Available online at www.sciencedirect.com

ScienceDirect

Procedia - Social and Behavioral Sciences 228 (2016) 436 – 442

Procedia
Social and Behavioral Sciences

2nd International Conference on Higher Education Advances, HEAd´16, 21-23 June 2016, València, Spain

Software Engineering Management Education through Game Design Patterns

Nuno H. Flores[ab]*, Ana C.R. Paiva[ab], Pedro Letra[a]

[a]*INESC TEC, Portugal*
[b]*Department of Informatics – Faculty of Engineering of the University of Porto, Portugal*

Abstract

Software engineering (SE) is an area with a wide range of concepts and knowledge. Such diversity of topics, requires the application of different teaching and learning techniques for an effective education. Serious Games is one of such techniques, yet its design tends to be complex, currently lacking a map of game design standards that comply with SE education requirements. This paper presents a process to identify the game design patterns that can be effective for teaching software engineering, specifically the software project management topic. Firstly, it begins by identifying the relationship between game design patterns and teaching and learning functions based on literature review. Secondly, it filters which of those teaching and learning functions is most relevant to software project management education, according to SE education specialists. Finally, it validates the relationship between game design patterns and software project management education through an empirical study conducted with master students. The results can be used as a basis for designing and developing serious games for teaching software project management.

Peer-review under responsibility of the organizing committee of HEAd´16

Keywords: Software Engineering Education; Software Engineering Project Management; Game Design Techniques; SimSE.

* Corresponding author.
E-mail address: nuno.flores@fe.up.pt

Peer-review under responsibility of the organizing committee of HEAd´16
doi:10.1016/j.sbspro.2016.07.067

1. Introduction

Software Engineering (SE), as a domain area, encompasses a wide and dense range of concepts and knowledge. The SWEBOK (Swebok, 2014), an organizational structure proposed by the IEEE, promotes a common and consistent understanding of those concepts, and how they relate to other areas. Effectively acquiring such diverse knowledge relies on a considerable amount of learning and teaching techniques (Claypool, 2005)(Schilling, 2012)(Yu 2014). One such technique is the use of *serious games*, but the inherent complexity of designing such games prevents its use on a large scale (Westera, 2008). Furthermore, the requirements for an effective and successful game do not always match the requirements for a proper SE learning task. This paper describes the applied process to elicit the learning and teaching functions (Grosser, 2007) relevant for SE education, focused on its *management* sub-area (according to the Swebok), as well as the identification of game design patterns (Bjork, 2004) that cover those specific functions. The resulting set should promote a more focused and effective design of game-based techniques for teaching SE topics. Validation of results relied on empirical methods, namely surveys and questionnaires to SE teachers and a case study performed with students, using a serious game designed for SE education (Navarro, 2006).

The results presented in this paper come from a second iteration of the presented process, performed to confirm and consolidate the preliminary results obtained in a previous iteration (Letra, 2015). This second iteration allowed the process to be refined in order to eliminate some bias thus improving the results quality, and to evaluate the game experience individually, instead of in pairs.

2. The ABC of Game-Based Software Engineering Management Education

When designing a game for teaching any kind of subject, the main concern should be to, not only keep the learner in a “flow” state (Csikszentmihalyi, 1975), but to meet the cognitive needs and nurture the player’s learning. The theoretical statement behind the work presented in this paper can be visualized as a triangle. In one vertex there is the cognitive aspects of learning, on another vertex, there are the game design best practices that create an effective gaming experience and on the other vertex there is the specific knowledge domain to be taught (or learned). Combining and connecting these 3 key aspects should provide a viable framework for developing game-based education solutions for any specific topic. To instantiate with each of these elements, the authors surveyed the literature in search of existing solutions.

2.1. Learning and Teaching Functions

To cope with the cognitive aspects of learning, the authors reviewed the work of Grosser (2007). Based on the contribution of Shuell and Muran (1994), Grosser created a list of 22 learning and teaching functions (LTFs) that cover all the pedagogical scope. In its definition, a *learning function* (*LF*) regards the learner's point of view on how to link new information to prior knowledge, how to organize information, and how to acquire cognitive and meta-cognitive knowledge. A *teaching function* (*TF*) defines the teacher's goal at ensuring the learner has the proper equipment (i.e., using the proper learning functions) in order to engage with the learning material in a meaningful way. Thus, an important TF is to identify and thoroughly analyse those LFs executed by learners, assisting them in acquiring and executing those LFs. LTFs can be grouped in 5 distinct categories (Figure 1).

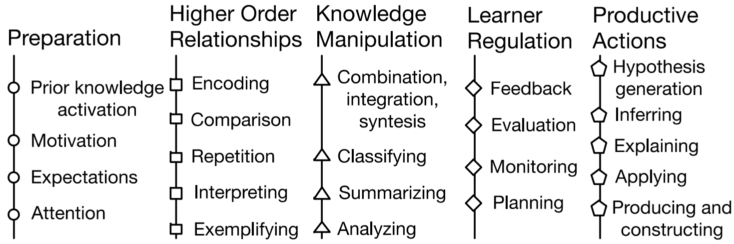

Fig. 1. Learning and Teaching Functions (LTFs) according to Grosser (Grosser, 2007).

For instance, *Prior knowledge activation*, focuses on reminding students of prerequisite information or asking oneself what is already known about the topic being learned, whereas *Interpreting*, aims at assisting learners in converting information from one form of representation to another. A description of each LTF can be found in (Grosser, 2007).

2.2. Game Design Patterns

To design an effective game, Bjork and Holopainen (2004) developed a collection of 296 design patterns relevant to games, most specifically, design of games. Design patterns are reusable good solutions for recurring problems within a specific context (Alexander *et al.*, 1977). As such, a game designer can rely on these "best practices" as building blocks to guide the development of an effective game, considering its specific requirements context. These patterns are divided in 11 categories (see Figure 2), regarding 4 different views on games: holistic, boundaries, time and structure.

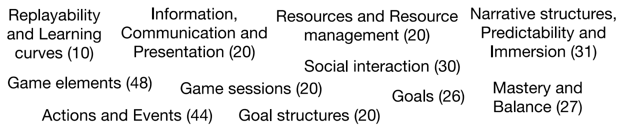

Fig. 2. Game Design Pattern categories with number of patterns between "()" . Source: Bjork (2004)

Patterns such as *Role-playing* (Social interaction), *Randomness* (Mastery and Balance), *Asymmetric Information* (Information, Communication and Presentation) or *Rewards* (Actions and Events) are just a few examples of the whole pattern set.

2.3. The ABC Triangle

The domain of knowledge focused by the presented work is SE project management. As such, the completed key aspect triangle can be viewed in Figure 3.

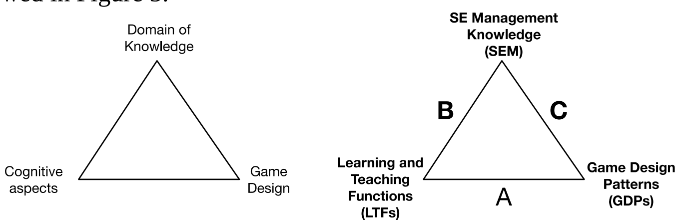

Fig. 3. The key aspect ABC triangle, instantiated to the presented work context.

With this in mind, the main question a game designer poses when conceiving and devising an educational game for a specific context is: "*What game design patterns to use so that the specific cognitive aspects of the specific domain of knowledge are covered?*". In order to answer this question, all edges of the triangle must be known, that is, LTFs and GDPs (edge A), LTFs and SEM (edge B) and GDPs and SEM (edge C) must map onto one another. Therefore, the main goal of the proposed approach is to provide evidence that these aspects relate, that is, there is a mapping between the concepts, which correspond to the "edges" of the triangle (named "A", "B", and "C", thus the "ABC" triangle metaphor).

3. Methodology

The applied process to define the mappings (edges) between the concepts (vertices) of the triangle proceeded as follows:

Step 1: Mapping between LTFs and GDPs – Edge A

Through literature review, the authors found that this step had already been done by Kelle *et al.* (2011), where a map was established between Shuel and Moran's learning and teaching functions and Bjork and Holopainen's game design patterns. In a nutshell, they connected the LTFs to the pedagogical taxonomies of Gagne (1965), Heinich *et al.* (2001), Kolb (1984) and Keller (1983) as to elicit the LTF pedagogical requirements, later selecting the GDPs that would best support there requirements.

Step 2: Mapping between LTFs and SEM – Edge B

In order to identify and validate which specific LTFs are used in SEM education, a questionnaire was presented to high education professionals. Its questions inquired which LTFs would be more relevant, concerning a specific sub-topic of SEM, namely Initiation and Scope Definition, Software Project Planning, Software Project Enactment, Review and Evaluation, Closure, SE Measurement, SE Management Tools. The expected results were a common set of LTFs, specific to SEM education.

Step 3: Mapping between GDPs and SEM – Edge C

Taking on the results of the previous steps, the specific LTFs (step 2) were then mapped (step 1) to GDPs, thus providing the mapping between these and SEM knowledge. This mapping was validated through an experiment with students, where they played an educational (serious) game about SEM, which contained the specific game design patterns. The expected results were based on knowledge intake and overall satisfaction on playing the game.

4. Survey to software engineering management education professionals

In order to minimize the inherent ambiguity of the LTFs and narrow down the relevant set to the present context, a survey was conducted, aimed at SE education professionals. The targeted subjects were high-education specialists, from both national and international institutions, ranging from 5 to 20 years of experience teaching SEM. The subjects would then answer a questionnaire where they would indicate which of the 22 LTFs would be most suited for effective learning of a specific sub-area of SEM.

Analysing the 10 (together with the other 5, from the first iteration (Letra, 2015)) responses obtained, it was determined that, for each sub-area, there was, at least, one LTF that all subjects indicated as 100% relevant to provide effective learning. Other LTF's were not completely agreed upon, so they were discarded. As such, and using the mapping from LFTs to GDPs (edge A), and the results of the survey (mapping of LFTs to SEM, **edge B**), a mapping between GDPs and SEM (**edge C**) can be obtained, as shown in Figure 4.

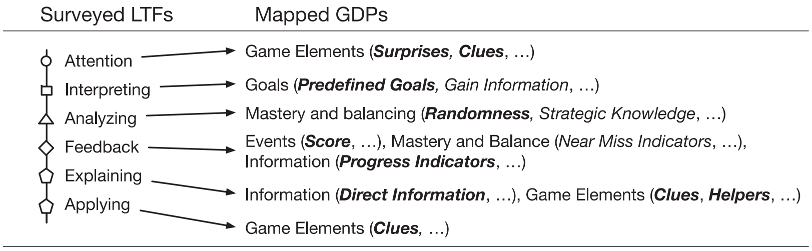

Fig. 4. Selection of Game Design Patterns from the LTFs deemed (surveyed) relevant to SEM Education.

5. SimSE Case Study

To validate the effectiveness of the mapped game design patterns, an empirical study involving students was conducted. The main goal was to observe the students playing an educational (serious) game about SE project management, containing the mapped GDPs (in bold, in Figure 4), while measuring learning effectiveness. The selected game was SimSE (Navarro, 2006), a simulation game for learning about SE project management.

5.1. Subjects

The experiment subjects were 26 MSc students of the 2nd year of the Integrated Master in Informatics and Computing Engineering (MIEIC), from Faculty of Engineering of the University of Porto. The general profile of 2nd year students had, if at all, very basic notions of SE project management, thus fitting the purpose of the experiment. Nevertheless, all students where scrutinized for their background and grades (average), being split into two homogenous groups: Group A would undertake experiment treatment A, while Group B would undertake experiment treatment B.

5.2. Experiment Protocol

The student groups undertook the following experiment protocol, as depicted in Figure 5.

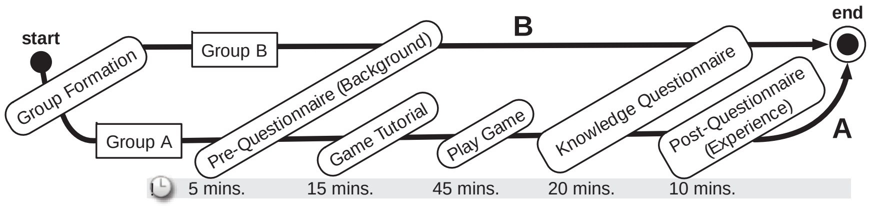

Fig. 5. Empirical study protocol.

Treatment "A" (Group A): Group A answered a pre-questionnaire for background check, watched a game tutorial (to learn how to play the game), played the SimSE game and x§afterwards answered a questionnaire (available at www.fe.up.pt/~apaiva/QuestionnaireCSE15.pdf) to evaluate the knowledge intake about SEM gained from playing the game. Finally, a post-questionnaire was answered to discard possible external threats to validity. Opposite to the first iteration, this time the students played the game individually, instead of in pairs.

Treatment "B" (Group B): Group B answered the same pre-questionnaire as Group A and answered the knowledge intake questionnaire. Acting as a "control" group, performing only these two steps allowed for an initial measurement of specific knowledge about SEM, so that these results could be compared with the same results of Group A, but after playing the SimSE game.

5.3. Results

Analysis of results from both Pre- and Post-questionnaires indicated that both the previous background knowledge about SEM and external threats to validity where conveniently discarded (See Table 1.). Both questionnaires were designed using a 5-point Likert (1932) scale.

Table 1. Pre- and Post-Questionnaires results.

Pre-Questionnaire (Background)					Post-Questionnaire (Overall Satisfaction)			
Question	Group A	Group B	*Goal*		Question	Group A	Goal	
Vast SE Knowledge	$\bar{x} = 1{,}89$, $\sigma = 0{,}56$	$\bar{x} = 2{,}13$, $\sigma = 0{,}53$	$x < 3$	✔	Environment suitable	$\bar{x} = 4{,}49$, $\sigma = 0{,}27$	$x > 3$	✔
SEM Experience	$\bar{x} = 2{,}42$, $\sigma = 0{,}21$	$\bar{x} = 2{,}15$, $\sigma = 0{,}71$	$x < 3$	✔	Enjoyed game	$\bar{x} = 4{,}21$, $\sigma = 0{,}52$	$x > 3$	✔
SE Team Experience	$\bar{x} = 1{,}54$, $\sigma = 0{,}70$	$\bar{x} = 2{,}37$, $\sigma = 0{,}26$	$x < 3$	✔	Game visually confortable	$\bar{x} = 4{,}39$, $\sigma = 0{,}60$	$x > 3$	✔
Know SimSE	$\bar{x} = 1{,}00$, $\sigma = 0{,}00$	$\bar{x} = 1{,}00$, $\sigma = 0{,}00$	$x < 3$	✔	English interfered	$\bar{x} = 1{,}04$, $\sigma = 0{,}62$	$x < 3$	✔
Played SimSE	$\bar{x} = 1{,}00$, $\sigma = 0{,}00$	$\bar{x} = 1{,}00$, $\sigma = 0{,}00$	$x < 3$	✔	Tutorial helpful	$\bar{x} = 3{,}52$, $\sigma = 0{,}24$	$x > 3$	✔

Comparing responses from both knowledge questionnaires (Group A and Group B) a **39,12% score improvement** was observed (backing the 36,54% score achieved at the first iteration), indicating that the game actually promoted the intake of knowledge about SEM.

6. Conclusions

This paper described the process of gathering and validating a set of game design patterns relevant for Software Engineering Management education (SEM), using an existing mapping technique from learning and teaching functions to game design patterns. Validation of the learning and teaching functions specific for SEM, was accomplished through surveys to SE education specialists, while an empirical study placed computer science students with low knowledge of SEM playing the game "SimSE", as to validate the identified game design patterns. The results showed a clear improvement in knowledge intake after playing the game, either individually or in pairs. These results may be used in the design and development of games for SE education in general, noting that to improve the quality of the game, the maximum number of the identified game design patterns should be considered, covering as much learning functions as possible.

Acknowledgments

This work is financed by the ERDF – European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 Programme within project «POCI-01-0145-FEDER-006961», and by National Funds through the FCT – Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) as part of project «UID/EEA/50014/2013».

References

Alexander, C., Ishikawa, S. & Silverstein, M. (1977). *A pattern language: Towns, Buildings, Construction.* Oxford University Press, UK.

Bjork, S. & Holopainen, J. (2004) *Patterns in Game Design*, C. River Media, Boston, MA.

Claypool, K.,Claypool, M. (2005). Teaching software engineering through game design. *SIGCSE Bull.* 27(3), 123-127.

Csikszentmihalyi, M. (1975) Play and intrinsic rewards. *Journal of Humanistic Psycology*, 15(3), 41-63

Gagne, R. M. (1965). *The Conditions of Learning*, Holt, Rinehart and Winston, Austin, TX.

Grosser, M. (2007). Effective teaching: linking teaching to learning functions, *South African Journal of Education*, 27(1), 37-52

Heinich, R., Michael, M., Smaldino, S. E. & Russell, J. D. (2001) *Instructional Media & Technologies for Leraning, 7th ed.*, Prentice-Hall, Upper Saddle River, NJ, USA.

Kelle, S., Klemke, R. & Specht, M. (2011). Design patterns for learning games, *Int. J. Techonology Enhanced Learning*, 3(6), 555-569

Keller, J. M. (1983) Motivational design of instruction. *Instructional Design Theories and Models: An Overview of their Current Status*, 1, 386-434, Routledge, NY, USA.

Kolb, D.A. (1984) *Experential Learning: Experience as a Source of Learning and Development*, Prentice-Hall Englewood Cliffs, NJ, USA.

Letra, P., Paiva, A. C. P., Flores, N. (2015). Game Design Techniques for Software Engineering Management Education. *18th Int. Conf. Computation Science and Engineering*, Porto, Portugal.

Likert, R. (1932) A technique for the measurement of attitudes. *Archives of Psychology*, 22, 1-55.

Navarro, E. (2006). SimSE: A Software Engineering Simulation Environment. University of California, Irvine, USA.

Schilling, W. W. & Sebern, M. J. (2012), Teaching Software Engineering: An Active Learning Approach. *2012 ASEE Annual Conference*, San Antonio, Texas, USA.

Shuell, T. J. & Moran, K. A. (1994). Learning theories: historical overview and trends. *The International Encyclopedia of Education, 2nd ed.*, 3340-3345, Pergamon, Oxford, UK.

Swebok. (2015). Guide to software engineering body of knowledge. IEEE Computer Society. Accessed at September 2015 from http://www.swebok.org.

Westera, W., Nadolski, R., Hummel, H. G. K. & Wopereis, I. G. J. H. (2008). Serious Games for higher education: a framework for reducing design complexity. *Journal of Computer Assisted Learning*, 24(5), 420-432

Yu L. (2014). *Overcoming Challenges in Software Engineering Education: Delivering Non-Technical Knowledge and Skills (1st ed.)*. IGI Global, Hershey, PA, USA.