CENTERIS 2013 - Conference on ENTERprise Information Systems / PRojMAN 2013 - International Conference on Project MANagement / HCIST 2013 - International Conference on Health and Social Care Information Systems and Technologies

Construction of a Web-based Project Management Simulator: Proposal, Process and Features

João Nascimento [a, *], Paulo Resende da Silva [b], João Samartinho [c]

[a] *Departamento de Informática e Métodos Quantitativos, Escola Superior de Gestão e Tecnologia do Instituto Politécnico de Santarém, Portugal*
[b] *Departamento de Gestão, Escola de Ciências Sociais, Universidade de Évora, Portugal*
[c] *Departamento de Informática e Métodos Quantitativos, Escola Superior de Gestão e Tecnologia do Instituto Politécnico de Santarém, Portugal*

Abstract

This paper proposes a model of a simulation system of Project Management based on the Web environment (Project Management Virtual Environment - PMVE), its features, and the process of its construction. This system aims to support the process of teaching / learning the subject of project management. When compared with other work in this field, there are two aspects that standout: the use of the Web to enhance the interaction between the agents involved in the teaching / learning process, and the variety of skills considered in the system. More than focusing on a small set of processes dealing with technical work management, the solution presented here involves nearly all of the project management life cycle. The description of this solution starts by presenting the context of the problem being solved, highlighting some of the features of the project management activity. Next, the conceptual perspective behind the construction of the system is presented, followed by the technological solution and respective features.

Keywords: Software Engineering, Project Management, Computer Based Trainning

* Corresponding author. Tel.: +351964166892
E-mail address: joao.nascimento@esg.ipsantarem.pt

1.   Introduction

The *Project Management Virtual Environment* - PMVE is a technological solution built for the *Web* environment, which is created in order to assist those who are involved in the teaching / learning courses related to Project Management, especially in undergraduate and graduate levels of education, but its application is not limited to academia. This solution is designed to support school staff, and training professionals in general, when the objective is developing students' skills in Project Management, and facilitating the students' knowledge assimilation, through a controlled scenario on the activity of managing projects. As regards Wateridge [1]:

> "*Undoubtedly the main method for managers to learn aspects of project management in the past is through experiential learning*".

This means that, in the past, there was no formal teaching on Project Management provided; the Project Manager has to deal with real projects to learn this art. Sometimes, dealing with failure was not an easy task.

Based on the results of the studies from Kerzner [2] and Burgetz [3], a high percentage of competencies in project management is better seized through experience. But, according to Thamhain [4], teaching should not be made only based on experience. To maximize the effect of training, it is necessary to join the expository sessions of theoretical aspects with a trial oriented instruction.

The solution presented here was built with the intention of complementing expository sessions. It was not meant to be used alone, or even to replace the teacher. It complements the teacher's job, facilitating the transmission of skills in project management. The teacher is responsible for the setting up of the system, and the experience of the students will be strongly influenced by the work of the teacher.

The system development process took place in two complementary phases: one concerning the analysis and design, where the steps (or a subset of steps) were performed by the application of the RIPPLE methodology, according to O'Docherty [5]. This methodology is based on the unified modeling language (UML), created by Booch, Jacobson and Rumbaugh [6]. In this methodology (RIPPLE) the author adds a sequence (which is not present in the original specification of the UML) of steps, and a set of new artifacts to maximize the outcome of problem analysis and conception of the system being built. In the second stage of the development process, the system was constructed according to the solution designed on the first stage. This was done by choosing and applying of the appropriate tools.

This paper is structured in four sections: the first provides a brief introduction to the work, the problem being addressed and the objective of the work; the second describe in more detail, the context of the problem being tackled and how the solution is reached; section 3 presents briefly the way the solution is structured, in terms of conceptual, logical and physical levels; the last section presents the main conclusions and the work to be achieved in the future.

2.   Background and Context of the Problem

Project Management teaching is an activity that is more recent than the profession itself. References to the term "accidental profession", used by Levine [7] and Herkeens [8], refer to situations of professionals with extensive experience in technical work that, suddenly, are placed, without any specific training, in the project management function. Competent technicians who assume leadership positions are not always successful, with the same ease as happened in their technical careers.

Project management is no longer an activity limited to the application of techniques and tools in order to, for example, define the scope of the project, estimate required resources (in its broadest sense) or create schedules on technical and management work. Project management has become a complex and multidisciplinary activity, according to the PMBoK 2013 [9].

The more complex is the context in which projects are developed, the more complex the activity of

managing these projects gets. Presently, due to the high degree of competition, the adversity induced by the context of crisis, in which the majority of organizations coexist, the successful realization of projects requires a level of accuracy, which is only possible through an ascertained management job. Implement solutions to solve increasingly difficult problems, or seize opportunities increasingly hard to find, is only possible by the application of a set of best practices on project preparation and leadership.

Simulation becomes an approach to consider, when thinking about giving the students (or even effective project managers) the experimental perspective of the job. Through simulation, it is possible to experiment the activity of managing a project. This approach does not require as much time as it would be necessary on a real situation, nor does it assume the consequences that would arise from any failures that might occur. The simulation allows the execution time acceleration by setting aside certain kinds of tasks that are not normally performed by the manager (but which take time, and other resources, to be done). Consider, for example, the technical work, which is given to technical teams and constitute the largest consumer of project time (and other resources). Such achievement leave more time for what really matters – making management decisions. However, we must keep the connection with reality by trying to confront the student with real situations.

A large majority of project management simulators developed in the academic environment have their focus on the engineering or technical work, concerning a single task, or a methodology, or a specific application area – software engineering, for example. This is the case of the "Master Manager" presented by Claudia Werner [10], the "Prosim" [11], the "SIMSE" [12], or the "SESAM" [13].

The focus on a single project management aspect, or a small set of aspects [14], is also a limitation of all simulators found. Even considering some commercial tools, for example: the "Planview", the "Primavera" and "NIKU 6" [15] the scenario is the same.

3. The Project Management Virtual Environment

The following subsections will be devoted to the description of the resulting system, by presenting it in a conceptual and technological perspective, and by highlighting some of its features through the presentation of graphical user interfaces.

This platform may be classified as a computer based training system (CBT), based on its characteristics and nature, whose structure includes a mechanism of decision assessment by comparison.

The methodology used to develop the system was selected and adapted from some references found in literature. One of the premises initially established for the selection of the development methodology was that it should be consistent with object orientated approach. Based on this assumption, some of the methodologies were identified (RUP reported in Kroll [16] and GRAPLE, presented by Schummer [17]) and (the elected) RIPPLE [5]. All these methodologies identified are based on the UML, but RIPPLE was adopted.

The choice, and the customization, of the methodology were done so that it could be applied to this work, where the main constraint was the time vertex. The artifacts that were considered expendable were excluded from the set of work to do, and those artifacts considered relevant were executed in a way that a functional platform could be built.

The artifacts produced were organized in five groups: **Genesis** (informal requirements); **Requirements** (Actor List, Use Case List, Use Case Diagram, Use Case Description, Activity Diagram and User Interface Design); **Conception** (Class Diagram, Sequence Diagram and Database Schema); **Construction** (Source Code); and **Deployment** (Components and setup manual).

In technological terms, the choice of tools was made considering the availability and the skills of the authors of this work. So, Microsoft Visio 2010 was the tool used to produce all the UML Diagrams involved. The construction of the system was made using Microsoft Visual Studio (initially 2008 edition and then) 2010 edition. The database was developed using Microsoft SQL Server 2008 Express Edition.

3.1. *Conceptual Viewpoint*

This is a system to support the teaching / learning, therefore there are at least two kinds of users: Teacher and Student. However, to ensure a better match to reality, it was decided to use another abstraction. Conceptually, there is no specific role for the teacher or for the student, as the perspective here is managing projects. Both the teacher and the student can perform the same functions – Project Manager. There is only one difference between the two perspectives behind Project Manager: the student manages the project to show competences to the teacher, creating a project structure, and teacher does exactly the same; *i.e.* manages the project, but to show to the student how should / could he have done.

There is, however, a user with special privileges – Administrator. This user will naturally be a supervisor, or someone responsible for the maintaining operation of the system.

The features of the system are organized into three modules: Project Portfolio Management (PPM); Framework Management (FM), and Project Management (PM).

As shown in *Figure 1*, and described in the following text, both the teacher and the student can use functionalities related to the FM and PM modules. The Administrator is the only user who can carry out the administration (maintenance) of the project's portfolio in the system, through the module PPM. There are some specific functionalities of the Administrator, as can be seen later in this paper.
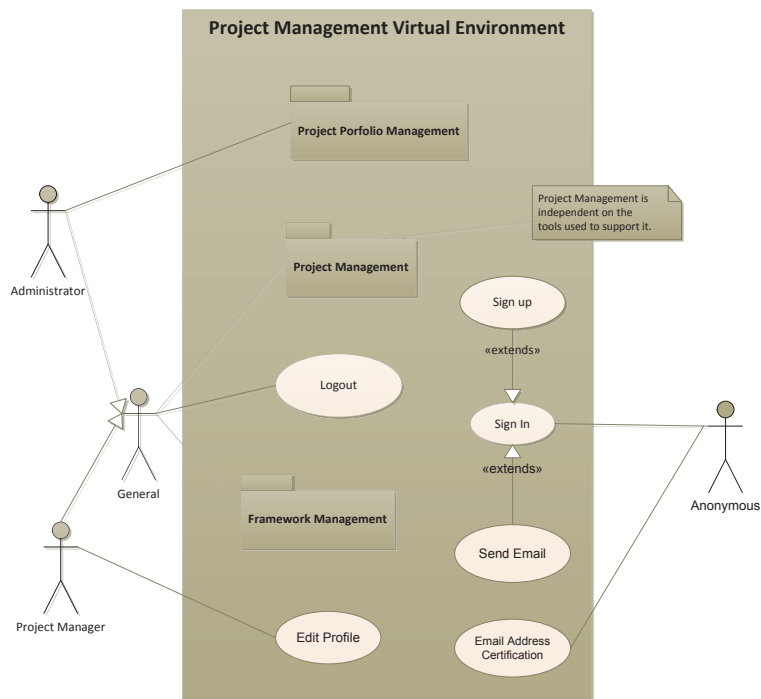

Figure 1 – Top level *Use Case Diagram*

- The PPM module has a set of functionality related to the management of the exercises of projects to deliver to students. This module has the following features:
  - i)   Creation of new projects (by duplication based on another, or from scratch);
  - ii) Management of the project's phases;
  - iii) Definition of the project's objectives;
  - iv) Linking users (and it's roles) to the project;

v) Configuration of constraints;

vi) Set the mandatory knowledge areas of the project;

vii) Management of the messages (between users) at the forum.

All of these functions (that must be addressed in the configuration of a new problem/project to be solved by the student) are available only for the Administrator, as shown in Figure 2. However, as can be perceived from the following paragraphs, some of them can also be made by other users in a similar way (*e.g.* the student can define which knowledge areas should be addressed in a project as well), when they are not imposed by the Administrator.
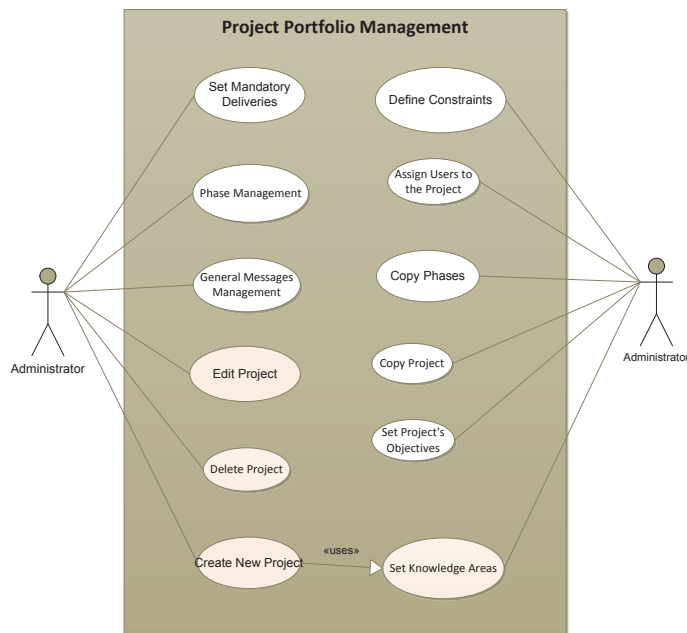

Figure 2 – PPM *Use Case Diagram*

*   The FM module contains a set of general functions (which can be used by different kinds of users, at different circumstances of the project), and a set of functions specific to the Administrator. The first group of functionality provides the creation of Project Management Life Cycles (PMLC) concept presented in PMBOK [18], and the related elements. These are available to the student, while in Project Managers role, in order to define the overall architecture of the project management to be used. A PMLC includes aspects such as: Knowledge Areas, Process Groups, Processes, Results, Techniques, and Tools. For all these elements, in addition to its characteristics, it is possible (and desirable) to establish their relationship with the corresponding items in the system's base model. In this module, and also related with PMLCs, the student can define which are the dependences between processes, as producers of results (outputs) and consumers of results (inputs) from other processes. This is the essence of the network that characterizes one PMLC.

    Processes may also be related to techniques and tools, in the way that a process applies a technique and/or a specific tool (or group of tools). Finally, the user can (and must) define the resemblance between each process in its user defined PMLC and the PMLC standing in the base of the system (the one used to make possible the matching across multiple PMLC defined by users).

Figure 3 shows that in this module there are also specific functionalities for the Administrator, which are identical to the Student's functionalities, but the items managed by the Administrator have a structuring function: which pertain to establish foundations for the comparison between the various PMLCs that students can create on the system.
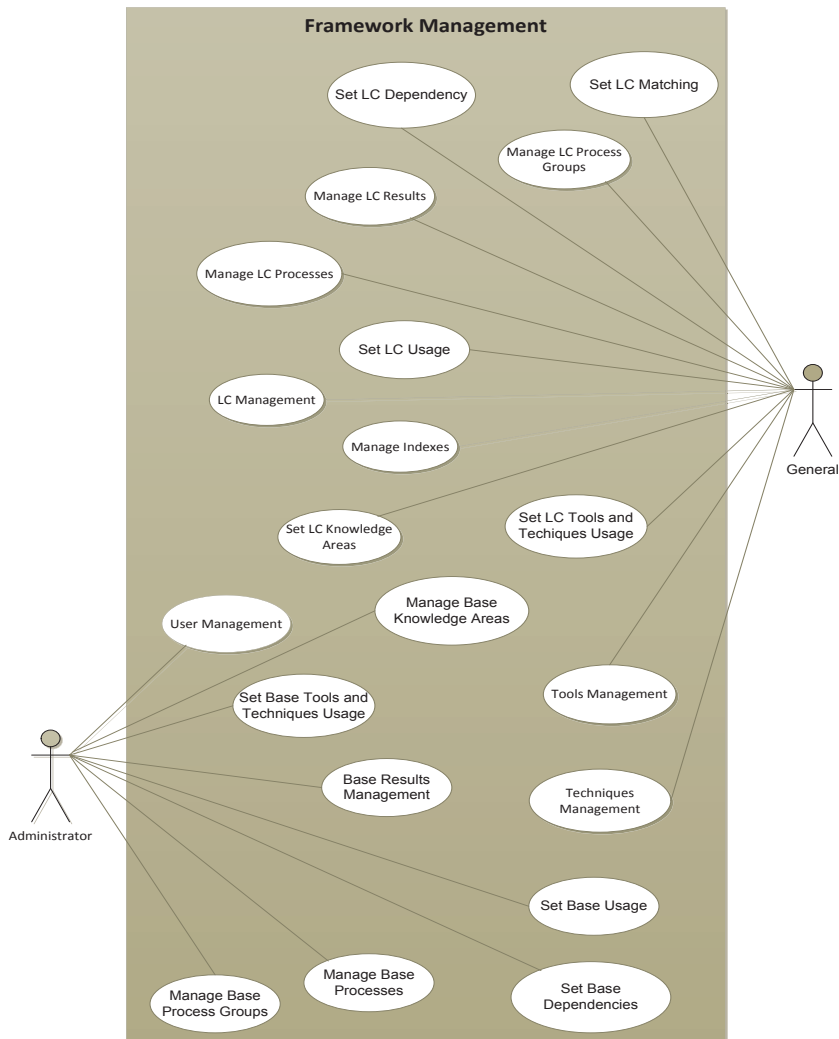

Figure 3 – FM *Use Case Diagram*

- The PM module includes features available to all users, as shown in *Figure 4*. This module gives the users all the functionality that enables them to respond to the requested: preparation, planning, execution, controlling, and monitoring the project.
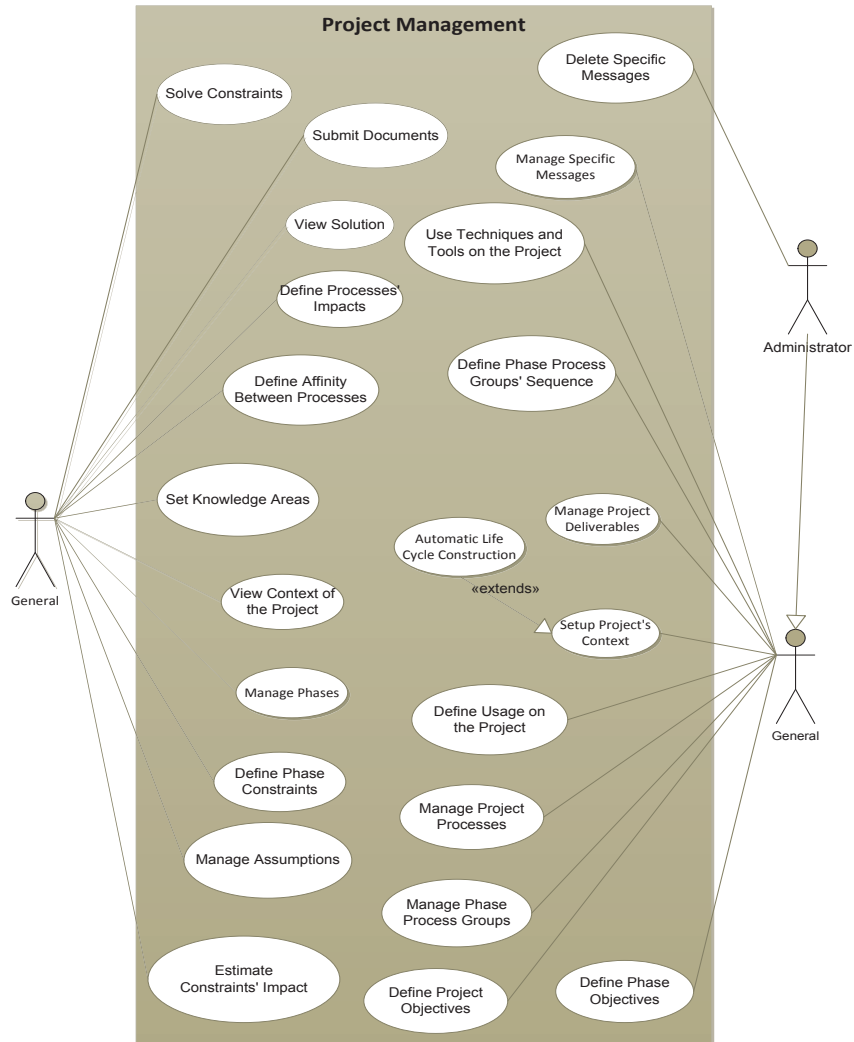
Figure 4 – PM *Use Case Diagram*

In addition to these features, a forum where users can exchange messages is available. There are two levels on this forum: (1) the Administrator can perform all functions in the forum (on the messages left by other users), and (2) the other users are limited to consult messages (they have access to), and put new messages or threads. The messages are organized into topics of discussion. There are two levels of board: one private specifically for projects, whose messages can only be accessed by users associated with this project, and a public forum, where messages can be viewed by all users registered in the system and properly authenticated.

The Class diagram represents, in terms of data, the elements, and its relation, needed so the data is stored. *Figure 5* shows the complexity associated to data storage.
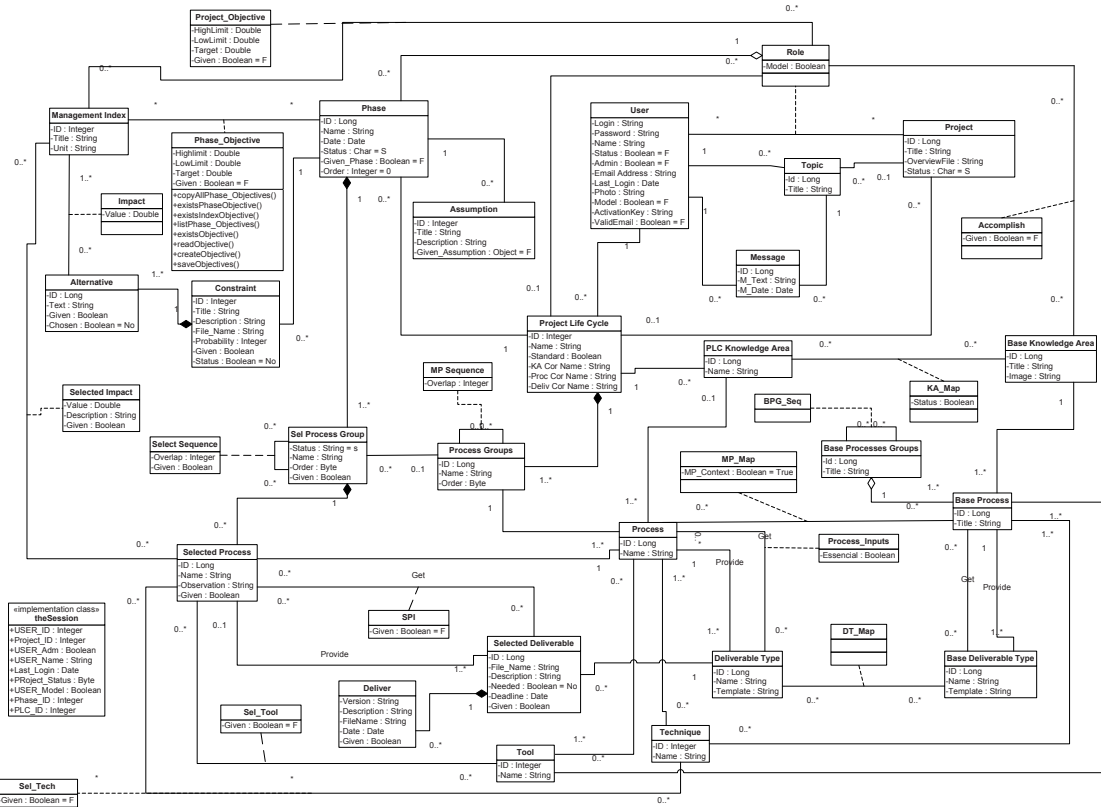
Figure 5: *Class Diagram*

After Classes have been defined, it is now possible to describe functionality (identified at *Use Case Diagrams*) in more detail. This is where the UML *Sequence Diagram* is applied. *Figure 6* shows one sample constructed during the design of the system. It is about the feature that enables the user to alter his profile.
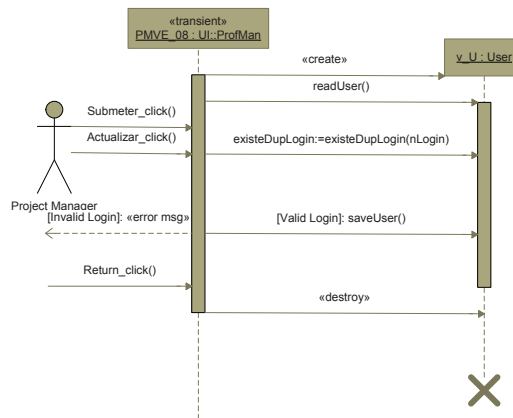

Figure 6: *Sequence Diagram (one sample) - the case "Edit Profile"*

3.2. Technical Viewpoint

The technologies used in the development and operation of the system are all from *Microsoft* suite. All source code was developed using (initially the 2008 version and, as soon as it became available, migrated to) the 2010 version of the *Microsoft Visual Studio*. The supporting data storage system was made using *Microsoft SQL Server 2008 Express edition*.

During the development phase, no framework was used. All the code was developed in *Visual Basic* (*VB.Net*), following the rules of object orientated development. In terms of source code structure, there are three layers, as shown in *Figure 7*. The *data layer* is supported on a database engine; the *business layer* has the implementation of all classes and its attributes and methods (ensuring data processing and data access), and; the *graphical user interface* is implemented with html pages and vb.net code.

Database Server
(MS SQL Server)

Business Rules
and Data Access
(VB.Net)
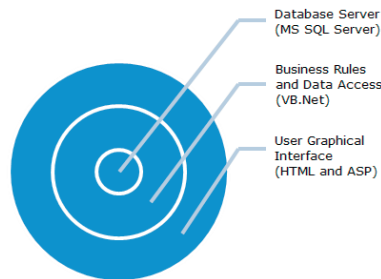
User Graphical
Interface
(HTML and ASP)

Figure 7: Layers of the system.

The operation of the system will be dependent on a *Web server* that supports ASP.NET.

The system's tests were made on a computer with Operating System *Microsoft Windows XP Professional Edition* with *IIS* (*Internet Information Service*), version 6.0, and the database engine used was *Microsoft SQL Server 2008 Express*.

To access the system, the user only needs to have an *Internet browser* installed on his computer. The whole system was designed to be used with *Microsoft Internet Explorer* 7.0 or above. *Figure 8* shows the technologies involved in exploring the system.
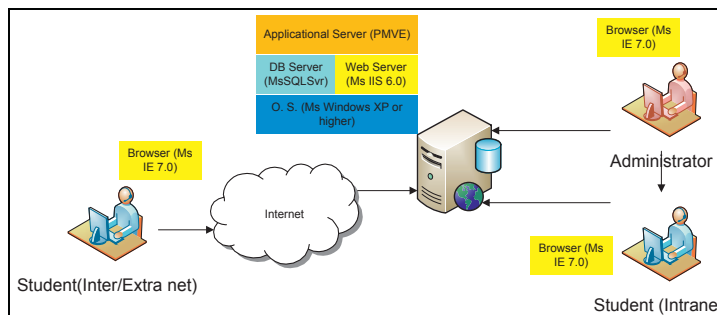

Figure 8: System's Usage Context

4. Conclusions

This paper presented the PMVE system, through a description, in terms of conceptual and technological perspective, highlighting its features.

Regarding the conceptual solution, this was achieved by the application of an agile software development,

resulting in three different modules: PPM module, for project portfolio management; FM module for system framework management; and PM module, for Project Management. In technological terms, in order to put this system to work there are necessary a combination of several essential services, that are needed to enable the teaching / learning activity in Project Management.

The aim was to keep a low set of technologies needed to make the exploration of the system. In fact, it only takes an operating system like *Microsoft Windows*, with *Internet Information Services* version 6.0 or later. Additionally, we will need a database engine such as, for example, *Microsoft SQL Server 2008* (free edition is sufficient to start).

From the user's point of view, the system requires only one connection to the server where the system is hosted (either on the local network or through the Internet) and a Web browser.

This system has limitations. For now it is only available in Portuguese language. One future work to be done is the necessary adjustments to extend the interface to other languages, allowing the user to choose his preferred language.

In terms of performance assessment, there's the need to implement a more user friendly functionality. Instead of comparing between his own project and the project solution (inserted on the system by another project manager – teacher or other professional), the trainee should look for performance indexes.

In the future, it is expected to be able to present to the student a set of performance indexes, indicating, for example, the level of work considered, and not, on the project and the unnecessary management effort considered.

There is also a possibility of interconnection with (commercial, or not) project management tools in order to export the project structure created in the system, or import a structure from a file produced by those tools.

Finally, because all the design was done and thoroughly documented, there is the chance to rebuild the system using any kind of open source technology. This would be an advantage in terms of costs for educational institutions and training centers, if they wish to use the system.

5. References

[1] Wateridge, J., 1997. Training for IS/IT project managers: A way forward. International Journal of Project Management, (pp. Volume 15, Issue 5, Pages 283-288, ISSN 0263-7863, 10.1016/S0263-7863(96)00085-3.).

[2] Kerzner, H., 1987. In Search of excellence in project management. Journal of Systems Management, pp. 30-39.

[3] Burgetz, B. A., 1991. Project manager or water walker. CMA Magazine, pp. 65, 29.

[4] Thamhain, H., 1989. Developing project management skills. Project Management Institute Seminar/Symposium, (pp. 652-659). Atlanta.

[5] O'Docherty, M., 2005. Object-Oriented Analysis and Design: Understanding System Development with UML 2.0. England: John Wiley & Sons Ltd.

[6] Jacobson, I., Booch G., Rumbaugh, J., 1998. The Unified Software Development Process. Addison Wesley Longman.

[7] Levine, A. H., 2002. Pratical Project Management: tricks, tactics and tools. New York: John Wiley & Sons.

[8] Herkeens, G. R., 2002. Project Management. McGraw-Hill.

[9] PMI, 2013. A Guide to Project Management Body of Knowledge, 5th Edition. Newtown Square, Pennsylvania: PMI.

[10] Werner, C. et al., 2013. Obtido em 05 de 02 de 2013, de http://reuse.cos.ufrj.br/site/pt/: http://reuse.cos.ufrj.br/site/pt/

[11] Lakey, P., 2003. A Hybrid Software Process Simulation Model for Project Management. in Proceedings of the 6th Process Simulation

[12] Navarro, E. et al., 2013. Obtido em 15 de 03 de 2013, de http://www.ics.uci.edu/~emilyo/SimSE/news.html

[13] Drappa, A., Ludewig, J., 2000. Simulation in Software Engineering Training. Proceedings of the 22nd International Conference on Software Engineering (pp. 199-208). ACM.

[14] Martin, A., 2000. A simulation engine for custom project management education. International Journal of Project Management, Vol. 18(3), 201-213.

[15] Mendonça, H. et al., 2004. Girassol: A Web-Based Software Project Management Tool. SERP'04: Proceedings Of The International Conference On Software Engineering Research And Practice, (pp. VOLS 1 AND 2 : 361-365).

[16] Kroll, P. e Kruchten P., 2003. The Rational Unified Process Made Easy. Addison Wesley. Modeling Workshop (ProSim 2003). Portland, Oregon, USA.

[17] Schmuller, J., 2004. Teach yourself UML in 24 hours, 3ª edição. SAMS publishing.

[18] PMI, 2008. A Guide to Project Management Body of Knowledge - PMBoK. Newtown Square, Pennsylvania.: PMI.