SESSION

THE USE OF ROBOTS AND GAMIFICATION IN EDUCATION

Chair(s)

TBA

An Evaluation of Simulation in Lego Mindstorms Robot Programming Coursework

Frank Klassner[1,2] and Sandra Kearney[1,2]
[1]Department of Computing Sciences, Villanova University, Villanova, PA USA
[2]Center of Excellence in Enterprise Technology, Villanova University, Villanova, PA USA

Abstract – *Robotics programming has become a significant element of undergraduate computer science curricula over the past decade. This paper presents an initial examination of the effectiveness of simulators in helping undergraduates in computer science courses produce moderately debugged code on simulated robots before further refining it on physical robots (Mindstorms NXT). The examination included a study of how students evaluated their experience programming robots in Java with and without a simulator in novice through advanced level courses. The paper describes the simulator for the study, the robot projects studied by the students, and the survey instrument used in the examination. It concludes with results on gender differences in student evaluation of the robot simulators, and overall student assessment of the simulator's learning effectiveness.*

Keywords: Robot Simulation, Computer Science Education

1 Introduction

A rich long set of pedagogic research lines have shown that projects using physical robots such the Lego Mindstorms platform can be effective for grounding topics in computer science (CS) undergraduate courses such as CS0 [1, 2], CS 1/2 [3, 4], AI [5,6] and Operating Systems [7], and others outside CS. [8] The role of simulation in computer science courses has also been explored since the first work on Karel the Robot [9] down to the recent Alice [10] investigations. Several college and graduate-level robot simulation packages have also been created over the last two decades, including Player/Stage/Gazebo [11] and Microsoft Robotics Studio. [12]

In a study conducted in this area by the first author [13], over 200 computer science majors were surveyed on their perception of 25 Lego Mindstorms laboratory modules' teaching efficacy in a variety of computer science courses. One survey question in the study asked students across four types of CS courses whether a robot simulator would have made a significant difference in their programming experience. For those modules in which the environmental interaction of robot chassis design or control software design played a significant role in the lab, student surveys indicated that a simulation environment for the Mindstorms platform would likely have benefited them by (1) reducing their time spent verifying - and often rebuilding - their chassis and by (2) increasing the time available for improving the performance of their robots' control software.

Combining this observation with the research cited at the beginning of the section, it is natural to ask what role simulation can play <u>in conjunction</u> with physical robots in computer science education, particularly in cases where students might be distracted from programming issues by "messy" robot interactions with the real world. There is also the question of how to educate computer science majors about the industrial practice of using simulation to design and test robot systems before they are construed.

This paper reports on a research project that evaluates the usefulness of simulation in conjunction with physical Lego Mindstorms robotic programming exercises. To this end, it first presents the simulator that was designed for the project and then describes the Lego chassis and programming projects that were used in courses in the project. The paper then explains the survey by which students reported their experiences while doing the projects with and without the robot simulator. The paper concludes with an analysis of the survey reports and discussion of future work.

2 Robot Simulator

The robotic simulation package used in the experiments is called **RAMS** (Robot Analysis, Modeling, and Simulation). It was developed at Villanova to support various ready-made Lego Mindstorms chassis designs for students to test control software before trying the software on physical robots.

2.1 ROAMS – the Base for RAMS

Jet Propulsion Labs staff used their Dshell software platform for creating the ROAMS (**RO**ver **A**nalysis, **M**odeling and **S**imulation) simulator for modeling Mars robot rover missions. [14] ROAMS was designed to model a wide range of rover systems with an easy to use graphical user interface. ROAMS was designed to be a virtual rover that could be used to test rover systems beyond just using the hardware alone. These needs require that ROAMS support a detailed physics model of the rover platform along with the kinematics and dynamics incorporated in it. ROAMS includes a suite of sensors to get feedback information from wheels, steering motors, encoders, cameras, light sensors etc. ROAMS included models for the environment with which the rover will be interacting. The reliability of ROAMS is clearly demonstrated by NASA's Mars Rover Program's success.

ROAMS and Dshell have been released to the public research sector under the caveat that users must demonstrate evidence of US citizenship or permanent residence. The ROAMS platform was selected as the base from which to build the RAMS simulator used in this research project, since it included a powerful physics engine that could be adapted for undergraduate use in computer science curriculum. Although ROAMS presented a complex interface, we did not want to start with a less robust platform only to run the risk later of discovering that students' simulator experience was not close enough to "real-world physics." We felt it would be easier in our experiments to simplify ROAMS modeling or its interface when needed than to build up another simulator package's physics engine on our own.

2.2 RAMS Design

Java is the primary programming language in the Villanova computer science curriculum, and the Lego Mindstorms robot platform coupled with the Java-based LeJOS library [15] is the one that has been most commonly used in computer science courses at Villanova. To facilitate comparisons between student programming experiences with and without the simulator, the RAMS system was designed with a compiler that could retarget the same LeJOS code to a physical or to a simulated robot at the click of a button, without requiring students to rewrite any code when switching between physical and simulated platforms.

The RAMS interface developed as a modified interface for the ROAMS platform has three windows: an Eclipse editor window for editing and compiling lab module code, a simulator PlanView window for observing a simulated robot performing in real time under the control of student code, and a RAMS Client window for showing real time state changes in the sensors, LCD display and motors of the robots. RAMS is executed within a Fedora shell. Figures 1 and 2, respectively, show the PlanView and Client windows. The RAMS Client window as illustrated shows all the features that are possible to display, but it can be configured (by an instructor, for example) to show only those features of a robot that are germane to a given laboratory project.

Figure 1. RAMS PlanView Window

Figure 2. RAMS Client Window.

Students controlled the simulation by clicking on a start/stop button on the Client window. Data logs for each of the robot features could also be generated. The PlanView window was configured to track automatically the motion of the robot, but its perspective of the robot could be manually adjusted during a simulation. As shown in the PlanView window, environmental elements such as chairs, floors, and desks could be included in the simulation to aid in students' understanding of how to program their robots for the room in which they were working.

The research project was conducted using the RAMS with simulator models based on the NXT Mindstorms platform, but with minor modifications the RAMS simulator can use the LeJOS libraries for the current EV3 Mindstorms platform.

2.3 Robot Chassis Designs

To make it feasible for students to conduct the robot laboratory projects as closed lab experiences, the lab modules in this research were based on chassis that would be already constructed for the students. In this way students had the maximum in-class time available for programming and testing their control software. Two robot chassis designs were developed and modeled using the Spatial Operator Algebra modeling language of ROAMS. The simulator's underlying ROAMS architecture permits suitably-trained instructors to define and add new robot models to the list of available models.

Figure 3. Tribot

Figure 4. Spiderbot

The Tribot shown in Figure 3 is a three-wheeled robot that is based on the same-named model in the NXT Construction Manual that comes with Mindstorms NXT kits. Note that the sonar sensor in shown as the “head” of the robot was not used in the lab modules described later in the next section. This chassis was selected for use in the experiment due to its straightforward design. Students could use the LCD for debugging messages.

The spiderbot shown in Figure 4 is a six-legged robot that was designed based on the Strider robot presented in Valk’s NXT development book. [16] The robot uses three motors to drive its legs (one motor for each pair of legs). This chassis was selected for use in the experiment due to its complex appearance. When properly coordinated, the legs move the robot in a sliding/crawling manner in the direction faced by the sonar-sensor “head” of the robot.

3 Simulator Evaluation Regimen

3.1 Courses Surveyed

Four computer science courses served as the testing environment for Lego Mindstorm Robot Programming simulation research. These were:

CSC2400 Evolution, Learning; Computational & Robotic Agents (students do not require computing knowledge)
The course explores how software designers and artificial intelligence researchers draw inspiration from life forms and how they learn and evolve to design programs and robotic agents that learn and adapt to changes in their environment.

CSC 1051 Algorithms and Data Structures I (students should have some familiarity)
The course explores problem analysis, algorithm design, and the implementation of algorithms in an object-oriented programming language. Extensive programming is required using the Java programming language

CSC 2053 Platform Computing (Students should have advanced computing knowledge)
The course includes identifying platform facilities and constraints, event driven programming, MVC pattern, client/server considerations, security-performance-accessibility issues, web/mobile programming, and application programmer interfaces.

CSC 5930 Game Development (Students should have advanced computing knowledge)
The course involves a large-scale, software design and development project, and several small-team game design projects. Students explore many facets of game design such as project management, software design, marketing analysis, game design tools and software.

The courses required a broad range of knowledge and experience, from students with no programming language knowledge or experience to students with more than five computer programming languages and in-depth understanding of computer science. The students in these computer science courses had three basic levels of programming experience: Novice, Intermediate, and Advanced.

3.2 Robot Projects

Two laboratory projects were developed for students to use. In the first, Tracing Geometric Shapes, students were asked to develop Java code to make a Tribot follow a path that traced out squares and other convex shapes with uniform sides on a floor, culminating in producing a square figure-8 pattern on the floor. They were asked to investigate how loops and conditional statements could be used to simplify their code, and to understand how they need to modify the angles in which they turn their robots at the vertices of shapes for shapes with more and more sides.

In the second laboratory module, Locomotion with Genetic Algorithms, students were asked to run a genetic algorithm (GA) on the robot to help it learn control parameters (speed, direction, duration of motion) for its motors so that it could move forward in a coordinated manner. Depending on the experience-level of the course in which this module was conducted, students sometimes only experimented with GA parameters such as mutation rate, crossover policy, or population size to see how each influenced the convergence of a supplied GA program. When students had more programming experience, they also were permitted to make code changes to the supplied codebase beyond parameter-tuning.

3.3 Survey Design

The survey was designed as a questionnaire. It was divided into four parts: General demographics, specific questions related to the lab, open comments in five related areas and specifics about students major study area and academic performance.
The questions were both structured and unstructured, providing for quantitative and qualitative analysis of the answers. The structured questions were limited to a single response providing researchers the opportunity to evaluate the strength of agreement or disagreement within the 11 learning components, represented by the related statement. The format of the structured questions was designed to outline the expectations of the learning objectives, the background of the robotics platform, module completion, satisfaction and promotion of this type of learning program.

4 Results

The research spanned two years, and four academic semesters, from the spring of 2013 to the fall of 2015. Research surveys were completed by 144 students, composed of nine different groups. There were 79 respondents that identified as male and 52 respondents identified as female. The remaining students did not report their gender identity.

The student population was primarily sophomore and juniors with the population represented specifically by 18 freshmen, 57 sophomores, 28 juniors, 24 seniors, and one post graduate. This information is displayed in Figure 5. All other students did not report their current grade level. The ethnic groups consisted of 85 Caucasian, non-Hispanic; 18 Asian and Pacific Islander; 4 Black, Hispanic; 2 Caucasian, Hispanic, with the remaining respondents not reporting ethnicity. The research participants were from two major academic groups, computer science and liberal arts.

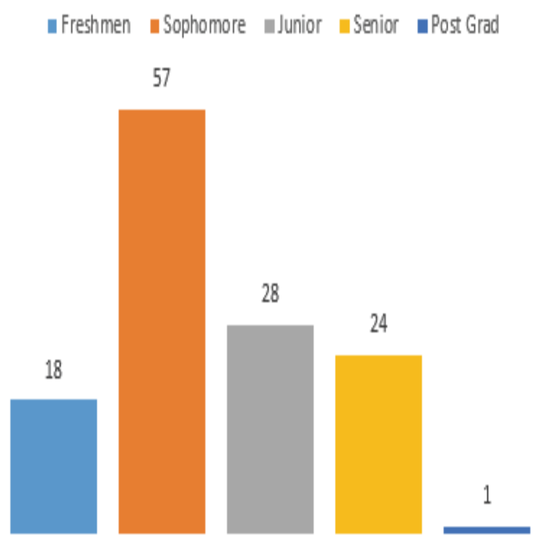


Figure 5. Self-Reported Academic Level

The unstructured questions of the survey covered five basic context areas:

- Interesting aspects
- Greatest difficulties
- Proposed changes to the modules
- Positive and negative aspects
- Challenges or benefits.

The structured questions were centered on 11 statements, designed to capture the spirit and context of learning through the use of either a robot simulation or no simulation. The classification was divided across agreement or disagreement of the statements as it related to the learning opportunity. For example, Question 1 asked students to rate their agreement with the statement, "After reading the learning objectives, it is clear to me what I should learn from this module," while question Q9 asked students to rate their agreement with the statement "I recommend that this module be used in future offerings of this course." There were four categories: gender, computer programming experience, class level, and academic major.

Figure 6 provides a summary of the survey data from the structured questions. It indicates that there was a high degree of neutrality expressed about the use of simulation, across all statements other than Question 1.

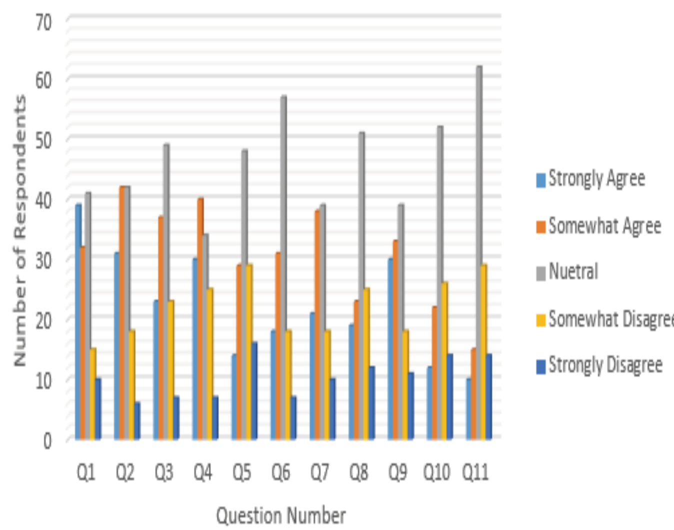


Figure 6. Total Responses to Structured Statements

The results of the Male-Female difference were of interest and warrant additional research. [17] Females were less likely to strongly agree or disagree with any of the survey statements, unlike the male respondents. Females showed a neutral to slightly positive response. However, the overall data responses from males reflected neutral to strong positive on the learning experience. With the combined results there is a strongly neutral position on the simulation. It is worthy to note, the most advanced academic group of participants had no females in the class or sampling.

The data in Figure 7 reflects the findings of Male-Female differences research. Questions in each subgroup are listed in the same order left-to-right (Q1 to Q11). [17]

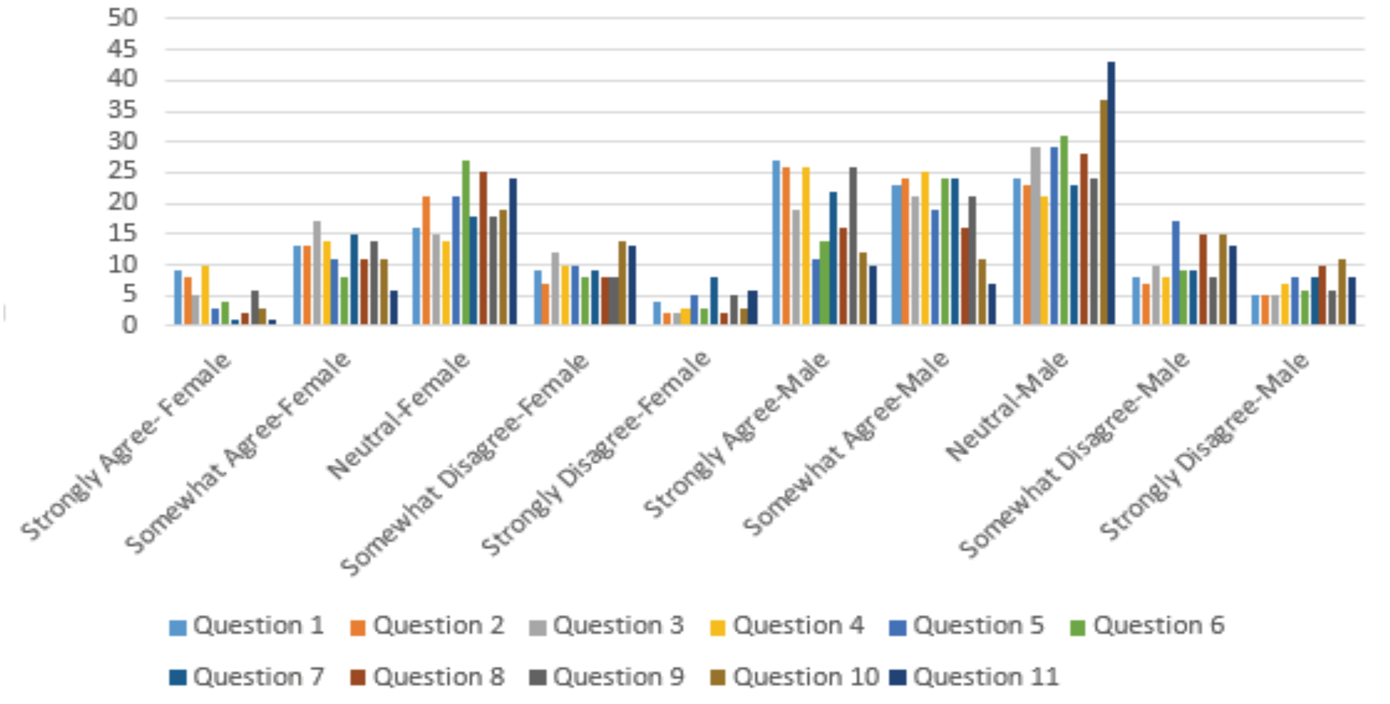


Figure 7. Response by Gender

Additional detailed analysis depicts the breakdown of academic majors, class level, and degree of programming experience. The small number of advanced participants did not provide enough data for adequate evaluation based on the level of experience. There were however, distinct differences between the Novice and Intermediate participants. Novice participants reflected neutral to more positive than even the small number of advanced students. Intermediate students reflected neutral to strongly positive.

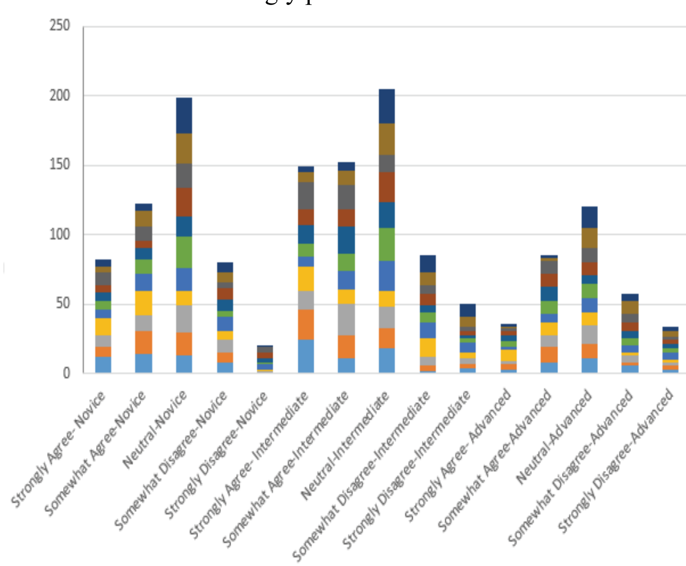


Figure 8. Response by Level of Experience

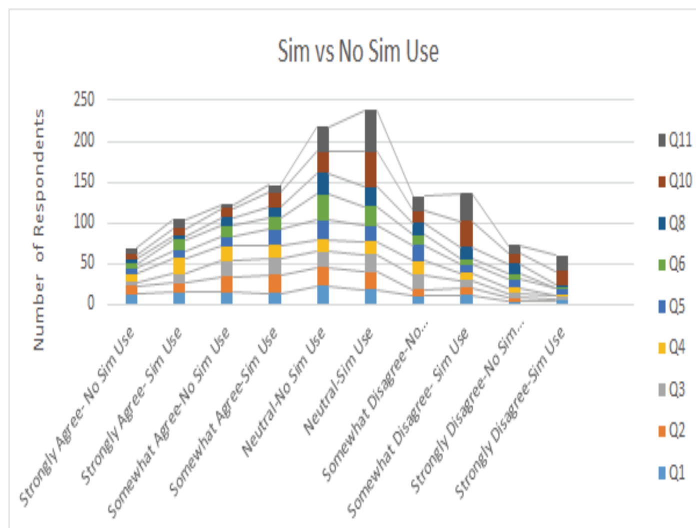


Figure 9. Response by Use/No-Use of Simulator

The initial hypothesis was that novice users would be much less likely to learn with the use of simulation due to the abstract aspects. That is, students in novice-level courses might tend to see the simulation as extra work rather than as a stage to help them produce debugged code. The final two questions of the survey (Q10 and Q11) asked students to determine if simulation improved or caused no change in their experience. The results show a slightly more positive agreement, but overall students were neutral about use or non use of the simulator. This contributed to an inconclusive hypothesis. The results are better defined by the qualitative answers given by students in the unstructured portion of the survey.

The qualitative data gave evidence to possible reasons for the neutrality. The responses to questions reflected frustration with glitches in the software and the extensive amount of time for lab completion.[18] Students were satisfied with utilizing robots in their learning program, they simply wanted better composability, specifically semantic composability that provided a smooth learning experience. As with research completed in the combined simulation and software engineering of complex systems, the evidence of the study reveals the need for additional research with regard to composability and learning simulations. Learning robotics can be a complex activity, when broken down into components and combined with good composability it may be shown in future research just how closely learning and building learning simulations relates to the development of complex computer systems.[19]

5 Future Work

Future research should build on the baseline activity in the use of simulators in education, specifically computer science and robotics. Additional research activities might include a gender study and a look at gender neutral simulations. In addition, incorporation of composability and component based modeling will provide previously uncharted results to improve simulation in teaching and learning.

6 Acknowledgments

The authors are very grateful to the team of student programmers who worked on adding RAMS features to the ROAMS codebase: (in chronological order) Alex Palkovic, Taylor Clifton, Tyler Young, Brian Kim, and Andrew Grace. They also thank Dr. Abhinandan Jain of Jet Propulsion Labs for his advising of the students on working with the ROAMS software as well as for his work in developing a model for the Spiderbot. This research was supported by NSF DUE TUES Grant #1023006. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

7 References

[1] J. Anderson and J. Baltes. "Robotics and AI as a Motivator for the Attraction and Retention of Computer Science Undergraduates in Canada," Proceedings, AAAI 2008 Spring Symposium *Using AI to Motivate Greater Participation in CS*, March, 2008, pp. 2-7, AAAI Press.

[2] S. Tejada. "AI, AIBO, and Art: Inspiring Interaction with Computer Science," Proceedings, AAAI 2008 Spring Symposium *Using AI to Motivate Greater Participation in CS*, March, 2008, pp. 92-97, AAAI Press.

[3] D. Blank, "Robots Make Computer Science Personal," *Communications of the ACM,* 49(12), pp. 25-27, 2006. ACM Press.

[4] J. Cummins, M. Q. Azhar, and E. Sklar "Using Surveyor SRV-1 Robots to Motivate CS 1 Students," Proceedings, AAAI 2008 AI Education Colloquium, pp. 23-27, AAAI Press. July, 2008.

[5] B. Tribelhorn and Z. Dodds. "Roomba and Mac OS X: Cross-platform Vision and Robotics for AI." Technical Report from the 2006 AAAI Robot Workshop. January, 2007, AAAI Press.

[6] F. Klassner, "Launching into AI's 'October Sky' with Robotics and Lisp." AI Magazine, vol. 27, no. 1, Spring 2006. AAAI Press, p. 51-65.

[7] F. Klassner and C. Continanza, *Mindstorms Without Robotics: An Alternative for Systems Courses.* F. Klassner & C. Continanza, Proceedings of SIGCSE 2007, Covington, KY. March 3-6, 2007.

[8] S. P. Imberman and S. Zelikovitz. "Robotics as a Component of a General Education Course," Proceedings, AAAI 2008 Spring Symposium *Using AI to Motivate Greater Participation in CS*, March, 2008, pp. 50-54, AAAI Press.

[9] R. Pattis, Karel the Robot, 2nd Edition, John Wiley and Sons, 1995

[10] The Alice Site: http://www.alice.org (accessed 4/11/2016).

[11] Player, Stage, and Gazebo, USC Robotics Research Lab website. http://robotics.usc.edu/?l=Projects:PlayerStageGazebo (accessed 4/11/2016)

[12] Microsoft Robotics Developer Studio. https://www.microsoft.com/en-us/download/details.aspx?id=29081 (accessed 4/11/2016)

[13] F. Klassner, *MindStorms as an Electronic Tangible Across the Computing Curriculum.* Journal of Computing Sciences in Colleges, Consortium for Computing Sciences in Colleges, 2012; 27 (3): 52-44.

[14] A. Jain, *"Structure-Based Computational Modeling Architecture for Robotics"*, 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, May, 2013.

[15] LeJOS NXJ website: http://www.lejos.org/nxj.php (accessed 4/11/2016)

[16] L. Valk, "The Lego Mindstorms NXT 2.0 Discovery Book: A Beginner's Guide to Building and Programming Robots," No Starch Press, 2010.

[17] Martell, Richard F.; Lane, David M.; Emrich, Cynthia "*Male-female differences: A computer simulation.*" American Psychologist, Vol 51(2), Feb 1996, 157-158.http://dx.doi.org/10.1037/0003-066X.51.2.157

[18] Eric Winsberg. 2010. "Science in the Age of Computer Simulation." University of Chicago Press, Chicago, IL, USA.

[19] Bartholet Robert G.; Brogan David C. Reynolds, Paul F.; Carnahan Jr. Joseph C. "*In Search of the Philosopher's Stone: Simulation Composability Versus Component-Based Software Design,*" Proceedings of the 2004 Fall Simulation Interoperability Workshop, Orlando, FL, September 2004.

Games Programming in Computer Science Education

D. Vrajitoru[1], **P. Toprac**[2]
[1]Department of Computer and Information Sciences, Indiana University South Bend, IN, USA
[2]Department of Computer Science, University of Texas at Austin, TX, USA

Abstract – *In this paper, we investigated the role of content and processes as they relate to games and programming in post-secondary computer science courses. Our examination found themes that can help computer science instructors teach both introductory and advanced programming topics. Our analysis was focused primarily on the theoretical and practical subject matter that is taught in game programming classes and, secondarily, in how game-based programming projects and assignments can help students learn core computer science concepts and improve coding skills.*

Keywords: game programming, curriculum design

1 Introduction

Game programming has been a popular topic in computer science education for many years. The popularity of game programming with computer science students is due to two causes. First, programmers and potential computer science majors are often passionate about playing games and participating in the game culture and related media, including literature, film, conventions, live action role-playing, and so on. Thus, they are driven by a desire to create games and be recognized within the gaming community. Second, the game development industry provides significant job opportunities, which also fuels students' interest in the field. Job opportunities are present for those who are interested in working for established, relatively stable, game companies, as well as for those who would rather work as independent freelancers or for small entrepreneurial startups.

Although game programming courses are often offered as electives for computer science majors, there are a large number of universities that offer entire degrees or programs in game programming. Such programs include the Bachelor's of Science in Computer Game Development from DePaul University in Chicago, IL, the Certificate of Computer Science in Game Development from the University of Texas at Austin, and the Bachelor's of Science in Game Design & Development from the Rochester Institute of Technology, NY. Moreover, computer science departments offer game design and development courses and programs in collaboration with departments or schools of fine arts, and even English departments.

Several studies have examined the curriculum of game programming courses based on their syllabi. The typical purpose is to help instructors of these courses design their assigned classes more effectively. For instance, [1] examines what topics are appropriate for game courses, what would be the best choice of language/API considering the class level, and various ways to find useful resources. In [2], the authors survey a range of game courses and describe how digital games can support the learning process in computer science education. In [8], the authors discuss how content related to computer hardware can be learned by means of a digital game. The study presented in [11] illustrates the use of game development for teaching problem solving techniques. Our paper examines in more detail some of the aspects mentioned in [2], particularly those regarding game courses in computer science at different levels of student expertise.

In [4], the authors discuss how active learning as a tool to teach critical thinking and problem solving is attractive to computer science students. The paper also touches on the issue of retention of computer science majors and how active learning methods can improve it. Our paper also discusses the course CSCI-B100 Tools for Computing mentioned in [4].

In the papers [3] and [10], the authors examine the interdisciplinary nature of some of the upper-level game development classes, which we will also discuss in Section 4 of this paper. As part of the interdisciplinary nature of game development courses, managing teams is discussed in [7].

Studies can also be found on using games or other entertaining topics in introductory computer science classes, such as Gothic novels [5], role-playing games [6], and gamification techniques in [7] and [9]. Our paper also discusses the use of games in early programming classes.

This paper is organized as follows. Section 2 examines introductory programming classes, as well as core course following them. Section 3 discusses middle level courses with specialized content in game development, while Section 4 takes a look at upper level courses. We finish the paper with some conclusions.

2 Early Programming Classes

Let us start by examining programming classes that take place early in students' computer science track, such as the CSCI-B100 Tools for Computing, taught at Indiana University South Bend (IUSB). Typically these are the first or second programming classes taken by students.

The goal of such classes is to introduce both major and non-major students to programming and possibly attract undecided students to computer science. These classes typically employ simple programming languages, such as Visual Basic or Processing, because they are easier to learn without prior experience in coding. The advantage of these programming languages is that they come with visual tools that can teach basic programing concepts without requiring students to learn the strict syntax of coding using text. Using visual programming tools lessen cognitive load and can improve motivation and learning over coding using text. Another technique that is often employed to improve motivation and learning is using project-based learning. That is, students must complete practical mini-projects throughout the semester in a lab environment under instructor supervision.

Game development projects can also be used in introductory courses to improve learning and motivation, because students can quickly see that basic data structures and programming concepts are necessary in these projects. The game development process affords immediate visual feedback of progress during programming. In addition, games provide an element of fun and entertainment, making it easier to engage students in the task at hand and making them more invested in the outcome of their programming assignments.

For example, a game of Pong [12] can teach beginner programmers about variables – notably the two paddles and the ball. Pong presents a problem requiring the solution to an iterative process where the ball moves in the stage area with a given speed and direction. In order to solve the problem, students are introduced to the idea of conditionals by checking for collisions of the ball with the stage boundaries or walls and performing a bouncing operation when this occurs.

More advanced concepts, such as functions, can be introduced through many different games. For example, in a game of Tic-Tac-Toe, a function can be written to verify the winning condition. The concept of simple arrays and string manipulation can be taught through a game of Hangman. Here, a string is needed to hold the word to be guessed. A Boolean array is used in the background to mark the letters in the word that were guessed correctly so far.

More advanced core courses can also benefit from using games as programming projects. Games provide practical applications for data structures and object-oriented programming, and are more fun and appealing to the students. For example, game assignments can be used in courses such as CSCI-C243 Data Structures taught at IUSB, which serves as a prerequisite for and gateway to most of the upper-division core and elective courses. This class teaches major data structures, including tables, trees, and graphs, and algorithms. It is important for students in this class to not only learn and understand data structures but also grasp their pervasive importance in all areas of computer science. The latter can be achieved both by providing lists of applications for each data structure, and by assigning practical programming tasks that make use of the abstract data structures that are introduced.

For instance, two-dimensional arrays serving as tables can be explained through arcade-style games such as PacMan [13] or Sokoban [14]. In these games, the world is two dimensional with discrete coordinates and can be represented as a 2D array or matrix of integers. An even better approach would be to define an enumerated type for the stage components, containing constants for walls, spaces, food, and other game elements. Then, the matrix representing the game area would use it as the base type for the cells.

A game of Breakout [15] can teach the concepts of boundary checking in data structures, as well as navigation. As a ball bounces against the walls, conditions of being within the boundaries of the arena must be checked every time. In the classic version of the game, the player destroys bricks by bouncing a ball against them, which is represented as a 2D array. Breakout also teaches students the complex relationships between integer coordinates in the table and real coordinates in the arena, in order to display the bricks properly. This leads to the concept of type conversion from a higher precision data type to a lower precision one.

More complex data structures, such as stacks and queues, can be somewhat abstract and difficult to grasp for the students. Card games are ideal applications for making abstract concepts more concrete. For example, a shuffled deck can be stored in a stack or a queue, depending on its purpose in the game. In the classic version of Klondike Solitaire, the game starts with eight “tableau” piles of cards facing down. Only the top cards can be turned face up and played from each of the piles. Other cards can be placed on top of the piles in descending order and in alternating color. Thus, tableaus can be represented as stacks. To win the game, all the cards must be moved to four foundation piles by suit and in ascending order. Again, cards can only be added to the foundations on top, which also means that they are best represented as stacks. Finally, the deck of playable cards can be recycled, so that when the cards are exhausted, playing starts over. To accomplish this, either a pair of stacks or a queue can be used, the latter being better suited for the purpose of the class.

Binary trees are also challenging for students to learn, and, again, games can make this subject less abstract. For example, one can show, in a game of Tic-Tack-Toe, how the current state of the game can be expanded into a tree where each branch is a possible move leading to another state. This can lead to explaining how artificial intelligent (AI) agents could use this tree to play the game. Even though AI is probably too early for students to implement at the time, the discussion helps to effectively introduce the concept, which can be built upon latter in the curriculum.

Object Oriented Programming (OOP) is often taught in the second or third semester of programming courses within the computer science curriculum. Developing games can help improve the students' comprehension and retention of OOP. Game projects showcase well-structured hierarchies of classes, some of them with storage purposes, such as stacks and queues, and others with more operational purposes, such as an interface class. A game project typically contains a Game Master or Manager class that oversees all the game objects, rules, and goals. This helps students understand not only the individual purpose of each class, but also the need for an overall structure. Incidentally, the term Game Master is also used when playing role-playing games. So, students who play these types of games may intuitively understand the topic better and also feel increased engagement with the subject.

3 Medium Level Classes

In this section, we will examine courses that are taken in students' sophomore or junior years. These are typically elective courses in a general computer science undergraduate degree program, or foundation or core courses in a degree program specialized in game development. Examples of such courses include the CSCI-C490 Games Programming and Design taught at IUSB and GAM 374 Fundamentals of Game Programming I taught at DePaul University, Chicago, IL.

Students in these classes have already completed many core courses, as well as probably rigorous algorithms and data structures courses. Although instructors typically find these students to be capable programmers, students still need to practice and extend their coding skills. At the same time, these courses train and hone students' programming skills and give them a solid game development experience.

These game programming electives usually teach students fundamental concepts in game development, relevant algorithms, design topics, and game-specific program structures. Examples of typical game algorithms include fair shuffling methods for card games, efficient randomized content generation, and kinematics-based object movement. Collision detection is another important topic that requires a significant amount of time during the semester to cover. Another focus is the various game loops for different types of games.

Students better appreciate the importance of algorithms when applied in concrete applications. For instance, Prim's algorithm is at the core of procedurally generated content, which allows the instructor to create a connection between game development and graph theory. Other graph algorithms have applications to games, notably path-finding algorithms for maze-like structures.

In terms of design, these courses emphasize strong program structure and object-oriented design. They also contain many topics related to game design, such as the mechanics, dynamics, and aesthetics analysis of games. In the C490 class at IUSB, students typically must write a midterm paper examining a game of their choosing from several points of view, such as genre, theme, rules and mechanics, reasons for failure or success, as well as the social context of the game. By performing class presentations of their papers, students train their communication skills while either sharing their enthusiasm for their favorite game or warning peers of bad decisions made by some game creators.

These higher-level courses are very likely to be intensive in terms of their use of libraries or APIs. These may not be entirely new concepts for the students, but they have not worked with them extensively before. A large part of the instruction may be dedicated to getting students more comfortable with using specialized game libraries or engines. For example, some classes may be taught in C++ or Java with direct use of OpenGL. As an alternative, an API such as Flash / ActionScript may be used, or even a game engine such as Unity or Unreal. The challenge in this case is to understand the tools provided by the game engine and how best to make use of them in the development process. Although it takes extra effort to learn how to use theses libraries, the gain is in being able to focus on higher-level game aspects by letting the libraries take care of lower level tasks.

An important aspect of the mid-level classes is that they are programming-intensive. As mentioned previously, although students have a good programming background, they still need to develop and improve their coding skills by working on real-world projects. In lower level classes students see programming as a goal in itself, the task to be accomplished to earn a good grade. Mid-level classes allow students to move towards seeing programming as a tool to accomplish something specific. Students are not focused anymore on just the algorithms for their own sake, but are solving a complex problem that requires solving many other smaller ones.

Furthermore, students get to experience how much of a challenge creating a good game can be and how important debugging is for the development process. Since many games are intended to be played over and over, any coding or design flaws are more likely to be revealed over time than in other programs that students may have written previously. Thus, students learn to appreciate the need for good programming practices and for extensive testing techniques. Students can also better understand their instructor's insistence on efforts to improve program performance. Generally, correctness of the program is a major goal for any coding activity, but efficiency may seem to beginner programmers as a lower priority. When working on games, students can see first-hand how an efficient algorithm performs better for their purpose than a weaker one. Good program structure and programming practices are thus seamlessly integrated into their learning objectives.

4 Advanced Level Classes

In this section, we examine advanced courses taken by students who are seniors, but may be juniors. An example of such a course is CS 354T Game Development Capstone: 3D Games taught at the University of Texas at Austin.

The focus of these courses is game development in terms of interdisciplinary teamwork, project management, and relatively complex software system development of significant size. The teams can be between 3 and 7 students, with bigger teams providing a working experience closer to that of the game industry. Oftentimes, these courses are open to students from various majors: computer science, fine arts, music, and others. The team composition reflects the variety of specialties and interests that creates a balance of the skills necessary to produce a game that approximates those in the real world.

While in the early game programming courses the instructor typically emphasize programming skills and OOP program organization, the advanced courses are used as a platform for training software engineering skills. The teams typically work on a single project for the entire semester, which requires the students to manage their time and make critical decisions during the development process. Team communication is crucial and specialized tools, such as TeamSpeak and Slack, must be learned and used throughout the semester. Source code control and sharing software, such as GitHub, TortoiseSVN, or Perforce, are necessary even for small teams. Students may apply general software engineering concepts that they learned in another class and, in addition, learn some that are specific to game development.

In earlier classes, students' efforts are concentrated on making their games work. In later classes, their assignments begin with the design process. Students must start by making design decisions about the game they want to make, such as the genre, theme, and scope of content creation. They must choose, adjust, and justify all the game aspects, from basic mechanics to storyline. Students must design an interface and choose from different ways of delivering narrative text or tutorial help. They learn to pitch their game to an audience and change plans based on feedback. Moreover, students must balance the time they have with the skills they share collectively to decide how much they can accomplish in one semester.

Along the way, students learn a considerable amount of content related to the entire game development process from concept to delivery. If the class is using the agile development methodology, as most do, students must deliver and present iterations of the game on a regular basis, as well as important milestones for the game, including the first playable version, alpha, beta, and final release by the end of the semester. Students are confronted with feedback about their games when their games are being playtested by participants with different levels of player expertise and different preferences, from the team members themselves to classmates to possibly complete strangers. The instructor must also rely on peer feedback from team members about themselves, each other, and the project in order to assign grades. This feedback must reflect not only the overall achievement of the project, but also the contribution of each team member to the effort.

These kinds of courses help students transition from solo programmers to team members and even project managers. Students learn collaborative and leadership skills that they will need later in the workplace.

5 Conclusions

In this paper we examined various ways in which game development content can help achieve learning objectives in a computer science program as students progress through the curriculum. We started with beginner courses where writing small games can make learning to code more fun and attractive. In more advanced early core classes, game programming can be used as concrete examples for many of the algorithms or data structures that students must learn. Medium level courses are where students learn fundamental concepts related to games while improving their programming skills through challenging practical exercises. Finally, advanced game development classes teach collaboration, project management, and software engineering skills. Thus, game programming can improve computer science education at all levels with content that is attractive and motivating for students.

6 References

[1] Ian Parberry, "Challenges and Opportunities in the Design of Game Programming Classes for a Traditional Computer Science Curriculum", *Journal of Game Design and Development Education*, Vol. 1, pp. 1-17, 2011.

[2] U. Wolz, T. Barnes, I. Parberry, M. Wick, "Digital Gaming as a Vehicle for Learning" In *Proceedings of the 2006 ACM Technical Symposium on Computer Science Education*, pp. 394-395, Houston, TX, Mar. 1-5, 2006..

[3] H. Bourdreaux, J. Etheridge, and A. Kumar, "Evolving Interdisciplinary Collaborative Groups in a Game Development Course"; *Journal of Game Design and Development Education*, Vol. 1, pp. 25-37, 2011.

[4] H. Hakimzadeh, R. Adaikkalavan, and J. Wolfer, "CS0: A Project Based, Active Learning Course", *International Transaction Journals of Engineering Management and Applied Sciences and Technologies*, http://www.tuengr.com/Vol 2(5), 2011.

[5] H. Bort, D. Brylow, M. University; and M. Czarnik, "Introducing Computing Concepts to Non-Majors: A Case

Study in Gothic Novels", *Proceedings of the Special Interest Group in Computer Science Education Conference (SIGCSE 2015)*, Kansas City, 2015.

[6] D. Toth and M. Kayler, "Integrating Role-Playing Games into Computer Science Courses as a Pedagogical Tool", *Proceedings of the Special Interest Group in Computer Science Education Conference (SIGCSE 2015)*, Kansas City, 2015.

[7] C. Latulipe, N. B. Long, C. E. Seminario, "Structuring Flipped Classes with Lightweight Teams and Gamification", *Proceedings of the Special Interest Group in Computer Science Education Conference (SIGCSE 2015)*, Kansas City, 2015.

[8] M. Sanchez-Elez and S. Roman, "Learning Hardware Design by Implementing Student's Video-Game on a FPGA", *Frontiers in Education: Computer Science and Computer Engineering*, Las Vegas, NE, July 27-30, 2015.

[9] M. Mejias, K. Jean-Pierre, Q. Knox, E. Ricks, L. Burge, III, and A. N. Washington, "Meaningful Gamification of Computer Science Departments: Considerations and Challenges", *Frontiers in Education: Computer Science and Computer Engineering*, Las Vegas, NE, July 27-30, 2015.

[10] D. Rosca, J. Kostiou, and C. Locke, "Video Games to Motivate a Multi-disciplinary Capstone Experience", *Frontiers in Education: Computer Science and Computer Engineering*, Las Vegas, NE, July 27-30, 2015.

[11] E. Bachu and M. Bernard, "Visualizing Problem Solving in a Strategy Game for Teaching Programming", *Frontiers in Education: Computer Science and Computer Engineering*, Las Vegas, NE, July 21-24, 2014.

[12] Pong, Atari, http://www.ponggame.org/.

[13] PacMan, Namco Bandai Games, http://www.bandainamcoent.com/game/pac-man.html.

[14] Sokoban, Thinking Rabbit, http://sokoban.info/.

[15] Breakout, Atari, www.atari.com/arcade.

Innovations in Education using Gamification: A Case on Vietnamese Tertiary Institutions

Prof Avinash Shankaranarayanan[1], **Prof Christine Amaldas**[2]
[1]Royal Melbourne Institute of Technology (RMIT) International University
[2]Ritsumeikan University, Japan

Abstract - *Recently, the concept of Gamification has been widely applied to teaching environments. Motivation in students is a common variable utilized for measuring student performance and feedback within and outside of a classroom environment. In this paper, the challenge is to look at alternate teaching and learning methodologies in improving the motivational levels of students coming from non-IT background towards learning ICT technologies. The research focuses on two sets of student populations: test and control group deployed using a pre and post-gamification phase introduced at a Vietnamese Institution. A participatory model of Action Research (AR) was applied as the conceptual framework observing pre and post-motivational levels of students in understanding the existing motivation of students in a selected classroom using feedback instruments where the use of game components were non-existent. An online survey was also conducted obtaining regular feedback from students in both phases at the time of content delivery.*

Keywords: Action Research, Content Delivery, Feedback, Gamification and Motivation in Student Populations.

1 Introduction to Gamification

"In most online communities, 90% of users are lurkers who never contribute, 9% of users contribute a little, and 1% of users account for almost all the action [13]".

Gamification could be defined as 'the art of applying motivational or game like dynamics to retain consumers within a socially interactive domain space' as stated by [1]. The design of gamified environments could be based on traditional business models applying game like dynamics to existing traditional markets or virtual electronic markets within a specified time frame which is controlled by the gamifier or schematic designer. Zichermann and Cunningham [19] defines Gamification as the process of game-thinking and game mechanics to engage users and solve problems. Typical examples of applying gamification in businesses includes: Airline companies using frequent flyer points; hotels and food joints utilizing loyalty programs; grocery stores exercising the concept of green stamps and vouchers. These were some of the typical Gamification mechanisms previously used in the fields of marketing [10,11,16]. Gamification may be a new term in this era but the idea of using game-thinking and game like mechanics to solve motivation problems and engaging audiences (students in our case) is not exactly new. The military has been using games and simulations for hundreds (if not thousands) of years, and the U.S. and Australian military has been a pioneer in the usage of video games across geographical locations. Simulation systems are now being primarily used in Australia defense to train top pilots of very young ages using flight Sims. Hollywood has been hot on the adoption of gamification for years with movies like War Games, Call of Duty, etc. [10,11,16]. This is probably one of the reasons why people spend so much time on games which was queried as a research question by numerous researchers in trying to answer this inquiry.

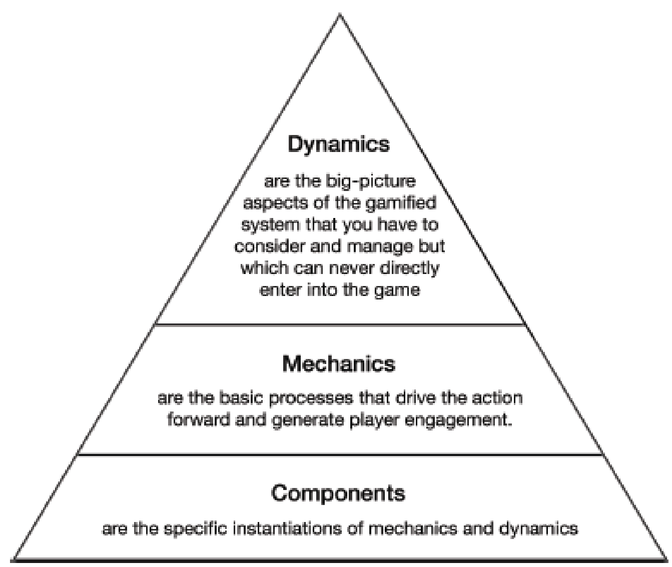


Figure 1: The game component hierarchy [4]

Jane McGonigal [10] tries to apply this query by labeling gamers as super-empowered hopeful individuals who exhibit four distinctive behavioral traits [5,6,7] as shown in Figure 1.

- **Urgent Optimism** – the desire to act and the belief in achieving success.
- **Social Fabric** – the ability to trust and form stronger social bonds through game playing.

ISBN: 1-60132-435-9, CSREA Press ©

- **Blissful Productivity** – the belief that the task they are engaging with is meaningful, hence, the dedication towards the game task itself.
- **Epic Meaning** – the strong attachment to a meaningful and awe-inspiring story that they are personally involved in and striving to make their mark on it [7].

Likewise, Priebatsch and Chatfield [3,15] have presented seven design concepts of computer games that help gamers immerse themselves into the world that the game presents [7,10]. Taking the example of a primary or kindergarten school children, teachers use quizzes or building blocks or playing scrabble in groups as a way of learning and then, rewarding the kids with chocolates or sweets is a simplistic example of Gamification applied to teaching and learning. Another term closely related to Gamification would be edutainment or the art of learning through media and entertainment. The major difference is the team factor or interactions that take place through team based motivation and learning.

2 Theoretical Objectives

2.1 Traditional Teaching Approaches

Although there are numerous approaches to teaching, there is no “one-size fits all” solution to teaching and learning as this is dependent on the concepts, methodology and most crucially the learners that can be observed in Table 1 indicating the differences between traditional teaching approaches to innovative teaching approaches empowered by technology.

By default, we find that the list of teaching methods categorized below closely matches to Student Centric versus Teacher Centric learning which makes or breaks students into Surface learners and Deep learners:

Surface Learning: Students were only focussing on core topics relevant to passing a course rather than gaining the knowledge for life-long learning, e.g. reading the PowerPoint slides, hand-outs, recommended textbooks.

Deep Learning: Students were required to read widely and explore the web-links available from e-Textbook resources to aid them in their research during the 1st and 2nd year of their university life. Coaching the students in the library, strengthens the student-teacher bond and enhances enthusiasm of the students to learn more. Coaching and facilitating student learning and overall comprehension of the course materials were given central importance towards peer learning where other materials included watching videos related to the course (blended activities) involving active Blackboard participation and discussions.

This enables the students to become better researchers kindling their life long learning experience.

Table 1: Traditional versus Innovative Teaching Practices

Traditional	Current
Asynchronous; Problem-based Learning; Peer Learning; Structured Learning; Paper based Assignments; Group Discussions; and Others.	Synchronous; Blended / Hybrid Learning; Flipped Classroom; Learning Management System (LMS) / Content Management System (CMS); Virtual Classrooms, Webinars, Podcast, Blogs, Social Media, Mobile devices and Computer Based Training (CBT) and e-Examinations; Using Lego bricks (see Figure 2); Virtual Reality (VR) Based Learning; Augumented Reality (AR) Based Learning (see Figure 3); Motion Sensory Control, Gyroscope (Android, Kinetics XBOX One, Nintendo Wii), e-SmartBoards; Gamification; and Others.

Gamification is widely applied to classroom environments and other areas wherein games or gaming mechanisms have an impact and influence on our day to day mundane lives. Other areas includes: sojourning or vacational trips; learning new language(s); managing our everyday exercises like jogging or walking; and even building relationships overseas using our mobile phones or the internet [11]. What we had once taken lightly as “gaming or having fun” has quickly become the way we interact in the future with relevance to work and everyday life. So, can children/students learn from games? The answer to this question is?

Absolutely! Research by Dr Arne May at Germany’s University of Regensburg clearly indicates that learning a new task produces a demonstrable increase in the brain’s gray matter in mere matter of weeks. And brain scientists all over the world agree that games (challenge-achievement-reward loop), promotes the production of dopamine in the brain, reinforcing our desire to play [16].

Gamification enables us to help align our interests with the intrinsic motivations of our students, amplified with the mechanics and rewards that make them feel welcome, bringing their friends, and keep returning to the gamified environment (retaining phase).

Students in various courses may not be in a position to answer concepts without the lecturer’s help in a variety of subject settings. This may result in queries raised to the teacher repetitively with the same questions followed by the need for the teacher to answer similar questions individually. Enquiring, deliberating and sharing knowledge are key parts of higher education wherein the challenge is to develop a model of improving the engagement of students at a tertiary institute such as a university.

Figure 2: The Lego Workshop

Students should gain all-encompassing knowledge in their chosen disciplines; fulfilling the necessary requirements of the required learning outcomes. Most students are unable to grasp every part of each discourse topic by themselves, resulting in numerous questions. In order to overcome this pertinent challenge, game design features and methods are used to augment student's motivation to engage more effectively within and outside the classroom actively. This technique is called Gamification, "the use of game design elements in non-game contexts" [5,6,7].

3 Methodology

Since its inception, Gamification is now an established and widely used technique to enhance the user's experience through edutainment and goal based conceptualization. We shall now try and address the following research question formulated based on the proposed methods:

> *"Can Gamification be applied as a tool for increasing the motivational levels of Non-ICT students towards learning ICT technologies making them deep learners?"*

Fascinated by collecting points and earning badges, the authors had the idea to enhance respective classrooms with game design elements. With this idea in mind, the authors aim to evaluate the use of Gamification in higher education in order to improve students' engagement epitomized by the online community in a tertiary institution in Vietnam. Werbach and Hunter [4,17] advocate us to build a Gamification System organized in a decreasing order of abstraction. They introduced three abstraction layers: dynamics; mechanics; and components. In short, this paper aims to document the expedition of planning and implementing a Gamification system into a classroom environment. To provide answers towards a niche area of research assisting the reader to entirely comprehend the subject matter, this paper has been divided into numerous phases. The goal of this approach is to guide the reader contextually into 6 Phases: The first phase, (Phase 1: Inceptions) helps to understand the foundations behind motivation, engagement and Gamification.

Figure 3: Smartphone Virtual Reality (VR) Based Learning

Proceeding to the second level (Phase 2: Theoretical Objectives), the authors give a synopsis about relevant work previously attempted in the research and development phase. The journey of building and evaluating Gamification in the context of classroom based teaching and learning, the third level (Phase 3: Methodology) provides an overview of the milestones. Moving to the fourth phase (Phase 4: Conceptual Framework), the goals of the paper were demarcated. The penultimate level demonstrates the concepts behind the system (Phase 5: The Results and Findings) indicating the learning and assessment outcomes. In order to provide the data for the findings and discussions, the last level (Phase 6: Conclusion and Future Directions) aims to evaluate the implementation of Gamification within and outside the classroom contexts. All the six phases enables the reader(s) to walkthrough a virtual journey summarizing the vital aspects of the research work undertaken; and evaluating the results in the context of learning and teaching as a future endeavour.

4 Conceptual Framework

The research methodology involved the use of Action Research (AR) based framework which involved a continuous feedback process in place to ensure alignment of the courses offered at all campuses of the selected institution. The participatory research component involved the motivation of Non-IT students accessed in a Vietnamese tertiary institution (denoted as Institution A for the sake of argument) within the context of gamification towards learning IT content and using Game components in teaching and learning environments. So the idea was to gamify a non-game environment such as a classroom (Computer Literacy for Business Students – CLB at Institution A) environment where the student backgrounds were not from IT but from Business, Social Sciences, Accounting and Finance. The feedback component was linked to Badges (Experience points) which were counted with relevance to the assessment component namely percentage was converted into points.

Figure 4: Conceptual Framework

Students motivational levels were assessed based on the following Instruments: Participatory research (included in class observation and feedback), F2F Interviews with selected students, Social Networking and Gamification Tools, Questionnaires and Student Participation and Feedback (Refer to Figure 4 for the functional aspect of the pilot project executed).

5 Results and Findings

During this course, approximately 50 percent of class time was used for traditional instruction (direct instruction, whole-group presentation activities, etc.) and the other 50 percent of class was devoted to students working on self-selected activities and projects to meet the learning objectives of the course.

Table 2: Pre and Post Gamified Statistics

Pre-Gamified Sessions (Week 1 – 7)		Post-Gamified Sessions (Week 8 – 15)	
Total Posts:	129	Total Posts:	1105
Total Contributions:	310	Total Contributions:	2330
Instructors' Responses:	50	Instructors' Responses:	186
Students' Responses:	114	Students' Responses:	244
Average Response Time:	0.23 Hrs	Average Response Time:	1.3 Hrs

The sampling data as shown in Table 2 was applied to a student population of around 160 students which included 70 males and 90 females with a ratio of 1:3 studying at the designated Institution A.

Their progress in an online gamification system was recorded using Blackboard (BB) and third party tools by claiming points for their completed work and moving up to new levels in the leader boards based upon their points earned.

Some activities were completed without technology usage while others required the utilization of technologies by our students which comprised of: Google Docs; Prezi; Google Spreadsheets; Facebook; Twitter; and Zondle.

Students were given 10 minutes per week in class to reflect upon their learning, and to share their work as portfolios. In order to enhance their learning outcomes, students have to retain their reflections on their USBs for easy retrieval on a weekly basis.

Students were encouraged to submit the USB's by Week 15 to showcase their contributions assessing every week of the work undertaken by them in the class as well as online (maintaining a 1:1 ratio of Blended vs. F2F Learning).

5.1 Leaderboards and Badges

The classroom environment was gamified within the scope of making students no longer asking about their grades eventually.

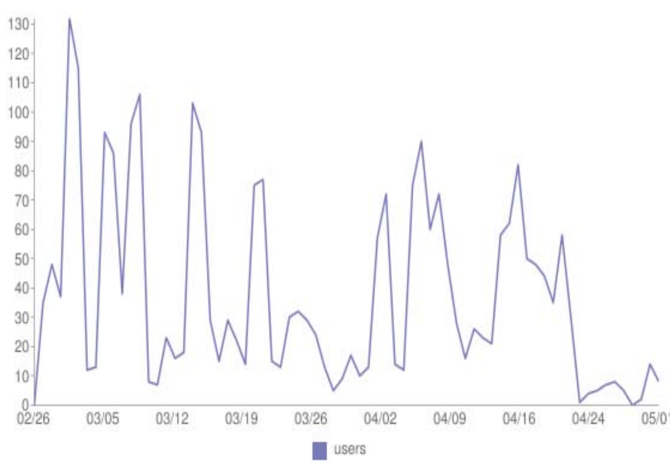


Figure 5: Online Student Participation (Blended Statistics)

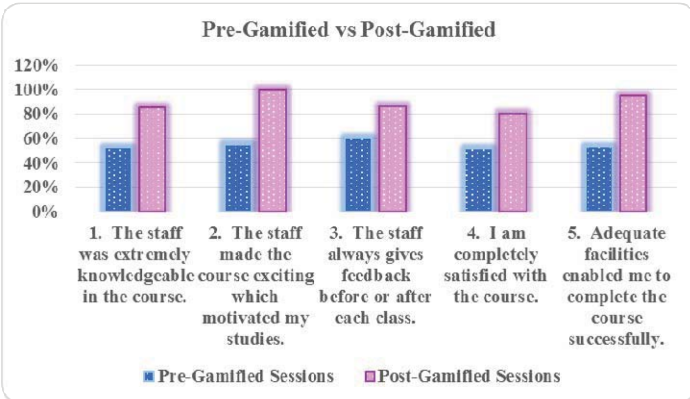


Figure 6a: Student Learning Experience

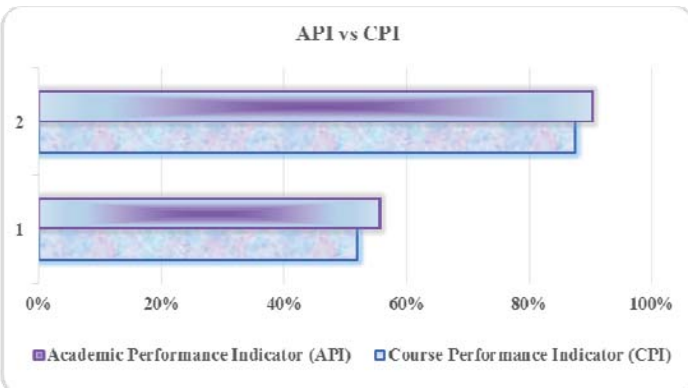


Figure 6b: Performance Indicators

This was something of a habit that they had to 'Unlearn' from their previous learning experiences and courses. Instead of motivating them in reaching the top of an "academic" leader board, we allowed students to gain experience points as opposed to marks.

Ultimately, students did focus more on academic skills (refer to Figure 6a,b which indicates the students' continuous feedback about the academic staff, followed by the course evaluation) as required for the assessments and the students started to venture back into surface learning from deep learning due to the structural aspects of the course. Therefore, it was observed that focus from conceptual learning to mark orientation eventually came back after the gamification process was stopped.

6 Conclusion and Future Directions

Overall, the Gamification project implemented in this course was a success. The feedback process included: observations; reflections; and data collection about students' learning and motivation.

Results were compared to a pre-gamification and post-gamification assessment context as indicated by Figure 5 which shows that there was a high level of motivation during the application of gamification using a goal oriented approach for this gamification environment for weeks 8 – 15.

Although it can be argued that there was a statistically insignificant gain in student performance (which cannot be completely perceived in a one semester project as seen in Figure 5), the project was still successful because of the improvements in student motivational levels and attitude (based on the feedback and reflections) as perceived from the results of pre/post-gamification data from the previous non-gamification environment refered in our previous works [4,5].

Only by carefully unpacking consumer (learner centric in our case) emotions and desires, we can design something that really stays and only through the power of gamification, we can make that experience predictable, repeatable, and financially rewarding [11].

Therefore, seeing educators and academic researchers embracing the concept of gamification should come as no surprise at all. As our society becomes more and more game obsessed, much of the conventional wisdom about how to design teaching environments and online content for our students is no longer absolute.

To further engage our audiences, we need to consider reward structures, positive reinforcement, and subtle feedback loops alongside mechanics like points, badges, levels, challenges, and leader-boards [11].

Hence, we can safely state based on our research question's inquiry that Gamification can be applied as a tool for increasing the motivational levels of Non-ICT students towards learning ICT technologies.

7 References

[1] Amaldas, Christine. and Shankaranarayanan Avinash, Business Communication for Technopreneurs Student Handbook (Textbook Ed.), 2015. ISBN:978-981-09-1376-2.

[2] Amaldas, Christine. and Shankaranarayanan Avinash, Organizational Behavior and Cognitive Development Student Handbook (Textbook Ed.), 2015. ISBN: 978-981-09-1379-3.

[3] Amaldas, Christine. and Shankaranarayanan Avinash, Technology Management for Technopreneurs Student Handbook (Textbook Ed.), 2015. ISBN: 978-981-09-7982-9.

[4] Shankaranarayanan, Avinash. and Amaldas, Christine, (2013), The Art of Cloud Computing for Businesses, Aries Media Publishers, ISBN: 978-981-09-0152-3.

[5] Amaldas, Christine, Shankaranarayanan Avinash, and Kiminori Gemba. "Understanding Academic Assessment Mechanisms: A Case Study on the Impact of Western Education Influences in Asia." Proceedings of the International Conference on Frontiers in Education:

Computer Science and Computer Engineering (FECS). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2012.

[6] Biggs, John. "Asian learners through Western eyes: An astigmatic paradox." Australian and New Zealand Journal of Vocational Education Research 2.2 (1994): 40.

[7] Chatfield, T. TED: 7 ways games reward the brain, 2010. Retrieved November, 10, 2016.

[8] Cimigo, V. Vietnam NetCitizens Report - Internet Usage and Development in Vietnam Cimigo Vietnam., Cimigo Vietnam, May 2013.

[9] Deterding, Sebastian. "Gamification: designing for motivation." interactions 19.4 (2012): 14-17.

[10] Deterding, Sebastian, et al. "From game design elements to gamefulness: defining gamification." Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments. ACM, 2011.

[11] Deterding, Sebastian, et al. "Gamification: Toward a definition." CHI 2011 Gamification Workshop Proceedings. 2011.

[12] Hofstede, Geert. "The cultural relativity of the quality of life concept." Academy of Management review 9.3 (1984): 389-398.

[13] Loebbecke, Claudia, Tung Bui, and Wilh Backhaus-Str. "System Dynamics based Modeling of the Global System of Mobile Communication (GSM) Market: A Comparative Study of Germany, Hong Kong and Vietnam." (1997).

[14] McGonigal, Jane. "Jane McGonigal: Gaming can make a better world.[Video file]." (2010).

[15] McGonigal, Jane. "Gamers have skills. Let's tap'em." Christian Science Monitor 5 (2007).

[16] Mobile, V. Snapshots of Vietnam's Mobile Market, Dataxis News, (June 2009).

[17] Nielsen, Jakob. "Participation inequality: Encouraging more users to contribute." (2006).

[18] Platt, Jennifer. "“Case study” in American methodological thought." Current Sociology 40.1 (1992): 17-48.

[19] Priebatsch, Seth. "The game layer on top of the world." TEDx Talks (2010).

[20] Tweed, Roger G., and Darrin R. Lehman. "Learning considered within a cultural context: Confucian and Socratic approaches." American Psychologist 57.2 (2002): 89.

[21] Werbach, Kevin, and Dan Hunter. For the win: How game thinking can revolutionize your business. Wharton Digital Press, 2012.

[22] Windsor, Duane, and George Greanias. "The public policy and management program for case/course development." (1983): 370-378.

[23] Zichermann, Gabe, and Christopher Cunningham. Gamification by design: Implementing game mechanics in web and mobile apps. " O'Reilly Media, Inc.", 2011.

Prof Avinash Shankaranarayanan is currently working as an Academic/Consultant teaching Accounting & Finance, Business, ICT, International Human Resource Management and Financial Risk Management courses at RMIT International University. He is also an Adjunct Professor at VELS University, India; Trier University, Germany; and the Institute for Applied Sciences (IFaS), Germany. Avinash has been actively involved in publishing several research papers and periodicals in International Conferences and Journals of high standards such as the IEEE and ACM. His research and teaching interests includes: Applied Social & Cognitive Psychology in Education; Game Dynamics; High Performance Grid Computing; Bioinformatics; Material Flow Management; Renewable Energy Systems; Policy and Decision Making. Avinash comes from a diverse discipline of Engineering and Social Sciences and is fluent in a wide range of scholarly domains specializing in Higher Education and Engineering. He serves as an Editor and Steering Committee member on numerous research entities including, the Journal of the Environmentalist and Biologica.

Prof Christine Amaldas is currently teaching core Business and IT technologies at Ritsumeikan while serving as a visiting Professor and Consultant for numerous organizations and international tertiary institutions spanning the Asia Pacific region. As an expert researcher and academic, she has been the author of several Books, Chapters, Conferences and Journal periodicals. She has Chaired and presided over a number of Conferences and has given numerous Keynote speeches and Guest talks. She specializes in Asia Pacific Studies, Education, Information Security and Fraudulence, Ethics and Ethical Governance in ICT, Holistic Medicine & Energy Healing and Governance (Corporate, Public, IT and Dynamic). In 2010, she was the first person to receive the 2010 Teaching and Research Award from the Royal Melbourne Institute of Technology University, Australia for her excellence in teaching and research methodology. Moreover, she was the recipient of the prestigious Best Paper Award at the ANNA World Congress 2011, Chennai, India. Furthermore, she was the recipient of the prestigious ‘2012 Young Scientists Award’ from ASTER, Japan and has received numerous Grants & Awards since 2012. She is currently busy writing many articles and books in multidisciplinary areas of research.

Game Design for Computer Science Majors

Philip J. Bernhard
School of Computing, Florida Tech, Melbourne, Florida, 32901

Abstract - We describe our experience developing and delivering a course in computer game design to students majoring in computer science. Constraints and objectives for the course are described, plus lessons learned from our experience, including things that worked and things that didn't. In general, we have found the course provides a very good platform that integrates a variety of topics from algorithms & data structures, graph theory, software engineering, statistics, probability, and psychology.

Keywords: Game design, Compute Science undergraduate education

1 Background

Several years ago we decided to offer a course on games in the school of computing at Florida Tech. The decision was motivated by several factors. Most noticeably, we had observed a decline in enrollments in recent years, and we thought that adding a course on games might help attract students. There was also a lot of interest and demand from current and incoming students; over half of our incoming freshman routinely expressed an interest in game design or development and we felt compelled to respond. There were, however, several constraints and issues that we had to deal with.

First, although many different types of courses on games are possible (design, physics, graphics, engines, etc.), lack of resources limited us to the addition of just a single (elective) course. This is in contrast to some departments that offer several courses on the topic, sometimes in the form of a specialization, concentration, minor, or in some cases a complete major [1]. This led to the second issue – what topics do we cover from the multitude of those available?

Finally, although there were several members of our faculty in related areas, e.g., graphics, none specialized in games, per se. This created the obvious concern - how could we possibly teach a course on games to a sizeable group of students who have been exposed to computer games from a very early age, and who play them on a frequent basis?

We quickly realized, however, that although there were plenty of games and concepts with which we were not familiar, we did have more experience than we thought. We hadn't played Halo, Warcraft, or Sims, for example, but we had played Tetris, Minesweeper, Pacman, plus numerous card and board games. In short, we weren't serious gamers, but we had plenty of experience with casual and non-computer games. One of the things we discovered is that most of the important concepts directly transferred from the experiences that we did have. There was certainly plenty of room to grow, but there was also a decent basis from which to start.

2 Course Content

In this section we describe the course content, how it was initially selected, and how it evolved. We also describe some of the principles that guided us during development and delivery of the course.

2.1 Initial Delivery of the Course

Given our lack of experience in the area, and our lack of a specific topical focus, we decided to implement the first iteration of the course in a manner similar to what we personally experienced as students in several humanities courses. Specifically, a book was selected for the course, along with other readings from various conferences and magazines. At the end of each class students were given a reading assignment for the next class, and were required to bring a 1-2 page summary of that reading along with questions, issues, points of contention, etc. Class time was spent discussing those issues, and no formal lecture was given. In summary, our initial delivery of the course had an informal, seminar-style format. Finally, for lack of a better title, the course was called *Game Design*, and offered as a special-topics seminar.

This particular approach had several advantages. First, as novices in the area ourselves, it gave us time to learn the material, basically alongside the students. Second, we have always found student enthusiasm in this class to be very high, and with the seminar format we were able to exploit this and rely on students to drive the discussion. Often times very simple questions such as *"Which are better, console games or PC games?"* prompt an in-class debate that required very little input or additional direction from the instructor. On the other hand, since there were no formal exams or quizzes, the seminar-style approach did make it difficult to assess student outcomes and assign final grades.

Over the period of several years, the course evolved into a form more typically used for science or engineering courses. This included a more precisely defined set of topics, a semester-long game design group project, individual writing assignments, plus quizzes, and a full set of power-point slides.

2.2 Current Course Content

One of the first things we learned was there is a broad range of subtopics under the topic of games, encompassing enough material for several courses. For example, individual courses on game design, implementation, physics, artificial intelligence, modelling, graphics, graphic design, storytelling, dialog, scripting, engine development, and project management, among others, could be offered. Courses that focus on specific game development tools such as Unity, Blender or Unreal Engine are also possible. As noted previously, due to our restriction to a single course, we had to select a relatively narrow set of topics that would be appropriate for, and of interest to our computer science students.

Currently, our course covers the following areas, all of which are part of the 2008 IGDA Curriculum Framework [2]:

- Game Vocabulary – What is a game? What are the key components of a game? How are games different from puzzles, toys, and sports?
- The Video and Computer Game Business – Publishers, developers, the International Game Developers Association (IGDA), the Entertainment Software Rating Board (ESRB) and other trade groups.
- Game Design Concepts – Mechanics, reward systems, level design, immersion, balance, uncertainty.
- Artificial Intelligence – Traditional vs. game AI, path finding, state machines, dynamic game balancing.
- Procedural Generation – Algorithms such as midpoint displacement (and variations), fault formation, cellular automata, random number generation.
- Game Physics – Collision detection & resolution, space partitioning.

2.3 Teaching from a List of Rules

Whether it's online resources, books or other sources of information, one thing we discovered is that a lot of game design material takes the form of rules. This is best exemplified by the 400 Project [3], which had the goal of compiling a list of 400 game design rules. In this project each rule is relatively short – consisting of an ID# (1 through 400), an imperative statement (summary or title), an explanation in 250 words or less, a domain for the rule, and the name of the rule contributors. Figure 1 shows an example of one rule from the project.

As of the writing of this paper, the list consists of just over 110 rules, submitted by several contributors, most of whom are well known game designers. The project started in 2001, and over the past few years the list does not appear to have changed.

ID #: 106
Imperative Statement: *Have Fun in the First Minute*
Explanation: *In casual games it is critical to make sure the player is having fun right away. If the game is an expensive, boxed game then this rule is not critical (although still good to follow).*
Domain: *Casual Games*
Contributors: *Steve Meretzky*

Figure 1: A sample rule from the 400 Project.

Stating game design principles or concepts as rules is natural and very common. Although not quite as explicit as the 400 project, we have seen game design principles specified as rules in magazine articles, conference and journal publications, and books.

In general, we have found most such rules to be concise, insightful, and easy to explain. The one problem we had with this format, however, is that by itself it doesn't make for the best classroom presentation. Early in the development of our course we found our in-class slides to be dominated by list after list of such rules. This did not make for the best classroom presentation, or the best basis for discussion, and students very quickly lost interest.

More recently we have avoided letting this format dominate our materials, and we would recommend that other course developers do the same. One way to do this is to focus on a small number of high-priority game design rules, and supplement their presentation with supporting material, such as examples of YouTube videos of published games.

2.4 On-Line Resources

It should come as no surprise that the internet provides an abundant source of both instructional and non-instructional game design material. As noted in the previous section we have found YouTube in particular to be a tremendous source of pre-recorded video transcripts presenting many of the concepts we teach. These frequently come from games the students are familiar with, making it easy to keep them engaged. Additionally, most game engines and related tools have instructional videos available, which help students learn the tools on their own. Finally, on-line books, courses, the Wikipedia, and even blogs and user groups provide more material than we could ever use.

Of course, quality is always a concern with materials on the internet, and anyone using such material should consider their sources and take appropriate precautions. Just like any academic topic, the internet is full of both good and bad material on the subject of game design.

2.5 Vocabulary

One area where this last point is particularly apparent is in the vocabulary of gamers, designers and developers. Put bluntly, there is a lot of jargon associated with game design & development, and different authors, bloggers, and teachers will frequently use terms and phrases in conflicting and ambiguous ways. For example, the phrase "game mechanic" is one that we struggled with for some time. The large number of explanations on the internet created just about as many problems as it solved when we were trying to provide a reasonable definition to our students.

Of course, sometimes this is more a reflection of the subject matter than any particular author. For example, the term *game* itself is notoriously difficult to define precisely, as has been discussed by many authors. In such cases, instead of asking how a word is, or should be defined, we change the question to focus more directly on what we are really trying to teach. For example, instead of asking how the word *game* is, or should be defined, we ask *what are the essential elements that make up a typical game?*

3 Assignments and Projects

In this section we describe the course project, individual assignments, and mechanisms used to evaluate the students. This includes a semester-long game design team project, individual research/writing assignments, and on-line quizzes.

3.1 Project Description

From the very first delivery of the course we decided to have a semester-long, group project that involved design and implementation of a game prototype. At the beginning of the semester the class is divided into groups, typically consisting of approximately half a dozen students each. Students are allowed to form groups themselves, both in terms of membership and roles. However, in some cases teams are modified for students who can't find one themselves. Depending on the specific semester, and class size, this has resulted in anywhere from 4 to 9 teams.

Each team is required to give three presentations throughout the semester:

- Game Concept, Platform & Tool Selection – A brief summary of the game genre, target audience, major goals & objectives for the player, plus an indication of the hardware platform (PC, console, mobile device) and software tools to be used during development (game engine, modeling tool, software development tools).
- Game Design – A summary of the game design (not the software), the mechanics, gameplay, characters, terrain, audio and graphic design, reward systems.
- Game Demonstration – An in-class demonstration of the final prototype.

The first presentation is typically given in the first 3 weeks of the semester, the second is given around midterm, and the third is given the last week of class.

3.1.1 Selecting and Enforcing Prerequisites

An important lesson we learned after several iterations of the course was that by enforcing appropriate prerequisites, we were able to rely on the students' abilities to figure out development tools on their own, outside of class. This was particularly true for game & physics engines, such as Unity and Unreal Engine, plus modeling tools such as Blender.

In our curriculum computer science majors are all required to take two introductory programming courses, plus a course on algorithms & data structures before taking the game design class. This, combined with the fact that some students had already used such tools, plus the multitude of instructional YouTube videos and other resources on the internet, ensured they were able to identify, download, install, and quickly learn the required tools with virtually no classroom discussion.

This selection of prerequisites was anecdotally supported by the fact that in the few situations where we did approve a student taking the class without all of the prerequisites – most notably algorithms & data structures – those students struggled with the class project, significantly more than other students. Frequently students from other majors that don't have the prerequisites still believe they have some other experience with game design that is sufficient, but in all cases we have found that not to be the case.

Conceivably, if the project were modified to focus only on design with no prototype, or if the course as a whole were refocused on teaching how to use tools such as game engines, then the course might meet the needs of such students better. This latter change would, of course, be a substantial deviation from our current course content, which does not focus on tool details at all. We also believe it is not necessary for our current audience. Being able to leave this responsibility to the students has proven to have many advantages and allows us to focus on issues that we feel are more appropriate for the majority of our audience.

3.1.2 Tool & Platform Independence

With few constraints we allowed teams to select whatever hardware and software tools they wanted. As noted in the previous section, students were then required to acquire and learn the tools on their own. We did this for several reasons. In particular, many students are already biased against certain game development engines and platforms, and this flexibility allowed us to attract the broadest possible audience. It also helped free us from having to teach a specific tool in class, and allowed the class to adapt from semester-to-semester as different tools enter and leave the market, without us having to update course materials in a significant way.

As noted in the previous section, we do not teach specific tools or how to use them. It was not our personal interest, nor possible given the combination of goals and constraints that we had for the course. That having been said, courses on specific tools could be offered, and many are in various schools, but we did not want to take this approach.

We also found students more than willing and capable of selecting appropriate tools. A sizable number of the students have already used game design or development tools, mostly on their own time. We have had no complaints from students feeling overburdened by having to select and learn the tools and in only a few cases have we had a team suffer a substantial failure due to poor tool selection or inability to learn one.

3.1.3 Use of On-Line Assets

As noted in Section 2.4, the internet provides access to a large quantity of online resources that can be used during game design and development. This is particularly true for what is commonly referred to as game content or assets such as clip-art, character sprites, music clips, partially/fully developed terrain, fully functional objects or characters, and photos. Often times these are freely available, while other times they are sold with licenses that allow them to be integrated into games. In one recent, somewhat interesting case, students asked if they could use a "dancing zombie" in their game, which they were able to purchase over the internet for under $25.

Initially we allowed students to integrate such assets into their games. We allowed this for several reasons. Most notably, a typical game will incorporate many such assets, and during a one-semester course we expected that students wouldn't have enough time to develop all the art and music themselves. Furthermore, a lot of what goes into development of a game has little to do with the (art) assets, but rather the game rules, mechanics, and physics. These later items are precisely where we believe the focus of the course should be for our students.

Of course, it should come as no surprise that students took substantial advantage of their ability to use online assets. We now require however, that they 1) give credit whenever they use such assets, and 2) include in their game prototype something other than just a loosely connected collection of downloaded assets. In other words, the majority of the effort that goes into their game is expected to be original development, and reused assets should be kept to a minimum.

More recently we have also concluded that students should document proof that they are not violating a copyright restriction or intellectual property law by using such assets. This became particularly apparent when in a recent project students incorporated a scene and music from the movie Star Wars. Consequently, in future delivery of the course we plan to require students prove that their use of such an asset is legal, even if only in the context of a college course.

3.1.4 Game Project Theme Restrictions

One of the first things students are required to do at the start of their project is to come up with a game concept. For example, students might decide to develop a first-person shooter that takes place in Europe during world-war II, a Mario-style side scrolling game, or a car racing game. Sometimes students decide to develop their games in 2-D, while others use 3-D. In the past we have not restricted students too much in this regard, except that they not develop a game that will create a public relations problem for the university, should it get outside exposure. For example, several semesters ago one team proposed developing a marijuana growth and harvesting simulation game. Although such a game has the potential for several interesting horticultural aspects (fertilizing, watering, light exposure, etc.), students were nonetheless encouraged to choose a less controversial plant.

We have seen examples of courses in other universities where students are restricted to 2-D games in one course, and 3-D games in another. While we do see the merits of such an approach, we have not used this particular restriction, but we have used others. For example, many of our students have developed games that have a Florida Tech university theme. One game in particular, called "Trolley of Terror," simulated the driving of a passenger trolley around campus. This was based on the actual, real trolley that traverses the Florida Tech campus, picks up passengers and delivers them to their destinations. Similarly, faculty, administrators, and even the students themselves have made appearances in their games as avatars, and in general students have found this very entertaining and motivating.

The use of university themes has become so common that we have considered restricting the students from doing this. Although it has definitely helped keep the students engaged and entertained, it has made the game design process easier for them than it probably is in the real world. In other words, it's relatively easy for them to develop a university themed game that gets laughs from the entire class, but more difficult to do so otherwise. For similar reasons we have considered discouraging students from developing games around current topics such as a recent election and its candidates - something that appears interesting and funny at the beginning of the semester, but has frequently gone stale by the end.

3.1.5 Final Demonstration Evaluation

We have found the final in-class demonstration to be very informative and, frankly, a lot of fun. After each team gives their demonstration, one of the unexpected things we do is ask two or more non-team members to play the game.

The reaction of the players, and the observing class, helps give us a feel for the playability of the game and the degree of success at creating a product that's fun. Of course, performing such demonstrations in class does require classroom media support, including a computer, projector, screen, and speakers or a sound system.

Although we don't track attendance for most classes, we do require all students to attend all presentations. Otherwise students tend to skip those classes when their team is not presenting. We have found student engagement and interaction is especially high during the presentations, provided the students are in attendance. This is particularly true of the final demonstration.

3.2 Individual Assignments

As noted previously, and by other authors, enthusiasm on the part of students for this topic is typically very high. This made it easy to prompt classroom discussion. Even students who haven't prepared for class or completed required reading are more than happy to express their opinion on just about any topic related to games.

One way we harness this energy is to give students individual, quick turn-around assignments that requires them to research/investigate a topic, write a short summary of their findings, and discuss their findings in class. Typically, these assignments are based on a game design concept previously discussed in class, but in general their focus and format varies greatly.

Sometimes we ask students to find an example of a game concept as it appears in a specific game. Other times we ask them to evaluate a specific game to see how it implements a concept. Regardless, in all such cases the objective is to prompt classroom discussion. Simply put, on the day when such an assignment is due, we ask a select few to tell us what they discovered, and typically this results in a lively discussion.

3.3 On-Line Quizzes

Evaluating and grading student performance has continued to be a challenge. During early iterations of the course this was due to the fact that we simply didn't have a good feel for the material ourselves, so it was difficult to know what the students were capable of, and how we should evaluate them on this type of material. Over time we have developed a better idea of what to expect on the projects, what kinds of game designs were realistic and what were not, how close students should get to a completed game by the end of the semester, etc.

More recently we have started using on-line quizzes to evaluate students' understanding of the topics discussed in class. The quizzes are generally objective-style questions, such as T/F or multiple-choice. All quizzes are open-book and open-notes. The students are even allowed to use google and access the course slides during the quiz. The only restriction is that they do the quizzes by themselves. In order to minimize the effect of free access to materials, we also put a limit on the time students have to take the quiz; approximately 30 seconds per question. Finally, students are only given one attempt at each quiz.

Students are informed of the time-limit in advance, and advised to study sufficiently so that their use of materials during the exam is minimized. Collectively, all of these constraints, plus our own growing understanding of the material, has resulted a more reasonable and acceptable grade distribution.

4 Conclusions

We first offered this course in response to student demand and declining enrollments. At the time, we had several faculty members who specialized in various aspects of computer graphics and vision, but no one who was an expert in game design or development. This created some concern at first – how could we possibly teach a course on games to a sizeable group of students who have been exposed to computer games from a very early age, and who play them on a frequent basis?

We quickly realized however that although there were plenty of games and concepts with which we were not familiar, we did have more experience than we thought. We hadn't played Halo, Warcraft, or Sims, for example, but we had played Tetris, Minesweeper, Super Mario, plus numerous card games and board games. In short, we weren't serious gamers, but we had plenty of experience with casual and non-computer based games, and most of the important concepts directly transferred. There was certainly plenty of room to grow, but there was also a decent basis for which to start. We would encourage anyone considering such a class to give it a try.

5 References

[1] Colman, R., Krembs, M., Labouseur, A., and Weir, J., *Game Design & Programming Concentration Within the Computer Science Curriculum*, SIGCSE'05, pp. 545-550, 2005.

[2] IGDA Curriculum Framework, The Study of Games and Game Development, International Game Developers Association, February 2008.

[3] Barwood, H., *The 400 Project*, www.finitearts.com/pages/400page.html.

SESSION

SCIENCE, TECHNOLOGY, ENGINEERING AND MATHEMATICS - STEM EDUCATION AND COMPUTATIONAL SCIENCE

Chair(s)

TBA

ADDRESSING GLOBAL EDUCATION CONCERNS-TEACHING COMPUTATIONAL THINKING

John N. Carbone, PhD, James A. Crowder, PhD
Raytheon Intelligence, Information and Services
{ jcarbone@raytheon.com, jacrowder@raytheon.com }

ABSTRACT

Although education is sometimes characterized by the daily heroic efforts of some teachers, voices across the global education continuum acknowledge numerous significant education failures. It is well known that many countries excel educating their masses where others fail. Countries perceived to excel are also known to place extraordinary pressure upon their students. Others, which also excel, have also been known to still have high illiteracy rates. Additionally, some countries focus more attention on improving only good students, leaving many behind; a "survival of the fittest" philosophy. Specifically, education continues to be hampered by serial learning, complacency and regurgitation education characterized by "teaching to standardized tests" and reward systems which continue to emphasize successful memorization instead of qualitative learning. Thousands of years of educational methodologies are not changed easily. Hence, it is well known that preparing global youth for complexities and technologies far beyond the 21st century requires significant qualitative education reform. Globally, studies and analyses have continuously concluded that lack of skills are an epidemic and that we fall behind significantly each and every year. Therefore, this paper addresses these concerns and provides potential solutions for improving global education challenges by combining cognitive psychology research, complexity management theory and proven Computational Thinking (CT) benefits. These concepts are then integrated into existing learning systems for facilitating broad effective and qualitative educational change, even for improving the physical brain composition of individuals not naturally disposed to CT benefits.

KEYWORDS

Computational Thinking, Global Education, Complexity Theory, Complexity Management, Cognitive Psychology

1. INTRODUCTION

It is well known that the act of learning and developing new insights continuously develops from the combination of experiences, actions and inquiries, whether scientific or personal. Additionally, it is well known that the level of education individuals achieve is also influenced by personal, cultural, and socio-economic factors. However, taking account of these factors and differences in education systems is not enough. There still exists an epidemic facing education today; talented individuals are lost within many education systems and fail reaching their full potential. Although global statistics are difficult to aggregate continually, the global Organization for Economic Cooperation and Development (OECD), founded in 1961, which now includes experts from 34 nations, has been tracking global education achievement via a triennial international survey aiming to evaluate education systems worldwide by testing the skills and knowledge of 15-year old students. This program is known as the Program for International Student Assessment (PISA) whose most recent report was published in 2012. This comprehensive assessment represented more than 80% of the world economy from over 65 participating countries and economies. The report outlines, in great detail, the factors and relationships behind recent and historical successes and failures while also providing detailed recommendations, improvements and detailed surveys of student happiness and disenchantment. The 2009 OECD global study concluded that 28% (avg) of people between the ages of 25-64 had not completed upper secondary of education and a 2010 OECD study concluded global average dropout rates for higher education over 30% with the United States leading the pack of developed nations with over 50%. By the 2012 study three countries had improved their numbers of top performers and simultaneously reduced low performers while 25 countries improved in mathematics, and 32 countries had improved their reading scores. Most importantly, the 2012 study pointed to high potential improvements gleaned from telling signs related to the positive multi-disciplinary effects of mathematics [1].

"The OECD's new Survey of Adult Skills finds that foundation skills in mathematics have a major impact on individuals' life chances." The survey shows that poor mathematics skills severely limit people's access to better-paying and more-rewarding jobs; at the aggregate level, inequality in the distribution of mathematics skills across populations is closely related to how wealth is shared within nations. Beyond that, the survey shows that people with strong skills in mathematics are also more likely to volunteer, see themselves as actors in rather than as objects of political processes, and are even more likely to trust others. Fairness, integrity and inclusiveness in public policy thus also hinge on the skills of citizens."

Hence, the many nation study concluded that mathematical computational skills and related characteristics are key components for improving and transforming global education. Mathematics is defined to be the broad abstract science of numbers, quantities and spatial understanding and therefore, if generally available, could potentially provide significant value in modeling understanding for the masses and broader applications across many disciplines. Fisher describes the goal of training socially-engaged, computational thinkers as paramount. The professional societies and accrediting boards, most notably IEEE and ABET, are insisting on "an ability to analyze the local and global impact of computing on individuals, organizations, and society." (ABET criteria, p. 19) and like characteristics among university computing graduates. Fisher goes on to say that these institutions recognize that segregating technical training from technology's broader impacts do not serve society well.

Therefore, requiring application based learning. The potential solutions this paper proposes reflect on "how" to harness these computational powers within mathematics and "how" to improve education qualitatively for all students even when they are not naturally disposed to be quality mathematicians. The key to enhancing global education with CT is first instilling a questioning mind as opposed to a simply receiving and parroting one [4].

1.1 Learning Components, Constraints, Opportunities

To discover pragmatic solutions the next sections explore well-known human motivations for learning from well-researched fields in Psychology, Cognition, Biochemistry, Neuroscience, and constraints imposed by complexity and the general human condition.

From the time humans developed opposing thumbs the human race has continued to automate and optimize. Whether preparing for the onset of deep winter, or more efficiently hunting for food, or finding a way to cave paint at night, understanding the environment and everything in it was absolutely fundamental to survival. Thus, prior to spoken and written language we learned by *doing* which became infused in our DNA. Today, this remains unchanged regardless of the advanced scientific methods and implements at our disposal. What is true is that the further we forge, enhance, automate, and improve, the greater complexities we encounter. We vigorously attack each new challenging complexity with what we have learned for centuries. We consistently discover new needs for developing a new type of hammer or nail, or even a hammer that can simultaneously drive many nails at once. Thus far, our survival has been predicated by the understanding of our environment, creating language, telling stories and then finally only a few thousand years ago actually writing down to pass on the continuously learned fundamental truths, whether on papyrus, paper or digital files. Subsequently, we entered the era of books. We began to focus more on memorization of material and less *doing*. We took tests and became experts in a given discipline. Where once only the strongest survived, now the most well-read and best-memorized could better survive and became more cherished. The automation objective became working smarter not harder, increasingly surviving and advancing with less labor. Continuous automation and optimization fueled by survival and then the industrial revolution, continues in complex dramatic fashion in our BIG data world today.

However, through the advancements in computation, across all of the sciences, fueled by computer science and the significant brain research of the likes of Nobel laureate Dr. Eric Kandel and Dr. Timothy Tully point to significant near term opportunities for brain function and educational improvement. Their research shows that the basic mechanics of human memory formation are not much different from those of simple creatures like flies and snails. The difference seems to lie within the complexity of the wiring that links them. In the early 1950's a well-known literature study involving a severe epileptic patient known as H.M. resolved some great misconceptions. In an attempt to cure the epilepsy doctors removed the Hippocampus, a small ridge in the center of the brain. His seizures subsided completely and his reasoning remained intact, but he lost short-term memory almost completely although he could remember everything from childhood. Hence, the Hippocampus was crucial for perception translation into memory but not the final storage location. Dr. Kandel used this research to begin his landmark research of the sea slug which has 20,000+ nervous system cells. In contrast humans have 100 billion brain cells. The outcome of his research concluded that human brain cells perform a very subtle electrochemical handshake and friendship dance that reinforces links between them. Short-term memories are more chemically fleeting and longer term memories are synapses cemented in place with new proteins which can also erode, but are less likely, over time.

Hence, although the research shows potential for brain supplements, it also dramatically points out the ability to increase the chemical interchange in short term and long term memories by natural human means. Positive and negative emotions and related confidence and failures are well known, from psychology and neuroscience, to generate strong memories for happiness and survival. Therefore, this paper supposes that to enhance education we must be able to accentuate the correct electrochemical reactions with our knowledge of human nature and remove the friction of certain constraints. The following paragraphs discuss the constraints of complexity and the management of it with regard to the formation of proper memories.

2. ESCALATING COMPLEXITY

Our approach to learning has served us well except that most discipline learning has evolved over centuries as inwardly focused areas of specific and valuable myopic expertise. Hence, as information content and science continues expanding at "warp" speeds and engineering continues to explode with new complex interdependencies, disciplines become wrought with increasing levels of uncertainty and unknowns. Therefore, to achieve continuous advancement, to manage the complexity, we are driven to broaden our scope of investigation beyond the boundaries of our expertise and knowledge of our existing discipline's First Principles and concepts; the components comprising the ontology/language, concepts, algorithms etc. which make up the core of each discipline.

In order to advance knowledge and manage related complexities today requires modern mechanisms, which simply do not exist in many engineering disciplines today. New mechanisms are required to manage complexities for efficient engineering solutions across many disciplines. A fundamental realization and understanding of what forms of complexity exist [15] is required. This must be accompanied by education research, new learning and building of methodologies that can transcend traditional disciplinary or organizational boundaries, to enable appropriate solutions to large complex problems by teams of people from diverse languages and backgrounds [3].

2.1 Defining Complexity

Suh, describes that significant confusion exists within the definition of what is complex. He explains that many people attempt to understand complexity in terms of physical entities instead of focusing upon what is trying to be achieved. He describes many types of complexity (e.g. Computational, Algorithmic, and Probabilistic). Suh's computational thinking approach, describes 4 types of complexity: Time-Independent Real and Imaginary Complexity and Time-Dependent Combinatorial and Periodic Complexity. Suh mitigates overall system complexity by using a few specific set of actions: Reduce the Time-Independent Real Complexity, Eliminate Time-Independent Imaginary Complexity wherever possible, Transform Time-Dependent Combinatorial Complexity into Time-Dependent Periodic Complexity which decomposes the complexity into easier understandable and operable units. Suh describes this as determining the Functional Periodicity in whatever domain you are operating within (e.g. Biological, Chemical, Thermal, Circadian, Temporal, Geometric etc.).

2.2 **Managing Complexity**

Suh, stipulates that complexity must be viewed within the functional domain. Hence, fundamental management of complexity becomes a process of defining what we want to achieve or understand within the functional domain of a discipline, project or need. These pieces of information become the Functional Requirements (FR). How we plan to achieve a goal becomes a set of Design Parameters (DP). Figure 1, shows this concept using a Probability Density Function (PDF) description. When the System Range is completely within the Design Range then Functional Requirements can easily be achieved and is therefore considered as not complex

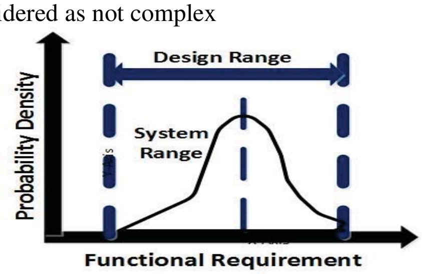


Figure 1, Design vs. System PDF

However, when the System Range extends outside of the Design Range boundaries then the design and the satisfaction of Functional Requirements become more difficult to achieve and is therefore considered more complex. Axiomatic Design processes are used to achieve optimized designs, domain decomposition into four important domains: Customer, Functional, Physical, and Process and using two important axioms. The Independence Axiom, to *maintain the independence of functional requirements* and hence minimize their overlap where possible and the Information Axiom which states to *minimize the information content* throughout the iterative design creation wherever possible. The outcome of performing these actions during any system design phase is hopefully a completely Uncoupled Design where all functional requirements are independent with each being satisfied by independent design parameters and functions. In software terminology axiomatic design provides the logical decoupling of dependent interfaces and supports development of what is common and what can therefore be abstracted effectively. This type of brain exercise is an example of the unique analogous perspective to what Computational Thinking is and how it can be applied.

3. COMPUTATIONAL THINKING

Computer Science evolution, created a now inherent revolutionary way of thinking and approaches to problem solving, primarily due to the vast application of computing for satisficing computational needs across every scientific domain. There are numerous existing publications primarily focused upon what Computational Thinking is and numerous educators/professors who have voiced speculation that Computational Thinking is either inherent or not, and hence cannot be learned. Some universities prescribe preexisting domain coursework as the mechanism for enhancing the development of Computational Thinking. Very few, if any, explore the steps to actually develop/train or rewire an individual's thought processes to allow them to benefit from Computational Thinking's value.

True efficient and optimized research cannot be accomplished today without Computational Thinking. Additionally, engineers working in varying domains today generally know their domain well but have little or no understanding, theory, tools, concepts, methods or guidance to assist them in understanding other disciplines much less the ability to engineer across disciplines developing viable solutions within complex environments. Computational Thinking is key to future design and engineering across the Multi-, Trans-, disciplinary global evolution. Hence, to enhance the quality of learning, the next sections explore details of how humans think, remember and learn.

3.1 Brain Structure Shaping Characteristics

Years of psychological research show that different people learn differently and that varying differences in learning ability, among others, is tied to varying degrees of ones ability to sense. Some are more visible learners and others not. The blind intensely develop their other senses to make up for vision. Some learn with more depth when accompanied with high degrees of emotion and/or pain. Hence, research shows that the structure of knowledge shaped in our minds is distinctively shaped within each individual based upon an extensive set of individual characteristics. Yet all of these important physiological differences are completely unaccounted for within the architecture of every machine/system we have developed, continue to develop and interact with every day.

3.1.1. Computational Thinking Models

Many believe Computational Thinking have 4 major components, as illustrated in Figure 2 [3, 4]: Decomposition, Abstraction, Pattern Recognition, Algorithms.

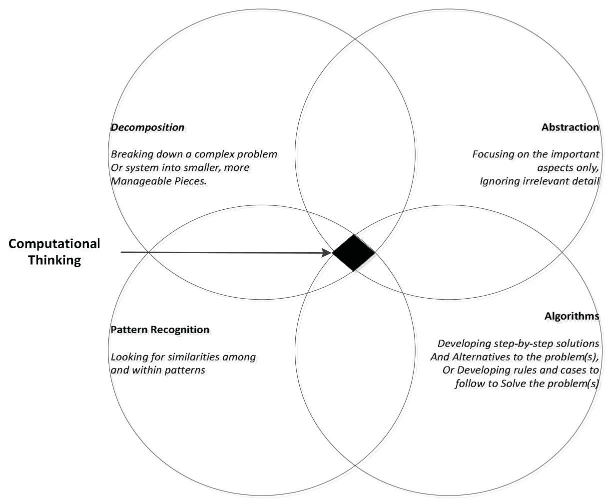


Figure 2, Major Components of Computational Thinking

While these skill sets are distinctly what Computer Science education focuses on, they are applicable to essentially all human computational thinking. Computational Thinking should be an integral part of all STEM education curriculums; providing Systems Thinking and Computational Problem Solving across all disciplines [8].

3.2 Brain Structure Shaping from Language

Significant research in computational linguistics and semantics has exposed many simple and complex structural characteristics within the ~10,000 spoken and ~3000 written languages. Some of the oldest languages map the physical world with replicated iconic forms and letters. The Korean language developed a written language by capturing physical shapes of the face and mouth during speech. Some example written languages, like Chinese and Japanese, developed around creating concise representations of their environment. Some written languages have a small number of letters or symbols while others have hundreds of letters/symbols. Many of these characteristics of language can have an effect on our capacity to learn. It is said that not until the 7^{th} grade, can a Chinese youngster read a newspaper effectively, because of significant language complexity, symbol quantity and comprehension volume. Therefore, specific language characteristics and structures seem to point to having a direct correlation between one's ability and quality of learning, how and how much humans remember, and the potential for impacting the shape of individual brain knowledge structures.

3.3 Cognition and Human Thought

Historically, there has been significant research in Human Cognition and the processes involved in humanistic thinking. In 1957 Newell et al. [16] and Simon [17] together developed models of human mental processes and produced General Problem Solver (GPS) to solve "means-end analysis" to solve problems by successively reducing the difference between a present condition and the end goal. GPS organized knowledge into symbolic objects and related contextual information which were systematically stored and compared. Almost a decade later Sternberg [14] described a now well-known paradigm called the Sternberg Paradigm where, observations of participants were taken during experiments to determine how quickly the participants could compare and respond with answers based upon the size and level of understanding of their knowledge organized into numerical memory sets. Sternberg Paradigm is known for (1) organizing knowledge and modifying context while using a common process for describing the nature of human information processing and (2) human adaptation based upon changes in context. Similarly, Rowley and Hartley [13] described the development of knowledge as the organization and processing required to convey understanding, accumulated learning and experience. Object Oriented Design (OOD), as defined by Booch and Rumbaugh et al. [9], organizes knowledge and attributes describing details of objects in the form of general objects of information, using a bottom-up approach, iteratively building its components and attributes through a series of decisions. More recently, Gruber [11] described the collection of knowledge and context on the web as "collective intelligence" and Van Ittersum et al. [12] organizes knowledge and context as individual stand-alone knowledge components in agricultural systems which can be linked using a software infrastructure. Finally, Ejigu et al. [10] defined the organization of knowledge and context as a well-defined process of collection and storage. Carbone aggregated and formalized these processes into a Recombinant kNowledge Assimilation (RNA) cognitive workflow.

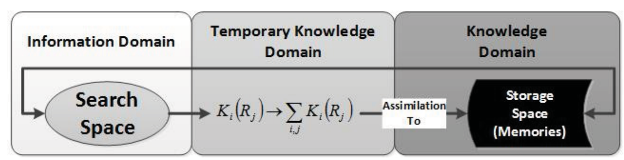


Figure 3, High-Level Information to Knowledge Flow

Figure 3, describes the high level cognitive information discovery to knowledge work flow. Figure 3, depicts the next level of detail of human cognitive workflow. The RNA common process contains five functional sub-processes, labeled Discovery, Decomposition and Reduction, Compare & Contrast, Association, and Normalization, which perform synthesis of knowledge and context within the framework depicted down the left side of Figure 3. These sub-processes operate in the process domain, as described by Suh [15]. The RNA steps in Figure 4 are as follows: Discovery encompasses the review and understanding of existing knowledge and /or in the case of disciplines, the review of a discipline's fundamentals and/or First Principles, Decomposition & Reduction decomposes the domain knowledge into "bite size" digestible bits of information and reduces the representative domain knowledge to a core capability, Compare & Contrast a cognitive examination process assimilating facts and information, comparing each to the other, looking for evolving associations, Association for establishing and assigning relationships between any two objects of information, Normalization for functionally combining commonalities into a normalized form and validating the result. Finally, recursion is depicted as the domain knowledge feedback loop, which represents the iterative recursive refinement taking the knowledge gathered during the first iteration and using it as input into the next iteration of the RNA process.

3.3 Challenges for Infusing Computational Thinking

Lessons learned from human nature tell us that change is slow and evolutionary and that changes occur most often incrementally. Those trying to implement change, learn to leverage existing processes for achieving success and to achieve buy-in and support from those within the most direct path to fundamental future change. Specifically, within education there are standards bodies, auditing boards and prerequisites at all echelons of education, which cannot be ignored. Additionally, there are also legal aspects, which are different between states, nationally and internationally, as well as, the effects of local and global demographics. As an analogy, if you have ever watched a foreign film without subtitles, at some point you get an idea of what is going on, but do you really understand the point of the movie? Hence, if a student misses a key foundational math concept or skill, keeping up in class becomes much more difficult. Students can then feel confused and frustrated, much the same way someone feels while watching the non-subtitle foreign film or simply needing to, but not understanding a language [2].

Most schools treat students the same because it is assumed that all students learn the same way and at the same rate. Teachers are expected to cover vast amounts of material in a short period of time, not to mention the teaching focus on specific country and/or state standards tests, tests to graduate from secondary education, advanced placement, or even university placement testing. If a student doesn't get a chance to master knowledge or a skill for whatever reason – that's too bad. Teachers must move quickly to stay “on schedule". As a result, we see too many students fall behind with ambiguous soft knowledge instead of clear, concise understandings.

Common Core did not fail because there wasn’t true passion to change flaws in our educations systems. Common Core failed primarily because the approach attempted to wholesale change everything without taking all of these considerations into account along with increased focus upon demographic and timeline differences, as well as, costs associated with differing states, countries and/or regions, which significantly affects adoption rates and ability to integrate new educational principles and even simple ideas. Therefore, affecting revolutionary Computational Thinking change will require a focused evolutionary approach while taking all of stated items into account.

Common Core did not fail because there wasn’t true passion to change flaws in our educations systems. Common Core failed primarily because the approach attempted to wholesale change everything without taking all of these considerations into account along with increased focus upon demographic and timeline differences, as well as, costs associated with differing states, countries and/or regions, which significantly affects adoption rates and ability to integrate new educational principles and even simple ideas. Therefore, affecting revolutionary Computational Thinking change will require a focused evolutionary approach while taking all of stated items into account.

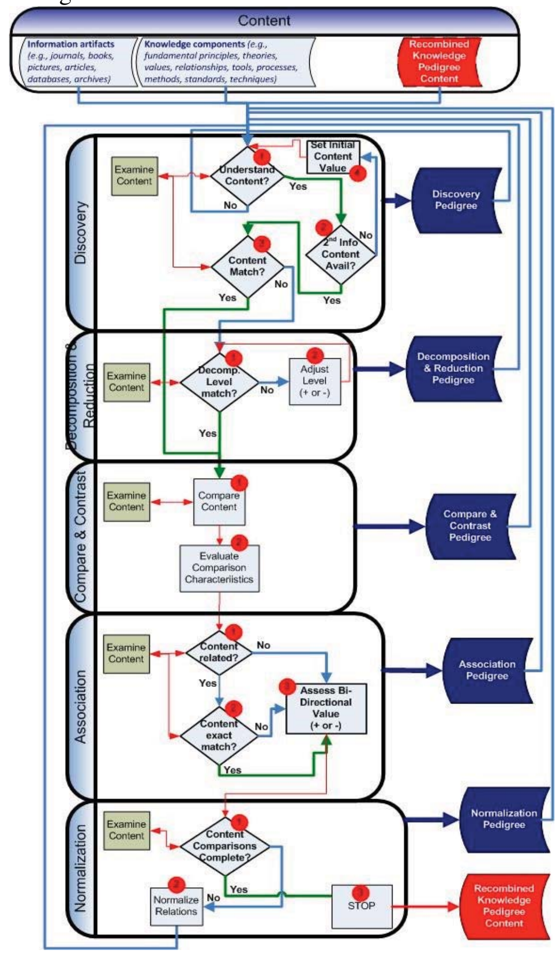


Figure 4, Recombinant kNowledge Assimilation (RNA) Cognitive Workflow

3.4 Computational Thinking Approach

The challenges expressed in the previous paragraphs may lead one to reasonably believe that generating a viable approach to integrating Computational Thinking into global education will be difficult. The reality is that integrating Computational Thinking is not difficult if the teachers and the students who know their challenges, culture and environments are the ones making changes from the inside out. The key to enhancing global education with CT is first instilling a questioning mind as opposed to simply a receiving and parroting one.

Taking into account the processes psychologists have researched for decades regarding how humans learn and the well documented benefits of how design engineers manage complexity, axiomatically, leads us to propose an approach for combining them. Thus, optimizing thought processes into curriculums qualitatively to develop CT within individuals not naturally disposed to its benefits. Teaching CT does not require massive changes in process, curriculum, laws etc. Teaching CT requires an understanding of the nuances of learning processes and how to optimize them within the current frameworks and mechanisms used for teaching. The end result has brought powerful results to the computing industry and will surely do the same of education. There are three core mechanisms required to fundamentally enhance education at each endpoint---the person/student, while standards bodies continue to require and infuse new required learning content:

1) Provide mechanisms for ***educators*** to support them making small evolutionary changes, modifying their approaches periodically to match changing students and environments, without modifying their required curriculum,
2) Provide mechanisms to involve ***students*** which they can learn on their own, which help them work directly with their teacher more effectively,
3) Provide ***administrators*** understanding so they may provide proper support for CT changes, as well as, realizing benefits and mitigation of perceived risks.

3.5 CT Approach: Environment "DO Power"

Fundamentally, CT approach takes into account what many teachers already know, but empowers them with some simple suggestions, concepts and exercises which have the potential to create lasting change. The following suggestions focus upon changing the paradigm from, "Why should I care about formulas, grammar, Roe –vs- Wade etc.?..."I will never use this information in life" to the power to "DO".

From psychology and cognition research we know that positive attitudes are beneficial for sick patients and for qualitative learning. Teachers know that the right classroom environment can make all the difference in the quality of learning. Many take great pride and ownership in creating a positive atmosphere within their realm of control whether in a jungle, a one-room schoolhouse or an urban jungle.

Hence, at the beginning of each newly formed class, each teacher should perform the "DO Power" Exercise workshop. The exercise workshop involves integration of two simple steps, 15 min each, where everyone is involved. Firstly, from Biological Systems we know that the greater the diversity the better our chance for adaptability and survival. Hence, to instill some adaptability at the beginning of the workshop and each new semester classroom the students should move and sit for diversity (e.g. gender, ethnicity, culture, etc.) The teacher should facilitate but allow the students to have some freedom as long as they are somewhat out of their comfort zone and sit with someone different.

The first ~15 min, students put themselves in the mind of a teacher and teacher like a student. Everyone together picks a scribe to document the ideas and steps for creating 3-5 ideas that they believe would create a "Do Power" class culture. Obviously, this also requires teachers to perform a detailed accounting of time constraints to cover educator selected material, curriculum options, and potential classroom teams (educators and students). Second, ~15 min the same teacher student mind swap occurs. The students and teacher develop 3-5 compelling small or large project ideas from each perspective, which can apply to the curriculum the teacher is required to teach. Choose another scribe together and have the scribe document the idea details output (e.g. what is it?, benefits to the students and to society, am I solving a problem?, what disciplines are involved…math, art, architecture, science, physics).

Finally, the teacher and students choose how to aggregate the output and the teacher then adapts the ideas into the curriculum and should work with her students to capture and document additional items they all learn over and above the required curriculum so they may see the fruits, value and application of their collaborative labors for psychological positive reinforcement.

3.6 CT Approach: The Curriculum

The previous discourse held that it was not good practice to impede standards bodies in their continuing efforts to instill necessary curriculum, unlike Common Core, and proposed some techniques to support students and educators in enhancing classroom atmosphere and qualitative learning. This segment describes current curriculum, used to allegedly embolden CT characteristics in the minds of university students and supplies mechanisms supporting development of curriculum potentially decomposed into finer grained components and potentially adapted for K-12 as well.

Some university Computational Thinking curriculum includes: number theory, probabilistic methods, algebraic structures, graphs, matching, finite automata, Turing machines, diagonalization, proof, algorithms, data structures, invariants, programming language paradigms, induction, intractability, computability, pipelining, distributed computing, operating systems, automated computation, game trees, and artificial intelligence [5]. Upon inspection it is not surprising that the course work described applies to varying types of computer science related degrees. The fact that the curriculum is identified for teaching computational thinking is yet another. These subjects provide students with objectives for greater computational understanding and specific computer skillsets [6].

However, the ability to visualize patterns rapidly, to abstract, to gain confidence in order to critically question the stated rules of physics for example, which we know can change, requires a way of learning which is not only about the math and the numbers. Creating Computational Thinkers is also about potentially rewiring and/or producing structural changes within the mind and within the way students learn. In this way they may Computational Think via application based learning and their own personal critical thought [7].

A well-known professor's young son once proudly provided him a picture of horse. After thanking him the professor slightly pressed the son to critically think and inquired, "How do you know this is a horse?" Can you imagine the questions that the professor's response generated? Another example is a standard secondary education question, "How tall is Mount Everest?" If the objective is to critically think then the answer is not necessarily simple. If students were pushed to think critically they might respond with additional queries: "Do you mean from sea level?", "What year?", "What time of year?" etc. and generate greater depth in understanding.

Therefore, positive environments, application-based curriculum, implementation of critical thinking techniques has great potential to promote denser knowledge, qualitative understanding and confidence to question by default. When also supported with appropriate curriculum and institution of continuous progress measures, students learn in-depth, they decompose the problem, improve understanding and therefore improve abstraction and recognition of patterns and therefore learn how to and improve Computational Thinking.

3.7 CT Approach: Example Curriculum and Program

The University of Alabama, Birmingham developed an unprecedented university-high school collaborative program a few years ago with inner city schools within the City of Birmingham and with students and high schools across the state of Alabama. This program used "DO Power" as part of a workshop, which included, students, teachers and faculty from local high schools as well as university engineering professors and faculty from a number of other universities. The local chapter of Girls Inc. was also well represented. Each participated to help create their now 3+ year program which has garnered significant success in its ability to instill Computational Thinking and improve scores among students involved in the program. The program was the brain-child of Dr. Murat Tanik, currently the Chair of the Electrical and Computer Engineering Department and the Wallace R. Bunn Endowed Chair for Telecommunications. Among other unique components the program university professors act as mentors for high school students from their junior year in high school as they work through projects and exercises.

4. SUMMARY AND NEXT STEPS

In summary, many see the challenges in global education today and many may see some relationships of this CT approach to mechanisms used within Science, Technology, Engineering and Math (STEM) programs, which have been in wide implementation over the years. Others may see relationships to trade schools, which are project based as well. *The key to enhancing global education with CT is first instilling a questioning mind as opposed to simply a receiving and parroting one.* The quality of learning in project based STEM is undeniable, however, many schools and teachers have lacked the multi-disciplinary skill sets and training as just as we do in industry. Some schools in Texas for example, are now moving from STEM to magnate schools to make it somewhat easier to adapt curriculum to a more coarse-grained global focus category (e.g. Math, Law magnet school). Therefore, the approach to CT described here is applied using application-based micro-projects, developed within a positively reinforced environment, from the inside-out, balanced by students, teachers (the ultimate curriculum providers) and administrators.

REFERENCES

1. Weintrop, D., Beheshti, E., Horn, M., Jona, K., Trouille, L., and Wilensky, U. 2016. Defining Computational Thinking for Mathematics and Science Classrooms. Journal of Sicence, Education, and Technology, Vol. 25 (1), pp. 127-47.
2. Voogt, J. 2015. Computational Thinking in Compulsory Education: Towards and Agenda for Research and Practice. Education and Information Technologies, Vol. 20 (4), pp. 715-728.
3. Daily, S. 2014. Embodying Computational Thinking: Initial Design of an Emerging Technological Learning Tool. Technology, Knowledge and Learning, Vol. 20 (1), pp. 79-84.
4. Eisenberg, H. 2010. Bead Games, or, Getting Started in Computational Thinking without a Computer. International Journal of Computers for Mathematical Learning. Vol. 15 (2), pp. 161-166.
5. Bagley, S. and Rabin, J. 2010. Students' use of Computational Thinking in Linear Algebra. International Journal of Research in Undergraduate Mathematics Education, pp. 1-22.
6. Mittermeir, R., Syslo, M. 2008. Informatics Education - Supporting Computational Thinking. Third International Conference on Informatics and Secondary Schools - Evolution and Perspectives, Proceedings of ISSEP, Torun, Poland.
7. Czerkawski, B., and Lymann, E. 2015. Exploring Issues about Computational Thinking in Higher Education. TechTrends, Vol. 59 (2), ppl 57-65.
8. Papert, S. 1980. Mindstorms: Children, Computers, and Powerful Ideas. Basic Books, Inc.
9. Booch, G., Rumbaugh, J., and Jacobson, I. 1998. The Unified Modeling Language Users Guide. Addison Wesley Publishing, Boston, MA, ISBN: 0-201-57168-4.
10. Ejigu, D., Scuturici, M., and Brunie, L. 2008. Hybrid Approach to Collaborative Context-Aware Service Platform for Pervasive Computing. Jounal of Computers, Vol. 3, pp. 40.
11. Gruber, T. 2008. Collective Knowledge Systems: Where the Social Web meets the Semantic Web. Web Semantics: Science, Services, and Agents on the World Wide Web, Vol. 6, pp. 4-13.
12. Van Ittersum, M., Ewert, F., Heckelei, T., Wery, J., Olsson, J., Andersen, E., Bezlepkina, I., Brouwer, F., Donatelli, M., and Flichman, G. 2008. Integrated Assessment of Agricultural Systems – A Component-Based Framework for the European Union (seamless). Agricultural Systems, Vol. 96, pp. 150-165.
13. Rowley, J. and Hartley, R. 2008. Organizing Knowledge: An Introduction to Managing Access to Information. Ashgate Publishing Company.
14. Sternberg, S. 1966. High-Speed Scanning in Human Memory. Science, Vol. 153, pp. 652-4.
15. Suh, N. 2006. Application of Axiomatic Design to Engineering Collaboration and Negotiation. Proceedings of the 4th International Conference on Axiomatic Design, Firenze, Italy.
16. Newell, A., Show, J., and Simon, H. 1957. Preliminary Description of General Problem Solving Programs-i (gps-i). WP7 Carnegie Institute of Technology, Pittsburgh, PA.
17. Simon, H. 1961. Modeling Human Mental Processes. In Papers presented at the May 9-11 Western Joint IRE-AIEE-ACM Computer Science Conference, Los Angeles, CA.

A computational approach to introduce Sumudu Transform to students

Jun Zhang
Dept. of Math and Computer Science
University of Maryland Eastern Shore
Princess Anne, MD 21853, USA
jzhang@umes.edu

Fangyang Shen
Dept. Computer Systems Technology
New York City College of Technology
City University of New York
Brooklyn, NY 11201, USA
fshen@citytech.cuny.edu

Chunlei Liu
Dept. of Math and Computer Science
Valdosta State University
Valdosta, GA 31698, USA
cliu@valdosta.edu

ABSTRACT

Computational science is not only important itself, as it is considered as the third pillar for scientific research, but also drives to advance all of sciences and technologies. It is even believed that the most scientifically important and economically promising research frontiers in this century will be conquered by those most skilled with advanced computing technologies and computational science applications. As computational science is getting more and more important in interdisciplinary studies in sciences and technologies, more academic institutions begin to offer such computational mathematics or scientific computing courses or programs to undergraduate and graduate students. Sumudu Transform is relatively new, but has many good properties for solving problems in pure and applied mathematics. The authors propose to introduce Sumudu Transform in computational approach to undergraduate or graduate students in computational related courses, such as computational mathematics and scientific computing.

General Terms

Type: Regular Research Paper

Keywords

Sumudu Transform, computation, application, education

1. INTRODUCTION

The Laplace Transform is one of the most famous integral transforms and has been well studied and widely used in mathematics and engineering for centuries. Many undergraduate and graduate programs offer courses on the Laplace Transform. The Sumudu Transform is relatively new, but it is as powerful as the Laplace Transform and has some good features ([1-5]). Sumudu transform was introduced and studied in a traditional way as other integral transforms by many researchers recently. In this paper, the authors shall introduce the Sumudu Transform through a new approach, a computational approach which can be implemented into a system to solve problems automatically, to undergraduate and graduate students in computational related courses, such as computational mathematics and scientific computing.

Assume that f is a function of x. The Sumudu Transform of f is defined as

$$F(z) = S[f(x)] = \int_0^\infty \frac{1}{z} e^{-x/z} f(x)dx. \quad (1)$$

We shall refer to $f(x)$ as the original function of $F(z)$ and $F(z)$ as the Sumudu Transform of the function $f(x)$. We also refer to $f(x)$ as the inverse Sumudu Transform of $F(z)$. The symbol S denotes the Sumudu Transform. The function $\frac{1}{z}e^{-x/z}$ is called the kernel of the transform.

The following are some basic properties of the Sumudu Transform:

(1) *Linearity*

$$S[c_1 f(x) + c_2 g(x)] = c_1 S[f(x)] + c_2 S[g(x)]. \quad (2)$$

(2) *Convolution*

$$S[(f * g)(x)] = zS[f(x)] * S[g(x)]. \quad (3)$$

(3) *Laplace-Sumudu Duality*

$$L[f(x)] = S[f(1/x)]/z, S[f(x)] = L[f(1/x)]/z. \quad (4)$$

(4) *Derivative*

$$S[f^m(x)] = S[f(x)]/z^m - f(0)/z^m - \cdots - f^{m-1}(0)/z. \quad (5)$$

More properties can be found in [1, 2, 3, 4, 5].

A very interesting fact about Sumudu Transform is that the original function and its Sumudu Transform have the same Taylor coefficients except a factor $n!$. This fact is illustrated by the following theorems:

Theorem 1.1 [6]: If

1. $f(x)$ is bounded and continuous,
2. $F(z) = S[f(x)]$, and

3. $F(z) = \sum_{n=0}^{\infty} a_n z^n$,

then

$$f(x) = \sum_{n=0}^{\infty} a_n \frac{x^n}{n!}. \quad (6)$$

Theorem 1.2 [2]: The Sumudu Transform amplifies the coefficients of the power series function,

$$f(x) = \sum_{n=0}^{\infty} a_n x^n, \quad (7)$$

by mapping $f(x)$ to the power series function,

$$S[f(z)] = \sum_{n=0}^{\infty} n! a_n z^n. \quad (8)$$

On the other hand, generating functions are a very important technique in discrete mathematics and algorithm analysis. For a given sequence a_n, there are two classical generating functions:

$$g(x) = \sum_{n=0}^{\infty} a_n x^n, \quad h(x) = \sum_{n=0}^{\infty} \frac{a_n}{n!} x^n. \quad (9)$$

$g(x)$ is called the ordinary generating function of the sequence a_n and $h(x)$ is the exponential generating function. Theorem 1.1 and Theorem 1.2 are inverse to each other and give a complete relationship about coefficients under Transform. Based on Theorem 1.1 and Theorem 1.2, there is another interesting fact about Sumudu Transform:

Proposition 1.3: The Sumudu Transform of an exponential generating function is its ordinary generating function; the inverse Sumudu Transform of a ordinary generating function is its exponential generating function.

These theorems serve as a base to calculate the general terms of Taylor series expansions.

2. APPLICATION IN COEFFICIENT CALCULATION

Based on the above discussion, the coefficients of Sumudu Transform can be calculated from the coefficients of the original function, using the following algorithm:

Algorithm 2.1

(1) Input: $f(x)$.
(2) Calculate the Sumudu Transform $S[f(x)]$.
(3) Calculate the nth coefficient $a_n = [z^n]S[f]$.
(4) Output: $a_n/n!$.

In the Maple software, this algorithm in can be implemented as the following `coefficient` procedure.

```
with(inttrans);
with(genfunc);
coefficient := proc(f, x)
  laplace(f, x, s);
  subs(s = 1/t, %)/t;
  rgf_expand(%, t, n));
  simplify((%)/n!);
  return %
end
```

This implementation uses the Laplace Transform function in the Maple software. An implementation without using Laplace Transform would be like the following:

```
with(genfunc);
coefficient := proc (f, x)
  int(f * exp(-x/t)/t, x =
   0..infinity)   assuming t > 0;
  simplify(%, radical, symbolic);
  simplify(rgf_expand(%, t, n)/n!);
  return %
end
```

Because of Maple's limitations on parameterized improper integrals, it seems that the *laplace* procedure is better implemented than the `simplify` procedure. Tests show that the first `coefficient` procedure performs better than the second in some cases.

Theoretically, if the Sumudu Transform of a function is rational, then it can be calculated by the `coefficient` procedure automatically without human interaction. In practice, because of Maples's limitations on parameterized improper integrals, the `coefficient` procedure does not always work. However, it still works for a wide range of functions, including all the functions in Table 2.1, which have rational Sumudu Transforms.

The *coefficient* procedure can be used to extract the nth coefficients for an infinite number of functions. Table 2.1 shows an example list of functions which can be directly calculated in Maple by the *coefficient*. The table is by no means complete. For instances, one can get immediate answers for: $sin(x)^{40}, cos(x)^{40}, sinh(x)^{40}, e^{ax}sinh(bx) + e^{cx}sin(dx)$, etc, that are not in the table. Here we assume a, b, c and d are constants.

3. AUTOMATIC PROVING OF IDENTITIES

For $f(x) = e^{ix} - cos(x) - isin(x)$, by applying the `coefficient` procedure to extract the nth coefficients of $f(x)$, we have: $a_n = 0$ for all $n = 0, 1, 2, 3, \ldots$. Since $f(x)$ is analytic, we prove $e^{ix} = cos(x) + isin(x)$.

This is the famous Euler's formula. Before Euler's work, mathematicians did not know the relationship between exponential functions and trigonometric functions, so these

Table 1: Test Results

(Assume n is an arbitrary but large enough positive integer.)

Table 2.1 Sample tests with Maple	
$f(x)$	$[x^n]f(x)$
a	0
x	0
x^{100}	0
$sin(ax)$	$\frac{I((-Ia)^n-(Ia)^n)}{2\Gamma(n+1)}$
$cos(bx)$	$\frac{(-Ia)^n+(Ia)^n}{2\Gamma(n+1)}$
e^{ax}	$\frac{a^n}{\Gamma(n+1)}$
$e^{bx}sin(ax)$	$\frac{-I((-(b^2+a^2)/(-b+Ia))^n-((b^2+a^2)/(b+Ia))^n)}{2\Gamma(n+1)}$
$e^{bx}cos(ax)$	$\frac{(-(b^2+a^2)/(-b+Ia))^n+((b^2+a^2)/(b+Ia))^n)}{2\Gamma(n+1)}$
$sinh(ax)$	$\frac{a^n-(-a)^n}{2\Gamma(n+1)}$
$cosh(ax)$	$\frac{a^n+(-a)^n}{2\Gamma(n+1)}$
$e^{bx}sinh(ax)$	$\frac{-(b-a)^n+(b+a)^n}{2\Gamma(n+1)}$
$e^{bx}cosh(ax)$	$\frac{(b-a)^n+(b+a)^n}{2\Gamma(n+1)}$
$sin(ax)cos(bx)$	$\frac{I(-(I(-b+a))^n+(-I(-b+a))^n-(I(b+a))^n+(-I(b+a))^n)}{4\Gamma(n+1)}$
$sin(ax)^2$	$\frac{(-2Ia)^n+(2Ia)^n}{-4\Gamma(n+1)}$
$cos(ax)^2$	$\frac{(-2Ia)^n+(2Ia)^n}{4\Gamma(n+1)}$
$sin(ax)^4$	$\frac{-4(-2Ia)^n-4(2Ia)^n+(-4Ia)^n+(4Ia)^n}{16\Gamma(n+1)}$
$cos(ax)^4$	$\frac{4(-2Ia)^n+4(2Ia)^n+(-4Ia)^n+(4Ia)^n}{16\Gamma(n+1)}$
$xsin(ax)$	$\frac{(-Ia)^n+(Ia)^n}{-2a\Gamma(n)}$
$x^2sin(ax)$	$\frac{I(-(-Ia)^n+(-Ia)^n n+(Ia)^n-(Ia)^n n)}{-2a^2\Gamma(n)}$
$xcos(ax)$	$\frac{I((-Ia)^n-(Ia)^n)}{2a\Gamma(n)}$
$x^2cos(ax)$	$\frac{-(-Ia)^n+(-Ia)^n n-(Ia)^n+(Ia)^n n}{-2a^2\Gamma(n)}$
$sin(ax)+cos(bx)$	$\frac{I(-Ia)^n-I(Ia)^n+(-Ib)^n+(Ib)^n}{2\Gamma(n+1)}$
$cosh(ax)+cos(ax)$	$\frac{a^n+(-a)^n+(-Ia)^n+(Ia)^n}{2\Gamma(n+1)}$
$sinh(ax)+sin(ax)$	$\frac{a^n-(-a)^n+I(-Ia)^n-I(Ia)^n}{2\Gamma(n+1)}$

two sets of functions were studied separately. It took a genius like Leonhard Euler to prove this formula. With the Sumudu Transform, the simple `coefficient` procedure can help us to prove this formula.

Next we consider the product of series.

Theorem 3.1: Given two power series expansions

$$g(x) = \sum_{n=0}^{\infty} b_n x^n, \qquad x \in B(0, r)$$

and

$$h(x) = \sum_{n=0}^{\infty} c_n x^n, \qquad x \in B(0, R),$$

the power series of their product $g(x)h(x)$ is given by

$$g(x)h(x) = \sum_{n=0}^{\infty} a_n x^n, \qquad x \in B(0, r) \cap B(0, R),$$

where

$$a_n = \sum_{k=0}^{n} b_k c_{n-k}. \qquad \text{for } n = 0, 1, 2, \cdots$$

This theorem is very easy to verify and can be found in many textbooks.

Applying Theorem 3.1, if $f(x) = g(x)h(x)$ and we know the n coefficients of the Sumudu Transform of $g(x)$ and $h(x)$, then we can calculate the n coefficient $f(x)$ using the following algorithm.

Algorithm 3.1

(1) Input: $g(x), h(x)$.
(2) Calculate the Sumudu Transform $S[g]$.
(3) Calculate the Sumudu Transform $S[h]$.
(4) Initialize: $c_n = 0$.
(5) For $k = 0$ to n do
  Calculate the kth coefficient $p_k = [z^k]S[g]$.
  Calculate the $n-k$th coefficient
   $q_{n-k} = [z^{n-k}]S[h]$.
  Add $p_k q_{n-k}/k!(n-k)!$ to c_n.
  End for
(6) Output: c_n.

For example, if $f(x) = sin(x)cosh(x)$, let $g(x) = sin(x)$, $h(x) = cosh(x)$, then

$$\begin{aligned} S[g] &= \frac{z}{1+z^2}, \\ S[h] &= \frac{1}{1-z^2}, \\ [z^n]S[g] &= \frac{I(-I)^n - II^n}{2}, \\ [z^n]S[h] &= \frac{1+(-1)^n}{2}, \\ [x^n]f &= \sum_{k=0}^{n} \frac{(I(-I)^k - II^k)(1+(-1)^{n-k})}{4k!(n-k)!}. \end{aligned}$$

By applying `coefficient` directly to $f(x) = sin(x)cosh(x)$, we have

$$[x^n]f = \frac{I((1-I)^n - (1+I)^n + (-1-I)^n - (-1+I)^n)}{4\Gamma(n+1)}.$$

Then we have the following identity:

$$\sum_{k=0}^{n} \frac{(I(-I)^k - II^k)(1+(-1)^{n-k})}{4k!(n-k)!}$$
$$= \frac{I((1-I)^n - (1+I)^n + (-1-I)^n - (-1+I)^n)}{4\Gamma(n+1)}.$$

4. AUTOMATIC SOLVING DIFFERENTIAL EQUATIONS

Sumudu Transform can help us develop algorithms, which can be implemented in computer algebra systems such as Maple, to solve differential equations automatically. For example, considering the following form:

$$\sum_{i=0}^{m} p_i(x) y^{(i)}(x) = \sum_{j=0}^{k} q_j(x) h_j(x) \qquad (10)$$

where $p_i(x)(i = 0, ..., m)$ and $q_j(x)(j = 0, 1, ..., k)$ are polynomials, $h_j(x)$ are non-rational functions with rational Sumudu transforms. Here m, k are nonnegative integers. Let's make some preparation before solving the equation.

Theorem 4.1: Assume f has a continuous derivative of order n in some open interval $x \in B(0, R) = \{x : |x| < R\}$ for some $R > 0$, and define $E_n(x)$ for $x \in B(0, R)$ by the equation

$$f(x) = \sum_{k=0}^{n-1} \frac{f^{(k)}(0)}{k!} x^k + E_n(x).$$

Then $E_n(x)$ is called the Lagrange Remainder of order n, given by the integral

$$E_n(x) = \frac{1}{(n-1)!} \int_0^x (x-t)^{n-1} f^n(t) dt,$$

$E_n(x)$ can be written in another form,

$$E_n(x) = \frac{f^{(n)}(c)}{(n)!}x^n.$$

for some c between 0 and x.

Then we have the following:

$$[x^n]f(x) = [x^n]E_n(x).$$

Assume f and g are functions of x, a and b are constants, d is a nonnegative integer.

Proposition 4.2 (Linear Pairs)

$$\begin{aligned}[x^n](af+bg) &= a([x^n]f)+b([x^n]g),\\ [z^n](S[af+bg]) &= a([z^n]S[f])+b([z^n]S[g]).\end{aligned}$$

The first formula of linear pairs can be obtained from Theorem 4.1 directly, and the second one from the linear property of Sumudu Transform.

Theorem 4.3: Assume that the power series $\sum_{n=0}^{\infty} a_n x^n$ converges for each $x \in B(0,R)$, then the function f defined by the equation

$$f(x) = \sum_{n=0}^{\infty} a_n x^n, \qquad x \in B(0,R),$$

has a continuous derivative $f'(x)$ for each x, $|x| < R$, given by

$$f'(x) = \sum_{n=1}^{\infty} n a_n x^{n-1}, \qquad x \in B(0,R).$$

It is important to know that the series $\sum_{n=0}^{\infty} a_n x^n$ and $\sum_{n=1}^{\infty} n a_n x^{n-1}$ have the same radius of convergence, as can be shown by applying the Root Test.

Proposition 4.4 (Derivative Pairs)

$$\begin{aligned}[x^n]f^{(d)} &= ((n+d)(n+d-1)\cdots(n+1))([x^{n+d}]f),\\ [z^n]S[f^{(d)}] &= [z^{n+d}]S[f].\end{aligned}$$

The first formula in derivative pairs can be derived by applying Theorem 4.3 repeatedly for d times. The second can be easily verified by applying the Sumudu derivative property above. We assume d is a positive integer and $n > d$ in this situation.

Suppose $p(z)$ and $g(z)$ have the following Taylor series expansions

$$\begin{aligned}p(z) &= p_0 + p_1 z + p_2 z^2 + \cdots + p_l z^l\\ g(z) &= b_0 + b_1 z + b_2 z^2 + \cdots + b_n z^n + \cdots\end{aligned}$$

where $p_l \neq 0$ and l is a non-negative integer.

Proposition 4.5: If $f(z) = p(z)g(z)$, we can compute $[z^n]f(z)$ by comparing coefficients,

$$[z^n]f(z) = \sum_{k=0}^{m} p_k b_{n-k}. \tag{11}$$

In this section, we assume that $h_j(x)$ are non-rational functions, but their Sumudu transforms $S[h_j]$ are rational. There are a lot of such functions [1]. For example: $e^x, sin(x)$, $e^x cos(x)$, and $sin^{(40)}(x)$.

Assume form (10) has a solution which can be expanded to

$$y(x) = r_0 + r_1 x + \cdots + r_n x^n + \cdots \tag{12}$$

Based on the theorems and propositions above, by comparing coefficients, we can calculate the coefficient of the general term r_n for the solution $y(x)$ as follows:

(1) Calculate $c_i = [x^n]p_i y^{(i)}$ by Proposition 4.4 and 4.5. for $i = 0, ..., m$.
(2) Add $s = \sum_{i=0}^{m} c_i$.
(3) Calculate the Sumudu transform $S[h_j(x)]$, for $j = 0, ..., k$.
(4) Calculate the $t_j = [x^n]q_j h_j$ by Theorem 1.1 and Proposition 4.5 .
(5) Add $t = \sum_{j=0}^{k} t_j$.
(6) Calculate some r_i's for small i's as the initial values by comparing coefficients.
(7) Solve the linear difference equation $s = t$,with initial values from (6). Return r_n.

Example Let's look at the following example. Assume

$$\begin{aligned}p_0(x) &= 1+x,\\ p_1(x) &= x,\\ q_0(x) &= 1+x,\\ q_1(x) &= 1+x,\\ h_0(x) &= sinh(x),\\ h_1(x) &= cosh(x),\\ p_1 y' + p_0 y &= q_1 h_1 + q_0 h_0.\end{aligned}$$

We are going to trace the algorithm with this example.

Step (1): Calculate

$$\begin{aligned}c_0 &= [x^n]p_0 y = r_n + r_{n-1},\\ c_1 &= [x^n]p_1 y' = n r_n,\end{aligned}$$

Step (2): Add

$$s = c_1 + c_0 = (n+1)r_n + r_{n-1},$$

Step (3): Calculate the Sumudu transforms:

$$S[h_0] = \frac{x}{1-x^2} = \sum_{n=0}^{\infty} \frac{1-(-1)^n}{2n!},$$

$$S[h_1] = \frac{1}{1-x^2} = \sum_{n=0}^{\infty} \frac{1+(-1)^n}{2n!}$$

Step (4): Calculate

$$t_0 = [x^n]q_0 h_0 = \frac{1-(-1)^n}{2n!} + \frac{1-(-1)^{(n-1)}}{2(n-1)!},$$

$$t_1 = [x^n]q_1 h_1 = \frac{1+(-1)^n}{2n!} + \frac{1+(-1)^{(n-1)}}{2(n-1)!}$$

Step (5): Add

$$t = t_1 + t_0 = \frac{1}{n!} + \frac{1}{(n-1)!}$$

Step (6): Calculate

$$r_0 = 1$$

Step (7): Solve the recurrence equation with initial value from step 6.

$$(n+1)r_n + r_{n-1} = \frac{1}{n!} + \frac{1}{(n-1)!}$$

$$r_n = \frac{3+2n+(-1)^n}{4(n+1)!}$$

So, we have:

$$y(x) = \sum_{n=0}^{\infty} \frac{3+2n+(-1)^n}{4(n+1)!} x^n$$

In Step (3), the Sumudu transform can be calculated from the Laplace transform, which is a standard implementation in most algebra systems, and the Laplace-Sumudu Duality. The Sumudu transforms are rational functions, so their Taylor expansions can be easily calculated by methods in [8-10]. In Step (6), the number of initial values needed is the order of the recurrence relation $s = t$. In Step (7), the linear recurrence equation can be solved with algorithms from [7]. The rest of the steps are straightforward.

Noticed that this algorithm can be implemented in computer algebra systems like Maple to solve such equations automatically.

More algorithms can be developed in similar ways to solve more differential equations.

5. CONCLUSIONS

The Sumudu Transform is related new but very powerful, and has plentiful applications. We used a few sample applications to show that Sumudu Transform is a powerful tool for computation.

Generating functions are a very useful tool to design and analyze algorithms. The ordinary generating function and the exponential generating function are the only two widely used generating functions. As the bridge between these two generating functions, the Sumudu Transform of an exponential generating function is its ordinary generating function and the inverse Sumudu Transform of an ordinary generating function is its exponential generating function.

In this paper, we provide some examples and algorithms to show that Sumudu Transform can be used to solve problems in a computational approach. The computational solutions can be implemented in computer algebra systems such as Maple to solve the problems automatically. These examples and algorithms are straightforward but serve as good teaching material for undergraduate or graduate computational courses.

6. REFERENCES

[1] Zhang, J. (2008), Sumudu Transform based coefficients calculation. Nonlinear Studies, Vol. 15, No. 4, pp. 355-372.

[2] Belgacem, F. B. M. (2006), Introducing and Analysing Deeper Sumudu Properties. Nonlinear Studies, Vol. 13, No. 1, pp. 23-41.

[3] Belgacem, F. B. M. and KARABALLI, A. A. (2006), Sumudu Transform Fundamental Properties Investigations and Applications. Journal of Applied Mathematics and Stochastic Analysis, Volume 2006, Pages 1-23.

[4] Kiligman, A and Altun, O. (2014), Some Remarks on the Fractional Sumudu Transform and Applications. Applied Mathematics and Information Sciences. 8, No. 6, 2881-2888.

[5] Zhang, J. (2007), A Sumudu based algorithm for solving differential equations. Computer Science Journal of Moldova, vol.15, no.3(45).

[6] Widder, D. V. (1971), An Introduction to TRANSEORM THEORY. Academic Press.

[7] Petkovsek, M, Wilf, H. S and Zeilberger, D. (1996), A = B. A K Peters, Ltd.

[8] Ravenscroft,R. A. (1994), Rational generating function applications in Maple V: Mathematics and Its Application, R. Lopez (ed.), Birkhaüser.

[9] Ravenscroft, R. A and Lamagna, E. A. (1989), Symbolic summation with generating functions. Proceedings of the International Symposium on Symbolic and Algebraic Computation, Association for Computing Machinery, 228-233.

[10] Salvy, B and Zimmermann, P. (1994), GFUN: A Maple package for the Manipulation of generating and holonomic functions in one variable. ACM Transactions on Mathematical Software, Vol. 20, 163-177.

Leveraging Interscholastic Competition in Computer Science Education

David Hicks and Jeong Yang
Electrical Engineering and Computer Science
Texas A&M University-Kingsville
Kingsville, TX, USA
{david.hicks, jeong.yang}@tamuk.edu

Abstract - *This paper reports on a project that focuses on teaching computer science related topics to high school students. During the first phase of the project a computer science course has been developed and taught to a group of students to help them prepare for an interscholastic computer science competition. The students' performance in the competition was tracked and has been analyzed and compared to students from previous years that were not given the computer science course. The initial results of the first phase of the project are encouraging, and have influenced the design of the second phase of the project in which the computer science course will be taught again and augmented with a tutorial system to reinforce and complement the course material. An extensive set of student performance metrics will be tracked and analyzed during the second phase to enable it to be thoroughly evaluated and reported.*

Keywords: *University Interscholastic League, UIL, Computer Science, STEM Education*

1 Introduction

The limited resources that are available for public education purposes sometimes result in important subjects not receiving the support they deserve. This can be especially true in smaller communities where schools are often forced to operate with even fewer resources. This paper reports on a project that focuses on teaching computer science related topics to high school students. It examines some preliminary results of the project along with a look at ongoing efforts. The students involved are juniors and seniors attending a public high school in a small city in South Texas. Despite the interest of the students in learning computer science related subjects, the school currently has no formal state sponsored program to teach these topics.

According to the Texas Comptroller of Public Accounts [1], among the 12 regions in Texas, the population of the South Texas region is the fastest growing in the state. At the same time, the region has a higher share of school districts ranked barely 'Academically Acceptable' than the rest of the state, in particular, in the Science, Technology, Engineering and Mathematics (STEM) category. The percentage of students who met the 'State Standard' in Math and Science is below the state average on TAKS (Texas Assessment of Knowledge and Skills) test performance. TAKS was a standardized test used in Texas to assess students' academic skills, which was replaced by the State of Texas Assessments of Academic Readiness (STAAR) test in 2014. More quality education must be provided to improve students' performance in this important area.

According to the survey conducted by the Computer Science Teachers Association (CSTA) in 2007, qualified teachers with a teaching certification play a key role to ensure that K-12 students are well educated to pursue a career in an area of Computer Science (CS), and yet many CS teachers are not properly prepared to teach that subject [2]. Most of the high schools in South Texas do not include CS courses as either elective or required courses in their curriculum, and thus the high school students in the area do not have much opportunity to receive proper CS education at the secondary school level. With the purpose of improving the situation, several suggestions were made by Yang et al [3]. One of them is the implementation of a vertical curriculum alignment between local high schools and universities [3] so that university professionals could help high schools to develop related courses and to sustain them to provide an array of benefits to students.

The project described here is being conducted by computer science faculty members at Texas A&M University Kingsville (TAMUK) along with the participation of several undergraduate computer science students. During the first phase of the project (Spring 2015), a course designed to teach computer science related topics at an appropriate level for high school students has been developed and was taught by a faculty member with the undergraduate students helping as teaching assistants. The results of the first phase of the project have influenced the structure of the second phase (Spring 2016). In particular, the computer science course is being augmented with the addition of a tutorial system designed to reinforce and complement course materials. The tutorial system will be accessible as a web site and be available for students to further study topics at their own pace.

The primary goal of the project is to teach a variety of important computer science related topics to these interested and motivated students in order to help them better prepare for eventually attending university level courses and/or entering the job market to pursue a computer science related career. An intermediate goal of the project, and an important motivating factor for the high school students, is to teach them topics and material that will help to improve their performance in a popular state organized interscholastic league computer science competition. Fortunately these goals align well with one another.

In the following section the high school computer science competition that serves as an incentive for student participation in the project is briefly described. This is followed by a description and evaluation of teaching the computer science course during the first phase of the project. The current status of the project is discussed next, including a description of the tutorial system under development during the second phase of the project. The final section briefly concludes the paper.

2 The University Interscholastic League Computer Science

The University Interscholastic League (UIL) of Texas organizes a variety of extracurricular academic contests among high school students within the state. Computer Science is one of the topics included in the Science Technology Engineering Math (STEM) category of UIL competitions [4]. The computer science contest includes both an individual component, made up of a written exam, and a team component that consists of both a written exam and a programming part. The competition takes place first at the local (invitational) level and then progresses to the district, regional, and state levels with the winners at each stage advancing to the next level.

The written exams of the competition consist of a series of 40 multiple choice questions that focus on general computer science related topics, including the evaluation of short code segments. In scoring the written exams, 6 points are awarded for every question answered correctly, no points are either awarded or subtracted for questions left unanswered, and 2 points are deducted for incorrect answers. Written exams are 45 minutes in duration.

The programming part of the competition consists of 12 programming problems of varying degrees of difficulty. Teams work together on programming solutions (in the Java programming language) and submit their answers for grading once completed. In scoring the programming problems, 60 points are awarded for every correct solution. An incorrect solution receives a 5 point deduction and contest judges provide feedback returning any incorrect solutions along with an indication of why a solution is not correct. If time allows teams are permitted to rework any incorrect solutions and submit them again for evaluation. The duration of the programming part of the competition is 2 hours.

The high school participants in this project are motivated students with an interest in computer science. Most of the students naturally also have an interest in participating in the UIL computer science competition. This provides an important incentive for students to participate in the project, as it aims to expand their knowledge of basic computer science related topics and thereby increase their competitiveness.

3 Computer Science Course Taught

The computer science course consists of material introducing and covering a variety of basic computer science related topics. The specific topics that are covered by the course closely match the material likely to be encountered by the high school students during the UIL computer science competition. (Note the course contents also closely match material covered by the AP Computer Science A courses and exams [5]). The course contents include:

- number bases and number base conversions
- basic data types and their representation
- performing basic computations in a program
- control flow of program execution
- problem solutions using iteration (looping structures)
- applying modularity in program design using functions
- file input and output operations
- representing data as arrays
- manipulating text data as strings
- understanding and using basic data structures

3.1 Methodology

The Java language is used in the course to teach programming skills and demonstrate practical coding examples. The course is structured to meet two times a week for 1.5 hours. The first weekly class meeting focuses on classroom work examining basic computer science related topics and introducing the Java programming language. At these meetings, the TAMUK faculty members teaching the course are assisted by the undergraduate students acting as teaching assistants, including working individually with high school students as necessary.

During the second weekly class meeting, the high school students get hands-on practice by completing exercises and programming assignments designed to reinforce classroom materials. On those days the undergraduate students work closely with the high school students on programming exercises designed by TAMUK faculty members. They help the high school students to design, code, test, debug, and run their exercise solutions. At the start of the course, the undergraduate students also help the high school students

Table 1. Results from District level of UIL CS competition.

		Year 2009		Year 2010		Year 2012		Year 2015	
		Score	Place (Total 17)	Score	Place (Total 13)	Score	Place (Total 10)	Score	Place (Total 15)
Individual Results	Student 1	12	6	8	8	20	5	26	5
	Student 2	0	9	6	9			10	7
	Student 3	-8	12	-4	10			4	8
	Student 4	-14	16	-8	11				
		Score	Place (Total 4)	Score	Place (Total 3)	Score	Place	Score	Place (Total 4)
Team Results	Team 1	4	3	10	3	x	x	100	2
	Advance to Region	No		No		No		Yes (As alternate)	

install, configure, and learn to use an appropriate programming environment for completing exercises.

During the first phase of the project, the weekly course meetings were held for a period of two months from the last week of January to the last week of March in 2015, prior to their District level of competition. About 70% of the major topics listed above were covered for the written portion of the competition. The general concept of each topic was first introduced and then related hands-on questions and exercises extracted from the previous year's written tests were given to students to provide an opportunity for them to further engage with the material.

For the programming portion of the test in the team part of the competition, the topics covered included installation of the NetBeans IDE (Integrated Development Environment) and the JDK (Java Development Kit), editing/compiling/running code in Java, creating input/output text files, using variables with various types of values, selection and repetition statements, and applying problem solving and logic thinking. However, due to the circumstance that the computer science course was not a regular required class, but instead an after-school activity, student attendance was not stable. Six students randomly showed up for the class meetings.

3.2 Evaluation

As described previously, a preliminary version of the computer science course was taught during phase one of the project. The top three high school students attending the computer science course that year (in 2015) entered the local UIL District level meet participating in both the written and programming parts of the computer science competition.

Table 1 summarizes their results along with the results of the previous years in which students from the high school have participated in the District level of the UIL computer science competition (without having taken the computer science course). As shown in Table 1, individual level student scores improved substantially in 2015, the year that students attended the computer science course. While in previous years some students had received negative individual level scores for having answered questions incorrectly, in 2015 all three students received positive individual level scores. The level of improvement in individual score performance is illustrated in Figure 1, which shows the sum of the top three scoring students for each year that students from the high school have participated in the competition.

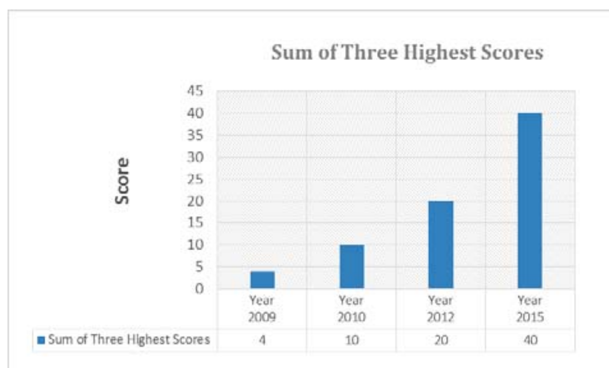


Figure 1. Sums of three highest individual scores of UIL CS competition.

An improvement in the students' team level performance in the UIL computer science competition can also be seen in Table 1. The team score for 2015 is substantially higher than that of any of the previous years in which students participated in the team competition. Figure 2 graphically illustrates the improvement in team scores. Also of note in Table 1 is that in 2015, the team placed second and became the first team from the high school to advance to the regional level of the UIL computer science competition.

The results of phase two of the project – during which the teaching of the computer science course will be supplemented through student use of the tutorial system - will be tracked more extensively. In particular, at Tuesday class meetings

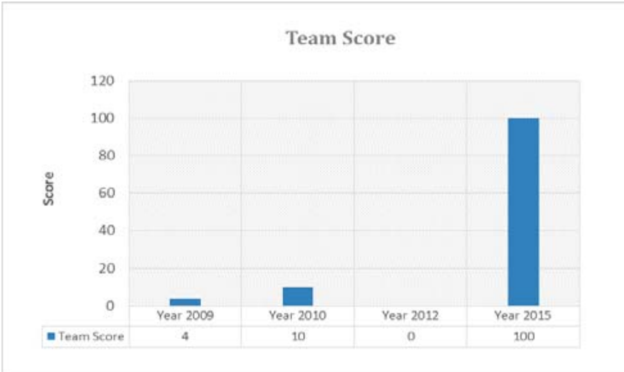


Figure 2. Team scores from District level of UIL CS competition.

students will receive exercise questions over the material covered. The ability of students to answer those questions will be scored and recorded. At Thursday class meetings the ability of students to solve programming problems will also be scored and recorded. The performance of students at UIL competitions will be recorded, including both the written exam and programming parts. A thorough analysis of all of these measurements will enable phase two of the project to be thoroughly evaluated and reported.

4 Current Status

The two educational resources that form the basis of this project are the computer science course designed to introduce and teach relevant topics at an appropriate high school level and a tutorial system being designed to reinforce the material covered during class and also support additional topic exploration by students.

4.1 UIL CS course

The UIL computer science course was taught to high school students for the first time during phase one of the project (results described above). During phase two of the project the computer science course is being taught to another group of high school students and will be supplemented by a tutorial system. The class meets twice a week, on Tuesday and Thursday afternoons, for approximately 1.5 hours, after the conclusion of the students' regular high school classes. The classes follow the structure described previously, with Tuesdays consisting of mostly classroom work focused on basic computer science topics, and Thursdays focused on programming assignments. A small group of undergraduate students from TAMUK serve as teaching assistants.

4.2 Tutorial System

The need for an additional resource that can both reinforce material that has been covered in class and help students as they work on programming assignments became evident during phase one of the project when the computer science course was taught for the first time. The resultant tutorial system is being developed during phase two of the project. It will be made accessible through a standard web browser so the high school students can easily access it and work through material at their own pace.

The tutorial system consists of two main components, a list of topics covering the knowledge of basic Java concepts and a programming capability for editing, compiling, and executing Java programs. The students will be provided a list of topics from which to make a selection. Comprehensive self-study materials will be provided for the selected topic to prepare for the written portion of the UIL CS competition. The students will also be provided with an environment to program in Java and compile and run programs to practice and prepare for the programming portion of the UIL CS competition. The tutorial system will include both written and video based materials. Once completed and deployed, the tutorial system will enable interested students to explore any of the topics covered in the computer science course – those most likely to be encountered during their computer science competition.

5 Conclusion and Future Work

The primary goal of the project described here is teaching computer science related topics to high school students. A UIL computer science course has been developed and was taught during phase one of the project. Feedback from the first phase indicated the need for an additional teaching resource to enable students to practice with and explore materials covered in class at their own pace. A tutorial system is being developed for this purpose and will supplement the computer science course during phase two of the project. It is anticipated that this combination of resources will be effective at teaching these topics to high school students. Results concerning a variety of student performance metrics will be tracked and analyzed during the course of the project and reported in future publications. A natural extension of this project will be to attempt to replicate it in other high schools, especially those whose students also participate in the same UIL computer science competition. It is also expected to make a profound improvement in educating high school students from a wide range of areas in Computer Science, therefore broadening overall participation.

References

[1] Texas Comptroller of Public Accounts (2008). Texas in Focus: South Texas. Retrieved from http://www.window.state.tx.us/specialrpt/tif/southtexas/index.html.

[2] Khoury, G. (2007). Computer science state certification requirements: CSTA Certification Committee report. Retrieved from http://csta.acm.org/ComputerScienceTeacherCertification/sub/TeachCertRept07New.pdf.

[3] Jeong Yang, Young Lee, Sung Park, Monica Wong-Ratcliff, Reza Ahanger, and Marie-Anne Mundy,

"Discovering the Needs Assessment of Certified STEM Teachers for the High-Need Schools in South Texas," Journal of STEM Education: Innovations and Research, Volume 16, Issue 4, pp. 40-45, 2015.
[4] UIL CS. Retrieved from: http://www.uiltexas.org/academics/ stem/computer-science.
[5] AP Computer Science A. Retrieved from: https://apstudent.collegeboard.org/apcourse/ap-computer-science-a/course-details.

Towards Math Integration in the Computer Science and Technology Curriculum

D. Heier, K. Lemm, M. Reed and E. Sand
Department of Computer Science and Technology
University of Jamestown, Jamestown, ND, USA
dheier@uj.edu

Abstract - *The University of Jamestown is a small liberal arts college with a Department of Computer Science and Technology (CS&T) that offers Bachelor of Arts degrees in computer science, information technology, and management information science. In the fall of 2014, we developed a curriculum modification plan and made significant changes to the mathematics requirements for these degree programs. The main objective of the changes was to replace the math course requirements with courses taught by the CS&T professors that integrate mathematical concepts with programming and computer technology concepts. Our expectation was that these changes would make mathematical concepts more relevant and valuable to students and better prepare students to apply those concepts as they progress through their degree programs and subsequently into their careers. Now that two new courses are fully developed, this paper reports on the progress*

Keywords: Math, Integration, Technology, Computer. Science. Information, Curriculum

1 Introduction

The University of Jamestown is a small liberal arts college with a Department of Computer Science and Technology (CS&T) that offers Bachelor of Arts degrees in computer science, information technology, and management information science. In the fall of 2014, we developed a curriculum modification plan and made significant changes to the mathematics requirements for these degree programs. The main objective of the changes was to replace the math course requirements with courses taught by the CS&T professors that integrate mathematical concepts with programming and computer technology concepts. Our expectation was that these changes would make mathematical concepts more relevant and valuable to students and better prepare students to apply those concepts as they progress through their degree programs and subsequently into their careers.

Now that two new courses are fully developed, the purpose of this paper is to report on the progress made in this transformation and to provide specific details of the new courses. The department has previously taught a junior level course in computer foundations (CS 330) which is described as a survey of discrete mathematical structures and its applications pertaining to the study of computer science and information technology. This course has not changed. We now have added two new courses which are CS 130: Introduction to Computer Principles and CS 230: Introduction to Algorithm and Data Analysis (1). The details of these new courses are provided in sections 3.1 and 3.2.

2 Background Research

The primary motivation for changes implemented in 2015 was based on updated ACM standards and research referenced by The Liberal Arts Computer Science (LACS) Consortium.

2.1 ACM Standards

The ACM, in the decades since the 1960s, along with leading professional and scientific computing societies, has endeavored to tailor curriculum recommendations to the rapidly changing landscape of computer technology (2). As the computing field continues to evolve, and new computing-related disciplines emerge, existing curricula are updated. At the University of Jamestown, ACM guidelines have been used to keep our programs in computer science and information technology current and relevant. The most recent ACM guidelines for computer science were approved in December of 2013. We have also researched larger schools such as Stanford University and the efforts they have taken to keep their curriculum in line with what is happening in the field (3).

The ACM guidelines suggest that there is a ‘deep and beautiful connection between mathematics and many areas of computer science” (2). The ACM notes the full set of mathematics in computer science programs varies broadly by institution and that, for example, restrictions on the number of courses that may be included in a program may lead to mathematics requirements that are “specially circumscribed to CS majors.” The ACM goes on to suggest “Students moving on to advanced coursework in specific areas of computing will likely need focused mathematical coursework relevant to those areas.” They further state that “Such coursework requirements are best left to the discretion of the individual programs and the areas of CS they choose to emphasize."

The 2013 guidelines list five example CS programs and dozens of example courses (2). One of the example programs was from Stanford University. That program contained two example courses that we felt would be appropriate for our programs. Those courses were Mathematical Foundations of Computer Science and Probability Theory for Computer Scientists. The foundations course had much in common with the foundations course we currently offer, but was positioned as an introductory course in the CS program. Further research revealed that several CS programs had introductory coursework that extended beyond the introduction to programming course. This led us to design CS 130, Introduction to Computer Principles, as a course to deliver basic computer mathematics concepts along with an introduction to the computing topics that students will be studying as they progress in their programs.

The second course adopted from Stanford was "Probability Theory for Computer Scientists." As Stanford developed this course, they determined that students found the content more relevant and valuable than a generalized course of probability theory (4). They believe the course "helps effectively prepare students for applying probability in computing contexts and using it as a tool for data analysis and modeling." Several other Universities have used this Stanford course as a model for their own courses. The Computer Science department at Villanova University believed that adopting a similar course would "allow students to appreciate the role of probability and statistics in computing" (5). This led us to design CS230, Introduction to Algorithm and Data Analysis.

2.2 LACS Consortium

The LACS Consortium is a group of computer scientists who work to model, advance, and sustain the study of computer science in liberal arts colleges throughout the nation (6). The group is actively engaged in curriculum development, scholarly research, and other projects that seek to advance high-quality undergraduate computer science education within the special setting of liberal arts institutions. In December of 2013, a few of the LACS members published a paper titled "The Roles of Mathematics in Computer Science". This paper suggests there is a gap between math's practical and intellectual roles in computer science and that required mathematics courses align poorly with the needs of computer science (7). The authors propose that the "content of required mathematics is of limited relevance to computer science as a whole. The remaining balance is under-utilized in the computer science curriculum." They state that "such a curriculum produces graduates who are ill-equipped to use mathematics in their professional careers." The authors note three solutions to the problem: a) computer science educators can be alerted to the problem, b) mathematics requirements can be used more efficiently, c) mathematics can be integrated into computer science courses. The conclusion of the paper charges faculty to reform the role of mathematics in their curricula. They encourage departments to prune mathematics courses with limited application to computer science and ensure more computer science courses use mathematics to illuminate computing concepts. The CS&T department at the University of Jamestown was very interested in improving the math skills of its students and decided the best approach would be to integrate mathematics and computer science concepts into courses offered and taught by the CS&T department.

2.3 Additional Considerations

There is another consideration that needed mentioning. Previously, students in our programs complete their math requirements at any point during their college career. For example, a computer science student may take statistics as a freshman, or during his or her last semester, as a senior. This eliminated the possibility of utilizing the statistics concepts learned from the math course in computer science coursework and providing examples and illustrations of how these concepts apply to computer science. Since the courses are offered by another department not related to the CS&T department, requirements and course sequencing was not easy to manage. At times, students were taking their math classes online from other institutions which also meant the content was not consistent for all of our students. This change intends to solve this problem by allowing the computer science instructors to build up relevant mathematics skills in our students as they progress through their majors.

For several years, the CS&T faculty have felt that the way mathematics was integrated into our majors was ineffective and were looking for a way to improve the application of mathematics concepts within our teaching lessons. The CS&T faculty expect the findings from the Sanford implementation that indicate that "students find the contextualized content of this class more relevant and valuable than general presentations of probability theory" will be replicated in our implementation of similar coursework.

3 Math Intensive Courses

The Computer Science and Technology department now teaches three classes that we consider math intensive and replace courses that would have traditionally been taught by a Mathematics department. These courses are CS130: Introduction to Computer Principles, CS 230: Algorithm and Data Analysis, and CS 330: Computer Foundations.

3.1 Introduction to Computer Principles

The Introduction to Computer Principles course was designed to present a breadth-first overview of the computer science discipline. The core of this course is the study of algorithms. Students are introduced to computing hardware and software, networks, programming languages, applications, and mathematical concepts basic to computer science. At the completion of this course, students should be able to demonstrate familiarity with concepts such as algorithmic problem solving, abstraction, pseudo code, binary numbers, Boolean logic, the Von Neumann architecture, system software, use of computers in todays' society and the social, ethical, and legal issues raised by pervasive computer technology.

3.1.1 Textbook Selection

The textbook selected for this course is titled and 'Invitation to Computer Science' and is published by Cengage (8). The publisher describes the textbook as follows:

> This flexible, non-language-specific text provides a solid foundation using an algorithm-driven approach that's ideal for your students' first course in Computer Science. Expanded chapter exercises and practice problems, feature boxes and the latest material on emerging topics, such as privacy, drones, cloud computing, and net neutrality, keep your course in touch with current issues. Optional online language modules for C++, Java, Python, C#, and Ada correspond seamlessly with this edition and give you the flexibility of incorporating a programming language to expand concepts from the text. … The optional CourseMate provides helpful study tools, such as flashcards, quizzing, and games, as well as an online Lab Manual containing 20 laboratory projects that map directly to the main text.

3.1.2 Course Outline

The following is a list of the main topics covered in this course:

- An Introduction to Computer Science.
- Algorithm Discovery and Design.
- The Efficiency of Algorithms.
- The Building Blocks: Binary Numbers, Boolean Logic, and Gates.
- Computer Systems Organization.
- An Introduction to System Software and Virtual Machines.
- Computer Networks and Cloud Computing.
- Information Security.
- Intro to High Level Languages.
- Multiple Programming Paradigms.
- Artificial Intelligence.
- Computer Graphics and Entertainment.
- Making Ethical Decisions.

3.1.3 Instructor Comments

At the completion of the course, the instructor made the following notes in an effort to make improvements to the course and increase student learning for future groups.

- Students were unfamiliar with the binary nature of a computer system. This resulted in difficulty breaking down algorithms into step-by-step pseudocode statements. Start with some 'simpler' examples.
- Students were particularly interested in the unit on computer networks, cloud computing, and information security. Include current events and expand the cloud references to include information about how their data is secured on the college network. (Possible guest speaker from our IT department).
- An individual term paper and small group presentation was assigned in conjunction with the Ethics unit. Effort was made to instruct students on using tools in Word to assist with writing the term paper and also how to rehearse and organize a presentation.

3.2 Algorithm and Data Analysis

The Algorithm and Data Analysis course was designed to introduce students to statistical concepts, so they can make educated decisions in computer and business careers. Additionally, students are taught how to interpret statistical results that computerized processes create, interpret and apply statistics in a business and managerial context, and utilize statistical software to assist business decision-making. Some of the topics covered in this course are numerical descriptive measures, basic probability, discrete probability distributions, sampling distributions, fundamentals of hypothesis testing, two-sample tests and one-way ANOVA, chi-Square tests and simple linear regression.

3.2.1 Textbook Selection

The textbook selected for this course is titled 'Business Statistics' and is published by Pearson (9). The publisher describes the textbook as follows:

> Statistics is essential for all business majors, and this text helps students see the role statistics will play in their own careers by providing examples drawn from all functional areas of business. Guided by principles set by major statistical and business science associations (ASA and DSI), plus the authors' diverse experiences, the Seventh Edition of Levine/Szabat/Stephan's Business Statistics: A First Course continues to innovate and improve the way this course is taught to all students. This brief version, created to fit the needs of a one-semester course, is part of the established Berenson/Levine series. MyStatLab provides users with countless opportunities

ISBN: 1-60132-435-9, CSREA Press ©

to practice, plus statistics-specific resources and tools that enhance students' experience and comprehension.

3.2.2 Course Outline

The following is a list of the main topics covered in this course:

- Introduction to MyStatLab and Excel.
- Defining and Collecting Data.
- Organizing and Visualizing Variables.
- Numerical Descriptive Measures.
- Basic Probability.
- Discrete Probability Distributions.
- The Normal Distribution.
- Sampling Distributions.
- Confidence Interval Estimation.
- Fundamentals of Hypothesis Testing: One-Sample Tests.
- Two-Sample Tests and One-Way ANOVA.
- Chi-Square Tests.
- Simple Linear Regression.

3.2.3 Instructor Comments

At the midway point of this course, the instructor has made the following notes in an effort to make improvements to the course and increase student learning for future groups.

- Students should have experience and comfort in using Excel as Excel is used to generate statistical values.
- Work in some small group projects and presentations earlier in the semester. An end-of-semester project is planned, but earlier projects would be beneficial.

3.3 Computer Foundations

This course provides a survey of the mathematics that pertain to the study of computer science and information technology. At the completion of this course, the student should be able to demonstrate understanding of the following objectives developed from the Core-Tier I Objectives of the ACM Curriculum (2):

- Use examples to explain the basic terminology of functions, relations, and sets.
- Perform the operations associated with sets, functions, and relations.
- Relate practical examples to the appropriate set, function, or relation model, and interpret the associated operations and terminology in context.
- Convert logical statements from informal language to propositional and predicate logic expressions.
- Describe how symbolic logic can be used to model real-life situations or applications, including those arising in computing contexts such as software analysis, program correctness, database queries, and algorithms.
- Apply formal and/or informal logical reasoning to real problems, such as predicting the behavior of software or solving problems.
- Define and identify sequences using a recursive relationship.
- Apply counting arguments, including sum and product rules, inclusion-exclusion principle and arithmetic/geometric progressions.
- Compute permutations and combinations of a set, and interpret the meaning in the context of the particular application.
- Illustrate by example the basic terminology of graph theory, as well as some of the properties and special cases of each type of graph/tree.
- Demonstrate different traversal methods for trees and graphs, including pre-, post-, and in-order traversal of trees.
- Model a variety of real-world problems in computer science using appropriate forms of graphs and trees.
- Calculate probabilities of events and expectations of random variables for elementary problems such as games of chance.
- Differentiate between dependent and independent events.

3.3.1 Textbook Selection

The textbook selected for this course is titled and 'Mathematics for Information Technology and is published by Cengage Learning (10). The publisher describes the textbook as follows:

> This text delivers easy-to-understand and balanced mathematical instruction. Each chapter begins with an application, goes on to present the material with examples, and closes with a summary of the relevant concepts and practice exercises. With numerous illustrations included, you'll be able to understand the content from a number of different angles. Whether you're majoring in electronics, computer programming, or information technology, you'll find Mathematics for Information Technology to be a valuable resource.

3.3.2 Course Outline

The following is a list of the main topics covered in this course:

- Sets.

- Logic and Proof.
- Binary and Other Number Systems.
- Straight Line Equations and Graphs.
- Solving Systems of Linear Equations Algebraically and with Matrices.
- Sequences and Series.
- Vectors.
- Probability.
- Statistics.
- Graph Theory.

3.3.3 Instructor Comments

This course has been in the computer science curriculum for many years. At University of Jamestown, it is taught in alternating years. It is scheduled to be taught in the spring of 2017. Some of the content that has been in this course, will now have been introduced in previous courses. This may affect the amount of time that is needed to cover the traditional topics and allow for additional material to be covered and applied in different ways.

4 Outcomes and Student Perception

The proposed changes of our curriculum are expected to improve the overall intended outcomes of our programs in CS, IT and MIS. Specifically, the intended outcomes for adding CS 130 and CS 230 to the programs are:

- Students will gain an introductory overview of the variety of computer science study topics in their freshman year.
- Students will be able to formulate mathematical proofs using logic well before their senior year.
- Students will be able to apply mathematical tools such as induction and recursion well before their senior year.
- Students shall be able to recall key definitions from set theory well before their senior year.
- Students will be able to distinguish between various computational models by the midpoint of their degree.
- Student will be able to think critically on the difficulties of key questions in foundations of computer science by the end of their freshman year.
- Students will gain an understanding of machine data representation and storage by the end of their freshman year.
- Students will be able to understand the combinatorial nature of problems by understanding the multiple possible outcomes from real problems and how various factors may affect an outcome by the midpoint of their programs.
- Students will gain a working knowledge of probability theory and how these theoretical fundamentals can be successfully applied in practice by the midpoint of their programs.
- Students will gain an appreciation for probabilistic statements and how understanding the difference between various statements requires an understanding of probabilistic analysis by the midpoint of their programs.
- Students will be exposed to a variety of applications where probability allows us to solve problems that might otherwise be out of reach without the tools that probability can bring to bear by the midpoint of their programs.
- Students will receive an introduction to machine learning and how this has grown to impact many applications in computing in their freshman year.
- Students will understand how machine learning focuses on analyzing large quantities of data to build models that can then be harnessed in real problems, such as filtering email, improving web search, understanding computer system performance, and predicting financial markets by the midpoint of their programs.

Assessment of the degree to which these objectives have been met will be measured by performance on unit and final exams, student discussions, and course evaluations by the students. A final exit survey is given to all graduating seniors and these outcomes will be evaluated by students in this survey. We currently do some pre- and post- testing in our classes for assessment purposes and we have some baseline data we could use to compare future results on these tests.

We are currently teaching the second class in this series. Initial assessment data will not be finalized until the class concludes. However anecdotal evidence via student feedback is currently available. All first year students who took the initial class were successful in the course and have decided to remain in the program.

Students who are in the Algorithm and Data Analysis class have given feedback about the class. They appreciate that the topics are being taught in a way that relates to their chosen major. They like the practical application that is being utilized in teaching the class.

5 Conclusion

For a number of years the faculty members in the CS&T Department have been searching for ways to encourage students to take mathematics classes early in their education so application can be made to the programming and other skills being learned in our three majors. When the new ACM standards emphasized integrating math concepts into classes taught in the computer science departments, we saw an opportunity to revise our curriculum and offer classes that are directly related to courses in computing. Since these courses will be taught by our department, we can schedule them so they can be taken concurrently with relevant computing classes. These changes were implemented during the 2015-2016 academic year and will continue to be evaluated over the next several years.

6 References

[1] University of Jamestown. (2016). 2015-2016 Academic Catalog. Retrieved from: http://www.uj.edu/academics/

[2] Association for Computing Machinery (ACM), IEEE-Computer Society - Joint Task Force on Computing Curricula. (2013). Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science. Retrieved from: http://www.acm.org/education/

[3] Mehran Sahami, Alex Aiken, and Julie Zelenski. (2010). Expanding the frontiers of computer science: designing a curriculum to reflect a diverse field. Proceedings of the 41st ACM technical symposium on Computer science education (SIGCSE '10). ACM, New York, NY, USA, 47-51. DOI=10.1145/1734263.1734279

[4] Sahami, M. (2011). A course on probability theory for computer scientists. Proceedings of SIGCSE 2011, the Forty-Second Technical Symposium on Computer Science Education (New York: ACM, 2011): 263-268

[5] Joyce, D. (2011). Probability and Statistics for Computer Science – Who Should Teach It? Proceedings of the 2011 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS2011)

[6]The Liberal Arts Computer Science (LACS) Consortium. (2014). What We Do. Retrieved from: http://lacs.myopensoftware.org/

[7] Douglas Baldwin, Henry M. Walker, and Peter B. Henderson. (2013). The roles of mathematics in computer science. ACM Inroads 4, 4 (December 2013), 74-80. DOI=10.1145/2537753.2537777

[8] Schneider, G. M., & Gersting, J. (2015). Invitation to Computer Science, 7th Edition, Course Technology, ISBN: 978-1305075771.

[9] Berenson, M. L., Stephan, D. F., Levine, D. M., & Krehbiel, T. C. (2012). Business Statistics: A First Course, 7th Edition, Pearson Higher Ed.,ISBN: 978-0321979018

[10] Basta, M., Delong, S. Basta, N., (2013), Mathematics for Information Technology, 1st Edition, Delmar Cengage Learning, ISBN: 978-1111127831

Teaching and Learning via Undergraduate Research Projects: Modeling Cosmic Flights Case Study

A. Vaninsky
Mathematics Department, City University of New York/Hostos Community College, Bronx, NY, US

Abstract - *This paper suggests that research projects should be one of the main teaching tools for STEM students from their sophomore year and above. This proposal is based on and further develops the educational part of undergraduate research. The paper presents an example of an inter-disciplinary project completed in a community college by a diverse group of students pursuing a career in Mechanical Engineering. The project was focused on the modeling the Moon – Earth flight using Maple software. It comprised the theoretical basis, deriving the differential equations of the movement, and computer modeling using Maple. The obtained results were presented at the Joint Mathematics Meeting of the American Mathematical Society*. The i*mpact on students' creativity and learning abilities is discussed.*

Keywords: STEM Education, Project-based education, Technology-based education, Cosmic flight modeling.

1 Introduction

Thanks to the Internet, students' learning is changing, and some universities adopt the challenge-driven model of education. During the course, students work on serious practical problems. Instead of attending lectures, they work on projects in teams, attempting to solve real-life problems. This is one of the new approaches among the broader changes in the contemporary higher education. Earlier such attempts are known, for example, in Experimental College, Wisconsin (1927) and Experimental University, Paris (1968). Both survived just a few years. A belief is that the Internet and technology give a new life to this attitude. See, for example, [6] for discussion. Our objective is to demonstrate how the teaching process may be organized in practice.

The role of undergraduate research was discussed in literature sources from different perspectives. Thus, Angulo and Vaninsky [2] analyzed undergraduate research in a minority - serving community college. They provided an example of a specially developed mathematics program aimed at academic preparation leading to the increase of the efficiency of the following field work at the NASA premises. This paper states that undergraduate research should become the mainstream of the undergraduate education rather than being a just complimentary tool. This way of teaching is closely related to psychology of education and educational neuroscience – the topics deserving special consideration beyond the scope of this paper, see, for example, Vaninsky [9].

The idea behind this project stems from the research on the modeling extraterrestrial urbanization, Merrifield [7]. One of the related problems is a rescue of the inhabitants in case of natural disasters or hostile actions of the other objects of different nature. A possible solution to the problem is placing a rescue satellite in a stationary orbit, where it circulates for a long time, and use it when necessary for evacuation. Literature sources reveal that such ideas were in mind since at least 1960s as the Moon - Earth projects. This paper presents a tool serving to support one of aspects of the Moon - Earth projects by modeling the trajectories of a satellite launched from a stationary Moon orbit. Theoretical basis and the project outline were borrowed from the publication of Edwards [5]. During the project, students studied theoretical material related to Physics, Mathematics, and Computer Science, learned to conduct and lead scientific discussions, practiced in compilation of a research journal, and made group and individual presentations. The final presentation was made at the Joint Mathematics Meeting of the American Mathematical Society, Vaninsky et al. [8].

This project was partially supported by the Title V grant awarded to Hostos Community College of The City University of New York.

2 Project description

The project comprises three parts: Theoretical basis, Differential equations of the motion, and Computer modeling using Maple.

2.1 Part 1. Theoretical basis

The project is based on the Second Newton's Law, the Gravitation Law, and the Third Kepler's Law. The Newton's Law substantiates the launch process, while the Gravitation Law shapes the trajectory. The Third Kepler's Law is used to compute the Universal Gravitation Constant. In this paper below we follow publication of Edwards [5] to derive a system of differential equations.

The Law of Gravitation states that two masses m_1 and m_2 are attracted to each other with the force F proportional to each of the masses and inverse proportional to the square of the distance between them:

$$\bar{F} = -\frac{Gm_1m_2}{|\bar{r}|^2}\hat{r} = -\frac{Gm_1m_2}{|\bar{r}|^3}\bar{r} \qquad (1)$$

where G=6.6726×10^{-11} Nm^2/kg^2 is the Universal Gravitational Constant. The Newton's Third Law states that the forces between the two masses are equal in magnitude but opposite in direction.

Let $<Y_1,Y_2>$ be the coordinates of a satellite in a fixed coordinate system, and $<b_1,b_2>$, the coordinates of a planet – the Earth or Moon, as appropriate. Then the gravitation force acting on the satellite is

$$\bar{F} = -\frac{Gm_1m_2}{d^3}\left(\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} - \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}\right) \qquad (2)$$

where d is the distance between the objects.

If two planets come to play – the Earth and the Moon, then, from the Second Newton's Law, it follows that the movement of the satellite located at the point $\bar{y}$ = <y1,y2>is described by a system of differential equations

$$m\bar{y}'' = -\frac{Gmm_M}{\rho^3}\left(\bar{y}-\bar{b}_1\right) - \frac{Gmm_E}{R^3}\left(\bar{y}-\bar{b}_2\right) \qquad (3)$$

where ρ and R are the distances to the Moon and Earth, respectively, and m_M and m_E are their masses, correspondingly.

To simplify this system, we use a special system of measurements as follows. A unit of mass is equal to the total of Earth and Moon masses: M_E = 5.975 × 10^{24} kg and M_M = 7.35 × 10^{22} kg, that is 6.0485×10^{24} kg. The system of differential equations is written in terms of relative mass of the Moon:

$$\mu = \frac{M_M}{M_M + M_E}, \qquad (4)$$

so that μ = 0.012298 and mass of the Earth is 1- μ = 0.987702.

The unit of length (L) equals to the distance between the centers of the Earth and Moon equal to 384,400 km or about 238,900 miles. The origin of the coordinate system was taken to be the center of mass of the Earth-Moon system. The Moon is rotating around the Earth with the period of P = 27.32166 days. A unit of time was chosen so that this value of P constitutes 2π new time units. This means that the new unit of time is P/2π = 4.3484 days = 104.3616 hours = 375,701.76 sec.

We introduce a rotating coordinate system $<X_1,X_2>$ that shares the origin with the fixed coordinate system and rotates with the same angular speed as the Moon around the Earth. The advantage of such a system is that at any moment in time, the Earth is located at the point (-μ,0) and the Moon is located at the point (1-μ,0) on the X_1 axis. This is convenient for setting up the initial conditions below.

The Kepler's Third Law states that

$$P^2 = 4\pi^2a^3/GM, \qquad (5)$$

where a is the average distance between two planets, M is their total mass, and P is the period of rotation. In the new system of units of measurement, a=M=1, and P=2π. It follows from here that in this system of measurement the Universal Gravitational Constant G = 1.

2.2. Part 2. Differential equations of the Moon - Earth satellite movement

Theoretically, this problem is a part of the three - body problem [3]. Two systems of coordinates were used in this project: fixed and rotating ones - simultaneously. The fixed coordinate system was used for the derivation of the equations of the motion, while the rotating coordinate system, for setting up initial conditions - position and velocity. It was assumed that the satellite started from the point that was farthest from the Earth and in the direction opposite to the movement of the Moon. This approach allowed us to use the relative movement of the two planets to the advantage of the satellite movement.

The coordinates of the satellite in the two systems are related via the rotation matrix. Due to the special choice of the units of time, the circular frequency equals to one that simplifies the formulas.

The system of differential equations (3) in the fixed coordinate system $<Y_1,Y_2>$ was transformed to the rotating coordinate system $<X_1,X_2>$ by using a rotation matrix

$$A = \begin{bmatrix} \cos t & -\sin t \\ \sin t & \cos t \end{bmatrix}, \qquad (6)$$

where time t is measured in the new artificial units, so that the circular frequency ω =1. The two coordinate systems are related as

$$\boldsymbol{Y} = A\boldsymbol{X}, \qquad (7)$$

so that

$$A'' = -A,\ A^{-1} = A^T,$$
$$A^{-1}A' = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \qquad (8)$$

By substitution of equation (7) into equation (3) and using relationships (8), we arrive to the system of algebraic-differential equations shown below.

$$x_1'' = x_1 + 2x_2' - \frac{M(x_1 + m\cdot\mu)}{r_1^3} - \frac{m\cdot\mu(x_1 - M)}{r_2^3},$$

$$x_2'' = x_2 + 2x_1' - \frac{Mx_2}{r_1^3} - \frac{m\cdot\mu\cdot x_2}{r_2^3}. \qquad (9)$$

We solve and investigate this system using computer algebra system Maple.

2.3 Part 3. Computer modeling using Maple

During the project, students acquired hands-on experience of programming in Maple. To familiarize them with the methodology and technique of coding in Maple, a series of one-hour workshops was organized, based on publication [1]. During the workshops, the student studied the following topics. I. Computer Algebra System Maple. Entering and evaluating numerical and symbolic expressions. Understanding the output. II. Named expressions. Functions and procedures. Palettes. III. Solving linear and non-linear algebraic equations and systems of equations in numerical and symbolic forms. IV. Understanding and graphing implicit functions. Computation partial derivatives in numerical and symbolic forms. V. Differential equations – understanding, entering in Maple, numerical and symbolic, general and particular solutions. This was the most important part of the workshops that actually required three hour-long meetings. The students studied finding a general solution of a differential equation with symbolic parameters, substituting numerical values for them, and graphing a particular solution. Also they learned to draw and analyze directional fields. VI. Linear algebra. This workshop began with defining a matrix in Maple and performing algebraic operations with matrices. Then matrix multiplication and finding an inverse matrix were considered. Computation of a determinant and checking its properties with regard to a matrix product, transposition and inversion followed. Finding eigenvectors and eigenvectors of a matrix and their geometric interpretation concluded the workshop. Each workshop comprised a brief theoretical introduction, guided and independent practice and a homework to be submitted online.

In the project, we used a statement *dsolve()* with parameter output = *listprocedure* to solve an algebraic-differential system of equations shown. Maple allowed us to obtain the numerical solution suitable for drawing graphs and further analytical operations. The "lazy computations" principle embedded into the Maple system improved the accuracy of the computations. Due to this principle, the computations are not performed immediately as they appear in the script, but are instead evaluated at the later time, when the result is actually needed. This helps to improve the performance of the computer code.

The system of differential equations shown above was appended with the following algebraic expressions:

$$R = \sqrt{(x_1 + \mu)^2 + x_2^2}\ ,$$
$$\rho = \sqrt{(x_1 - M)^2 + x_2^2}\ , \qquad (10)$$

equal to the distance from the satellite to the Earth and Moon, respectively. In these formulas μ = 0.012298 and M= 1- μ = 0.987702 are the masses of the Moon and Earth, respectively in the new units of measurement. The Maple solved the system of differential equation in rotating coordinate system and simultaneously converted the solution to the fixed system of coordinates.

The Runge-Kutta-Fehlberg method of the fifth order with variable step embedded in the Maple computer system was used to solve the system of differential equations numerically, see [4] for details. This method allows to achieve the required accuracy, while keeping the calculation time acceptable.

In the project, we assumed that the satellite was launched from a point maximally distant from the Earth and in the direction opposite to the rotation of the Moon around the Earth. This allowed us for using the rotation of the Moon to the advantage of the satellite launch.

The Maple script is shown in Fig. 1. The script developed for this project may serve as a basis for future investigations. Fig. 2 and 3 show an example of the outcome.

$$\mathrm{mu} := \frac{1000}{82314};\ M := 1 - \mathrm{mu};\ r1 := \sqrt{(x1(t) + \mathrm{mu})^2 + x2(t)^2};$$
$$r2 := \sqrt{(x1(t) - M)^2 + x2(t)^2};$$
$$soln\_1 := dsolve\Bigg(\Bigg\{\frac{d^2}{dt^2}x1(t) = x1(t) + 2\cdot\frac{d}{dt}x2(t) - \frac{M\cdot(x1(t)+\mathrm{mu})}{r1^3} - \frac{\mathrm{mu}\cdot(x1(t)-M)}{r2^3},$$
$$\frac{d^2}{dt^2}x2(t) = x2(t) - 2\cdot\frac{d}{dt}x1(t) - \frac{M\cdot x2(t)}{r1^3} - \frac{\mathrm{mu}\cdot x2(t)}{r2^3},\ y1(t)$$
$$=\cos(t)\cdot x1(t) - \sin(t)\cdot x2(t),\ y2(t) = \sin(t)\cdot x1(t) + \cos(t)\cdot x2(t),$$
$$x1(0) = \mathrm{x1\_0},\ \ x2(0) = 0,\ \mathrm{D}(x1)(0) = 0,\ \mathrm{D}(x2)(0) = Dx20\Bigg\},$$
$$numeric,\ range = 0\,..120,\ parameters = [x1\_0, Dx20],\ output$$
$$= listprocedure\Bigg);$$
$$soln\_1\left(parameters = \left[\frac{994}{1000}, -\frac{21245}{10000}\right]\right);\ with(plots):$$
$$odeplot(soln\_1, [x1(t), x2(t)], 0\,..5.5, numpoints = 1000);$$
$$odeplot(soln\_1, [y1(t), y2(t)], 0\,..5.5, numpoints = 1000);$$

Figure 1. Maple script for the Orbit 2. Height above the Moon – 673km; Velocity – 2174 m/s.

Different satellite orbits were investigated, corresponding to a variety of combinations of low-high orbits and low-high initial velocities. The obtained results demonstrated high sensitivity of the trajectories to the change in the initial values. Furthermore, the shapes of the orbits in the fixed and rotating coordinate systems turned out to be quite different. The period of rotation changed from 23.9 to 76.1 days; the higher the orbit, the greater the period. An escape velocity, that is the one that leads to leaving the Solar System, was also determined.

3. Conclusions

This paper claims that STEM students at the sophomore level and beyond should be mostly taught in the research – project style. In the case study provided in this paper, the participants studied theory of flight in the atmosphere and in the vacuum, and combined the Second Newton's Law, the Gravitation Law, and the Third Kepler's Law to derive a system of differential equations in fixed and rotating coordinate systems. They also familiarized themselves with the methods of numerical solution of such systems, studied programming in Maple and used it to find and investigate the solution.

Active involvement in research had led to higher interest in the learning process and was a factor of the increase in both learning outcomes and retention. Our experience reveals that for the teaching and learning processes should be based on using technology at the highest possible level to be efficient. Drilling and memorization should be minimized or avoided. Students should learn to keep research records, conduct discussions and be able to present their findings at a conference. The latter allows them to better understand the value of the obtained results. Diversity of the research team is a positive factor. It stimulates better cooperation and prepares the participants to the realities of real life.

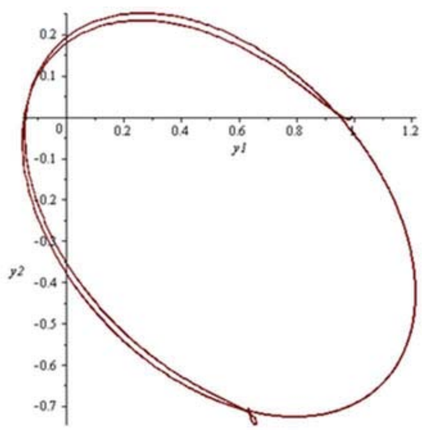


Figure 2. The trajectory of the Orbit 2 in the fixed coordinate system (view from Cosmos)

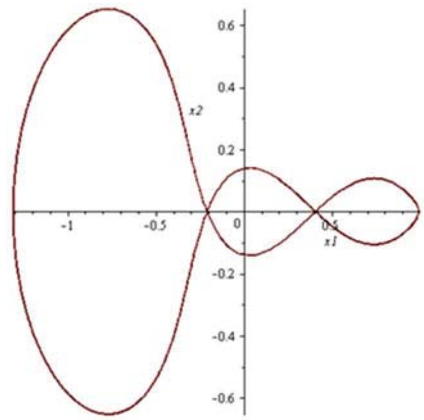


Figure 3. The trajectory of the Orbit 2 in the rotating coordinate system (view from the Earth).

4. References

[1] Abell, M., Braselton, J. "Maple by Example", 3rd Ed., Pearson. 2005.

[2] Angulo, N., Vaninsky, A. "STEM – NASA Undergraduate Research in a Minority - Serving Community College." Council for Innovative Research. Journal of Social Science Research, 2014, Vol. 4, No. 2, 542-547. Available at http://cirworld.org/journals/index.php/jssr/index.

[3] Arenstorf, R. "Periodic Solutions of the restricted three body problem representing analytic continuations of Keplerian elliptic motions". Amer. J. Math. 85 (1963) 27-35.

[4] Dormand, J. "Numerical Methods for Differential Equations". CRC Press. 1996.

[5] Edwards, C. "Periodic Moon-Earth Bus Orbits". Available at ww.math.uga.edu/~hedwards.

[6] "Flying high." The Economist, June 25, 2016, 53-54.

[7] Merrifield, A. "The Urban Question under Planetary Urbanization'. International Journal of Urban and Regional Research, Vol. 37.3, 909–922. Wiley Online Library, 2013 Available at http://onlinelibrary.wiley.com/doi/10.1111/ j.1468-2427.2012.01189.x/epdf.

[8] Vaninsky, A., De La Cruz, D., Darko, S., Garcia, J., Tambourine, C. "Bridging Mathematics, Physics, and Computer Science in an undergraduate research project 'Modeling the Earth - Moon Satellite Orbit'". Joint Mathematics Meeting of The American Mathematical Society, Seattle, WA, January 6-9, 2016. # 1116-K1-1789.

[9] Vaninsky, A. "Bridging Neuroscience and Technology for Teaching and Learning Mathematics", MathAMATYC Educator, 2013, 4(2), 4 - 8.

PERPETUAL PODS: HIGHER EDUCATION OF STEM STUDENTS – PART II

Stacey Franklin Jones, D.Sc.

Abstract

Demand for STEM workers has and will continue to increase in the foreseeable future. Moreover, the misalignment between what employers need, and skills taught and delivered, continues to be expressed as a critical problem for U.S. competitiveness.[i] *More intense team exposure during formative college years to better prepare those entering the workforce lends to this introduction of Perpetual Pods (or P-Pods™). P-Pods™ are perpetual scholarly units of students – undergraduate, graduate and pre-college; faculty, industry, government, community and alumni partners. They form, bond around, and focus on a specific interdisciplinary interest, initiative or cause. Expected yield is exploration and contribution to the greater body of knowledge; creation of a human network for discovery, research and other growth opportunities; and the establishment of a pipeline for the perpetual advancement of a specific interdisciplinary interest.*

ReCap: Majoring in STEM – Necessary, but still not Sufficient

While the statistics vary, the general consensus is that the demand for science, technology, engineering and mathematics (STEM) workers has and will continue to increase in the foreseeable future. This is good news for students who have endured the rigor of college level study to earn STEM degrees. However, the lack of alignment between what employers need, and what skills are taught and delivered, has become a critical problem for U.S. competitiveness.[ii] Project based learning has emerged as a promising educational approach to mastering skills for the workforce. In fact, there are universities that have completely abandoned their traditional lecture-based pedagogy.[iii] Also topping a list of the Five Ways to Better Prepare Students for Careers is "teamwork". These components – project based learning and teamwork - considered together yield statements such as "Employers want people who understand how to manage a project, how to deliver a product on time and on budget, and how to work in teams with little or no oversight". [iv]

Engagement in real world professional activities prior to entering the STEM workforce also increases the preparedness and competitiveness of the individual graduates. The desired outcome of such engagement activities may vary. However, in general the goal of engagement activities is to address, a priori, many of the following areas that according to Association of American Colleges and Universities (AACU) employers have identified as associated with lack of preparedness for the workforce[v]:

- Being innovative or creative
- Applying knowledge or skills to real world
- Locating, organizing and evaluating information
- Staying current on global developments

- Critical/analytical thinking
- Awareness/experience of diverse cultures both inside and outside the U.S.
- Ethical judgment and decision making
- Written communications

Internships are common avenues to acquire such experience. They have been described as " … a gateway for young students coming out of college, enabling them to learn practical skills while also applying what they have learned in the classroom" [vi]. Internships also help students match up with the right company prior to entering the job market.[vii] This is often a result of having the opportunity to try on a job before committing. But, even if the student and/or employer don't wed for future employment an equally powerful residual benefit may be the creation and/or expansion of a network.

Networking prior to leaving the university environment can contribute to preparation, landing a job that fits, and overall professional growth. An unnamed scientopia.org author states "Networking is NOT about only looking forward... Networking is more about looking around in all directions. It is important (in lots of ways) that interactions are with people at ALL stages of their career."[viii] Its value can be immediate, mid and/or long term when navigating a 35-45 year career.

What is a Perpetual Pod (or P-Pod™)*?*
Conceptually, P-Pods™ are perpetual scholarly units of students – undergraduate, graduate and pre-college; faculty, industry, government, community and alumni partners. They form, bond around, and focus on a specific interdisciplinary interest, initiative or cause to

1. Explore and contribute to the greater body of knowledge;
2. Network for discovery, research and other growth opportunities which include but are not limited to engagement and collaboration on grants, contracts, internships and other sponsored programs review and recommendation on curricula; and
3. Create a pipeline for the perpetual advancement of their respective members and the expansion of the collective pod.

In a very practical sense, a P-Pod™ is an open yet connected family-like unit that vertically explores a topic, subtopic or area of a discipline. The unit may be comprised of a unique set of faculty, students, alumni; corporate, government and/or other partners from different but interrelated disciplines. They engage in formal and informal interactions that promote high and focused interactivity around a specific area of discovery. The continuous engagement provides an atmosphere and resources to stimulate a natural and perpetual bonding process.

Formal interactions may include industry site visits, internships, seminars, speaker series, conferences, research, and work on grants or contracts. Informal interactions may take the form of campus meetings, peer-to-peer mentoring, service projects, and social networking. Any

ISBN: 1-60132-435-9, CSREA Press ©

activity that includes all or part of the unit and promotes critical and creative inquiry, leadership, and/or engagement is legitimate. The unit or pod will continue to expand with the addition of new members – whether by inclusion of a new student or cohort, or discovery of an additional connected alumnus, etc.

Putting it Together: An Example – Bloodstain Analysis Patterns (BPA) P-Pod™

A hypothetical XYZ University has distinguished undergraduate and graduate programs in biology, chemistry, mathematics, physics, computer science and engineering disciplines. Given the explosion of television series and movies that highlight and glorify the work of scientists in solving crimes, many students find the area of forensics interesting. A mathematics faculty at has dabbled in Bloodstain Analysis Patterns (BPA) defined as "the interpretation of bloodstains at a crime scene in order to recreate the actions that caused the bloodshed"[ix] primarily for presentation of mathematical problems in class. An XYZ U. mechanical engineering faculty member has done considerable work in developing instruments that produce bloodstain patterns for evaluation and comparison. An alumnus who majored in computer science works at a company that designs BPA software that performs directional analysis given varying impact patterns. Another alumnus of XYZ U. who majored in behavioral science and earned an advanced degree in chemistry, is a retired homicide investigator who now teaches chemistry at the local high school. A pod is formed under the leadership of the aforementioned faculty. Initially, they function similar to a special interest club engaged in demonstrative BPA educational activities and sessions. However, this embryotic pod encounters a challenge in BPA validation that if addressed can make a significant contribution to the reliability of a particular and/or set of BPA techniques – thus, contributing to the overall BPA body of knowledge. There is a cooperative agreement solicitation released by a federal agency with interest in development and validation of more reliable BPA techniques. Once funded, the effort can support faculty, and student assisted research at the alumnus' company. The pod's effort and its participants expand with the inclusion of their government partner and there is additional interest expressed by students in the subsequent cohort(s). Some of the undergraduates continue on to graduate studies at XYZ U. The network continues to expand and the corporate partner, now teamed with a hardware engineering company, advances its position to a competitive bid on a major contract for design of a reliability measurement instrument and applicable technique(s) that may eventually establish new global BPA standards. There is international interest giving way to study abroad. The company hires several of the pod graduates, now alumnus, to satisfy the contract. The P-Pod™ morphs and expands – naturally.

It is not difficult to see how the pod objectives of exploration and contribution

ISBN: 1-60132-435-9, CSREA Press ©

to the greater body of knowledge; networking for discovery, research and other growth opportunities which include but are not limited to engagement and collaboration on grants, contracts, internships and other sponsored programs review and recommendation on curricula; and the creation of a pipeline for the perpetual advancement of their respective members and the expansion of the collective pod, are met. Moreover, in more integrated university-industry environments such as Silicon Valley, similar activity exists both informally and informally. However, for many smaller, less strategically located, and/or economically restrained institutions a more intentional effort may be necessary. The purpose of this writing is to spur consideration of an intentional effort to form such pods at these institutions.

Higher Educating STEM College Students Continued

The first position related to the promise of "higher educating" STEM college students explored exposure to Agile. The idea was to infuse a modified Agile methodology to be incorporated at different points or phases of the undergraduate and graduate matriculation. This Part II Perpetual Pods (P-Pods™) targets development of a human resource unit or pod that naturally forms, bonds around, and focuses on a specific interdisciplinary interest, initiative or cause as outlined above. It too is boundless in its execution and presents yet another opportunity to address voids in STEM college level preparation expressed by employers. The exact implementation is also flexible and adaptable to the particular higher education environment. Moreover, while the scope of this writing is competitive advantage and workforce readiness for STEM college students; the idea of infusing the P-Pods™ into other disciplines – not just science, technology, engineering and mathematics – also holds significant promise.

CITED REFERENCES

[i] [Staff] "Prepare the American Workforce to

[ii] Ibid

[iii] Helguera, Maria and Pow, Joseph "A Project-Based Pedagogy for Technical Education Among First Year College Students." *RIT Carlson Center for Imaging Science.* Rochester Institute of Technology, 30 April 2014

[iv] Caron, Sarah. "Five Ways to Better Prepare Students for Careers." *The Education World.* Education World, 2011. Web. 30 May 2014

[v] Jaschik, Scott. "Well Prepared in Their Own Eyes" *Inside Higher Ed*, Teaching and Learning. Web. 25 January 2015

[vi] Gonzalez, Carlos. "Engineering Internships: The key to Finding a Job." *Machine Design*, Web September 2015.

[vii] Ibid

[viii] Unamed author. "The benefits of networking – thoughts on what that really means". *scientopia.org*. Portraits of the scientist of the young woman. Web. October 2015

OTHER REFERENCES

[Staff] "Introduction: A Simplified Guide To Bloodstain Pattern Analysis" *National Forensic Science Technology Center, Crime Scene Investigator Network.* Web. 2000–2016 Crime Scene Resources

[Staff] Advertisement. Hemospat BPA Software. Web https://hemospat.com/#/0 [March 2016]

ISBN: 1-60132-435-9, CSREA Press ©

Agile Process Experience for High School Students

Massood Towhidnejad, Jayson Clifford, Alexandria Spradlin

Embry-Riddle Aeronautical University, Daytona Beach, FL

Abstract - *There are a number of obstacles faced by K-12 students in the pursuit of secondary education and subsequent careers in the computing and engineering fields. This paper describes a project in which a group of ten high school students participated in an industry-strength software development project using agile processes. The project required that the students develop a semi-autonomous vehicle during a nine week summer internship with limited financial resources. The students faced a number of challenges during the internship stemming from the time constraints and lack of experience. The final product failed to meet several of the more ambitious project requirements but had the minimum required functionality.*

Keywords: K-12 STEM Education, Software Engineering, Agile Process

1 Introduction

The demand for computing professionals is increasing, and all data points to the fact that this demand will not be met given current trends. Therefore, it is important to find ways in which students are exposed to the computing field as early as possible and to encourage the pursuit of education and careers in computing fields. One obstacle for student pursuing computing education is the existence of a myth that computing fields are abstract fields and the fact that students in middle school and high school cannot draw a relationship between computing and its application to the fields in which they are interested. Through this INSPIRE-CT project, we try to encourage students to take a second look at computing and see the role of computational thinking in our everyday life.

This paper describes an activity under the INSPIRE-CT (Computational Thinking) project funded by the National Science Foundation (NSF) CPATH-2 program NSF-DUE-0939028. The INSPIRE-CT project explores vertical integration of student teams to improve student learning and to raise student interest in computing. In this experience, ten high school students were selected to participate in a software development internship program hosted by the NextGeneration Applied Research (NEAR) Lab at Embry-Riddle Aeronautical University (ERAU) (www.near.aero). The program lasted nine weeks, during which the interns met at the NEAR Lab three days a week and collaborated on a project. The project involved developing a product requested by a NEAR Lab employee playing the part of a customer. The final product would be a semi-autonomous vehicle to be controllable remotely from a ground station. The vehicle would also have a fully-autonomous mode in which it would navigate between waypoints by traveling over sidewalks. The detail description of the internship, including the selection of the participants, team organization, product requirements, and activities are discussed in [1]. This paper mainly concentrates on the development process used by the students, and some of the challenges and rewards students experienced as a result of working with a well-defined process.

2 Development Environment

A goal of the project was to allow the high school students to work in a professional environment and to experience working as part of a team in a software development organization. To this end, we hosted ten high school students at ERAU's NEAR Lab. The NEAR Lab serves as the technical arm of the university, providing support to the Federal Aviation Administration (FAA)'s Florida NextGen Testbed facility located at the Daytona Beach International Airport.

The NEAR Lab has eleven full-time staff two of which hold a Ph.D. degree, eight hold a master's degree, and one holds a bachelor's degree all in computing and engineering fields. The staff have experience in software development ranging between three to over twenty-five years. The NEAR Lab staff is almost fully funded by external grants and contracts from government and industry. The projects are funded by organizations such as the NSF, FAA, National Oceanic and Atmospheric Administration (NOAA), Boeing Corporations, New York Port Authority, etc. In addition to its full time staff, there are a number of university faculty who are also working part-time on supporting the projects. Throughout the year, between 5 and 15 graduate and undergraduate students are also hired by the lab to work on various projects.

The NEAR Lab occupies the majority of the first floor of ERAU's simulation building. It is divided into three components: the office suites, the demonstration suite, and the development and integration area. In the demonstration suite, the NEAR lab represents a microcosm of the United States National Air Space (NAS), where we develop products that simulate gate-to-gate air travel in NAS. The products in the demonstration area include the airline dispatch, remote air traffic control tower display, en-route air traffic control center, Boeing 737 plane simulator, etc. It was important to allow the high school students to work in the NEAR Lab so that they could appreciate the complexity of the products and systems (and systems of system). The complexity of the systems and the reason for defining and following a process were highlighted at the beginning of the project. The high school students were more interested in rolling up their sleeves to begin solving the problem (writing code, and building hardware). In another world, they were anxious to start hacking a solution rather than identifying the real requirements, coming up with a design, and then implementing a solution. Needless to say, they were in the most part against documenting their work.

3 Participants and Team Organization

During the ERAU fall 2014 semester, thirty students from a pool of over seventy applicants were selected to participate in a one day a week job shadowing program. The students continued to job shadow during the spring 2015 semester, and ten students were recommended by their NEAR Lab mentor to participate in a nine week summer internship program at the lab. The following represent the demographics of the ten interns:

- The interns' ages ranged from fifteen to seventeen, and their grades ranged from tenth to twelfth. Nine of the interns were male.
- With the exception of one intern, they had all taken an advanced placement computer programming class prior to participating in the internship. The majority of the participants had also taken advanced mathematics courses and some had completed the equivalent of Calculus I.
- The majority of the interns attended the same high school. Consequently, most of the participants had experience working with one another.
- Most of the students did not have experience working on a team project, and the ones with experience had worked on small teams (2-3 people).

Given the nature of the project, the development team required expertise in both hardware and software. There were a number of students with a higher level of interest in one of these two areas, and the remaining students did not have a strong preference for either. This allowed us to divide the interns into two groups of five. The two groups remained fluid throughout development, but the participants tended to remain with those they had grouped with on the first day. During the first iteration, the two groups worked in the same room, making communication amongst them efficient. At the start of the second iteration, the groups were separated. There were two reasons behind this decision:

- We wanted to simulate a professional working environment, where projects are distributed across the nation and/or internationally.
- Due to the fact that the majority of the hardware related equipment was located in the lab and the software team did not need any access to the lab equipment other than during integration, we moved the software group to a classroom (located on the second floor of the building).

When the two groups were separated, their means of communication changed. They decided to use an instant messenger to communicate throughout the workday but often found it difficult to understand one another. To remedy this problem, the two groups set designated times for video chat meetings before lunch and before each workday ended. After encountering some difficulties with this method, a leader was elected in each group, and the leaders were allowed to meet in person at the end of each workday to discuss the progress made that-day and plans for the following day. The forced separation also introduced technical difficulties. Although the requirements for the project were identified and documented, some of the requirements were interpreted differently by the two groups, and when the requirement was critical for integration, the teams had a hard time integrating all the components of the system at the end of the second iteration. All of these challenges these pointed to the importance of documentation and the identification of a clear communication path. At the beginning of the third iteration, we co-located the two teams again.

4 Process overview

Over the years, a number of agile methodologies have been used within the software engineering community, such as Crystal Clear [2], Extreme Programming [3], and Scrum [3]. The Agile Manifesto emphasizes frequent and rapid delivery of working prototypes of the system to the customer that are iteratively refined and developed. The team defines the specifics of its own design process based on the guidance of the agile methodology selected with the anticipation that it will be revised periodically based on the team's experience on the project. Since the team designates the process (with some input from the team's manager), the overhead required can be significantly reduced.

As emphasized by Cockburn [4], the different agile methodologies only set guidelines on the process carried out by the team, and it is up to the team to adapt to what best suits their organization. Over the past ten years, we have used an agile process called Crystal Clear process methodology as part of ERAU's senior design classes. Since 2010, we adapted this process as part of our software development process at the NEAR Lab. The simplest, least overhead process is called Crystal Clear.

Crystal Clear is designed specifically to work with small or medium sized teams. Some of the properties of this process include: frequent delivery via two week iterations, reflective improvement via reflection workshops at the end of each iteration, osmotic communication by co-locating teams and utilizing charts and boards to share information, personal safety, focus through a flexible plan and fixed deliverables per iteration, and a technical environment capable of supporting automated testing, configuration management, and frequent integration of team software. The following is a high-level description of a modified Crystal Clear process that was used by the high school students.

The two primary goals of the modified process used are to deliver usable/tested software and to teach students how to work in teams on highly challenging projects. This process aims to accomplish these goals by instilling a sense of personal safety and accountability, breaking the project into small, easily implemented parts, and being highly adaptive. To facilitate, there are several methodologies used, including stand-up meetings, blitz planning, iterative development, walking skeleton, early victory, and reflection workshop.

Stand-up Meetings are common across the majority of agile processes encountered. Before work begins on each workday, a stand-up meeting is held between the members of each individual team leader, followed by a stand-up meeting between the team leaders and, if present, the faculty mentors. During a stand-up meeting, the participants should actually stand. This allows them to focus on the meeting away from any laptops and to keep the meeting brief. A stand-up meeting typically lasts five minutes.

Blitz Planning (Project Planning) is based off of the XP planning game. Students write out features to be delivered from the final product on a note card. On the front side of the note card, they write out the task, to whom it is assigned, and the number of iterations required to complete the task. On the back, a detailed description of the task is provided. The note cards are then organized chronologically by the team. Next, the team builds a project timeline and identifies issues along the critical path that dictate certain task dependencies. The result is a set of high-level tasks that must be completed during each iteration.

Iterative Development forces the team to break down the project into many small chunks of time with well-defined deliverables called time boxes. These time boxes, or iterations, serve a multitude of purposes. They allow the teams to be highly adaptive by decreasing the length of feedback loops between customer and developer. In addition, these iterations allow the team to recognize the project progress and provides an opportunity to celebrate the successful completion of the iteration (see early victory below). For this project, the iteration was defined as a two week time box. The two week duration gave students enough time to fully realize new features and to integrate them into the walking skeleton but was a short enough time scale that they could remain constantly busy and focused.

Walking Skeleton and Early Victory is an essential element of Cockburn's Crystal Clear process. A walking skeleton is a functional system of hardware and software representing the work to-date. With each iteration, the team adds more features to the skeleton, fleshing it out, toward the final whole. Due to the immaturity of the students, it was very important to maintain the integrity of the iteration goals and to make sure that students who had completed their portion of the work did not start on deliverables for future iterations until they made sure that the members who were still working on the current iteration's deliveries were able to complete their work and if not, help them catch up. Lack of following the walking skeleton concepts may result in "trailing limbs"; i.e. features that are implemented but not attached. When features are not integrated, they cannot be adequately tested, and there is the risk that the design may change such that the features must later be discarded.

Reflection Workshop is another part of Crystal process, which is also included in some form in the majority of agile methodologies. It provided the opportunity for the students to address the issues that exist within the process and/or the technical environment. Students identified what problems they encountered during the current iteration, identified possible solutions to these problems, and indicated which solutions from previous iterations have worked that they wish to continue. The reflection workshop occurs at the end of every iterations and lasts no more than thirty minutes. The results of the workshop should be recorded and posted so that all can see and reflect upon the lessons learned. Figure 1 displays the result of the reflection workshop after two iterations.

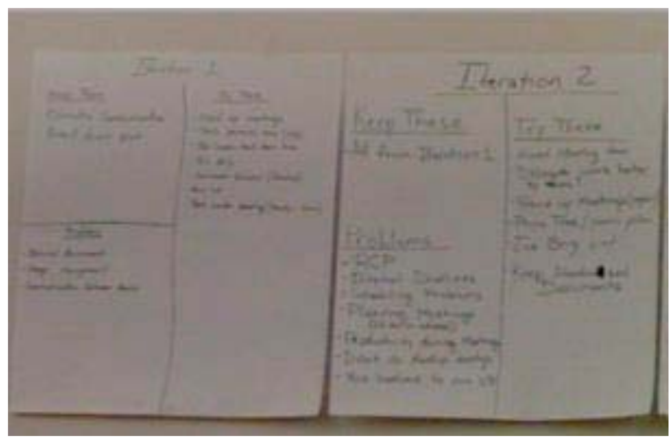


Figure 1: Reflection workshop posters from two iterations.

5 Project Requirements

The simulated project customer was a member of the university who was looking for a semi-autonomous vehicle that could be driven either remotely or autonomously as desired. While in the autonomous mode, the vehicle was required to navigate the campus walkways, while avoiding any obstacles it encountered. The vehicle was required to transmit high definition video to the ground station at all times. While operating in the semi-autonomous mode, the vehicle was required to be controlled remotely through the ground station, using a joy stick. Finally, the vehicle needed to carry a payload of at least 3 lbs, in order to be able to carry the hardware necessary for safe operation, such as a light source and a source of sound to warn pedestrians of its presence.

The ground station, would allow the user to remotely drive the vehicle. Additionally, the user would be able to set waypoints on a graphical user interface (GUI) and to switch the vehicle between the fully-autonomous and semi-autonomous mode. The video feed would be viewable from the ground station at 1080p with a minimum of 30 fps. The location of the vehicle, the charge on the battery, and the vehicle's movement speed would be observable from the ground station.

In order to simulate real-world development scenarios, vague budgetary constraints were placed on the interns, encouraging them to utilize resources that were readily available at the NEAR Lab. This idea was also enforced by the time constraints.

The project requirements and scope were intentionally beyond the interns' capabilities given the project's time constraints and budget. The reason behind this was to allow the interns to evaluate their own capabilities, and by following the process (blitz planning), to renegotiate the project requirements and de-scope the project to a more manageable scope.

6 Final Products

A number of the final requirements were never implemented due to the time and budgetary constraints. However, the finished vehicle was still drivable remotely both over the Internet and with an RC controller. The finished vehicle was built primarily with parts for RC cars. The students also used modeling software and the Taz 5 3D printer available at the NEAR Lab to create various components, including struts for the vehicle's chassis.

The vehicle was equipped with a Ubiquiti AIRCAM mounted to the front strut, a Ubiquiti Rocket M2 to provide Internet access, and a Raspberry Pi B+ that responded to commands from the ground station broadcast over Crossbar. A programmable electronic speed controller (ESC) compatible with the brushless motors that accompanied the chassis was purchased. The programmable ESC was programmatically limited in its voltage output so that the vehicle's maximum speed would remain well under 10 mph. The vehicle's speed was further limited by the choice of battery. A 3S 12.4 V LiPo battery was used. The vehicle was not equipped with sensors for obstacle detection, so the assisted-manual mode was never implemented. The vehicle was never tested for IP64 compliance (in the initial requirements), but water-sensitive components were placed in a 3D printed plastic box onboard the vehicle.

Software was successfully written for all of the hardware features that were implemented. Stub code was also written for the hardware features that were cut due to time constraints (GPS, infrared sensors, etc.). The software developed for this project was split into two major functional subgroups: the software present on the vehicle's Raspberry Pi B+ and the software used on the ground station. The vehicle code was written in Python and consisted of a primary script that performed the communications and vehicle control, a script to handle network changes that would determine the IP address of the Ubiquiti Rocket M2, and a script to automatically run the Python program when the Raspberry Pi B+ booted. The ground station code consisted of a website written using HTML and JavaScript and a Python script to take input from a USB joystick and communicate it to the vehicle over Crossbar.io, a WAMP router.

Early in the project's development, it became clear that it would be necessary to use some form of document repository and version control system. Due to the open, academic nature of the project, using a public repository on GitHub fit this need. The repository can be found at [5]. All of the previously discussed code artifacts and documentation generated during the development of the system requirements and design can be viewed. The interns were not familiar with version control prior to the internship, and using GitHub proved to be a learning process.

7 Conclusion

This internship provided the high school students with an opportunity to work in a software development industry setting under the mentorship of professional software developers. As part of the project, students were exposed to an agile development process that they were required to follow throughout the project. The agile process allowed the students to experience more aspects of the design process and project management than more traditional development processes such as the waterfall model or spiral models. There were a number of advantages to following the crystal clear process:

- The blitz planning forced the students to decompose the project into easily understandable functional requirements. This is one of the most important elements of engineering solutions and is unfortunately not practiced enough during K-12 education.
- The process forced the students to learn about project planning and about identifying a project's critical path. Again, this is not knowledge that typical a high school student gains prior to post-secondary education.
- Having a walking skeleton at the completion of each iteration forced the students to decompose the project into a number of smaller deliverables. It also allowed the students to taste success more frequently. Even though the final product did not deliver all of the project requirements, students still felt they were successful as they had satisfied a number of the functional requirements.

At the completion of the internship, students were asked to complete a survey. The survey points to the following facts:

- Almost all students believed that the internship experience was very good, and they had gained technical and professional knowledge that they did not believe was available to them through their high school educations.
- They thought that the interaction with the customer was very good, but they complained about the changes

to the requirement by the customer (this was actually designed into the experience by the PI).

- They appreciated the fact that they were required to follow the process, and the walking skeleton was a good feature of the process used to assess the team's progress throughout the project.
- They now have a better understanding of what it takes to work on a complex project and be part of a large team.
- They now have a better understanding of what potential career opportunities exist if they purse a computing degree.
- Most of the students thought that they underestimated the complexity of the project, but as they proceed through the life cycle, they better realized what they could and what they could not deliver.

Overall the students were very satisfied with the internship experience and most asked if they could participate in the same program during the following summer.

8 References

[1] M. Towhidnejad, A. Spradlin, T. Bassa, “Software Industry Experience for High School Students”, Proceedings of ASEE 2016 conference.

[2] A. Cockburn, "Crystal Clear a Human Powered Methodology for Small Teams," Reading, MA: Addison-Wesley, 2004.

[3] A. Cockburn, “Agile Software Development: The Cooperative Game“(2nd Edition). 2nd ed. Boston, MA: Addison-Wesley Professional, 2006

[4] J. Highsmith and A. Cockburn, "Agile Software Development: The Business of Innovation," Computer, vol. 34, pp. 120-122, 2001.

[5] T. Bassa, "near-netcar," Internet: https://github.com/ThomasBassa/near-netcar, 2015.

SESSION

CAPSTONE DESIGN PROJECTS

Chair(s)

TBA

ISBN: 1-60132-435-9, CSREA Press ©

ISBN: 1-60132-435-9, CSREA Press ©

E-Pass: Implementation of an NFC-Based Electronic Pass System

Erwin Ramon Alejo, Jose Miguel Sevilla, Michael Jordan Apuya, Dominic Francis Villar, Ken Ferens, Marcia Friesen
Dept. of Electrical and Computer Engineering, University of Manitoba, Winnipeg, MB, Canada
Ken.Ferens@umanitoba.ca

Abstract— This paper reports on a capstone design project that was done in partial fulfilment for the degree of Bachelors of Science in Electrical and Computer Engineering (ECE) at the University of Manitoba. The capstone design project program in ECE allows students the opportunity to demonstrate the skills they have attained throughout their studies by designing, implementing and testing a solution to a medium complexity problem. Their demonstration of the skills is a required component in the accreditation of the program, as per the requirement of the Canadian Engineering Accreditation Board (CEAB), which requires that students demonstrate the specified 12 graduate attributes at the time of their graduation. The purpose of the paper is to allow other universities and other interested parties across the World to learn about the programs the Department of ECE is doing to satisfy their accreditation requirements. Specifically, this design projects demonstrates the attributes of Knowledge, Design, Problem Analysis, Investigation, Teamwork, Communication, and Impact on Society.

Keywords— Capstone design project, accreditation, CEAB graduate attributes, NFC communications, NFC scanner, Android HCE, electronic pass card, Raspberry Pi, kiosk, data networking.

1. Introduction

The Department of Electrical and Computer Engineering (ECE) at the University of Manitoba offers a “capstone” design course, ECE 4600 Group Design Project, which is normally undertaken by students in the final year of their ECE program. The project work constitutes a significant design experience based on the knowledge and skills acquired throughout their undergraduate program and gives students an exposure to the concepts of engineering design in a team environment. Students are required to demonstrate, within a fixed time period, the ability to organize, conduct and report on an original study of an electrical/computer-engineering project. The requirements consists of five parts: a project proposal, the engineering log-book, written and oral progress reports, a formal engineering report, and an oral presentation. The authors in [21] provide a 1997 review of literature on teaching engineering design through project-oriented capstone courses. The authors in [22] provide a review of North American current practices in final year engineering design courses.

The following details one of the projects done by undergraduate students at the Department of ECE, U of M, 2016. It aims to improve a traditional fare collection system through an implementation based on NFC technology. The system was developed in the context of a bus transit service. Four primary subsystems were developed: a device scanner, a customer kiosk, an Android NFC HCE application, and a common database server for customer information. The system allows customers to use an E-Pass, either a reloadable NFC card or an Android smartphone to act as the transit fare.

The scanner subsystem consists of a Raspberry Pi with an NXP OM5577 NFC kit. It is designed to scan devices on buses. Another version of the scanner is designed to work with the kiosk subsystem, which also consists of a specialized scanner with a touchscreen display. The kiosk is where new cards are initialized and balance can be added to a user’s card. An Android application was developed to allow users to transfer their card information to their smartphone and use it as the E-Pass using HCE. All three subsystems communicate as clients over the internet with a database server to keep track of customer information.

The developed system successfully allows users to initialize their cards and switch between using their card or smartphone as an E-Pass. Both cards and smartphones can be verified with the scanner. The database can keep track of customer transactions and can be accessed through a website. Payload information on both the card and smartphone is encrypted and database caching was introduced on the scanner and kiosk to account for network connectivity issues. The created system leaves room for improvements that were outside the scope and timeline of this project, such as total data encryption between all subsystems, testing communication between several clients and the database on a large scale, and UI and visual improvements.

Near-field communication (NFC)-based connectivity is slowly becoming popular in applications that require pairing.

NFC is a technology that is based on the principle of electromagnetic induction. NFC capable devices exchange data by NFC through radio frequencies. The radio frequency emitted by one NFC device that causes the passive induction within the other NFC device to transmit data. This effectively allows two NFC devices to exchange data by simply being within proximity to each other.

Since NFC has been gradually gaining attention as of late, most Android phones right now in the market embed NFC chips to support NFC-based applications. Host card emulation (HCE) is an application made by Google that makes use of NFC of the phone that allows the data to route directly to the host device.

This project implements an NFC-based transit fare collection system using E-Passes. E-Passes can either be an NFC card or a smartphone running an HCE application. The E-Pass system consists of four major subsystems: the device scanner, the customer kiosk, the Android HCE app and the customer database server. The NFC card used is the Mifare Ultralight EV1. The HCE app turns a smartphone into a complimentary device that works interchangeably with the card.

The device scanner subsystem is placed inside a bus and is responsible for scanning both the NFC card and the HCE app. It is also responsible for writing back the updated balance and time stamp on the NFC card. Aside from being integrated to the NFC card and the HCE app, this subsystem also connects to the database to update balances and request needed data.

The customer kiosk subsystem will be placed in retailers where the NFC card will be sold and activated. It interfaces a touchscreen user interface with another version of the scanner to work with NFC cards. This subsystem is responsible for activating a new bus pass by connecting to the database where it initializes a new customer. It can also add balance to a customer's card.

The HCE app subsystem is an application on an Android smartphone and is responsible for emulating the NFC card. The HCE app allows the user to switch between having their NFC card or their smartphone as the currently activated E-Pass. It also can add balance to the user's associated account by updating the database. Additionally, the HCE app allows the user to check for the most updated balance and timestamp by requesting information from the database.

In today's day and age, the vast majority of the world's population have smartphones, most of which users heavily rely upon for day-to-day tasks. The core functionalities of a smartphone are so innovative to the extent that it can already replace one's personal computer or laptop due to the sophistication and advancement of technologies it contains. In fact, mobile payment is one of the many useful applications that is currently in smartphones right now. It provides easy payment transaction between consumers and merchandisers. The use of mobile payment constitutes the use of NFC to allow two NFC-capable devices to exchange data by tapping.

The main idea of implementing an electronic pass system into transit systems like Winnipeg Transit, is to provide commuters with the convenience of having their smartphone and a reloadable NFC card act as bus passes. Commuters do not have to deal with the problem of lining up to bus pass retailers monthly to get new passes on time. Also, using this type of system lessens the need for mass production of bus passes since commuters will use one reloadable NFC card that will only be replaced if lost or stolen. Adapting to this type of system also prevents commuters from dealing with the annual price hike of bus passes.

Aside from the main idea of using NFC and smartphones, this project also allows the practical application of knowledge gained from the University of Manitoba Electrical and Computer Engineering course curriculum. This project incorporates the use of several programming languages, client-server software architecture, TCP/IP communication, Android application development, GUI creation, and database management.

2. Related Work

Near Field Communication (NFC) is not a newcomer to the technology scene, yet is a relative newcomer with respect to integration with smartphones. Smartphone consumer applications based on proximity communication, such as electronic flight boarding passes or electronic transit fares, are the primary drivers for NFC. Specifications ISO/IEC 14443 and ISO/IEC 18000-3 are the major specifications for NFC, ensuring downstream compatibility for existing and emerging NFC applications [18]. ISO/IEC 14443 defines the technology which ID cards use to store information, such as the NFC tags found in books, clothing, etc. ISO/IEC 18000-3 specifies the Radio Frequency IDentification (RFID) communication and protocols used by NFC devices. Like traditional passive RFID tags, only one of the two NFC devices needs to be powered for tag interrogation.

As NFC is a subset of RFID, it operates in the unlicensed 13.56 MHz band and performs many of the same functions as RFID tags and contactless smartcards. The main mode of communication is through coils which to allow electromagnetic induction interactions between the NFC devices. Once powered, the tag effectively shunts its coil based on the data it wants to transmit, thereby inducing an impedance change seen by the interrogating device. NFC operates in one of three communication modes: Read/Write, Peer-to-Peer, and Card Emulation [19].

In Read/Write mode, an NFC reader/writer reads data from NFC smart objects and acts upon that information. With an NFC-enabled smartphone, for example, users can automatically connect to websites via a retrieved URL, send short message service (SMS) texts, obtain coupons, etc.

In Peer-to-Peer mode, any NFC-enabled reader/writer can communicate to another NFC reader/writer to exchange data with the same advantages of safety, security, intuitiveness, and simplicity inherent in Read/Write mode. In this mode, one of the reader/writers behaves as a tag, during the creation and operation of a communication link. For example, two smartphones with readers/writers can communicate with each other.

An NFC device in card emulation mode can replace a contactless or proximity smartcard, enabling NFC devices to be used within the existing contactless card infrastructure for operations such as ticketing, access control, transit, tollgates, and contactless payments. These types of applications extend the use of the smartphone and leverage its underlying technology [20].

3. System Components

Scanner

The scanner subsystem is responsible for scanning customer E-Pass devices. An E-Pass device is either an NFC card or a phone running the HCE app. In a real-world implementation, both transit vehicles and kiosks would contain scanner systems. The subsystem consists of its software, which runs in bus mode or kiosk mode, and its hardware, a Raspberry Pi with an NFC controller and indicator LEDs. In general, the scanner waits for devices to be scanned and performs required actions when a device is detected. In bus mode, the scanner verifies E-Pass devices and rewrites NFC cards with a new payload as necessary. The scanner then communicates with the database over an internet connection for database updates and caching. In kiosk mode, the scanner relies on commands from the kiosk CUI to determine its actions when a device is detected, such as blanking, initializing, or verifying a card.

HCE App

The HCE app on the Android smartphone serves as another electronic pass that can be used interchangeably with the Mifare Ultralight EV1 NFC card. This allows and gives commuters the flexibility to switch and use their smartphones as their daily bus pass. This is desirable to most commuters because it prevenst commuters from losing or forgetting their bus passes. The HCE app is not limited to being a bus pass on its own because it offers the ability to switch back to the NFC card if the commuter feels like so. Moreover, the HCE app provides real-time balance checking and time stamp verification to let the users know if their transfers are already expired. Also, the app allows the users to add balance to their account alleviating the hassle of lining up to activation kiosks. All these features of the HCE app are a click-of-a-button away.

Database and Servers

In this project, we used three servers: a database server, an application server and a web server. The database server stores all the necessary information about each customer while the application server interfaces between the clients and the database server. The web server allows customers to check the current contents of their account and check their transaction history. The database server we used in this project was a MySQL server, the application server was a custom Java Socket server and the web server was developed using node.js and expressjs.

Hardware

The E-Pass system makes use of several hardware components (Fig. 1). The scanner consists of a Raspberry Pi, an NFC controller, and light-emitting diodes (LEDs). The kiosk also uses a Raspberry Pi with an NFC controller in addition to a touchscreen. NFC cards and Android smartphones are both used as E-Passes. Each component works together with the developed software to provide the needed functionality for the system.

Fig. 1 E-Pass system hardware.

Raspberry Pi SBC

The Raspberry Pi was selected to run the scanner and kiosk software for this project. The Raspberry Pi was selected because of its small size, Linux-based operating system, and compatibility with the NFC development kit chosen, the OM5577. A Raspberry Pi 1 Model B+ with an 8 GB micro SD card and a Raspberry Pi2 Model B with a 16 GB micro SD card were used for development, testing, and demonstration.

NFC Controller SBC Kit

The OM5577 is a development kit that interfaces NXP's PN7120 NFC controller with a Raspberry Pi [1]. The OM5577 is displayed in Fig. 2. This kit consists of the PN7120 NFC controller board which contains the PN7120 NFC controller chip and an RF Antenna. This controller board is then stacked on top of a Raspberry Pi interface board, which is essentially a Raspberry Pi shield. The shield and controller board combination is then placed on top of the Raspberry Pi and plugged into its general-purpose input/output (GPIO) pins. The OM5577 takes up pins 1-26 of the GPIO pins. Pins 3 and 5 are used as SDA and SCL respectively for the I2C protocol, pin 16 is used for IRQ, and pin 18 is used for VEN. Refer to Appendix for OM5577 schematics [2]. The OM5577 kit also comes with a MIFARE Ultralight EV1 NFC card, which is one of the E-Pass devices. NXP provides a Linux NFC API software stack for use with the OM5577 called linux_libnfc-nci [3].

NFC Open Source Kernel Driver

The "nxp-pn5xx" kernel is maintained by NXP on their GitHub NXPNFCLinux [4]. This driver manages I2C communication between User software on the Raspberry Pi and the PN7120 NFC Controller. From a high-level perspective, this allows users to read and write byte streams from and to the controller without explicitly implementing the I2C protocol. This driver is included in the OM5577 Raspberry Pi Linux demo

ISO but for installation along with other drivers (such as the Touchscreen's Waveshare kernel driver), it should be manually installed and built into the kernel. The guide provided in NXP's "NFC in Linux" presentation was used to install the driver into a fully updated firmware of Raspbian [5].

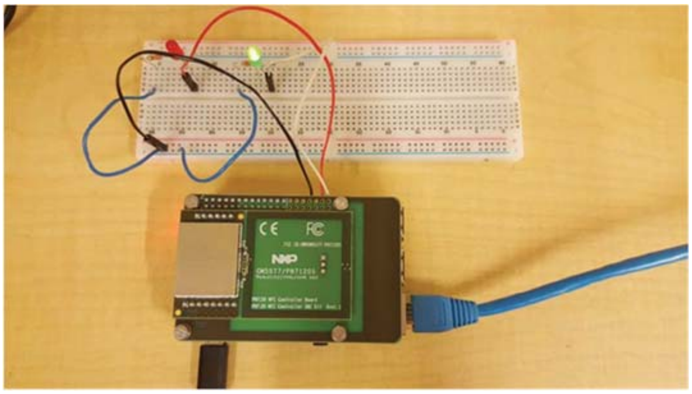


Fig. 2 Raspberry Pi connected to OM5577.

MIFARE NFC Card

The Ultralight card (Fig. 3) is flexible, as thin as a sheet of paper, and about the size of a credit card. It contains 80 B of memory split into 20 pages of 4 B each [6]. There are 34 B reserved for card information and formatting. There are 46 B available for the card's payload, with 7 B reserved for NDEF header information. The remaining 39 B are available for the payload, which is 32 B long. When a device is scanned by the scanner, LEDs are used to indicate the validity of the device's transaction.

Fig. 3 MIFARE Ultralight card provided in OM5577 kit.

Touchscreen

The Kiosk uses a Wontop 7 inch 800 × 480 Capacitive Touch Screen Interface (Fig. 4) to allow customers to use the Kiosk application. The touchscreen contains an HDMI port and a USB Mini-A port. The HDMI port receives standard display output from the PI and the USB port receives power from and sends touch data to the Pi. The Pi receives the touch driver through the driver provided by Waveshare. The Touchscreen interface aims to provide a minimalistic and intuitive experience for customer-card and customer-database interaction to ensure quick and easy use. At the time of purchase, the manufacturers claimed that this product is fully compatible with the Pi. After installation however, the existing kernel which had been previously installed was completely overwritten with the kernel provided by Waveshare. This became a major problem since both the Touchscreen and PN7120 must both be installed in order to develop and run the Kiosk application.

Fig. 4 Wontop 7 inch 800 × 480 Capacitive Touch.

4. Software Design

For this project, development focused mainly on the software running the E-Pass system. The scanner software runs in either bus or kiosk scanner modes and provides E-Pass verification functionality. The kiosk application communicates with the scanner's kiosk mode running on the same Raspberry PI to scan cards and generates the user interface displayed on the kiosk's touchscreen. The Android application provides HCE on phones with NFC capability in order to allow it to be a functional E-Pass. The database server maintains customer information and communicates with the scanner, kiosk, and HCE app to process E-Pass transactions. A variety of programming languages were used, including C, Java, Python, and SQL.

The scanner software is entirely written in C, runs on the Raspberry Pi, and manages transactions involving NFC cards. The source code for the scanner software is located on GitHub [11]. The scanner is based on the demo application [1] that NXP provides with the OM5577. Specifically, the demo application's polling and reading modes are modified to suit the project's objectives. The software runs in two modes: bus mode and kiosk mode. Bus mode is the version of the software running on a system that accepts card payments on a bus while kiosk mode is the version of the software running on a system that is present at a kiosk. In bus mode, the scanner subsystem interfaces with NFC cards, phones running the HCE app, and the database server. In kiosk mode, the subsystem interfaces with NFC cards and the kiosk CUI. The program is compiled with the included makefile and run with the command "sudo ./scan" followed by an argument of either "bus" or "kiosk" to indicate which mode to start the scanner in. Superuser permission is required to use the wiringPi library.

5. Integration and Testing

Each of the project's subsystems were initially developed individually to ensure basic functionality. After each subsystem was confirmed to be working in isolation, the system's total integration commenced. The system is based on a client-server software model where the database acts as the server and all other subsystems are the clients. The database provides information as needed to each subsystem client. In return, the clients can remotely update the customer information stored on the database. The kiosk mode scanner acts as a server to the kiosk CUI, which sends commands to the scanner to determine scanner activity. The scanner also integrates with the HCE app to allow smartphones to be used as E-Passes.

The Java Socket Server uses a number of special messages to notify clients. The server sends a "126" in byte form to notify a client that it has successfully logged in using the credentials it sent. On the other hand, the server sends a "-127" to notify a client that the login failed. After successfully inserting a new entry into the database, the server sends out a "130" to notify the client (kiosk) that it is sending the CID of the newly inserted entry. When a client requests database entries, the server sends a "127" before sending the actual entries. This notifies the client that the server will be sending the database entries that were requested. Lastly, the server sends a "-128" after sending the data that a client requested. This indicates the end of the message and tells the client that the server has sent all the requested data.

The server also requires the clients to send special messages to the server to notify it. The server listens for two special messages. The first code is a "-1" which tells that server to disconnect the client that sent it. Unless the server receives a -1 from the client, either explicitly or through some kind of error, the server will keep that specific client connection open. The second code is a "-128" which indicates the end of a message. If the server receives a message that is not -1 or -128, it assumes that it is part of the message from the client and simply appends it to anything it received after the last -128 or whatever was sent before if it is the first message upon connecting. Unless the server receives a -128, it will simply append everything it receives from the clients. The processing of messages only happen once the server receives a -128 from the client.

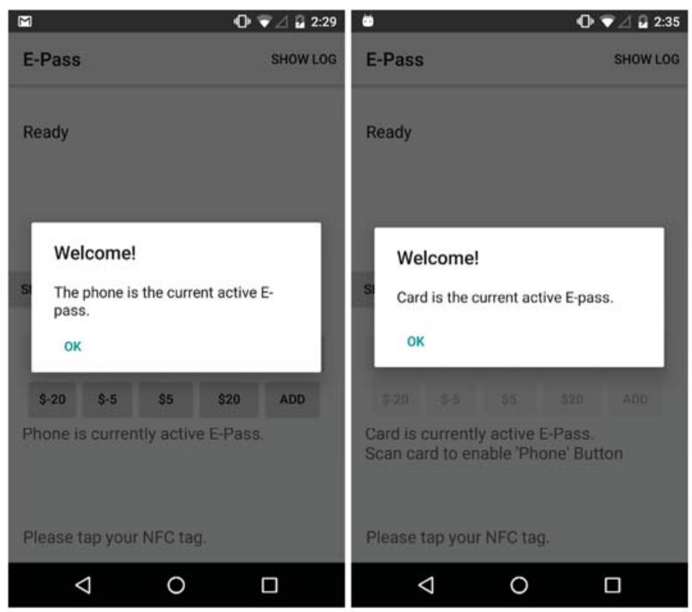
The scanner software is able to store and update a local cache of the database contents to access information when there is no network connection. Caching only occurs in bus mode, as the kiosk UI manages caching for a kiosk system. The main caching function is getFullCache which uses sendMessageToServer to send a cache request to the database over TCP. The database responds with an XML-formatted message describing each customer's CID, Device, Valid, Balance, and Timestamp values. The scanner saves the response in a local XML file; the cache is overwritten with each update. The getFullCache function is called at the beginning of WaitDevArrival, when a card with a CID not in the local cache triggers a database check, and after every successfully sent database update.

When attempting to send a balance update command, and there is no connection to the database, the scanner updates its own local cache. The libxml library [15] is used to manage the cache. The xmlUpdateBalance function is used to update the balance field of a given CID in the cache. Additionally, a copy of the balance update message to be sent is stored in a queue. The next time a successful connection to the database is made after a transaction, the entire cache is updated from the database and the queued updates are all de-queued and sent to the database. Because the queued messages saved in local memory with a linked list, they are lost if the program stops or the scanner turns off. The kiosk CUI also implements its own version of database caching.

Card Emulation

Aside from the setting of the account in this sub-module as mentioned from the previous section, the execution of commands that correspond to each button in the GUI are handled in this sub-module. When a button is pressed on the HCE app GUI, the Card Emulation Fragment class uses the variable acvtivity that is an instance of the MainActivity class. All the commands associated to the GUI buttons are in the Main Activity class.

When the "Phone" button is pressed, the command responsible for activating the smartphone as an active e-pass is executed.



Checking current balance. Checking time stamp.

Fig. 5 Active device messages.

Also, the HCE app requests the user to tap the NFC card to transfer identical payload as what is set to the account except the valid byte is set to 0. When the "E-Pass" app is cleared from the most recent apps opened, Fig. 5a shows what the app looks like when re-launched.

The "Card" button has the same functionalities as the "Phone" button only that it does the opposite: it activates the card and deactivates the smartphone. And it executes the commands for extracting the updated time stamp and updated balance that are copied to the NFC card. The HCE app account is set with an identical payload as the NFC card except the valid byte is "0". The "Check" button executes the command for extracting the updated balance and the "Time" button executes the command that extracts the updated time stamp. Lastly, the "Add" button executes the command that adds up the balance set by the user to their user account. The actual results of the "Check", "Time" and "Add" commands are depicted in Fig. 6a, Fig. 6b and Fig. 6c, respectively.

Checking current balance. Checking time stamp. Adding balance.

Fig. 6 Update messages.

6. Conclusion and Future Work

This project used NFC technology as a basis for an effective fare collection system. There were four subsystems developed: a device scanner, a customer kiosk, an Android NFC HCE application, and a customer information database server. The combination of each subsystem together resulted in a functioning bus transit fare collection service.

The scanner subsystem was designed on a Raspberry Pi. NFC functionality was added with NXP's OM5577 NFC kit. The scanner is used for transit fares and with the kiosk for E-Pass management. The kiosk subsystem works with the scanner to allow customers to initialize cards and add balance through a touchscreen user interface. An Android application uses HCE to allow users to transfer NFC card information to their phone and active it as their E-Pass. Customer information is maintained and updated through a common database server.

The proposed features for the system were successfully implemented. The scanner is able to detect both NFC cards and smartphones running the HCE app as valid forms of payment. The scanner includes the bus transfer system to allow users free use for a period of time after a transaction. It can also cache customer information in case of network issues between itself and the server. The kiosk's touchscreen interface was created and also caches customer information. Both the scanner and HCE app are able to encrypt and decrypt payload information transferred over NFC. The scanner, kiosk, and HCE app all communicate with the database over TCP/IP to update and read customer entries. Database access is restricted by requiring login credentials. While all proposed features were included, there is still opportunity to further develop the system.

There are some examples of possible extensions to the E-Pass system that were outside the timeline and scope of this project. For example, in addition to payload encryption, all communication between the database and the other subsystems can also be encrypted. Additionally, because of the networking between each subsystem, load testing needs to be done to evaluate the system's performance when multiple clients are communicating with the server. The scanner's balance writing method can be improved to allow for total balance writing instead of incremental or decremental writing. This would allow the scanner and kiosk to synchronize payload information with the database whenever needed. The scanner's indication method could also be expanded with the use of speakers or a small display for detailed feedback. Finally, the kiosk's and HCE application's user interfaces could be improved and expanded on.

7. Acknowledgements

The authors will like to acknowledge the technical support of Technology lab at the Department of Electrical and Computer Engineering, University of Manitoba for advice during this design project.

References

1 NXP Semiconductors. (2015, Jul 1). "PN7120 NFC Controller SBC Kit Quick Start Guide" [Online]. Available: http://www.nxp.com/documents/application_note/AN11646.pdf [September 30, 2015].

2 NXP Semiconductors. (2016, Oct 7). "PN7120 NFC Controller SBC Kit User Manual" [Online]. Available: http://www.nxp.com/documents/user_manual/UM10878.pdf [February 29, 2015].

3 NXP Semiconductors. (2015, Aug 27). NXPNFCLinux/linux_libnfc-nci: Linux NFC stack for NCI based NXP NFC Controllers [Online]. Available: https://github.com/NXPNFCLinux/linux_libnfc-nci [November 16, 2015]

4 NXP Semiconductors. (2015, Oct 12). NXPNFCLinux/nxp-pn5xx: NXP's NFC Open Source Kernel mode driver [Online]. Available: https://github.com/NXPNFCLinux/nxppn5xx [October 14, 2015]

5 NXP Semiconductors. (2015, Jul). "NFC in Linux: How to get started with the PN7120S controller board" [Online] Available: http://www.themobileknowledge.com/sites/default/files/NFC_in_Linux_v1.0%20Public.pdf [February 29, 2016]

6 NXP Semiconductors. (2012, Oct 2). "MIFARE Ultralight as Type 2 Tag" [Online]. Available:

http://www.nxp.com/documents/application_note/AN1303.pdf [November 11, 2015].
7 Gordon Henderson. (2015). Wiring Pi [Online]. Available: http://wiringpi.com [October 17, 2015].
8 wakdev. (2016, Feb 25). NFC Tools -Android Apps on Google Play [Online]. Available: https://play.google.com/store/apps/details?id=com.wakdev.wdnfc&hl=en [February 29, 2016].
9 Derek He. (2016, Jan 29). WaveShare 7-inch user space driver [Online] Available: https://github.com/derekhe/waveshare-7inch-touchscreen-driver [February 29, 2016]
10 Wikipedia, “List of NFC-enabled mobile devices,” [Online]. Available: https://en.wikipedia.org/wiki/List_of_NFC-enabled_mobile_devices. [Accessed 17 February 2016].
11 Group 16 (2016, Mar 1). 2016UofMECEG16/scanner: Scanner Software for 2016 UofM ECE Group 16 Capstone [Online]. Available: https://github.com/2016UofMECEG16/scanner [March 1, 2016].
12 Google, “Host-based Card Emulation | Android Developers,” [Online]. Available: http://developer.android.com/guide/topics/connectivity/nfc/hce.html#HCE. [Accessed 28 February 2016].
13 Google, “NFC Basics| Android Developers,” [Online]. Available: http://developer.android.com/guide/topics/connectivity/nfc/nfc.html#ndef. [Accessed 28 February 2016].
14 Google, “NFC Basics | Android Developers,” [Online]. Available: http://developer.android.com/guide/topics/connectivity/nfc/nfc.html . [Accessed 29 February 2016].
15 Daniel Veillard (2016). The XML C parser and toolkit of Gnome [Online]. Available: http://xmlsoft.org/index.html [February 2, 2016].
16 Dave O’Hallaron (1999, Nov 29). CS 213 Lectures, Fall ’99 [Online]. Available: http://www.cs.cmu.edu/afs/cs/academic/class/15213-f99/www/class26/tcpserver.c [December 28, 2015].
17 Dave O’Hallaron (1999, Nov 29). CS 213 Lectures, Fall ’99 [Online]. Available: http://www.cs.cmu.edu/afs/cs/academic/class/15213-f99/www/class26/tcpclient.c [December 28, 2015]. network programming -C: socket connection timeout (2010, Apr 8). [Online]. Available: http://stackoverflow.com/questions/2597608/c-socket-connection-timeout [February 20, 2016].
18 http://www.nearfieldcommunication.org/technology.html
19 http://www.nxp.com/products/identification-and-security/nfc-and-reader-ics/nfc-everywhere/nfc-technology:NFC-TECH
20 Want, Roy. "Near field communication." IEEE Pervasive Computing 3 (2011): 4-7.
21 Dutson, A.J., Todd, R.H., Magleby, S.P., & Sorensen, C.D., “A review of literature on teaching engineering design through project-oriented capstone courses”, *Journal of Engineering Education*, 86(1), 17-28, 1997.
22 Colin Vincent, Peter Wild, “Current Practices in Final Year Engineering Design Courses”, Available http://library.queensu.ca/ojs/index.php/PCEEA/article/viewFile/3809/3769, [11 April 2016].

A Data Analytics Approach to a Computer Science Senior Capstone Project Management Tool

Dan Tappan
Department of Computer Science, Eastern Washington University, Cheney, WA, USA

Abstract - *Managing students in computer science senior capstone courses is comparable to herding cats. Many students can not or will not perform their tasks, and they go to great lengths to hide this behavior from their teammates and the instructor. This poster introduces a proof-of-concept web-based management tool that helps all stakeholders track activities, report on progress, and identify issues and concerns before they become problems, as well as reflect on the development process from an educational perspective.*

Keywords: capstone, project management, data analytics

1 Introduction

Software engineering has a bad reputation of being far too similar to the age-old joke from the public domain in Figure 1. Despite endless warnings to students in their senior capstone course in computer science that they need to take the development process seriously, the outcome almost inevitably resembles “The Cartoon” to a significant degree. A big part of disciplined development on a reasonably complex real-world project, which is indeed the experience that the students are supposed to be gaining, is to learn how to iteratively make a small plan, execute it, and verify that the actual results reasonably match the expected results. If they do not match, then some remedial action is necessary before continuing. However, this process only works if students have all three of these elements, continuously apply them in a serious, disciplined manner, and honestly assess and report their performance to their teammates, their project sponsors, and the course coordinator. Needless to say, typical team dynamics result in one or more students not carrying their weight and trying to hide this behavior.

This poster showcases a prototype web-based tracking system that helps students account for their activities and those of their teammates in an informative, intuitive way that is not especially onerous. Nobody enjoys generating status reports, but they are a necessary evil in a field where anarchy and disorder are the norm, even among professionals. Students, left to their own devices, generally fare far worse.

This experiment followed eight teams totaling 30 students as each team worked on its own independent project over 23 weeks covering two academic quarters. The development methodology was Agile, which offers considerable freedom, but it also demands a reasonable level of maturity and discipline. The intent was for students to be able to partition their work into bite-sized activities that aligned well with Agile thinking, doing, and verifying. Status reports helped students identity gaps and disconnects in the plans and their execution, as well as monitor the behavior of themselves and their teammates. They also provided a rich opportunity for students to reflect on the process to learn from it, instead of fixating on the product, which is really not the true purpose of the course.

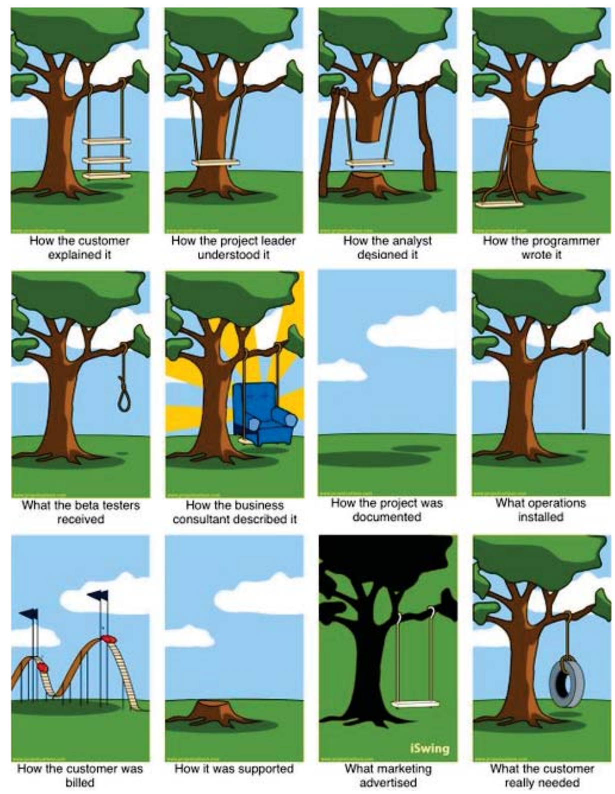


Figure 1: Software engineering reality

2 Report elicitation

The Agile sprint period was one week, Saturday to Friday, with the submission period on Friday. An earlier pilot test in the author’s prerequisite software engineering course had successfully used a shorter period, so this approach is very flexible. There are two kinds of reports to submit.

2.1 Individual reports

Each team member completes their individual report on their own. It contains elements that will contribute to the public and private summaries in the next section. The first section is public and elicits the progress that a student has made. It starts with basic time-keeping of hours of effort per day. Students' grades are not directly tied to these numbers, but because there is an average expectation over the project, students in the past had vastly inflated their numbers. The second section, also public, elicits new activities that were started, open activities that were continued from previous sprints, activities that were abandoned, and those that were transferred to other teammates or shared. Each automatically receives its own unique reference number that is retained throughout the project. A student must briefly describe the activity, including an estimate on completion time for new and open ones.

The third section is private and consists of three subsections that evaluate the claims made by teammates in the last sprint. For each teammate, it summarizes the hours and the four categories of activities and requires the student to indicate whether they concur, do not concur, or are not sure. The latter two require a brief explanation. It also asks whether performance is meeting expectations. Finally, there is a field for general comments.

2.2 Team report

The entire team completes the single team report together. It relates to the general experience of the team as a whole over the last sprint, not to specific activities of individual members. The reflective framework first asks the team to articulate briefly for each of these questions which aspects were both the easiest and hardest:

- *understand*: comprehending what needs to be done;
- *approach*: planning to solve it;
- *solve*: implementing the actual solution;
- *evaluate*: demonstrating that the performance of the solution is consistent with the problem and everything else in the project.

It also requests an estimate on how far along the project is and whether this pace is on target to finish on time. Teams routinely neglect communication with the project sponsors, so the next question asks whether there was any, and if not, when the last contact occurred. The final question addresses whether there are any issues, concerns, or comments not captured elsewhere.

3 Report generation

Reports are HTML-based emails that go to all stakeholders.

3.1 Public reports

The public report goes to each member of each team and the project sponsors. It summarizes the hours in a variety of intuitive statistical ways and depicts trend information over the project to this point. It also organizes the four types of activities from all team members into a readable summary, also with trend information. Finally, it presents the contents of the team report.

3.2 Private reports

The private report goes to the coordinator only. It contains everything in the public report, as well as the private entries. As the coordinator has many teams and students to manage on a weekly basis, the results are presented in such a way that skimming it reveals whether deeper investigation is warranted. A colorful heat map and matrix of green, yellow, and red dots help cross-reference each team member with themselves and each other. There is also an automatic tie-in with GitHub to produce a graphical representation of activity in the code repository by individual and team.

4 Results and discussion

In its first deployment as a proof of concept, this approach has already far and away demonstrated that it expedites the processes of keeping track of teams and individuals. In fact, instructors scheduled to teach this course in the future are envious and want access to this tool. It is difficult to compare this group of students against past ones because each offering of the course involves different projects, but there have been far fewer cases of deceptive behavior, or they have not continued very far before being questioned. The students very quickly got a feel for how to articulate their progress and interpret that of their teammates, which has done wonders for managing expectations. Software development is not a constant, linear activity, and certainly not for students in an academic setting. They all have ups and downs, easy weeks and difficult ones, etc. As long as their teammates are informed of, understand, and accept this performance, then they are functioning effectively as a team. This approach had a notable effect on improved quality in the final products. Anecdotal evidence suggests that it has also improved the process of software development, and especially the behavior of the people involved, who are traditionally the most troublesome factor.

The data elicited throughout the process are both quantitative and qualitative. No amount of manipulation is going to entirely automate the process of making sense of them, but the expectation is that there are likely to be clear patterns related to various aspects of performance. Faculty who have previously taught any course already have a general feeling for when, where, why, and how certain things go right or wrong. This approach does not replace such wisdom, but it appears likely to be helpful in managing the large amount of activities and information from so many students on completely independent projects over a long period. Other analytical measures will likely be added over time to highlight areas of interest and concern, etc. in an intuitive manner that is easier for all stakeholders to identify, monitor, and resolve.

Survey Analysis of the Expectations and Outcomes of a Computing and Technology Interdisciplinary Capstone

A. Nettles, J. Livingston, K. Willett, A. Douglas (students), and M. Estep (faculty)
Department of Computing and Technology, Cameron University, Lawton, OK, USA

Abstract - *The Cameron University Computing and Technology Capstone course was an interdisciplinary project that allowed students to develop a real-world software application. Despite having been taught this way for years, a study had never been initiated that determined whether outcomes intended by students, teachers, and clients were being met. A student group in Cameron's Research Topics in IT course endeavored over the course of the spring 2016 semester project to propose such a study to conclude whether this class was achieving its goals for all parties affected. To accomplish this goal, the group utilized Google Forms software to compile two brief online surveys for Capstone students, which would be conducted at key points in the Capstone class. Once an adequate number of responses had been collected, the group compiled the results in order to analyze the outcomes.*

Keywords: Capstone, Survey, Case Study, Research, Analysis

1 Introduction

The Computing and Technology interdisciplinary Capstone course was designed to meet the needs of students destined to make their mark in the technically challenged world by developing a real-world software application. Not only does this prepare students for future careers, but the integration of multiple disciplines provides an environment "for the challenges of any one computational domain by virtue of their exposure to multiple domains in which computational thinking plays prominent a role" [1]. Previous case studies were written by and conducted from the perspective of faculty members. Research Topics in IT, the compliment course to Capstone, was created as a way for students to conduct case studies on the Capstone experience from their own perspectives [2]. In an effort to assess the efficiency of this program for the first time this year, a study was now being initiated by students to determine whether the needs of students, teachers and clients were met. With combined team effort, periodic assessment, and a compiled analysis of results, a selection was made to determine the best viable product which could be used as a solution to meet the expected demands of a client.

1.1 Capstone overview

1.1.1 Structure

An authoritative explanation of the structure of the Capstone class is available in the paper: *Web-Based Database Project for the Lawton, OK Police Department: Interdisciplinary Systems Software Development at Cameron University* by Cameron faculty Smith, Zhao, Estep, and Johari [3]. The goal of the class was to complete a moderately sized software project that used software engineering principles to construct a web-accessible database program for a local business or government entity on a *pro bono* basis. This program could be completely original or based on an existing website, as was the case at one University of South Carolina Capstone course [4]. The class was structured as if it were a mock company with two teams competing to produce the product they hoped the client would use as their solution. Separate Multimedia (MM), Computer Information Systems (CIS), Computer Science (CS), and Information Assurance and Security (IAS) sections, each headed by a faculty member, existed for each respective discipline in the Department of Computing and Technology.

When not needed for part of the Capstone, the students met after class to perform tasks deemed necessary by their respective professors. From time to time, especially after the end of the first half of the 16-week term, the individual sections met as a collective whole to work on the project. Each subsection, be it CS, CIS, etc., met at the beginning of the semester, and two leaders were selected by the faculty in charge of that section. Then through review of blind resumes, these leaders formed two separate teams. These teams were then further divided according to what the sub-team leader felt was appropriate and could be fluid during the semester project. The team leader selected at least one assistant leader who served as leader upon the main leader's absence and gathered reports from others as the main leader needed. The leader could also delegate other positions, such as being in charge of documentation, online research, or tracking teammates' PHP code contributions. Each of these sub-teams formed two super teams, each with a representative sub-team from the areas of CS, CIS, MM, and IAS. In addition to having a representative of each sub-team in each instance of the super team, there was also a chief leader of the super team from the CIS section who served as a mock boss of all the sub-team leaders and coordinated the teams' overall work. A backup person was chosen by the faculty to compliment the chief leader in case of absence or other difficulties that could have prevented him from fulfilling his duties. Outside the command structure described above was a group from the English department that had helped the Capstone class over the last few iterations. This group consisted of one senior tech editor and two tech writers who assisted the super teams in reviewing documents and work sent between the super team and the client. Each super team was assigned one tech writer, but the senior tech editor position was not under the chain of command of either super team.

1.1.2 Time flow of product development

The class had a tentative structure timeline presented to each team member at the beginning of class on what was expected and when. The class began with the above described resume writing and readings as well as the dividing of the classes into sub-teams. Next to happen was the initial interview of the combined class of the client. During this meeting the students got to know exactly what the client wanted in the finished product. The professors left the exact questioning of the client to the students and were more than willing to let the students struggle with this part and fail to ask pertinent questions as part of the learning experience. The first eight weeks of the class were primarily the domain of the CIS sub-teams. They were responsible for setting up the backend database by designing first a plan for how entities in the database would be represented and later implementing said database. While the CIS team was designing the database, they held meetings with the heads of the CS and IAS teams to see if their plans were feasible.

Approximately eight weeks into the class, the teams held their first in progress report (IPR) for the client. This meeting served to reassure the client that work was indeed getting done on their project and allowed the client to provide feedback. This part was important because the client would not always know what was wanted, and the requested solution was not necessarily what was needed. After the first IPR, the project largely changed hands from the CIS team acting as the lead team to the CS team acting as the head team. The role of the CIS team did not end completely however because they were still needed to make changes to the SQL database as necessary. At this point in the class, the IAS team, through the use of penetration testing, analyzed the security mechanisms of the database and made change requests to he CIS team. Despite these other teams' involvement, the work in this second half was mostly the domain of the MM and CS team. The CS team began the arduous task of making a web interface using PHP code that would access the database previously completed that would try to meet three seemingly mutually exclusive goals: being feature complete, being user friendly, and being secure. The MM team at this point assisted the CS team by working on the aesthetics of the web pages constructed by the CS team and doing any graphic work for the web pages.

Approximately one month before the project was due in May, the super teams held a second IPR, this time headed by the CS team that presented the almost finished version of the project to the client. The client had the chance to point out anything that was miscommunicated to the team or any last minute wish list items for the team to add in, time permitting. The end of the class was unique for many students because the university opened up the computer labs on one or two Saturday and Sundays for the team members who were able to come in and rush to meet deadlines. The class concluded by giving the client a final report, and students were required to dress in business formal attire and participate in group presentations to the client. After the presentation, the client was to choose between the two products developed by the competing super teams, but the client chose to merge the best features of each finished product into a third product. As Smith mentions, this practice provided internship opportunities for interested students to then work over the summer or fall semesters on the new final project [3].

2 Survey overview

2.1 Motivation

After reading relevant research papers on the subject, the group discovered that the approach they would be using was not unwarranted. Jareo, et al. in their paper *Tool for Assessing Student Outcomes,* asserts that universities have been unsuccessful in assessing "whether their students are exhibiting the outcomes they desire" [5]. They go on to say how "Researchers have pointed out that course grades do not, in and of themselves, provide good measurements of student outcomes," [5]. They elaborate on this claim by mentioning that because the average computing class contains a writing component, individual programming assignments, projects, and exams that a student could do very poorly or fail on any one component of the class and still survive with an A or B final grade [5]. They mention in particular the writing component is difficult to assess because all of the other grades can cover poor performance in that area up [5]. Therefore, they point out that a survey mechanism like this is warranted just to be able to see if a student is indeed learning what the teacher, department, and university is intending students to learn in any given class [5]. Furthermore, the approach to conducting web-based surveys is warranted because Jareo, et al. mentions that research shows typical paper surveys have a return rate below 25% [5].

2.2 Questions

All questions in the survey were reviewed by each team member working on this paper and were deemed worthy of asking. In constructing the survey, the group tried to make a survey brief enough to appeal to students and retain their attention. This challenge of having questions meaningful enough to be worth asking was discussed by Jareo et al. [5]. In consulting Jareo's research as a source, the group also made the survey web-accessible to aid in administration, collection, and analysis of the surveys. After undergoing three revisions by the team members, the first survey was distributed to the lead faculty member in charge of the overall Capstone experience this year, Mr. Dave Smith. Upon pointing out his objections, as well as possible redundancies of some questions, the survey was revised one last time before approval was given to distribute the surveys on February 17, 2016. After approval was given, they were given out on February 24.

3 First survey

The first survey results were collected on February 24, and after analyzing the participation, the group found that by February 29 there were 26 responses. The survey participants were asked if both their sub-teams and their overall team were on track at this point in the semester, which is approximately halfway done, and 100% of participants said that they felt that the project was indeed on track. Concerning the two questions that were about how realistic this class was and whether it was worth their time, only one response on each of these questions said the class was not worthwhile. The multimedia team in particular felt like they were underused in this project and that the project could get done just as easily with far fewer participants from that department.

3.1 Analysis

The first survey yielded meaningful data, giving the Research Topics in IT group the opportunity to make important observations.

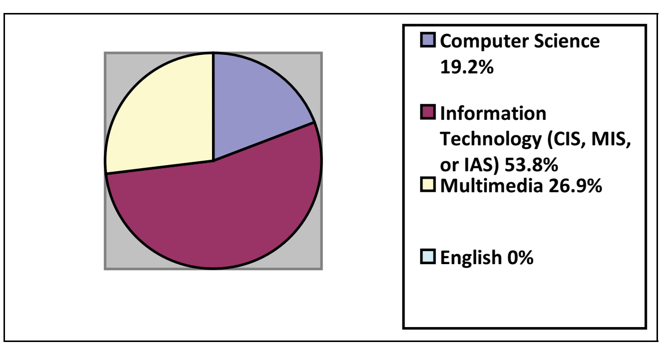


Figure 1

The chart shown in Figure 1 indicates the major concentrations of all participating students. Information Technology was the majority, and no English major participated.

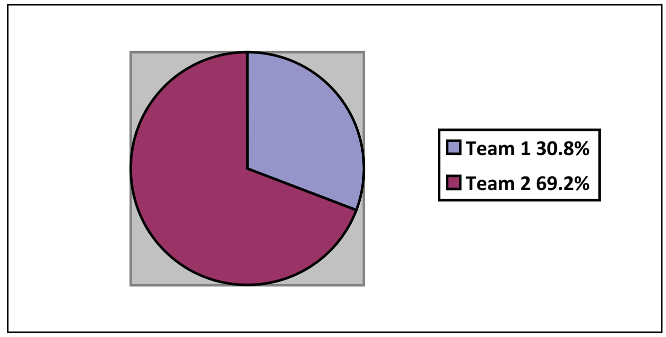


Figure 2

The chart shown in Figure 2 indicates the teams that participated in the survey. Of two teams, the majority of the students that participated were in team 2.

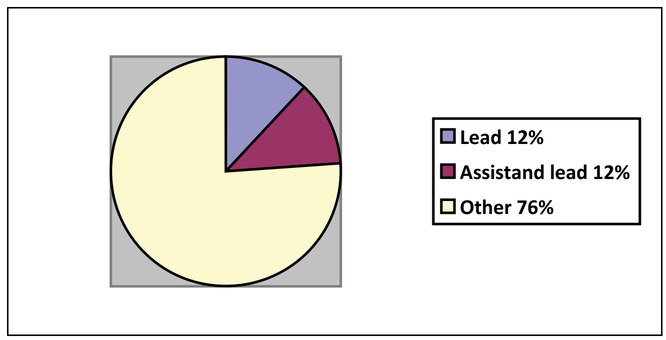


Figure 3

The chart shown in Figure 3 indicates the role each participant played in the project, for which the majority of students chose roles other than lead or assistant lead. It was vital to consider the opinion of the students that were not in a lead role.

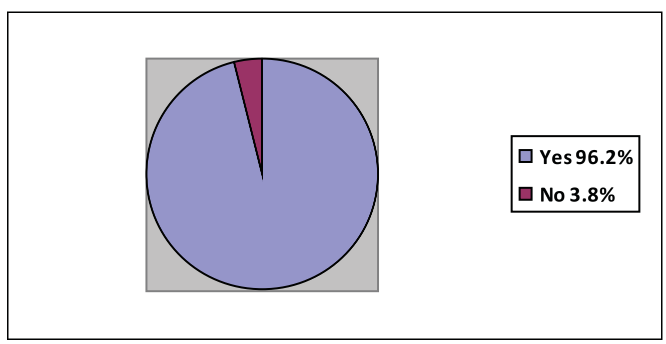


Figure 4

The chart shown in Figure 4 indicates the expectations of the project and whether or not it was valuable to the participants' major. One Multimedia student responded no and briefly explained that he believed multimedia students were not given enough tasks and that the part of the project they were working on was not benefiting them.

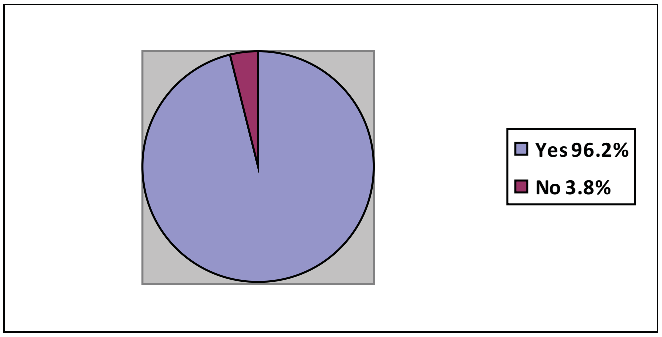


Figure 5

The chart shown in Figure 5 indicates the opinions of the participants on whether the Capstone course had been worth the time. All but one participant selected yes, who was the same student that selected No in Figure 4.

4 Second survey

The second survey results were collected on April 6, and after analyzing the participation, the group found that by April 13 there were 18 responses. The survey participants were asked if their sub-teams were on track at this point in the semester, which is nearly finished, and 94.1% of participants said that they felt that the sub-teams' project was indeed on track. Participants were also asked if the project teams were on track at this point, to which 70.6% responded that the project was on track.

4.1 Analysis

The second survey yielded important data for the research team to work with, including new information about the class's pacing and worthwhileness.

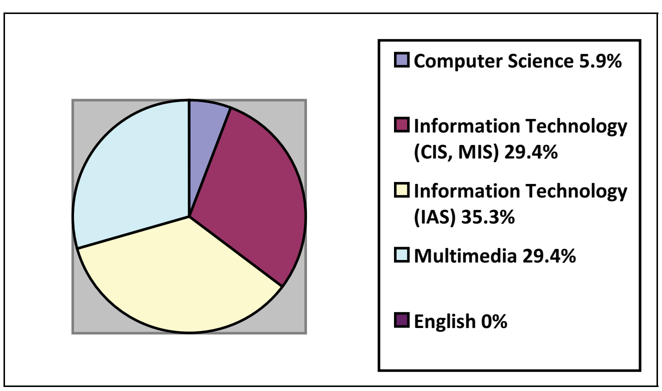


Figure 6

The chart in Figure 6 indicates the major concentration of all participating students. Information Technology (CIS, MIS), Information Technology (IAS), and Multimedia participants were the majority. Only one Computer Science major participated, and no English or other majors participated.

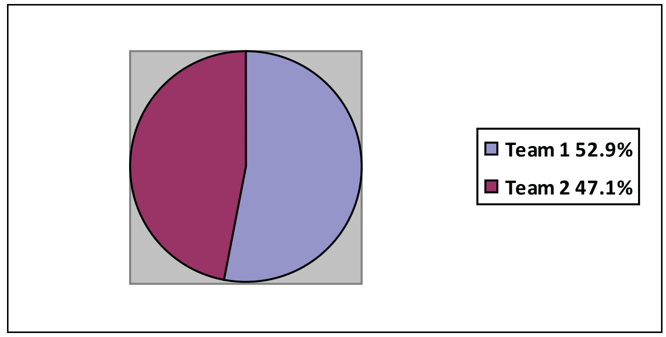


Figure 7

The chart in Figure 7 indicates the teams that participated in the survey. Team 1 had the most participants with a slight majority.

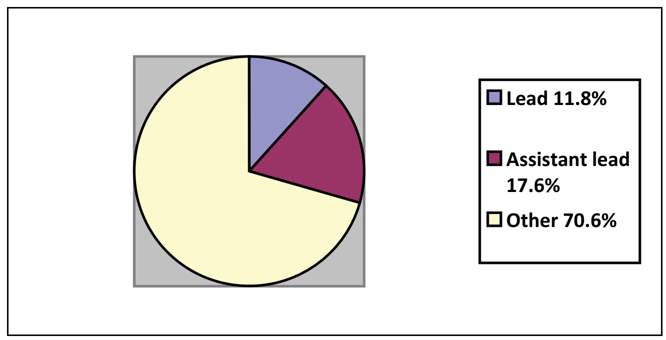


Figure 8

The chart in Figure 8 indicates the role each participant played in the project, for which the majority of students chose roles other than lead or assistant lead.

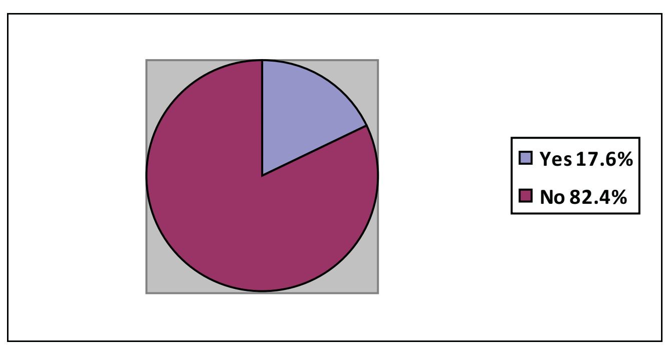


Figure 9

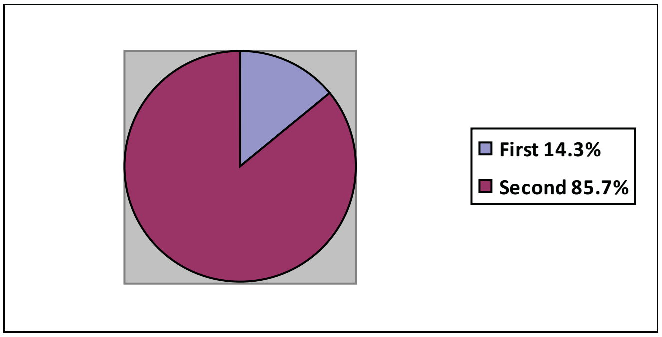


Figure 10

The charts in Figures 9 and 10 indicate students' opinions about the pacing of the class. Figure 9 shows students' responses to the question of whether the pacing between the first and second eight weeks was even. Three students felt that the pacing of the first and second eight weeks was approximately the same. As shown in Figure 10, 12 of the 14 students who thought the pacing was unequal said that the second eight weeks was faster-paced than the first.

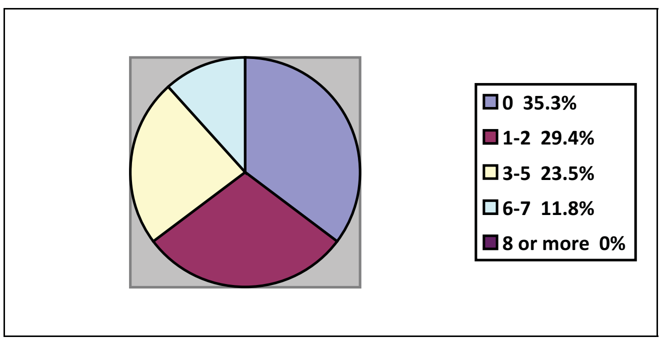


Figure 11

The data in figure 11 displays students' opinions about how many days the Capstone course was not worth their time. The most frequent answer was zero days (all days worthwhile) with six responses.

4.2 Changes made

Since the first survey, several questions were added, modified, or removed, following suggestions of the lead faculty member, to help gather refined meaningful information. In the question about what major the students had, the IT answer was split into one answer for IAS and one answer for CIS or MIS. A question asking what students are personally working on was removed. A question asking if the project was sufficiently complete in order to meet the next deadline was also removed. Three questions about the pacing of the class were added, along with one question asking how many days the students felt that coming to class was not worth their time.

5 Analysis of results

Key findings extrapolated from the surveys are as follows: 1) most students felt that the Capstone course was worthwhile and beneficial to them, 2) Multimedia students did not feel that the class was worthwhile, 3) the majority of students (82%) felt that the course was unevenly paced and the faculty should address this issue, and 4) six people found at least three class meetings not worth their time. The group recommended the faculty remedy these issues by making every class a worthwhile learning experience for all students to attend.

6 Conclusion

This new initiative taken by the Cameron University Computing and Technology Capstone course was one that demanded coordination, cooperation, and team effort. Good leadership, delegation, coordination, and completion of tasks assigned, were all essential factors that determined the quality software application produced. The competitive nature of the study would certainly add volume, as teams were expected to work efficiently to get their final product pleasing to the client, thus chosen to be used as a solution to meet the needs of a business government entity. It was a hope that this survey study would aid in fulfilling the expectations of all concerned.

7 References

[1] L.K. Soh, A. Samal, S. Scott, et al. "Renaissance Computing: An Initiative for Promoting Student Participation in Computing." Proceedings of the 40th SIGCSE Technical Symposium on Computer Science Education, SIGCSE 2009, Chattanooga, TN, March 4-7, 2009.

[2] M. Estep, K.D. Smith, C. Zhao, A. Johari, and K.D. Smith, "Creating synergy between undergraduate senior computing research and applied capstone courses." Proceedings of the 2015 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS'15), Las Vegas NV, July 27-30, 2015.

[3] K.D. Smith, C. Zhao, M. Estep, A. Johari, and W. Carney. "Web-Based Database Project for the Lawton, OK Police Department: Interdisciplinary Systems Software Development at Cameron University". Proceeding of the 2014 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS'14), Las Vegas NV, July 21-24, 2014.

[4] J.G. Flowers. "Improving the Capstone Project Experience: A Case Study in Software

Engineering." Proceedings of the 46th Association for Computing Machinery Annual Southeast Regional Conference (ACM SE '08), Auburn AL, March 28-29, 2008.

[5] M. Jareo, D. Parks, and R. Tashakkori. "Tool for Assessing Student Outcomes." Proceedings of the 45th Association for Computing Machinery Annual Southeast Regional Conference (ACM SE '07), Winston-Salem, NC, 2007.

Database and Security: Creating a Secure Database for a Capstone Application Development Project

J. Dodson, W. Rae, R. Shrestha, S. Thapa, C. Trantham (students), and M. Estep (faculty)
Department of Computing and Technology, Cameron University, Lawton, Oklahoma, USA

Abstract – *During Cameron University's Spring 2016 IT Capstone course, students were given the chance to gain experience in a simulated real-world software development company, for n actual client. Their client, Comanche County Memorial Hospital's (CCMH) Surgical Resources Department, needed a system to track incoming bones and surgical implants purchased from vendors. The authors of this paper were accountable for creating a database and guaranteeing its security, which was crucial because of its sensitive information. The cyber security teams developed policies in conjunction with existing hospital policies, and used reputable penetration tests and protocols to ensure that the application was free from vulnerabilities by attacking its code and identifying and eliminating weaknesses. This paper documents all methods used by the team as it made progress throughout each phase of the project.*

Keywords – Capstone, Real-world Project, Security Testing, Vulnerabilities, Software Development, Database.

1 Introduction

During the Spring 2016 Capstone IT 4444 course at Cameron University, approximately forty-four students from the Information Technology, Computer Science, Multimedia, and English departments came together to work for a simulated real-world software development company. As in the past Cameron capstone courses, the students gained hands-on experience with the tools and techniques used by professionals in their fields and learned more about teamwork, working within the project scope, and satisfying the needs of a client [1,2,3]. These students were divided into two teams and were given the task of developing a fully-functioning bone and implant tracking system to be used by various employees of CCMH. In later sections, this paper will examine the development and execution of the security protocols and tools used to test the system.

1.1 Forming teams

Each student submitted two copies of their resumes to the Capstone instructor, one version complete and one version anonymized. From these resumes, the Capstone instructor picked two project leaders. These project leaders were given the anonymized version of the resumes and hand selected their teams, equally splitting students from each department. The chosen students were then divided into groups that defined their role in the project. These groups included a database team, a cyber security team, a multimedia team, a middleware programming team, and a technical writer. Each group was assigned a team leader who guided the team through the project, delegated tasks, and made sure milestones were met. The project leader also assigned a deputy project leader to assist when needed.

1.2 Initial client interview

The Capstone instructor gave the two teams a brief summary of the project. Each team was tasked with creating questions for the client to be asked during the initial client interview. Both teams had representatives from each of their sub-teams ask the client questions. Several team members took notes which served as an initial reference for what the client required. The purpose of this interview was to determine the project requirements and to develop a course of action, complete with milestones for each team.

1.3 Purpose

The purpose of this study is to analyze the methods used by the security teams during the security hardening of the system and to create a reliable, relocatable method to be used during application development by other groups. Reliability and a secure application were the two key requirements of the client.

1.4 Scope of project

The scope of the project for the security teams included the following: (a) information gathering, (b) systems vulnerability and penetration testing to include PHP code, HTML code, database MySQL code, and (c) validation of the data.

1.5 Not in scope of project

Activities not included within the scope of the project by the security teams included the following: (a) user training in best business practices, (b) disaster recovery procedures, (c) emergency response procedures in the event of a systems breach or hardware failure, and (d) social engineering to acquire sensitive information from the client.

2 Identifying Vulnerabilities

The security teams concluded that identifying possible vulnerabilities was the first step in the development of their testing strategies. To identify the possible vulnerabilities of their system, the teams used the OWASP Top Ten List of Application Security Risks [4] shown in Figure 1.

OWASP Top 10 – 2013

A1 – Injection
A2 – Broken Authentication and Session Management
A3 – Cross-Site Scripting (XSS)
A4 – Insecure Direct Object References
A5 – Security Misconfiguration
A6 – Sensitive Data Exposure
A7 – Missing Function Level Access Control
A8 – Cross-Site Request Forgery (CSRF)
A9 – Using Components with Known Vulnerabilities
A10 – Unvalidated Redirects and Forwards

Figure 1. OWASP Top 10 Application Security Risks

3 Vulnerability and Configuration Testing Tools

To properly test the database and web application for vulnerabilities and proper configuration, the security teams had to create a controlled environment. Besides performing manual code and configuration reviews, automated security tools had to be chosen to assist in checking for known vulnerabilities. Rather than test on the Cameron University or CCMH production networks, the security teams opted to build a testbed environment.

3.1 Testbed

The security teams' decision to build a testbed environment separate from production networks was made to create a controlled environment for accurate testing, and to limit the risks of introducing untested and potentially unsecure code to a production environment. Also, the potential risks of running the various security related tools on a production environment outweigh the benefit in such a scenario.

Cameron University provided two desktop computers in a special security Virtual LAN (VLAN) for testing. Students also used their own personal laptops for further testing of the code.

All computers used were running Microsoft Windows 7 as their host operating system. Students then loaded Oracle VM Virtualbox on the machines, "a general-purpose full virtualizer for x86 hardware targeted at server, desktop and embedded use" [5]. This allows a user to create virtual machines with many different software environments and test configurations and run attack simulations without any risk to the host machine or network in which the virtual machine resides [6].

3.2 Kali Linux

With Oracle VM Virtualbox, the students opted to build a Kali Linux Version 2016.1-based virtual machine. "Kali Linux is a Debian-based Linux distribution aimed at advanced penetration testing and security auditing. Kali contains several hundred tools aimed at various information security tasks, such as Penetration Testing, Forensics and Reverse Engineering." Kali Linux is developed and maintained by the security company Offensive Security [6,7].

Once the creation of the Kali Linux virtual machine was complete, the students used open source MySQL Community Server version 5.6.28 for database hosting functions [8] and phpMyAdmin 4.5.3 for MySQL administration functions [9]. Webpage hosting was handled using the open source Apache web server that is included with Kali Linux. Figure 2 shows the phpMyadmin interface.

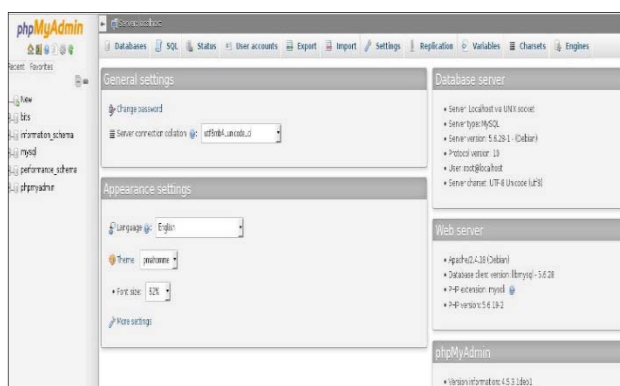


Figure 2. phpMyAdmin Interface

With the setup described above, the security teams tested various configurations and attacks against the application without fear of interrupting production work or permanently breaking server software requiring lengthy reloads or refresh of software.

3.3 Nessus

Tenable Nessus is a widely-used multi-function vulnerability scanner. Scanning takes place from the Tenable website utilizing a victim IP address [10]. The security teams used Nessus to perform different types of scans. These scans include the following:

- Host Discovery Scan
- Bash Shellshock Detection
- Web Application Test
- Basic Network Scan
- Advanced Scan

The students ran Nessus in Kali Linux, which was accomplished by launching Nessus from the command line with the command */etc/init.d/nessusd start*. This caused Nessus to run in the background. The team then used Ice Weasel, Kali Linux's Internet browser, to open the local host at 127.0.0.1:8834 so the Graphical User Interface (GUI) for Nessus appeared. From this point, each separate scan was configured and launched. From the results, the students gathered information including current vulnerabilities categorized by risk, top hosts, and operating system, vulnerability, and host count comparisons.

3.4 SQLMap

"SQLMap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over database servers" used from the command line [6,11]. The students used this tool to perform an SQL injection attack scanning the database server to provide information which included usernames, passwords, password hashes, and the database structure. SQLMap found vulnerabilities that could be used to exploit the database to find critical information.

3.5 OWASP Zed Attack Proxy (ZAP)

The OWASP Zed Attack Proxy (ZAP) is a free tool used to find security vulnerabilities in web applications. Zap is maintained internationally by volunteers. [6,12]. The students used this tool to scan the software for any vulnerabilities and to scan the localhost where the database resided on the local system. Once the scan was completed, the students were able to see the vulnerabilities that the tool found. The tool also printed the recommendations for fixing these vulnerabilities. The OWASP Zap tool's initial interface is shown in Figure 3. From there, the students conducted their scan by entering the URL of the site they wish to attack.

Figure 3. OWASP ZAP

3.6 XSS-Me

"Cross-Site Scripting (XSS) is a common flaw found in todays web applications. XSS flaws can cause serious damage to a web application. Detecting XSS vulnerabilities early in the development process will help protect a web application from unnecessary flaws. XSS-Me is the Exploit-Me tool used to test for reflected XSS vulnerabilities" [13]. The students used this tool to test all the forms of the database to make sure that cross-site scripting could not occur. The test included having to input data into the forms and then run the test. The test would then try to run algorithms to see if they would work within the database. Figure 4 shows the student's XSS-Me test results.

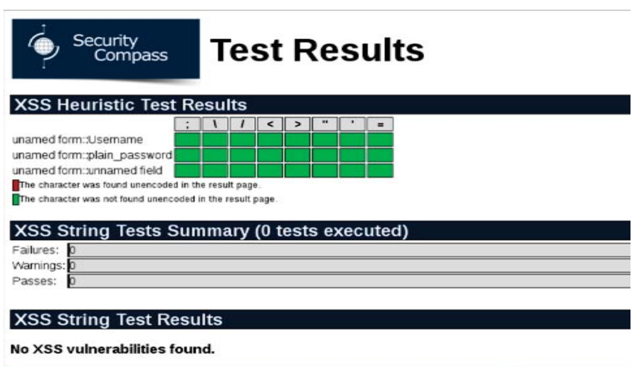


Figure 4. XSS-Me

4 Security Implementation during Application Development

An important step in building a secure web application is to implement security into the application during the development process. During this Capstone project, the security teams worked closely with the computer science teams to ensure security was properly implemented. The following sections will cover the secure storage of sensitive items through hashing/salting methods and ensuring a strong password policy is in effect.

4.1 Hashing of stored passwords and security answers

Hashing utilizes an algorithm to change a password to an encrypted value [6]. Password hashing takes strings and uses an algorithm to get a different value from the original strings. In order to secure the databases for CCMH, the security teams used a password hash function. The user's passwords and security questions were stored using hash with salt (adding a random string to the beginning or end of the hash) [6].

Database hashing/verification functions used in application were built into PHP. To hash, the students used the code < *password_hash($passwordString, PASSWORD_DEFAULT);* >, which generated a random salt, and hashed the password using the "DEFAULT" bcrypt algorithm, as designated by PHP. This constant was designed to change over time as new and stronger algorithms were added to PHP. For that reason, the length of the result of using this identifier could possibly change over time. This function automatically used the newer and stronger algorithms in the future without any changes to the code. The password hashing code can be seen in Figure 5.

```
function unlockAccountInfo()
{
    if(isset($_POST['UID']) && $_POST['UID'] != "") {
        $unlockUserID = $_POST['UID'];
        $con = connect();
        $sql = "UPDATE account SET InvalidAttempts = 0 where userID = '$unlockUserID'";
        $result = $con->query($sql);


        $sql = "SELECT * from account WHERE userID = '$unlockUserID'";
        $result = $con->query($sql);
        $row = $result->fetch_assoc();

        $accountID = $row['AccountID'];
        $oldPassword = $row['Password'];
        $date = date("Y-m-d");

        $stmt = $con->prepare("INSERT into passwordhistory(AccountID, Password,DateChanged) VALUES (?,?,?)");
        $stmt->bind_param('iss', $accountID, $oldPassword, $date);
        $stmt->execute();
        $stmt->close("Y-m-d");

        $twoDigitYear = date('y');
        $password = "CCMH";
        $password .= $twoDigitYear;
        $password .= "is";
        $expDate = date("Y-m-d", time() - (60*60*24) );
        $password = password_hash($password, PASSWORD_DEFAULT);
```

Figure 5. Password Hashing Code

The output generated by this hashing and salting function resembled something similar to *$2y$10$QjSH496pcT5CEbzjD/vtVeH03tfHKFy36d4J0Ltp3lRtee9HDxY3K.* In this output:

- $2y$ designated which hashing algorithm was used.
- 10$ designated the "cost". Essentially, this determined CPU cycles used to salt and hash the data.
- The remaining characters were the random salt and actual hash of the password.

To verify if the user's entered password is correct, the students called the function < *password_verify($passwordFromLoginForm, $hashedPasswordFromDB);* >. This was another built-in function (as of PHP version 5.5.0) which determined the hashing algorithm, cost, and salt from the given hash, hashed the entered password using those criteria, and compared the two hashes. It returned a Boolean true/false. The password verification code can be seen in Figure 6.

```
if($result){
    $result = $result->fetch_assoc();
    if($result["InvalidAttempts"] >= 3){
    close($db);
        ?><script>
            alert("User is locked out.  Contact BITS Admin.")
            window.location = "index.php";
        </script><?php
    }else if(valid_input($result["AccountID"], $result["Password"])){

        //loginSuccess is a variable to hold 0, 1, or 2.  0 means no match,
        //1 means match original password, 2 means match temp password.
        $loginSuccess = password_verify($passwd, $result["Password"]);
```

Figure 6. Password Verification Code

4.2 Password policy

To ensure that a strong password policy was in effect, the security teams obtained the current accepted password policy from the client, and then ensured the same standards were applied to the capstone project. The following password requirements were implemented into the project:

- Passwords must be at least 8 characters in length.
- Passwords must have at least one uppercase letter.
- Passwords must have at least one number.
- Passwords must be changed every 90 days.
- Passwords cannot be reused within a 365-day period.
- Accounts that are inactive for more than 60 days will be disabled for 30 days then deleted.
- Accounts are locked out after 3 unsuccessful login attempts and the application administrator must be contacted for the account to be unlocked.

The account lockout policy in which the user account is locked after three unsuccessful login attempts was implemented to defend against brute force attacks in which an unauthorized user can repeatedly attempt to gain access to an account, while attempting to guess the password. If this occurs, the only way to unlock the account is to contact the administrator and have him/her unlock it. Figure 7 shows the system's lockout popup box which appears when a user is locked out of the system.

Figure 7. Lockout Popup Box

If a user forgets his or her password, the password reset policy could be used, but only before the account is locked out. To reset a password, the user must click on the password reset option and answer personalized security questions that were set up during account creation process. If the user cannot remember the answers to these security questions or the account is locked, he or she must contact the system administrator. This is to protect the system from malicious brute-force attacks against the security questions. A sample set of personalized security questions is shown in Figure 8.

Figure 8. Personalized Security Questions

5 Results

As mentioned previously, many of the vulnerabilities were avoided or mitigated during the development phase. As the programmers were developing their code, the security team would manually review the code for known vulnerabilities using the OWASP Top Ten as a guide. Vulnerabilities that existed, such as missing password policy enforcement, secure password storage, and sensitive data exposure through browser caching of web form data, were all found and reported on a daily basis to the development team. The development team would then correct the vulnerabilities by adding or correcting programming coding and then upload the new code to a shared folder on Dropbox.com. The following day, the security team would manually recheck the code again and also check it with scans using the various tools. This process would repeat throughout the semester until the application was complete. The week before the final due date, the security team performed one final test of the application via manual code review and automated scans.

This final test found no critical vulnerabilities. However, there were vulnerabilities that were considered low or medium, but were either infrastructure related vulnerabilities, or downgraded due to the nature of the implementation of the application for the client. The application is being implemented in an internal intranet environment only. Only authenticated domain users will have access to the network and the application will not have any remote or internet access. Using the OWASP Top Ten as a guide, a sample of some of the final results are mentioned in the following sections.

5.1 A1 - Injection

After testing, the teams found no vulnerabilities that would offer the potential for injections. The code is protected from injections because of input validation and using prebuilt SQL queries. All queries are prebuilt, such as the one shown in Figure 9, and the application does not allow user's data to be used in building any queries. The user's data is then used to filter the results of the query in the middleware and only return the parameter the user requested.

```
function queryVendor()
{
 $con = connect();
 $sql = "SELECT * FROM vendors";
 $result = $con->query($sql);
```

Figure 9. Example of Prebuilt Query

A user's inputted data that is being stored in the database is first filtered through different PHP functions to eliminate the ability of the users to inject computer code into the database. The PHP built-in functions called htmlspecialchars, trim, and stripslashes are used to eliminate the ability of an attacker to perform injection attacks on the database. Each webpage and data field has been tested to ensure that data entered into each data field on every webpage is sanitized through input validation methods. An example of code is provided in Figure 10 that shows how input data is ran through the "test_input" function.

```
function test_input($data)
{
 $data = trim($data);
 $data = stripslashes($data);
 $data = htmlspecialchars($data);
 return $data;
```

Figure 10. Test Input Function

The Trim function is a PHP function that replaced any whitespace with a single space. The stripslashes() function removed slashes from the user data. The PHP function htmlspecialchars() removed special characters that were used in programing languages and replaced it with a different symbol that could not be used in attacks. For example, the ‘<' (less than) symbol becomes '<' and the ‘>' (greater than) symbol becomes '>'. These symbols were then eliminated from user input data because they were used in a majority of computer coding languages. An example of how to call the PHP function is shown in Figure 11.

```
if(is_numeric($_POST['lotNum']))
{
 (int)$_POST['lotNum'];
 $lotNum = $_POST['lotNum'];
 $_SESSION['lotNum'] = $lotNum;
 $lotNum = test_input($lotNum);
}
```

Figure 11. Example of Calling test_input Function in the Code for addForm.php

5.2 A2 - Broken authentication and session management

The teams tested for exposed session IDs in the URL, such as URL rewriting. The result was negative as the URLs with session IDs were hidden and not exposed in the browser, as shown in Figure 12.

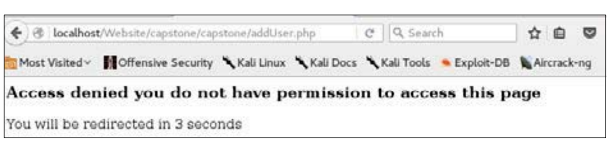


Figure 12. URL is Hidden

Furthermore, the user is forced to re-authenticate or prove that they are the user when trying to access a webpage or asset. This is due to the previous session ID being destroyed after login, as shown in Figure 13.

```
    //Destroy any previous session data before creating new
session.

    //If cookie is set, remove PHPSESSID
    if ( isset( $_COOKIE[session_name()] ) ){
        setcookie( session_name(), "", time()-3600, "/" );
    }

    //overwrite current session with blank array to remove
previous data
    $_SESSION = array();

    //start new session
    session_start();
```

Figure 13. Session ID Being Destroyed

5.3 A4 - Insecure direct object references

No vulnerabilities were found in the testing of direct references to restricted resources. The PHP code verifies who has permissions to access each resource. The pages use the “POST” method and do not use a “GET” method (The GET method would show parameters in the URL). Also, every page checks the user permissions and if the permissions are incorrect, it returns the user to their main page.

```
if(isset($_POST["addUser"]))
{
        addAccount();
}
else
{
        header("Refresh: 3; url=main.php");
        echo '<h3>Access denied you do not have permission to access this page</h3>';
        echo 'You will be redirected in 3 seconds';
        exit(); // Quit the script.
}
?>
```

Figure 14. Verification of Who Can Access the addUser Function

Figure 14 shows the code that verifies who can access the addUser function. If the user does not have the permission, then access is denied and the user is returned to the main screen.

5.4 A8 - Cross-Site Request Forgery (CSRF)

As shown previously in Figure 14, the code verifies permissions for each page view. If the user does not have permissions to the page, they are redirected back to their main page.

The result of testing for CSRFs was negative. This is because the unique token was in a hidden field, and therefore, was not included in the URL, as shown in Figure 15.

Figure 15. Token Not Included in URL

6 Recommendations

Recommendations to the client were all infrastructure-related and should be integrated as part of a total IT security policy. The implementation of SSL encryption to be used between client browser and web server should be implemented immediately. Error messages from web servers or the database server should be limited, or perhaps the client could implement custom error pages.

Logging and auditing all access to the application and database should be performed, as well as restricting physical access to the servers. ACLs (access control lists) can be implemented to control access at the network level to the front and back end server. Implementation of a software configuration management plan should be started.

Other recommendations were to remove unnecessary features or services, disable unneeded accounts, change default account passwords, and grant users least number of privileges possible. Finally, keeping PHP versions up to date to ensure that any vulnerabilities that existed in older versions are mitigated would be an effective way to prevent future issues with the system.

7 Summary

The Spring 2016 Capstone IT 4444 course at Cameron University gave the students the chance to gain experience, develop skills, and apply their abilities in a simulated real-world software development company with a real client. Furthermore, the security teams had the chance to experience what conducting security testing is like in a business environment by using their knowledge and skills to test the Bone and Implant Tracking System. By utilizing testing tools available to them, the security teams aided in the development of a secure system in cooperation with the computer science teams. The tests students conducted helped identify vulnerabilities in the system and taught the teams significant skills that could only be experienced in this type of learning environment.

8 References

[1] M. Estep, K.D. Smith, C. Zhao, A. Johari, and K.D. Smith, "Creating synergy between undergraduate senior computing research and applied capstone courses." Proceedings of the 2015 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS'15), Las Vegas NV, July 27-30, 2015.

[2] C. Zhao, M. Estep, and K.D. Smith, "Teaching software engineering through a real-world project: A new approach." Proceedings of the 2013 International Conference on Software Engineering Research & Practice (SERP'13), Las Vegas, NV, July 22-25, 2013.

[3] K.D. Smith, M. Estep, C. Zhao, F. Moinian, and A. Johari, "Teaching case combined discipline capstone teams: Using service learning to provide a business solution." Issues in Information Systems journal, vol. 15, no. 2, p.8-13, 2014.

[4] OWASP, "OWASP top ten project." Available: https://www.owasp.org/index.php/Top10#OWASP_Top_10_for_2013. [Accessed: May 12, 2016].

[5] Oracle, "Virtual Box." Available: https://www.virtualbox.org. [Accessed: May 12, 2016].

[6] J. Muniz and Aamir Lakhani, "Web Penetration Testing with Kali Linux." Birmingham-Mumbai: Packt Publishing, 2013.

[7] Offensive Security, "What is Kali Linux?" Available: http://docs.kali.org/introduction/what-is-kali-linux. [Accessed: May 12, 2016].

[8] MySQL, "Download MySQL community server." Available: http://dev.mysql.com/downloads/mysql/5.7.html. [Accessed: May 12, 2016].

[9] phpMyAdmin, "Bringing MySQL to the web." Available: https://www.phpmyadmin.net. [Accessed: May 12, 2016].

[10] Tenable Network Security, "Nessus." Available: http://www.tenable.com/products/nessus-vulnerability-scanner. [Accessed: May 16, 2016].

[11] B. Guimaraes and M. Stampar, "SQLMap." Available: http://sqlmap.org/. [Accessed: May 16, 2016].

[12] OWASP, “OWASP Zed Attack Proxy project.” Available: https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project#tab=Main. [Accessed: May 16, 2016].

[13] Security Compass, "XSS Me." Available: https://addons.mozilla.org/en-US/firefox/addon/xss-me/. [Accessed: May 16, 2016].

SESSION

PROGRAMMING AND SOFTWARE ENGINEERING COURSES AND RELATED ISSUES

Chair(s)

TBA

Engaging CS Students in Co-Curricular Undergraduate Research

Cynthia Y. Lester[1] and Olatide Omojaro[2]
[1]Mathematics, Computer Science, Engineering, Perimeter College, Georgia State University, Atlanta, GA, USA
[2]School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA

Abstract – *With the rapid growth and need for applications and platforms, today's technology companies are looking for employees who possess knowledge and skills in software development. However, for students attending two-year institutions where software engineering courses are not typically offered and internships are even more difficult to obtain, gaining this knowledge and/or experience is even more difficult. The aim of this paper is to present a small scale project in which students explored the software development process. While in and of itself this is not unique, the setting, software development strategy, challenges and lessons learned make the project worth exploring. The paper engages educators who teach in the two-year institution space and discusses a framework that could be used to engage students beyond the traditional classroom environment, giving them a co-curricular experience needed to transition to a 4-year institution, obtain internships or industry work experience in software development.*

Keywords – Agile method, Computer Science, Extreme programming, Software engineering, Undergraduate education

1 Introduction

The United States Department of Commerce, Economics and Statistics Administration in its July 2011 report stated that science, technology, engineering, and mathematics (STEM) occupations are projected to grow by 17.0 percent between 2008 and 2018, compared to 9.8 percent growth for non-STEM occupations [1]. Additionally, STEM workers command higher wages, earning 26 percent more than their non-STEM counterparts. Moreover, STEM degree holders enjoy higher salaries, regardless of whether they are working in STEM or not [1]. By 2018, the bulk of STEM jobs will be in Computing (71%) followed by traditional Engineering (16%), Physical Sciences (7%), Life Sciences (4%) and Mathematics (2%) [2]. These statistics provide an impetus for more students to choose STEM areas as fields of study. Moreover, it has been reported that of the fastest growing jobs in the last five years, seven of the top ten are technology positions that involve software development and data mining. The U.S. News and World Report ranks the job of Software Developer as number 3 of the top 100 best jobs in 2015 [3]. Yet, it is often difficult for undergraduate students to gain experience with the software development process without an internship, co-op or taking a formal class in which processes are introduced and projects are developed. Furthermore, for students attending two-year institutions where software engineering courses are not typically offered and internships are even more difficult to obtain, gaining this knowledge and/or experience is even more difficult.

This paper focuses on the development of a small scale project aimed at giving beginning engineering and computer science students an exposure and experience with the software development process. While in and of itself the idea is not unique, the setting, software development strategy, challenges and lessons learned make the project an expository discourse on how to engage students at this level in an effort to provide them with necessary skills needed to transfer to a 4-year degree program or accept employment in the workforce. The paper is organized into the following sections. Section 2 introduces the project framework. Sections 3 and 4 present the project and implementation. Section 5 presents observations and next steps. Section 6 presents concluding thoughts.

2 Project Framework

2.1 Introduction

The basis for the work was the need to provide students with software development experience beyond what is traditionally taught during the first two years that students matriculate at a 2-year institution. Typically at a 2-year institution, students focus on core classes that can be transferred to a 4-year institution or gain skills that will allow them to enter the workforce immediately. In computer science these introductory courses may include a breadth-first course, programming and data structures courses. The curriculum is robust enough to propel students to a 4-year degree while also flexible enough to allow students to work in a variety of professions, as encouraged in the Computer Science Curricula 2013 Final Report [4]. However, more advanced topics are not covered leaving some students desiring a more in-depth look into computer science prior to graduating or transferring from their 2-year institution. Hence the need to create co-curricular research experiences for these students beyond the traditional classroom environment.

Students had approached the researcher about providing a more in-depth experience because either of their desire to transfer to a 4-year institution or to accept a position in industry. They thought that having a research experience in software engineering would strengthen their applications and make them more competitive against their counterparts. Further, research findings suggest that in the new academy, faculty-student collaborative research opportunities are important and can be especially consequential for students identified "at risk," including first-generation college students and minorities [5],[6]. Research experiences are effective avenues for allowing students to apply theoretical concepts learned in class. They provide students with the opportunity to explore solutions to real-world problems while encouraging them to hone their problem-solving skills, improve their analytical and critical thinking, practice the soft skills of communication and engage in teamwork. These experiences have been shown to impact student retention as well as influence students' decision to persist in and pursue STEM careers [5], [6]. Therefore, research experiences for undergraduate students, especially in STEM disciplines, are an essential instrument for developing the next generation of STEM professionals.

The researcher had previous experience in developing projects for undergraduate students and used this as a basis for project development [7-9]. However, previous projects were developed for senior level students who had taken courses in database management, operating systems, algorithms, etc. Consequently, the researcher used the work by Schilling and Sebern as a framework which describes how a senior level course was changed to a sophomore level course [10]. But the researcher was still challenged to develop a project in which students could gain experience in the software development process based only having had courses in Java programming who unlike the students in Schilling and Sebern's work had courses in Software Development and Data Structures. Furthermore, the project would not be offered as part of an existing course but as a co-curricular activity. There is a big push in higher education, especially at the 2-year level, to ensure that students are taking only courses that transfer and/or are part of the core curriculum. Consequently, elective courses or special offerings are not encouraged.

Taking these challenges into consideration the following questions were considered in designing the student engagement:

- How will project success be determined? How will student learning be assessed?
- What real-world project could students investigate?
- What requisite skills would a student need to successfully complete the project?
- How can the project be designed to be completed in one semester (sixteen weeks)?

Therefore, in an effort to answer the first questions, student learning outcomes and project content were developed which are presented in the next sections.

2.2 Student Learning Outcomes

The idea was to offer students a scaled-down version of a software engineering course. When initially designed, it was decided to provide a high level overview of software development rather than focus on specific processes. This would allow students to be exposed to the breadth of software development and gain broad-based knowledge that could be easily transferred. Consequently, the learning objectives developed were general and stated that at the end of the research experience, students should be able to:

- Describe in detail the software process
- Identify various software process models and determine which model should be used for project implementation
- Implement various phases of a software process

2.3 Project Activity Content

In an effort to introduce theoretical concepts that students would see in a software engineering course when they transferred, the following topics were introduced:

- Introduction to software engineering concepts
- Overview of software process models
- Introduction to requirements building
- The use of modeling
- Design and implementation
- Software testing and evolution

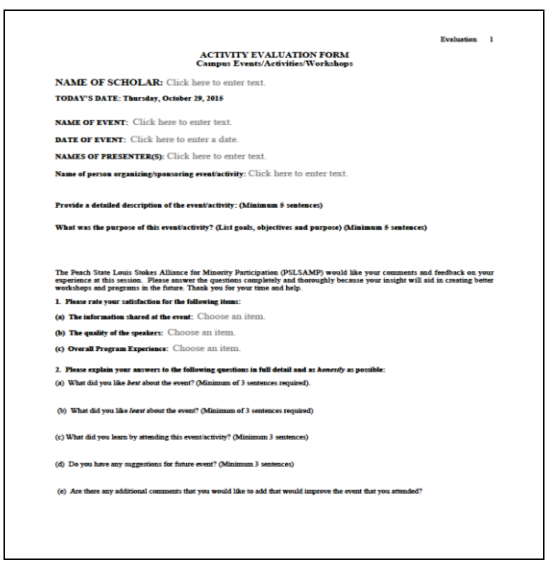
The next step included determining a practical project, in which students could engage, that had stakeholders and end users. Again, considering that this was a co-curricular activity and the idea was to introduce students to software development, not just have them code a project, this step was very intentional and resulted in conversations with colleagues about their software application needs. The next section presents the project that was chosen.

3 The Student Activity Survey System

3.1 Project Description

The Office of the STEM Initiatives at Georgia State University, Perimeter College has several programs in which STEM students can engage, some of which include the Mathematics Engineering and Science Achievement (MESA), the Education, Nurture Leadership in STEM (ENLISTEM), the Peach States Louis Stoke Alliance for Minority Participation (PSLSAMP), and the STEM Talent Expansion Program (STEP). The goal of these programs is to provide support for students so they can excel in math and science and graduate with a degree in science, engineering, computer science, and other math-based fields with the ultimate goal of increasing the number of students receiving associate or baccalaureate degrees in established or emerging fields within STEM. The programs also seek to diversify the STEM hiring pool through inclusive excellence. Students in the programs benefit from academic excellence workshops; academic advising/counseling; assistance with the transfer process to a 4-year institution; career advising; summer research/internship opportunities; and, linkages with student and professional organizations. The programs are externally-funded and require data collection for each of their respective granting agencies. Data collected from the student includes general demographic information and activity participation. Data from program coordinators include not only the data collected from the students but also academic information. The effectiveness of these programs for the students is assessed by collecting and analyzing feedback from qualitative surveys as well as their participation in the activities hosted by the programs. Overall effectiveness of the programs is assessed by meeting objectives and outcomes as a function of the respective grants.

Currently, the system in place for gathering student feedback and student participation is a paper-based form, depicted in Figure 1. The Activity Evaluation Form is completed by the student attending an activity and then submitted to the program coordinator or campus advisor. The submitted forms are verified and compiled by the respective program coordinators, who are then tasked with analyzing and summarizing the results. These results are compiled into one large report and shared with the funding agency and the institution's administration. Collection of forms, analysis of data and creating the summary report are labor intensive for the program directors and campus coordinators and present a time-commitment challenge to their primary role of teaching-faculty member. Therefore, the goal of project activities was to automate the process and alleviate the need for a paper-based form.



Evaluation 1

ACTIVITY EVALUATION FORM
Campus Events/Activities/Workshops

NAME OF SCHOLAR: Click here to enter text.

TODAY'S DATE: Thursday, October 29, 2015

NAME OF EVENT: Click here to enter text.

DATE OF EVENT: Click here to enter a date.

NAMES OF PRESENTER(S): Click here to enter text.

Name of person organizing/sponsoring event/activity: Click here to enter text.

Provide a detailed description of the event/activity: (Minimum 5 sentences)

What was the purpose of this event/activity? (List goals, objectives and purpose) (Minimum 5 sentences)

The Peach State Louis Stokes Alliance for Minority Participation (PSLSAMP) would like your comments and feedback on your experience at this session. Please answer the questions completely and thoroughly because your insight will aid in creating better workshops and programs in the future. Thank you for your time and help.

1. Please rate your satisfaction for the following items:

(a) The information shared at the event: Choose an item.

(b) The quality of the speakers: Choose an item.

(c) Overall Program Experience: Choose an item.

2. Please explain your answers to the following questions in full detail and as *honestly* as possible:

(a) What did you like *best* about the event? (Minimum of 3 sentences required).

(b) What did you like *least* about the event? (Minimum of 3 sentences required)

(c) What did you learn by attending this event/activity? (Minimum 3 sentences)

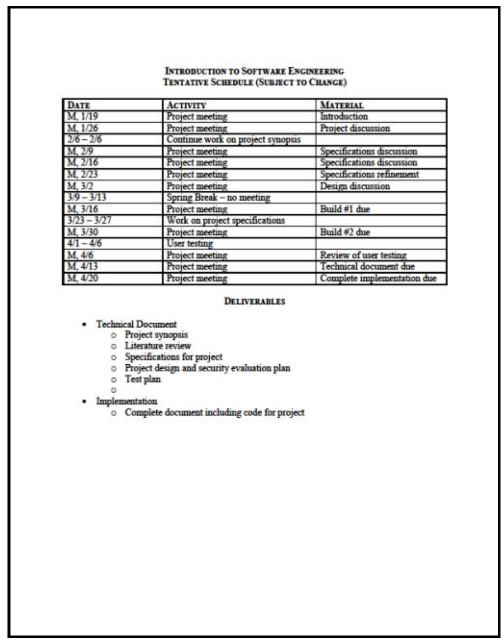
(d) Do you have any suggestions for future event? (Minimum 3 sentences)

(e) Are there any additional comments that you would like to add that would improve the event that you attended?

Figure 1. Activity Evaluation Form

3.2 Introduction to software development

Students were given the basic premise of the project. Since they were also participants in at least one of the programs supported by the Office of STEM Initiatives, they were also familiar with the paper-based form (having used it themselves) and the challenges that their usage placed on campus advisors and/or program coordinators. However, before meeting the stakeholders to determine user requirements, students were given material to read so that student learning outcomes could be met and so that students understood that software development included more than just programming [11-13]. Additionally, because students were not used to developing applications according to a development process, a tentative weekly outline was provided which included meeting dates and deliverables. Figure 2 shows the weekly outline.



INTRODUCTION TO SOFTWARE ENGINEERING
TENTATIVE SCHEDULE (SUBJECT TO CHANGE)

DATE	ACTIVITY	MATERIAL
M, 1/19	Project meeting	Introduction
M, 1/26	Project meeting	Project discussion
2/6 – 2/6	Continue work on project synopsis	
M, 2/9	Project meeting	Specifications discussion
M, 2/16	Project meeting	Specifications discussion
M, 2/23	Project meeting	Specifications refinement
M, 3/2	Project meeting	Design discussion
3/9 – 3/13	Spring Break – no meeting	
M, 3/16	Project meeting	Build #1 due
3/23 – 3/27	Work on project specifications	
M, 3/30	Project meeting	Build #2 due
4/1 – 4/6	User testing	
M, 4/6	Project meeting	Review of user testing
M, 4/13	Project meeting	Technical document due
M, 4/20	Project meeting	Complete implementation due

DELIVERABLES

- Technical Document
  - Project synopsis
  - Literature review
  - Specifications for project
  - Project design and security evaluation plan
  - Test plan
  -
- Implementation
  - Complete document including code for project

Figure 2. Weekly Outline

The first meeting included an introduction to software engineering and its historical connection with computer science and engineering [14]. Students were encouraged to familiarize themselves with technical terms and to start thinking about the software methodologies and the most common activities that they would be required to engage.

The most fundamental activities that are common among all software processes to which students were introduced included [14]:

- *Software specification* – the functionality of the system and constraints imposed on system operations are identified and detailed
- *Software design and implementation* – the software is produced according to the specifications

- *Software validation* – the software is checked to ensure that it meets its specifications and provides the level of functionality as required by the user
- *Software evolution* – the software changes to meet the changing needs of the customer

Students were reminded that the activities that formulate this view of software engineering came from a community that was responsible for developing large software systems that had a long life span. Additionally, the teams that used this methodology were typically large teams with members sometimes geographically separated and working on software projects for long periods of time [14]. Therefore, software development methodologies that resulted from this view of software engineering were often termed as "heavyweight" processes because they were plan-driven and involved overhead that dominated the software process [14]. Since the student participants were co-located and the software application to be developed was not considered large-scale, additional methodologies were explored.

The next section presents an overview of an alternative to heavyweight processes, agile development. Agile methods were introduced during the second and third week and extreme programming ultimately became the strategy used for the project implementation.

3.3 Agile Methods

In an effort to address the dissatisfaction that the heavyweight approaches to software engineering brought to small and medium-sized businesses and their system development, in the 1990s a new approach was introduced termed, "agile methods." Agile processes are stated to be a family of software development methodologies in which software is produced in short releases and iterations, allowing for greater change to occur during the design [14]. A typical iteration or sprint is anywhere from two to four weeks, but can vary. The agile methods allow for software development teams to focus on the software rather than the design and documentation [14]. The following list is stated to depict agile methods [14], [15]:

- *Short releases and iterations* - allow the work to be divided, thereby releasing the software to the customer as soon as possible and as often as possible
- *Incremental design* – the design is not completed initially, but is improved upon when more knowledge is acquired throughout the process
- *User involvement* – there is a high level of involvement with the user who provides continuous feedback
- *Minimal documentation* – source code is well documented and well-structured
- *Informal communication* – communication is maintained but not through formal documents
- *Change* – presume that the system will evolve and find a way to work with changing requirements and environments

More specifically, the agile manifesto states [16]:

"We are uncovering better ways of developing software by doing it and helping others to do it. Through this work we have come to value: Individuals and interaction over processes and tools. Working software over comprehensive documentation. Customer collaboration over contract negotiation. Responding to change over following a plan. That is, while there is value in the items on the right, we value the items on the left more."

While agile methods are considered as lightweight processes as compared to their predecessors, it has been stated that it is sometimes difficult especially after software delivery to keep the customer involved in the process [14]. Moreover, for extremely small software projects, the customer and the user may be one in the same, further complicating the development process.

While there are many well-known agile methods, Crystal, Adaptive Software Development, Feature Driven Development, it was decided to investigate further the use of extreme programming (XP) to determine its feasibility for the small-scale project development. The next section includes a high-level overview of XP and how it was used to facilitate project development.

3.4 Extreme Programming

Extreme programming (XP) is probably one of the best known and most widely used agile methods [17], [18]. It was originally coined by Beck because the method was developed as an alternative to legacy development models and pushed customer involvement to "extreme" levels [17]. It was also designed to address the needs of software development by small teams who faced changing requirements and system environments. Other characteristics of XP include team participants program in pairs, there is unit testing of all code, application features are not developed until they are actually needed, simplicity in code, constant communication with the customer and among the programming team [14]. Figure 3 illustrates the XP release cycle that students followed [14].

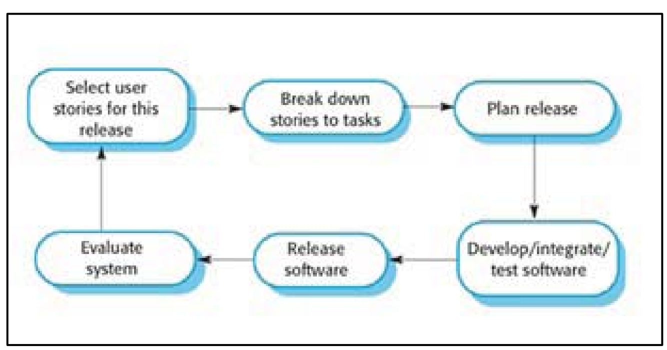


Figure 3. XP Release Cycle

4 Implementation

4.1 User Stories

Stakeholders were interviewed to ascertain requirements. Figures 4-6 are examples of the user stories that were discussed and created.

Collect the information
Students should be able to input participation information into the application inclusive of name, name of the event, date, name of program organizing/sponsoring event/activity, provide a detailed description of the event/activity and what was the purpose of this event/activity.

Figure 4. Example of User Story Card

Search for Data
Program director should be able to search for activity information based on a participant's name.

Figure 5. Example of User Story Card

Generate a report
Program Director or campus coordinator wants to generate a report based on the activity on a given date to include the number of students who participated.

Figure 6. Example of User Story Card

4.2 Implementation Strategy

The next steps included breaking down user stories into tasks and investigating strategies that could be used for implementation of the tasks that resulted in the development, integration and testing of the software. Some criteria that were utilized in the examination of a viable implementation strategy included familiarity and ease of use of the programming language, the testing and deployment environment, and compatibility with the institution's hosting servers.

The first strategy tried was an implementation of a PHP user interface, logic and database backend. This implementation was chosen because PHP is a general-purpose server-side scripting language that follows the object-oriented programming (OOP) with which students were familiar, having been exposed to OOP in a previous class. Further, PHP is specifically suited for web development, running on web servers and could be used with relational database management systems. Another advantage of the PHP implementation was the embedding of logic handling PHP code with user interface handling HTML code. However, after spending time developing the application for the first release, it was determined that the strategy was not working because there were unanticipated challenges in using the programming language to develop an easy to use front-end that could handle the complex and tedious data interactions needed. So, another strategy was considered and attempted.

The second strategy was an implementation of a Java Graphical User Interface and logic on top of a Microsoft Access (MS-Access) relational database backend. The implementation of the front-end was eased by using a series of graphical components, such as menu buttons, combo-boxes, multi-select lists, date pickers, and sliders. Furthermore, the connection layer between the graphical user interface front-end and the MS-Access relational database backend was implemented in Java. The use of this implementation strategy proved more successful yielding a finished product on a stand-alone machine. However, there once again unanticipated challenges when the application was deployed on the web server.

Consequently, a third implementation strategy was investigated which used the ASP.NET platform. Implementing the application as an ASP.NET Model-View-Controller (MVC) 5 application provided a similar development environment as the Java environment. Utilizing this strategy led to successful deployment of the application. Figure 7 shows the final theoretical framework developed and deployed.

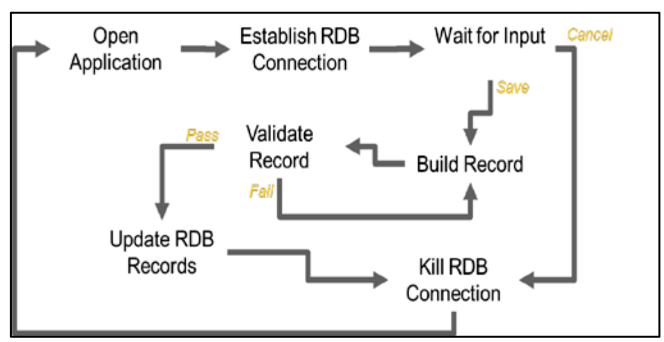


Figure 7. Theoretical Framework

4.3 Testing

According to Sommerville, XP places more emphasis than other methods on the testing process and that system testing is essential to XP [14]. He further states that the key features of testing in XP are [14]:

- Test-first development
- Incremental test development from scenarios
- User involvement in the test development and validation
- The use of automated test harnesses

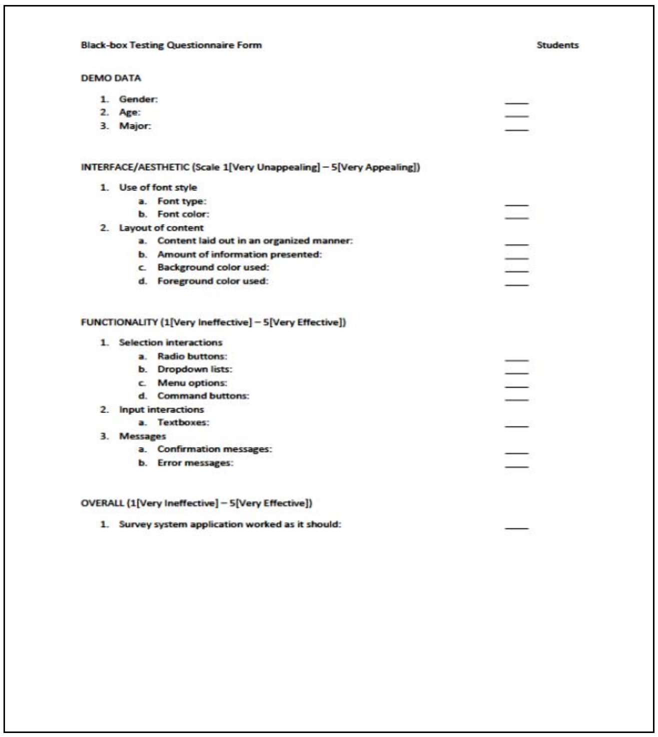
Using these key features as a basis for testing the application, the test cases were developed based on user stories. Automated testing was not utilized. However, unit

ISBN: 1-60132-435-9, CSREA Press ©

testing and user testing were used. Figure 8 illustrates an example of a task card developed from a user story while figure 9 is a simple questionnaire users were asked to complete as it relates to the usability of the application.

Test 1: Identification number
Input: String of characters to represent a student's identification number **Tests:** Check that only numbers can be entered Check if the string to too short or too long **Output:** Error message if digits are not entered or if string is to too short or too long.

Figure 8. Task Card


Black-box Testing Questionnaire Form   Students

DEMO DATA

1. Gender: ___
2. Age: ___
3. Major: ___

INTERFACE/AESTHETIC (Scale 1[Very Unappealing] – 5[Very Appealing])

1. Use of font style
   a. Font type: ___
   b. Font color: ___
2. Layout of content
   a. Content laid out in an organized manner: ___
   b. Amount of information presented: ___
   c. Background color used: ___
   d. Foreground color used: ___

FUNCTIONALITY (1[Very Ineffective] – 5[Very Effective])

1. Selection interactions
   a. Radio buttons: ___
   b. Dropdown lists: ___
   c. Menu options: ___
   d. Command buttons: ___
2. Input interactions
   a. Textboxes: ___
3. Messages
   a. Confirmation messages: ___
   b. Error messages: ___

OVERALL (1[Very Ineffective] – 5[Very Effective])

1. Survey system application worked as it should: ___

Figure 9. User Testing Survey

5 Results and Future Work

The purpose of the paper was to describe the development of a small-scale project aimed at giving beginning engineering and computer science students exposure to the software development process. As previously noted, while in and of itself the research is not unique, the setting, and software development strategy of the project set the stage for a discussion on how to engage students at this level in co-curricular research projects that provides them a theoretical framework they can transfer to a 4-year degree program or skills needed to accept employment in the workforce.

It must be noted that there are numerous challenges with the work. The first challenge was introducing the theoretical concepts in a setting that was outside a normal classroom environment. Students had only been introduced to computer science through the typical first courses in the discipline which included a breadth-first course and two courses in Java programming. While these courses introduced the software development paradigm, unlike upper-level division courses, there is no connectedness between topics. Consequently, several weeks were spent on an introduction to software engineering, terminology, and "getting students to understand that development is more than just programming" before project activities could begin.

Another challenge was finding the right solution for implementation. Students were excited to take on the challenge of learning PHP because of its popularity and widespread use. After doing some research, they intended to use XAMPP server solution package as the testing environment, specifically the Apache HTTP Server. However, after spending a release cycle with no tangible outcome, the team abandoned this idea for another solution that better utilized their skill set. While this was a valuable learning experience, time to complete project activities became an obstacle.

The end result was an application that contained the majority of the features that the stakeholders had initially expressed. There were some tasks that remained incomplete, specifically report generation. However, the outcome as stated was not for students to merely "code a project" but to walk away from the research experience being able to describe in detail a software process; to identify various software process models and determine which model should be used for project implementation; and, to implement various phases of a software process.

In discussions with the team leader, it was revealed that the there was a feeling that the "best work" had not been done because the entire application was not completed by the deadline. However, it was explicitly expressed by the researcher that the purpose of project activities was not necessarily to deliver a completed project but that research is undertaken to increase knowledge, which they agreed that by engaging in project activities had done.

Future work includes redesigning the research experience for students. As noted by Schilling and Sebern, student comprehension and outcome obtainment is essential [10]. While without question, students were engaged and had the fundamental skills to be successful, the type of project chosen was too complex. It may have been more realistic an outcome to implement the application's front-end and then in future releases consider the database and report generation. Additionally, the researcher will give more thought on how to better introduce the theoretical concepts needed before project activities begin. Students need to be provided with a basic foundation but attempting to introduce agile methods within the context of software development and software engineering was too intensive and time consuming.

6 Summary

In closing, as the global market place continues to become increasingly competitive, students are attempting to find different ways in which to differentiate themselves from the crowd. With the fastest growing jobs in the last five years involving software development and data mining, students are clamoring for opportunities to gain experience and enhance their skills in software development. However, it is often difficult for undergraduate students to gain experience with the software development process without an internship, co-op or taking a formal class. Moreover, for students attending 2-year institutions where software engineering courses are not typically offered and/or special topics courses/independent study do not exist, gaining this knowledge and/or experiences becomes even more difficult. Consequently, it is left to educators to find creative ways to engage students who want these experiences as well as those who may not yet have the requisite skills needed for a traditional internship or National Science Foundation sponsored research experiences for undergraduate students offered at 4-year institutions. The author hopes to bring forward this conversation on how to better prepare 2-year students for jobs in the software arena while also providing them with the theoretical foundation needed for transfer to a senior college.

7 Acknowledgements

The author wishes to thank Olatide Omojaro for serving as the leader on this project.

8 References

[1] D. Langdon, G. McKittrick, D. Beede, B. Khan and M.Doms. "STEM: Good Jobs and for the Future," 2011. U.S. Department of Commerce, Economics and Statistics Administration. http://www.esa.doc.gov/sites/default/files/stemfinalyjuly14_1.pdf. [Retrieved: June, 2016].

[2] R. Munce and E. Fraser, 2012. "Where are the STEM Students? What are their Career Interests? Where are the STEM Jobs?" https://www.stemconnector.org/sites/default/files/store/STEM-Students-STEM-Jobs-ExecutiveSummary.pdf. [Retrieved: October, 2015].

[3] http://money.usnews.com/careers/best-jobs/rankings/the-100-best-jobs [Retrieved: June, 2016].

[4] Computer Science Curricula 2013 (CS2013) http://www.acm.org/education/CS2013-final-report.pdf. [Retrieved March, 2016].

[5] B.A. Nagda, S.R.Gregerman, J. Jonides, W. von Hippel, W. and J.S. Lerner, 1998. "Undergraduate Student-Faculty Research Partnerships Affect Student Retention," *Rev. Higher Educ*., Vol. 22, pp. 55-72.

[6] M. Boylan. 2006. "The Impact of Undergraduate Research Experiences on Student Intellectual Growth, Affective Development, and Interest in Doing Graduate Work in STEM: A review of the empirical literature," *Doctoral Education and the Faculty of the Future,* Cornell University, Ithaca, NY. http://www.ilr.cornell.edu/cheri/conferences/doctoralEducation.html. [Retrieved: June, 2016]

[7] C. Lester. 2008. *An innovative approach to teaching an undergraduate software engineering course.* Proceedings of the International Conference on Software Engineering Advances. IEEE Computer Society Press.

[8] C. Lester. 2007. *Encouraging Teamwork in an Undergraduate Software Engineering Course.* Proceedings of the International Conference on Software Engineering Advances. IEEE Computer Society Press.

[9] C. Lester. 2010. *Training undergraduate students in user-centered design*. Proceedings of the International Conference on Advances in Computer-Human Interaction. IEEE Computer Society Press.

[10] W.W. Schilling and M.J. Sebern, M. J. 2012. Teaching Software Engineering: An Active Learning Approach Paper presented at 2012 ASEE Annual Conference, San Antonio, Texas. https://peer.asee.org/22014. [Retrieved: March, 2016].

[11] Van Vliet, P. J. A. and Pietron, L. R. 2006. "*Information Systems Development education in the Real World -- A Project Methodology and Assessment*," Journal of *Information Systems Education* (17:3)

[12] M. Rizwan and J. Qureshi. 2012. "Agile software development methodology for medium and large projects," in *Software, IET*, vol.6, no.4, pp.358-363.

[13] M. Stocia, M. Mircea, and B. Ggilic-Micu. 2013. "Software Development: Agile vs. Traditional" Informatica Economică vol. 17, no. 4/2013.

[14] I. Sommerville. 2011. *Software Engineering 9th Ed.* Addison Wesley, 13:978-0-13-703515-1, Boston, MA.

[15] F. Tsui and O. Karam. 2011. *Essentials of Software Engineering 2nd Ed.* Jones and Bartlett Publishers, 13:978-0-7637-8634-5.

[16] http://www.agilemanifesto.org [Retrieved: June, 2016].

[17] K. Beck. 1999. Extreme programming explained: Embrace the change. Addison Wesley.

[18] R. Jefferies, A. Anderson and C. Hendrickson. 2000. Extreme programming installed. In: The XP Series. Addison Wesley.

Programming in a Virtual Reality Environment

M. R. Zargham and **Bharat Kamsani**
Department of Computer Science
University of Dayton
Dayton, Oh USA

Abstract - *This paper represents a new way of teaching an introductory programming course. Students use a simple environment, generated with Unity, to write a code for designing a roller-coaster. As they code, they see the development of their roller-coaster in a 3D environment step by step. Once they have completed the design of their roller-coaster, they can wear Oculus to ride on their design in a virtual environment. Students find such an environment more interesting and creative than a typical computer science programing course environment. Students write codes for creating a virtual environment and are excited to do changes as necessary. They make every effort to do a good project while having fun and learning coding.*

Keywords: Oculus; Roller-Coaster; Virtual Reality; Computer Science Education; Programming.

1. Introduction

We have created an environment in which students can write simple codes consisting of several statements to create a roller-coaster of their choice. In general, the idea is to teach the introductory programming class in such a way that follows project-driven learning process and encourages students to develop problem solving and teamwork skills while fostering creativity and logic. The goal is to not only provide students with some "programming maturity," but to also engage them with existing projects related to their interest. At present we are emphasizing on one project (roller-coaster). In future we will be adding more projects such as: car racing, 3-D store, and games.

Once students have learned the basics of programming through various design projects, they will be encouraged to design games in an environment such as Unity by learning more programming statements and techniques.

To teach a programming course in this way, will attract students with diverse educational backgrounds to form teams to tackle various problems. Beside students in computer science, electrical/computer engineering and business, students from other disciplines such as mathematics, physics, chemistry, biology, communication, and arts are encouraged to enroll and participate.

2. Roller-coaster Environment

We have used Unity game development software package, to design an environment that the user can program. Figure 1 represent the starting scene when there is no roller-coaster. The small window on the left-top corner is used by the user to enter his/her code for building the roller-coaster. This window takes data to configure the roller-coaster. It takes care of graphical user interface with a text box, and four buttons. Textbox is used to get code from user. There are four buttons are for generating track, add new track, start animation and set animation in loop.

Figure 1. Initial scene.

Students can use the environment to design a 3D modeling of a roller-coaster based on their choice. At present, we have implemented only few statements to work with; the syntax of these statements are given below.

Declaring variable:
var variable_name;
var variable_name = constant;

Example:
var i; //creates a new variable *i* with value 0.
var limit=10; //creates a new variable *limit* with value 10.

Generating track:
track (length, direction, direction_angle, self-rotation_angle, bend_angle);
(or)
track length direction dirction_angle self-rotation_angle bend_angle;

In the above **track** statement, the length indicates the size of the track; the direction could be **forward**, **up**, **down**, **left**, **right**; the angle refers to the degree of track-turn indicated by the direction parameter; the self-rotation indicates the angle of track rotation in respect to its initial position; the bend angle represents the angle of twist of the track with some displacement.

Example:
track 10 up 45 0 0; //creates a track of length 10 going in upward direction with angle of 45.
track 100 left 30 0 10; // creates a track of length 100 turning left 30 degree and bending at the same time for 10 degree.

Generating random numbers:
generate variable_name;
generate variable_name start_value end_value;

Example:
generate i; //creates a new variable i with a random number between 0 and 100.
var j;
generate j; //initialize variable j to random value between 0 and 100.
generate i 1 10; //generates a number between 1 and 10 and stores in variable i.

For statement:
for (initiation, condition, increment/decrement expression)
{
Statements
}

Example:
for (i=0, i<10, i=i+2) {**track** 1 forward 0 0 0;}
//creates a track of length 5.
for (j=0, j>=10, j++) {**track** 1 forward 0 0 0;}
//creates a track of length 10.
for (k=0; k<10; k++)
{
track 1 forward 0 0 0; //creates a track of length 10.

}

If statement:
if (condition){ Statements}
else if (condition){ Statements}
else { Statements}
where condition could be:
variable condition_operator variable/constant
Condition_operator could be:
<,<=,>,>=,=, = =

Example:
var i;
generate i;
if(i<10){}
else if(i<50){}
else {}

While Statement:
while (condition){ Statements}

Example:
while (x > height)
{
track 10 forward 0 0 0;
}

Assignment:
variable = variable(or)constant (and/or) operator variable(or)constant;
Operator could be:+,-,*,/,%

Example:
var i;
i=i+10;
i=45*i;
i=j-k;

functions:
brake 10; //decreases speed by 10.
boost 30; //increases current velocity by 30.

object time(in seconds);
Object_name Object_time;
Object_name could be: **goblin**, **elf**, **pumpkin**, **bear**.
Object_time indicates the amount of time (in seconds) that the object appears.

Example:
pumpkin 10; //creates a pumpkin object at current track which lasts 10 seconds.

Displaying/Writing Value
display variable_name;
display i; // it writes the value of i into display window.

Based on the above statements, a possible code can be developed as below:

```
track 40 up 60 0 0;
track 40 down 120 0 0;
track 30 up 60 0 0;
track 30 forward  0 0 0;
var i;
generate j 1 400;
i = j%4;
if(i=0)   { pumpkin 20; }
else if(i=2)  {  goblin 20;}
else  { bear 20;  }
i = maximumheight;
j=currentheight;
display i j;
track 10 up 60 0 0;
while(j<i){
track 1 forward 0 0 0;
j=currentheight;
}
track 40 down 120 0 0;
for(i=0,i<3,i++){
track 60 up 360 0 30;
}
brake 50;
track 10 forward 0 0 0;
```

In the above statements, all statements must have either ';' at the end, or end of line character. Also, all braces must be put in separate lines.

The environment can be controlled by the following keys:

Space – play/pause animation.
A - Move left
D - Move right
W - Move up
S - Move down
Q - Zoom in
E - Zoom out
Mouse_Right_Key+Drag -rotate camera.

Figure 2 shows the design of one track. Figure 3 represents a complete code for a roller-coaster and its design. At any time the user can modify its code by deleting or inserting statements. Once the user is satisfied by the design, he/she will click the run button to start the animation. Figure 4 shows the start of the animation. If a Virtual Reality Oculus is connected to the computer, the users can actually feel that they are riding on their own roller-coaster design.

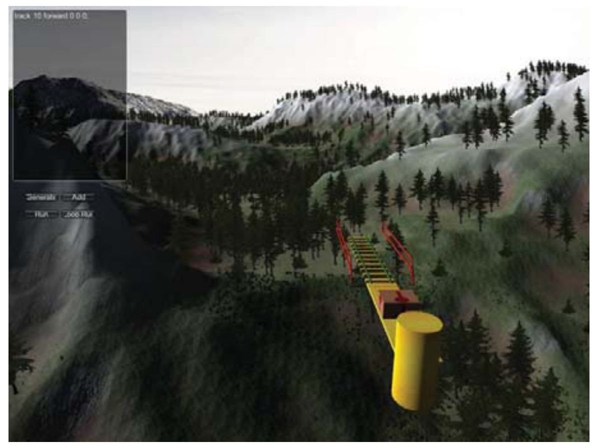


Figure 2. Execution of the first statement.

Figure 3: The code for a rollercoaster.

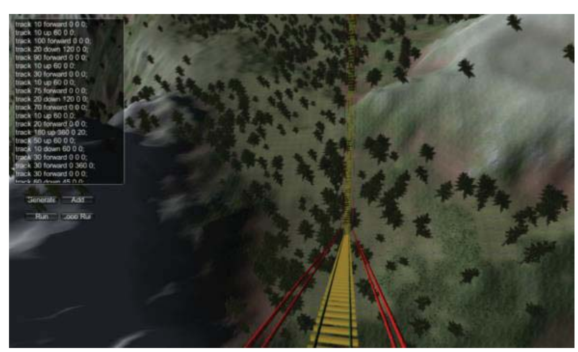


Figure 4. Start of animation.

Figure 5. A ten years old student is observing his design through Oculus.

The code editor allows students to add/delete a statement to/from their partial code very easily. The effect of such modification on their partially build roller-coaster, can be seen instantly by clicking on “wooden” or “Steel.” The user can use “for loop” to build a large track by repeating of a section of the track many times. The “while loop” can be used to design a track which reaches certain height. The “if” statement can be used to make a track dynamic by assigning it different types of objects.

3. Conclusions

In this paper, we have described a 3D environment that can be assigned in an introductory computer science course to retain students. We have implemented such an environment for 7 students, ages from 10 to 15, and preliminary results indicate that we have gained their interest. Students learned how to use for-loop and if statement and then used those statement to design their own roller-coater which they loved to ride on it. We are planning to design more environments for different objects. We are hoping that students in K to 12 also take advantages of this software package to learn coding in a fun way.

4. References

[1] Blum, L. 2001, “Women in Computer Science: The Carnegie Mellon Experience”, women@scs2.2., to appear in The Future of the University: The University of the Future.

[2] Klawe, M, 2013, “Increasing Female Participation in Computing: The Harvey Mudd College Story,” IEEE Computer, pp. 56-58, 2013.

Closed Labs in Programming Courses: A Review

Radu P. Mihail and Krishnendu Roy
Department of Mathematics and Computer Science
Valdosta State University
1500 N Patterson St.
Valdosta, GA
{rpmihail, kroy}@valdosta.edu

ABSTRACT

Recent trends show nationwide increases in undergraduate CS enrollment. Retention of students in CS programs continues to be an area with room for improvement. Research to increase retention focuses on modifications in the introductory computer science courses (CS1 and CS2). Many CS1/2 courses have a laboratory component. In these labs students typically write code, most often under the supervision of the instructor or a lab assistant. While labs are perceived by many CS instructors to be essential for introductory CS courses, we ask the following question: what characteristics of labs help increase the percentage of students with satisfactory learning outcomes in introductory CS courses? Based on the vast amount of evidence showing that retention is higher when students are more successful in the introductory level classes, we survey existing CS education literature that address CS1/2 closed labs. In this paper we present what we learned about common practices in the design and implementation of CS1/2 labs.

Keywords

CS1, CS2, Introductory Programming, Closed Laboratories, Design, Theory, Human Factors, Languages, Measurement

1. INTRODUCTION

The Association for Computing Machinery (ACM) released in 2013 the latest curriculum guidelines for undergraduate programs in computer science. In part because not all topics relevant to a computer scientist can be taught in a two or even a three course sequence, many departments have to make choices that involve trade-offs and potential loss of exposure to topics. This multidimensional design space leads to hot debates regarding the relative importance of a set of learning outcomes and sparse coverage due to lack of time.

In spite of this debate, the vast majority of undergraduate programs offer, at some point in their curriculum, one or more courses where concepts in computer science are taught using a programming language. In most cases, students are exposed for the first time to syntax and core concepts, and these courses are the first in a sequence of two or more, which we call introductory programming courses. Introductory programming courses typically have a laboratory component, where students can practice programming.

Lab components of introductory programming courses come in two main flavors: open and closed labs. Open labs offer students a space to work, mostly not constrained by time (i.e., open most of the day) and staffed by tutors or teaching assistants. The courses using open labs may encourage students to work on the homework in these labs, but attendance is not required. Closed laboratories consist of scheduled class time, supervised by an instructor or a teaching assistant. In these sessions, the students are given concrete instructions and their work is graded. The instructor is there to provide help in the form of direct instruction.

In this paper, we review existing literature in an attempt to summarize current practices in closed labs. We conjecture that retention is highly dependent on a constructive and positive experience with introductory programming courses served by closed labs. Although a vast amount of universities offer introductory programming courses, closed laboratory assignments and their learning outcomes are largely a matter of local choices.

More specifically, we ask the following questions:

1. What are the characteristics of an effective lab?
2. What are the assessment methodologies employed effective closed labs?
3. What types of lab design choices result in effective introductory programming closed labs?
4. What are the benefits and harms of contextualization (i.e., focusing on a specific application domain such as game development or software engineering)?
5. How do teams play a role in student learning?
6. How does the size of the university affect the use of labs?

In this paper we reviewed publications in the ACM Digital Library, IEEE Explore, Journal of Computer Science in Colleges, and Taylor and Francis CS Education Journal related to closed labs. In Sections $2-6$ we present a summary of research answering the questions posed above.

2. ASSESSING EFFECTIVE LABS

In this section we present details about how various closed lab related publications reported their assessment methodology.

Closed lab sessions can be measured with respect to retention, performance on exams and homework assignments and self reported measures. The literature that directly addresses lab efficacy is relatively sparse. Several studies include discussions on benefits of labs in CS1 courses. However, not many include rigorous assessment of student learning in closed labs. We believe (as also briefly alluded to in [31]) that one of the reasons assessment is difficult is simultaneous changes for all sections. Therefore when labs are introduced, there are no non-lab sections that can be used as a baseline for comparison. Researchers have to fall back on data from previous offerings of the course.

Titterton et al. [30] present their experience with offering a “lab centric” course. They describe common problems with the adoption a lab centric approach, such as shortage of facilities, staff and the determination of teaching credit. Kumar et al. [14] compare student learning and retention in a course with closed labs to that with open labs. Their result shows that there was no significant gain in student retention from the open lab to the closed lab course. Their research did show some improvement in test scores in the closed lab section, compared to the open labs. The authors also found similar results in the context of closed labs and online testing. These results were reported in [15].

Titterton et al. [31] used surveys and analyzed student performance in lab-centric and non-lab versions of the same course. They did this in two different courses (CS2 - Data Structures, and Computer Architecture). They found that lab-centric courses helped students increase exam scores. They also found conflicting evidence that students in lab-centric courses did worse on homework projects.

McCauley at al. [18] proposes an experimental design to evaluate the effectiveness of closed laboratories. This design accounted for reports that different instructors for lecture and lab make a difference. In a follow-up study by McCauley et al. [19], they reported no effect on learning outcomes given different lecture and lab instructors, but significant increased student satisfaction was observed when the lecture and lab were taught by the same instructor.

Soh et al. [26] measured the effects of closed labs on students' performance on placement exams, take home assignments, end of course exams, and attitude towards CS1 course. They included pre- and post-tests in their labs and analyzed student performance. They found students' lab performance was a significant predictor of exam outcomes as well as homework assignment scores. The lab performance was not a significant predictor of their placement exams. The same authors published subsequent results in Soh et al. [25] where they found positive correlation between lab performance and exams/homework-assignments.

Horton et al [10] compared and contrasted traditional and flipped CS1. They report better learning outcomes as measured by the final exam, but no changes in student attitudes or likelihood to pass was observed.

2.1 Conclusions on Lab Characteristics

In this section we reviewed the literature to answer the question: what are the assessment methodologies employed effective closed labs? We found no consensus on how to assess the effectiveness of closed labs. When formal assessment was done, student retention, their grades on exams, and end of course surveys were some of the assessment metrics. There was no discussion on whether any of these methods provide better insights over others. Many papers use anecdotal and experiential (here is what we tried and here is what happened) data to support the positive effects of their labs.

3. LAB DESIGN CONSIDERATIONS

The variety of curriculum choices in computer science programs result in a diverse set of structural designs for closed labs. We found that there is insufficient research to determine which types of lab designs result in effective closed labs. We conjecture that effectiveness of closed labs may be influened by local contraints (e.g., university resources and student-faculty ratio). In this section, we review historical uses of labs and the methods for evaluating lab design to support our finding.

Beginning around the late 80s to early 90s, when microcomputers became financially accessible to many universities, many researchers acknowledged that programming labs are useful, and presented their own recipes for success. Examples of early work are [3, 28]. Two decades later, works that address programming labs are similar, in that they rarely offer a comparison to other approaches. Unfortunately, there is no clear evidence that 20 years have produced any measurable progress in terms of student achievement following closed lab courses. One possible way to measure improvement over time is to evaluate designs by separating domain specific and instructional design choices.

A framework provides a conceptual model that allows the separation of domain specific and instructional design choices in the overall design and administration of labs. Separation of these choices is non-trivial. Consider the size of a CS department and the type of institution it is home to. Large departments in research institutions are more likely to hire an instructor who is responsible for large lectures and coordinating teaching assistants who administrate labs (open and/or closed). In small departments, introductory courses are typically taught in a rotating fashion by tenure-track faculty who have their own notions of optimal choices of content and instructional design.

Two decades ago, Waller [33] proposed a framework for the development and administration of lab materials in CS1 and CS2 courses. His system allowed for the incorporation of extant materials, but provided no explicit instructional design research. Kumar [14] compared open and closed labs with respect to student retention, performance and project completion. They found that closed lab did not improve retention or project-completion, but observed qualitative improvement in student learning for closed labs in the first part of the course.

Soh et al. [25] present a framework for closed laboratories where instructional design is separated from content design. They embed instructional design features as well as an infrastructure to keep track of resources over time, in light of constantly changing software development and information technology. Their proposed framework consists of a lab design phase, in which CS faculty and researchers discuss and select CS topics. After this discussion, labs are implemented as stand-alone modules, each with handouts, worksheets, and pre- and post-tests. Instructional research questions can be answered by making modifications at the implementation stage. Deployment of labs is done in collaboration with course and lab instructors, who also monitor the labs, suggest fixes to problems and provide additional help to students.

3.1 Lab Design Conclusions

We reviewed literature to answer the following question: what types of lab design choices result in effective introductory programming closed labs? There is insufficient evidence to determine which lab designs lead to effective closed labs. Closed labs have been designed to fit the specific needs of a local university and thus are unique to each university (or instructor). Furthermore, case studies of labs address and do a great job of discussing implementation choices, but lack a framework that separates instructional design and pedagogical research. We recommend that researchers use a design framework (such as described by Soh et al. [25]) when designing and evaluating labs. We then hope that a set of consensus of best practices can emerge using a hypothesis driven research methodology that separates content from delivery.

4. CONTEXTUALIZATION

The topics taught in introductory programming courses can easily be grounded in applications (e.g. video games) to make the learning experience more motivating and to inspire students early in the program. We refer to the use of such applications as contextualization. We review the literature and discuss approaches that use games and real-world scenarios. The literature indicates that contextualizing introductory programming courses is beneficial for retention and student learning. Furthermore, there is evidence that lab effectiveness in contextualized courses is increased, but generalizing is premature due to limited sample sizes.

4.1 Games

Most CS faculty were told by some of their students that the students' choice of major was driven by video games and desire to write their own. This has motivated work on creating game-centric introductory programming courses, where labs are framed around the various aspects of game development.

Many game-centric approaches have been proposed [2, 4, 20], and the ones specifically aimed at CS1/2 courses [16, 23, 11, 5, 29] tend to focus on the course as a whole, and not on the specifics of the laboratory experience. This body of literature presents evidence that supports the hypothesis that using games as contextualization improves student learning outcomes and retention.

Chen et al. [4] report their experience with a game-centric course. Their labs were incremental, as object oriented programming concepts were introduced as a means to deal with the growing complexity of implementing the functionality of a simple flying-shooting game. The labs were supported by a framework library called *Game* that handles the game loop, windowing, controls and displaying graphics. The authors discuss the potential pitfalls with such an approach. For example, students might spend too much time on art or they might reduce functionality of the game to meet their programming abilities. The format of their lab sessions was closed, three hour sessions with two student groups. The authors reported that the lab time was not sufficient, but support from TAs and office hours alleviated the problem.

4.2 Real-World

Real-world scenarios are another form of contextualization. We live in the age of of massive data collection and analysis. Real-world scenarios can motivate students by showing them how to use skills learned in the classroom to analyze data and solve current problems.

Efforts to make introductory programming more relevant include using data from other domains. For example, Anderson et al. [1] use publicly available data from oceanography, DNA, elections, etc. They reported success as defined by the ability of students in CS1 to do real-world data analysis, but no quantitiative data was given.

Reilly et al. [24] discuss the introduction of new CS1 labs based on real-world problems. They reported more students completing the new labs compared to their status-quo labs. The researchers believe students did better on exams, but have no way to confirm due to the experimental design.

4.3 Interactive Multimedia

Another form of contextualization consists of incorporating digital media (images, sounds, and video clips) manipulations in introductory programming courses. Media Computation, is perhaps the most widely adopted example of media focused contextualization. In Media Computation students manipulate image pixels to create various effects and filtering. They also work with video and sound editing through code. Guzdial [8] describes his work over ten years on a media-centric introductory programming course. We refer the interested reader to read [8] for a comprehensive guide to media computation related literature.

4.4 Conclusions about Benefits and Pitfalls of Contextualization

We reviewed the literature to answer the following question: what are the potential benefits and pitfalls of contextualization? Given the success of using contextualization in courses as a whole as we discussed above, we believe that contextualization in labs has many benefits. Perhaps the greatest benefit is increasing student motivation to learn computer programming as we discussed in 4.2. Furthermore, the literature on real-world contextualization found that more students completed labs when the labs incorporated real-world problems.

While using contextualization in labs has benefits, there are also pitfalls to consider. One possibility is that using contextualization could increase workload for instructors. Chen et al. found that students needed more TA support and office hours to overcome insufficient time to complete games during labs. When planning to use contextualization, instructors need to be able to provide more student support. Additionally, creating the game assignments or finding real-world data may increase the time needed to plan lesson plans for labs.

Another common concern with contextualization is whether students can easily bridge the gap between contextualized knowledge and more general decontextualized knowledge that they will acquire in various CS classes beyond CS1/2. Guzdial addressed this difficult question in [7].

5. TEAMS

Introductory programming courses often include active learning activities like pair programming, group assignments, peer instruction etc., with the goal of fostering students' engagement and participation. We expected similar approaches to be broadly adopted in closed labs, but found few studies that provided details about active learning activities in the context of closed labs. In this section we present our findings.

Perhaps the most common intervention used in closed labs is collaboration where students are asked to work on an assignment together. Pair programming, where one student is acting as the driver and the other as the navigator, is a common form of collaboration. Walker [32] shows examples of closed lab implementation with collaboration. He describes a lab-centric course where most of the classes were labs, not lectures.

Nagappan et al. [21] and Williams et al. [34] present details about the effectiveness of pair programming in CS1 labs. Their results show that pair programming has positive effects on students' self-sufficiency, performance on tests and projects, and retention. Porter et al. [22] discuss efficacy of pair programming in their open as well as closed labs and how that increased retention.

Titterton et al. [31] discuss gated collaboration. In gated collaboration, during a lab, a question is posed to students. After a student answers the question (usually online), they get to view other students' answers and discuss various answers.

5.1 Conclusions about Teams

We reviewed the literature to answer the following question: how do teams play a role in student learning? There is some evidence that teamwork plays an important role in lab efficacy. Pair programming is the most common approach to implement teamwork and collaborative learning. The Community of Inquiry framework (widely adoped in the education community), proposed by Garrison et al. [6] proposes three "presences" for effective learning: teaching, cognitive and social. Pair programming falls under the social presence by facilitating engagement with peers and discourse relevant to student learning. The relatively limited number of publications related specifically to closed-lab student-engagements interventions leads us believe that this will be an active research area in near future.

6. OTHER CONSIDERATIONS

Closed laboratory for introductory programming appears to be the most widely adopted style by universities across the country and world. Despite their popularity, few studies with solid instructional and research designs have been done to assess best practices in a data-driven manner. A decade ago, Soh et al. [25] proposed a design framework for labs that incorporates instructional research. Their follow-up work [27], has been cited 3 times (excluding self-citations). In much greater numbers, case studies seem to dominate the literature regarding closed labs.

We observed that a subset of recent work within the introductory programming realm report shifting focus toward more dynamic (i.e., more stimuli) teaching tools (e.g., game-centric). There is a growing number of studies outside of the CS education community that indicate benefits of serious game-based approaches (e.g., [17, 13, 12]). Within CS education, controlled trials rarely have large ($n \geq 100$) samples sizes. We believe game-centered approaches have great potential, but support from data is imperative for further progress.

We often encounter distracted students, who, in spite of our best efforts, lag behind others who are more focused. Anecdotally, attention span is reduced today. We have not found any work in the CS education literature that takes a critical look at how distracting factors (e.g., cell phones, social media) impact learning. This problem is most acute when both the lecture and closed laboratory are in a room full of social-media-capable machines. We believe we need to take a closer look at course time allotment and be aware of attention span. Hakimzadeh et al. [9] took an encouraging step in this direction by proposing short 10-15 minute lecture sessions followed by individual learning activities, group analysis, peer learning, pair programming and discussion.

We believe closed laboratories should primarily contribute to maximizing the percentage of students' with satisfactory learning outcomes. A multi-dimensional design space and a continually evolving toolbox makes it difficult to establish a set of best practices. We believe that evidence-based controlled trials will eventually converge to a set of best practices, invariant to teaching technologies, but we have much work to do.

7. WHAT DID WE LEARN?

The main motivation of this work is to identify the best practices for success in CS1/2 closed labs. The main question we posed was: what are the characteristics of an effective lab? We surveyed five aspects of closed labs to answer this broad question. They were broad design frameworks, assessment of effectiveness, role of teams, contextualization, and local constraints like university size.

We found that there is insufficient evidence to conclude about any specific lab design framework being superior. We found a few ways to design closed labs. Some of the frameworks embedded instructional design aspects, while others did not. Assessing the effectiveness of a specific closed lab design framework is challenging in itself. We do not believe comparing design frameworks to decide the superiority of a framework is practical.

Assessing effectiveness of closed lab is another area where we found limited results from evidence-based assessments. There is no standardization about what is considered effective and metrics for measuring effectiveness. While some papers measured student retention as a metric for success, others used qualitative data from surveys.

There is some consensus about positive effects of contextualization in closed labs. Research shows benefits of various types of contextualizations in CS1/2 closed labs. We believe we will continue to see more contextualized CS1/2 courses and labs in future which will branch out to newer domains like cyber security, mobile computing etc.

Just like contextualization, there is consensus about the role of team work in increasing student success in CS1/2 labs. Pair programming was shown to work in many studies with convincing quantitative data. Other team oriented interventions like gated collaboration have also been shown to work.

To summarize, we can conclude that incorporating contextualization and teamwork related activities helps CS1/2 closed labs in several metrics. While there is some research to design effective frameworks that can be easily adopted, there is no broadly adopted framework. Consensus guidelines for assessing the success of closed lab is also an open issue.

8. FUTURE RESEARCH

We identified several future research directions related to closed labs. Most of them stem from the inconclusive nature of our conclusions presented in the previous section.

Below is a list of future research directions, we believe, are suitable for closed CS1/2 labs:

- While it is difficult to compare closed lab design frameworks, it might useful to investigate how other STEM fields can be leveraged to improved closed CS1/2 labs.
- We do not expect a universally adopted assessment methodology in near future, but we believe we should learn from other STEM fields to see how their assessment methodologies can better inform CS closed lab assessment strategies.
- We learned that contextualization helps. We did not find any study that addressed how the backgroond of an instructor teaching a contextualized course affects student learning outcomes. We would like to know what the minimum level of expertise in the contextual area that a CS faculty should have to successfully teach a contextualized introductory programming course/lab.
- Finally, a central repository of information to relevant to success in closed labs should be created.

9. CONCLUDING REMARKS

To the best of our knowledge, no review paper on closed lab practices in CS education, similar to this one has been recently published. This paper was motivated by the aforementioned gap and discussions among colleagues in our department. As a medium-sized regional comprehensive institution, our faculty teach introductory programming on a rotating schedule. While we keep learning outcomes and student outcomes consistent, the administration of the course (e.g., choice, duration and type of lab) is left to the discretion of the course instructor. In light of this reality, we took an opportunity to conduct controlled, theory-driven experiments (currently underway) about novel enhancements to closed laboratories. We hope that this will inspire others to consider possible improvements to student learning by tweaks in their current practice. A large majority of our written end-of-semester student comments on teacher/course evaluations for introductory programming courses mention labs as an important and useful experience, thus memorable. We conjecture that gender-sensitive tweaks to labs have a strong potential to increase diversity in our major. Moreover, we ask the astute reader-practitioner to consider other cultural sensitivities in their lab designs, for these first impressions of our field of study are long-lasting.

10. REFERENCES

[1] R. E. Anderson, M. D. Ernst, R. Ordóñez, P. Pham, and B. Tribelhorn. A data programming cs1 course. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, pages 150–155. ACM, 2015.

[2] K. Becker. Teaching with games: the minesweeper and asteroids experience. *Journal of Computing Sciences in Colleges*, 17(2):23–33, 2001.

[3] D. Chavey. A structured laboratory component for the introductory programming course. In *ACM SIGCSE Bulletin*, volume 23, pages 287–295. ACM, 1991.

[4] W.-K. Chen and Y. C. Cheng. Teaching object-oriented programming laboratory with computer game programming. *Education, IEEE Transactions on*, 50(3):197–203, 2007.

[5] D. C. Cliburn. The effectiveness of games as assignments in an introductory programming course. In *Frontiers in Education Conference, 36th Annual*, pages 6–10. IEEE, 2006.

[6] D. R. Garrison, T. Anderson, and W. Archer. Critical inquiry in a text-based environment: Computer conferencing in higher education. *The internet and higher education*, 2(2):87–105, 1999.

[7] M. Guzdial. Does contextualized computing education help? *ACM Inroads*, 1(4):4–6, 2010.

[8] M. Guzdial. Exploring hypotheses about media computation. In *Proceedings of the ninth annual international ACM conference on International computing education research*, pages 19–26. ACM, 2013.

[9] H. Hakimzadeh, R. Adaikkalavan, and R. Batzinger. Successful implementation of an active learning laboratory in computer science. In *Proceedings of the 39th annual ACM SIGUCCS conference on User services*, pages 83–86. ACM, 2011.

[10] D. Horton, M. Craig, J. Campbell, P. Gries, and D. Zingaro. Comparing outcomes in inverted and traditional cs1. In *Proceedings of the 2014 conference on Innovation & technology in computer science education*, pages 261–266. ACM, 2014.

[11] R. Ibrahim, R. C. M. Yusoff, H. M. Omar, and A. Jaafar. Students perceptions of using educational games to learn introductory programming. *Computer and Information Science*, 4(1):p205, 2010.

[12] D. Ifenthaler, D. Eseryel, and X. Ge. *Assessment for*

game-based learning. Springer, 2012.

[13] K. M. Kapp. *The gamification of learning and instruction: game-based methods and strategies for training and education*. John Wiley & Sons, 2012.

[14] A. N. Kumar. The effect of closed labs in computer science i: an assessment. *Journal of Computing Sciences in Colleges*, 18(5):40–48, 2003.

[15] A. N. Kumar. Closed labs in computer science i revisited in the context of online testing. In *Proceedings of the 41st ACM technical symposium on Computer science education*, pages 539–543. ACM, 2010.

[16] S. Leutenegger and J. Edgington. A games first approach to teaching introductory programming. *ACM SIGCSE Bulletin*, 39(1):115–118, 2007.

[17] C. S. Loh, Y. Sheng, and D. Ifenthaler. *Serious Games Analytics: Methodologies for Performance Measurement, Assessment, and Improvement*. Springer, 2015.

[18] R. McCauley, W. Pharr, G. Pothering, and C. Starr. A proposal to evaluate the effectiveness of closed laboratories in the computer science curriculum. *Journal of Computing Sciences in Colleges*, 19(3):191–198, 2004.

[19] R. McCauley, C. Starr, W. Pharr, R. Stalvey, and G. Pothering. Is cs1 better with the same lecture and lab instructor? *ACM SIGCSE Bulletin*, 38(2):54–60, 2006.

[20] R. P. Mihail, J. Goldsmith, N. Jacobs, and J. W. Jaromczyk. Teaching graphics for games using microsoft xna. In *Computer Games: AI, Animation, Mobile, Interactive Multimedia, Educational & Serious Games (CGAMES), 2013 18th International Conference on*, pages 36–40. IEEE, 2013.

[21] N. Nagappan, L. Williams, M. Ferzli, E. Wiebe, K. Yang, C. Miller, and S. Balik. Improving the cs1 experience with pair programming. In *ACM SIGCSE Bulletin*, volume 35, pages 359–362. ACM, 2003.

[22] L. Porter and B. Simon. Retaining nearly one-third more majors with a trio of instructional best practices in cs1. In *Proceeding of the 44th ACM technical symposium on Computer science education*, pages 165–170. ACM, 2013.

[23] R. Rajaravivarma. A games-based approach for teaching the introductory programming course. *ACM SIGCSE Bulletin*, 37(4):98–102, 2005.

[24] C. F. Reilly and N. De La Mora. The impact of real-world topic labs on student performance in cs1. In *Frontiers in Education Conference (FIE), 2012*, pages 1–6. IEEE, 2012.

[25] L.-K. Soh, A. Samal, and G. Nugent. A framework for cs1 closed laboratories. *Journal on Educational Resources in Computing (JERIC)*, 5(4):2, 2005.

[26] L.-K. Soh, A. Samal, S. Person, G. Nugent, and J. Lang. Analyzing relationships between closed labs and course activities in cs1. In *ACM SIGCSE Bulletin*, volume 37, pages 183–187. ACM, 2005.

[27] L.-K. Soh, A. Samal, S. Person, G. Nugent, and J. Lang. Closed laboratories with embedded instructional research design for cs1. *ACM SIGCSE Bulletin*, 37(1):297–301, 2005.

[28] G. Struble. Most effective lab exercises. *Computer Science Education*, 1(2):177–180, 1989.

[29] K. Sung, P. Shirley, and B. R. Rosenberg. Experiencing aspects of games programming in an introductory computer graphics class. In *ACM SIGCSE Bulletin*, volume 39, pages 249–253. ACM, 2007.

[30] N. Titterton and M. J. Clancy. Adding some lab time is good, adding more must be better: the benefits and barriers to lab-centric courses. In *FECS*, pages 363–367, 2007.

[31] N. Titterton, C. M. Lewis, and M. J. Clancy. Experiences with lab-centric instruction. *Computer Science Education*, 20(2):79–102, 2010.

[32] H. M. Walker. A lab-based approach for introductory computing that emphasizes collaboration. In *Computer Science Education Research Conference*, pages 21–31. Open Universiteit, Heerlen, 2011.

[33] W. A. Waller. A framework for cs1 and cs2 laboratories. In *ACM SIGCSE Bulletin*, volume 26, pages 198–202. ACM, 1994.

[34] L. Williams, E. Wiebe, K. Yang, M. Ferzli, and C. Miller. In support of pair programming in the introductory computer science course. *Computer Science Education*, 12(3):197–212, 2002.

Software Engineering for Systems Engineers

David J. Coe and Jeffrey H. Kulick
Department of Electrical and Computer Engineering
University of Alabama in Huntsville, Huntsville, Alabama, USA

Abstract - *As cyber physical systems evolve, the breadth of expertise that systems engineers need to command is rapidly expanding. Nowhere is this more apparent than the automotive field, where with the advent of autonomous driving features, automobiles now include tens of millions of lines of code. At the request of our Department of Industrial and System Engineering, the authors have developed a course for System Engineers that are going to participate in the design of cyber physical systems. These students typically have limited amounts of domain specific knowledge (for example, rocket guidance) and even less experience in software engineering of large scale cyber physical systems. The goal of the course proposed here is to provide experience in developing a multi-component cyber physical system, verify it meets requirements through simulation, and then auto generate designs for various target technologies such as embedded processors, virtualized processors, and special purpose processors.*

Keywords: Software engineering, systems engineering, Simulink, Raspberry Pi, Model-Based Systems Engineering

1 Introduction

Software-intensive development projects have earned a reputation for being late, over budget, low quality, or failures with respect to satisfying user needs. There has been considerable research on the root causes of software project failures. The most common causes of software project failures include unrealistic or unarticulated goals, poorly defined requirements, unmanaged risks, inaccurate estimates of required resources, and the inability to handle the project's complexity [1].

One example of such a project is the development of the F-35 Joint Strike Fighter (JSF) [2][3]. When completed, the collection of mission and support software will total in excess 24 million lines of code. In 2014, A report produced by the Government Accountability Office (GAO) warned that limited progress in mission systems software testing may cause delivery delay and increase cost. Additional delays have been reported in the development of the Automated Logistics Information System (ALIS), which is the software that supports F-35 operation, maintenance, and supply chain. The ALIS has failed to meet basic functional requirements such as fault detection, fault isolation, and false alarm rates [3].

The development of complex cyber-physical systems such as the F-35 begins with the systems engineering process. Systems engineers develop the concept of operations from which they derive a set of system-level requirements. The systems engineers analyze these requirements with an eye towards allocating each requirement to hardware or software. At that point, the hardware development process and software development process derived detailed requirements from their respective set of allocated requirements, produce designs, and begin implementation.

While there is some overlap in the skill sets of systems engineers and software engineers, the overall preparation required for the two degrees differs significantly. As the software complexity of cyber-physical systems increases, the engineering community must develop efficient ways to cross-train engineers from various engineering domains in software engineering knowledge beyond introductory courses in programming. This additional training is especially important for systems engineers who have great influence in the shaping of project requirements.

The authors have been asked to develop a course that will give systems engineers additional software engineering knowledge. The proposed course will also benefit other engineering disciplines involved in the development of a complex, cyber-physical system such as mechanical engineers or aerospace engineers. Below we discuss the proposed platform, course structure, and sample projects.

2 Software Engineering Crash Course

There are several challenges with respect to development of this course. First, one must identify a set of course objectives within a reasonable set of constraints. The course, for example, cannot assume that the typical systems engineer has prior instruction in computer programming beyond that of an introductory course, a typical requirement of many systems engineering programs. Secondly, one must determine a way to convey the desired course content. Lastly, all content will ideally fit into a single course to facilitate incorporation into systems, mechanical, or aerospace engineering curriculums.

2.1 Course Objectives

The design of the course must satisfy an ambitious set of objectives.

- Introduce Model-Based Systems Engineering (MBSE) as a technique for specifying and developing software-intensive cyber-physical systems
- Familiarize systems, mechanical, and aerospace engineers with the software development life cycle
- Introduce Hardware-in-the-Loop (HWIL) testing concepts
- Demonstrate the value of including MBSE verification in the development cycle
- Illustrate the use of MBSE as a means of achieving an Agile development process
- Provide an introduction to system safety and security concepts

2.2 Proposed Delivery Platform

To achieve the course objectives, Mathworks' Simulink has been selected as the primary modeling tool [4]. Simulink provides a high-level, graphical language for representing the model. With Simulink, an engineer may develop a block model of the desired system and simulate its operation. Figure 1 below shows an example of a block model of a thermostat-controlled heater represented in the Simulink language. When modeling of the system, an engineer has access to a palette of predefined blocks that can be used to compose the model of the system. The predefined blocks include an underlying executable representation to facilitate rapid development and verification of the models.

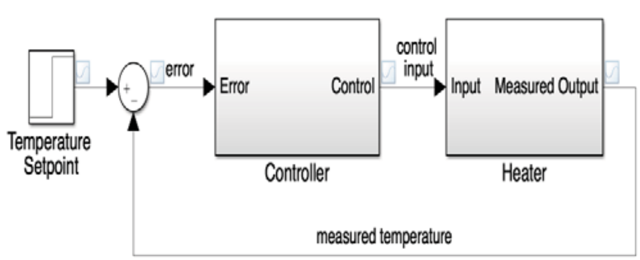


Figure 1 – Sample Simulink block model of a thermostat-controlled heater

Wizards are also available to assist an engineer in tuning the behavior of the more complicated model blocks such as signal filters and controllers. For example, if an engineer can describe the desired characteristics of the filter, then they can use the filter design wizard to automatically create an implementation of that filter even if that engineer has no background in filter design theory. Similarly, an engineer can use a wizard to automatically tune a proportional-integral-derivative (PID) controller block for a given application. Using these wizards, an engineer may rapidly explore a design space to identify a set of parameter values that will satisfy the system requirements.

Another advantage of Simulink is that it includes verification tools that can be used to evaluate the model virtually before deployment into hardware. Once the model has been verified, an executable version of the Simulink model can be either directly downloaded directly from Simulink into the target hardware or a C-code representation of the model may be synthesized and subsequently compiled and deployed to the hardware. Once deployed, the engineer may then conduct on-target testing to identify shortcomings in the model. If discrepancies are found between the virtual model and the deployed software, the engineer can correct the model and redeploy. Because of the wide variety of targets that the Simulink auto coders support, system engineers can experiment with key non-functional requirements such as size, weight, and power (SWAP),

The Raspberry Pi 3 (Pi 3) is an ideal hardware platform for the proposed course [5]. The board is low cost (~$40) to minimize the overhead required to offer the course. It includes a quad-core ARM processor and 1 GB dynamic RAM, that can be used to illustrate parallel programming concepts. Disk drive manufacturers are offering compact 300 GB drives for $45 that are form factor compatible with the Pi 3. The Pi 3 can interface with a variety of sensors and actuators to create a representative cyber-physical system, and it includes wired and wireless networking. Figure 2 below shows a photograph of the Raspberry Pi 3 Model B.

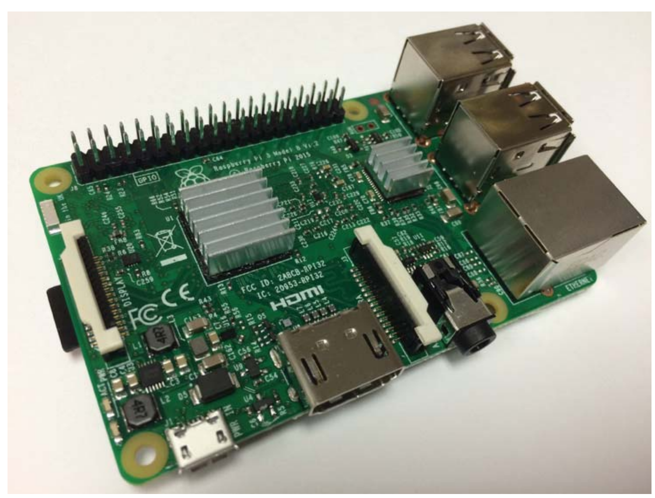


Figure 2 – Photograph of Raspberry Pi 3 Model B [5]

2.3 Discussion

In many real-world scenarios, the hardware is being developed in parallel with the software. The use of model-based engineering allows the creation of virtual plant models that can be used to exercise software against potentially accredited models to allow early identification incompatibilities rather than identifying these issues later in the design cycle which is well known to be very expensive. As subsystem models are verified and implemented in hardware, they can be systematically integrated for hardware-in-the-loop testing, which in our case can consist of multiple Pi 3 boards with their ancillary cyber physical devices.

The goal is not to turn an engineer from another discipline into a software engineer, but rather to give systems, mechanical, and aerospace engineers a better understanding of the challenges of software engineering to improve their ability to manage development of software-intensive cyber-physical systems. To this end, the course must introduce the complete software development life-cycle including requirements specification and refinement, design space exploration, implementation, and testing since each of these activities presents a different set of obstacles. With Simulink's ability to synthesize a hardware or software implementation of the model, a systems engineer can deploy a design directly to hardware even with limited background in programming.

The course will also convey a basic knowledge of computing hardware and architecture. Many cyber-physical systems have hard, real-time performance specifications that must be met. Thanks to the proliferation of mobile devices, engineers have at their disposal low-cost multicore processors that, if utilized properly, may greatly improve system performance and throughput.

Cyber-physical systems frequently have both safety and cybersecurity concerns. While systems engineers may have a knowledge of system safety analysis, they typically lack a background in cybersecurity. With the advent of the Internet of Things (IoT), billions of cyber-physical systems will be connected to the internet. As with system safety, cybersecurity cannot be added as an afterthought. Both must be addressed throughout the development life-cycle, including during development of the concept of operation and system requirements. The course must introduce the concept of threat modeling for the identification of system-level cybersecurity requirements and system safety analysis for identification of System-level safety requirements.

Under an existing grant from the National Security Agency, the authors are developing a hardened version of the XEN hypervisor designed to run on embedded systems. The advent of multicore SOCs such as Xilinx ZINQ processors with multicores, hardware virtualization support, trust zones and integrated FPGA fabrics will allow the deployment of embedded systems with significantly improved security properties then is found in current embedded processors. The introduction of these capabilities in an embedded environment will significantly improve the systems developed by systems engineers participating in this program.

2.4 Representative Course Projects

A typical project that will be part of the program will be the development of a power grid control system consisting of silicon photo cell generators, thermal generators, rotating reserve (e.g. flywheel) generators, transmission lines, and loads.

Students will start out learning Simulink by building models of the individual plants. These models will be run under Simulink and exercised by a test harness to verify that the plant models conform to specifications and can be used as accredited models. The various plant models will have differing rates of response (thermal plants spin up more slowly than rotating reserve generators), and differing rates of reporting status (some components may report every minute, some every second).

Students will then construct controllers for each of these plants. The goal of the controllers will be to comply to a set of design requirements regarding control response. Students will be required to maintain a requirements trace to the various parts and components that contribute to meeting the requirements. This will allow rapid identification of model components that need to be changed when requirements change (which they will during the course).

Students will again test their integrated controller/plant model against requirement based scenarios such as rainy and sunny days, summer and winter days, and other anomalies such as loss of power transmission lines.

Following verification that the controller operates correctly in simulation mode, the students will deploy the controllers, one by one, into target hardware. Various target hardware configurations will be provided such as a system running a Linux operating system controlling all cores, small real time operating systems, hand rolled cyclic executives, and where possible, implementations using virtual controller elements. How and whether the system continues to meet requirements on these various targets as well as the SWAP properties of these targets will be a part of the final evaluation.

2.4.1 Development Model

As much as feasible, an agile model of software engineering development will be used. Clearly some aspects of the work, such as producing a detailed set of requirements for which test cases can be developed will require modifications to the classical "what me worry" associated with agile development. Also, many of the graduate students at our university are full time employees of local companies so that this program has to have a significant distance learning component. The low cost of the target hardware will allow students to acquire their own hardware for testing at home. Many of the toolboxes required for this course are available campus wide for all users, and students can purchase a student package for nominal cost for at home use.

2.4.2 Evaluation

Evaluation of the success of this course will entail more than just students receiving satisfactory grades. Local industry will be invited to participate in the development of the course content, sponsor course projects, and participate in an after action lessons-learned meeting with the students.

3 Conclusions

Model Based System and Software engineering are not new disciplines. Normally the former is practiced by system engineers while the latter is practiced by software engineers. This course will be an attempt to broaden the education of system engineers (and others in mechanical and aerospace engineering, for example) so that the testing and verification methodologies that are currently used by large institutions such as NASA can be pushed down into the general public education program.

4 References

[1] Robert N. Charette, "Why Software Fails," *IEEE Spectrum*, September 2, 2005.

[2] GAO Report GAO-14-322, 'F-35 Joint Strike Fighter: Problems Completing Software Testing May Hinder Delivery of Expected Warfighting Capabilities,' March 2014, http://www.gao.gov/assets/670/661842.pdf.

[3] Jeremiah Gertler, "F-35 Joint Strike Fighter (JSF) Program," *Congressional Research Service*, April 19, 2014, https://www.fas.org/sgp/crs/weapons/RL30563.pdf

[4] MathWorks Simulink, http://www.mathworks.com/products/simulink/

[5] Raspberry Pi 3, https://www.raspberrypi.org/blog/raspberry-pi-3-on-sale/

SESSION

ACTIVE LEARNING, E-LEARNING, LEARNING STRATEGIES, COLLABORATIVE LEARNING, AND TOOLS

Chair(s)

TBA

ISBN: 1-60132-435-9, CSREA Press ©

Developing Professional Competencies in Soft-Skills in Information Technology Students: A Tale of Three Interventions

P. Meso[1] **and D. Kerven**[1]
[1]School of Science and Technology, Georgia Gwinnett College, Lawrenceville, Georgia, USA

Abstract - *Imparting professional-grade soft skills in information systems/information technology (hereafter collectively referred to as IT) professionals is an important part of any college IT curriculum. However, this is not an easy feat due to the nature of IT as a subject and the classroom Lecture & Lab model for most often used for instruction. Nonetheless, experts advocate for the enhancement of soft skills in IT students owing to the integral value of these skills for future success as IT professionals. Many IT professors find themselves using various pedagogical "work-arounds" to foster soft skills development. This paper presents three techniques introduced to enhance students' soft skills in an IT ethics and professionalism course and reports on their effectiveness in fostering soft-skills.*

Keywords: Soft-skills; Ethics; Engagement; Active Learning

1 Introduction

Professionalism is an important component in the education of information technology majors. [1] The ACM and IEE Computer Society in their Curriculum Guidelines for Undergraduate Degree Programs in Information Technology characterize professionalism, as a "pervasive theme" that should be woven into multiple classes within the curriculum. [2] Professionalism in IT is construed as a demonstration of competency and integrity in technical knowhow, systems and process know-how as well as what is generally referred to as "soft-skills" – the innate, usually intangible attributes pertaining to people-skills, critical-thinking/reasoning, communication-efficacy and collaboration, among others.

While the imparting of technical competency to IT students is relatively straightforward, it is not as easy to develop soft-skills within students. Part of the problem has to do with the nature of IT content. IT subject matter is for the most part technical, methodical, structured, requiring much more of the "left-brain" capabilities than the "right-brain" capabilities. Another contribution to this is the very structure of IT pedagogy. Most IT courses are structured to overly rely on the lecture/lab approach to teaching IT concepts and competencies. With this approach, students receive instruction and learning about a particular topic via a lecture to enhance their cognitive competence - the know-what and know-why skill - and then complete a series of hand-on exercises within a lab or structured-homework setting as a way of developing their technical competence – the know-how skill - in the topic. Thus, there is a built in bias towards technical competence (know-what, know-why and know-how competencies) at the expense of the subtler "soft-skills."

2 Literature Review

"[J]ust learning the technical theory and skills is not enough and does not constitute a complete educational program in any computing field. Students must also develop the communication skills, teamwork skills, and ethical analysis skills required of practicing professionals." [3] Soft skills are a necessary component in an effective IT program.

Many studies support the importance of soft skills by IT students. For example, Noll et al. identify the need to integrate effective oral and written communication, collaboration, project management and interpersonal relationship skills into IT education from a survey of a diverse population of IT industry personnel. [4] McMurtrey et al. found in their study of IT field personnel that soft skills, including problem-solving critical thinking and collaborative ability, were deemed the most important for entry-level IT professionals. [5] Finally, Aasheim et al., from their study of personnel self-identifying as in IT leadership roles, validated their 2006 survey results identifying personal and interpersonal skills as the most significant. [6] The top four skills identified in the 2006 survey were honesty/integrity, communication skills (oral and written), analytic skills, and ability to work in teams. [7] The new study conducted in 2010 included additional skills; however, the top skills in the new study remained soft skills. Specifically attitude, willingness to learn new skills, and professionalism, newly introduced to the 2010 survey, joined the 2006 winners at the top of the skills list. [6] These results suggest a need on the part of faculty to develop courses and approaches that foster soft skills among IT students.

Many IT programs include courses in professional practice and ethics. [8] Given the nature of the soft skills highlighted by the various studies, the professional practice and ethics classes would provide a suitable platform for fostering these skills. [3] However, these courses are often difficult for IT faculty to teach. [8,9,10] A primary reason cited by researchers in IT ethics is that the traditional lecture format familiar to most IT faculty is not well suited for this type of course. [10,11] Researchers have suggested that use of a discussion format and/or case studies approach yields better educational results. [9,10,12] These formats more actively engage the students in the learning process.

Injecting creative activities is a technique applied in IT education to increase student engagement. [11] This technique is often applied in the technical IT area, where hands-on class projects are assigned and assessed. Project-based learning has a long history in IT, for example, in the areas of software development [13], systems [14], and web development [15]. In the area of IT professional practice an ethics, the adaptation of creative activities has more recently been introduced. For instance, Nygard et al. assigned a group project involving analysis of ethical questions, where team members were required to role play viewpoints of various stakeholders within the ethical situation. [3] Howard also employed a creative activity in her ethics course. In her course, she included a final project where the students had an open ended choice of presentation method as well as topic. [11] For the most part, her students chose a large group discussion format; however, two groups chose more dynamic forms of presentation: an interactive poll with discussion and a mock trial. [11]

A number of researchers have also examined the use of undergraduate research in computer science and information technology programs and found such usage engaging and educationally beneficial to students. [16,17,18] Specifically, both Pastel and Tashakkori incorporated research into specific undergraduate courses, Human Computer Interactions and Digital Image Processing, respectively, through assignment of a course related research project. [16,18] However, this technique does not appear to have significant usage in the area of professional practice and ethics.

These techniques serve as the foundation for the foregoing interventions to foster greater soft skill development in a professional practice and ethics class.

3 Course Context

The course is titled "Professional Practice and Ethics" and is required for all students with an information technology major at Georgia Gwinnett College. This course focuses on teaching students fundamental principles of professionalism, professional practice and ethics. Emphasis is placed on key areas of contention concerning ethics within the information technology profession, key among these being security, privacy, and intellectual property. Also covered at great depth are issues pertaining to professionalism such as personal professional responsibilities, communications acumen, people skills, professional conduct, image and reputation, as well as issues relating to professional certifications and the processes used to certify individuals within the profession.

Conventionally, this course has been taught by way of lectures on the key topics complemented by case-study projects on each topic. While this allowed for reasonable and effective coverage of the pedagogical content, students had trouble internalizing these key concepts within an information technology or organizational workplace. This led to efforts to enhance the delivery of the course with a view to better engage the students and make the course more impactful to them through several interventions.

4 Interventions

Three interventions were introduced into the classroom not only to enhance the students understanding of the underlying professional practice and ethics principles but also to immerse the student into situations/contexts that may contribute to the development of various "soft-skills." The first intervention entailed use of cases to develop group-led interactive active-learning tasks. Student groups were assigned a case for which they would develop an activity and facilitate coverage of the activity to enable the rest of the class learn the underlying principles within the case. The second intervention entailed reading and critiquing of scholarly articles. In the third intervention, students were assigned to an ethics research project completed in groups of three members.

4.1 Interactive Exercise

Ten cases related to course topical areas were included as part of the course. Each student was asked to research and analyze the ten cases during the course of the semester. In addition, each case was assigned to a team of three students. Each team was tasked with developing an interactive activity demonstrating the underlying case principles. The interactive task could take any form – jeopardy quiz, scavenger hunt, simulation, role-play, human-video, small-groups assignment, debate, audition, experiment, interview, game-show, etc. However, the activity could not be a lecture. The objective was for each team to develop an activity that would allow the members of the class to engage with the case in an "active-learning" manner and interactively explore the principles from the case by completing the activity.

4.2 Scholarly Articles

Students were assigned a set of ten course topic related scholarly articles to read and critique. The critique on each article was limited to a two-page executive-summary-style written paper submission. These articles were then discussed in class with each group getting an opportunity to moderate the classroom discussion – emphasis here being on leveraging oral communications on the one hand, and leadership/moderation skills on the other.

4.3 Research Project

The research project was designed to allow students (1) to ask questions beyond the materials presented in class, (2) to synthesize their own model of how the underlying principles in those materials are interrelated and (3) to experimentally test their model through survey-based data collection and structural equation modeling (SEM) statistical data analysis. The research project was completed in two parts with the first part being due about two-thirds into the semester and the second part at the end of the semester.

In the first part, the instructor provided guidelines on how to develop a causal model (by teaching them about latent variables, indicators and causal relationships) and how to scientifically conduct a literature review (by teaching them how to search for relationships, indicators of latent variables and/or hypotheses of interest as they read and associated the various scholarly articles and cases covered in the course). Students were organized into groups of three. Each group revisited all the cases and scholarly articles assigned during the semester and used that content as a basis for developing a causal model detailing a hypothesized beliefs regarding the interrelationships among five key concepts: Ethics, Morals, Professionalism, Risk, and Value. The causal model included each of these concepts as a latent variable. Each relationship within the model is envisioned to represent a testable hypothesis.

As a final task within this first part of the research project, each group developed an executive-summary styled report that articulated its causal model and provided justifications for that model. Toulmin's model for argumentation [19] was used as the basis for articulating these justifications. Requiring that each group justify its causal model ensured that a group did not merely slap together an arbitrary model but rather had critically thought through the model and established substantive, content-backed, reasons for each latent variable and each relationship contained within their causal model.

In the second part of the research project, students were taught how to develop a survey instrument that operationalized the indicators in the causal model and how to administer such instrument via web-based survey tools. The students then develop a single, course-wide, survey instrument themed on peer-to-peer file sharing and administered this instrument among peers across the campus.

Peer-to-peer file sharing was selected as the theme for three reasons. First, it is an area of great familiarity among college students. Secondly, it is an area of wide contention both within university and college settings and also within professional information technology settings. Finally, it ably educes almost all aspects of key importance concerning information-technology ethics – privacy (both information and personal), security (both information and personal), intellectual property, professional conduct, professional image/reputation, and personal professional responsibility among others.

Model validation and testing was done using the partial least squares approach as implemented in the Smart PLS software. [20] For this part of the research project, students were taught how to use Smart PLS to assess causal models and how to report their findings in the format of a scientific report. The students then assessed their hypothesized relationships, articulated the results of their analysis within a scientific research report, and presented their results to the class.

5 Research Design

In order to investigate the impact of these interventions on the development of soft-skills, a post course survey was used to collect data. A model-building approach was used on this data to discover which pedagogical mechanism impacted or elicited which skills-development outcomes significantly.

5.1 Data Collection Procedure

Students enrolled in the course were requested to voluntarily participate in an Institutional-Review-Board (IRB) approved research study associated with the course. Students who volunteered were asked to complete a post-course survey on their experiences and perspectives of the tasks and content they had completed in the course. The survey was anonymous and conducted after students had completed all the course requirements. The survey collected data on the pedagogical mechanisms embedded in each of the interventions and the desired outcomes. Some data were also collected from instructor observation and assessment of student performance.

The specific pedagogical mechanisms include:
- Attending class,
- Group work (group oral presentations of research project and scholarly articles, case-based activities),
- Case activity,
- Research project activity,
- Class discussions (discussion of scholarly readings and discussion of research project results), and
- Efficacy of scholarly articles (base on research project).

The desired outcomes include:
- Communication skills;
- Collaboration skills;
- Critical/analytical reasoning skills;
- Analytical reading, and
- Understanding of subject matter.

Since the survey data was collected from students after course completion, the only feasible way to statistically validate the survey was after data collection. This statistical validation was conducted using Smart PLS software's measurement-model testing capabilities and discussed in further detail below. Post-data-collection validation of the survey is one of the limitations of this study. This validation does serve as a starting point for refinement of the survey for use in future research.

5.2 Data Analysis

A model-building approach was used to discover which pedagogical mechanism impacted or elicited which skills-development outcomes significantly. Said otherwise, an

ISBN: 1-60132-435-9, CSREA Press ©

exploratory model was created that linked (i.e. hypothesized that) each pedagogical mechanism impacted each desired soft skill outcome. Then, the process of elimination was used to remove all the relationships that did not produce a statistically significant result.

6 Results

The results of this research design applied to five sections of the course over two semesters are described below.

6.1 Participant Populations

Fifty students volunteered to participate in the study. Demographic statistics are provided in Table 1.

Table 1: Demographic Data of Survey Participants

Item	Sub-items	No.	% of sample	Mean
Gender	Female	8	16%	n/a
	Male	42	84%	
Age	18-20	3	6%	21-25
	21-25	35	70%	
	26-30	9	18%	
	31-35	2	4%	
	36-40	0	0%	
	41 and above	1	2%	
Major	Digital Media	10	20%	n/a
	Enterprise Systems	6	12%	
	Software Development	19	38%	
	Systems and Security	12	24%	
	Other	3	6%	
Class Year	Freshman	0	0%	Junior
	Sophomore	3	6%	
	Junior	31	62%	
	Senior	16	32%	

6.2 Validation of Survey Instrument

Survey instrument validation occurs simultaneously with model evaluation within PLS. First, Cronbach's alpha values are calculated for each construct to assess instrument reliability. Table 2 provides these results. Given that this is an exploratory study, the threshold for instrument reliability is 0.6. [21] The Cronbach's Alpha scores obtained indicate that all constructs exhibited sufficiently high scores. Second, convergent reliability was assessed. For exploratory studies, convergent validity is deemed adequate when composite reliability, measured by the rank correlation coefficient (rho) value of each construct, exceeds 0.70. [21] Results for this test are presented in table 2. All the constructs in the model were found to have adequate convergent validity.

Table 2: Cronbach's Alpha and Composite Reliability Scores

Construct	Cronbach's Alpha	Composite Reliability
Analytical Reading	0.815	0.915
Usefulness of Article's Critiques	0.940	0.952
Attend Class	1	1
Case Activity	0.908	0.956
Collaboration & Communication	0.661	0.853
Critical Reasoning/Argumentation	0.742	0.885
Discussion	0.628	0.829
Research Project Group Work	0.722	0.877
Understanding	0.923	0.942

The instrument's construct validity was assessed by employing confirmatory factor analysis (CFA) algorithm. This determines how individual indicators loaded onto their respective constructs. All of the indicators loaded adequately because each had a value greater than 0.7. [22] Further, an analysis of the indicator's cross loadings revealed that each indicator does load most highly on its appropriate construct in comparison to any of the other constructs in the model. [23,22]

Fourth, the instrument's discriminant validity was assessed by comparing the square root of the average variance extracted (AVE) of each construct to the correlations of that construct with the other constructs in the model. The AVE and square root of AVE values are greater than the correlations among the constructs. Further, all the AVE values were greater than 0.5. As a result, all the constructs in the theoretical model demonstrated satisfactory discriminant validity.

6.3 Exploratory Model

Having been satisfied that the survey instrument passes the fundamental tests of validity, the data collected by that instrument is used to assess the perceived value of the pedagogical mechanism on fostering soft skills. An exploratory model was developed that hypothesized a relationship between each of the pedagogical mechanisms and each of the desired outcomes. Relationships not found to be statistically significant based upon the collected data were removed from the model. The analysis was performed using the partial least squares structural equation modelling techniques as implemented in Smart PLS software. [20] The statistical analysis of the data is provided in Table 3. The resultant model including only statistically significant relationships is depicted in Figure 1.

Table 3: Final Results from Assessment of the Survey Instrument

Dependent Variable	Plausible Hypothesis	Path Coefficient	Sample Mean	Standard Deviation	T Statistics
PEDAGOGICAL MECHANISM	Attend Class -> Discussion	0.2991	0.2955	0.1213	2.4671
	Article Critiques -> Discussion	0.3916	0.3913	0.0834	4.6967
	Article Critiques -> Research Project	0.5432	0.5491	0.0785	6.9185
SOFT SKILL	Attend Class -> Critical Reasoning	0.197	0.194	0.0584	3.3745
	Article Critiques -> Analytical Reading	0.3878	0.3956	0.1405	2.7603
	Case Activity -> Analytical Reading	0.2481	0.2451	0.1281	1.9358*
	Discussion -> Analytical Reading	0.2007	0.1939	0.1003	2.0017
	Discussion -> Critical Reasoning	0.2913	0.3042	0.0969	3.0069
	Research Project -> Collaboration & Communication	0.859	0.859	0.197	4.363
	Research Project -> Critical Reasoning	0.3593	0.3549	0.1114	3.2255
OVERALL UNDERSTANDING	Attend Class -> Understanding	0.6923	0.6847	0.0844	8.2071
	Research Project -> Understanding	0.2168	0.2192	0.0713	3.0406
* hypothesis was supported at the 90% level of confidence					

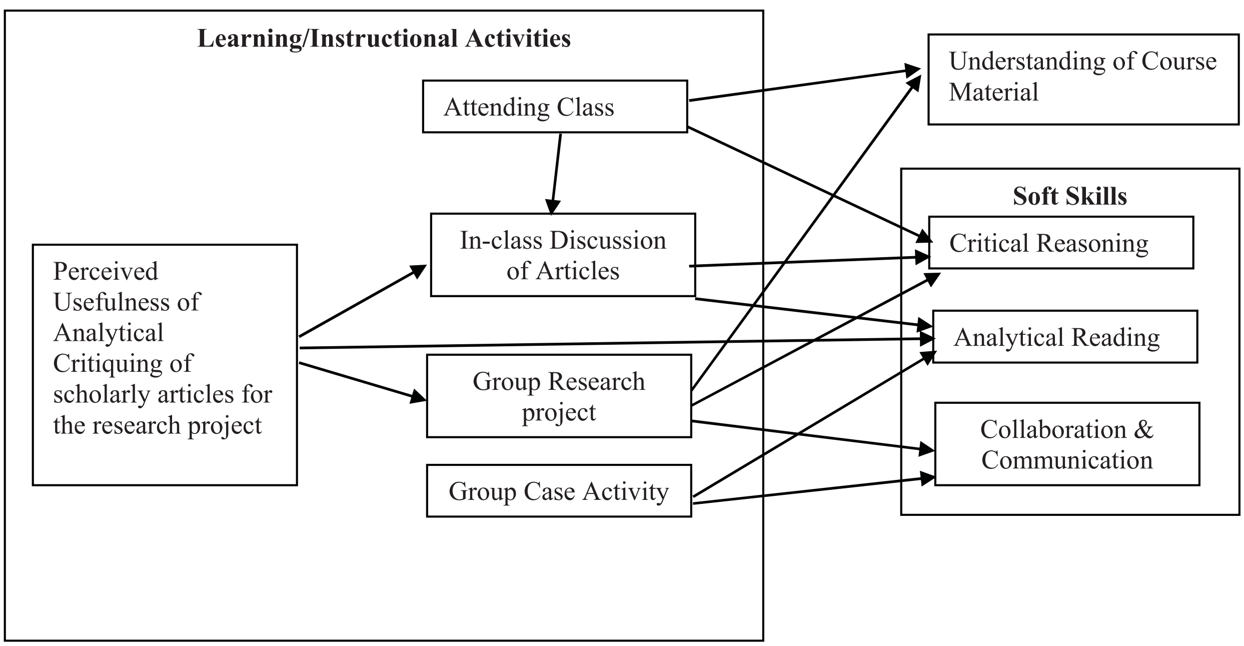


Figure 1 Causal Model

6.4 Predictive Power of the Resultant Model

R^2 values represent the amount of variance within a dependent variable that is explained by the set of independent variables that relate to that dependent variable. R^2 values are interpreted based on their magnitude, - larger values indicating higher predictive power of the independent variables on the dependent variable. Table 4 presents the R^2 values for the survey. The values indicate that the independent variables accounted for 30% to 67% of the variance observed in the dependent variables. This in itself is an additional indicator of the contributory effects of the active-learning tasks to student's learning outcomes. It also provides validity to the relevance of the resultant theoretical model as articulated in figure 1. A natural next step will be the empirical validation of this model in different settings and contexts.

Table 4: R^2 Values for the Dependent Variables Derived from the Survey Instrument

Construct	R Square
Analytical Reading	0.428
Collaboration and communication	0.601
Critical Reasoning	0.476
Discussion	0.346
Research Project Group Work	0.304
Understanding	0.669

7 Conclusion

This study set out to examine and identify if indeed a multi-method active learning strategy enhances development of professional soft skills among information technology students. The study finds that there are tangible benefits in designing information technology courses to be more engaging. Results obtained in this study indicate that tasks that foster a student's engagement with both (a) the content of the course and (b) other students within the course facilitate development of skill sets that may be otherwise difficult to impart in students. A clear example of these kinds of skills, as demonstrated in this paper, is the acquisition of professional soft skills.

The primary limitation of this study is that it is exploratory in design. The purpose of this study was to develop a plausible causal model that relates pedagogical approaches/strategies to "hard to teach" but necessary skill sets – in this case, development of professional soft skills among IT students. The model that was produced from this study shows promise. It has allowed the articulation of a plausible causal-model that theorizes the relationship between pedagogical approaches and students' development of professional soft skills as well as students' overall understanding of a course's core content. However, this plausible causal-model is still in need of validation in one or more explanatory-design studies - preferably operationalized in multiple different contexts. Therefore, while the results from this study are encouraging, we look forward to future studies that employ the model and approaches articulated in this study. Our hope is that the model developed in this study, and the results that we obtained, will hold in other settings as well.

A secondary limitation of this study is it relatively small sample size as well as the post-hoc approach to validating the data-collection instrument employed in the study. Future explanatory studies involving much larger samples where the instrument developed in this study can be empirically re-validated will contribute to the improvement or validation of this model.

8 References

[1] Gordana Dodig-Crnkovic, "On the Importance of Teaching Professional Ethics to Computer Science Students," in *Computing and Philosophy Conference, E-CAP 2004*, Pavia, Italy, 2005.

[2] Association for Computing Machinery and IEEE Computer Society, "Curriculum Guidelines for Undergraduate Degree Programs in Information Technology," 2008.

[3] Kendall E Nygard et al., "Strategies for Teaching Ideation and Ethics in Computer Science," in *Proceedings of the International Conference on Frontiers in Education: Computer Science and Computer Engineering*, Las Vegas, Nevada, 2012, pp. 1-7.

[4] Cheryl L Noll and Marilyn Wilkins, "Critical Skills of IS Professionals: A Model for Curriculum Development," *Journal of Information Technology Education*, vol. 1, no. 3, pp. 143-154, 2002.

[5] Mark E McMurtrey, James P Downey, Steven M Zeltmann, and William H Friedman, "Critical Skill Sets of Entry-Level IT Professionals: An Empirical Examination of Perceptions from Field Personnel," *Journal of Information Technology Education*, vol. 7, no. 1, pp. 101-120, 2008.

[6] Cheryl Assheim, Jordan Shropshire, Lixin Li, and Christopher Kadlec, "Knowledge and Skill Requirements for Entry-Level IT Workers: A Longitudinal Study," *Journal of Information Systems Education*, vol. 3, no. 2, pp. 193-204, Summer 2012.

[7] Cheryl L Aasheim, Lixin Li, and Susan Williams, "Knowledge and skill requirements for entry-level information technology workers: A comparison of industry and academia," *Journal of Information Systems Education*, vol. 20, no. 3, pp. 349-356, 2009.

[8] Randy Connolly, "Beyond good and evil impacts: rethinking the social issues components in our computing curricula," in *Proceedings of the 16th annual joint conference on Innovation and technology in computer science education*, New York, NY, 2011, pp. 228-232.

[9] Elizabeth V Howard, "Facing the Challenges of Teaching IT Ethics," in *Proceedings of the 7th conference on Information technology education*, New York, NY, 2006, pp. 95-98.

[10] Alton F Sanders, "A Discussion Format for Computer Ethics," *SIGCSE Bull.*, vol. 37, no. 1, pp. 352-355, February 2005.

[11] Elizabeth V Howard, "Encouraging Engagement in an IT Ethics Course by Fostering Creativity," *J. Comput. Sci. Coll.*, vol. 23, no. 1, pp. 8-13, October 2007.

[12] Raquel Benbunan-Fich, "Guidelines for Using Case Scenarios to Teach Computer Ethics," *SIGCAS Comput. Soc.*, vol. 28, no. 3, pp. 20-24, September 1998.

[13] Douglas D Bickerstaff, "The evolution of a project oriented course in software development," in *SIGCSE '85 Proceedings of the sixteenth SIGCSE technical symposium on Computer science education*, New York, NY, 1985, pp. 13-22.

[14] Daniel Farkas, "Choosing group projects for advanced systems courses," in *SIGCSE '88 Proceedings of the nineteenth SIGCSE technical symposium on Computer science education*, New York, NY, 1988, pp. 109-115.

[15] Paula Gabbert and Kevin Treu, "Reality check: working with meaningful projects in and out of the classroom," *J. Comput. Sci. Coll.*, vol. 17, no. 2, pp. 191-198, December 2001.

[16] Robert Pastel, "Integrating Science and Research in a HCI Design Course," in *Proceedings of the 36th SIGCSE technical symposium on Computer science education*, St. Louis, MO, 2005, pp. 31-35.

[17] Steven Hadfield and Dino Schweitzer, "Building an Undergraduate Computer Science Research Experience," in *Proceedings of the 39th IEEE International Conference on Frontiers in Education*, Piscataway, NJ, 2009, pp. 1193-1198.

[18] Rahman Tashakkori, "Encouraging undergraduate research: a digital image processing approach," *Journal of Computing Sciences in Colleges*, vol. 20, no. 3, pp. 173-180, February 2005.

[19] Toulmin, *The Uses of Argument*., 1958.

[20] C. M. Ringle, S. Wende, and A. Will, *SmartPLS 2.0 (beta)*., 2005.

[21] J. Nunnally, *Psychometric Theory*, 2nd ed. New York, NY: McGraw-Hill, 1978.

[22] David Gefen, Detmar W. Straub, and Marie-Claude Boudreau, "STRUCTURAL EQUATION MODELING AND REGRESSION: GUIDELINES FOR RESEARCH PRACTICE," *Communications of AIS*, vol. 4, pp. 1-79, August 2000.

[23] W. W. Chin, "The partial least squares approach to structural equation modeling," in *Modern Methods for Business Research*, G. A. Marcoulides, Ed. London, UK: Lawrence Erlbaum Associates, 1998, ch. 10, pp. 295-336.

Engaging Students in Software Engineering Through Active Learning in Software Engineering

Rajendran Swamidurai[1] **and David Umphress**[2]
[1]Mathematics and Computer Science, Alabama State University, Montgomery, AL, USA
[2]Computer Science and Software Engineering, Auburn University, Auburn, AL, USA

Abstract - *This paper presents our three years' experience in designing and implementing a practitioner-centered based undergraduate software engineering course. To engage the students in software engineering through active learning, we have adapted a single-person software process called Practitioner-Centered Software Engineering (PCSE). The objective of PCSE is to have individuals developers/students enact engineering activities at their own skill level, as opposed to dictating engineering activities in the traditional fashion. As such, PCSE is not a software process, but a framework for identifying an initial software process and modifying that process to meet specific, individual developer/student needs. We describe the implementation of the PCSE-based undergraduate software engineering course, and then present the challenges in successful implementation of PCSE-based pedagogy in undergraduate software engineering courses.*

Keywords: Practitioner-centered software engineering; PCSE; active learning; test driven development; software process; personal software process.

1 Introduction

Employment in the software development profession is expected to increase by 22% over the next decade, making it one of the rapidly growing professions in the country. A large increase in the demand for computer software in mobile, healthcare and cyber security industries is the main reason behind this rapid growth in expected employment [1].

Though employment for computing profession is going to increase by 27 percent by the next decade in general, employment of computer programmers is projected to decline 8 percent from 2014 to 2024 [2]. It is clear that U.S. companies don't want computing graduates who just having programing or coding abilities. Students' who are inculcated with broader software engineering process technical and business skills than simply programming skills are more attractive to employers [3].

Equipping students with knowledge about practical software development is problematic in today's academic arena. Colleges teach students the principles of software development, but that instruction is mostly theoretical and abstract. Developing working software requires specific knowledge in software engineering industrial practices. The traditional university curricula do not address these areas in any depth. We have taken a first step in departing from the traditional curricula by orienting an undergraduate course to software engineering practices. Course material on software engineering, including software process, is readily available. What is missing is the mechanism to expose students to real-world software issues encountered in the software industry.

This paper presents an experience in constructing, teaching, and assessing a course that immerses students in real-world software engineering practices through active learning. Our course walked students through producing a working solution by having them use a one-person process, known as Practitioner-Centered Software Engineering (PCSE), developed specifically to apply cutting-edge industry techniques at each point in the software lifecycle. The process was taught incrementally throughout the course, starting with techniques that enhance writing code, and having an engineering activity, which addresses another lifecycle activity added each week. Students worked in an interactive environment in which they were instructed on new techniques and then mentored in the use of those techniques through a series of hands-on exercises.

2 Practitioner-Centered Software Engineering (PCSE)

The Practitioner-Centered Software Engineering (PCSE) is an agile alternative to PSP. It focuses on carrying out fundamental principles of software engineering while letting the developer select appropriate practices. It emphasizes common high-leverage practices by concentrating on high-yield individual performance (which, in turn, can foster team integration) to cultivate discipline, agility, and quality.

PCSE is a tailored collection of different elements from various software processes such as Personal Software Process (PSP) [4], Team Software Process (TSP) [5], Extreme Programming (XP) [6], Feature Driven Development (FDD) [7], SCRUM [8], Rational Unified Process (RUP) [9] etc.

PCSE incorporates the following Capability Maturity Model Integration (CMMI) process areas:

- Project planning.
- Project monitoring and control.
- Risk management.
- Configuration management.
- Process/product quality assurance.
- Measurements.
- Requirements management.
- Technical solution.
- Verification.
- Validation.
- Product integration.

Usage of this model helps the software engineers make accurate plans; consistently meet commitments; improve quality, predictability, productivity, and customer satisfaction; and deliver high-quality products. Using this model, students experience a software development effort from inception to final delivery. It also means that standards and guidelines are small and manageable, and can be distilled for classroom use with little loss of industry intent. This is in sharp contrast to software development using heavy weight software processes in which standards and guidelines are so comprehensive as to be difficult to adapt to student use.

2.1 Software Engineering

Andrew Koenig notes that "software is hard to develop for many reasons: we must figure out what to do, do it, and ensure that we have done it correctly" [10]. The so-called "traditional" engineering fields such as mechanical, civil, electrical, chemical came into being for the very purpose of addressing the same issues. The difference is that the traditional engineering fields address building complex physical things instead of building software. Just as electrical engineering is the business of developing worthwhile solutions through power systems and civil engineering is the business of developing worthwhile solutions through bridges and roads, software engineering can be defined in the same sense: software engineers are in the business of developing worthwhile solutions and they just happen to articulate those solutions in software instead of physical systems.

2.2 Cognitive Activities

The four fundamental activities of engineering are: ***Envision***, identifying a desired future state; ***Synthesize***, determining how to reach that future state; ***Articulate***, taking the actions necessary to reach the future state; and ***Interpret***, ensuring that the future state has been achieved. Applying these activities to software at a high level of abstraction, we have, respectively, ***analysis, design, construction***, and ***test***. This is not suggesting that all these activities take place in sequence. In fact, how we order these activities, how much of each activity we perform and how we carry out each activity determines to a great extent the quality of the software solution itself. Performing these activities in a seat-of-the-pants fashion gives us – and, more importantly our customers – little confidence that we have systematically come up with a good solution.

The term "software process" was coined to describe how and when engineering activities are carried out. The idea was initially exciting due to the parallels it draws to manufacturing processes used in industry to produce quality goods. Over the years, attempts to codify, measure, and enforce process cast a bureaucratic pall over the term, so much so that "software process" is eschewed by many developers because it smacks of attempts to overregulate an essentially creative endeavor.

The tenet engineering activities – envision, synthesize, articulate, and interpret – are abstract in the sense that they address the "what" of problem solving, they don't address how to go about carrying them out. This distinction is important. If an activity is carried out in a defined fashion, we refer to it as a "practice". Put simply: activities are abstract, practices are specific.

The objective of PCSE is to have individuals developers enact engineering activities at their own skill level, as opposed to dictating engineering activities in the traditional fashion. As such, PCSE is not a software process, but a framework for identifying an initial software process and modifying that process to meet specific, individual developer needs. PCSE, thus, views "process" not as prescriptive software development, but as a conscious recognition of the way in which to build software.

For example, suppose we are trying to determine the problem that is to be addressed with a software solution. We are in the "envision" activity, specifically performing requirements analysis. Absent any particular approach to conducting requirements analysis, we consider it an activity with no prescribed practice. If we don't feel comfortable that we know how to accomplish it, we break it down in engineering fashion: we envision, synthesize, articulate, and interpret requirements. In other words, we elicit requirements, analyze requirements, specify requirements, and validate requirements. If we have a specific way of carrying out, say, elicitation, we then say we have a practice for it and we need not define it any further. If we don't have a practice for it, we repeat the engineering process until we get to a level of granularity where we can identify a practice. This process is explained in figure 1.

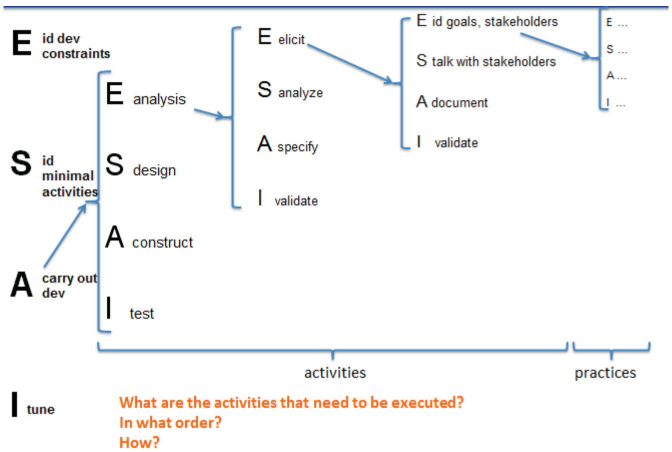


Fig.1. Cognitive activities involved

Given this concept, PCSE addresses the issues of

- What activities need to be carried out
- What practices are used to carry out activities
- In what order is everything done
- How well is everything being done
- What activities and practices need to be changed during development to improve quality

3 Ingredients of the PCSE

The ingredients of the PCSE are:

1) **Minimal Guiding Indicators (MGI):** A statement of project goals and the means by which we know if we have achieved them. The MGI is typically expressed in terms of cost, schedule, and performance.
2) **Minimally Sufficient Activities (MSA):** The least possible set of conceptual lifecycle activities needed to produce a working solution of a given quality. MSAs describe how objectives will be achieved.
3) **Minimally Viable Process (MVP):** Orders MSAs so as to structure the effort to be as non-invasive as possible, yet provide enough structure to be viable. Done by listing the lowest-level MSAs and determining work flow. MVP identifies the relationship of the MSAs.
4) **Minimally Effective Practice (MEP):** Instructs developers in what to do. Done by specifying lowest-level MSAs. MEPs describe how MVPs will be carried out.

The process of identifying the necessary ingredients (MGI + MSA + MVP + MEP) is described in figure 2 and an actual example of PCSE is shown in figure 3.

Having identified the necessary ingredients (MGI + MSA + MVP + MEP) information, we have a basis for alleviating software development pain points. But in an industrial setting, we may not necessarily know the goal process. We start with the status quo and tune according to practitioner needs – hence the name: Practitioner-Centered Software Engineering (Refer figure 4).

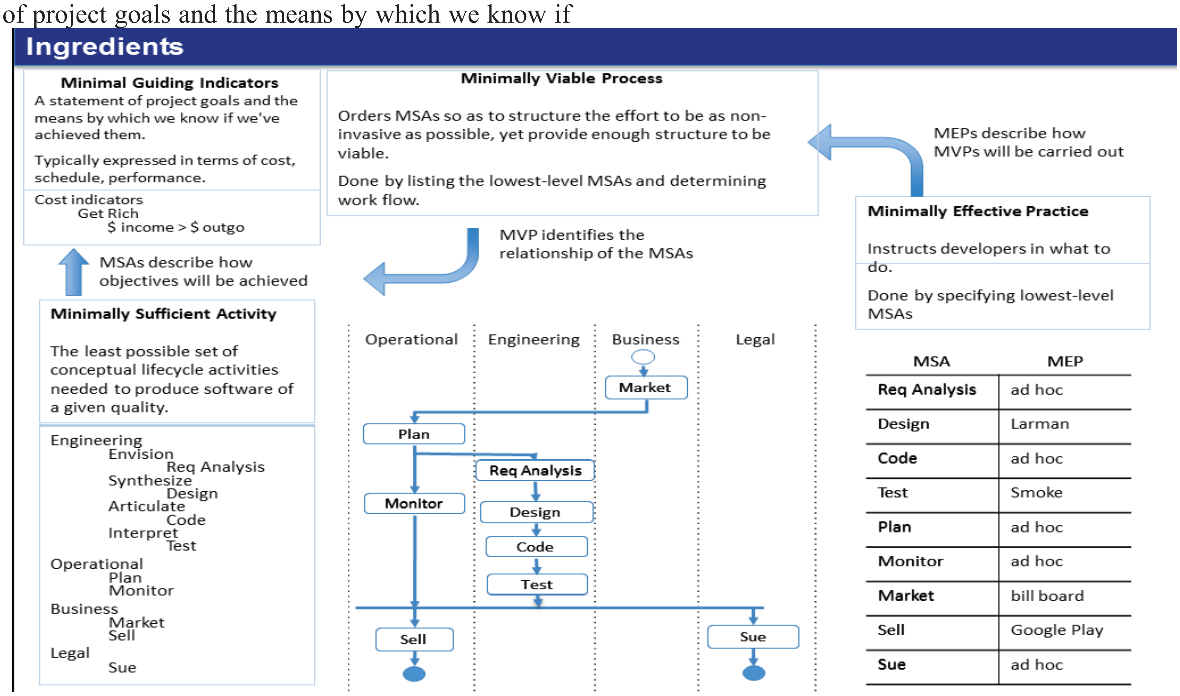


Fig.2. Ingredients of PCSE

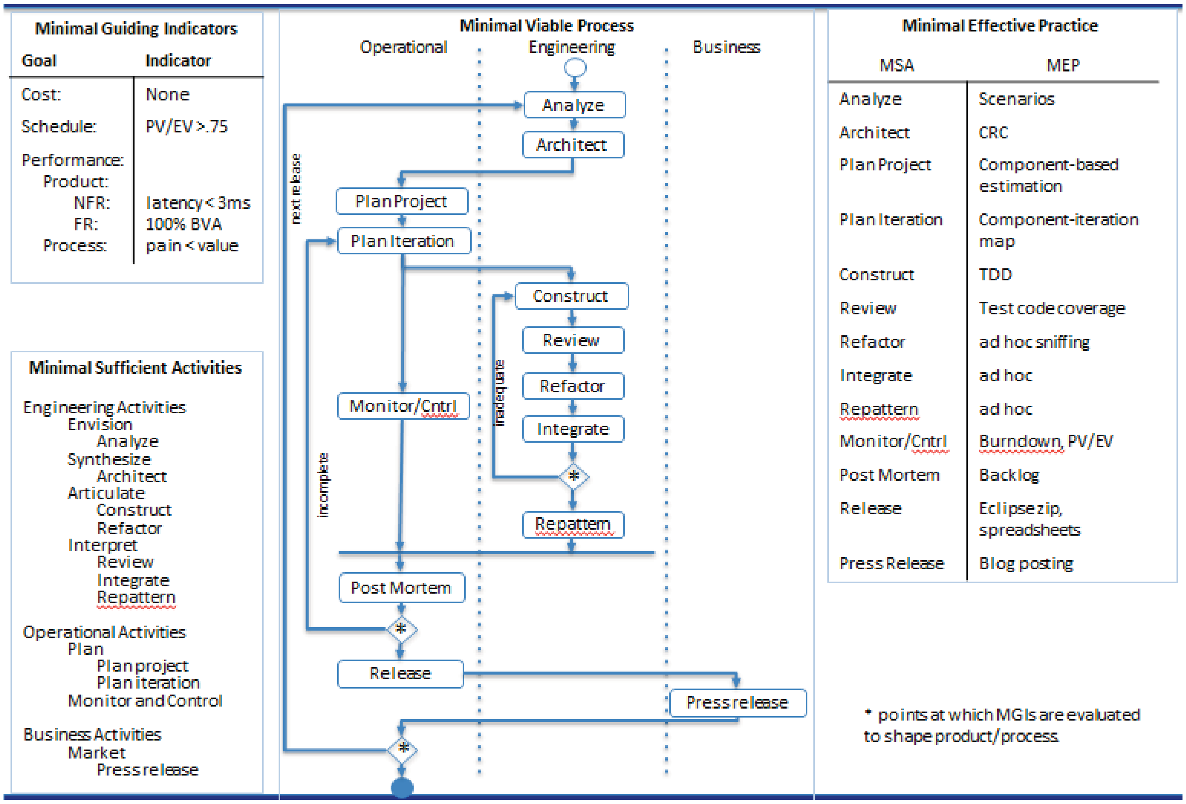


Fig.3. An Actual Example of PCSE

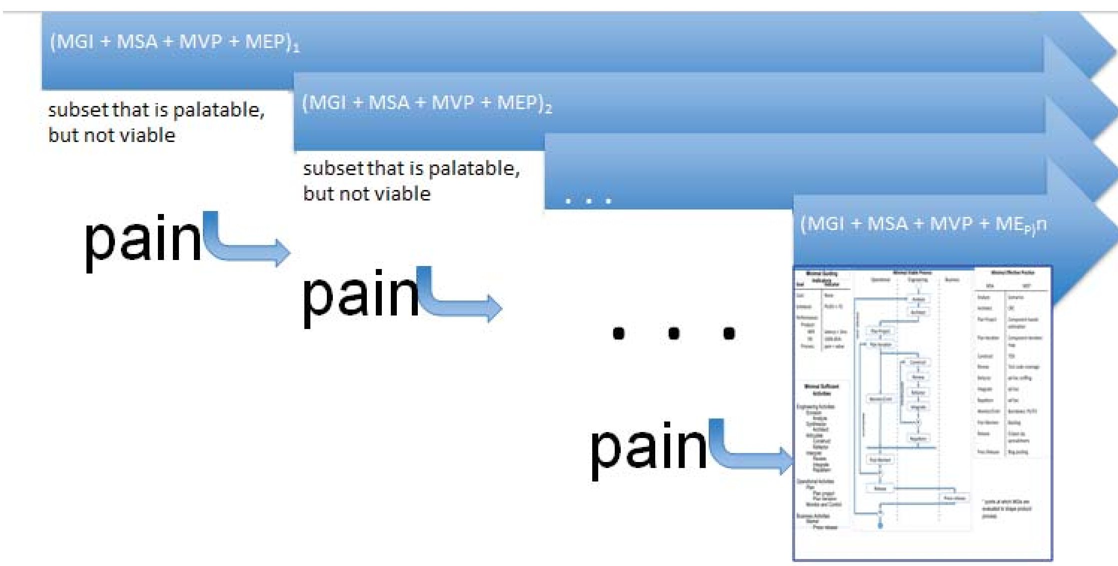


Fig.4. The Practitioner-Centered Software Engineering (PCSE)

4 Implementing PCSE in Software Engineering Course

The Software Engineering II course at Alabama State University is typical in content, focusing on software lifecycle and associated tools and techniques. The instructional language is C++ and the instructional IDE is Eclipse (eclipse.org). Software Engineering II is a 3 credit hour course and meets for 2 clock hours of lecture per week and 2 clock hours of lab per week. All students meet together for the same lecture, but they meet separately in small lab sessions. The course is normally offered every spring semester and each class has no more than 30 students.

For each of the lifecycle topics, students were given a brief description of the relevance of the lifecycle phase and were then walked through an interactive example during class. Concepts were solidified with lab assignments. The instructional modules and lab materials were prepared by integrating the PCSE practices described above.

The topics presented, in order, were

- Week 1: Software engineering overview
- Week 2: Lifecycles
- Week 3-4: Eclipse, unittest
- Week 5-7: Construction using TDD
- Week 8: Analysis using scenarios
- Week 9-11: Design using CRC cards
- Week 12-14: Estimation and scheduling
- Week 15: Wrap up and feedback

5 Summary and Conclusion

For the past three years we were using PCSE to teach Software Engineering II. In our first two offering, Spring 2013 and Spring 2014, we introduced the students to the Eclipse platform outfitted with the PyDev Python development plug in. This provided the students an interactive development environment suitable for classwork, and equipped them with skills in working with a software tool in heavy use in industry. The students came to the course with weak development skills and no prior knowledge about Python. This necessitated an unexpected amount of time having to be devoted to using Eclipse and writing rudimentary Python. We had to allow extra time for each assignment, and dropped the last two programming assignments because of lack of time. Since the pre-requisite courses (CSC 211, CSC 212 and CSC 280) for CSC 437 Software Engineering II being taught using C++ language, the student feedback suggested us to use C++ with Eclipse rather than Python with Eclipse. Therefore in year-3, Spring 2015, we offered CSC 437 using Eclipse and C++.

We integrated the PCSE practices discussed above into instructional modules, each lasting one week. The modules were taught using examples that were worked through interactively during class. The students then worked on a programming assignment that incorporated the new instructional concept into concepts previously taught. This allowed the instructors to evaluate the students on their performance and to give points as to how they might improve.

6 Acknowledgment

This material is based upon work supported by the National Science Foundation under Grant No.1238912.

7 References

[1] U.S. Bureau of Labor Statistics, Occupational Outlook Handbook, 2010-11 Edition, http://www.bls.gov/oco/ocos303.htm#projections_data.

[2] U.S. Bureau of Labor Statistics, Occupational Outlook Handbook, 2015-16 Edition, http://www.bls.gov/oco/ocos303.htm#projections_data.

[3] Swamidurai, R.; Umphress, D., "Engaging students through practitioner-centered software engineering," In Proceedings of the IEEE SoutheastCon 2015, vol., no.,pp.1-5, 9-12 April 2015, DOI:10.1109/SECON.2015.7132985

[4] Watts S. Humphrey, PSP(sm): A Self-Improvement Process for Software Engineers, Addison-Wesley Professional, March 2005.

[5] Humphrey, Watts S. The Team Software Process (sm)(TSP (sm)). Carnegie Mellon University, Software Engineering Institute, 2000.

[6] Kent Beck, Extreme Programming Explained: An Embrace Change, Addison-Wesley, 2000, ISBN 0201616416.

[7] Hunt, John. "Feature-driven development." Agile Software Construction (2006): 161-182.

[8] Schwaber, Ken. "Scrum development process." Business Object Design and Implementation. Springer London, 1997. 117-134.

[9] Kruchten, Philippe. The rational unified process: an introduction. Addison-Wesley Professional, 2004.

[10] http://www.drdobbs.com/cpp/why-is-software-so-hard-to-develop/240168832

[11] Stephen Montgomery, Object-Oriented Information Engineering: Analysis, Design, and Implementation, Academic Press

[12] Beck, K. & Cunningham, W., A Laboratory for Teaching Object-Oriented Thinking, OOPSLA'89 Conference Proceedings (1989), pp. 1-6.

[13] Beck, K. Test-Driven Development by Example, Addison Wesley - Vaseem, 2003

[14] Velocity, http://guide.agilealliance.org/guide/velocity.html

[15] Humphreys, Gary, Project Management Using Earned Value, Humphreys and Associates, 2001, ISBN 0-9708614-0-0

A Computer Design and Assembly Active Learning Project

Muhittin Yilmaz, PhD,
Texas A&M University-Kingsville (TAMUK), 700 University Blvd., MSC 192, ENGC 322, Kingsville, Texas, USA

Abstract: *An active learning activity for a computer design and assembly with practical, technical and budgetary constraints was implemented to study real-life design concepts and project-based design outcomes, supplementing the traditional lecture style, implementing a potential interview process to synthesize the innovative design ideas and requirements, and relevant academic experiences as well as perspectives. The student teams conducted extensive research, analyzed hardware/software requirements as well as budgetary constraints, evaluated various design alternatives and conducted interviews with their own questions to subsequently improve the project design perspectives. The project evaluation has illustrated the effectiveness of the active learning activity to enhance the student long-lasting educational experience, i.e., the hybrid approach was instrumental to enhance student technical and interpersonal skills.*

Keywords—Computer architecture, active learning, practical design

1. Introduction

Instructional methodologies have targeted superior student learning experiences, especially under complex as well as real-life design problems with realistic constraints that may be challenging or inherently insufficient to cover satisfactorily in traditional teaching-centric educational formats. Active learning is one valuable approach to encourage students for intense research, information gathering and evaluation, and engineering design to enforce superior course content synthesis and comprehension levels. As the students lead the process, the active learning approach effectively engage students into practical as well as meaningful implementation activities of the course materials under new conditions, resulting in superior learning experiences [1, 2], higher student performances in science, engineering, and mathematics [3] and with a qualitative case-study for the USA analysis [4], in which group discussions, project as well as case study analysis, reading assignments, project papers and reflective writing are found to be the best strategies of active learning. As different active learning components were introduced, a journal-club experience as an active learning implementation in a natural science undergraduate course indicated better prepared student body to follow scholarly results in literature, where students were assigned literature reading sections complementing the regular course content and were asked to participate class discussions with a-priori written questions [5]. A medical course utilized an active learning component to achieve better relevant-rubric assessment, indicating superior learning outcomes [6]. An electronic measurement and instrumentation course involved a hands-on active learning implementation to enforce student theoretical understanding, resulting in better academic scores [7]. A civil engineering active learning activity involved management courses with a research framework and subsequent two-year research project activity, aiming to achieve better student learning experience and sustainable knowledge development [8]. A System engineering field study utilized the Lego Mindstorm as an active learning tool to reiterate handling unforeseen situations or obtaining a comprehensive system-level view of key system engineering lifecycle, yielding effective implementations of theoretical techniques covered in lectures [9]. A two-semester robotics curriculum was developed to offer various active learning components to students, including open-ended hands-on engineering design process, for superior academic and interpersonal skills [10-12]. A freshman Electrical engineering course included active learning components to design a protocol simulator and founded the final simulator product and active learning components popular, robust, and suitable for problem-based learning as well as student invention of their unique ideas [13].

Computer Architecture effective design and teaching approaches have also been studied. It was already recognized that proper applications, algorithms, architectures, and active learning were four important pillars of computer science and engineering education [14]. A three-year study about a computer architecture course illustrated significant improvements in student performances and satisfaction levels after active learning components of collaborative as well as problem-based learning methods integrated into the course curriculum [15]. Recent industrial and curricular developments have proposed project or case-study usage to apply theoretical knowledge gained in classrooms, as recommended by the electrical and computing discipline governing bodies, i.e., IEEE-Computer Society and Association for Computing Machinery (ACM) [16]. A digital logic course was successfully enhanced with active learning components such as hands-on laboratory activities and student team project design as well as implementation, resulting in superior student comprehension levels, critical thinking, problem solving and interpersonal skills [17]. Another digital logic course active learning implementation successfully integrated theoretical foundations to hands-on laboratory experiments and associated cost-effective visualization tools, resulting in superior student logic function demonstrations and deeper understanding of the digital logic circuits [18]. A course containing digital logic design and computer architecture utilized a major design project to effectively motivate students and to provide project-based learning benefits [19]. An active learning environment for an intermediate computer architecture course was also developed with suitable design activities in the classroom, laboratory implementations, and online assignments [20]. Computer design was introduced to students with appropriate levels of theoretical knowledge and significant hands-on design activities, challenging the student intelligence and knowledgebase for a successful completion [21]. An interactive reduced-instruction set computing processor and memory simulator was also developed to utilize active learning methods for superior theoretical concept comprehension such as multicache operations, pipelining and superscalar execution [22]. A survey study found that research-based instructional strategies including active learning are known to the electrical and computer engineering faculty teaching digital design, circuits, and electronics and that the class time implementation of the strategies seemed to be the most significant concern, possibly resulting faculty members utilize out-of-class activities [23].

This research describes an active learning activity as well as project-based design outcomes about designing a feasible computer with specific constraints during a senior-level Computer Architecture and Design course, at the Electrical Engineering and Computer Science Department of Texas A&M University-Kingsville (TAMUK), a minority serving institution. The active learning experience aimed to introduce real-life design concepts under significant constraints to supplement the traditional classroom instructions for a course in traditional lecture style, to implement a potential interview process to synthesize the project design requirements and the relevant academician educational experiences as well as perspectives for superior student problem-solving, communication, time-management and interpersonal communication skills.

2. The Computer Architecture and Design Course

The course is a senior-level core course for Computer Science and an elective course for Electrical Engineering students at the Electrical Engineering and Computer Science Department of TAMUK and is regularly offered every Fall semester. The course is a 3-credit course having 3-hour/week traditional lectures with the course materials distributed to the students. The computer architectural aspects were enhanced with organizational perspectives in terms of theoretical and practical concerns. The course actively supported two ABET outcomes: "(a) Ability to apply knowledge of mathematics, science, and engineering", and "(e) Ability to identify, formulate, and solve engineering problems", with pre-defined rubrics. One of the outcome (e) components included this active learning project of computer design and assembly, in addition to a separate architectural development activity.

The active learning project was implemented during the Fall semester and included twenty-one students, including minority groups such as Hispanics, African-Americans, and international students. All course students attended the traditional lectures, given by the course instructor, throughout the semester and participated in individual quizzes, short project assignments and exams, in addition to the active learning implementation of computer design and assembly.

The content coverage included the major computer architecture topics, as listed with approximate expected coverage durations: Introduction to Computer Architecture and Design (1 Week), Data Representation and Arithmetic Operations (2), Digital Logic Components (1), The instruction Set Architecture (2), Machine Languages (1), Datapath and Control (2), Memory Operations (3), Input and Output (2), and Pipelining (1). The course grading scheme and associated percentages included: Exam-I (20%), Exam-II (20%), Quizzes (15%), Projects (15%), and Final (30%). All exams were cumulative and closed book-notes, with the final exam allowing a half-page (8.5"-5.5") handwritten individual cheat sheet. The type of take-home exam was considered but not utilized, partially the time-consuming subjects being moved to the projects. Planned and/or pop-style quizzes were given both to motivate the student timely study as well as preparedness, to test the course concepts, and to ensure attendance. Quizzes had different time durations or point values depending on subject difficulty levels or amount of time needed. It was a class policy that missing a quiz due to an unexcused absence to receive a zero grade. There were two types of projects: several individual short projects and one team final design project, both of which aimed toward synthesis of theoretical concepts and engineering analysis, design and applications in terms of active learning activities. While short projects focused on topics covered, the final design project required a comprehensive study of a contemporary issue in detail. The final project technical findings and evaluation were announced to be assessed, at the instructor's discretion, during the oral presentation by the instructor and the classmates, if time permitted. The final course grades were to be assigned to reflect the subject comprehension scale that was determined by the instructor.

3. The Computer Design Project

After covering major computer theoretical backgrounds and demonstrating a sample computer design activity with several constraints to exhibit the proper synthesis and implementation steps in the first half of the semester, the active learning project of computer design and assembly was assigned to synthesize innovative design ideas as well as interview questions for a working computer design, during the second half of the semester, and the students were asked to form teams with three or four students while promoting diversity with maximum mutual interactions to nurture all relevant student skills. As most teams (5) included three students, one team included four while two teams included one student each, resulting in eight different computer design. The computer design project included specific project-related technical and budgetary guidelines, and required significant theoretical as well as practical research and appropriate implementation under new situations, commercial components and corresponding specifications, project design, management and leadership, communication, teamwork, and critical as well as innovative thinking skills, a potential interview with a computer technology professional, and a complete description of the practical computer and documentation. The student teams conducted extensive research, analyzed hardware/software requirements as well as budgetary constraints, evaluated various design alternatives and conducted interviews with their own questions to subsequently improve the project design perspectives. The student teams and the course instructor maintained a constant communication as well as interaction both to monitor the team design progress and to further explore design alternatives. The student teams were asked to document their project theoretical developments with clear justifications for their choices and implementations under technical and budgetary constraints in a technical report.

The active learning project was implemented as the semester project of the course during the second half of the semester, after exposing students with the needed major computer architecture components, both in theoretical perspectives and hands-on classroom demonstrations. The project instructions included the items, listed as (in their original format)

- The term project will be completed by teams of 3-4 students who will be assigned in the class. Based on the instructor's discretion (and time permitting), oral presentations may be required.
- The project initialization steps, detailed analysis of component selections/justifications for each specification (especially, when multiple products are available for the same specification), price comparisons, etc., **MUST** be explained clearly.
- All copied materials especially the images of the components used in this project must be referenced and credited.
- A separate budget table including the component names, quantities, prices, manufacturers and internet locations is to be prepared and submitted. The budget limit cannot be exceeded. Also, if the final cost is well below the budget limit, the spending selections and component choices must be fully justified.
- There will be one final team report that is to be written by using a word processor and all figures, tables, etc., are to be numbered, placed appropriately, referred in the text, etc. Clarity, organization and presentation of the report will be part of the final grade. **Possible extra credit**: the team prepares a list of interview questions (at least, more than ten meaningful questions) related to the project, reaches out to a professor with proper background, and conducts a short interview to explore different perspectives and choices. (Proper procedures for interview solicitation, execution, and reporting must be followed).

The project required a computer design according to the major specifications, listed below, and hypothetically assembling the working computer machine with available commercial products accordingly;

a) Each group has a total of $800 hypothetical budget that is to cover all components including the %8.25 Texas sales tax (You can assume that there is no shipping/handling costs, no rebates, and no required software purchase).
b) Each computer will have at least one Blue-Ray driver feature.
c) The computer will possibly be used for
    a. NASA Space exploration robotics system data analysis,
    b. Analysis of non-linear dynamics of robust systems, image processing, magnetic levitation systems, superscalar system simulations, etc.,
    c. Real-time hurricane monitoring sensory system data recording and webcasting,
    d. Doctoral student audio-visual class presentations for pharmacy and international economics disciplines.
d) The number of USB and Firewire ports as well as required hard-drives (including the RAID configuration) will be selected by the team with full

justification.

e) Three additional *unique features* that each team will decide and justify. Extra points will be awarded based on the originality and uniqueness of these features.

The project interview aimed to provide theoretical, engineering, historical, and commercial insights during the computer design by approaching a relevant professor, or any qualified computer industry professional, and utilizing the interview responses to explore different perspectives and choices. As there was a potential extra credit for the interview, the guidelines emphasized, at least, more than ten meaningful project-related questions to challenge the team technical skills and knowledgebase, and subsequent integration to the team computer design. Since the interview was potentially sensitive in terms of arrangements as well as professor availability, the teams were clearly instructed to follow proper procedures for interview solicitation, execution, and reporting. There were two teams who prepared interview questions, conducted interviews with an active computer science department seasoned instructor and a campus information technology director. Due to lack of space, the interview questions and responses were omitted.

The typical computer design required extensive team discussions about the given hardware and software specifications and their implied computer architecture requirements. Each team was allocated a total of \$800 hypothetical budget to cover all components including the %8.25 Texas sales tax. As the project envisioned an active learning experience such as designing a sample computer for a large organizational upgrade, no shipping/handling costs, no rebates, and no required software purchase were assumed as they were typically offered freely by computer manufacturers or the organization continuous utilization of the same or upgraded software components.

Although the course students were not exposed to the anticipated disciplines, namely, NASA space exploration robotics system data analysis, analysis of nonlinear dynamics of robust systems, image processing, magnetic levitation systems, superscalar system simulations, real-time hurricane monitoring sensory system data recording and webcasting, and doctoral student audio-visual class presentations for pharmacy and international economics, and associated software components, they were required to conduct research on each mentioned discipline and to identify major software and hardware requirements. It was observed that the project software application specifications as well as requirements were discussed extensively among the team members and were translated into plausible computer component specifications such as the processor speed or the amount of RAM for satisfactory performance while the project hardware specifications were considered under the budget constraint. The major data analysis and program execution requirements of the applications were translated into superior processors, larger as well as faster main memory, large hard drive storage, large input/output ports as well as storage capabilities, e.g., redundant array of independent disks (RAID) and superior motherboards. Data recording or displaying operations seemed to focus on superior graphics card, suitable and mostly larger monitors, wireless mouse as well as speaker selections. It was also noticed that the input/output operations for data exchanges were overlooked to some extent while basic input/output tools were treated as part of the motherboard selection with sufficient input/output capabilities, higher clock frequency as well as higher bus data rates. When a team faced a trade-off for a component as it was expected between mainly cost and performance dimensions, it was required to fully justify the selection in terms of the project requirements, budgetary constraints, technological perspectives, etc. As there were many different component selections by the student teams, an illustrative computer design components were described.

Teams appeare to consider plausible processor, main memory and motherboard combinations to support their research findings about hardware requirements under the budgetary constraints. Most teams identified a candidate processor for the expected heavy numerical calculations in project software tools and arranged the motherboard and main memory to support the superior processor operations. Although the main processor selection was Intel Core i-series and Core i5 was the top choice, as shown in Fig. 1, for their expected nominal and boosting performance levels, built-in security, superior visualization capability, and satisfactory cache sizes, there were Core i7-960, Advanced Micro Devices (AMD) FX-6300 Vishera 6-Core, AMD FX-8350 8-core processors, after some teams studied data analysis and 3D modeling benchmark results. The processor comparison was about, performance, cost, parallel and graphical processor performances, and manufacturer brand names.

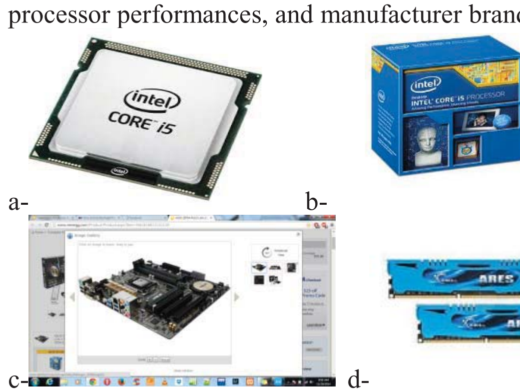


Figure 1. The Computer Design Project a)-b) Intel Core i5 Processor, c) the ASUS Z97M-PLUS Motherboard, and d) Main Memory (RAM)

The motherboard selection was also critical as there were many available options. However, teams tended to utilize many motherboard features to comply with the project requirements while saving some funds for the rest of the project specifications. Teams utilized motherboards with faster chipsets for program execution as well as data exchange, with appropriate processor socket type such as LGA 1150 for Intel Core i5 processor, with a number of USB 2.0 and USB 3.0 connections, potential support for HDMI, SATA, eSATA as well as RAID configurations, reliability and precision with proper set of chipsets, potential support for the largest main memory capacity with highest data transfer rates, with largest data communication protocols and associated slot availability, easy usage and update of Basic Input-Output System (BIOS) tools, different network communication capability, and online review scores, as one of the selection is shown in Fig. 1 for the ASUS Z97M-PLUS LGA 1150 motherboard - Intel Z97 HDMI SATA 6Gb/s USB 3.0 Micro ATX Intel chipset. Other motherboards included ASUS B85M-E/CSM, ASRock H94 PRO4, MSI 760GM-P34, ASRock 970 EXTREME4, ASUS M5A97 LE, ASUS Z97-PRO LGA1150, and ASUS P8B75-M/CSM LGA 1150.

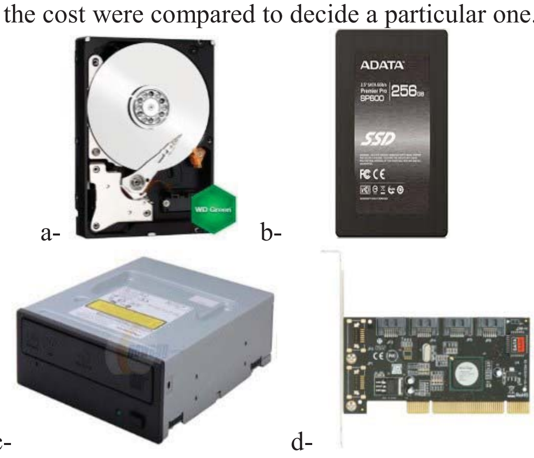
The third component was typically to identify the main memory (RAM) of the computer, supported by the motherboard clock frequency operation ranges. Since the larger the main memory the faster the computer response and there were several data operations specified by the project, teams seemed to put a lot of emphasis on utilizing the largest capacity while complying with the motherboard support and budgetary constraints. The RAM size was seen to be 8 Gigabytes with mainly DDR3 technology by all teams, as one illustrative RAM is shown in Fig. 1.

Once the three fundamental computer components were selected, teams seemed to be diverging in terms of their next components. There were high quality monitors, wired or wireless keyboard and mouse combos, and stand-alone as well as built-in speaker selections with varying output power levels, as a sample set is shown in Fig. 2. The monitors were selected by comparing the actual viewing space and associated resolution, superior display technology and power consumption levels, and were between 17-21.5 inches in size and with LCD or LED technology. Wireless keyboard-mouse combo was seen to offer flexibility during presentations, as part of the project specification, and was interfaced with the main computer via USB input/output protocol. Also, an optical mouse was preferred by one of the team to provide reliable interactions with the computer. Furthermore, some teams considered the 'wireless' keyboard-mouse combo as one of the unique feature. In addition, some teams preferred to emphasize the computer execution performance such that they seemed to obtain basic components, possibly to save some funds for high-end performance critical components.


a-
b-
c-

Figure 2. The Computer Design Project a) Monitor, b) Keyboard-Mouse, and c) Speakers

Long-term permanent storage options and faster data input/output operations included magnetic disk drives up to 1 Terabytes, solid state drives, a Blue Ray driver that was specifically instructed as part of the project specification, and RAID implementations, shown in Fig. 3. As solid state drives were seen as a 'unique' feature, it was recognized to increase both the performance of the computer considerably and the cost of the computer, resulting in a delicate trade-offs such that some teams preferred large capacity hard disk drives instead of solid state drives. The RAID 0-1-5 implementations were obtained by preferring motherboard feature or an external RAID controller. In a design, up to four 250 Gigabyte magnetic disk drives were also utilized in the RAID 10 configuration, resulting in a 500 Gigabyte effective storage size in RAID 1, both for superior data input/output and reliable real-time data backup performance levels. Although there was only limited discussion about the required Blue Ray drive as it is a standard component, the data transfer rate, the read-write speeds, the manufacturer, and the cost were compared to decide a particular one.



a- b-
c- d-

e-

Figure 3. The Computer Design Project a) Magnetic Disk Drive, b) Solid State Drive, c) Blue Ray Drive, d) RAID Controller, and e) Video Card

Graphics card selection also seemed to be important both in terms of anticipated higher graphical operations and cost of the component. Out of built-in or external graphics cards, the student teams tried to obtain the largest graphics card memory while monitoring the project budget as graphics cards are very expensive among many required computer components, resulting in mainly external cards with 1-2 Gigabytes memory size. Video card CrossFire support was also considered as a unique feature to utilize a number of graphics cards, possibly having newer and older technologies, both to improve graphics performance and to maintain lower cost of the computer. A sample graphics card is shown in Fig. 3.

a- b- c-

Figure 4. The Computer Design Project a) Computer Case, b) Power Supply, and c) Webcam-Microphone

Computer cases, a sample shown in Fig. 4, were mainly chosen by studying the cost, the number of internal and external expansion slots, form factors, the number of available cooling fans and ventilation options, aesthetically-important LED lighting, and sturdy building material seemed to be supporting decision factors. Power supply of a computer, shown in Fig. 4, was also seen as an important design issue to provide sufficient and reliable amount of power for computer components. As the sufficient power capacity was seen to be the main factor to consider, the cost, the number of power cable outputs as well as connector types, power efficiency ratings such as Bronze Certified for more that 85% and Gold Certified for more than 95% power efficiency levels, reputation of the manufacturer, compatibility with processor specifications were observed to be the supporting factors for team decisions. The Webcam-microphone combo in Fig. 4 was also proposed to support the required project tasks, including webcasting operations. The webcam rotational capability as well as high-resolution sensor features were the main consideration points.

In addition, one team also proposed a laser printer as one of their ‘unique’ computer machine feature. Another team proposed an integrated processor with a graphical processor unit as part of their ‘unique’ feature set. In addition, IEEE 1394 (Firewire) PCI card was also proposed as a ‘unique’ feature, that was significant due to the Firewire input/output protocol superiority for large data transfer operations.

4. Evaluation

The computer design active learning project contributed to superior student long-lasting educational experiences under open-ended real-life design challenges under a number of constraints, highlighting its effectiveness.

The team performance and project effectiveness were quantitatively and qualitatively evaluated in a number of different methods, including the technical project reports according to the rubric-based student outcome assessments, verbal reports by individual students or student teams reiterating the project outcome, and classroom as well as office-hour discussions. The participating student open-ended computer design performances as well as report content presentation or justification qualities strongly suggests a successful implementation of the active learning project, effectively providing real-life educational design experiences. The corresponding class average for the computer design project was 92.3, indicating a high level of student research and preparation performances, with a range of 83-105 (after extra credits). Thus, the hybrid approach, including both traditional and active learning components, was observed to be instrumental to nurture student problem-solving, critical thinking, time-management, technical writing, and interpersonal communication skills to integrate the computer architecture theory with real-life design experiences.

5. Conclusions

An active learning project to offer an open-ended computer design project was successfully developed in the course and implemented by the students. As the students were exposed to the important computer design principles and available commercial components as well as associated selection process justification, the active learning activity was able to greatly enhance student comprehension levels, alleviating the lack of hands-on laboratory facilities.

Acknowledgment: The author thanks the course student team members and two computer industry professionals. The final reports were partially utilized for the presentation.

6. References

[1] C. C. Bonwell, J. A. Eison, "Active Learning: Creating Excitement in the Classroom," Association for the Study of Higher Education (ASHE)-ERIC Higher Education Report No 1, Washington, D.C, 1991: The George Washington University, School of Education and Human Development.
[2] G. Wiggins, J. McTighe, "Understanding by Design," Association for Supervision and Curriculum Development (ASCD), Alexandria, VA, 1998.
[3] S. Freeman, S. L. Eddy, M. McDonough, M. K. Smith, N. Okoroafor, H. Jordt, M. P. Wenderoth, "Active Learning Increases Student Performance in Science, Engineering, and Mathematics," Proceedings of the National Academy of Sciences of the United States of America, 111 (23), June 2014, pp. 8410-8415.
[4] Y. F. Chan, G. K. Sidhu, L. F. lee, "Active Learning in Higher Education: A Case Study," 2014 Taylor's 7th Teaching and Learning Conference, 2014, pp. 519-526.
[5] K. L. Anderson, "Active Learning in the Undergraduate Classroom: A Journal-Club Experience Designed to Accentuate Course Content," The American Biology Teacher, 78 (1), January 2016.
[6] A. M. Walton, A. N. Isaacs, A. T. McFarland, L. M. Czosnowski, S. A. Nisly, "Design of drug-induced diseases elective utilizing active learning," Currents in Pharmacy teaching and Learning, 8 (1), January-February 2016, pp. 90-95.
[7] P. Nikita, P. Jyoti, B. Shraddha, P. Preeti, S. Raghavendra, N. C. Iyer, "Active Learning in Electronic Measurements and Instrumentation Course through Hands-on," Journal of Engineering Education Transformations, Special Issue, 2016.
[8] Lam EWM, Chan APC, Chan DWM, "Active Learning Strategies for Building Engineering Students for Sustainable Knowledge Development," Civil Engineering and Architecture, 3 (5), 2015, pp. 136-142.
[9] C. B. Nielsen, P. Adams, "Active Learning via LEGO MINDSTORM in Systems Engineering Education," 2015 IEEE International Symposium on Systems Engineering (ISSE), September 28-30, 2015, pp. 489-495.
[10] M. Yilmaz, S. Ozcelik, N. Yilmazer, R. Nekovei, "Design-Oriented Enhanced Robotics Curriculum," IEEE Transactions on Education, 56 (1), February 2013, pp. 137-144.
[11] M. Yilmaz, S. Ozcelik, N. Yilmazer, R. Nekovei, "A Two-Semester Project-Based Robotics Curriculum," 2012 American Society of Engineering Education Annual Conference and Exposition (ASEE-2012), June 10-13, 2012, San Antonio, TX, USA.
[12] M. Yilmaz, S. Ozcelik, N. Yilmazer, R. Nekovei, "Forestry Robot Design," 2014 American Society of Engineering Education Annual Conference and Exposition (ASEE-2014), June 15-18, 2014, Indianapolis, IN, USA.
[13] D. A. J. Pearce, "The Development of a Protocol Simulator for Active Learning," International Journal of Electrical Engineering Education, November 5, 2015.
[14] R. M. Panoff, "The Four A's of CSE Education: Application, Algorithm, Architecture, and Active Learning," Computing in Science & Engineering, 1995, pp. 6-7.
[15] O. Arbelaitz, J. I. Martin, J. Muguerza, "Analysis of Introducing Active Learning Methodologies in a Basic Computer Architecture Course," IEEE Transactions on Education, 58 (2), May 2015, pp. 110-116.
[16] Association for Computing Machinery/IEEE-Computer Society (ACM/IEEE-CS), "Computer Science curriculum 2008: An interim revision of CS 2001," Report from the Interim Review Task Force, December 2008, located at http://www.acm.org/education/curricula/ComputerScience2008.pdf, Last accessed on January 14, 2016 .
[17] N. Yilmazer, M. Yilmaz, R. Seker, "A Project Based Hands-on Digital Logic Course," 2011 American Society of Engineering Education Annual Conference and Exposition (ASEE-2011), June 26-29, 2011, Vancouver, B.C., Canada.
[18] J. Pang, "Active Learning in the Introduction to Digital Logic Design Laboratory Course," 2015 American Society of Engineering Education Zone III Conference.
[19] C. M. Kellett, "A Project-based Learning Approach to Programmable Logic Design and Computer Architrecture," IEEE Transactions on Education, 55 (3), August 2012, pp. 378-383.
[20] J. Herath, S. Ramnath, A. Herath, S. Herath, "An Active Learning Environment for Intermediate Computer Architecture Courses," 2002 Workshop on Computer Architecture Education (WCAE), Alaska, USA.
[21] D. C. Hyde, "Teaching Design in a Computer Architecture Course," IEEE Micro, 20 (3), May/June 2000, pp. 23-28.
[22] M. I. Garcia, S. Rodriguez, A. Perez, A. Garcia, "p88110: A Graphical Simulator for Computer Architecture and Organization Courses," IEEE Transactions on Education, 52 (2), May 2009, pp. 248-256.
[23] J. E. Froyd, M. Borrego, S. Cutler, C. Henderson, M. J. Prince, "Estimates of Use of Research-based Instructional Strategies in Core Electrical or Computer Engineering Courses," IEEE Transactions on Education, 56 (4), November 2013, pp. 393-398.

FilmTies: A Web-based Tool for Teaching 3-D Cinematography

Donald R. Schwartz [1], William Bares [2]
[1]School of Computer Science and Mathematics, Marist College, Poughkeepsie, NY, USA
[2]Department of Computer Science, College of Charleston, Charleston, SC, USA

Abstract - *College film programs which emphasize production of film provide hands-on practice in skills including i) camera operation to choose and film camera angles and ii) editing available footage to select and order the shots to tell the story. Film programs also incorporate group critique sessions in which students write commentaries on the works of other students. The motivation of the FilmTies application is to enable college film programs to offer more opportunities for practicing key skills of camera composition, editing, and critiques in a collaborative Web-based environment which requires no specialized lab equipment. We propose that film classes adopt the FilmTies system to offer additional practice prior to the more time-consuming and costly exercises using real camera equipment. This paper presents an overview of the FilmTies environment.*

Keywords: Innovative Use of Technology in the Classroom, Collaborative Learning, Computer and Web-based Software for Instruction

1 Introduction

Cinematographers begin learning their craft by mastering commonly-accepted guidelines for picture composition and editing. Two of the more familiar guidelines are the Rule of Thirds and the 180-degree rule. The Rule of Thirds suggests framing subjects so they appear on imaginary horizontal and vertical lines that divide the frame into thirds. The 180-degree rule suggests keeping the camera on a consistent side of an imaginary line between two characters so that each consistently appears in either the left or right side of the frame. [1]

College film programs which emphasize production of film provide hands-on practice in skills including i) camera operation to choose and film camera angles and ii) editing available footage to select and order the shots to tell the story. However, intensive hands-on practice can require significant investment in equipment that includes film studio sets, cameras, lights, dollies, booms, jibs, editing suites, blank film stock, and film processing costs. Additionally, students typically pay lab fees and insurance fees to protect against lost or damaged equipment. Film programs also incorporate group critique sessions in which students write commentaries on the works of other students. The motivation of the FilmTies application is to enable college film programs to offer more opportunities for practicing key skills of camera composition, editing, and critiques in a collaborative Web-based environment which requires no specialized lab equipment. We propose that film classes adopt the FilmTies system to offer additional practice prior to the more time-consuming and costly exercises using real camera equipment.

2 Background

Existing cinematography assistants designed to teach guidelines such as the Rule of Thirds or the 180-degree rule utilize case-based or procedural algorithms to generate suggested shots which conform to a guideline, or to automatically determine whether a student's work satisfies that guideline [2]. However, mastery of these guidelines is not sufficient for creating artful cinematography since the impact of a shot also depends on the narrative context and individual style. Additionally, case-based or procedural methods do not afford an easy way for instructors to extend these systems since they involve specialized programming to determine how the system would respond to a student for each possible anticipated scenario. For example, to teach the Rule of Thirds, a rule-based system would program a rule that stated that if a character did not lie along one of the imaginary lines that split the frame into thirds, the rule-based assistant would suggest an example composition in which the character does lie along the imaginary Rule of Thirds lines.

FilmTies is similar to Director's Lens [3] in that the user can browse a gallery of suggested cameras from which to choose for filming a shot. In Director's Lens, suggested cameras are produced entirely by an artificial intelligence agent. FilmTies introduces collaborative, social, and database functionality. In FilmTies, human users post suggested alternate or improved shots as comments to a given first shot. Consequently, suggestions are much more accurately related to specific narrative contexts and include human-written comments that explain why an alternate shot might be chosen.

Existing Web-based social systems teach web page design and visual layout in social Web-based environments that encourage sharing of examples and critiques [4][5]. The OSCAR system aids photographers by retrieving exemplary images from its database by searching for exemplars having similar subject matter and composition features [6]. FilmTies differs from these prior works in that the content being shared represents the composition properties that can be automatically adapted to different 3D scenes using a constraint-based camera solver [7].

3 Overview of the FilmTies Environment

Upon connecting to the FilmTies website (Figure 1), users are invited to create an account (or to log in, if their account already exists). Users enter their ID and password, then hit Login.

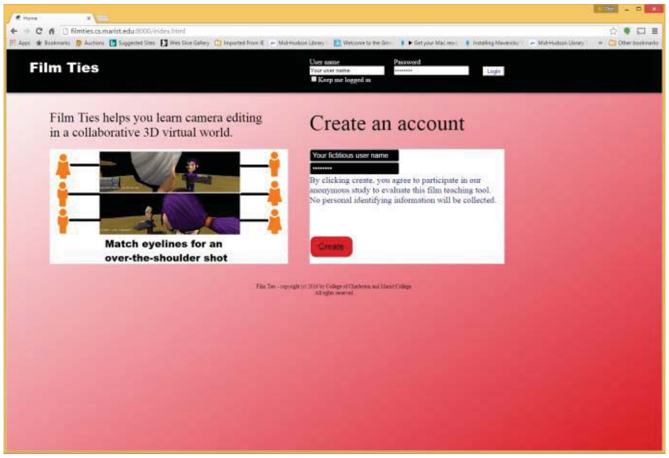


Figure 1: The Login Screen

After successfully logging in, the user will choose from a menu of available movies (Figure 2). This list of movies or movie clips will have been set up in advance by the instructor(s). Each movie is defined by a timeline of animated 3D events which are rendered in real time using WebGL, which makes it possible for students to freely move the virtual camera to choose how they would film the events in a movie.

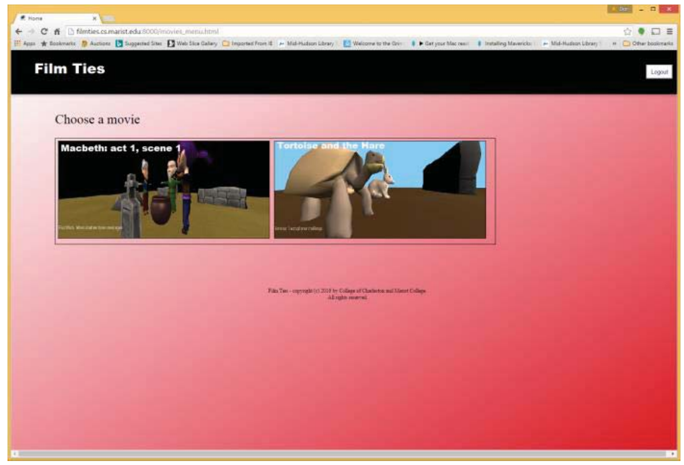


Figure 2: Available Movies

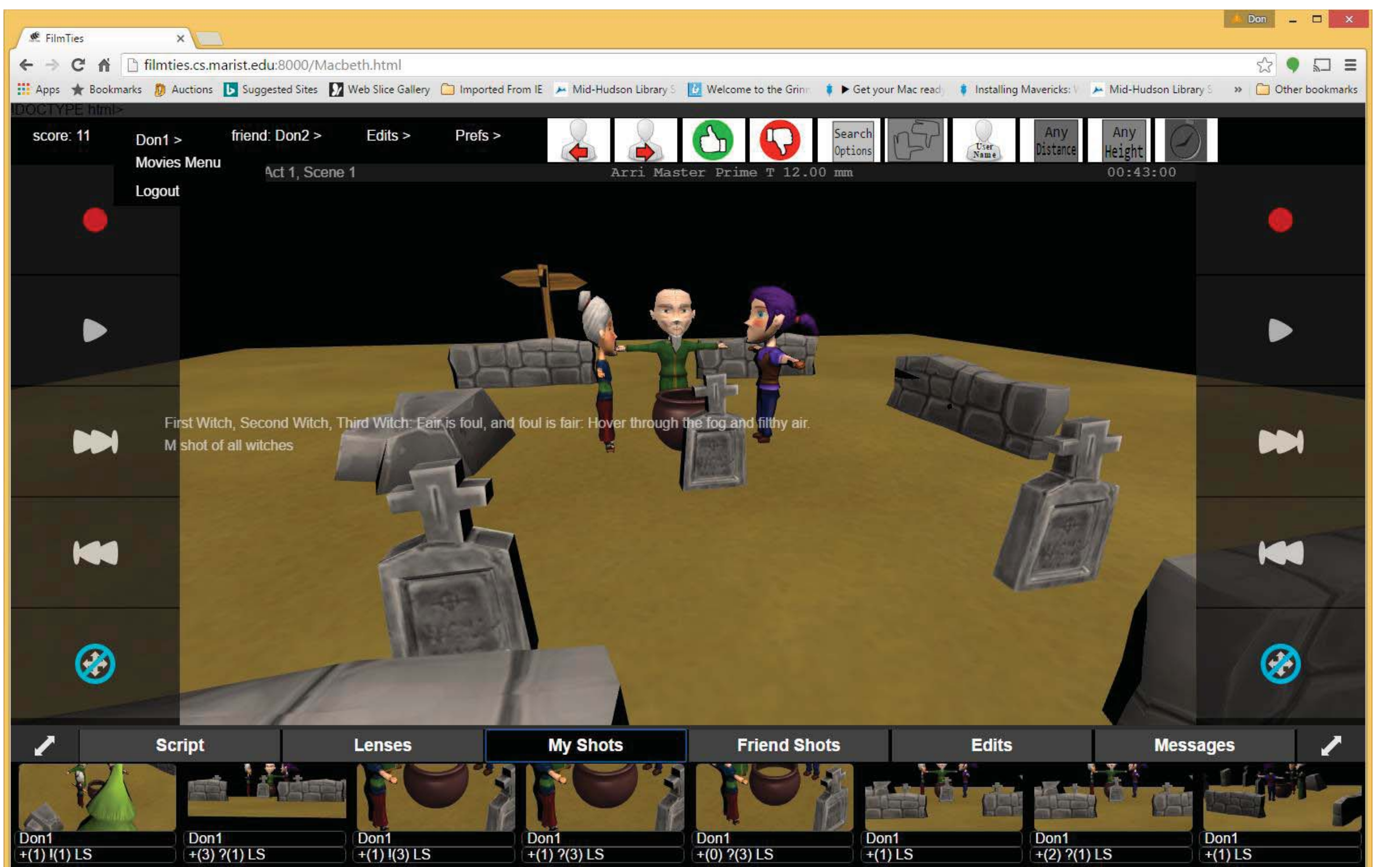


Figure 3: The Main Screen and My Shots Gallery

After choosing a movie, the user is shown the main FilmTies screen (Figure 3). The VCR-type controls along the left- and right-hand sides allow the user to play, pause, skip forward and skip backward through a list of narrative events. A narrative event is defined by its start time, names of characters involved, and dialog. For example, the opening scene of Macbeth at time 27 seconds has the character "third witch", who begins speaking the line of dialog "There to meet with Macbeth". Touching the red Record button allows the user to capture a snapshot, which is the 3D visual composition of the subjects seen in the 2D frame. It is defined by the configuration of the subjects in the 3D scene and by its composition, which includes the relative locations of the characters in the frame, the size of the characters within the frame, and their relative orientation with respect to the camera lens. Each recorded snapshot appears in the My Shots gallery filmstrip at the bottom of the screen, as shown in Figure 3.

The Script tab displays the dialog and optional shooting notes (provided by the instructor and/or director) for the current moment of time in the movie. The shooting notes allow the instructor to tell the student how to film a given moment of the movie. For example, at time 27 seconds, the instructor may suggest that the student film an over-the-shoulder (OTS) shot showing the third witch's face. The shooting notes can be customized, so that multiple learning outcomes can be taught using the same movie scene. The script should establish the standard by which student submissions are to be evaluated. The students use this script to set up a corresponding list of snapshots. For example, using the camera controls, the students are able to pause the movie and pan the camera around to get a shot of the character's face while that character is speaking.

The Lenses tab (Figure 4) allows the user to choose from among a set of predefined lenses, which vary by focal length. The tab displays a list of images that preview how the current camera view would be seen through each lens. This gives the user the ability to refine the level of zoom (close-up / medium / long shot) to be used for a given snapshot.

Figure 4: Available Lenses

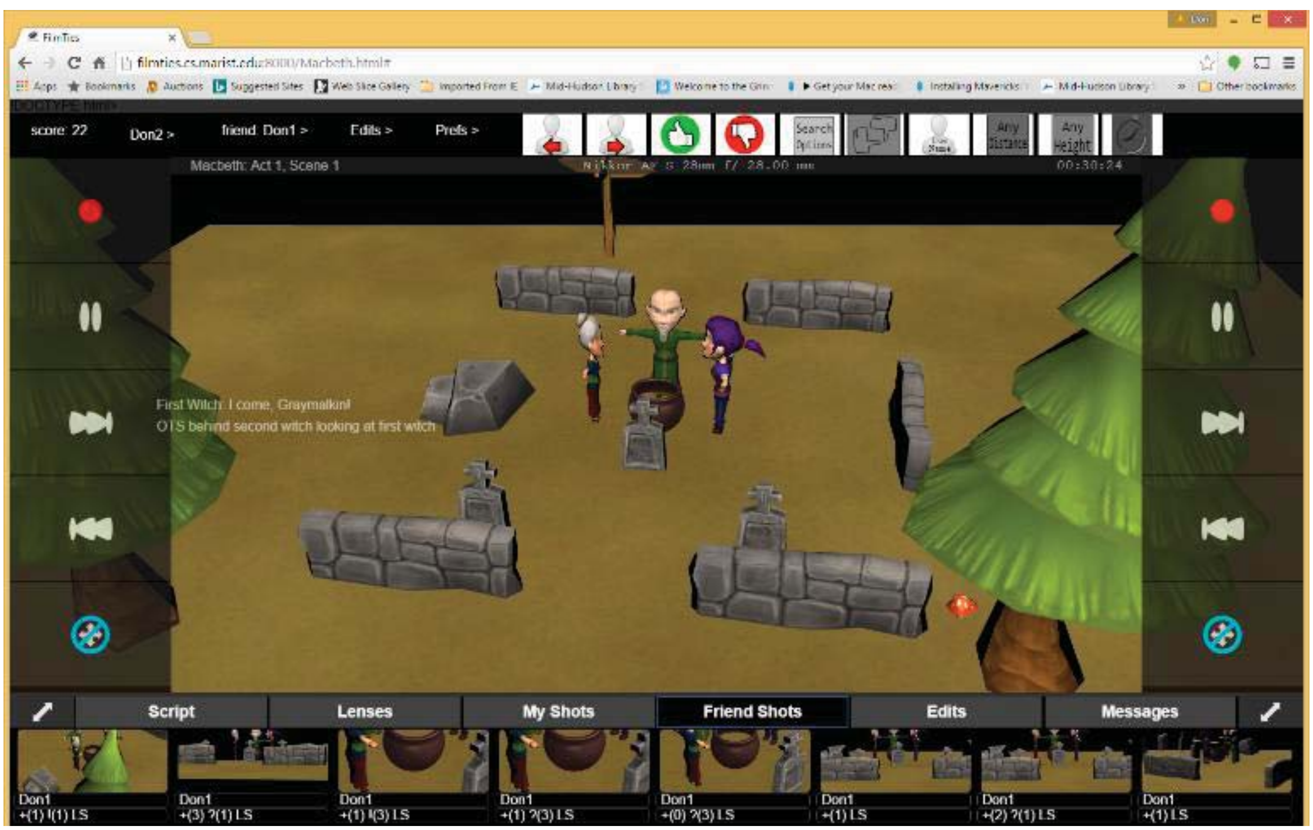


Figure 5: Friend Shots Gallery

The My Shots tab (Figure 3) displays all of the snapshots created by the current user. At any time the user can click to open the My Shots gallery to choose and re-use a previously created composition. For example, in editing a conversation between two characters, the user may choose to cut repeatedly between the same pair of close-up shots of the two characters. The system gives the user the ability to sort the snapshots according to various properties, including shot size (close, medium, long), camera height (low, eye-level, and high), popularity score (as voted on by other users), and time created.

The Friend Shots tab (Figure 5) allows the user to browse through all of the snapshots created by a given "followed user". Users can choose to "follow" other users, which means that they wish to have access to the followed users' ("friend's") snapshots. Selecting the Friend Shots tab will reveal that user's palette of snapshots. Like those in the My Shots gallery, the snapshots in this gallery can also be sorted according to the same set of criteria.

The Edits tab shows the movie frames (snapshots) recorded to film specific moments of time. They by default appear sorted by ascending order of movie time. The Edit represents the user's choice of shots recorded to film each specific moment of time in the movie.

The Messages tab (Figure 6) displays all of the feedback messages related to the most-recently selected snapshot (from My Shots or Friend Shots) or movie frame (from the Edits gallery). The usage of the Messages tab is described in more detail in section 5.

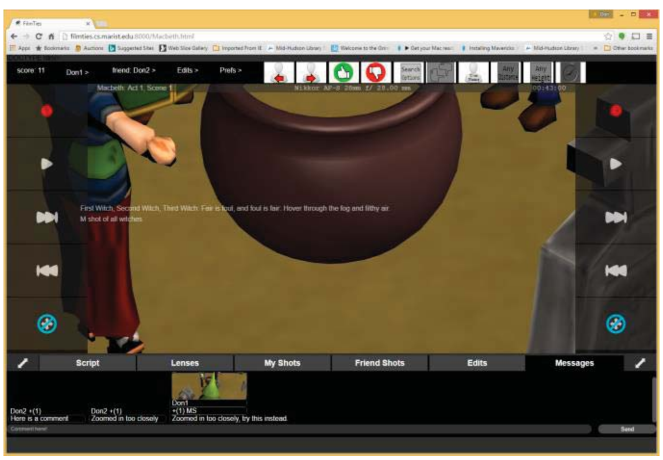


Figure 6: Messages Tab (used for feedback)

The labels at the top left portion of the screen identify the name of the current user and her social score (an indication of the number of "likes" this user received for her set of snapshots), allow the current user to return to the movie menu or to logout, and (Figure 7) identify the currently-selected friend/followed user (and allow the user to add/delete a followed user, including the teacher). The Edits dropdown (Figure 8) allows the user to create a new edit, open one of her existing edits, open a friend's edit, or move to a particular moment of time within the animation. The Preferences dropdown (Figure 9) provides the ability to loop the playback, to allow the movie to auto-pause at the next narrative point, to have the AI system automatically suggest comments based on comments made for similar snapshots, or to skip forward to the next script beat, item in the shot list, or time. It also allows the user to set the sort feature for the current gallery.

Figure 7: Options for Friend/Followed User

Figure 8: The Edits Options

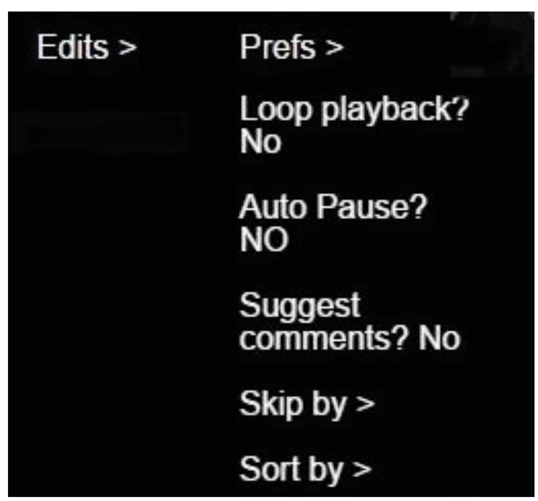


Figure 9: Preferences

ISBN: 1-60132-435-9, CSREA Press ©

The icons at the top right portion of the screen (Figure 10) allow the current user to select the previous or next followed user/friend, vote-up ("like") or vote-down ("dislike") the currently selected snapshot or movie frame, or choose a search element for the currently-displayed snapshot gallery.

Figure 10: Available Shortcut Icons

4 Providing Feedback on Friend Shots

While perusing a friend's shots or movie edit, the user can offer feedback, including voting up (liking) or down (disliking) the current snapshot, or, more importantly, providing feedback messages about the composition. Feedback messages can include descriptions of the best features of the snapshot, suggestions for improving the composition of the snapshot, or any other feedback the user wishes to share.

To provide a feedback message, the user first selects a snapshot, for example from the Friend Shots gallery (Figure 5). That snapshot then appears in the current camera view. For simple text feedback, the user then touches the Messages tab and enters the feedback message, which will then be associated with this snapshot and will immediately be made available under that friend's My Shots gallery. A special indicator (!) will flag that snapshot so that the friend is made aware that feedback has been received. (The number after the exclamation point indicates the number of feedback messages received for that snapshot.) The friend then selects that snapshot, then touches the Messages tab to see the feedback, as shown in Figure 11.

Figure 11: Feedback from Another User

Because snapshots represent the 3D composition of the shot (as opposed to simply being a 2D picture), FilmTies provides the ability for the feedback to include a newly constructed snapshot to illustrate the suggestion being offered. For example, suppose the teacher is evaluating a student's snapshots. For a particular snapshot, the zoom level might be significantly off, making it harder to identify which character is speaking. The teacher might provide a feedback message indicating these problems. However, because of how the composition of the snapshot is stored, the feedback message can also include a suggested snapshot composition created by the teacher to show a much more useful composition. To provide feedback that includes a suggested alternate snapshot composition, the user selects a snapshot as before, but this time she adjusts the camera location/zoom/etc. and touches the red Record button to capture the alternate snapshot. She then provides an optional text comment to describe why the alternate shot might be a better choice. Again, the friend is notified that an alternate snapshot has been provided for a particular snapshot in the My Shots gallery.

5 Example of a Typical Usage Scenario

The teacher will select a Movie from the current library of movies. She will develop a script with dialog, directing notes, establishing shots, etc. Students will create their own edit of the movie based on these notes, taking snapshots for each indicated moment of the movie (according to the shooting notes) to demonstrate their work. For example, as lines of dialog appear in the script, students might pause the movie, adjust the camera so that the speaker is in the middle of the frame, capture that snapshot, re-start the movie and proceed to the next narrative event and continue. Upon completion, students will have their Edit gallery containing a full slate of snapshots for the entire movie. At this point, a good exercise might be for the students to assess each other's work by adding the other students to their Friends list, then going through each Friend Shots gallery and providing feedback to one another's snapshots. This can help the student (since she'll see how other students completed the assignment) and the friend (who will hopefully get useful feedback about possible improvements to their work).

In a large class, it's quite possible that several students might make the same sort of mistake when depicting a particular moment of time within the movie. In most systems, the teacher would be required to provide the same feedback to each student, one student at a time. Because of how FilmTies captures snapshot compositions, we have been able to develop an AI system that can return a set of snapshots whose composition matches a given snapshot. If she so desires, the teacher can have the system automatically associate the current student's feedback message and alternate snapshot with all of the snapshots that are identified as "similar". This feature has the benefit of being able to significantly reduce the amount of time and effort required to provide useful feedback to all students.

6 Conclusions

The FilmTies system is a collaborative, web-based environment that can help college film programs give their students additional opportunities to practice key film-making skills, including camera composition, editing and critiques, without requiring specialized lab equipment. Students can use our system to practice camera operation to choose camera angles, edit existing footage to select and order shot sequences, and provide (and receive) feedback about other students' works. This feedback can include text as well as alternate shot suggestions, which can be created by changing the camera angles, lenses, shot distance, etc. Instructors can use the system to create assignments by selecting movies, providing directing notes, and creating shot lists and other details for the students to follow. Using FilmTies' collaborative features, instructors can also provide detailed feedback about student submissions, including the ability to have the system automatically associate a given student's feedback message and alternate snapshot with all of the other students' snapshots that are identified as "similar".

We evaluated the instructional effectiveness of the system by conducting a pilot study with ten Computer Science students at the College of Charleston. Seven out of ten demonstrated improved accuracy between a pre- and post-test to identify images that illustrated the Rule of Thirds. The pre- and post-tests consisted of a series of images and asked students to identify those images that correctly illustrated the Rule of Thirds.

7 Future Work

We are currently extending the FilmTies system to support classroom-management features to enable instructors to create class groups, invite students to participate in a class group, create shot list lessons, and track student completion of lessons. We are developing Web-based graphical authoring tools so that instructors can more easily create the script data files which define the 3D animated scenes, movie script dialogs, and shooting notes instructions.

8 References

[1] Blain Brown. Cinematography: Theory and Practice: Image Making for Cinematographers and Directors, 2nd Edition, Focal Press, 2011, 384 pages.

[2] Nicholas Davis, Alexander Zook, Brian O'Neill, Brandon Headrick, Mark Riedl, Ashton Grosz, and Michael Nitsche. 2013. Creativity support for novice digital filmmaking. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '13). ACM, New York, NY, USA, 651-660. DOI=http://dx.doi.org/10.1145/2470654.2470747

[3] Christophe Lino, Marc Christie, Roberto Ranon, and William Bares. 2011. The director's lens: an intelligent assistant for virtual cinematography. In *Proceedings of the 19th ACM international conference on Multimedia* (MM '11). ACM, New York, NY, USA, 323-332. DOI=http://dx.doi.org/10.1145/2072298.2072341

[4] Brian Lee, Savil Srivastava, Ranjitha Kumar, Ronen Brafman, and Scott R. Klemmer. 2010. Designing with interactive example galleries. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '10). ACM, New York, NY, USA, 2257-2266. DOI=http://dx.doi.org/10.1145/1753326.1753667

[5] Kurt Luther, Jari-Lee Tolentino, Wei Wu, Amy Pavel, Brian P. Bailey, Maneesh Agrawala, Björn Hartmann, and Steven P. Dow. 2015. Structuring, Aggregating, and Evaluating Crowdsourced Design Critique. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing* (CSCW '15). ACM, New York, NY, USA, 473-485. DOI=http://dx.doi.org/10.1145/2675133.2675283

[6] Lei Yao, Poonam Suryanarayan, Mu Qiao, James Z. Wang, and Jia Li. 2012. OSCAR: On-Site Composition and Aesthetics Feedback Through Exemplars for Photographers. *Int. J. Comput. Vision* 96, 3 (February 2012), 353-383. DOI=http://dx.doi.org/10.1007/s11263-011-0478-3

[7] Christophe Lino and Marc Christie. 2015. Intuitive and efficient camera control with the toric space. ACM Trans. Graph. 34, 4, Article 82 (July 2015), 12 pages. DOI=http://dx.doi.org/10.1145/2766965

ISBN: 1-60132-435-9, CSREA Press ©

Analysis of laboratory practices and computer simulations to identify a better way of learning

Rosete F. Juan C. [1], Guzmán R. Miguel A.[2], Estrada R. Felipe [3]

1 Mechanical Engineering Department, Technological Institute of Queretaro,
Av. Tecnológico S/N Esq. M. Escobedo, PC 76000, Querétaro, Qro, México
jc_rosete@yahoo.com.mx
2 Computer Engineering Department, Technological Institute of Queretaro

3 Computer Engineering Department, Technological Institute of Queretaro

Abstract. –*This paper exposes the results of a series of tests applied to students of Mechanical Engineering from the Technological Institute of Queretaro, Mexico, with the purpose to identify the level of understanding of the lectures of mechanics of materials while applying computing simulations. The results were useful to compare the performance of students that developed along the same time laboratory practices and experiments without using the simulators.*

Physics and strength of materials are topics that continuously have a large number of low notes from the students and the failure rate in both topics usually means desertion in the engineering program, given the impossibility of the student to reach the required competences in the topic.

For more than five years, a large amount of money has been invested in order to acquire educative software to help the education of the future engineers. The main idea was to help the activity of the professor by providing software resources that could explain the basis of the physics and behavior of the materials under stress.

After the experience with the simulators as the main way to provide the students with the required experiences in this field of knowledge, it is mandatory to review the results of such experience in order to validate the impact and the convenience to continue applying virtual tools instead to develop laboratory practices.

Keywords: Simulators, laboratory practices, physics education, mechanics of materials education.

Introduction

High desertion rates and low notes are typical problems that several countries experiment continuously around the world in their day-by-day educative process. It is important to understand the conditions that produce these problems, due to the huge amount of economic resources that every year the education bureau loses for this concept.

The number of students that every year stop studying an engineering career is high in relation to other fields of knowledge; most of the times, the simple answer to explain this problem is to state that the student did not had the required knowledge bases or the necessary motivation to continue studying. Nonetheless, it is important to find the main reasons that lead to the failure of a student.

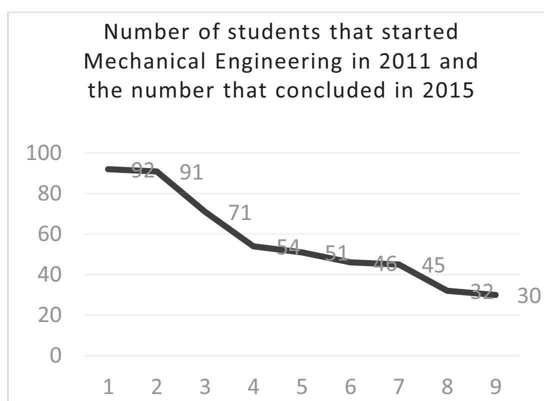


Figure 1. Number of students per semester

During more than 5 years, several software tools have been included in a large number of engineering lectures looking for a better understanding of the topics established in the official syllabus. The reasons behind this decision are:

1. A low number of hours per week by every lecture
2. A large number of students per group (more than 30)
3. A high number of groups to be attended per each teacher (5 groups in average)
4. Not enough equipment to do the laboratory tests. (One tensile stress machine, one torque machine, one fatigue test machine, one lathe to do the specimens…)
5. Absence of a laboratory assistant to help the students to do the required practices.

Due to these situations, the use of computer simulation of experiments and the teaching practice of using software tools were considered as the best way to improve the understanding of the topics reviewed in the classroom.

The first activity deployed for the Mechanical Engineering Department was to establish the lectures that have the largest number of students with low grades and the level of impact with the progressive lectures. From this analysis, it was possible to conclude that physics has been the study program with the largest number of students with low notes, condition that could explain the problems that the student experiment just one year later in Mechanics of materials.

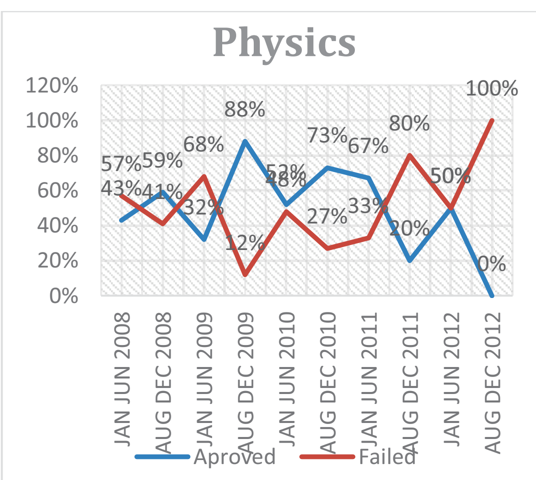


Figure 2. Physics lecture behavior in Mechanical Engineering

One the remarkable situations from the study is the up & down behavior of the grades obtained by the students with one semester having a high level of low grades, while the next one shows a better behavior only to be followed by another low level semester; the two semesters of year 2012 were particularly significant because 100% of the students failed in the lecture.

This condition is the result of several factors acting simultaneously, including a large number of lectures per day, the cultural influence of mass communication media, a lack of criteria to discern the quality of information found in the internet, and lots of misconceptions amongst others factors.

An important factor that we have to consider is the fact that the actual generations of students have access to a large quantity of educative resources. Apps available by mobile devices, lectures on line about almost every topic of an Engineering program, recorded lectures in multimedia format for several kinds of platforms and a higher quantity of books than ever before; so, it is not clear the reason of a low grade in a lecture.

2 Using software to simulate materials behavior

Given that the recommendations made for the Bureau responsible of certifying the quality of Engineering programs in Mexico (CACEI), it was detected the necessity to increment the number of equipment devices in laboratories in order to attend the number of students in the program. Under this condition, the investment in additional equipment and educative resources was mandatory. However, the financial situation of the Institute did not allow the acquisition of the totality of machinery proposed, so the final decision was to buy software to help in the lectures.

One of the lectures that worked with software simulation was Mechanics of Materials. This topic is fundamental in order to identify the mechanical conditions that can produce the failure of a material under the action of loads; so, it is the first step in the way to analyze the reason of failure in a structural component, identify the level of risk in a design and to design machinery and machine components.

2.1 Working with software in Mechanics of materials

The program of study of the lecture includes the next topics:

- Stress and strain of materials under loads
- Beams, normal and shearing stress, deflection and structural conditions
- Torque and power transmission
- Stress combination
- Failure theories

For all the topics included in the program, it is possible to find a large variety of educative software.

The best option to apply in the lecture was the highly recognized software MDsolids®, this software includes a wide variety of modules that explain all the topics with clear examples and easy to understand explanations.

Figure 3. Modules of software MDsolids®

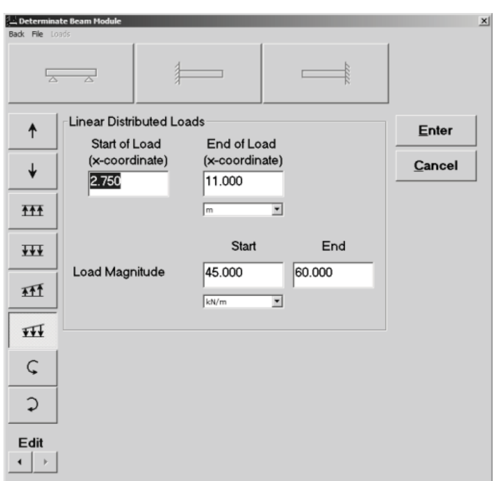


Figure 4. Beams module, loading a beam

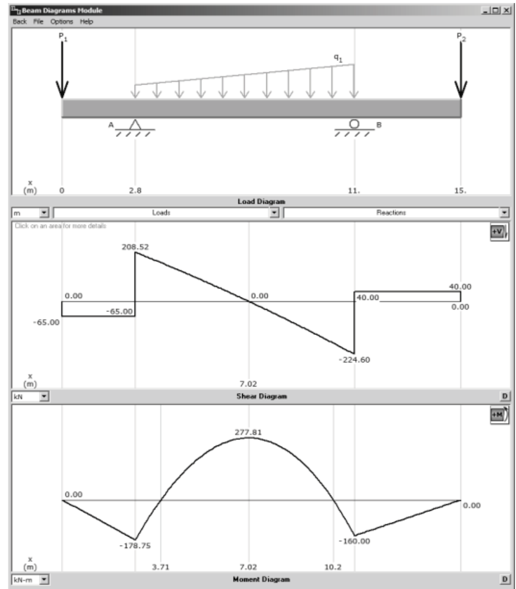


Figure 5. Shearing loads and bending moments acting in the beam

The academic staff of Mechanical Engineering Department evaluated the software; the professors developed several study cases with traditional procedures and then all of them were compared with the results that the software exposes. Then, a small group of students worked with the software in order to identify the facility to interact with the structure of the examples. After all this actions, the staff concluded that the software was a nice option to help the activity of the teacher into the classroom; then, the study of the lecture includes this software as a regular tool every semester and all the students are encouraged to install it as demo version or buying the book that includes the program.

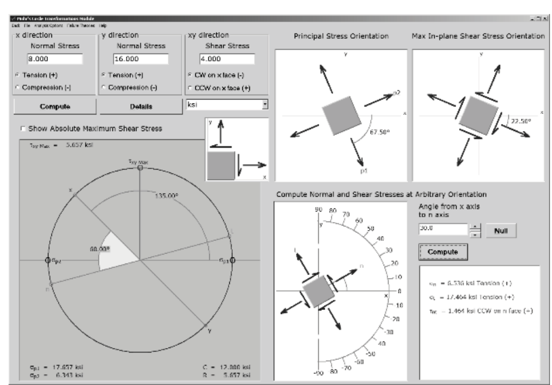


Figure 6. Mohr’s circle example

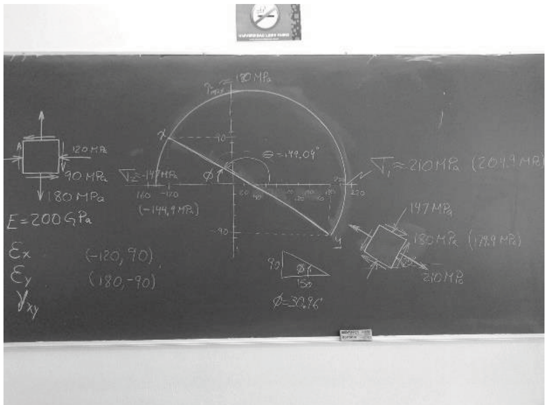


Figure 7. Solving Mohr circle in blackboard

2.2Validating the proposal

The structure of the study of mechanics of materials consists in five hours per week, with three hours of theory and two hours of practice, the common action of the teacher responsible of the lecture includes one demonstrative experiment and three sessions in computer every month. In this way, it is possible to complete a higher number of experiences for the students to understand the topics reviewed into the classroom.

After two years, working with the software the Department detected that a large number of the students continued failing the exams with a high number of cases of desertion given that the impossibility to get the credits of this lecture.

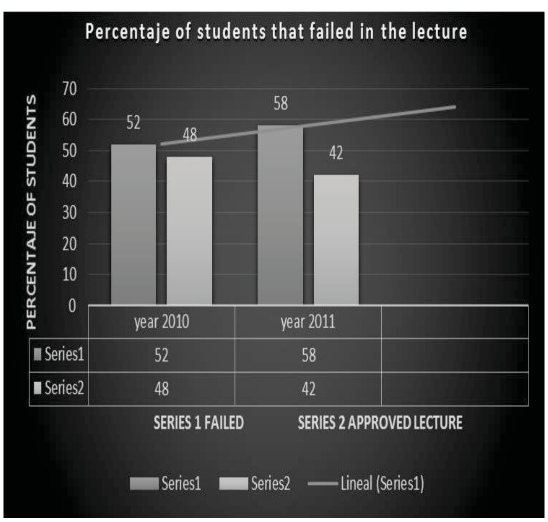


Figure 8. Behavior of the groups in Mechanics of Materials

These results forced again the deploy of a serial of actions intended to clarify other factors that could affect the performance of the students

The first action considered to evaluate the capacity of the software in order to produce long-term knowledge versus simple experiments applying didactic models some of them bought by the Institute others, developed and machined into the workshop.

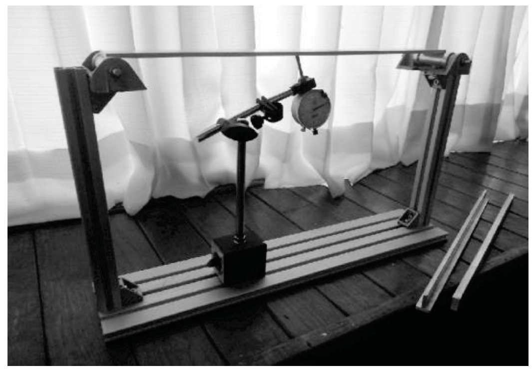


Figure 9. Simple beam system

Figure 10. Coplanar forces system

Both examples have an equivalent module in the software. Given that there are six groups of the same lecture the experiment consisted along 2014 and 2015 to separate the didactic resources under the next structure:

- Two Groups working with software (A)
- Two Groups working only with didactic models (B)
- Two Groups doing a mixture between both options (C)

After two years monitoring the results of the groups, the next graph shows the results of the groups.

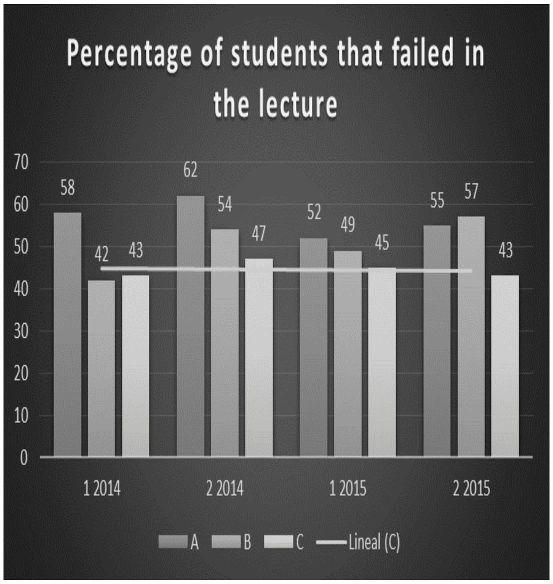


Figure 11.Percentaje of students with low notes by every model

The result, obtained from the teachers' final report show us that anyway, the percentage of engineering students that fail in the lecture is high. Nonetheless, there is a small difference in study group C; the students that worked with both simulator and with didactic resources developed a better performance that allow them to get the required competences.

It was possible to detect the next common factors for the group A:

1. High self-confidence about the topics reducing the number of hours of study, because "it is easy to learn with the software".
2. Low understanding of the structure and results provided by the software in order to identify if the component is capable to bear the stress
3. Low number of students in the group A reviewed the topics of the software with the theory of a book; most of them considered that the software is not wrong so, it is not necessary to read a book.

The detected conditions for the low performance for group B were:

- Given that, there is just one equipment in most of the times it was necessary to separate in small teams of no more than five students in order to assure the interaction with the didactic resource or machine. This condition meant less number of hours of practice.
- The total number of practices done per students is low in comparison with the number obtained for group A
- It is not possible to deploy a high variety of practices in every equipment given that the simple of the concept to do a laboratory practice

For all the groups the most relevant failures are:

- A lot of the students declared that it is better to take a picture from the exercise made for the teacher instead to copy the information received
- No one of them reviewed the picture to study or to remember the lesson.
- Most of them declared that it was not possible to install the software because no there is a mobile app.
- Close of 30% of the students said that they do not like to work with simulators; they prefer to work with real materials and equipment.
- Several times a student decides not to assist to the classroom cause all the time a colleague can send a picture of the exposed information.

3. Conclusions

With the obtained results it is possible to identify some of the difficulties that lots of students experiment in order to understand the behavior of an Engineering system with a simulator. In spite that most of them interact every day of technologies such as videogames, digital videos, chat on line, social networks and so on, most of the time the interaction is done with leisure purposes, instead of trying get relevant information about an academic topic. Study habits are changing with the use of technology and the impact of this change in the learning efficiency of Engineering students should be evaluated before continuing to migrate parts of the educative system to virtual environments.

It is interesting to point that a great quantity of engineering simulators include elements that tend to transform the experience into a game. This point of view is obvious if the goal is to produce a platform able to retain the consumer attention while avoiding boring sessions. However, we detected a problem with this approach, and it’s that the student does not review the topic with the importance that it deserves. It is possible to find a PLC software simulator that will include visually attractive graphics and examples as an elevator or a car parking system being controlled for a ladder program. It is a nice option to understand the logic of the instructions but it cannot really teach the inherent risks and serious injuries that a wrong program can produce in a real system.

In our study, the situation is very similar. Most of the students were not worried when the result of the simulator was different with the equivalent calculation. The reason is simple “it is not real”, and “I can recalculate again until the results are concordant”. Therefore, the suggested strategy is to continue including a mixture of resources; now, the target is to verify if software and didactic equipment are a better option to assist with the work in the classroom.

4 References

1. Mechanic Engineering Department. (2015) Rate of non-satisfactory results into a Mechanics of Materials course. Quality system report, 6 (25), pp 41
2. Stahl, G., Koschmann, T., Suthers D, (2006) Computer – supported collaborative learning: An historical perspective. Cambridge handbook of the learning sciences, pp 409 – 426
3. Hay D., Kinchin, (2008) Making learning visible: the role of concept mapping in higher education, Studies in higher education, 33 (3), pp. 295 – 311

A Mathematics E-book Application by Maple Animations

A. Alharbi[1*], F. Tcheir[1], and M. Siddique[2]
[1]Mathematics Department, King Saud University, Riyadh, Saudi Arabia
[2]Mathematics Faculty, ECPI University, USA

Abstract **-** *The role of technology and using specialized software in the educational process is growing in recent times. Many resources are available both commercial and academic, targeting a wide variety of audiences. In this project we developed an e-book application that complements the usual topics covered in calculus, and discrete mathematics. The e-book application is developed using animations designed by Maple as an interactive book viewer. It allows the user to read content and play animated illustrations which aids in understanding the material. The animated graphics are simple accessible and can be customized by a user with little programming background, they also run without the need to install the Maple software on the computer or to be connected to the internet.*

Keywords: Maple Animations, E-learning, E-book, Graphics, Visualizations.

1 Introduction

Mathematics is the foundation of many core concepts in science and engineering. A successful career in science and engineering is impossible without a solid mathematical base. However, mathematics is always considered a hard subject because many mathematical concepts in calculus, multivariate calculus, trigonometry, and discrete mathematics are difficult to explain to students on blackboard or on paper. Without a thorough understanding students are bound to lose their interest in this very important subject. Even if the students get some understanding, it is extremely difficult to visualize many graphs and curves of various equations. The goal of our education system is to keep our students motivated, enhance their learning capabilities, and develop the ability to reason precisely and think logically. Now the advent of powerful convenient software makes it imperative to reconsider how to teach these subjects. Colored graphics makes it possible to display curves, surfaces, and solids in two and three dimensions in a way that is both more effective and more engaging for the student. Fortunately, today's software allows us to bridge the gap between understanding mathematics and visualizing its basic principles. Students will see and implement some basic animated graphics in their studies and instructors can rely on such graphics to explain these subjects.

Maple software is one of the famous systems for symbolic and numerical mathematics. Animations support the learning outcomes in a significant way: they enable students to repeat a great number of necessary steps, and enrich their visualizations, which cannot be done effectively using only the pencil-and-paper method. Maple gives a lot of benefits for mathematics education. It creates an engaging lesson, an interactive classroom environment, which increases student comprehension, minimizes preparation time and facilitates visualization of abstract concepts. We have chosen Maplesoft to design our animations because of the simplicity of Maple and its powerful graphical capabilities. There are many other software's available in the market such as Matlab [12], Mathematica [13], Scientific Workplace [14], that have graphical, computational and programming capabilities, but Maple offers the best animation capabilities with simple commands. There are also a few mathematical companion websites provided by some universities, such as "Maple for Math Majors" from University of Perdue [10], which presents some examples and exercises to help their students in the calculus courses offered. There is also another website "Computational Mathematics" from University of Waterloo [8], which offers tutorials for students on some calculus concepts. There are also papers published on using Maple in Graph theory: "Simple Graphs by Maple" [11], and several published work on applying Maple animations and graphs in Calculus [1],[2],[4], specifically showing computations of differentiation , integration and function properties [5],[6],[7].

The e-book application we have developed is to help understand mathematics using Maple animations in undergraduate college levels through six main chapters. Currently we have completed the first four chapters only, and the remaining chapters will be designed in the next year 2017. The e-book developed is called *"Maple animations to understand Mathematics*". It is an interactive book viewer allowing the instructor to navigate through every chapter by selecting from the Table of Content viewer. The user can cycle through the loaded animated files and play them as seen in Figure 1. The animated graphics are simple accessible and need little programming background, they also run without the need to install the Maple software on the computers, or to be connected to the Internet. All the codes used to generate the animation graphs are provided, allowing the user to change the functions and the intervals to customize his own animations and graphics.

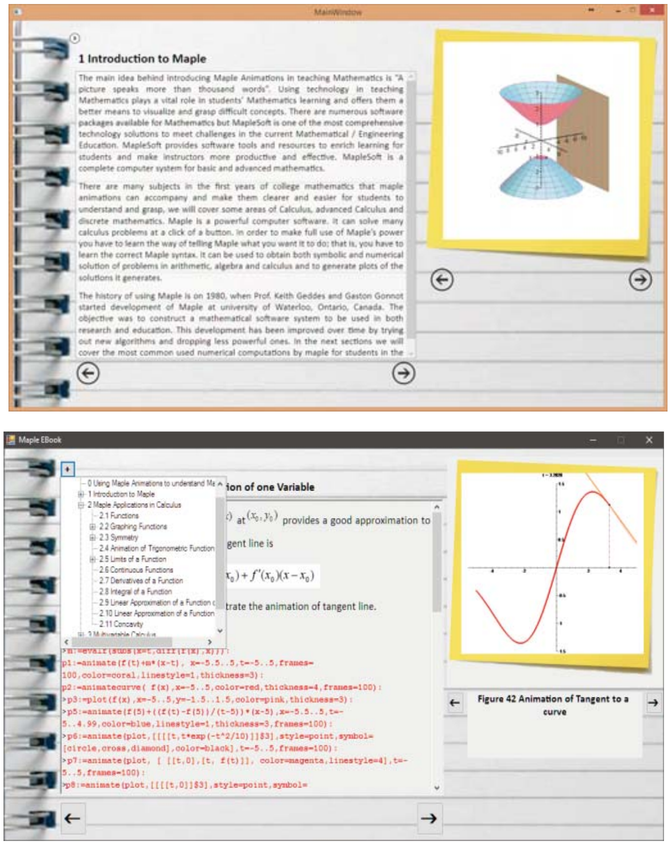


Figure 1: Design of *"Maple animations to understand Mathematic "* E-book application viewer.

2 Maplesoft

System Maple has been developing during the past 30 years together at several universities, the largest share of the work was done under the combined group of scientists called "Symbolic Computation Group" at the University of Waterloo in Canada and then in federal technical university ETH Zurich in Switzerland, where part of this group passed in 1990 [8]. Maple was commercialized and its further development is governed by a Canadian company Maplesoft Inc. based in Waterloo and currently Maplesoft^{TM} is a subsidiary of Cybernet Systems Co. Ltd. in Japan, the leading provider of high-performance software tools for engineering, science, and mathematics [9]. Maple allows you to perform both symbolic and numerical calculations, show graphs of functions of one and two variables. We can involve all computations by some text and create hyperlinks [15]. Maple worksheet created in this way saves files in its special format MW, which is stored in XML format. Files can be also exported to LaTeX, HTML, RTF and MathML, which is an extension of HTML for the presentation of mathematical texts on the web. Maple also allows automatic conversion of their commands and procedures in the programming languages C, Fortran77 and Java [3]. Maple is based on a programming language similar to Pascal fourth generation with many predefined functions and procedures. Maple involves many functions covering various parts of mathematics such as differential and integral calculus, linear algebra, algebraic equations, differential and difference equations, geometry, graph theory and logic. Visualization from the point of view tools of IT is understood as a set of tools for visual analysis. The basic statement for creating of two-dimensional graphs is named **plot**. Similarly, Maple uses statement **plot3d** to display the graph of a function of two independent variables. In Maple, we can not only draw graphs, but also we can perform animation, which arise from a sequence of images, depending on some animated variables. This command is named **animate** and it is part of package (plots).

3 E-Book Content

Our E-book currently contains four chapters covering the most important mathematical topics in Calculus and discrete mathematics, as follows:

- The first chapter introduces Maple software and its uses to familiarize the user to the Maple software in performing the basic mathematical computations. Simple solve commands: fsolve, isolve, Polynomials: Sorting and Collecting, Root Finding and Factorization. Graphics: 2D, 3D, Parametric Plots. Animation: 2D,3D.
- The second chapter includes concepts on Calculus specifically: limits, functions, derivatives, integration and graphing functions to illustrate their properties. Also animation of trigonometric function, limits of a function, definition of derivatives, Riemann integral, tangents and critical points are included.
- The third chapter covers some applications of calculus in several variables and algorithms to plot a graph in a three-dimensions together with some popular shapes and surfaces such as cones, parabolas, spheres, Hyperbolas, Hyperbolic Paraboloid and many more.
- The fourth chapter presents some concepts of graph theory that are important in discrete mathematics, displaying different types of graphs such as planar, Connected, Hamiltonian, Euler, isomorphic Bipartite, Matching, Planar and coloring graphs.
- Each chapter contains some specialized references for extra readings. Finally, an index is included to facilitate the use of our book and to easily find the subjects needed.

3.1 The E-book Design Method

The Maple E-Book application is an interactive book viewer. It allows the user to read contents and play animated illustrations. Chapters and materials are stored as Rich Text Format files (RTF) in a specific folder using a specific naming system. Upon application startup, it loads all the materials from the folder and populates a tree which represents the Table of Contents. Animated illustrations are Graphics Interchange Format files (GIF) named in a certain format which corresponds to the according chapter or sub-chapter. These files are loaded with each selected chapter. The user can navigate through every chapter by selecting from the Table of Contents viewer. The user can cycle through the loaded animated files and play them. The animation is set to loop to give the impression of a continuous play which aids the user in understanding the selected topic. Material can be easily added by adding to the material folder. New added material will be loaded on application restart.

The application can be loaded on any Windows PC or Tablet. It is best suited for tablets for the look and layout of the application that replicates a student's notebook. The application doesn't need an internet connection to operate. All materials are loaded on the device. The application can be easily modified to display any material. No specific software such as Maple needed to be installed. The application can be used as an interactive book. It can be loaded with any topic with animation files. So it can be used for other book materials.

The application is written in .NET (4.5) environment using C# language. It is compatible with Windows 7 and later versions. The process of designing the e-book went through the following steps:

1- The mathematical concepts were chosen from calculus and Discrete mathematics.

2- The exercises and graphs for these concepts were then designed.

3- Maple codes were written which plot the graphs.

4- Maple Animate was used to make the graphs animated graphs.

5- The graphs were modified in color, camera view angle coordinates, movement, and rotation angle, then saved as Graphical interchange format (GIF).

Example 1: Animation of formation of a Sphere is conducted through the following steps:

```
> f:=x->sqrt(1-x^2):a:=-1: b:=1:
> p1:=spacecurve([0,0,z],x=-1..1,y=-1..1,z=-1..1,thickness=3):
> Surface:=[u,f(u)*cos(v),f(u)*sin(v)]:
  Range:=u=a..b,v=0..2*Pi:
> p2:=animate3d({[f(u)*cos(v*t),-
  f(u)*sin(v*t),u]},Range,t=0..1,labels=[X,Y,Z],
  tickmarks=[3,3,3],frames=100,numpoints=2000,style=line):
> display(p1,p2);
```

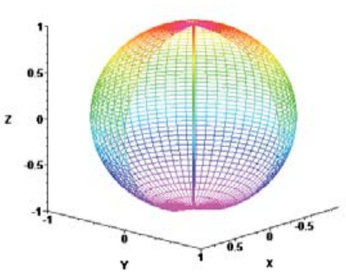


Figure 2: animation of formation of a sphere

Example 2: Vertical traces of hyperboloid of one sheet are hyperbolas. The following maple animation demonstrates this fact.

```
> f:=x->3*sqrt(1/2+x^2): a:=-3: b:=3:
> p1:=[f(u)*cos(v),f(u)*sin(v),u]:
P2:=plot3d(p1,u=-3..3,v=0..2*Pi,labels =[X,Y,Z]):
>p3:=animate3d([k,y,z],y=-9..9,z=-3..4,k=-
3..3,frames=50,labels=[X,Y,Z],color=gray):
> display(p2,p3);
```

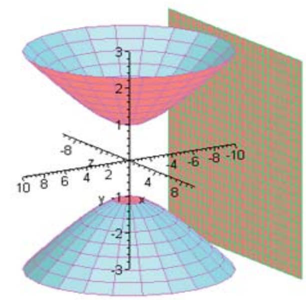


Figure 3: Vertical Traces of hyperboloid of one sheet are hyperbolas.

4 Survey Results

A group of students were chosen from several mathematics departments studying in different levels and different courses such as: Numerical analysis, Calculus I and II, and Discrete mathematics, with a total of 175 students. Also a group 50 of math instructors from different colleges were also approached to give their opinion of the product. Moreover, 40 Masters and PhD students from different science and engineering majors also volunteered to try our e-book. They were all given a copy of the e-book to evaluate and use for a few days in their studies A survey was later sent to them to record their opinion of the e-book and to give their suggestions and recommendations.

Figure 4 shows the percentage responds to the survey questions. The results show 97% of the users enjoyed the e-book and were satisfied with its benefit. Most of the suggestions were to have the product available in the central library for other students, also there were suggestions to add material on, solving differential equations, and solving systems of linear equations. All these suggestions will be considered when developing the remaining chapters. There is also plan to make the e-book available as an application for Tablets both Android and I-pad and therefore making it available to address a broader audience.

Dear Student\ instructor

Thank you for your support in evaluating our E-book *"Maple animations to understand Mathematics".*

	Questions	Strongly agree	Agree	Agree to some extent	Disagree	Strongly disagree
1	I am satisfied with the E-book application.	97	3			
2	Instructions are easy to understand and use.	86	10	4		
3	I can easily reach the chapters and the animations.	90	5	5		
4	I find the animation graphs interesting.	95	5			
5	The material covered by the book is relevant to my studies/work.	85	10	5		
6	The animations helped me understand the mathematical concepts.	90	7	3		
7	I would recommend this E-book to my follow students/colleagues.	90	10			
8	I am looking forward to use this E-book as a reference in my studies.	90	6	4		
9	I would like to buy this product to help me on my studies/work.	92	5	3		

What do you suggest to change in this E-book?

____

What would you like to add to this E-book?

____

Figure 4: Results of the Survey of *"Maple animations to understand Mathematic "* E-book.

5 Conclusions

The use of animations as part of the lectures has brought to the classroom the dynamics of three dimensional objects to create a more active way of teaching mathematics. Animations bring the mathematical concepts alive and with interactive animations students can manipulate the animations to understand mathematical concepts. Incorporating computer animations together with traditional hand computations has helped students reach the desired goals more successfully. From our experience with our E-book we can notice the need to use technical programs such as Maple to facilitate mathematical concepts to student and expand their understanding and knowledge for all disciplines especially to those who have difficulties in grasping mathematical concepts.

The product should be developed in the future to include other mathematical concepts in the remaining two chapters. Workshops should be conducted to instructors to train them on Maple animations usages to enhance class room teaching. Instructors should encourage the use of Maple and other computer software's in their courses through tutorials, homework, and computer assignments. We realize that our younger generation is technology smart and learn fast through different mediums and technologies, so we need to in cooperate e-learning methodologies and introduce technological methods and software in teaching mathematics to help them successfully reach their goals.

Acknowledgment

This research project was supported by a grant from the "Research Centre of the Female Scientific and Medical Colleges", Deanship of Scientific Research, King Saud University.

6 References

[1] Bauldry, W.C, and J.R Fielder."Calculus Laboratories with Maple", Brooks/Cole Publishing Co, 2008.

[2] Bill Blyth and Amin Naim,"Problem Solving using Maple Animations", RMIT University, Australia,2012.

[3] Blyth B., "Visualization of Slicing Diagrams using Maple", EMAC Proceedings, University of Technology, Sydney, pp. 7-12, ANZIAM, 2003.

[4] Blyth B., "Multiple Integration - Visualization and Animation using Maple", Proceedings of the 17th International Conference on Technology in College Mathematics: ICTCM New Orleans, USA, 2004.

[5] Devitt S," Calculus with Maple", Brooks/Cole Publishing Co, 1993.

[6] Ellis W., Lodi E.," Maple for the Calculus Student: A Tutorial", Brooks/Cole Publishing Co.,1993.

[7] Pražák Pavel, (cz), Gavalcová Tatiana," Animation of essential Calculus concepts in Maple", Aplimat, 2012 .

[8] Website: "Computational Mathematics." , University of Waterloo, 200., (http://www.math.uwaterloo.ca/navigation/CompMath/).

[9] Website : E-book," Maplesoft", (http://www.maplesoft.com/products/maple/)

[10] Website : Roger Kraft," Maple for Math Majors", http://math.purduecal.edu/~rlkraft/mfmm/index.html

[11] Website: Carlo F.," Simple graphics with maple. Barenghi", http://www.mas.ncl.ac.uk/~ncfb/maple.pdf

[12] Website: Matlab http://www.mathworks.com/academia/?s_tid=gn_acad

[13] Website: Mathematica ,https://www.wolfram.com/mathematica/

[14] Website: Scientific workplace, www.mackichan.com/products/swp.html.

[15] Website: E-book," Maplesoft, Calculus study guide", (http://www.maplesoft.com/products/studyguides/CalculusStudyGuide/index.aspx).

SESSION

MANAGEMENT AND ORGANIZATIONAL METHODS & RELEVANT TOOLS + ASSESSMENT AND ADVISING STRATEGIES

Chair(s)

TBA

Using Beacons for Attendance Tracking

D. Deugo[1]
[1]The School of Computer Science, Carleton University, Ottawa, Ontario, Canada

Abstract – *Students experiencing difficulty with their studies don't often identify themselves as students in need of support. The Science Student Success Centre at Carleton University actively seeks out these science students to give them the help they require. One indicator of need is a student's grades. Another indicator is attendance in class. In this paper, we describe how we use beacon technology to help track attendance. While our existing online card swiping and QR Code scanning attendance tracking system was suitable for smaller class sizes, our beacon-based attendance tracking system is now suitable for classes and events with over 1000 attendees.*

Keywords: Attendance, Tracking, Android, Beacon

1 Introduction

In [1, 2, 3], we described different variants of our attendance tracking system and its usage in classes to examine its performance and usability. Our motivation for building the system came from The Science Student Success Centre (SSSC) at Carleton University. The SSSC collaborates with other on-campus departments to help develop a student's academic and professional skills. The SSSC also takes an active role in helping science students struggling in computer science, math, and science courses with the following mission:

- Increase the engagement and retention of students in the Faculty of Science
- Foster the growth and achievements of high performing students
- Identify and support students who may be experiencing difficulty with their studies
- Inform students of professional and academic development opportunities
- Aid in the recruitment of outstanding students to the Faculty of Science

It was our belief that identifying students with less than 60% attendance records and having the SSSC talk with and mentor these students would help with their overall success in the faculty of Science. We had already been successful mentoring students with grades less than 60% by discussing with them the issues they face with their courses and suggesting possible techniques and actions they can take to overcome the issues.

Taking attendance using paper and pen was one approach we could have used, but we knew it was slow and prone to errors. In [2, 3] we developed a student card scanning based system for tracking attendance in classes containing less than 50 students. However, we needed to expand that capacity. In [1] we added the ability for students to generated custom QR codes that were scanned by users or instructors using a custom application that ran on a smartphone or tablet. The scanned information was sent back to our system's server and a student's attendance at a class or event was recorded. The approached doubled our ability to check students into a class within a required timeframe. However, we needed to expand that capacity by at least an order of magnitude to handle the potential for very large classes or event attendance tracking.

1.1 Goals

Reiterating our goals and objectives from [3]. Our main goal is to provide a fast and efficient attendance tracking system. In addition, the system must work in any and all classrooms at Carleton, including its electronic classrooms – those with computers and projectors – and those classrooms containing no computers. A further goal is to provide a system that requires minimum hardware, and can be maintained at minimum cost. Our new goal is to be able to track attendance of classes or events with over 1000 students or participants.

1.2 Objectives

To meet the goals, we have the following objectives:

- Use easily found, inexpensive hardware for the system.
- Make use of mobile devices to help with the attendance tracking process but not specifically require the use of student owned mobile devices
- Use open source software to minimize development and maintenance costs
- Installation should be fast and simple
- Attendance reports must provide end users with the greatest flexibility for manipulating the collected attendance data
- Support both Mac and PC platforms, and Android and iOS mobile devices

1.3 Outline

In section 2, we survey beacon technology and beacon-based attendance tracking systems. Section 3 describes our system and how it uses beacons in classes to take attendance. Section 4 describes our results. Section 5 provides our conclusion and reviews our system's feature set.

2 Background

There are numerous beacon vendors providing many different styles of beacons for different needs. A 2015 report by Aislabs [5] examined 26 different vendors, including the beacons we use from Radius Networks [6]. Beacons are generally very small devices that are often powered by a battery such as the 3-volt CR2032 battery. Some beacons, like the ones from Radius Networks, can be powered from a wall outlet or a USB outlet. Beacons emit a Bluetooth Low Energy (BLE) signal that can be detected by a BLE enabled mobile device. Beacons have a transmit power and advertising interval. The former dictates how beacon's signal can travel (up to 50 meters) and the latter indicates how often the signal is sent, e.g. 100 to 500 milliseconds. Both the power and advertising setting impact the battery life of a beacon. Depending on one's settings, and a beacon's battery, the battery life can be as short as 0.6 of a month to upwards of 56.3 months.

Each hardware beacon can support one or more of the following beacon specifications:

- iBeacon [7]
- AltBeacon [8]
- EddyStone once called UriBeacon [9]

[10] provides an overview of the first two specifications. In summary the iBeacon and AltBeacon specifications are similar. In short both support beacons transmitting a universally unique identifier (UUID) and a few bytes of configurable data specific to the device that is static once configured. Eddystone beacons are configured with arbitrary blobs of data that are served back as messages to Android and iOS apps. One can update attachments remotely. One of the important features of the Eddystone specification is the following:

> *Eddystone is an open beacon format from Google that works with Android and iOS. Eddystone includes a number of broadcast frame types, suitable for different types of deployment. [11]*

As hardware and specifications stabilize, so too do the number of uses for beacon technology. [12] describes over 100 different use cases for examples of iBeacon technology. Retail, Hospitality, Tourism, Education, Healthcare, Entertainment, and the Travel industries are all prime areas for use of beacons. In addition, the Corporate, Automotive, Real Estate and Adverting industries are areas that can benefit from the technology. In this paper, we are interest in using beacons to help track attendance at events or classes.

Several different systems are using beacons for event tracking. Špica International, a company based out of Slovenia, has created an application named All Hours [4] that enables employees to clock in at work with their smartphones. Within a range of about 20 meters, employees use their smartphones as identification once an Estimote Bluetooth beacon is detected by their smartphone application. The application then registers an employee as in the premises. Their backend system provides various reports on when employees arrive and leave the premises and their system can also enforce a company's working time policies by permitting a company to set its work hour rules.

SessionRader is a U.S. company that offers secure and verified local beacon check-ins at any location. The benefits they suggest [13] of their system are the following:

- No Badge Required
- Passive + Interactive
- No Scanning or Lines
- Integrates with SmartWatch
- Instant Automatic Attendance
- Wireless Check In

In their system, attendees are either running the associate iOS mobile application and are informed when they are in range of a beacon allowing them to check in, or the detection of the beacon is integrated with the device's notification support, alerting an attendee of the possibility of checking in. A backend system with application support enables users to view attendance data in real time.

DoubleDutch, a company with offices in the U.S. Netherlands, Hong Kong, and the U.K., use beacon technology to support capturing who is attending a session using their Head Count application. Their application runs on both iOS and Android devices that support Bluetooth 4.0 or greater. Provided an iBeacon-enabled device is available for each session, attendees can conform their check in when in range of the beacon.

In contrast to using beacons, [14] uses a combination of QR codes that each attendee have, typically on their conference badge, and conference support staff having iOS and Android devices running their custom application to scan and track attendees. Their system provides real time reports and registration details to registered users. The disadvantage of this system is that it focuses in the conference domain, where the number of events and attendees are usually small. A mid-size university can have hundreds of classes and over 25,000 students to track on a daily bases. The sizes of the education domain make typical costing models using this approach with attendees and sessions expensive.

The above approaches suggest several important features for any attendance tracking system:

- Real time reports
- Fast Scanning
- Attendee identification
- Mobile device application
- Attendee id and registration

All of these features are outlined in our approach, described in the next section.

3 Approach

As indicated in [1, 2, 3], our system users are instructors or people that are taking attendance in classes or at different events. They are provided with a user id, which is their email address, and a unique password. After logging into the system using the Event Tracker website, shown in Figure 1, users can download their attendance real time reports for any class or event.

In [2, 3], users attached a USB card reader to users computers and swiped student cards in order to track attendance. We also had a mobile application that students could use on their devices to identify themselves as attending the class or event. Their location was taken from their current GPS location of the device. In [1] we added the ability for students to generated custom QR codes that were scanned by users or instructors using a custom application that ran on a smartphone or tablet. The scanned information was sent back to our system's server and a person's attendance at a class or event was record.

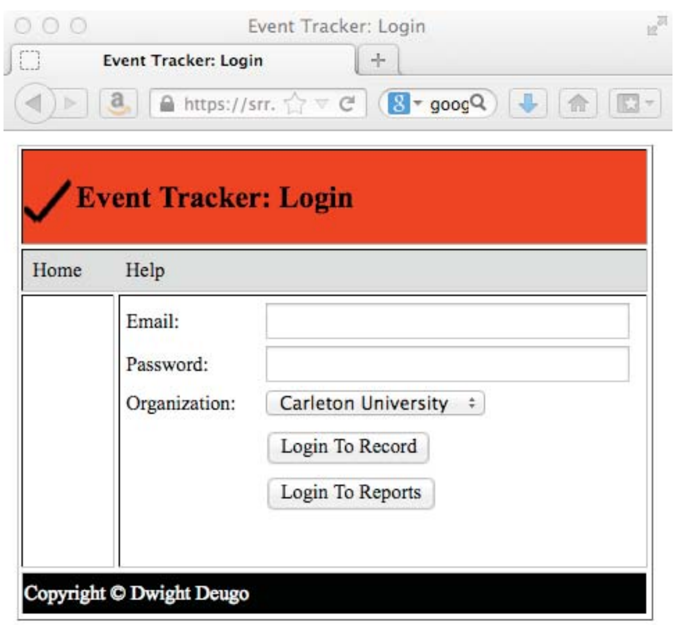


Figure 1: Login Screen

In the current version of our system, the above capabilities are still supported. However, now we have added beacon support with a custom attendee mobile application that detects beacons in order to permit attendee logging. We use beacons for two reasons. The first reason is that by having each attendee use the associated mobile application it allows us to process attendee logging in the thousands. Previously the scanner and QR Code approaches only permitted logging of up to 100 attendees in any short period of time. The second reason is that the previous mobile application with GPS support proved highly inaccurate, permitting attendees to log their attendance when they were not in the room or building. In the following subsections we will examine our approach to these problems in more detail.

3.1 Beacons

Our system use beacons from Radius Networks [6]. Available from the Radius Network Store at $21.00 U.S. for one,

> *"RadBecon USB is a fully standalone Bluetooth Smart proximity beacon using iBeacon, AltBeacon and Eddystone™ technology, implemented in a tiny USB package."*

The RadBecon provides proximity services for iOS, Android and other mobile environments and is powered by a standard USB port. This means that these beacons can be plugged into a USB AC adaptor, car adaptor, or a computer with a USB port.

3.2 Attendance Setup

Tracking attendance using our system involves the following components: User, Event, Beacon, and mobile application.

The system's administrator assigns a user's email address, password, organization, and the events or classes they are able to track. The system administrator has access to the backend of the system and is outside the scope of this paper. After entering their email address and password into our web interface, a user can look at their attendance reports for their associated events. Reports are in the form of an Excel spreadsheet. Each entry in the Excel spreadsheet provides information on an individual checking in or out of an event. This information includes the following:

- First Name
- Last Name
- ID Number
- Email Address
- Event Name
- Organization
- Time
- Direction (In/Out)
- Latitude
- Longitude

- Altitude
- Accuracy
- Device

The Device entry can be Android, iOS or web, identifying the three methods of data entry available for our system. This paper discusses the Android approach using our Android mobile application. In the case of Android entry, the accuracy is always 1 and the latitude, longitude and altitude are 0.

To track attendance, a user must ensure that the beacon that is registered to their event is available. Beacons are configured with three values: UUID, Major, and Minor. A UUID contains 32 hexadecimal digits, split into 5 groups, separated by dashes. Major and Minor values are numbers assigned to the beacon, in order to identify them with greater accuracy than using the UUID alone. They are unsigned integer values between 1 and 65535. This configuration is done by the system administrator and outside the scope of this paper. Once configured, a user brings the beacon to the corresponding event and plugs it in, or the beacon is permanently installed and powered.

The final step in the setup is that each attendee must install our EventTracker application on their Android device. The only device requirement for the application is that it supports BLE, also called Bluetooth Smart or Version 4.0. BLE is designed to provide significantly lower power consumption and is the Bluetooth specification that our beacons use.

3.3 Mobile Application

We had the goal for our EventTracker mobile application that is was easy to use. To meet that goal we wanted to minimize the number of touches an attendee makes in order to indicate their attendance at an event.

After downloading and starting up our application, the attendee, and from their perspective a user, is presented with the view shown in Figure 2. It is in this view the attendees enters the information that they wish recorded when their attendance is taken, namely their first and last names, their email address, id and language preference. It is worth noting that an attendee does not indicate the events they are going to attend. Event identification is done on a discovery bases. After entering the information the Save Info button must be clicked in order to commit the information to the device. If not done, the attendee will see this screen and need to fill in their information each time the application starts up. However, at anytime the attendee can always come back to this screen by selecting the User tab to change and then save the information again.

By selecting the menu button (three dots in the top right corner of the view) a user can configure the backend server IP address the application will use. For most attendees, they will never need to change this value. However, there are scenarios where this value will need to change dynamically and we have configured this into our application. It is just a simple matter of selecting the menu, entering the new address and selecting the OK button. Figure 3 shows the corresponding IP Address configuration dialog.

Once the attendee information has been entered and the server IP address has been configure, an attendee will see the screen shown in Figure 4 when the application starts up. This is the screen where attendees initiate the scanning for events that they can register their attendance. The first step for an attendee is to click the Scan For Events button. This initiates a search by the application for any beacons that are within range of the device. As a result of the scan, the application will gather each detected beacon's UUID and its Major and Minor values. These values are then sent to the server to gather information about the events that are associated with the values. This information is returned to the application and presented to the attendee. Figure 5 shows an example of the information that is presented to an attendee. In this example, three different events were indicated from the detected beacon(s).

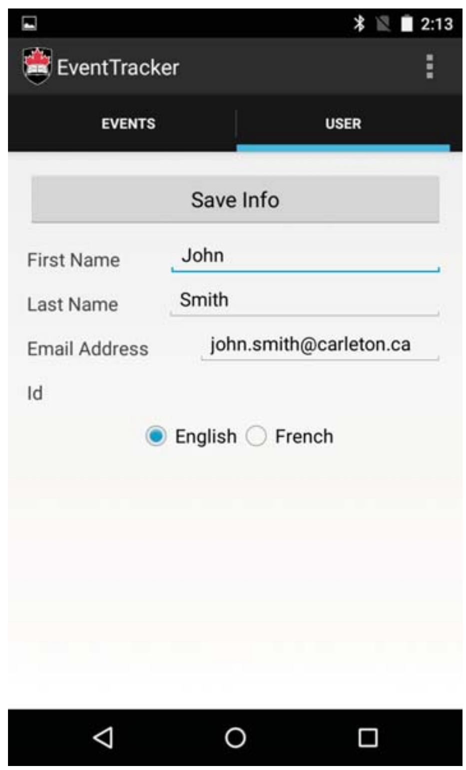


Figure 2: Attendee Information

The last step in the process for an attendee is to select which event they are attending and then click the Attend Event button. Once done, the application sends the attendee's information back to the server to indicate the attendee's presence at the corresponding event. Once complete an attendee see the status line indicating “Success!”, as shown in Figure 5.

3.4 Report Generation

After logging in to the Event Tracker web front end, as shown in Figure 1, an end user can generates up-to-the-minute Excel reports of the events they are tracking. By selecting Reports in the left hand menu or selecting Login to Reports in the home screen a user is taken to the Reports Screen shown in Figure 6. All of a user's events are listed in this screen and by selecting one and clicking the download button, an Excel spreadsheet of results is downloaded for the user's review, as shown in Figure 7.

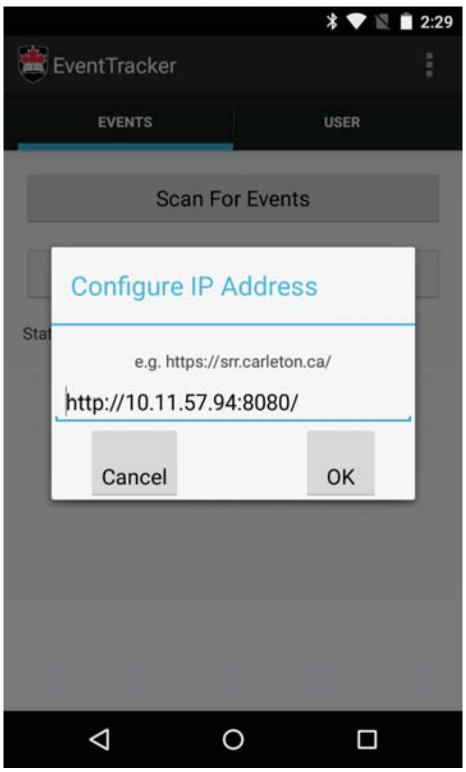


Figure 3: Configuration

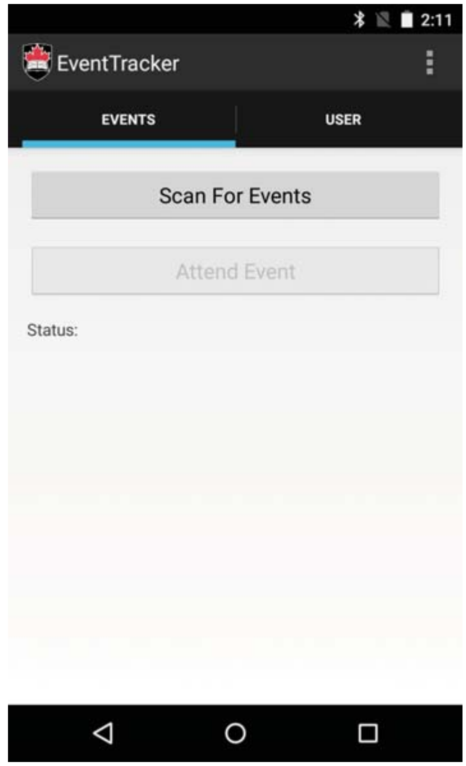


Figure 4: Scanning

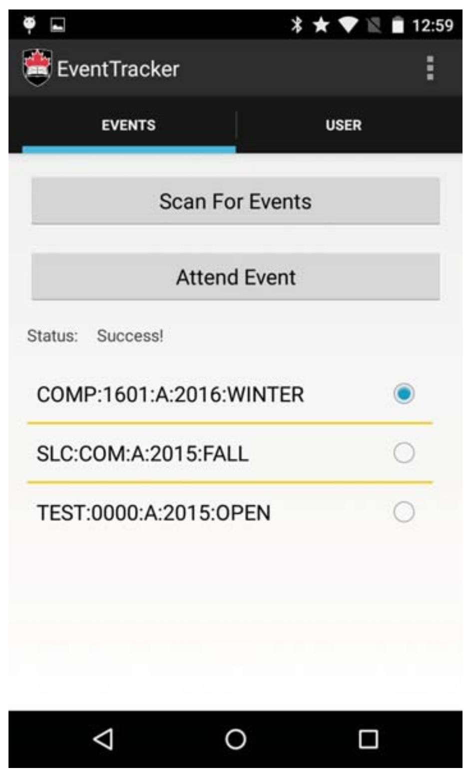


Figure 5: Attending

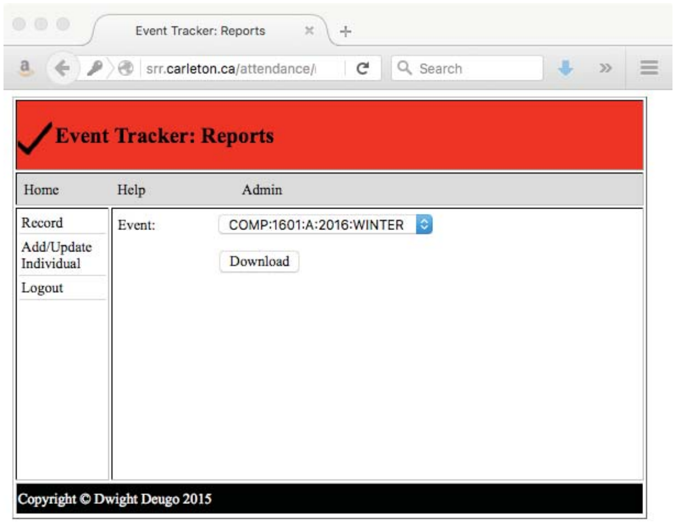


Figure 6: Reporting

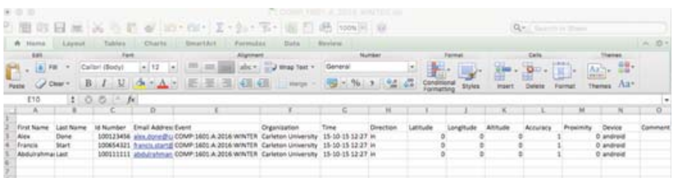


Figure 7: Excel Spreadsheet

4 Results

In [3] our system used the swipe of a student card to indicate a student's attendance in class. The issue we faced with this version of the system was that it required us to register each student once. As reported, it took an average of 50.34 seconds to register a student. In small class sizes, this one-off operation was acceptable, but in classes of over 50 students, this time requirement in the initial class(es) was too high. The other issues that we faced was that the time required to swipe in students for a class took anywhere between 8 and 16 minutes for class sizes of 50, either before or after class. Given the swipe timings found, we decided that the use of student cards just didn't work in classes of 50 or more.

In [1] we added the ability for students to generated custom QR codes that are scanned by users or instructors using a custom application that ran on a smartphone or tablet. The scanned information was then sent back to our system's server and a student's attendance at a class or event was record. One improvement with this approach was that students only needed to generate their custom QR code once, and this activity could either be done offline or in parallel with other students, using our QR code generation website. A second improvement with this approach was that it took an average 5.3 seconds to scan an individual's QR code. The implications was that in the first ten minutes before an event we could scan a little over 100 attendees, doubling the size of event or class our application could support over [3].

Our goal was to be able to handle events with over 1000 attendees and with our current approach and custom mobile application we have been able to reach that goal. This was achieved by distributing the workload. Now every attendee has a device that can record attendance, rather than having to go through a single device. Moreover, the number of attendance recording actions permitted is only limited by the number of simultaneous incoming connections our server can support. Our server is based on Tomcat [15]. As part of Tomcat's default configuration,

> *The default maximum queue length for incoming connection requests when all possible request processing threads are in use is 100.*

This number is only the default. As reported in [16], it is possible to have over 13000 concurrent connections in Tomcat by manipulating its Connector configuration and the Java heap size Tomcat is running on. This number, combined with the fact that the maximum time to execute a single connection request is 40 ms, means that we can support attendees in the 1000s, which meets our goal.

5 Conclusion

We have created a mobile application and server-based system that be used for tracking attendance for events or classes with well over 1000 people attending. Moreover, if attendees attend different events, they do not need register with subsequent events, as their personal information is only recorded once in their mobile application.

Another issue we have resolved is a technique to ensure that an attendee is actually at an event when being tracked. Having the mobile application detect beacons at the event and then only allowing an attendee to register for only the events associated with the beacons enables this feature.

Our working system has the following features. Ultimately, attendance tracking data is stored on our server. We support the creation of a user with corresponding events they wish to track attendance. Users can login to our system with their user id and password in order to retrieve real time Excel reports of who has been tracked at one of their events. We have a custom mobile device application that permits attendees to enter their personal information, detect beacons and register attendance for events associated with those beacons. We use beacons that support standard protocols requiring only to be plugged in for power.

We believe we have a system that is ready for product development and will be moving in that direction in the near future.

6 References

[1] Deugo D. "Using QR-Codes for Attendance Tracking". The 11th International Conference on Frontiers in Education:

Computer Science and Computer Engineering (FECS 2015), 267-273. 2015.

[2] Deugo, D. Tracking Attendance Pilot. The 10th International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS 2014), 101-107. 2014.

[3] Deugo D. "Attendance Tracking". The 9th International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS 2013), 79-85. 2013.

[4] All Hours - Employee attendance tracking with Beacons. Web 14 March. 2016. <http://allhours.com>

[5] The Hitchhikers Guide to iBeacon Hardware: A Comprehensive Report by Aislelabs. Web 14 March. 2016. <http://www.aislelabs.com/reports/beacon-guide>

[6] We Make Proximity Work; Radius Networks. Web 14 March. 2016. <http://www.radiusnetworks.com>

[7] iBeacon for Developers. Web 14 March. 2016. <https://developer.apple.com/ibeacon>

[8] AltBeacon: The Open and Interoperable Proximity Beacon Specification. Web 14 March. 2016. <http://altbeacon.org>

[9] Beacons: Google Developers. Web 14 March. 2016. <https://developers.google.com/beacons/?hl=en>

[10] Understanding the different types of BLE Beacons. Web 14 March. 2016. <https://developer.mbed.org/blog/entry/BLE-Beacons-URIBeacon-AltBeacons-iBeacon>

[11] Platform Overview: Beacons, Google Developers. Web 14 March. 2016. <https://developers.google.com/beacons/overview>

[12] Web 14 March. 2016. <http://blog.mowowstudios.com/2015/02/100-use-cases-examples-ibeacon-technology>

[13] Over 100 use cases and examples for iBeacon technology. Web 14 March. 2016. <http://sessionradar.com/attendance-tracking>

[14] Attendance & Session Tracking. Web 14 March. 2016. <http://www.eventrebels.com/products/er-mobile-onsite>.

[15] Apache Tomcat. Web 14 March. 2016. <https://tomcat.apache.org>

[16] Cool, Tomcat is able to handle more than 13,000 concurrent connections, <http://blog.krecan.net/2010/05/02/cool-tomcat-is-able-to-handle-more-than-13000-concurrent-connections>

Incorporating a Modified Group Advising Approach into a Computer Science Program

Ron Davis
Department of Electrical & Computer Engineering
Tennessee State University
Nashville, Tennessee 37209
rdavis8@my.tnstate.edu

Xihui Zhang, Jill Simpson, Carol Gossett
Department of Computer Science & Information Systems
University of North Alabama
Florence, Alabama 35632
{xzhang6, jmsimpson1, chgossett}@una.edu

***Abstract*—The department of Computer Science & Information Systems at the University of North Alabama implemented an advising program for its students that incorporated the efficiency and consistency of group advising while retaining the personal interaction of one-on-one advising between a faculty member and a student. The department instituted a modified group advising program that began with a thirty-minute presentation to students as a group. Using advising packets that were composed of a student's transcript, a four-year curriculum plan, a pre-requisite chart, a projected two year course offering worksheet, each student met individually with a faculty member to create a schedule for the following semester. Although participation in this program was voluntary, within a couple of semesters of implementation, approximately half of the students in the department chose to be advised in this manner. The amount of time spent by faculty advising dropped from thirty minutes per student on average to approximately seven minutes. Furthermore, the quality of the advising offered to students improved because the information presented and the advising forms used in the process were reviewed and updated each semester.**

Keywords—advising methods; retention; recruitment; success in computer science; mentoring strategies

I. Introduction

Starting in the fall semester of 2009, the chair of the Department of Computer Science & Information Systems at the University of North Alabama decided to implement a modified group advising program as an alternative to the traditional individual student to faculty member advising method used by the faculty and students in the department. As others had discovered before, given the increasing complexity of curriculum and administrative requirements, advising has become a process that requires additional expertise on the part of a faculty member and one that uses more of their time as well [1]. Students have other burdens on their time outside of the classroom such as employment or family issues that hinder the traditional advising model [1] and the department chair wanted a program that could offer them additional flexibility to allow students to accommodate these needs. The department chair also wanted to offer an additional advising option that helped reduce the advising burden placed on faculty members in the department while at the same time providing students with an improved advising experience. The goals for the program were as follows:

1. Give students a better advising experience by providing them all of the information they need to make their own decisions about class schedules. This meant that the information provided had to be accurate, timely, and consistent with each faculty member in the department. Given this, advising forms and checklists were created and regularly updated to make the class selection process quick, simple, and foolproof. The forms and worksheet used by the department for modified group advising were developed to minimize the students' need to refer to the college catalog simply because many students are intimidated by the size of an average catalog and choose not to use it [2]. The modified group advising system also gave students another way to be advised if they did not work well with their assigned faculty advisor in the system.

2. Make students aware that graduating in only four years was possible and show them how it can be done. The key to doing this was the creation of several forms: a departmental four year course plan, a pre-requisite chart, and a projected two year course offering departmental worksheet. These forms were updated for each catalog year and contained all of the required information necessary for the student to graduate in four academic years. If a student selected classes based on the information provided by these forms and assuming they did well in the classes, they could complete their degree in four years. These forms also helped students see obscure details that commonly delay graduation such as having a minimum of 120 hours of credit. These forms also assisted the department with recruiting new students and retaining current ones. Potential students and their parents were frequently put at ease to see that it was possible to finish the program in a period of four years.

3. Reduce the advising workload of all faculty members in the department. On average, it was found that properly advising students on an individual basis took 30 minutes per student and each faculty member was assigned 25 advisees on average. Therefore, advising 25 students individually would take a faculty member about 750 minutes (30 minutes per student $*$ 25 students = 750 minutes) per semester. Under the modified group advising program, a single faculty member would convey information to an entire group of students for

30 minutes. After this, additional faculty members would join the meeting and it would take around 5 minutes to advise each student individually. Under this arrangement, a group of 25 students could be advised in 155 minutes total faculty time (30 minutes group + (25 students * 5 individual minutes per student) = 155 minutes total). In reality, it did not take 155 total minutes to conduct a modified group advising session from start to finish because multiple faculty members were handling individual advising at the same time. A typical modified group advising session took between 45 – 60 minutes to complete.

4. Make students aware of non-advising information that was important to them. Modified group advising sessions gave the department the opportunity to present information of any nature to students. This was particularly valuable because students were never together for any other reason. The department believed that effectiveness of communicating with students via email was spotty at best [1] and making announcements during classes was viewed as too much of an interference with normal class activities. Examples of this kind of information included but were not limited to: the availability of internships, ACM club membership, and scholarships and participating in an overseas study.

II. Literature Review

Education has been summarized as "communication between a faculty member and a student" and that "the more intimate that communication, the more likely the results will be long lasting" [3]. However, in regards to the traditional individualized advising approach, faculty members have noted many obstacles to this ideal. They point out that many students arrive to the advising session ill-prepared and ready to be told exactly what to do [2]. Also, faculty members have stated that they are burdened by other responsibilities that leave little time to prepare for advising, and they are frustrated by some faculty members "carrying disproportionate advising loads to compensate" for other faculty members who do not perform well in this area [2]. Faculty members also noted that advising should only be assigned to those who do it well and that advising should count in promotion and tenure decisions [2]. Other research points out that the success of any advising program lies in building a relationship between the student and the advisor [4] and that a good advising program can also increase student retention [5]. Also, the key for faculty to be successful in advising lies in receiving good training in this area. Many faculty are hesitant or unwilling to serve as advisors because "they don't want to put the student's degree completion in jeopardy" due to their lack of preparedness [6].

The Council for the Advancement of Standards in Higher Education (CAS) has published a list of goals for the advising process. These goals include:

- Help students understand themselves.
- Help students make decisions about educational and career goals.
- Monitor their academic progress.
- Encourage students to assess their own progress.
- Educate students about university policies.
- Direct students to resources that can help them [5].

These goals are complicated by the fact that the current generation of undergraduate students (millennials) see themselves differently and they present a special challenge to educators. Generally, millennials think of themselves as special and deserving of individualized attention. They are also accustomed to being protected, working in teams, and relying on others for support. Additionally, as a group, they tend to respond negatively to failure and have a lack of intrinsic desire [5]. Given these characteristics, some research suggests that a "dual or split model of advising" is appropriate for millennials because it "provides quick, straightforward advice about majors, courses offerings, and requirements" while still meeting their need for individualized attention [5].

Students who are performing poorly academically or who are on academic probation are not uncommon among computer science majors given the difficulty of the coursework. These students present their own unique challenges to advising. Research has noted that advising these types of students in a group environment is beneficial for them [7] and that group advising for all students is an "effective and efficient" advising technique [8]. One paper notes that success in a computer science program is mostly based on previous computer experience, math and verbal skills, and age [8]. However, effective advising can be a contributing factor to the success of computer science students [9].

Direct comparisons of the group advising approach to traditional one-on-one advising have been completed and published. These evaluations discovered that group advising has a significantly lower average advising time per student that can save the faculty member as much as 30 hours per semester. In addition, it was determined that both the student knowledge of college procedures and policies and the satisfaction of students with the advising process was unaffected by the method of advising [10].

III. Implementation

To use the modified group advising program each semester, the first step was to select the days and times for the sessions. While scheduling, care was to avoid Mondays (students would forget over the weekends) and Fridays (students simply are not going to attend on a Friday). During the fall and spring semesters, the department normally held four sessions during daytime hours while avoiding common class times. Also, one night session was held to accommodate evening students. The dates selected were usually prior to pre-registration milestones (i.e., date that juniors could begin to register, dates that sophomores could start registering). It was also determined that it was more effective and efficient to have "Junior/Senior" sessions and "Freshman/Sophomore" sessions. Given this structure, the department could tailor the information being presented to the appropriate group. For example, upper level students did not need to know about the requirements for being admitted into a major because they had already been admitted into one. Therefore, modified group advising sessions normally looked similar to the following:

- Session 1: Junior/Senior level, daytime
- Session 2: Junior/Senior level, daytime
- Session 3: Any level, evening
- Session 4: Freshman/Sophomore, daytime
- Session 5: Freshman/Sophomore, daytime

Dates, times, and locations of modified group advising sessions were posted throughout the building and sent to students via email. While participation in a modified group advising was voluntary, students that wanted to attend were required to sign up for a session with the departmental administrative assistant. This was done so the administrative assistant could print the following for each student and put them into a folder. This constituted an individual advising packet for each student:

1. Student's current transcript.

2. The computer science four year plan of study form, as shown in Fig. 1. This form contained the entire curriculum for the undergraduate computer science degree. Classes are

UNIVERSITY OF NORTH ALABAMA
Effective Fall 2014
Computer Science
Four Year Plan

	First Semester		Grade	Sem/ year	*Second Semester*		Grade	Sem/ year
FRESHMAN	EN 111 or EN 121	3 hrs.			EN 112 or EN 122	3 hrs.		
	HI 101 / HI 101-H / HI 201	3 hrs.			HI 102 / HI 102-H / HI 202	3 hrs.		
	Math 125 prerequisite*	3 hrs.			MA 125	4 hrs.		
	1st Natural Science	4 hrs.			*2nd Natural Science*	4 hrs.		
	CS 135	3 hrs.			CS 155	3 hrs.		
	TOTAL	**16 hrs.**			**TOTAL**	**17 hrs.**		
SOPHOMORE	1st Literature Course	3 hrs.			2nd Literature Course	3 hrs.		
	COM 201	3 hrs.			*1st Social & Behav. Sci. Elec.*	3 hrs.		
	CS 255	3 hrs.			CS Prog. Lang Elective	3 hrs.		
	MA 126	4 hrs.			MA 227 *or 3rd Nat. Sci.*	4 hrs.		
	CS 245	3 hrs.			CS 310	3 hrs.		
	TOTAL	**16 hrs.**			**TOTAL**	**16 hrs.**		
JUNIOR	CS 311	3 hrs.			MA 237	3 hrs.		
	CS 355	3 hrs.			1st CS Advanced Elective	3 hrs.		
	MA 345	3 hrs.			2nd CS Advanced Elective	3 hrs.		
	2nd Social & Behav. Sci. Elec.	3 hrs.			*General Elective*	3 hrs.		
	Fine Arts/Humanities Elec.	3 hrs.			**Apply for Graduation**			
	TOTAL	**15 hrs.**			**TOTAL**	**12 hrs.**		
SENIOR	CS 410W	3 hrs.			CS 455	3 hrs.		
	CS 420	3 hrs.			3rd CS Advanced Elective	3 hrs.		
	CS General Elective	3 hrs.			*General Elective*	3 hrs.		
	General Elective	3 hrs.			*General Elective*	3 hrs.		
	General Elective	3 hrs.			*General Elective*	1 hr.		
	TOTAL	**15 hrs.**			**TOTAL**	**13 hrs.**		

*Or ACT score =>28; other prerequisites may be required for eligibility to enter MA 125

- Natural Science paired sequence choices: BI 111/112, CH 111/111L/112/112L, ES 131/132. GE 111/112, PH 251/252 (all lab sciences)
- 3rd Nat Sci. can be one not already counted from above paired sequence or any other (at least 4 hours) from list on *2014-2015 Undergraduate Catalog*, page 61.
- Literature Courses paired sequence choices: EN 211/212, EN 221/222, EN 231/232, EN 233/234
- Social & Behav. Sci. Elec. choices: COM 205, EC 251, EC 252, ED 299, FL 101, FL101H, FL 201, FL 204, GE 102, GE 260,HPE 175, HPE 213, PS 241, PY 201, SO 221, SO 222, SRM 200
- Fine Arts/Humanities Elec. choices: AR 170, AR 281, AR 282, COM 133, FL 100, FR 101/111, FR 102/112, FR 201, FR 202, GR 101/111, GR 102/112, GR 201, GR 202, MU 222, MU 244, PHL 201, PHL 205, RE 221, RE 231, SP 101/111, SP 102/112, SP 201, SP 202, TH 210
- CS Programming Language Elective choices: CS 315, CS 325, CS 335, CS 390, CIS 315
- CS Advanced Elective choices: CS 360 or CIS 406, CS 421, CS 447 or CIS 445, CS 470
- CS General Elective choices: Any CS Programming Language Elec. or CS Advanced Elec. not already chosen, CIS 486, CS 480, CS 490
- Note: cannot count both CIS 445 and CS 447 for the CS degree major requirements
- Note: cannot count both CIS 406 and CS 360 for the CS degree major requirements

Additional Requirements:
Overall institutional GPA at least 2.0? ______
Major GPA (all MA and CS courses highlighted above) at least 2.0? ______
At least 36 hours at 300-400 level? ______
At least 30 hours at UNA? ______
At least 18 hours 300/400 hours in required in MA and CS at UNA? ______
Total hours at graduation 120 (beginning Fall 2014)? ______
At least 60 hours at a senior college? ______
Anticipated Graduation date? ______
Application deadline for Graduation? ______

Fig. 1. Computer Science Four Year Plan of Study Form

presented in a recommended, but not required, semester order. Courses in yellow are major courses, classes in blue are computer science electives, and courses in green are programming electives. At the bottom of the form, obscure college catalog requirements were listed so they could be checked off when completed. (Note that all of the advising forms and worksheets mentioned in this article are available for download in their original size and color format from: https://drive.google.com/open?id=0B0ysDLCX7zsnZGpobG8tT1BsWXM.)

3. The computer science pre-requisite form, as shown in Fig. 2. Since the department and college rigorously enforce pre-requisite requirements because of ABET standards, it was much more helpful for students to see course pre-requisite information in the visual format as shown versus a written

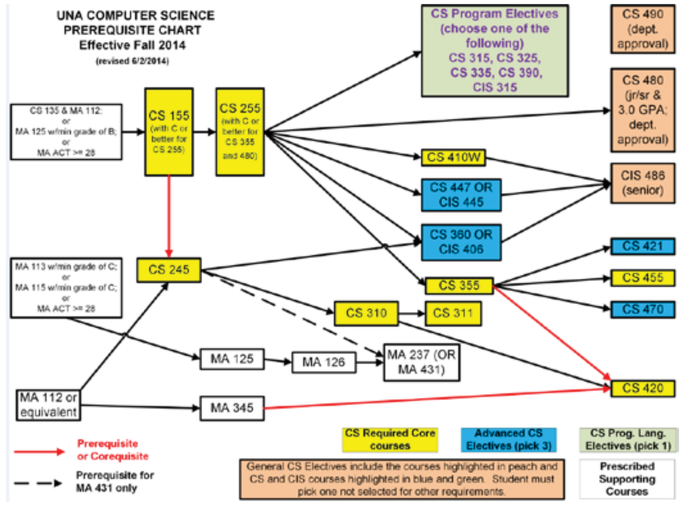


Fig. 2. Computer Science Pre-Requisite Chart

format as presented in the college catalog. Also, note that the colors used on the four year plan match those used on the pre-requisite chart and that they have the same meaning.

4. The specific "PIN Number" unique to each student that allowed students to register once the registration system was open for them.

5. The two year projected departmental course offering worksheet, as shown in Fig. 3. This is a worksheet that shows the projected offering schedule for every computer science course over the next six semesters. This allowed students to think and plan their course schedules for the next two to three semesters. Additionally, this form had the added benefit of helping computer science classes met mimimum class size requirements because students would routinely plan to take classes with each other to fulfill these requirements. Also, note that using the form required discipline on the part of the faculty and the department. If a class was on the projected schedule, alterations to the class were not allowed. For example, the worksheet shows that CS 310 would be offered during the day as a traditional face-to-face class in the fall semester only. The professor would not be allowed to change it to an online course or an evening class since it had been published on the form and made available to students. This gave students a great deal of confidence when planning out their schedules. But, students also realized that while a class would always be offered, if it did not meet the minimum enrollment size during pre-registration it was subject to being cancelled.

All of these forms were made available to both students and faculty members via the Internet. Additionally, copies

Projected CS Course Offerings						
Course	2015 (planned)			2016 (preliminary)		
	Spring	Summer	Fall	Spring	Summer	Fall
UNA Service Courses						
135	Day	Online	Day	Day	Online	Day
CS Major: All Students						
155	Day	Day	Day	Day	Day	Day
245			Day			Day
255	Day	Day	Day	Day	Day	Day
310	Day			Day		
311			Day			Day
355			Day			Day
410			Evening			Evening
420			Day			Day
455	Evening			Evening		
CS Major: Programming Language Electives (choose 1)						
CIS 315		Day	Day		Day	Day
CS 315				Day		
325						Day
335	Day					
390			Day			
CS Major: Advanced CS Electives (choose 3)						
360	Day					
421	Day					
430				Day		
*CIS 445	Day		Online	Day		Online
*CS 447						
470				Day		
CS Major: General CS Electives (choose 1 from this list or an additional one from the above electives lists)						
CIS 486	Day		Day	Day		Day
**CS 480	Internship	Internship	Internship	Internship	Internship	Internship
CS 490	indep study/ permission required	indep study/ permission required	indep study/ permission required	indep study/ permission required	indep study/ permission required	indep study/ permission required

Fig. 3. Two Year Projected CS Course Offering Worksheet

were printed and placed in a rack located in the hallway outside of the office of the departmental administrative assistant so students could access them as needed. It became common for all students, whether they participated in the modified advising program or not, to use these forms to plan their course schedules for upcoming semesters. It also became commonplace for faculty members to use these forms for their advisees in both modified group advising sessions and during traditional one-one-one advising meetings with their advisees.

When the modified group advising program was first implemented, the department offered randomly drawn door prizes, sodas, and candy to entice students to come to the meetings. After a couple of semesters, it was realized that these enticements were no longer necessary and eliminated everything but the candy.

At the time of a modified group advising sessions, students would arrive and sign in on a sign-in sheet. The administrative assistant would use this list to email all of the professors in the department to let them know if one of their advisees had attended a modified group advising session. If a student had not signed up with the administrative assistant beforehand for a session, they would be sent to her so she could quickly create an advising packet for them.

To start the meeting, a single faculty member, assisted by a PowerPoint presentation, would speak for about thirty minutes to the entire group. The PowerPoint presented important generalized advising and non-advising information and was typically organized as follows:

- First Thoughts
    - Pin numbers required for registration.
    - Importance of pre-registration.
    - Departmental student organizations.
- Graduation
    - Graduation requirements.
- Making a Schedule
    - Use your transcript, the departmental four year plan of study, the pre-requisite chart, and the projected course offering worksheet provided to make a schedule.
    - Focus on earning admission into the major.
    - Focus on completing the math sequence.
    - Note that pre-requisites are rarely waived.
    - Finish course sequences in back to back semesters if possible.
    - If you follow these guidelines, you will see that your schedule almost “writes itself.”
    - Think ahead two or three semesters.
    - Focus on learning and high GPA rather than heavy course loads.
    - Know your strengths and weaknesses in regards to online and early hour courses.
- Additional Details
    - Math and science requirements.
    - Possible minors.
- Final Thoughts
    - Online/hybrid courses.
    - Overseas study.
    - Importance of GPA.
    - IT certifications.
    - Co-op programs.
    - Internships.

Once this presentation was complete, other faculty members would join the session. The advising packets would be distributed to each student. Using their transcripts, students would mark their completed classes and classes they were currently taking on the four-year plan. Students would then meet individually with a faculty member to discuss and select their course schedules for the next semester.

IV. Results and Lessons Learned

Overall, the department found that once students attended a modified group advising session, they would frequently choose to use it again semester after semester. Students liked the certainty that they could come to a short meeting, be advised individually, and leave with their registration PIN number and a proposed schedule for the next semester in hand. Students also liked the convenience of not having to worry about working around their assigned advisor's schedule or dealing with making an appointment to meet with a faculty member. After a couple of semesters, the department was routinely advising half of the department's students through this program.

Other benefits included the ability to steer students towards departmental courses that historically had trouble making. Advisors would tell students "if it fits in your schedule, it would be helpful to your fellow CS students for you to take this particular CS course this semester to help it make." This tactic was extremely useful in helping summer semester class offerings meet minimum enrollment requirements.

Encouraging students to minor in a departmental degree or to participate in other departmental offerings (e.g., HCI/UX program, robotics program) helped to ensure that departmental classes would meet the minimum number to make and would help grow these programs. In addition, modified group advising sessions gave the department a forum to answer students' questions about new classes and new programs offered by the department.

Several faculty members in the department thought the advising forms were so useful that they would not advise students in any manner until they had completed the four-year plan document. The department also realized that the modified group advising process could be a bit of an administrative burden. Having a competent administrative assistant to handle the administrative process is vital to ensure the success of the program.

The department chair was also convinced that this process saved students time and money. It prevented them from making scheduling mistakes, taking unnecessary extra classes, incurring late registration or class change fees, or staying in college for extra semesters. The department chair also believed that by preventing students from making scheduling mistakes, departmental faculty and staff members were saved the time and effort it took to react to and resolve these mistakes (e.g., course substitutions).

An additional benefit of the modified group advising program is that it came to serve as an unofficial training program for new faculty members in the department on advising issues. Traditionally, new faculty members were not assigned advisees during their first year of employment to give them time to settle into the university and to allow them to become familiar with administrative policies and the nuances of the department's curriculum and courses. While it was not required, several new faculty members chose to attend the presentation portion of a modified group advising session and then shadow other faculty members as they advised students in order to improve their skillset in this area.

V. Future Work

In the future, the plan is to survey both faculty members and students who had participated in this modified group advising sessions for their feedback. The surveys will be designed and implemented to measure their satisfaction of and the experience with the modified group advising process, and seek their specific suggestions to improve it. Both qualitative narratives and quantitative data will be collected. The survey results will be used to improve the effectiveness, the efficiency, and the overall quality of our modified group advising approach.

References

[1] J. Aucoin, E. Jones, S. Letvak, P. Crane and K. Gilbert, "Targeted Group Advising", *Nurse Educator*, vol. 29, no. 6, pp. 226-227, 2004.

[2] R. Dillon and B. Fisher, "Faculty as Part of the Advising Equation: An Inquiry into Faculty Viewpoints on Advising", *NACADA Journal*, vol. 20, no. 1, pp. 16-23, 2000.

[3] J. Cogdell, "The role of faculty advising in science and engineering", *New Directions for Teaching and Learning*, vol. 1995, no. 62, pp. 65-70, 1995.

[4] J. Coll and C. Zalaquett, "The Relationship of Worldviews of Advisors and Students and Satisfaction with Advising: A Case of Homogenous Group Impact", *Journal of College Student Retention: Research, Theory and Practice*, vol. 9, no. 3, pp. 273-281, 2007.

[5] T. Montag, J. Campo, J. Weissman, A. Walmsley and A. Snell, "In Their Own Words: Best Practices for Advising Millennial Students about Majors", *NACADA Journal*, vol. 32, no. 2, pp. 26-35, 2012.

[6] M. Cuevas, M. Loya and R. Bachman, "Faculty Advising Fellows: Lessons Learned", *Mentor: An Academic Advising Journal*, vol. 1, 2015.

[7] M. Austin, E. Cherney, J. Crowner and A. Hill, "The Forum: Intrusive Group Advising for the Probationary Student", *NACADA Journal*, vol. 17, no. 2, pp. 45-47, 1997.

[8] J. Battin, "Improving Academic Advising Through Student Seminars: A Case Study", *Journal of Criminal Justice Education*, vol. 25, no. 3, pp. 354-367, 2014.

[9] P. Oman, "Identifying Student Characteristics Influencing Success in Introductory Computer Science Courses", *AEDS Journal*, vol. 19, no. 2-3, pp. 226-233, 1986.

[10] J. Grahn, P. Kahn and P. Kroll, "A Faculty Team Approach to Group Advising", Journal of College Student Personnel, vol. 17, no. 1, 1982.

Driving Strategic Advantage into Information Systems Programs at a Small Texas University

John M. Medellin*, Kirk Fischer, James King
Faculty at the Master of Science in Information Systems Program
University of Mary-Hardin Baylor
Belton, TX 76513-2599, USA

Abstract— ***The University of Mary Hardin-Baylor is a small Texas university that was founded in 1845 and has a thriving Master of Science in Information Systems program. The school has experienced explosive growth in enrollment over the past five years. University leadership tasked a group of professors to recommend and implement enhancements that provide strategic advantage for students entering or returning to industry. The cost of this initial venture was under $10,000USD and provides significant price performance versus traditional methods.***

In this paper, the approaches to selection of enhancements and implementation focus are discussed. This is an initial study in what small schools can do to enhance their offerings and drive a very high value proposition for students, faculty and employers. The team implemented cloud environments in OpenStack and Microsoft Windows Enterprise Server 2012 with Hyper-v on a late model processor containing Second Level Address Translation. In addition, the students are modeling solutions that use the Standards Performance Evaluation Corporation SPECjbb2015 for java B2B environments and the SPECvirt2013 for virtualized/Cloud environments. These projects also have the advantage of preparing them for CompTIA certifications. New horizons are included for development of additional facilities like a hybrid cloud and additional certifications.

Keywords—curriculum design, cloud computing, student stem competencies, SPEC Corporation, OpenStack, Windows Hyper-V, CompTIA

I. INTRODUCTION

Baylor University was chartered by the Republic of Texas in 1845. Judge R. E. Baylor wanted to provide an organization where values-based learning in a higher education setting could be delivered to the growing population of the Republic of Texas. The male part of the University relocated from Independence, Texas to Waco Texas and merged with Waco College to form Baylor University in the 1890s. That institution still operates to this day and is a top University in a variety of disciplines.

Baylor College for Women moved from Independence to Belton, Texas in the 1890s and remained a female-only University until the early 1970s. The name was changed to Mary Hardin-Baylor College in honor of an alumni and benefactor in the 1930s. It formally became University of Mary Hardin-Baylor in the 1960s. The name still remains today as does the pride in the longest tradition of service in higher education to the citizens of Texas.

The Master of Science in Information Systems program was founded in the 1990s in order to accommodate the growing need for higher education in Computer Science for Business. The degree continued to attract graduates in a moderate fashion through the first decade of the 2000s and experienced explosive growth in the 2010s. The program has around 330 active students with graduates usually numbering approximately 150 per year. This program is one of around 100 offered in the United States.

The faculty had been investigating additional avenues to expand the opportunity for graduates through curriculum enhancement. Many of these graduates were competing with much larger programs (such as the academic sibling; Baylor University) and an expeditious approach was needed to get additional world-class credentials into the repertoire of the students. It would take years, if not decades to achieve the scale, reputation and credentials that these larger competitors had in their portfolios.

This paper discusses the various approaches that were evaluated by the faculty and the implementation of the enhancements that were considered most effective in driving superior credentials for the students. The participants in this effort believe the model will scale to larger Universities.

II. APPROACHES EVALUATED

The discussions for enhancement ranged from the purely theoretical, pedagogical or very hands-on. The summary of these approaches reviewed is discussed below.

A. *Theoretical enhancement approach*

The discussions were approached from the perspective that doing more teaching of theory concepts would enhance the students' chances to compete. The advocates of this approach cited how leading universities (those more focused on research) relied on investigative studies to enhance their own (and their students') reputation as thought leaders [4]. Some universities would rely on investigative grants from both public and private sources to study their particular domains and also enhance the student experience [8]. While UMHB's faculty are engaged in research, it is a teaching institution. The pure-theory approach seemed inconsistent with the university's mission.

B. *Pedagogical enhancement approach*

The discussion on implementing additional pedagogical approaches centered around providing more training to the instructional staff on engaged learning in the classroom. The learning pyramid drives significant participation by the student in the process and ultimately delivers higher results in recollection [7]. This is a very worthwhile approach to driving better results in student learning and has been adopted by many educators.

It was the opinion of the faculty that most of the instructors had achieved a high level of learning effectiveness at the University. This was empirically witnessed by the better than average scores received on the Individual Development and Educational Assessment (IDEA). The IDEA methodology and process compares scores from the University in question to the scores received at other Universities from student assessments of the instructor, material and delivery mechanisms [2].

The group felt the pedagogical methods that were employed were not, by themselves, going to enhance the competitive advantage for jobs in the market.

C. *Hands-on enhancement approach*

The proponents of this approach recommended a curriculum offering which focused on students' having the ability to demonstrate competencies and capabilities in real situations. These simulations could also lead to a curriculum which provided the ability to secure some key certifications that would allow them to compete with experienced candidates. Items like the Microsoft OS certifications [6] or the CompTIA certifications [3] were strongly proposed as compliments to the hands-on curriculum.

The group felt that a focus that led only to enhancing the curriculum in such manner would begin to change the educational offering to resemble a technical training school rather than a university. It was felt that these certifications would add to the value but should take a second place to imparting agnostic concepts that would withstand the test of time.

D. *Driving new technologies and methods*

The strongest support was for driving new technologies and methods into the curriculum. The group felt that the offerings could be better served by adding some of the technology disruptors [1] including:

1. Cloud computing and virtualization
2. Enhanced Web programming concepts
3. Standardized testing and benchmarking
4. Performance measurement
5. Analytics and large data

The above technologies were debated as to which combination would drive the largest advantage for the students. The discussion also focused on the ability of the faculty members to deliver the above without significant re-training in those fields. At the end of the analysis, all agreed a combination of some or all of the above should be included in the next academic offering.

III. A COMPETENCIES-BASED APPROACH

The faculty felt that more input was required from external sources in order to define the enhancements. A series of inquiries were commissioned including:

- Engaging outside experts
- Identification of standards/tools that were used in industry
- Identification of credentials that could potentially enhance the students abilities

The actions taken in these areas are detailed below.

A. *Engagement of outside experts*

The MSIS program has an advisory panel composed of industry leaders that meets on an annual basis. The panel is composed of 8 senior executives from Fortune 500 corporations. Their focus is on IT services, about half of them are client facing (they serve external consumers) and half are internal (they serve the organization).

A panel discussion was held with these executives and the academic leadership of the program. The recommended competencies that would be required for jobs in these companies were as follows:

1. Mobile computing
2. Cloud computing and virtualization
3. Java programming
4. Web programming
5. Program management
6. Testing and benchmarking
7. Performance measurement
8. Analytics and large data
9. Cyber Security

The program already had strong offerings in mobile, java, program management and cyber security. It was felt that some existing courses could potentially be enhanced by industry certifications.

B. *Identification of standards and tools used in industry*

The faculty team investigated standards and tools that were either used or would be used in larger corporations. The findings by category (which the university might have access under certain licensing) included:

1. Cloud computing and virtualization
    a. OpenStack
    b. MS Windows Server 2012 (Hyper-V)
    c. VMWare
2. Web programming
    a. XML
    b. SGML
3. Testing and benchmarking
    a. IBM Rational Suite
    b. LoadRunner
    c. Standards Performance Corporation (SPEC)
4. Performance measurement
    a. Standards Performance Corporation (SPEC)
    b. IBM Tivoli Suite
5. Analytics and large data
    a. Hadoop 1.0 and 2.0
    b. MongoDB
    c. R suite

After further debate and cost evaluation, the IBM and VMWare tools were temporarily excluded from the next phase of analysis. Most instructors felt that the mainline concepts delivered by those packages could be taught by using the other tools selected.

C. *Identification of potential credentials*

Certification in the IT discipline is sought by employers as a means of ascertaining a level of competency. Certifications exist from a variety of bodies and they range from vendor-specific technology to more domain-specific bodies. Many current graduates actually take additional certification courses and exams to make them more competitive in the market. The team investigated which of the certifications would mean most to students/employers and could be delivered within reasonable parameters. The initial recommendations were to pursue the following:

1. Program management (through the PMI institute) [10].
2. Cloud computing (through CompTIA) was identified as a growing credential and with demand from a variety of job boards.

Other certifications could be offered as the new offering progressed.

D. *Initial phase of curriculum enhancements*

The final recommendation for an initial phase (due in the Fall of 2016 semester) included a combination of:

1. Enhancements to existing subjects in order to enable greater hands-on approaches.
2. Development of new "pilot" courses to test student (and employer) appetites for these skills.
3. Creation of leading-edge environments for use by students.
4. Creation of an initial "cohort" approach to linking several courses so that projects would begin in one course and be elaborated in subsequent courses during the student's program of study.

A principal strategic objective was to enable the student to communicate to the employer the leading theories and technologies that had been studied and implemented ahead of some larger schools. This would provide a significant strategic advantage over peers and larger programs.

IV. Preparation for Implementation

Given the approval to proceed, the team began a series of processes to implement in time for the Fall 2016 semester.

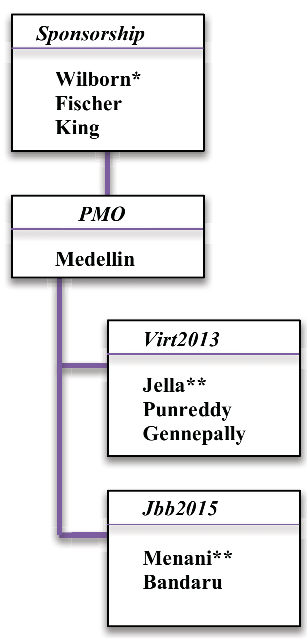
A. *Team acquisition and project launch*

The graduate programs director empowered the proposing professors to begin the recruitment process. The initial estimates called for two graduate assistants with the following skill sets:

- Experience in Java deployments and application development.
- Experience in infrastructure development, support and maintenance.

Applications were received from 52 interested parties. Several were selected and two finalists were offered the assitantships. The interest in participating in such a program was so large that additional students volunteered to participate. Three more students were accepted in this category. The organization diagram for the project is below:

Figure 1: Organization Diagram



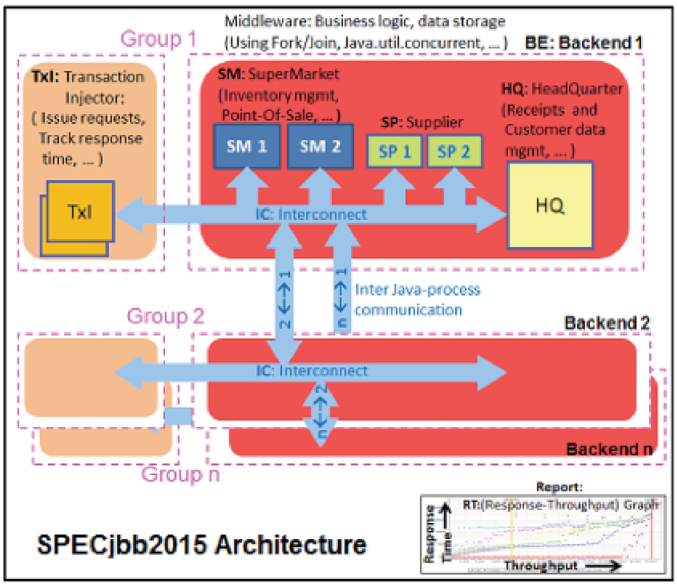
Key: *=Dean of the Graduate School and Research, **=Team Leader

The project was to execute an average of 28 hours per person per week and would be tasked with delivery of an initial environment by May 2, 2016. Weekly status reports and semi-monthly meetings were held to track completion and issue resolution. Formal project plans were developed according to PMI/PMP guidelines and reporting was also executed under them.

B. *Environment and model installation*

The University made an expenditure to acquire the SPECjbb2015 [11] java standard model and the SPECvirt2013 [11] virtualization standard model. The overall architecture for the SPECjbb2015 is in Figure 2 and the architecture for the SPECvirt2013 is in Figure 3 below.

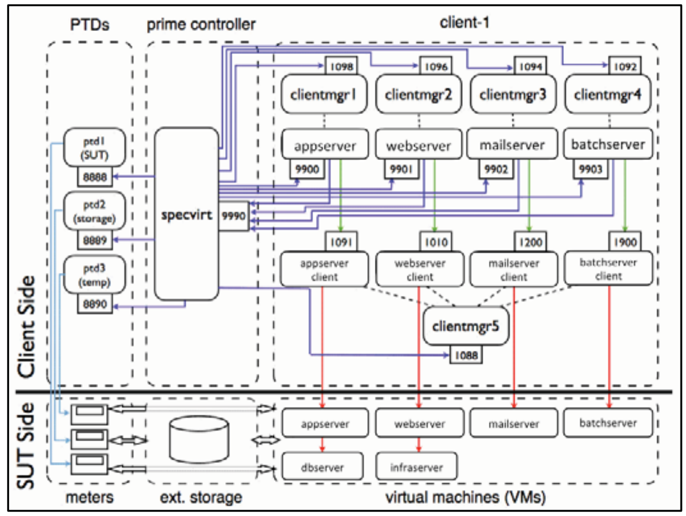
Figure 2: SPECjbb2015 Architecture



The SPECjbb2015 (Figure 2) constitutes a Business to Business and Business to Consumer simulation for a Supermaket Chain. The model can be extended to include several super markets and several central offices in a variety of virtual machine settings.

The SPECvirt2013 (Figure 3) offers a set of tools to measure the efficiencies of various components of the virtual environment. The software works as an overlay on to the virtualized environment and measures each Virtual Machine performance as well as the total cloud environment. It is particularly useful when teaching students the principles of balancing loads in a variety of environments. Web environments in particular are prone to spiking due to unforcasted demand [9] a condition that is prevalent in public cloud offerings. This tool provides metrics of usage and performance along different types of scale tests in the web environments of the SPECjbb2015 application stack.

Figure 3: SPECvirt2013 Architecture



A donor from the University community provided a late model computer with key components that were recommended by the SPEC corporations (including Second Level Address Translation; SLAT and 16GB of real addressable RAM). The unit that was sourced contained the following specifications:

- Hewlett-Packard Envy 15t
- Intel i6700 quad core processor
- 16G RAM
- 1TB Hybrid SSD
- 4GB NVIDIA GTX 950M chip

The team proceeded to implement the environment in the following sequence:

1. Install Windows Enterprise 2012 R2 and Windows-Hyper-V (hypervisor).
2. Create two (production and test) virtual machines with Windows-10 host Operating System.
3. Install SPECjbb2015 single VM, supermarket and corporate office (compiled under current java version).
4. Execute base level of transaction engine, run and compare results to install documentation.
5. Vary execution of transaction engine by 10% and record results.
6. Re-execute the final steps measuring the impact on the virtual machine and on the overall cloud using the SPECvirt2013 tool set.
7. Finalize documentation and prepare demo environments.
8. Prepare for turn over to next set of graduate assistants (some will graduate and others will stay on for another semester).

The efforts under way were on target for their delivery dates as of the writing of this document.

C. *Curriculum definition – trial cohort approach*

Given the above teams and scope, the Faculty turned their attention towards incorporating these new technologies into a "trial cohort" set of students. The subjects that would be impacted and their respective impacts were as follows:

1. (MSIS 6322) Systems Analysis, Modeling & Design: 50 students would participate in defining enhancements to the SPECjbb2015 suite. These enhancements would include OOAD work products [5]: (a) Use Cases (b) Activity Diagrams (c) Sequence Diagrams (d) Package Diagrams (e) Class Diagrams and (f) Pseudo-code for implementation.

2. (MSIS 6391) Architecture & Models in the Cloud: A select group of 25 students would deploy the newly developed cloud environment to measure performance impacts of additional code on the virtual machines and on the total cloud.

3. (MSIS 6395) System Construction & Implementation: The 50 students in MSIS 6322 would develop and implement their designs into the SPECjbb2015 code base (some would receive extra credit for actual integration of their designs). Students that had already taken MSIS 6391 could receive extra credit on either front; integration OR measurement of their work products on the cloud infrastructure.

The course MSIS 6391 would provide the necessary training in order to sit for the CompTIA Cloud Essentials certification.

D. *Additional curriculum enhancements*

The following additional "trial" courses would also be offered as part of the Fall 2016 semester offering:

1. (MSIS 6392) Large Data & Analytics: a survey of the data science and approaches to hypothesis formulation, data planning, data assembly and modeling in support of key decision making.
2. (MSIS 6393) Web Programming: a review of programming theories aimed at applications with the XML and SGML paradigms. Focus on efficient methods for web programming and maintenance.

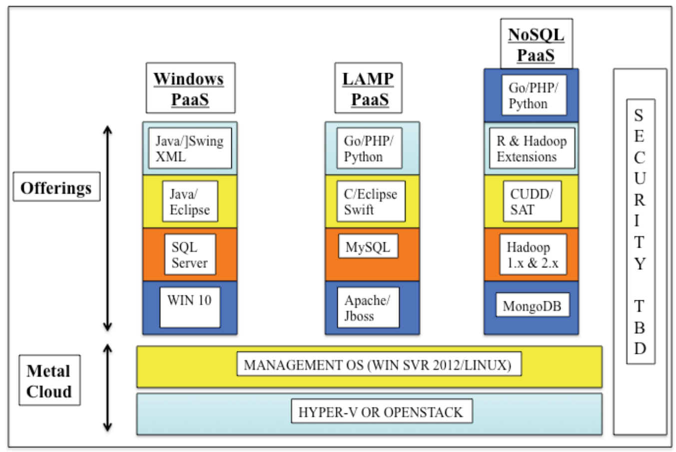
V. GOING FORWARD PLANS

This is now considered a strategic initiative at the university and will continue to evolve through the efforts of the faculty and the dedicated graduate assistants. Plans are to continue to fund the assistantships and potentially grant additional monies for new products.

A. *Future vision: the "CRUCloud"*

The students at the University of Mary Hardin-Baylor are known as the "Crusaders" and refer to themselves as "The CRU". A future vision of this program is to realize the implementation of a hybrid cloud environment (which has been named ***"CRUCloud"***) with the architecture indicated in the figure below.

Figure 4: Target CRUCloud Architecture



Not shown in the figure above is a second cloud that will reside in the Microsoft Azure environment. Students in the program are allowed a small Azure environment while enrolled. The future objective will be to allow them to develop their designs on the CRUCloud and migrate them for large scale testing in the MS Azure environment.

B. *Short term enhancements*

Additonal enhancements are being contemplated in the following areas:

1. Offering Project Management Institute certification exams coaching as part of the Program Management course.
2. Implementation of additional certifications for Java.
3. Implementation of coaching programs for the CompTIA cloud security certification.

Alternatives continue to be evaluated by the enhancement committee as part of their university charter.

VI. ACKNOWLEGEMENTS

Special thanks go out to Drs. Ho-Chang Chae and Russell Torres of the University of Mary Hardin-Baylor for their leadership and support in developing the web programming and data analytics courses referred to in this text.

References

[1] H. Alkhatib, P. Faraboschi, E. Frachtenberg, H. Kasahara, D. Lange, P. Laplante, A. Merchant, D. Milojicic, K. Schwann; "What Will 2022 Look Like? The IEEE CS 2022 Report" IEEE Computer Volume 48, Issue 3, p. 68-76.

[2] Benton, S. L., & Cashin, W. E.; "Student ratings of instruction in college and university courses". In Michael B. Paulsen (Ed.), *Higher Education: Handbook of Theory & Research*, 2014, Vol. 29 (pp. 279-326). Dordrecht, The Netherlands: Springer.

[3] https://certification.comptia.org

[4] J.A. Fulcher; "The advantage of industrial R&D in the training of university students" TENCON 2001. Proceedings of IEEE Region 10 International Electrical and Electronic Technology (v. 2, p. 881-883).

[5] C. Larman; "Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development (3rd Edition)" c. Pearson Education 2005, Upper River, NJ.

[6] http://www.microsoft.com/en-us/learning/windows-certification.aspx

[7] http://siteresources.worldbank.org/DEVMARKETPLACE/Resources/Handout_TheLearningPyramid.pdf

[8] R. Pirinen; "Learning within externally funded research and development projects in University of Applied Sciences: An analysis" Proceedings of the 14th International Conference on Interactive Collaborative Learning (ICL 2011) p. 175-182.

[9] A. Salam, Z. Gilani, S. UI Haq; "Deploying and Managing a Cloud Infrastructure" p. 130, Chapter 5. John Wiley & Sons, 2015.

[10] http://www.pmi.org/Certification.aspx

[11] https://www.spec.org

Evaluating Program Assessment Report

Nasser Tadayon
Department of Computer Science and Information Systems, Southern Utah University
Cedar City, Utah, USA

Abstract - *The term meta-assessment is used to evaluate an assessment process with the purpose of weighing the quality of assessment. The objective of meta-assessment is to investigate the factors that leads to the improvement of the assessment process in order to get better informed decisions about student learning outcomes. The process for meta-assessment should use both quantitative and qualitative methods with all possible resources such as external reviewer. To the external viewer, one of the main methods used to evaluate an assessment of the program, is to examine the program final report. The Joint Committee on Standards for Educational Evaluation (JCSEE) introduces a set of standards that are classified into categories of Utility, Feasibility, Propriety, Accuracy, and Evaluation Accountability.*

In this paper, the internal/external review of the assessment report as one of the most common meta-assessment process used at many institutions is checked against JCSEE standards. In conclusion, the strength, the weakness, and some suggestions for improvement of the process is pointed out.

Keywords: *Evaluation, Program Report, Assessment, Meta-assessment, Standards*

1 Introduction

The concept of assessment and evaluation of processes has been in use for many decades and it has evolved and shaped based on its application and intent. There is a distinction between the words “assessment” and “evaluation” even though in many cases they have been used interchangeably and as synonym. Based on MIT Teaching and Learning lab [1], assessment is defined as systematic collection of data to monitor the success of a program/course in achieving intended learning outcomes for students and evaluation is a judgment by the instructor or educational researcher about whether the program or instruction has met its Intended Learning Outcomes. In WIKIPEDIA [2], assessment is based on its purpose but “Educational assessment” is defined by the process of documenting knowledge, skills, attitudes, and beliefs. In other hand, evaluation is defined as “a systematic determination of a subject's merit, worth and significance, using criteria governed by a set of standards”. There are many other interpretations of the words from educational entities who are involved in assessment process.

There are variety of assessment processes used in academic programs both within and across institutions mainly because of their difference in goals and objectives. However, within academic, the assessment processes used, have two conflicting objectives: quality improvement and accountability. [3] Universities mostly emphasize on quality improvement, which has been their main tool in competing against other institutions, while the government is mostly concern with accountability, which guarantees certain standard for the quality of the services provided to society by higher education institutions. [4] Educational institute may have many levels of assessment used for different accreditation at program or university level. However, most institution involved in assessment process, agree that the main objective of their assessment or evaluation is to obtain continues improvement. In general, evaluation involves both quantitative and qualitative analysis whereas assessment has more emphasis on broader aspect of collection, analysis, and interpretation of information associated to a particular outcome/objective [5]. Assessment processes are usually adjusted based on the outcome/objective of the assessment. There are different types of assessment tasks that can be classified as diagnostic, formative, integrative, and summative. [6]

There are two aspect of evaluating assessment process. One relates to verifying assessment which relates to identify set of activities to ensure that the assessment process is correctly implemented. The other is validation which refers to set of tasks that ensure that the assessment process is traceable to the objective.

Evaluator concern for a consumer-oriented, professionally developed set of guidelines or evaluation criteria led to the development of standards for judging an evaluation. A profession- wide Joint Committee on Standards for Educational Evaluation (JCSEE) published a set of thirty standards in 1981 refined in 2011, called the Standards for Evaluation of Educational Programs. The standards were approved as an American National standard, were classified in to different categories.

2 Program Evaluation Standard Statements

The evaluation standard has helped institutions explain the quality of assessment at every level of a university from program to department to college to the university. Meta-assessment helps to identify a university shortcoming or area of improvement within its assessment practices and weights the assessment process against the objective of assessment. Explicit or implicit, a primary role of an assessment office and people who are involved, is to improve assessment practice. However, many assessment processes cannot provide data about the aggregate quality of assessment.

The program evaluation standards are classified into Utility, Feasibility, Propriety, Accuracy, and Evaluation Accountability as follows:

- Utility Standards: The utility standards are intended to increase the extent to which program stakeholders find evaluation processes and products valuable in meeting their needs.

- Feasibility Standards: The feasibility standards are intended to increase evaluation effectiveness and efficiency.

- Propriety Standards: The propriety standards support what is proper, fair, legal, right and just in evaluations.

- Accuracy Standards: The accuracy standards are intended to increase the dependability and truthfulness of evaluation representations, propositions, and findings, especially those that support interpretations and judgments about quality.

- Evaluation Accountability Standards: The evaluation accountability standards encourage adequate documentation of evaluations and a meta-evaluative perspective focused on improvement and accountability for evaluation processes and products.

3 Meta-Assessment

Meta-assessment goes beyond assessment as it examines elements and tools of assessment as well as the necessary and sufficient conditions and the need for assessment. The formative and ongoing evaluation can be used to improve assessment by analyzing the quality of process used and its relevant to the objective. In contrast, the summative evaluation is used to rank the overall assessment results. The are many article related to formative assessment vs summative assessment in general [7] and the use and structure of a formative meta-assessment [8].

Meta-assessment involves evaluating the quality of the overall assessment process itself. [9] Ory suggested to consider the following tips related to conducting meta-assessment:

- At the outset decide why your university should conduct meta-assessment. If it is to help with accreditation then make sure that your evaluation criteria, which are usually articulated through a rubric, are synced with the accreditor's standards.

- If you would like the process to help faculty and staff learn about assessment, then make sure that this group participates in evaluating assessment reports. This participation should include training from an assessment expert.

- Last, help administration make decisions based on meta-assessment results. For example, if the results indicate that a few programs are struggling with assessment then the administrators could allocate money to the respective program coordinators for training. [10]

Meta-assessment relates to the question of process evaluation vs outcome evaluation. When involved in meta-assessment of programs, the term ipsative vs normative evaluation [11] has been raised which means if the objective (in most cases, it would be improvement) is apparent if the assessment result shows improvement from one year to other (ipsative) or it is compared against another (normative).

There is a sequence of activities that should be followed in order to implement an assessment. The TLL (Teaching and Learning Lab) at MIT, identified the general steps and summarized it as follows:

- Identify Intended Learning Outcomes
- Identify Research Questions
- Develop Research Design
- Select Sampling Frame
- Select Appropriate Data Collection Methods
- Construct Measurement Instruments
- Select Appropriate Data Analysis Techniques
- Consider Communication and Dissemination of Findings

Same as any assessment process, meta-assessment must also have a sequence of steps with the main objective as to advance and improve the assessment process which means the closing of the loop in their process. This means that the results of evaluations must be used to improve assessment process otherwise it would be redundant activity. As an example, the following rubric is used in meta-assessment for closing the loop at Ohio State University [12]

Meta-assessment Rubric for Closing the Loop - Program/Unit: ____________________

Performance Criteria	Exceeds the Standard (3)	Meets Standards (2)	Developing (1)
Targets for Improvement	Targets for improvement were composed based on area of weakness as identified in previous year's assessment. All proposed targets were implemented prior to the next assessment cycle.	Targets for improvement were composed and were relevant to the previous year's assessment. Some or most of the proposed targets were implemented prior to the next assessment cycle.	Targets for improvement were composed, but were irrelevant to the previous year's assessment or were not implemented.
Re-Measuring	The Program/Unit recreated the same assessment, using the same sample size and tools as the previous year in order to collect data for a valid comparison.	The Program/Unit attempted to recreate the same assessment. Sample size might have varied slightly or the tool may have been similar, but not exactly the same (NSSE data compared to internally developed survey about the same information, for example).	The Program/Unit attempted to re-measure the same outcome, but either the sample size or the tool was so different from the previous year that no valid comparison could be made between the previous and current year's data.
Evidence	The Program/Unit included supporting evidence, including a detailed analysis and comparison of the previous and current years' data.	The Program/Unit included supporting evidence of the current year's data, but failed to include a comparison to the previous year's data.	Supporting evidence is anecdotal, invalid, or absent.
Improvement	Implementation of targets for improvement resulted in achievement of the outcome.	Implementation of targets for improvement resulted in improvement of the outcome.	No improvement to outcome.

However, the steps that can be used to assess most institutional and departmental assessment plans and programs with the goal to improve students' academic achievement should exemplify the following five principles: [13]

- Mission and educational goals are reflected in the assessment process.
- The conceptual framework is effective.
- Institutional personnel are involved in the design of the assessment process.
- Data are collected through multiple measures.

- An assessment of assessment activities is established

Furthermore, the evaluation of assessment process in university should be established by means of an office, director, and committee, that would review the entire assessment process not just the result. The assessment office is responsible to evaluate and provide feedback on the assessment plan in terms of its design, utility, external reviews, standards, and the recommendations suggested by the data.

4 Assessment Report

Academic Program Review Committee (APRC) is the university level committee based on Policy 6.41, to oversee the process of academic program reviews and monitor and improve the quality of degree programs offered at institution. As such, the committee reviews programs based on the reports compiled by the college housing the programs. Each program report is provided by the department offering the program. This process is part of cyclical academic program review and reports. The review process would also go through external reviewers before presenting to the commissioner's office. As a part of review process for the programs, academic program review committee, would assess the report as well as all related data based on several categories which are defined as:
Section 1: Purpose and R411 Data

1.A. Mission Statement (Including program goals and objectives)

1.B. R411 Data Form

Section 2: Operations

2.A. Faculty Characteristics

2.B. Administrative Support

2.C. Program Resources

2.D. Student Development

2.E. Program Climate

Section 3: Instructional Programs (address each academic unit)

3.A. Curriculum

3.B. Student Learning Outcomes

3.C. Assessment

3.D. Special Considerations or Issues

The general report for a program at institutional level should contain statements addressing the following items:
1.A.1 Mission Statement: Clear published mission statement and goals reflecting its purpose, characteristics, and expectations, give direction for its efforts, and derive from, and are generally understood by, its community.

1.A.2 Definition of Mission Fulfillment: The program defines mission fulfillment in the context of its purpose, characteristics, and expectations. Guided by that definition, it articulates program accomplishments or outcomes that represent an acceptable threshold or extent of mission fulfillment.

2.A.1 Qualification and Sufficiency of Faculty: Consistent with its mission, intended outcomes, services, and characteristics, the program employs a sufficient number of qualified faculty to achieve its educational objectives and to assure the integrity and continuity of its programs and services, wherever offered and however delivered.

2.A.2 Faculty Evaluation: Faculty are evaluated in a regular, systematic, substantive, and collegial manner based on clearly established criteria that reflect duties, responsibilities, and authority of their position.

2.A.3 Professional Development for Faculty: The program provides faculty with appropriate opportunities and support for professional growth and development to enhance their effectiveness in fulfilling their roles, duties, and responsibilities.

2.B.1 Qualification and Sufficiency of Staff: Consistent with its mission, intended outcomes, services, and characteristics, the program employs a sufficient number of qualified administrative leadership and other personnel to achieve its educational objectives, assure the integrity and continuity of its programs and services, wherever offered and however delivered, and maintain its support and operations functions.

2.B.2 Staff Evaluation: Administrative leadership and other personnel are evaluated in a regular, systematic, substantive, and collegial manner based on clearly established criteria that reflect duties, responsibilities, and authority of the position.

2.B.3 Professional Development for Staff: The program provides administrative leadership and other personnel with appropriate opportunities and support for professional growth and development to enhance their effectiveness in fulfilling their roles, duties, and responsibilities.

2.C.1 Financial Stability: The program demonstrates financial stability.

2.C.2 Resource Planning and Development: Resource planning and development include realistic budgeting, enrollment management, and responsible projections of grants, donations, and other non-tuition revenue sources.

2.C.3 Physical Infrastructure: Consistent with its mission, intended outcomes, and characteristics, the program's physical facilities and equipment are accessible, safe, secure, and sufficient in quantity and quality to ensure healthful learning and working environments.

2.C.4 Technological Infrastructure: Consistent with its mission and characteristics, the program has appropriate and adequate technology systems and infrastructure to support its management and operational functions, and it's academic and support services, wherever offered and however delivered.

2.D.1 Student Development: Students receive effective and sufficient support and opportunities beyond the classroom in an effort to facilitate their academic success and to enhance their overall development.

2.E.1 Program Work Environment: The program has a positive and stimulating work environment in which mutual respect, shared responsibility, and equitable problem solving are demonstrated and differences are utilized as strengths for advancing the program.

2.E.2 Program Contribution and Reputation: The program shares responsibility at the university level, is engaged with the community outside the institution, and is reputed to be functional, contributing, and talented.

3.A.1 Admissions and Graduation Requirements: Admission and graduation requirements are clearly defined and widely published.

3.A.2 Curriculum Content: The program provides a curriculum with appropriate content and rigor and consistent with its learning outcomes.

3.A.3 Curriculum Coherence: The curriculum demonstrates a coherent design with appropriate breadth, depth, sequencing of courses, and synthesis of learning.

3.B.1 Course and Program Learning Outcomes: Academic programs identify and publish expected course and program student learning outcomes that are clearly stated.

3.B.2 Alignment with Institutional Learning Outcomes: The course and program learning outcomes are aligned with the institutional student learning outcomes.

3.C.1 Assessment of Outcomes: The program documents, through an effective, regular, and comprehensive system of assessment, achievement of its intended outcomes and the students who complete its educational courses, programs, and degrees, wherever offered and however delivered, achieve identified course and program learning outcomes.

3.C.2 Assessment of Internal and External Environment: The program regularly monitors its internal and external environments to determine how and to what degree changing circumstances may impact its mission and its ability to fulfill that mission.

3.C.3 Assessment of Assessment Processes: The program regularly reviews its assessment processes to ensure they appraise authentic achievements and yield meaningful results that lead to improvement.

3.C.4 Dissemination of Assessment Results: The program disseminates assessment results and conclusions concerning mission fulfillment to appropriate constituencies.

3.C.5 Use of Assessment Results: The program uses the results of its assessment to inform its planning and practices that lead to enhancement of the achievement of intended outcomes including student learning achievements.

The feedback from APRC on the above items is to identify and define a list of Strengths, Weaknesses, and Recommendations for the programs.

5 Report Standards meeting the JCSEE Standards

The items within section 3.C of the program review relates to the assessment of the program. Based on JCSEE standard, the question is, does the evaluation of the assessment process in the programs by APRC, meets the JCSEE standards? We investigate each of JCSEE Utility standards [14] against policy standard.

Utility Standard:

Utility standards focus primarily on the qualities that prepare stakeholders to use the processes, descriptions, findings, judgments, and recommendations in ways that best serve their needs. Within Utility standard, the questions to answer would be

Does our program make a unique contribution to the community/agency/organization?

1.A.1 - Mission Statement
2.E.2 - Program Contribution and Reputation

To what extent is the program meeting its stated goals?

1.A.1 - Mission Statement
1.A.2 - Definition of Mission Fulfillment

How do we think our program should be working?

1.A.1 - Mission Statement
2.D.1 -Student Development
2.E.1 - Program Work Environment

What are the discrepancies between the intended program and the program-in-action?

3.C.1 - Assessment of Outcomes

Are there better ways to do what we're doing?

2.A.3 - Professional Development for Faculty
2.B.3 - Professional Development for Staff
2.D.1 - Student Development
2.E.1 - Program Work Environment
3.C.5 - Use of Assessment Results

What should be the future direction of our program?

1.A.1 - Mission Statement
1.A.2 - Definition of Mission Fulfillment

Are we doing a good job of reaching and servicing our potential user groups?

2.A.2 - Faculty Evaluation
2.B.2 - Staff Evaluation
2.D.1 - Student Development
3.C.1 - Assessment of Outcomes
3.C.2 - Assessment of Internal and External Environment
3.C.3 - Assessment of Assessment Processes

How can we adapt this program in light of budget cuts?

2.C.1 - Financial Stability
2.C.2 - Resource Planning and Development
2.C.3 - Physical Infrastructure
2.C.4 - Technological Infrastructure

Are we optimizing use of our human and fiscal resources?
2.A.1 - Qualification and Sufficiency of Faculty
2.B.1 - Qualification and Sufficiency of Staff
2.C.3 - Physical Infrastructure
2.C.4 - Technological Infrastructure

What does it mean to have a program that is adaptive and responsive?
2.C.3 - Physical Infrastructure
2.C.4 - Technological Infrastructure
2.E.1 - Program Work Environment

How can we do a better job of advocating for our program?
3.A.1 - Admissions and Graduation Requirements
3.A.2 - Curriculum Content
3.A.3 - Curriculum Coherence
3.B.1 - Course and Program Learning Outcomes
3.B.2 - Alignment with Institutional Learning Outcomes
2.A.3 - Professional Development for Faculty
2.B.3 - Professional Development for Staff
2.D.1 - Student Development
3.C.4 - Dissemination of Assessment Results
3.C.5 - Use of Assessment Results

Other standards are mostly based on the evaluation/assessment process. They may not be directly/indirectly associated or mentioned in the program report document.
The followings are the Feasibility, Propriety, Accuracy, and Evaluation Accountability standards:

Feasibility Standards

The feasibility standards are intended to increase evaluation effectiveness and efficiency. They are mostly based on the way the evaluation is done and may not directly/indirectly mentioned or documented.

F1 *Project Management: Evaluations should use effective project management strategies.*

F2 *Practical Procedures: Evaluation procedures should be practical and responsive to the way the program operates*.

F3 *Contextual Viability: Evaluations should recognize, monitor, and balance the cultural and political interests and needs of individuals and groups*.

F4 *Resource Use: Evaluations should use resources effectively and efficiently.*

Propriety Standards

The propriety standards support what is proper, fair, legal, right and just in evaluations.

P1 *Responsive and Inclusive Orientation Evaluations should be responsive to stakeholders and their communities*.

P2 *Formal Agreements Evaluation agreements should be negotiated to make obligations explicit and take into account the needs, expectations, and cultural contexts of clients and other stakeholders.*

P3 *Human Rights and Respect Evaluations should be designed and conducted to protect human and legal rights and maintain the dignity of participants and other stakeholders.*

P4 *Clarity and Fairness Evaluations should be understandable and fair in addressing stakeholder needs and purposes*.

P5 *Transparency and Disclosure Evaluations should provide complete descriptions of findings, limitations, and conclusions to all stakeholders, unless doing so would violate legal and propriety obligations*.

P6 *Conflicts of Interests Evaluations should openly and honestly identify and address real or perceived conflicts of interests that may compromise the evaluation*.

P7 *Fiscal Responsibility Evaluations should account for all expended resources and comply with sound fiscal procedures and processes*.

Accuracy Standards

The accuracy standards are intended to increase the dependability and truthfulness of evaluation representations, propositions, and findings, especially those that support interpretations and judgments about quality.

A1 *Justified Conclusions and Decisions Evaluation conclusions and decisions should be explicitly justified in the cultures and contexts where they have consequences.*

A2 *Valid Information Evaluation information should serve the intended purposes and support valid interpretations.*

A3 *Reliable Information Evaluation procedures should yield sufficiently dependable and consistent information for the intended uses*.

A4 *Explicit Program and Context Descriptions Evaluations should document programs and their contexts with appropriate detail and scope for the evaluation purposes*.

A5 *Information Management Evaluations should employ systematic information collection, review, verification, and storage methods*.

A6 *Sound Designs and Analyses Evaluations should employ technically adequate designs and analyses that are appropriate for the evaluation purposes*.

A7 *Explicit Evaluation Reasoning Evaluation reasoning leading from information and analyses to findings,* *interpretations, conclusions, and judgments should be clearly and completely documented*.

A8 *Communication and Reporting Evaluation communications should have adequate scope and guard against misconceptions, biases, distortions, and errors*.

Evaluation Accountability Standards

The evaluation accountability standards encourage adequate documentation of evaluations and a meta-evaluative perspective focused on improvement and accountability for evaluation processes and products.

E1 *Evaluation Documentation Evaluations should fully document their negotiated purposes and implemented designs, procedures, data, and outcomes*.

E2 *Internal Meta-evaluation Evaluators should use these and other applicable standards to examine the accountability of the evaluation design, procedures employed, information collected, and outcomes*.

E3 *External Meta-evaluation Program evaluation sponsors, clients, evaluators, and other stakeholders should encourage the conduct of external meta-evaluations using these and other applicable standards*.

6 Conclusions

The cyclic program evaluation based on the program report, contains the data as well as the general concepts and details of the program assessment. The program report is the only base of review and therefore evaluation of the program by APRC. The program report content is checked against recognized standard for evaluation by JCSEE. The program report maps to the JCSEE standards for evaluation based on utility but in order to meet other standards, the evaluators need to be part of the program.

7 References

[1] "TLL," [Online]. Available: http://tll.mit.edu/.

[2] "Wikipedia," [Online]. Available: https://en.wikipedia.org/wiki/Main_Page.

[3] L. Weber, The Legitimacy of Quality Assurance in Higher Education, Council of Europe Publishing, 2007.

[4] B. Howard. [Online]. Available: http://www.jcsee.org/.

[5] L. Suskie, "Assessing student learning," 2004. [Online]. Available: https://www.gvsu.edu/cms3/assets/.../evaluation-assessment.pdf.

[6] G. T. Crisp, "Integrative Assessment: Reframing Assessment Practice for Current and Future Learning," Assessment & Evaluation in Higher Education v37, pp. 33-43, 2012.

[7] "Carnegie Mellon University," [Online]. Available: https://www.cmu.edu/teaching/assessment/basics/formative-summative.html.

[8] M. Rodgers, "Improving Academic Program Assessment: A Mixed," Springer, pp. 384-395, 2013.

[9] J. Ory, "Meta-assessment: Evaluating assessment activities," Research in Higher Education, pp. 467-481, 1992.

[10] K. H. F. &. M. R. Good, "The Surprisingly Useful Practice of Meta-Assessment," 5 11 2013. [Online]. Available: https://illinois.edu/blog/view/915/99344.

[11] "Normative vs Ipsative," [Online]. Available: http://orgcommit.com/blog/ipsative-vs-normative/.

[12] "meta-assessment-rubric-for-closing-the-loop," 6 2015. [Online]. Available: http://cssl.osu.edu/posts/documents/next-steps-meta-assessment-rubric-for-closing-the-loop.pdf.

[13] D. A. WALKER, "A MODEL FOR ASSESSING ASSESSMENT ACTIVITIES," College Student Journal, 1999.

[14] "TLL Teaching," [Online]. Available: http://tll.mit.edu/help/assessment-and-evaluation.

Internet Assessment Considerations for Business Instruction

C. Balch
Department of Business & Administration Northern Arizona University, Yuma Branch Campus, Arizona, USA

Abstract - *Assessment is an essential part of instruction. Without assessment of participant learning outcomes, there is little way to accurately measure the efficacy of the instruction and the level of participant understanding. This paper reviews methodologies and concerns regarding development of internet assessment tools.*

Keywords: Assessment, Methodology, Internet, Learning outcomes

1 Introduction

Careful targeting of learning outcomes is essential to any learning process. It is equally important for to assess whether learners have achieved learning outcomes. Businesses, colleges, and universities have vested interest in assessment efficacy.

To be professionally useful, assessment should be based on proven techniques and practices. Assessment planning key considerations include:

1. Significant criteria to assess,
2. Appropriate methodology for collection of data,
3. Useful analysis of the data collected, and
4. Appropriate response to the results.

While accurate data collection is fundamental to assessment, proper data collection methodology is often given little attention. Except by chance, inaccurate data provides results that have little or no validity or reliability. "Garbage In Garbage Out" (GIGO) is a term commonly used in the computer industry to remind us that, if the data being analyzed are incorrect, the result will likewise be incorrect. While no assessment design system can correct for improper sampling or poor item construction, it is essential to use an assessment methodology that allows for and supports best practices in presentation, sampling, data collection, and analysis.

1.1 Considerations in Internet Assessment Delivery

The Internet is particularly well suited for some types of assessment and therefore very attractive to many evaluators of learning outcomes. Watt [1] concluded "…if you haven't done Internet survey research -- you will." Given that the Internet is an increasingly valuable option in assessment, researchers need to carefully consider a number of issues related to Internet assessment design and delivery.

Methodology scholars have identified a number of issues that must be considered if Internet assessment is to be of high quality. These issues include:

1. Considerations in Internet Assessment delivery and integrity [2-6].
2. Developing ways of selecting nonbiased samples, insuring that participants perceive and respond to Assessments as expected[7],
3. Ensuring that participants complete Assessments [8], and
4. Reaching an appropriate sample [9-11]

Various researchers [9, 12-14] have noted that many current Internet assessments do not generally follow best design principles as established by research. Failure to follow best practices can result in faulty data collection and therefore lead to erroneous conclusions.

A related issue in Internet Assessment design is the fact that researchers occasionally desire to explore Assessment design possibilities unique to Internet/Computer delivered Assessments. These design possibilities create both new opportunities and new issues particularly in regards to participants understanding the interface..

Issues include differences in perception of assessment instruments [15, 16] Different browsers, operating systems, hardware specifications, and user selected options all change how assessments are perceived. It may

ISBN: 1-60132-435-9, CSREA Press ©

be inappropriate to aggregate data collected from what are arguably different instruments. [13]

Consistency of experience is of concern to the researcher as advertising content is not the same for all participants. For example, on pages that display advertisements, one participant might see an advertisement for clothing while another might see one for books. The direct or subliminal association with this advertising content can change participant answers to questions [17, 18].

Lack of user response is an issue [13, 19]. While it is often possible to assign a penalty such as a grade or certification for lack of an assessment response many users may never even see the request to participate in a learning assessment. Spam filters and lack inbox and/or web page review are two common reasons for lack of participants seeing the request to respond.

Criteria in evaluating assessment authoring system features should include:

1. Number of available item types,
2. Ability to export data,
3. Ability to scale to research needs, and
4. Compliance with best practices in web Assessment presentation.

Particular emphasis should be placed on the system's ability to present Assessment content in a format consistent with best Internet Assessment practices. For example, many "free" Web page Assessment systems place advertisements and/or other distracting graphics in the instruments they create and host. Free web systems also tend to have severe limitations in terms of the size of the Assessment they can accommodate and in the number of respondents allowed.

Table 1: Review of considerations in the comparison of Internet Assessment Delivery Methods by Delivery Type [20]

Consideration \ Type	Ease of Creation	Data Collection Options	Ease of Delivery
Email	Easy	Minimal	Very Easy
User downloaded applications	Difficult	Open	Can be difficult to install
Internet Assessment delivery	Moderate	Large	Very

Table 1 Continued

Consideration \ Type	Participant Acceptability	Ease of Data Collection	Required Participant Knowledge
Email	Very acceptable	Difficult	Little
User downloaded applications	Low	Low	High
Internet Assessment delivery	High	Can be high	Low

Table 2: Various Factors in Internet Assessment Design [21]

Consideration	Internet Based Assessments
Coverage	Varies from low to high depending on availability of audience contact information and audience access to the Internet.
Participation percentage	Low, usually around 50%.
Distribution Speed	Very high.
Return Cost to user	Minimal or no cost/effort (some users pay by units of connection time/data used).
Informed consent	Cannot be collected in traditional ways.
Incentive availability	Difficult to use with anonymous Assessments but otherwise similar to other modalities.
Failure to connect with intended participant	More frequent than with other modalities.
Labor needed to distribute	Low.
Labor needed to create from a defined list of times	High, but labor can be reduced through use of various Assessment creation tools.
Expertise to construct items	High. Internet based Assessments require tradition design skills and understanding of unique Internet Assessment issues.
Incremental distribution cost.	Extremely minimal unless incentives are used or there is a cost to discover participants.

1.2 Web Page Perception

A participant's initial perception or preattentive processing of the web instrument is important because it influences much of the user's later interaction with the web Assessment content. Somewhat like the old adage that first impressions are the most important, what participants first see of a web page strongly influences the rest of their experience with the Assessment.

Initial perception of web based assessment is different from the participants perception of paper based assessment where the entire content is exposed at once. With internet assessment it is difficult to predict what content the participants' will first experience as Internet page organization tends to be fluid and contain unique features like animation which may be distracting from other content or take longer to load than other content.

Initial perception theories of web based assessments tend to cluster around either perception of object types such as images and scanning paths that reflect the path and duration that participants' eyes follow on pages. In addition, unlike paper, computers are often owned by the participant. Ownership may have profound effects on participant perception of the assessment and how the interact with it..

According to Koshman [22], participants perceive some parts of a website before others through a process called preattentive processing. Preattentive processing occurs quickly -- often in less than a quarter second. For example, for a given location on the page, animation tends to draw the participants' attention before static images, which are in turn viewed before text. Josephson and Holmes [23] noted that what is observed first on web pages depends on a variety of factors, including page content loading order and type of content. When there is a delay of more that a quarter second between the loading of different content areas such as text and images, content that is loaded first is observed first despite how the content might otherwise be observed with a completely loaded page.

Researchers have noted that preattentive processing patterns are evolving as sophisticated users alter their scanning patterns to avoid advertisements that are now typically placed in areas that research once found were getting first viewing. Even animated images in locations once thought to be ideal are now often ignored by persons who have been trained to avoid content that is not of interest or content that does not generally contribute to the content they wish to explore [23, 24] The top and right portions of the screen are now the most ignored web page areas. This should be of particular interest to researchers who use web based assessments as they should avoid placing graphics and instructions in those areas.

Research has found that web pages tend to be viewed in a variety of different patterns or Scanpaths. Participants from a Western culture tend to start reading at the upper left quadrant of printed pages while they tend to ignore instructions and response options on the right side or margin of printed pages. Web page participants tend to pay increased attention to content at the top or the bottom of pages. I find the bias towards the bottom of web pages paradoxical because the fluid nature of web page content makes it difficult for the researcher and/or web designer to control what content will appear at the bottom of web pages.

Internet Assessments present unique challenges in Assessment design for grouping of items. For example, it is possible to use objects such as drop down menus, which because they are animated and require user interaction to be perceived, are not possible with traditional paper Assessment instruments. Web based Assessments also lend themselves to the effective use of color. In addition, dynamic changes in web page content may occur both through programming and through participant changes to their computer environment such as when the participant changes the font size or the size of web pages.

As shown in Table 3, the fluid nature of screen level grouping is of particular concern to a web designer as participants' computers employ a variety of different screen sizes and resolutions (TheCounter.com, 2006). As only one screen of information can be shown at a time, all Assessments are automatically grouped at the screen level. Unfortunately this screen level grouping can change from participant to participant.

Table 3: Most popular screen resolutions of users Worldwide

Screen Resolution	Percentage
1024 x 768	56.15%
1280 x 1024	15.79%
800 x 600	12.04%
1280 x 800	4.09%
1152 x 864	3.90%

Loss of grouping due to low screen density might show just the following	Grouping interaction due to high screen density
Q2) What are your favorite months? ❑ January ❑ February ❑ March ❑ April ❑ May ❑ June	Q1) In what month does spring semester start? ○ January ○ February Q2) What are your favorite months? ❑ January ❑ February ❑ March ❑ April ❑ May ❑ June ❑ July ❑ August

Figure 1: Comparison of screen shots showing more or less grouping than desired

Note: In the left panel response options are not perceived it their entirety as the rest of the months do not fit.

Note: In this panel, there is interaction between the two questions on the screen.

1.3 Instructions

It is the obligation of the researcher to provide instructions that reduce the cognitive effort of participants so as to provide a clear and unambiguous experience with the Assessment instrument. Participant confusion results in bad data and bad data results in wrong conclusions. Since the layout of internet Assessment pages can change depending on the participants' computer environments and since some participants may not be willing to invest significant effort in the Assessment, special care must be taken to provide concise and clear instructions. There are a number of concerns and potential conflicts in maintaining participant focus and understanding as participants interact with Assessments.

Instructional clarity is essential to proper participant interaction. Unfortunately, greater instructional clarity can result in lengthier instructions. Assessments that take longer for the participant to complete are in conflict with a Assessment design objective of minimizing the amount of time a participant must spend taking an Assessment [14]. Clear instructions and proper Assessment design will minimize misunderstanding on the part of participants, as well as minimize the time needed to complete an Assessment.

1.4 Handicap Issues

Access for visually challenged users is a another serious concern for web design. The Section 508 requirements of the US government developed for handicapped users are often cited as a foundation for generic or standard web page design. For example, Section 508 requirements depreciate and discourage the use of frame objects which break the page into sections with different sources. Framed sections are difficult for assisted reading devices to explore and frames additionally may require the viewer to enable programming languages, such as Javascript, for navigation.

Universal readability for the handicapped is most easily accomplished when the structure and function of the page is kept simple. It is suggested that not only the code for pages be kept to the least common denominator but that pages be presented in such a way that the rendering of page content is not significantly altered across a variety of browser environments.

Section 508 also encourages the use of well described images when images are used at all. In particular, graphical interfaces, such as mapped images, are very difficult for visually challenged users to navigate as they cannot perceive the image areas that correspond to the map links.

1.5 Pattern Responses

The term "pattern response" refers to the manner of response of some Assessment participants to questions based not on instructions but on their personal habit or preference as they first look at (view) then respond to an Assessment page. When the reasons for pattern responses are known, Assessment designers can attempt to reduce them and consider them in the data analysis phase.

There are a variety of pattern response effects that researchers should be aware of. Participants who feel compelled to complete an Assessment will often respond to certain locations regardless of the content [25]. Pattern response also extends to grouping of options within Assessment items.

The layout and grouping of response options can have significant impact on participant responses. Christian and Dillman [26] found that responses to items arranged in a linear format have a significant bias towards selection of the top options or choices. On the other hand, there is a bias towards responses to item options on the left of non-linear layouts.

The response area of least resistance or the response pattern of respondents may change from participant to participant with some participants finding it most convenient to answer at the top, middle, or bottom of the Assessment page without care for the content of their responses. While inappropriate response patterns can be detected and the resultant data discarded, instruments should be designed to avoid these issues. For instance, a student responding to a course evaluation might have a genuinely positive or negative opinion of the course. If all of the items are weighted in the same direction, the student's response could create a false positive for a pattern response. Alternatively changing the direction of the weighting would be more likely to reveal the students true feelings about the course.

2 Discussion/Conclusion

Assessment is an essential part of the learning process. With proper considerations internet assessment methodologies are cost efficient and effective in measuring participant learning. Conversely, internet assessment tools are effective at measuring the efficacy of a learning unit.

References

[1] S. Watt, C. Simpson, C. McKillop and V. Nunn, "Electronic course surveys: does automating feedback and reporting give better results? Electronic version]," *Assessment & Evaluation in Higher Education,* vol. 27, pp. 325-337, 2002.

[2] P. M. Gurney, E. Chambers, L. Grant, S. Shah and M. Sullivan, "The Internet: Marketing Researcher's Panacea or Pandora's Box?" vol. 4, pp. 27-45, Spring, 2004.

[3] Z. Simsek and J. F. Veiga, "A Primer on Internet Organizational Surveys," vol. 4, pp. 218, 2001.

[4] N. C. Schaeffer and S. Presser, "The science of asking questions," *Annual Review of Sociology,* vol. 29, pp. 65-88, 2003.

[5] D. J. Solomon, "Conducting Web-Based Surveys. ERIC Digest," 2001.

[6] R. C. Yeaworth, "Use of the Internet in Survey Research," *Journal of Professional Nursing,* vol. 17, pp. 187-193, 2001.

[7] J. Bartlett and M. Alexander, "Response rate, speed, and completeness: A comparison of Internet-based and mail surveys," *Behavior Research Methods, Intruments & Computers,* vol. 34, pp. 46-49, 2002.

[8] (). *Balch Internet Research and Analysis Tool*. Available: http://birat.net.

[9] M. P. Couper, M. W. Traugott and M. J. Lamias, "Web survey design and administration," vol. 65, pp. 230-253, Summer, 2001.

[10] D. Heerwegh and G. Loosveldt, "An evaluation of the effect of response formats on data quality in web surveys," *Soc. Sci. Comput. Rev.,* vol. 20, pp. 471-484, 2002.

[11] (Aug 5, 2003; Aug 5). *The Ever-shifting Internet Population: A new look at Internet access and the digital divide*. Available: http://www.netratings.com/downloads/NetView.pdf.

[12] M. Carbonaro, J. Bainbridge and B. Wolodko, "Using Internet Surveys To Gather Research Data from Teachers: Trials and Tribulations," *Australian Journal of Educational Technology,* vol. 18, pp. 275-292, 2002.

[13] B. Eaton, "The problems and pitfalls of `net survey methodologies," *Communications Week,* pp. 54, 1997.

[14] (19 July, 2000; 19 July). *"Top Ten Mistakes" Revisted*. Available: http://www.useit.com/alertbox/990502.html.

[15] D. Y. P. Leung and D. Kember, "Comparability of data gathered from evaluation questionnaires on paper and through the Internet," *Research in Higher Education,* vol. 46, pp. 571-591, 2005.

[16] (23 October, 2003; 23 October). *How Users Read on the Web*. Available: http://www.useit.com/alertbox/9710a.html.

[17] D. Dillman, R. Tortora and D. Bowker, "Principles for constructing web surveys: SESRC Technical Report 99-01," pp. 16, 1999.

[18] (25-Aug-05,; 25-Aug-05). *Factors that influence reading and comprehension in self-administered questionnaires*. Available: http://survey.sesrc.wsu.edu/dillman/papers.htm.

[19] A. Tse, "Comparing the response rate, response speed and response quality of two methods of sending questionnaires: Email versus mail," *Journal of Market Research Society,* vol. 40, pp. 353-361, 1998.

[20] C. Balch, "Balch Internet Research and Analysis Tool development case study," 2007.

[21] C. Balch and T. Doise, "Barriers to adoption of educational technology," in *Technology,* 2005, .

[22] S. Koshman, "Testing user interaction with a prototype visualization-based information retrieval system." *Journal of the American Society for Information Science & Technology,* vol. 56, pp. 824-11, 2005.

[23] Josephson, S., & Holmes, M. E., "Attention to repeated images on the World-Wide Web: Another look at Scanpath theory. ," *Behavior Research Methods, Instruments, & Computers,* vol. 34, pp. 539, 2002.

[24] (4 Oct, 2004; 4 Oct). *Point of view: Common knowledge: The purpose of general education Electronic version]*. Available: http://chronicle.com/prm/weekly/v51/i07/07b02001.htm.

[25] (Access Date). *Any opinions about Opinio web survey software*. Available: http://aoir.org.

[26] L. Christian and D. Dillman, "The influence of graphical and symbolic language manipulations on responses to self-administered questions," *Public Opin. Q.,* vol. 68, pp. 57-81, 2004.

SESSION

TEACHING METHODOLOGIES AND STRATEGIES

Chair(s)

TBA

ISBN: 1-60132-435-9, CSREA Press ©

Consolidating Windowing Transformations Discussion with Applications to Screen Savers Program in Computer Graphics

Hassan Farhat
Computer Science
University of Nebraska at Omaha, USA

Abstract - *Consider the screen saver output shown in Figure 1. The screen saver is an output of a program written in the OpenGL API. Missing from the output is the rotation animation of each scene within its own viewport. The scenes generated are based on parametric equations and the concept of turtle graphics. The program resulting in the screen saver output incorporates many of the special 2D windowing transformations found in the OpenGL fixed-function model pipeline.*

We propose a methodology of covering the needed important windowing transformations and mathematical derivation during the early part of a first course in computer graphics. The windowing transformations used are the affine transformation, a generalization of linear transformations. In the later part of the course, these same concepts can be applied using matrix algebra as a method of concatenating transformations. Covering these topics, and formulating the mathematical basis, early in a first course in computer graphics eliminates many of the ambiguities found in the transformations in the OpenGL pipeline. We include additional constraints not found in OpenGL to enhance the topics coverage. WebGL is a graphics API developed from a version of OpenGL used in mobile devices and is based on OpenGL ES (embedded systems). Hence, the methodology of coverage applies to WebGL as well.

While current computer graphics APIs offer a very powerful set of graphics tools, the underlying understanding of the mathematics used in constructing the APIs remain important. We feel consolidating the coverage of topics on windowing early in the course is also important. In particular the different windowing transformations found throughout the graphics pipeline.

Keywords: Computer Graphics, Affine Transformations, Linear Transformations, Viewport, Display Coordinates, World Coordinates.

1 Introduction

Many of the computer graphics textbooks used today employ OpenGL in the study of computer graphics algorithms [1-4]. More recently, WebGL is being used as a graphics API [5, 6]. Other graphics APIs used in recent texts include Windows Presentation Foundation (WPF) [7]. Earlier textbooks on computer graphics varied in the APIs used as there was no common acceptable standard. For example, during the early emergence of computer graphics, graphics commands were screen-based commands. APIs such as ReGIS (Remote Graphics Instruction Set) were simple hardware instructions given to special graphics terminals such as the VT240 (Video Terminal) introduced in the early 80s. ReGIS and the VT240 were products of Digital Equipment Corporation (DEC) a very popular company with computer designs such as the PDP 11 and the VAX [8]. Programs such as Pascal were written and stored on the VAX as a server. Within the programs, ReGIS hardware graphics instructions (assembly) were included. The instructions were placed in output statements such as write and writeln in Pascal. The instructions are then interpreted by a graphics processor. The processor is part of the VT240 terminal design, the client. The graphics instruction set referenced locations on the screen directly. These included position instructions such as p[x,y] and vector instructions such as v[x,y]. The instruction p[x,y] resulted in placing the cursor at location (x, y) on the screen and v[x, y] resulted in drawing a line from the current cursor coordinates to location [x,y].

In such a graphics API, all transformations were completed manually through matrix algebra manipulations and the concepts of modeling and world coordinates were not fully included in graphics APIs. During the late 70s attempts at standardizations were being studied. These attempts included introducing the Graphical Kernel System (GKS) as a 2D graphics API standard and the Programmer's Hierarchical Interactive Graphics System (PHIGS) introduced as a 3D graphics standard.

Early computer graphics texts were written using these APIs [9, 10]. Not until the introduction of OpenGL in the early 90s did the graphics standard become widely used. OpenGL started by Silicon Graphics Incorporated (SGI) as a GL (Graphics Library) running on special IRIS workstations designed by SGI. Today OpenGL is a very popular graphics API standard with many versions being introduced regularly. These include OpenGL ES (Embeded System). The popularity of OpenGL continues with new variations of the

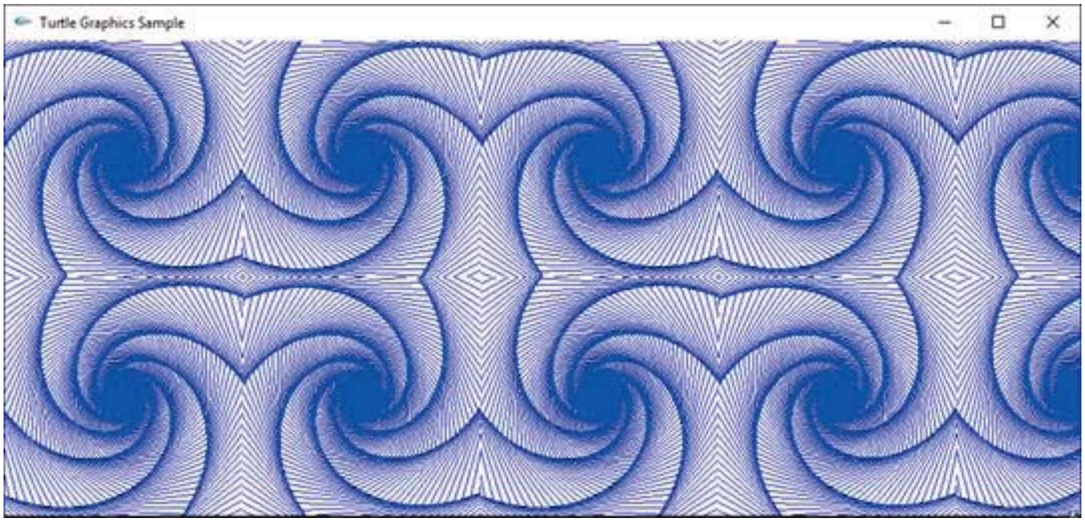


Figure 1

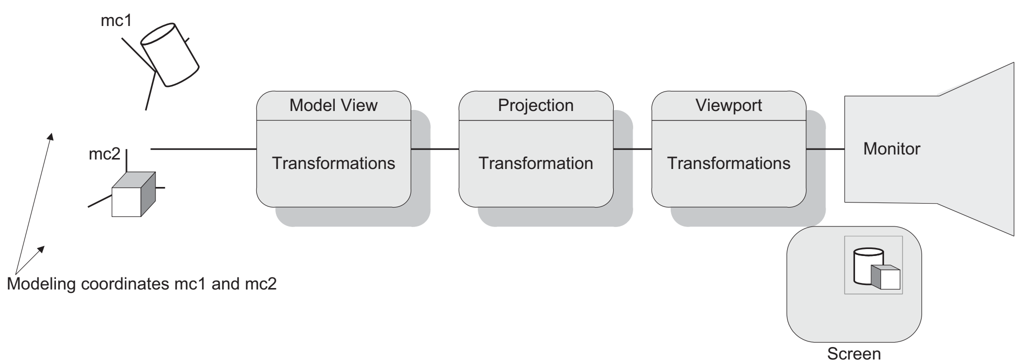


Figure 2: Simplified graphics pipeline

API still being developed. This is the case for WebGL, a graphics API based on OpenGL ES.

Consider Figure 2 of a simplified OpenGL fixed-function pipeline. As can be seen from the figure, object descriptions are not given in screen coordinates. An object description will undergo many transformations before it is rendered on the monitor screen.

The pipeline includes many important preprocessing steps needed prior to rendering the scene on the monitor screen. We feel an early introduction to a subset of these steps in a unified fashion serves in: a) providing a more solid understanding of the overall functionality of the pipeline, b) consolidates the topics, and c) removes many ambiguities in dispersed coverage. The windowing subset is an important subset as it takes a scene defined in many coordinate systems through the pipeline to eventually render the scene in screen coordinates.

We propose a teaching methodology that starts coverage of windowing early in an introductory course on computer graphics using the OpenGL API. This coverage can be extended to WebGL as well. While today's graphics APIs provide the tools to render scenes without the need to study the mathematical foundations, the topic is important for thorough understanding of the APIs [11, 12].

The early coverage on windowing transformations can start with simple aspects of linear mappings. Later in the course, when matrix algebra is introduced, these concepts have natural extensions to studying windowing under this algebra, thus providing more solid understanding of these aspects of the pipeline. In expanding the discussion, additional functionality not found in graphics APIs can be added in the discussion. Further understanding of these concepts can be achieved using the screen saver program. The program makes extensive use of these transformations to achieve the visual effects. From a user perspective, rendering in screen coordinates as is done in ReGIS seems easier and more natural. The discussion is intended to simplify the transformation task by computing the transformations needed to address screen pixels directly.

The paper is organized as follows. In section 2, we present the different OpenGL windowing commands to motivate the discussion on transformations. In section 3, we present simple affine transformations and see how the same concepts can be used in the derivations of the many windowing transformations in the pipeline. Section 4 extends these concepts to matrix algebra with composite transformations discussed. In section 5 we look at the applications of these concepts to the screen saver program. The conclusions are given in section 6.

2 The Different Transformations

The paper discusses transformations in the context of the OpenGL library. OpenGL is composed of two graphics libraries: graphics library (gl or GL) and graphics utility library (glu or GLU). The gl library is the core OpenGL library while the glu is an extension to gl written with functions from gl. This is similar to writing macros in assembly. To distinguish between the two libraries function calls names have prefixes gl and glu.

Transformations are applied to objects within the same coordinate system or to transform objects to another coordinate system. Figure 2 shows a simplified view of the different transformation phases in OpenGL. An object in modeling coordinates is transformed to world coordinates, also called object coordinates. In world coordinates, another coordinate system is defined, the viewing coordinates, based on the concept of synthetic camera. These coordinates define the orientation and position of a camera in world. The coordinates of objects in world coordinates are then computed in the viewing coordinate system. This is accomplished using the Model View transformation stage in OpenGL.

In viewing coordinates a view volume is defined in this coordinate system. The view volume can be a frustum or a right parallelepiped. This is accomplished in the projection phase of OpenGL. In this phase, the view volume is projected to a projection plane. In OpenGL, we characterize the stage using glMatrixMode(GLenum mode) where mode is an enumerated constant equal to GL_MODELVIEW or GL_PROJECTION.

The paper is about special two-dimensional (2D) rendering transformations where the view volume is a right parallelepiped and the projection used is orthographic. Here projection lines are perpendicular to the projection plane and the viewing coordinate's origin is equal to the world coordinates origin. We can apply this work to 2D world coordinates as OpenGL defaults assume, with the exception of the Z direction, the viewing coordinates x and y axis coincide with the world coordinates x and y axis . This simplifies the discussion of the different windowing transformations as the viewing coordinates (x, y) are read directly from the world coordinates (x, y).

Where to place the projection plane on the screen is not part of the OpenGL API as OpenGL does not have window or input/output commands. To deal with interactive input and window rendering commands, we use the GLUT (OpenGL Utility Toolkit) library in windows. The library includes commands that specify where to display a scene relative to screen coordinates of a monitor. Since rasterization starts at the top-left corner of the screen, the origin is chosen to be the top-left corner of the screen. In these coordinates y increases downwards and x increases from left to right. Where to display the scene is the display coordinates.

OpenGL does have a function, glViewport. The glViewport defines a mapping from normalized device coordinates (OpenGL normalized display) to a viewport. When combined with GLUT, glViewport(GLint x, GLint y, GLint w, GLint h) specifies (x, y) as coordinates of the lower leftcorner with respect to the display window (the display origin is at lower leftcorner of display with x increasing to the right and y increasing upwards), w is the width of the viewport in pixels and h is its height in pixel as well.

As can be concluded, the OpenGL pipeline undergoes many windowing transformations that are transparent to the user. We look at the different OpenGL commands for a simplified pipeline for the 2D special case. When referring to Figure 2, only parts of the world scene is eventually displayed on the screen within the display window. Which parts to render for the 2D case is given in the function call gluOrtho2D(left, right, bottom, top). The function is a special case of 3D clipping volume in gl. Objects outside of the clipping area or volume are clipped and not rendered. For the 2D case, the area is a rectangle. The coordinates (left, bottom) and (right, top) are of a rectangle's lower leftand upper right corners. The defaults are (–1. –1) and (1, 1).

The function call glViewport (x, y, w, h) defines where the world object (between (left, bottom) and (right, top)) is to be rendered within the display area. The default viewport area is the entire display area. The display area (a rectangle) is defined using glut functions. If the display area is not specified, the default is a window of size 300 pixels by 300 pixels along x and y. The upper left corner of the display is the upper right corner of the screen. In display coordinates, the origin of the display is assumed to be at the lower leftcorner of the display. To generate a display on the screen, we use three glut functions, glutInitDisplayMode, glutInitWindowSize(width, height) and glutInitWindowPosition(x, y).

We next look at the different needed transformations to accomplish the process of rendering a point on the screen.

3 The Needed Transformations as Affine Transformations

We start by assuming a default viewport, where the world object is mapped to the entire display area. The mappings used in the derivation are affine transformations. With affine transformations lines are mapped to lines. To map circles to circles, the display aspect ratios (height/width) are set so as to be equal to the aspect ratios of the world clipping area.

The OpenGL and glut display functions result in mapping the lower leftcorner of the world clipping area to the lower leftcorner of the display. Similarly for the upper right corners. Assume the world clipping area has the coordinates (X_{world_min}, Y_{world_min}) and (X_{world_max}, Y_{world_max}) instead of (left, bottom) and (right, top). Assume the display rectangular lower leftand upper right coordinates are ($X_{display\_min}$, $Y_{display\_min}$) and ($X_{display\_max}$, $Y_{display\_max}$). Given the coordinates of a point in world (X_w, Y_w) we need the transformations that result in the display coordinates, (X_d, Y_d). The affine transformations to accomplish this are

$$X_d = aX_w + b \qquad Y_d = cX_w + d \tag{1}$$

The parameter a, b, c, and d are derived by simple substitutions as

$$\begin{cases} X_{display\_\min} = aX_{world\_\min} + b \\ X_{display\_\max} = aX_{world\_\max} + b \end{cases} \tag{2}$$

Hence $S_x = a = \dfrac{X_{display\_\max} - X_{display\_\min}}{X_{world\_\max} - X_{world\_\min}}$

In equation (2), S_x, is a scaling factor along the x-axis. Similarly we can compute the scaling factor along the y-axis, S_y, the c parameter. S_y is obtained from S_x in equation (2) by replacing the variable X by Y, a by c and b by d. The computations of b and d are derived from $S_x X_{world\_min} + b = X_{display\_min}$, and similarly for d, with the X replaced by Y, S_x by S_y and b by d. This yields:

$$\begin{gathered} \left(\frac{X_d - X_{display\_\min}}{X_w - X_{world\_\min}}\right) = \left(\frac{X_{display\_\max} - X_{display\_\min}}{X_{world\_\max} - X_{world\_\min}}\right), \\ \left(\frac{Y_d - Y_{display\_\min}}{Y_w - Y_{world\_\min}}\right) = \left(\frac{Y_{display\_\max} - Y_{display\_\min}}{Y_{world\_\max} - Y_{world\_\min}}\right) \\ \left(\frac{X_d - X_{display\_\min}}{X_w - X_{world\_\min}}\right) = \left(\frac{w_{display}}{w_{world}}\right), \\ \left(\frac{Y_d - Y_{display\_\min}}{Y_w - Y_{world\_\min}}\right) = \left(\frac{h_{display}}{h_{world}}\right) \end{gathered} \tag{3}$$

Here, it can be emphasized that equations (1), (2) and (3) form the basic set of window transformations equations that can be applied for transformations between the four spaces: world, screen, display and viewport. The computation of the parameters a, b, c and d follows similarly by starting with proper initial values supplied by the OpenGL functions.

4 Alternative Derivations of the Needed Transformations

In the later part of the course, discussion on transformations can be carried in matrix algebra, thus emphasizing the concepts. We expand the discussion and follow the steps using matrix algebra to compute the screen coordinates of a point given in world. These computations work when OpenGL uses the initial defaults for viewing transformation where the x and y in world coordinates are equal to the x and y in viewing coordinates. Discussing transformation initially at this level, removes many of the ambiguities by a detailed computation of a user-defined point through the OpenGL pipeline to eventually render on the screen.

The initially user-defined point in world, (Xw, Yw), is first mapped to viewport coordinates (Xv, Yv), second, the viewport coordinates are computed in display coordinates as (Xd, Yd). The display coordinates are then computed in screen coordinates (Xs, Ys) as the actual pixel coordinates on the screen.

Using homogeneous coordinates, each 2D point (X, Y) is represented as 3D point with the Z value set to 1, (X, Y, 1). The transformation that maps a point in world coordinates (Xw, Yw, 1) to a point in viewing coordinates (Xv, Yv, 1) is derived similarly to section 3 but using matrices.

A call to glViewport includes the width of the viewport, w_v, the height, h_v, and the coordinates of the lower left corner ($X_{viewport\_min}$, $Y_{viewport\_min}$). OpenGL maps the lower left corner of world to the lower left corner of viewport, assumed as (0, 0) in viewport coordinates. Equation (3) can be used to compute the viewport coordinates with the display subscripts replaced with viewport coordinates. This results in the matrix representation

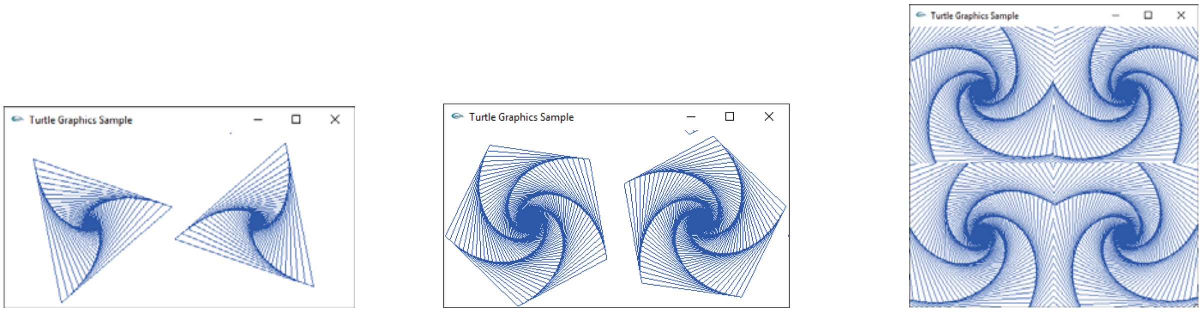
$$\begin{pmatrix} X_v \\ Y_v \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{w_{viewport}}{w_{world}} & 0 & -\frac{w_{viewport}}{w_{world}} X_{world\_\min} \\ 0 & \frac{h_{viewport}}{h_{world}} & -\frac{h_{viewport}}{h_{world}} Y_{world\_\min} \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X_w \\ Y_w \\ 1 \end{pmatrix} \tag{4}$$

From (Xv, Yv), the computation of the point in display, (Xd, Yd), is computed by simple translation. In matrix form

$$\begin{pmatrix} X_d \\ Y_d \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & X_{viewport\_\min} \\ 0 & 1 & Y_{viewport\_\min} \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X_v \\ Y_v \\ 1 \end{pmatrix} \tag{5}$$

Similarly, we can compute the screen coordinates. Here, however, the screen coordinates have the origin in the upper left corner while the display coordinates have the origin in the lower left corner. Hence, a translation is performed in X, with reflect and translate performed in Y. In matrix form

$$\begin{pmatrix} X_s \\ Y_s \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & X_{display\_\min} \\ 0 & -1 & h_{display} + Y_{display\_\max} \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X_d \\ Y_d \\ 1 \end{pmatrix} \tag{6}$$



(a) (b) (c)

Figure 3: Sample viewports displays based on 3-gon and 5-gon convex geometry, part (c) includes zoom-in.

In the equation, the coordinates ($X_{display\_min}$, $Y_{display\_max}$) is provided in the glut function call glutInitWindowPosition(X, Y) where (X, Y) is the coordinates of the upper right corner of the display given in screen coordinates. The display origin, however, is assumed as the lower leftcorner. Hence the Y in the glut function call is the maximum Y value in the display. As to the height of the display, $h_{display}$, it is given as part of the glut function call glutInitWindowSize(width, height) with height equal $h_{display}$.

5. The Screen-Saver Program

An interesting application of the computations given above is the screen-saver program shown in Figure 1. The program tiles the display with viewports. The viewports are constructed from an initial set of n points forming n-sided convex regular polygon, convex n-gon. The user enters the number of points, n, interactively. The program then computes the points as points on a unit circle with the points forming n equal sectors and forms the initial n-gon.

Following this, parametric equations are used to iteratively compute a set of n new points and display another n-gon. This processes is then repeated until the desired shape is obtained. Within each viewport, the scene is rotated. Mouse clicks are used to zoom-in (left-mouse click) or zoom-out (right-mouse click), creating interesting views.

Figure 3, shows screen captures for n = 3 and n = 5. The display was adjusted to include two viewports as shown in the figure. Part (c) shows the animation of part (b) with zoom-in applied using mouse-clicks.

6 Conclusion

Early computer graphics instructions were screen-based instructions. Today graphics instructions work mostly in different domains such as modeling and world domains. In this paper we proposed a methodology of covering the needed important windowing transformations and mathematical derivation during the early part of a first course in computer graphics. The coverage can be introduced early in a computer graphics course. The details of the mathematics needed to compute the different coordinates in the many coordinate systems and ending in screen coordinates is presented using simple transformations. This removes many of the ambiguities in the rendering steps. The windowing transformations used are the affine transformation, a generalization of linear transformations. In the later part of the course, these same concepts can be applied using matrix algebra as a method of concatenating transformations. In the discussion we used the different windowing transformation in the OpenGL and its extensions, gl, glu and glut.

References

[1] F. Hill, S. Kelly, “Computer Graphics using OpenGL, 3rd edition”, Prentice Hall, 2007.

[2] E. Angel, E. Shereiner, “Interactive Computer Graphics, A Top-Down Approach With Shader-Based OpenGL, 6th edition”, Addison Wesley, 2012.

[3] E. Angel, “OpenGL, A Primer, 3rd edition”, Addison Wesley, 2008.

[4] D. Hearn, M. Baker, W. Carithers, “Computer Graphics with OpenGL, 4th edition”, Prentice Hall, 2011.

[5] K. Matsuda, R. Lea, “WebGL Programming Guide, Interactive 3D Grpahics Programming with WebGL” Addison Wesley, 2015.

[6] E. Angel, E. Shereiner., “Interactive Computer Graphics, A Top-Down Approach With WebGLL, 7th edition”, Pearson, 2015.

[7] J. Hughes, A. Van Dam, M. McGuire, D. Sklar, J. Foley, S. Feiner, K. Akeley, “Computer Graphics Principles and Practice, 3rd Edition”, Addison Wesley, 2014.

[8] Digital Equipment Corporation, “GIGI/ReGIS Handbook”, Digital Equipment Corporation, 1980.

[9] J. Foley, A. Van Dam, S. Feiner, J. Hughes, "Computer Graphics Principles and Practice, 2nd edition", Addison Wesley, 1990.

[10] D. Hearn, M. Baker, "Computer Graphics, 2nd edition", Prentice Hall, 1994.

[11] D. Rogers, J. Adams, "Mathematical Elements for Computer Graphics, 2nd edition", McGraw Hill, 1990.

[12] D. Rogers, "Procedural Elements for Computer Graphics", McGraw Hill, 1998.

Teaching Security in Mobile Environments

Lila Ghemri [1], Shengli Yuan[2]
[1]Department of Computer Science, Texas Southern University, Houston, TX, 77004
[2] Department of Computer Science, University of Houston Downtown, Houston, TX, 77002

Abstract - *Mobile devices are quickly becoming the prevailing mode of access to the internet, social and entertainment media for a majority of people, especially the younger generation. The growing use of mobile devices has also attracted the interest of hackers who have turned their attention to designing or reformatting malware and spyware to specifically infect mobile devices. In this paper, we argue for the necessity to educate students, as users and/or application developers, about mobile security and privacy. We present three educational modules which have been each designed to cover a specific aspect of the mobile environment: Infrastructure Security, Device Security and Application Privacy and Security. These modules are fully operational and offer students the required knowledge on the field of mobile security.*

Keywords: Mobile infrastructure, Mobile devices and application, Privacy, Security.

1 Introduction

A recent Pew report showed that about two thirds of Americans own a smartphone. Additionally, the report stated that for 19% of Americans, their phone was the main and only platform for accessing the internet [1]. Globally, the number of mobile phones users has increased by 41% from 2010 to 2015 and now amounts to 5.2 billion; that is about 73% of the world's population. Also, the use of mobile media and applications has increased by 76% in 2014, with 300 million applications downloaded in 2015. Besides utilizing their mobile devices for socializing, people are increasingly using them to accomplish financial transactions such as checking their bank account, online shopping, paying their bills, etc.

Numbers show that iOS related shopping sessions increased by 174% in 2014, while on the Android platform; the same sessions were up by 220% [2]. This situation leads to security and privacy risks, as personal and financial data are stored in and accessed through these devices. It is then incumbent on computer science instructors to inform their students, both as users or developers of mobile software about the growing security threats in the mobile environment.

2 Motivation

There has been a surge in malware design for mobile environments. Indeed, there has been a notable 136% increase in mobile adware (410,000 apps) between 2013 and 2014 [3]. Additionally, cyberattacks are becoming more sophisticated, providing access to personal information such as contacts, or financial information which can be used to launch phishing attacks [4]. Investment in technology to prevent security risks has not been accompanied with a similar growth in skilled labor within the field, so much so that 30% of organizations complain of lack of experts in the field of security analytics and mobile security [5].

The purpose of this paper is to present three teaching modules that aim at addressing this need by offering

material related to mobile security. The material is organized into independent modules that can readily be used. The modules have been designed so that they can be used off-the-shelf, without requiring much customization on the part of the instructor. We believe that the modular approach will encourage an easy adoption by instructors and provide students with the required background in the topic.
The rest of this paper is organized as follows, the next section will discuss the characteristics of modular teaching and how suited it is to our work. The following sections will introduce in details each of the module in terms of coverage and topics. We will then conclude.

3 Modular Teaching

One of the most challenging tasks for a computer professional is keeping up with the advances in the field. Currently two main trends are taking place in the computing area: *Big Data and Data Analytics* and *Wireless and Mobile Computing*. Each trend has its attending topics, such as Big Data platforms Analytical tools for *Big Data* and Mobile Application programming, Wireless networking for *Mobile Computing*. Instructors in the field of Computer Science (CS) are faced with the need to offer courses on these topics so as to ready students for a competitive job market and provide good employment prospects to them. However, it is also becoming apparent that the CS curriculum is already loaded with the "traditional" CS courses and can hardly accommodate more courses. Faced with this conundrum, one solution is to use modules as a teaching aid to support and complement relevant courses. Modules provide a framework in which new skills can be taught within a limited time. This approach is appropriate for teaching topics that are important but do not require a whole course devoted to them. Modular teaching also allows teachers whose expertise is in a topic to gain knowledge in a different but related topic. They have the added benefit of being able to fit into many subjects. For example a module in Mobile Applications Privacy and Security can fit into a security course or a mobile app development course or an ethics course, etc. Thus modules provide flexibility in planning teaching, since instructors can elect to pick one or more modules depending on their need and time availability. A successful teaching module has to be clearly defined, include instructional material together with learning activities, and means to assess learning through a varying range of questions and tests [6].
Figure 1 describes the desirable features of a teaching module.

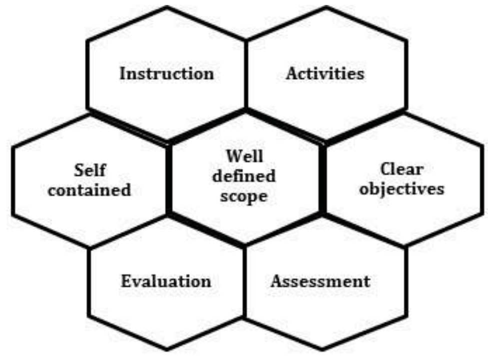


Figure 1: Module Desirable Features

4 The "*Enriching Security Curricula and Enhancing Awareness of Security in Computer Science and Beyond*" Project

The "*Enriching Security Curricula and Enhancing Awareness of Security in Computer Science and Beyond*" project is a collaboration between three Houston based universities: the University of Houston-Central (UH) as the lead institution, Texas Southern University (TSU) and the University of Houston-Downtown (UHD). The project includes eight sub projects covering a wide array of security and privacy topics and ranging in difficulty from material designed for non-CS students to content more appropriate for graduate CS students. Each sub project comprises several modules. Each module includes relevant teaching material, short questions and hand-on activities and can be offered as a standalone or as part of the topic. In this paper, we will focus on Privacy and Security in Mobile Environments.

4.1. Privacy and Security in Mobile Environments

The mobile environment landscape includes three main areas that we chose to focus on and for which modules have been designed.
The core issue from an IT security perspective is protecting the mobile infrastructure, protecting the mobile asset and protecting the user data from malicious code embedded in mobile applications. The three modules aim at addressing each of these security aspects (Figure 2).

a) Mobile Infrastructure Security,
b) Mobile Devices Security
c) Mobile Applications Privacy and Security

These three areas provide a comprehensive coverage of the mobile environment and afford the student a suitable working knowledge of the topic.

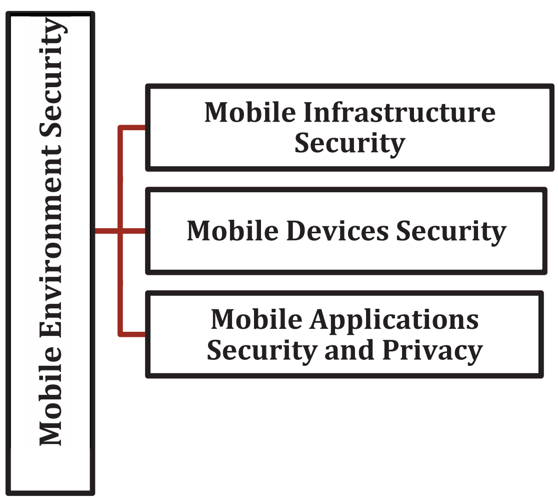


Figure 2: Mobile Environment Security Module Organization

In what follows, we present the details of each module.

4.1.1. Mobile Devices Security

In the Mobile Infrastructure Security module, we cover the security issues in two of the most widely deployed network infrastructures, i.e., the Cellular Networks and the Enterprise Wi-Fi Networks. These two types of networks support the majority of the wireless network traffic and have very significant impacts on business, economy and social lives of people [7], [9].

The objectives of the modules include:
1) Understand the security and privacy threats to the mobile infrastructure and,
2) Understand the basic strategies and approaches to enhance mobile infrastructure security and privacy.
The module requires 4 hours of lecture and 2 hours of hand-on learning activities.
The lectures focus on the following topics:

1. Carrier networks: Cellular networks
2. Enterprise networks: Wi-Fi networks.
3. Threats: Eavesdropping, interceptions, tracking, unauthorized access, malware [9], [10].

Suggested hands-on activities:
This instructional material is complemented by hands-on lab exercises. By performing these exercises, students will understand the risks attached to wireless communications, as well as the security and vulnerability of WEP and WPA by conducting a simple eavesdropping attack on an 802.11 Wi-Fi wireless network. Lab equipment includes 2 PCs with wireless NICs, one wireless router/access point and one laptop with wireless NIC running Aircrack-ng. Possible steps are:

1. Install and configure a 802.11 network on the 2 PCs and the Wi-Fi router/access point;
2. Set the security mode on the PCs and the router/access point to WEP;
3. Initiate and maintain constant traffic between the 2 PCs via the wireless router/access point;
4. Start Airecrack-ng on the laptop; Eavesdrop and try to break the traffic;
5. Set the security mode on the PCs and the router/access point to WPA;
6. Start Airecrack-ng on the laptop; Eavesdrop and try to break the traffic;

4.1.2. Mobile Devices Security

The purpose of this module is to teach students about the security of mobile devices. Mobile devices present great security challenges. As more people carry their mobile phones everywhere and use them to store data, some of which is critical, security of these devices is becoming vital. The mobility of the device makes it prone to being borrowed, lost or stolen, so much so that physical security is assumed to be compromised

The objectives of this module are for students to:

1) Understand the security and privacy threats to mobile devices,

2) Understand the basic strategies and approaches to enhance the device security such as configuration, user authentication and data encryption, and

3) Understand detection of security threats and mitigation approaches [11], [12].

The module requires four hours of lecture and three hours of hands-on activities. The topics covered in this module are:

1. Mobile devices features and characteristics: This section presents the characteristics of a mobile device, such as Wi-Fi and Bluetooth connections, GPS sensors, small screen, etc.
2. Secure local data storage: Authentication policies are presented together with their suitability to the mobile environment.
3. Safe browsing environments: The lack of display space on mobile devices can cause security issues when browsing the web. Mobile devices' heavy reliance on URL links makes it difficult to distinguish safe links from compromised ones.
4. Spyware, malware and phishing: Attacks that affect mobile applications such as mobile viruses, phishing attacks, cross-site Request Forgery (CSRF) are presented.
5. Security risks of mobile devices to "traditional" IT systems: Corporations can either provide employees with an approved company device that they manage or adopt a BYOD (bring your own device) policy. This section discusses these choices and the implications of each [13], [14].

Suggested hands-on activities:

Activity 1: Establishing a PIN for a device SIM Card and backing up the data on a mobile device.
Activity 2: Privacy and security of geolocation. Study the pros and cons of saving the user location data in a database versus storing the data in the device.
Activity 3: Define device rooting and study the pros and cons of jailbreaking a device.
Activity 4: Research mobile phone location apps. Find the procedure to remotely wipe data from a mobile phone.

4.1.3. Privacy and Security of Mobile Applications

The number of mobile applications has reached the millions, with more than one million and a half mobile applications on Google Play, closely followed by Apple Store. The chart below (Figure 3) shows the distribution of downloaded applications by platform [15]. This increase in the number of mobile apps has also been accompanied with an increase in interest for mobile app programming, so much so that a majority of CS programs include such courses in their curricula. The module of Privacy and Security of Mobile Applications aims at providing students, who have a background in programming mobile applications, with the necessary knowledge of the security threats that mobile applications present [16], [17], [18]. This module also introduces some design principles that programmers can apply to code more secure applications.

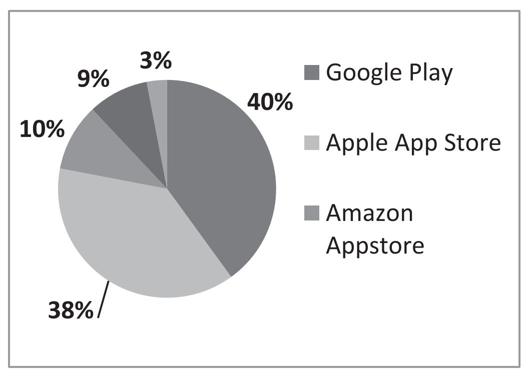


Figure 3: Distribution of Mobile Applications by Platform.

Mobile applications provide specific services that help organizations conduct business operations and rely on user data for authentication and verification of identity. These data are the target of cyber-attacks and it is important that students know about these threats. The topics addressed in this module are:

1. Security terms: Terms that are commonly used in security parlance such as vulnerability, exploit, zero-day attacks, black-hat hacker, and white-hat hacker and so on are introduced.

2. Coding Vulnerabilities: Bad programming techniques, such as unnecessary use of global variables, unbounded, non-initialized variables, pointer boundary, and others can be the source of serious security threats. The student is introduced to these techniques and offered more secure programming alternatives.

3. Google Android Security: This section introduces Google Android Mobile Java platform and its approach to security such as sandboxing, controlling access and permissions to resources.

4. iOS Security: This section introduces the Apple security framework based on mandatory app signing, security enclaves, etc.

5. Mobile HTML Security: Mobile HTML is used to code mobile versions of desktop websites. Security challenges, such as HTTP redirect and phishing are introduced [19].

6. Mobile Geolocation: Geolocation can present a threat to the privacy of the device holder. Best practices to mitigate this risk and how to manage resource permissions are presented [20].

Suggested hands-on activities:

In order to enable students to recognize the security risks related to mobile applications as well as how to mitigate them, **we** designed hands-on activities, such as managing an app security using the Manifest file. We also suggest using off the shelf, public domain**,** lab material that has already been designed and tested. One such repository is the Android Security Labware developed by Li Yang at the University of Tennessee in Chattanooga Information Security (InfoSec) Center [21] and PLab Information Assurance and Security Education on Portable Labs [22].

5 Conclusion

As mobile devices become more ubiquitous and their use more widespread, the various security challenges they bring about need to be addressed. Three modules were presented that can help instructors in the task of informing their students about mobile infrastructure security, mobile devices security and mobile applications privacy and security. Each module consists of instructional material; a related set multiple choice questions and hands-on learning activities. Currently these modules are being presented and tested for adoption and improvement in the three home institutions. The material is freely available, with more security related modules, at http://capex.cs.uh.edu and we invite all interested instructors to use them and provide the team with feedback.

6 Acknowledgements

This work was supported by NSF grant 1241772. Any opinions, findings, conclusions, or recommendations expressed herein are those of the authors and do not reflect the views of the National Science Foundation.

7 References

[1] A. Smith. 2015. U.S. Smartphone Use in 2015. Pew Research Center. Retrieved October 2015 from http://www.pewinternet.org/files/2015/03/PI_Smartphones_0401151.pdf

[2] S. Perez. 2015. App Usage Grew 76% In 2014, With Shopping Apps Leading the Way. Retrieved 1/2016 from http://techcrunch.com/2015/01/06/app-usage-grew-76-in-2014-with-shopping-apps-leading-the-way/

[3] G. Suarez-Tangil, J. E. Tapiador, P. Peris-Lopez, A. Ribagorda. 2014. "Evolution, Detection and Analysis of Malware in Smart Devices". *IEEE Communications Surveys & Tutorials*. 16, 2 (2014), 961-987

[4] A. Porter Felt, M. Finifter, E. Chin, S. Hanna and D. Wagner. 2011. A survey of mobile malware in the wild. In *Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices (SPSM'11).* ACM New York, NY, 3-14.

[5] B. Solis. 2015. Disruptive Technology Trends 2015-2016. Retrieved January 2016 from http://www.briansolis.com/2015/01/25-disruptive-technology-trends-2015-2016/

[6] K. Sejpal. 2013. Modular Method of Teaching. *International Journal for Research in Education*. 2, 2 (2013), 169-171.

[7] M.S. Gast. 2005. 802.11 Wireless Networks: The Definitive Guide. O'Reilly

[8] A. Gardezi. 2006. Security in Wireless Cellular Networks. Retrieved on 12/2015 from http://www.cse.wustl.edu/~jain/cse574-06/ftp/cellular_security/index.html

[9] N. Krawts. 2006. Introduction to Network Security. Charles River Media

[10] A.N. Bikos; N. Sklavos, 2013. "LTE/SAE Security Issues on 4G Wireless Networks, Security & Privacy", *IEEE Volume: 11, Issue: 2. 2013*, Page(s): 55 – 62.

[11] Isaca. 2010. Securing Mobile Devices, White Paper Report. Retrieved on 12/2015 from:

[12] http://www.isaca.org/KnowledgeCenter/Research/Documents/SecureMobileDevices_whp_Eng_0710.pdf

[13] P. Ruggiero, J. Foote. 2011. Cyber Threats to Mobile Phones. Retrieved July 2012 from http://www.uscert.gov/reading_room/cyber_threats_to_mobile_phones.pdf

[14] Xcubelabs. 2012. Mobile Device Management. Retrieved 3/2016 from http://www.xcubelabs.com/blog/mobile-device-management-enable-manage-and-secure-your-mobile-environment

[15] K. W. Miller, J. Voas, G. F. Hurlburt. 2012. BYOD: Security and privacy considerations. *IEEE IT Professional*. 5 (September 2012), 53-55:

[16] Statista, 2015 Retrieved 5/15/2016 from http://www.statista.com/topics/1002/mobile-app-usage/

[17] H. Dwivedi, C. Clark and D. Thiel. 2010. Mobile Application Security. McGraw-Hill Professional Publishing. New York, USA.

[18] S. Siddharth, P. Doshi. 2010. Five common Web applications vulnerabilities. Retrieved on June 2012 from http://www.symantec.com/connect/articles/five-common-web-application-vulnerabilities.

[19] G. Wassermann, Z. Su. 2008. Static detection of cross-site scripting vulnerabilities. In *Proceedings of 30th ACM/IEEE Conference on Software engineering (ICSE'08).* ACM New York, NY, 171-180.

[20] N. Doty and E. Wilde. 2010. Geolocation privacy and application platforms. In *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Security and Privacy in GIS and LBS (SPRINGL'10)*. ACM, New York, NY, 65-69.

[21] L. Yang. 2014. Capacity Building through Curriculum and Faculty Development on Mobile Security. Retrieved on May 2015 from http://www.utc.edu/faculty/li-yang/mobilesecurity.php

[22] D. Lo. 2015. Information Assurance and Security Education on Portable Labs. Retrieved November 2015 from https://sites.google.com/site/iasoncs/home

Kinesthetic Touches For a Theory of Computing Class

Judy Goldsmith
University of Kentucky
329 Rose St.
Lexington, KY 40509
goldsmit@cs.uky.edu

Radu Paul Mihail
Valdosta State Univeristy
1500 N Patterson St.
Valdosta, GA 31698
rpmihail@valdosta.edu

ABSTRACT

Theory of computing courses do not usually evoke memories of fun for most students. Instead, fragments of formalisms from automata theory and memories of the existence of rigorous proofs come to mind. In this paper we bring arguments from educational psychology and pedagogy that movement and fun can be a part of the teaching and learning process in theory of computing courses. Kinesthetic Learning Activities (KLAs) consist of physical interactions between the students and their environment in order to achieve or optimize an educational objective. We propose a set of KLAs and performances to assist teachers with topics that most students find hard to grasp. Our hope is to inspire CS educators to not only adopt and modify these KLAs, but to create new ones. Our greater hope is that words like "fun" and "games," along with more accurate understanding of important results in theoretical computer science, come to mind when students are asked to describe their theory course.

1. INTRODUCTION

This is a paper about real practices in real Theory of Computing classes that bring motion into the teaching and/or learning process. We argue that there are pedagogical reasons to involve motion in what has often been taught in the best formal mathematics tradition.

Our university offers theory of computing courses at the undergraduate and graduate level. The syllabi for both include regular languages, deterministic and nondeterministic automata, and their equivalence, as well as proofs that there are non-regular languages; context-free grammars and pushdown automata, and their equivalence; Turing machines and their variants, a brief introduction to decidability, and to computational complexity. Our undergraduate course is considered part of the discrete math sequence, and also includes a section on logic, at the cost of depth and rigor in the other topics. For the undergraduate course, student learning outcomes include:

- Fluency in the elements of automata theory
- Basic understanding of properties of formal languages and associated concepts including grammars and regular expressions
- Basic understanding of models of computation including Turing machines

The undergraduate course is required for computer science majors; the graduate course is a core course, and can be replaced by an algorithms course. Thus, the graduate course is usually taken by students who are more comfortable with mathematical formalisms. The undergraduate course is an oft-postponed challenge for many of our students.

1.1 Order Matters

The standard order for a traditional theory of computing course is based on the Chomsky hierarchy of formal languages. It starts with regular languages (and automata theory), then introduces context-free languages, with their corresponding grammars and pushdown automata. Some courses then cover context-sensitive languages, but many have traded that topic for more time for Turing machines and their model variants, computability, and then a brief introduction to computational complexity, ending with NP-completeness.

It is common to end the course with NP-completeness in the last week of classes, and the halting problem soon before that, although anyone who has taught for a few years, or has recently been a student, knows that those last few weeks are the time when students are least likely to absorb and understand hard concepts. In other words, computability and complexity often go flying past even the better students while those students are busy with major programming projects, emergency extra-credit assignments, and general end-of-semester panic.

We have taken to introducing the course with Turing machines. This has two major advantages. First, it eases them into the notion of formal models of computation with something that is "Turing complete," meaning that it can compute whatever their laptops/watches/supercomputer accounts can compute. Secondly, it introduces the hard parts of the course (undecidability and NP-completeness) before their brains are full from the rest of the semester.

One side effect of ending the course with CFLs and pushdown automata is that the students can make connections to courses on compiler design, parsers, and other basic computer science concepts, so they go out believing that this is a relevant course for the curriculum. A secondary effect is that the beliefs that they can master the (recent) material, and that it is relevant, seem to have a positive effect on teaching evaluations (at the end of the semester).

ISBN: 1-60132-435-9, CSREA Press ©

2. KINESTHETIC LEARNING

Bloom's taxonomy of educational objectives [13] remains an important tool to improve student learning through systematic analysis and research of learning goals and process. This taxonomy, and later revisions, e.g., [4], suggest that learning can be divided into three domains: cognitive, affective and psychomotor, or kinesthetic. Anderson [4] and others in the educational psychology community (e.g., [36, 20, 35, 23, 37]) encourage educators to consider all three domains, and include aspects thereof, when designing lessons.

Given our physical senses, we perceive information visually (e.g., white/black board, overhead projector with text, pictures, diagrams, videos), auditorily (sounds) and kinesthetically (taste, touch and smell). A growing body of literature suggests that as people mature, the majority specialize in one input modality (i.e., they become visual, auditory or kinesthetic learners), though some rely on multiple input modalities [17]. Teachers tend to focus their teaching methods on the input modality that works best for them [6]. Thus, those students whose primary modality is not in use in the classroom are at a real disadvantage.

Kinesthetic Learning Activities (KLAs) involve physical interaction between the students and their environment. Begel et al. [7] argue that research on kinesthetic learning in higher education is sparse, and even more sparse in engineering related fields. KLAs offer students a change to literally and physically participate in their own education, and thus have potential to energize an otherwise static group of learners. For the interested reader, Begel [7] puts forward a few guidelines on designing KLAs, and recommends that future research be directed toward assessing KLAs and building a set of best practices.

Mathematics provokes anxiety for a variety of students. One might think that teaching abstract, rigorous mathematical thinking, by its nature, must be dry of artistic expression. Karl Schaffer and Erik Stern [27] are determined to change that. The two men perform and choreograph where modern dance intersects mathematics. Combinatorics and set theory exhibit symmetries and structures that Shaffer and Stern bring to life using dance. For example, one of their routines involves attempting a hand-shake that fails numerous times, resulting in the dancers falling past each other. In a recursively defined set whose basis clause is the dancers, and the inductive clause joins two arbitrary elements using a hand-shake, the dance demonstrates the various possibilities and constraints of set building.

In their book [32], Shaffer and Stern propose dance activities and practical activities for teachers to use in the classroom. For example, distinguishing symmetries induced by reflection, rotation and translation is practiced using simple movements and relative positioning of a group leader and a group of followers.

A motivated teacher, excited to explore kinesthetic learning, faces several obstacles: students' reluctance to participate, students' failure to understand the point of the exercise, or the teacher's own failure from losing control of the activity [30]. Student reluctance to participate in general problem-solving, according to Felder and Brent [16], is in part due to a sudden withdrawal of a support structure given to students by teachers from the first grade on. Pollard et al. [30] argue that this problem can be circumvented by adding elements of kindergarten activities such as toys and play. The idea is simple: toys materialize abstract concepts and playing with the toys can help build internal representations of complex ideas or algorithms.[1]

Attempts at enriching computer science students' learning through kinesthetic enhancements has been done in the past. For example, Silviotti et al. [33] suggest KLAs for a distributed computing course, where algorithms are enacted by people and data structures built from people. The authors argue that these activities promote the understanding of concurrency through simultaneously active agents, and locality of scope through physical limits. The authors describe several distributed algorithms enacted kinesthetically.

Most of us have seen videos of folk-dance troupes performing sorting algorithms [1]. The site csteaching tips points out that YouTube abounds in videos of CS students doing similar exercises, and algorithms teachers can bring these dances to their classrooms [24].

Dr. Craig Tovey, at Georgia Tech, brings to life Dantzig's simplex method [14] for finding the optimal solution(s) to linear programs,[2] using students as columns of a matrix and rubber bands for the edges of a simplex (hypertetrahedron). He illustrates the concept of pivoting through an edge of the rubber band simplex that remains static as one of the simplex's vertices moves to a better choice. This lesson is publicly available on YouTube at: `https://www.youtube.com/watch?v=Ci1vBGn9yRc`.

Friss de Kereki [15] proposes and evaluates a set of KLAs for an object oriented (OO) flavor of CS1. The experimental design had two control groups and one intervention group. The author reports higher motivation, satisfaction and passing rates for the intervention group. Similarly, Adorjan et al. [2] propose improvements to an OO CS1 using Gardner's taxonomy of intelligence [19], dubbed multiple intelligence (MI). Adorjan et al. provide a list of activities suitable for CS1, along with a mapping to MIs. For Gardener's MI4 (bodily-kinesthetic intelligence), Adorjan proposes five activities.

The CS Unplugged movement, started by Mike Fellows, has accumulated a book [12] and website [22] with many activities suitable for outreach to kids as well as for the CS classroom. Much has been written about the use of these and similar activities, e.g., [8, 9, 10, 11, 18, 21, 26], and some about the efficacy thereof [34].

Peacock [29] looks at potential teacher learner style mismatches for English as a foreign-language instruction. He cites Reid's [31] hypothesis that claims such mismatches cause learning failure, frustration and demotivation. Pea-

[1] One of the authors, old enough to remember the New Math, remembers using wooden blocks to count in various bases — though perhaps that was after kindergarten.

[2] Not to be confused with Nelder and Mead's [25] unconstrained optimization method based on the same geometric simplex.

cock gives teachers several recommendations to accommodate a balanced style. For example, kinesthetic learners should be given problem-solving activities, role-play and drama, and encouraged to actively participate in the learning environment. He also notes that Chinese learners prefer kinesthetic and auditory learning styles. This finding is also supported by Park [28].

Ambudkar [3] presents a dance activity to support student learning of the interoperability and communication between the layers in the open systems interconnections (OSI) model. The dance team consisted of seven rows (corresponding to the seven layers of the OSI model) and two columns (corresponding to the sender and receiver). The dancer's movements indicated how data moved between layers and between the sender and receiver.

Anewalt [5] includes toys, games and the Alice programming environment to foster an active learning approach for CS0 students. Algorithmic thinking and accuracy were reinforced by asking students to write instructions for making peanut butter and jelly sandwiches. The instructor later unveiled tools and materials, then randomly paired students were asked to modify their algorithms to use only the tools and materials available. The students then precisely followed their algorithms to make the sandwiches. She describes other activities related to basic programming principles, such as cookie-cutter and cut-out shapes to illustrate difference between classes and objects, "fishing" the argument (miming the action of fetching the argument value) to assign a parameter during a function call and the data transfers therein, as well as others.

Theory of computing courses tend to be devoid of physical activity,[3] and are thus optimizable by adding relevant and purposeful dramatic, visual, and physical elements to cater to kinesthetic learners. Our contribution to this aspect of computer science education consists of a set of kinesthetic learning activities, which we hope will be adopted by and improved upon by others.

3. KINESTHETIZING THE THEORY OF COMPUTATION CLASS

For this section, we assume that the reader has seen a theory of computing class, though details may have been forgotten. While the statement of the pumping lemma is destined to be forgotten, we speculate that drawing smiley faces for accept states on finite automata and frowning faces for reject states might make basic finite automata definitions more memorable.

3.1 Power Set Construction

In many classes labeled "Theory of Computing" or "Finite Automata and Formal Languages," regular languages are introduced early on. They might be defined by deterministic or (equivalently) nondeterministic finite automata, regular expressions, and/or regular grammars. Eventually, most classes claim or prove that whichever of those models were introduced are equivalent. The proof that any nondeterministic finite automaton N has an equivalent deterministic finite automaton D is referred to as the power-set proof. The construction starts by defining the start state of D, s_0^D, to be the set containing the start state, s_0^N, of N and any states reachable from s_0^N by ϵ-transitions (where ϵ is the empty string). Then, for each state s^D in D corresponding to a set S of states of N, and each element σ of the alphabet Σ, we create a state (if it isn't already in D) consisting of the set of states reachable from any state in S via σ followed by ϵ-transitions. We label the transition by σ.

We then might prove that the two automata compute the same language. What we hope is that the students understand the construction, and we might ask them to perform it on a homework or exam.

Dr. Craig Tovey, at Georgia Tech, has his students enact the construction. First, they simulate a deterministic finite automaton. He writes each character of the string on a separate sheet of paper. Students holding one sheet each line up as the input string on one side of the room. Once a student's character is "read" the student throws the sheet into a trashcan and has to sit on the floor on the other side of the room. This conveys the one-time read process.

For the power set construction, he brings in cardboard squares with state numbers, and draws a nondeterministic automaton on the board. Students volunteer to "be" the states of the nondeterministic automaton.

Tovey then asks the start state to climb onto a table. For each symbol in the string, for each student on the table, the student points out those states that are reachable from himself via that symbol. The reachable states then take their place on the table and the others climb down.

"Does it get crowded on the table?" we asked.

"Yes."

"Isn't there a danger that the table will flip over?" we asked.

"That's what makes it so exciting," he said.

We have tried this exercise. to the amusement of the class. Tovey also simulates a pushdown automaton in class. It's the same except for an extra line of people to be the elements in the stack.

3.2 Proving a Language is not Regular

There are two approaches to showing that a language is not regular. The more commonly taught one is the Pumping Lemma, which says that a language L is regular iff there is an n [the size of the minimal deterministic finite automaton for L] such that for all strings $s \in L$ of length $> n$, there are x, y, z such that $|xy| \leq n$ and $|y| > 0$ such that for all $i \in N$, $xy^iz \in L$.

Got that? The quantifier sequence after the "iff" is

$$\exists n \forall s \exists x, y, z \forall i.$$

This is really hard for most students to grasp, much less

[3] Here we discount the vigorous writing of mathematical formalisms with chalk or dry-erase markers that some instructors do, and the physical acts of note-taking in which we wish students would engage.

apply. However, the underlying idea is a simple application of the pigeonhole principle: if we have a finite automaton with n states, and a string with more than n symbols, then when the automaton processes the string, at least one state must repeat. Therefore, the path from that state back to itself can be repeated arbitrarily many times (or excised, when $i = 0$) before the string leads to an accept state.

In order to cement the idea of a repeated loop, we ask all the students in the class to draw a circle in the air with their fingers. Since we have not yet tried the Power Set Construction exercise, this is usually our first class participation exercise. At first many are reluctant, but are willing to humor the crazy teacher.

The second approach to showing a language is not regular is via the Myhill-Nerode Theorem. This begins by defining an equivalence relation R_L for a language L. We say xR_Ly iff for all strings z, $[xz \in L \Leftrightarrow yz \in L]$. It is difficult to show this, but not so difficult to show that x and y are not equivalent. For instance, for the language $L = \{1^n0^n | \ n \in N\}$, consider $x = 1^i$ and $y = 1^j$ for $i \neq j$, and let $z = 0^i$. Then $xz \in L$ but $yz \notin L$.

The Myhill-Nerode Theorem states that L is regular iff R_L has finitely many equivalence classes. (These correspond to states in a minimal finite automaton that accepts L.) To show that L is not regular, one shows (as in the previous example) that there are infinitely many non-equivalent strings. In the previous paragraph, we have argued that for any two distinct natural numbers i and j, 1^i is not L-equivalent to 1^i.

Dr. Mike Fellows says that he explains the notion $[xz \in L \Leftrightarrow yz \in L]$ by starting his hands in different places (representing x and y), making the same series of moves (z), and ending up in the same place. We conjecture that having the students do this, and also have them end up in different places, might help make the equivalence relation more memorable, if not clearer.

3.3 Formal Grammars and Buffalo

It is well known that the word "buffalo" can be a noun, verb, and adjective. For instance, "Buffalo buffalo Buffalo buffalo" might be parsed as "Bovines harass other bovines from a city in upstate New York." Thus, one can illustrate a parse tree for English without having to resort to complex vocabulary. However, one might rightly be concerned about shorter sentences, since this verb is transitive: the subject of the sentence must have someone or something to buffalo. However, there is a tap dance sequence called a buffalo, because the rhythm of the taps matches that of the word. Thus, "Buffalo buffalo" is a valid sentence.

Teachers eager to learn the step are advised to consult YouTube or their local tap dance instructor. It is worth noting that (a) the move involves a shuffle step, and (b) it is often used in tap routines as an exit. Thus, one can shuffle off with buffalos.

3.4 Enacting Turing Machines

When we teach about Turing machines, we often start by writing a TM program to decide the language of palindromes over a small alphabet, say, $\{0, 1\}$. This is convenient because the language of palindromes is not regular, and also (as we later mention) can be recognized in time $\mathcal{O}(n^2)$ with a one-headed, one-tape TM, but in time $\mathcal{O}(n)$ by a two-tape or two-headed TM. We later show that there's a simulation of a 2-tape TM by a 1-tape TM with an $\mathcal{O}(n^2)$ time blowup; the language of palindromes is an example where we see the quadratic speedup directly.

In order to describe the TM, we act out the states: "I'm remembering I saw a 1. Is this a blank? No. Move right. I saw a 1. Is this a blank?...." (This is said while moving in the same direction, one step per "Move right". Ideally, that direction should be to the students' right, rather than the instructor's.)

We have found, over the years, that having students attribute such statements to states and state/symbol/action tuples cuts down on the instances of students randomly writing down states and transitions just to write something.

3.5 The Infinite Loop Dance

One of the central concepts in teaching about Turing machines (or your favorite sequential model of computation) is the notion of an infinite loop. For instance, to show that a decidable language L is semi-decidable, one can modify the Turing machine that decides L by replacing every "No" output by an infinite loop. But if a professor asks students, "can you write an infinite loop," a surprising number of them refuse to admit that they can — meaning that they won't admit that they *have* done so.

The two simplest Turing machine infinite loops are the one-state loop that ignores the input and moves the read head right, and the one that ignores the input and moves right, left, right, left, etc.

Both can be demonstrated in choreography. The first should involve the professor side-stepping at least as far as the door, if not out into the hallway. The second, which does not require leaving the room, is to step right, then left, then right, then left, then right, then left, then involves a moue and a vocalization of the "etc." It doesn't hurt to explain out loud that the "etc." is part of the Infinite Loop Dance.[4]

3.6 Undecidability

The proof that the Halting Set is undecidable is both elegant and, at first introduction, usually baffling. We feel strongly that the students should see it, but we don't think that it is terribly persuasive. In fact, to many students, it looks like a conjuring trick. We spend some time setting up another argument that not all languages are decidable. The argument is as follows.

1. There are at least two sizes of infinity, countable and uncountable.

[4]For one of us, there is a soundtrack to this dance. It is the Romanian folkdance tune Alunelul. The pattern, if you include stamps in the step count, is 7,7,7,7,4,4,4,4,2,2,3,2,2,3. Or, to the right 5 steps, stamp stamp, to the left 5 steps, stamp stamp, repeat; to the right 3 steps, stamp, to the left 3 steps, stamp, repeat; to the right 1 step, stamp, to the left 1 step stamp, to the right 1 step, stamp stamp, to the left 1 step, stamp, to the right 1 step, stamp, to the left 1 step, stamp stamp.

2. The set of finite strings over a finite alphabet Σ of size $|\Sigma| > 1$ is countable, but the set of infinite sequences over Σ is uncountable.

3. Turing machines can be encoded as strings over a finite alphabet. Therefore there are countably many TMs, and thus, countably many decidable languages.

4. There are uncountably many infinite binary sequences, and thus uncountably many languages. Therefore, not all languages are decidable.

To set this up, we start with questions about Hotel Infinity.[5] First, "One dark and stormy night, you arrive at Hotel Infinity. It's at the edge of the universe. You are told that all the rooms are full, but you need a room to sleep in. Can you find a room without making anyone leave?"

Each question is left hanging until the next class period, or until someone asks (sometimes several classes later). After they get the idea that everyone can move from room n to room $n+1$ and still have a room, they usually find a way to house "you, and infinitely many of your friends" by sending folks in room n to room $2n$. They often get stuck on "and each of your friends has a disjoint set of infinitely many friends," but you can remind them that powers of primes form infinitely many disjoint sets. (There are many proofs that there are infinitely many primes. The one where you assume there are only finitely many, $\{p_0, p_1, \ldots, p_q\}$ and then consider the number $(\Pi_{i \leq q} p_i) + 1$ — a number not on that list, and not divisible by any prime — allows us to dissect a proof by contradiction. It's useful to set up that structure, from "Spoze not" to the $\Rightarrow\Leftarrow$ contradiction mark, as a template.)

By this time, the students are reasonably comfortable with infinity. They have shown that $\aleph_0 \times \aleph_0 \cong \aleph_0$.[6]

That's when we prove that the set of subsets of $\mathbf{N}$, or the set of infinite binary sequences, or ... is uncountable.

This is usually in the second week of class. We promise that this is as hard a concept to wrap one's head around as any in the class. They disbelieve, but hope.

Later, when we introduce the notion of a universal TM, we introduce the notion of a TM as a (finite) string. We stop and ask how many TMs there are. After some discussion, they usually conclude that the answer is "countably infinitely many". They are often pleased to be using this notion that they thought was just introduced, earlier in the semester, to amuse and/or baffle them.

When we have completed the standard Halting Problem proof by contradiction, we say, "But you already knew that not all languages are decidable." We walk them through the proof that there are uncountably many languages (the same proof by diagonalization that we gave to show that there was an uncountable set), and remind them that there are only countably many decidable languages.

[5] This is not original to us, and we are not sure of the origin.
[6] $\aleph_0$ is the cardinality of $\mathbf{N}$, the set of natural numbers; $\aleph_0$ is the size of any countably infinite set.

4. CONCLUSIONS

We have not done controlled experiments to see whether tracing a loop with their fingers helps students comprehend or remember the pumping lemma. We can, however, say with great confidence that the students woke up when asked to do so. We can also say that our own students reference the infinite loop dance during and after class. In fact, when we taught it to a colleague's class, the colleague put it into his final exam. ("For 1 point, get up and do the infinite loop dance.") Not only did all the students, one by one, do it, but they asked their professor to do it as well. For those students with math (or formalism) anxiety, the performance aspect of choreographed theory, and the chance to move around in goofy ways in the class, gives them a reason to come to class.

5. REFERENCES

[1] `Algo-Rythmics`. http://algo-rythmics.ms.sapientia.ro/ and https://www.youtube.com/user/AlgoRythmics, 2013. Accessed August 2, 2015.

[2] A. Adorjan and I. Friss de Kereki. Multiple intelligence approach and competencies applied to computer science 1. In *Frontiers in Education Conference, 2013 IEEE*, pages 1170–1172. IEEE, 2013.

[3] B. Ambudkar. Introducing network design to students via a dance activity. In *Technology for Education (T4E), 2013 IEEE Fifth International Conference on*, pages 123–126. IEEE, 2013.

[4] L. W. Anderson, D. R. Krathwohl, and B. S. Bloom. *A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives.* Allyn & Bacon, 2001.

[5] K. Anewalt. Making CS0 fun: an active learning approach using toys, games and alice. *Journal of Computing Sciences in Colleges*, 23(3):98–105, 2008.

[6] W. B. Barbe and M. N. Milone Jr. What we know about modality strengths. *Educational Leadership*, 38(5):378–80, 1981.

[7] A. Begel, D. D. Garcia, and S. A. Wolfman. Kinesthetic learning in the classroom. In *ACM SIGCSE Bulletin*, volume 36, pages 183–184. ACM, 2004.

[8] T. Bell, J. Alexander, I. Freeman, and M. Grimley. Computer science without computers: new outreach methods from old tricks. In *Proceedings of the 21st Annual Conference of the National Advisory Committee on Computing Qualifications*, 2008.

[9] T. Bell, J. Alexander, I. Freeman, and M. Grimley. Computer science unplugged: School students doing real computing without computers. *The New Zealand Journal of Applied Computing and Information Technology*, 13(1):20–29, 2009.

[10] T. Bell, P. Curzon, Q. Cutts, V. Dagiene, and B. Haberman. Introducing students to computer science with programmes that don't emphasise programming. In *Proceedings of the 16th Annual Joint conference on Innovation and Technology in Computer Science Education*, pages 391–391. ACM, 2011.

[11] T. Bell, F. Rosamond, and N. Casey. Computer science unplugged and related projects in math and computer science popularization. In *The Multivariate Algorithmic Revolution and Beyond*, pages 398–456. Springer, 2012.

[12] T. Bell, I. H. Witten, M. Fellows, R. Adams, J. McKenzie, M. Powell, and S. Jarman. *CS Unplugged*. Lulu.com, 2015.

[13] B. S. Bloom. *Taxonomy of Educational Objectives: The Classification of Education Goals. Cognitive Domain. Handbook 1*. Longman, 1956.

[14] G. B. Dantzig, A. Orden, P. Wolfe, et al. The generalized simplex method for minimizing a linear form under linear inequality restraints. *Pacific Journal of Mathematics*, 5(2):183–195, 1955.

[15] I. F. de Kereki. Incorporation of kinesthetic learning activities to computer science 1 course: Use and results. *CLEI Electronic Journal*, 13(2), 2010.

[16] R. M. Felder and R. Brent. Navigating the bumpy road to student-centered instruction. *College Teaching*, 44(2):43–47, 1996.

[17] R. M. Felder and L. K. Silverman. Learning and teaching styles in engineering education. *Engineering Education*, 78(7):674–681, 1988.

[18] M. Fellows, T. Bell, and I. Witten. Computer science unplugged. *Computer Science Unplugged*, 2002.

[19] H. Gardner. *Multiple Intelligences: The Theory in Practice*. Basic books, 1993.

[20] T. F. Hawk and A. J. Shah. Using learning style instruments to enhance student learning. *Decision Sciences Journal of Innovative Education*, 5(1):1–19, 2007.

[21] P. Henderson. Computer science unplugged. *Journal of Computing Sciences in Colleges*, 23(3):168–168, 2008.

[22] S. Jarman, T. Bell, and I. Freeman. CS Unplugged. Accessed August 2, 2015.

[23] G. P. Krätzig and K. D. Arbuthnott. Perceptual learning style and learning proficiency: A test of the hypothesis. *Journal of Educational Psychology*, 98(1):238, 2006.

[24] C. Lewis, T. McKlin, T. Berry, and A. Schlesinger. csteachingtips. `http://csteachingtips.org/tip/use-physical-activities-demonstrate-sorting-algorithms-and-help-students-build-intuition-about`. Accessed August 2, 2015.

[25] J. A. Nelder and R. Mead. A simplex method for function minimization. *The computer journal*, 7(4):308–313, 1965.

[26] T. Nishida, Y. Idosaka, Y. Hofuku, S. Kanemune, and Y. Kuno. New methodology of information education with "computer science unplugged". In *Informatics Education-Supporting Computational Thinking*, pages 241–252. Springer, 2008.

[27] S. Ornes. Math dance. *Proceedings of the National Academy of Sciences of the United States of America*, 110(26):10465, 2013.

[28] C. C. Park. Learning style preferences of Asian American (Chinese, Filipino, Korean, and Vietnamese) students in secondary schools. *Equity and Excellence in Education*, 30(2):68–77, 1997.

[29] M. Peacock. Match or mismatch? learning styles and teaching styles in EFL. *International Journal of Applied Linguistics*, 11(1):1–20, 2001.

[30] S. Pollard and R. C. Duvall. Everything I needed to know about teaching I learned in kindergarten: bringing elementary education techniques to undergraduate computer science classes. In *ACM SIGCSE Bulletin*, volume 38, pages 224–228. ACM, 2006.

[31] J. M. Reid. The learning style preferences of ESL students. *TESOL Quarterly*, 21(1):87–111, 1987.

[32] K. Schaffer, E. Stern, and S. Kim. Math dance. *MoveSpeakSpin, Santa Cruz*, 2001.

[33] P. A. Sivilotti and S. M. Pike. The suitability of kinesthetic learning activities for teaching distributed algorithms. *ACM SIGCSE Bulletin*, 39(1):362–366, 2007.

[34] R. Thies and J. Vahrenhold. On plugging unplugged into CS classes. In *Proceeding of the 44th ACM Technical Symposium on Computer Science Education*, pages 365–370. ACM, 2013.

[35] A. Vincent and D. Ross. Personalize training: determine learning styles, personality types and multiple intelligences online. *The Learning Organization*, 8(1):36–43, 2001.

[36] D. T. Willingham, E. M. Hughes, and D. G. Dobolyi. The scientific status of learning styles theories. *Teaching of Psychology*, 42(3):266–271, 2015.

[37] V. Zimmerman. Moving poems: Kinesthetic learning in the literature classroom. *Pedagogy*, 2(3):409–412, 2002.

Co-Curricular Technical Writing Activities Promoting Student Success in Computing Fields.

Greg Simco[#1], Meline Kevorkian [#2]

[#1]*College of Engineering and, Nova Southeastern University*
3301 College Avenue, Fort Lauderdale, FL 33314, USA
greg@nova.edu

[#2]*Office of Academic Affairs, Nova Southeastern University*
3301 College Avenue, Fort Lauderdale, FL 33314, USA
melinek@nova.edu

The 2016 World Congress in Computer Science, Computer Engineering, and Applied Computing

***Abstract*—Building quality programs based on workforce skills, such as technical writing, is reconsidered an effective means for recruitment and retention in higher education, especially in the areas of science, technology, engineering and math (STEM). In computer science, technical writing can be especially challenging due to the complexity of technical documents. This work incorporated the use of co-curricular technical writing activities for computing majors at Nova Southeastern University. The results support student success and growth in employability skills.**

Keywords: Computer Science, Technical Writing, Professional Expression, and Student Success

I. Introduction

Writing is the most common form of technical communication, yet many employers complain that college graduates are too often poor writers and communicators. Technical writing skills provide student success in and out of the classroom. Hooley, Hutchinson & Moore (2012) discuss how college graduate employability is also becoming an essential selection criteria by potential students. In addition, according to the 2015-2016 Criteria for Accrediting Computing Programs, criterion 3, student outcome, students must have an ability to communicate effectively with a range of audiences.

Undergraduate students are often underprepared for college coursework and often need skills development and reinforcement for success. According to The Condition of College & Career Readiness 2015, 31% of the ACT-tested graduating class are not meeting any of the Benchmarks, which will make it difficult for them in their post–high school experiences.

Employability refers to the capability to gain initial employment and retain the job. College graduates need to be employable professionals with a sustainable skill set in written communication to articulate workplace initiatives and benefit industry.

According to White House, earning a post-secondary degree or credential is a prerequisite for the growing jobs of the new economy. President Obama has reinforced the need every young American to obtain a post-secondary credential or degree with currency in the labor market. Unfortunately, nearly half of those who enroll leave without a degree and many who get a degree are unable to find and retain employment.

Technical writing is a necessary skill for STEM careers, particularly in computer science. The literature effectively supports this concept:
"In the STEM fields, the single most important thing we do to teach our students better communication skills is to teach them how to write a lab report. The reason a lab report is so important is not just because it is the end product of a research project, but because each section of a lab report has particular function that often correlates with different types of communication that a STEM professional needs to use" (Bennert, n.d., p.1).

Symonds, Schwartz, & Ferguson (2011) discuss how failure rates at the postsecondary level are alarming. Among all four-year colleges, just 56 percent of students meet this goal. At community colleges, fewer than 30 percent of students manage to earn an A.A. degree within three years. Also, the majority of students who go on to college fail to earn a degree on time, and many of those never successfully complete their degree supporting the United States now has the highest college dropout rate in the industrialized world.

Many employers share that new employees often require remediation. According to the survey, *Are They Really Ready to Work? Employers' Perspectives on the Basic Knowledge and Applied Skills of New Entrants to the 21st Century U.S. Workforce*, only 23.9 percent report that new entrants with four-year college degrees have "excellent" basic knowledge and applied skills, and important deficiencies exist at every level. Additionally, employers noted new hires exhibited significant deficiencies in written and oral communication skills.

These complaints are not only found in industry. Computer science faculty, like most faculty complain of poor writing skills among students. This work found that technical writing workshops for computing students found self-reported benefits that translated into perceived ability to be successful in coursework that required technical writing. For sustainable success in both coursework and industry, technical writing, is equally important to technical knowledge.

The implications for computer science educators are to create graduates who are more agile. The ability to take technical knowledge and articulate it to a particular audience is the main ingredient in industry and skill for student success in upper-level computing coursework. Technical writing is necessary to truly master computer science content. This further prepares future graduates in a workforce where they will move across different jobs and sectors. Workforce agility is supported through improved technical writing through co-curricular technical writing workshops

Examples of the workshops are found in the table below.

Table 1

Example 1 **Communicating Technical Information to Non-Technical Audiences** Those who work with technology and related fields often have to explain technical issues in accessible language to non-specialists. Come find out strategies you can use to make sure your message fits your audience.
Example 2 **Hack Your Brain: Creativity Workshop** Employers want workers with excellent technical skills, but they also want them to have solid communication skills and problem-solving abilities. Each of the three unique "Hack Your Brain" workshops will present different ways participants can develop creative approaches to everyday challenges and use technology to boost their creativity.
Example 3 **Designing Effective Posters** You've done the research, developed the proposal, or gotten the grant—but now you need to communicate your findings or pitch your idea to an audience using a poster and/or oral presentation. This workshop will help participants identify the most effective tools and techniques for making their posters attractive and informative.

Example content is below in table 2.

Example 1

Basic Usability Test

1. Write a brief set of "how-to" instructions
   a. determine audience
   b. determine language level
2. Observe participants while they complete task/follow instructions
   a. take notes
   b. get feedback
3. Discuss observations
   a. identify potential revisions to instructions

Example 2

Activity: Analyzing Non-Technical Audience Needs

1. Discuss the audiences for technical information
   a. Identify differences between technical and non-technical audiences
   b. Discuss genres and media through which non-technical audiences access technical information
2. Write a description of a technical process for target user
   a. Identify strategies for communicating with non-technical audiences
   b. Write a profile of an individual user within your target audience
   c. Write a technical process description for the target audience
3. Conduct Debriefing
   a. Discuss strategies and choices made in process description drafts
   b. Discuss writing process for technical information

Example 3

Activity: Innovation in Technical Communities

1. Discuss sources of innovation
   a. Conduct the "Game of Chance" activity to draw attention to the role of constraints and context on the formation of new ideas.
   b. Discuss the Japanese practice of "chindogu"—the creation of "not unuseless objects."
   c. Discuss the difference between technology as an "object" and as an "experience."
2. Propose new ways to experience a technology.
   a. Have teams of participants conduct a tagmemic analysis to generate new perspectives on an everyday technology, looking at the object as a "particle," "wave," and "field."
   b. Have teams propose innovations that change how we experience these technologies, drawing prototypes of their designs.
3. Conduct Debriefing
   a. Discuss the proposed innovations.
   b. Discuss the process of idea and prototype development.

Example 4

Activity: Writing Clearly & Concisely for Technical Documents

As Group

1. Discuss need for concise technical language.
2. Review examples of "wordy" technical sentences.
3. Discuss revision strategies for making sentences more concise.

As Individuals

4. Read sample wordy sentences.
5. Revise to make more concise.

As Group (Debriefing)

6. Share and discuss several revised sentences.

II. Co-curricular technical writing Results

Working with data collected for the technical writing workshops, the faculty measured the self-reported impact of co-curricular technical writing workshops on their connected classroom learning to work readiness skills, ability to do better in coursework, outlined experience and technical skills (above and beyond the degree) that are highly sought after for employment.

Table 3

Technical Writing Survey Results		
Statement	**Agree**	**Disagree**
The presentation connected classroom learning to work readiness skills.	100%	
The presentation outlined experience and technical skills (above and beyond the degree) that are highly sought after for employment.	100%	

The overall results of technical writing workshops led to a self-reported improvement in the written quality of coursework as well as the ability to better articulate solutions and outcomes. To this end, faculty should be discussing ways to add co-curricular activities to their computer science curriculum to better address the needs of the workforce and foster student success. By designing a curriculum that promotes technical writing skills for success we can produce greater researchers and industry leaders in computing fields.

III. Actions for Improving Technical Writing

In computing disciplines, proficiency in a skills or subset of skills without the ability to articulate your work is not enough to success in the classroom more the workplace. Additionally, the lack of technical writing prevents students from pursuing graduate level education. To promote student success implement co-curricular technical writing support including agile exercises that provide a clearer understanding of technical writing expectations and identification of common errors.

IV. Implications of the Findings

There is a lack of literature that provides a conceptual framework for increasing the technical writing skills for computer science and related STEM fields. Additional research and methodologies must address expanding the concept of conversational programmers (Chilana, Singh, & Guo, 2016). This would include those individuals working in computing fields as end-user programmers to better interface with those professionals that require programming tasks. To be successful, the computer science and technology professionals must learn to translate their understanding of their workplace projects into various genres of communication such that they are able to address the challenges of targeting the appropriate audience. Thus a common ground and form of communication must be established between computing and non-computing professionals that provides a bi-directional mapping of processes and technical artifacts to support effective project conversation skills in the workforce.

References

1. Hooley, Y. Hutchinson, J. and Moore, N. (2012). Supporting STEM students into STEM careers: A practical introduction for academics. Derby: International Centre for Guidance Studies (iCeGS), University of Derby.

2. ABET. (2015, October 15). Criteria for Accrediting Computer Programs. Retrieved February 29, 2016, from http://www.abet.org/wp-content/uploads/2015/10/C001-16-17-CAC-Criteria-10-20-15.pdf

3. ACT. (2015). The Condition of College & Career Readiness 2015.Retrieved February 29, 2016, from http://www.act.org/content/dam/act/unsecured/documents/Condition-of-College-and-Career-Readiness-Report-2015-United-States.pdf

4. WHITEHOUSE. Education: Knowledge and Skills for the Jobs of the Future. (n.d.). Retrieved February 29, 2016, from https://www.whitehouse.gov/issues/education/higher-education

5. Bennett Jr., P. (n.d.). Using Rubric to Teach Science Writing. Essays on Teaching Excellence. *Toward the Best in the Academy, 20*(8).

6. Symonds, William C., Schwartz, R., & Ronald F. Ferguson. (2011). Pathways to prosperity: Meeting the challenge of preparing young Americans for the 21st century. Cambridge, MA: Pathways to Prosperity Project, Harvard University Graduate School of Education.

7. Chilana, P., Singh, R., and Guo, P. (2016) Understanding Conversational Programmers: A Perspective from the Software Industry. Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI 2016).

Teaching Artificial Intelligence Using Lego

Dr. Jianna Jian Zhang Irgen-Gioro
Jianna.Zhang@wwu.edu

Department of Computer Science, Western Washington University
Bellingham, WA, USA

We present 12-years teaching experience on Artificial Intelligence using Lego RCX, NXT, and EV3 Robotics platforms for both undergraduate and graduate classes at Western Washington University. We present the curriculum design, methods used, and outcomes of this kind of teaching. We discuss the advantages and disadvantages of using Lego. We suggest using a systematic method to raise the quality of teaching computer science, particularly to address how to teach students to code their idea efficiently using our research methodology of flowcharting-to-code. We will address some of the current problems in our education system for computer science and present our future plan.

Keywords — University Education, Robotics, AI, Lego

I. INTRODUCTION

Lego has a long and impressive history as an effective educational tool for all ages. Many educators adopt Lego to teach math, computer science, and engineering in different levels [5,6,8] including educational therapy for children with autism [1]. Lego Robot Kits such as RCX, NXT, and EV3 can be used to make prototypes for a real world robots using Artificial Intelligence (AI). In this paper, we present and discuss methods used to teach AI using Lego Mindstorms Robotics Kits in the classroom for undergraduate and graduate students at Western Washington University (WWU) since 2004.

We will first briefly introduce the history of RCX, NXT, and EV3, and the schematic for each with comparison discussion. In Section III, and IV, we will present the class curriculum design, practice, and the changes over the last 12 years. We will present and discuss the outcome of the classes, conclude our current research, and present future teaching plan and strategy in Section V.

II. HISTORY OF RCX, NXT, AND EV3

In 1939, with 10 employees [3], Lego started as a toy company. By 1986, Lego created software that enables a PC to control light, and sound sensors. After 12 years, in 1998, Lego released the Robotic Command Explorer (RCX), a plug and ready to program microcontroller brick [3,4]. RCX is a bench mark which made a significant leap for Lego education. At WWU, we created both introductory and AI application robotics classes using RCX. Due to the highly reusable Lego electronical and mechanical parts, students can create several different robots and apply different AI techniques during one academic quarter. With only 32kb of RAM, students must learn how to create memory-efficient code which is a permanent important skill of computer programming. Not long after the creation of RCX, Lego released its NEXT (NXT) generation of RCX in 2006, and Evolution (EV3) microcontroller in 2013 (See Figure 1). With NXT and EV3 bricks, we started to teach more AI algorithms including Artificial Neuron Network (ANN), Reinforcement Learning (RL), and Genetic Algorithms (GA) to tackle more difficult AI problems such as indoor navigation, and color shade recognition.

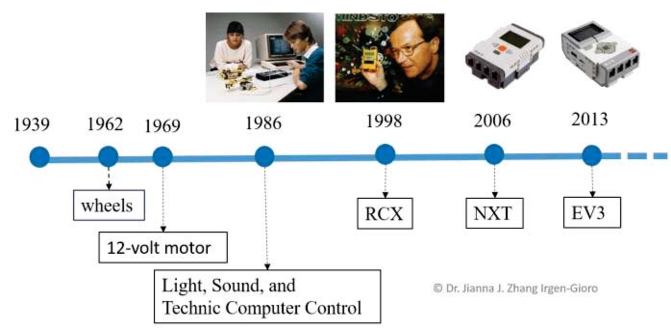


Figure 1. Summary of Lego Development Timeline [3,4]

A. *RCX: Robotic Command Explorer [5,6,8]*

Inside of the RCX, is an 8-bit Hitachi H8/3292 microcomputer with 32 kb RAM and 16 kb flash memory. The 16 kb flash is divided into 5 program slots. Instructions are executed with a speed of 16 MHz. The numeric display of the RCX is very useful for real time debugging. The numeric LCD has 43 segments that can be used to display sensor or internal values at run time. Instead of USB, RCX uses infrared signals to communicate with a PC or another RCX. The popular languages for RCX includes Forth, C, Pascal, and Java using BricxCC [16] (famous free ware), and Mindstorms SDK. A very useful feature, that disappeared in later versions of Lego microcontrollers, is a power adapter jack to allow continuous operation without consuming batteries.

B. *NXT: The Next Generation [5,7,9]*

Inside of NXT is a 32-bit AT91SAM7S256 microcontroller with 64 KB RAM and 256 KB flash memory which equivalent to a typical PC in 20th Century. On top of the main processor, NXT uses a co-processor, an Atmel 8-Bit ATmega48, to control motors. The communications between main processor and the co-processor is through a I2C bus. The execution speed is 48 MHz which is three time as fast as the RCX. Two other significant advantages of the NXT over the RCX are Bluetooth communication and the 100 × 64 LCD matrix display. On top of the original supporting languages, NXT also support Ada. However, it has no power adapter and consumes batteries much faster than both RCX and EV3.

C. EV3: The Evolution (3rd Generation) [5,9]

EV3 has a 32-bit ARM9 processor while NXT has an ARM7 processor. The size of the EV3 RAM is increased from 64 KB to 64 MB and the size of the flash memory is increased to 16 MB. The system clock speed is increased to 200MHz which is 3 times faster than that of NXT and 12 times faster than that of RCX. EV3 uses the same pixel matrix display with larger screen, 178 × 128. New features of EV3 over NXT and RCX is Wi-Fi connection to a network. It requires a wireless adapter which is also called a "dongle".

III. CURRICULUM DESIGN AND PRACTICE

The CS robotics classes at WWU were created in 2003. We have used Lego RCX, NXT, and EV3 as the platforms for both first-year, third-year undergraduate and MSc. graduate robotics courses over the past 12 years. These courses provide an introduction to robotics, AI algorithms, and how to program AI. Students start by learning to build a variety of robots then programming motors and sensors. Applications using AI algorithms depends on different levels of the courses. The general goal is to provide students a first-hand experience in quantitative and symbolic reasoning. It is not uncommon for students to be curious about AI, but fearful that it is too difficult to learn. We try to provide a positive learning environment for students who do not have AI background knowledge, and change serious and fearful learning process into a natural and interesting learning process. The following guideline defines the curriculums.

Students will be able to:

- Clearly define target problems
- Design strategies to solve these problems
- Design and build robots that are suitable to solve these target problems
- Analyze both the fundamental and complex logical relationship between input and output
- Make intelligent decisions using AI techniques to deal with physical environments
- Use flowcharts method to present decisions to the problems
- Translate ideas using flowcharts-to-code method
- Document program code
- Test and record results
- Discuss and analyze the test results
- Make future plans

With this guideline, students are required to creatively build their own robots using classroom knowledge for the final project. Throughout the course of hands-on learning, students gain a deeper understanding of AI algorithms and the applications of these algorithms in different physical environments. We found that students' motivation and efforts in learning AI theory and concepts increase as they observe intelligent behaviors from their own robots.

Due to the hardware changes and cost increases over the last 12 years, our curriculums have changed. For example, we originally used three RCX light sensors in an ANN line following robot because the cost of RCX light sensors are much cheaper ($1.76) than that of NXT ($24.95) and EV3 ($39.95). The learning strategy or the application algorithms evolved into more complexed ones that must be work with less sensor input, for example learning color recognition or balancing.

IV. TEACHING AI USING LEGO BRICKS

In this section, we highlight a few cases of our approach for teaching AI with Logo Bricks. We have three levels of robotics classes, and currently using RobotC with NXT and EV3. The first year class emphasizes teaching plain RobotC programming with simple AI techniques such as left-hand rule for line followers and maze solvers. While the third-year and graduate classes, we apply reinforcement learning, artificial neuron networks, and genetic algorithms.

A. Teaching a First-Year GUR Robotics Class

Line following and maze solving are two most commonly used projects to teach robotics classes at the beginner's level. Most of the students who signed up for this class did not have prior robotics or programming experiences. They are working towards different majors from different departments at WWU. Line following and maze solving are two of most commonly used projects to teach introductory robotics classes. We strive to teach the following:

- Understand how Lego motor and sensor work
- Design intelligent strategies for the robots to follow
- Present these strategies in flowcharts
- Translate these flowcharts to program code

Currently, the "Introduction to Robotics" class uses the NXT platform with RobotC. The Lego Mindstorms kit includes touch, light, sound, and ultrasonic sensors. Although most of the students in this class are not in computer science major. we want them to gain an appreciation for how the hardware works in addition to learn how to program. For example, the NXT touch sensor returns default value 1 or 0 (touched or not touched). We do not only show the electronic circuit, but also the internal structure of the sensors, and how to program them. We use the light sensor combined with the ultrasonic sensor to teach a simple, yet powerful, AI technique called "left-hand" or "right-hand" rule. The environment is a loop of black line on a

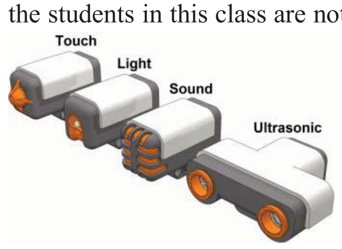


Figure 2. NXT Sensors [12]

white surface, or walls of a maze. As a light sensor reads darker color, it directs the motors turn towards the white surface and vice versa. For the maze example, when the ultrasonic sensor reads the walls are “too far away”, the robot should turn itself closer to the wall such that the robot continues to follow either left or right-hand side of the walls as designed by the students. This simple AI application makes students truly understand what AI is about, and aware that AI is everywhere in our life.

Unlike searching in a huge deterministic database, calculate with large quantity of numbers, or processing 3-D graphics, the program that we create for robots must have the ability to deal with ever changing real physical world. This can be seen when considering the simple light sensor example above. When the lighting condition changes at the different time of the day (sun rise, down, sunny, rainy, etc.), immediately students noticed that their robots behave differently. “It does not listen to me”, as some of the students commented. This demonstrates the need for dealing with environmental changes, and we must “give” the robots the ability to do so. We further direct students to think about how would a person deal with this situation, and how could we code the robot to collect the environmental statistics, such as color intensity, and then automatically calculate a threshold. Students now realize that it is not hard to code an intelligent behavior limited to a fixed environmental condition, but it is very difficult to make a robot adapt to the ever changing environments. At this point, we can introduce some more advanced AI: learning behavior.

For this introductory class, we introduce a simple learning behavior: automatically set the light thresholds by collecting environment statistics. That is, read the black and white color under any lighting condition, and then calculate the thresholds on the fly. With this learning behavior, robots can follow the black line under any lighting condition. As result, students suddenly realize that we can code a learning behavior using less than 10 lines of code.

We teach students to use an ultrasonic sensor to detect the walls of a maze. The task is to keep a constant distance between the robot and the walls of the maze when the robot is moving forward to the end of the maze. Note, we eliminate loops of walls in the maze to simplify the learning environment for first year students. Two major difficulties we want to address here. The first one is that the ultrasonic sensor reading for less than 3cm or longer than 180cm are both equal to infinity. Theoretically, the NXT ultrasonic sensor has a reading 0-254 cm, but the reality is quite different. The readings of distance 0 and 255cm are equally set to the value “??????”. This error default reading not only occurs with Lego ultrasonic sensors but also with general ultrasonic sensors, such as Arduino compatibles. The second difficulty is the blind spots. There blind spot is about 12-15° between the transducer to the receiver of the sensor. Students must learn to deal with these problems while programing the left-hand or right-hand rule.

We now discuss our teaching method to transfer these ideas into a computer language such as RobotC. The approach starts from defining the target problem. We require each student describe the target problem using their own language. They must define the general problem, and divide the general problem into sub-problems. Then the students must decide what kind of robot should be used. Based on this information, we worked with the students to define a logic table.

In a logic table, we first lay out the input and output variables and then display the logical relationship between these variables. Table 1 illustrates an example of a logic table that shows the relationship among a light sensor, a sonar sensor, and two motors. The task is to follow a black line and avoid objects on the black line using a right-hand rule (Example 1).

Table 1. Logic Table Example 1

Light	Sonar	Left Motor	Right Motor	Action
<= Thr	>= Max	slower	Faster	Turn left
> Thr	>= Max	faster	Slower	Turn right
Don't care	< Max	stop	Fast	90° left turn
		Master (full speed)	Slave (30%)	Circle around

Each row shows one relationship, and it is the flowchart’s job to show how these individual actions arranged sequentially or parallel in order to fulfill the overall line following and object avoidance task. We must help students understand that for each set of input (situation), there must be a set of output (actions). Note, the logic table does not necessarily show the sequence of actions but the logic between sensors and motors.

After the logic table is generated, we show students how to create a flowchart based on the logic table and task description. We strongly encourage students to draw a flowchart first, examine the flowchart carefully to see if there are any logic errors, problems, redundancies, and or any inefficiencies before they start programming. However, students, particularly most of the computer science students are not used to drawing flowcharts. They tend to start coding without thoroughly understanding the problem and the logistics. This results in inefficient time usage. That is, taking longer time to debug in order to solve unknown logical errors. In the worst case, they have to rewrite the entire code due to “unknown” or “hard to figure out” logical errors. This inefficient style also has a tendency to produce redundant code, waste CPU time, and memory. We should not waste memory just because the size of memory increased dramatically. On contrast, we should keep educating our students to pay attention to using their resources efficiently.

How do we make a flowchart? There are two simple rules:

- Using correct symbols (just like using correct alphabet for English language for communications, Figure 3)
- Directed graph with clockwise direction (mixing both counter clockwise and clockwise arrows make a diagram hard to read)

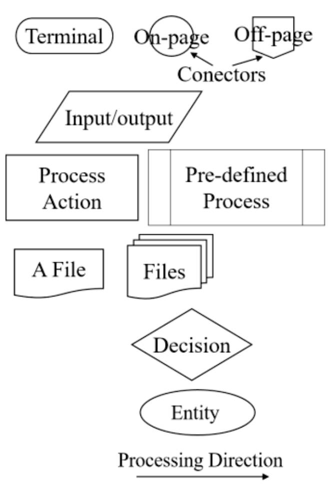


Figure 3. Flowchart Symbols

Figure 3 shows the essential flowchart symbols, and Figure 4 shows the flowchart for Example 1. All lines have its own direction, the entire flow is in clockwise direction, and the whole chart should be simple and clean. The arrows should be connected to the next action sequentially, and the loops should be indicated by clockwise arrows. To avoid cross over lines and crowding, we should use either on-page or off-page connectors. If one flowchart becomes crowded, we should break it up.

Students must think very carefully. Which sensor value has higher priority? How do we make a loop of sequential processes to fulfill the task? Where are the loops, and where to put the condition for each loop? Because the logic table only shows logically related segments of information for coding, we must teach students to make the sequential connections. We begin with asking students to determine which condition the program must check first? For our example, it is the sonar sensor because when the robot is too close to an object, the robot must go around the object no matter what color the light sensor reads. Students noticed that when the sonar sensor was used as the first condition in the loop, then the rest of the parts of the flowchart are merged together naturally. Translating the flowchart to code becomes an easy task if the students are familiar with RobotC syntax, which we have taught through many examples prior to the lectures on creating flowcharts.

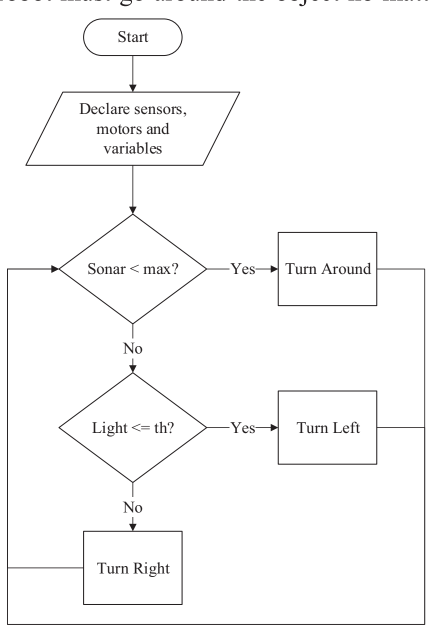
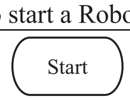


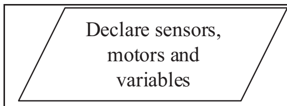
Figure 4. Example 1 Flow Chart

Here are the translation steps. First we show students how to start a RobotC program:




```
task main( ) {
}
```

Then using the built-in method of RobotC to declare sensors and motors:

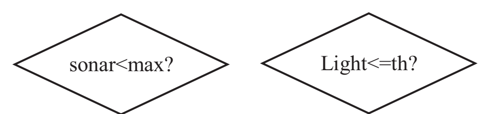



```
#pragma config (Sensor, S1, light, sensorLightActive)
#pragma config (Sensor, S2, sonar, sensorSONAR)
#pragma config (Motor, motorA, leftWheel,
                tmotorNXT, PIDControl, encoder)
#pragma config (Motor, motorB, rightWheel,
                tmotorNXT, PIDControl, encoder)
   //*!!Code automatically generated by 'ROBOTC' … !!*//
```

Next the variables are declared:

```
task main()  {
  //declare variables
  int max = 25;  //max distance between robot and object
  int th = 40;     //threshold for distinguish dark color
  //check sonar sensor
  //check light sensor
}
```

Now we show students how to use the input sensor values as control conditions using the "if" statement:




```
task main()  {
   //declare variables
   int max = 25;  //max distance between robot and object
   int th = 40;     //threshold for distinguish dark color
   //check sonar sensor
   if (SensorValue[sonar] < max)   {
      //turn around
   }
   //check light sensor
   else if (SensorValue[light] <= th)   {
      //turn left
   }
   else //turn right
}
```

The code is almost complete except the loop which repeats the sequence of processes. We guide students in the completion of this task by pointing out the flowchart again. Immediately students realized that the loop can be created with the addition of a "while" statement before the "if" statement. The resulting program produced with students in a class is shown below:

```
task main()  {
     //declare variables
     while(true)  {
          //check sonar sensor
          if (SensorValue[sonar] < max)   {
           //turn left
          }
          //check light sensor
          else if (SensorValue[light] <= th)  {
             //turn left
          }
          else   {  //turn right   }
     }
}
```

We have presented a systematic and logical method called "flowchart-to-code" to develop computer programs. This approach is especially useful for larger and more complex software.

B. Teching Atificial Neuron Networks with Lego Robots

We have been teaching ANN using RCX, NXT, and/or EV3 since 2003. The basic technique we use in higher level undergraduate and graduate classes are the same as we have shown in the previous section. We first explain the theory of ANN, show samples of the topology of forward networks, explain how to compute the gradient, and then the computation of the delta value for the back propagation. It is particularly effective when we connect the algorithm with a real problem such as learning to distinguish colors or learning to follow a black line instead of hard coding the behavior.

Students are quick to understand the concept of ANN, but tend to have difficulties applying the theory and concept to real world projects. In the past we have used the RCX with three light sensors to learn line following behavior around a complex course (see Figure 5). The general idea is to train a robot to take correct actions when it sees a black line.

Figure 5. Line Track Used by ANN Robots

This may seem straight forward, but it is not as simple as it sounds because the robot is not in an ideal virtual world and make sharp turns accordingly. The test results were not ideal compared with hard coded robots, however all the ANN RCX robots can correctly follow the most part of the black line shown in Figure 5. With faster speed, say > 75% power level, they all have difficulties following the zigzag sharp corners closely. The typical topology for the RCX line flowing ANN robot is 3×5×2. Three input light sensor value is fed into the hidden layer, and the output are two actions: left turn or right turn.

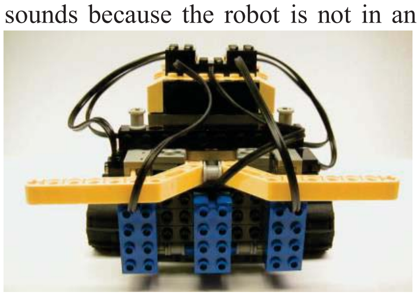


Fig. 6. RCX Line Follow ANN Robot

We applied ANN and general AI algorithms to many other projects such as learning to walk, self-balancing, Tic Tac Toe game engine, a Hanoi Tower problem solver, searching for objects in an open area, learning to type on a keyboard, learning to generate a piece of acceptable music, and more.

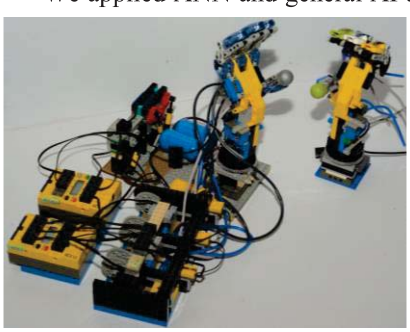


Fig. 7. Pnuematic Hand Learn to Grap

Figure 7 shows another robot that we use for research and teaching ANN. These robot hands are based on the pictures posted on the Brickshelf [10]. It is not only a good platform for ANN learning but also good platform for reinforcement learning. We use two RCXs, six motors, and two air tanks pneumatic system. The details of this project will be published in a separate paper in the near future.

A NXT Turing Machine shown in Figure 8, is another classroom example which implements an abstract Turing Machine. This Lego NXT Turing Machine is built based on [13] with many modifications. We use 16 lift arms to represent 16 bits of I/O (2-symbol: 0 or 1), and 3-states (L, M, R). The initial head position is on the top of the 4th bit of the "tape". The initial instruction is one byte located from the 4th to the 11th bits. There are two kinds of operations for current application: do math or display text. We use right hand side 8 bits (0-7) to represent the first operand, and the left hand side 8 bits (8-15) to represent the second operand, where both can be any of the 128 ASCII code. The input data can be entered by users or randomly generated by the NXT brick. The tape (with the lift arms on it) moves back and forth. We use an infinite number of finite states to represent the infinitely long tape while the head of the Turing Machine stays in place. If a lift arm is away from the head reading point, the input for that particular bit is a "0", and "1" otherwise. Let the input string from the Turing Machine be: $\{0,1\} \mapsto Ascii\ code$ such that the following finite states hold:

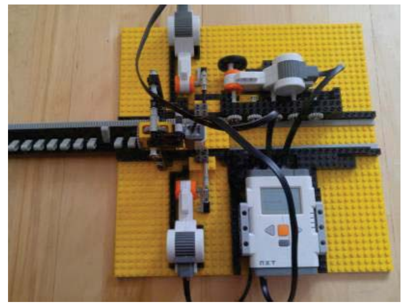
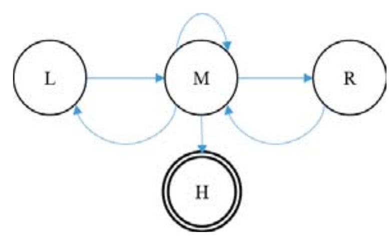


Fig. 8. NXT Turing Machine



The corresponding state table for the NXT Turing Machine is defined as the follows:

Previous State	Current State	Scanned Symbols	Print Symbol	Move Tape	Next State
Start	M	O: Ascii	0/p	fwd	L
M	M	0	0/p	stop	H
M	M	1	1/p	idle	M
M	L	R1: Ascii	0/p	rev	M
L	M	R2: Ascii	0/p	rev	R
M	R	No scan	0/p	fwd	M
R	M	No scan	0/p	fwd	L

Table 1. NXT Turing machine State Table

We move the “tape” forwards or backwards based on the M state value scanned at the beginning of operation. Then the tape will move to the beginning position which is R. Then the machine will scan two values, R1 and R2. For example, if the operator is Ascii code “+” or “/”, the machine will perform the addition or division of R1 and R2 respectively. While if the operator is “T”, then the Turing Machine reads a sequence of Ascii code, and translate them into text. The text is displayed on the LCD panel. In the future, we want to make it into text-to-speech Turing Machine.

Although the RCX and NXT can be used for many AI algorithms, currently both RCX and NXT are no longer supported by Lego company. We have now switched to EV3 bricks. As discussed in previous sections, EV3 has improved tremendously in comparison with the RCX and NXT. Figure 9 shows a multi-purpose EV3 robot which we currently use to teach both introductory and higher level robotics classes. We made this robot as simple as possible to save material and building time. It can be used for line following, navigation, obstacle avoidance, detecting color, climbing stairs, and as a platform for AI applications.

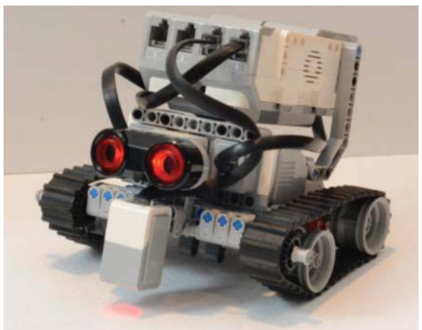


Fig. 9. EV3 Color Recognition ANN Robot

For example, we use this robot to teach classical Backpropagation ANN. The learning task is to make the robot recognize 5 different colors: red, green, blue, yellow, and brown while it is traveling on a black ring on a white surface. Based on our classroom experiments, EV3 has less accurate reading for “blue” than that of “red” and “green”. RobotC language provides a convenient way to obtain the standard Lego color: 0 = colorNone (not color object detected); 1 = colorBlack; 2 = colorBlue: 3 = colorGreen: 4 = colorYellow: 5 = colorRed: 6 = colorWhite; 7 = colorBrown. However, if any color that does not belong to the standard Lego color, this mode would fail to classify them. To learn colors that are different from the standard Lego color, we have to use the raw EV3 RGB value. Theoretically, each of the R, G, and B channels have a value from 0 to 1024. That is over one million (1,070,598,144) distinct RGB color sequences. However, after many tests, we found that the maximum RGB value never exceed 800. Thus in reality, the input RGB color sequence is less than 510,081,600. As we can see, the input space is still huge. To make this computation possible, we use the hierarchical ANN techniques as shown in our previous research [11].

We first roughly classify these half million input RGB sequences into three classes using a generalization. That is, the first class of the RGB input sequences are those causing the R-output neuron to fire, the second class of the RGB input sequences are those causing the G-output neuron to fire, and the third class of the RGB input sequences causing the B-output neuron to fire. Then we use these three classified RGB sequences to train three identical but separated sub ANNs. The final output of the entire ANN system can recognize 125 colors in the RGB gamut cube. The general approach for this project is the classical Backpropagation using Gradient Decent. The topology of the ANN is: 3×3×3 for the first ANN, and 3×40×125 for the three identical sub ANNs. We will report our testing results for this more complex example in a separate paper in the near future.

Currently, we request students apply a simpler version of the ANN color recognition exercises. We use Lego predefined 8 color as input training data on the classical Backpropagation ANN. We request graduate level students apply the more advanced Hierarchical ANN approach. No matter how much more complex the programming task may be compared to those for the first year students, we found that the teaching method explained in Section IV is very effective. Students are taught to use a systematic method to program AI, and help them reduce confusion when implementing the details.

C. *Teaching Reinforcement Learning with Lego Robots*

Reinforcement Learning is also taught in our classroom using Lego robots. Figure 10 shows a NXT Lego guide “dog” robot that uses a Q-Learning algorithm to train itself not just listen to the owner’s command, but also learn to disobey when it encountered a dangerous situation [14] [15]. The dangerous situation including, but not limited to, objects blocking the way, a hole or a ditch nearby when the owner gives command to move to that direction.

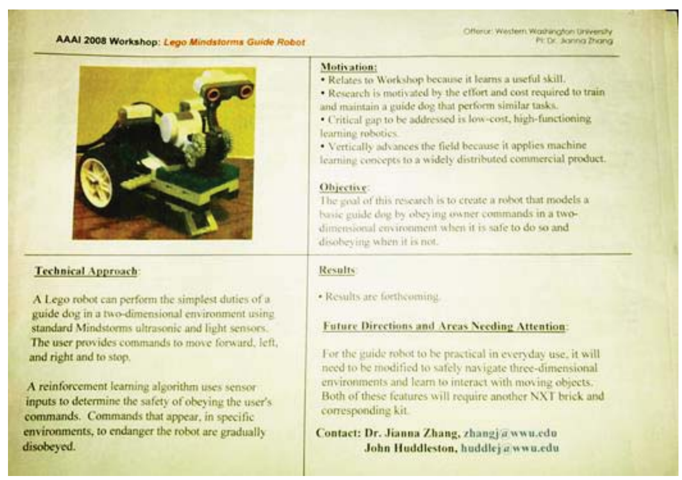


Fig. 10. NXT Guide Dog Robot, 2008 [14] [15]

This NXT guide dog robot operates in the 2-dimentional *n*×*m* grades-world using an ultrasonic and a light sensor. A policy must set to enable the initial learning. We must first set the reasonable policy for initial learning. We set this initial policy as follows: the robot can move one step forward, left, right, and reverse within the closed environment. For example, given the environment is a 3×5 grades, and the current position of the robot is $p = (1, 1)$, then the robot cannot move to (1, 2), (2, 1) but not outside the limit. The goal of the learning is to obtain an optimal policy to avoid obstacles automatically, and disobey the owner’s command in a dangerous situation. The general Q-Learning function given this initial policy on action "*a*" given a state "*s*" can be written as

$$Q(s, a) = r(s, a) + \gamma \text{ argmax}_{a'} Q(\delta(s, a), a') \quad (1)$$

where "*r*" is the reward value for action *a* in state *s*, γ is an arbitrary learning speed between 0 and 1, and *a′* is one of the next actions that is allowed by the current policy. We want to maximize the Q-value for each given state and generate two things: an optimal action and a new Q-value. For example, if the next action "*a′* = left" gives the highest Q-value based on the current state value $\delta(s, a)$, then the robot would turn left. After performing this action, the robot would update its Q-table with the optimal Q-value. The robot repeats this learning process from different starting positions, and eventually it can continually improve its performance by looking up the learned Q-table: the state-action pair. This practice helps students gain an understanding of the fundamental machine learning theory.

D. Teaching Genetic Algorithm with Lego Robots

The students are asked to reuse their Backpropagation ANN programs to learn the general Genetic Algorithm (GA). Instead of using the Backpropagation, we show students how to use GA to generate candidate ANNs, evaluate these candidates, select parent pool of ANNs from these candidates, and then use GA operators to produce child pool of ANNs. For example, we start with random generated ANNs (sets of weights), then test them to see how many colors that each ANN classified correctly. These test results show how "good" each ANN is. Several selection methods are introduced, such as Ranking, Rolette Selection, Tournament, Steady State, and Elitism Selections. Students can use one of these selection methods or combine them to generate a pool of parent ANNs. After that we explain how to use GA operators, such as Cross Over and Mutation, to produce a pool of child ANNs. This process of measurement, selection, and mutation are repeated over and over until several satisfactory ANNs are generated. We found that by modifying previous ANN programs to learn GA, not only helps students learn a new machine learning algorithm, but also reinforce their knowledge on ANN.

V. Future Teaching Plan and Strategy

Our 12-year classroom experience shows that using Lego robots to teach AI is economical and effective. Students can extend their creative ability by making different kinds of robots more easily, spend less time on the mechanical construction, have more time on learning AI algorithms, and programming to deal with real world complexities. We regard this first-hand programming experience as a very important preparation for students to be successful in their future careers. Using Lego is economically desirable compared with other equivalent commercial robots. However, using Lego as the classroom equipment is labor intensive for maintenance, such as tracking and sorting Lego parts as well as the sensors, wires, and batteries.

In the future, we want to create more demo examples on color sensor because there is still much to discover. Currently, the raw RGB value do not show the full range, but we do not fully understand why. More investigation must be done to test the electronics, mechanics, firmware, and RobotC language on EV3 color sensor.

Another interesting sensor is the EV3 Gyro. This sensor is not stable. The well-known problems are its drifting and lagging. To make this sensor useful with RobotC programming language, we have to hard code a counter drifting every mSec There must be better solutions. As for the lag, the most popular solution is to use a third party Gyro sensor. There is some problem located in RobotC language or the firmware. We may create more sophisticated RobotC library as a solution.

We plan to connect cameras to Lego EV3 brick, because we believe computer vision is an important key to indoor navigation robots. We want to further develop our teaching methods to help students learn better and faster. We feel that the current education system of teaching computer science needs more focus on quality instead of quantity. Using robots as real world programming platform helps students raise their comprehension level, and increase the ability to problem solve.

References

[1] Ferrandez, Jose Manuel ; Paz, Felix ; Barakova, Emilia I. ; Bajracharya, Prina ; Willemsen, Marije ; Lourens, Tino ; Huskens, Bibi; "Long-term LEGO therapy with humanoid robot for children with ASD", Expert Systems, 2015, Vol.32(6), p.698(12) [Peer Reviewed Journal].

[2] Gomez-de-Gabriel, Jesus Manuel; Mandow, Anthony; Fernandez-Lozano, Jesus; Garcia-Cerezo, Alfonso, "Mobile Robot Lab Project to Introduce Engineering Students to Fault Diagnosis in Mechatronic Systems", IEEE Transactions on Education, Aug. 2015, Vol.58(3), pp.187-193 [Peer Reviewed Journal]

[3] Lego Group, "Lego History Timeline", http://www.lego.com/en-us/aboutus/lego-group/the_lego_history, 2016.

[4] Mindstorms, "History of LEGO Robotics", http://www.lego.com/en-us/mindstorms/history, 2016.

[5] Official Lego Vikipidia, "Lego Mindstorms", https://en.wikipedia.org /wiki/Lego_Mindstorms#RCX, 2016.

[6] Hitachi, "Hitachi Single-Chip Microcomputer", http://www.legolab .daimi.au.dk/CSaEA/RCX/Manual.dir/H8Hardware.pdf, 3rd Edition, accessed 2016.

[7] ATMEL, "AT91SAM ARM-based Flash MCU", http://www.atmel.com /Images/6175s.pdf, 2012.

[8] University of Aarhus, Department of Computer Science, RCX Manual,

[9] Lego Mindstorms EV3, https://en.wikipedia.org/wiki/Lego_Mindstorms _EV3, accessed 2016.

[10] Brickshelf folder: MOC Bionicle Technic Mecha hand legotic, Folder created: 2003/11/30 13:10:26, Folder modified: 2006/01/26 21:13:35, accessed 2016.

[11] Zhang, J., Blachford, D., and Tien, J., "Chinese Handwriting Recognition System Using Artificial Neural Network" in Chinese Language Instruction and Computer Technology, in Traditional Chinese Language, US Department of Education, Printed in Taiwan, June 2005, pp. 259-273.

[12] LEGO® Education, https://education.lego.com/en-us/, accessed 2016.

[13] Mario Ferrari, Giulio Ferrari, Kevin Clague, J. P. Brown, Ralph Hempel, "LEGO Mindstorms Masterpieces: Building Advanced Robots", 1st Edition, Syngress, 2003.

[14] Jianna Zhang and John Huddlston, "Guid Robot: a Q-Learning Lego Robot", The AAAI 2008 Workshop on Mobility and Manipulation (held during the 23rd AAAI Conference on Artificial Intelligence), 2008, Chicago, USA.

[15] Anderson, Monica; Jenkins, Odest Chadwicke; Oh, Paul, "The 17th Annual AAAI Robot Exhibition and Manipulation and Mobility Workshop", AI Magazine. Volume: 30. Issue: 1, 2009, pp. 95-102.

[16] Bricx Command Center, http://bricxcc.sourceforge.net/, accessed 2016.

ISBN: 1-60132-435-9, CSREA Press ©

SESSION

NOVEL EDUCATIONAL PROGRAMS, COURSES AND STRATEGIES + WEB UTILIZATION + SURVEY AND REVIEWS + NOVEL PROJECTS

Chair(s)

TBA

An Integrated Multimedia Computer Science Summer Program for Middle School students

Ching-yu Huang[1], Richard Olszewski[2], Rocco Tomazic[3]
Kean University[1,2], chuang, olszewri@kean.edu
Freehold Borough School District[3], rtomazic@freeholdboro.k12.nj.us

Abstract – **Computers and technology are becoming an ever more essential component in business and human life. It is very important to incorporate modern technology and computer science knowledge into science curriculum in middle school to increase the desire from students in this area to learn more about computers. We proposed a multimedia summer program for promoting computer science education in middle school. The program was successfully run at Freehold Borough School District in New Jersey from 8 am to 12 pm, four days per week for 3 weeks, from 7/6 to 7/23, 2015. Topics covered, such as website creation, media editing, and Microsoft Excel Macro, enticed the students to establish new skills, allowing further growth and interest in the field. With this program, we hope to foster a desire from the students to head out to college interested in computer and technology majors, and pursue related careers.**

Index Terms – Multimedia, web, technology, education, summer program

1. INTRODUCTION

According to a study by code.org, there will be one million more technology jobs than students by 2020 [1]. There is a need to have more qualified Computer Science (CS) teachers, better and interesting curriculum to attract and educate more students in the secondary schools. A comprehensive study conducted by Liu [2] compared several K-12 teacher training programs and found their lessons can improve CS education. Most young students in recent decades have had exposure to technology in one form or another such as smart phones, mobile devices, smart boards, and educational software on computers. They might know how to use the technology, but most of them might not know the design. It is very important for students to experience technology early so they can be aware the options to study computer related courses further in high school and select information technology majors in college.

Many states in the USA and other countries are changing curriculum to include the computer and technology prior to a college education. A CE21 project aims to establish Maryland high school students to have access to high-quality CS courses [8]. Israel launched the Science and Technology Excellence program (STEP) in 2011 to strengthen science and technology K-12 education [9]. Learning computer programming usually requires a student to have an Algebra background so the coding curriculum is usually after 9th grade. Buffum [5] showed how the computer science programs currently available can link up with current curricular changes, like the new "Common Core State Standards" approach to learning in middle schools. The Computer Science Teachers Association (CSTA) built a model computer science curriculum and set student learning objectives and for K-8 grades [6].

A study by Stewart-Gardiner [3] showed that early exposure to educational computer games will influence middle school girls to pursue computer science in high school and college. In the modern world, computer technology should not just be limited to programming; it should also cover architecture, design, hardware and software. Computer and technology technical jobs include writing requirements, user interface design, human computer interaction, data and system analysis, engineering, architecture and system configuration, as well as coding, quality assurance and control, security, testing, maintenance, administration and support [9]. It is more important to trigger students' interest than to teach a highly rigorous curriculum for learning technology in middle school. A pathway in multimedia curriculum proposed by Huang [4] is suitable for our proposed computer science summer program that focuses on interactive multimedia software such as Microsoft (MS) Excel, MS-Movie Maker, and Weebly - a website creator.

2. TEACHING METHODS & MATERIALS

The summer program was run in a computer lab at Freehold Intermediate School, NJ. The instructor was a junior undergraduate student who is majoring in computer science and he used slides shown through a projector for teaching. For middle school students, it is more attractive for students to learn how to utilize the application while learning some design and coding concepts. Hands-on experiences are very important to technology and computer science education. Every student had a computer to do exercises. The computers run on Microsoft Windows systems. Since every student's learning habits and reactions to new teaching materials are different, it is better to have close-age Teaching Assistants (TA) to help the individuals. Therefore, six high school students were recruited to assist the instructor for this summer program.

The program was from 8am to 12pm, four days per week for 3 weeks, from 7/6 to 7/23. There were a total of 23 middle school students in two groups; there were 10 students in group 1 (8-10am), and 13 students in group (10-12pm). Due to the relatively small age gap between the students and instructors, the middle school students were able to communicate better and learn more quickly.

The students were given the opportunity to build websites and make small movies as assignments for the class. Making movies and websites with the students allowed them to bring out their creative side. Our other program in the curriculum, Excel Macro, helped show more of the practical side to programs and computers.

At the end of every week, Professor Stewart-Gardiner, a professor from Kean University, came in and introduced three different hands-on exercises related to Computer Science at the basic level (without use of a computer.) The students enjoyed these exercises, as they took a nice break from the computer each Thursday, while still learning about what can be accomplished with the proper knowledge. Post program analysis showed that the program sparked increased interest in computers and technology at Freehold Intermediate School.

3. CURRICULUM DESIGN

The program designed for the 2015 summer experience covered a simple motive: to teach students at the middle school level about the useful software in a computer. We decided to cover the more interesting topics for the beginning of the session, along with movie editing and website development. Microsoft Excel was introduced further on, covering data analysis, basic programming knowledge, and basic chart creation.

The class started the session with building a basic website which could be added to later on. After each section was finished, the class went back to add the newly created content to the website. Throughout the course length, the students added their videos, charts, and other creations onto their sites. Going back to post the new content helped keep everything cohesive in the program, and also helped to keep the knowledge of how they accomplished each part fresh in their mind. For example, this approach allowed the students to start off making sites about their favorite soccer player, then later on adding a personal video about that player in particular.

3.1 Web Design

Websites are the entry point for any person using the Internet. For this reason, the first part of the program seemed logical to be website design. The website would give students something to build that they could expand upon in the class, while giving something to share with parents and friends. It allowed them to have a direct and immediate impact on the web. Weebly, the website which allows free website creation with a plethora of easy-to-use tools, was the tool of choice used to teach website design.

As we moved from website creation to movie making, the idea of embedding a video went hand-in-hand with each other. We first started with embedding videos from YouTube, as “first steps” into expanding their websites. The students would make separate pages dedicated to these videos, and link them with different drop-down menus. Modifications to how a person could navigate a website better, while still being aesthetically pleasing to users, was something that came easily to the students as they worked. Other students around them would immediately try to use the site, and would give the creator instant feedback on what could make it better.

3.2 Making Movies

As previously mentioned, using Windows Movie Maker was the next phase of the program. With the two sessions of this program, each student made a video roughly two minutes in length. The videos covered content that either interested them greatly, or correlated to their site (e.g. a top ten video of a favorite team, or talking about a favorite show, etc.) Each student, after creating their personal video, had to learn how to add it to their websites as well. This helped further to link the programs back to each other.

The class covered important parts of video creation, how to create different parts that make a solid, concise video. The requirements for a completed video were the following: a title, a body, two captions, two effects, and a credits sequence. No student had any issue meeting the requirements, showing quick understanding of the program.

3.3 Data Analysis

Microsoft Excel was expected to be the hardest of the three programs in the class, even before the topic was discussed. It was not as flashy as making videos or websites, opting for practical uses like making spreadsheets and charts instead. The first couple days were spent learning the software, how to navigate the interface, entering in values, and making different charts with the list of sample values set into the spreadsheets.

Once the core concepts were understood by the students, the class was set to move into basic Microsoft Excel functions. Simple and useful functions, like SUM, ADD, and CONCATENATE, were used to build a solid foundation on how to interact with the functions, how to call them, and what they can be used in real world scenarios.

The class then moved onto making Microsoft Excel charts, a tool to anyone who may work in an office setting, as well as giving the students an edge on how to make clean charts for their future projects in school. The idea of making charts to show data in a visual format was the goal. Showing the students that it is sometimes easier to look at a chart to understand data, rather than a spreadsheet, was the main focus.

3.4 Basic Programming

As the end goal of the program, the students were expected to leave the summer course with an understanding of basic programming. The project had plans to use Microsoft Excel’s Visual Basic for Applications (VBA), or Excel Macro, to cover the basics of input/output, function calls, and GUIs, or Graphics User Interface. Showing the students their first few lines of code filled the students with determination. It was their first opportunity to make something entirely their own on the computer, more so than a movie or website. Over the course of a couple days, the students encountered hurdles. Testing different things out with the few functions they knew had them discover different quirks behind the coding language. The students did improve, however, with certain students taking initiative with Excel Macro and creating their own unique programs. These involved a calculator, with a very solid GUI, and a detailed grading scale. Some students did not enjoy the programming portion, however. A slower pace may have been needed for the subject, but improvements made by select students over the span of the topic showed that those students could take away a desire to learn more about programming after only limited exposure.

3.5 Data Presentation

With significant assistance from Kean University's Professor Stewart, the instructor was shown what was needed to successfully deliver the program. With her experience and knowledge, she showed how the subject should be taught at younger levels of education. A teacher for the program should be one that is comfortable with letting students take their time to understand the core concepts, while also giving the students who are obviously ahead the space and support to grow. They will also need to be able to teach less with their words and more with examples and hands-on time, something that helps to make the learning curve more gradual.

Methods of teaching at the high school and college level do not apply in this program. Getting to see what experienced teachers do, how they handle situations with younger students, and how they demonstrate the time and patience needed for the class are all important features to learning effective instruction. To teach young students advanced topics like programming, without either relating it to something they already know or making it engaging with the students, will leave a lot of them confused and falling behind. Teaching them through games and basic ideas, like Stewart did, helped the students grasp harder concepts by relating different ideas to parts they completed in the games. These are the components that are needed to engage younger students and to fire up their interest in computers and technology.

SUMMER PROGRAM PILOT STUDY

For the undergraduate instructor, being a new teacher came with some new experiences. Teaching new concepts and ideas to students, while not previously having any experience in teaching this age level, might well be the hardest part of the internship. But learning to teach was enhanced by learning how younger students learn. Any new instructor needs to be determined that they will help students to learn the content, one way or the other.

With the students' timely formative feedback, they gave an impression of where to take the lessons. Certain students would need more time on parts of the project, while others showed complete knowledge of the subject. The curriculum that was created was adjusted to fit the class after the first few days. The students responded well to this. Those students that had shown talent, the students that had previous knowledge of subjects, were given more of a challenge with different projects. The students that had issues with certain concepts were given extra help. Other students that had also completed the project were also able to help guide the other students as well.

RESULTS AND CONCLUSION

Surveys: Before and After

A survey of proficiency was given to each student at the beginning and the end of the program. Each student was asked the same set of questions, some questions pertaining to time allocated to the computer at home, while others asked about previous knowledge of the programs used in the class. The results were varied, as expected, with many having drastically different experiences with the computer before coming into the course. That preparation became essential for the program running as smooth as it did. Planning out multiple ways to introduce material, expecting different students to understand material in various ways – it was all a challenge. It all paid off in the end, however, with the two sections of students leaving the program with a new idea of computers and the purposes they can serve to others. In fact, one of the students in the 2015-2016 school year was on a winning team in the PicoTurbine Competition.

The program was a little different to what is currently taught in middle schools today. With it covering the basics of website creation, video creation, and Visual Basic programming, it tapped into many different areas of learning and creativity for the students. It gave certain students the ability to make their own piece of the internet to share with others. It gave an understanding of what goes on behind the scenes for all websites, at least at the surface. For subjects outside of computer science, it will expand their ways of thinking and presenting their work. Whether they apply the ideas and concepts they learned in the program to subjects like Math or English, they will start to think, "How would I approach this problem now?" Would they go about it in a systematic way, detailing each step as they go, or would they think of a more creative solution instead? Computers allow for so much more than just the study of the workings of a computer, more than a straightforward learning of topics. The students leaving this program went into their next academic year with an extra way to think about problems. The extra step they took over the summer, to learn something new and introduce themselves to new ways of thinking and learning, will allow them to excel and find new ways to show what they know.

If this program will be continued in the future, several improvements can be arranged. First, and the biggest change, is to make the program longer in length. Over the three week program, it felt rushed on certain topics. The students felt it as well, as a good portion of the students needed an extra couple steps until they felt comfortable in what they were doing.

ISBN: 1-60132-435-9, CSREA Press ©

Another change to be made, to help it fit in better with the school curriculum, is to ensure each topic of the summer program is covered in the beginning of the school year for technology courses. If there was any prior knowledge to what would be taught in the fall, the program could be modified to include those topics. That would give the students the ability to learn more in what they are already interested in, engaging them with subjects they want to learn. With the help of high school teaching assistants, each student enjoyed different parts of the program, giving their own ideas on what to do different the next time. One of the important factors in making the program successful is that the high school teaching assistants used their closer communicative dialog to guide the middle school students.

Students need schools to help them prepare for their future. The future that elementary and middle school students will face will be highly reliant on computers and the understanding of technology to solve problems. K-12 education cannot allow instruction in computers and technology to be just a highly specialized elective at the high school level.

The society of tomorrow will require more computer and technology professionals than ever before. In order to meet this demand, higher education and departments of computer science have to form partnerships with K-12 school districts to find ways to develop more interest in computers and at a younger age. This project provided an example of how that partnership might be formed, developed, and executed.

ACKNOWLEDGEMENT

This program is funded by the Foundation Faculty Research Award (FFRA) at Kean University. We really appreciate Kean University's support.

REFERENCES

[1] Promote Computer Science , https://code.org/promote

[2] Jiangjiang Liu, E.P. Hasson, Z.D. Barnett, Peng Zhang. "A survey on computer science K-12 outreach: Teacher training programs", *Frontiers in Education Conference (FIE)*, Rapid City, SD, 12-15 Oct. 2011.

[3] Stewart-Gardiner, C., Carmichael, G., Latham, J., Lozano, N., Greene, J., "Influencing Middle School Girls to Study Computer Science Through Educational Computer Games", *Journal of Computing Sciences in Colleges*, pp 90-97, Vol 28, June 2013.

[4] Ching-yu Huang, Jing-Chiou Liou, Grace Huang, "Pathway from Learning Multimedia Software to Computer Science Education," *Proceedings of the 11th International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS'15)*, Las Vegas, Nevada, July 27-30, 2015.

[5] Philip Sheridan Buffum, Allison G. Martinez-Arocho, Megan Hardy Frankosky, Fernando J. Rodriguez, Eric N. Wiebe, and Kristy Elizabeth Boyer. "CS Principles Goes to Middle School: Learning How to Teach "Big Data"", *Proceedings of the 45th ACM Technical Symposium on Computer Science Education* (*SIGCSE*), pp 151-156. New York, NY, ACM, 2014.

[6] Daniel Frost , Anita Verno , David Burkhart , Michelle Hutton and Karen North. "A Model Curriculum for K–12 Computer Science Level I Objectives and Outlines", Computer Science Teachers Association (CSTA), pp 2-44, 2009.

[7] Marie desJardins and Susan Martin. "CE21–Maryland: The State of Computer Science Education in Maryland High Schools", *Proceedings of the 44th ACM Technical Symposium on Computer Science Education* (*SIGCSE*), pp. 711-716, Denver, CO, March 6-9, 2013.

[8] Bargury, O Muller, B Haberman, D Zohar, A Cohen, D Levy, R Hotovel. "Implementing a New Computer Science Curriculum for Middle School in Israel", *Proceedings of the FIE'12 Conference*, Seattle, Washington, pp 886-891, October 3-6, 2012.

[9] "What jobs are available in the computer industry?" http://www.computerhope.com/issues/ch000764.htm

AUTHOR INFORMATION

Ching-yu Huang, Assistant Professor, Department of Computer Science, Kean University.

Richard Olszewski, Junior student, Department of Computer Science, Kean University.

Rocco Tomazic, Superintendent, Freehold Borough School District, Freehold, NJ.

EIT Digital Doctoral School: A new program for ICT innovators

Anders Flodström
EIT Digital
KTH Royal Institute of Technology

Maurizio Gabbrielli
EIT Digital
DISI, University of Bologna
INRIA, Focus Team

***Abstract*—We describe the European EIT Digital Doctoral School, whose mission is to educate tomorrow's leaders and innovators in digital technologies by combining excellent technical programs with deeply embedded Innovation and Entrepreneurship education.**[1]

I. Introduction

The last thirty years have witnessed a dramatic change in the perspective of doctoral level education in the world. Such a change is due to the following facts:

1) The percentage of doctoral graduates among the population is constantly increasing, as shown in figure 1.
2) The percentage of doctoral graduates who can find a job in Academia is dramatically decreasing: some indications can be obtained from figures 2 and 3, as well as from the data in [7].
3) The number of young researchers working in the private sector is increasing in many countries: for example, the data for France are given in [6] and shown in figure 4.

While the first fact is quite simple to explain, since the level of education in general is increasing, the second one is more complicated and has to do with societal, economic and also technological issues, whose analysis is beyond the scope of this paper. It is however a quite clear trend, which will probably last for many years. The third fact is somehow related to the second one, even though the relation is more complicated that a simple cause-effect one.

These facts have the quite obvious consequence that, for an increasing number of persons, a Ph.D. degree is now seen as a gate to high level positions in the industry, both in large companies and in new startups. And such a consequence, in its turn, has a direct corollary which can be stated as follows: "Universities should also do more to help their Ph.D. students to gain skills and contacts that will come in handy beyond academia" [5]. In particular, it would be very important to provide to Ph.D. students, together with a good technical formation, some basic education on innovation and entrepreneurial activities.

Despite this quite clear evidence the current situation in Europe in not completely satisfactory for ICT. In fact, on one hand there exist traditional Ph.D. programs on ICT disciplines which provide excellent technical content but do not touch on innovation and entrepreneurial aspects. On the other hand, these aspects are provided in business oriented programs, which however fail to provide the technology mastering which today is needed to successfully operate in the innovation area of ICT.

There is then an increasing need for Industrial Doctorate programs which are able to coniugate technical education with business and innovation oriented training. This is by now clearly recognized at EU level, where indeed specific programs of the Marie Sklodowska-Curie Actions (MSCA) framework are targeting this need. In particular, the European Industrial Doctorates within the Innovative Training Networks program (ITN-EID) aim to provide a doctoral level education to "a new generation of creative, entrepreneurial and innovative early-stage researchers, able to face current and future challenges and to convert knowledge and ideas into products and services for economic and social benefit". In such an ITN-EID program it is explicitly required that the Ph.D. candidates receive a supplementary training on Innovation and Entrepeneruship (I&E) aspects and that, together with the usual academic advisor, they are mentored also by a person from the non-academic environment.

These requirements are exactly those which, since 2012, are the basis of the EIT Digital Doctoral School, where students receive a technical education in one of the 18 partner universities in Europe, while the I&E education is provided in our 7 Doctoral Training Centres located in five different countries in Europe.

Exploiting synergies with other European programs, such as the MSCA ITN-EID, we believe that the EIT Digital Doctoral School can play a unique role in generating European ICT leaders that can face the challenges of ICT innovation in Europe and drive the process to achieve a better quality innovation, thus contributing to Europe's competitiveness and growth.

The remaining of this paper is organized as follows. Section II presents in some more detail the structure of EIT Digital Doctoral School, focusing on the organization of I&E Education and providing some evidence of the value of our mode coming from our industrial partners and our students, while Section III contains some closing remarks.

[1] The first author is Education Director of EIT Digital, the second one is the Director of the EIT Digital Doctoral School.

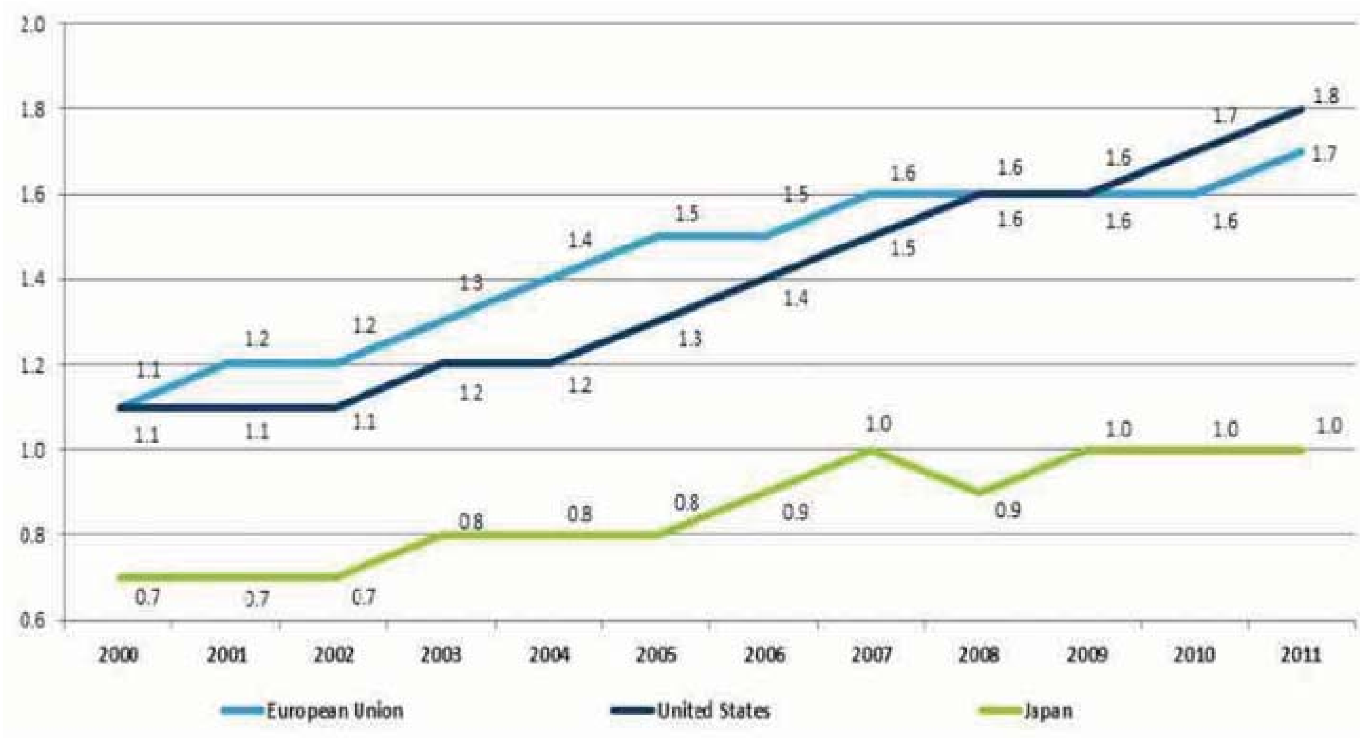


Fig. 1. New doctoral graduates (ISCED 6) per thousand population aged 25-34, EU, US and Japan, 2000-2011 (UNESCO, EUROSTAT, OECD).

II. Structure

EIT Digital [1] is a Knowledge and Innovation Community (KIC) of the European Institute of Innovation and Technology (EIT) [2]. As a such, the mission of EIT Digital is to lead European digital innovation and entrepreneurial education, thus driving Europe's digital transformation. It does this by mobilising an ecosystem of over 130 top European corporations, SMEs, start-ups, universities and research institutes and using a pan-European network of Co-Location Centres (CLC) in Berlin, Eindhoven, Helsinki, London, Paris, Stockholm, Trento, as well as in Budapest and Madrid.

EIT Digital is focused on entrepreneurship and is at the forefront of integrating education, research and business by bringing together students, researchers, engineers, business developers and entrepreneurs, focusing on the following Europe's strategic, societal challenges: Digital Industry, Digital Cities, Digital Wellbeing and Digital Infrastructure.

The EIT Digital education programme includes a Master School, a Doctoral School and a Professional School. Here we focus on the Doctoral School (DSL), whose mission is to educate tomorrow's leaders and innovators in digital technologies by combining excellent technical programs with deeply embedded Innovation and Entrepreneurship (I&E) education.

The technical program is carried out in a partner university according to the local rules. We have currently 18 partner universities in six different European countries (Hungary, Finland, France, Italy, Sweden and Spain), hence rules and requirements for Ph.D. education are quite different. However all the Ph.D. programs of these universities have a very good quality, as assessed by standard international rankings.

On the other hand the I&E education, which is the main activity our Doctoral School, is organized and coordinated by the central Doctoral School office in such a way that, despite the different implementations in various parts of Europe, we can maintain a uniform level of content and quality. More precisely, the I&E education is delivered in our Doctoral Training Centers (DTCs): these are places, located in EIT Digital CLC, which provide geographic and thematic focus and aim to create a critical mass of doctoral candidates, in a single place, around a few single themes or technologies and surrounded by an ecosystem of involved industries. There are currently 7 DTCs in Budapest, Helsinki, Madrid Paris, Rennes, Sophia-Antipolis and Trento, and each one of them is closely

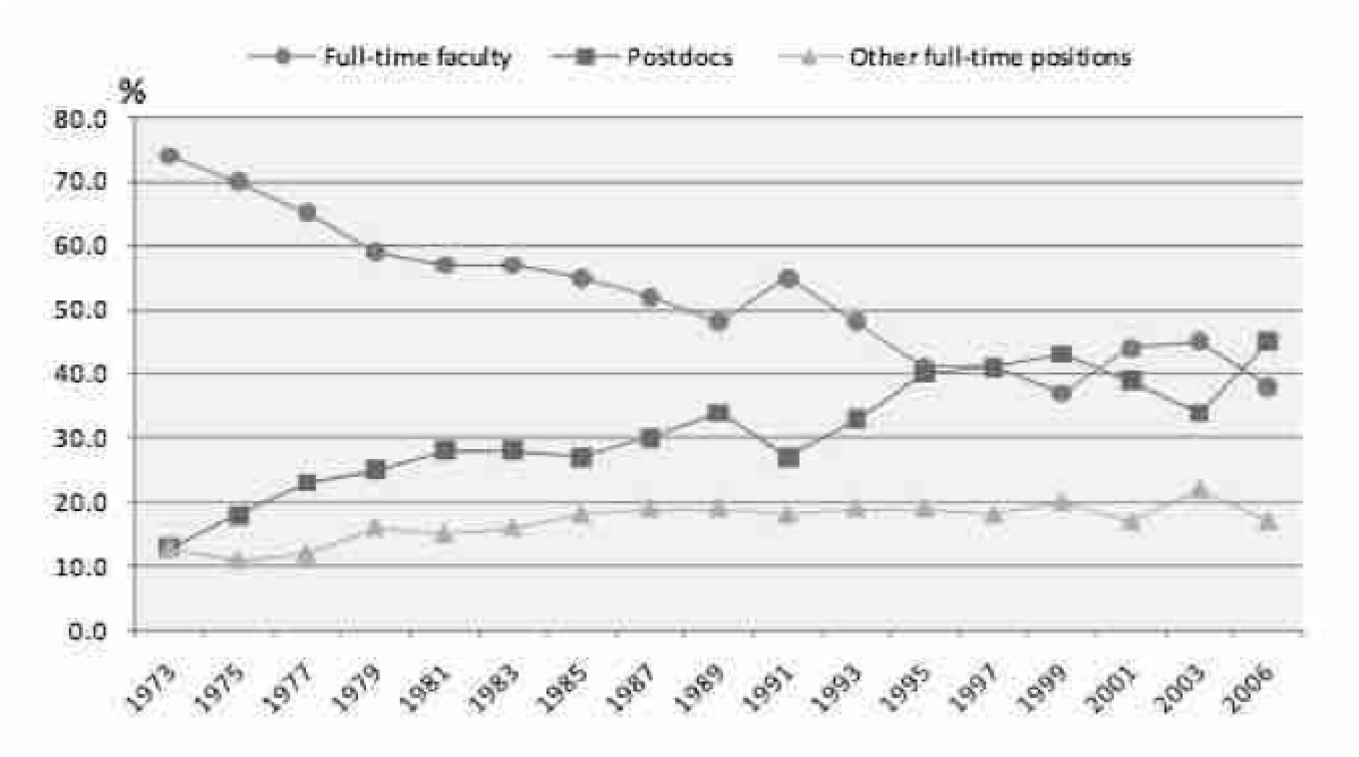


Fig. 2. Trends in employment of S&E Ph.D. graduates in the US 1973-2006 (NSF).

connected to some local partner universities and companies which support its activity. It is also worth noting that in the colocation centers there are ongoing project activities, supported by EIT Digital, in the context of the previously mentioned strategic areas (Digital Cities, Digital Health, Digital Industry and Digital Infrastructure). Being in the colocation centers then our students can be exposed directly to relevant projects of our partner companies.

The I&E education consists of three main components:

- Four courses to be attended during the doctoral studies;
- One period of six months of mobility abroad;
- A six months period of internship in an industrial environment, called Business Development Experience (BDExp).

The courses provide a specific education toward the innovation and the business world which usually is not contained in the standard Ph.D. technical programs in ICT. We have the following four modules:

Raising Awareness (RA). Typically a two days course introducing the EIT Digital DSL education and discussing mainly the methodologies to assess the value of research in innovation, trying to focus on customer centric perspectives.

Opportunity Recognition (OR). This a five days course which address the problem of identifying value elements in research and relating them to innovation and business activities, thus answering to the needs of industry, customers and more generally society. Initial aspects related to concept design and business modeling are also considered.

Business Modeling and Development (BMD). This course usually span over 15 weeks (part time) and address the typical aspects related to business modeling and planning, with a particular focus on ICT areas. We also address aspects related to the ability to exploit business value elements in research and technology and we touch on communication skills.

Growth and Harvest (GH). Usually a one-two weeks course covering mainly the application of concepts, methods and tools pertaining to business development, to manage and finance the growth of a company and to exploit innovation.

The second component of our education is a period of geographical mobility of six months, preferably in a partner university or company in Europe, in which the students have

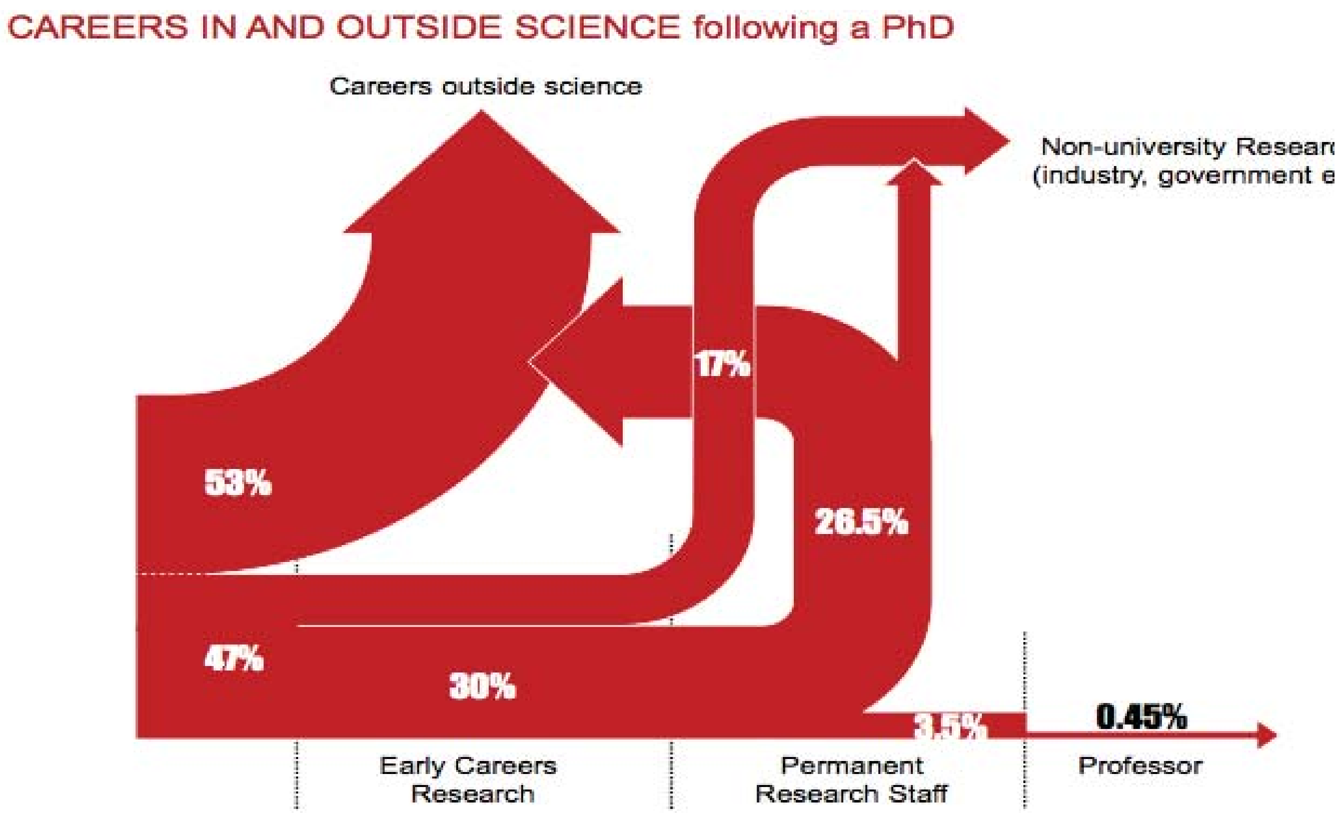


Fig. 3. Careers in and outside science in UK (Royal Society).

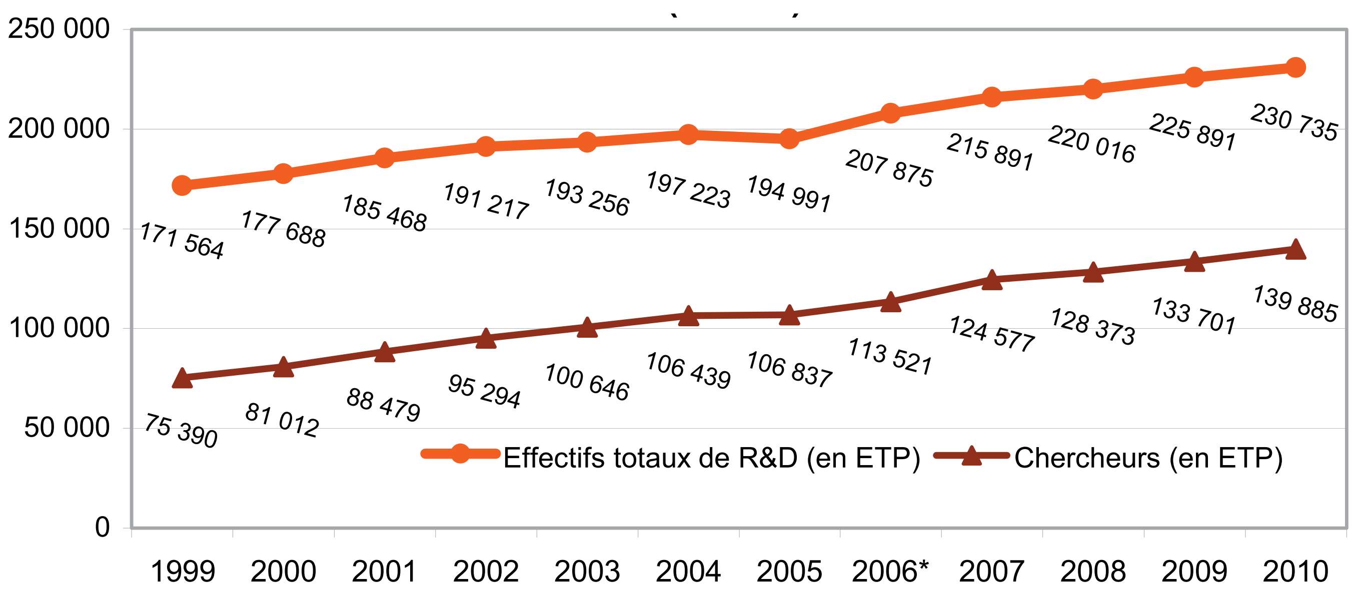


Fig. 4. Researchers in private companies in France. Data in FTE. Source: MESR-DGESIP/DGRI-SIES C1.

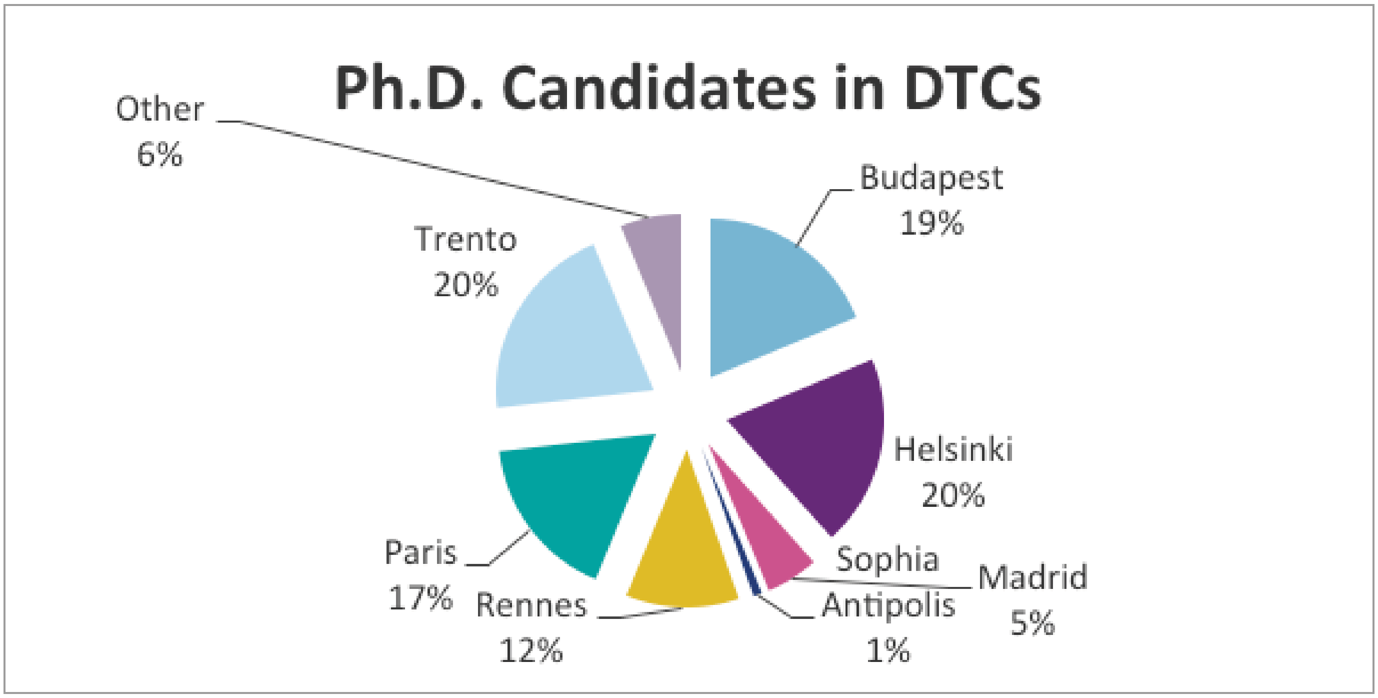


Fig. 5. Distribution of students per DTC.

a chance to visit other laboratories and to get in contact with different research and innovation environments.

Vey important for our programme is also the Business Development Experience (BDExp): this is a six months period which lets doctoral candidates or fresh doctors be immersed (for at least six months) in a real business-oriented context. The BDexp may be implemented as an internship in some company, or as a pre-incubation stay in an innovation friendly place, or even by starting-up a new venture. The BDexp should yield a written document which reports on the learning outcomes of this period, which should include skills in applying, synthesizing, and evaluating prior I&E studies, as well as new skills learned during this period. In the ideal case the BDExp should be linked to the doctoral thesis, so that the technical results obtained during the ph.d. studies can be linked to the business world. While in the past the BDExp was done after the completion of the normal Ph.D. activity (usually three years), we are now switching to a more integrated model, where the immersion in the industrial context starts from the very beginning of the doctoral studies. This has the advantage of a more direct contact between the research environment and the industrial one, with the possibility of receiving – in both directions – more direct feedbacks, thus better exploiting the synergies of the two different worlds.

A. Some numbers

The EIT Digital Doctoral School was started in November 2012 and since then has constantly augmented the number of students, now reaching a really European dimension, with five countries and 18 universities involved, around 130 students – with 63% of them coming from Europe and 21% women – and targeting at 200 students in two years.

The Figure 5 shows the distribution of the students in the DTCs.

The first six students who finished our program graduated in April 2016 an we expect some more twenty students to graduate by the end of 2016. Thirty-five Ph.D. candidates have been doing their mobility in fourteen different countries, including US, and around fifteen students are about to finish their Business Development Experience (BDExp).

B. Some results

All the students who have finished or are finishing our program have received a positive feedback from the labor market: some of them have been employed in relevant positions in R&D labs of a major companies in Europe, while some other successfully created their own startup (six startup have been already created in Finland, France, Hungary and UK).

Large companies appreciate our graduates for the technical mastering of new digital technologies, combined with the ability to turn them into business opportunities. The following is a quote from the CTO of IT R&D Division and Director of Central Software Institute Europe, at Huawei European Research Center [3]:

> Huawei's European Research Center (ERC) aims to become a key technology player in the European ecosystem. In execution of this mission, Huawei is building new and innovative technologies in collaboration with its customers. The scope of the EIT Digital Doctoral School of providing innovation-related education is important for both Huawei and

> for the innovation agenda of the EC. Through the EIT education Radu[2]acquired valuable skills for managing and driving innovation which contribute well to our day-to-day business of creating advanced technologies and establishing new partnerships for joint innovation with European customers.

Also our graduates are appreciating our program: the following is a the opinion of one of them, who is now starting his own startup [4]:

> The EIT Digital Doctoral School programme was a very important training. It developed the skills and the knowledge that I needed for starting a business and that I could never have obtained otherwise, in a ”standard” PhD programme. Thanks to the EIT DSL, I am still a scientist, but now partly a CEO as well.

III. Conclusions

Our first graduates are already actively working as innovators in the R&D divisions of major companies and in their own start-ups. These examples confirm the added value of our students: a deep technical mastering of new digital technologies, combined with strong problem solving skills, and the ability to turn them into business opportunities. We are now fine-tuning our programme in order to find the right balance between academic and industrial education and we are also working on the integration with other national and European programs in order to build up thematic, permanent networks for industrial PhD education in Europe. As for other educational programs, one of our main challenges is now to measure the impact of our Doctoral School at societal level.

References

[1] EIT. https://eit.europa.eu. Last retrieved in March 2016.
[2] EIT Digital. http://www.eitdigital.eu/. Last retrieved in March 2016.
[3] Goetz Philip Brasche. Personal communication, 2016.
[4] Wilfried Dron. Personal communication, 2016.
[5] Editorial. There is life after academia. *Nature*, 513(7516), 2014.
[6] Direction générale pour l'enseignement supérieur et l'insertion professionnelle. Direction générale pour la recherche et l'innovation. *L' état de l'emploi scientifique en France*. http://cache.media.enseignementsup-recherche.gouv.fr/file/Personnels_ens._sup_et_chercheurs/41/6/EES2013_261416.pdf, 2013. Last retrieved in March 2016. In French.
[7] The Royal Society. *The Scientific Century securing our future prosperity*. 2010. ISBN: 978-0-85403-818-3.

[2]Radu Tudoran is the first graduate of EIT Digital Doctoral School, now working at Huawei European Research Center.

MakerView: An integrated camera-monitor network for promoting collaboration in educational Makerspaces

Samuel Woolf , Ethan Danahy,
CEEO, Tufts University, Medford, MA, USA

Abstract - *MakerView, a new technology development from Tufts University, utilizes an integrated network of cameras and monitors to promote participation, collaboration, and evaluation amongst educational Makerspaces. Previously, these spaces have functioned in isolation, which is a juxtaposition to the general Makerspace ethos, as the Maker community is often the pinnacle of collaboration. By providing a glimpse into the everyday happenings of these spaces, MakerView has the potential to greatly increase participation, both in person and virtually, in these Makerspaces. The MakerView project is comprised of three key technical elements: remote camera nodes that can easily be installed anywhere, strategically placed TV monitors that give an observer a window into a space, and a central web-server and user-interface for system coordination. Leveraging low-cost Wi-Fi enabled microprocessors, MakerView utilizes a cloud based network, connecting previously disjoint spaces, projects, and communities.*

Keywords: Makerspace, collaboration, community, Internet of things, education

1 Introduction

In a world that is becoming increasingly connected, it is important to leverage new technologies to support collaboration. MakerSpaces, which pop up in schools, libraries, and businesses worldwide, are a prime example of twenty first century collaboration. They are hotbeds of human to human idea sharing, skill teaching, and product development [1]. MakerView enhances the incredible human centric benefits of MakerSpaces, using a connected IOT network to promote education, collaboration and participation.

2 System Overview

MakerView has a clear goal: improve the inherent collaborative and educational benefits of MakerSpaces by leveraging the capabilities of low-cost, internet enabled technologies [2][5]. MakerView strives to use visual and contextual information to virtually connect these spaces, thus bolstering their appeal and utility [9]. We believe that by pushing pertinent information to participants, we can passively enhance user behavior, increasing the benefits of makerspaces for current makers, and increasing participation among potential makers. Additionally, moving the modern MakerSpace into a more public view has the potential of furthering collaboration and education [3][4].

2.1 Technical Implementation

On a conceptual level, MakerView is comprised of three key elements; MakerView Nodes, MakerView Access Points, and the MakerView Server. First, a Node is comprised of an internet enabled camera and can be installed in any room around a campus. These units have been specifically designed for easy setup, so one can dynamically add any number of camera Nodes to a MakerView network. Second, Access Points take the form of connected TV monitors that are installed in strategic locations around a campus. By displaying a series of images and videos pulled from the MakerView cameras, these 'digital windows' give a peek into Makerspaces, motivating a casual passerbyer to engage in the growing community. Third, the MakerView Server houses a repository for the Camera Node data, and pushes new content to a central web interface where users can access all of the information. The website view provides users with real time Makerspace data as well as historical information catered to researchers and management.

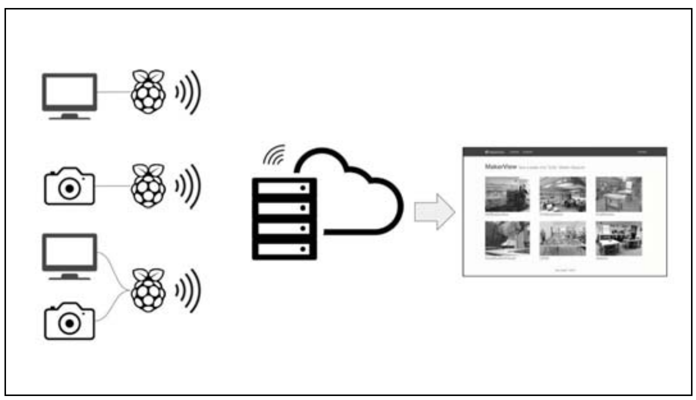


Figure 1. A diagram displaying the MakerView system design, highlighting Camera Nodes, Access Points, and joint Camera-Monitor nodes. The data is sent to a cloud storage server, which, in turn, pushes new content to the web interface.

MakerView's current implementation focuses on easy, low cost scalability, maximizing the use of off-the-shelf components and reproducible designs. Each Camera Node is inexpensive and can be quickly installed. Comprised of a Raspberry Pi, a PlayStation Eye camera, and configurable software, a new Camera Node costs under $50 and requires less than 15 minutes of setup time. Similarly, a new Access Point is composed of a TV monitor connected to an internet enabled computer (the Tufts MakerView implementation uses a Raspberry Pi). These monitors constantly display an Access Point specific website page, which includes intelligently compiled content, determined by both Access Point location and software configuration (i.e. easy to include bus schedules, announcements, etc.). Both the Camera Nodes and the Access Points have been strategically designed to run on the same core architecture. Thus, when adding another physical component to the system, one can install either a freestanding Camera Node or Access Point, or combine the two into a multifunctional station. The data is stored on the cloud via an on campus server, running a version of the free, open source cloud storage software, OwnCloud [10].

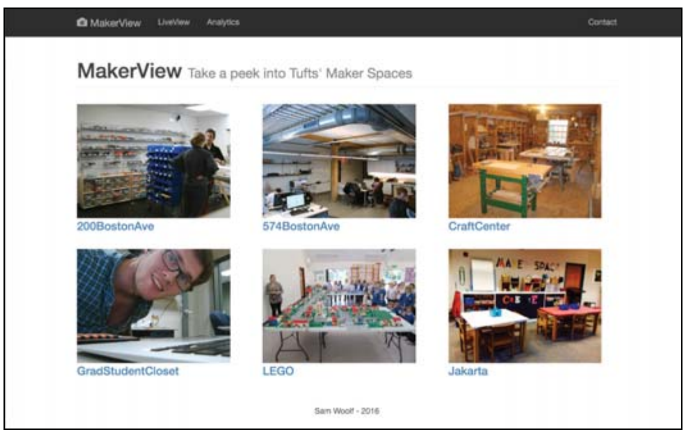


Figure 2. An image of the MakerView's web-interface front page. The website allows the user to see the current status of a community's Makerspaces.

3 Initial Impact

A prototype of the MakerView system has been implemented on the Tufts University campus, and is currently running, connecting spaces and collecting data. The university supports a variety of Maker facilities spread across the campus, creating a spatial divide that affects how makers use the spaces. Often a student will simply head towards the geographically closest MakerSpace without even considering the benefits (different tools, different people) that the other MakerSpaces could provide. By installing MakerView in just two of the spaces on campus, students now have a better understanding of the full Maker Network, viewing it as a conjoined service as opposed to autonomously functioning spaces. This connection has the potential to boost both interpersonal collaboration and space utilization. Now, we are in the process of scaling MakerView, first to the other spaces on campus, and then to other institutions. As it scales, the true collaboratory and educational benefits of this technology begin to emerge.

Already in this nascent form, MakerView provides substantial benefits. The network has already altered how management views the spaces. Previously, estimations of attendance and hours of use had been based on hearsay. Now, with the new data provided by MakerView, definite answers to these questions can be obtained. In examining one week of data, we found that fewer people utilize the spaces than had been previously estimated. Additionally, the spaces are underused, remaining closed for a large percentage of the week. Armed with this new information about actual use, the Tufts maker community has begun an effort increase Makerspace utilization.

Additionally, MakerView has the potential to lower the barrier of entry for new students thinking about utilizing the infrastructure, tools, and social networks inherent to the Makerspaces [11]. Either through the Access Points or web interface, a prospective maker now has a full mental image of the space, and immediately begins to picture him or herself working in the facilities. Now venturing across campus to a new area is far less daunting, as the student already has an association with the space.

4 Conclusions and Future Work

With the successful launch of a minimum viable product, we eagerly look towards the implications of future developments. We are now working on a key element of MakerView: analytics. The existing network supplies a large quantity of high quality image and audio data. With this data, MakerView will be able to report a wealth of useful statistics and feedback. By monitoring space usage over time, the community will be able to better understand which locations and times of day/week/year are the most popular, and allocate resources accordingly (i.e. more staff, change in hours of operation). Additionally, Makerspace organizers will be able to get instant feedback on pilot studies (i.e. does the additional 3D printer entice more students).

There is a strong want for user-specific information that may not be conveyed by the existing image data. A data visualization tool is currently under development to allow users to glean a greater understanding of Makerspace status from the web interface [8]. This tool will allow users to see who is currently using the room (i.e. a trainer for tool usage, or an on-duty proctor), as well as give insight into tool availability. Additionally, a user could be notified when a specific team or project occupies a space.

The next iteration of MakerView will be accompanied by an increased testing effort. Only through deeper analysis of the data collected by the MakerView system will its benefits be proven. Thus, these subsequent tests will focus a quantitative analysis on increases in user participation and user collaboration (both intra- and inter- Makerspace)

At the moment MakerView connects people and spaces by giving users access into a remotely located makerspace. This virtual connection provides many aforementioned benefits. In time, MakerView hopes to enhance the interactions catalyzed by this virtual connection, providing a method for space-to-space collaboration [7]. These improvements open up possibilities such as enabling remotely located designers to co-design an idea in physical space. The current model of Makerspaces draws like-minded people together in a single location to share ideas and collaborate. MakerView strives to provide this 'local community experience' to individuals with shared interests regardless of physical location [6].

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 1119321. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

5 References

[1] Martin, Lee (2015) "The Promise of the Maker Movement for Education," *Journal of Pre-College Engineering Education Research (J-PEER)*: Vol. 5: Iss. 1, Article 4.

[2] Kuznetsov, S., & Paulos, E. (2010). Rise of the expert amateur: DIY projects, communities, and cultures. Proceedings of NordiCHI, Reykjavik, Iceland.

[3] Papert, S. (1993). The children's machine: Rethinking school in the age of the computer. New York, NY: Basic Books.

[4] Ito, M., Gutierrez, K., Livingstone, S., Penuel, B., Rhodes, J., Salen, K., … & Watkins, S. C. (2013). Connected learning: An agenda for research and design. Irvine, CA: Digital Media and Learning Research Hub.

[5] Schrock, Andrew Richard (2014) "Education in Disguise: Culture of a Hacker and Maker Space," *InterActions: UCLA Journal of Education and Information Studies*, Vol. 10: Iss, 1.

[6] Kurti, R Steven; Kurti, Debby L; Fleming, Laura (2014). "The Philosophy of Educational Makerspaces Part 1 of Making an Educational Makerspace," *Teacher Librarian*, Vol. 41: Iss 5.

[7] Brave, Scott; Ishii, Hiroshi; Dahley, Andrew (1998). "Tangible interfaces for remote collaboration and communication," CSCW98: Computer Supported Cooperative Work.

[8] Carleton, Allison A.; FitzPatrick, Catherine M; Pommier Theresa M; Schwartz, Krista S. (1997). "Remote collaboration system with annotation and viewer capabilities" Patent US 6061717 A, Mar 1997.

[9] Hlubinka, Michelle; Dougherty, Dale; Thomas, Parker (2013). "Makerspace Playbook: School Edition," Maker Media

[10] Owncloud Inc., OwnCloud, 2011. Web. 12 May, 2016. <https://owncloud.com/>.

[11] Stockman, Angela. *Documenting Learning in the Makerspace: A Peek Inside Our Process at Roy B. Kelley School*, 2016. Web. May 12, 2016.

Improving an online course that teaches Human Computer Interaction using Quality Matters through informal reviewers

Suhair Amer
Department of Computer Science,
Southeast Missouri State University, One University plaza, Cape Girardeau, MO, USA 63701
samer@semo.edu

Abstract- *In this paper we discuss the steps followed to approve and online course teaching Human Computer Interaction. In addition, we provide brief explanation of what have been done to satisfy each of the standards required by Quality Matters to approve the course through informal reviewers at our institution.*

Keywords: Quality matters, human computer interaction

1. Introduction

Quality Matters (QM) [QM] is a process used to improve the quality of online courses. It involves the use of a rubric and peer review to evaluate the quality of the design of the course. It has been developed by the National Standards, Research Literature, Best Practices and Instructional Design Principles. It lists 40 standards, divided into eight areas. Even if the course meets QM, it still required continuous improvement.

The reviewers are trained before performing a review. They provide feedback to the course's instructor with suggestions of how to improve the course. The instructor then can improve the course so that it eventually meets QM standards. There are two types of reviews. Official reviews gain QM recognition and have specific requirements for the team members. One member must be from outside the institution and another must be an expert in the subject matter. Informal reviews are completed at an institution and are performed by trained reviewers from the institution. This type of review does not gain QM recognition. In both types, all reviewers must be certified by QM.

2. Process of approving the course

To start a university wide attempt to have almost of our online courses meet QM standards, such decision was explained during university wide talks to faculty. Handouts of information about Quality Matters along with the implementation plan were sent to faculty members and administrators. Faculty were reminded in meetings about the importance and the deadlines related to approving different courses. Faculty attend a 5 week workshop, taught once a week, discussing different techniques to be used to build or improve an online course using the QM rubric. The courses were taught by QM instructors. Upon completion, faculty received a certificate of completion. Faculty then worked on developing or updating their courses to meet Quality Matters rubric. Faculty receives support and training. Once the course is submitted, a reviewer is assigned and he/she will send, once finished with the review, the score and any comments and suggestions back to the faculty. If the course has met Quality Matters standards, no additional action is required from faculty and he/she can indicate that it has been approved to students. If it did not meet, the faculty is given the chance to make changes, or provide feedback regarding they are not updating an entry and then resubmit. This iterative process is complete once the reviewer indicates that the course with the new changes has met the standards.

3. Describing the courses

The course covers and discusses Theory and practice of human computer interaction; designing, implementing and evaluating human computer interaction. The first objective of the course is that the student will define interdisciplinary concepts in the field of Human-Computer interaction. The second object is that the student will apply the Human-Computer interaction concepts into building and evaluating interactive systems. The third objective is that the student will critically analyze and prepare reports on Human-Computer interaction topics. We are currently using [Rogers et al. 2011] as the text book for this course. We cover all chapters of the book. The students need to complete 10 quizzes and 10 reports covering different HCI aspects, theories, and techniques. Towards the end of the semester the students are required to design, implement and evaluate a system. Designing a system with a good interface is very important as well as evaluating the system which involves subjects/users.

4. Actions to satisfy standards

In this section we will discuss what have been done to satisfy each of the standards for the online course. Our institution utilizes Moodle for the delivery of courses. We have used weekly format, where units and their corresponding activities are published and should be completed weekly. Moodle provides this information in blocks with a main block appearing at the top of the Moodle page. For each unit/week, we list the objectives of the unit/week. We list resources and what should be completed in that week. Activities may include an assignment, quiz, forum discussion, paper, report, or presentation. In the main block we list the syllabus, course objectives, faculty information, policies, etc.

4.1. General Standard 1

"The course overview and introduction set the tone for the course, let learners know what to expect, and provide guidance to ensure learners get off to a good start." [Rubric]

To satisfy this standard, we placed in the main block many relevant information such as a getting started page or section. It also states the pre-requisites for the course. There is also a "News forum" which should keep any announcements made to the class. The students are recommended to visit it every time they log on just to make sure they have received any recent announcements. It has been recommended that we should use consistent terminology to refer to sections. It was also recommended that we provide an explanation to what " news forum" is used for especially for new students. Since this is an online class, sometimes it is difficult to know what will student start looking at. Although, we assume that they will go in order, they in many times skip important sections. Therefore, if we have something placed at the very top of the page where it could be easily overlooked, we may consider emphasizing it using a different font or typeface or otherwise highlighting it to call student attention to it immediately.

We list the purpose and structure of the course and explain it in several webpages titled as such. Although we don't require students to engage in any online discussion, we provide rules for discussion/email conversation in our syllabus. It has been recommended that it should not be brief and we should expand our discussion of the topic or linking it to the university's page on the topic. We also provide links to all of the policies that the annotation for the Standard recommends. We devote a full page of the course to the minimum tech requirements for the course, meeting the Standard. We have a page listing the pre-requisites for the course and the required competencies. We included the minimum technical skills expected of the learner in the course. All of which meet the standard. We also have an introductory page, which includes short summary of instructor and how to get hold of him/her. It was recommended to add a photo of instructor and some personal information such as hobbies to create a connection between the instructor and the students. The first assignment involves the student posting an introduction to the rest of the students.

4.2. General Standard 2

"The learning objectives or competencies establish a foundation upon which the rest of the course is based." [Rubric]

We list the course learning objectives and course/program objectives in the syllabus that is listed under a "Getting started" header making them easy to find. The course objectives use the terms apply, define, analyze and prepare which are specific and observable terms as defined by the Standard. We used Bloom's taxonomy [OIT] to create the objectives. We list module/unit learning objectives and describe how it relates to the course level objectives. In addition, our objectives are written in a manner that makes them comparatively easily grasped by those who have mastered the pre-requisites for the course. Each of the unit objectives is linked to the course objective that it supports and we show which course objectives are met by each unit, meeting the Standard. We clearly state the relationship between objectives and activities in a format similar to (table 1).

Table1: Relating course objectives to class activities

Course Objectives /	Activities that contribute to achievement of the objective or outcome
1.The student will define interdisciplinary concepts in the field of Human-computer interaction.	1- Quizzes 1 to 10 2- final Quiz
2.The student will apply the Human-computer interaction	Three phases of project:

concepts into building and evaluating interactive systems.	1- Establish requirements and design a simple interactive system. 2- Implement a simple interactive system 3- Data analysis and Evaluation of the simple interactive system.
3.The student will critically analyze and prepare reports on Human - computer interaction topics.	1- Reports 1 to 10

The annotation for the Standard states "Expected content mastery is appropriate to the type and level of the course. Taxonomies that describe levels of learning can be helpful in categorizing learning objectives or competencies by level and in enabling reviewers to determine whether the objectives or competencies correspond to the course. For example, a first-year course is likely to include objectives or competencies that are lower in the cognitive realm than those in an upper-level course. Objectives or competencies in a lower-level course may use verbs such as "identify," "describe," or "apply," which align with assessments such as multiple-choice quizzes, essay questions in exams, or solving problems."[QM]. The terminology we used is appropriate for a junior level course in this discipline. We include more terms appropriate for an upper level course that emphasizes objectives higher in the cognitive realms such as "differentiate," "design," or "justify".

4.3.General Standard 3:

"Assessment is implemented in a manner that corresponds to the course learning objectives or competencies and not only allows the instructor a broad perspective on the learners' mastery of content but also allows learners to track their learning progress throughout the course." [Rubric]

Our course assessments measure the stated learning objectives for the week. Each unit / week we required the student to complete a quiz and a report. In the report they answer essay questions that apply the concepts covered in that unit. We provide a page detailing the ways in which points will be assigned to the different activities as the annotation for the Standard requires. Table 2 explains the rubric used to grade a report. Reports consist of short or long easy questions. Each question has a different grade depends on the length of answer required and steps to be carried out to complete the questions. We also state that no late work will typically be accepted although exceptions can be made, which is also a component of meeting the Standard.

Table 2: Rubric used to grade reports.

Student earns	If he/she
81 %- Full mark	Completes all questions Shows full understanding of concepts studied in the chapter and applies it correctly to answer the questions with minor errors (including formatting)
41 % - 80 %	Completes all /some questions Shows partial understanding of concepts studied in the chapter and applies it correctly/partially correctly to answer the questions with minor-moderate mistakes including formatting problems
1% - 40 % of grade	Completes all /some questions Shows partial understanding of concepts studied in the chapter and applies it partially correctly or incorrectly to answer the questions with major mistakes including formatting problems.
Zero marks	No answer provided

In our first design of the course, we located the rubric of each activity as part of the question. Therefore, the rubric, for example, for the project phases were placed with their description in the weeks towards the end of the semester. The reviewer pointed out that the annotation for the Standard states that grading rubrics and information should be located toward the beginning of the course. Accordingly a copy of the project grading rubric is placed as well in the syllabus.

It is required that "Multiple assessment strategies are used in both the online and face-to-face settings, including alternative assessments that require learners to apply what they learn and to think critically."[Rubric]. We utilized 4 different methods of assessments during the course: forum, reports, quizzes and term project.

Since it is more effective if the student receive frequent timely feedback, I make sure that weekly activities are graded within 3 to 5 days after the due date. This has been indicated in the syllabus and students are sent a notification every week to check their newly updated gradebook. Reports and project phases require manual grading whereas some of the questions on quizzes can be automatically graded and hence will provide immediate feedback.

4.4.General Standard 4

"The focus of this Standard is on supporting the course objectives and competencies, rather than on qualitative judgments about the instructional materials." [Rubric]

Instructional material is carefully chosen to contribute to the achievement of the stated course and module/unit learning objectives. In online courses, students heavily depend on the text book used. In addition, I make sure to have additional resources available for students to access. Still every semester, more resources are added and we make sure that the already supplied resources are checked and updated as needed. In addition, the course clearly explains both the purpose of the instructional materials and how they are used in learning activities. For example, we include a note similar to the following for each week:
How to prepare and Resources:
1- Read Chapters 1 and 2
2- Check additional web resources, assignment comments, teaching material (slides) and case studies by visiting the "Resources" and/or "case studies" tabs.
3- Complete quiz
4- Complete Report

It is very important also to cite all instructional materials. This includes citing the book in the syllabus and any other additional resources. In a fast changing field, we update the text book and all resources as newer ones come available. Also, we make sure the different and variety instructional material is up to date. We use an appropriate textbook for the course and provide access to the publisher provided materials and other additional resources, which incorporates a variety of instructional materials. We also differentiate between what is required and what is optional. This is done at the beginning of the semester and also repeated again every time it is mentioned.

4.5.General Standard 5

"Course components that promote active learning contribute to the learning process and to learner persistence." [Rubric]

Activities scheduled were carefully picked to promote the achievement of course objectives. Since this is the first course that introduces Human Computer Interaction in our curriculum we had to start from the lower levels of the Blooms taxonomy. For the first weeks of the semester. As the semester advanced, more activities would require the student to apply techniques and theories studied. The students, for example, were required to complete a quiz measuring their comprehension of a theory. Then they were required in the report/ assignment to apply the theory to complete a small project. Finally, towards the end of the semester, they were asked to analyze, design, implement and evaluate an interactive system following the theories they covered throughout the previous weeks.

It is required to have learning activities that would provide opportunities for interaction that support active learning. There are three types of interaction: learner-instructor, learner-content, and learner-learner. With our interaction with the student via forum and email correspondence and the students working individually by reading the book, additional resources and online modules to grasp the material we satisfied the first two types of interaction. We lack allowing learner-learner interaction in this course. We do ask students to introduce themselves to each other at the beginning of the semester. We allow students to engage in "Hallway" forum discussions but they are not graded on such activity and many decide not to use it. Since, our course objectives do not ask us to ask students to work in groups; we are not forcing them to do it. This is because if we do ask them then we should make sure that they know what it means to work in a group and how to do it.

With regard to our plan for response time and feedback, we clearly indicate what students should expect in terms of response time and feedback and have this information in the syllabus. We clearly state what is meant by learner interaction, devoting an entire page of the syllabus to an explanation of it.

4.6.General Standard 6

Examples of tools that can be used are discussion boards, chat rooms, grade book, social media, games, whiteboard, wikis, blogs, etc. At our university, we are using Moodle and we use most of its functionalities to deliver the material and for students to complete and submit their work. In addition, the department allows current students to download tools and program from Degree Works [Degree]. For their project, they utilize tools that are available for them for free. The course also recommends use of a laptop. We can suggest where students may rent or borrow a laptop if students do not have one.

One advantage of using Moodle, which is used by the university, is that the technologies used are kept current, meeting the standard. Also, since we are using the publisher provided materials as a component of the course, providing a link to the publisher's privacy policy would be appropriate. We also provide a page devoted to course privacy policies in general.

4.7.General Standard 7

"It is important to ensure online learners know they have access to and are encouraged to use the services that support learners at the institution. In the Learner Support Standard, four different kinds of support services are addressed: technical support, accessibility support, academic services support, and student services support." [Rubric]

The course's website offers a full page directing students where they can go to find technical support and other online courses. In that page we include information about technical support, describing how to get online help and after hours help with complete contact information. "Learning support" page describes hours of operation of library, writing labs, academic assistance, learning assistance program, and testing services. We also provide general support information such student financial services, counseling, career center, student conduct, veterans services, bookstore, advising, financial aid, billing and accounting. Technology help is essential especially when the student faces technical problems which may not be addressed by the faculty but is concerned with using Moodle and the student's device. Contact information for tech support is provided. Finally, information regarding computer labs is also provided. Although this course is offered online, students living near campus can still have access to the labs. Policies regarding their use is also included.

We provide a page discussing accessibility and Disability Services. The website also provides a link to the institution's accessibility policy. In addition to a statement that informs the learner how to obtain an institution's disability support services.

4.8.General Standard 8

"The course design reflects a commitment to accessibility, so that all learners can access all course content and activities, and to usability, so that all learners can easily navigate and interact with course components." [Rubric]

Students can easily navigate the course's website and it components. We are utilizing Moodle which is what is used at our university and by all courses. The university is constantly updating it to suit the university's needs. We used weekly layout where the student is asked to complete activities every week. Within a week block, we show the following information:

- Objective of this week/unit.
- A link relating week objectives to activities.
- Steps to how to prepare for this week and links to resources.
- Drop boxes to activities dues including their instructions.
- Link to a quiz.

We don't use many non-text materials, and aside from what is found on the website of publisher provided materials, no alternative means of delivery are needed. We make sure that when we add video or some other alternative means of delivery that we consider the needs of diverse learners. Our course is clearly laid out with consistent headings and appropriate material groups together, making the course highly readable.

5. Amendments

The course did not pass the first time it was submitted for review. It was awarded 88 out 100 which was not a passing grade. Minor changes were requested. After completing most of the amendments that we thought really will improve the quality of the course and after clarifying why some of the suggestions are not appropriate to the course, the course passed with a score of 99.

Next are some of the recommendations that the reviewers pointed out and our fixes. In addition to our list regarding "Etiquette expectations", we added the university's link under policies section which expands on that subject. We clearly added a link/page under policies section for "Criteria for the evaluation of learner's work Page". We clearly labeled a page/link for the student detailing how to track their learning progress which we named "Tracking your learning Progress - Policies regarding Grades and Feedback Page". Since we had listed optional and additional resources that a student may use in addition to the required text and material, we had to clearly label that as well. We updated and grouped all reference to material, books, resources, etc. and placed it under one section/page titled "Student textbook/ resources (what is required and what is optional) Page". We had to clearly label the following information: "Interaction", "Copyright / privacy policies ", "Disability Services/ Academic Accommodation ", "Online Support and accessibility of technologies", and "Help Desk - Telecommunications - Policies Governing the Use of Computing Facilities and Services."

6. Conclusion

In this paper we discussed the process followed to approve an online course teaching Human Computer Interaction. Brief explanation of what have been done to satisfy each of the standards required by Quality Matters to approve the course through informal reviewers at the institution has been provided. In general, this approval process did improve the quality of the course and increased the satisfaction of both the instructor and the students. It took a period of time to have the course qualified for submission, and the reviewer also took time to review the course and provide feedback. The instructor had to make additional changes depending on reviewer comments and resubmit. Finally, the course was approved and met Quality Matters and is currently offered at the university.

7. References

[QM] https://www.qualitymatters.org/

[Rogers et al. 2011] Yvonne Rogers, Helen Sharp and Jennifer Preece. Interaction Design: beyond human-computer interaction. 3rd edition. Wiley 2011. ISBN 978-0-470-66576-3

[OIT] https://oit.semo.edu/oit/InstructionalDesign/BloomsTaxonomy.aspx

[Degree] http://www.semo.edu/registrar/degreeworks.html

[Rubric] http://www.elo.iastate.edu/files/2014/03/Quality_Matters_Rubric.pdf

The Study of a Way to Solve a Problem with Design Thinking

Risa Ogawa, Sari Shinkawa, Takayuki Fujimoto
Graduate School of Information Science and Arts, Toyo University
Kujirai 2100, Kawagoe-City, Saitama, Japan
s3B101600021@toyo.jp, methyl69@gmail.com, me@fujimotokyo.com

***Abstract*—An environment surrounding people is in a continuous state of change: a change of population characteristic due to demographic aging; a technological growth; energy shortfall; a change of industrial structure. Their pace of change is accelerated. To produce something better in this society, innovating the society is necessary. Purpose of design thinking is to make an innovation happen and to provide a new value. This paper takes the research of attitude survey on problem resolution and the experiment of design thinking, and joined them to understand the consciousness of university students. In addition, it demonstrates effects of design thinking, and describes its importance and necessity.**

Keywords—Design thinking, Problem resolution, Empathy, Problem definition, Creation, Prototype

I. INTRODUCTION

An environment surrounding people is in a continuous state of change: a change of population characteristic due to demographic aging; a technological growth; energy shortfall and a change of industrial structure. Their pace of change is accelerated.

These changes will create a situation to which conventional acquaintances, methodology and concept of values are not applicable. Moreover they will shift origin of competing power and innovation. Fallibleness that does not translate well to recent model of success, knowledge and experience is increasing in the society. We need to make a new merchandise development and expanded audience by out-of-the-box innovation, and also compelling needs for the service and merchandise development to change our lifestyle are required. [1]

II. BACK GROUND

Generally, the service and articles commerce consist of "functional value" and "sentimental value."

The functional values are brought by products or services directly and means the function and performance. As in the case of a car, for example, they are methods of transport, fuel cost and safety, and it is characteristic of them to be judged from the objective point of view regarding function and specifications.

On the other hand, the sentimental value is the one for the emotional satisfaction with the product use or the one which background of service and merchandise development have. These characteristics tend to be not universal are low generality, because of the sense of subjective value of the user.

In the age with the circumstance of growing market and economy, functional value of goods and service are crucially important. People gained significant advantage on competence from steady offering excellent function at low cost with large-scale production-and consumption. In this time, people were required to make much effort to develop technologies, to improve performance of exciting goods, service and technology, and to reach better cost-effectiveness.

However, in a period of low growth and with the satisfaction of basic cravings, the value requested for goods and service greatly changed. That is the shift from the functional value to the semasiological one.

There are actual conditions with which technology and knowledge are commoditized. If people develop the perfect technique, they will get caught in a price war with low-labor-cost overseas. So, it is difficult that people continue making innovation happen by only increasing functional values.

People were apt to choose sophisticated and multifunctional goods (=high-performance goods) in past days. By contrast, with user's needs' changes, user's sense of values and life style became diverse nowadays. The semasiological values have been seen to be important, as the values that mentally enrich people's lives by use of the goods and services. Therefore, it has become important to design the values obtained from the use of goods and services. [2]

In this circumstance, "Design thinking" gets attentions.. This is a behavioral observation method for discovering potential needs. [3]

The good ideas for technology and ways of working have brought social innovations. [4] The society became dramatically affluent.

However, there are a lot of problems, for example, environmental issues and poverty issues. It is difficult to shift

ISBN: 1-60132-435-9, CSREA Press ©

to the better society by only bringing innovation of machines and the working ways. This is because it is humans to think and solve the problems, not just machines and working ways.

The innovation of life itself is necessary to change for the better society in the future. The purpose of design thinking is creating innovation by human ideas and contributing the new values to the society. [5]

In this paper, we conducted the experimental test to demonstrate effects of design thinking.

III. Purpose of research

A. What is "Design thinking"

(1) Focusing the users

There are a lot of companies that produce businesses from the viewpoints: "We can create goods by using the existing technology"; "We want to get into this market to get a lot of business chances." However, the absolute values of the products are determined by the users.

Generally speaking, for creating innovation, we need to research the overlapped part of the three elements: the business value; technological feasibility; the value for the users. In these three elements, design thinking especially focuses on the users than technology and market.

(2) The realization of the dialogue-conscious process

Design thinking puts emphasis on communication with team members. The prime reason is "speed." By communicating thickly, they can quickly advance the process, of tentative, prototype, test and improvement.

By using design thinking, people create goods and services with the dialogues with users. They focus on: empathy with the users (needs discovery); problem definition; creation (brain storming etc.); making a prototype; test. Furthermore, they define problems from the feedback test. [6]

By listening to the users' voices in each process and repeating this process many times, the goods and services that meet users' needs are created.

(3) Repeating Prototype, Test and Improvement

Design thinking gives weight to repeat a set of process. In consequence, world-class goods and services with high-usability come off.

(4)Allowance for the diverse problem resolutions and goals

There are two mindsets, "Find ideas and outputs as many as possible", and "Find methods of solving a problem as many as you can". They connect many founded ideas to the problems and resolutions. In design thinking, if many methods of solving a problem are created, it receives recognition, because it puts emphasis on "quantity and speed over quality." [7][8]

B. Sumarry of Purpose of Research

Recently, "Design thinking" has attracted attention as a method of solving a problem, and is already in use at universities and companies. [9][10] However, there is not a survey to reveal how much it is different from the today's young people's consciousness for problem resolution, and there is not a study to demonstrate the effects of "Design thinking."

In this paper, we conduct the experimental tests that demonstrate the effect of design thinking, and the survey research of young people nowadays. Furthermore, we manifests detail effects and necessities of design thinking by this study.

IV. 4. Content

This paper takes the research of attitude survey in problem resolution and the experiment of design thinking, and joined them to understand the university students' consciousness of the attitude towards problem resolutions. In addition, it demonstrates effects of design thinking, and describes its importance and necessity.

A. Research of attitude survey in problem resolution

In this paper, as premises, we held questionnaire of attitude survey for problem resolution targeting university students, and did analytical validation.

In the questionnaire, subjects were required to put their age and gender, and to answer questions about important things for them to clear a problem. By quantifying the attitude survey results.. We planned to find out tendency and shed light on the real situation of important things for them to clear a problem.

B. The Experiment of design thinking

In this paper, we carried out the experiment to demonstrate identify design-thinking effects targeting university students and grad students.

In this experiment, we made a co-gender groups of five people (including one with a user role). There were six groups; three groups who don't know design thinking and the other three groups who learnt design thinking. All the six groups were required to work on the same question, and we made an observation about the subjects, quantified the problem resolution process, made a comparison and researched the trend. We assumed that these steps would explain effects of design thinking.

V. 5. Research of attitude survey in problem resolution

A. Questionnaire

1. Put five options in order of priority to solve a problem..

① Profit

② Meeting a need

③ Speed

④ Empathy

⑤ Idea

2. Out of five choices choose one, which is important for you to put out an idea or produce something. .

① Quantity

ISBN: 1-60132-435-9, CSREA Press ©

② Relatively quantity
③ Neutral
④ Relatively quality
⑤ Quality

3. Do you improve something after completion incorporating people's responses and feedbacks?
① I make improvements many times using people's feedbacks in a positive manner.
② I make an improvement once with people's feedbacks.
③ I sometimes make improvements with people's feedbacks.
④ Neither
⑤ I never improve anything after completion.

B. Results and analysis

1) Results of "First priority"

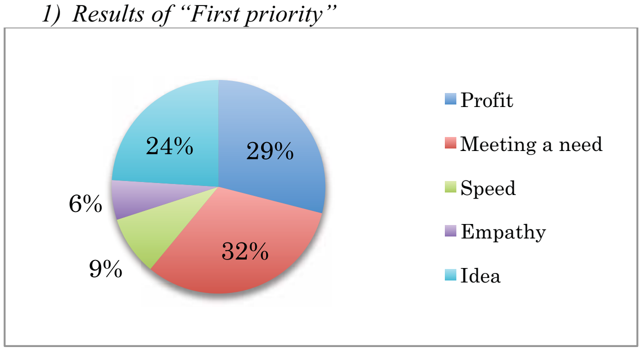


Chart1. Male Answers for "First priority"

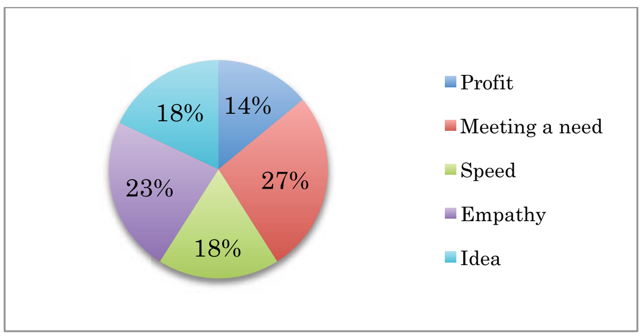


Chart2. Female Answers for "First priority"

We attempted a comparison between male and female, the highest rate was "Meeting a need."

The second highest of male is "Profit", whereas, "Profit" of female answers is the lowest. Besides, the second highest of female answers is "Empathy", on the other hand, "Empathy" of male answers is the lowest.

From these, it was found that when both sexes produce something, they have high regards for meeting a need. Whereas males are apt to attach weigh to "Profit" and give scant weight to "Empathy," females tend to attach weight to "Empathy" and give scant weight to "Profit."

2) Results of "Quantity and Quality"

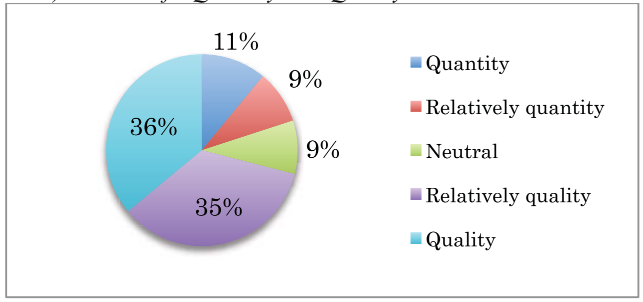


Chart3. Male Answers for "Quantity and Quality"

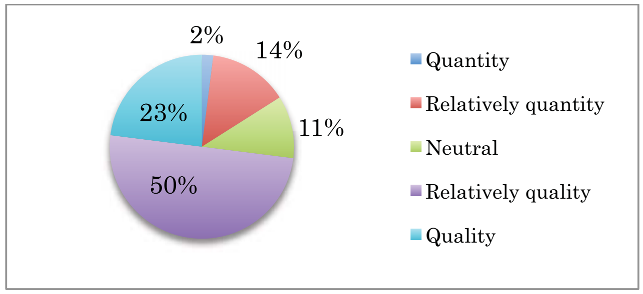


Chart4. Female Answers "Quantity and Quality"

The result of "Quantity and Quality" turned out to be high proportion of "quality" and "Relatively quality" in both male and female answers and low proportion of "Quantity" and "Relatively quantity", which is about two out of ten. From these it is evident that the vast majority of people value quality more than quantity to produce something new.

3) Results of "Presence or absence of improvement"

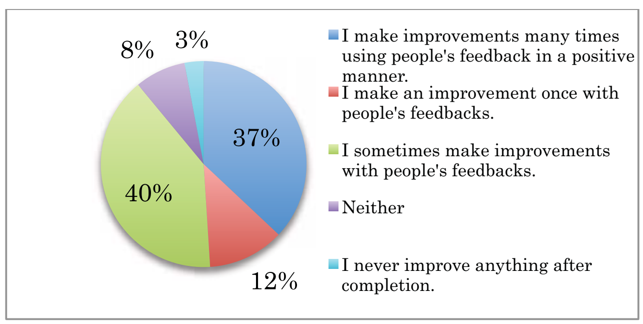


Chart5. Male Answers for "Presence or absence of improvement"

ISBN: 1-60132-435-9, CSREA Press ©

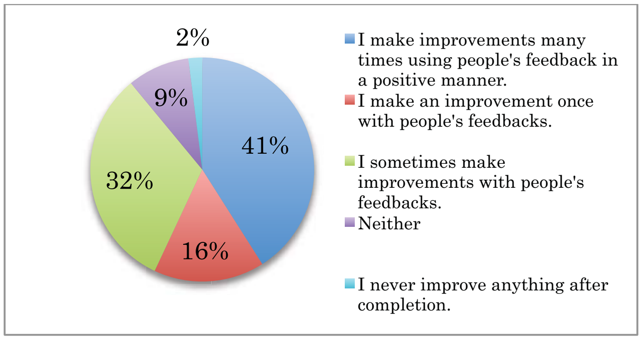


Chart6. Female Answers for "Presence or absence of improvement"

The results of "Presence or absence of improvement" ended in pretty much the same way in both men and women. We found that the number of the people (added percentage of Answer.1 and 2), who make some kinds of improvements definitely when producing something new, accounts for about half of the whole.

VI. Survey of Design Thinking

A. Investigation objecst and the experimental method

We conducted the experimental tests for 6 people per each 6 groups of 36 university students and grad students who don't know design thinking. They are 18 males and 18 females aged from 9 to 27.

One plays the part of the user in each group. 5 people except the user were required to make the new item which meets the needs of the user with 30 items from a hundred-yen store within a limited time of 60 minutes. After the test, the person of the user made evaluation for the finished items by a run of the specified points.

3 groups could use 60 minutes freely, and the other 3 groups were required to study the process of design thinking first and practice it.

In testing, we set up two cameras in the room and took the footage to record subjects' actions and acts, and examined the possibility of design thinking according to the specified points.

Table1. Process of design thinking group

1.Empathy You need to understand the users' feeling, and take interest in their life. ①Observation ②Get engaged/ Interview ③See and listen Repeat ① and ②, and drill down a little deeper ・Handle experiencing to seize a chance to visualize ・Make a list of the all information in the wall, and make relationships of information	10min
2.Problem definition Express the viewpoint as a problem definition, combing users, needs and insights Make an improvement list for the provided topics	5min
3.Creation Expand the possibilities for goods. Don't just look for correct ideas Make solutions for problem definition ・Mind map Sketch	5min
4.Prototype Make for thinking, and try for studying ・Try anyway ・Don't spend too much time for one prototype (not to have a personal attachment) ・Think with the users' eyes	15min
5.Test A chance to see the users ・Improve a prototype and a solution ・Learn about the users ・Review a viewpoint ・Request comparison to the user	5min

After this process, test subjects repeated 1 to 5, and they turned off one product.

The following are the six points of the evaluation by the user after the completion of the product.

1. Reasonability
    1. Very reasonable
    2. Moderately reasonable
    3. Neither
    4. Not reasonable
    5. Not at all reasonable
2. Practical utility
    1. Very practical
    2. Moderately practical
    3. Neither
    4. Not practical
    5. Not at all practical
3. Element of surprise

ISBN: 1-60132-435-9, CSREA Press ©

1. Very surprising
2. Moderately surprising
3. Neither
4. Not surprising
5. Not at all surprising

4. Joy
   1. Very joyful
   2. Moderately joyful
   3. Neither
   4. Not joyful
   5. Not at all joyful
5. Openness to users' feedbacks
   1. Very open
   2. Moderately open
   3. Neither
   4. Not open
   5. Not at all open
6. Score this product.

   ________ Points

We requested questioned subjects to evaluate the about above six components. We show the fractions of conclusions the results graphically, and drew a comparison between the free time group and the design thinking group.

B. The method of analysis

On the basis of prerecorded documents by two cameras, we recorded the subjects' actions and behaviors, and classified them into some components. to carry out a comparative analysis by quantification. .

Table2. Verification points

①Empathy	Conversation time with the user Number of questions for user Amount of writing in Whiteboard
②Problem definition	Number of problems Discussion time Amount of writing in Whiteboard
③Creation	Number of idea Discussion time Amount of writing in Whiteboard
④Prototype	Working hours Number of creations
⑤Test	Number of feedbacks Number of improvements Time for feedbacks
⑥Other	Time for non-specified activities Number of finished products Users' evaluations

We rounded up the results and verification for each group, stroke an average per each of the free time groups and the design thinking groups, and made a comparison.

VII. Consideration

The free time groups and the design thinking groups made a substantial difference in the experiment results of the users' evaluation.

Regarding the free time groups, there is a paucity of time for "Empathy", "Test", and "Number of questions for the users" and "Number of feedbacks." It is found that, the free time groups have less engagement with the users comparing with the design thinking groups.

The acts of the free time group are totally different from the user-centered thinking of the design thinking. Also, the free time groups spent more time for "Time for non-specified activities (time for nothing or time for observing products from a hundred-yen shops)" and "Prototype" then the design thinking groups.

The degree of completion of the finished products by the free time groups seemed to be greater than ones by the design thinking groups, because they spent more time for "Prototype." However, this degree of completion of the products didn't lead to a high acclaim. It shows that the important key is not the perfectibility of the products but the products which meet the users' needs.

Findings also shows that conversations with the users and the feedbacks from the users are important.

There are no big differences between the numbers of "Creation." Therefore, it can be said that the key point is the difference of the ideas' quality.

ISBN: 1-60132-435-9, CSREA Press ©

The free time group's "Problem definition" time, which is the basis for the ideas, is less than the design thinking groups'. New ideas occur with "Problem definition" as a basis. , Even if the number of the ideas is the same, the weakness of that basis leads to the differences of ideas' quality. Also, in design thinking, the dialogue with team members is important.

From these, it is found that there is an obvious gap between the users' assessments for the free time groups and the design thinking groups.

Next, we considered today's young people's consciousness for problem resolution.

The five points, "Profit", "Meeting a need", "Speed", "Empathy" and "Idea" are important to solve problems. From a viewpoint of design thinking, it is important that we attach great importance to "Meeting a need" and "Empathy."

Most young people nowadays think "Meeting a need" is primary. In spite of the fact that they are conscious of that, the subjects could not be regarded with high esteem by the users in results.

In the question of priority for "Quantity" and "Quality" in problem resolutions, the rate of "Quality" and "Relatively quality" is predominantly high.

From a viewpoint of design thinking, the importance is put on "Speed" and "Quantity." If people try improving the quality from the beginning, they have the potential of paying too much attention to one thing.

This is against to the process of design thinking. In design thinking, people do not spend much time on one thing. They generate as many ideas as possible, build a prototype, test and make improvements. The important thing is "Quantity." If there are many ideas, they widen the thinking and this provides better quality of ideas. Without noticing this fact, the subjects seemed to be restricted by the Japanese method with which people tend to take a fair amount of time before bringing the product to market.

Also, in the survey results of presence or absence of improvements in problem resolution, it shows that only around half of the whole make definitely use of the users' feedbacks.

In design thinking, the tests for the users and the feedbacks from the users are most important. It has been revealed that they provide a better quality of products, by survey results of the research for effectiveness of design thinking.

Thus, it has become clear that young people notice the importance of "Meeting a need" in problem resolution, however, there are the reality is that they do not know how to do.

This paper clearly demonstrated that design thinking is very important for the future generation to play important roles, under radical changes of the society surrounding us.

REFERENCES

[1] http://www.buildinsider.net/enterprise/designthinking/01

CCL Inc. Jun Sudo, The shift for semaiological value from functional (in Japanese)

[2] Kensuke Ohno, Yasuhisa Kato, Yoko Asano, "Service design creating appealing service and spirit of design thinking (in Japanese)", NTT technology journal 27(9), 54-57,2015-09, Telecommunications Association

[3] Motoichi Adachi, Takayuki Fujimoto, "Technique sparking the idea at brain (in Japanese)", Apl.2013, Kodansha

[4] Translation(2006): Kenichi Ohmae, "High-concept−−−The time of people creating 'new things' (in Japanese)", Mikasa Shobo

[5] Hasso; Meinel, Christoph; Leifer, Larry J, (2011). 『Plattner』

[6] Translation (2012): Yoko Goshi, "How to get ideas that can solve problems", PNN Inc.

[7] Robson, Mike (2002) [1988]. "Brainstorming". Problem-solving in groups (3rd ed.). Aldershot, Hampshire, UK; Burlington, VT: Gower. p. 41.

[8] Jonathan Littman, Translation(2006): Chikara Suzuki, "Master of innovation! ──10 human material that make company working out", Hayakawa Shobo

[9] Fortune Magazine （June 05 2012） 『15 top MBA employers IDEO』

ISBN: 1-60132-435-9, CSREA Press ©

A review of Security Assessment in E- Banking

Laleh Nosrati [1], Leili Nosrati [2]
[1]Industrial Engineering, Tarbiat Modares University, Tehran, Iran
[2]Computer Science, Islamic Azad University, Tehran, Iran

Abstract - *Today, as a result of lots of weaknesses in the online banking system, account hacking has caused much money losses. In this paper we will talk about different methods of online banking and mobile banking classification.*

In this process, different protocols of mobile banking system which are used till now, and also their various reported security problems are gathered. Moreover, encryption algorithms were compared and our method which has two layers of security has been presented.

Keywords: *online banking, mobile banking, privacy in mobile banking, mobile banking vulnerabilities, security of mobile banking, Privacy in mobile banking*

1 Introduction

Electronic banking has lots of advantages such as: convenience, quickness, and cost saving for both banks and customers who use it. Gradually, it has changed traditional banking industry and it is a combination of the most sensitive tasks which are happening through the internet.

1.1 Internet-Banking

Internet banking refers to use the website of your bank to process information. This means using the bank site in order to manage loans, transfer funds and checks in digital form, as well as, pay bills and control the accounts. Banking websites also allow you to download statements in PDF format, hence, if you want, you can keep your statement records. As another advantage, if you are interested in going green, it is also possible to go fully paperless and you can just view those statements and then delete them from your computer to save the world from the printing of unnecessary papers [1].

1.2 SMS-Banking

SMS banking is similar to Internet banking which helps you to do some of your financial works without going personally to the bank. However, it is different in that you primarily are using text messages to perform certain actions. Depending on your bank, you may be able to use text messages, to receive an account balance at a same time and get alerts when your account dips below a certain balance or when a large amount of money is withdrawn from your account [2].

1.3 Mobile-Banking

Mobile banking is a term used for performing balance checks, account transactions, payments, credit applications and other banking transactions through a mobile device [3].

Mobile devices, Smart phones, and tablets can be taken here and there easily. Furthermore, they provide users easy access to personal and financial data via applications that allow the movement of stored data on the devices and also allow data to be sent to and stored with a third force. Although, they can also be robbed, poison with malwares and fraud. Howbeit, smart phones are here to stay, and the way consumers use them may change every day, but it is clear that mobile banking via smart phones and tablets is on trend to grow rapidly in the coming years.

The released data of Malauzai Software, Inc. on 14th of August 2013, has confirmed the growing use of mobile banking applications [4]

Customers of Mobile banking have been allowed to perform these three transactions: [5]

- Saving money for accounts which are accessible with mobile
- Cash in/out transactions with accounts which have been stored
- Transferring money

Mobile banking can be done through the web browser on the mobile phone, to achieve the bank's web page via text messaging, or by using an application downloaded to the mobile phone [6].

Mobile banking gives full access to set apart transactions, personal details of bank, as well as making credit installment and transferring funds directly. Customers should use the service which has to register with their bank, with a valid identity as required by law after the telecom firm installs the software on the applicant's device. As the number of mobile banking providers increased, a fierce competition has grown in continuous improvement in service quality [7].

Bank management technologies are among the major changes in internal banking systems that also have exercised a positive influence on banking achievements and proprietary [8].

Socio-economic impact of mobile-banking systems in the developing world is scarce. Even less attention has been paid to the social, economic, and cultural contexts neighboring the use of these systems, mobile banking is uprising that is driven by the world's one of the fastest growing sectors which is mobile communication technology. The implications of the results provide practical recommendations to the all concerned mobile banking security and payment challenges. This paper reports on the aiming at identifying and classifying Mobile banking payment security challenges. We provide an overview of the-state-of-the-art of mobile banking security challenges, and a key for reading and interpreting them. In addition, this paper presents a number of interesting findings, including mobile banking challenges of payment security and application of mobile banking, and the importance of inter-relationships between financially as well as our smart phone security challenges.

One of the new challenges of mobile banking is online threats; mobile users are always active on online downloading various applications, songs and official mail or other personal files over the internet. Around five million users all over the world use their mobile phones to make different kinds of payment or view various payments, and pay utilities. Most customers receive this information in text messages format, but a number of banks now allow customers to download secure software that can access on banking system and third party. However the important question is how secured is this payment system, [10], mobile phones have emerged as one of the most ubiquitous technologies in human history.

1.3.1 Benefits of Mobile-Banking

With mobile banking, customers can do their banking activities anytime, anywhere, and cheaper [11]. In other words, the bellow points show the benefits of mobile banking.

1.3.2 Vulnerabilities of Mobile-Banking

1.3.2.1 Distributed Denial of Service (DDOS) Attack

DDOS attack is ranked as third highest threat as FBI said. DDOS is the most common attack of banking system. DDOS attack, orbit the attack to target system. Before an attack is happen, attacker will attack network by scanning open ports [12].

1.3.2.2 Malware

Malware is the term for maliciously crafted software code. There are special computer programs that enable intruders to fool you into believing that traditional security is protecting you during online banking actions. Moreover, it is possible to perform the following operations for this type of malicious software account information theft [13]:

- Fake web site substitution
- Account hijacking

1.3.2.3 TCP/IP Spoofing

Here, an attacker gains unauthorized access to a mobile device or a network by making it show up that a malicious message has come from a trusted machine by “spoofing” the IP address of that machine [14].

1.3.2.4 Backdoors

Access to a mobile program that avoided security mechanisms is backdoor. A programmer may sometimes install a back door so that the program can be accessed for troubleshooting or other purposes. Anyhow, back doors have been used by attackers to install themselves, as chunk of an exploit [15].

1.3.2.5 Tampering

Tampering is an intentional modification of products in a way that would make them harmful to the consumer [16].

1.3.2.6 Exploits

Exploit is a piece of software, or a data which acts as a bug or vulnerability in order to matter surprising behavior to exist on computer software, or hardware [17].

1.3.2.7 Social Engineering and Trojans

Trojans act as no authorized programs. Can delete, block, modify, and copy data. However, Trojan is not like a viruses and worms, it is not able to self-replicate [18].

2 Risks

2.1 Risks in e-Banking Devices

- Not encrypted SMS
- Spoofing SMS
- Passwords would find by hackers through stolen devices [19]

2.2 Risks in Wireless Application Protocol

- Eavesdropper attacker
- Unencrypted data during switching between protocols
- Attacker can contact to unencrypted data [20]

2.3 Risks in Servers

- System may crash
- Server may be failed
- Virus may be attack

3 Proposed Method

In our method Channel Manager which is middle-ware software to integrate Internet Banking, Mobile Banking, and EFT Switch, and etc. has been selected as core banking solution. Channel Manager provides an API in form of web services to handle transactions or queries expected from Internet Banking, Mobile Banking, and EFT Switch.

In this method, requests only from known entities such as mobile Banking and some other applications have been accepted from channel manager. Each entity which wishes to perform an operation must get registered through Channel Manager Application. Each entity provides a separate shared secret key. The entity will be responsible to preserve this key securely, and use it to generate a checksum for every request sent out and verify checksum by Channel Manager Application for every received request. Channel Manger validates the check sum and timestamp for the authorization. Check sum is encrypted with 'SHA-256' (Message Digest) with shared secret key. If check sum validates successfully, channel manager will validate the timestamp of the service which should be within the specified time.

4 Conclusions

Using E-Banking to access banking financial systems made customers do their banking jobs independent of time and location. In addition, it allows customers to take full advantage of the latest technology.

In this paper, we tried to gather all different kinds of online banking systems. However, we focus on e-banking, specially, mobile banking, the benefits of e-banking, all vulnerabilities which found till now, and different protocols of mobile banking. Moreover, different algorithms of encryption which used in mobile banking have been compared. At last we proposed our own method which is more secure for mobile banking system. Our method has two security layers which are Authentication and Authorization. In addition, for preparing the security of the Network Layer, our method will authorize the Message format, which is encrypted with 'SHA-256' format. Our findings ring a bell to the research community.

5 References

Post: Feasible model and assessing demand potential", International conference on emerging economies-Prospects and challenges (2012)

[2] Jeong, B. K., & Yoon,"An Empirical Investigation on Consumer Acceptance of Mobile Banking Services", Business and Management Research, 2(1), 31-40, T. E. (2013)

[3] J. D. Pitts, “Surfing the Payment Channels, Mastering the Fraud Tsunami”, JDP Enterprises, Carrollton, TX, (2010).

[4] Balebako, R., & Cranor, L., "Improving App Privacy: Nudging App Developers to Protect User Privacy", Security & Privacy, IEEE, 12(4), 55-58, (2014)

[5] Md. Shoriful Islam," Systematic Literature Review: Security Challenges of Mobile Banking and Payments System", International Journal of u- and e- Service, Science and Technology Vol. 7, pp. 107-116(2014)

[6] Elkhodr, M., Shahrestani, S., & Kourouche, K.," A pro-posal to improve the security of mobile banking applications", In ICT and Knowledge Engineering (ICT & Knowledge Engineer-ing), 2012 10th International Conference on (pp. 260-265). IEEE, (2012)

[7] He, W., "A Review of Social Media Security Risks and Mitigation Techniques", Journal of Systems and Information Technology, 14(2), 171-180, (2012)

[8] He, W. , "A Survey of Security Risks of Mobile Social Media through Blog Mining and an Extensive Literature Search", Information Management and Computer Security, 21(5), pp.381–400.

[9] Hameed Ullah Khan, "E-banking: Online Transactions and Security Measure", Research Journal of Applied sciences, Engineering and technology 7(19): 4056-4063, (2014)

[10] Rajpreet Kaur Jassal1, Ravinder Kumar Sehgal," Study of Online Banking Security Mechanism in India: Take ICICI Bank as an Example ", IOSR journal of computer engineering (IOSR-JCE), p-ISSN: 2278-8727 Volum13, Issue1 ,PP114-121,(2013),

[11] Jeong, B. K., & Yoon, T. E. , "An Empirical Investigation on Consumer Acceptance of Mobile Banking Services", Business and Management Research, 2(1), 31-40 , (2013)

[12] "Alternative Graphical Authentication for Online Banking Environments", Proceedings of the Eighth International Symposium on Human Aspects of Information Security & Assurance (2014)

[13] Hameed Ullah Khan, "E-banking: Online Transactions and Security Measure", Research Journal of Applied sciences, Engineering and technology 7(19): 4056-4063, (2014)

[14] Vorugunti Chandra Sekhar, "A Secure Account-Based Mobile Payment Protocol with Public Key Cryptography", Mrudula Sarvabhatla, ACEEE Int. J. on Network Security, Vol. 03, No. 01, (2012)

[15] Jaak Tepandi, "Security Analysis of Mobile ID", Ordered by Department of State Information Systems, (2008)

[16] Yacine Challal, Hamida Seba, "Group Key Management Protocols: A Novel Taxonomy", International journal of information technology volume 2 number 2 ISSN 1305-239X ,(2005)

[17] Ibrahim M. Al-Jabri, M. Sadiq Sohail," Mobile banking adoption: Application of diffusion of innovation theory ", Journal of Electronic Commerce Research, VOL 13, NO 4,(2012)

[18] G. Bella, F. Massacci, and L. C. Paulson,"The verification of an industrial payment protocol: The SET purchase phase" In V. Atluri, editor, 9th ACM Conference on Computer and Communications Security, pages 12–20. ACM Press, (2002)

[19] M.Ebrahimi, S. Khan, Shujjat kahn, UMer Bin Khalid,"Symmetric Algorithm Survey: A Comparative Analysis", International Journal of Computer Applications (0975 – 8887) Volume 61– No.20,(2013)

[20] Haiyong Xie, Li Zhou, and Laxmi Bhuyan, “Architectural Analysis of Cryptographic Applications for Network Processors”, Department of Computer Science & Engineering, University of California

[21] ElBahlul ElFgee, Ahmed ARARA, "Technical Requirements of New Framework for GPRS Security Protocol Mobile Banking Application", International workshop on intelligent techniques in distributed systems (ITDS), Procedia Computer Science 37, (2014)

[22] Limor Elbaz & Hagai Bar-El, “Strength Assessment of Encryption Algorithms, (2000)

[23] Bruce Schneier, "The Blowfish encryption algorithm9", Dr. Dobb's Journal of Software Tools, 19(4), p. 38, 40, 98, 99, (1994)

[24] Jeong, B. K., & Yoon, T. E., "An Empirical Investigation on Consumer Acceptance of Mobile Banking Services",Business and Management Research, 2(1), 31-40, (2013)

[25] X. Lai and J. Massey. A proposal for a new block encryption standard. In Proceedings of the EUROCRYPT 90 Conference, pp. 3 89-404, (1990)

[26] Wheeler, D.J., & Needham, R.J., "TEA, a tiny encryption algorithm", In Fast Software Encryption– Proceedings of the 2nd International Workshop, 1008. (1994).

[27] Kelsey, John; Schneier, Bruce; Wagner, David "Related-key cryptanalysis of 3-WAY, Biham-DES, CAST, DES-X NewDES, RC2, and TEA", Lecture Notes in Computer Science 1334: 233–246. Doi: 10.1007/BFb0028479, (1997)

[28] Anjali Patil, Rajeshwari Goudar, "A Comparative Survey of Symmetric Encryption Techniques for Wireless Devices", International journal of scientific & technology research volume 2, ISSUE 8, (2013)

[29] James Nechvatal, Elaine Barker, Lawrence Bassham, William Burr, Morris Dworkin, James Foti, and Edward Roback, "Report on the Development of the Advanced Encryption Standard (AES)", Volume 106 Number 3 , (2001)

Crowdsourcing for Emergency Response

Liliya I. Besaleva and Alfred C. Weaver
Department of Computer Science, University of Virginia, Charlottesville, VA 22904 USA

Abstract - *In the past decade the importance of social networks and mobile technologies has been growing rapidly. The convenience, ease of use and affordability of handheld devices have caused an unprecedented increase in Internet connectivity across all age and social groups in various sectors of our lives, including emergency services. An ongoing movement among crisis management organizations is the incorporation of Web 2.0 and Web 3.0 tools into their practices for the improvement of their critical situations response. Simultaneously, the level of trust in the capabilities, safety and reliability of social networks demonstrated by the general public has been growing equally quickly. As a result, social media is becoming a ubiquitous part of the process of disaster management. In this paper, we provide a detailed discussion on how social media are rapidly overtaking traditional methods for emergency response and review current ideas, examples and trends for the usage of online services for disaster control and the seamless transition between platforms and device types.*

Keywords: crowdsourcing; social networking; mobile applications; emergency response

1 Introduction

The influence of social media is growing rapidly. According to the Search Engine Journal (SEJ) [20], as of 2012 nearly 91% of American adults who are online use social media for a variety of purposes ranging from private to business matters. Social networking sites such as Facebook, Twitter, LinkedIn, Google+, and others are starting to be considered mainstream sources [17] of information and entertainment for hundreds of millions of people around the globe. And while the overall growth rate of content creation and active usage is starting to slow down as a part of the natural process of maturation of the web, two exceptions to the rule are *social networking* sites and *mobile applications*, both of which continue to grow rapidly [8]. In fact, by September 2014 the number of unique mobile users in the world surpassed 50% of the world's population [30]. Consequently, it has become a staple for content providers to have information available in a variety of formats and versions for both mobile and desktop devices. Thus, the possibilities for their use have grown far beyond their initial entertainment, socialization and recreational purposes. From raising awareness about important social issues, to creating precise demographic profiles of millions of web users for the purpose of targeted advertising, to winning political campaigns, the power of online networking is significant, and the knowledge it offers is immense. The problem many institutions have is that with all this information available, choosing the most useful pieces of it becomes a challenging task.

Simultaneously, another trend has arisen—the use of social networking for disaster management and improving public health and safety. Earthquakes, tsunamis, tornadoes, violent storms, influenza epidemics and unpredicted acts of mass aggression (e.g., Boston Marathon bombing) have proven difficult to manage despite all of our technological advancements. When a disaster occurs, sometimes the best (or indeed only) way to save numerous lives is to generate a timely reaction to its outcomes. That puts extra emphasis on the importance of high-quality information management for emergency communications. This is particularly visible in critical situations because internal and cluster communication within law enforcement, relief groups, non-governmental organizations, civil societies and other agencies is crucial to getting their work accomplished in a timely manner. Without sufficient coordination due to inadequate information flows, we end up witnessing great devastation which could have otherwise been reduced. One example is Hurricane Katrina (2005) in the southern part of the United States where at least 1,836 deaths [28] and property damages estimated at $81 billion USD [16] were recorded. According to Mahamed Gad-el-Hak in his book *Large-Scale Disasters: Prediction, Control and Mitigation* [9, page 129]: "inability of disaster response managers to validate and process relevant information and make decisions in a timely fashion" resulted in greater than anticipated losses.

Historically, society has modeled its responses through circumstantial observation, using trained professionals (e.g., police, fire, EMS) and occasionally volunteers. But more recently, many of the leading rescue and disaster management organizations have been turning towards social networking websites to take advantage of their huge user base, accessibility, and fast response. [8][13] Hence, one of the positive characteristics of social media, which is its capacity to harness collective knowledge for learning and problem-solving, is no longer being overlooked. Additionally, in 2015 a new trend among online platforms for advancements in interconnectivity and shared content has been on the rise. Examples such as Vine allows its users to be posting simultaneously on Twitter, Facebook and Tumblr through its mobile app [25], and services such as Share This permit a completely customized social networking experience in a one-stop-shop location for posting, monitoring and content analysis are becoming the new power players in the interim between our transition from Web 2.0 to Web 3.0. As a result, the "crowd" feels more knowledgeable in their decision-making and more empowered in their efforts of bringing attention to socially impactful situations, such as mass-scale disasters.

The advantages of mining data coming from various sources (and the consequent creation of fine-grained maps of the surroundings following a disaster) have earned praise from both citizens and government. As a result, we have witnessed a number of cases where online networks have been a primary source of information for both victims and rescue authorities. One of the most recent examples is the 2012 Superstorm Sandy which was ranked by the National Oceanic and Atmospheric Administration (NOAA) and the American Red Cross (ARC) [2] as the number one US weather event for 2012. Throughout Hurricane Sandy, the public turned to social media for updates and assistance, and more than ever before, response agencies, organizations and community groups used social media to organize and direct resources where needed. For example, the New York Office of Emergency Management provided hourly updates and evacuation orders via Twitter, and New Jersey Gov. Chris Christie relayed updates about the storm, available aid and evacuation orders via his personal Twitter account.

The reason behind the rising usefulness of social networks is their highly crowdsourced information, constant availability, increased accessibility and ease of use. While the local widespread communication and information services (e.g., telephone, television, radio) often crash during large natural disasters, the fabric of the social networking websites can remain intact due to its distributed and agile nature. Social networking can provide a reliable means of communications for both those involved and those witnessing an incident.

2 Applications of Social Media

Social networks can provide detailed knowledge about their users (consider Facebook that knows your age, job title, music preferences, etc.) With that kind of data, advertisers can easily and cost-effectively customize their products and advertisements for the correct target groups, predict future trends, and keep themselves updated with the latest consumer feedback about their brand. But the uses of social media do not end there. For instance, LinkedIn (a vastly popular professional business network*)*, provides ways for smaller inner networks to develop such as the *Advanced Social Work Practice Network* (ASWPN) [1] with nearly 1,000 active members from around the world. The purpose of that inner-circle network is the enhancement of good practices and continuous professional development of advanced practitioners from the UK. Another such group is the *Network for professionals working with vulnerable children and adults* [14] which includes social work and health practitioners, managers and academics.

Vivid examples of the use of social media can be seen in the fields of health and social welfare. Websites providing first-person experiences and advice make it easier for patients and survivors to open up about their conditions and ask for help. A popular one is the *patientslikeme* website [15] where over 150,000 people look for answers to unusual, embarrassing, or simply hard-to-talk-about health-related questions. It is a safe community where people feel protected behind their online personas, and no longer experience discomfort when revealing their real-life stories to others because it is a place of shared sympathy and understanding. Additionally, many social networks for survivors or friends and families of victims of disastrous events exist; they are used for connecting people needing that kind of support to cope with their grief or share their inspirational survival stories. The social importance of such communal social networks has been recognized by various authorities, including the Red Cross, and the Association for Computing Machinery (ACM) [7] [22] which is the world's largest educational and scientific computing society.

Social media has a vast presence in the academic world as well. From huge professional and educational online communities like the ACM itself, to smaller Twitter groups (e.g. @Info4Practice, @LSEImpactBlog), and Facebook pages of universities/colleges or scientific groups, social networks are everywhere helping to establish better connections between the students and the educational organizations. That helps in building not only better communications between both sides, and increased flexibility and accessibility as well, but also a sense of pride in the institution.

In the last couple of years local authorities have also embraced social media to enable their staff to connect more effectively with the local communities. That is a direct way for them to engage with residents, community groups and partners using blogs, Twitter, Facebook, YouTube, Flickr and LinkedIn. These social networks allow local authorities to get involved in local conversations or collect "the wisdom of the crowd" about events occurring in their locality. These organizations are slowly increasing their level of recognition

2.1 State-of-the-Art Solutions

State-of-the-art solutions for disaster response are based in part upon data retrieval from social networks. Many crisis response systems turn to social networks (Twitter, Facebook, etc.) to both locate and provide the latest news relating to a critical situation. Growing demand makes the development of successful and effective tools even more vital and urgent. And with the increased acknowledgement of the power of crowdsourced data, even organizations that have been slow to adapt, such as the Federal Emergency Management Agency (FEMA), have joined the effort to harness the power of the crowd [5]. Currently, numerous examples of somewhat successful systems used even in large-scale situations exist such as:

Ushahidi [6] is an open source crisis map platform created in 2007, and deployed in multiple locations, inc. Kenya, Mexico, Afghanistan, Haiti, New York, California, and Washington DC. It leverages web technologies for multiple source data integration (phones, e-mail, social media sites such as Twitter and Facebook) and provides an up-to-date publicly available crisis map that is in turn available to relief organizations. The platform uses crowdsourcing for information collection, and support cooperation enabling among various organizations. Ushahidi was a prominent player in the management of the massive *Snowmageddon* snowstorm in the District of Columbia in the winter of 2010.

GeoCommons [7] is a tool for the community of GeoIQ users for building an open repository of data and maps for the world. Some of the features it provides are map visualization, temporal analysis visualization, data upload and search features, filter and access data features, and dataset show-and-edit features.

Both of those technologies were successfully used in Haiti in 2010, and more recently in the big Texas floods of 2015, and helped in saving hundreds of lives there. A direct gathering of information coming from the online ‘crowd’, fortified with quick sorting and analysis of that data, were of tremendous help for shedding light in a moment of disorder and panic.

Another highly cited and exemplary work is the technology developed in *Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors* [23] from The University of Tokyo where Twitter is used for earthquake detection and reporting. In that work, the micro-blogging service is used as a source of immediate textual responses to an earthquake, which are then analyzed, and a conclusion about the occurrence of an earthquake is made. In places with very technologically savvy users such as Japan, that system was even capable of detecting real earthquake events with a probability as high as 96%.

2.2 Issues of Existing Systems

Sadly, most of the information coming from social media sources cannot be considered entirely trustworthy. These publicly available websites normally do not perform any data integrity checks. Therefore, the entries could be malicious, or sent for the purpose of creating confusion. That is a problem some have attempted to investigate and deal with but without affirmative success. An exemplary work is *Seeking the Trustworthy Tweet: Can Microblogged Data Fit the Information Needs of Disaster Response and Humanitarian Relief Organizations* [3] which discusses the untrustworthy nature of data coming from social media, as well as the different types of issues arising from it. As stated in that paper, currently no solutions exist—but what looks promising as the most scalable and effective approach is the use of machine learning, entity extraction, and text classification techniques for data analysis.

Another side of the social networking approach is its decentralized nature. Post, tweets, and blog entries are spread throughout the entire Web. While the distributed way of collecting and presenting information could be considered good for data richness, it also means people often need to visit several different data centers to locate all the information they seek. Also, that information is usually not in any purified state (cleansed of “noise”), but instead comes in a variety of forms, each of which requires additional effort on the part of the user.

3 Social Importance of Applications

In the past few years, the level of respectability and social acceptance of social networks has grown exponentially. According to recent nationwide surveys [19] the percentage of job offerings online on social networking sites surpasses 75% of all positions available, and the majority of hiring managers want to use online resources for recruiting purposes. Similarly, online websites are also present in most other aspects of our lives and are rapidly penetrating even the most resistant areas of business and governmental. As a result, many new opportunities for using Facebook, Tumblr, Twitter, etc. have arisen, and the social importance of those websites is increasingly appreciated.

A vivid example of just how significant online networking is for society was its use during the earthquake and resulting tsunami in Fukushima, Japan in 2011 which has 8,649 confirmed deaths and over 13 thousand missing persons. Lacking knowledge about the health and safety of loved ones, and with mobile phone networks jammed due to overloading, many people chose social media sites such as Facebook, Twitter, Mixi (an exclusively Japanese website), etc. to establish communications and find out the latest news.

According to the annual Twitter report there was an almost immediate growth of over 500% in tweets in Japan right after the Tōhoku earthquake as people reached out to their loved ones. Later on in June, Twitter posted a visualization of the tweets sent to and out of Japan within one hour of the disaster striking (see Fig. 1,2, for video see [24, 25]). That gave a powerful message of just how heavily people depend on simple online platforms for immediate connection and help in case of danger.

What allows accessing online social networks, when even standard calls are impossible to make because of excessive network traffic, are the different telecommunication standards and protocols such as SMS used for mobile Internet in comparison to those used for telephone calls [27] [29]. Hence, neither of those services affects the other one.

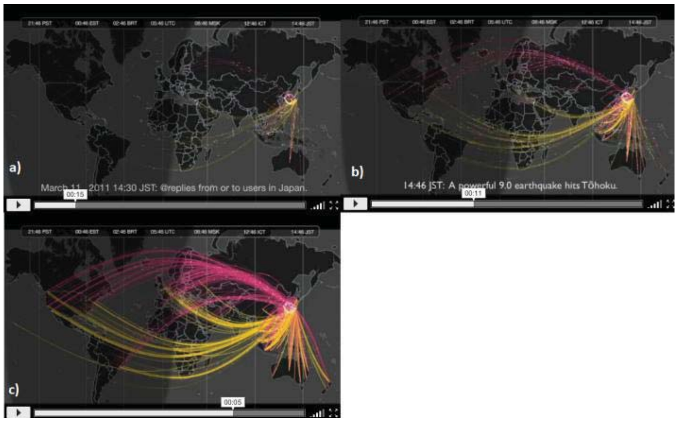


Figure 1. Tweets within one hour of Tōhoku earthquake (yellow represents tweets coming from Japan, pink represents tweets going to Japan). a) Normal Tweet load before earthquake; b) Immediate response after earthquake; c) One hour after earthquake.

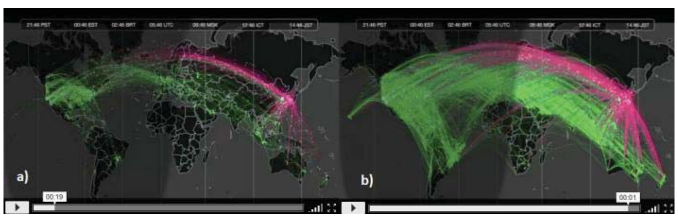


Figure 2. Retweets of Tweets originating in Japan within one hour of Tōhoku earthquake (pink - original Tweets, green - retweets). a) First several minutes after earthquake; b) One hour after earthquake.

Whereas previously these social media sites might have been joined out of curiosity, suddenly they ‘came of age’ in Japan and became a lifeline for thousands around the globe. Survivors used it for keeping in touch with the rest of the world, rescue organizations collected data for analysis, charity groups raised money for the needy, governments made appeals for public order, while everyone else kept a wary eye on current events. With record numbers of users (over 11 million active users of Facebook and over 17 million active users of Mixi in Japan alone), the social arena became an additional layer of our lives where we can go to search for those lost in the devastation, donate, support, pay respects, or simply keep up-to-date with the latest happenings [18][24].

Another example of online networking power was the 12 January 2010 cataclysmic earthquake in Haiti. According to reports from the Red Cross, within less than 48 hours from the disaster striking, more than US$8 million in donations was received via text messages due to campaigns spread across the social media websites. Numerous videos, images, and texts of the personal experiences of witnesses of the earthquake were available online within a few minutes of it happening. That was a major wake-up call for many about the undeniable power of social networking in various online communities. According to online reports [12], tools as simple as Facebook mobile application widgets were a central way of organizing rescue missions and saving hundreds of people stuck under rubble with no other means of communication but social media sites.

Many more examples exist as well, but what we gather from each and every one of them is just how big a role social networks really play in our lives nowadays. Online networking is without a doubt here to stay. The social importance of websites such as Facebook, Twitter, Google+, and many others is growing day by day, and will soon introduce us to a world where our online presence will go hand-in-hand with our real-life personas as two inseparable parts.

4 Future Trends

As mentioned by Craig Fugate in his statement on the FEMA website [4], much has been done to aid the operations of relief organizations v social media. At the same time, most local governments still employ outdated intranets within their organizations, and very few Web 2.0 and Web 3.0 technologies are being used [10][11]. In order to make the practices of the officials more efficient, effective, and inclusive a number of cities have plans for increasing their involvement in online communities. And the next step to be taken in that direction is going mobile. So, making mobile versions of websites, creating relevant applications, and urging for expanded collaborations with existing networks (e.g., Craigslist, Google, Apple, Microsoft) and traditional organizations through mass-technology tools, are the likely choices of the rescue authorities in the years to come.

Another very new trend is building NASA-style mission control centers (MCC) for social media. As information coming from networking websites could be so vast, both in terms of numbers of entries but also in their interconnections, it has become increasingly difficult to maintain a constantly up-to-date stream of easily comprehensible data about the latest events. Therefore, growing numbers of global organizations are building multimillion-dollar control centers: dedicated physical hubs for monitoring and responding to the torrent of social commentary and queries flooding in via Facebook, Twitter and other channels. An example of such a MCC related to disaster management is the latest Red Cross Digital Operations Center (see Fig. 3).

With help from this new MCC the Red Cross is hoping to be able to expand its outreach to volunteers willing to help out to those in need at the right time and place.

Figure 3. ARC Digital Operations Center for Disaster Monitoring through Social Media

5 Conclusion

There is a plethora of articles, scientific works, reports and applications of social media available at the moment. Naturally, by creating more accessible materials about a certain topic they only attract even further curiosity and discussion, and the popularity of the topic grows exponentially. A similar phenomenon can be seen in the field of emergency management through crowdsourcing and online media. Tools that did not exist even 5 years ago are now primary means of communication for hundreds of millions around the world. Regardless of its mass recognition, this is still a field which is yet to be fully explored. The importance of social networks is bound to increase dramatically over the next decade. We are currently in a greatly prolific stage for the Web 2.0 technologies which are steadily becoming a standard of work in numerous organizations. Slowly but surely, even more advanced Web 3.0 technologies are finding their way into commercial software which disaster management organizations will eventually adopt.

6 References

[1] Advanced Social Work Practice Network (ASWPN), http://www.linkedin.com/groups/Advanced-Social-Work-Practice-Network-2639368/about [last visit 4/9/2016].

[2] American Red Cross, "Superstorm Sandy: One month Update", http://www.redcross.org/images/MEDIA_CustomProductCatalog/m15840096_SandyUpdate-MonthOne.pdf [last visit 4/9/2016].

[3] A. H. Tapia, K. Bajpai, J. Jansen, J. Yen and L. Giles, "Seeking the Trustworthy Tweet: Can Microblogged Data Fit the Information Needs of Disaster Response and Humanitarian Relief Organizations", in Proceedings of the 8th International ISCRAM Conference – Lisbon, Portugal, May 2011.

[4] C. Fugate, "Understanding the Power of Social Media as a Communication Tool in the Aftermath of Disasters", http://www.dhs.gov/ynews/testimony/testimony_1304533264361.shtm [last visit 4/9/2016].

[5] FEMA, Mobile app, https://www.fema.gov/mobile-app [last visit 6/3/2015]

[6] H. Gao, G. Barbier , R. Goolsby and editor D. Zeng, , "Harnessing the Crowdsourcing Power of Social Media for Disaster Relief". IEEE Computer Society, 1541-1672/11/2011 IEEE.

[7] Klasnja, P., Hartzler, A., Powell, C., Phan, G., & Pratt, W. "HealthWeaver Mobile: Designing a Mobile Tool for Managing Personal Health Information during Cancer Care". ACM Conference on Human Factors in Computing Systems 2010.

[8] K. Ala-Mutka, D. Broster, R. Cachia, C. Centeno, C. Feijóo, A. Haché, S. Kluzer, S. Lindmark, W. Lusoli, G. Misuraca, C. Pascu, Y. Punie and J. A. Valverde, "The Impact of Social Computing on the EU Information Society and Economy", JRC Scientific and Technical Reports, EUR 24063 EN – 2009.

[9] M. Gad-el-Hak, Large-Scale Disasters - Prediction, Control and Mitigation, Cambridge, September 2008.

[10] M. J. Magro, "A Review of Social Media Use in E-Government". Administrative Sciences". 2012; 2(2), pp.148-161.

[11] [M. McDonald, J.S. McDonald, G. A. Merwin Jr., Keith A. Merwin and M. Richardson, "Assessment of Web 2.0 Applications Employed by Human Resource Departments in U.S. Cities", in E. Downey, & M. Jones (Eds.), Public Service, Governance and Web 2.0 Technologies, 2012: Future Trends in Social Media, pp. 122-138.

[12] M. S. Bruno, The tale of an Android phone in the earthquake in Haiti, http://nuvohaiti.blogspot.com/2010/02/tale-of-android-phone-in-earthquake-in.html [last visit 4/9/2016].

[13] N. A. Christakis and J. H. Fowler, "Connected: The Surprising Power of Our Social Networks and How they Shape Our Lives", published by Little, Brown and Company, Hachette Book Group, September 2009.

[14] Network for professionals working with vulnerable children and adults, http://www.linkedin.com/groups/Network-professionals-

working-vulnerable-children-2584873/about [last visit 4/9/2016].

[15] Patients like Me, http://www.patientslikeme.com/ [last visit 4/9/2016].

[16] R. D. Knabb, J. R. Rhome and D.l P. Brown, “Tropical Cyclone Report Hurricane Katrina”, 23-30 August 2005, National Hurricane Center, 20 December 2005.

[17] R. L. Briones, B. Kuch, B.F. Liu and Y. Jin, “Keeping up with the digital age: How the American Red Cross uses social media to build relationships”, Public Relations Review, Volume 37, Issue 1, March 2011, Pages 37-43.

[18] Regus, “A global report from Regus, A global survey of business social networking”, http://www.regus.com/images/Social%20Success%5B1%5D_tcm8-21559.pdf [last visit 4/10/2013].

[19] S. Adams, New Survey: LinkedIn Still The Top Social Networking Site For Job Seekers And Recruiters, http://www.forbes.com/sites/susanadams/2012/06/05/new-survey-linkedin-still-the-top-social-networking-site-for-job-seekers-and-recruiters/ [last visit 4/9/2016].

[20] Search Engine Journal – SEJ, “The Growth of Social Media: An Infographic”, http://www.searchenginejournal.com/the-growth-of-social-media-an-infographic/32788/ [last visit 4/9/2016].

[21] Share This, http://www.sharethis.com/#sthash.SIPYDWT0.dpbs [last visited 6/3/2015]

[22] S. R. Hiltz, P. Diaz and G. Mark, “Introduction: Social media and collaborative systems for crisis management”, ACM Trans. Comput.-Hum. Interact. 18, 4, Article 18 (December 2011), 6 pages.

[23] T. Sakaki, M. Okazaki and Y. Matsuo, “Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors”, in Proceedings of the 19th international conference on World Wide Web (WWW '10). ACM, New York, NY, USA, pp. 851-860.

[24] Twitter, Personal Messages From Japan., http://www.flickr.com/photos/twitteroffice/5885172082/ [last visit 4/9/2016].

[25] Twitter, Spread of Information., http://www.flickr.com/photos/twitteroffice/5884626815/ [last visit 4/9/2016].

[26] Vine, Simultaneous sharing: https://vine.co/ [last visit 6/3/2015]

[27] Wikipedia, Evolution-Data Optimized., http://en.wikipedia.org/wiki/Evolution-Data_Optimized [last visit 4/9/2016]

[28] Wikipedia, Hurricane Katrina., http://en.wikipedia.org/wiki/Hurricane_Katrina [last visit 4/9/2016].

[29] Wikipedia, High-Speed Downlink Packet Access, http://en.wikipedia

[30] http://wearesocial.net/blog/2014/09/world-mobile-phone/ [last visit 6/4/2015]

"SDSS: Smart Diet Support System" Creativity Support System Development and Evaluation

Motoichi Adachi[1, 2], **Takayuki Fujimoto**[2]
[1]Motoichi Entertainment, Inc. Tokyo, Japan
[2]Graduate School of Engineering, Toyo University, Saitama, Japan

Abstract - *In this thesis, we carried out evaluation tests to verify the effectiveness of SDSS: Smart Diet Support System, which has been developed on the basis of our research findings on meal intake and mental alertness. The evaluation tests were planned to verify system effectiveness by score variation of the exams and workbooks distributed in the open market. The results showed that the hypotheses of this study seems to be fundamentally valid.*

Keywords: Smartphone, SDSS, Application, Apps, Mental Alertness

1 Introduction

Phenomenon such as *sluggish in a fed state* or *alert in a small fasting state*, are well known perceptive mechanism of human beings. For example, occupations like planners or writers, who are always required to create new ideas, have taken strong interest in the relevance between meal intake control and creativity for ideas despite the lack of scientific evidence. Common patterns are widely established: they have small meals on purpose before important meetings where they need to provide new ideas; they prefer not to eat too much before writing. These patterns were not necessarily based on scientific evidence or verification, though they are assumed to exist empirically and they are often the case of belief based on popular theory.

However, according to the hypothesis: "a properly fasting state makes it easier to create ideas (namely, makes you alert), and the relevance between meal intake and *alertness (creativity)* is becoming recently clear in brain science.

For example, a study carried out by the Japan National Cerebral and Cardiovascular Center Research Institute shows [1]: "Approximately a 30% restriction on eating (diet control) increases Brain-Derived Neurotrophic Factor (BDNF) to become brain nourishment."

Also, according to a study by Professor Shuzo Kumagai from Kyushu University, it is becoming clear that dietary restriction has a great influence on the increase of BDNF expression [2].

Besides an empirical basis, scientific studies have begun to reveal that *an increase of BDNF* (*fasting state*) leads you to become *alert*.

Also, some of the studies abroad are revealing the relationship between fasting state and mental alertness, such as the result which scientifically supports that an increase of BDNF (fasting state) leads you to become *alert*[3] [4].

We have approached various studies on the relevance between meal intake and creativity [5]. Our verification and questionnaire investigations show quantitative results on influence with which meal intake works for creativity [6].

2 SDSS: Smart Diet Support System Development

From the survey results in our study, we formed the hypotheses indicated below.

【Hypothesis1】Meal intake time can be decided by subtracting a certain lapse time from the time in which you would like to be alert.

【Hypothesis2】Meal size can be decided by calculating the time in which you would like to be alert.

【Hypothesis3】You can control the time in which you would like to be alert by controlling meal time and meal size.

Based on the hypotheses above, we have devised and developed SDSS, a feeding support system to control meal intake time and meal size based on the time in which you would like to be alert, as a smartphone application software [7]. (Fig1. Fig2.)

Fig 1. SDSS Initial Screen

Fig 2. SDSS Operation Screen

This system enables us to be alert at a time which we would like to be alert by controlling meal size in our daily lives. It determines the meal time which is required by the user in order for them to make the most of their alert state at a specified time by using back calculation and proposes the most appropriate meal size and meal time based on two conditions: time slot in which you would like to be alert; meal size which you want to have. Basically, it is required to target a properly fasting state and a potent effect on dieting can be expected.

3 SDSS Evaluation Test

3.1 Evaluation Methods

We carried out evaluation tests to verify the effectiveness of SDSS proposed in this study. Score variation of the workbooks to test creativity was used for evaluation.

The examinees were a group of 93 randomly chosen people aged from sixteen to sixty-four.

3.2 Evaluation Methods

We have selected the books which have some scientific evidence out of the various tests and workbooks in the market to examine creativity, thinking ability and flexibility and used them as subjects for evaluation tests. In this test, how SDSS use/ disuse influences creativity and flexible thinking was clarified by measurement. Thus, it can be said that verifying the effectiveness of SDSS by score variations of the tests from those books is reasonable.

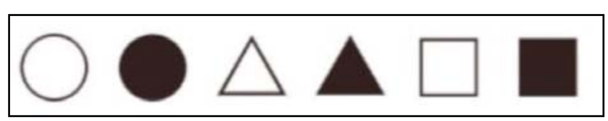
3.3 Sample Questions form Test

The following are sample questions and sample answers from the evaluation test.

［Sample Question1］Line up comic panels in an appropriate order to represent a story.

［Sample Answer1］

［Sample Question2］Rearrange the figures below according to the following five rules.



① The white circle is to the white square’s right.

② The black circle is second to the right of the white circle.

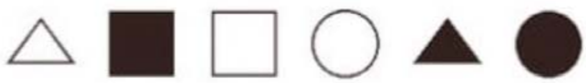
③ A square is next to a square.

④ The black triangle is between a circle and another circle.

⑤ Far left is a triangle.

ISBN: 1-60132-435-9, CSREA Press ©

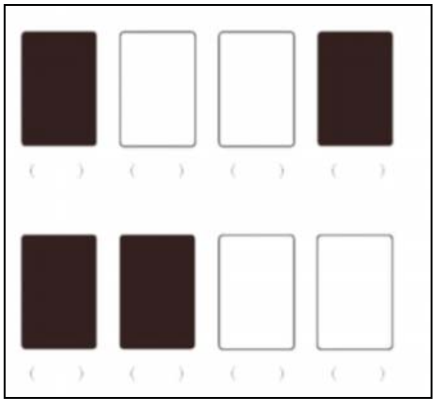
［Sample Answer2］



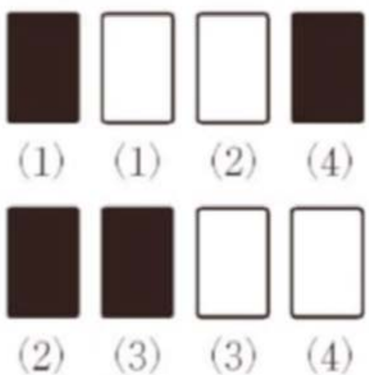
［Sample Question3］There are eight white and black cards numbered from one to four. Guess the number of each card according to the rule A and B.

(A)Left is a smaller number.

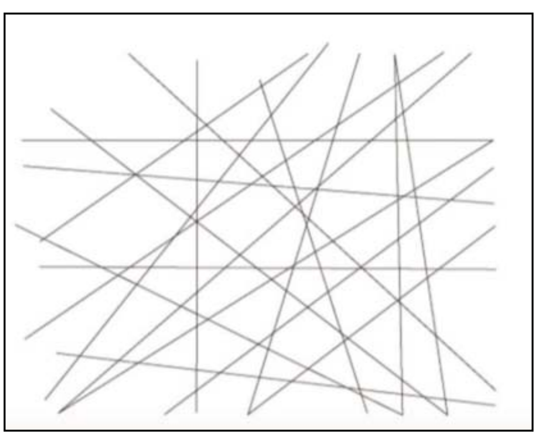
(B)If a number is the same, the black card comes left.



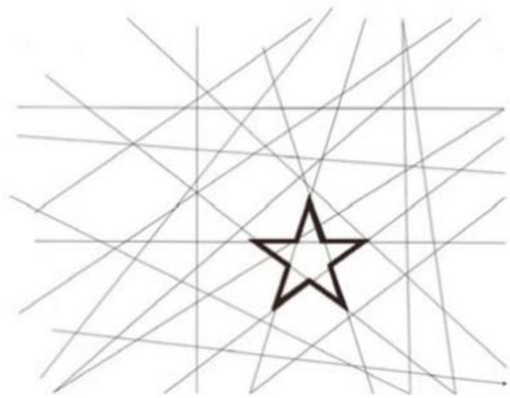
［Sample Answer3］



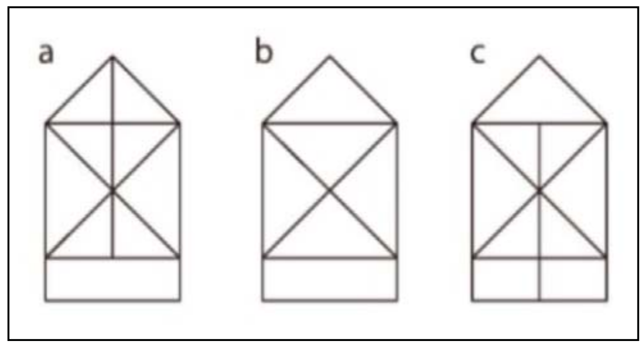
［Sample Question4］Extract one accurate star-figure from the picture.



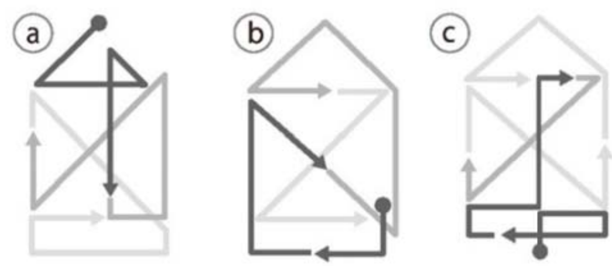
［Sample Answer4］



［Sample Question5］From the following picture a, b and c, choose which one is possible to draw without lifting your pencil from the paper.



［Sample Answer5］All possible.



4 Method and Results of Evaluation Test

4.1 Test Method

In this test, we set a time limit of 30 minutes and counted score. The test time of 30 minutes was set on the assumption that it would take almost 10 minutes to go through each section considering the volume and difficulty of the questions.

One set has 12 questions and the point allocation is 10 points per 1question. We have prepared questions on a 120-point scale and the test was held targeting 31 examinees per each of the 3 groups, in total 93 people. Extraction and grouping of examinees were run randomly.

The following are 3 groups with different test conditions.

【Group1】No preset restrictions, the examinees take the tests whenever they choose.

【Group2】With use of SDSS, the examinees take tests at an appropriate time slot during which they are supposed to be alert.

【Group3】With use of SDSS, the examinees take tests at a time slot during which they are not supposed to be alert.

4.2 Test Results

Results of the test of 31 people per each of the 3 groups (93 people in total) with different test conditions is shown in Fig.1.

The test score follows normal distribution. Since each group has different examinees, two groups have no paired samples. To analyze the difference of the two groups, an f-test was conducted at first to examine equality of variances. Then, a t-test appropriate for F-test results was held to confirm whether the combination of the groups showed significant difference. An applied level of significance was set at 5%. A spreadsheet software, Microsoft Excel 2011 was used for analysis.

Table 1. Evaluation Test Results

Group1 With App. Appropriate Time			Group2 With App. Inappropriate Time			Group3 Without App. Whenever		
Age	Sex	Score	Age	Sex	Score	Age	Sex	Score
49	F	60	33	F	90	50	M	70
43	F	60	62	F	80	29	F	70
33	M	100	33	M	40	45	F	100
38	M	80	62	M	90	31	M	90
41	F	70	29	F	90	31	M	50
34	F	80	45	F	80	32	M	70
42	M	110	54	F	50	32	M	80
51	F	100	40	F	80	37	M	110
40	M	70	51	M	80	50	M	60
32	M	90	54	M	30	18	M	70
48	M	100	49	M	50	19	F	100
50	M	90	55	F	80	62	M	70
43	M	110	56	F	70	64	M	60
25	M	70	60	M	70	58	F	80
25	M	70	27	M	50	48	F	100
34	F	80	29	F	90	23	M	60
18	F	90	24	M	60	28	F	80
22	M	90	49	M	80	45	F	50
19	F	110	58	F	50	42	M	90
34	F	90	40	F	70	42	M	70
24	F	70	35	F	50	40	F	70
32	M	90	32	M	70	27	M	90
24	M	70	42	F	100	25	F	70
39	F	100	49	M	50	29	F	60
55	M	60	19	M	70	35	M	90
62	F	80	16	F	50	36	F	70
58	F	90	64	M	60	16	F	80
63	M	60	63	F	60	21	M	60
60	M	90	60	M	100	29	M	70
21	M	70	20	M	90	54	F	60
48	F	90	42	F	70	33	M	60

Data acquired by the examinee test dealt with the assumption that there was no outlier, and the null hypothesis: There is no difference in the score of the two groups, was verified.

4.3 Group1 vs Group2

T-test results for group1 and group2 is indicated in Table2.

An f-test performed preceding the t-test showed that the P-value is 0.179 by one-tailed testing, which is more than 0.05, and it was confirmed that there is no significant difference. Therefore the t-test was conducted assuming equality of variances.

Table 2. T-test results for group1 and group2

Data		
	Group1	Group2
Mean	83.5483871	69.35483871
Variance	236.9892473	332.9032258
Observations	31	31
Degrees of freedom	30	30
	60	
f-test		
Significance level	0.05	
F	0.711886305	
P(F<=f) one-tailed	0.178584631	
F-critical one-tailed	2.07394375	
Significant difference one-tailed	No	
t-test		
Significance level	0.05	
Pooled variance	284.9462366	
Hypothesized mean difference	0	
t	3.310361093	
P(T<=t) one-tailed	0.000790098	
t critical one-tailed	1.670648865	
P(T<=t) two-tailed	0.001580196	
t critical two-tailed	2.000297822	
Significant difference one-tailed	Yes	
Significant difference two-tailed	Yes	

Both group1 and group2 examinees used SDSS. However only those in group1 used it in an appropriate way. Therefore, it was expected that group1's score would be higher. In fact, according to a comparison of the two means, group1's is higher: group1's is 83.5 and group2's is 69.4. The t-test for this combination was conducted by a one-tailed test to confirm if the difference is significant. As a result, the t-value is 3.31, the P-value is 0.0008, and the degrees of freedom is 60. Since the P-value is less than significance level of 0.05, there is a significant difference between group1 and group2.

4.4 Group1 vs Group3

T-test results for group1 and group3 is indicated in Table3.

An f-test performed preceding t-test showed that the P-value is 0.491 by one-tailed testing, which is more than 0.05, and it was confirmed that there is no significant difference. Therefore a t-test was conducted assuming equality of variances.

Table 3. T-test results for group1 and group3

Data		
	Group1	Group3
Mean	83.5483871	74.51612903
Variance	236.9892473	238.9247312
Observations	31	31
Degrees of freedom	30	30
	60	
f-test		
Significance level	0.05	
Variance ratio	0.99189919	
P(F<=f) one-tailed	0.491187892	
F critical one-tailed	2.07394375	
Significant difference one-tailed	No	
t-test		
Significance level	0.05	
Pooled variance	237.9569892	
Hypothesized mean difference	0	
t	2.305223052	
P(T<=t) one-tailed	0.012314343	
t critical one-tailed	1.670648865	
P(T<=t) two-tailed	0.024628686	
t critical two-tailed	2.000297822	
Significant difference one-tailed	Yes	
Significant difference two-tailed	Yes	

Group1 examinees were supported by SDSS whereas group3 examinees were not supported by anything. Therefore, it was expected that group1's score would be higher. In fact, according to a comparison of the two means, group1's is higher: group1's is 83.5 and group3's is 74.5. The t-test for this combination was conducted by a one-tailed test to confirm if the difference is significant. As a result, the t-value is 2.31, the P-value is 0.0123, and the degrees of freedom is 60. Since the P-value is less than significance level of 0.05, there is a significant difference between group1 and group3.

4.5 Group2 vs Group3

T-test results for group2 and group3 is indicated in Table4.

An f-test performed preceding t-test showed that the P-value is 0.184 by one-tailed testing, which is more than significance level of 0.05, and it was confirmed that there is no significant difference. Therefore a t-test was conducted assuming equality of variances.

Table 4. T-test results for group2 and group3

Data		
	Group2	Group3
Mean	69.35483871	74.51612903
Variance	332.9032258	238.9247312
Observations	31	31
Degrees of freedom	30	30
	60	
f-test		
Significance level	0.05	
Variance ratio	1.393339334	
P(F<=f) one-tailed	0.184370373	
F critical one-tailed	2.07394375	
Significant difference one-tailed	No	
t-test		
Significance level	0.05	
Pooled variance	285.9139785	
Hypothesized mean difference	0	
t	1.201728728	
P(T<=t) one-tailed	0.117095262	
t critical one-tailed	1.670648865	
P(T<=t) two-tailed	0.234190523	
t critical two-tailed	2.000297822	
Significant difference one-tailed	No	
Significant difference two-tailed	No	

Despite the use of SDSS, group2's examinees were supported in an incorrect way. Group3's examinees were not supported by anything. Therefore, it was expected that the influence on group2's score by SDSS use would not be identified. Also, it was assumed that there is significant difference between group2 and group3. The t-test for this combination was conducted by a two-tailed test to confirm if there is a significant difference. As a result, the t-value is 1.20, the P-value is 0.2342, and the degrees of freedom is 60. Since the P-value is more than significance level of 0.05, there is no significant difference between group2 and group3.

5 Evaluation Test Analysis

A significant difference between group1 and group2 was identified. Also, there was a significant difference between group1 and group3. By one-tailed testing confirmation, it can be said that group1's score is higher compared to group2 and group3. On the other hand, since no significant difference between group2 and group3 was identified, it can be said that there is no score difference in the case of inappropriate SDSS use.

In the t-test, the bigger the sample size is, the stronger the testing ability will be, and a significant difference is subject to occur. To examine whether this test is appropriate, the effect size was calculated using a spread sheet file created by Atsushi Mizumoto from Kansai University [8]. Figures are indicated in Table5.

Table 5. T-test Effect Size

Group	Effect Size	Interpretation
1, 2	0.39	Medium
1, 3	0.29	Small
2, 3	0.15	Small

Regarding group1, 2, effect size is more than 0.30, medium. Also in group1, 3, effect size is not negligible. Though 0.29 of its effect size is less than 0.30, they are very close in value. Therefore, regarding these two combinations of groups, the null hypothesis: There is no difference in scores between two groups is abandoned, and it can be said that group1's test score is higher than the other groups.

On the other hand, in regards to group2,3, the effect size is small. It would have something to do with the considerable possibility with which examinees happen to take tests at a time on which they feel alert because of individual differences, despite meal intake at an inappropriate time, which is different from the guided time by SDSS. To consider matters with this combination, more detailed analysis on inappropriate SDSS uses is required.

6 References

[1] National Cerebral and Cardiovascular Center Research Institute: Department of Molecular Pathogenesis/ Pathophysiological Clarification and Therapeutic development for diseases related to cerebral nerve/surgery,

http://www.ncvc.go.jp/res/divisions/etiology/et_005/index.html

[2] Japanese Association of Exercise Epidemiology・Journal RESEARCH IN EXERCISE EPIDEMIOLOGY Vol. 9, (March, 2007)

[3] Hyperphagia, severe obesity, impaired cognitive function, and hyperactivity associated with functional loss of one copy of the brain-derived neurotrophic factor (BDNF) gene.,Diabetes December 2006 vol. 55 no.12

[4] Brain-derived neurotrophic factor (BDNF) and food intake regulation,Bruno Lebrun, Bruno Bariohay, Emmanuel Moyse, André Jean Autonomic Neuroscience: Basic and Clinical Volume 126, Complete , pp.30-38, 30 June 2006

[5] Motoichi Adachi, Takayuki FUjimoto, "Proposal on Feeding Support Application Software based on Research into the Effect of a "Fasting State" on "Mental Alertness" ", The Internatinal Conference on Software Engineering Research & Practice（SERP2014）,July21-24, 2014, pp.291-295, USA

[6] Motoichi Adachi, Takayuki Fujimoto, "SDSS : Proposal on Feeding Support Application Software which Enables the User to Create a State of "Mental Alertness"", 1st IEEE/ACIS International Workshop on Information Design and Media Technology 2015, June 2015, USA

[7] Motoichi Adachi, Takayuki Fujimoto,"Putting SDSS into Practical Use: A Creativity Support System to Enable a State of"Mental Alertness"", International Journal of Computer Science and Network Security, VOL.15 No.11, November 2015

[8] Atsushi Mizumoto "Effect Size Calculation Sheet (MS Excel)" http://www.mizumot.com/stats/effectsize.xls

SESSION

ACCREDITATION AND RELATED ISSUES

Chair(s)

TBA

ISBN: 1-60132-435-9, CSREA Press ©

The Use of Course Benchmarking Technique (CBT) to Assess Students' Outcomes for ABET Accreditation

Ahmad A. Rabaa'i, and Aaron R. Rababaah[1]
[1]Department of Computer Science and Information Systems (CSIS), American University of Kuwait (AUK), AsSalmyah, Kuwait

Abstract – *Accreditation of educational programs is an increasingly important topic. However, educational program accreditation, in general, and ABET accreditation, in particular, could be daunting process which requires considerable time, effort and resources. This paper describes the implementation, use, and applicability of an effective, efficient and sustainable assessment method used by the Computer Science and Information Systems (CSIS) department at the American University of Kuwait (AUK) known as Course Benchmarking Technique (CBT). Other educational institutions seeking ABET program accreditation could benefit from utilizing this assessment method.*

.***Keywords:*** *Accreditation, Assessment, ABET/CAC, AUK, Student Outcomes, Kuwait*

1 Introduction and Background

Educational program accreditation has become an important topic around the globe. Program accreditation is a review process that educational programs in institutions undergo periodically for improvement of academic quality and for public accountability [1]. For instance, accreditation is considered as a type of quality assurance tool under which educational programs are evaluated by an external organization to determine if appropriate standards are met. If those standards are met, accredited status is granted by the external body [2].

Founded in 1932 under the name Engineers' Council for Professional Development, Accreditation Board for Engineering and Technology (ABET) accredits post-secondary degree-granting programs. ABET is a recognized accreditation agency of college and university programs in applied science, computing, engineering, and technology. ABET is a nonprofit organization whose responsibilities include "organizing and carrying out a comprehensive process of accreditation of pertinent programs leading to degrees, and assisting academic institutions in planning their educational programs" [3]. Its objective is to "promote the intellectual development of those interested in engineering and related professions, and provide technical assistance to agencies having engineering–related regulatory authority applicable to accreditation"[3]. The Computing Accreditation Commission (CAC) of ABET accredits computing programs, in particular Computer Science, Information Systems, and Information Technology. (See www.abet.org for more details).

The department of computer science and information systems (CSIS) in the college of Arts and Sciences at the American University of Kuwait (AUK) was established in 2004. The CSIS department is offering two separate Bachelor of Science degrees; Bachelor of Science in Computer Sciences and Bachelor of Science in Information Systems. Currently, the CSIS department is seeking ABET accreditation for the computer science program.

Part of the CAC-ABET accreditation requirements include students' outcomes assessment and continuous improvement (criterion 3 and 4) [3], [4]. This paper presents the course benchmarking technique (CBT) used by the CSIS at AUK to assess students' outcomes. In this paper, we focus on just three outcomes as example, and discuss the CBT we used to develop an assessment methodology. This paper aims at providing information on this technique for other educational institutions seeking ABET program accreditation. First, an abbreviated overview on ABET program assessment is presented. Section three lists the CSIS computer science program students' outcomes and a summary list of performance indicators which are used to assess the three students' outcomes as example. The proposed CBT is then discussed in section four followed by concluding remarks.

2 ABET Program Assessment

Program assessment is a critical component of attaining accreditation through the Computing Accreditation Commission of the Accreditation Board for Engineering and Technology (CAC-ABET) [1] [2] [4] [5] [6]. In fact, assessment of student work to assess how well students are meeting specified outcomes is an essential part of the ABET accreditation. For instance, a program seeking accreditation through ABET must have a periodic assessment process that determines the level of attainment of students' outcomes [4].

The assessment process includes collecting and analyzing the data to support a conclusion. It is essential to demonstrate that students' outcomes for the program are being measured and accomplished. In addition, your assessment process needs to show how results are applied to further improve your program (i.e. the continuous improvement process) [7].

ABET accreditation is outcomes-based assessment [9] [10] [11]. In 2000, ABET adopted what was at the time a revolutionary approach to accreditation through implementing outcomes-based assessment [6]. As stated by ABET, the purpose of this change was to focus on what students were learning as opposed to what students were being taught [12]. While the ABET criteria has been modified since that time, the concept of outcome-based assessment has remained the same [6].

In the definition section of the Criteria for Accrediting Computing Programs [5], ABET defines students' outcomes as "what students are expected to know and be able to do by the time of graduation. These relate to the knowledge, skills, and behaviors that students acquire as they progress through the program", while assessment is defined as "one or more processes that identify, collect, and prepare data to evaluate the attainment of student outcomes. Effective assessment uses relevant direct, indirect, quantitative and qualitative measures as appropriate to the outcome being measured. Appropriate sampling methods may be used as part of an assessment process." [5]. There are many ways in which programs can assess student outcomes, including both direct and indirect methods [1] [2] [6].

Though ABET does not provide a specific definition of "direct" or "indirect" assessment, Gloria Rogers effectively defines direct assessment as "Direct examination or observation of student knowledge or skills against measurable learning outcomes", and indirect assessment as "A process to ascertain the perceived extent or value of learning experiences" [13]. That is, "direct assessment is using a specific student artifact, such as an exam or report, and assessing against a set of criteria. Indirect assessment may include responses to a survey or questionnaire to determine a "perception" of how well a student achieved a certain outcome" [6]. Menhart [14] states that direct assessment methods, such as: tailored exam questions, quizzes, and laboratory assignments measure students' performance and allow faculty members of a program to ultimately assign student grades. On the other hand, indirect assessment methods, which are less applicable to students' outcomes and more so to the program educational objectives, include senior exit surveys, alumni survey, industry advisory board input and employers' survey, do not directly measure students' performance, but can provide useful information.

Additionally, Menhart states "Direct assessment via exam questions, quizzes, and lab reports provides a definitive measure of the students' capabilities. Indirect assessment can only complement direct measures. However, direct assessment usually requires more time to structure and obtain. It may not be as easy as the indirect method to compare data from different semesters because the actual exam questions etc., will most likely change in different course offerings."

3 Students' Outcomes (SOs)

ABET requires each computing program seeking accreditation to develop a clear set of students' outcomes (SOs), collect assessment data, determine the extent to which the outcomes are attained, and use the results of the evaluation to improve the program [1]. SOs describe what the students are expected to know and be able to do at the time of graduations [8]. These outcomes should relate to the learning domains such as: knowledge, cognitive skills, interpersonal skills, and communication that students acquire as they progress through the program [8].

In their criteria for the accreditation of all computing programs, CAC-ABET provides a well-known list of expected SOs. The list is augmented by additional characteristics under specific program criteria for computer science, information systems and information technology [1]. For computer science programs this list is commonly referred to as the "a through k" outcomes. ABET allows programs to use additional outcomes; however, it is required that the "a through k" outcomes are used and assessed by each program [6].

The CSIS department adopted the same "a through k" SOs list as provided by the CAC-ABET [5], these include: (a) an ability to apply knowledge of computing and mathematics appropriate to the program's student outcomes and to the discipline, (b) an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution, (c) an ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs, (d) an ability to function effectively on teams to accomplish a common goal, (e) an understanding of professional, ethical, legal, security and social issues and responsibilities, (f) an ability to communicate effectively with a range of audiences, (g) an ability to analyze the local and global impact of computing on individuals, organizations, and society, (h) recognition of the need for and an ability to engage in continuing professional development, (i) an ability to use current techniques, skills, and tools necessary for computing practice, (j) an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices, and (k) an ability to apply design and development principles in the construction of software systems of varying complexity.

It is apparent that these SOs are broad statement "encompassing fundamentals of computer science and mathematics appropriate to computer science" [7]. As such, the first step in assessing SOs is to break down these outcomes into simpler measureable factors that allow one to determine the extent to which the outcome is met [6]. As prescribe in

ABET literature, these measurable factors are known as "performance indicators" or "performance criteria". As mentioned previously, in this paper we chose three SOs; a, c, and j. The following is a list of the performance indicators for these outcomes.

Outcome A:

- A1: Identifies an appropriate mathematical or computing concepts/skill required to solve a specific problem.
- A2: Can explain a particular concept/skill within computing or mathematics when asked to do so.
- A3: Successfully applies a mathematical or computing concept/skill to solve a specific problem.

Outcome C:

- C1: Use common modeling techniques to design a solution.
- C2: Writes computer programs that solves a problem
- C3: Writes test cases for a problem and/or implementation.
- C4: Students can use and choose tools for modeling a design solution

Outcome J:

- J1: Demonstrates an ability to apply mathematical modeling to computing problems
- J2: Demonstrates an ability to develop different algorithms for a computing problem
- J3: Demonstrates an ability to evaluate algorithm efficiency
- J4: Demonstrates an understanding of Complexity classes their models of computation and their resources for a given problem.

The above list was chosen after careful evaluation and discussion of different performance indicators proposed by faculty members at the CSIS department. Three performance indicators were used to assess SO A, while four performance indicators were used to assess SOs C and J. SOs measured by these indicators collectively provide a reasonable basis for stating claims on the extent to which the SOs A, C and J are attained. Similar sets of performance indicators have been developed for all other SOs, but will not be reported and discussed in this paper.

4 The Proposed Method - CBT

As aforementioned, direct assessment is essential, but its collection and efficient use continues to pose many problems [13]. Assessment, in general and direct assessment in particular, can be time consuming and minimizing faculty time is of key importance [15]. With regard to ABET, the greatest difficulty encountered with direct assessment is taking this data and using it routinely in a continuous improvement process, without overwhelming faculty members who are participating in the assessment process [14].

ABET provides sufficient flexibility for a program to tailor its particular assessment approach based upon its own unique constraints [16]. Among the many direct assessments alternatives, in this section we relate to our experience with the course benchmarking technique (CBT) used in the CSIS department. The proposed CBT is effective, efficient and sustainable and will ease the data collection as well as the assessment processes and will put less burden on the faculty members involved in the assessment process.

As recommended by Rao et al [17], we insured that our SOs and associated PIs were supported by the core courses of the CS program. Table 1 presents all core courses of the CS program at AUK.

Table 1: CS Program Core courses

Course No.	Course Name
CSIS120	Programming I
CSIS130	Programming II
CSIS150	Ethical Issues in CSIS
CSIS210	Data Structures and Algorithms
CSIS220	Computer Organization & Architecture
CSIS250	Database Systems
CSIS255	Web Technologies
CSIS310	Operating Systems
CSIS320	Principles of Programming Languages
CSIS330	Software Engineering
CSIS401	Mobile Computing
CSIS405	Analysis of Algorithms
CSIS476	Computer Security
CSIS490	Capstone Project I
CSIS491	Capstone Project II

In the following three sub-sections we will present case studies of the aforementioned SOs namely A, C and J.

4.1 SO A

As it can be seen in Figure 1 below, SO A is mapped to several courses and only being assessed to the selected benchmark course of CSIS405 – Analysis of Algorithms[1]. Figure 4 presents the completed assessment form. For each assessment form, three items are reported as follows:

Observations:
A1 meets the expectations but its borderline.
A2 does not meet expectations.
A3 exceeds expectations.

Analysis & Justification:
We believe that the results of A2 were due to two root causes: (1) Our experience (in class and through their written tests) with the students shows that their spoken/written English is

[1] x = this implies that the PI is enabled at an introductory, intermediary or advanced levels but will not be assessed. A = this implies that the PI is enabled at an introductory, intermediary or advanced levels and will be assessed against that SO.

not satisfactory to allow them explain concepts well. (2) Some students complained about the length of the exam where, this question was the last one in the exam, by the time they reached it they were either tired or ran out of time.

Recommendations:

All issues were discussed in department meetings (Feb 9, Feb 16 & Feb 23, 2016) and the following were reached: A1: Keep a close eye on it since it is borderline. If performance drops next assessment cycle, we must investigate A1 and devise a plan of action forward. A2: Recommended solutions: (1) “Concept explanation" should be emphasized in lower division courses such as: CSIS120, 130 and 210. A final exam question will be added to these courses to address this issue. (2) The length of the test was dictated by its purpose to assess all learning objectives, in addition to PIs of outcome A and outcome J. Therefore, test 4 should be dedicated to assessing outcome A and the Final exam should be dedicated to assessing outcome J. Left to the discretion of the instructor, some questions may be designed to assess more than one PI. A3: No action was required this assessment cycle.

4.2 SO C

As shown in Figure 2, SO C is mapped to several courses and only being assessed to the selected benchmark course of CSIS330 – Software Engineering. Figure 5 presents the completed assessment form. The results of this assessment form were as follows:

Observations:
Sample size is very small (3 CS students)
C1 Different methods of assessments where used (project and Final exam)
C1 and C2 meet expectation
C3 does not meet expectation

Analysis & Justification:
It was noticed that 2 out of 3 assessed students lack the required skills which address PI C3. These students were not able to write test cases for a problem and/or implementation.

Recommendations:
All issues were discussed in department meetings (Mar 1 & Mar 8, 2016) and the following were reached: C1: Use less assessment methods to assess this PI. This should ease the process next assessment cycle. C3: Recommended solutions: (1) students should be exposed to testing in CSIS210, (2) While CSIS260 - Systems analysis and design is an elective course, CSIS260 instructor should emphasize this issue. (3) Instructor of CSIS330 should emphasize testing in more assessed activities and observe the results. C2: No action was required this assessment cycle.

4.3 SO J

As can been seen in Figure 3, SO J is mapped to several courses and only being assessed to the selected benchmark course of CSIS405 – Analysis of Algorithms. Figure 6 presents the completed assessment form. The results of this assessment form were as follows:

Observations:
J1: exceeded expectations.
J2, J3 and J4: do not meet expectation

Analysis & Justification:
J2: 4 students from the class did not attempt answering question 4 in the exam. This amounts to 40% of the class, a significant percentage given the unreliably small sample size (10 CS students in total). Some students complained about the length of the exam. All learning objectives, in addition to PIs of outcome A and outcome J were to be assessed in this test. J3&J4: We believe that students are not spending enough time and effort to do their homework assignment themselves which affects their grasping the material through hands-on.

Recommendations:
The issues were discussed in department meetings (Feb 16 & 23, 2016) and the following were reached: J2: (1) needs to be addressed in at least 2 assessments (tests), in a format that is introduced prior to the test (in-class example/assignment), to eliminate factors attributing to poor exam attempts. (2) Make Test 4 dedicated to assessing outcome A and the Final for outcome J. Left to the discretion of the instructor, some questions may be designed to assess more than one PI. J3&J4: It is necessary to emphasize the importance of assignments, and discourage outsourcing them. Random samples of students and assignments must be taken and their work is to be verified by asking the students to explain their assignments. +ve credit can be given to those who show strong evidence that they have done the work and –ve credit can be given otherwise.

		Computer Science Core Courses														
	Learning Outcomes	**120**	**130**	**150**	**210**	**220**	**250**	**255**	**310**	**320**	**330**	**401**	**405**	**476**	**490**	**491**
a	**An ability to apply knowledge of computing and mathematics appropriate to the discipline**	X	X		X	X			X	X			X		X	X
a1.	Identifies an appropriate mathematical or computing concepts/skill required to solve a specific problem	x	x		x	x			x	x			A		x	x
a2.	Explain a particular concept/skill within computing or mathematics when asked to do so	x	x		x		x		x	x			A			x
a3.	Successfully applies a mathematical or computing concept/skill to solve a specific problem	x	x		x	x			x	x	x		A		x	x

Figure 1: SO A PIs Definitions and Curricula Mapping

		Computer Science Core Courses														
	Learning Outcomes	**120**	**130**	**150**	**210**	**220**	**250**	**255**	**310**	**320**	**330**	**401**	**405**	**476**	**490**	**491**
c	**An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs**		X		X	X	X		X	X	X		X			X
c1.	Use common modeling techniques and modelling tools to design a solution						x				A					x
c2.	Writes computer programs that solves a problem		x		x	x			x	x	A		x			x
c3.	Writes test cases for a problem and/or implementation		x		x						A					x

Figure 2: SO C PIs Definitions and Curricula Mapping

		Computer Science Core Courses														
	Learning Outcomes	**120**	**130**	**150**	**210**	**220**	**250**	**255**	**310**	**320**	**330**	**401**	**405**	**476**	**490**	**491**
j	**An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices**				X				X				X		X	X
j1.	Demonstrates an ability to apply mathematical modeling to computing problems				x								A			
j2.	Demonstrates an ability to develop different algorithms for a computing problem				x								A			
j3.	Demonstrates an ability to evaluate algorithm efficiency				x				x				A			
j4.	Demonstrates an understanding of Complexity classes their models of computation and their resources for a given problem											x	A			

Figure 3: SO J PIs Definitions and Curricula Mapping

Student Outcome (A)

SLO(A): *An ability to apply knowledge of computing and mathematics appropriate to the discipline*

Performance Measures/Appraisals	List of courses which address the PI	Where data are collected	Assessment method	Semester and year of data collection	Inadequate 1	Approaches Standard 2	Meets Standard 3	Exceeds Standard 4	Poor <= 40%	Good/Excl >= 60%
					Sample shows little ability to achieve the PI	Sample shows some modest ability to achieve the PI with several important flaws	Sample shows a reasonable ability to achieve the PI with a few minor flaws	Sample shows an excellent ability to achieve the PI		
A1 – Identifies an appropriate mathematical or computing concepts/skill required to solve a specific problem	**CSIS 120, 130, 210, 320, 405, 490, 491**	CSIS 405	Final Exam Q1	Fall 2015	4	0	6	0	40.0%	60.0%
A2. – Explain a particular concept/skill within computing or mathematics when asked to do so	**CSIS 250, 320, 405**	CSIS 405	Final Exam Q9	Fall 2015	0	5	2	2	55.6%	44.4%
A3. – Successfully applies a mathematical or computing concept/skill to solve a specific problem	**CSIS 120, 130, 210, 320, 330, 405, 490, 491**	CSIS 405	Final Exam Q7	Fall 2015	0	3	3	4	30.0%	70.0%
								Overall	**41.4%**	**58.6%**
								PI Satisfied?	**no**	**no**

Figure 4: Assessment Data of SO A

Student Learning Outcome (C)

SLO(C): An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs

Performance Measures/Appraisals	List of courses which address the PI	Where data are collected	Assessment method	Semester and year of data collection	Inadequate 1	Approaches Standard 2	Meets Standard 3	Exceeds Standard 4	Poor <= 40%	Good/Excl >= 60%
					Sample shows little ability to achieve the PI	Sample shows some modest ability to achieve the PI with several important flaws	Sample shows a reasonable ability to achieve the PI with a few minor flaws	Sample shows an excellent ability to achieve the PI		
C1.- Use common modeling techniques and modelling tools to design a solution	CSIS-130,210, 250, 320, 330, 405, 491	CSIS-330	Project - Design Document Final Exam (Q1, 3, 4, 5)	Fall 2015		2	4		33.3%	66.7%
C2. - Writes computer programs that solves a problem	CSIS-130,210, 250, 320, 330, 405, 491	CSIS-330	Project - Demo	Fall 2015		1	2		33.3%	66.7%
C3.- Writes test cases for a problem and/or implementation	CSIS-130,210, 320, 330, 405, 491	CSIS-330	Final Exam - Q2	Fall 2015	2			1	66.7%	33.3%
								Overall	41.7%	58.3%
								PI Satisfied?	no	no

Figure 5: Assessment Data of SO C

Student Learning Outcome (J)

SLO(J-CS Major): An ability to apply mathematical foundations, algorithmic principles, and computer theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

Performance Measures/Appraisals	List of courses which address the PI	Where data are collected	Assessment method	Semester and year of data collection	Inadequate 1	Approaches Standard 2	Meets Standard 3	Exceeds Standard 4	Poor <= 40%	Good/Excl >= 60%
					Sample shows little ability to achieve the PI	Sample shows some modest ability to achieve the PI with several important flaws	Sample shows a reasonable ability to achieve the PI with a few minor flaws	Sample shows an excellent ability to achieve the PI		
J1. Demonstrates an ability to apply mathematical modeling to computing problems	CSIS 210, 405	CSIS 405	Final Exam Q7	Fall 2015	0	3	3	4	30.0%	70.0%
J2. Demonstrates an ability to develop different algorithms for a computing problem	CSIS 210, 405	CSIS 405	Final Exam Q4	Fall 2015	6	3	1	0	90.0%	10.0%
J3. Demonstrates an ability to evaluate algorithm efficiency	CSIS 210, 405	CSIS 405	Final Exam Q2	Fall 2015	4	2	4	0	60.0%	40.0%
J4. Demonstrates an understanding of Complexity classes their models of computation and their resources for a given problem	CSIS 210, 405	CSIS 405	Test 3 Q3	Fall 2015	5	1	3	0	66.7%	33.3%
								Overall	61.5%	38.5%
								PI Satisfied?	no	no

Figure 6: Assessment Data of SO J

5 Conclusions

While the importance of direct assessment for CAC-ABET accreditation cannot be over-stated, faculty involved in the assessment process recognize that assessment tool utilization and analysis of results is a time-consuming activity and ways to minimize the time-commitment is appreciated [16] [17] [18].

This paper describes the implementation, use, and applicability of an effective, efficient and sustainable assessment method used by the Computer Science and Information Systems (CSIS) department at the American University of Kuwait (AUK) to assess the CS program, known as Course Benchmarking Technique (CBT) and uses three SOs, namely A, C and J as case studies. Mapping of SOs and their associated PIs to core CS courses, data collection, analysis along with data interpretation is presented herein to give an overview of the evaluation process. Other educational institutions seeking ABET program accreditation could benefit from utilizing the proposed CBT.

6 Acknowledgment

The authors acknowledge the contributions of other faculty members in the CSIS department at AUK in the development of the student outcomes, the performance indicators as well as the rubrics, the assessment and evaluation. The authors also acknowledge the continuous support and guidance from the Dean of the Arts and Sciences College, Prof. Rawda H. Awwad and the Associate Dean, Prof. Ali M. Charara.

7 References

[1] Lakshmanan, K. B., Sandeep R. Mitra, and T. M. Rao. "Assessing Student Learning in Computer Science-A Case Study." Proceedings of the International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2013.

[2] Al-Yahya SA, Abdel-Halim MA. “A successful experience of abet accreditation of an electrical engineering program. Education”, IEEE Transactions on. Vol. 56 No. 2, 165-73, (May 2013).

[3] ABET. “About ABET [Online]” available at: http://www.abet.org/about-abet/ accessed 3rd of March, 2016.

[4] Somervell, Jacob P. "Assessing Student Outcomes with Comprehensive Examinations: A Case Study." Proceedings of the International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2015.

[5] ABET. “Criteria for Accrediting Computing Programs, 2016-2017 [online]” available at: http://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-computing-programs-2016-2017/ accessed 3rd of March, 2016.

[6] Matthew A. Dettman “Developing a Sustainable Assessment Plan for EAC-ABET Accreditation.” Proceedings of 11th Latin American and Caribbean Conference for Engineering and Technology. (August 14 - 16, 2013).

[7] Shryock, Kristi, and Helen Reed. "ABET accreditation: Best practices for assessment." Proceedings of the ASEE Annual Conference and Exposition, Austin, TX. 2009.

[8] Alzubaidi, Loay, Israa Jabur, and Rihab Alzubaidi. "Course Assessment Tool Using the Combination Approach." Proceedings of the World Congress on Engineering and Computer Science. Vol. 1. (October, 2015).

[9] Crouch, D, B., Schwartzman, L, “Computer Science Accreditation: The Advantage of Being Different,” Proceedings of the 34th SIGCSE. Technical Symposium on Computer Science Education, 36-40, 2003.

[10] Yaverbaum, G, Lidtke, D, Reichgelt, H, Zweben, S, “Outcomes-based Computing Accreditation: Program Assessment,” Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education, 281-282, 2007.

[11] Rigby, S, Dark, M, “Using Outcomes-based Assessment Data to Improve Assessment and Instruction: A Case Study,” ACM SIGITE Newsletter, Vol. 3, No. 1, 10-15, 2006.

[12] ABET. “History of ABET [Online]” available at: http://www.abet.org/about-abet/history/ accessed 3rd of March, 2016.

[13] Rogers, Gloria. “Assessment 101”, Community Matters: ABET Community Newsletter, (April 2006).

[14] Menhart, Steve. "Direct Versus Indirect Assessment Methodologies." Proceedings of the 2011 Midwest Section Conference of the American Society for Engineering Education, Russellville, AK. (2011).

[15] Yue, Kwok-Bun. "Effective course-based learning outcome assessment for ABET accreditation of computing programs." Journal of Computing Sciences in Colleges, Vol. 22, No. 4, 252-259, (2007).

[16] Kerbs, Robert. “An Accreditation Framework for Global Markets.” Proceedings of the 37th ASEE/IEEE Frontiers in Education Conference, (2007).

[17] Rao, M, R, Junaidu, S, Maghrabi, T, Shafique, M, Ahmed, M, et al., “Principles of Curriculum Design and Revision: A Case Study in Implementing Computing Curricula CC2001,” Proceedings of the 10th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education, 256-260, 2005.

[18] Hogan, T, R., Harrison, P, R., Schulze, K, G., “Developing and Maintaining an Effective Assessment Program,” ACM SIGCSE Bulletin, Vol. 34 No. 4, 52-55, (December 2002).

Accreditation: Observations of a recent program evaluator

Benjamin Bishop
Department of Computing Sciences, University of Scranton, Scranton, PA, USA

Abstract – *ABET accreditation can be a valuable tool for improving the quality of computing programs. However, many computing departments have important misconceptions about the accreditation process. This work provides guidance, and identifies a number of common pitfalls.*

Keywords: ABET, accreditation, assessment, education

Short research paper

1 Introduction

Many computing programs benefit from the ABET accreditation and review process. However, misunderstandings of the requirements create problems for these departments, and prevent other departments from seeking accreditation. This paper is intended to assist the audience in avoiding a number of common issues. This information is presented entirely from the perspective of a single program evaluator, and is not intended to speak on behalf of ABET.

1.1 Terminology

In order for a computing program to become accredited (or remain accredited), it must meet general criteria [1] and program-specific criteria[1] [2] from ABET's Computing Accreditation Commission. Additionally, the ABET Accreditation Policy and Procedure Manual [3] must be adhered to. Accreditation decisions are based on the presence or absence of **shortcomings** identified during a program review with respect to the criteria. In order of seriousness, these shortcomings are designated as program **deficiencies**, **weaknesses** or **concerns**.

A deficiency represents an area where the program fails to meet the requirements of the criteria. This sort of shortcoming is very serious, and ultimately prevents programs from attaining accreditation. For example, CAC general criterion 4 requires that "the program must regularly use appropriate, documented processes for assessing and evaluating the extent to which the student outcomes are being attained". If a program lacked such a continuous improvement process, it would be a clear deficiency.

Weaknesses are areas where the program doesn't fully meet the standard set by the criteria, but does comply in some ways. Shortcomings at this level still allow for program accreditation, but may require more frequent reviews. As an example, CAC general criterion 1 requires that "students must be advised regarding curriculum and career matters". If, during the program review, the team finds that an advising process is in place, but is not entirely adequate to serve the needs of the students, it may be appropriate to note this shortcoming as a weakness.

Concerns identify areas where the program currently meets the criteria, but there is a danger that the program will fall out of compliance if there is some change in circumstances. These shortcomings do not impact accreditation decisions, but may be helpful to consider in order to prevent future problems. Note that concerns **are not** a less serious form of a weakness - programs with only concerns fully comply with the criteria. As an example, the CAC program criteria for Computer Science require that "some full-time faculty members must have a Ph.D. in Computer Science". If a department has three non-doctoral junior faculty and three doctoral senior faculty that are all nearing retirement, a concern may exist that these senior faculty members may decide to retire simultaneously due to an institutional retirement incentive program leaving the program out of compliance.

1.2 Accreditation process

ABET provides a detailed outline of the accreditation process [4]. From the perspective of the institution, some of the most important steps are the request for evaluation, self-study and visit.

For programs that are currently accredited, the request for evaluation (RFE) initiates the review process. Currently, the RFE is due in January of the year that the program is seeking accreditation.

[1] Whether specific program criteria apply depends on the name of the academic program.

The self-study is document that is prepared by the institution describing the program and the context in which the program is offered. The self-study is one of the most important documents involved in the process. It is reviewed very carefully by the visiting team prior to the on-site visit.

The on-site visit involves a team of volunteers from the relevant ABET commission who examine the program over the course of around three days. The visitors will interview faculty, staff, students and administrators, tour facilities and review documentation. This allows the team to validate information from the self-study and gather additional information. Volunteers are normally diplomatic and operate in the spirit of genuine program improvement. The visit is intended to be a cooperative rather than adversarial experience.

2 Observations

In this section, a number of observations/suggestions are listed with the intention of assisting programs that are seeking accreditation. They are listed in roughly order of decreasing importance.

2.1 Manage expectations/resources

Some of the most disastrous visits are caused by a mismanagement of expectations. This may involve a confident chairperson assuring the institutional leadership that accreditation is a "sure-thing". Under no circumstances should such expectations be created.

Accreditation is a complex process. It is difficult to foresee all issues that may arise during a visit. For this reason, some uncertainty must always be accepted – no matter how well-prepared a department feels. Additionally, faculty must be prepared to take on the work required for assessment. It may be appropriate to provide resources (i.e. release time) to help offset this workload.

2.2 Understand what is required of the team

ABET program evaluators are not compensated for their efforts. A program evaluator may spend one week reviewing the self-study and transcripts prior to the visit. Additionally, it would not be unusual for the program evaluator to spend all waking hours working during the visit itself (evenings may be spent analyzing information and preparing documentation). As a result, program evaluators have very little time to "dig" for missing information. A well-prepared institution will provide all necessary documentation and will follow-up with the team to ensure that it has all information that it needs.

2.3 Student outcomes vs. Program educational objectives

A significant amount of confusion exists about student outcomes (SOs) and program educational objectives (PLOs). PLOs are defined [5] as "broad statements that describe what graduates are expected to attain within a few years of graduation. Program educational objectives are based on the needs of the program's constituencies". Student outcomes are defined as "statements that describe what students are expected to know and be able to do by the time of graduation. These relate to skills, knowledge, and behaviors that students acquire as they progress through the program".

In other words, SOs are relatively straight-forward, and can be tested. On the other hand, PLOs reflect the overall aim of what the program hopes to accomplish.

2.4 Enabling outcomes vs. assessment of outcomes

CAC general criterion 3 requires that "the program must enable students to attain, by the time of graduation…" a list of specific student outcomes. This enabling typically involves coverage of these topics in particular courses. This does not mean that every student necessarily must meet every outcome at graduation, or that every aspect of these outcomes must be assessed – it is that students must have had the opportunity to meet the outcome.

2.5 Accreditation applies to programs

Accreditation applies specifically to programs rather than departments or courses. As a result, it is necessary to separate out different populations during assessment. For example, in departments that have both IT and CS programs, these populations should be segregated during assessment in courses that might serve both majors.

2.6 Department culture

ABET's requirements should be interpreted from the perspective of genuine program improvement rather than "going through the motions". Particularly with regard to assessment, unless department leadership is genuinely interested in using assessment as a tool for improving programs, faculty participation will be half-hearted.

2.7 "Bean counting"

Program criteria may require a certain number of credits in particular areas. For example, the CAC Computer Science program criteria requires "one year of science and mathematics…" A year is interpreted to mean 24 credits.

Programs that do not meet these requirements are clearly deficient. So, it is important to ensure that credit requirements are met prior to seeking accreditation. One other issue to be careful about is that the visiting team may disagree with the institution about the classification of a particular course. For example, a department may wish to count discrete mathematics within the 1 1/3 years of Computer Science courses, but the team classifies the course within the one year of science and mathematics.

2.8 Documentation

Documentation is essential to the work of every visiting team. In addition to documents such as transcripts, academic catalogs etc., the team may review minutes of department, assessment committee, and industry advisory board meetings. There is often significant variation between institutions with regard to the availability of this documentation.

Institutions should ensure that all relevant documentation is being collected. If not, what changes can be made to ensure such documentation is developed? Is there a staff person that could accept these duties?

2.9 "Closing the loop"

Assessment is an iterative process. For example, some assessment instrument may show opportunity for improvement in a particular area. In analyzing the data, the assessment committee may recommend a change in order to improve this metric. The relevant faculty member may actually implement the change. Then the assessment committee may follow-up with a reassessment to determine if the change has had the desired effect. This cycle is then repeated, leading to continuous improvement.

An assessment process that lacks one or more of these critical steps has not "closed the loop" and is incomplete. Figure 1 shows a graphical overview of this process. It is important to note that periodic revision of Program Educational Objectives and Student Outcomes is also required.

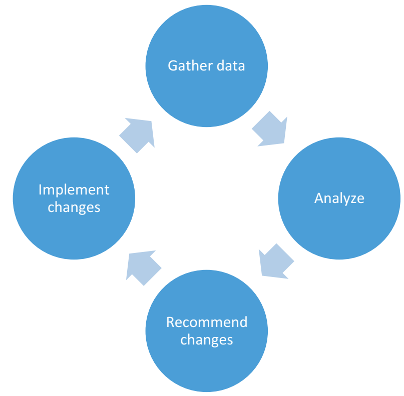


Figure 1 – The assessment process

2.10 Distribution of work

It may be somewhat unusual for 100% of the faculty in a department to enthusiastically carry out the work required for accreditation. Some variation in level of commitment may be expected. When distributing work, redundancy and realistic planning for individual failure may be valuable.

The concept of a "weakest link" may be helpful. Even if 90% of the work is outstanding, if the remaining 10% is incomplete and represents a critical piece, overall accreditation may be jeopardized. It is helpful to have a small group of reliable faculty that may be able to step in under such circumstances.

2.11 Assessment instruments

The strongest assessment techniques involve **direct** assessment – that is actually measuring what students have learned. For example, a student's oral presentation ability could be evaluated using a rubric (direct), rather than asking the student to evaluate their own speaking ability (indirect).

It is necessary to be very careful when involving graded coursework in assessment. This is because students are often graded based on factors unrelated to what is being assessed. As an example, it may be appropriate to assign a poor grade to a student that did not submit a particular assignment. However, it would not be appropriate to assign a poor score for assessment purposes – lack of data is not the same as negative data.

2.12 Tailor assessment to institutional needs

A great deal of flexibility exists for institutions to define their own PEOs/SOs based on institutional priorities. By making these priorities part of the assessment process, it is possible to ensure that the institutional character is preserved and expanded. For example, if an institution values community engagement, PEOs and

SOs could be extended to include items related to this priority.

2.13 Excellence vs. minimum performance

There may be a perception that assessment is about setting a minimum bar that all students are expected to meet. However, assessment targets can also be set to focus on excellence. For example, rather than having a target of 90% of students meeting some minimal hurdle, 20% of students could be expected to reach some much higher hurdle. Again, these decisions should be reached based on institution-specific priorities

2.14 Openness

The institution should always try to be as open as possible during the accreditation process. In particular, if any irregularity exists with regard to waiving/modifying requirements, it is better to be up-front and direct in explaining these issues. A common example would be constraints imposed on institutions due to articulation agreements.

Transcripts should be truly representative of the student population. If the team later suspects that the transcript sample was not representative of the student population, trust will be undermined.

2.15 Reactions to findings

The most constructive reaction to adverse findings is to begin working as soon as possible in order to address the problem. ABET's accreditation process allows time for an institutional due process response. Depending on the particular shortcomings identified, this may be an opportunity to partially or fully correct the issue prior to the final accreditation decision.

2.16 Leverage existing structures

In many cases, it is possible to implement assessment smoothly by making use of structures that are already in place. For example, are there particular courses that would be a good fit for assessment activities – a capstone course, a first-year seminar perhaps?

What about opportunities for engaging stake-holders for PEO/SO revision? Does the department already run an industrial advisory board? Alumni events? Meetings with large employers?

3 Conclusions

In this paper, a number of observations regarding the ABET accreditation process were presented. These observations were intended to aid institutions in gaining accreditation as well as extracting the maximum possible benefit from the accreditation process.

4 References

[1] http://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-computing-programs-2016-2017/#generalcriteria

[2] http://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-computing-programs-2016-2017/#programcriteria

[3] http://www.abet.org/accreditation/accreditation-criteria/accreditation-policy-and-procedure-manual-appm-2016-2017/

[4] http://www.abet.org/accreditation/get-accredited-2/get-accredited-step-by-step/

[5] http://www.abet.org/network-of-experts/for-current-abet-experts/training/glossary/

SESSION

POSTER PAPERS

Chair(s)

TBA

ISBN: 1-60132-435-9, CSREA Press ©

SAS Programming in the Pharmaceutical Industry

WORLDCOMP'16 Poster Paper Submission

Chris Helwig (EMCC)
and
Julia Bao (U. of Phoenix)
juliabao@email.phoenix.edu

ABSTRACT

This project examines the use of SAS, which is both a statistical programming platform and a programming language, in the pharmaceutical industry for the purpose of analyzing clinical trial data used to gain approval of new drugs and medical devices. This material has been used to teach computer information students about statistical programming techniques, as well as to acquaint them with the field of pharmaceutical industry software development.

General Terms

Computer Science Education, SAS, Pharmaceutical, Biostatistics.

Keywords

SAS, Computer Science Education, Pharmaceutical, Biostatistics.

1. Clinical Research

The Part of process of discovering new drugs and registering them for FDA approval and marketing. [1]

Pharmaceutical companies and research agencies engage in drug discovery, development, clinical trials, manufacturing, and marketing approval. [1]

2. SAS

SAS is a statistical software system used in a range of industries. [2]

SAS uses data steps to read data into memory and proc steps to calculate descriptive or inferential statistics, generate summary reports, and create summary graphs and charts. [3]

```
Data InvestigatorName;
Input Country$ Site Investigator_Name$;
Datalines;
US 1 Name1
US 2 Name2
US . Name3
;
Run;  [2]
```

SAS's implementation of structured query language, can sort, summarize, select, subset, join and concatenate datasets, treating the dataset conceptually like a database table with rows and columns. [4]

```
Proc SQL;
Create table NewTable as
Select * from Demographic_Data;
Quit;
```

SAS Macros can be used to perform repetitive tasks quickly and efficiently. [5]

Macro programs start with %, Macro variables start with &. [5]

Use %Let statement, a global statement to name a macro and give it a value. [6]

The SAS proc tabulate procedure can arrange data from a data set in table format. [7] Consider a table with columns for each dosage level and rows showing number of patients, medicine effectiveness, adverse event or side affect.

Also included would be detailed demographic data on race, age, gender, and whether subject was taking other medication. [8]

```
Data Drug_Study;
   input gender $ race $ drug $ @@;
/* Format is M or F, race, and whether drug is Active or Placebo
*/
   proc tabulate data = drug;
   table (gender race),
     drug*(n pctn<gender race>='%');
/* Generate s a table showing columns with Active and Placebo
with a count (N) and percentages broken down by race and gender
*/
```

SAS also has a graph capability to generate charts. The general form of the command is:

```
Proc Gchart Data=Demographic_Data;
Vbar Race;
Run; [9]
```

ISBN: 1-60132-435-9, CSREA Press ©

SAS also generates charts showing frequency distributions using proc freq.

3. Examples

Suppose there is some sample clinical data related to a diet pill, and an investigator wants to determine if the baseline average weight of study participants has gone up or gone down. SAS can calculate such baseline comparisons. [10]

Or consider an example of a clinical trial of a drug to remove warts. SAS On Demand for Academics was used to run the SAS code. [11]

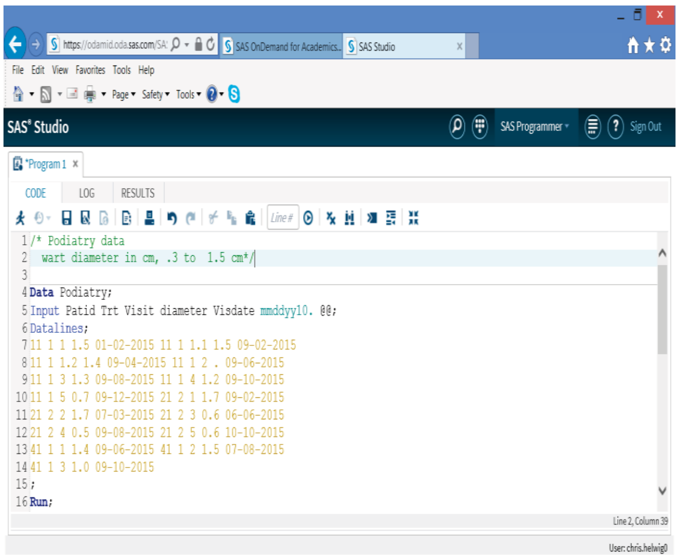


Figure 1. Screen Shot of SAS on demand for Academics

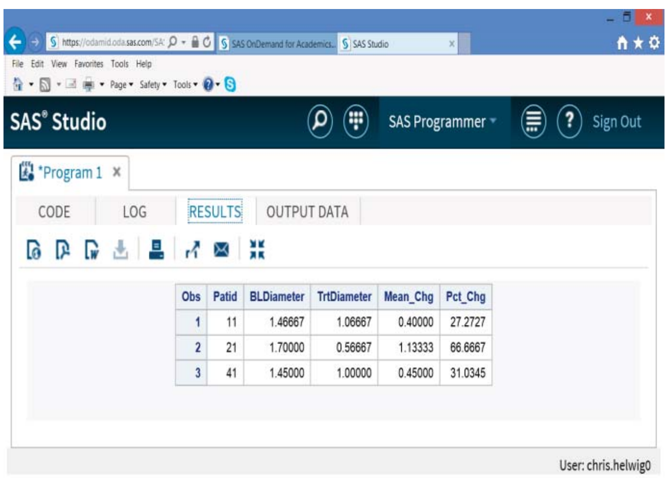


Figure 2. Output of basline comparison example

```
/* Podiatry data
  wart diameter in cm, .3 to  1.5 cm*/

Data Podiatry;
Input Patid Trt Visit diameter Visdate mmddyy10. @@;
Datalines;
11 1 1 1.5 01-02-2015 11 1 1.1 1.5 09-02-2015
11 1 1.2 1.4 09-04-2015 11 1 2 . 09-06-2015
11 1 3 1.3 09-08-2015 11 1 4 1.2 09-10-2015
11 1 5 0.7 09-12-2015 21 2 1 1.7 09-02-2015
21 2 2 1.7 07-03-2015 21 2 3 0.6 06-06-2015
21 2 4 0.5 09-08-2015 21 2 5 0.6 10-10-2015
41 1 1 1.4 09-06-2015 41 1 2 1.5 07-08-2015
41 1 3 1.0 09-10-2015
;
Run;
Proc Means Data=Podiatry (where = (1<=visit<=2)) Noprint;
By patid;
Var diameter;
Output out=BL(drop=_TYPE_ _FREQ_) mean=BLDiameter;
Run;
Proc means data=Podiatry (where = (3<=visit<=5)) Noprint;
By patid;
Var diameter;
output out=DDT(drop =_TYPE_ _FREQ_) mean=TrtDiameter;
Run;
DATA Change_From_Baseline;
Merge BL DDT;
by Patid;
Mean_Chg = BLDiameter - TrtDiameter;
Pct_Chg = (BLDiameter - TrtDiameter)/BLDiameter*100;
Run;
Proc Print data=Change_From_Baseline; [10]
```

4. CDISC ADaM

CDISC, which stands for Clinical Data Interchange Standards Consortiam, is a standards organization for the clinical data field. [12] Part of this standard consists of the Analysis Data Model, which includes items such as variable name, label, data type, length, and comment.

5. CDISC SDTM

SDTM is the CDISC standard for the Study Data Tabulation Model, which governs human clinical trial data tabulations submitted to the Food and Drug Administration (FDA) for approval. This model provides detailed guidelines for how various medical interventions, findings, and measurements need to be documented, presented and statistically analyzed prior to submission as part of a regulatory approval process. Interventions refer to therapeutic treatments or surgical procedures; findings refer to lab test results, ECG results, or other observational evaluations. [13]

6. Biostatistics

Biostatistics is the "application of statistics to a wide range of topics in biology. The science of biostatistics encompasses the design of biological experiments, especially in medicine" and drug studies; "the collection, summarization, and analysis of data from those experiments; and the interpretation of, and inference from, the results. A major branch of this is medical biostatistics, which is exclusively concerned with medicine and health." [14]

One statistical test that may be performed with SAS is the Proc Ttest. This statistical test can be used to compare if there is a statistically significant difference between two compared groups. [15]

Another statistical test that can be performed is proc REG for regression analysis. This test involves a dependant variable that can be predicted based on one ore more independent variables. For example, regression analysis could be used to predict blood pressure based on dosage of a new drug, cholesterol level based on patient age, or degree of wound healing based on the surface area of burn wound. [16]

Another possible study would be to consider the effectiveness of different kinds of stents used in aortic aneurism surgery. For example, in an endovascular aneurism surgery, stents are moved through the arteries via incisions near the groin to the aorta. A test could be used in a clinical study of the effectiveness of different versions of such stents, comparing factors such as diameter of aneurism, likelihood of rupturing, or symptoms such as back pain or a pulsating feeling near the navel. [17]

Another possible clinical study might evaluate the effectiveness of different models of artificial knees used in knee replacement surgeries, investigating outcome results such as range of motion, pain, and mobility levels, both before and after the procedure, and comparing one artificial knee to another.

7. Training

Several training options are available to learn SAS pharmaceutical programming. Philadelphia University and SAS Institute Inc. are collaborating to provide such training. The program covers: SAS® Fundamentals, SAS® Report Writing and ODS, SAS® Programming, SAS® Macro Language and SAS® SQL Processing. It also covers an overview of the pharmaceutical industry, data issues, study design, data analysis and biostatistics. [18]

Another similar training program is offered by UC San Diego, which has online certificate programs in SAS and Biostatistcs. [19]

SAS Institute also offers its own Pharmaceutical Certificate. [20]

8. References

[1] SAS Clinical Programming in 18 Easy Steps by Y. Prasad, p. 1. While the FDA regulates drugs in the U.S., several international equivalent organizations regulate drugs overseas, such as the European Medicines Agency, the Chinese State Food and Drug Administration, and the Indian Ministry of Health and Family Welfare's Drugs Controller General of India. www.fda.gov/Food/GuidanceRegulation/FSMA/ucm291803.htm. While the FDA engages internationally, a key issue in many regions of the world is how well regulatory policies are effectively enforced in practice.

[2] Prasad, Ibid. at p. 15.

[3] Ibid. at p. 16.

[4] Ibid. at p. 248.

[5] Ibid. at p. 222.

[6] Ibid. at p. 225.

[7] Ibid. at p. 376

[8] Sharpening Your SAS® Skills by S. Gupta at p. 155.

[9] SAS Clinical Programming in 18 Easy Steps by Y. Prasad at p. 397.

[10] Ibid. at p. 343. The podiatry example in this poster is based on a modified version of the weight change example in SAS Clinical Programming in 18 Easy Steps. The data is hypothetical.

[11] http://www.sas.com/en_us/industry/higher-education/on-demand-for-academics.html

[12] SAS Clinical Programming in 18 Easy Steps by Y. Prasad at p. 301.

[13] Ibid. at 264, 267.

[14] https://en.wikipedia.org/wiki/Biostatistics

[15] SAS Clinical Programming in 18 Easy Steps by Y. Prasad at p. 199-200.

[16] Ibid. at 213.

[17] http://www.webmd.com/heart-disease/endovascular-repair-for-abdominal-aortic-aneurysm; www.mayoclinic.org.

[18] http://www.philau.edu/scps/CPE/Certificates/SAS.html

[19] http://extension.ucsd.edu/programs/index.cfm?vAction=certDetail&vCertificateID=201&vStudyAreaID=4

[20] http://www.philau.edu/scps/CPE/Certificates/SAS.html

[21] http://www.fda.gov/Drugs/ResourcesForYou/Consumers/PrescriptionDrugAdvertising/ucm071964.htm The FDA regulates drug advertising as well as the drug approval process.

Implementation of Virtual Computing Environments for Online Computer Science Instruction

John P. Buerck[1], **Srikanth P. Mudigonda**[1], **Matthew J. Grawitch**[1] **and Susana V. Salinas**[1]
[1]School for Professional Studies, Saint Louis University, St. Louis, Missouri, USA

Abstract *– Online courses are an essential component in University education. To support online instruction, infrastructure is needed to host virtual computing environments that meet the instructional needs of computing disciplines. This poster outlines how Saint Louis University designed, developed and implemented a private cloud-based virtual computing infrastructure to support computer science instruction.*

Keywords: Virtual Computing Environments, Online Education

1 Poster Overview

The focus of this poster is to share our experiences of designing, developing, implementing and using a locally-hosted private cloud virtual computing environment for computer science instruction within a university domain. Additionally, we will provide detailed information relating to both the technical aspects of our design and implementation processes, and the design of learning and teaching activities that utilize the infrastructure provided by the environment.

The poster will be accompanied by a demo/walk-through of the virtual computer lab environment from the perspective of a student and a course instructor. This demo will show a sample set of learning, teaching, and evaluative activities that are facilitated through the environment. Conference attendees will have the opportunity to seek detailed responses on specific questions related to various aspects of the design and implementation of our environment, along with integrating it into our curriculum.

2 Key Poster Outcomes

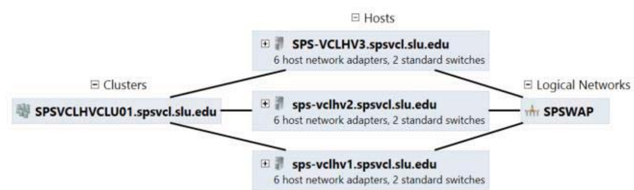
- Identify hardware and software infrastructure supporting scalability of virtual computing.
- Understand budgeting, licensing and security for virtual computing.
- Explain private cloud virtual computing from the administrator, instructor, and student roles.

3 Poster Relevance

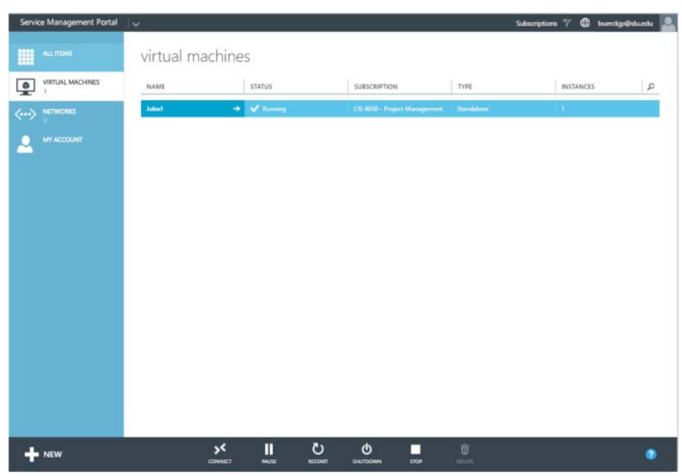
University leaders, administrators, instructors, and information technology professionals looking to use virtual computing for strategic advantage within their institution will benefit by learning about opportunities, challenges, and key milestones through the use of firsthand accounts. Session participants can learn from our experiences and determine how a private cloud-based virtual computing environment can be implemented within their academic institution using a blueprint that is either based on or similar to ours.

4 Selected Screen Captures of Demo

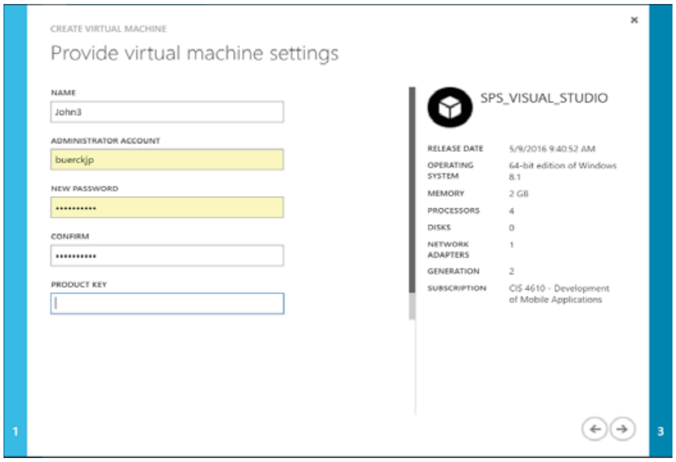
4.1 Server & Network Configuration



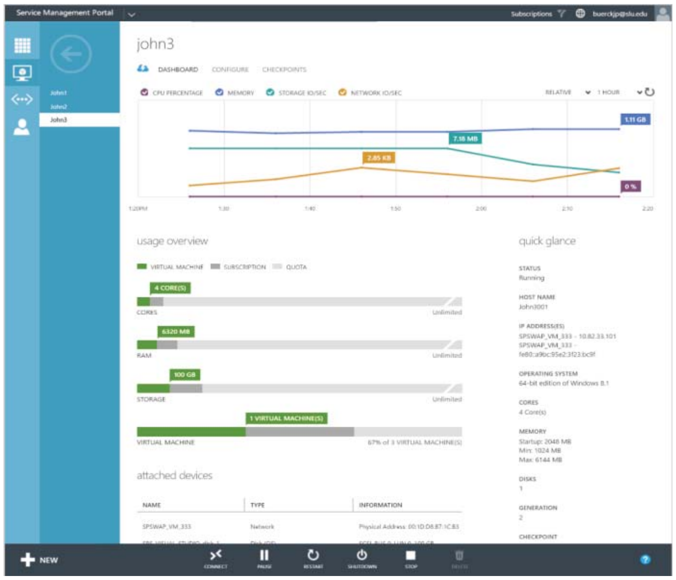
4.2 Server Management Portal



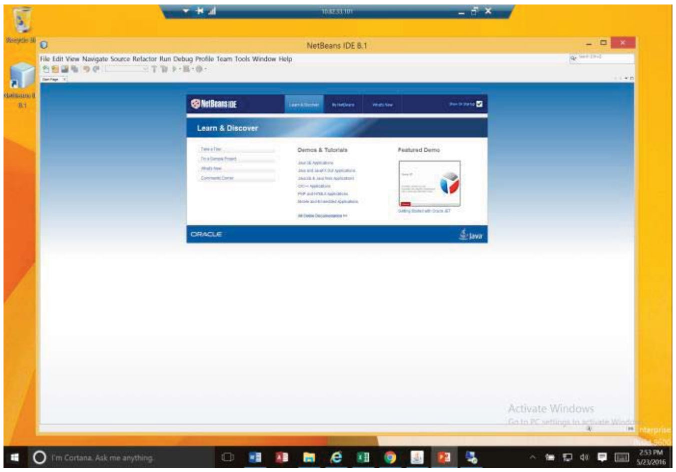
4.3 Virtual Machine Setup – Student View



4.4 Server Management Dashboard



4.5 Virtual Desktop Running Net Beans IDE



5 Conclusion & Future Enhancements

The Azure virtual computing environment went into production in the spring of 2016. Three online programming courses used the virtual desktop as it primary lab/development environment. Positive feedback was received from both students and faculty. Future enhancement to the virtual computing environment will include 1) increased server capacity, 2) increased storage and 3) the use of Parallels Remote Application Server.

Standardized, Rubrics-Based Design and Delivery of Online Courses: Lessons Learned

Srikanth P. Mudigonda[1], **John P. Buerck**[1] **and Matthew J. Grawitch**[1]
[1]School for Professional Studies, Saint Louis University, St. Louis, Missouri, USA

Abstract – While moving from a ground-based course delivery mode to an entirely online mode, we developed a nuanced understanding what it takes to implement a standardized, rubrics-based approach for designing and delivering our courses. In our poster session, we present the lessons we learned during the process, along with the challenges we encountered.

Keywords: Rubrics Development, Online Education, Curriculum

1 Poster Overview

The focus of this poster is to highlight the key issues that need to be recognized and addressed carefully when moving from a course-delivery modality that is entirely on ground to one that is entirely online. Advances in instructional and learning technologies, on the one hand, with the changing profiles and expectations of learners, on the other hand, have created new opportunities and challenges for educators.

The experiences that we share in our poster session will be of interest and significance to academic administrators such as department chairpersons, directors, and coordinators, who have recognized the role of technology in the design and delivery of courses in hybrid/flipped/blended and online formats. Specifically, we identify the key set of learning and evaluative activities that are common across courses in the curriculum, such as creation and delivery of lectures, interactive question-and-answer and problem-solving sessions, small-group activities and discussions, monitoring of student participation and contribution to group activities, and design and delivery of evaluative components.

The poster will describe the challenges we faced and the lessons we learned when designing and implementing our courses using a standardized, rubrics-based approach. Where possible, recommendations for other academic-practitioners involved in delivery of online curricula will be provided.

2 Key Poster Outcomes

- Identify key learning and evaluative components of an online course.
- Understand challenges and opportunities afforded by relevant instructional and learning technologies that are designed to provide detailed feedback to students in a systematic ways
- Match technologies and course activities to meet course objectives.

3 Poster Relevance

The poster will provide a “big picture” view of the online course design and delivery process and how various technologies can be leveraged to facilitate the process. Additionally, we will provide the opportunity to view and interact with a “sample” version of a course that we have designed and implemented in our learning management system. Attendees can navigate through our course website and learn how different learning and evaluation activities have been implemented and facilitated through the website.

An Application of Reflective Learning Journal with Goal -based Scenario for Course Learning Outcomes

YoungTaek Jin , TaeWon Lee[1]

[1] Dept. of Computer Engineering, Hanbat University, Daejeon, South Korea

***Abstract* -** A number of studies have shed light on the effectiveness of a reflective learning journal. Previous research has shown that it can enhance course learning outcomes, increase problem-solving skills and motivate the adoption of self-controlled learning strategies. However, it has been also argued that many issues such as providing adequate learning experiences and feedbacks needs to be tackled. This paper attempts to solve these issues using a goal-based scenario. Based on our experimental study, we suggest that incorporating goal-based scenarios into reflective learning journal may offer a solution to some of the problems associated with it.

Keywords: Reflective learning journal, Goal-based scenario, Course learning outcomes

1 Introduction

Outcome based education focuses on what students have rather than what teachers teach them[1].Thus, the educational paradigm requires continuous improvement in teaching methods and educational environment that can help students improve their outcomes. Since course level outcomes contribute to student's outcomes, many methods for enhancing these course outcomes have been studied including classroom assessment techniques and problem - based learning [2,3,4].

At first, we needed a method to investigate which teaching methods and student's learning strategies are effective to enhance course outcomes. Reflective learning journal[5] is a journal which allows students to record their thoughts and insights about their learning experience. It encourages students not only to review and consolidate their knowledge and skills, but also to plan future learning based on past learning experiences. Among many techniques to enhance course outcomes, it was selected because of its simplicity and positive results from many studies[6]. However, while applying this technique in six courses over three semesters, we have encountered several problems.

In this paper, we investigate a new way to overcome such problems. Specifically, we used a goal-based scenario in order to solve the issues from applying the reflective learning journal.

2 Our experiences from reflective learning journals

Based on a well-established basic format adapted from the existing literature[7], Reflective learning journal in the study consists of various questions about learning contents, learning strategies and learning evaluation. For example, one of the questions about the learning evaluation viewpoint is: "How can I relate the knowledge and skills that I have learnt in the topics to another situation/context?".

In addition to the basic questions, we also used additional one or two more questions to acquire the teaching and learning data that we need. We have applied it in five different courses - 'C programming', 'Digital signal processing',' Information system analysis and design', 'Digital System Design', and 'Software Engineering' course. These courses are different in lecture types(theory or theory with lab.), grades of the students(freshman-senior), number of students(10-56) and submission period(3-5 times).

For the three courses('C programming', 'Digital signal processing',' Information system analysis and design') in the first semester, we used the basic questions of reflective learning journals to obtain the overall perception of their learning. At the beginning of the semester, we provided adequate guidance so that students could readily write the journals. Students were asked to submit their journals before, during and after class.

After analyzing the first application, we focused more on their learning strategies and learning evaluation in the next three courses ('C programming', 'Digital System Design', 'Software Engineering') in the second semester, Through this experimental study, we have obtained the following analysis results:

1) Providing realistic learning experiences that match course learning outcomes is essential. That is, it is important to provide students an opportunity to integrate what they have learned in a class and critically view such knowledge.

2) Feedbacks must be provided to each student. However, it requires time and teacher's extra efforts

3) We had assumed that the student him/herself will acknowledge their own learning methods; however, students were unaware or in some cases ignorant to specific learning strategies.

To overcome these problems, we propose integrating goal-based scenario associated into reflective learning journals.

3 Goal-based scenario

The goal-based scenario [8] is one of the problem-based learning approaches such as Case Study, Problem Based Learning and Action Learning. It provides specific hypothetical situations or examples that students may face in reality, and thus helps students relate to those situations. Based on the scenarios, students can clearly understand the context/situation in which particular knowledge is used and learn to apply the earned knowledge to solving specific problems. We selected the goal-based scenario among many other problem-based learning approaches because it is a well-structured method. Compared to existing learning methods, it provides students with an authentic and highly-organized scenarios to which students can relate. In our study, goal-based scenarios consist of learning goal, mission, a cover story, a role and resource[9]. The scenario structure is depicted in Figure 1. According to course learning objectives, scenarios are provided for students to understand concepts and practice skills. They are asked to submit not only the reports and assignments, but also reflective journals.

To find out the effectiveness of our proposal, we authored four scenarios (Use case writing scenario, UML analysis and design modelling scenario, testing scenario and configuration management scenario) according to course objectives in software engineering course of the third semester. For example, we have provided various practical situations through scenario to teach configuration management concepts and principles.

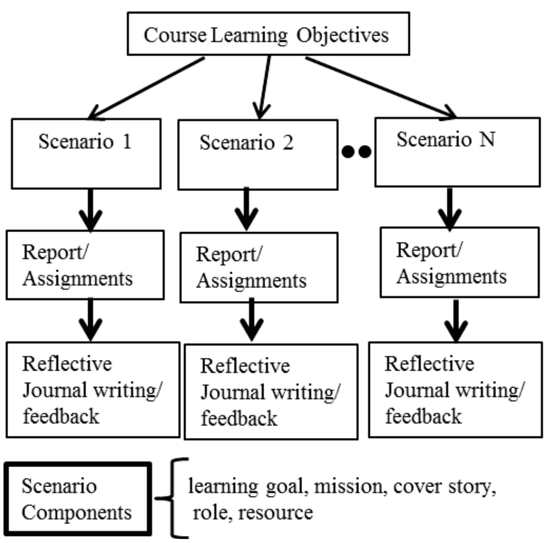


Figure 1: Scenario structure with reflective learning journal

4 Lesson learned and Conclusion

Compared to journal writing without real or simulated situation, the contents of journal are more clear and specific enough to evaluate course outcome and derive the improvements of teaching methods. Also, the assessment results of assignments and reports indicated that course outcomes have been increased compared with the previous results of same course.

In addition to this direct measure, we measured the effectiveness of using goal-based scenarios in conjunction with reflective learning journals using surveys and focus group interviews.

1) Provision of learning experiences:
Strongly Agree:33%, Agree:54%, Neutral:13%
2) Positive effect about learning attitude:
Strongly Agree:23%, Agree:50%, Neutral:27%
3) Help for improving learning strategies:
Strongly Agree:23%, Agree:40%, Neutral:33%, Disagree:4%

In this paper, we have introduced a goal-based scenario technique with reflective learning journals.

Through the experimental study, we have obtained positive results of our approach in order to review students' learning processes and investigate which teaching methods and student's learning strategies are effective to enhance course learning outcomes. This is because goal-based scenarios and feedbacks have provided a realistic situation to which students can relate.

While our research has revealed the benefits of incorporating goal-based scenario with reflective learning journals, there are some potential limitations that should be addressed such as scenario authoring and provision of scenario library.

5 References

[1] Joni E. Spurlin et al, "Assessing student learning-Ensuring Undergraduate students are learning what we want them to learn", edited by joni E. Spurlin, Sarah A. Rajala, and Jerome P.Lavelle, p.23-58,Stylus publishing, 2008.

[2] Angelo et al, "Classroom Assessment Techniques: A Handbook for College Teachers", 2nd ed. 1993.

[3] Margaret C. Lohman, "Cultivating Problem-Solving Skills Through Problem-Based Approaches to Professional Development", Human Resource Development Quarterly Volume 13, Issue 3, 2002.

[4] Janet L. Kolodner and Mark Guzdial, "Theory and Practice of Case-Based Learning Aids", Theoretical Foundations of Learning Environments, Edited by David Jonassen,Susan Land, 2000.

[5] Susan E. George, "Learning and Reflective Journal in Computer Science", Univ. of South Australia, Twenty-Fifth Australasian Computer Science Conference (ACSC2002), 2001.

[6] A. Srikantha Phani, "Reflective learning in a large core course in mechanical engineering", University of British Columbia, 2012.

[7] Reflective Learning Journal Teacher Guide, https://www.polyu.edu.hk/.../Reflective%20Learning%20Journal%20.

[8] Roger C. Schank et al, "The Design of Goal-Based Scenarios". Journal of the Learning Sciences, 3(4), 1993.

[9] Chung-Yuan Hsu, "Formative research on the goal-based scenario model applied to computer", The Journal of Applied Instructional Design · Volume 1 · Issue 1, 2011.

A Pedagogy-Supported English-Medium Instruction Course

Yung-Ting Chuang
Department of Information Management, National Chung Cheng University, Chia-Yi County, Taiwan

ABSTRACT
Education is one of the most important elements in our lives, as it provides a direct gain in knowledge. Thus, in order to provide better academic achievement, more and more instructors adapt collaborative learning to their classes. Moreover, there seems to be an increasing numbers of EMI (English as Medium of Instructions) courses added to university course offerings in countries where English is not the first language, for supporting university internalization and addressing the global status of English. However, some scholars have indicated that the EMI courses might decrease the overall learning due to the poor lecture comprehension and passive in-class engagement. Thus, in order to encourage more collaborative learning and obtain better lecture comprehension, the author introduces an EMI pedagogy in this paper. The author first describes how she applied her pedagogy to her programming courses to increase students' overall learning in her EMI courses. Then, the author plans to ask students to evaluate their experiences and feedback, and demonstrates the effectiveness of this EMI pedagogy to improve the overall lecture comprehension, encourages more class engagement, and promotes collaborative learning.

Keywords
English as medium of instruction, pedagogy.

1. SYSTEM DESIGN
Education is one of the most important elements in our lives, as is provides a direct gain in knowledge. Thus, in order to provide better academic achievement, more and more instructors adapt collaborative learning to their classes. Collaborative learning, by definition, means that students have to become socially active and achieve learning goals in a group-based environment. According to [1][2][3][4][5], the benefits of collaborative learning include learning concepts more effectively, encouraging more participation, improving learning satisfaction, developing teamwork skills, and promoting higher-order thinking. Moreover, recently there seems to be an increasing numbers of EMI (English as Medium of Instructions) courses added to university course offerings in countries where English is not the first language for supporting university internalization and addressing the global status of English [6][7]. According to [9], the number of EMI courses offered from all universities at Taiwan has risen from 2,013 to 4,099 within 5 years. However, some studies like [8] argue that EMI courses might affect the overall learning of course content, such as affecting students' poor lecture comprehension and passive engagement in class.

In order to resolve students' poor lecture comprehension and encourage more in-class engagement in EMI courses, the author introduces her EMI pedagogy in this paper. The author first describes how she designed and implemented her EMI pedagogy to better achieve lecture comprehension and greater in-class engagement in her programming courses. Next, the author plans to ask students to evaluate their experiences and provide feedback about this pedagogy. The goal of this paper is that students can therefore increase their learning motivation, and thus can ultimately achieve greater learning outcomes.

2. COURSES DESIGN AND PEDAGOGY
There are four parts in the pedagogy, which are: 1) Lecture and in-class discussions, 2) In-class laboratories, 3) Programming assignments, 4) Group projects and presentation. The instructor and teaching assistant are the moderator for the in-class laboratories and discussions, providing necessary help for the students.

First, the instructor first starts the lecture by introducing the important programming concepts, and then have in-class discussions that require students to work on some theoretical-related problems with their group partners, using either English or Chinese, and finally explain their answers after they have reached to the agreed answers. Next, the instructor assigns some in-class laboratories and requires students to work on programming-related problems with one partner, whereas the purpose of these in-class laboratories is that students with insufficient English proficiency can catch up on lecture materials with their peers, and gain more accomplishments. For these in-class laboratories, both instructor and teaching assistants are the moderator that provide necessary help for the students. By having these in-class laboratories once per week, it could further increase students' lecture comprehension and collaborative learning. Then after the students have obtained the course materials from lectures, in-class discussions, and in-class laboratories, they would need to accomplish their programming assignments alone at home. Lastly, the students would need to work on their programming projects with their partners, and finally present their project ideas to the whole class at the last lecture.

3. VALIDATION
So far the author has only applied this EMI pedagogy to her programming courses, but not yet evaluate students' feedback of this EMI pedagogy. Thus, the author will ask students to evaluate about their experiences, and demonstrate the effectiveness of her system about the collaborative learning, lecture comprehension, class engagement, and learning outcomes.

4. ACKNOWLEDGEMENT
This research was supported in part by MOST 104-2410-H-194-090-MY2 of the Ministry of Science and Technology and National Chung Cheng University Teaching Excellence Project by the Ministry of Education, Taiwan.

5. REFERENCES
[1] Felder, R. M., & Soloman, B. A. (1988). Systems thinking: An experimental course for college freshmen. Innovative Higher Education, 12(2), 57-68.

[2] Felder, R. M., & Brent, R. (2009). Active learning: An introduction. ASQ Higher Education Brief, 2(4), 1-5.

[3] Oakley, B. A., Hanna, D. M., Kuzmyn, Z., & Felder, R. M. (2007). Best practices involving teamwork in the classroom: Results from a survey of 6435 engineering student respondents. Education, IEEE Transactions on, 50(3), 266-272.

[4] Haller, C. R., Gallagher, V. J., Weldon, T. L., & Felder, R. M. (2000). Dynamics of peer education in cooperative learning workgroups. Journal of Engineering Education, 89(3), 285-293.

[5] Johnson, D.W. 1991. Cooperative Learning: Increasing College Faculty Instructional Productivity. ASHE-ERIC Higher Education Report No. 4, 1991. ASHEERIC Higher Education Reports, George Washington University.

[6] Ferguson, G. (2007). The global spread of English, scientific communication and ESP: Questions of equity, access and domain loss. Iberica: Revista de la Asociacion Europea de Lenguas para Fines Especificos (AELFE)(13), 7-38.

[7] Crystal, D. (2012). English as a global language: Cambridge University Press.

[8] Chang, Y. Y. (2010). English-medium instruction for subject courses in tertiary education: Reactions from Taiwanese undergraduate students. Taiwan International ESP Journal, 2(1), 55-84

[9] Lin, S. Y. (2010). 5-year 50 billion program reaping fruits: 103 percent increase for English-medium courses. Epoch Times. Retrieved from http://www.epochtimes.com/b5/10/4/1/n2864074.htm

SESSION

LATE BREAKING PAPERS

Chair(s)

TBA

ISBN: 1-60132-435-9, CSREA Press ©

In the Beginning: Establishing a Baseline of Awareness, Exposure, and Motivation for Students in a Nanoscience and Nanotechnology Course

Deborah Worley[1], Naima Kaabouch[2], Matt Cavalli[3], Kanishka Marasinghe[4], Nuri Oncel[4], David Pierce[5], Brian Tande[6], Julia Zhao[5]
[1]Educational Leadership, [2]Electrical Engineering, [3]Mechanical Engineering, [4]Physics and Astrophysics, [5]Chemistry, [6]Chemical Engineering,
University of North Dakota, Grand Forks, North Dakota, USA

***Abstract* -** This paper describes the preliminary results of a study about students' awareness of and exposure to nanotechnology. The study also explores students' motivations for investigating nanotechnology through interdisciplinary coursework. To collect this information, the *Nanotechnology Reflection Survey* was administered at a research university in the upper Midwest to students who enrolled in a nanoscience and nanotechnology (NSNT) course. The course was team taught by faculty across various disciplines including Electrical Engineering, Mechanical Engineering, Chemical Engineering, Chemistry, and Physics. The data collected from the survey will be used to inform the design and delivery of future nanoscience and nanotechnology (NSNT) courses as part of constructing a complete nanoscience and nanotechnology (NSNT) curriculum. The overall study is funded by a National Science Foundation (NSF) Nanotechnology Undergraduate Education (NUE) grant.

Keywords: nanoscience, nanotechnology, engineering student awareness, exposure, motivation

1 Introduction

Nanoscience and nanotechnology represent a growing sector of academic and industry [1-4]. Advancements in our global society push us even further to acquire the next generation of technologies for utilizing nanoscience in various engineering disciplines. A society that produces more and better prepared STEM graduates will provide the talent needed to ensure future technological leadership, and the jobs that go along with that success. To address this need, many institutions are offering more courses and opportunities for students to pursue nanoscience and nanotechnology as an academic discipline and, ultimately, as a career field. For institutions just beginning this journey, it is important to establish a baseline of knowledge about the students who express interest in nanoscience and nanotechnology.

The purpose of the project is to learn more about students' level of awareness about nanoscience and nanotechnology, and to understand what kind of exposure they have had to nanotechnology prior to enrolling in a course on the subject matter. In addition, we seek to understand students' motivations for investigating nanotechnology. This information was collected as part of the *Nanotechnology Reflection Survey* [5]. This instrument is part of a larger project that is dedicated to introducing the nanoscience and nanotechnology field into undergraduate engineering curricula at a public state institution in the upper Midwest [6]. With this overarching purpose in mind, the project team members are developing and implementing new NSNT courses aimed at growing students' interest and responding to the needs of training future workforce in the field. The overall study is funded by a National Science Foundation (NSF) Division of Engineering Education and Centers (EEC) Nanotechnology Undergraduate Education (NUE) grant.

2 Methodology

Students enrolled in a nanoscience and nanotechnology course in fall 2015 were asked to complete the *Nanotechnology Reflection Survey*. The internal evaluator visited the class twice to distribute statements of informed consent and to distribute and collect the survey instrument. The first visit took place within the first four weeks of the course. The second visit took place in the last week of the course. The same instrument was used in both visits. Participation was voluntary; no compensation was offered to students for completing the survey. Eighteen students completed the survey.

The *Nanotechnology Reflection Survey* [5] was designed by Diefes-Dux and colleagues to be used as a measure of variables related to awareness, exposure, and motivation specifically designed an instrument to measure relevant aspects of student awareness of, exposure to and motivation for students to study nanotechnology [7]. There are eight questions about students' awareness of nanotechnology, six questions about students' exposure to nanotechnology, and 17 questions about students' motivations for studying, or investigating, nanotechnology. The instrument also includes a series of questions that are used to measure students' knowledge levels about nanotechnology, and their use of simulation and other pedagogical tools to learn about nanotechnology – these items will be discussed in a future paper.

3 Results

In this paper, we report basic descriptive statistics for the questions about students' awareness of, exposure to, and motivations for investigating nanotechnology.

3.1 Awareness of Nanotechnology

Students' awareness of nanotechnology was measured using eight items from the *Nanotechnology Reflection Survey*. All items used a five point scale, ranging from (1) strongly disagree to (5) strongly agree. The results for these items are included in Table 1.

Table 1.
Students' Awareness of Nanotechnology

I can:	*Mean* (start of course)	*SD* (start of course)	*Mean* (end of course)	*SD* (end of course)
Name a nanoscale-sized object.	4.78	0.43	4.80	0.41
Describe one way nanotechnology directly impacts my life.	4.06	0.94	4.67	0.49
Name a field of study that currently conducts nanotechnology research.	4.61	0.50	4.80	0.41
Describe one way nanotechnology may benefit society/humankind	4.78	0.43	4.80	0.41
Name an application of nanotechnology.	4.78	0.43	4.80	0.41
Describe a process to manufacture objects at the nanoscale.	3.44	1.25	4.60	0.51
Name an instrument used to make measurements at the nanoscale.	4.56	0.78	4.67	0.49
Describe one way nanotechnology may directly impact my life in the future.	4.67	0.59	4.80	0.41

3.2 Exposure to Nanotechnology

Students' exposure to nanotechnology was measured using six items from the *Nanotechnology Reflection Survey*. All items used a five point scale, ranging from (1) not at all/never to (5) a great deal. The results for these items are included in Table 2.

Table 2.
Students' Exposure to Nanotechnology

I have:	*Mean* (start of course)	*SD* (start of course)	*Mean* (end of course)	*SD* (end of course)
Heard the term nanotechnology.	4.28	0.83	4.73	0.46
Read [something] about nanotechnology.	3.89	0.96	4.60	0.63
Watched a program about nanotechnology.	3.11	1.23	3.73	1.22
Had one [or more] instructors/ teachers talk about nanotechnology in class.	3.67	1.19	4.13	0.92
Participated in an activity involving nanotechnology [lab, project, research].	2.89	1.53	4.27	1.16
Taken a class about nanotechnology.	3.22	1.44	4.13	1.13

3.3 Motivations for Investigating Nanotechnology

Students' motivations for investigating nanotechnology were measured using 17 items from the *Nanotechnology Reflection Survey*. All items used a five point scale, ranging from (1) strongly disagree to (5) strongly agree. The results for these items are included in Table 3.

Table 3.
Students' Motivation to Investigate Nanotechnology

I plan to:	*Mean* (start of course)	*SD* (start of course)	*Mean* (end of course)	*SD* (end of course)
Read a fiction story about nanotechnology.	2.44	0.98	2.64	1.08
Formally teach nanotechnology concepts (e.g. as a teaching assistant)	2.56	0.98	3.21	0.80
Investigate the implications of nanotechnology.	4.17	0.62	4.00	0.96
Informally/casually teach someone something about nanotechnology.	4.11	0.68	4.00	0.78
Seek information about internships or Co-op experiences with companies engaged in nanotechnology.	3.94	0.73	4.50	0.65
Read a news story or popular magazine article about nanotechnology.	4.06	0.94	4.21	0.58
Give a presentation related to nanotechnology to an audience I perceived as having more experience with nanotechnology than I.	3.11	1.08	3.21	0.89
Explore nanotechnology as part of this course.	4.61	0.50	4.43	0.76
Read a research journal article about nanotechnology.	4.06	0.80	4.00	0.96
Enroll in a course about nanotechnology.	4.50	0.51	4.50	0.65
Attend a non-course related seminar about nanotechnology.	3.65	0.79	3.57	1.34
Visit an industry or business that specializes in nanotechnology.	3.94	0.73	4.14	0.66
Give a presentation related to nanotechnology to an audience I perceived as having less experience with nanotechnology than I.	3.28	1.07	3.57	1.02
Watch a program about nanotechnology.	3.94	0.64	4.00	1.11
Apply or interview for a nanotechnology related work or research experience.	4.00	0.69	3.93	1.21
Investigate fields of study in which I can learn more about nanotechnology.	4.17	0.62	4.43	0.65
Obtain a work experience or undergraduate research opportunity related to nanotechnology.	3.89	0.96	4.07	0.73

4 Discussion

The highest means for ways in which students reported being aware of nanotechnology belonged to two items: (1) Name a field of study that currently conducts nanotechnology research and (2) Describe one way nanotechnology may benefit society/humankind. The lowest reported mean was related to describing a process to manufacture objects at the nanoscale. These results indicate that students may be theoretically attuned to nanotechnology but they may not yet be aware of the breadth of application or ways in nanotechnology can be used in industry. On a related note, students reported being exposed to nanotechnology mostly through formal classroom learning. For example, more students had read about nanotechnology by the end of the course when compared to the students who reported having done so toward the beginning of the course. Largely, it does not appear that students actively report out of classroom opportunities as ways to engage with and become exposed to nanotechnology, or at least they do not do so at a level that is equal with in-class learning. Students' motivations for investigating and studying nanotechnology range quite a bit. The highest means for motivations were related to the nanoscience and nanotechnology course in which they were enrolled. The lowest means belonged to voluntary and informal learning opportunities, such as reading a fiction story about nanotechnology. These results are not surprising, however, given the themes that were observed in the awareness and exposure responses.

5 Conclusion

There is congruence in how students consider the sources of their awareness of, exposure to, and motivations for investigating nanotechnology. They rely quite a bit on learning in formal settings, although there is a need to integrate real-world scenarios and examples of application into the classroom to the greatest extent possible. The development of nanoscience and nanotechnology as a curriculum is one of the main goals of the institution where this study is being conducted. The nanoscience and nanotechnology courses that are developed through this larger project will rely on information gleaned through the *Nanotechnology Reflection Survey* to structure and organize course content. Moreover, the results of the survey will help us understand how students experience the early iterations of the nanoscience and nanotechnology courses and will provide valuable insight into how these courses can be improved for the future to better educate tomorrow's technological leaders.

6 References

[1] A. Delemarle, B. Kahane, L. Villard, P. Laredo, "Geography of knowledge production in nanotechnologies: A flat world with many hills and mountains," *Nanotechnology Law and Business, 6*(1), 2009, pp. 103-122.

[2] F. Müller, M. Guggisberg, H. Burkhart, T. Gyalog, "Nano-World: A showcase suite for technology-enhanced learning," IEEE Education Engineering Conference, *EDUCON 2010*, art. no. 5492456, 2010, pp. 1075-1080.

[3] K.A. Singh, "European nanotechnology masters recognition scheme - supporting technical higher education," *Materials Research Society Symposium Proceedings, 1105*, 2008, pp. 8-17.

[4] S. Jeschke, N. Natho, O. Pfeiffer, C. Thomsen, "Networking resources for research and scientific education in nanoscience and nanotechnologies," *Proceedings of the 2008 International Conference on Nanoscience and Nanotechnology, ICONN 2008*, art. no. 4639290 , 2008, pp. 234-237.

[5] H.A. Diefes-Dux (2014), "Nanotechnology Reflection Survey," Retrieved from https://nanohub.org/resources/20771

[6] N. Kaabouch, D. Worley, M. Cavalli, B. Tande, J. Zhao, D. Pierce, G. Marasinghe, Nuri Oncel, "No Small Feat: Building A Nanoscience and Nanotechnology Program for Engineering Undergraduate Students," ASEE Conference, New Orleans, LA (June 2016).

[7] H.A. Diefes-Dux, M. Dyehouse, D. Bennett, P.K. Imbrie, "Nanotechnology awareness of first-year food and agriculture students following a brief exposure," *Journal of Natural Resources & Life Sciences Education, 36(1),* 2007, pp. 58–65.

A Concept Map of Concurrency Control in Database Management Systems

Jamal Alsabbagh
Department of Computing and Information Systems
Grand Valley State University, Allendale, Michigan, USA

Abstract – *Concurrency control is a core service offered by database management systems and is covered at various levels of detail in typical database courses. Teaching the topic, however, requires coverage of several concepts and technique. This paper is not intended to be a tutorial on concurrency control but rather it presents a one-page concept map that visually depicts how the underlying concepts and techniques relate to each other and the tradeoffs made in order to integrate a subset of the pieces to build the functionality of a concurrency control subsystem of a typical database management system. The purpose of the map is to provide students with an overall picture of the topic to help them navigate the various subtopics and organize their learning.*

Keywords: Concurrency Control. Teaching tools. Concept maps.

1 Introduction

Concurrency control is one of the fundamental services offered by database management system (DBMS) engines and plays a central role in facilitating the implementation of database applications that can support hundreds or even thousands of concurrent transactions per second and yet maintain the integrity of the database should one or more of the concurrent transactions fail. Being a core service, concurrency control is a standard module in database courses and textbooks such as [3, 5, 6, and 7]. Teaching and learning the module, however, presents a challenge mainly due to the fact that students need to learn several concepts, techniques, and the reasoning behind how the pieces fit together in order for a DBMS to offer the functionality of concurrency control. This paper is not intended to be a tutorial on concurrency control but rather it presents a one-page concept map that visually depicts how the underlying concepts and techniques relate to each other and the tradeoffs made in order to integrate a subset of the pieces to build the functionality of a concurrency control subsystem of a typical database management system.

One of the challenges that we face as teachers when covering a multi-lecture topic such as concurrency control is how to keep students motivated and engaged during coverage of the many prerequisite subtopics needed for the final integration into a whole. Furthermore, during coverage of the individual subtopics, students cannot be expected to mentally organize the subtopics and the relationships among them since they have not yet developed the framework to organize their knowledge. It is a well-known fact that a learner's motivation and ability to organize new knowledge have a significant positive impact on their ability to learn and apply that knowledge [1]. One visual technique that is widely used to facilitate knowledge organization is concept (or conceptual) maps that are frequently used in design and education. (A very good entry to the topic with a large number of articles and resources is [2].) This paper presents a concept map that we have developed and used several times to streamline the coverage of concurrency control in a database course.

Our course module on concurrency control consists of about seven one-hour lectures. The agenda for the very first lecture is to first motivate the need for concurrent execution of transactions since serial execution cannot feasibly support the high throughput and low response time expected in multiuser database applications. Second, we trace four examples of schedules (i. e. interleaved transactions) to demonstrate the anomalies of dirty read, lost update, unrepeatable read, and the phantom problem (shown in diagram 1 of figure 1) that certain schedules can cause. Third, we emphasize the fact that while the first three of the four anomalies are due to updates to values, the phantom problem is fundamentally different since it is due to insertion or deletion of objects in the database. Finally, we conclude the first lecture by introducing the ACID (atomicity, Consistency, Isolation, and Durability) properties of transactions.

The second lecture is devoted to explaining the structure of our one-page map of concurrency control that will be described in section 2 below. Section 3 gives brief comments on the remaining lectures of the course module.

Finally, Section 4 of this paper includes some concluding remarks.

2 The concurrency control concept map

As mentioned in the introduction, this paper is not intended to serve as a tutorial on concurrency control. In fact, the technical details mentioned in this section are presented only at the level necessary to explain the structure of the map to students and correspond very closely to the level presented in our second lecture of the course module. The goal is to use the map afterwards as a focal reference point when covering the details of the subtopics in subsequent lectures.

The map, depicted in figure 1, consists of six diagrams and six arrows. Each diagram is essentially an IS-A hierarchy among closely related concepts and/or techniques. On the other hand, the arrows in the map represent associations among the various components in the diagrams.

Having covered the anomalies depicted in diagram 1 in the first lecture, we now motivate students by pointing out that what we have covered and what we will cover about concurrency control is of immediate practical importance to a database developer. In particular, we point out that some applications can safely relax the isolation property of ACID in the quest for higher levels of concurrency. Towards that end, diagram 2 shows the isolation levels supported in SQL and arrow (A) shows that the programmer can explicitly choose an appropriate isolation level for a transaction depending upon what anomalies the transaction can tolerate based upon the database system functional requirements.

Diagram 3 shows a concurrency-based classification of schedules whereby a schedule can be serial (which is ACID) or interleaved (which may or may not be ACID). Furthermore, arrow (B) indicates that a serializable schedule is an interleaved schedule that is equivalent to at least one serial schedule and, therefore, is ACID. We point out at this point that the most common notion of equivalence used in DBMS is that of conflict equivalence giving rise to the name conflict-serializable schedules in diagram 3.

Diagram 4 shows the classification of schedules based upon recovery which is needed if and when a transaction in a schedule fails. The purpose of the classification here is to investigate the compromises that systems make since while a high level of concurrency is desirable it can increase the cost of recovery. Therefore, systems tend to allow only the schedules that require a recovery mechanism that uses as few system resources as possible for bookkeeping and for rolling back a failed transaction. The diagram shows four classes of schedules. One class is the non-recoverable schedules that, although admit high concurrency, are undesirable since they may require the rollback of an already committed transaction and thereby violating the durability property of ACID (a motivating example we give students here is whether it is acceptable for the bank to call you and ask you to 'un-spend' and return the $20 that you withdrew earlier from the ATM because the database had to roll back the withdrawal transaction after it had committed and you received you money?) The other three classes (recoverable, cascadeless, and strict) allow for varying levels of concurrency whereby recoverable schedules allow the highest degree of concurrency while strict schedules allow the least concurrency but have the advantage of requiring the least bookkeeping among the three – in fact it needs to log only the before image (BFIM) of a value prior to an update.

At this point, it is quite instructive to point out the relationship between the concurrency-based classification (diagram 3) and the recovery-based classification (diagram 4). A good Venn diagram showing the containment relationships among the various classes of schedules is figure 17.9 in [6] which we use to emphasize to students that one should not jump to a false conclusion at this point. More specifically, the concurrency-based classification is actually orthogonal to the recovery-based classification. For example, a schedule can be strict but not serializable and vice versa.

Diagram 5 shows a classification of concurrency control protocols. We emphasize here that since transactions do not communicate directly with each other, there is a need for a protocol (i.e. a set of rules) that every transaction must adhere to when requesting resources. In our course we cover the 2-phase locking (2-PL) protocol though we mention that there are others. As the diagram shows, there are four different types of 2-PL that essentially differ from each other in terms of the level of concurrency each one permits and its susceptibility to deadlocks. At this point we point out to students that we have now arrived at an important conclusion which is: *Strict 2-PL produces only strict schedules (arrow D) that are also conflict serializable (arrow C) since strict 2-PL is a subtype of 2-PL.*

The discussion on locking until now has left open the question of what specifically to lock and students may justifiably think that locking is applied at the field level. Arrow (E) threads into the discussion the fact that a database is a hierarchy of objects and consists of tables each of which consists of pages that in turn contain rows that consist of fields. Therefore, and as depicted in diagram 5, locking at the field, record, or even at the page level will

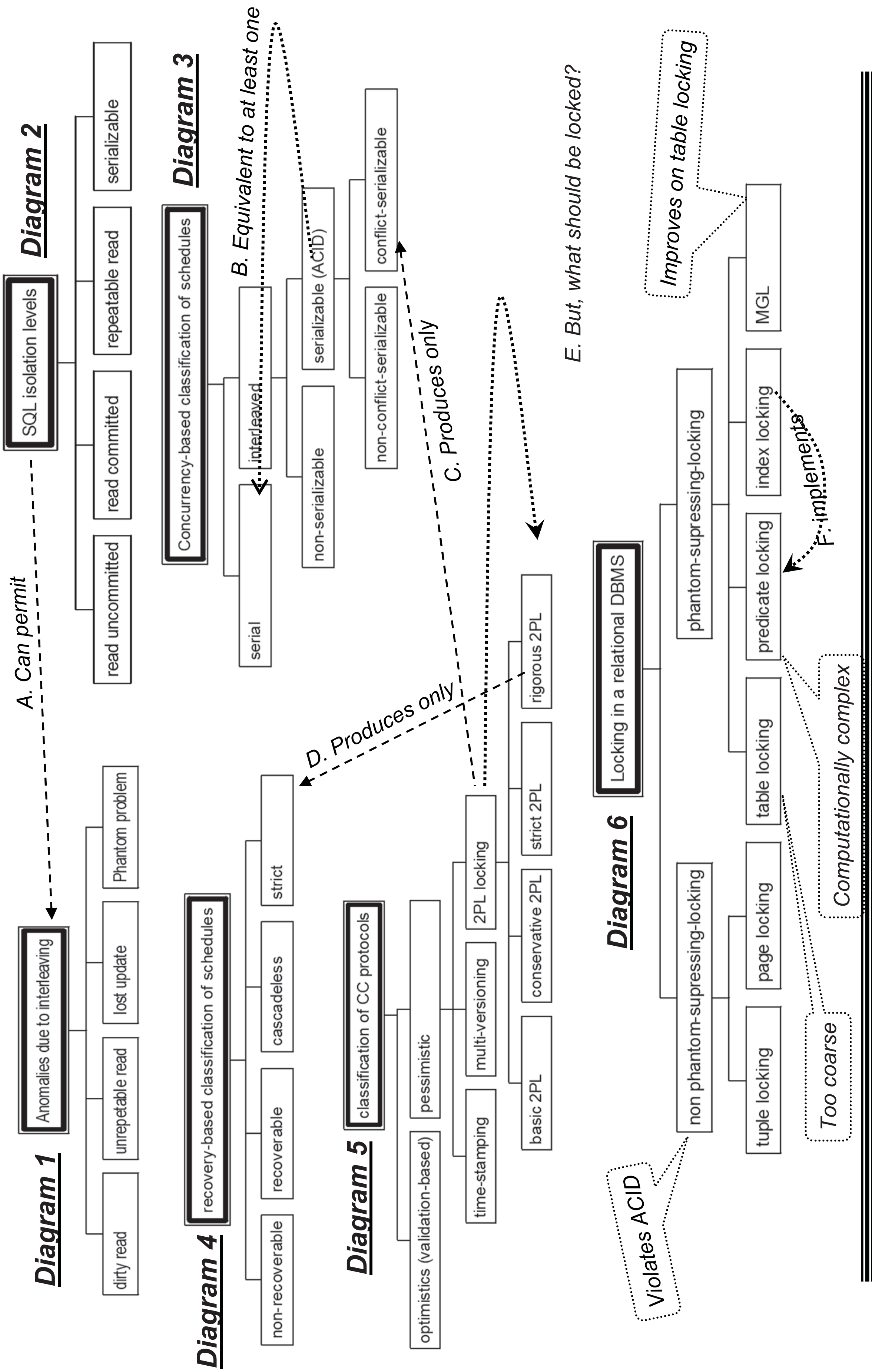


Figure 1: Concurrency Control Map

not solve the phantom problem which was shown earlier in diagram 1. For example, if a transaction T1 is scanning an employees table to find all employees whose age is between 25 and 30 but is still only half way during the scan when another transaction T2 inserts a new employee whose age is, say 26, in the part of the table that T1 has already scanned then T1 will produce an incorrect count.

As diagram 5 shows, one way to solve the phantom problem is to lock at the table level. However, such locking will be at too coarse of a level and will lower the level of concurrency in cases when different transactions in a schedule need to access disjoint parts of the table. Another solution to the phantom problem is predicate locking whereby the lock is associated with a logical condition rather than a physical object. For example, in the above scenario of T1 and T2, the lock will be associated with the predicate (age > = 25 and age <= 30) thereby prohibiting records satisfying the condition from being inserted by T2 while T1 has not completed its task. The problem with predicate locking is that in the presence of several concurrent transactions the predicates can become too complex to analyze efficiently. An alternative solution to the phantom problem is index locking (arrow F) whereby locks are applied to the lowest node of a B-tree index that covers the predicate and thus closing the paths down the tree on any new record whose condition matches the condition associated with the locked node.

Finally, diagram 6 depicts the technique of multiple granularities locking (MGL) which addresses the fact that a schedule often contains transactions that need different granularities of locking where some may need to lock a whole table while others may need to lock only a single row.

3 The remaining lectures in the course module

After the first two lectures mentioned in sections 1 and 2 above that serve as an overview of the landscape, the remaining five lectures in the course module on concurrency control are devoted to covering the details. We typically cover the topics shown in each one of diagram 3 through 6 in about one lecture per diagram. We regularly refer to the map while covering every subtopic in order to give context to the discussion. In covering the topics, we present several examples from commonly used database books such as the ones listed in the references section.

4 Concluding remarks

The map has been used several times in our database course. We found that the approach has been helpful in several ways. First, the map helps to motivate students very early on by showing them that the upcoming topics are important not only from an academic and theoretical perspective but also from a practical aspect since the programmer can declare the desired level of isolation among concurrent transactions. Second, the map is referenced throughout the module in order to streamline the discussion and give context to the subtopic(s) being discussed at the time by pointing out where the subtopic(s) appear on the map. Third, the map helps in imparting to students the fact that there is a certain elegance to the topic due to the fact that the practical solutions found in today's DBMSs are the result of integrating several concepts and techniques that address the underlying subproblems. Finally, for those who are interested in going beyond textbook discussions about the system aspects of DBMS including concurrency control and other topics, an authoritative comprehensive publication, [4], should be interesting.

5 References

1. Ambrose, Bridges, DiPietro, Lovett, and Norman, How Learning Works: 7 Research-Based Principles for Smart Teaching, Jossey-Bass, 2010.

2. Cmap Tools by IHMC Website http://cmap.ihmc.us (accessed in June 2016).

3. Elmasri and Navathe, Fundamentals of Database Systems, 7th edition, Addison-Wesley, 2016

4. Hellerstein, Stonebraker, and Hamilton, "Architecture of a Database System", in Foundations and Trends in Databases, J. M. Hellerstein, Ed., Foundations and Trends, Vol. 1, Hanover, MA: now publishers inc., 2007, pp. 141-259. Available at: http://db.cs.berkeley.edu/papers/fntdb07-architecture.pdf (accessed in June 2016).

5. Kifer, Bernstein, and Lewis, Database Systems: An Application Oriented Approach, Complete Version, 2nd edition, Pearson Publishing, 2006

6. Ramakrishnan and Gehrke, Database Management Systems, 3rd edition, McGraw-Hill, 2003.

7. Silberschatz, Korth, and Sudarshan, Database Systems Concepts, 6th edition, McGraw Hill, 2011.

Monotony and concavity intervals of non transcendental functions with *Mathematica*

R. Ipanaqué[1], R. Velezmoro[1] and N. Farias[1]
[1]Department of Mathematics, National University of Piura, Castilla, Piura, Perú

Abstract— *Finding the monotony and concavity intervals of a function is of paramount importance in the construction of functions graphs. This paper presents a new package,* `GraphsOfFunctions`*, developed in* Mathematica*, which incorporates a command that returns the monotony and concavity intervals of non-transcendental functions. The information is displayed in tables that are very similar to those obtained in a traditional way; in addition, the package includes a command which obtains the graphs of functions with the regions of increase and decrease indicated by different colors; and another command that gets a similar graph for the regions of concavity.*

Keywords: monotony, concavity, GraphsOfFunctions

1. Introduction

Making the construction of the graphs of the functions requires a lot of skill from students in a traditional calculus course at the undergraduate level [2], [3], especially while finding the monotony and concavity intervals. This paper presents `GraphsOfFunctions` –a new package developed in *Mathematica* [4]– which is intended as a tool for students to verify the results obtained in their calculations.

This tool is limited only to non-transcendental functions, i.e. those functions that do not involve trigonometric, trigonometric inverse, exponential and logarithmic functions. The package `GraphsOfFunctions` has been implemented in *Mathematica* v10.3.1, although later releases are also supported. The output obtained is consistent with *Mathematicas*'s notation.

The structure of this paper is as follows: Section 2 provides some mathematical background on monotony and concavity intervals. Then, Section 3 introduces the new *Mathematica* package which obtains them. The performance of the commands are discussed by means of using some illustrative examples. Finally, Section 4 closes ends the main conclusions of this paper and some further remarks.

2. Mathematical Preliminaries

2.1 Definitions of Increasing and Decreasing Functions

A function f is increasing on an interval if for any two numbers x_1 and x_2 in the interval, $x_1 < x_2$ implies $f(x_1) < f(x_2)$.

A funtion f is decreasing on an interval if for any two numbers x_1 and x_2 in the interval, $x_1 < x_2$ implies $f(x_1) > f(x_2)$.

2.2 Test for Increasing and Decreasing Functions

Let f be a function that is continuous on the closed interval $[a, b]$ and differentiable on the open interval (a, b).

1) If $f'(x) > 0$ for all x in (a, b), then f is increasing on $[a, b]$.
2) If $f'(x) < 0$ for all x in (a, b), then f is decreasing on $[a, b]$.
3) If $f'(x) = 0$ for all x in (a, b), then f is constant on $[a, b]$.

The intervals where the funtion f is increasing or decreasing are called *monotony intervals*.

2.3 Definition of Concavity

Let f be differentiable on an open interval I. The graph of f is concave upward on I if f' is increasing on the interval and concave downward on I if f' is decreasing on the interval.

2.4 Test for Concavity

Let f be a function whose second derivative exists on an open interval I.

1) If $f'' > 0$ for all x in I, then the graph of f is concave upward on I.
2) If $f'' < 0$ for all x in I, then the graph of f is concave downward on I.

The intervals where the funtion f is concave upward or concave downward are called *concavity intervals*.

3. The Package `GraphOfFunctions`: Some Illustrative Examples

This section describes some examples of the application of this package to cases of monotony and concavity intervals.

First, we load the package:

```
In[1]:=<<GraphOfFunctions.m
```

x	⟨-∞, -√5⟩	-√5	⟨-√5, 0⟩	0	⟨0, √5⟩	√5	⟨√5, ∞⟩
$f(x)$	↘	0	↘	-1	↗	0	↗
$f'(x)$	-	0	-	0	+	0	+

Fig. 1: The monotony intervals of $f(x) = \frac{(x^2-5)^3}{125}$.

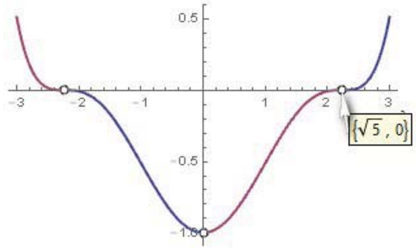


Fig. 2: Graphic of the function $f(x) = \frac{(x^2-5)^3}{125}$ indicating, with different colors, the regions corresponding to the monotony intervals.

3.1 Monotony Intervals

The first example is given by the function $f(x) = \frac{(x^2-5)^3}{125}$. The `MIntervals` command calculates the monotony intervals of f and returns a table with information inherent to such intervals.

```
In[2]:=MIntervals[(x^2-5)^3/125,x]
```

$Out[2]$ =*See Figure 1.*

The `MPlot` command returns a graphical output which shows, in different colors, the regions corresponding to the monotony intervals of an indicated function.

```
In[3]:=MPlot[(x^2-5)^3/125,{x,-3,3}]
```

$Out[3]$ =*See Figure 2.*

The next example calculates the monoty intervals of the function $f(x) = \sqrt[3]{x^2 - 2x}$.

```
In[4]:=MIntervals[Surd[x^2-2x,3],x]
```

$Out[4]$ =*See Figure 3.*

Now, the regions to the monotony intervals is obtained as:

```
In[5]:=MPlot[Surd[x^2-2x,3],{x,-2,4}]
```

$Out[5]$ =*See Figure 4.*

x	⟨-∞, 0⟩	0	⟨0, 1⟩	1	⟨1, 2⟩	2	⟨2, ∞⟩
$f(x)$	↘	0	↘	-1	↗	0	↗
$f'(x)$	-	None	-	0	+	None	+

Fig. 3: The monotony intervals of $f(x) = \sqrt[3]{x^2 - 2x}$.

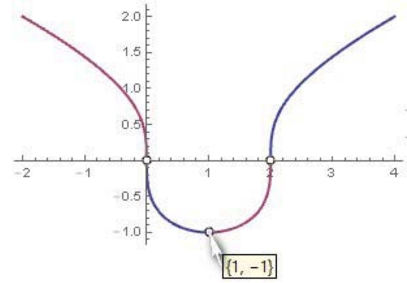


Fig. 4: Graphic of the function $f(x) = \sqrt[3]{x^2 - 2x}$ indicating, with different colors, the regions corresponding to the monotony intervals.

x	⟨-∞, -√2⟩	-√2	⟨-√2, -1⟩	-1	⟨-1, 0⟩	0	⟨0, 1⟩	1	⟨1, √2⟩	√2	⟨√2, ∞⟩
$f(x)$	↘	0	↗	1	↘	0	↗	1	↘	0	↗
$f'(x)$	-	None	+	0	-	0	+	0	-	None	+

Fig. 5: The monotony intervals of $f(x) = \left|x^4 - 2x^2\right|$.

A more complicated example is given by the function $f(x) = \left|x^4 - 2x^2\right|$.

```
In[6]:=MIntervals[Abs[x^4-2x^2],x]
```

$Out[6]$ =*See Figure 5.*

```
In[7]:=MPlot[Surd[x^2-2x,3],{x,-2,4}]
```

$Out[7]$ =*See Figure 6.*

It is interesting to see an trascendental function for which a satisfactory solution is obtained. However, as already mentioned in the introduction to this paper, this is not possible for all functions of this type.

```
In[8]:=MIntervals[x^2 E^(2/x),x]
```

$Out[8]$ =*See Figure 7.*

```
In[9]:=MPlot[x^2 E^(2/x),{x,-3,4}]
```

$Out[9]$ =*See Figure 8.*

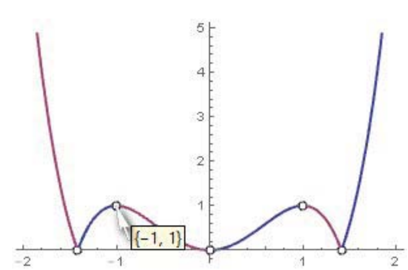


Fig. 6: Graphic of the function $f(x) = \left|x^4 - 2x^2\right|$ indicating, with different colors, the regions corresponding to the monotony intervals.

x	$\langle -\infty, 0\rangle$	0	$\langle 0, 1\rangle$	1	$\langle 1, \infty\rangle$
$f(x)$	↘	∞	↘	e^2	↗
$f'(x)$	-	None	-	0	+

Fig. 7: The monotony intervals of $f(x) = x^2 e^{2/x}$.

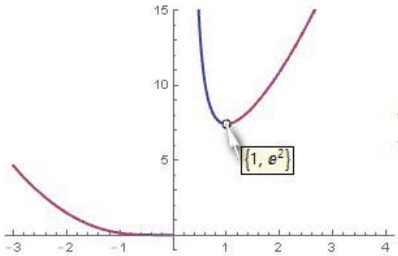


Fig. 8: Graphic of the function $f(x) = x^2 e^{2/x}$ indicating, with different colors, the regions corresponding to the monotony intervals.

3.2 Concavity Intervals

The results of this section are quite similar to those of the previous one, the difference being that, here, concavity intervals are obtained. The first example is given by the function $f(x) = \frac{(x^2-5)^3}{125}$.

```
In[10]:=CIntervals[(x^2-5)^3/125,x]
```

$Out[10]$ =*See Figure 9.*

```
In[11]:=CPlot[(x^2-5)^3/125,{x,-3,3}]
```

$Out[11]$ =*See Figure 10.*

The last example is given by the function $f(x) = \left|x^4 - 2x^2\right|$.

x	$\langle -\infty, -\sqrt{5}\rangle$	$-\sqrt{5}$	$\langle -\sqrt{5}, -1\rangle$	-1	$\langle -1, 1\rangle$	1	$\langle 1, \sqrt{5}\rangle$	$\sqrt{5}$	$\langle \sqrt{5}, \infty\rangle$
$f(x)$	$\cup$	0	$\cap$	$-\frac{64}{125}$	$\cup$	$-\frac{64}{125}$	$\cap$	0	$\cup$
$f''(x)$	+	0	-	0	+	0	-	0	+

Fig. 9: The concavity intervals of $f(x) = \frac{(x^2-5)^3}{125}$.

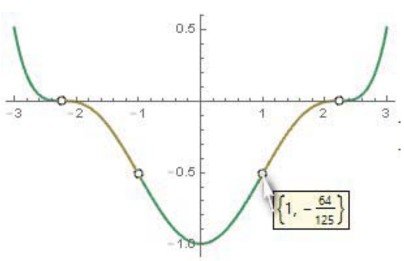


Fig. 10: Graphic of the function $f(x) = \frac{(x^2-5)^3}{125}$ indicating, with different colors, the regions corresponding to the concavity intervals.

x	$\langle -\infty, -\sqrt{2}\rangle$	$-\sqrt{2}$	$\langle -\sqrt{2}, -\frac{1}{\sqrt{3}}\rangle$	$-\frac{1}{\sqrt{3}}$	$\langle -\frac{1}{\sqrt{3}}, 0\rangle$	0	$\langle 0, \frac{1}{\sqrt{3}}\rangle$	$\frac{1}{\sqrt{3}}$	$\langle \frac{1}{\sqrt{3}}, \sqrt{2}\rangle$	$\sqrt{2}$	$\langle \sqrt{2}, \infty\rangle$
$f(x)$	$\cup$	0	$\cap$	$\frac{5}{9}$	$\cup$	0	$\cup$	$\frac{5}{9}$	$\cap$	0	$\cup$
$f''(x)$	+	None	-	0	+	4	+	0	-	None	+

Fig. 11: The concavity intervals of $f(x) = \left|x^4 - 2x^2\right|$.

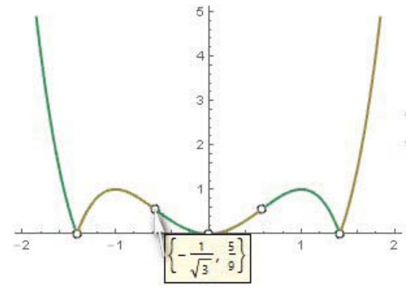


Fig. 12: Graphic of the function $f(x) = \left|x^4 - 2x^2\right|$ indicating, with different colors, the regions corresponding to the concavity intervals.

```
In[12]:=CIntervals[Abs[x^4-2x^2],x]
```

$Out[12]$ =*See Figure 11.*

```
In[13]:=CPlot[Abs[x^4-2x^2],{x,-2,2}]
```

$Out[13]$ =*See Figure 12.*

The yellow box (Figs. 2, 4, 6, 8, 10, 12) is meant to indicate that by placing the mouse cursor on a certain point, its coordinates are obtained.

4. Conclusions and Further Remarks

In this paper, we introduce a new *Mathematica* package for obtaining the monotony and concavity intervals of non-transcendental functions. The performance of the package is discussed by means of some illustrative and interesting examples. All the commands have been implemented in *Mathematica* version 10.3.1 and are consistent with *Mathematica*'s notation and results. The powerful *Mathematica* functional programming [1] features have been extensively used to make the program shorter and more eficient. From our experience, *Mathematica* provides an excellent framework for this kind of developments. Perhaps the weakest feature of the package is not supporting transcendental functions. Further work will be carried out in order to improve this issue, and other features as well.

References

[1] Roman E. Maeder. *"Programming in Mathematica."* Addison-Wesley Longman Publishing Co., Inc., 1991.

[2] M.H. Protter. and P.E. Protter. *"Calculus with Analytic Geometry."* Jones and Bartlett, 1988.

[3] R. Larson and B. Edwards. *"Calculus of a Single Variable: Early Transcendental Functions."* Cengage Learning, 2010.

[4] S. Wolfram. *"The Mathematica Book, Fifth Edition."* Wolfram Media, 2003.

Digitizing the Content of a Whiteboard to Aid Traditional Educators and Enhance the Learning Experience of Students

T. Shnoudi[1], L. Abu Hadba[2], M. Zabadne[3], O.M.F. Abu-Sharkh[4], B.H. Sababha[5]
King Abdullah II Faculty of Engineering,
Princess Sumaya University for Technology, Amman, Jordan
{tamer.shnoudi[1], luay.abuhadba[2], mohamed.joumaazabadne[3]}@students.psut.edu.jo, osama@psut.edu.jo[4], B.Sababha@psut.edu.jo[5]

Abstract - *In this paper, we introduce and implement a device that scans and erases the content written on a whiteboard. The proposed device is directed toward educators who are still comfortable with the traditional methodologies of conducting lectures inside classrooms. The device performs an easy and fast scanning process and then converts the input into an electronic image. The image is stored as a bitmap file on an attached memory device such as an SD card which can be provided to the students for later access. The scanning operation shall help the students concentrate on the lectures and listen carefully to the instructor rather than being distracted by copying the content of the whiteboard on their notebooks. The proposed device is also capable of erasing the whiteboard automatically after it is being scanned to facilitate this operation.*

Keywords: Whiteboard, Scanning, Electronic Documents, Erasing, E-learning.

1 Introduction

Whiteboard has been extensively used in educational institutions for several decades. Recently, with the advancements in technologies, many electronic devices such as laptops, tablets and electronic boards have been serving as tools for efficiently conducting lectures inside classrooms. Many educators have adopted these technologies while considerable numbers of educators, especially elderly people, are still comfortable with the traditional methodologies of teaching. Furthermore, many educators believe that the material of a course which includes mathematical analysis should be delivered by writing on a whiteboard, while the material which is descriptive in nature should be delivered using electronic means. Moreover, regular whiteboards are much cheaper than the electronic boards such as the SmartBoard [1] and the Mimio [2]. Therefore, as of the time of writing this article, the use of the regular whiteboard is still the dominant methodology for conducting lectures inside the classrooms of most, if not all, educational institutions around the world. Unfortunately, the tiresomeness of having to copy the content of a whiteboard is what keeps many students away from interacting with the lecturer which in turn distracts their attentions and considerably affects their understanding of the given material. Hence, many efforts have been provided in the literature to facilitate the use of regular whiteboards while avoiding the use of special equipment such as the ones used in electronic boards. The main and mostly used approach is based on taking images or videos of the whiteboard by using static or Pan-Tilt-Zoom (PTZ) cameras and then process the produced files while applying computer vision and image processing algorithms to achieve the best results [3]-[17]. In order to enhance the visual quality of the taken images and remove any distracting regions such as walls and shadows, Zhang and He described in [3] a system that automatically locates the boundary of a whiteboard, crops out the whiteboard region, rectifies it into a rectangle, and corrects the color to make the whiteboard completely white. In other works [4]-[5], Zhang and He proposed a Real-Time Whiteboard Capture System (RTWCS) which analyzes the sequence of captured video images in real time, classifies the pixels into whiteboard background, pen strokes and foreground objects such as educators and extracts newly written pen strokes. In [6], Wienecke et al. proposed a video-based system for recognizing unconstrained handwritten text. The system is characterized by an incremental processing strategy in order to facilitate the recognition of text as soon as they have been written on the whiteboard. In [7], Berard proposed the Magic Table which uses computer vision for scanning and spatially organizing texts and drawings on the surface of a whiteboard. Digitization of the physical ink is done by a-posteriori capture of the strokes. The main components of the Magic Table are a whiteboard, dry-ink pens, erasers, a video projector and a video camera. Saund proposed in [8] the ZombieBoard system which uses image mosaicking techniques to obtain high-resolution images of large surfaces using relatively low-resolution cameras. In [9], Xu recommended the use of PTZ cameras in addition to static cameras to enhance the quality of the captured content. Xu also used several computer vision techniques to calibrate the cameras and locate the newly updated whiteboard region to be scanned. Prabhu et al. discussed in [10] that capturing the content of a whiteboard in the presence of foreground objects such as educators has been a challenging task for camera-based approaches. Therefore, they proposed a model for whiteboard content documentation through foreground object detection. The detected object regions are inpainted with the recent whiteboard content. In [11] and [12], Dickson et al. described a classroom-capture system, Presentations Automatically Organized from Lectures (PAOL), that captures the content of a whiteboard within the setting of a classroom environment. The system acquires a sequence of images from high-resolution, fixed view cameras and extracts a series of content-rich key frames. PAOL creates multimedia Flash presentations using high-resolution cameras, a computer screen capture device and a wireless microphone. In [13]-[15], Vemulapalli and Hayes proposed methods to improve handwritten mathematical character recognition accuracy by utilizing both the recorded audio and video in a classroom. Other camera-based patented methods that capture the content of a whiteboard can be found in [16] – [17].

All of the aforementioned camera-based approaches are costly, especially the ones which require the employments of operators to control the system and manually adjust static or PTZ cameras to record lectures. Furthermore, since cameras have to be in distance from the whiteboard to be able to capture its content, the dimensions of classrooms affect the quality of the captured videos and images. Moreover, the complex algorithms and methods which are proposed to enhance the captured videos from different aspects, as the ones mentioned above, raise the computational power and slow down the processing and handling of the captured videos.

In this paper, we follow a completely different approach in capturing the content of a whiteboard. We propose a device that can be mounted on any regular whiteboard in a classroom. The device can scan the content of the whiteboard in few minutes with only a single click. It is also capable of erasing the content of the whiteboard after the scanning process or whenever needed. The implementation of the device is cheap, cost-effective and is affordable to educational institutions. It does not need certain setups or special handling neither in the classroom environment nor in the use of special equipment. The device is portable and can be mounted/unmounted on a whiteboard in few minutes. It also does not require any special operator to handle or control. The educator himself can operate the device with only a single click using a user-friendly interfacing LCD. The scanned image of the captured content is processed as a bitmap file which is stored on an SD card for later use. Since the device is mounted directly on a whiteboard, there are no foreground objects and hence special algorithms are not needed to handle their existence.

The rest of the paper is organized as follows. We introduce and discuss the design of the proposed system in Section 2. We then discuss the image formation in Section 3. We then describe the operation of the proposed system in Section 4. The implementation of the system, with discussions of the obtained results, is provided in Section 5. Finally, we conclude our paper in Section 6.

2 System Design

The design involves many systems, which are built using different types of components, to perform the scanning and erasing operations. The systems are illustrated in Figure 1. The interaction between the systems is achieved by using a controlling system which consists of two microcontrollers as shown in the figure. The second microcontroller is responsible of handling the LCD while the first microcontroller is responsible of handling the rest of the systems. The two microcontrollers are connected using a serial connection. We discuss the design of each system in the following.

A. Sensing System

As for the scanning process, we use infrared (IR) sensors to detect the light emitted from the whiteboard (black or white). The sensors are placed vertically and in close proximity of the whiteboard to detect the object by sensing the reflected IR beam from the whiteboard. In order to reduce the interference that may occur between adjacent sensors which may lead in turn to erroneous data, all adjacent sensors are placed in opposite direction of each other. In order to manage the acquired data using the microcontroller, all the outputs of the sensors are connected to the microcontroller through two levels of multiplexers which perform the selection process. Three selection lines are used for the first level and four selection lines are used for the second level. Two levels of multiplexers are used in order to cope with the many used sensors and cover the whole area of the whiteboard. The selection lines are all connected to the microcontroller to control the data acquisition process since each sensor is read one at a time.

B. Storage System

In order to store the data acquired by the sensors, a storage device is connected to the microcontroller. The microcontroller arranges the acquired data in a bitmap formatted file. The recommended storage device is an SD card since it is a popular and cheap device that can be easily obtained and handled by most electronic devices such as laptops and tablets.

C. Mechanical Movement System

Two stepper motors are used in the mechanical part to perform both the scanning and erasing operations. These motors are controlled by two h-bridges and two limit switches. The first stepper motor is connected to the metal rod and the eraser. It is used to control the eraser by bringing it forward, towards the whiteboard to perform the erasing operation when desirable, or moving it away from the whiteboard. The latter is the default position of the eraser. On the other hand, the second motor is connected to the wheels, the belt and the rail of the whiteboard. It is used to move the device along the whiteboard in order to perform the scanning and/or the erasing operation. The two h-bridges are used to let the motors move in both directions, backward and forward

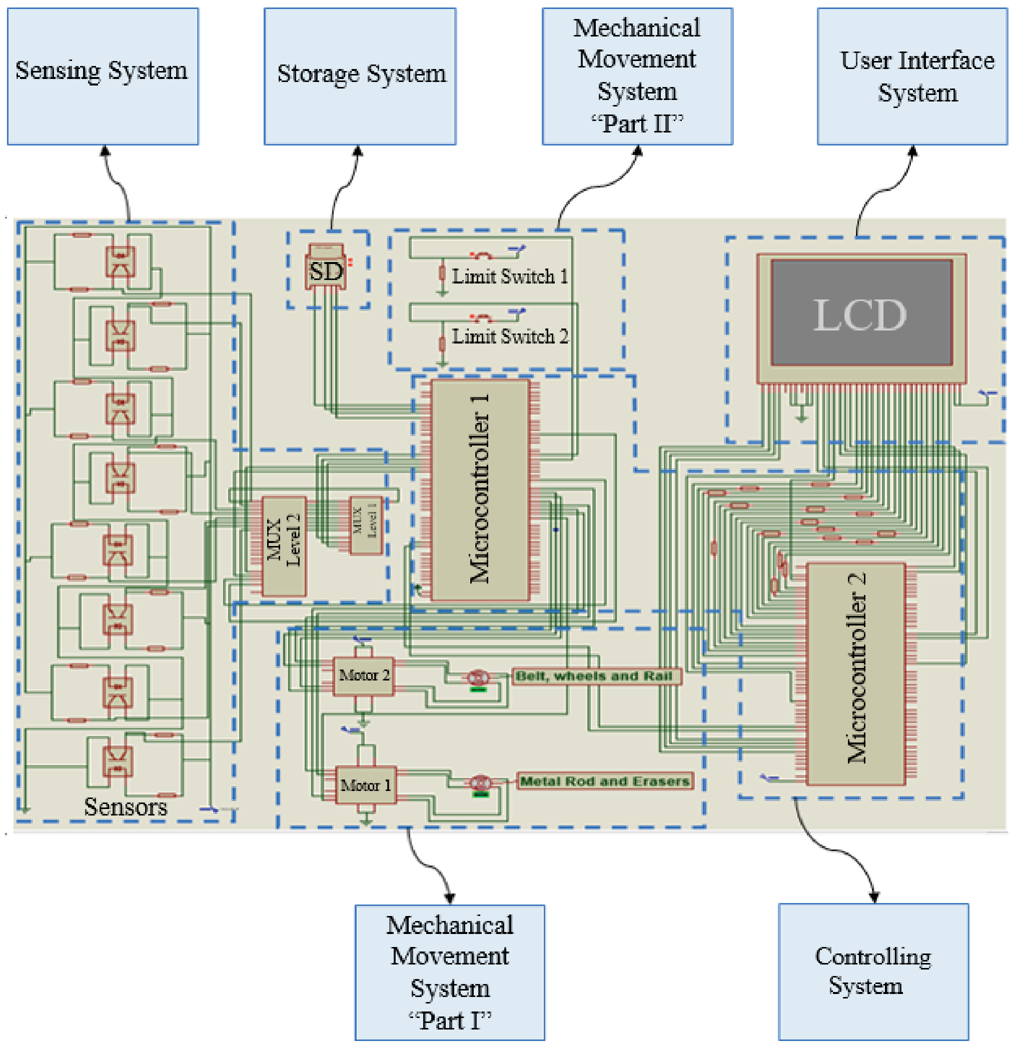


Figure 1. The proposed system design.

and hence, cover the whole area of the whiteboard. As for the limit switches, they are placed at the edges of the whiteboard and used to prevent the device from traveling beyond the predetermined points.

D. User Interface System

A touch screen LCD module is added to the device, in our design, in order to make the interaction easier between the user and the device and perform the scanning and/or erasing processes by simply choosing one of three options, Scan, Clean or Scan & Clean.

3 Image Formation

The acquired data from the sensors is processed during the scanning process and stored in a bitmap file (BMP). This file format has a structure that is capable of storing two-dimensional array images. It includes both width and height, in addition to the resolution and color types of the stored data which is in our case, black and white. The bitmap file contains headers, where some of them are mandatory and the others are optional. They are explained in the following.

A. Bitmap File Header

The 14 bytes at the beginning of the file are used to identify the BMP format 'BM' in ASCII, the size of the BMP file in bytes and the offset where the data shall start. This header is mandatory.

B. DIB Header

The DIB header includes detailed information about the BMP file format. It has 12 bytes which specify the width and height of the BMP file and also the number of bytes in the DIB header. This header is mandatory.

C. Extra Bit masks

Extra bit masks specify the color format bytes which include the number of color planes, the color depth, the number of colors in the color palette, in addition to the horizontal and vertical resolution of the image. This field is optional.

D. Pixel Array (Bitmap Data)

The Pixel Array consists of 3 bytes per pixel. In our design, we use two colors, black which has the hexadecimal code 0x00 0x00 0x00 and white which has the hexadecimal code 0xFF 0xFF 0xFF. The black material written on the whiteboard absorbs most of the infrared light emitted by the IR sensor while other colors reflect the light back to the receiver part of the sensor. This information is mandatory.

4 System Operation

As mentioned in Section 2, three processes can be handled by the device through the LCD: Scan, Clean or Scan & Clean. The flow of the processes is illustrated in the flowchart shown in Figure 2. They are also briefly explained as follows:

A. Scan

This process is initiated by the user by choosing the "Scan" option on the LCD. It starts by checking the SD card whether it is inserted in the module or not. The device will stay in hold until an SD card is properly inserted inside the module. After that, it checks whether the first limit switch is high or low. When it is low, the motor moves the device to its initial position. Then, the device starts scanning the whiteboard until the second limit switch is high. At that point, the device returns to its initial position and the scanned content are saved into the SD card as a bitmap file for later use.

B. Clean

This process is initiated by the user by choosing the "Clean" option on the LCD. It starts by checking whether the first limit switch is high or low. When it is low, the motor moves the device to its initial position. After that, the stepper motor moves the eraser toward the whiteboard and the other stepper motor moves the device along the whiteboard to erase its content until the second limit switch is high. At that point, the device returns to its initial position.

C. Scan & Clean

This process is initiated by the user by choosing the "Scan & Clean" option on the LCD. It performs the two operations mentioned above in a single click. It first performs the scanning process and then the cleaning process.

5 System Implementation

To validate our design, we implement the device using the following components, a whiteboard with 200 centimeters in width and 120 centimeters in height, TCRT5000 IR reflective sensors, 74HC4051 multiplexers, a 74HC4067 multiplexer, a Lexar SD card, two stepper motors, two limit switches, two dual L298n H-bridges, two ATmega2650 microcontrollers and a SainSmart 3.2" SSD1289 Touch Screen LCD.

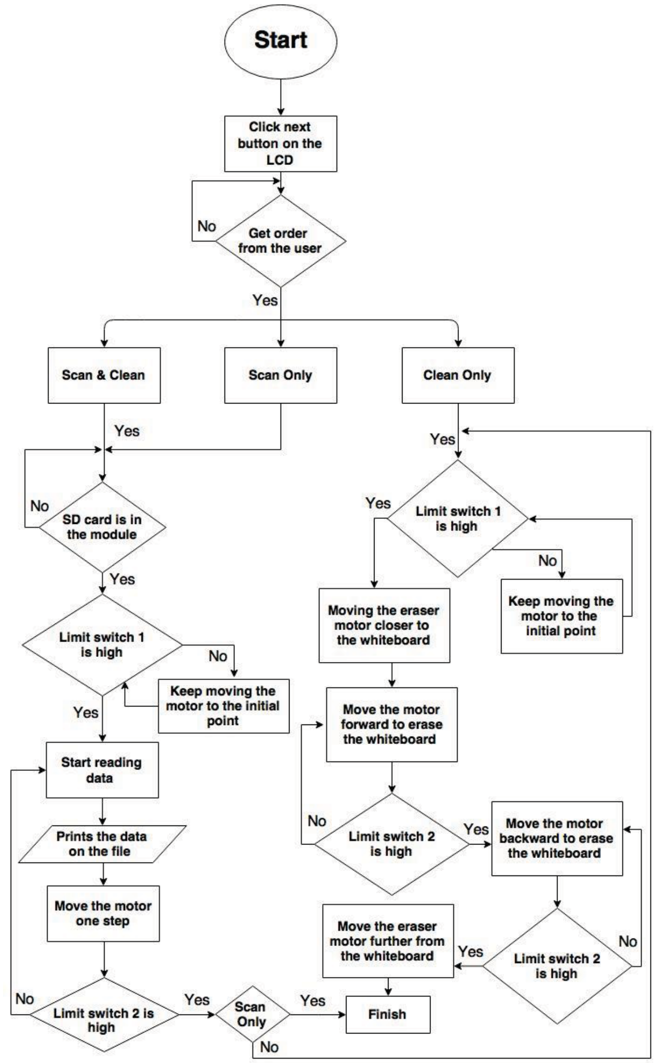


Figure 2. The flowchart of the system operation.

The TCRT5000 sensor is made up from two parts, an emitter and a detector. A Light Emitting Diode (LED) operates as the emitter while a Bipolar Junction Transistor (BJT) operates as the detector. The BJT is used as a current switch and it works by converting the reflected light on its base into a logical zero or logical one depending on the sensed data. The wavelength of the radiated light is 950mm. The sensing range of the TCRT5000 sensor is around 12mm. 117 sensors are used in order to cover the whole area of the whiteboard. Every 8 sensors are connected to one multiplexer with 8 channels (74HC4051) and the output of each multiplexer is connected to the multiplexer with 16-channels (74HC4067) forming two levels as explained earlier. Therefore, in order to span over all sensors, 15 multiplexers with 8-channels are used at the first level and only one multiplexer with 16-channels is used at the second level.

Figure 3. The prototype of the proposed design.

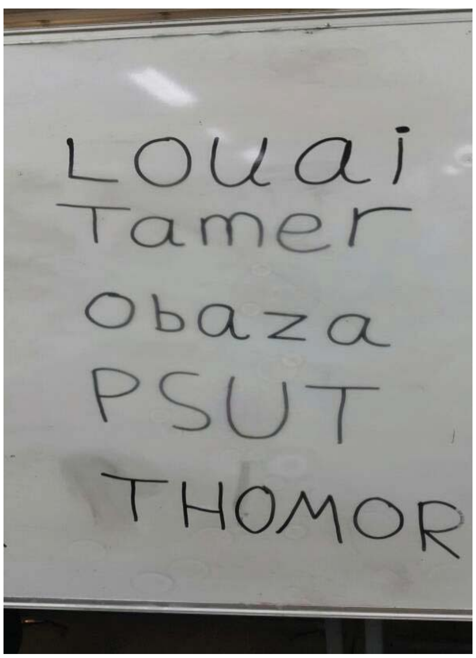


Figure 4. The content of a whiteboard.

To ensure the safety of users and in the same time seek ease of handling (light weight), the integrated components of the device are encapsulated inside a well-finished wooden frame. The prototype is shown in Figure 3.

After performing the scanning process, the content written on the whiteboard, which is shown in Figure 4, are saved on an SD card as a bitmap file. The SD card is then inserted into an SD card reader to show the obtained results on a computer.

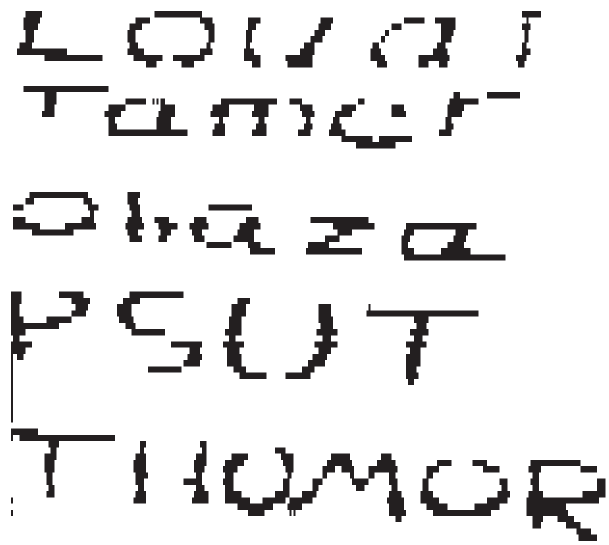


Figure 5. The captured Image (BMP format).

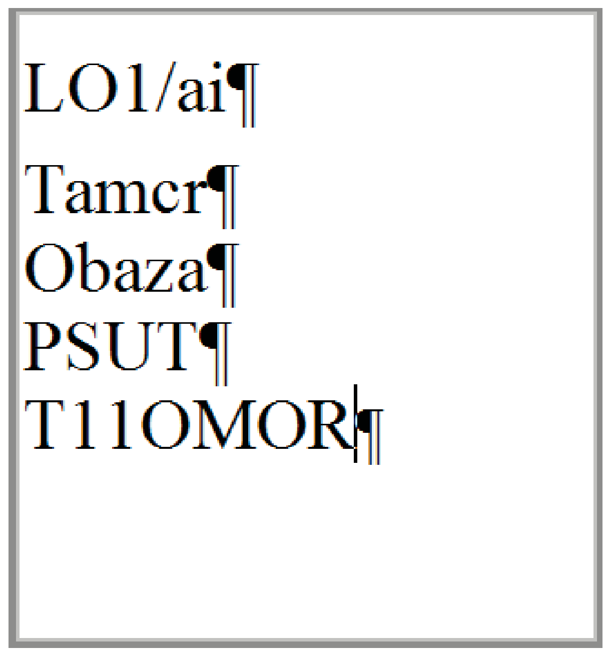


Figure 6. The obtained content after OCR.

The obtained BMP image is shown in Figure 5. Note that it almost reassembles the original content of the whiteboard. The obtained low resolution image is because of the used sensors which were mainly chosen because of their cheap price compared with others. Nevertheless, after processing the image using an Optical Character Recognition (OCR) program, most of the content written on the whiteboard was interpreted correctly as shown in Figure 6.

6 Conclusion

In this paper, we proposed a device that can be easily mounted on a whiteboard to perform two main operations, scanning and erasing. The system is a low cost device, implemented using commercial off-the-shelf (COTS) components. As for our implementation, the cost of the prototype is almost the same as the cost of the used whiteboard. Generally speaking, regular whiteboards are very cheap when compared with electronic boards or even with static or PTZ cameras. Therefore, educational institutions can easily afford the price of the proposed device. Because the device is designed to be portable, an educational institution can just have few of them to be used in the classrooms and at the class sessions where educators do not use electronic means in conducting lectures. The use of the device in such sessions saves considerable time for students and also makes them concentrate on what is taught in the lectures rather than getting distracted by writing on their notebooks.

As a future work, we intend to use IR sensors which can distinguish colors and also have much better resolution than the ones we used in our implementation in order to improve the quality of the obtained content. We also intend to implement a wireless transfer system that sends the scanned image over a wireless connection rather than storing it on an SD card.

7 References

[1] Smarttech. (2016, June 06) SmartBoard [Online]. Available: http://www.smarttech.com.

[2] Mimio. (2016, June 06) The Mimio [Online]. Available: http://www.mimio.com.

[3] Zhengyou Zhang and Li-Wei He. "Whiteboard scanning and image enhancement"; Digital Signal Processing (Elsevier), Vol. 17, pp. 414-432, May 2007.

[4] Li-Wei He and Zhengyou Zhang. "Real-Time Whiteboard Capture and Processing Using a Video Camera for Teleconferencing"; In Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol. 2, pp. 1113-1116, March 2005.

[5] Li-Wei He and Zhengyou Zhang. "Real-Time Whiteboard Capture and Processing Using a Video Camera for Remote Collaboration"; IEEE Transactions on Multimedia (IEEE), Vol. 9, Issue. 1, pp. 198-206, January 2007.

[6] Markus Wienecke, Gernot A. Fink and Gerhard Sagerer. "Towards automatic video-based whiteboard reading"; In Proc. of the Int. Conf. on Document Analysis and Recognition, Vol. 7, pp. 188-200, March 2003.

[7] Francois Bérard. "The Magic Table: Computer-Vision Based Augmentation of a Whiteboard for Creative Meetings"; In Proc. of the IEEE International Conference in Computer Vision, October 2003.

[8] Eric Saund. "Bringing the Marks on a Whiteboard to Electronic Life"; In Proc. of the Second Int'l Workshop, Vol. 1670, Issue. 0302-9743, pp. 69-78, October 1999.

[9] Richard Yi da Xu. "A Computer Vision based Whiteboard Capture System"; IEEE Workshop on Applications of Computer Vision, January 2008.

[10] Nagesh Prabhu et al. "Whiteboard documentation through foreground object detection and stroke classification"; IEEE International Conference on Systems, Man and Cybernetics, pp. 336-340, October 2008.

[11] Paul Dickson, Richards Adrion and Allen Hanson. "Whiteboard Content Extraction and Analysis for the Classroom Environment"; IEEE International Symposium on Multimedia, pp. 702-707, December 2008.

[12] Paul Dickson, Richards Adrion and Allen Hanson. "Automatic capture and presentation creation from multimedia lectures"; Annual Frontiers in Education Conference (IEEE), Vol. 38, Issue. 0190-5848, pp. T2A-14-T2A-19, October 2008.

[13] Smita Vemulapalli and Monson Hayes. "Synchronization and combination techniques for audio-video based handwritten mathematical content recognition in classroom videos"; International Conference on Intelligent Systems Design and Applications (IEEE), Vol. 11, Issue. 2164-7143, pp. 941-946, November 2011.

[14] Smita Vemulapalli and Monson Hayes. "Ambiguity detection methods for improving handwritten mathematical character recognition accuracy in classroom videos"; International Conference on Digital Signal Processing (IEEE), Vol. 17, July 2011.

[15] Smita Vemulapalli and Monson Hayes. "Audio-video based character recognition for handwritten mathematical content in classroom videos"; Integrated Computer-Aided Engineering, Vol. 21, Issue. 3, pp. 219-234, January 2014.

[16] Zhengyou Zhang and Li-Wei He. "System and method for whiteboard scanning to obtain a high resolution image"; U.S. Patent 7119816, May, 2005.

[17] Eric Saund. "Device for transcribing images on a board using a camera based board scanner"; U.S Patent 5528290, June 1996.

Trivariate Bspline solids with *Mathematica*

R. Ipanaqué[1], R. Velezmoro[1], F. Gutiérrez[1], and R. Urbina[1]
[1]Department of Mathematics, National University of Piura,
Urb. Miraflores s/n, Castilla, Piura, Perú

Abstract— *Trivariate Bspline solids are used for solid objects modeling in* $\mathbb{R}^3$. Mathematica *incorporates a several commands in order to manipulate symbolic and graphically Bspline basis functions and to graphically manipulate Bsplines curves and surfaces; however, it does not incorporate any command for the graphical manipulation of trivariate Bsplines. In this paper, we describe two new* Mathematica *commands:* `SolidPlot3D`*, use to plot the trivariate Bspline basis functions, and* `TrivariateBspline`*, use to plot the trivariate Bspline solids. The output obtained is consistent with* Mathematica*'s notation. The performance of the commands is discussed by means of using some illustrative examples.*

Keywords: Solids, NURBS, Trivariate

1. Introduction

Currently, trivariate Bspline solids are extensively used for solid objects modeling in $\mathbb{R}^3$ [2], [4], [5], [6]. The most popular and widely used symbolic computation program, *Mathematica* [3], incorporates a several commands used to graphically and symbolically manipulate the Bspline basis functions, and to graphically manipulate Bsplines curves and surfaces [7], [8]; however, it does not incorporate any command to manipulate trivariate Bsplines graphics.

In this paper, we describe two new *Mathematica* commands: `SolidPlot3D`, used for plotting the trivariate Bspline basis functions, and `TrivariateBspline` (which defines a graphics object), used to plot the trivariate Bspline solids. Both commands are coded based on the code that presents R. Maeder [1] and the commands incorporated by *Mathematica* [8], for that reason its options supported come to be the same as those commands in which them are based. The commands have been implemented in *Mathematica* v10.4 although later releases are also supported. The output obtained is consistent with *Mathematica*'s notation.

The structure of this paper is as follows: Section 2 provides some mathematical background on trivariate Bspline solids. Then, Section 3 introduces the new *Mathematica* commands for manipulating them. The performance of the commands are discussed by using some illustrative examples.

1.1 Mathematical Preliminaries

An order d Bspline is formed by joining several pieces of polynomials of degree $d-1$ with at most C^{k-2} continuity at the breakpoints. A set of nondescending breaking points $t_0 \leq t_1 \leq \ldots \leq t_m$ defines a *knot vector*

$$\mathbf{T} = (t_0, t_1, \ldots, t_m)\,,$$

which determines the parametrization of the basis functions.

Given a knot vector $\mathbf{T}$, the associated Bspline basis functions, $N_{i,d}(t)$, are defined as:

$$N_{i,1}(t) = \begin{cases} 1 & \text{for } t_i \leq t < t_{i+1} \\ 0 & \text{otherwise}\,, \end{cases}$$

for $d = 1$, and

$$N_{i,d}(t) = \frac{t - t_i}{t_{i+d-1} - t_i} N_{i,d-1}(t) + \frac{t_{i+d} - t}{t_{i+d} - t_{i+1}} N_{i+1,d-1}(t)\,,$$

for $d > 1$ and $i = 0, 1, \ldots, n$.

The trivariate Bspline solid is a tensor product solid defined by a topologically paralelepepidal set of control points $\mathbf{p}_{ijk}$, $0 \leq i \leq n_1$, $0 \leq j \leq n_2$, $0 \leq k \leq n_3$ and two knot vectors $\mathbf{U} = (u_0, u_1, \ldots, u_{n_1+d_1})$, $\mathbf{V} = (v_0, v_1, \ldots, v_{n_2+d_2})$ and $\mathbf{W} = (w_0, w_1, \ldots, w_{n_3+d_3})$ associated with each parameter u, v, w. The corresponding trivariate Bspline solid is given by

$$\mathbf{r}(u,v,w) = \sum_{i=0}^{n_1} \sum_{j=0}^{n_2} \sum_{k=0}^{n_3} \mathbf{p}_{ijk} N_{i,d_1}(u) N_{j,d_2}(v) N_{k,d_3}(w)$$

A trivariate NURBS solid can be represented as

$$\mathbf{r}(u,v,w) = \frac{\sum_{i=0}^{n_1} \sum_{j=0}^{n_2} \sum_{k=0}^{n_3} h_{ijk} \mathbf{p}_{ijk} N_{i,d_1}(u) N_{j,d_2}(v) N_{k,d_3}(w)}{\sum_{i=0}^{n_1} \sum_{j=0}^{n_2} \sum_{k=0}^{n_3} h_{ijk} N_{i,d_1}(u) N_{j,d_2}(v) N_{k,d_3}(w)}$$

where $h_{ijk} > 0$ is a weighting factor. If all $h_{ijk} = 1$, the trivariate Bspline solid is recovered.

2. Manipulation of trivariate Bspline solids

2.1 The new command `SolidPlot3D`

The `SolidPlot3D` command generates a three-dimensional solid plot of a certain f expression as a function of three variables. The syntax of this command is:

`SolidPlot3D[`$f, \{x, xmin, xmax\}, \{y, ymin, ymax\}, \{z, zmin, zmax\}, options$`]`

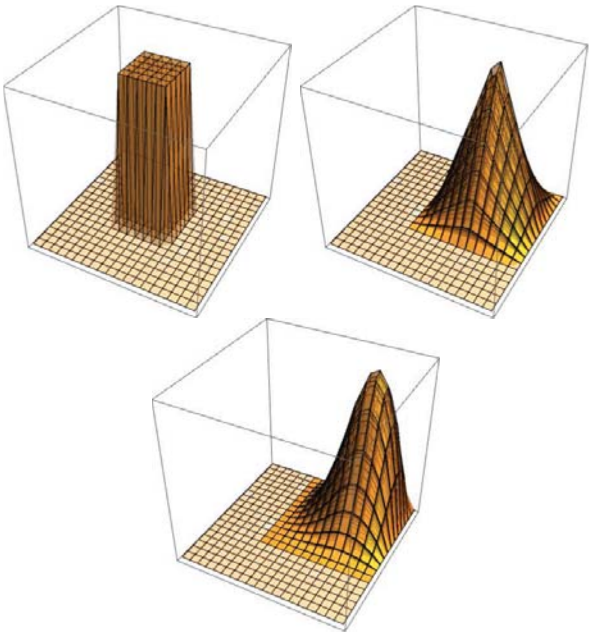


Fig. 1: Trivariate Bspline basis functions for $\mathbf{U} = \mathbf{V} = \mathbf{W} = \{0, 0, 0, 1, 2, 3, 3, 3\}$ and: (left) $d_1 = d_2 = d_3 = 0$; (center) $d_1 = d_2 = d_3 = 1$; (right) $d_1 = d_2 = d_3 = 2$.

The options supported by this command are: some options of the `Plot3D` command (`AxesLabel`, `Background`, `BoxRatios`, `PlotPoints`, `PlotRange` and `PlotStyle`) and all options of the `Graphics3D` command.

For instance, this shows a triconstant, trilineal and triquadratic trivariate Bspline basis function:

```
In[1]:=U=V=W={0,0,0,1,2,3,3,3};
In[2]:=SolidPlot3D[
        BSplineBasis[{#,U},3,u]
        BSplineBasis[{#,V},3,v]
        BSplineBasis[{#,W},3,w],
        {u,0,3},{v,0,3},{w,0,3},
        PlotStyle->Directive[Opacity[.2]],
        BoxRatios->1,
        PlotPoints->20
                  ]& /@ {0,1,2}
```

Out[2] =*See Figure 1.*

In fact, the trivariate Bspline basis function are solid. This is best seen in the figure 2.

Note that `BsplineBasis` is a built-in *Mathematica* command, so it is possible to obtain analytical expressions for any trivariate Bspline basis function, for example this shows a trilineal trivariate Bspline basis function:

```
In[3]:=U=V=W={0,0,0,1,2,3,3,3};
In[4]:=BSplineBasis[{1,U},3,u]
       BSplineBasis[{1,V},3,v]
```

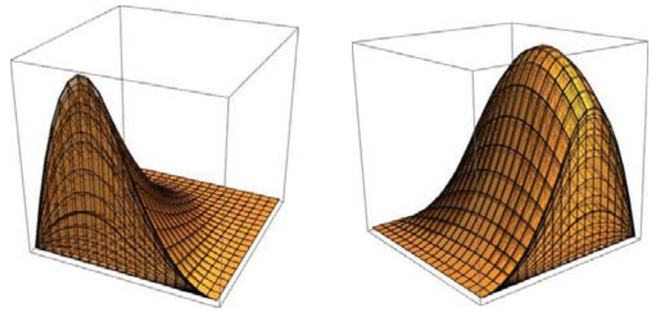


Fig. 2: Two different views, with different cuts, of the triquadratic trivariate Bspline basis function ($\mathbf{U} = \mathbf{V} = \mathbf{W} = \{0, 0, 0, 1, 2, 3, 3, 3\}$ and $d_1 = d_2 = d_3 = 2$).

```
       BSplineBasis[{1,W},3,w]//
       PiecewiseExpand
```

Out[4] =

$$\begin{cases} -(u-3)(v-3)(w-3) & 2 \le u \le 3 \wedge 2 \le v \le 3 \wedge 2 \le w \le 3 \\ (u-1)(v-3)(w-3) & 1 \le u < 2 \wedge 2 \le v \le 3 \wedge 2 \le w \le 3 \\ (u-3)(v-1)(w-3) & 2 \le u \le 3 \wedge 1 \le v < 2 \wedge 2 \le w \le 3 \\ -(u-1)(v-1)(w-3) & 1 \le u < 2 \wedge 1 \le v < 2 \wedge 2 \le w \le 3 \\ (u-3)(v-3)(w-1) & 2 \le u \le 3 \wedge 2 \le v \le 3 \wedge 1 \le w < 2 \\ -(u-1)(v-3)(w-1) & 1 \le u < 2 \wedge 2 \le v \le 3 \wedge 1 \le w < 2 \\ -(u-3)(v-1)(w-1) & 2 \le u \le 3 \wedge 1 \le v < 2 \wedge 1 \le w < 2 \\ (u-1)(v-1)(w-1) & 1 \le u < 2 \wedge 1 \le v < 2 \wedge 1 \le w < 2 \end{cases}$$

2.2 The new command `TrivariateBspline`

The new command `TrivariateBspline` defines a graphic object and for to view this object you must use the `Graphics3D` command. The syntax of this command is:

`TrivariateBspline[`*pts*, *options*`]`

The options supported by this command are: `PlotPoints`, `SplineDegree`, `SplineKnots` and `SplineWeights`.

As a first example we compute the trivariate Bspline solid of order 3 for a set of points as follows:

```
In[5]:=pts = Table[{u,1/(1+u^2+v^2),
       1/(1+u^2+w^2)},{u,-1,1,1/2},
       {v,-1,1,1/2},{w,-1,1,1/2}];
```

```
In[6]:=Graphics3D[{Meshing[pts],Yellow,
       Opacity[.2],TrivariateBspline[pts]},
       ViewPoint->{3.,-1.6,0.8}]
```

Out[6] =*See Figure 3 (above).*

(The command `Meshing` has also been coded by the authors.)

```
In[7]:=Graphics3D[{Yellow,
       TrivariateBspline[pts]},
       ViewPoint->{3.,-1.6,0.8},
       Boxed->False]
```

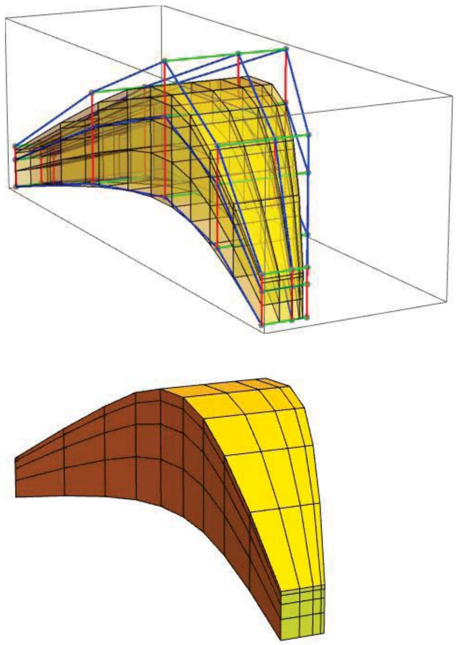


Fig. 3: Trivariate Bspline solid.

Out[7] =*See Figure 3 (below).*

```
In[5]:=pts = {{{{-2, 0, 0},
   {-3/2, 0, 0}, {-1, 0, 0}},
   {{-2, -2, 0}, {-3/2, -3/2, 0},
   {-1, -1, 0}}, {{0, -2, 0},
   {0, -3/2, 0}, {0, -1, 0}},
   {{2, -2, 0}, {3/2, -3/2, 0},
   {1, -1, 0}}, {{2, 0, 0},
   {3/2, 0, 0}, {1, 0, 0}},
   {{2, 2, 0}, {3/2, 3/2, 0},
   {1, 1, 0}}, {{0, 2, 0},
   {0, 3/2, 0}, {0, 1, 0}},
   {{-2, 2, 0}, {-3/2, 3/2, 0},
   {-1, 1, 0}}, {{-2, 0, 0},
   {-3/2, 0, 0}, {-1, 0, 0}}},
   {{{-2, 0, 1}, {-3/2, 0, 1},
   {-1, 0, 1}}, {{-2, -2, 1},
   {-3/2, -3/2, 1}, {-1, -1, 1}},
   {{0, -2, 1}, {0, -3/2, 1},
   {0, -1, 1}}, {{2, -2, 1},
   {3/2, -3/2, 1}, {1, -1, 1}},
   {{2, 0, 1}, {3/2, 0, 1},
   {1, 0, 1}}, {{2, 2, 1},
   {3/2, 3/2, 1}, {1, 1, 1}},
   {{0, 2, 1}, {0, 3/2, 1},
   {0, 1, 1}}, {{-2, 2, 1},
   {-3/2, 3/2, 1}, {-1, 1, 1}},
   {{-2, 0, 1}, {-3/2, 0, 1},
   {-1, 0, 1}}}, {{{-2, 0, 2},
   {-3/2, 0, 2}, {-1, 0, 2}},
   {{-2, -2, 2}, {-3/2, -3/2, 2},
   {-1, -1, 2}}, {{0, -2, 2},
   {0, -3/2, 2}, {0, -1, 2}},
   {{2, -2, 2}, {3/2, -3/2, 2},
   {1, -1, 2}}, {{2, 0, 2},
   {3/2, 0, 2}, {1, 0, 2}},
   {{2, 2, 2}, {3/2, 3/2, 2},
   {1, 1, 2}}, {{0, 2, 2},
   {0, 3/2, 2}, {0, 1, 2}},
   {{-2, 2, 2}, {-3/2, 3/2, 2},
   {-1, 1, 2}}, {{-2, 0, 2},
   {-3/2, 0, 2}, {-1, 0, 2}}}};
```

```
In[9]:=uk = {0, 0, 0, 1, 1, 1};
       vk = {0, 0, 0, 1, 1, 2, 2, 3, 3,
             4, 4, 4};
       wk = uk;
```

```
In[12]:=w1 = {1, 1, 1};
        w2 = {1, 1/Sqrt[2], 1,
              1/Sqrt[2], 1, 1/Sqrt[2],
              1, 1/Sqrt[2],1};
        w3 = w1;
        w = Table[
             w1[[i]] w2[[j]] w3[[k]],
             {i, 1, 3}, {j, 1, 9},
             {k, 1, 3}];
```

```
In[16]:=Graphics3D[{Meshing[pts21],
         Yellow,
         TrivariateBspline[ pts,
         PlotPoints -> {5, 35, 5} ]}]
```

Out[16] =*See Figure 4 (top-left).*

```
In[17]:=Graphics3D[{Meshing[pts],
         Yellow,
         TrivariateBspline[ pts,
         SplineWeights -> w,
         PlotPoints -> {5, 35, 5} ]}]
```

Out[17] =*See Figure 4 (top-right).*

```
In[18]:=Graphics3D[{Meshing[pts],
        Yellow,
        TrivariateBspline[ pts,
        SplineKnots -> {uk, vk, wk},
        PlotPoints -> {5, 35, 5} ]}]
```

Out[18] =*See Figure 4 (bottom-left).*

```
In[19]:=Graphics3D[{Meshing[pts],
         Yellow,
         TrivariateBspline[ pts,
          SplineWeights -> w,
          SplineKnots -> {uk, vk, wk},
          PlotPoints -> {5, 35, 5} ]}]
```

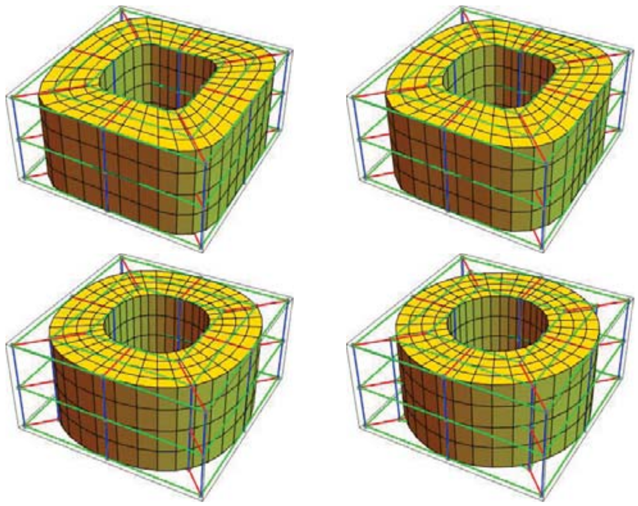


Fig. 4: Modeling a hollow cylinder with trivariate NURBS solid.

$Out[19] =$ *See Figure 4 (bottom-right).*

Moreover, figure 4 shows the adjustment process, by varying the node vectors and weights, modeling a solid hollow cylinder with trivariate NURBS. In practice this is achieved by changing the default values of `SplineWeights` and `SplineKnots`. Similarly, in figure 5 the modeling of a solid torus and in figure 6 the modeling a solid helix are seen.

2.3 Conclusions and Further Remarks

In this paper two new *Mathematica* commands: `SolidPlot3D` (to plotting the trivariate Bspline basis functions) and `TrivariateBspline` (to plotting the trivariate Bspline solids) are introduced. The options supported for both commands come to be the same as those commands in which them are based. The commands have been implemented in *Mathematica* v10.4 although later releases are also supported. The output obtained is consistent with *Mathematica*'s notation. We are currently working on the extension of these results to the four-dimensional case. The obtained results will be reported elsewhere.

References

[1] R. E. Maeder. *"Programming in Mathematica."* Addison-Wesley, Boston, 1991.

[2] V. Ng-Thow-Hing and others. "Shape reconstruction and subsequent deformation of soleus muscle models using B-spline solid primitives"; BiOS'98 International Biomedical Optics Symposium, 423–434, May 1998.

[3] S. Wolfram. *"The Mathematica Book."* Fifth Edition. Wolfram Media, 2003.

[4] B. Schmitt and others. "Constructive sculpting of heterogeneous volumetric objects using trivariate b-splines"; The Visual Computer, Vol. 20, Issue 2–3, 130–148, 2004.

[5] T. Martin and others. "Volumetric parameterization and trivariate B-spline fitting using harmonic functions"; Computer Aided Geometric Design, Vol. 26, Issue 6, 648–664, 2009.

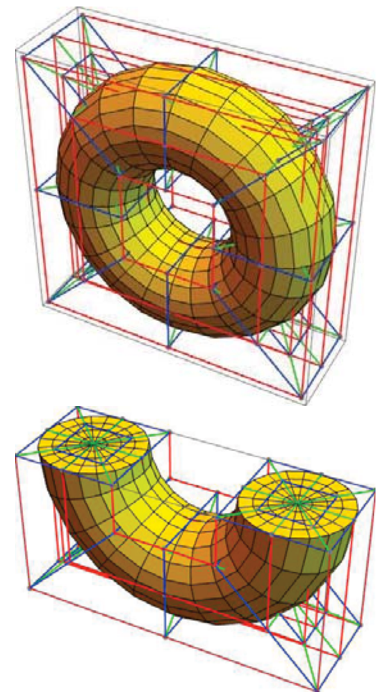


Fig. 5: Modeling a solid torus with trivariate NURBS solid.

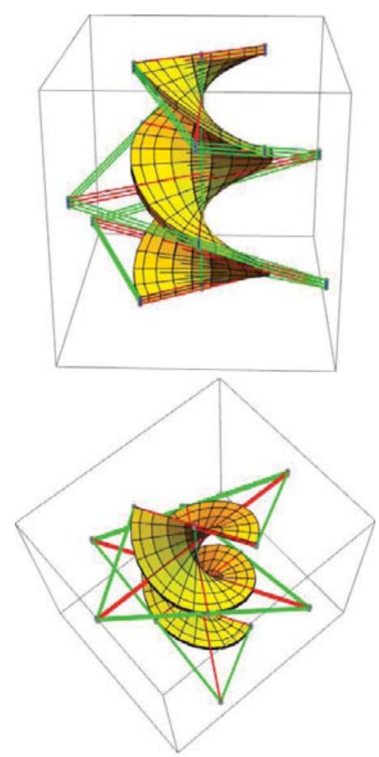


Fig. 6: Modeling a solid helix with trivariate NURBS solid.

[6] J. H. Kim and Y. J. Lee. “Trivariate B-spline approximation of spherical solid objects”; Journal of information processing systems, Vol. 10, Issue 1, 23–35, 2014.
[7] Wolfram. (2016). Work with Spline Functions. [Online]. Available: https://reference.wolfram.com/language/howto/WorkWithSplineFunctions.html
[8] Wolfram. (2016). Graphics Objects. [Online]. Available: https://reference.wolfram.com/language/guide/GraphicsObjects.html