

Please quote as: Thillainathan, N. (2013): A Model Driven Development Framework for Serious Games. In: Informatik 2013 Doctoral Consortium, Koblenz, Germany.

# A Model Driven Development Framework for Serious Games

Niroshan Thillainathan

Information Systems  
Kassel University  
Pfannkuchstr. 1  
34121 Kassel  
thillainathan@uni-kassel.de

**Abstract:** Serious games have become more important in the field of education. The tight connection of education to computer game playing by presenting learning content in a game-like environment creates a better usage experience thus leading to higher usage and potentially better learning outcomes. To create successful serious games, it is necessary to combine programming knowledge and didactical knowledge to properly include learning objectives. Most educators are not able to develop such games on their own due to lack of programming and game design skills. In my dissertation, I am focusing on a solution based on model driven development techniques that shall allow educators to create didactically sound serious games. My idea is it to provide a domain specific modeling language for serious games and a visual modeling tool, which allows the generation of games from visual models.

## Introduction

Serious games are applications, which focus on the tight connection of education to computer game playing by presenting learning objectives in a game-like environment. A positive effect of playing computer games is that the player's motivation to continue playing is at a high level throughout the game. This is highly preferable for education and increases the attractiveness of serious games for learning [BS09]. In class, serious games are used by educators to reach the young generation of learners, who frequently get in touch with new technologies. Modern technologies enable not only to address the user on an intellectual level, but also to stimulate all of his senses [DBB12]. Offering new learning contents through a game induces higher motivation to play/learn and thus has a higher learning success than presenting it in the classical way [Pi07]. Mouaheb et al. (2012) describe six different characteristics for serious games. A serious game (i) is a learning process, (ii) is a game, (iii) is an application of video game technologies, (iv) targets multiple learning objectives (to teach, train, educate, heal), (v) applies in almost every field (education, health, advertising etc.) and (vi) is intended for all age groups (children, adolescents and adults) [Mou12]. So, in conclusion serious games can be an effective tool to transmit educational content to learners.

However educators face the problem, that existing serious games are either too specific on a topic or too generic, which does not allow adapting the game for their own purposes. Therefore, either the development process must be handed over to a professional game developer, which is associated with high cost or the serious game must be planned and developed by the educators themselves. Serious game development consists of three major development parts: (i) game design, (ii) software development and (iii) development and integration of pedagogical elements. Although educators are experts in the third part, unfortunately the lack of programming knowledge and game design skills lead to the main problem that they are unable to develop such games ([Ch09], [YL10]). Sparse research has been conducted to address this problem. So, the challenge here is to enable these non-technical domain experts (educators) to develop serious games adapted to their own learning content, without having knowledge about software development or game design.

To address this problem, the aim of my dissertation is to develop a framework based on model driven development techniques which allow the generation of serious games from models. Model Driven Development (MDD) is a methodology in software engineering, which consists of techniques for automated generation of software code from formal models. Domain-Specific Modeling Languages (DSMLs) are formal languages, which are explicitly designed for a specific domain to formalize the application structure, behavior and its requirements with models. In the case of my research the domain is serious gaming. The elements of the language will represent game-related functionality as well as didactical concepts. In MDD, applications are modeled at a higher abstraction level instead of writing code in a programming language. The generation of software from models is the main task of the so called transformation engine and generator. Consequently, all changes made to the software will be made by updating the model [Sc06].

Model Driven Development for serious games has several benefits for non-technical educators. DSMLs allow them to focus on creating models representing their didactical goals and in return they free them from the task of game development. This higher level of abstraction allows easier maintenance and documentation of games, which is highly desirable for any educators [KWB03].

A domain specific modeling language (DSML) and an associated visual game editor, with which a serious game can be modeled and developed visually, form together my serious game development framework. In my research the language will need to provide high-level modeling concepts (and associated graphical representations) for expressing games and pedagogical aspects as well as game logic (behavior of the game). Based on this DSML a serious game editor is developed as a plugin for the game engine Unity3d. My developed tool allows modeling the game and its logic in an intuitive fashion. A semantic validator checks if the design conforms to the semantics of the DSML. Furthermore a code generating component is developed, which performs the generation of software code for serious games.

By now my research is limited by supporting only one game genre, in this case point and click graphical adventure games. Especially the slow pace during game play, the

possibility to study the environment and the fact that all tasks are based on problem-solving are relevant from a pedagogical point of view. These characteristics increase the learning success while playing [Va08].

## **Related Work**

Only little research concerning modeling languages for serious games are present in current scientific literature. Tang & Hanneghan (2008) propose a domain specific modeling language for serious game design, which allows two types of modeling: data and visual modeling. Data modeling is described as the definition of objects, flows and processes. Whereas visual modeling is presented as the positioning of in-game components within the game world [TH08]. Assessment and adaptation are not supported by this language, but they are crucial for serious games.

Gomez-Rodriguez et al. (2011) describe a solution named GAM-ING, which is based on INGENIAS, an Agent Oriented Software Engineering methodology. The main focus of their work is to allow the specification, design and development of actors, scenes, context, light, cameras, and environment of a serious game by using different types of meta-models [Go11]. The disadvantage of this language is that it is rather complex for non-technical educators.

Marchiori et al. (2011) have developed the story-flow language, which aims at developing narrative educational games based on game storyboards and are represented by a visual language. This language allows to model potential actions of players and their consequences to give an abstraction of the story-flow, which can be automatically transformed into playable 2D educational video games [Ma11]. The focus of this language is on narrative educational games and the description of the game behavior. Details about the design of game worlds are not given.

The presented modeling languages have several weaknesses and disadvantages. Either the existing approaches are lacking in functionality or are too complex to be used by non-technical domain experts. So, this induces the need for a solution for the development of serious games, which allows educators to develop their own serious games, without technical or game design knowledge.

## **Research Approach and Methodology**

My research follows the Design Science Research Methodology (see Figure 1) of Peffers et al. (2007), which aims to solve a real world problem by constructing and evaluating artifacts. This iterative process includes six steps: problem identification and motivation, definition of the objectives for a solution, design and development, demonstration, evaluation, and communication [Pe07]. Step 1, the problem identification and motivation is described in the introduction part of this paper. The last step of this iteration process will be done by publishing my research results on scientific conferences and journals.

The remaining steps 2-5 will be dealt within the scope of the following research questions of my dissertation:

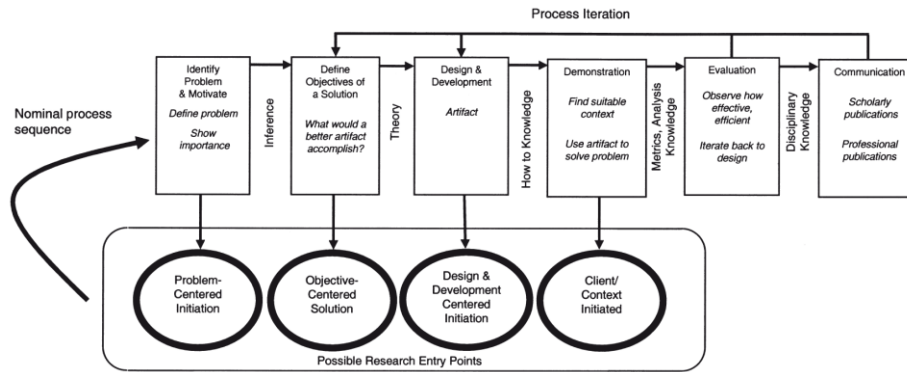


Figure 1: DSRM Process Model; Source: [Pe07]

***RQ1: What are requirements for a Domain Specific Modeling Language for Serious Games?***

The aim of this research question is to get fundamental information about the serious game development process. This step contains both the examination of existing game development models and a requirements analysis for serious game development. Furthermore an in-depth literature review has to be conducted to identify the specific requirements for a DSML for serious games. This research question addresses the “Define Objectives of a Solution” step in the DSRM Process Model. The output of this research question will be a set of requirements for the development of a DSML for serious games.

***RQ2: Which elements can be derived for a Domain Specific Modeling Language for Serious Games?***

The aim of this research question is to design a DSML for serious games. Therefore the requirements list from RQ1 will be used as a basis. Each requirement will be analyzed and elements for the language will be derived, until the requirements are met. During this syntax definition process further DSML related elements, e.g. connectors, will be determined. This is followed by the specification of the semantics of this DSML. This research question addresses the “Design & Development” step in the DSRM Process Model. The output of this research question will be the syntax and semantics of a DSML for serious games.

***RQ3: What are software architectural features of a game editor which allows the development of Serious Games based on the Domain Specific Modeling Language?***

The aim of this research question is to construct a serious game editor, which allows the development of a serious game and its logic in an intuitive fashion. To achieve this, the aforementioned DSML and its elements will be transferred into the game editor, so that both the structure and the logic of the game can be modeled visually using the language.

At the same time the editor allows to set the appearance and content of the serious game. To check if the design conforms to the semantics of the DSML a semantic validator is built. Finally to get the application code for the serious game a code generating component is developed, which parses the designed model and generates the target software code representing the serious game with not only the appearance and educational content, but also the game logic. This research question addresses the “Demonstration” step in the DSRM Process Model. The output will be a software tool, which enables to develop a serious game containing its content and game logic.

***RQ4: What implications can be derived related to the use of the framework, containing the Domain Specific Language and the game editor?***

The aim of this research question is to evaluate the domain specific modeling language as well as the game editor. Therefore different game scenarios will be developed, which will be modeled by educators. First this is done by the use of the DSML followed by the use of the game editor. This research question addresses the “Evaluation” step in the DSRM Process Model. The output of this research question will be the evaluation results for both the DSML and the game editor.

## Results so far

In the process of this research, parts of my research questions have been answered. First an in-depth literature review was conducted to get requirements for the development of serious games. Unfortunately, no publications dealing with requirements for serious games could be found in scientific literature. Instead, a list of serious game influencing factors could be identified, which have an effect on the learning success and motivation of players. So, these factors have been analyzed and at the current stage of my research I have deduced 24 requirements for the development of serious games. Table 1 shows the so far requirements list from RQ1.

Table 1: Requirements list for the development of serious games

	<b>Requirement</b>	<b>Derived from source</b>
1	A serious game must provide object-focused interaction mechanisms and could give the possibility to learn/play in social groups.	[An10], [FJ06], [KKB10], [KM04], [YCG10], [ZS10]
2	A serious game should convert a difficult learning objective into repeated game tasks to enhance knowledge retention.	[An10], [FJ06], [KKB10], [KM04], [YCG10], [ZS10]
3	A serious game must have player-world interaction mechanisms to introduce learning objectives to the player.	[Pr01b], [YLK10]
4	A serious game should provide a progressive way of incremented use of new skills.	[YCG10], [ZS10]
5	A serious game must provide players virtual worlds they can relate to and match to the represented content.	[YCG10]
6	A serious game must provide game mechanics to enable	[GAD02], [YLK10],

	the player to have control over his gaming experience and to explore the virtual world.	[YCG10]
7	A serious game must base on game rules.	[GAD02], [YLK10]
8	A serious game should allow the player to communicate with characters within the game.	[Pr01a]
9	A serious game must have a feedback system to give feedback on user actions.	[KKB10], [KM04], [LC10], [Li09], [YCG10]
10	A serious game must have a Graphical User Interface (GUI) system to show texts and textures.	[KKB10], [KM04], [LC10], [Li09], [YCG10]
11	A serious game must have a system to assess and measure learner's progress.	[Mo08], [Wi09], [ZS10], [Mo08]
12	A serious game must have a system to manage achievements.	[SM07], [YCG10]
13	A serious game must have solvable, clear and understandable goals.	[KM04], [SM07], [Wi09], [YLK10]
14	A serious game should accommodate to learner's style.	[Mo08], [YCG10]
15	A serious game must have a system to measure players' learning progress and adjust to learner's skill.	[Pr01a]
16	A serious game must have different difficulty levels to solve tasks.	[GAD02]
17	A serious game should have game world which reflect the content they represent.	[An10], [FJ06], [Mo06]
18	A serious game must have the ability to display videos.	[GAD02], [Ri09], [ZS10]
19	A serious game must have the ability to play audio.	[GAD02], [Ri09], [ZS10]
20	A serious game should have a character to which the player can identify to.	[An10], [LC10]
21	A serious game must have objects and characters.	[An10], [LC10]
22	A serious game must have virtual worlds that are authentic and exciting for the player with curious and surprising elements.	[GAD02], [SM07], [Wi09]
23	A serious game should have a virtual world parallel to the real world without having consequences on the real world.	[GAD02], [HAB05], [Wi09], [YLK10]
24	A serious game must have a virtual world limited in space and time.	[GAD02], [HAB05], [Wi09], [YLK10]

To answer RQ2 the requirements from RQ1 were used as a basis, to construct the elements for the DSML. Therefore, all requirements have been analyzed individually and language elements, which support the fulfillment of these requirements have been derived. I have noticed, that the requirements 2, 4, 7 and 13 can only be fulfilled by the serious game developer, in this case the educator who designs the game, as they have no

direct influence via structure or logic elements. The educator has to consider these requirements during the game development. The remaining requirements have been analyzed and the outcome of this process is a first version of the Serious Game Logic and Structure Modeling Language (GLiSMo). Table 2 shows which language elements have been derived from which requirements. GLiSMo is a modeling language, which allows the design of the structure as well as the logic of a serious game. The structure describes how a serious game is built and contains the layout of the game world, which characters and objects are included and how the player interaction takes place. Figure 2 provides an insight into the meta model of GLiSMo's structure modeling, which describes the elements and their relationships. This meta model follows the notion already known from UML class diagrams, with associations represented by straight lines and compositions by a straight line with a black rhombus on one end.

Table 2: Language elements of GLiSMo's structure modeling

<b>Element</b>	<b>Derived from requirement</b>
Serious Game Root	No derivation needed
Act and Scene	5, 17, 22, 23, 24
Object	1, 3, 5, 21
Character and Inventory	3, 20, 21
Video and Audio Manager	18, 19
GUI Manager	10
Feedback Manager	9
Reward Manager	11, 12

The *serious game root* element plays a special role, as it is the point of origin for all other elements. These elements can have one or more *acts*, which have similarities to levels in common games. They have one or more goals and are divided by one or more *scenes*. A scene describes a specific place within the game world. *Objects* are placed inside the game world to have the player interact with them. The *character* element represents on the one hand the player itself. On the other hand it represents all social beings within the game world. A character can appear in a specific scene, in an act containing several scenes or in the whole game, like the character played by the player. *Inventory* elements are used for storing objects taken by the player. The *feedback manager* manages the textual feedback given to the player, whereas the *reward manager* controls the scoring and rewarding within a serious game. The playback of audio and video within the game is controlled by the *audio manager and video manager*. Finally, the *GUI Manager* is used to show texts and textures on the screen to display buttons, textboxes, scores and the game menu.

Table 3: Language elements of GLiSMo's logic modeling

<b>Element</b>	<b>Derived from requirement</b>
States	No derivation needed
Action	3, 6, 8
Task/Assessment	9, 11
Adaptation	14, 15, 16



Stream	No derivation needed
Event/Message	No derivation needed
Fork/Join and Branch	No derivation needed

Serious game logic modeling characterizes the behavior of the game and shows how the game reacts on specific actions performed by the player or events occurring during game play. Furthermore, this also describes the assessment of player actions and the game adaption according to the adaption results. Table 3 again shows which language elements have been derived from which requirements.

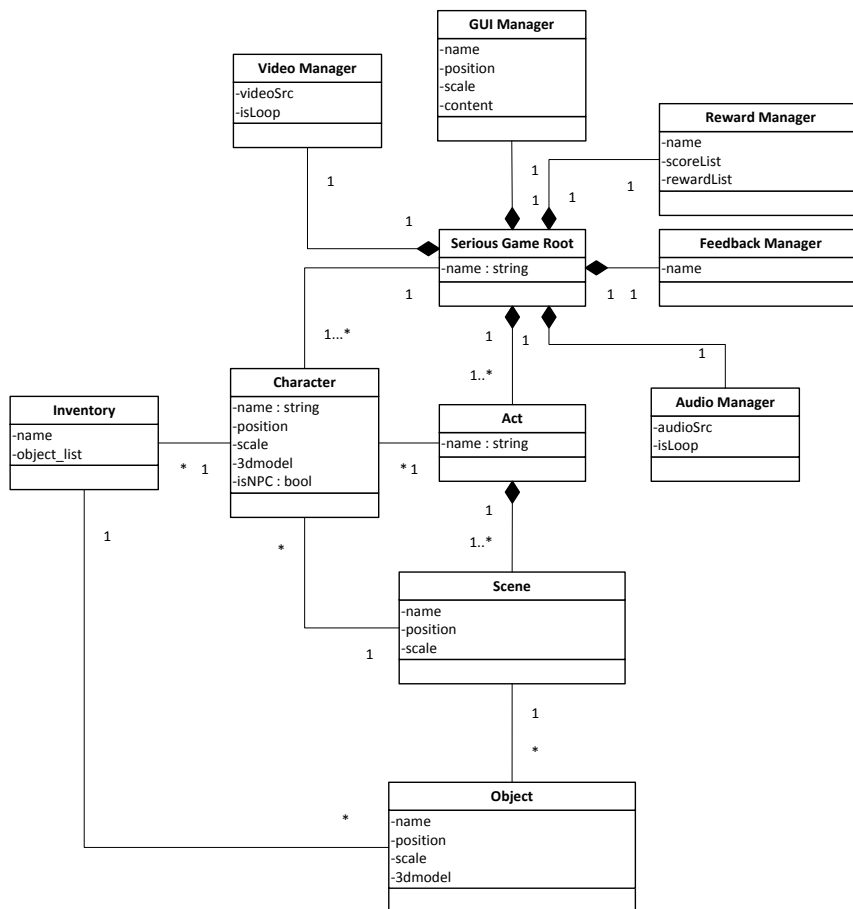


Figure 2: Meta model of GLiSMo's structure modeling

To define the boundaries of a logic model, there are two different states – the *initial-* and *final-state* – which follows the UML activity diagram's notion. *Action* elements enable the player to interact with the game world. Examples for different interaction methods are *select object*, *take object*, *use object*, *put object* or *discussion with character*. *Task*

elements represent different tasks like multiple-choice questions. These elements are associated with *assessment* elements, which evaluate the result of tasks and initiate following processes to give feedback and rewards to the player. Elements called *streams*, *events* and *messages* are used to manage the information and control flow by connecting different elements. This is also supported by using *fork and join* as well as *branch* elements to manage parallel and branched flows. Requirements 14, 15 and 16 describe the need for an adaptation method for serious games. The intention is to monitor the player's learning progress and adjust the learning content and its difficulty to the player's skills. This can be done by simply letting the player select the difficulty level at start and based on this selection a predefined game content will be presented. Another way is to provide different tasks with various difficulty levels and dynamically chose the right task on the basis of previously scored points.

## Exploration of a Multiple-Choice Learning Scenario

To give a first evaluation of GLiSMo's feasibility to model structure and logic of a serious game, I describe a multiple-choice task as an exemplary learning scenario including. The scenario takes place at a company where the player character is representing a trainee. He finds himself in a hall, which is connected to two rooms. One is an office with a NPC, furniture and a computer, the other is a storeroom with a box which contains four computer parts. Furthermore, the NPC holds the key for the storeroom. The first goal is to talk to the NPC and learn about the task. The task is to repair the computer. Therefore the player should get the key from the NPC, open the storeroom and get a specific computer part from the box. This computer part has to be handed over to the NPC, which will check if the correct part was brought. If this is incorrect, then the NPC will give feedback telling him to go back and bring the right part. As soon as the task is accomplished correctly, this scenario ends. Figure 3 shows the structure model and Figure 4 the associated logic model of this described scenario. As this is just a demonstration, the position and scale attributes are not mentioned in this example.

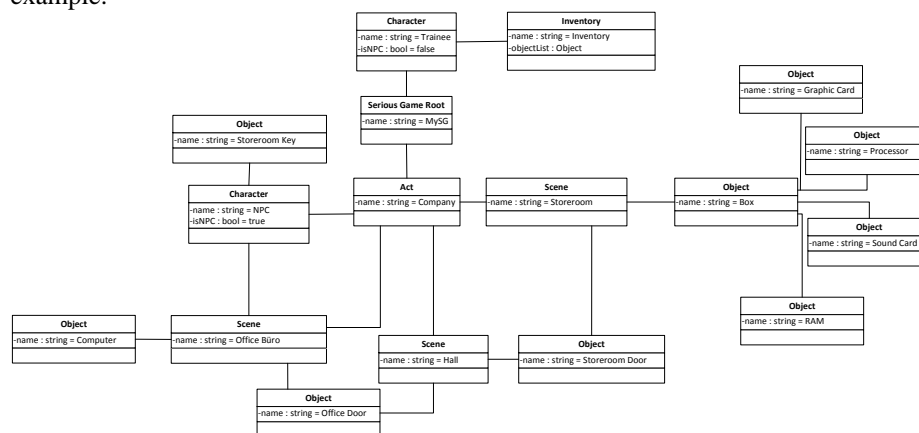


Figure 3: Structure Model of an exemplary serious game scenario

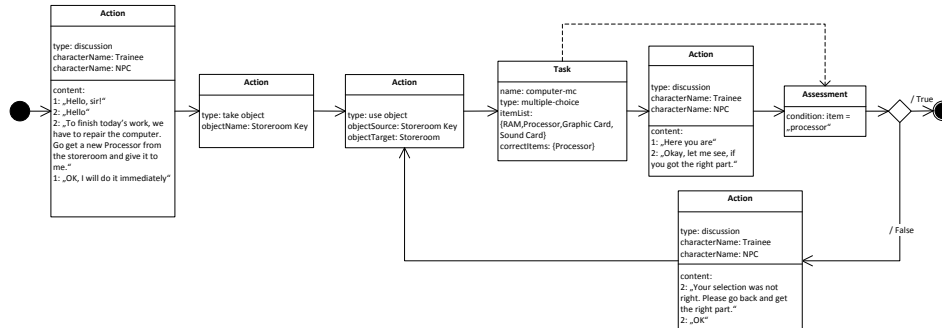


Figure 4: Logic Model of an exemplary serious game scenario

## Research Implications and Limitations

GLiSMo, the domain specific modeling language to model serious games is the theoretical contribution of my research. This can be used as a basis for further research by extending the language by new elements to cover more game genres and accordingly model a larger variety of serious games. The practical implication of this dissertation is that the developed tool gives educators the ability to create serious games on the basis of visually designed models. As this is developed as a plugin for Unity3d, still all functionality of Unity3d can be used to develop serious games. Furthermore the plugin can be extended to design mobile serious games for mobile devices such as tablets and smartphones.

However, the task of game creation in common is a very complex procedure to be supported by just one tool developed within the scope of my dissertation. Therefore, GLiSMo as well as the editor is focusing only on one game genre, in this case point and click graphical adventure games. With a first version of the language and the tool, my aim is to show a proof of concept and provide point of contacts for further research and researchers. At this point in time Unity3d is planned to be used as the required game engine and IDE for development, without influencing my research, as it serves as the basis for proof of concept.

Another limitation of my research is, that I am not focusing on the didactic evaluation of serious games generated with my tool. As described in the introduction, serious game development consists of game design, software development and development and integration of pedagogical elements. Latter will still be developed by the educators. The aim of my research is to support educators in game design as well as in software development, so that by adding pedagogical elements by their own, educators are able to develop serious games adapted to their own learning content.

## Conclusion

This dissertation project aims at enabling non-technical educators to develop serious games on their own. Model driven development seems to be a promising approach to reach this goal. Therefore, my idea is it to provide a framework based on model driven development techniques, which consists of a domain specific modeling language, a visual design tool for serious games and a transformation engine and generator that render the model into compilable code.

## References

- [An10] Annetta, L.A.: The "Is" have it: A framework for serious educational game design. *Review of General Psychology*; **14**; 2010; 105-12
- [BS09] Brennecke, A., Schumann, H.: A General Framework for Digital Game-Based Training Systems. IADIS International Conference Game and Entertainment Technologies, Algarve, Portugal 2009
- [Ch09] Chandler, H.M.: *The Game Production Handbook*. Jones & Bartlett Publishers, Ontario 2010
- [DBB12] De Gloria, A., Bellotti, F., Berta, R.: Building a Comprehensive R&D Community on Serious Games. *Procedia Computer Science*; **15**; 2012; 1-3
- [FJ06] de Freitas, S., Jarvis, S.: A Framework for developing serious games to meet learner needs. *Interservice/Industry Training, Simulation, and Education Conference*, Vol. 2006. NTSA, Baltimore 2006
- [GAD02] Garris, R., Ahlers, R., Driskell, J.E.: Games, motivation, and learning: A research and practice model. *Simulation & Gaming - An Interdisciplinary Journal of Theory, Practice and Research*; **33**; 2002; 441-67
- [Go11] Gomez-Rodriguez, A., Gonzalez-Moreno, J.C., Ramos-Valcarcel, D., Vazquez-Lopez, L.: Modeling serious games using AOSE methodologies. 11th International Conference on Intelligent Systems Design and Applications (ISDA), Cordoba, Spain 2011 53-8
- [HAB05] Habgood, M., Ainsworth, S., Benford, S.: Endogenous fantasy and learning in digital games. *Simulation & Gaming - An Interdisciplinary Journal of Theory, Practice and Research*; **36**; 2005; 483-98
- [KKB10] Kazimoglu, C., Kiernan, M., Bacon, L.: Enchanting e-learning through the use of interactive-feedback loop in digital games. 3rd Conference on Human System Interactions (HSI), Rzeszow, Poland 2010 502-9
- [KM04] Kirriemuir, J., McFarlane, A.: *Literature Review in Games and Learning Futurelab Series*, Ceangal 2004
- [KWB03] Kleppe, A.G., Warmer, J.B., Bast, W.: *MDA explained: the model driven architecture: practice and promise*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA 2003
- [LC10] Lin, H.Z.S., Chiou, G.F.: Modding Commercial Game for Physics Learning: A Preliminary Study. Third IEEE International Conference on Digital Game and Intelligent Toy Enhanced Learning (DIGITEL), Kaohsiung 2010 225-7
- [Li09] Linehan, C., Lawson, S., Doughty, M., Kirman, B.: Developing a serious game to evaluate and train group decision making skills. 13th International MindTrek Conference: Everyday Life in the Ubiquitous Era. ACM, New York, NY, USA 2009 106-13

- [Ma11] Marchiori, E.J., del Blanco, Á., Torrente, J., Martínez-Ortiz, I., Fernández-Manjón, B.: A visual language for the creation of narrative educational games. *Simulation & Gaming - An Interdisciplinary Journal of Theory, Practice and Research*; **22**; 2011; 443-52
- [Mo06] Morales, C., Martínez-Hernández, K., Weaver, G., Pedela, R., Maicher, K., Elkin, E., Danforth, D., Nattam, N.: Immersive chemistry video game. 33rd International Conference and Exhibition on Computer Graphics and Interactive Techniques, New York, NY, USA 2006 50
- [Mo08] Moreno-Ger, P., Burgos, D., Martínez-Ortiz, I., Sierra, J.L., Fernández-Manjón, B.: Educational game design for online education. *Computers in Human Behavior*; **24**; 2008; 2530-40
- [Mou12] Mouaheb, H., Fahli, A., Moussetad, M., Eljamali, S.: The Serious Game: What Educational Benefits? *Procedia - Social and Behavioral Sciences*; **46**; 2012; 5502-8
- [Pe07] Peffers, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S.: A design science research methodology for information systems research. *Journal of Management Information Systems*; **24**; 2007; 45-77
- [Pi07] Pivec, M.: Editorial: Play and learn: potentials of game-based learning. *British Journal of Educational Technology*; **38**; 2007; 387-93
- [Pr01a] Prensky, M.: The digital game-based learning revolution. *Digital game-based learning*; 2001;
- [Pr01b] Prensky, M.: *Digital game-based learning*. McGraw-Hill, New York 2001
- [Ri09] Ritterfeld, U., Shen, C., Wang, H., Nocera, L., Wong, W.L.: Multimodality and interactivity: connecting properties of serious games with educational outcomes. *CyberPsychology & Behavior*; **12**; 2009; 691-7
- [Sc06] Schmidt, D.C.: Model-driven engineering. *Computer*; **39**; 2006; 25
- [SM07] Sørensen, B.H., Meyer, B.: Serious Games in language learning and teaching—a theoretical perspective. 3rd International Conference of the Digital Games Research Association, Kaohsiung, Taiwan 2007 559-66
- [TH08] Tang, S., Hanneghan, M.: Towards a Domain Specific Modelling Language for Serious Game Design. 6th International Game Design and Technology Workshop, Liverpool, UK 2008
- [Va08] Van Eck, R.: Building Artificially Intelligent Learning Games. *Intelligent Information Technologies: Concepts, Methodologies, Tools, and Applications*. IGI Global 2008 793-825
- [Wi09] Wilson, K.A., Bedwell, W.L., Lazzara, E.H., Salas, E., Burke, C.S., Estock, J.L., Orvis, K.L., Conkey, C.: Relationships Between Game Attributes and Learning Outcomes Review and Research Proposals. *Simulation & Gaming - An Interdisciplinary Journal of Theory, Practice and Research*; **40**; 2009; 217-66
- [YCG10] Yusoff, A., Crowder, R., Gilbert, L.: Validation of Serious Games Attributes Using the Technology Acceptance Model. Second International Conference on Games and Virtual Worlds for Serious Applications, Braga, Portugal 2010 45-51
- [YL10] Younis, B., Loh, C.: Integrating serious games in higher education programs. Academic Colloquium, Ramallah, Palestine 2010
- [YLK10] Yessad, A., Labat, J.M., Kermorvant, F.: SeGAE: A Serious Game Authoring Environment. 10th International Conference on Advanced Learning Technologies (ICALT), Sousse 2010 538-40
- [ZS10] Zaibon, S.B., Shiratuddin, N.: Adapting Learning Theories in Mobile Game-Based Learning Development. Third IEEE International Conference on Digital Game and Intelligent Toy Enhanced Learning. IEEE, Kaohsiung 2010 124-8