Running Head: Service Learning and Computer Games

Service Learning and Computer Games

Bruce R. Maxim, Matthew Sable, Margaret Turton, John Cristiano

University of Michigan-Dearborn

Abstract

This paper describes use of industry-based capstone design courses to provide service learning opportunities to students. In this type of course, students work as members of small teams to complete software development projects. These projects proceed from requirements gathering, to analysis, design, implementation, and delivery of products to real-world clients. In recent years, several of these projects have involved the development of serious games for real-world clients in the non-profit sector. Serious games and simulations can be good candidates for student projects that provide opportunities to manage projects with real-world development constraints and deadlines. These projects can help students learn how to use their roles as computing professionals to address community needs. Initial student survey data indicates satisfaction with the service learning experience.

BACKGROUND

The idea of a capstone design course for undergraduate computing students is not new. Two of the authors have been teaching them for many years. Capstone courses have traditionally tried to provide senior students with experiences similar to those encountered in professional computing practice. Some instructors argue that the purpose of such a course is to help students integrate theoretical computing concepts with the demands of real-world project management (Maxim and Akingbehin, 2006).

At the University of Michigan-Dearborn, most of the student capstone projects implemented during the past fourteen years do not involve game development. Several recent capstone software engineering projects have involved developing serious games for real-world clients in the non-profit sector.

The primary goals for the requested serious game projects are usually educational in nature. Students in our Computer Science Game Design application track need to create game-based capstone design projects. One purpose of this paper is to examine the lessons learned by students developing serious games as capstone design projects. The expectation is that students working on serious games will learn many of the same lessons as students working on traditional software projects.

Serious Games

Serious games make use of the artistic medium of games to deliver a message, teach a lesson, or provide an experience. Serious games may be entertaining, but that is not their primary purpose. Video games can teach hand-eye coordination, spatial relationships, and encourage exploratory experiences. Immersion in simulated environments has increased learning speed and

retention for some tasks. Video games can engage players for two or more hours, yet these same students may lose interest in classroom activities after only fifteen minutes (Michael and Chen, 2006).

Several constraints present in serious game creation provide for interesting design tradeoff considerations during the software development process. Developers often design serious games for wider audiences and not just hardcore game players. One of the design challenges present for serious games is that the games produced often need to run in lower end computing environments without sacrificing the game's core-learning objectives. Reality does not map exactly in a computer game, so software engineers must weigh each simplification assumption carefully to avoid damaging the verisimilitude of an educational experience. Shortcuts from entertainment games (i.e. use of random numbers, time compression, headache removal, perfect communication) may not be appropriate when the learning outcomes for a game may have life and death consequences in the real-world (Michael and Chen, 2006).

The intellectual challenges present in making a good game are part of the reason people want to work on the game development process. Game development projects are often large and complex projects. The workflow to complete a large game is non-trivial. Project managers need to coordinate the efforts of programmers and third party game asset creators. Game developers need to manage both technical risk and project risk to ensure the timely completion of a good serious game. Creating and modifying game engine code requires good algorithmic programming knowledge on the part of the developers (Blow, 2004). In short, serious games require serious software engineering skills to complete them on time and within budget.

Service Learning

As a teaching methodology, service-learning falls under the philosophy of experiential education. Service-learning programs are distinguished from other approaches to experiential education by their intention to benefit the client and the recipient of the service equally, as well as to ensure equal focus on both the service being provided and the learning that is occurring. More specifically, it often integrates meaningful community service with instruction and reflection to enrich the learning experience, teach civic responsibility, encourage lifelong civic engagement, and strengthen communities for the common good (Furco, 1996).

We believe it is important to foster a sense of community involvement among the computing students on our campus and provide them with a sense of purpose for their work. An important goal with some educators is to foster a sense of professional activism within their students. While Decker (Decker, 2007) claims that it remains unclear how this mentality can be reliably created, he does cite optimistic data that suggests that even during high school, students are predisposed to performing community service.

Martin reports a study in which students were forced to participate in what she referred to as social impact analysis, as one means creating an attitude of professional activism. By becoming intimately aware of the technology used in a real-life organization, both the client and the student were forced to consider the social and ethical components of the technology the client was using. This led the students to provide the client with ideas for alternatives or future upgrades to improve both productivity and social impact. This suggests that students can become much more interested in human aspects of the environment and much more in tune with social implications of the profession (Martin, 2007).

This immersion approach can be beneficial for instilling the social impact of one's work and a desire for activism. Tsang notes that most capstone design courses do not allow for serious reflection upon one's accomplishments in the context of community impact (Tsang, 2007). Tsang claims that if reflection were included, it would drive students to provide community service in the future. He does note also that service learning projects can be provided to community organizations with little or no cost to them with the risks being mitigated to, "Could this project be done by professionals?" and "What happens if the students don't get it done?"

In our approach, a community partner makes no financial contribution and sacrifices only his or her time to meet with the students. The students become very aware of the organization's structure and its needs as a result. Their work products are provided to the client free of charge, with virtually no risks to the client.

Through our project, we can assist in educating the public about community needs and resources. In our project, the game players gain a better understanding of the community partner's activities, with the hope that this will ease their apprehension of getting involved with the community itself.

COURSE DESCRIPTION

The computing capstone design experience at the University of Michigan-Dearborn is organized as two, two credit-hour courses (CIS 4961 and CIS 4962) which students complete over two consecutive semesters. These courses are required of all computing majors. Most students taking these courses complete projects for off campus clients. The capstone projects generally require about 500 hours of student effort to complete.

Students enroll in CIS 4961 after they complete all required software engineering courses. The major activities in CIS 4961 are requirements gathering and project planning (including risk management and quality assurance efforts). The major activities in CIS 4962 are product design, implementation, and testing. Many game projects usually make use of a rapid prototyping process, so a clear distinction between the analysis and design phases of a project may not exist.

Students work in three or four person teams. Students negotiate assignment to a particular project. We have found from observation that students work harder on projects they select, rather than those they are assigned to. Students select their own teammates and determine their own plan for rotating team leadership. We have observed that students tend to organize themselves so that one person is the hardware expert, one person is the software expert, and a third team member takes charge of documentation and coordination of activities. The asset creators similarly tend to specialize in the areas of 2D art creation, 3D animation, and audio design.

Classes meet for two hours each week for 56 semester contact hours over a period of 8 months. The ACM/IEEE Computing Curricula 2004 recommendations suggest that 11 lecture hours be devoted to social, ethical and professional issues. We include this material in our capstone design experience. Students select papers from a textbook containing a collection of

contemporary papers on computing ethics and professional issues and lead discussions on their selected papers (Spinelo and Tavini, 2004).

During the first two weeks of CIS 4961, class time is devoted to course introduction and project organization issues. During this time potential client projects are investigated by the students for feasibility. Most clients contact the instructors prior to the start of the 8 month project cycle. About 30% of our clients have worked with our students on previous projects. Many of our clients have read about our projects in either print or on line media. Some of our clients are the employers of current or former UM-Dearborn students. Since we have been doing these projects for 15 years, some of our clients are former graduates from our department.

After project teams assemble, class meetings consist of seminar-type class discussions on professional issues or team presentations of significant project milestone artifacts. These presentations might consist of a brief progress report, a structured walk-through of a work product, or a product demonstration.

In addition to the two hours of class-time each week, students put in an additional six hours per week out-of-class on their project. The out-of-class time in the capstone course consists of team interaction, project planning, software design, product implementation, presentation preparation, report writing, meeting with clients, and consultation with the instructor. The six hours of outside work is very important as a means of fostering team development.

The role of the instructor in our course is that of a coach or mentor not a project manager. The students handle routine client contact. Project scheduling and progress tracking is also handled by the student teams. The instructor is available to help student teams resolve unusual problems with the project and the client. The instructor provides feedback on the milestone

documents and presentations. Students revise their milestone documents based on the feedback from the instructor and their classmates following the presentation of their documents.

A final presentation is required of all teams at the end of the second semester and includes a product demonstration and report. The final project presentation is very important as a vehicle for assessing oral communication skills.

At the end of each semester the student teams reflect on their experiences. This involves producing both individual and team post mortem documents. These one page documents ask students to summarize what went right, what went wrong, and what lessons were learned.

Students must present a letter of acceptance from their client to the instructor in order to receive a grade for CIS 4962. The use of the client acceptance letter is a very important element of our course to drive home to students the importance of satisfying their clients' needs.

PROJECTS

Several student teams from the University of Michigan-Dearborn have created serious games for their capstone design project. About one out of every five projects is concerned with game development when Dr. Maxim teaches this course. The descriptions of three of these projects appear in this section of the paper. The projects may be accessed from the University of Michigan-Dearborn GAME Lab web site (GAME Lab, 2010). It is interesting to note that in each case, students were motivated to continue work on these projects for several months after receiving the client acceptance letter and final course grade.

Street Smart Detroit

The title of one serious game project is Street Smart Detroit. This project involved the development of a single level for a first person action game using the Torque 3D Game Engine. The main premise of the game level is to allow a player to experience the life of a homeless person trying to survive one night on the streets of Detroit. The player interacts with several non-player characters and attempts to find food, warmth, personal identification documents, and shelter to win the game. The player's challenge is to see if he or she is smart enough and tough enough to live (and eventually get off) the streets of Detroit. The client for this game is a project manager from City Connect Detroit.

The educational goal for this project is to design game play to raise the player's awareness of the problem of homelessness. Through the medium of a game, the player views the issues of homelessness through the eyes of a homeless person. The game includes some facts about homeless people with the goal of educating the player. The client's expectation for the game is to change player attitudes and make people see the homeless as human beings.

Examination of the student post mortem documents from this project provides some insight into what students learned from this experience. It should be noted that the Torque environment was new to this group of students.

- Art assets need to be developed first.
- Managing a large repository of game assets as well as program code presented the team with new project management experiences.
- More time should be spent learning their game development environments.
- Strong team leadership is needed even in teams using democratic team structures
- Solid team work building on each member's strengths is needed to complete a successful project on time.
- Source code version control tools need to be deployed early in the project and used diligently.
- Important to keep the game requirements up to date during project to ensure smooth negotiation of the client acceptance letter.
- Obtain early (and frequent) feedback from client on all game prototypes.
- Do not procrastinate and do work between semesters

These students (N = 3) added comments on working with their client to their postmortems. One of the software engineering challenges in this project involves ensuring that the delivered game runs on the relatively modest hardware specifications of client's laptop. The students were able to locate and negotiate the voluntary participation of content producers (artists and writers) because this project was developed for a non-profit group. Trying to keep the client from expanding the scope of the project with each delivered prototype was also a challenge. The

developers also needed to keep in check their desires to add fun features to the game when they came in conflict with the client's desires to have an accurate portrayal of life on the streets. These students indicated that they gained a greater appreciation for the challenges faced by homeless persons.

Campus of Hope

A second project involved the creation several interactive learning simulations (or mini-games) in Second Life to educate users on how food is collected and flows through the food bank network to reach people in need. One game allows players to drive a truck among food pick up points using the shortest route possible. A second allows users to experience the canned good sorting process that follows food collection. This helps drive home a message regarding the effort required to process donations. The third game allows players to manage efficient food distribution among several regions so that no region runs out of food. These games were developed for a larger project called the Campus of Hope.

Examination of the post mortem documents from this projects provide some insight into what students learned from this experience. The Second Life environment was new to this group of students and they had a change of client early in the project.

- More effort should be spent learning a new game development environment early.
- There are times when it is better for individuals to work by themselves rather than waiting for a time when they can time can all meet to solve a problem.
- When it is difficult to meet as a group, this delayed the implementation of software quality assurance activities (such as defect tracking) rendering them less effective.
- Good communication within the team and with the client is essential.

- Make sure client is committed and available as the project is started.
- It is difficult to track progress when working with a group of developers.
- Project would have benefitted from more user-testing earlier in the project.
- Be more proactive and organized with documentation as the project proceeds.
- Time management can be a huge problem when several people are interacting with an external client.
- Be proactive about schedule slippage; take action as soon as warning signs appear.
- Update your project schedule as the project proceeds.
- Do not procrastinate and do work between semesters.

This information was supplemented with survey data from students (N = 9) working on the Campus of Hope project. In aligning with the goal of fostering community involvement, students working on the Second Life mini-games claimed they learned a great deal about their community partner, not only through the interpersonal interaction with their client but also through the task of creating a product to meet the needs of the community partner in a virtual environment. This seems to have fostered a greater understanding of general practices and protocols of food banks such as sorting, delivery, and distribution. Although none of them claimed to have any prior experience with community involvement, they suggested that they would be more willing to get involved in the near future as a result of this project.

Virtual Shopper

A third student team developed a 3D game using the Torque game engine to provide players with a virtual shopping experience. The educational goal of this game is to provide

practice in selecting economical and nutritious food while staying within a limited food budget. This game was also developed as part of the Campus of Hope project.

The game is oriented towards players who are 18 years or older and also eligible for bridge card/welfare status. The primary game play involves moving throughout the store, viewing items, picking up chosen items, and picking up some starting coupons, as well as interacting with characters in the store. When players select an item, nutrition information is given to them, helping them to determine whether the selected item should be purchased or placed back on the shelf. The coupons the player receives will be generated at random from the list of available coupons. These coupons give players better prices on some items as well as provide a more realistic shopping experience, especially for those on a very tight budget. Players are successful if they can purchase the necessary food items with the money they have with them. Players rely on information gained from completing a six week nutrition course offered by the Gleaners of Detroit.

Examination of the post mortem documents from this project provides some insight into what students learned. These students had some experience with the Torque environment prior to beginning this project.

- An identifiable team leader is needed.
- Formal source code version control tools need to be used, not Google Groups.
- Important to keep the game requirements up to date during project to ensure smooth negotiation of the client acceptance letter.
- Start early even on things that seems to be simple.
- Begin implementation during the first semester and work during the semester break.
- Keep team mates updated on all work completed, have mandatory weekly status meetings

- Trust your team mates to deliver what they promise and accept the result.
- Write down plans and commitments to avoid misunderstandings.
- Function creep as project proceeds must tracked.
- Duplication of texturing work needs to be prevented, better communication is needed.

Supplemental survey data collected from the Virtual Shopper students (N = 4) suggested that they learned a lot about the community partner's Operation Frontline program and gained a better understanding of how to balance nutrition with a budget. Some of these students did have previous community involvement and, overall, they do feel more compelled to get involved in their community in the future.

SUMMARY

Table 1 contains the average scores for course evaluation items completed by all students for three offerings of the senior design course (4 = completely agree, 0= completely disagree) and the overall course rating (4=highest, 0=lowest). The data shown in this table involves students working on serious games and more traditional software development projects. The evaluation forms for each course indicate above average levels of satisfaction on the part of the students. The average for most student evaluation items in the College of Engineering and Computer Science is 3.0 or less.

Table 1

Student Course Evaluations for CIS 4962

	Fall 2007	Win 2009	Win 2010
Course fulfilled my needs	3.8	3.5	3.7
Course objectives were clear	3.8	3.6	3.8
Course prerequisites are adequate	3.6	3.7	3.6
Course was challenging and interesting	3.8	3.3	3.8
Overall course rating	3.8	3.2	3.5

The number of serious game projects that our senior design students have completed for non-profit groups is too small in number to allow for meaningful statistical analysis. Students submit course evaluation forms anonymously so it is not possible to separate them by project type. It is the authors' observation that students are willing to spend many hours engaged in both

the thoughtful design and careful implementation of components for serious game projects. It is interesting to note that students working on serious game projects at our institution have often continued working on them for several months after delivering them to their clients. The author has supervised over two hundred traditional senior design projects and it is extremely rare when students elect to continue working on these projects after receiving their client acceptance letter.

The number of requests for this type of senior design project is increasing as more people gain knowledge of our game design courses and research activities. Serious game requests seem to come from clients who have specific agendas for the use of the games within their organizations. These clients often have very firm delivery deadlines and computing resource constraints. The students seem to appreciate this and have often learned a great deal about their clients and the services they provide to their stakeholders.

We hope to involve students in working with more non-profit and faith-based clients in the future. As we do, we will undertake more systematic evaluation of the types of benefits students gain though participation in the development of serious games for these clients. A challenge to completing projects for non-profit clients is providing support for the projects after their final deployment. As students graduate their knowledge often leaves with them.

References

Blow, J. (2004) “Game development is harder than you think”, *Queue* (Vol. 1, No. 10, Feb 2004), ACM Press, NY, pp.29-37.

Decker, R. (2007) "Professional Activism: Reconnecting Community, Campus, and Alumni Through Acts of Service." In Tsang (ed.). *.). Projects That Matter: Concepts and Models for Service-Learning in Engineering*, Virginia: Stylus Publishing, VA., pp. 53-57.

Furco, A. (1996) “Service-Learning: A Balanced Approach to Experiential Education.” In *Expanding Boundaries: Serving and Learning.* Washington, DC: Corporation for National Service, 1996, 2-6.

GAME Lab, (2010) “University of Michigan-Dearborn GAME Lab”, http://gamelab.cis.umd.umich.edu/ (retrieved September 4, 2010)

Martin, C. D. (2007) "Integrating Service-Learning Into Computer Science Through a Social Impact Analysis." In Tsang (ed*.). Projects That Matter: Concepts and Models for Service-Learning in Engineering*, Virginia: Stylus Publishing, VA., pp. 99-107.

Maxim, B. R. and Akingbehin, K. (2006) “Experiences in Teaching Senior Design Using Real-World Clients”*, Proceedings of the 2006 IEEE Frontiers in Education Conference* (vol. 1, 2006), San Jose, CA, October 2006: T2H13-T2h17.

Michael, D. and Chen, S. (2006) *Serious Games: Games that Educate, Train, and Inform*, Thomson Course Technology, Indianapolis, IN.

Spinello, R. and Tavini, T. (2004) *Readings in Cyber Ethics*, Jones and Bartlett, Sudbury, MA.

Tsang, E. (2007) "Service-Learning as Pedagogy for Engineering: Concerns and Challenges." In Tsang (ed.). *Projects That Matter: Concepts and Models for Service-Learning in Engineering*, Virginia: Stylus Publishing, VA., pp. 27-30.

Author Note

Bruce R. Maxim and Matthew Sable, Department of Computer and Information Science, University of Michigan-Dearborn; Margaret Turton, Holmes Institute, Victoria, Australia, John Cristiano, Henry Patton Center for Engineering Education and Practice, University of Michigan-Dearborn.

This research was supported in part by grants from the Ford Motor Company Fund and the Henry Patton Center for Engineering Education and Practice.

Correspondence concerning this article should be addressed to Bruce R. Maxim, Department of Department of Computer and Information Science, University of Michigan-Dearborn, Dearborn, MI 48167. E-mail: bmaxim@umich.edu.