

Integration of Mobile Game Development into Introductory CS Courses: Lessons Learned

Stan Kurkovsky

Department of Computer Science
Central Connecticut State University
New Britain, CT 06050

Abstract—Computer games are widely used to improve student motivation and engagement in Computer Science courses. The process of design and implementation of a playable game, however, is a challenging task and is best accomplished in upper-level courses. Mobile games, on the other hand, are simpler and, thus, easier to program for students in lower-level courses. By exposing these students to a breadth of advanced topics in computing, we can also demonstrate that CS reaches far beyond coding. This paper describes our evaluation of several learning modules designed for introductory CS courses that use mobile game development as a learning context aimed to boost student motivation and engagement with the course material.

Index Terms—mobile game development, introductory courses, motivation, engagement

I. INTRODUCTION

From single courses to entire degree programs, computer games have been used as a mechanism for recruitment and improving retention by leveraging students' enthusiasm about games [9]. However, creating a playable game in the framework of an academic course is a demanding task, especially for the students in introductory courses. This task can be better accomplished by students in advanced CS courses who have more experience in software development and exposure to other CS topics. At the same time, mobile games typically have simpler playing scenarios and are often easier to program, which makes it more feasible for students in introductory courses to develop playable games [4]. Students often find it easier to relate to mobile technology and applications since they play an increasingly important role in their lives. We believe that using mobile game development as a motivational learning context in introductory computing courses could improve student chances for success and boost student motivation to continue their studies of CS [7]. Using mobile game development as a learning context in introductory courses can provide a simple and elegant way to motivate students and communicate the diversity and power of many advanced CS areas in a manner that engages students in experiential education [8]. Furthermore, research literature indicates that participatory learning methods such as those used in mobile game development can level the playing field for different types of students [10]. Current research and our own experience show that most CS students seem to be very interested in computer game development, and introducing students to this topic early in the curriculum could serve as a good tool to increase student retention [1]. Additionally,

current research in CS education shows that students tend to perform better when they perceive their course material as relevant to their everyday experiences [2].

II. DEVELOPMENT OF LEARNING MODULES

With the supported from an NSF TUES/CCLI grant, we are creating a set of several curricular modules consisting of laboratory projects and related instructional materials for introductory computing courses in Java. Every module focuses on a single mobile game and aims to introduce students to a single advanced topic, e.g. artificial intelligence, algorithms, computer networking, databases, software engineering computer security, or human-computer interaction. By showing students these and other non-programming and diverse aspects of computing early in the curriculum, this approach may help dissolve a widely popular misconception that “CS is all about coding” [1]. Each module is also designed to offer students a chance to practice at least one fundamental topic, such as classes, arrays, or loops.

All learning modules developed in this project have the same structure illustrated in the sample module on computer networking described below.

Learning objectives: Introduce students to the concepts of computer networking; gain hands-on experience with implementing a communication protocol; apply object-oriented programming techniques in a practical context.

Learning outcomes: Upon a successful completion of this project the students will be able to: a) understand the basic concepts of computer networking; b) implement a simple two-way communication protocol; c) develop software that incorporates inheritance, interfaces, exceptions, event-driven graphical user interfaces.

Problem statement: The game Battleship is a guessing game played by two opponents, which has been played since early 1900's throughout the world as a pencil and paper game. In a typical game, each opponent uses a 10x10 board to place a number of ships of different configurations; each player has the same number and configuration of the ships, but is not allowed to see the opponent's board. Upon positioning of the ships, the game is played as a series of turns where each opponent announces the target square to be shot at. The game proceeds until all of the ships of one opponent have been sunk. The objective of this project is to complete an implementation of

this game for mobile devices so that two opponents can play against each other over a wireless network connection.

Programming project: A mobile implementation of this game is provided where opponents play the game on their Java-enabled mobile devices. A basic infrastructure for sending and receiving messages between the two devices is in place. Students are required to complete the project by developing a communication protocol for messages exchanged by the devices to inform each other about the game turns (the target square on the board) and their outcomes (hit or miss). To successfully complete this project, students do not need to know the details of communication infrastructure provided in this project. The code scaffolding provided to the students also handles all details relevant to maintaining the state of the game between the two devices (e.g. who's turn is next). This project is ideally suited for students working in teams of two, who first can develop and test out the protocol of a simple message/response exchange using a paper-based game, then work out the data format of the messages and responses, how they will be analyzed, and what decisions will need to be made by the game logic.

Demonstration game and sample solutions: Instructors are provided with a complete working implementation of the game. Running it will demonstrate to the students how the game is intended to work without revealing the specifics of the project's solution. Students are provided with a clear interface exposing the methods for message passing and making game-play decisions so that they can focus on the learning objectives and not on the inner workings of the rest of the implementation. However, since the source code with low-level technical details of message passing using Java ME sockets is available, highly motivated students have an avenue for expanded learning opportunities in computer networking. Sample solution available to the instructors allow them to assess the amount of time and effort needed by their students to complete the project and may also help them in answering student questions.

Lecture notes: Instructors are provided with a set of lecture notes introducing computer networking at a level accessible to college freshmen. These materials emphasize the main concepts of computer communications (such as types of networks, protocols, networking devices, latency, etc.) and use examples from the Battleship game and other existing computer games to illustrate these concepts. Lecture notes contain a detailed description of the game and outline the strategies for developing a successful communication protocol between the opponents' devices. These notes also contain a range of additional introductory material that goes beyond the scope of the topics related to the Battleship game (such as network protocol layers, network management, network security) in case if the instructor should decide to include them in pre- or post-project lectures and discussions.

Taking a step further: Students and instructors wishing to explore this project further are offered a number of directions to build upon this project. For example, students could implement a capability to chat between players by adding extra features to the communication protocol and extending the game logic.

III. EVALUATION OF OUR APPROACH

Several of the currently developed modules have been tested at Central Connecticut State University (CCSU) and Rose-Hulman Institute of Technology (RHIT). In testing these modules, we employed three assessment strategies: 1) student course grades and course completion, 2) the Classroom Survey of Student Engagement (CLASSE) administered at the end of the term, and 3) the Motivated Strategies for Learning Questionnaire (MSLQ) administered at the beginning and end of the academic term. The modules were used relatively late in the courses, during the last few weeks of the term.

In the first year of the project period, a total of 161 students enrolled across eight sections of introductory computing courses at CCSU and RHIT that were in the scope of this study, although the total number of students who completed each assessment varied. A total of 84 students were enrolled in a test section in which one or more curricular modules using mobile game development were used to convey basic concepts, and 77 students were enrolled in paired control sections taught by the same faculty member in which mobile game development was not used as a curricular feature. Not all students completed the course, however, nor did all of those students who completed courses participate in the both administrations of MSLQ and the one administration of CLASSE. The course population varied between the two institutions: RHIT course was open only to computing and engineering majors, while CCSU course was open to students of any major, as long as they met the course prerequisites.

According to the interim report [6], results from the first year of the assessment of the project yielded findings in various areas of interest. We highlight a few of these areas.

A. Student Persistence

Persistence rates appear independent of administration of mobile game development curricular modules but were significantly different between the two institutions, with students at CCSU persisting at lower rates. Because mobile game development curricular modules were administered toward the ends of academic terms, there would have been limited opportunity for them to affect student persistence. The differences between universities in persistence rates very likely reflect the selectivity of the institutions, composition of the class population, and academic inputs of students taking the courses, with students at the less selective CCSU exhibiting lower persistence rates and lower levels of academic performances.

B. Course grades

Student course grades were independent of the use of the curricular modules. This suggests that the mobile game development modules as administered at both institutions did not affect student performance on the assessments used by these instructors.

Direct measurement of student learning during the first year of the project relied upon course exams and course grades in these introductory courses, and no meaningful differences were observed between test and control groups. Differences between students at the two institutions were pronounced, with CCSU students in all earning an average course grade of 2.56, more

than a full letter grade below students in all sections at RHIT who earned an average course grade of 3.63. Further, no RHIT students earned course grades below C, and all finished the course, compared with 22 CCSU students (29%) who earned course grades below C (2.0) or withdrew from the course (six of these students earned grades of C- which is considered passing at CCSU with 1.7 grade points, but RHIT does not use the minus designation for letter grades). Many students did not complete one or more assessments and the performance differential between test and control groups is not meaningfully different.

C. Student ratings of instructor and course quality

Patterns of student ratings of instructor quality and course quality varied by institution. Among RHIT students, ratings of instructor quality and course quality as measured on CLASSE at the end of the course were lower for test sections than for control sections at statistically significant levels. No significant differences were observed among test and control sections at CCSU. Perceptions of course quality exhibited similar patterns, although CCSU results for the test group tilted slightly more positive, albeit at a level that did not rise to statistical significance.

Absent other pronounced differences between test and control groups at RHIT, it seems plausible that the implementation of the mobile game development modules in test sections prompted this difference in the perception of effective teaching. Causes for this may have stemmed from the integration of a new and perhaps unusual curricular feature into a seamless course based on other pedagogies. It is possible that additional course planning, increased familiarity with the game development modules, or modifications to instructor materials could prompt more even ratings. It is also possible that some teaching or learning styles may better lend themselves to use of the mobile game development modules than others. CCSU results may be more dispersed across this scale than the RHIT results because of heterogeneity among the student populations, although all of these explanations remain speculative.

D. Student engagement with course

Students in test groups reported differences in terms of course engagement in areas of note taking, making connections across courses, course difficulty, and interest in course material. Some of these differences suggested mobile game development modules were associated with positive effects, while in other areas, for RHIT students only, findings suggested these modules may have had a negative effects in the areas of instructor comfort and communication as well as in interest in course material.

Note taking. Students in test sections were more likely to take notes than those in control sections. While the number of respondents at each institution was too low on this assessment instrument to yield statistically significant results at the campus level, this difference between test and control groups on this item was 24% among CCSU students and 12% among RHIT students. Reasons for this difference are not immediately clear, but they may involve the need to pay closer attention during demonstrations of mobile game development curricular units or to write down instructions during these classes.

Putting together ideas or concepts from different courses. Students in test groups at both institutions reported putting together ideas or concepts from different courses when completing assignments or during class discussions more frequently than did students in control groups. No students in either test or control groups at CCSU reported putting together ideas or concepts from different courses when completing assignments or during class discussion, but 26% of students in the test group reported making such connections 3 to 5 times compared to 13% of those in the control group. Students at RHIT made such connections much more regularly, with 59% of students in the test group reporting putting together ideas or concepts from different courses when completing assignments or during class discussion 3 to 5 or 5 or more times, compared to 47% of students in the control group who made such connections. It is possible again that RHIT students, who enter the institution with on average stronger academic backgrounds are accustomed to such habits more so than students at CCSU.

Perceptions of course difficulty. While no differences between test and control groups at RHIT were observed in terms of how students perceived the difficulty of course material, among CCSU students, those in test sections reported that course material was less difficult than did students in control sections. While 35% of students in the control group at CCSU reported the course material was very difficult, just 11% of students in the test group reported a similar level of difficulty. It is important to note, however that respondents in the control groups were distributed bi-modally with a large cluster (39%) reporting the course material was just somewhat difficult, while 17% reported it was difficult. By contrast, less than 10% of RHIT students in either the test group or the control group reported the course material was very difficult. Again this difference may be related to the higher selectivity of RHIT as an institution and students who may be better prepared to negotiate college-level work in a CS course.

Interest in course material. RHIT students in the test group reported lower levels of interest in course material than did students in the control group, although these differences were significant only at a borderline level. These proportions were reversed among CCSU students, but the lower number of respondents to CLASSE at CCSU did not allow these differences to rise to a generally accepted level of statistical significance. While 78% of RHIT students in the control group reported they were very interested in course material at the end of the term, only 59% of students in the test group reported the same high levels of interest. By contrast, among students at CCSU, 30% of students in the control group reported they were very interested in course material, while 52% of students in the test group reported this same high level of interest. Of further note, 18% of students in the test group and 26% of students in the control group at CCSU reported they were either uninterested or very uninterested in course material, while just one student among the RHIT test and control groups (about 1.5% of respondents) reported being uninterested in course material, suggesting again that RHIT students may have been predisposed to be more engaged in introductory computer science concepts that students at CCSU.

Taken as a whole, findings about student engagement in the course present a mixed bag of results that suggest limited

positive effects of using the mobile game development curricular modules in increased note taking, making connections across courses, and for the students at CCSU, whose academic preparation for college was not as strong as those at RHIT, perceptions of course difficulty decreased. Use of the mobile game development curricular modules with the better prepared RHIT students, however, exhibited some negative effects in the area of instructor comfort and communication as well as student interest in the material. It is possible that these negative findings were a result of students who expected a more traditional classroom experience with these concepts or that the curricular modules did not fit seamlessly into the rest of the course as taught by RHIT instructors.

E. Student motivation

In general motivation among students in CCSU computer science classes declined, but this decline was less pronounced in test sections than control sections. By contrast, RHIT student motivation stayed level or increased slightly over the course of the course of the term for both the test group and the control group, with motivation gains among the control group slightly outpacing those in the test group.

Student motivation was measured using the short (31-question) version of MSLQ; this instrument was administered in the first two weeks of the term and again at the end of the term. Results were analyzed only for students who participated in both administrations of the instrument. In general, student motivation among CCSU students declined, but this decline was less pronounced in test sections than control sections. At RHIT, however, student motivation stayed level or increased slightly during the term for both the test group and the control group, with motivation gains among the control group slightly outpacing those in the test group. These changes on MSLQ indices amounted to differences of as much as half a standard deviation between test and control groups, but they did not rise to the level of statistical significance.

IV. SUMMARY AND FUTURE WORK

Preliminary results described here suggest that in each of the areas highlighted above, the test groups did at least as good as the control groups. In fact, students in the test groups showed higher levels of engagement with the course material in which the modules were presented and their motivation did not decline as much as in the control groups. We will continue to refine the modules and adjust the points in the courses in which the mobile game development projects are offered to the students. Also, giving students more time to complete the work related to these curricular modules and to apply what they learn to other topics may produce more positive results.

As we pointed out in our current evaluation, there is still a substantial room for improvement in this project. We believe that incorporating mobile game development projects into the introductory computing curriculum will promote teaching and learning, help improve student experience in the critical introductory courses in computing when student attrition is at its highest, and ultimately will help train outstanding computer scientists. Currently, we are finalizing our mobile game development modules to address the concerns raised as the result of the ongoing project assessment. We shared one of the modules at the 2011 ACM Symposium on Computer Science Education (SIGCSE 2011) and at several conferences of the Consortium for Computing Sciences in Colleges (CCSC) [5] and it was very well received.

ACKNOWLEDGMENT

This work was supported in part by NSF awards DUE-0941348 and DUE-0941658. Dr. Delvin Defoe works on this project at Rose-Hulman Institute of Technology.

REFERENCES

- [1] T. Barnes, H. Richter, E. Powell, A. Chaffin and A. Godwin, "Game2Learn: building CS1 learning games for retention," SIGCSE Bull. 39, 3 (Jun. 2007), 121-125.
- [2] T. Beaubouef and J. Mason, "Why the high attrition rate for computer science students: some thoughts and observations," SIGCSE Bull. 37, 2 (Jun. 2005), 103-106.
- [3] M. Biggers, A. Brauer, and T. Yilmaz, "Student perceptions of computer science: a retention study comparing graduating seniors vs. CS leavers," Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education, pp. 402-406, 2008.
- [4] B. Burd, J. Goulden, B. Ladd, M Rogers and K. Stewart, "Computer games in the classroom, or, how to get perfect attendance, even at 8 am," SIGCSE Bull. 39, 1 (Mar. 2007), 496-496.
- [5] D. Defoe, S. Kurkovsky and E. Graetz, "Short mobile game development projects for introductory CS courses: conference workshop," Journal of Computing Sciences in Colleges, vol. 26, no. 5, pp. 141-143, May 2011.
- [6] B. J. Hosch, "Mid-point assessment report for NSF CCLI grant: Using mobile game development to improve student learning and satisfaction in introductory computer science courses," Aug 2011.
- [7] S. Kurkovsky, "Engaging students through mobile game development," In Proceedings of The 2009 Technical Symposium on Computer Science Education (SIGCSE 2009), Chattanooga, TN, March 4-8, 2009.
- [8] S. Kurkovsky, "Making the case for mobile game development," In Proceedings of The 14th ACM-SIGCSE Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE 2009), Paris, France, July 6-8, 2009.
- [9] S. Leutenegger and J. Edgington, "A games first approach to teaching introductory programming," SIGCSE Bull. 39, 1 (Mar. 2007), 115-118.
- [10] M. A. Lundeberg and S. Diemert, "Influence of social interaction on cognition: connected learning in science," Journal of Higher Education, 66(3), 312-335, 1995.