# LECTURE QUIZ - A MOBILE GAME CONCEPT FOR LECTURES

Alf Inge Wang, Terje Øfsdahl and Ole Kristian Mørch-Storstein
Dept. of Computer and Information Science
Norwegian University of Science and Technology
N-7491 Trondheim, Norway
alfw / ofsdahl / olekrst@idi.ntnu.no

**ABSTRACT**
This paper describes a new mobile game application, Lecture Quiz, which can be used within lectures in higher education to promote strong student participation and enable variation in how lectures are taught. The new idea is to provide a lecture game using the devices and infrastructure already available in lecture halls like the teacher's portable PC, a large screen and a video projector, network connections, and the students' mobile phones. The architecture of the game application consists of a server, a client for the students and a client for the teacher. The game is a multiplayer quiz game with a variation in game modes where an unlimited number of players can play simultaneously. The main contribution of this paper is a description of a novel game concept for lectures and an evaluation of the application in use. The paper also reports technology choices we made when implementing the game and how we solved the problem of variation in latency between the server and various mobile clients running on different wireless networks.

**KEY WORDS**
Mobile and Wireless Computing, Education, Communication Networks, and Multimedia Systems.

## 1 Introduction

Video games have the last couple of years become more and more used within schools to teach children mathematics, geography, English and other subjects. Experiences show that video games can be effective and compelling context for children's learning [15, 9, 8]. The usual way of integrating video games in schools is to let the children practice on certain skills in a computer lab supervised by a teacher. Previous research shows that it can even be beneficial for academic achievement, motivation and classroom dynamics to use video games within a classroom [17]. It is also believed that the teaching methods based on educational games are not only attractive to schoolchildren, but also to university students [18]. Research on video games for higher education has been done before, e.g. [3, 14, 13], but we believe there is an untapped potential that needs to be explored. Research within this domain should focus on two main areas: development of new game concepts and game technologies for higher education, and evaluation of these concepts and technologies. A video game can be integrated mainly in two ways with a university course. *First*, video games can be used instead of traditional exercises. This approach should motivate the students to put extra effort into the exercises and gives the teachers an opportunity to monitor the students when they are doing the exercises. *Second*, video games can be used within lectures to improve the participation and motivation of students. This paper focuses on the latter, and describes the game Lecture Quiz – being a game that is easy to integrate with lectures.

In the initial phase of the design of a new game concept for lectures, we considered the existing technological infrastructure already available in lecture halls. All lecture halls at the Norwegian University of Science and Technology (NTNU) are equipped with a stationary PC for the lecturer connected to local monitor and a video projector displaying an image at a large screen (various sizes depending on the size of the lecture hall). There are also connections to connect a private laptop to the video projector, and there are both wired and wireless network (WiFi) connections available. In addition, close to 100% of our students own a mobile phone where most of these phones can run Java 2 Micro Edition (J2ME) applications.

The goal of developing a lecture game was to motivate students, make students learn, and to improve student participation in lectures. Thus, it was important that as many students as possible could participate in the game simultaneously. We therefore decided to include the students' mobile phones into the game concept along with the equipment already available in the lecture halls. The game concept that we developed was a variant of the existing Sony Playstation 2 quiz-game series Buzz! [1]. The main difference is that in our game we allow unlimited number of players and that the players will use their personal mobile phones as game controllers.

Game-based learning in mobile and ubiquitous learning environments has gained interest in recent years, e.g. [16]. Studies show that wireless technology used in an educational setting can increase mental activity [10], facilitate interactivity [19], and promote social interaction [6]. This paper will focus on the game concept, the software architecture, and the technical aspects of the lecture game concept. The main contribution in this paper is a presentation of a novel game concept that can be used in any lectures, experiences from actual usage of the game concept, and

some technical considerations.

The rest of the paper is organised as follows. Section 2 describes and motivates for the game concept used in Lecture Quiz. Section 3 describes the software architecture and the technical considerations made when implementing such systems. Section 4 describes an evaluation of Lecture Quiz with emphasis on data communication. Section 5 describes similar approaches, and Section 6 concludes the paper.

## 2 The Game Concept

In this section we will present the motivation and a description of game concept of Lecture Quiz as illustrated in Figure 1 featuring student participation and interaction.

### 2.1 Choice of game concept

In our search for a game concept for lectures, we found a large number of projects and products of varying success. The most important consideration when designing a game is to make the game captivating and interesting to use. In addition, for games used in education, the players must learn something useful from playing. Malone has listed three main characteristics that makes things fun to learn: they should provide the appropriate level of challenge, they should use fantasy and abstractions to make the game more interesting, and they should trigger the player's curiosity [12]. Quiz games are likely to trigger player's curiosity as well as provide a challenge for the player, but do not provide much fantasy for the player. However, we believe that the lack of fantasy can partly compensated if the game is a multiplayer game where the social interaction becomes an important motivating factor for playing. We also considered the social interaction to be important for games used in lectures, to promote more active students in general. In a quiz game, the level of the challenge is adjusted by the questions asked, and the alternatives provided. Thus, the challenge level can be adjusted to the right level for most participants but not tailored for the individual differences. The main benefit of choosing a quiz game concept is that such games can be used in any course independent of the subjects being taught. Quiz games are most useful for testing and rehearsing theory, and are less useful for testing skills like doing calculation in Mathematics or similar (although, it can also be used for such purposes).

### 2.2 Game Modes

Our architecture (see Section 3) allows the design of multiple game modes to facilitate various multiplayer quiz variations similar to game modes available in games like Buzz! [1]. In our current implementation we have developed two game modes with names reflecting how the students performance is presented:

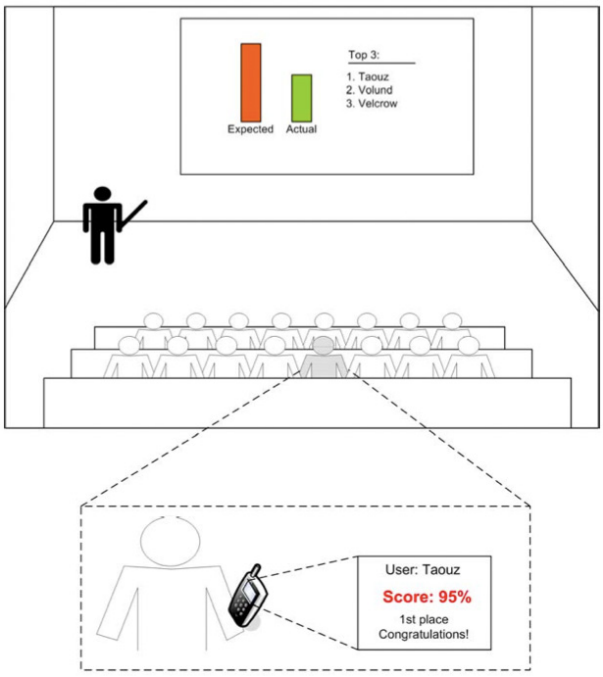


Figure 1. Illustration of the Game Concept

**Score distribution:** In this game mode, the presentation of the students answers distributed on the various alternatives is shown in an animated 3D bar diagram and the correct answer is highlighted as shown in Figure 2a). The graphics is scaled in size (both the bars and the text), enabling a variation in number of alternatives.

**Last man standing:** In this game mode, the players have to answer correct to play in the next round. After each question, all the players that have participated in the round are displayed on the screen as avatars with a nickname beneath, see Figure 2b). The players that have answered incorrectly get shot and will not make it to the next round. The goal of this game mode is to be the last man standing as shown in Figure 2c). Also for this mode, the graphics automatically scales the avatars and the text so they will be larger when few players are left (see Figure 2 b and c).

## 3 The Software Architecture

The Lecture Quiz game has been designed using the well-known client server architectural pattern. A deployment view of the Lecture Quiz game is shown in Figure 3. The system consists of one server and one teacher client and one student client. The server manages a database of questions and alternatives, tracking game scores, manages game modes, and routing messages between the server and the clients. The server program is accessed by the clients over the network using TCP/IP and is typically run one of the

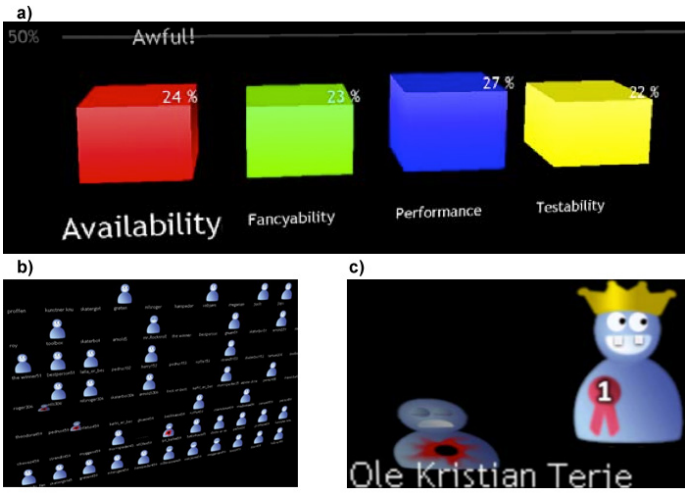


Figure 2. Lecture Quiz game modes

University's web servers. The teacher's game client is used to initiate and control the game using a keyboard, and is used to visualise the performance of the students. This client is usually run on the teacher's laptop PC. For the student client, the Lecture Quiz system support two alternatives – one is running on a mobile phone and one on a laptop PC. The only difference between these two variants is that the PC client runs an applet to wrap the mobile application enabling students with laptop PCs to run game clients from their machines.

### 3.1 Technology Choices

The server application of the system was implemented in Java SE 6 and MySQL well integrated with an Apache Web server, to enable cross-platform portability and easy integration with the rest of the server applications running at our department. For the mobile clients, we had to make a choice between three implementation approaches: 1) native applications, 2) .NET Compact Framework (CF) from Microsoft, 3) Java 2 Micro Edition (J2ME) from Sun Microsystems. To make a choice, we considered what the mobile devices of the students could run. Only a few students have mobile phones or PDAs that are able to run .NET CF. Further, only few students have mobile devices that allow native applications. Thus, our choice was to use J2ME that made it possible to run the client on all kinds of devices including PCs using a Java applet.

It was harder to choose the implementation platform for the teacher client application, as it was important to that this client application could produce nice graphics and animations. Our choice was between two alternatives: 1) using C++ or C# in combination with Microsoft's DirectX or 2) using Java SE in combination with OpenGL. The first alternative is an attractive choice with good development environments, high-speed graphics and good developer support, but it lacks of portability across operating systems. The second alternative was chosen, as many lectures run other operating systems than Microsoft Windows such as Linux and Mac OS. The main disadvantage choosing the Java and Open GL was a potential loss in performance.

### 3.2 Network Considerations

Our architecture is independent of the type of network used in the system. However, we recommend using a wired connection between the teacher game client and the server to ensure stability and high transfer speed. Between the student game client and the server, a wide variety of wireless network technologies are available: GSM, GPRS, EDGE, 3G, WiFi and Bluetooth. Of these networks, GSM is not recommended because of slow transfer speed and long delays in establishing connections. We also considered providing support for Bluetooth connections directly between the server and the mobile devices in the lecture hall. This approach was abandoned, as it has limited the number of players that could participate. Even when using a Bluetooth hub, the maximum number of players would be maximum 21. From experiences from another project [20], we know that a Bluetooth connection with 21 clients can become slow, has a long connection time and is unstable. Thus, we chose to do all communication between the student game client and the server over the telecom networks or the local WiFi infrastructure. The main disadvantage with using the telecom networks is that the students must pay the telecom providers a small fee using their network to transfer data to play the games during the lectures (about 30 cents per game). As more mobile phones also support WiFi, this problem might be less of a problem in the future.

## 4 Evaluation of the Lecture Quiz

The Lecture Quiz game was implemented by two master students (the second and third author) in four month. This section describes an evaluation, the first of experiences from usage, and some issues related to communication.

### 4.1 Experiences from Using Lecture Quiz

The Lecture Quiz game was used in a summary lecture in the course Software Architecture 8th of May 2007, as well being tested by several students not being part of a particular lecture. Lecture Quiz has received good reviews from the students on concept, engagement and playability. For the teacher the game concept was easy to integrate with the lecture, as the PowerPoint and a video projector was used in the rest of the lecture.

In the test in the software architecture lecture, the students were told in the beginning that they were going to play a new quiz game in the last part of the lecture. We observed, that the students were very concentrated during the

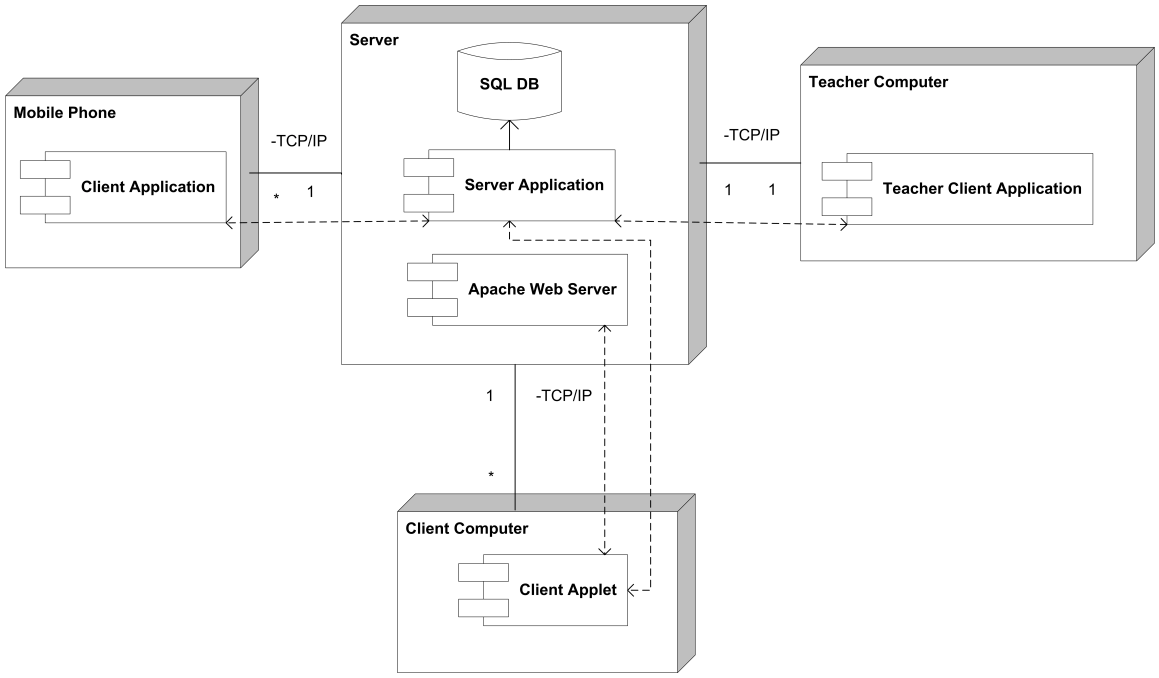


Figure 3. Deployment view of the Lecture Quiz game

presentation of the power point slides in order to do their best in the game. The two game modes were used, starting with *Score distribution* and continuing with *Last man standing*. The same questions were used for both game modes, thus giving the students the opportunities to learn from the first game round. The students said after the lecture that they thought this approach was useful both for learning and for motivating students to show up in lectures.

### 4.2 Issues Related to Data Communication

From the start of the Lecture Quiz project, we were aware of initial latencies of up to 6-10 seconds for TCP/IP packages sent over GPRS networks [5]. As an important part of the gameplay is synchronised by a countdown timer (e.g., 20 seconds to pick an answer), this unpredictable latency would make the game unfair and cumbersome to use. Thus, we had to find ways to minimize the perceived delay for the player.

From tests we ran, we discovered that the initial latency differed considerably between the mobile network providers. At times, the initial delay of some providers was several seconds longer than others. However, this was only a problem for the first packet sent after a longer delay. The average latency could be drastically reduced by sending small amounts of data directly after the mobile client returned an answer. E.g., for one telecom provider the latency was reduced from 1828 to 344 milliseconds, by transmitting an extra data packet.

Our solution to synchronise the mobile clients was to send one packet of data (dummy or real) every 2 second and at the same time introduce an intentional delay of 2 seconds before the clients could start to answer. The introduced delay was camouflaged in the Teacher client by playing a sound of a bell and displaying the countdown timer before the question was displayed (2 seconds and 300 milliseconds) after the dummy package was sent to the clients.

The solution to this problem is illustrated in Figure 4. $\Delta Normal$ illustrates the normal difference in delay for two mobile clients to be ready to receive answers from the players, $\Delta wait$ is the introduced delay to cover the differences in latency between clients and $\Delta Optimized$ is the optimised difference in delay after using the solution described above.

Our solution reduced the observed delay on the mobile clients from maximum 7-8 seconds on the slowest telecom providers to approximately 300 milliseconds. A difference of 300 milliseconds will not affect the gameplay and fairness of the game.

## 5 Related Work

As far as we know, there are no papers describing exactly the same game concept using the technical infrastructure in lecture halls. However, there are some related approaches that we will describe in this section.

In [4], the TVREMOTE Framework is presented. The framework was designed to allow hundreds of students to participate and give feedback during lectures using their

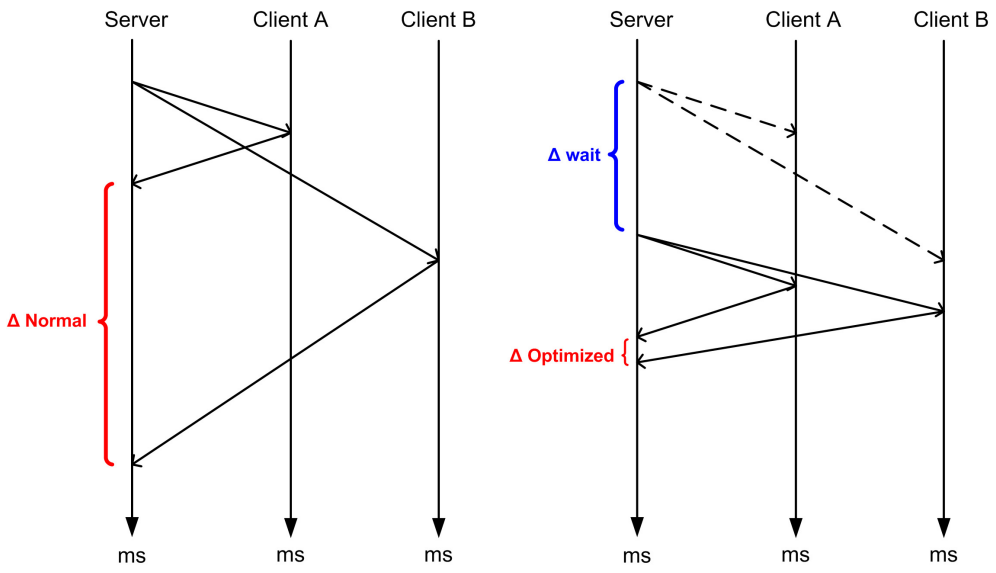


Figure 4. Synchronisation between server and clients

mobile phones over the GPRS network. The system features polling of students' opinions and electronic question submission. The teacher can also broadcast links and notes, as well as multiple-choice questions. The teacher collects the feedback and reads it from a private display. The main motivation for this tool is for the teacher to get a statistical distribution of correct and incorrect answers. Studies using the TVREMOTE show that students generally appreciate a short explanation as to why a given answer is correct.

Linnell et al. describes Classroom Presenter being a Microsoft PowerPoint plug-in developed at the University of Washington [11]. This plug-in allows students to write comments directly onto the digital slides with a stylus or a keyboard input using a tablet PC handled out at the beginning of the lecture. Exercises are broadcasted to the tablet PCs, and the students write their answers onto a blank space of the slide. The teacher can then browse through the replies.

WIL/MA is a tool developed at the University of Mannheim and is a Java implementation of a digital hand raising, spontaneous student comments and multiple-choice questionnaires [21]. The clients run on PDAs and a WiFi connection is used to transfer data. The teacher obtains the students responses on his private screen on a PC.

ClassInHand is another approach developed at Wake Forest University, and features a presentation controller, real time quiz and student/teacher interaction [22]. The students' clients run on Windows Mobile PDAs and the teacher is also running his/her client on a PDA. The teacher's client is used for information retrieval and this information can later be presented to the students independently of the ClassInHand system.

ClickPro is a commercial classroom polling application developed by Avrio Ideas for teaching in elementary school [2]. The system is based on infrared technology, giving the students special remote controls to submit the answers of multiple-choice questions. The teacher controls the application on a PC and the answers are displayed using a TV or a video projector. Each student is assigned a number that is displayed on the screen so the student can see if his answer is correct or not.

Buzz was originally conceived as commercial trivial game for Sony Playstation 2. With government funding, a new version of the game named *Buzz! The School quiz* was designed especially as a tutoring tool [7]. The game will be marketed towards education institutions and comes with content covering Stage 2 National Curriculum for private schools in the UK. A new feature, *create a quiz* is included to allow tutors to hold revision exercises on a given subject. The game will use the Playstation 2 platform and wired special purpose buzz-controllers that allow up to maximum 8 players.

Apart from Buzz! The School quiz, none of the systems above was intentionally designed as a game. Many of the systems provide functionality for running quiz, but the main focus is for the teacher to get the knowledge level of the students, as the result is not presented for the students. The ClickPro system has many similarities to Lecture Quiz, but was neither designed as a game e.g., including different game modes. In addition, ClickPro does not use the available technical infrastructure in lecture halls and requires special purpose devices to use the system. Although, Buzz! The School quiz is a game, this is a game tailored for a particular syllabus and cannot be used to support lectures in general. In addition, the game requires special controllers and a game consol to run.

## 6 Conclusion

In this paper we have presented a new game concept that can easily be integrated with lectures. The game is a multiplayer game that utilise the equipment and infrastructure available in lecture halls like the teacher's PC, network connections, video projector and the students' mobile phones. The game has been used with success in a software architecture lecture, and the students claimed that they learned from playing the game and those such games would make it more likely that they will follow lectures. The mobile game clients communicate with the server using wireless telecom networks like GPRS, EDGE and 3G. We found that when transferring the first packages over the wireless network, it could take up to 6-10 seconds before the data was sent. However, when the first package was sent, there were only small delays sending another package within 2 seconds. We solved the problem of synchronising the mobile clients by sending dummy or real data packages every 2 seconds and introducing a 2 seconds delay before the clients could start to answer. This reduced the difference in delay between various mobile clients and networks from maximum 7-8 seconds to approximately 300 milliseconds.

We will continue to explore the area of lecture games by creating new game modes in Lecture Quiz, evaluating the use of Lecture Quiz in real lectures, and designing and implementing other new game concepts that can be used in lectures.

## References


[1] Sony Playstation 2. Buzz! Web: http://www.buzzthegame.com, 2008.

[2] Aclass Technology. EduClick - Overview. Web: http://www.aclasstechnology.co.uk/eduClick/index.html, 2007.

[3] Alex Baker, Emily Oh Navarro, and André van der Hoek. Problems and Programmers: an Educational Software Engineering Card Game. In *ICSE '03: Proceedings of the 25th International Conference on Software Engineering*, pages 614–619, Washington, DC, USA, 2003. IEEE Computer Society.

[4] Henning Bär, Erik Tews, and Guido Rößling. Improving Feedback and Classroom Interaction Using Mobile Phones. In *Proceedings of* Mobile Learning 2005, pages 55–62, 2005.

[5] A. Calveras, J. Paradells, C. Gomez, M. Catalan, and J. C. Valles. Optimizing TCP/IP parameters over GPRS and WLAN networks. In *Vehicular Technology Conference*, volume 4, pages 2569–2573. IEEE, 2003.

[6] Camila Cortez, Miguel Nussbaum, Raul Santelices, Patricio Rodriguez, Gustavo Zurita, Monica Correa, and Rafael Cautivo. Teaching Science with Mobile Computer Supported Collaborative Learning (MCSCL). In *WMTE '04: Proceedings of the 2nd IEEE International Workshop on Wireless and Mobile Technologies in Education (WMTE'04)*, pages 67–74, Washington, DC, USA, 2004. IEEE Computer Society.

[7] Emma Boyes. Buzz! spawns School Quiz. Web: http://www.gamespot.com/news/6164009.html, 2007.

[8] James Paul Gee. *What video games have to teach us about learning and literacy*. Palgrave-MacMillan, New York, NY, 2003.

[9] Henry Jenkins. How should we teach kids Newtonian physics? Simple. Play computer games. *Technology Review*, March 2002.

[10] Susana Juniu. Implementing Handheld Computing Technology in Physical Education. *Physical Education, Recreation & Dance*, 73, 2002.

[11] Natalie Linnell, Richard Anderson, Jim Fridley, Tom Hinckley, and Valentin Razmov. Supporting Classroom Discussion with Technology: A Case Study in Environmental Science. To Appear ASEE/IEEE Frontiers in Education, 2007.

[12] Thomas W. Malone. What makes things fun to learn? Heuristics for designing instructional computer games. In *SIGSMALL '80: Proceedings of the 3rd ACM SIGSMALL symposium and the first SIGPC symposium on Small systems*, pages 162–169, New York, NY, USA, 1980. ACM Press.

[13] Lasse Natvig, Steinar Line, and Asbjørn Djupdal. Age of Computers: An Innovative Combination of History and Computer Game Elements for Teaching Computer Fundamentals. In *FIE 2004: Proceedings of the 2004 Frontiers in Education Conference*, 2004.

[14] Emily Oh Navarro and André van der Hoek. SimSE: an Educational Simulation Game for Teaching the Software Engineering Process. In *ITiCSE '04: Proceedings of the 9th annual SIGCSE conference on Innovation and technology in computer science education*, pages 233–233, New York, NY, USA, 2004. ACM Press.

[15] Seymour Papert. Does Easy Do It? Children, Games, and Learning. *Game Developer Magazine*, page 88, June 1998.

[16] Marc Prensky. *Digital GameBased Learning*. McGraw-Hill, 2001.

[17] Ricardo Rosas, Miguel Nussbaum, Patricio Cumsille, Vladimir Marianov, Monica Correa, Patricia Flores, Valeska Grau, Francisca Lagos, Ximena Lopez, Veronica Lopez, Patricio Rodriguez, and Marcela Salinas. Beyond Nintendo: design and assessment of educational video games for first and second grade students. *Computers & Education*, 40(1):71–94, 2003.

[18] Mike Sharples. The design of personal mobile technologies for lifelong learning. *Computer & Education*, 34(3-4):177–193, 2000.

[19] Elliot Soloway, Cathleen Norris, Phyllis Blumenfeld, Barry Fishman, Joseph Krajcik, and Ron Marx. Log on education: Handheld devices are ready-at-hand. *Communcation of the ACM*, 44(6):15–20, 2001.

[20] TellU and NTNU. MOOSES: Multiplayer On One Screen Entertainment System. Web: http://www.mooses.no, 2007.

[21] UCE Servers & Clients, Lecture Lab. WIL/MA, Wireless Interactive Lecture in Manheim. Web: http://www.lecturelab.de/, 2007.

[22] Wake Forest University. ClassInHand. Web: http://classinhand.wfu.edu, 2007.