

The Impact of Game Design on Students' Interest in CS

Yolanda Rankin
Northwestern University
Evanston, IL 60202, USA
847-467-5635

yrankin@northwestern.edu

Amy Gooch
University of Victoria
Victoria, B C, Canada
801-414-3498

Amy.a.gooch@gmail.com

Bruce Gooch
University of Victoria
Victoria, B C, Canada
250-472-5758

brucegooch@gmail.com

ABSTRACT

We examine the effect of game design on students' attitudes, specifically interest in attaining a Computer Science degree, continued development of programming skills and experience in game design. Students in a Computer Science Survey course are given the task of applying software engineering principles in the context of game design. Using the Game Maker platform, students are divided into teams and participate in collaborative game design for 2.5 weeks. Pre and post survey results reveal that game design can have both a positive and negative impact on students' attitudes about Computer Science, game design and further development of programming skills. Furthermore, it is imperative that assignments are carefully planned and cover appropriate material in respect to time constraints if the goal is to positively influence students' attitudes and lead to positive learning outcomes.

Categories and Subject Descriptors

K.3 [Computers in Education]: Software Engineering for game design.

General Terms

Computer Science Education, Software Engineering.

Keywords

Game Design, Computer Science Education, Software Engineering

1. INTRODUCTION

Game design has become increasingly popular in computer science curriculum as academia responds to the decreasing number of students declaring computer science as their major [6], [8]. Much research has investigated the integration of game design into

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GDCSE Conference '08, Feb. 28 – March 3, 2008, Cozumel, Mexico.

Copyright 2008 ACM 978-1-60558-057-9/08/02...\$5.00..

computer science curriculum [1], [9] – [13], [19], [20]. One of the underlying reasons for incorporating game design into the curriculum is to formulate common ground with young people who play games for recreational purposes. For example, 84% of children ages 8 – 17 years old own at least one video game and play them for a minimum of one hour per day [13]. Thus, the argument has been that college students represent a segment of the population that grew up playing video games, and this affinity for video games can be leveraged to increase student interest in computer science [6].

Previous research explores pedagogical strategies that apply computer science concepts in the context of game design. Parberry et al. [9] - [11] teach capstone courses that require students to apply their comprehensive knowledge of computer science concepts to the game design process. Bayliss [1] demonstrates game development as a tool for encouraging incoming freshmen to enroll in computer science courses and become computer science majors. Youngblood [20] proposes game segments as an effective pedagogical tool for advanced students to master design principles as they modify code templates. Wolz et al. [19] share their approach to teaching game design across disciplines, including Art, Communications, Computer Science, Interactive Media, Music and Writing. More articles exist that outline pedagogical strategies for incorporating game design for the purpose of enhancing recruitment and retention of computer science majors [5], [12], [13]. Trytten [17] discusses the challenges of teaching introductory Computer Graphics with some game applications in a team-based laboratory setting. However, less has been said about the effects of game design on students' interests and attitude about computer science [1]. Do video games provide sufficient motivation for student engagement? Are there certain pedagogical strategies related to game design that increase or impede student interest in Computer Science? To what extent are students interested in game design? This article attempts to address these questions as we examine the effects of a game design assignment in a Computer Science (CS) Survey course.

2. GAME DESIGN STUDY

2.1 Participants

56 undergraduates enrolled in the Computer Science Survey class offered at the University of Victoria in British Columbia. 20 students self-selected to participate in the study. Fifteen students indicated that they were Computer Science (CS) majors including joint degree programs such as Computer Science/Geography. Three students identified themselves as non-CS majors and three students had not declared a major. Prior to the game design assignment, students completed a pre-game assignment survey that assessed attitudes and interest about Computer Science, programming and game design. In fulfillment of course requirements, students were given 2.5 weeks to complete the game design assignment. Students were divided into groups of 2-4 students and given the task to design and implement a 2D game using the *Game Maker* Platform developed by YoYo Games [7].

2.2 Waterfall Lifecycle & Game Design

Using the *Waterfall Lifecycle* model as a guide for software development, students applied software engineering principles to the context of game design [15]. The assignment correlated phases of game design to the *Waterfall Lifecycle* model. See table 1.

Table 1. Software Engineering in the Context of Game Design

Game Design Process	Waterfall Lifecycle	Project Deliverables
Conceptualization Phase	Requirements	Project Schedule
	Specifications	Specification Sketch
	Architectural Design	Initial Design Document
		Game State Diagram
Prototype Phase	Detailed Design	Executable file of Alpha version of game
	Implementation	Revised Design Document User Guide
Playtesting Phase	Testing & Debugging	Playtest Feedback Executable file of Beta version of the game
	Finished product	Final Design Document Final User Guide

2.3 Conceptualization

The conceptualization phase began the game design process as students discussed ideas about the kind of game they wanted to design [4]. The conceptualization phase gave students the opportunity to appreciate the formal (objectives, rules, resources) and dramatic elements (story, characters, challenge) of game design [4]. Each team submitted a specification sketch that explained the game concept, included mock-ups of interface, identified game controls, and described the flow of game-play [15]. Additionally, students were asked to develop a project schedule with clearly identified milestones. These milestones are equivalent to product deliverables (e.g. Design Document). Each team generated a game state diagram to visualize the architectural design. This stage corresponds to the *Requirements Gathering*, *Specification* and *Architectural Design* phases of the waterfall lifecycle model [15].

2.4 Prototype

We selected the *Game Maker* development platform, because it had a short learning curve, did not require prior programming experience, and supported rapid prototyping [7]. Team members worked together to determine the distribution of the workload for the *Detailed Design and Implementation* stages of the *Waterfall Lifecycle model*. Using the drag and drop interface of the *Game Maker* platform, 23 teams transformed their game ideas into an executable file, creating an Alpha version of the game. At a minimum, the Alpha versions had art work for sprites, defined game objects and system behaviors such as game structure, and executed basic events (e.g. start game) [4], [15].

2.5 Playtesting

Completion of the Alpha version allows students to engage in one of the most critical phases of game design---playtesting [4]. Students critiqued other teams' prototypes to generate user feedback. This feedback served as input into the iterative design process of evaluating requirements, specifications and architectural design [15]. Upon receiving feedback from other users about their game, conceptual changes were incorporated into the implementation phase to produce the Beta or final version of the game. Thus, students had the opportunity to experience the real-world process of iterative software development.

2.6 Final Presentation

Student groups gave a ten minute presentation of the conceptual design and implementation of their 2D game to their peers and a panel of reviewers. Students also reported significant problems and successes encountered during development, distribution of workload amongst teammates, and how much time they spent on developing the game. Prizes including \$10 gift certificates to the student union were awarded for the categories of

Best Presentation, Best Novel Game, Best Overall Game, and Best Game as Voted by Peers.

3. DATA ANALYSIS & RESULTS

3.1 Pre and Post Game Design Surveys

Though the class consisted of 56 undergraduates, only 20 students completed both the pre-game assignment and post-game assignment surveys. Both surveys asked students to rate their interest on a Likert scale from 1 to 7 with 1 indicating absolutely no interest at all and 7 indicating extreme interest for the following categories:

- enrollment in a game design course,
- pursuit of a CS degree,
- further development of programming skills,
- and enrollment in additional CS courses.

3.2 Interest in Game Design

Post-game survey results showed a 100% increase (from 3 to 6) for students who absolutely had no interest in game design and an overall 70% decreased interest in game design. Twenty percent of students who indicated decreased interest in game design dropped by 3 points on the Likert scale. Fifteen percent of students appeared to be unaffected by the game design assignment, indicating continued strong interest in learning how to design video games.

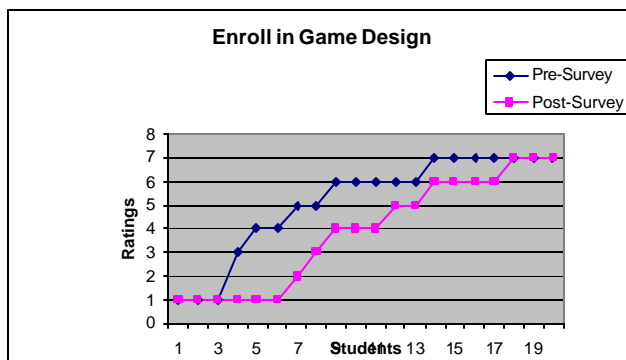


Figure 1. Student Reported Interest in Game Design

3.3 Interest in Attaining a CS Degree

Fifty percent of students were more interested in CS degree prior to game design assignment in contrast to the 30% of students who indicated growing interest in CS degree after completing the assignment. Twenty percent of students demonstrated same degree of interest in pursuing a CS degree after completing game design assignment. One Undecided Major dropped from 3 (not

much interest) to 1 (not interested at all) in terms of interest in CS degree. Sixty-seven percent of non-CS majors dropped from 6 (interested) to 4 (neutral) in terms of interest in CS degree. Seventy-three percent of CS majors indicated less interest in CS degree, dropping by as much as 3 points in some instances.

3.4 Development of Programming Skills

Overall 25% of participants indicated an increased interest in further developing their programming skills, and 40% demonstrated reduced interest. The game design assignment had no effect on the remaining 35% who completed the Post-Game Survey.

3.5 Enrollment in Additional CS Courses

Completion of the game design project had no impact on 65% of students' plans to definitely enroll in CS classes; this is probably due to large concentration of CS majors in the CS Survey course. However, 20% of students indicated less likely to enroll in CS classes. Only 15% of students indicated a positive change in regards to taking more CS classes. One CS major dropped from 7 (extremely interested) to 5 (somewhat interested); One CS major changed from being neutral to enrolling in additional CS courses and 2 CS majors indicated definite plans to enroll in more CS courses, demonstrating a 20% increased rating. Sixty-seven percent of non-CS majors indicated decreased likelihood in taking additional CS courses. One Undecided Major dropped from 7 (definitely will enroll) to 1 (definitely will not enroll), indicating negative response to the game design assignment; the other Undecided Major indicated no change in definite plans to take more CS courses.

4. DISCUSSION

4.1. Student Deliverables

As educators, we were thrilled by the student output for the Game Design Workshop. For example, the group for "Faith of Time" implemented inheritance in their game engine to make it easier to add new enemies for additional levels. The "Dodge Ball" group extended the game engine by implementing a real-time physics module. All of the teams created extensive websites and back stories using HTML which they were only taught the week before. All of the teams successfully created games that would have been marketable when arcade games were at their height. To view student work go to <http://www.csc.uvic.ca/~csc212/forum/viewforum.php?f=9>. We recorded anecdotal evidence from the final presentations that can be summed up in the final statement: novice programmers can not build a game equivalent to Blizzard's *World of Warcraft* in two weeks. Students appeared to have reached the conclusion that the reward-to-work ratio for game development is not one-to-one.

Perhaps this is not a misconception on their part, but points to the idea that the students may have been disappointed in their output.



Figure 2. Screen shot of game entitled Faith in Time.

4.2 Effects of Game Design on Students

We incorporated game design into the CS Survey course to determine if video games would positively influenced motivation and interest in the discipline of CS and game design. We discovered that for the majority of non-CS and Undecided majors, the workshop appeared to have a negative impact on their interest in pursuing a CS degree, taking additional game design courses and improving their programming skills. In comparison, CS majors indicated commitment to a CS degree even in the cases where they showed decrease interest in game design. In fact, some CS majors indicated that they were less interested in improving their programming ability despite the fact that *Game Maker* did not require extensive programming skills to create a video game. On the surface, this may suggest that game design does not provide sufficient motivation for recruitment and retention in CS. However, it points to the problem of not properly integrating the technical details of implementation with the practice of game design, placing equal emphasis on both. Students who did not possess prior programming experience became bogged down in the technical details of implementation and focused less on design issues. Students who possessed prior programming experience and expressed interest in game design excelled in the implementation phase and produced high quality games, but also failed to appreciate the significance of conceptualization. The solution is to allot equal time and emphasis to both the conceptualization phase and the implementation phases, identifying the direct correlation between conceptualization to the implementation and playtesting phases of game development.

4.3 Peer Review of Teammates

We used a team-based pedagogical approach to facilitate students developing negotiation skills and other soft skills necessary for team work [8]. Eleven teams evaluated their team members for individual contribution, ability to encourage others and ability to be flexible when resolving conflict. Sixty-four percent, 7 out of 11 teams, rated their teammates an overall average score of 8.7 or higher for contribution and overall average score of 8.5 or higher for encouragement on a scale from 1 to 10 with 1 being the lowest and 10 being the highest. In contrast, only 45% of team members thought their peers were flexible in terms of scheduling time to meet to work together and negotiating conflict. This suggests further participation in team-based projects can assist students with their negotiation skills.

4.4 Limitations of Study

In our enthusiasm we did not plan sufficiently for direct evaluation of the course beyond the pre- and post-game surveys. For future research purposes, we will designate a control group to better identify the contributing factors of game design for student motivation and interest. The pre- and post-game surveys will be administered to the control group in the same time interval designated for the game design assignment. This will enable us to draw conclusions about the effects of game design on student interest and motivation. Because the participation in the study was voluntary and students enrolled in the course were under no pressure to participate in the study, fewer than 50% of enrollees actually participated. Thus, we did not have an accurate picture of the true effects of the game design assignment on students' motivation and attitudes for the majority of the class. To ensure that we have an adequate number of participants that will support generalizations of the effects of game design, we will conduct this study with more CS Survey courses and increase our subject pool.



Figure 3. Screen shot of game Legend of the Link.

5. CONCLUSION

Students applied software engineering principles to game design: 1. Students used the waterfall model of software development and carefully documented the process of designing software. 2. Students met and often exceeded the learning goals for algorithms and software development models. 3. Students acquired skills in areas such as sound/music, artwork and animation. 6. Students became acquainted with version control, backing-up data, and merging code from individual programming and content creation efforts. Three of the groups did background research in software development and employed an extreme programming model for developing their games. Our survey results indicate that the assignment might be better received and increase student interest in CS and game design if the students were given more time and equal emphasis on conceptualization and implementation phases of game design. Students indicated that there is more to writing a 2D game than can be satisfactorily accomplished during a two week project. Overall, we found that students will work hard and exceed learning goals expressed in the guise of game development.

6. ACKNOWLEDGMENTS

We express our gratitude to the students in CSC212 at University of Victoria for selflessly giving us access to their ideas and feedback. We say thank you to Adam Parkin for assistance with the logistics for the game lab and Marcus Shute for assisting us with the data analysis.

7. REFERENCES

- [1] Bayliss, J. (2007). The Effects of Games in CS1 -CS3. *Journal of Game Development*. Vol. 2 Issue 2. 2007 pp. 7 – 17.
- [2] CARLESS S. 2005. Postcard from SGS 2005: Computer gaming to enhance computer science curriculum. <http://www.gamasutra.com/features/20051101/carless01.shtml> <http://www.gamasutra.com>
- [3] Elliot, J., Adams, L., and Bruckman, A. 2002. No Magic Bullet: 3D Video Games in Education. *In Proceedings of the International Conference of Learning Sciences (ICLS 2002)* Seattle, Washington, 2002.
- [4] Fullerton, F., Swain, C., and Hoffman, S. *Game Design Workshop: Designing, Prototyping, and Playtesting Games*. CMB Books, San Francisco, CA, 2004.
- [5] Guzdial, M. and Forte A. 2005. Design process for a non-majors computing course. Technical Symposium on Computer Science Education of the 36th SIGCSE Technical Symposium on Computer Science Education, St. Louis, MO, 2005. pp. 361 – 365.
- [6] Guzdial, M. and Soloway, E. 2002. Teaching the Nintendo generation to program: Preparing a new strategy for teaching introductory programming. *Communications of the ACM* Vol. 45, Issue (4) April 2002 pp. 17 – 21
- [7] Habgood, J. and Overmars, M. *The Game Maker's Apprentice: Game development for beginners*. Apress, Berkely, CA. 2006
- [8] Michaelson, L., Knight, A., and Fink, L. *Team-Based Learning: A Transformative use of small groups in college teaching*. Stylus Publishing, Sterling, VA. 2004
- [9] Parberry, I. 2006. Game Development in Computer Science Education: From outcast to mainstream. *Journal of Game Development*. Vol. 2 Issue 2. pp. 5 – 6.
- [10] Parberry I., Kazemzadeh M., and Roden T. 2006. The art and science of game programming. In *Proceedings of the 2006 ACM Technical Symposium on Computer Science Education*, Houston, TX (2006), pp. 510 – 514.
- [11] Parberry I., Roden T. and Kazemzadeh M. 2005. Experience with an industry-driven capstone course on game programming. In *Proceedings of the 2005 ACM Technical Symposium on Computer Science Education*, St. Louis, MO (2005), pp. 91 – 95.
- [12] Rankin, Y., Lechner, T., and Gooch, B. 2007. Team-based Pedagogy for Object-Oriented Game Design. In *Conference Proceedings SIGGRAPH 2007 Educator's Program*, Los Angeles, CA.
- [13] Rankin, Y., Gooch, B., and Gooch, A. 2007. *Interweaving Gaming into Core Curriculum*. In *Conference Proceedings of Microsoft Academic Days Game Development Conference 2007*, Nassau, Bahamas.
- [14] Roberts, D., Foehr, U., and Rideout, V. 2005. *Generation M: Media in the Lives of 8-18 Year-olds Report* Kaiser Family Foundation.
- [15] Rucker, R. 2003. *Software Engineering and Computer Games*. Addison-Wesley, Harlow, England.
- [16] Snyder, N. 2006. *Universities See a Sharp Drop in Computer Science Majors*, Tennessee.com September 2006.
- [17] Trytten, D. 2005. A Design for Team Peer Code Review. In *Proceedings of SIGCSE 2005*, February 23 – 27, 2005, St. Louis, MO. pp. 455 – 459.
- [18] Vegso, J. 2005. Interest in CS as a Major Drops Among Incoming Freshman, *Computing Research News*, May 2005 Vol. 17/No. 3. <http://www.cra.org/CRN/articles/may05/vesgo>
- [19] Wolz, U., Ault, C., and Nakra, T. M. 2006. Teaching Game Design through Cross-Disciplinary Content and Individualized Student Deliverables. *Journal of Game Development*. Vol. 2 Issue 2. pp. 19 - 31
- [20] Youngblood, G. M. 2006. Engaging Students in Advanced Computer Science Education Using Game Segments. *Journal of Game Development*. Vol. 2 Issue 2. 2006 pp. 33 – 45.