DOI: 10.22122/jrrs.v15i2.3496 Published by Vesnu Publications

Applying Model Driven Approach for Gamification of Educational Courses: Action Research

Seyedeh Hasti Mousavi-Dehooyi[1], Shekoufeh Kolahdouz-Rahimi[2], Leila Samimi-Dehkordi[3]

Original Article

Abstract

Introduction: The importance of education in today's world is evident for anyone. Nowadays, with the advancement of technology, educational centers seek electronic approaches to train the students. On the other hand, education authorities encounter difficulties like abundant educational material, limited learning time, lack of interactive media, and poor quality of education that negatively influence student's motivation and engagement. An effective solution in this regard is gamification in the electronic learning. Additionally, software developers encounter many complexities in producing courses that use game elements for learning, including lack of precise understanding of the concepts of the field, which makes the development of the software a time-consuming process.

Materials and Methods: In this study, the process of designing and implementing courses was simplified for developers by modeling the concepts of gamification using the model driven approach. A metamodel was designed for e-learning domain using Eclipse framework, and then a model was has developed based on this metamodel that represented a gamification course.

Results: A meta-model was created for modeling gamification of electronic education using a model driven approach and then a model was developed based on this metamodel that indicated a gamification course for educational purposes.

Conclusion: In this study, the learning process becomes more attractive by applying gamification, in addition to facilitating e-course development process, which was usually implemented in web platforms using the model driven approach. Different models may be specified conforming to the metamodel presented in this study according to the requirements of customers. These models may be used in transformation programs to generate implementation code for e-courses.

Keywords: Gamification; Model driven; Education; Metamodel; Developers

Citation: Mousavi-Dehooyi SH, Kolahdouz-Rahimi S, Samimi-Dehkordi L. **Applying Model Driven Approach for Gamification of Educational Courses: Action Research.** J Res Rehabil Sci 2019; 15(2): 107-14.

Received: 09.04.2019 Accepted: 30.04.2019 Published: 05.06.2019

Introduction

In today's world, education is one of the most important needs of human society. History has shown that human progress is achieved through education and learning knowledge. Teaching topics with a high volume and educational materials has always been a challenging for learners and teachers (1). On the other hand, ignoring individual differences and limited education time has reduced motivation among students. Therefore, teachers encounter two major problems in this area, including reduced student motivation and low time and high volume of teaching materials (2). Here, the use of gamification and e-learning courses with a positive impact of learning can enhance motivation and reduce the time required for the teaching process, in addition to facilitating learning the high volume of materials to learners (1). In this regard, training courses are needed to be designed and implemented to be able to use the best elements of the games to gamify these courses. This requires expertise in the field of games and the production of training courses (3). The present study addressed the three domains of Model-Driven Software Engineering (MDSE), training, and gamification.

Games have long been known as one of the

1- MSc Student, MDSE Research Group, Department of Software Engineering, University of Isfahan, Isfahan, Iran
2- Assistant Professor, MDSE Research Group, Department of Software Engineering, University of Isfahan, Isfahan, Iran
3- Assistant Professor, Department of Computer Engineering, School of Engineering, Shahrekord University, Shahrekord, Iran
Corresponding Author: Shekoufeh Kolahdouz-Rahimi, Email: sh.rahimi@eng.ui.ac.ir

enjoyable human activities and are goal-oriented activities that are performed according to the rules defined for them (3,4). Gamification can be defined as the use of game design elements in non-game fields (5). There are several classifications in gamification design, one of the most well-known one is Mechanic-Dynamic-Aesthetic (MDA) framework. In this framework, attention to the dimensions of mechanics, dynamics, and aesthetics is of great importance (5).

Game mechanics describes the specific components of the game at the level of displaying the data and algorithms (5) and may severely affect the motivation and engagement of the user (6). It is important to note that the game mechanics are different from the game rules (5). The game rules determine the approved behaviors that are followed when performing the related mechanics (7). The most common elements of game mechanics include point, leaderboard, levels, and achievement system (5).

Game dynamics describe user behaviors on the game mechanics during the runtime. Therefore, the dynamics show the reason for the user's motivational behavior towards the game mechanics (6). The most common game dynamics include rewards, status, achievements, self-expression, competitions, and altruism (5).

Aesthetics describes the desired emotional responses in the player when interacting with the game system (7) and includes elements such as feeling, narration, challenge, friendly cooperation, exploration, expression, and submission (5).

One of the applications of gamification is to use it to change the behavior of an individual or individuals in a community. In fact, gamification can be regarded as a technology created to influence user behavior without forcing him to change his behavior (6). In order to influence behavior, one must understand how behavior is created and what affects it. Factors influencing behavior creation and motivation include motivation and engagement. Motivation is the desire to do something that promotes human behavior and consists of two types, internal and external (7). Engagement is a sign of emotional desire and involvement in the company and completion of activities (6) and indicate how much a person can engage in doing something.

Regarding the use of the gamification elements, researchers have found that these elements have a positive impact on the engagement, motivation, and overall performance of learners through instant feedback and collaboration (6). In previous studies, the model driven approach has been used to gamify education (8-11), however, limited game elements were used. The aim of the present study is to model gamification in the production of e-learning courses using the model driven approach to increase the abstraction level and make the software developers needless of understanding the details that are not familiar with.

Materials and Methods

In this study, a meta-model was designed for the domain of gamification in the field of education, which included the most important concepts necessary for modeling this domain.

MDSE: Software development consists of activities that are planned to increase the quality of software and increase its production speed. This process has always been along with some complexities for software engineers, most of which have been solved with the help of methodologies designed to develop software (12). One of these methodologies, which has been very popular in recent years by software engineers, is MDSE, which can be defined as a strategy to use the benefits of modeling in software engineering activities (12). The results of various quantitative and qualitative studies have suggested that MDSE methods enhance efficiency and effectiveness in software development (8-11). In MDSE, a simple software formula is created based on a combination of models and transformations (13). Obviously, the models and transformations must be expressed in the form of some notations, which is called a modeling language in MDSE (13). In the model driven concepts, the model can be considered as a representation of a system that by removing unnecessary details, leads to the increased user understanding of the system (14). A metamodel in this approach is a model that describes the elements of modeling and the possible structure of models in an abstract way and defines the structures of a modeling language and their relationships (15). In the model driven approach, transformations are programs that convert models to lower or higher level models in accordance with the specific rules defined in the transformation (15). In this approach, the domain-specific modeling language (DSML) is used to identify the concepts of a particular domain to more focus on that domain and extract all its features and capabilities (13).

The goal of software developers is to produce high quality codes (12). In the model driven approach, many stages and products of software development can be produced (semi-) automatically (14), which leads to an increase in the quality of the software product and code (12). The development process in MDSE is illustrated in figure 1. First, the main concepts of the domains were extracted by the model driven expert and the metamodel was designed based on the main elements of the domains and the relationships among them using Eclipse software (Eclipse 2019, 12th Edition, Eclipse Foundation, Ottawa, Ontario, Canada). Then, using a modeling language creation framework such as Xtext, a DSML was developed. In the next step, using software like Sirius, a graphics editor was created to graphically generate models. Afterwards, with the help of a suitable transformation language such as ATLAS Transformation Language (ATL) or Acceleo, programs were created to transform the created models into codes executable in destination platforms such as Android, web platforms, etc. This code can be generated (semi-) automatically.

In the present study, the key elements employed in the production of the gamification-based education courses were extracted and a metamodel was designed based on these concepts.

Results

After examining the gamification and e-learning domains, the concepts were extracted in the form of the metamodel elements and their characteristics, which are described in the following. All of these entities and the relationships between them are displayed in figure 2.

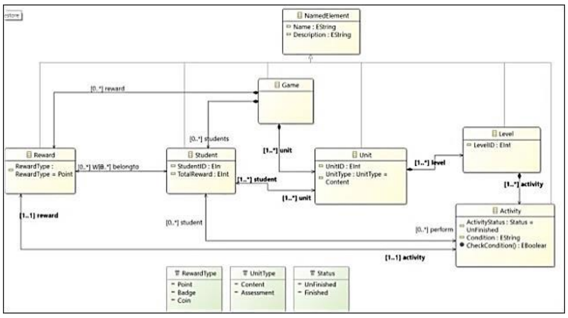

Figure 2. Education gamification metamodel

NamedElement: This element has a name characteristic and description, and all the elements in the metamodel are inherited from this class.

Game: This element is the root class, from which one can access all the metamodel elements.

Unit: Each game class can contain one or more units. This class has the UnitStatus feature, which is an enumerated data type called Status, and specifies whether or not the stage is completed, and the UnitID feature, which shows the unique identifier of each unit. The UnitType feature also specifies the type of the section designed that can be in the form of course content or content evaluation.

Level: Each unit class can contain one or more levels and defines the steps a user must go through. This element has the LevelStatus feature, which has a function similar to the UnitStatus feature in the Unit element.

Activity: This element specifies the activities to be performed on a level. Each level can include one or more activities. It also includes the ActivityStatus, ActivityID, and Condition features, which specify the activity status, activity identifier, and condition required to complete the activity, respectively. This class is related to the Reward class, because for every activity completed by the user, that is rewarded. The CheckCondition method also specifies whether the condition has been met.

Reward: This element indicates the reward received by the user. The RewardType feature of this class determines the type of reward given and is of the enumerated data type of the same name, which can be a point, a badge, and a coin.

Student: displays the user of this program, which has a unique identifier for each user and the sum of the rewards he has received so far. This element is in a many-to-many relationship with the Activity element; as the user can perform different activities and also, the activity can be performed by several users. This element is also related to the Reward element, which determines what rewards each user has received. In addition, it should be determined in which section each user is working and which section he has completed.

With this approach, different metamodels can be produced. The XMI format is one of the models built based on this metamodel, which is demonstrated in figure 3.

Figure 1. Software development process in model-driven software engineering (MDSE)

Figure 3. XMI model produced by the metamodel

This model indicates the gamification for training patterns in software engineering, which consists of a section called Analysis Pattern and includes two stages: Planning and Supporting Pattern. Two activities were designed for the first stage and one activity for the second stage appropriate for them. Furthermore, two users, Ali and Reza, participated in this course section and had to complete the required steps and activities. For each activity performed, the corresponding reward was defined, which would be given to the user if the specified condition was met.

The ultimate goal in MDSE is to produce high quality code automatically or semi-automatically (14). Therefore, the model presented in figure 3 can be utilized to produce models with a lower abstraction level or code. Since modeling as a text is tedious for users and increases the possibility of errors in modeling (8), by creating a graphics editor such as the one shown in figure 4, modeling will be easier for the user. After modeling, programs are written for the model to model or model to code transforms using the transformation languages. The gamification codes created can be implemented on each supportable platform.

Discussion

In the present study, using more diverse game elements and special design with the aim of education in the gamification context, training in various fields was performed by combining education, gamification, and the model driven approach. Moreover, using the key elements to produce education courses, a metamodel was designed and a model following this metamodel was produced. Education has a very special place in today’s world and its trustees are always trying to improve its quality by using modern technologies (16). Therefore, many training centers seek to offer their educational courses electronically (16). Furthermore, education faces challenges such as the lack of motivation among learners (6). In the present study, it was tried to make the learning process more attractive using gamification and, using the model driven approach, make the process of developing e-courses, which are often implemented on web platforms, easier. Therefore, a metamodel was presented to gamify education. Different models of this metamodel have been produced taking into account the customer needs, which can be used in the transformation programs written to generate the e-course implementation code. With the help of the metamodels, various models of these courses are designed by software developers. In the next steps, these models can be converted into a code necessary to be implemented on the destination platforms with the help of the model-to-code transformation programs.

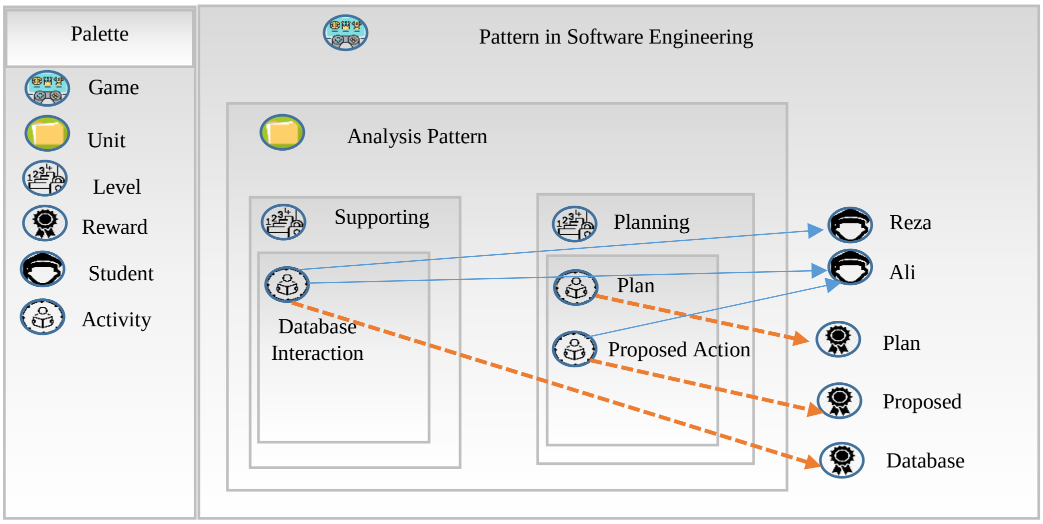

Figure 4. A sample graphics editor

Currently, one of the interesting and controversial issue in education is the gamification-based learning using game design elements in e-learning activities (4). On the other hand, MDSE is one of the new approaches that has been welcomed by developers in recent years (16). Therefore, various investigations have been accomplished in the specific model driven field or gamification and their combination in different scientific areas such as education (3,8-11). In the following, the differences between the present study and other studies are noted.

In a study, Cosentino et al. provided a model driven approach to supporting gamification of education. For modeling the game environment, they designed a game metamodel, and two complementary metamodels, one associated with the software products and the other related to the player's status in the game, as well as a text syntax to define the game components and mechanics. The framework presented in this study collected the project data when an action took place, and performed the game mechanics. It then processed the data generated by the gamification and finally displayed the game information (3). In this approach, the problems arising from fraud and violation of privacy are addressed by encrypting the status models. The framework introduced in this study was applied to the Unified Modeling Language (UML) learning gamification; however, it is also used to learn languages such as Structured Query Language (SQL). One of the drawbacks of this approach was the ignorance of other aspects of gamification, such as competition and progress, which could be applied with elements such as scoreboards and progress bars. Additionally, this framework has been designed specially to UML training and has limited use (3). One of the advantages of the present study over the one by Cosentino et al. (3) is the possibility of creating models for teaching each topic and field. Therefore, different items can be used in teaching.

DoPrado and Lucredio proposed a model driven game development (MDGD) approach to produce electronic games. This method combines several domain specific languages (DSLs) with design patterns and game engines so that the code generated can be integrated with the manual code. It offers three DSLs and three editors for three different aspects of camera, character, and scenario. The developer can produce his own model according to these aspects. The models are then saved in an XML file format and read by the Freemarker template processor, and based on the templates, generate the implementation codes according to the templates. The Java code generated using some design patterns is integrated with the manual code and run on the JMonkey engine. In general, a set of tools and code generators called JMEGenerator is presented, which is an educational tool for managing the course and connecting learners and educational resources with learning activities (13). The study by doPrado and Lucredio used only the model-driven approach to generate game code semi-automatically (13), but in the current study, the game design elements were used in the training section and the model-driven approach was used to model the concepts of the training and gamification domains.

In another study, Jurgelaitis et al. offered a metamodel for UML training that identified the structure of the course and the main elements of gamification. In the proposed metamodel, the semantic section was defined for teaching the principles of the unified process (UP) methodology and the syntax section for teaching different UML diagrams. For the gamification of the UML education, a leveled structure was provided to train specific diagrams and part of the UP methodology at each level, and did not allow access to higher-level content until the previous level had been completed and the required score had been obtained. The game elements used in this course included points and a variety of rewards such as coins, badges, scoreboards, content locking, and trading. The gamified course was implemented on the Moodle platform. In evaluating this work, the scores of the spring semester were examined in two consecutive years, but it was not clear on what basis these scores were determined and what role the points, badges, and other achievements played in calculating these scores (1). The study by Jurgelaitis et al. specifically addressed the UML training and did not allow teaching other training materials (1); however, the present study provides training in many areas.

Savic et al. tried to apply the model driven principles to the development and management of the electronic course. With the help of the platform-independent model (PIM), they made it possible to store and manage courses. The source model consisted of a series of separate components defined by the machine-readable method, each component showing a specific aspect. The source model consisted of the components of Learning objectives, Learning source, and Instructional strategy. The model used E-Learning Instructional Design Meta-Language (ELIDL). This language was an XML-based meta-language that selected an abstract syntax for defining rules and organized courses (14). This PIM could be converted into four platform-specific models (PSMs) and into the Learning Activity Management System (LAMS), Shareable Content Object Reference Model (SCORM), IMS Learning Design, and Sakai formats. Savic et al. provided an architecture for the course management system, and in a case study, the web programming course was modeled and then transformed into the four afore-mentioned PSMs. Their study did not use game design elements to gamify their electronic courses (14); however, in the present study, the game elements were carefully selected to create motivation and engagement among learners.

In another study, Arawjo et al. presented a digital educational game called Reduct, which took a new approach for teaching the basic concepts of JavaScript beginner programming. These concepts included functions, Boolean data type, comparison operators, conditional statements, and mapping of functions on sets. In this work, a graphic block was employed to design the problems and the student was asked to write the appropriate code in the blank space according to the desired answer. The player ran the code written by himself and in the same way, performed 72 levels designed in this game as long as he wanted. The researchers conducted two laboratory and online evaluations of the game, which were well accepted by the participants in terms of quality and attractiveness of learning (15). The difference between the present study and the study by Arawjo et al. was that a serious game was designed in this work (15); while the present study was of a gamification nature, and the two are different. The second difference was the specific education of programming and the game was not able to teach other fields, and the third difference was that in the present study, the model driven approach was employed to produce software products, but the software applied in the study by Arawjo et al. was created by the conventional software development methods (15).

Bucchiarone et al. considered the impossibility of reviewing or introducing new game elements and mechanisms during system execution to be an important challenge in the development and evolution of the gamification systems. Therefore, they suggested a gamification designing framework, which consisted of well-defined languages for game design and its components and behavioral details. This framework included a layered architecture in which the highest layer formed the foundation of other layers. The architecture included five Gamification Model Languages (GaML) to introduce the basic elements of the gamification systems, the Game Model Language (GML), which provided more details on how to assemble game components to create a gamification app, and the Game Instance Model Language (GiML) was used to create various samples of a game defined in GaML. Game Simulation Language (GsML) was applied to simulate the mode changes that occur in the game, and finally, Game Adaptation Model Language (GadML) included a suggestion system that could create challenges given the skill level and background of the player (16). The study accomplished by Bucchiarone et al. applied four metamodels and modeling languages to create gamification systems, complicating the design process for the user when using these languages (16); However, in the present study with one metamodel, the user will be able to more easily create a gamification system.

Limitations

In the present study, other game elements such as the store to replace or exchange various rewards received by the user could be considered which due to the limitations of the implementation platforms, it was not possible to add this element or similar elements to the metamodel. In addition, there are many ready-made frameworks on which the output code of the present study can be implemented, but in many existing frameworks, there are no sections for monitoring, privacy, and prevention of fraud, or these sections can be used for a fee.

Recommendations

For future work,, an editor can be developed to generate courses graphically to make modeling easier for the user. Moreover, the game elements introduced in the present study can be used in different parts of the health area and for different age groups. For example, in order to increase motivation in the field of fitness or consumption of appropriate foods or production of gamified education courses in the field of medicine and rehabilitation, the metamodel designed in the present study can be used.

Conclusion

In the present study, it was tried to make the learning process more attractive using gamification, and with the help of the model driven approach, make the process of developing electronic courses, which are often implemented on web platforms, easier. Different models of the metamodel presented in the this study can be produced based on the customer's needs, which can be used in the transformation programs with the aim of generating the code for implementing e-courses.

Acknowledgments

The present study was derived from the proposal of the MSc degree program in software engineering under the supervision of Dr. Shekoufeh Kolahdouz-Rahimi and consultation of Dr. Leila Samimi-Dehkordi, and was approved by the University of Isfahan, Isfahan, Iran.

The present study is one of the articles submitted to the Secretariat of the 5th International Conference on “Computer Games; Challenges and Opportunities” with a special focus on therapeutic games (February 2020, Isfahan), which was praised by the editorial board of the Journal of Research in Rehabilitation Sciences (JRRS). The authors would like to appreciate the Cyberspace Research Institute, National Cyberspace Center for supporting the publication of this article. The Entertainment Industry Innovation Center, University of Isfahan, which played an important role in collecting data and accomplishing this project is also appreciated.

Authors’ Contribution

Seyedeh Hasti Mousavi-Dehooyi: Study design and ideation, providing study equipment and samples, data collection, analysis and interpretation of results, manuscript preparation, specialized manuscript evaluation in terms of scientific concepts, confirmation of the final manuscript to be submitted to the journal office, responsibility for maintaining the integrity of the study process from the beginning to publication, and responding to the referees’ comments; Shekoufeh Kolahdouz-Rahimi: Study design and ideation, attracting financial resources for the study, supportive, executive, and scientific study services, analysis and interpretation of results, manuscript preparation, specialized statistical services, specialized manuscript evaluation in terms of scientific concepts, confirmation of the final manuscript to be submitted to the journal office, responsibility for maintaining the integrity of the study process from the beginning to publication, and responding to the referees’ comments; Leila Samimi-Dehkordi: study design and ideation, supportive, executive, and scientific study services, analysis and interpretation of results, specialized statistical services, manuscript preparation, specialized manuscript evaluation in terms of scientific concepts, confirmation of the final manuscript to be submitted to the journal office, responsibility for maintaining the integrity of the study process from the beginning to publication, and responding to the referees’ comments.

Funding

The present study was based on the analysis of a part of the information extracted from the MSc thesis of software engineering and was implemented with the financial support of the University of Isfahan. The University of Isfahan did not comment on data collection, analysis, and reporting, manuscript preparation, and final approval of the study for publication. The present study was published in JRRS, with the financial support of the Cyberspace Research Institute of the National Cyberspace Center, and sponsored by the Fifth International Conference on Computer Games with an Approach to Therapeutic Games. This research institute did not contribute to the designing, compiling, and reporting this study.

Conflict of Interest

The authors declare no conflict of interest. Dr. Shekoufeh Kolahdouz-Rahimi has attracted funding from basic studies related to the present study from the University of Isfahan and has been working as an assistant professor at the School of Computer Engineering, University of Isfahan since 2013. Dr. Leila Samimi-Dehkordi also works as an assistant professor and faculty member at the School of Engineering, Shahrekord University, Shahrekord, Iran and is a graduate of the PhD program in Software Engineering, University of Isfahan. Seyedeh Hasti Mousavi-Dehooyi has been a graduate student of Software Engineering, School of Computer Engineering, University of Isfahan since 2018.

References

1. Jurgelaitis M, Ceponiene L, Ceponis J, Drungilas V. Implementing gamification in a university-level UML modeling course: A case study. Comput Appl Eng Educ 2019; 27(2): 332-43.
2. Topalli D, Cagiltay NE. Improving programming skills in engineering education through problem-based game projects with Scratch. Comput Educ 2018; 120: 64-74.
3. Cosentino V, Sebastien G, Cabot J. A model-based approach to gamify the learning of modeling. CEUR Workshop Proceedings 2017; 1954: 15-24.
4. Groh F. Gamification: State of the art definition and utilization. Proceedings of the 4th Seminar on Research Trends in Media Informatics; 2012 Feb 14; Ulm, Germany. p. 39-46. 1-1-2012.
5. Matallaoui A, Hanner N, Zarnekow R. Introduction to gamification: Foundation and underlying theories. In: Stieglitz S, Lattemann C, Robra-Bissantz S, Zarnekow R, Brockmann T, editors. Gamification: Using game elements in serious contexts. Cham, Switzerland: Springer International Publishing; 2017. p. 3-18.

6. Al Marshedi A, Wanick V, Wills GB, Ranchhod A. Gamification and behaviour. In: Stieglitz S, Lattemann C, Robra-Bissantz S, Zarnekow R, Brockmann T, editors. Gamification: Using Game Elements in Serious Contexts. Cham, Switzerland: Springer International Publishing; 2017. p. 19-29.
7. Zichermann G, Cunningham C. Gamification by design: Implementing game mechanics in web and mobile apps. Sebastopol, CA: O'Reilly Media; 2011.
8. Volter M, Stahl T, Bettin J, Haase A, Helsen S, Czarnecki K, et al. Model-driven software development: Technology, engineering, management. Hoboken, NJ: Wiley; 2013.
9. Brambilla M, Cabot J, Wimmer M, Baresi L. Model-driven software engineering in practice. 2nd ed. San Rafael, CA: Morgan and Claypool Publishers; 2017.
10. Frank U. Domain-specific modeling languages: requirements analysis and design guidelines. In: Reinhartz-Berger I, Sturm A, Clark T, Cohen S, Bettin J, editors. Domain engineering: Product lines, languages, and conceptual models. Berlin, Heidelberg: Springer Berlin Heidelberg; 2013. p. 133-57.
11. Sendall S, Kozaczynski W. Model transformation: the heart and soul of model-driven software development. IEEE Software 2003; 20(5): 42-5.
12. Wang D, Hoppe HU. A model-driven approach to developing a web-based environment to support problem-based learning [Thesis]. Duisburg, Germany: University of Duisburg-Essen; 2018. 2019.
13. DoPrado EF, Lucredio D. A flexible model-driven game development approach. Proceedings of the 9th Brazilian Symposium on Components, Architectures and Reuse Software. 2015 Sep 21-22; Belo Horizonte, Brazil. p. 130-9.
14. Savic G, Segedinac M, Milenkovic D, Hrin T, Segedinac M. A model-driven approach to e-course management. Australas J Educ Technol 2018; 34(1): 14-29.
15. Arawjo I, Wang CY, Myers A, Andersen E, Guimbretiere F. Teaching programming with gamified semantics. Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, CHI '17; 2017 May 6-11; Denver, CO, USA. p. 4911-23.
16. Bucchiarone A, Cicchetti A, Marconi A. GDF: A gamification design framework powered by model-driven engineering. Proceedings of the 22nd International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C); 2019 Sep 15-20; Munich, Germany. p. 753-8.