Towards Social Gaming Methods for Improving Game-based Computer Science Education

Andrew Hicks
University of North Carolina - Charlotte
9201 University City Blvd.
Charlotte, NC, 28213
1-304-210-1440
AHicks37@uncc.edu

ABSTRACT
Participation, retention, and overall performance in introductory CS students could be improved by removing or reducing some of the pitfalls along these students' paths. In introductory coursework, there is little or no need for collaboration, which may discourage socially-oriented students. Furthermore, the dual challenge of learning a programming language and covering more abstract, conceptual concepts may discourage students with little prior experience, and frustrate those who are already capable programmers. To remedy this, we will investigate social mechanics for learning games. We hypothesize that students will develop the skills necessary to succeed in the game more readily if friends are present to motivate and challenge them. We plan to explore effective social mechanics, and create efficient methodologies for their implementation and development. We believe that social gaming has the potential to revolutionize the way students learn, and that through our research, we will discover the most effective ways to bring students together and motivate creativity, innovation, and learning using social gaming.

Categories and Subject Descriptors
L.5.1 – Game-based Learning / Gaming

General Terms
Human Factors, Social Gaming, Games, Education, Motivation

Keywords
Serious Games, Social Networking, Education, CS1

1. INTRODUCTION
Creating a sense of community in an educational context by encouraging interaction between groups of students has been shown to significantly increase both retention rates and student success [13, 14]. Without community, people are likely to be "anxious, defensive, and unwilling to take the risks involved in learning" [15]. Previous research encouraging peer-to-peer interaction has been effective in fostering communities between groups of students. In these communities, students feel free to display their work, comment upon and rate the work of others, and motivate themselves to improve and create based on the achievements of their peers.

Additionally, there has been recent insight into the functional disparity between introductory computer science curriculum and high-level coursework. An undue level of focus on programming at the expense of computational thinking, problem solving, and mathematics may be one source of student frustration and confusion. Efforts made to teach important abstract conceptual information without overloading students with the specifics of programming met with success, both in introductory coursework, and in high school and middle school environments. Many of these approaches use simplified vocabularies to represent such programming concepts as control structures (loops, branches, and switch-case statements), methods, and basic object-oriented programming ideas.

Unlike any other medium, games have the potential to inspire creativity in an engaging, accessible environment. Players of popular online games develop a sense of camaraderie that exceeds that found between peers in a traditional classroom, and develop the skills necessary to succeed in the game more readily if friends are present to motivate and challenge them. Games also have the ability to communicate large amounts of information via simple, abstract visualizations while maintaining consistency with their virtual environments. These qualities of games show their amazing potential as educational tools. For this reason, we propose an investigation towards the use of community-building mechanics in games for education.

2. BACKGROUND
We believe that graphical approaches to CS material similar to those used in environments like Scratch are ideal for teaching introductory CS concepts because they avoid syntactical roadblocks while leaving the logic of the student's code completely intact. This presentation also lends itself well to visualizing the performance of algorithms like those taught in an introductory algorithms class. Additionally, using a graphical interface allows for striking visual feedback which previous studies have shown positively affects student attitudes and performance [3].

Our previous work on educational games for teaching computer science demonstrated that players enjoyed the approach of interacting with the world via code [5]. However, one issue that became apparent was the choice of language; the game was designed to accept code written in C#, but some students were unfamiliar with this language, having learned C++ or Java. While

FDG 2010, June 19-21, Monterey, CA, USA

this did not present any major difficulty in this case, as technologies and the computer science curriculum continue to develop, the gap between in-game code and in-class code is sure to widen.

Previous work on encouraging community development has been encouraging. Such efforts as the REU program at UNC Charlotte [6] and the Educational Game Design Class at University of Colorado Boulder [12] have demonstrated promising results and innovative techniques. These communities exist in the real world; however, we hypothesize that creating a community learning space in a shared virtual environment would help assuage any concerns about scheduling, location, and resources while providing benefits to learners. Our previous work towards developing an online social community, documented in [11] demonstrated that students are more likely to have meaningful interactions if encouraged to do so by game mechanics; these same motivations could help drive valuable learning interactions like sharing of ideas and critical evaluation of peers.

3. RESEARCH GOAL

The goal of this research is to investigate the effectiveness of community-driven, game-based educational tools. Through this, we hope to discover new methods for implementing effective social mechanics in games. We will develop metrics to enable us to measure and describe the social behavior and problem-solving aptitude of the players, and investigate the effects of various mechanics. We will implement our findings in a learning game, *BOTS;* an online, multiplayer game that allows players to engage in simple programming and problem-solving puzzles in an engaging visual environment. This environment, sharing elements of successful social games and websites like *Farmville, Fantastic Contraption,* and the online *Scratch* community, will encourage consistent social interaction and sharing of knowledge [7][8]. We hypothesize the following:

- **An educational game with social mechanics will achieve significant learning gains.**

Game based approaches have been successful at teaching the type of material we are focusing on. [4][9][10]
Our game should impart the same level of learning as previous efforts in this area.

- **In-game social interaction will increase student involvement in class material.**

Efforts to improve communication between students and encourage peer-to-peer interactions, such as undergraduate research experiences and pair programming groups, have shown effectiveness [1][6][11]. We hope to encourage the same types of interactions with our game, expanding game mechanics to include peer evaluation and creative collaboration.

- **The existence of a consistent metaphor between presented materials will improve the experience.**

Many game-based approaches towards teaching computer science exist, but most use very different metaphors for similar concepts [2][3]. This could cause confusion if multiple tools are used in one class; therefore, we will unify content under one metaphorical umbrella; our game world.

4. RESEARCH PLAN

BOTS will be developed as a game-based environment in which programming challenges can be easily deployed, presented, and evaluated while maintaining student engagement with the material through dynamic, interesting gameplay and omnipresent social interactions. We have created a proof-of-concept for the gameplay mechanic of BOTS currently in the initial phases of playtesting. In the prototype, players have access to a limited vocabulary of programming components to create and modify the programming "puzzles" presented in the game.. We created the prototype with a four-person team during a course on the creation and evaluation of serious games at UNC Charlotte, Fall 2009. Members of the original team are designing a study to evaluate various components of the prototype; learning gains, player enjoyment, and clarity of visualization.

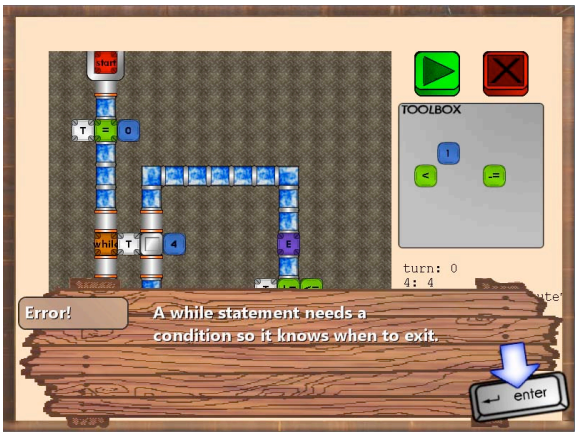

Figure 1 – Code elements and visual feedback

The next step of our proposed research is to develop an environment in which students can engage in social interaction. To implement this, we follow the "Flow of Inspiration Principles" outlined in [12]: Players will be able (and encouraged) to make their in-game creations public, to interact with and rate other students' solutions, and provide or request advice on difficult problems, similar to the Scratch community. [10]. We hypothesize that adding social functionality to the game will enhance student creativity by inspiring and rewarding the best and most interesting solutions to proposed problems, and encourage each player with feedback, encouragement, and an audience of observers.

We believe this project represents a novel combination of puzzle-based, role-playing, and social gaming mechanics with programming and computational thinking material. *BOTS* will allow students the freedom to create their own "machines" in game to share with friends, and the social mechanics will reward successful students and motivate progress for new players. The game environment will also encourage players to offer criticism, comments, and cooperation to each other, forming a community of peers which will serve to increase learning gains and foster a strong bond between players which, hopefully, will carry over into the real world.

5. PRELIMINARY RESULTS

Our previous work on Serious Games, [5], has shown the use of coding in-game to be effective, but noted that the use of varying programming languages could be a difficulty. Despite this, our

game achieved significant learning gains between pretest (M = 1.9355, SD = 1.1236) and posttest scores (M = 3.1250, SD = 1.8212, $t(15)$ = -3.048, p< .05). This effect was quite large (Cohen's d = .78).

Our work done on social mechanics in games, [11], has been met with positive results. Our social networking game, Snag'Em, was deployed at an academic conference (the STARS Celebration 2009 in Tallahassee), at a week-long event at Spellman College, and is currently in a semester-long, college-wide evaluation at the College of Computing and Informatics at UNC Charlotte. This game was met with great enthusiasm and broad participation, with 80 of 280 attendees registering for our game, half of these participating enough to accumulate a positive score.

6. DISCUSSION

The development of this project will result in a refined methodology for designing, refining, and evaluating player-to-player interaction in Serious Games, as well as methods for evaluating performance across a span of gameplay. This will have applications reaching far beyond serious games; investigation into the development of social mechanics has potential for improving any multiplayer gaming experience, and our proposed study on extended gameplay periods can help developers create a smoother, more even learning curve. Once we have fully investigated these realms, we will present design principles which will allow developers and educators to present material, regardless of domain, in a social context, and establish methods to measure the efficacy of these new games and tools. We believe that social gaming has the potential to revolutionize the way students learn, and that through our research, we will discover the most effective ways to bring students together and motivate creativity, innovation, and learning using social gaming.

7. REFERENCES

[1] Aragon, C. R., Poon, S. S., Monroy-Hernández, A., and Aragon, D. 2009. A tale of two online communities: fostering collaboration and creativity in scientists and children. In *Proceeding of the Seventh ACM Conference on Creativity and Cognition* (Berkeley, California, USA, October 26 - 30, 2009). C&C '09. ACM, New York, NY, 9-18.

[2] Barnes, T., Richter, H., Powell, E., Chaffin, A., and Godwin, A. 2007. Game2Learn: building CS1 learning games for retention. In *Proceedings of the 12th Annual SIGCSE Conference on innovation and Technology in Computer Science Education*(Dundee, Scotland, June 25 - 27, 2007). ITiCSE '07. ACM, New York, NY, 121-125

[3] Barnes, T., Powell, E., Chaffin, A., and Lipford, H. 2008. Game2Learn: improving the motivation of CS1 students. In *Proceedings of the 3rd international Conference on Game Development in Computer Science Education* (Miami, Florida, February 27 - March 03, 2008). GDCSE '08. ACM, New York, NY, 1-5.

[4] Bayliss, J. D. 2009. Using games in introductory courses: tips from the trenches. In *Proceedings of the 40th ACM Technical Symposium on Computer Science Education* (Chattanooga, TN, USA, March 04 - 07, 2009). SIGCSE '09. ACM, New York, NY, 337-341.

[5] Chaffin, A., Doran, K., Hicks, D., and Barnes, T. 2009. Experimental evaluation of teaching recursion in a video game. In *Proceedings of the 2009 ACM SIGGRAPH Symposium on Video Games* (New Orleans, Louisiana, August 04 - 06, 2009). S. N. Spencer, Ed. Sandbox '09. ACM, New York, NY, 79-86.

[6] Dahlberg, T., Barnes, T., Rorrer, A., Powell, E., and Cairco, L. 2008. Improving retention and graduate recruitment through immersive research experiences for undergraduates. In *Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education* (Portland, OR, USA, March 12 - 15, 2008). SIGCSE '08. ACM, New York, NY, 466-470.

[7] David J. Malan , Henry H. Leitner, "Scratch for budding computer scientists", ACM SIGCSE Bulletin, v.39 n.1, March 2007

[8] de Kereki, I.F., "Scratch: Applications in Computer Science 1," Frontiers in Education Conference, 2008. FIE 2008. 38th Annual , pp.T3B-7-T3B-11, 22-25 Oct. 2008

[9] Eagle, M. and Barnes, T. 2009. Experimental evaluation of an educational game for improved learning in introductory computing. In *Proceedings of the 40th ACM Technical Symposium on Computer Science Education* (Chattanooga, TN, USA, March 04 - 07, 2009). SIGCSE '09. ACM, New York, NY, 321-325.

[10] Maloney, J. H., Peppler, K., Kafai, Y., Resnick, M., and Rusk, N. 2008. Programming by choice: urban youth learning programming with scratch. In *Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education*(Portland, OR, USA, March 12 - 15, 2008). SIGCSE '08. ACM, New York, NY, 367-371

[11] Powell, E. M., Finkelstein, S., Hicks, A., Phifer, T., Charugulla, S., Thornton, C., Barnes, T., and Dahlberg, T. 2010. SNAG: social networking games to facilitate interaction. In *Proceedings of the 28th of the international Conference Extended Abstracts on Human Factors in Computing Systems* (Atlanta, Georgia, USA, April 10 - 15, 2010). CHI EA '10. ACM, New York, NY, 4249-4254

[12] Repenning, A., Basawapatna, A., and Koh, K. H. 2009. Making university education more like middle school computer club: facilitating the flow of inspiration. In *Proceedings of the 14th Western Canadian Conference on Computing Education* (Burnaby, British Columbia, Canada, May 01 - 02, 2009). R. Brouwer, D. Cukierman, and G. Tsiknis, Eds. WCCCE '09. ACM, New York, NY, 9-16.

[13] Tinto, V. Taking retention seriously: Rethinking the first year of college. *NACADA Journal*, 19:5-9, 1999.

[14] Bell-Watkins, K., Barnes, T., and Thomas, N. 2009. Developing computing identity as a model for prioritizing dynamic K-12 computing curricular standards. *Journal of Computing Sciences in Colleges*, 24, 3 (2009), 125-131.

[15] Wegerif, R. 1985. The social dimensions of asynchronous learning networks. *Journal of Asynchronous Learning Networks*, 1985.