IMPROVEMENT OF A LECTURE GAME CONCEPT
Implementing Lecture Quiz 2.0

Bian Wu, Alf Inge Wang
Dept. of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway

Erling Andreas Børresen, Knut Andre Tidemann
Dept. of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway

Keywords: Educational game, Multiplayer game, Computer-supported collaborative learning, Software engineering education, Evaluation, System Usability Scale (SUS).

Abstract: A problem when teaching in classrooms in higher education is lack of support for interaction between the students and the teacher during the lecture. We have proposed a lecture game concept that can enhance the communication and motivate students through more interesting lectures. It is a multiplayer quiz game, called Lecture Quiz. This game concept is based on our current technology rich and collaborative learning environment and was proved as a viable concept in our first prototype evaluation. But based on our previous implementing experiences and students' feedbacks about this game concept, it was necessary to improve this first lecture quiz prototype in four aspects: 1) Provide a more extensible and stable system; 2) Easier for students to start and use; 3) Easier for the teachers to use; and 4) Good documentation to guide the further development. According to these aims, we developed the second version of Lecture Quiz and carried out an evaluation. Through comparing the evaluation data from second version with first version of Lecture Quiz, we found that both surveys show that the Lecture Quiz concept is a suitable game concept for improving lectures in most of aspects and that Lecture Quiz have been improved in several ways, such as editor for the teachers to update the questions, improved architecture that could be easy to extend to the new game modes, web-based student clients to get an easier start than first version of lecture quiz, etc. The results are encouraging for further development of the Lecture Quiz platform and for exploring more in this area.

1 INTRODUCTION

Traditional educational methods may include lecture sessions, lab sessions, and individual and group assignments, in addition to exams and other standard means of academic assessment. From experiences at our university, we acknowledge that today's lectures mostly use slides and electronic notes and can still be classified as one-way communication lectures. In a typical lecture the teacher will talk about a subject, and the students will listen and take notes.

However, the exclusive use of such methods may not be ideally suited to today's students, particularly those in the generation born after 1982, or "Millennial students," as termed by education researchers (Raines; Oblinger and Oblinger, 2005; D. Oblinger, 2003). Millennial students prefer hands-on learning activities, and collaboration in education and the workplace. Female, African American, Hispanic, and other underrepresented students may also be inclined toward ways of learning and working that involve more group work and social interaction than traditional university education provides (Williams et al., 2007).

The technology has now evolved and smart phones, laptops and wireless networking have become an integrated part of students' life. These technologies open new opportunities for interaction during lectures. As game technology is becoming more important in university education, we proposed a way to make the lecture more engaging and interactive. In 2007 we developed Lecture Quiz, an educational multiplayer quiz game prototype (Wang et al., 2007; Wang, 2008) denoted as LQ 1.0 in this paper. It provides a possibility for the students to participate in a group quiz using their mobile phone

Wu B., Wang A., Børresen E. and Tidemann K..
IMPROVEMENT OF A LECTURE GAME CONCEPT - Implementing Lecture Quiz 2.0.
DOI: 10.5220/0003331300260035
In *Proceedings of the 3rd International Conference on Computer Supported Education* (CSEDU-2011), pages 26-35
ISBN: 978-989-8425-50-8

or laptop to give an answer. The questions are presented on a big screen and the teacher has the role as host of a game show. This prototype was created in a hastily manner to prove that the game concept was viable for educational purposes.

The implementation of LQ 1.0 was clearly a prototype that was made as a primary proof of concept. It lacked a good structure to serve as a platform for various lecture quiz games and was not designed with extendibility and modifiability in mind. As everything was hard-coded, it was difficult to extend this prototype to be used at a larger scale. There were also issues with an unstable application and the architecture itself was not built for large-scale usage. This could be easily identified by some limitations, such as that only one session was allowed per server and no ability for the teacher to edit quiz data without hacking into the database. We wanted to extend its structure to support playing many lectures at the same time. Besides of these limitations of the software architecture, we also wanted to add new features to save preparation time to make a quick and easy start of the game, provide guidance for the new developers, and quiz editor tools for teachers. In the light of this, the main aim was to develop second version of Lecture Quiz, denoted LQ 2.0, providing the following features: 1) A more extensible and stable system with a suitable architecture; 2) Easier to start and use; 3) Easier for the teachers to use; and 4) Good documentation as reference for the further development in future, such as add new game modes to the system.

The final goal was to give a good and solid base as an extendable lecture game platform, thus hopefully making it a regular part of university lectures.

2 RELATED WORKS

Here we will present a survey of similar approaches for lectures, the design criteria for Lecture Quiz and introduce the previous version and the improvements of second version.

2.1 Literature Review

There was no paper describing exactly the same game concept using the technical infrastructure in lecture halls for higher education when we implemented LQ 1.0. During implementing LQ 2.0 in 2010, we found some new similar quiz games used in education in different ways, but excluding the quiz used without any technology, such as (Schuh et al., 2008) or the quiz development frameworks, such as Quizmaker (Landay, 2010).

Using a game in a portable console (Larraza-Mendiluze and Garay-Vitoria, 2010) describes an educational strategy that directly situates students in front of a game console, where the theoretical concepts will be learned collaboratively through a question and answer game. PCs and Nintendo DS consoles were compared.

Moodle (Daloukas et al., 2008) is an online open source software aiming at course management. It focuses on a game module consisting of eight available games, which are “Crossword”, “Hangman”, “Snakes and Ladders”, “Cryptex”, “Millionaire”, “The hidden picture”, “Sudoku” and “Book with questions”. Their data are derived from question banks and dictionaries, created by users, both teachers and students. The rationale behind the design is to create an interactive environment for learning various subjects. Since learners are accustomed and attracted to gaming as well as they are able to gain immediate feedback on their performance, they should be easily engaged in them.

The baseball game (Han-Bin, 2009) implements an learning platform for students by integrating a quiz in virtual baseball play. Students can answer questions to get higher possibilities to win the game. The higher percentage they made right decisions, the better performance can be made in the baseball game. By integrating authoring tools and gaming environment, students will be focused in the contents provided by teachers.

Also, we found some related approaches prior to 2008 based on technology rich environment, described in LQ 1.0 (Wang, 2008). Such as, the Schools Quiz (Boyes, 2007), Quiz game for Medical Students (Roubidoux et al., 2002), the TVREMOTE Framework (Bar et al., 2005), Classroom Presenter (Linnell et al., 2007), WIL/MA(Lab), ClassInHand (UNIV.), ClickPro (AclassTechnology). Only the first two cases are designed as games.

2.2 Criteria for the Game Design

Our lecture game concept intends to improve the non-engaging classroom teaching by collaborative gaming. And its design is based on the eight elements that make the games more fun to learn.

2.2.1 Collaborative Gaming for Learning

Today's Millennial students (Raines; Oblinger and Oblinger, 2005; D. Oblinger, 2003) have changing preferences for education and work environments

that negatively affect their enrolments and retention rates into university course programs. To better suit these preferences, and to improve the lecture's educational techniques, teaching methods and tools outside of the traditional lecture sessions and textbooks must be explored and implemented. Currently, both work on serious games and collaborative classrooms focus on this issue. The proposed lecture game concept deals with both serious games and student collaboration research, proposing that educational games with collaborative elements (multiplayer games) will take advantage of the benefits offered by each of these areas. The result is an educational game that demonstrates increased learning gains and student engagement above that of individual learning game experiences. Collaborative educational games and software also have the potential to solve many of the problems that collaborative work may pose to course instructors in terms of helping to regulate and evaluate student performance (Nickel and Barnes, 2010).

Currently, research into the combination of serious games and collaborative work (for example, collaborative, or multiplayer educational games) is an underexplored area, although recently, computer-supported collaborative learning (CSCL) researchers have begun investigation how games are designed to support effective and engaging collaboration between students. Studies on social interaction in online games like Second Life (Brown and Bell, 2004), or World of Warcraft (Bardzell et al., 2008; Nardi and Harris, 2006) reveal how multiplayer games allow players to use in-game objects to collaboratively create new activities around them, and how social interaction in the games is facilitated and evolving. From these studies, we learn how to create multi-player games that effectively support, or even require, collaboration between players.

Collaboration does not necessarily mean competition between teams, or otherwise an adversarial approach (Manninenand and Korva, 2005) in the virtual environment, like above online multiplayer games. In the real world, a goal that requires a collaborative process, like solving a puzzle does create a conflict in the form of the interaction within the game (C. Crawford, 1982), but it is not a contest amongst adversaries. The team has to cooperate to reach a common goal. Up until recently, the lack of proper means of communication and interaction has made it difficult to support collaboration in computer games, and there exist few actual true collaborative games on the marked. So we would like to explore this issue by a case study of using multiplayer quiz game in lecture to see what will happen when combining serious game with collaborative works in the physical world.

2.2.2 Characteristics of Good Educational Games

This section presents eight important characteristics of good educational games based on computer supported collaborative learning and Malone's statements of what makes games fun to learn. The following list of characteristics is extracted as a reference for people designing educational games, shown in Table 1. Note that missing one of the characteristics may not mean that the game will be unpopular or unsuccessful, but including the missing characteristics in the game concept may make it better.

Our lecture games concept, both in LQ 1.0 and 2.0 are designed based on these characteristics.

Table 1: Characteristics of good educational games.

ID	Educational Game elements	Explanation	Reference
1	Variable instructional control	How the difficulty is adjustable or adjusts to the skills of the player	(Thomas, 1980; Lowe and Holton., 2005)
2	Presence of instructional support	The possibility to give the player hints when he or she is incapable of solving a task	(Lowe and Holton., 2005; Privateer, 1999)
3	Necessary external support	The need for use of external support	(Lowe and Holton., 2005)
4	Inviting screen design	The feeling of playing a game and not operating a program	(Lowe and Holton., 2005)
5	Practice strategy	The possibility to practice the game without affecting the users score or status	(Lowe and Holton., 2005; Privateer, 1999)
6	Sound instructional principles	How well the user is taught how to use and play the game	(Lowe and Holton., 2005; Boocock and Coleman, 1966; J Kirriemuir and McFarlane, 2003; Schick, 1993)
7	Concept credibility	Abstracting the theory or skills to maintain integrity of the instruction	(Elder, 1973)
8	Inspiring game concept	Making the game inspiring and fun	(Thomas, 1980; Kirriemuir, 2004)

2.3 Lecture Quiz 1.0

The developed prototype of LQ 1.0 consisted of one main server, a teacher client and a student client. To begin a session the students had to download the student client to their phone using Wifi, Bluetooth or the mobile network (GPRS/EDGE/3G). After the download was finished, the software had to be installed before the students were ready to participate. This was seen as a bit of a cumbersome process. The teacher client was implemented in Java and used OpenGL to display graphics on a big screen.

The prototype implemented two game modes. In the first game mode, all the students answered a number of questions. Each question had its own time limit, and the students had to answer within that time. After all the students had given their answers, a screen with statistics was displayed providing information on how many students that answered on each option. At the end of the quiz, the teacher client displayed a high-score list.

The other game mode was named *"last man standing"*. The questions were asked in the same fashion as with the plain game mode, but if a student answered incorrectly, he or she was removed from the game. The game continued until only one student remained and was crowned as the winner.

One of the main drawbacks of LQ 1.0 was that it lacked a good architecture, making it hard to extend, modify and maintain. It also lacks good documentation, and there was not quiz editor to add a new quiz or a question. The teacher had to manually edit the data in the database. The time spent on downloading and installing the software on the students' devices also made it less interesting for regular use in lectures.

2.4 Improvements of Lecture Quiz

Firstly, LQ 2.0 was based on above design methods and lecture quiz concept. But according to previous experiences and students' feedback, we made some improvements on these aspects. Table 2 shows the additional functional requirements in LQ 2.0.

Table 2: List of added new functional requirements.

Functional requirement
A developer can extend the game with new game modes
A teacher can update the question through a quiz editor
A teacher can tag questions for easier reuse and grouping
A server should be able to run several quiz sessions at the same time

In addition to functional requirements, we defined some non-functional requirements for LQ 2.0 described as quality scenarios (Len Bass et al., 2003). Table 3, 4, and 5 shows three quality scenarios for modifiability respectively.

Table 3: Modifiability scenario 1.

M1 - Deploying a new game mode for a client	
Source of stimulus	Game mode developer
Stimulus	The game mode developer wants to deploy a new game mode for one of the Lecture Quiz clients or the server
Environment	Design time
Artefact	One of the Lecture Quiz clients or the game server
Response	A new game mode is deployed and should be ready for use
Response measure	The new game mode should be possible to be deployed in few hours

Table 4: Modifiability scenario 2.

M2 - Creating a new client	
Source of stimulus	Client developer
Stimulus	The client developer wants to create a new client for the Lecture Quiz game
Environment	Design time
Artefact	The Lecture Quiz service
Response	A new client supporting to play the Lecture Quiz game.
Response measure	The server communication part of the client should be complete within two days

Table 5: Modifiability scenario 3.

M3 - Adding support for a new database back-end	
Source of stimulus	Server developer
Stimulus	Server developer wants to add support for another database back-end
Environment	Design time
Artefact	The Lecture Quiz server
Response	A new option for database storage in the server
Response measure	The new back-end should be finished in two hours

Also we required including a guide explaining how to create a new game mode for the Lecture Quiz server as well as for the clients. Such a guide would make it possible for an external developer to create new game modes with minimal effort.

As the ability for further development and expansion of the Lecture Quiz framework was an important part of our work, we decided to include detailed information on how this could be done. This information was intended for new developers wanting to pick up the Lecture Quiz system and continue development on the many aspects of it.

3 IMPLEMENTATION

In this section we describe how we have implemented the architecture for LQ 2.0. The main component in this architecture is the Lecture Quiz Game Service. The clients are implemented as flexible components that are easy to extend and improve. Figure 1 gives the system overview.

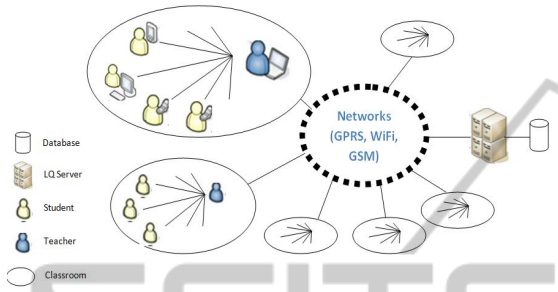

Figure 1: System overview of LQ 2.0.

3.1 Lecture Quiz Game Service

The Lecture Quiz Game Service is the server component that handles all the game logic. Both teacher and student clients connect to this server through its web service API. The server itself is implemented in Java EE 6 and was running on the Apache Tomcat application server during development, but should be able to run on any Java web container.

3.2 Database Design

The database design is given in Figure 2. Based on five main tables in the database, we have added two reference tables that help provide the needed relations between quizzes and questions, as well as the tags that could be as a new function for teachers to search certain questions.

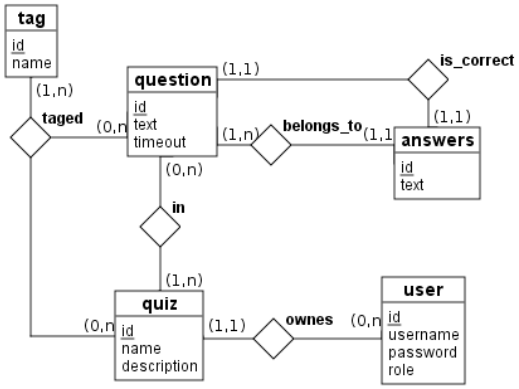

Figure 2: ER diagram of database.

3.3 Student Client

The student client was developed in Java using the Google Web Toolkit4 (GWT) and the AJAX5 framework. As with the teacher client, the main focus of this implementation has been on functionality and providing a reference as of how a client can be implemented. Hence, the graphical design is minimalistic that also fits the small screens and easy to download content for mobile phones.

3.4 Teacher Client

The teacher client is developed in Java SE 6. The development mainly focused on the functional parts of the client. Implemented in the teacher client is a simple menu system, a quiz editor to create and edit quizzes and questions, and a single game mode. When the teacher client is started, a connection check is performed to make sure the application can reach the Lecture Quiz web service. Figure 3 shows the interface of teacher clients.

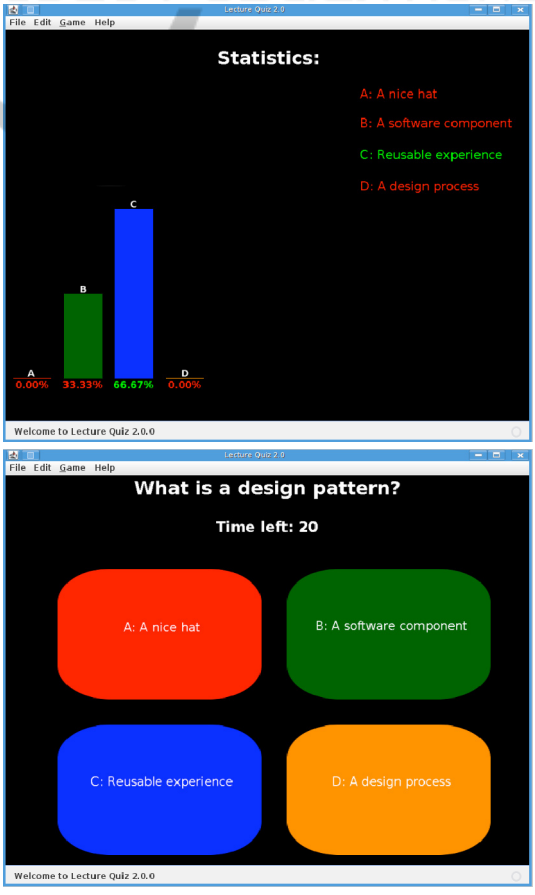

Figure 3: Screenshots from teacher client.

4 EVALUATION

In this section we will present an empirical experiment where our system was tried out in a realistic environment and the findings we found.

4.1 Experiment Delimitation

The goal of this experiment was to get an overall picture of how the Lecture Quiz service and clients worked in a real life setting, and comparing it to the similar experiment conducted for LQ1.0 in 2007 (Wang et al., 2007; Wang, 2008). We will point out and discuss trends based on these results and our experiences. Statistical analysis and thorough psychological analysis are out of the scope of the current aim.

4.2 Experiment Method

The goal of the formative evaluation was to assess engagement and usability of lecture quiz concept with a group of target users. The group of subjects included 21 students with average age of 22. The minimum number of participants was determined using the Nielsen and Landauer formula (Nielsen and Landauer, 1993) based on the probabilistic Poisson model:

$$\text{Uncovered problems} = N\,(1 - (1 - L)^{n})$$

Where: N is the total number of usability issues, L is the percentage of usability problems discovered when testing a single participant (the typical value is 31% when averaged across a large number of projects), and n is the number of subjects.

Nielsen argues that, for web applications, 15 users would find all usability problems and 5 participants would reveal 80% of the usability findings. Lewis (Nielsen and Landauer, 1993) supports Nielsen but notes that, for products with high usability, a sample of 10 or more participants is recommended. For this study it was determined that testing with 15 or more participants should provide meaningful results.

Usability and enjoyment of a game are two closely related concepts. According to the ISO 9241-11 (Jordan et al., 1996) definition, usability is derived from three independent measures: efficiency, effectiveness, and user satisfaction.

- **Effectiveness** - The ability of users to complete tasks using the system, and the quality of the output of those tasks
- **Efficiency** - The level of resource consumed in performing tasks
- **Satisfaction** - Users' subjective reactions to using the system.

Also, there are various methods to evaluate the usability. To measure usability we chose the System Usability Scale (SUS) (Jordan et al., 1996), which is a generic questionnaire with 10 questions for a simple indication of the system usability as a number on a scale from 0 to 100 points. Each question has a scale position from 1 to 5. For items 1,3,5,7 and 9, the score contribution is given by subtracting from the scale position. For item 2,4,6,8 and 10, the contribution is 5 minus the scale position. This implies that each question has a SUS contribution of 0-4 points. Finally, the sum of the scores are multiplied by 2,5 and divided by the number of replies to obtain the SUS score. The questionnaire is commonly used in a variety of research projects.

4.3 Experiment

This experiment tested the usability and functionality of LQ 2.0. The experiment took place on May 2010.

The purpose of this experiment was to collect empirical data about how well our prototype worked in a real life situation, especially regarding usability and functionality.

4.3.1 Participants and Environment

The experiment was conducted in a lecture in the Software Architecture course at our university, and all the participants were students taking this course. 21 students took part of this experiment, where 81% were male and 19% where female. As the test was conducted in a class of computer science students, most of the students consider themselves to be experienced computer users, but none of the participants had tried the software before the experiment. The test was lead by the teacher, and he controlled the progress of the game with the teacher client running on a laptop and displayed the quiz on a big screen by a video projector. The students used own mobile phones or laptops to participate through a web browser supporting java script. The Lecture Quiz server was running on a computer located outside of the lecture room.

4.3.2 Experiment Execution

21 of the students in class agreed to participate in the experiment. The lecture was a summary lecture in the Software Architecture course. In the first part of

the lecture, theory from current semester was summarized and discussed. The students were allowed to ask questions. The experiment took place in the second part of the lecture, after a short break.

The teacher client was started on a laptop, and an URL to the student client was shown on the projector. Each student logged in on the web client using a desired username and the quiz code displayed on the large screen processed by the teacher's client.

The experiment was executed without any problems. Everyone was able to answer the questions using their own mobile clients, and there was a relaxed atmosphere in the room. Some of the answer options made the students laugh a bit. In one of the questions, the teacher client was not able to display the statistics and correct answer. But this was displayed correctly on the student client. The problem was solved by the next question, and all the software seemed to handle this issue well. All of the 21 students that took part of the experiment did also answer the questionnaire.

4.4 Results and Findings

We will present our results mainly on two aspects: the usability and usefulness of the LQ 2.0.

4.4.1 SUS Score and Student Feedbacks

In this section we will present the results of the questionnaire. First we explain how the SUS score was calculated. Most of statements had five choices for the user to answer. From strongly disagree to strongly agree. These choices were displayed in the graphs as values from 1 to 5 respectively, and -1 means that the user did not answer this question.

To calculate our SUS score we had to discard 6 of the 21 returned questionnaires, as they had not answered all of the questions included in the SUS part of the questionnaire. That made it 15 valid questionnaires for our SUS calculation.

Our software got a SUS score of 84 out of 100. This is displayed in Table 6, and shows how Lecture Quiz scored on each question along with the results from LQ 1.0 (the previous experiment).

LQ 2.0's SUS score of 84 shows that it has high usability. The SUS score of the experiment in LQ 1.0 was 74.25. LQ 2.0 does mainly the same things from the students' aspect, except that the student client is web-based. Thus we conclude that the web-based approach was a success.

Also, if we look closely to the questions: 3, 4, 7, 10 from Table 6, it shows that LQ 2.0 has the scores of 4.53, 1.13, 4.73 and 1.27 respectively. We find the results relatively clear. The people that answered our questionnaire found LQ 2.0 both easy to use and easy to getting started with. All of these results are somewhat better compared to previous LQ 1.0. It shows *the system is easy to getting started with and use*.

This was an encouraging result, but we still had to face some negative feedback from students on the LQ 2.0 experiment. Some of the students commented that the graphical design of the software was not good. Many students complained that the answer buttons where to small, although this could be solved using the zoom function in their web browser. We are fully aware that we are not graphical designers, and that major improvements could be done on this area. But our main focus in this system was to get the technical issues on the back-end done right.

There were also some complains about the colour chosen as a background about option two on the teacher client. This colour was displayed differently from the projector than on a standard computer screen and this made the text almost unreadable. In the experiment, the teacher read out

Table 6: Lecture Quiz 1.0 and 2.0 SUS Scores.

	-	LQ 2.0		LQ 1.0	
ID	**Question**	**Avr**	**Score**	**Avr**	**Score**
1	I think that I would like to use this system frequently	3.53	3.53	3.6	2.6
2	I found the system unnecessarily complex	1.40	3.6	1.85	3.15
3	I thought the system was easy to use	4.53	3.53	4.02	3.05
4	I think that I would need support of a technical person to be able to use this system	1.13	3.87	1.35	3.65
5	I found the various functions in this system were well integrated	3.73	2.73	3.2	2.2
6	I thought there was too much inconsistency in this system	1.73	2.73	1.95	3.05
7	I would imagine that most people would learn to use this system very quickly	4.73	3.27	4.35	3.35
8	I found the system very cumbersome to use	1.73	3.27	1.95	3.05
9	I felt very confident using the system	4.33	3.33	3.55	2.55
10	I needed to learn a lot of things before I could get going with this system	1.27	3.73	1.95	3.05
--	**SUS score**		**84.00**		**74.25**

all the choices, so that all the students did get the information they needed. The colour problem was corrected after the experiment by choosing a darker background colour for the teacher client to improve readability.

From the teacher's perspective, LQ 2.0 was clearly an improvement over LQ 1.0 as the time to start a quiz was shorten dramatically and there were no technical issues the teacher had to attend. The teacher only needed to put an URL on the blackboard or on the large screen, and then let the students log into the system. This meant that Lecture Quiz did not introduce a break during the lecture.

4.4.2 Results from Usefulness Questions

Our questions and results regarding usefulness of using Lecture Quiz both in LQ 1.0 and 2.0 are shown in Table 7. In this part of the survey, we looked at the students' attitude towards the game compared to the previous version. We also had an open question part where the students could come with their comments.

From question 2 and 3 in Table 7, we can found that *most students did not find the system intrusive in the lecture*. Question 2 shows that most of the students (53%) thought they paid closer attention during the lecture because of the system. We find this as a positive result, as this was more evenly distributed in LQ 1.0. And question 3 shows that over 80% disagreed in some way that the system had a distracting effect on the lecture, where 60% strongly disagreed. This is a slightly better result than survey data from LQ 1.0, where 70% disagreed to this statement in some way. We guess that having the quiz at the end of the lecture, and not having to change lecture room as in 2007, may be factors changing this result.

From question 4 and 5, we found that *lecture quiz have positive effect to the learning*. Over half students agree that they learned more from the system and the lecture quiz at least do not have negative effective on learning compared to traditional lectures.

Also from question 6 we found that *the students found the system inspiring and fun*. From both surveys of LQ 1.0 and LQ 2.0, we see a clear trend that students (over 90%) think using the lecture quiz system in lectures make them more fun.

From question 7 in the LQ 1.0 survey, the majority thought that regular use of the system would make them attend more lectures. But in LQ 2.0 survey, the distribution of answers was more even. We guess there are more factors that affect the attendance rate, and maybe game factor is not the biggest one. This proves that more research is necessary before we can make a valid result on this question.

From question 9, we found that *the system worked as it should*. Out of the 21 returned questionnaires, 18 reported that the software worked as it should on their devices. One did not answer, one meant that the

Table 7: Usefulness questions.

ID	Question	Strongly disagree	Disagree	Neutral	Agree	Strongly agree	Version
1	I think that I am an experienced computer user	-	-	-	-	-	LQ 1.0
		0	0	5%	19%	76%	LQ 2.0
2	I think I paid closer attention during the lecture because of the system	10%	10%	30%	40%	10%	LQ 1.0
		5%	0	42%	32%	21%	LQ 2.0
3	I found the system had a distracting effect on the lecture	35%	35%	15%	10%	5%	LQ 1.0
		60%	25%	5%	5%	5%	LQ 2.0
4	I found the system made me learn more	5%	5%	40%	50%	0	LQ 1.0
		0	15%	25%	50%	10%	LQ 2.0
5	I think I learn more during a traditional lecture	5%	55%	25%	10%	5%	LQ 1.0
		15%	25%	40%	15%	5%	LQ 2.0
6	I found the system made the lecture more fun	0	0	5%	35%	60%	LQ 1.0
		0	0	10%	30%	60%	LQ 2.0
7	I think regular use of the system will make me attend more Lectures	15%	0	15%	45%	25%	LQ 1.0
		10%	15%	30%	20%	25%	LQ 2.0
8	I feel reluctant to pay 0.5 NOK in data transmission fee per lecture to participate in using the system	35%	15%	30%	10%	10%	LQ 1.0
		20%	25%	5%	20%	30%	LQ 2.0
-	LQ 2.0 question	Yes	No	If no, please describe the problem			
9	Did the client software work properly on your mobile/laptop?	90%	10%	Totally we got 20 responses, only two have problems.			

software did not work because of the problem with small buttons in the mobile screen; this could be solved when he zoomed in the mobile browser, and one complained that the software did not work in Opera Mini. The reason for the problem in Opera Mini is that LQ 2.0 is based on AJAX and therefore needs java script support in the browser. In Opera Mini the requests are compressed and handled on a central server before being sent to the mobile device, and thus java scripts do not work. And this student switched the browser before the formal experiment starts.

During the experiment the teacher client failed to show the statistics for one of the questions once. But the statistics where displayed correctly on all the student clients, and all answers was stored as they should. The quiz continued as usual when the teacher pressed the button to start the next question. This is only a minor bug in the teacher client and that the rest of the system works as expected. We were not able to reproduce this bug later.

As a whole, we had less technical problems than the comparable experiment in 2007, thus probably resulting in the users to be friendlier in their evaluation of the system. The results of this experiment are mostly positive and in most areas better than for the previous version of the system.

5 CONCLUSIONS

Through the data from the evaluation and by comparing with the first version of lecture quiz, we found that lecture quiz is a suitable game concept to be used in lecture from both evaluation data.

And LQ 2.0 improved lecture game quiz concept in several ways. The main feature of building a strong and easily modifiable web-based architecture is extendable game modes, the ability to run multiple game servers on the same database and run many different quiz sessions on the same server. The new student web-based client reaches more students as close to 100% of students have access to a web-browser using a laptop or a mobile phone. In addition, the quiz editor makes it easy for teachers to maintain the question database, and it is easy to extend the game with the new game modes through the architecture. All of these features can be the factors that made the survey and evaluation better than the last version in most of aspects. More elaborate experiments must be conducted to find whether Lecture Quiz improves how much the students actually learn.

REFERENCES

Raines, C. Managing Millennials; [cited October, 2010]. Available from: https://www.cpcc.edu/millennial/presentations-workshops/faculty-or-all-college-workshop/9%20-%20managing%20millennials.doc

Oblinger, D.; Oblinger, J. Educating the Net Generation. Boulder, CO: Educause. 2005.

D. Oblinger. Boomers, Gen-Xers, and Millennials: Understanding the New Students. Educause Review, July/August 2003; vol. 38, no. 4:pp. 37-47.

Williams, L.; Layman, L.; Slaten, K. M.; Berenson, S. B.; Seaman, C. On the Impact of a Collaborative Pedagogy on African American Millennial Students in Software Engineering. Proceedings of the 29th international conference on Software Engineering: IEEE Computer Society; 2007. p. 677-687.

Alf Inge Wang; Terje Øfsdahl; Mørch-Storstein., O. K. LECTURE QUIZ - A Mobile Game Concept for Lectures., In 11th IASTED International Conference on Software Engineering and Application (SEA 2007),: Acta Press; 2007. pages 128–142.

A.I. Wang, An Evaluation of a Mobile Game Concept for Lectures Software Engineering Education and Training, 2008 IEEE 21st Conference on; 2008. 197 p.

Schuh, L.; Burdette, D. E.; Schultz, L.; Silver, B. Learning Clinical Neurophysiology: Gaming is Better than Lectures. Journal of Clinical Neurophysiology. 2008; 25(3):167-169 110.1097/WNP.1090b1013e31817759b 31817753.

Landay, S. Online Learning 101: Part I: Authoring and Course Development Tools. eLearn. 2010; 2010(6).

Larraza-Mendiluze, E.; Garay-Vitoria, N. Changing the learning process of the input/output topic using a game in a portable console. Proceedings of the fifteenth annual conference on Innovation and technology in computer science education. Bilkent, Ankara, Turkey: ACM; 2010. p. 316-316.

Daloukas, V.; Dai, V.; Alikanioti, E.; Sirmakessis, S. The design of open source educational games for secondary schools. Proceedings of the 1st international conference on PErvasive Technologies Related to Assistive Environments. Athens, Greece: ACM; 2008. p. 1-6.

Han-Bin, C. Integrating baseball and quiz game to a learning platform. Pervasive Computing (JCPC), 2009 Joint Conferences on; 2009. p. 881-884.

Boyes, E. Buzz! spawns School Quiz.; [modified 2007]. Available from: http://www.gamespot.com/news/6164009.html

Roubidoux, M.A.; Chapman, C. M.; Piontek, M. E. Development and Evaluation of an Interactive Web-Based Breast Imaging Game for Medical Students. Academic Radiology. 2002; 9(10):1169-1178.

Bar H.; Tews, E.; G. Robling. Improving Feedback and Classroom Interaction Using Mobile Phones. In Proceedings of Mobile Learning; 2005. p. 55-62.

Linnell, N.; Anderson, R.; Fridley, J.; Hinckley, T.; Razmov, V. Supporting classroom discussion with technology: A case study in environmental science.

Frontiers In Education Conference - Global Engineering: Knowledge Without Borders, Opportunities Without Passports, 2007 FIE '07 37th Annual; 2007. p. F1D-4-F1D-9.

Lab, L. Wireless Interactive Lecture in Manheim: UCE Servers & Clients, Lecture Lab. WIL/MA; [cited 2007]. Available from: http://www.lecturelab.de/

UNIV., I. A. W. F. ClassInHand: Wake Forest University; [cited 2007]. Available from: http://classinhand.wfu.edu

AclassTechnology. EduClick-Overview; [cited 2010]. Available from: http://www.aclasstechnology.co.uk/eduClick/index.html

Nickel, A.; Barnes, T. Games for CS education: computer-supported collaborative learning and multiplayer games. Proceedings of the Fifth International Conference on the Foundations of Digital Games. Monterey, California: ACM; 2010. p. 274-276.

Brown, B.; Bell, M. CSCW at play: there as a collaborative virtual environment. Proceedings of the 2004 ACM conference on Computer supported cooperative work. Chicago, Illinois, USA: ACM; 2004. p. 350-359.

Bardzell, S.; Bardzell, J.; Pace, T.; Reed, K. Blissfully productive: grouping and cooperation in world of warcraft instance runs. Proceedings of the 2008 ACM conference on Computer supported cooperative work. San Diego, CA, USA: ACM; 2008. p. 357-360.

Nardi, B.; Harris, J. Strangers and friends: collaborative play in world of warcraft. Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work. Banff, Alberta, Canada: ACM; 2006. p. 149-158.

T. Manninenand; T. Korva. Designing Puzzle for Collaborative Gaming Experience - CASE: eScape. DiGRA 2005 Conference: Changing Views - Words in play 2005. Vancouver, Canada, 2005.

C. Crawford. The Art of Computer Game Design: Osborne/McGraw Hill; 1982.

Thomas, W. M. What makes things fun to learn? heuristics for designing instructional computer games. Proceedings of the 3rd ACM SIGSMALL symposium and the first SIGPC symposium on Small systems. Palo Alto, California, United States: ACM; 1980.

Lowe, J. S.; Holton., E. F. A Theory of Effective Computer-Based Instruction for Adults. Human Resource Development Review. 2005:4(2), p159-188.

Privateer, P.M. Academic Technology and the Future of Higher Education: Strategic Paths Taken and Not Taken. Journal of Higher Education, Vol 70. 1999.

Boocock, S. S.; Coleman, J. S. Games with Simulated Environments in Learning. Sociology of Education. 1966; 39(3):215-236.

J. Kirriemuir; McFarlane, A. Use of computer and video games in the classroom. Proceedings of the Level Up Digital Games Research Conference. Universiteit Utrecht, Netherlands.; 2003.

Schick, J. B. M. The Decision to Use a Computer Simulation. Vol. Vol. 27, No. 1 (Nov., 1993), pp. 27-36 Society for History Education; 1993.

Elder, C. D. Problems in the Structure and Use of Educational Simulation. Sociology of Education. 1973;46(3):p.335-354.

Kirriemuir, J. M., A. Literature review in games and learning. Report 8; 2004.

Len Bass; Paul Clements; Kazman, R. Software architecture in practice: Second Edition: Addison-Wesley Professional; 2003.

Nielsen, J.; Landauer, T. K. A mathematical model of the finding of usability problems. Proceedings of the INTERACT '93 and CHI '93 conference on Human factors in computing systems. Amsterdam, The Netherlands: ACM; 1993. p. 206-213.

Jordan, P. W.; Thomas, B.; Weerdmeester, B. A.; McClelland, A.L. Usability Evaluation in Industry, chapter SUS - A quick and dirty usability scale: CRC Press; 1996. pages 189–194.