International Journal of Engineering Education Vol. 28, No. 6, pp. 1316–1325, 2012
Printed in Great Britain

0949-149X/91 $3.00+0.00

e-SPM: An Online Software Project Management Game*

YVES WAUTELET
Hogeschool-Universiteit Brussel, Stormstraat, 2, B-1000 Brussels, Belgium. E-mail: yves.wautelet@hubrussel.be

MANUEL KOLP
Université catholique de Louvain, Place des Doyens, 1, B-1348 Louvain-la-Neuve, Belgium. E-mail: manuel.kolp@uclouvain.be

Today's software development has become a complex task and no one has the required skills or time to solve a sophisticated problem on his own. Software development needs the involvement of different roles and people having to use concepts and ideas for which they need to share a common understanding. In such a context, several software development methodologies have appeared in the last thirty years. Those methodologies use different development life cycles, one of the most famous being the iterative one notably used by the Unified Process (UP) and agile methods. To teach software project managers and computer science students the required skills to deal with such development processes few approaches exist. Indeed, nowadays one mostly learns from ex-cathedra courses and books or on an empirical basis when involved into 'real life' projects. Both learning approaches have major flaws since 'theory-only' fails to highlight the practical dimension and empirical learning exposes the organization to damageable mistakes. e-SPM is an online multi-users game simulating the tasks of a software project manager facing user requirements, development planning, human resources allocation, budget constraints, risk and quality management within a UP/UML project in the steel industry. The originality of the game resides in its design based on the documentation and experience of previously performed real-life projects. The game mainly focuses on hard (technical) skills for project managers but also introduces random events showing to the player which soft skills are required in practice; it has been validated using the competencies framework of a business school.

Keywords: software project management; e-learning; unified process; game modeling

1. Introduction

Adequate decision taking is from primary importance for firms operating in today's highly competitive and fast evolving environments notably within the development of systems managing the fundamental resource of the 21st century: *information*. Indeed, software development is becoming increasingly complex. Stakeholders' expectations are growing higher while the time to market has to be as low as possible. In order to be competitive in such markets, analysts, project leaders, software developers need adequate methodologies to model the organization, capture requirements and build efficient and flexible software systems. Developing software is unfortunately not an exact science. As pointed out in [1], effective project management is a combination of hard and soft skills meaning that technical skills alone do not constitute a guarantee of success. An efficient project manager should consequently be familiar with development processes, techniques and tools but also acquire abilities going beyond purely technical aspects [2–5] such as communication, team building, flexibility and creativity, leadership, organizational effectiveness, stress management, time management, change management, trustworthiness or conflict management.

The aim of this research is to build up and validate a software project management role game for object-oriented development driven by the *Unified Modeling Language* (*UML*) [6] models and following the *Unified Process* (*UP*) [7, 8]. This game takes a series of engineering concepts as fundamentals around disciplines that have to be repeated iteratively during each of the four phases[1]. The disciplines, the activities performed during them, the models taken as input or produced as output to these activities, are part of the gamer dimensions i.e., the aspects he has to deal with to successfully achieve his goals. The project management framework is consequently central into this game which primarily focuses on hard skills. The game however addresses more than only project planning (see [9] for a characterization). It is, indeed, based on the simulation of previously performed real life development projects in the steel industry. Random elements based on experience and lessons learned within this case study have been included. This does not constitute an exhaustive coverage of identified soft skills for project management but at least stimulates the player to overview what those are. The project and its artifacts are thus, within the game life cycle, subject to dynamic evolution.

The paper is structured as follows. Section 2 describes the research method notably through the project context and the development approach. Section 3 defines the core game model as well as

[1] As defined by the UP, i.e. *Inception*, *Elaboration*, *Construction* and *Transition*.

* Accepted 20 July 2012.

the gamer dimensions and the main game scenario. Section 4 highlights the game validation method. Finally, Section 5 concludes the paper.

2. Research method

This section presents the research context, briefly describes the Carsid project and defines the development approach.

2.1 Research context

The idea of developing an online software project management game appeared in 2007 at the *Université catholique de Louvain* (*UCL*) in the context of courses in software engineering and project management. The faculty members conducting the present research were indeed involved in teaching activities lacking of concrete exercises to illustrate the project management concepts taught in class and labs. The game was consequently targeted to offer a practical approach to master students in Computer Science from the *Louvain School of Engineering* (*LSE*) and master students in business engineering of the *Louvain School of Management* (*LSM*). The willingness was to offer a tool illustrating and applying theoretical knowledge given in ex-cathedra courses onto a real life project.

It was nevertheless not possible to cover all the theoretical aspects taught during those courses so that it was decided to restrict it to a particular development process. Indeed, part of those courses were (and are still) structured around the UP and UML models. It was consequently decided to develop a game in line with UP and dealing with the artifacts it defines (those are notably UML Models) but focusing on decision taking.

To realize such a development, we needed to build up a game that was closely related to reality; ideally aligned with real projects that had been performed in the past. Fortunately finding experienced project managers and real life software development projects was not a difficult issue since part of the work done at the *Information Systems* (*ISYS*) research lab at UCL (http://www.isys.ucl.ac.be) developing the game is focused onto modeling/developing software within real life projects conducted with industrial partners.

The first step has thus been to select one particular project that had been successfully realized in the past few years. From that project, knowledge could be extracted (see [10]) and simulated within the game allowing students to face a real life context. Among several projects, the development of a production management system for a coking plant in the steel industry owned by the Carsid company (Duferco Group—http://www.duferco.be) was selected because of:

- a high alignment between the project and the UP, i.e.,
  - the project narrowly followed the guidelines from the RUP knowledge base;
  - an extensive use of UML models and object-oriented technology;
  - CASE-tools for UML, like Rational Rose, Rational Software Architect and Rational Requisite Pro were used.
- an ex-post evaluation of the project had already been realized allowing to deliver adequate information. This evaluation is notably presented in [11, 12];
- the project had been realized by people either from the research department, master thesis students or people from Carsid; every person that had been involved was also still available for feedback;
- a complete documentation of industrial processes was available;
- the high organizational complexity of the industrial context where the software was deployed made it a particularly challenging case, perfectly suitable for our purpose.

2.2 The Carsid project

Carsid, a steel production company located in the Walloon Region, wanted to develop a production management software system for a coking plant. The aim was to provide users, engineers and workers with tools for information management, process automation, resource and production planning, decision making, etc. Coking is the process of heating coal into closed ovens to transform it into coke and remove volatile matter from it. Metallurgical Coke is used as a fuel and reducing agent in the production of iron, steel, ferro-alloys, elemental phosphorus, calcium carbide and numerous other production processes. It is also used to produce carbon electrodes and to agglomerate sinter and iron ore pellets. The production of coke is one of the steps of steel making but further details about other phases of the production process are not necessary to understand the case study.

2.3 Development approach

Once the case study had been selected, game scenarios were set-up. Two sources of information were used to design them:

- Interviews of every person involved onto the project providing a background for decisional aspects within the game. More precisely:
  - 2 project managers—one was a senior researcher from the ISYS department, the other an IT manager from the Carsid company;

  - 10 analysts—4 former master thesis students, 4 researchers for the ISYS department and 2 employees from the Carsid company;
  - 10 developers—1 former master thesis student and 9 employees from the Carsid company.
- Each of the artifacts produced within the project were rigorously documented notably using Rational RequisitePro.

This constituted the empirical base for the development; people having worked on the project and especially project managers were constantly involved into the development that followed itself the UP methodology.

3. The game

This section describes the core model developed to build the game, the dimensions that the gamer has to deal with and the main game scenario.

3.1 Core model

We will bring below the game to further formalization. For this purpose, a game meta-model specified as a UML-like class diagram is represented in Fig. 1 and formal definitions are used. It has been built up independently of the case study. The game we are describing is a multi-user game where the gamer plays essentially the role of a project manager. In a sentence, his core business is driving the software development process by managing the iterative development life cycle.

Definition 1 *A tuple* $(\{(act_i, q_{mfi}^{a}), \ldots, (act_{i+m}, q_{mfi+m}^{a})\}, Ga^{a})$ *is called a gamer a, where* act_i *is a project management activity. An activity is an issue the gamer has to deal with when playing the game i.e. managing the project, for example evaluating a threat, hiring human resources, etc. The gamer owns the ability to manage those activities at cost* q_{mfi}^{a}.

Definition 2 $<a, uc_n, it_m>$ *associating a use case* uc_n *to an iteration* it_m *by the gamer a, which is a Project Manager* a_i^{PM}.

The primary user function is to plan use cases achievement through an iteration plan. To achieve such a goal, the gamer has to perform a number of project management activities. Those activities include the evaluation of factors such as risk, quality and effort. Risk management is materialized into the game through the presence of threats. In function of the use case complexity, enough human resources must be allocated for the use case realization. An expected quality level in terms of correctness, reliability, efficiency, etc should be allocated to each use case so that resources allocation sometimes has to be increased to fulfil the expected levels.

Definition 3 *An activity* act_i *is* $<act_i^{pre}, \tau_i, act_i^{post}>$ *where* act_i^{pre} *describes the activity precondition,* τ_i *is a specification of how the gamer executes the activity and* act_i^{post} *describes the post-conditions i.e. the state of the gamer's environment after the activity is executed.*

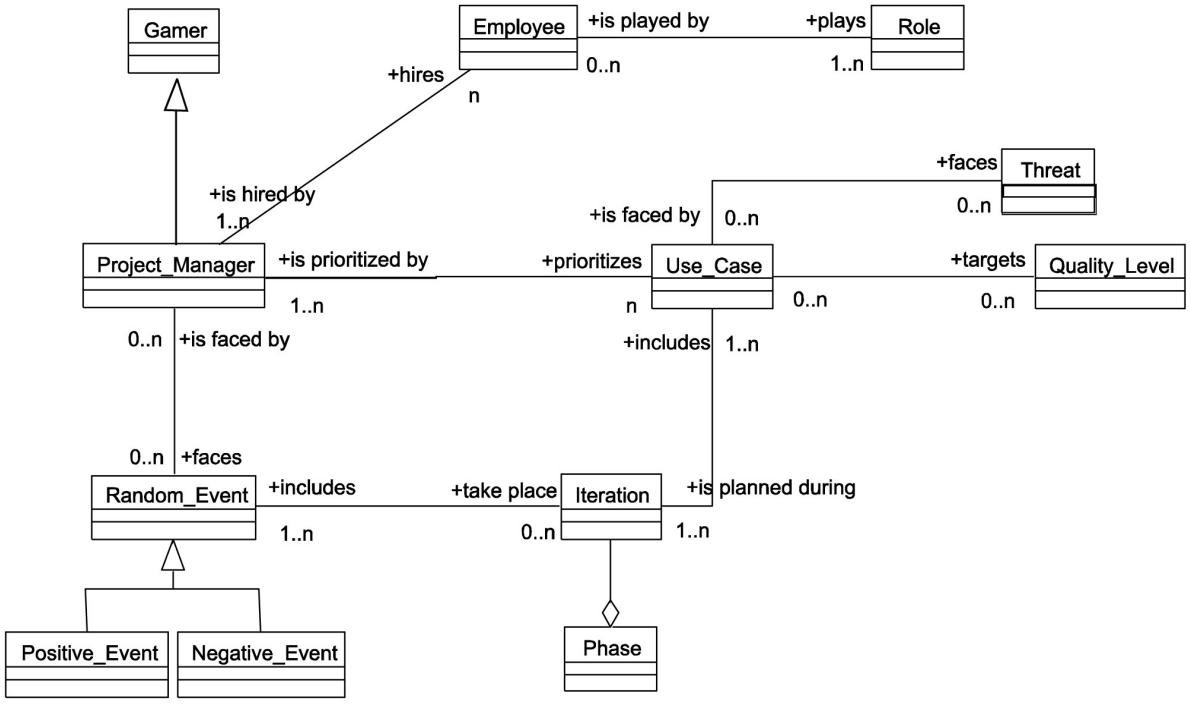

Fig. 1. The Project Management Game Meta-Model.

Definition 4 *$<uc_i^t, uc_i^q, uc_i^w, ucState_j>$ is a use-case uc_j, where uc_j^t represents the threats faced by the use-case, uc_j^q represents the quality levels required for the use-case and uc_j^w represents the use-case weight i.e., the number of resources required for use-case fulfilment. Finally, $ucState_j : uc_j \rightarrow \{act_i^{pre}\}\ \{act_i^{post}\}$ represents the activities pre-condition i.e. $\{act_i^{pre}\}$ and post-conditions i.e. $\{act_i^{post}\}$ for the use case fulfilment.*

Finally, the gamer score is computed on the basis of factors such as the time required to fulfil the project, the total cost of the project and the quality level of the 'realized' use-cases.

Definition 5 *$<score_i^t, score_i^c, score_i^q, scoreAmount_j>$ is a the gamer score $score_j$, where $score_j^t$ represents the time required by the gamer to complete the whole project, $score_j^c$ represents the total cost of the project fulfilment by the gamer and $score_j^q$ represents the use-cases effective quality i.e. the quality effectively measured in function of defined benchmarks, the ability to meet user requirements is included. Finally, $scoreAmount_j$ is a function returning the overall gamer score based on parameters $score_i^t$, $score_i^c$, $score_i^q$.*

3.2 Gamer dimensions

Figure 2 gives an overview of the dimensions offered by the software to the gamer using the formalism defined by the *Software Process Engineering Metamodel* (*SPEM*) [13]. Tasks belong to the following disciplines:

- *Business Modeling* and *Requirements Engineering*: the gamer receives a series of use cases specific to the Carsid case study. On the basis of a redundancy analysis he selects the use cases he has to effectively realize i.e. he has to plan for practical achievement in terms of design, implementation and test. This first step is already from

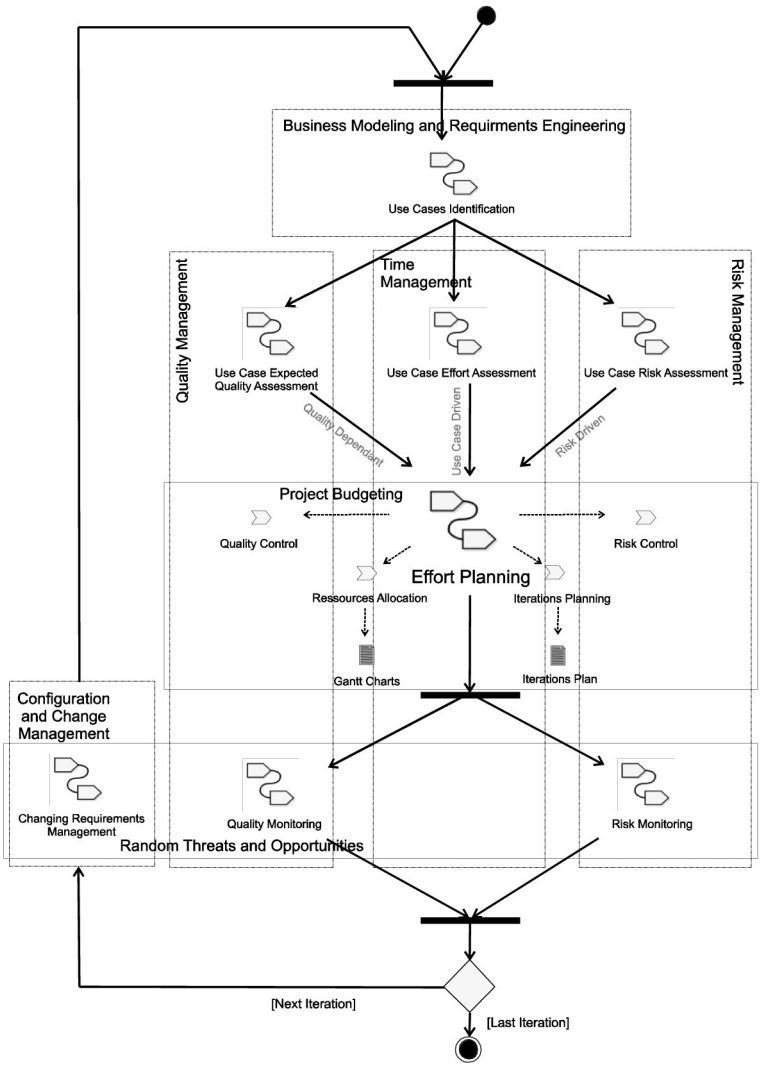

Fig. 2. Gamer dimensions.

strategic importance since cost and resources optimization cannot be envisaged with redundant developments;
- *Risk Management*: on the basis of some predefined threats, the overall risk of each use case must be evaluated by the gamer. The overall risk will be taken into account for use case realization (especially in terms of planning and resources allocation);
- *Quality Management*: quality expectations defined for various factors are associated with specific use cases. As a consequence, resources allocation has to be sufficient to meet the expected level;
- *Time Management*:
  - *Effort estimation*: on the basis of UML activity and or sequence diagrams describing in more details the use cases success scenarios, the gamer has to evaluate the effort (the number of resources having to be allocated) required to realize them in terms of design, implementation and test;
  - *Project planning and budgeting*: using the quality, effort and risk factors he has previously evaluated, the gamer has to plan a first overall project planning and expected cost. Once determined the gamer runs a one iteration simulation.

Furthermore, the game includes some non-deterministic dimensions. Those include:

- *Random threats and opportunities*: during the simulation, positive and negative events can occur. After each iteration simulation the gamer is informed with the factors that happened so that he can react in terms of planning, budgeting for the following iteration(s);
- *Configuration and Change Management*: random events can also subsequently modify use cases specification during the game. When those events occur, the gamer has to take their impact into account for the next iteration(s) planning.

Gamer performance is evaluated based on the:

- Ability to **respect planning**: if too few human resources (HR) are hired on the project, software developments will last longer than originally planned;
- Ability to **respect budget**: the gamer has a budget constraint. Hiring and even firing staff have different costs with respect to HR roles; the gamer has to optimize resources allocation;
- Ability to meet **users requirements and expected quality levels**: this benchmarks lowers if:
  - the project manager omits some non redundant use case realizations;
  - the project manager does not take into account evolutions and changes in requirements and threats;
  - quality levels are below expectations.

3.3 Main game scenario

We briefly describe, in this section, the e-SPM application itself by focusing on some of the decisions the gamer has to take. The implementation has been realized using the Java 2 EE platform [14].

Figure 3 depicts the project effort estimation module. The game supports effort estimation on the basis of *function points* and *use case points* techniques. Based on available documentation, the player estimates the number of steps in a use case success scenario or the data movements per function so that he can evaluate its weight. In the same way, the user also estimates the actors' weight. On the basis of the *global computed weight* balanced using environmental and technical complexity factors

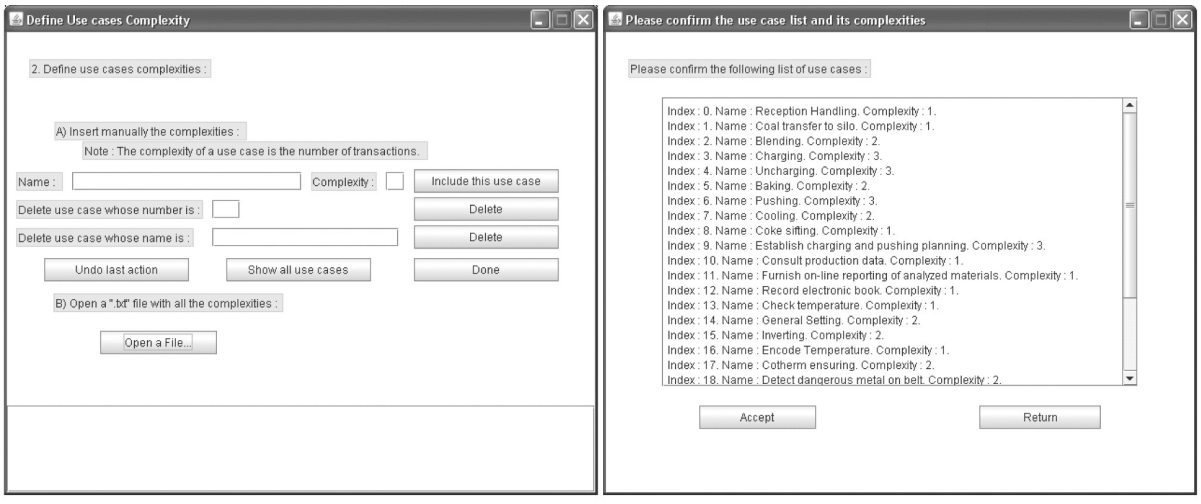

Fig. 3. Project effort estimation using use-case points.

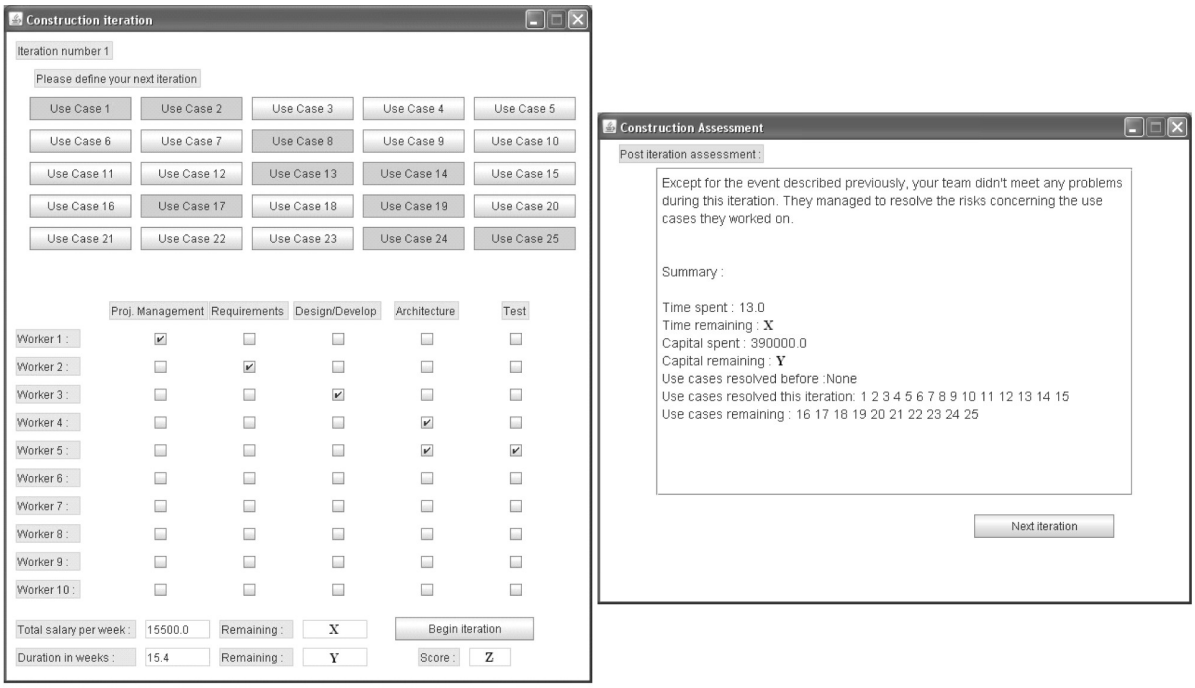

Fig. 4. Project iteration planning through use case realization.

also available as variables of the game, the overall project effort is determined.

Planning use case realizations for the coming iterations, evaluating their redundancy, allocating workers on specific disciplines are, among others, some of the main tasks of the gamer. Fig. 4 depicts, on the left, the main interface allowing to encode the data previously described. Once encoded, the software runs for one iteration simulation and provides the user with an iteration report, represented in the right part of Fig. 4.

Risk and quality management are another part of the gamer's mission. At the beginning of the game as well as after each iteration the gamer is given a *risk and quality matrix* i.e., a matrix tracing the possible exposure of the identified threats and quality factors on the use cases. The gamer has to take this into account when planning the coming iteration(s). Fig. 5 shows, on the left a window informing the gamer that new threats have been identified. e-SPM then automatically opens the risk and quality matrix in a spreadsheet (such as Excel or LibreOffice Calc) as shown in the right part of Fig. 5.

Finally, as pointed out earlier, random events can occur during each of the project iterations. The gamer is informed of each positive or negative event interfering when planning the next iteration(s). Fig. 6 depicts, on the left, one of the screens informing the gamer of an event occurrence. The game also includes other UML diagrams on which

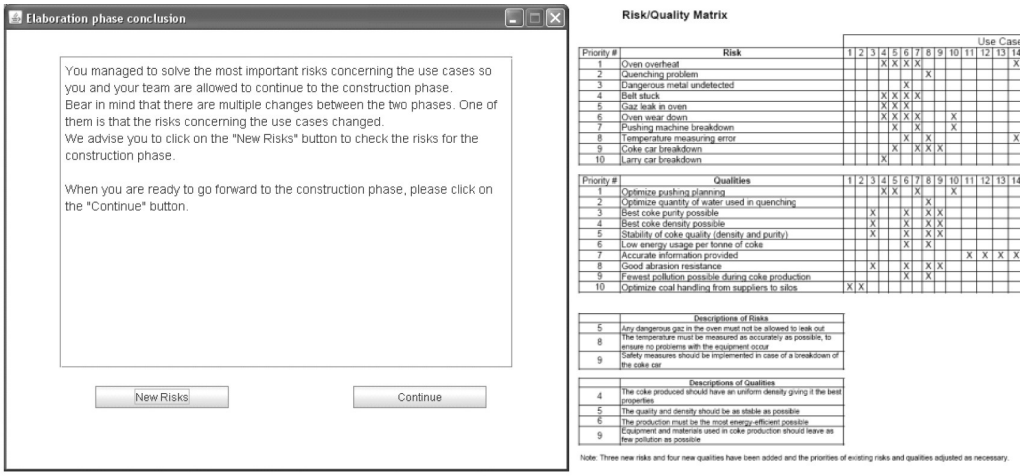

Fig. 5. Project risk management.

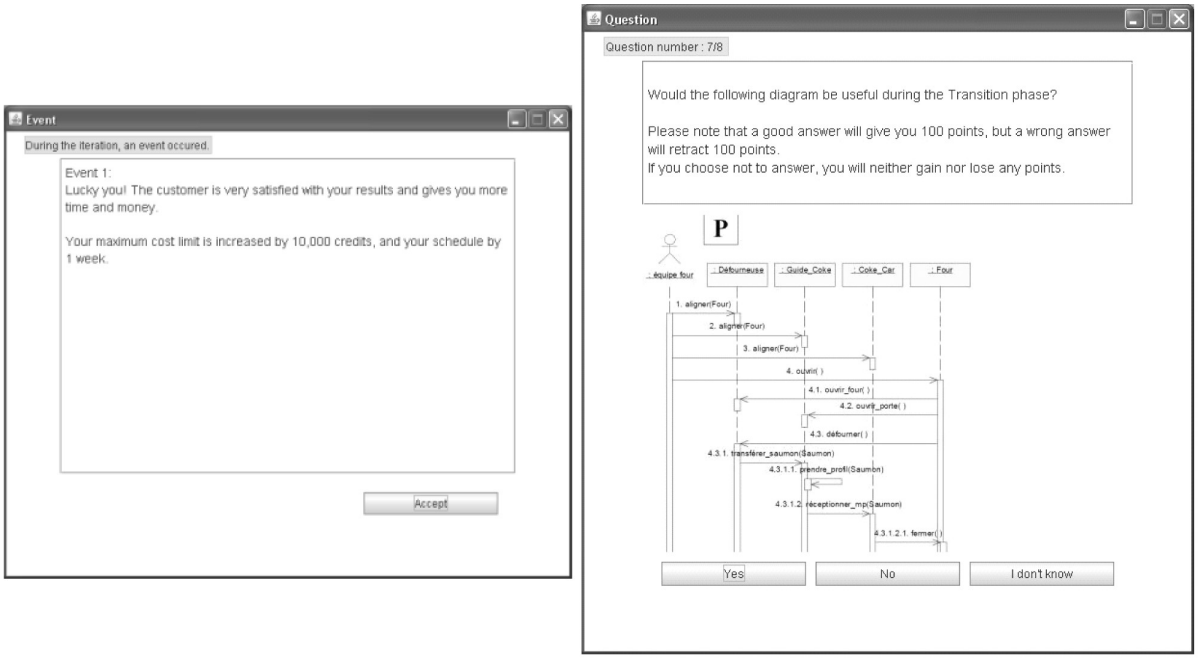

Fig. 6. Project iteration planning through use case realization.

questions are asked to test the theoretical knowledge of the player on the software development process and concepts. Adequate answers increase the gamers' score.

4. Validation

e-SPM has been validated in the context of the LSM competencies framework rather than on the basis of a specific e-learning framework as the one proposed in [15]. In order to clarify the intended learning outcomes (ILOs) of the masters in business engineering and in computer sciences (information systems), the Louvain School of Management (LSM) has been engaged since 2009 in the elucidation and definition of a 'competencies framework' (see [16]).

The formulation of the school ILOs through a competencies framework is aimed to be the cornerstone around which will be articulated the programs monitoring both *a priori*, by the design and continuous improvement of the learning curriculum and activities, and *a posteriori*, by structuring the collect and analysis of courses and programs' quality evaluation. The development of a common competencies framework will allow to go further in the harmonization of the learning goals, pedagogical approaches and quality monitoring tools and will ensure a high and similar quality across the programs.

More specifically, stakeholders have been involved in the identification of the competencies expected from LSM graduates: workshops have been organized, by specific disciplines but also transversally with representatives of employers coming from various industries and management functions, with young alumni and students as well as with faculty (professors and teaching assistants). These workshops have been dedicated to the elucidation of the learning outcomes aimed by the faculty, to the confrontation with the expectations of alumni and employers and to a first exchange of teaching practices used or usable to reach these learning outcomes.

Typically, the framework is organized as a compass as represented in Fig. 7 and developing nine competencies.

These competencies are:

1. Corporate citizenship: Act consciously, aware of their responsibilities, placing human and ethical considerations at the very heart of their thought and actions;
2. Knowledge and understanding: Master an active and integrated command of a multidisciplinary body of knowledge (content, methods, models, conceptual frameworks) essential to respond professionally in the various managerial domains;
3. Scientific and systematic approach: Analyze and resolve problems in multidisciplinary and complex management situations using a scientific and systematic approach;
4. Innovation and Entrepreneurship: Innovate, initiate and lead change;
5. Working effectively in an international and multicultural environment: Work as an inter-

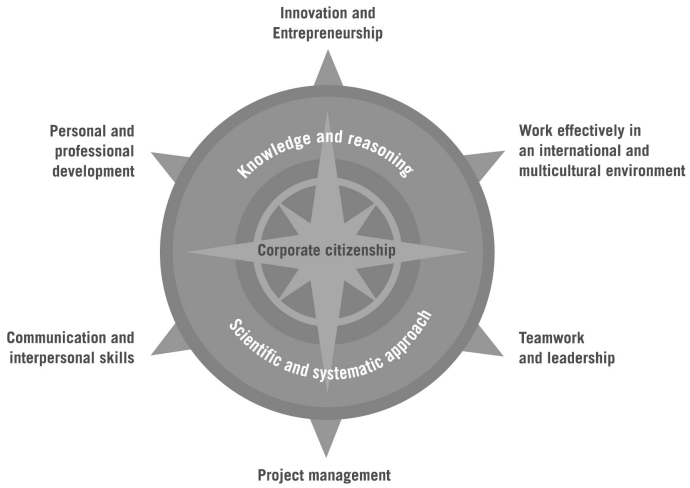

Fig. 7. the LSM competencies framework.

face between stakeholders with different rationales, belief systems, training, nationalities, cultures, etc;

6. Teamwork and leadership: Integrate and work in a team; exercise sensible leadership within the group;
7. Project management: Define and manage a project to completion, taking into account the objectives, resources and constraints that characterize the project environment;
8. Communication and interpersonal skills: Converse and communicate effectively and convincingly with stakeholders;
9. Personal and professional development: Display self-knowledge and independence, able to adapt quickly to new situations and develop positively.

e-SPM has been screened in specific and transversal workshops involving students, faculty, alumni and potential employers in the IT business using the framework. Table 1 summarizes the evaluation with

Table 1. Evaluation of e-SPM with the LSM competencies framework

	Application and project							Argumentation			
Competencies	Real Problem Case Study	Monodisciplinary Integrated Project	Multidisciplinary Integrated Project	Problem-Based Learning	Directed Team Labs	Computer Simulated Labs	Progression Monitoring	Argumentation and Discussion	Role Game	Professional Simulations	Micro-Teaching
1. Corporate citizenship		x									
2. Knowledge and understanding		x	x		x	x	x				
3. Scientific and systematic approach		x				x				x	
4. Innovation and Entrepreneurship		x		x		x			x		
5. Work effectively in an intl. and multicultural environment		x				x			x		
6. Teamwork and leadership		x			x	x		x	x		x
7. Project management		x			x	x	x		x	x	
8. Communication and interpersonal skills		x					x	x			
9. Personal and professional development		x			x	x	x		x		

respect to the nine competencies and the contributions it brings to students and teachers.

5. Conclusions

Developing software can involve nowadays hundreds of individuals for the most sophisticated software projects. It is not consequently anymore just a matter of analysts, designers and developers but requires people able to manage the whole software process adequately. Required skills (even hard ones) cannot be only acquired through an ex cathedra course but requires empirical experience. With that in mind, we have specified, formalized and developed a role game called e-SPM allowing students to learn the principal dimensions and concepts of software project management.

This paper has thus presented the motivation, conceptualization and results of a research aimed to build a game for iterative software development training. The game is aligned with the UP/UML process and based on the documentation and experience of a real life development project in the steel industry. Such a game helps educating the future project managers who will have to cope with a complex and constantly evolving environment. Benefits of adequate project management will be, among others, measured in terms of spent effort, time to market, perceived quality and even ability to meet user requirements.

The game in its actual state remains however subject to improvements. On the basis of a five year user-experience and an enhanced game engine, a new release of the software will be built up soon; it will notably introduce more uncertainty to better address the soft skills project managers should acquire, dispose of a more efficient user interface and allow competition between participants.

References

1. A. Sukhoo, A. Barnard, M. M. Eloff, J. A. V. der Poll and M. Motah, Accommodating soft skills in software project management, *Issues in Informing Science and Information Technology*, **2**, 2005.
2. J. Rose, K. Pedersen, J. H. Hosbond, and P. Kraemmergaard, Management competences, not tools and techniques: A grounded examination of software project management at WM-Data, *Information & Software Technology*, **49**(6), 2007, pp. 605–624.
3. S. C. Petter and V. K. Vaishnavi, *Facilitating* experience reuse among software project managers, *Inf. Sci.*, **178**(7), 2008, pp. 1783–1802.
4. C.-N. Bodea, M. Dascalu and M. Coman, Quality of project management education and training programmes, *IJKSR*, **1**(2), 2010, pp. 13–25.
5. K. U. Koskinen and P. Pihlanto, The project manager in the theatre of consciousness: A new approach to knowledge creation and communication, *IJKSR*, **1**(4), 2010, pp. 20–31.
6. OMG, Unified Modeling Language (UML). Version 2.4. Technical report, Object Management Group, 2011.
7. P. Kruchten, *The Rational Unified Process: An Introduction*, Addison-Wesley, 2003.
8. P. Kruchten, Tutorial: introduction to the rational unified process, In *ICSE*, p. 703. ACM, 2002.
9. E. Alba and J. F. Chicano, Software project management with gas, *Inf. Sci.*, **177**(11), 2380–2401, 2007.
10. M. Minovic, V. Stavljanin, M. Milovanovic and D. Starcevic, Adventure game learning platform, *IJKSR*, **1**(1), pp. 12–21, 2010.
11. Y. Wautelet, *A Goal-Driven Project Management Framework for Multi-Agent Software Development: The Case of I-Tropos*, PhD thesis, Université catholique de Louvain, 2008.
12. Y. Wautelet and M. Kolp, Goal driven iterative software project management, In M. J. E. Cuaresma, B. Shishkov, and J. Cordeiro, editors, *ICSOFT* (2), pp. 44–53. SciTePress, 2011.
13. OMG, Software & systems process engineering meta-model specification, Version 2.0. Technical report, Object Management Group, 2008.
14. K. Ahmed and C. Umrysh, *Developing Enterprise Java Applications with J2EE(TM) and UML*, Addison-Wesley, 2001.
15. B. Ertl, K. Ebner and K. Kikis-Papadakis. Evaluation of e-learning. *IJKSR*, **1**(3), 2010, pp. 33–43.
16. Louvain School of Management, LSM competency framework, http://lsmcompetencyframework.be, May 2012.

Yves Wautelet is an Assistant Professor of Information Systems at Hogeschool-Universiteit Brussel, Belgium. He formerly has been an IT project manager and a postdoc fellow at the Université catholique de Louvain, Belgium where he still is scientific collaborator. He completed a PhD thesis focusing on project and risk management issues in large enterprise software design. Dr. Wautelet also holds a Master of Management Sciences as well as a Master of Information Systems. His research interests include aspects of software engineering and enterprise information systems such as requirements engineering, agent-oriented programming and COTS-based development. He also focuses on the application of his research into industrial environments for the development of supply chain management and enterprise resource planning solutions. Yves Wautelet is also a reviewer for international scientific journals and member of the program committee of several international computer science conferences. Finally, he is the author or co-author of about 40 articles and papers in international journals, conferences, books and workshops.

Manuel Kolp is Full Professor in IT (Information Systems) and Vice-Dean at the Louvain School of Management (LSM), Université catholique de Louvain (UCL). He was a Post Doctoral Research Associate in Computer Science at the University of Toronto, Canada, for a couple of years and worked there with the Department of Computer Science in Requirements Management and Multi-Agent Software Engineering. He was also an adjunct professor in Knowledge and Information Management at the Faculty of Information in the same university. He has also been invited professor with the St. Louis University in Brussels and with the University of Brussels for several years. Previously, he was a FNRS (Belgian National Agency for Scientific Research) research fellow in Knowledge Representation and Object Oriented Information Systems at the Université catholique de Louvain and received his PhD degree from the University of Brussels. Manuel

Kolp has also been collaborating as a lead investigator on projects dealing with knowledge, information and data systems or e-business and ERP II applications and regularly acts as an IT expert and advisor for enterprises and organizations. He is also a reviewer for international scientific journals and expert for foreign national and international research agencies and the European Commission. Dr. Kolp is mainly involved in academic research at UCL on projects dealing with agent oriented e-business systems architectures and software project management. His research work also studies e-business and ERP II systems methodologies. In addition to be program committee member or associate editor of international conferences and workshops in information systems, software engineering and multi-agent systems, Manuel Kolp has been one of the organizers of six of them. Manuel Kolp is the author or co-author of about 120 articles and papers in international journals, conferences, books and workshops and has edited or written 4 books.