ICWL 2012

The 11th International Conference on Web-based Learning

2-4 September 2012
Sinaia, Romania

- Local Workshop Proceedings -

Edited By:

Dickson Chiu

Elvira Popescu

Qing Li

University of Craiova

Faculty of Automation, Computers and Electronics

Hong Kong Web Society

Table of Contents

i

CCSTED 2012

**1st International Workshop on Creative Collaboration through Supportive Technologies in
Education**
Organizers:

- **Gabriel Gorghiu (Valahia University Targoviste, Romania)**
- **Mikhail Fominykh (Norwegian University of Science and Technology, Trondheim,
  Norway)**
- **Tommi Inkilä (University of Oulu, Finland)**
- **David Lamas (Tallinn University, Estonia)**
- **Margarida Romero (Fundació per a la Universitat Oberta de Catalunya, Spain)**

Note. The papers included in these local proceedings will be revised by the authors and then published as post-proceedings volume in Springer Lecture Notes in Computer Science (LNCS).

Developing Online Collaborative Games for e-Learning Environments

Traian Anghel[1], Adrian Florea[1], Arpad Gellert[1], Delilah Florea[2]

[1] "Lucian Blaga" University of Sibiu, Computer Science and Electrical Engineering Department
`tanghel@live.com`, `{adrian.florea,arpad.gellert}@ulbsibiu.ro`
[2] "Samuel von Brukenthal" National College of Sibiu
`delilah_florea@yahoo.com`

Abstract. Based on our experience, we believe that games, competition and teamwork offer a pleasant and active way of learning. This is much more efficient when the learner has a smile on his face, when he is astonished and curious about next levels and finds the game sufficiently challenging and fun to try again. Our application proposal has the purpose of implementing an e-Learning platform for improving the teaching and learning process in somewhat abstract domains, such as computer architecture or object oriented programming, with the help of games. These games are time-dependent and are able to support collaboration between groups. To this date there are two learning games implemented: a crossword puzzle and a collaborative jigsaw puzzle, the last one supporting multiplayer mode for up to 16 simultaneous players, being simple, fast, fun and reliable. Our application allows geographically distributed students to concurrently and collaboratively play the same game.

Keywords: e-Learning system, collaborative tool, online games.

1   Introduction

The explosion of communication devices and pervasive computing systems in everyday life due to technological development in ICT, led for the young generation to the acquisition of skills in using computers, laptops, tablets, cell phones, Internet or social networks without too much effort and generally outside the classroom. In this context, the tendency of so called "digital" or "online generation" is not having the patience to follow a course, read a book (even electronic) but rather "play or practice instead of learning theory". A very important challenge in the teaching process is to keep students motivated throughout the course, and to help prepare outside the class hours. Learning through play, the desire to win, forcing him to pay attention to rules, to retain them, motivates him to stay focused on the studied topic. Competition and teamwork are also motivational for students.

There is one thing that teachers and learners have to agree upon: learning is boring, because of age and mentality differences [14]. Adults show commitment and disposition to learn, being aware of the need to be prepared throughout life (long life learning concept). Youngsters, however, want to acquire knowledge and technical skills with-

81

out too much effort. In education, as well as corporate facilities, trainings are not always seen as fun. All of this can be changed if games are implemented in the learning process. The learner's attention is kept awake by allowing him to interact with a virtual environment where he can test his skills, is allowed to restart the game at any time and, most important of all, is allowed to make mistakes (unlike real-life situations where mistakes are not desirable). Games are usually highly intuitive, user-centered; they have entered our social lives and support massive amounts of concurrent players (such as World of Warcraft, or, more recently, Minecraft).

Our application proposal has the purpose of implementing an e-Learning platform – Education Management Tool (EDM) – for improving the teaching and learning process in somewhat abstract domains, such as computer architecture, object oriented programming, operating systems, algorithms and data structures, with the help of games. These games are time-dependent and are able to support collaboration between groups. To this date there are two learning games implemented: a crossword puzzle and a collaborative jigsaw puzzle, the last one supporting multiplayer mode for up to 16 simultaneous players, being simple, fast, fun and reliable. The implemented application allows geographically distributed students to concurrently and collaboratively play the same game. Additionally, our work accomplishes many learning preferences of digital / online generation: parallel processing and multitasking, pictures processing, interacting/networking simultaneously with many others, acquiring quickly multimedia information, making the learning process funny [16]. The proposed puzzles are not used alone in teaching and evaluating the mentioned computer science subjects, they are periodically used in a mixture with classical teaching, practising with hand-on exercises and also evaluation.

Our approach mainly insists on how the professor could create his own game related to a computer science topic in the context of our previously introduced e-Learning platform (EDM tool [5]). The jigsaw puzzle is very appropriate for algorithms, computer architecture, microprocessor's organization and design because we use and refer to many pictures with (logical) schemes, instruction / data flow diagrams, memory hierarchy in the teaching process. Whatever is the engineering topic, in learning process the visual memory (visual learning style) is very important because, as an old Chinese proverb said, "a picture is worth a thousand words meaning". Also, we thought that a crossword puzzle is properly used to test student's knowledge in a funnier and more competitive manner. We have chosen examples from algorithms and other computer science fields because we teach these topics but our application allows that the didactic materials to be further extended, so that they cover other scientific fields too.

The organization of the rest of this paper is as follows. In section 2 we shortly review the Related Work in the field of educational software and mainly of game-based learning. Section 3 describes some of the latest developments of our application insisting on the games software design, concept and objectives and the user interface. In section 4 we present the system requirements. Based on a short interactive animated example, we explain the game's functionality. Finally, section 5 suggests directions for future work and concludes the paper.

2   Related work

Transforming by gamification the learning process into "flow" [12] – as enjoyable and valuable for students as possible – can produce high benefits in some educational areas. This approach is successfully used worldwide in teaching music through the Kodály method [13] which reviews and reinforces concepts by games, pictures, movements, songs and exercises. A game that revolutionized the gaming and learning communities is Minecraft. It is a simple game with simple rules, but can be extended so that it gets very complex. It is very similar to a role-playing (RPG) game, where the user can actually change the environment by adding/removing building blocks. Besides the fact that it is fun to use, this game has great educational potential. For example, students can be asked to research a historic event and then recreate it in the game, or build a roller coaster and do experiments on velocity and acceleration [1].

In [2] the authors stated out that in-class games have a positive impact on students, they are more engaged in the learning process and thus they also perform better in exams. In [3] the authors present their experience on how can puzzles and games used to teach and reinforce Computer Science concepts, since many topics of this field are well suited for coverage in such games. They show that instructional crossword puzzles and the Jeopardy!-style games are good methods to create in-class experiences that support learning terminology and basic concepts. In [4] the authors examined the use of puzzles and puzzle-like problems in teaching design and analysis of algorithms like brute-force search, divide-and-conquer, greedy, dynamic programming, backtracking, etc. As a difference, our puzzles are integrated into our own e-learning system, the EDM tool [5]. New puzzles can be easily generated through EDM.

In our previous paper [5], we presented an interactive flash application of the well-known Towers of Hanoi puzzle (with three rods), proposed by Edouard Lucas in 1883, which can be solved through the Divide and Conquer programming technique. We also implemented the generalized version of the Hanoi problem, also called Reve's puzzle, with four or more rods [6]. We used the Frame-Stewart algorithm to solve the generalized Hanoi problem, with four rods. Our proposed interactive Towers of Hanoi puzzle can be run in user mode or in simulation mode. In [7] we integrated into EDM Tool some interactive third party lessons focused on graph algorithms (Breadth-First Search, Depth-First Search, Dijkstra, etc.), on binary tree operations (insertion, deletion, traversal, search, etc.) and also on digital logic circuits.

Using simulators for teaching Computer Science is another widely used method. Besides their importance proved in the computer architecture research field, simulators have lately been extensively employed as a valuable pedagogical tool as they enable students to understand the theoretical concepts better and to visualize how microarchitectural components work and interact [8]. In [9] and [10] we proposed some interactive graphical simulators to teach Branch Prediction and Simultaneous Multithreading Architectures. Such simulators play a key role in translating all complex processing mechanisms in relevant and easy to understand information.

International e-Learning platforms such as Moodle are already very popular, available free of charge as open-source projects. Moodle stands for Modular Object-Oriented Dynamic Learning Environment and is currently the most popular of the free

e-Learning systems. It is provided without fee under the General Public License (GNU) available for any university to implement. It can be implemented on any broadly available operating system (Linux, Windows, Mac OS, etc.) and runs mainly under MySQL, though it can use any ODBC connection to other providers such as Oracle, IBM DB2, Sybase, Access, etc. [11]. It is a very complex system, managing anything from courses and lessons to exams and grades. It is a highly flexible system, but sometimes it may be hard to understand and/or use. In contrast, Education Management Tool is a targeted package that is easy to use and to understand (there is no training needed for understanding any concept in EDM), but still is powerful enough to suit the basic needs of the learner and teacher.

Other learning systems are Blackboard and WebCT, but both of them are expensive, sometimes too expensive for some universities to implement. Education Management Tool has the clear advantage of being free of charge.

3   Games software design: concept and objectives

3.1   Education Management Tool

Education Management Tool (EDM) is an e-Learning platform designed to help students in universities to access learning materials posted by the teachers, to complete online tests and obtaining immediate feedback, as well as retrieve information about results for their exams. It has the advantage of being free, accessible from anywhere and from any type of computing system device, not being restricted to the laboratory. It can manage an entire university: specializations, subjects, administrators, professors and students (see Figure 1). It is a complex tool which is easy to use, with three possible user types: Administrator, Professor and Student.
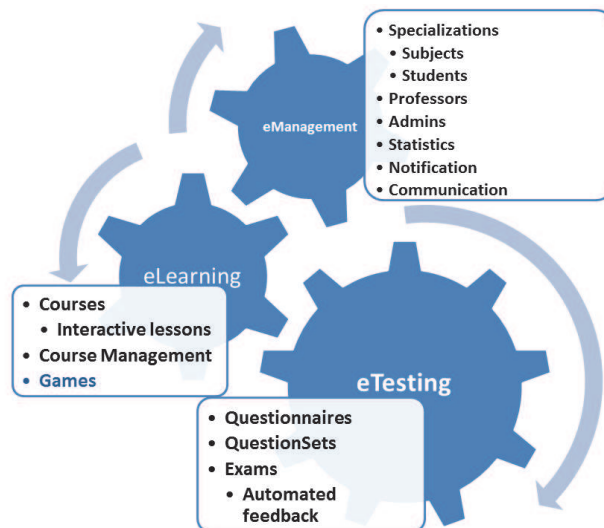

Fig. 1. EDM Tool Structure

Consisting of three modules, EDM is designed to meet the basic requirements of an e-Learning environment: learning, testing and management. The learning component consists of courses, developed by teachers in a manner similar to Google Docs, which are easily accessible by students. The courses may contain images and flash movies. The testing module consists of exams, customized and posted by the teacher. EDM provides a simple and intuitive exam management system that fits any workflow, supporting exam schedule, timed exams and highly editable quizzes. The management part offers the teacher the ability to manage grades and presence at courses together with a communication system for delivering messages to persons / groups.

Both of the games presented in this paper are implemented in Silverlight (a product from Microsoft similar to Adobe Flash) and are tightly integrated into the main application. From the Professor account, they are created and set up, and accessed afterwards from the Student account. A demo account has been set up within the system for anyone who is willing to try the new features of the system. It can be accessed at the web page http://edmtool.gotdns.com/, using the log in credentials (username/password) ism/ism. The games can be accessed by clicking on the left menu on the item "Games", then choosing from one of the two games available and finally joining a room. Because our application is in continuous development, we have not uploaded it yet on a professional server and keep it on the author's laptop, which may cause to not be accessible at any time.

3.2   Crossword Puzzle

During the lessons, the learners are taught to be able to understand and explain certain terms, but in real-life situations, they will have to recognize certain patterns and apply the required methods to solve them. Nowadays, in the age of information, when we don't have to carry around enormous amounts of knowledge, but to know how to find it using search engines (Google, Bing, Yahoo, etc.) or computational engines (Wolfram Alpha), our mind starts to behave differently. As a result, the learner can benefit more if faced with a game that allows him to guess / anticipate a certain term when he is presented with a definition or a situation where it is being used.

The rules for this game are simple: guess all words on the horizontal axis and you will get a combined word on the vertical one. When you correctly guess all terms, the game ends and your time is registered. The total time you have spent solving the crossword puzzle will be shown and compared to what other users have achieved, showing you a score leaderboard with the best times.

On the technical part, this game was easy to implement. After the game has been loaded and the terms and definitions are retrieved from the remote server, everything works on the client (the browser). When the game ends, another communication to the server is initiated, for storing the time achievement and showing the leaderboard.

Each professor is allowed to create a virtually unlimited number of crossword puzzles for the students. The creation of one is simple and intuitive, consisting of two basic steps: main term and associated terms. The main term must be a word which will be formed automatically upon completion of the puzzle. Then, the professor must enter for each character another term, along with its description, which will be

guessed by the student. When entering this information, the professor must respect some limitations. Each term (main or associated) must have a character number in the range of 2-13. It must consist only of lowercase characters, from the Latin alphabet (without spaces or other special characters). In the order of the occurrence of characters in the main term, each associated term (in the order of creation) must contain at least one corresponding character.

The user enters the game, performs a request on the server to retrieve the terms, along with their descriptions, and then displays them to the user. Next, the user repeatedly guesses the term for the provided description; if the term is incorrect, the system responds with an invalid answer message, and the procedure of entering a term is started from the beginning. If the word is true, the answer is marked as correct, and the term input procedure starts from the beginning, unless all terms have been already guessed. If this situation occurs, the game ends, the elapsed time is recorded, sent back to the server and some statistics are shown, including the current player's position, competing with the times of other players.

00:44
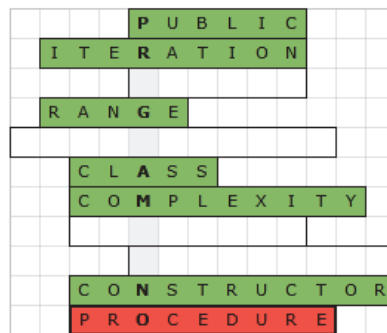
A set of instructions intended to solve a generic problem

Fig. 2. Playing *Crossword Puzzle*

3.3 Picture Puzzle

The inspiration for this game came from the fact that certain subjects must be learned in text mode, while others must be mapped mentally into an image, requiring proper usage of the visual memory. The application trains the student to reconstruct an image or a schema requiring his knowledge, similarly to a jigsaw puzzle. Unlike a jigsaw puzzle, this application does not give any feedback if two tiles have been joined correctly, until the whole image has been reconstructed, making him think twice when trying to put together the pieces of the puzzle.

Picture Puzzle can be played in multiplayer mode; up to 16 players can join the game simultaneously and collaborate to form the completed image. A remote server takes care of the synchronization and the communication between the simultaneously

connected clients. Each action that a player performs on the puzzle is reflected almost instantaneously on the other clients that are connected to the same room. This is done via http by using a long polling method technique of initiating callbacks to the other clients. In addition, recovery methods are implemented to be able to reconnect if the internet connection fails or the server crashes.

The implementation was a bit more complicated, but provided a better usage of the communication skills from the learner side. The paradigm of this game is the fact that the server must notify all connected clients that the game's state has changed (one of the users has moved a tile from one position to another). The server, as its name says, serves data to the client, and in the normal case, the client performs a request, and the server responds to that request. Therefore, the communication must be done via two sockets, one from the client to the server (as in the normal case) and one from the server to the client. Desktop applications are allowed to do that, but browsers, because of security reasons, allow only client-server connections. For callbacks to be still possible, a long-polling method is implemented that simulates server-client communication. The client initiates a request and listens for the response from the server, but the server does not respond until it decides it should do that. Although it is not one of the best methods, Facebook and Google Docs have demonstrated that it can be achieved, so that we started doing it as well. This approach has led to a completely different style of server-side and client-side programming, which was in itself a challenge. For example, two different clients should not use the same tile at the same time, so that we were forced to implement an exclusive check-out system for tile selection. This has been implemented on the client side as well as the server side in the following manner: when another client selected a tile, the server was notified, and checked if no other user has already selected the tile. Afterwards, all other clients were notified of that particular tile selection in order to keep a list of selected tiles from themselves, therefore disallowing selection of tiles that have been selected by other users. This approach also covers the case when two users select the same tile at the same time. Both of them are allowed from the client application to select the tile, but the server makes sure these types of requests are handled sequentially, and only one is allowed to select, while the other client action is dismissed.

From the technology point of view, we used WCF (Windows Communication Foundation – http://msdn.microsoft.com/en-us/netframework/aa663324) at the beginning, but it proved to be unstable, crashed a lot, was hard to implement and to debug. It carried around too much overhead; sometimes it was sending twice the actual data for no obvious reason. SignalR, an open-source library (http://signalr.net/), proved to be much more stable, easier to configure, and did not carry around too much overhead by using lightweight JSON (http://www.json.org/) encoded entities.

In EDM Tool, the games are created from the Professor account, and can be accessed by any student using the system. Upon creation, the following parameters must be set: a title, a description, one of the three levels of difficulty (Easy, Medium, Hard) and the image that should be completed by the students. Once this information arrives on the server, any client can connect and play the game collaboratively. When the game is over, a leaderboard is presented to each user, and the game is reinitialized, so that other users can join a new game.

Unlike Crossword Puzzle, where the user does not communicate with the server during game time, Picture Puzzle exchanges information with the server, sometimes even several times per second. Each action from the current user must be transported to the other connected players in a fast and reliable manner. For example, when the user joins the game, the server notifies everyone and sends the player information to all clients. The same procedure is applied when the user selects a tile (it must be blocked for all other players), when the tile is dropped or when a player leaves the game. An interesting challenge was having a common, synchronized timer for all players. In Crossword Puzzle, that was simple, because only one user had to see the timer. In this case, the timer must be initialized when the first user enters the game room and, for each client that connects afterwards, the elapsed time is sent to him. From the moment the timer info is received, the client increases the time counter on its own, because no further synchronization is needed. Another alternative was possible: instead of sending the elapsed time, a more accurate measure could have been to send a timestamp containing the date and time when the game started. But because people can possibly access the game from different time zones or have their clock set up incorrectly, this approach could have led to incorrect display of the elapsed time. Although much better, the current approach is still not the best option, because delays of a few seconds can also lead to inaccurate elapsed time measure, but these differences are acceptable. The best option would have been to measure the average time it takes for the server to respond, and then add that value to the received elapsed time.

Fig. 3. Playing Picture Puzzle

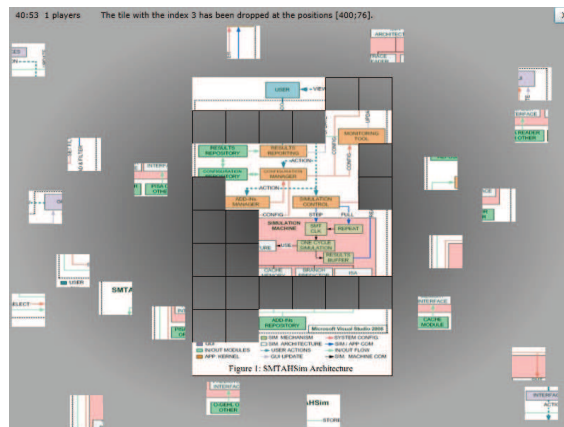## 4 The System Requirements

Because it is a client-server application, the system requirements have to be split up in two parts: server and client.

The server requires the operating system Windows Server 2008 to be installed, together with Microsoft Sql Server 2008 R2 as the database provider. Internet Information Services 7.5 at least has to be installed for the server-side assemblies to be

deployed and for the proper execution of the code in those assemblies, .NET Framework 4.0 or a later version is necessary. From the hardware point of view, we recommend a Dual Core processor with a frequency of at least 2 GHz, 4 GB of RAM and an Internet connection of at least 10 Mbps, for a fluid game experience for up to a few tens of players. Hardware requirements are pretty high, because on each client request, the server has to send data to all the connected players, unlike normal servers, which respond only to the connected client.

The client requirements are not as high as the server ones. From the software point of view, any browser that supports Silverlight and has version 5.0 installed will work (Internet Explorer, Google Chrome, Mozilla Firefox). The hardware requirements are also low, although there is some graphical processing done during the game, it will not require too much processing power from the machine.

5   Conclusions and Further Work

Nevertheless, the game ideas could be implemented in other basic tools and topics. From a pedagogical point of view, the proposed tool benefits the learning process, since it permits students to observe rather than learn through classical methods, discovering the fact that studying can be fun as well. Sometimes the information within lectures is not transferred completely to the students also because of the age and mentality differences. EDM Tool provides the much-needed bridge, to alleviate this misunderstanding and to encourage students to work with the concepts and ideas presented during courses. Students will use the experience with these games in later courses when more notions (about advanced processors, operating systems, etc.) are introduced. As far as concern the educational added value, in the crossword puzzle game we followed a common educational methodology in which students first tackle easy and small chapters based on few definitions, and then progressively continue to more complex ones with many and more difficult definitions and tasks.

No success story comes without problems. The system is far complete, as well as the games. Although the games run smooth and without major problems, we have experienced some issues connecting new players to the Picture Puzzle application. If the connect succeeds, the communication works without any problems, but in some cases, the messages sent from the client never reach the server for some reason.

As future work, we intend to determine the success of this game-based learning approach, through performing assessments using questionnaires (from EDM Tool). Thus, we want to quantify the learning degree of each student and the satisfaction degree if they are using games in the learning process. For now, we observed that the games have good acceptance among the students. In fact, they are using these proposed games to do their homework, obtaining encouraging results in class. In this sense, the instructors could see how student grades will improve with respect to the previous academic years. Besides the tests, we will consider a different approach inspired from machine learning. Thus, we think to improve our games using reward learning algorithms such as reinforcement learning (Q-learning or SARSA) [15]. Also, we want to add new features such as the option to save some "good but not fin-

ished configurations", resuming games from that point, etc. Another issue is security. Although our system does not manage bank accounts or sensitive information, it can be the target of malicious attacks. In the future, we will increase the security of our application by implementing data encryption and hashing where needed. For example, the students could easily figure out the answers for the Crossword Puzzle if they are curious enough to seek through the page source code.

References

1. Minecraft and Crafting To Learn
   `http://shoyulearning.wordpress.com/2012/01/11/minecraft-and-crafting-to-learn/`
2. Kirkland D., O'Riordan F., Games as an Engaging Teaching and Learning Technique: Learning or Playing?, 1st Irish Conference on Engaging Pedagogies, Dublin, Ireland, 2008.
3. John M. D. Hill, Clark K. Ray, Jean R. S. Blair, Curtis A. Carver, Puzzles and Games: Addressing Different Learning Styles in Teaching Operating Systems Concepts, 34th SIGCSE technical symposium on Computer science education, pp. 182-186, 2003.
4. Levitin A., Papalaskari M.-A., Using Puzzles in Teaching Algorithms, Technical Symposium on Computer Science Education, pp. 292-296, Covington, Kentucky, February 2002.
5. Florea A., Gellert A., Anghel T., Florea D., Enhanced Learning and Educational Management through Online Collaborative Technologies, Journal of Digital Information Management, Vol. 9, No. 1, pp. 33-42, February 2011.
6. Stockmeyer P. K.. Variations on the Four-Post Tower of Hanoi Puzzle, Congressus Numerantium 102, pp. 3-12., 1994.
7. Anghel T., Florea A., Gellert A., Florea D., Web-Based Technologies for Online e-Learning Environments, Seventh International Scientific Conference "eLearning and Software for Education", Bucharest, April 2011.
8. Yi J.J., Lilja D.J., Simulation of Computer Architectures: Simulators, Benchmarks, Methodologies, and Recommendations, IEEE Transactions on Computers, vol. 55, No. 3., 2006.
9. Radu C., Calborean H., Crapciu A., Gellert A., Florea A., An Interactive Graphical Trace-Driven Simulator for Teaching Branch Prediction in Computer Architecture, The 6th EUROSIM Congress on Modelling and Simulation, Ljubljana, Slovenia, September 2007.
10. Florea A., Ratiu A., Gellert A., Vintan L., A Visual Simulation Framework for Simultaneous Multithreading Architectures, The 25th European Conference on Modelling and Simulation (ECMS 2011), Krakow, Poland, June 2011.
11. Klaus Brandl, ARE YOU READY TO "MOODLE"?, Language Learning & Technology, Vol. 9, No. 2, May 2005, pp. 16-23
12. Csíkszentmihályi M., Flow: The Psychology of Optimal Experience, Harper and Row, New York, 1990.
13. Kodály Z., Selected Writings of Zoltán Kodály, Boosey & Hawkes, London, 1974.
14. Prensky M., Digital Game-Based Learning, McGraw-Hill, 2001.
15. Sutton R., Barto A.S., Reinforcement Learning: An Introduction, The MIT Press, Cambridge, MA, 1998.
16. Derryberry A., Serious games: online games for learning, 2010, `www.adobe.com/products/director/pdfs/serious_games_wp_1107.pdf`.