Model Driven Development of User Interfaces for Educational Games

Miroslav Minović†, Miloš Milovanović†, Mladjan Jovanović‡ and Dušan Starčević†
†Faculty of Organizational Sciences, ‡ Faculty of Electrical Engineering, Belgrade University

***Abstract* — The main topic of this paper is the problem of developing user interfaces for educational games. Focus of educational games is usually on the knowledge while it should be evenly distributed to the user interface as well. Our proposed solution is based on the model-driven approach, thus we created a framework that incorporates meta-models, models, transformations and software tools. We demonstrated practical application of the mentioned framework by developing user interface for educational adventure game.**

***Keywords* — Learning and adaptive systems, Human-centered design, Entertainment and Gaming.**

I. Introduction

With the rapid development of computers, it became clear that they will play an important role in business, science, but also as a gaming platform. The first computer games appeared between 1950. and 1960. and from then on, their development gained more and more speed. It was almost impossible to assume that they would become one of the most dominant social phenomena and that they would generate more revenue than movie industry during the last decade of the 20th century.

Such fast development allowed computer games to become more complex, more attractive, to have rich content and, at the same time, to attract more players. The popularity of computer games led to them being the important part of modern society. Because of this, during the last few years, the idea of using games in educational purposes became more and more popular.

Educational games have gone through the evolution process, from simple 2D games with low knowledge integration all the way to the complex 3D game whose purpose is to pass the knowledge to the player without breaking the game flow.

It is reasonable to assume that making the educational game as interesting as possible will improve learning. Also, achieving increased player engagement involves adding deep emotional experiences to video games [1].

There are several approaches to the development of educational games. Some games are using well-known, popular environment and set of rules, adapted for the purposes of education. On the other hand, some games are developed with a certain subject matter in mind. Games can also be used for teaching certain skills, or for simulations of real – life events. In some cases, the modification of popular games (game modding) was used for teaching.

This paper tries to honour the needs of a certain educational matter as well as the needs of a game player. User interface and interaction with the user is the area of ever growing research. On the other side although the user interface is important factor in video games, and is a subject of intensive designing, in educational games that matter is underappreciated. In educational games the focus is usually on the knowledge and interaction with player is usually in the service of that knowledge.

In our work we focused on presenting a new model that will allow educational games to better integrate with player's needs. Taking players profile in to account is equally important as the gameplay itself.

Second part of this paper addresses problems with traditional approach of educational game development. Part three gives a short review on bibliography regarding this matter. Part four presents our solution to the problems identified in the second part. In fifth part we give a case study that shows practical use of the proposed solution. Finally, we conclude the paper with our findings and the further research directions.

II. Problem definition

Process of making a good educational game is very complex. In order to fully satisfy the needs of game player, it requires considering many factors.

Creating a fun game mainly depends on game designer. The problem is how to have fun playing, and learning as well. Learning process can be rather uninteresting, especially if the learning content is unable to hold our attention by its own. That is why capturing students' attention and engaging them in the learning experience in a didactically correct way must be identified. Gameplay is a promising solution to achieve this goal, as it provides a structure and well defined rules to drive the user behaviour [2].

In order for a gameplay to be successful in holding players attention, proper knowledge integration is essential. If a game designer is not an expert in the area

This work is part of a project "Managing intellectual capital", funded by the Ministry of Science and Technology of the Republic of Serbia, contract no 13028-TP.

M. Minovic is with the Faculty of Organizational Sciences, University of Belgrade (e-mail: mminovic@fon.rs).
M. Milovanovic is with the Faculty of Organizational Sciences, University of Belgrade (e-mail: milovanovicm@fon.rs).
M. Jovanovic is with the Faculty of Electrical Engineering, University of Belgrade (e-mail: mladjan@rcub.bg.ac.yu).
D. Starcevic is with the Faculty of Organizational Sciences, University of Belgrade (e-mail: starcev@fon.rs).

978-1-4244-3960-7/09/ $25.00 2009 IEEE

which educational game covers, he will face the problem of making the knowledge as fun as the game itself. Traditional educational games that addressed only one matter are losing its impact on learners. The process of developing educational games is expensive and time consuming and the final results represent only one subject matter and it is not reusable. That is why there is a current trend of using existing game editors in process of educational game development. Downside of this approach is poor knowledge integration. For that reason we proposed and developed a software system that provided the ability of developing educational games with no programming skills and that allowed educators a good way to integrate knowledge [3]. Its upside is reusability of knowledge as well as multimedia game content (graphic, music…).

One other factor that influences capturing player's undivided attention is usability of educational game. Game interface is essential to player's experience. Educational games cannot be focused mainly on the knowledge. User, or in this case player (student) must be in the center of the design process. Interaction between the player and the game is highly important. If the main purpose of educational games is to teach than the learner must be taught the way he feels most comfortable. Every person has a specific psychological profile. That profile determines preferences for a person and plays an important role for the game design process. Disregarding the players profile while designing an educational game user interface, can prove to be the biggest reason for an educational game failure. Also disregarding it during the game play itself can be a problem.

Most of the educational games on the market fail to properly attend this issue. Our research focus lays on the problem of educational games. We are working on a framework for educational game development that includes meta-models, models, transformations and software tools. In this paper we will attempt to address the issue of designing user interface in educational games. Our focus will be on the specific part of the framework that considers the issue. The purpose is to enhance the learning experience through properly designed and adapted user interface.

III. Bibliography review

Educational games present a relatively young but fast growing research field. There are already many interesting and effective solutions. Some of the solutions are games focused on solving a particular problem, while other solutions are engines for creating educational games. Some types of games are more suitable for representing certain types of knowledge than the other. Choosing the right type of game is an important part of the work.

As mentioned earlier, main purpose for the educational games is to teach and pass knowledge. That is why a majority of games is focused mainly on the knowledge. Different skills and knowledge can be taught differently. Some games are using well-known, popular environment and set of rules, adapted for purposes of education - for example, the educational game based on "Who wants to be a millionaire?" quiz [4]. It uses all elements of the TV show, but questions are chosen by the teacher.

On the other hand, some games are developed with certain subject matter in mind, like games for teaching electromagnetism called Supercharged! [5] or a fantasy adventure game for teaching the basic concepts of programming [6]. In some cases, the modification of popular games (game modding) was used for teaching computer science, mathematics, physics and aesthetics [7]. Game design can be used to achieve similar goals - developing problem solving skills and teamwork [8].

Regardless of the rapid growth of this research field, user modeling for the purpose of educational games is still in its initial phases. User modeling can make games a more individualized experience [9]. While there are many examples of practical work in adaptive interfaces [10], there is very little practical work done on the field of user modeling and user interface adaptation in educational games. A good example is a game S.C.R.U.B [11], the arcade-style game for learning microbiology. It considers the connection between what is presented to the player and the perceived motivations and learning styles of the player. While there are numerous efforts that games can be applied to learning, relatively few attempts can be found where principles of learning and motivation theories were explicitly followed a priori in design [12].

IV. Proposed solution

Our approach is inspired by the model-driven development, where software development's primary focus and products are models rather than computer programs. In this way, it is possible to use concepts that are much less bound to underlying technology and are much closer to the problem domain [13].

TABLE 1: Mapping Educational Games Concepts to the OMG's MDA Levels.

OMG MDA Level	***Educational Game Metamodeling Architecture***	***Description***
M3 – Meta-metamodel	The Meta Object Facilities (MOF)	The MOF is an OMG standard that defines a common, abstract language for the specification of metamodels. MOF is a meta-metamodel – the model of the metamodel, sometimes also called an ontology
M2 – Meta models	The Educational Game Metamodel (EGM)	The Educational Game Metamodel provides a common and standardized language about phenomena from various domains relevant to the design of educational games. It is called a

		metamodel as it is the abstraction of platform specific models.
M1 – Models	Platform-specific Shemas (XHTML, SAPI, SWIXml Schemas...)	Platform specific models of educational game content.
M0 – Objects, data	Content data (XHTML, SAPI, SWIXml files...)	Instances of platform specific models.

Table 1 gives an overview of educational game development through MDA levels. It uses a platform-independent base model (PIM), and one or more platform-specific models (PSM), each describing how the base model is implemented on a different platform [14]. In this way, the PIM is unaffected by the specifics of different implementation technologies, and it is not necessary to repeat the process of modeling an application or content each time a new technology or presentation format comes along. The views on multimedia content from different levels of abstraction can be derived by *model transformations*. In MDA, platform-independent models are initially expressed in a platform independent modeling language, and are later translated to platform-specific models by mapping the PIMs to some implementation platform using formal rules. The transformation of the content models can be specified by a set of rules defined in terms of the corresponding higher level metamodels. The transformation engine itself may be built on any suitable technology such as XSLT tools. Our approach is based on standard technologies such as the Unified Modeling Language (UML) and XML, which are familiar to many software practitioners and are well supported by tools. Therefore, it is not necessary to develop complex solutions from scratch, and it is possible to reuse existing model-driven solutions and experiences from other domains. In our work we rely on existing UML modeling tools, XML parsers and software frameworks, developing only code that extends, customizes, and connects those components according to common and standardized language defined in the Educational Game Metamodel.

Interaction between player and game is realized using multiple channels of communication. In order to describe relevant concepts of communication between player and game, we propose metamodel as depicted by Fig 1.

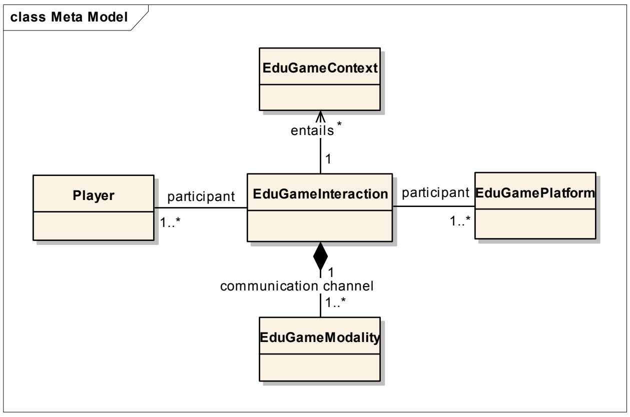

Fig. 1. Basic concepts of communication between player and game.

Main concept in metamodel of communication between player and game is GameModality. It is defined as a form of interaction between player and game, which engages human capabilities to produce some effects on users. Presented concept of GameModality is derived from existing metamodel of multimodal human-computer interaction [15]. Multimodal interaction can be established between multiple players (GamePlayer) and multiple game platforms (GamePlatform). GameContext defines set of conditions and facts that are pertinent to specific situation, and can affect interaction between player and game. GameContext is additionally classified into:

- Physical context – defines environment conditions such as kind of space in which interaction is established, temperature value, luminance level, noise, humidity;
- Situational context – defines current situation in environment from the point of view of the task that user has to accomplish;
- Common knowledge – represents a set of facts that are understood by humans and computer;
- Social context – defines social environment of the interaction.

Each modality engages human capabilities, producing some effect on the user. Effects are classified into four main categories [15]:

- Sensory effects describe the human sensory apparatus's processing of stimuli;
- Perceptual effects result from the human perceptual system's analysis of sensor data;
- Motor effects describe human mechanical actions, such as head movement or pressure;
- Cognitive effects occur at higher levels of human information processing and include memory, attention, and curiosity. In the design of educational games, these are the key asset.

In this way, user interface can be described in terms of messages (effects) that designer sends to the user. Effects are interconnected. For example, all perceptual effects results from sensory effects. These relationships enable designers predicting the result of using some effects.

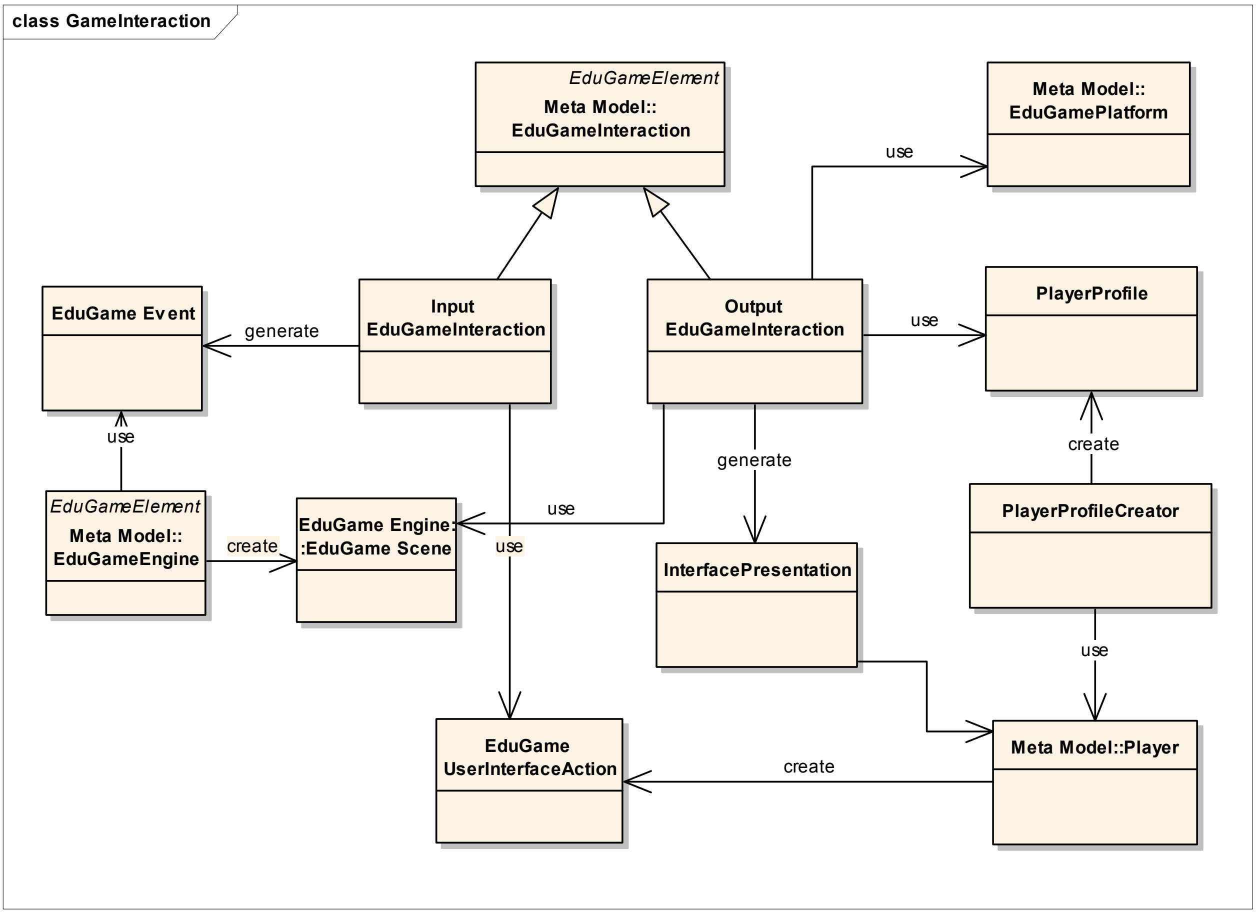

Fig 2. Detailed concepts of communication between player and game.

When we analyzed Game Interaction in more details, we found that EduGameInteraction could be specialized into Input and Output EduGameInteraction, representing user input and game interface output communication. EduGameEngine creates EduGame Scene, abstract concept which defines all elements of one scene in educational game, but without any user interface elements. Then, Output EduGameInteraction produces concrete InterfacePresentation based on given EduGame Scene and concrete EduGamePlatform. InterfacePresentation can be adjusted depending on PlayerProfile, created by PlayerProfileCreator. PlayerProfile is determined at the beginning of the game and can be dynamically adjusted during the game, depending on Player actions. For determining initial profile we can use psychological tests or try to capture it automatically, if possible, by analyzing Players actions produced at training game level. Since we are modeling educational games, we are mostly interested in finding out player's cognitive style and motivational state, in order to adapt educational content and game interface.

Input EduGameInteraction generates EduGame Event based on EduGame UserInterfaceAction created by Player. UserInterfaceAction represents low level actions performed by Player, such as mouse movement, or keyboard key press, if standard keyboard/mouse/screen interface is in use. EduGameInteraction handles such actions, trying to detect changes that significantly influence EduGame. When important action is detected, EduGame Event is raised, such as question answered or puzzle solved, and is being sent to EduGameEngine for further processing. EduGameEngine produces new EduGame Scene which triggers Output EduGameInteraction.

Clear separation between EduGameEngine and EduGameInteraction was our main aim. EduGameEngine represents educational game logic, while EduGameInteraction represents interaction with player. In this way we enabled game designer and educator separation of roles. Another benefit is reusable game interaction. To clarify, by applying this we gain the ability to keep game interaction while modifying just educational content. Such educational game can be applied to many different domains, without additional programming and user interface design. This brings a major contribution to educational game development.

In order to appropriately capture players profile proper classification is in order.

A. Game player profiles metamodel

We have classified players/learners according to multiple intelligence theory [16]. The eight intelligences are as follows [17]:

1. Linguistic: think in words,
2. Logical/Mathematical: think by reasoning,
3. Spatial: think in images and pictures,
4. Bodily/Kinesthetic: think through somatic sensations,
5. Musical: think via rhythms and melodies,
6. Interpersonal: think by bouncing ideas off other people,

7. Intrapersonal: think in relation to their needs, feelings, and goals,
8. Naturalist: think through nature and natural forms.

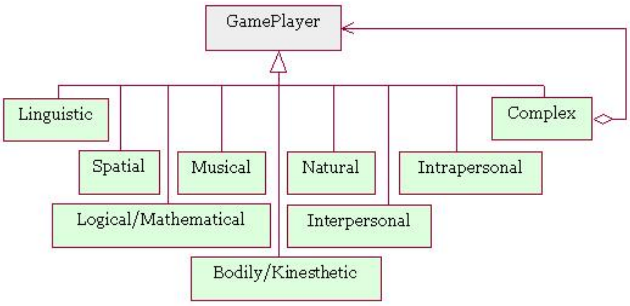

Fig. 3. Proposed player classification.

The fact that some people impose multiple types of intelligences justifies introduction of *Complex* type. Gardner states that most schools focus on the linguistic and logical/mathematical intelligences. In this way individuals who show gifts in the other intelligences: the artists, architects, musicians, naturalists, designers, dancers, therapists, entrepreneurs are learning-confined by traditional educational process. Unfortunately, many children who have these gifts do not receive much reinforcement for them in school. McCue [18] describes how computer games can provide a multiintelligence approach to learning by exemplifying particular intelligences. In our approach we are reusing multiple intelligence concepts in order to (1) comprise users with diverse learning preferences inherently predefined by particular intelligences (2) identify the dominant content of learning interface according to intelligence type as [17]: words (linguistic intelligence), numbers or logic (logical-mathematical intelligence), pictures (spatial intelligence), music (musical intelligence), self-reflection (intrapersonal intelligence), a physical experience (bodily-kinesthetic intelligence), a social experience (interpersonal intelligence), an experience in the natural world (naturalist experience).

V. Design Case Study

In this section we will apply our model-driven approach and present educational game user interface design example using high-level EGM terms. In real-life situation, every game use many different communication modes (GameModalities), and each of them is engaged to some extent, since typical game player profile is complex type. Thus, every learner has its own proportion of intelligences, which imply unique educational game interface. Relying on high-level EGM concepts and defining model transformations, gives us an opportunity to specify very specific game interfaces which will be well suited to concrete learner/user, supporting his own intelligence type.

Based on this mapping, we have been developing software tools that create game user interface description with higher-level markups, and which than transform this high-level markup into visual or audio representation.

Table 2: Illustrates model transformations on the example of Java SWIXml.

Model	*XML code*
PIM XML game definition fragment	<ACTION NAME="ATTACK"> <DESC>**Your army is under attack!**</DESC> <EFFECTS> <EFFECT NAME="Direct system response" CODE="DSR"/> </EFFECTS> </ACTION>
XSL Transformations	<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"> <xsl:output method="html" encoding="utf-8"/> <xsl:template match="ACTION [@name]"> <panel constraints="BorderLayout.CENTER"> <xsl:apply-templates/> </panel> </xsl:template> <xsl:template match="EFFECTS/EFFECT [@code = string(DSR)]"> <label font="Georgia-BOLD-12" foreground="blue"> <xsl:attribute name="text"> <xsl:value-of select="desc"/> </xsl:attribute> </label> <Graphics2D> <xsl:attribute name="image"> <xsl:value-of select="name"/>.gif </xsl:attribute> </Graphics2D> </xsl:template> </xsl:stylesheet>
PSM XML fragment	<panel constraints="BorderLayout.CENTER"> <label font="Georgia-BOLD-12" foreground="blue" text="Your army is under attack!"/> <Graphics2D image="ATTACK.gif"/> </panel>
Platform	SWIXml

We present transformation of high-level game definition into platform-specific game implementation (Table 2), based on our MDA approach. Presented example transforms an object of strategic game, action named ATTACK, which reflects the fact that player's troups are currently under attack. Platform-specific language is Java SWIXml which enable us to define whole user interface as XML document, which will produce concrete Java user interface, based on SWING user controls [19]. Transformations are implemented using

XSL transformations. ATTACK action is transformed as Java SWIXml panel and shown as animated gif with flashing army figure. In real game, ATTACK action is part of more complex concept SCENE, which defines complete strategic game scenario, corresponding to EduGameScene concept from metamodel given in previous section.

Given mappings with supporting transformations allows us to personalize the user interface to specific learner profile. Users can choose the presentation form which they prefer, or it could be done automatically since the game engine contains built-in psychology mechanisms for determining particular learning profiles according to intelligence type. In addition, it is possible to adopt content for different platforms, depending on device capability (eg. cell phone, PC, PDA).

Next, we give sample screenshots with generated user interfaces, for educational adventure game. First one is implemented as Java applet (Fig 4).

Fig 4. Web client game interface

In order to get the full impact on a player, appealing interface for our game is required. The game platform we developed has skinable interface. By making the required images you can change the way the game looks (Fig 4).

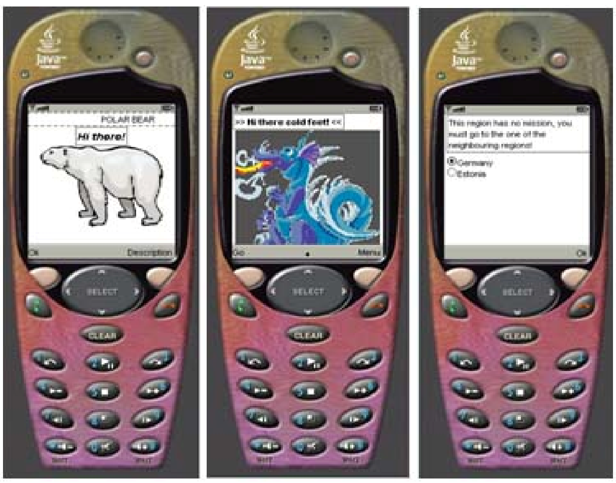

Fig 5. Mobile client game interface

Also different platforms could be supported, for now we developed client for J2ME platform for mobile phones, with required XSLT transformations.

VI. Conclusion

This paper presents an approach to modeling educational games user interface that incorporates principles of motivation theory as well as multimodal human-computer interaction. The overall challenge is to increase player's motivation for studying. Proposed modeling framework is aimed to design user interfaces that motivate users with diverse learning abilities and preferences to get actually involved in learning process. This is achieved by identifying game player profiles according to player learning abilities and preferences and adopting game interaction to player profile.

The efficiency of usage of our method depends very much on the efficiency of supporting tools. In our current approach we are relying on the existing UML modeling tools, and their integration mechanisms. The advantage is that the designers who are familiar with UML can do the design in their natural environments.

Applicability of our approach is demonstrated by toy example of MDA user interface development process and implementing educational adventure game user interface, for web and mobile platform. In our future work we plan to further improve our framework, as well as to extend our system to other available platforms, such as JavaFX, Adobe Flash, Symbian, Windows Mobile and other.

References

[1] M. Van Lent, W. Swartout: "Games: Once More, with Feeling", *Computer*, vol. 40, Issue 8, pp. 98 – 100, Aug. 2007.

[2] M. F. Costabile, A. De Angeli, R. Lanzilotti, C. Ardito, P. Buono, T. Pederson, "Explore! Possibilities and Challenges of Mobile Learning", in *CHI 2008 Proceedings,* Florence, Italy, 2008, pp. 145-154

[3] M. Minović, M. Milovanović, M. Lazović, D. Starčević, "XML Application For Educative Games", *Proceedings of European Conference on Games Based Learning ECGBL 08*, Barcelona, Spain, 2008

[4] G. Reinhardt, L. Cook: "Is This a Game or a Learning Moment?", *Decision Sciences Journal of Innovative Education*, vol. 4, no. 2, pp 301-304, 2006.

[5] K. B. Squire, "Electromagnetism supercharged! Learning physics with digital simulation games", *Proceedings of the 2004 International Conference of the Learning Sciences*, Santa Monica, 2004.

[6] R. Moser, "A fantasy adventure game as a learning environment: Why learning to program is so difficult and what can be done about", *Proceedings of the ITiCSE '97*, Uppsala, Sweden, 1997.

[7] M. El-Nasr, B. Smith: "Learning through game modding", *ACM Computers in entertainment* , vol. 4, no. 1, article 3B, 2006.

[8] B. Steiner, N. Kaplan, S. Moulthrop, "When play works: Turning game-playing into learning", *Proceedings of IDC '06*. Tampere, Finland, 2006.

[9] B. Magerko: "Adaptation in Digital Games", *Computer*, vol. 41, Issue 6, pp. 87-89, June 2008.

[10] L. Jiming, W. Chi Kuen, H. Ka Keung: "An Adaptive User Interface Based On Personalized Learning", *IEEE Intelligent Systems*, vol. 18, Issue 2, pp. 52-57, March 2003.

[11] (Online Sources) B. Medler, B. Magerko, K. Farley, M. Riedl (2008, 04, 27). *S.C.R.U.B.* Available: http://adam.lcc.gatech.edu/?page_id=11

[12] M. Jovanovic, D. Starcevic, V. Stavljanin, M. Minovic, "Surviving the Design of Educational Games: Borrowing from Motivation and

Multimodal Interaction", *Proceedings of Conference on Human System Interactions*, Crakow, Poland, May 2008, pp. 188-192.

[13] B. Selic: "The pragmatics of model-driven development", *IEEE Software*, vol. 20, no 5, pp. 19-25, September/October 2003.

[14] J. Bezivin: "From Object Composition to Model Transformation with the MDA", *Proceedings of TOOLS,* IEEE CS Press, Santa Barbara, USA, August 2001.

[15] Z. Obrenovic, D. Starcevic: "Modeling multimodal Human-Computer interaction.", *IEEE Computer*, vol. 37, no. 9, pp 62-69, 2004.

[16] G. Howard, "Intelligence reframed: Multiple Intelligences for the 21st century.", 2000

[17] (Online Sources) T. Armstrong, (2000) "Multiple Intelligences", Available: www.thomasarmstrong.com/multiple_intelligences.htm

[18] P. McCue, "The crucial role of animated children's educational games", *Proceedings of International Conference on Computer Graphics and Interactive Techniques*, Los Angeles, California, 2005

[19] (Online Sources) SWIXml project, Available: http://www.swixml.org/