Educational Multiuser Online Game, “debugger” game for Introductory Computer Science Class

Ilmi Yoon[1], Gary Ng[2], Zoran Millic[1], Supakit Kiatrungrit[3], Yiyi Miao[4], Sunggye Hong[5]

[1]Computer Science Dept, San Francisco State University
[2]Pacific Ecoinformatics and Computational Ecology Lab
[3]Dept. of Information Technology, San Francisco State University
[4]OPTWAT. INC, San Francisco
[5]Dept of Special Education, San Francisco State University

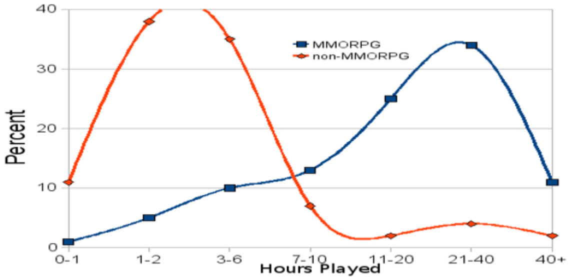
Abstract

Computer games became daily routine to young generation. Especially games with social interactions are building strong cultures to young students. Multiplayer Online Role Playing Game (MMORPG), “debugger” is developed to utilize this new culture for education. Introduction to Computer Programming is a challenging course for majority of students as concepts are abstract, non-intuitive and completely new to most of them. Considering the serious decline of CS major students across US, developing publicly available educational games for CS students is a timely effort. Debugger game aims to build a collection of educational mini games within a community of learners. Within the virtual world of debugger game, players fight with bugs by solving problems, play “CodeGame” to earn resources to build their own motherboard space, interact with other players via chatting, friends list, reputation list and other features. The debugger game is intended to take advantage of social interactions and community to retain players longer, promote players to solve more quests, and encourage players to discuss and learn from each other more actively, as recommended in peering learning or peer tutoring. The debugger game is up and running and a game client can be downloaded for playing at http://thecity.sfsu.edu/~debugger.

1. Introduction

Game-based learning has been one of active educational research area with hopes that playing game would effectively aid students’ learning. Large number of studies showed the educational impact of the game-based approach, educational games (or called as edutainment, serious game, gamification) have been adapted and used in diverse areas such as elementary math to professional medical trainings (Green ’03 & Prinsky ’01). Recent study shows that game-based education reaches beyond original objective of educational level and motivated players to achieve much more – promote the players potential upto extraordinary level (Cooper ’10).

Cooper utilized multiplayer online game as social interactions serve well to motivate players. Other studies also showed significance of social interactions in the game to retain players longer. Figure 1 shows the general play time of stand-alone game vs. MMORPG game; MMORPG game keeps players longer over time (Ng ’05). We observed that popular MMORPG games are extremely good at retaining their players for very long period time (months to years) and invite friends to play together (spread through human network). Recently, Farmville, a relatively simple farm nurturing (crops or farm animals) face book game made a nice show case that how quickly a game can spread through social interactions (50 million active players) and keep players to play over and over.



MMORPG gamers play significantly longer than non-MMORPG gamers. This graph showing percent of players by hours played.
(Source: Ng and Wiemer-Hastings, 2005, N=91)

Figure. 1 - MMORPG vs. non-MMORPG games, hours played; Source: Addiction to the Internet and Online Gaming.

Popular MMORPG games like World of Warcraft has several millions players. Lots of research has been done to analyze the main driving factors (Song ’07 & Yee ’02). These studies pointed out that players especially enjoyed the fact that there is a community of audiences to give compliments when player achieved high level or completed quest. It is an important finding which can significantly increase educational games' efficacy. Educational games that are well designed for good educational impact often suffer from the fact that players do not play voluntarily over and over.

The MMORPG game, "deBugger" was designed and developed to make use of these findings to assist Introductory Computer Science course. It is generally accepted that it is very difficult to learn to programming (Jenkins '02). For example Bergin and Reilly (Bergin '05) note that *It is well known in the Computer Science Education (CSE) community that students have difficulty with programming courses and this can result in high drop-out and failure rates.* (p. 293). The concepts are non-intuitive or/and overwhelming. Students need lots of exercises to digest the concepts, close interactions, guidance and help when they are lost. Average pass rate for this course at colleges across US is around 60% (Bennedsen '07). Considering that 70% decline of incoming college freshman intending to major in CS, it is imperative to develop and provide publicly available educational game for CS students.

Debugger game is MMORPG game there is persistent virtual worlds that continues to exist even after a user exits the world, and players are represented by their characters in that world. The characters have permanent status like game level, virtual money, health, list of friends, game items like weapons or tools.

In the debugger game, there are different styles of games that players can play with other players or against bugs; debugger title was inspired by the origin of the word (removing a bug to fix errors). These mini games (games inside of debugger game) are intended to develop competence in computer science concepts when played repeatedly over and over.

Design principles of debugger game are; (1) Make use of players attachment to the character. In commercial MMORPG games, players have shown strong attachments to their online status (or their own characters). To be able to level up their characters, players tolerate grinning process –known as boring, time consuming and repetitive tasks. Players happily spend hours and hours to succeed in a quest. While there is pure pleasure of succeeding in quests, studies show that players are strongly motivated by the fact that their successes in quest result in level-up or acquisition of special awards. This motivation can be easily linkedd to Educational MMORPG game. (2) Studies have shown that students learn from peers as much as from teachers (Kaufman '99, Christudason '03, Johnson '93, Falchikov '03). Students tend to ask questions for clarifications among peers first and then ask questions to teacher if no answer can be found from peers. In addition, peer pressure pushes students not to be left behind. Online game communities have shown that experienced players take great pleasure helping novice players. These advanced players can be further motivated to help new players like a TA, giving close individual interactions that teachers may not give in real world. Frustration from the learning can be alleviated by sharing the similar troubles of individual with peers. All these peers can be connected in the virtual world, available when needed, and increasing educational impact. (3) For shy students, they can hide behind their virtual character, so do not need to fear the failure. (4) Importantly, usual MMORPG games grow and adjust depending on the needs or desire of the player community unlike console games that have pre-fixed everything before manufacturing. Debugger game can do the same. In addition, debugger can collect all sorts of (educationally) meaningful player activities in the game server. These data can be very useful for iteratively design cycle (design, development, user test) of the debugger game for various users to maximize their learning outcome.

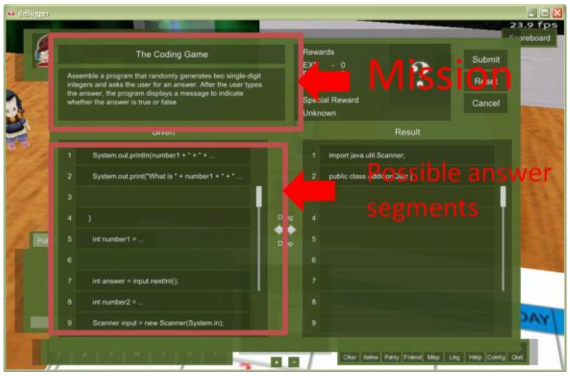
One thing to note is that debugger game is not intended to replace the ordinary class teaching, but to be used as an effective aid like a virtual TA or lab, so students can comprehend the core pedagogical components and then progress smoothly. The core idea of debugger game is helping students to master the concepts through voluntary repetitive practices (lab) and clarifications from other players (virtual TA).

Section 2 discusses the mini games and community features within the debugger game and section 3 presents the structure of the debugger world. Section 4 discusses on-going implementations and user trial plan and section 5 concludes the paper.

2. Mini games for Computer Science Learning

There are different approaches to introduce the computing concept to beginners. First and newer approach is starting with abstract and high level, usually utilizing objects like graphics, sounds, and scripts to get quick results and motivating to try further. Examples are MIT Scratch, CMU (storytelling) Alice and NYU Rapunsel. These are tangible (video+audio) and fun approach to problem solving and logical flow. However these approaches are still experimental and eventually CS students have to take programming course. So, still majority of college and AP CS courses take traditional approach of using practical programming languages like JAVA or C++. And CS AP Collegeboard uses this approach. When the second and traditional approach uses practical programming languages, courses cover the core concepts (problem solving with pseudo-code, variables, data types and operators, selection flow control, repetition flow control, methods, strings, arrays) in sequential order with slight variations. Mini games are designed to work for one concept at one time or one group of a few related concepts only, so each game can be designed with specific focus.

First mini game called GodeGame is for practicing program structure, importance of syntax rules, and simple program flow. When mini game starts, player will receive a simple mission randomly assigned according to the player's level and a list of possible answer segments will be displayed on the left panel as shown figure 2. To solve the mission, player picks up the answer segments that are correct in syntax and in proper program structure then arrange them on right answer panel in proper sequence by drag and drop. With this game, player does not have to solve the problem from scratch, but think of the functionality of given answer segments and then place them in right order. The answer segments can be either actual program code level or abstract high level like pseudo code as shown at figure 2. Abstract high level helps students learn problem solving using top-down or divide-conquer approach easily without thinking of the syntax level. Screen capture of playing this mini game is recorded and available at YouTube (YouTube link).



Assemble a program that randomly generates two single-digit integers and asks the user for an answer. After the user types the answer, the program displays a message to indicate whether the answer is true or false

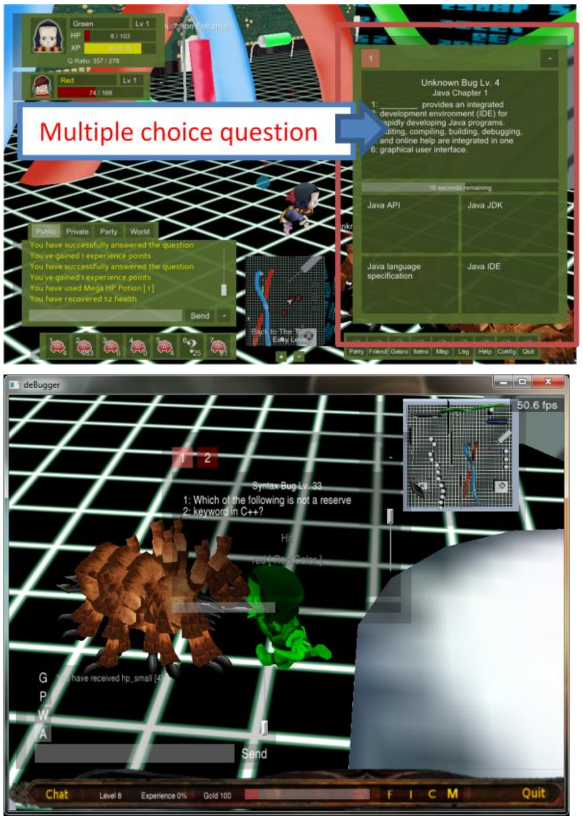
```
System.out.println(number1 + " + " + ...
System.out.print("What is " + number1 + " + " ...

}
int number1 = ...

int answer = input.nextInt();
int number2 = ...
Scanner input = new Scanner(System.in);
```

```
import java.util.Scanner;
public class Addition Quiz {
```

Figure. 2 – CodeGame Interface: mission is displayed at upper left corner and the options to drag are displayed below the mission. Individual options can be dragged and dropped.

Another mini game currently implemented is multiple choice questions as it is easier to implemented and used for user trial quickly. Multiple choice style is easy to ask diverse topics from definition, concepts and picking a correct execution result from a given code segments. These questions are organized into levels in the order of variables, data types and operators, selection flow control, repetition flow control, methods, strings, and arrays. So students can play the right level that they have learned.

To make the multiple choice questions to be fun, these questions are used while player is fighting with bugs in the battle, playing board game with other players, doing a quest with big boss bug with other players or use the question for PvP (Player vs. Player) game. Rewards for playing these games are increasing health, getting game gold, increasing level or getting a gift of game items. Game items can be also purchased with game gold. And game items are designed to add fun to the game and allow players to grey out two options from the multiple choices, protect player's health while being attacked by bugs or upgrade decorative items.



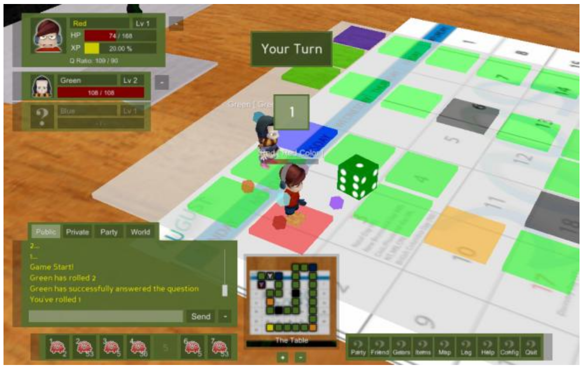
1: ________ provides an integrated
2: development environment (IDE) for
rapidly developing Java programs.
Editing, compiling, building, debugging,
and online help are integrated in one
6: graphical user interface.

You have successfully answered the question
You've gained 1 experience points
You have successfully answered the question
You've gained 1 experience points
You have used Mega HP Potion [1]
You have recovered 52 health

1: Which of the following is not a reserve
2: keyword in C++?

You have received hp_small [4]

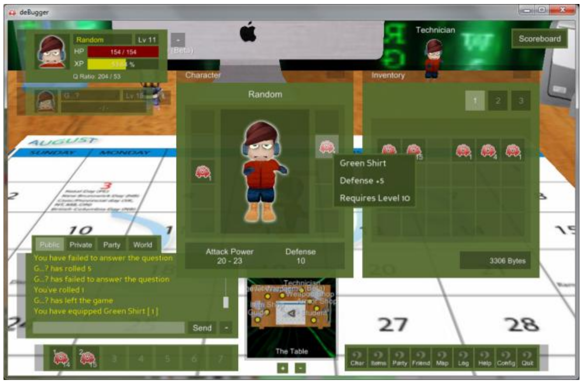
Figure 3a & 3b – A player is fighting with a bug. Bug asks a question and the player has to answer. Health gets damaged as time lapses. Player's health, level, gold are shown at upper left corner. Lower left corner panel serves as a message panel and chat panel.



2...
1...
Game Start!
Green has rolled 2
Green has successfully answered the question
You've rolled 1

Figure. 4 – Two players are playing a board game. Each player takes turn, roll the dice, and solve a problem to progress. Winner who arrives the end point faster receives game gold as reward.

In addition to those game features, there are many social activity features within the debugger game. Players can chat with other players in a few different ways. Public chat is for chatting with anyone in the same virtual space (room) and private chat is for chatting with ones that a player wants to chat. Also chatting with friends who are online is possible. A player can manage friend's list. Player can choose to show their level and other performance to the friends or public.



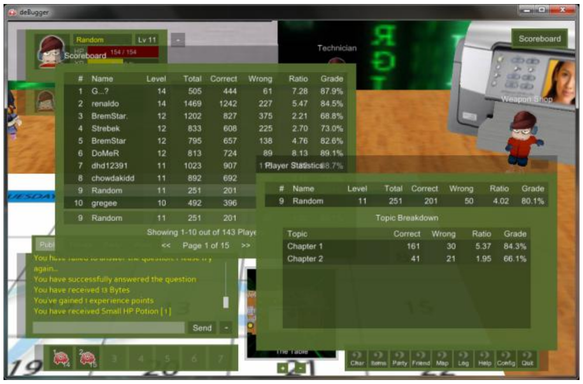
You have failed to answer the question
G...? has rolled 5
G...? has failed to answer the question
You've rolled 1
G...? has left the game
You have equipped Green Shirt [1]

Figure 5 – Player can check their inventory for game items. Players can purchase game items using game gold, trade with other players or give a gift to friends.

Player's achievement can be measured by levels, number of questions played and the accuracy ratio. Achievement board displays top 10 high score achievers to encourage players to play longer and better (Figure 6).

3. DeBugger Game Architecture

Social gaming and MMORPG are relatively recent game genres as they bloom over the availability of Internet. Social interactions within the game fire the excitement of the game and popular games easily develop communities of multimillion players. Once community of learners is created, students can learn not only from educational game components but also from their peers discussing their problems with them. Debugger game was created with the vision of nurturing such community of learners playing collection of games together, help each other and invite more friends to join.



Scoreboard

#	Name	Level	Total	Correct	Wrong	Ratio	Grade
1	G...?	14	505	444	61	7.28	87.9%
2	renaldo	14	1469	1242	227	5.47	84.5%
3	BremStar.	12	1202	827	375	2.21	68.8%
4	Strebek	12	833	608	225	2.70	73.0%
5	BremStar	12	795	657	138	4.76	82.6%
6	DoMeR	12	813	724	89	8.13	89.1%
7	dhd12391	11	1023	907	[illegible]	[illegible]	88.7%
8	chowdakidd	11	892	692			
9	Random	11	251	201			
10	gregee	10	492	396			
9	Random	11	251	201			

Showing 1-10 out of 143 Playe

<< Page 1 of 15 >>

#	Name	Level	Total	Correct	Wrong	Ratio	Grade
9	Random	11	251	201	50	4.02	80.1%

Topic Breakdown

Topic	Correct	Wrong	Ratio	Grade
Chapter 1	161	30	5.37	84.3%
Chapter 2	41	21	1.95	66.1%

You have failed to answer the question. Please try again...
You have successfully answered the question
You have received 13 Bytes
You've gained 1 experience points
You have received Small HP Potion [1]

Figure. 6 – Player's performance is measured in diverse ways. And players can see other players' performance unless blocked by the player.

To enable this, we need to create and support virtual world persistently utilizing game server. Game server runs all the time, connects all the clients, and updates DataBase Server as well as clients. Game client connects to the game server when a client wants to play the game, receive the latest update of the virtual world since the last time player logged out, and interact with players already online. Database server stores all the data of individual players (health, level, money, game items, friend's list, performance, and etc) and other data to maintain the game world persistent. Game protocol is developed to make the communication effective. DeBugger game also has a bug server that maintains all the bugs – how often they appear (spawn), what kind of game item they drop, how aggressively attack players, etc. Bug server also controls bugs with simple AI to make them wander around and handle collision naturally to make the game fun when player interacts with bugs.
Figure. 7 shows a flow of debugger game, showing each component and their connections to each other. Game client was developed utilizing Panda3D and python scripts. Game server was developed using JAVA, using MySQL as DB server. Bug server was extended from game client that already included collision handling.

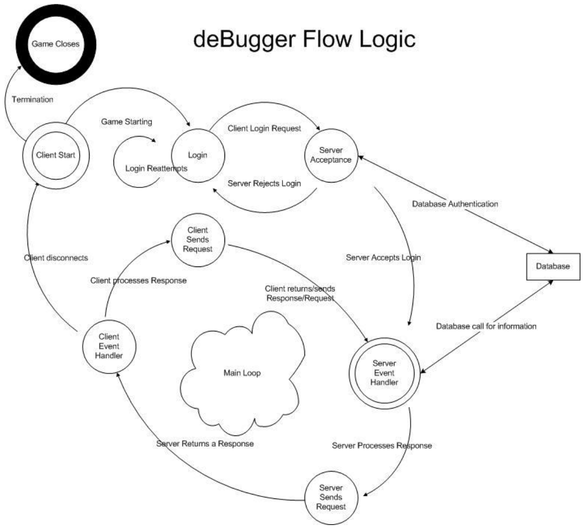


Figure 7 - deBugger Flow Logic used to show the cycle of information being passed between *Client* and *Server*

Currently, debugger supports two big virtual spaces (a desk and a motherboard) where players can meet and interact with each other and bugs. Figure 3b shows mother board space where players can explore and fight with bugs. Like other MMORPG games, this virtual space can be extended over time by collecting feedbacks from players.

4. On Going Implementation

Currently additional mini games are designed and being implemented and user trials are under IRB review for human subject protocol. We are trying the iterative game development cycle of design, implement and user test followed by a new cycle of revised design, implementation and user trial. The first user trial will focus on the CodeGame, bug battles, and other community features discussed in section 3.

Designs of additional games are as follows. Four more mini games have been designed. First mini game to be implemented is called “Variables” and designed for practicing concepts of variable, data type and operators. Students need to learn that variables are spaces in the memory where program stores value according to the data type and operators act on the values according to data type. This will be similar like Cisco Binary Game [ref] where user finds the memory space and fill the value according to the type. For example, 1 / 2 is 0.5 according to usual math, but 1 / 2 is zero in JAVA, C, C++ while 1.0/2 is 0.5. Repetitive playing of this game will help students master the three concepts combined together. Second game is for practicing selection flow control concept. It will utilize idea of racing game and player’s route will be selected depending on the value in the car and the condition at the branch. To be able to finish the lap in given time, player should choose the value wisely to utilize the conditions at the branch to stay at desired routes. This game will be extended to repetition flow control by running multiple laps. For the beginners, the car move very slowly to give enough time to think and the car will get accelerated as player progresses well. Third game is for practicing repetition flow control, especially nested repetition flow control using pattern printing style arcade game. Forth game is for practicing array and repetition flow control.

Strength of MMORPG approach for educational game is that game server can collect every activity of player; how long, how often, how many questions does player attempt to solve, what is the accuracy rate, and then how their performance at class correlate to the game performance.

5. Conclusion

Introduction to Computer Programming is a challenging course for majority of students as concepts are abstract, non-intuitive and completely new to most of them. Considering the serious decline of CS major students across US, developing publicly available educational games for CS students is a significant contribution. Debugger game aims to build a collection of mini games within a community of learners. The game will be iteratively refined through user trial and revision of the game to increase the educational efficacy. Debugger game tried to maximize the education efficacy utilizing the social interaction and the power of the community itself (players help each other). Once the human subject is approved, user test will be performed on the debugger game.

ACKNOWLEDGEMENT

This project is being funded by National Science Foundation Div. Of Biological Infrastructure, Biological Databases and Information, NSF DBI-0543614

6. References

(Bergin ’05) S. Bergin and R. Reilly. The influence of motivation and comfort-level on learning to program. In Proceedings of the 17th Annual Workshop of the Psychology of Programming Interest Group pages 293-304, University of Sussex, Brighton UK 2005.

(Bennedsen ’07) Jens Bennedsen and Michael E. Caspersen, Failure rates in introductory programming, SIGCSE 2007, pp 32-36.

(Christudason '03) Christudason, A. Successful Peer Learning, http://www.cdtl.nus.edu.sg/success/sl37.htm

(Cooper '10) Cooper, at al, Predicting protein structures with a multiplayer online game, Nature 466, pp 756-760, August, 2010.

(Falchikov '03) Falchikov, N., Blythman, M., Learning together: peer tutoring in higher education.

(Green '03) GREEN, C S, AND BAVELIER, D. 2003. Action video game modifies visual selective attention. Nature 423 (2003), 534-537, Letters to Nature.

(Johnson '93) Johnson, D.W.; Johnson, R.T.; & Holubec, E.J. (1993). Circles of Learning. Edina, MI: Interaction Book Company.

(Jenkins '02) T. Jenkins. On the Difficulty of Learning tp Program. In Proceedings for the 3rd Annual conference of the LTSN Centre for Information and Computer Sciences , Loughborough, UK August 27 - 29, 2002.

(Kaufman '99) Kaufman, D.B.; Felder, R.M.; & Fuller, H. 'Peer Ratings in Cooperative Learning Teams'. Proceedings of the 1999 Annual ASEE Meeting, ASEE, Session 1430'.

(Ng '05) Ng, Brian D. and Peter Wiemer-Hastings, Addiction to the Internet and Online Gaming, Cyber Psychology & Behavior 8.2 (2005)

(Papstergiou '09) Papstergiou, M., Digital Game-Based Learning in High School Computer Science Education: Impact on Educational Effectiveness and Student Motivation, Computers & Education, v52 n1 p1-12 Jan 2009.

(Prensky '01) PRENSKY, M. 2001. Digital Game-Based Learning. McGraw-Hill, New York. ISBN 0-07-136344-0.

(Song '07) Seungkeun Song, Joohyeon Lee (2007), Key factors of heuristic evaluation for game design: Towards massively multi-player online role-playing game, International Journal of Human-Computer Studies, no. 65, 709-723

(Yee '02) Yee, N. (2002) Understanding MMORPG Addiction

(YouTube link) http://www.youtube.com/results?search_query=SFSU+debugger&aq=f