End of Semester Software Problem Solving and Design Projects in a First-Year Engineering Course

Krista M. Kecskemety and Lauren N. Corrigan
Department of Engineering Education
The Ohio State University
Columbus, Ohio, 43210

***Abstract*—Introductory program courses often have a final project. The development of a game using programming languages learned is possible choice for this project. A large Midwestern public university has been assigning a programming project at the end of it's first semester, first-year engineering course. Previous projects have included an IR detection program and programming a train scenario. Recently, faculty have been piloting a game software design project in these courses. Student survey results have demonstrated that students prefer the game project to the other projects for many reasons including they are more enjoyable and they help develop necessary engineering skills. The results from this pilot have lead to many questions that will be explored in future studies.**

***Keywords*—Games; Software design project; First-year engineering**

I. Introduction

The use of game development in programming courses has been increasing in popularity. There is research showing the use of game development in upper level courses [1] and also in first-year courses [2]–[5]. There is also research in the area of choice in student projects and its impact on student motivation and interest [6]. This research study developed out of these two areas of choice and game software development.

At The Ohio State University, the first-year engineering honors program has a software design project at the end of the first semester problem solving course and the standard first-year engineering course has a software problem solving project at the end of the course. The semester focuses on problem solving using computer tools. Both honors and standard track cover programs like Excel and MATLAB, and the honors version also covers C/C++ programming. For many years the honors software design project was to develop a code to detect an IR signal and compute its frequency and the project for the standard course was to control the operation of a model train set. These projects provided students with real time programming and gave the students experience working with the controller that they would use in the second semester. For most of these students the second semester course involves designing, building and programming an autonomous robot or designing and controlling an advanced energy vehicle.

Recently, some faculty have piloted an open-ended game software design project in both of these courses. This work started in the honors course and student surveys have been used to examine the project. The game project has expanded into the standard course. This game project has allowed students the freedom to propose any game they desire, work with the instructor to decide the difficulty of the project, and then students may use a microcontroller, MATLAB or Linux to implement the game depending on the course. The work presented in this Research to Practice, Work in Progress paper will be an initial step towards answering the research questions: What are the student perceptions of the different software design project/problem solving project options and which produce more engagement? Does the project type (Game, IR, Train) impact the types of programming skills required/used to complete the project and in turn does it impact the learning of these skills?

This Research to Practice, Work in Progress paper presents the preliminary results obtained from student surveys showing statistically significant differences in student perceptions between the different projects. It also provides key trends and findings that will help develop future qualitative research that will include content analysis of the software codes developed in the project to assess programming skills learned. Through the Work in Progress format, we hope to gain added insights into our study.

II. Background

A. Literature Review

Games are increasingly being used in open-ended projects. One university added games into a robotics context [5]. Students in this class programmed a robot arm to play tic-tac-toe using MATLAB. Results of a study of this robotic game application focused on students motivation to work on the project [5]. A first-year engineering course at another university has a game project for their introductory MATLAB course [4]. One interesting observation mentioned by the instructors of this course was that students seemed more willing to explore MATLAB and find new commands when working with the games and GUIs [4]. Another study on games in a first-year engineernig course investigated using games in a LabVIEW programming course [3]. They

978-1-5090-5920-1/17/$31.00 ©2017 IEEE

found that students who completed the game project in the LabVIEW thought their creativity was more developed than the students in a C and C++ programming course without the game project [3].

B. Previous Research

This pilot study started in a honors sequence of the first-year engineering course in Autumn 2014. In this course students learned MATLAB and C and C++ programming in their first semester class. The final course software design project typically had been to program a microcontroller to detect an IR frequency. Through this pilot, some course sections offered students the choice to design a game. Many students chose to design the game using the microcontroller, although some faculty allowed students to use the desktop environment if the specific game warranted it. For example a typing game made more sense to implement with the desktop computer. A previous study [7] examined student perceptions of the two project types from Autumn 2015 and found promising results associated with the game design project.

C. Expanding the Pilot

Based on these positive results, the pilot project was translated into three 72 student sections of the standard first-year engineering course in Autumn 2016. However, the standard first-year course has many differences compared to the honors course. The standard course only covers MATLAB programming and uses limited programming of an Arduino microcontroller. Therefore, in order to implement the game project into the standard course, there would need to be different expectations in the tools used. In order to meet this need, the students were provided with MATLAB game pieces and boards developed by [4]. For example, for Connect 4 there was a game board and black and red pieces to play the game with. This can be seen in Fig. 1. This software was obtained through MathWorks and the Creative Commons Attribution-ShareAlike 3.0 Unported License [8].

Other changes were made to the project to increase its real world applicability and make students think about the value of their product. Students were required to conduct interviews with potential users of their games (typically their classmates) before creating their game. In these interviews the team was to ask the following questions:

- What are some games you have played recently?
- What are some games you might enjoy playing on MATLAB?
- In your opinion what is the optimal length a game should take to play?
- What aspects of a game make it enjoyable to you to play?

After two weeks of game development, the team was required to find someone to beta test their game. Again, this was typically their classmates but could have also included a friend or teaching assistant. After beta testing, they again conducted a similar interview that included suggested changes to the game. Once the game was finished, students were required to create an online notebook documenting their project, a five minute pitch and demonstration video, and an advertisement of their product. The online notebook platform was similar to what is used in the second semester course [9]. Some teams chose to add a ”price” for their product into the advertisement or the pitch and although this was not required, future versions of this project might include a financial analysis in the business plan. Figure 1 shows a sample advertisement for one team. The Connect 4 gameboard is also shown in this advertisement.

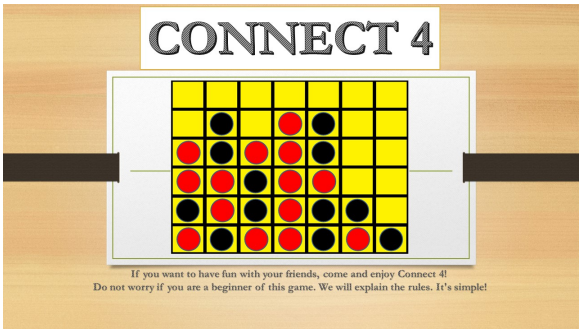

Fig. 1. Sample game advertisement created by a team who designed a Connect 4 game.

III. Methodology

A. Survey Creation

The survey questions were based on questions established by [3] and used previously by [7]. These questions are based on Engagement Theory where students are engaged in their learning if they are able to create, donate and relate [10]. The creation phase requires them to be able to define the problem and create something, the donate phase requires it to be an authentic scenario, and the relate phase is accomplished through teamwork. All three of these phases of Engagement Theory were a part of the software design project and therefore survey questions around this framework was appropriate for this investigation.

The survey consisted of the following of Likert scale questions where [Project] was either SDP for those students that completed the game project, or Train Problem Solving Project for the traditional train project:

1) I enjoyed the [Project] .
2) The [Project] enhanced my understanding of the design process.
3) The [Project] operated under a manageable timeframe.
4) The [Project] was a good example of applied engineering.

TABLE I
STUDENT SURVEY RESULTS COMPARING THE GAME AND THE TRAIN PROJECTS. STATISTICALLY SIGNIFICANT VALUES ($p<0.05$ ARE DESIGNATED WITH A *.

Questions		Train N=1098		Game N=204		Difference Game-Train
		Average	Std	Average	Std	Average
1.	I enjoyed the [Project] .	3.83	1.06	4.30	0.78	0.47*
2.	The [Project] enhanced my understanding of the design process.	3.92	0.97	4.27	0.76	0.35*
3.	The [Project] operated under a manageable timeframe.	4.04	0.91	4.24	0.83	0.20*
4.	The [Project] was a good example of applied engineering.	4.20	0.80	4.27	0.78	0.07
5.	The [Project] helped develop my engineering teamwork skills.	3.92	1.00	4.38	0.74	0.46*
6.	The [Project] helped develop my creativity in engineering.	3.75	1.04	4.27	0.82	0.52*
7.	The [Project] helped prepare me for my future as an engineer.	3.77	1.03	4.07	0.97	0.30*

5) The [Project] helped develop my engineering teamwork skills.
6) The [Project] helped develop my creativity in engineering.
7) The [Project] helped prepare me for my future as an engineer.

The survey also included the following open ended response question: *Please provide any additional comments you have on the [Project]. What was your favorite part of the [Project]? Any changes you would make to the [Project]?*

IV. PRELIMINARY RESULTS

The survey results can be seen in Table I with averages and standard deviations for each question as well as a difference in the averages between the Game and Train project. Statistical significance ($p<0.05$) was determined through a Mann Whitney U test. The train project had 1098 responses and the pilot game project had 204 responses. From these results it is clear that the game project had statistically significantly higher scores in all areas except one. The only area without a higher score was that it was "a good example of applied engineering". For this statement both groups rated this category highly.

The top three statements with the highest differences between the game and train project were the areas of enjoying the project, developing creativity, and developing engineering teamwork skills. It is not surprising that students found more enjoyment with the game project and it also is not surprising that they thought it developed more creativity. Since students completing the game project had many options to chose and many ways to implement the game once they chose it, creativity development is not surprising.

However, the teamwork development was something unexpected. Both projects had teams of four and had similar requirements in terms of documentation. There are a few factors that may have resulted in the increased teamwork development and further investigation is required to determine the impact of these. One reason teamwork may have been developed more with the game project is the higher programming difficulty of some of the tasks may have forced more teamwork compared to the train scenario. This is an area that could warrant another study to investigate if this was a factor. Moreover, many game project teams decided to create a few smaller games and therefore some may have divided up work that way. Compared to the train project where students are encouraged to divide up the programming, the project does lend itself more to having one student do the programming during the lab section. For example, one train project student mentioned in the open ended response:

> *"I didn't like how it made teamwork hard. Only one person could write the code at one time, so they felt as though they were doing the majority of the hard work."*

This is something that could be analyzed in the future since the students who completed the game were required to document how they split up the project. Another reason teamwork development may have been increased in the game project could be how the teams were developed. In the course teams were developed early in the semester and labs were conducted with these teams throughout the semester. In the train project courses, the train project was completed with this same team. However, in the game project course new teams were randomly created immediately prior to the start of the project. This could have had a large impact on the teamwork skill development and is something that would need to be investigated further. While some students mentioned not liking the group change, the following two open ended responses captured the positive change seen by game project students:

> *"I enjoyed the group change as it helped me better understand teamworking environments"*

and

> *"I enjoyed working with a new team. I feel that it helped me build my team-working skills, as well as my ability to communicate."*

V. Conclusions and Future Work

While the project has been a success on a small scale, large scale expansion of the project to all 2000 first-year engineering students will require some adjustments to the current model and additional training for teaching assistants and faculty involved in these courses. For faculty, learning to be comfortable with their students completing vastly different games will be something that will need to be coached throughout the project. Finding ways to have them help their students without needing to know how to program every possible scenario and game will be important. Further training will be necessary for the undergraduate and graduate teaching assistants involved in the program. When the pilot game project was instituted teaching assistants in that course were each assigned a game to work on. This allowed each instructional team to have some knowledge of the different types of games that might be assigned. While this worked well on a small scale, more formal training may need to be implemented in future iterations.

As mentioned previously, some other changes to the project may include further developing the business plan including and financial model for the game. One missing piece of the game project in the standard pilot is the lack of a controller or physical system to interact with. This is a large part of the train project and something that must be investigated in future expansion. One possibility is to use an NI Elvis board that will allow the MATLAB program to communicate with the board to light up LEDs or accept input. This could transform the game project to meet the additional learning objectives that the hardware can provide.

An additional study could look at the teamwork aspect of these projects and how a team of four can all receive the same learning experience through this project. One possible approach would be to try pair programming techniques [11]. Whatever technique is used, an assessment of the teamwork and learning outcomes would be needed.

Also, while student perceptions of the game project being more positive than the train project are a great start, further investigations are needed to determine the additional learning gained through the project. Another study would likely compare the computer programs generated through these different projects and score the programs based on levels of programming complexity and amount and use of basic programming skills. This will allow a direct comparison in the type and level of programming needed to complete the task.

Acknowledgment

The authors would like to thank the faculty and students who have participated in these projects and allowed their work to be shown.

References

[1] B. Maxim, "Serious games as software engineering capstone projects," in *2008 Annual Conference & Exposition*. Pittsburgh, Pennsylvania: ASEE Conferences, June 2008, https://peer.asee.org/4028.

[2] J. Estell, "Writing card games: An early excursion into software engineering principles," in *2005 Annual Conference*. Portland, Oregon: ASEE Conferences, June 2005, https://peer.asee.org/14554.

[3] E. Helber, M. Brockman, and R. Kajfez, "Gaming with labview: An attempt at a novel software design project for first-year engineers," in *Proceedings of the 6th First Year Engineering Experience (FYEE) Conference*, 2014.

[4] K. Ossman and G. Bucks, "First year student team projects using matlab," in *Proceedings of the 5th First Year Engineering Experience (FYEE) Conference*, 2013.

[5] T. R. Hamrick and R. A. Hensel, "Putting the fun in programming fundamentals - robots make programs tangible," in *2013 ASEE Annual Conference & Exposition*. Atlanta, Georgia: ASEE Conferences, June 2013, https://peer.asee.org/22397.

[6] T. Shepard, "Implementing first-year design projects with the power of choice," in *2013 ASEE Annual Conference & Exposition*. Atlanta, Georgia: ASEE Conferences, June 2013, https://peer.asee.org/19722.

[7] K. Kecskemety, A. Drown, and L. Corrigan, "Examining software design projects in a first-year engineering course: How assigning an open-ended game project impacts student experience," in *Proceedings of the 124th American Society for Engineering Education Annual Conference & Exposition*, 2017.

[8] K. Ossman and G. Bucks, "Matlab courseware: Game projects," MathWorks, https://www.mathworks.com/academia/highschool/courseware/games/?s_tid=srchtitle, 2014.

[9] R. Kajfez, K. Kecskemety, and M. Kross, "Electronic notebooks to document the engineering design process: From platform to impact," *Computers in Education Journal*, vol. 7, no. 1, pp. 37–46, January-March 2016.

[10] G. Kearsley and B. Shneiderman, "Engagement Theory: A Framework for Technology-Based Teaching and Learning," *Educational Technology*, vol. 38, no. 5, pp. 20–23, 1998.

[11] N. D. Fila and M. C. Loui, "Work-in-progress who's driving? structured pairs in an introductory electronics laboratory," in *2010 IEEE Frontiers in Education Conference (FIE)*. IEEE, oct 2010.