Game2Learn Summer Usability Study
Paige Matthews

Abstract:

As a response to the decreasing number of women and minorities in the computer science field, the Game2Learn team is developing a massive multi-player online role playing game (MMORPG) that incorporates a fun interactive game environment while teaching programming concepts. The preliminary step in the development of this game, StormHaven, is a usability study to determine if the games are enjoyable, if the subjects feel they can learn with games, and if the subjects would prefer to learn from games such as these. The qualitative data from this study, along with data from pre- and post- test comparisons, will serve as information for further development of StormHaven and the goal of recruiting and maintaining computer science students.

Introduction:

In order to recruit and maintain computer science students, the Game2Learn team is developing an interactive computer game to supplement in the teaching of introductory computer science. The first step in developing the game is a usability study to determine the effectiveness of learning computer science concepts through a game. StormHaven, the game being developed, will provide a fun interactive environment in which to learn programming. We predict that students will like learning programming from a game better than the traditional classroom homework and assignments.

For the study, two different mini games were designed to test and compare several concepts. The layout of the study is designed in order for the subject to be able to compare storylines, game layout, and interfaces. Throughout this paper, the reasons for the development of this game as well as the process including initial stages of development, one of the games developed for the study, as well as the usability study will be discussed.

Background:

According to the 2004-2005 Taulbee Survey, enrollment in computer science and computer engineering programs has decreased 14% from the previous year and 30% as compared to three years ago. Along with the general decrease in students enrolled in these programs, the percentage of women enrolled in bachelor degree programs decreased from 17% to 14.7% from the 2003-2004 year to the 2004-2005 year. The survey also detected an increase in the proportion of white, non-Hispanics, receiving these degrees. This group makes up 59.6% of bachelor's and master's recipients, which greatly outweighs the numbers of various other ethnicities (Zweben, 10-12).

One reason women and minorities are not a part of science, engineering, and mathematics is the non-supportive environment. One suggested method of adapting the workplace to better suit diversity is through intellectual diversity (Fancsali, McGinnis, 51). One method of intellectual diversity is implemented by learning through computer games. Learning through computer games not only teaches the intellectual skills required to function within the game, it also teaches basic knowledge of computers. For example, "some basic components of games are solving problems, answering questions, planning,

understanding instructions, and basic academic skills. Along the way, knowledge is gathered about operating the system, keystroke operation, familiarity with computer terms, and other general skills, which boosts confidence in utilizing a computer for pragmatic applications" (Natale, 25). With an increase in confidence, users will be more comfortable with the computer in general. When one's skills are weak, she will avoid spending time using these skills. However, with an increase in general computer skill and knowledge, the user will become more comfortable using the computer for these as well as other applications (Natale, 28-29). This self-efficacy, the belief in one's ability to execute successfully a certain course of behavior, influences the decision of whether to engage in a task, as well as the effort and persistence put into the task. Furthermore, female students base much of their choice of courses and careers on self-efficacy (Busch 1-2).

The results of one study indicate that "the most important predictor of computer attitudes is previous computer experience and encouragement" (Busch 6). While encouragement cannot be directly changed by academia as a whole, there are several ways to effect exposure and experience with computers; one way is through computer games. "When learning is divorced from doing a meaningful task-as are many arbitrary, decontexualized activities in the classroom-the learning becomes just another chore, low on the priority stack" (Soloway et. al. 40). By incorporating learning in a fun interactive computer game environment, the learning will not be meaningless or boring. There will also be more motivation to learn.

The video game market is predominately geared towards a male audience. Women in the gaming industry make up a mere 17% (Haines, 5). The lack of women in the video gaming industry as well as the current make-up of the gaming population contribute to the focus on males by the industry. In general, females do not like violent games with pressure, repetitive action, and speed. They are more interested in social interactions within a game (Ziemek, 183). "As most girls present it, their more limited involvement with computers, especially their lack of interest in games and their lack of interest in having a career in computing, has more to do with disenchantment than with anxiety or intellectual deficiency." Females are more interested in promoting human interaction than destroying (AAUW, 8).

According to ISDA, women comprise 43% of the computer gaming population (Case). Likewise, women make up of 50.4% of the online gaming population (Guernsey).

For the Game2Learn project to be successful, the game must be appealing to women as well men. We must focus on making a game that will appeal to both genders. We propose that creating a MMORPG with the appeal of the online games that women play, along with gearing quests towards women as well as men, will draw both women and men to the introductory course in which it is included.

Quest Development:
The initial stage of the project was brainstorming quest ideas for StormHaven. Quest ideas the team developed were both large and small. Initially the team developed ideas for the StormHaven in general. After we had the initial ideas, we then had to adapt the quests to fit the usability study. Likewise, we adapted the study to fit our quests as well.

Some of the ideas that my group developed were packing eggs in a crate to teach nested for loops, sorting keys in a chest to learn recursion, and playing songs for the king for while loops.

During the development of the mini-games for the study, one of the quests that I worked with was "Playing Songs for the King." The quest was not implemented for the usability study because of its complexity. The basic idea for this quest is that the player has to use loops and switches to instruct different musicians to play during appropriate songs. During the quest, there are different levels of difficulty. The player starts by converting if/then statements to a switch statement. Later, the player has to use other more difficult interfaces such as typing code to conduct more difficult songs.

Saving Sera:

Prior to running the usability study, there were two teams to work on the mini-games we developed. The game I primarily worked on was Saving Sera. Saving Sera is an exploratory island game where the subject gets to choose the quests he or she wants to complete. The game begins in the king's court where the evil Gargamel kidnaps the Princess Sera. Arshes, the player's character, is on a mission to save the princess Sera. Before Arshes can get off the island to save Sera, he must collect enough money by completing quests to board a boat.

There are three quests that Arshes can complete to earn money. He can complete the fisherman quest, the billboard quest, or the egg quest. During the fisherman quest, the player rearranges statements in a while loop to help the fisherman know when and how to fish. On the other hand, the billboard quest uses print statements and variables to post adds on different billboards throughout the island. The other quest on the island, the egg quest, uses nested for loops to drop eggs in cartons. The different interfaces for these quests are code jumble, fill in the blank, and dialogue trees respectively.

After getting enough money by completing quests on the island, Arshes boards the boat to Gargamel's lair. Arshes is then able to rescue the princess. However, the evil Gargamel sets a bomb. By answering questions about recursion through dialogue trees, the user is able to fight the bomb. Every time the player gets a question correct, he or she receives a piece of a diagram explaining the recursive sort quicksort. Lastly, in order to save everyone from the bomb, the player uses drag and drop to fill in a quicksort diagram.

Summer Usability Study:

For the usability study, two game prototypes were developed. Saving Sera, a two dimensional exploratory game, was developed in RPG Maker XP. Several different interfaces were implemented in this game. In Saving Sera, the subject has the ability to choose the quests he or she wants to complete. Each quest in this game consist of a separate interface; the interfaces for this game are fill in the blank, code jumble, symbolic programming, and dialogue trees. The other game, The Catacombs, a three dimensional game with linear quests, was developed in Neverwinter Nights. For The Catacombs, two versions were developed. Each version primarily used one interface. With the same basic storyline of a family trapped in the catacombs for each version, the primary interfaces used were symbolic programming, dialogue trees, and multiple choice.

During the study, each subject plays through both Saving Sera and one version of The Catacombs games. Prior to playing the games, the subject completes a demographic questionnaire to get background information as well as a pre-test to obtain an initial assessment of the subject's programming skills. During the playing of the games, as well as during the interview, the subject will be video tapped. A log will also be kept of actions taken during the game. After the completion of each game, interview questions are asked to get general feedback about the specific game. There are also questions during the interview about the subject's perception of learning from a game, as well as overall feedback about the project. Lastly, the subject completes a post-test. The post-test will be compared to the pre-test to look for any learning.

The primary data gathered from the study will come from the interviews with the subjects. We will also watch video logs, analyze play time logs, and compare the pre- and post-tests.

Results:

We are currently in the process of running the usability study. Thus far in the study, the response to learning from a game has been positive. The subjects have enjoyed the game and have responded to the interview questions with feedback that will be helpful in further development of StormHaven.

Conclusion:

We hope that from the usability study, we will find that the subjects would prefer to learn programming from a game more than from a traditional classroom setting. Although the study is a short amount of time to learn programming and the subjects have prior programming skills, we hope and expect that the post-test will have the same or better results than the pre-test.

Future Work:

The development of StormHaven is an ongoing project. This usability study was the preliminary study in the long term development of a computer game to teach introductory computer programming. Future works include testing the difference between a class with more traditional methods of homework and assignments and a class using a computer game for their homework assignments. This comparison would give a more accurate representation of the differences in learning from a computer game and traditional textbook learning.

References:

AAUW. *Tech-Savvy: Educating Girls in the New Computer Age*. American Association of University Women Education Foundation, New York, 2000.

Busch, Tor. Gender Differences in Self-Efficacy and Attitudes toward Computers. *Journal of Educational Computing Research,* 1995, vol. 12, 147-158.

Case, S. Women in Gaming. Microsoft, Jan. 12, 2004. Online: http://www.microsoft.com/windowsxp/using/games/learnmore/womeningames.mspx

Fancsali, Stephen; McGinnis, Laura Untapped Resources: Can Intellectual Diversity Promote Cultural Diversity in Technology? ACM Press 2005.

Guernsey, L. 'Women Play Games Online in Larger Numbers Than Men,' *The New York Times*, January 4, 2001.

Haines, Lizzie. Why are there so few women in games? *Media Training North West*. September 2004.

Natale, Marc J. The Effect of a Male-Oriented Computer Gaming Culture on Careers in the Computer Industry. Computers and Society, June 2002.

Soloway, Elliot E, Guzdial, M, Hay, KE.   Learner-Centered Design: The Challenge For HCI In the 21st Century. *ACM Interactions.* April 1994.

Ziemek, Tina R. Two-D or not Two-D: Gender Implications of Visual Cognition in Electronic Games. SIGGRAPH, *ACM Press*, 2006.

Zweben, Stuart. 2004-2005 Taulbee Survey. Ph.D. Production at an All-Time Hight with More New Graduates Going Abroad; Undergraduate Enrollments Again Drop Significantly. *Computing Research Association Taulbee Survey,* May 2006.
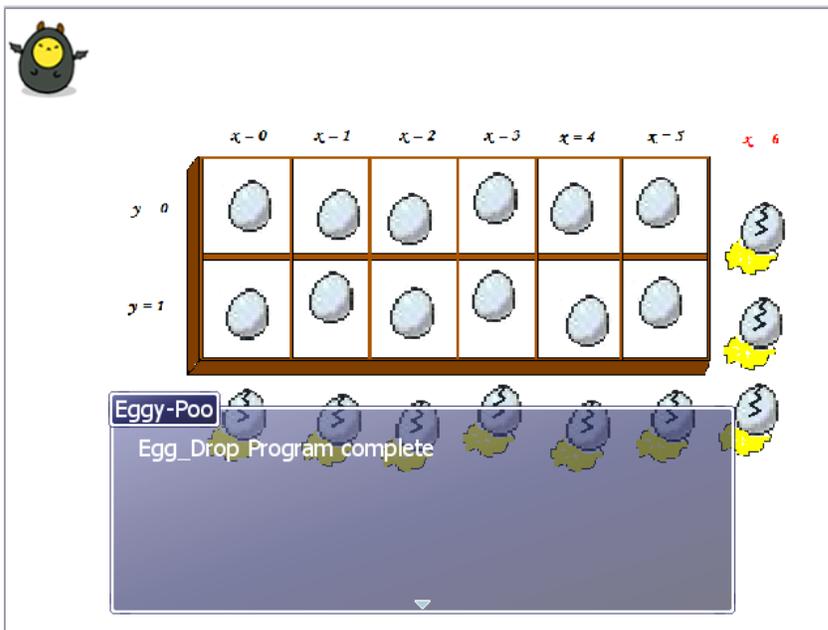
Example Quicksort Diagram for Saving Sera Final Quest:
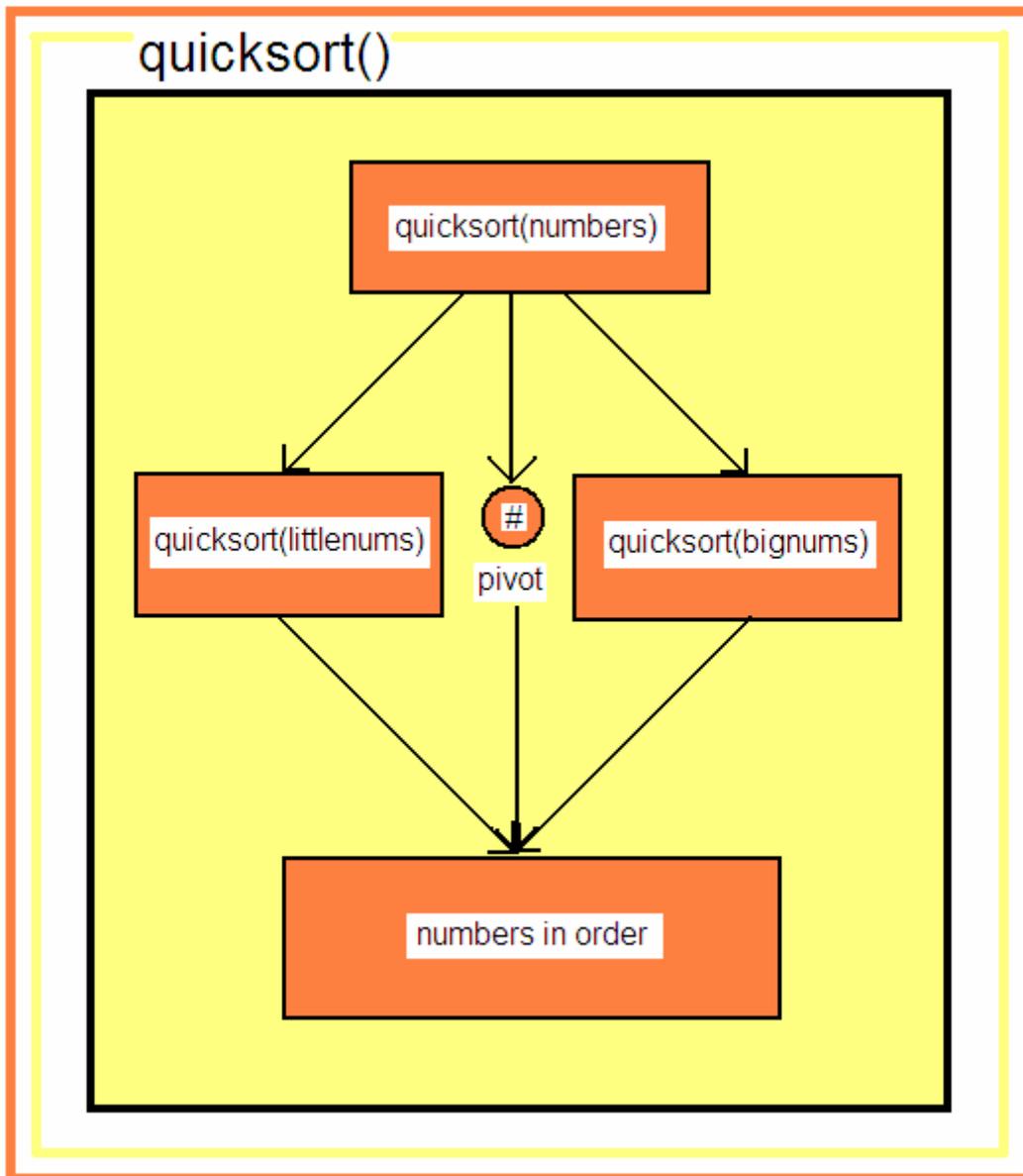
Screenshot from The Catacombs:

Screenshot from Saving Sera Egg Quest:

Example Pre/Post-Test:
Subject #:

For questions 1-3, match each of the segments to the letter that corresponds with how many times + will be printed.

```
___ 1. for (int i = 0; i <= 10; i++)
       {
          println( "+" );
       }
```
a. 0

b. 1

c. 9

```
___ 2. for (int j = 1; j < 10; j--)
       {
          println( "+" );
       }
```
d. 10

e. 11

f. indefinite number
   (infinite loop)

```
___ 3. int k = 10;

       while (k > 0)
       {
          println( "+" );
          k--;
       }
```

For questions 4-6, give the output of the block of code when x equals the following:

4. x = 27

5. x = 10

6. x = 8

```
if (x < 10)
{
   println( "y" );
   x = 11;
}
else if ( x < 15 )
{
   println( "n" );
}
else
{
   println( x );
}
```

00	01	02	03
10	11	12	13
20	21	22	23

7. Fill in the squares in the following code so its output will match the table above.

```
for ( int j = 0; j □□; j++ )
{

   for ( int k = 0; k □□; k++ )
   {
      print(  j + k + "   ");
   }

   print( "\n" );
}
```

8. Consider the following function:

```
void test ( int n )
{
   if ( n > 0)
   {
      test ( n - 1 );
   }

   println ( n + "   " );
}
```

What is printed by the call test(3)?

What is printed by the call test(0)?