

Project Management with Agile Processes

Miguel Pinto, n°56888

Instituto Superior Técnico
miguel.pinto@ist.utl.pt

Abstract. The software process development community has known a steady evolution over the past years. Alongside this growth, several software processes have been created. More recently, agile processes have increased their popularity. Its principles and practices fit with the needs of Client and software development teams. This increasing adoption has led software industry to develop tools that assist in project management with agile processes. This project analyzes and compares some of these tools, according to a defined reference model, and presents the implementation of ProjectIT-Enterprise system. This system, a collaborative tool for project management, seeks to ensure the project management according traditional or agile processes and the proper alignment between a process and a project, as well as to improve the communication and collaboration between all members of a software development team.

Keywords: Organizational Management, Process Management, Project Management, Process-Project Alignment, Agile Processes, ProjectIT-Enterprise.

1 Introduction

The software process development community has known a steady evolution over the past years. In 2001, a group of people with expertise in the area of Information Systems met with the aim of exchanging ideas on issues related to the software development process and wrote the Manifesto for Agile Software Development [4, 5]. This manifesto contains the values and principles of agile processes [5]. Both traditional and agile processes aim to ensure the quality of software produced. Thus, selection of the correct process for a given project is essential. Therefore, and due to its characteristics, the agile processes should be adopted in projects with volatile environments and where requirements are uncertain [1, 2, 3]. Traditional processes are better suited to projects with stable requirements and future requirements that are predictable [1, 2, 3]. However, both types of processes can be used at the same time, making RUP agile [6].

The starting point for this work was the identification of two communities related to the software development practice: project management and software process development. The project management community is related with management of scope, time, cost, quality, communication, organization and risk [4]. The software process development community, represented by examples such as RUP, XP, Scrum and OpenUP, reflects on aspects such as definition of processes, tasks, specific roles, work products and good practices [4]. Another important area in the existing software development is the organizational management area, reflecting on aspects such as the definition of skills, people management and its activities. Therefore, the goal of this project is to define a model and a set of mechanisms that allow the alignment of both organizational and processes dimensions with project dimension, considering the project management with agile processes, in particular both Scrum and XP processes. As the ProjectIT-Enterprise (see [4, 7, 22]) system currently allows this alignment to traditional processes, one of the challenges is guarantee the flexibility of the model referred in order to support the project management with both traditional and agile processes. Thus, the general goals of this project covers the following aspects: define a reference model for the analysis of agile tools; analyze the previous version of ProjectIT-Enterprise (ProjectIT-Enterprise/2007) according to the reference model and identify its limitations; development of a new version of ProjectIT-Enterprise (ProjectIT-Enterprise/2010) covering the identified limitations; evaluate the results through a case study.

In the next chapter, related work will be described. In the third chapter, the ProjectIT-Enterprise conception will be presented. The fourth chapter presents the design and architecture of ProjectIT-Enterprise. The fifth chapter presents a case study and the last chapter points the conclusions and relevant future work.

2 Related Work

2.1 Agile Processes

Primarily, in order to define the reference model for the analysis of agile tools, some agile processes were studied such as Extreme Programming (XP) [4, 11, 12], Scrum [4, 13, 14, 15, 23], Adaptive Software Process [24], Crystal [25] e OpenUP [26].

2.2 Agile Tools

This project analyzes four agile tools: VersionOne [16], Pivotal Tracker [17], RallyDev [18] and TargetProcess [19]. Additionally, the previous version of ProjectIT-Enterprise (see [20]) system was analyzed.

2.3 Reference Model

After the analysis of agile processes, a set of **activities** that must be supported by agile tools were identified and grouped in typical activities of a software development process:

- Requirements Management (**Backlog Management**)
- Quality Management (**Test and Defects Management**)
- Time Management (**Planning/Monitoring of Releases/Iterations; Reports and Charts**)
- People Management (**Management of Collaborative Teams**)

2.4 Comparative Analysis

This section analyzes the agile tools identified using the reference model presented in section 2.3.

General Characteristics

Table 2.1: Analysis of general characteristics of the tools

Characteristics	VersionOne	Pivotal Tracker	RallyDev	TargetProcess	PIT-Enterprise/2007
Installation Options	On-Site and On-Demand	On-Demand	On-Site and On-Demand	On-Site and On-Demand	On-Site and On-Demand
Multiple Projects	Yes	Yes	Yes	Yes	Yes
Web-Based	Yes	Yes	Yes	Yes	Yes
Supported Processes	XP, Scrum, DSDM and AgileUP	Scrum	RUP, XP and Scrum	XP and Scrum	RUP

Requirements Management

Table 2.2: Analysis of requirements management

Characteristics	VersionOne	Pivotal Tracker	RallyDev	TargetProcess	PIT-Enterprise/2007
Backlogs Prioritization (drag-and-drop)	Yes	Yes	Yes	Yes	No
Backlog Items Type	Feature, Enhancement and Defect	Feature, Bug, Chore and Release	User Story, Defect and Defect Suite	Feature, User Story and Bug	Activities
Items Estimation	Story Points	Story Points	Story Points	Hours	Days
Split Stories into Sub-Stories	Yes	No	Yes	Yes	No

Quality Management

Table 2.3: Analysis of quality management

Characteristics	VersionOne	Pivotal Tracker	RallyDev	TargetProcess	PIT-Enterprise/2007
Test Management	Integration with Fitnesse and HP Quicktest Pro	No	Integration with Fitnesse	Integration with Selenium, Nunit and JUnit	Yes
Defect Management	Integration with Bugzilla and JIRA	Yes	Integration with Bugzilla and JIRA	Integration with Bugzilla, JIRA and TestTrack Pro	No

Time Management

Table 2.4: Analysis of time management

Characteristics	VersionOne	Pivotal Tracker	RallyDev	TargetProcess	PIT-Enterprise/2007
Releases	Yes	Yes	Yes	Yes	Yes
Sprints (or Iterations)	Yes	Yes	Yes	Yes	Yes
Estimation of Releases	Story Points and Days	Days	Story Points and Days	Days	Days
Estimation of Iterations	Story Points and Days	Story Points	Story Points and Days	Days and Hours	Days
Estimation of Tasks	Hours	No	Hours	Hours	Days
Monitoring of Stories	Yes - Storyboard	Yes	Yes	Yes - Storyboard	No
Monitoring of Tasks	Yes - Taskboard	No	Yes - Taskboard	Yes - Taskboard	Yes
Charts	Iteration/Release Burndown/ Cumulative flow and Velocity Chart	Iteration Burn-Up, Release Burndown, Velocity Chart	Iteration/Release Burndown/ Cumulative flow and Velocity Chart	Sprint/Release Burndown/ Cumulative flow and Velocity Chart	No
Reports	Yes	Yes	Yes	Yes	No
Reports Format	PDF	No	PNG, PDF and JPG	No	No

People Management

Table 2.5: Analysis of people management

Characteristics	VersionOne	Pivotal Tracker	RallyDev	TargetProcess	PIT-Enterprise/2007
Definition of Teams	Yes	Yes	Yes	Yes	Yes
Definition of Roles Process	Yes	No	Yes	Yes	No
Indication of the work allocated to each element	Yes	Yes	Yes	Yes	Yes

3 ProjectIT-Enterprise – Conception

3.1 Domain Models

This section presents the domain models of new ProjectIT-Enterprise system. The next figure illustrates the domain Model of Organizational management area. In this area, the Human Resources Manager is responsible for defining and assigning skills to users. Each skill has a cost and corresponds to a set of organizational and technical skills (ex, “Programmer” and “System Architect”).

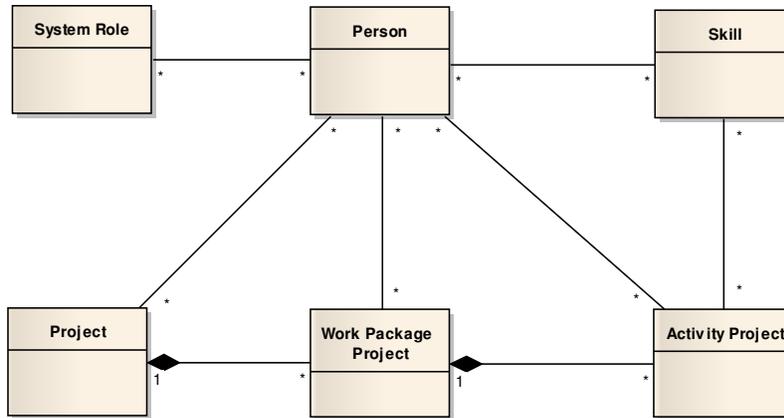


Figure 3.1: Domain model of organizational management area

The processes management area involves a number of aspects, including definition of tasks, roles, templates, disciplines and good practices to follow (see [4, 22]). The Domain Model is presented next.

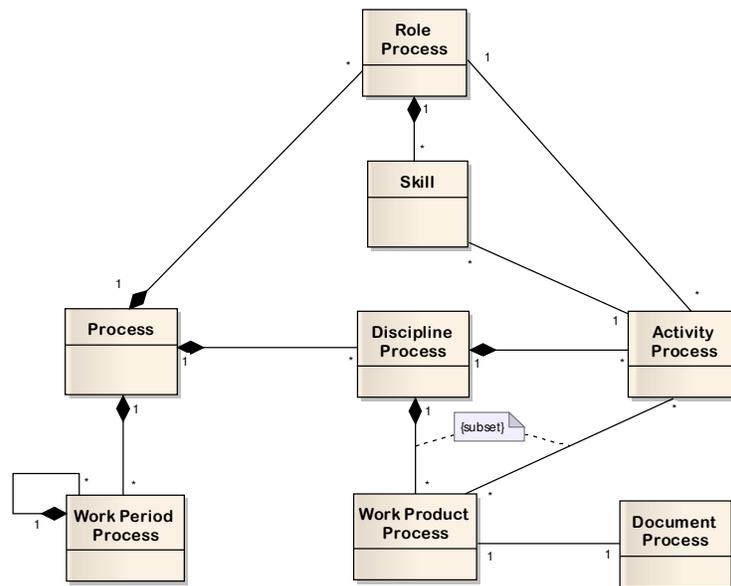


Figure 3.2: Domain model of processes management area

A Process has a set of disciplines. Each discipline is a logic composition of a set of activities, grouping them according to areas of interest (e.g., requirements, implementation, etc). An activity has information about the skills that are most appropriated for its execution. Each process is composed of work periods that represent a time window that allows project planning and monitoring. For example, XP would have been the following work periods: Releases and Iterations. Additionally, it is possible to define a hierarchy (e.g., in XP Release, Iteration and Task are level 1, 2 and 3). A process has also specific roles. For example, Scrum would have been the roles of Product Owner, Scrum Master and Team.

It is also noted that almost all the concepts defined in this area correspond to general concepts, i.e., a work product is actually a work product template (e.g., Word template for writing a requirements document) and an activity is actually a generic description of an activity. Only in Project management area these concepts are instantiated in both concrete work products and activities.

The projects management area involves aspects such as scope definition, requirements management, time management, risk management, cost management and communication management (see [4, 22]). Next is presented the domain model of projects management area.

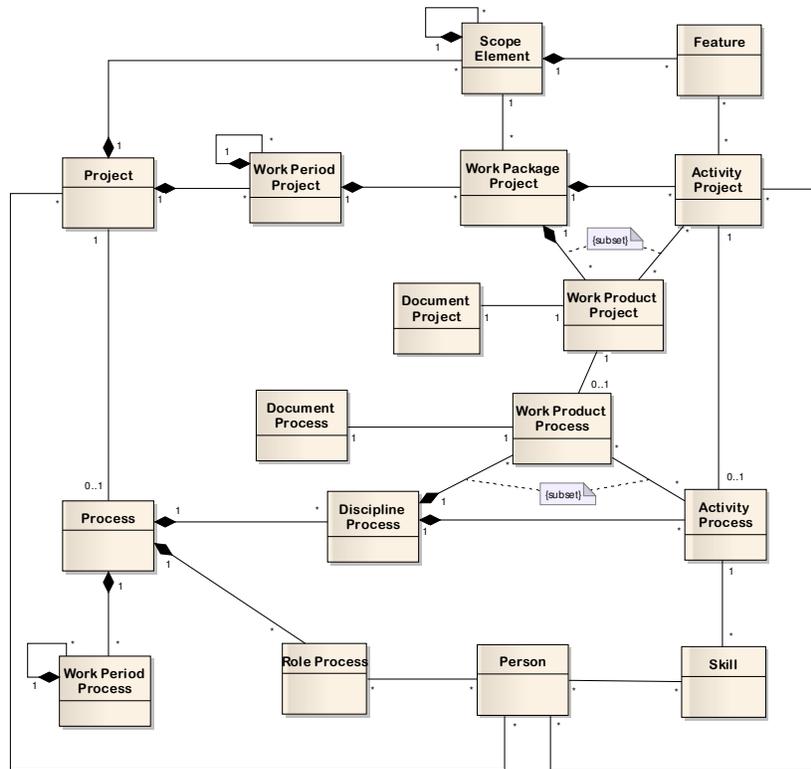


Figure 3.3: Domain model of projects management area

A project can be associated with a specific process, and if this alignment takes place, the project manager can select the disciplines and templates of activities/work products that he wants. By associating a process to a project, the work periods of project will be based on the terminology of the chosen process. Each work period is composed of work packages. In a traditional process, the concept of work package corresponds to a normal work package at WBS (Work Breakdown Structure) while in a Scrum process it would present the same characteristics of a Sprint Backlog, with a list of tasks set for each Sprint (Work Period Project). For each work package is necessary to describe its activities. Each activity has an effort, measured in Man-Days, and has elements responsible for its implementation.

3.2 Actors

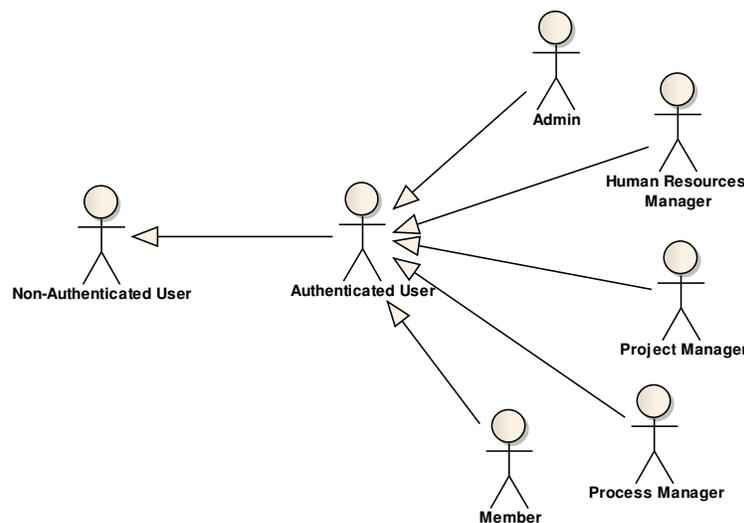


Figure 3.4: Actors of ProjectIT-Enterprise system

The actors' permissions to the modules developed are referred in the next chapter.

4 ProjectIT-Enterprise – Architecture and Design

ProjectIT-Enterprise has been developed on the WebComfort platform. The WebComfort is a CMS (Content Management System) Framework developed and supported by both Microsoft technologies ASP.NET 2.0 (C #) and SQL Server 2005. A fully description of WebComfort can be consulted in [8, 9, 10, 21].

ProjectIT-Enterprise consists of four main sections (or dynamic pages) namely, **Dashboard**, **Organizational**, **Processes** and **Projects**. Additionally, each **Process / Project** has its own section. Each section of the system has a set of specific modules WebComfort which were developed throughout this work. The presentation of the contents of a module (Module Layout) is implemented using ASP.NET User Controls (.ascx files). In order to edit the contents of a module, are used static pages (.aspx files, Module Support Page). The next figures presents the WebComfort modules developed for each section of ProjectIT-Enterprise.

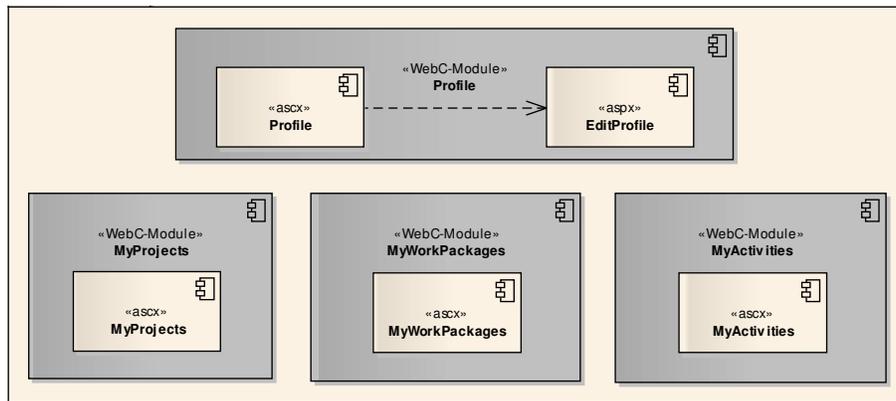


Figure 4.1: Modules developed for **Dashboard** section

The Profile module presents the personal information of the user, which can be edited by him, by accessing the EditProfile page. This module is only editable and viewed by authenticated user. The other modules allow the user to view its activities, its work packages and its projects under execution.

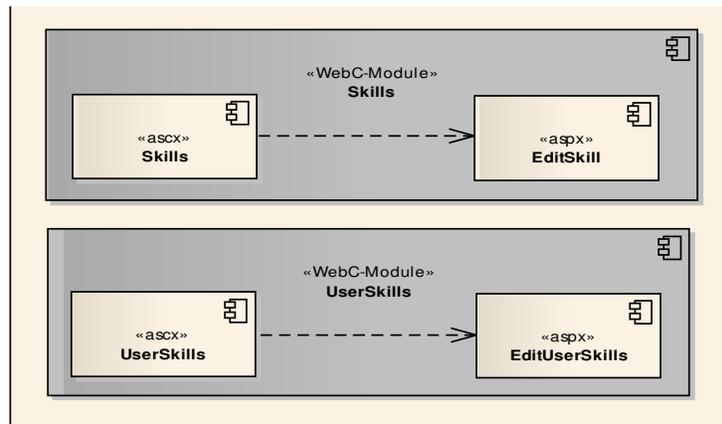


Figure 4.2: Modules developed for **Organizational** section

The Skills module shows the skills defined in the system and its cost associated. The UserSkills module presents the skills associated with each of the registered users. The content present in both modules is managed by Human Resources Manager. The non-authenticated user can only view the Skills module.

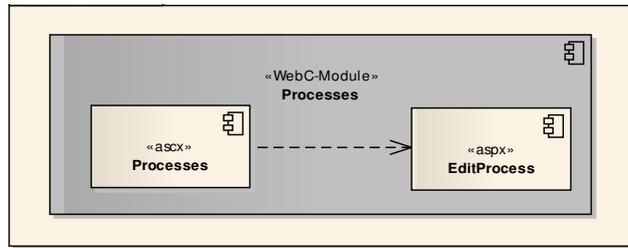


Figure 4.3: Modules developed for **Processes** section

This module presents the processes defined in ProjectIT-Enterprise. This module can be viewed by any user and is managed by Process Manager.

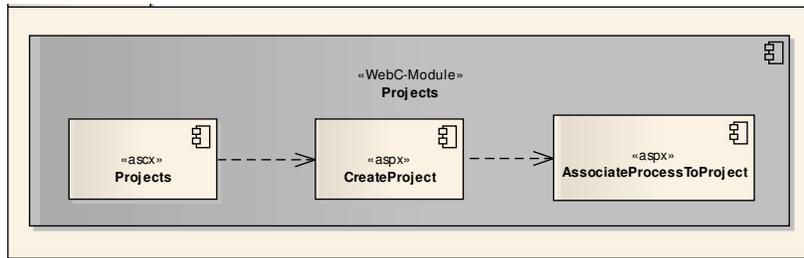


Figure 4.4: Modules developed for **Projects** section

This module presents the projects defined in ProjectIT-Enterprise. From this module, it is possible create new projects, by accessing to CreateProject page. In this page, a Project Manager may choose to associate a template of a process. If yes, he can choose a set of disciplines, activities, work products and work periods to instantiate in the project through a wizard, on the AssociateProcessToProject page. This module can be viewed by any user and is managed by Project Manager.

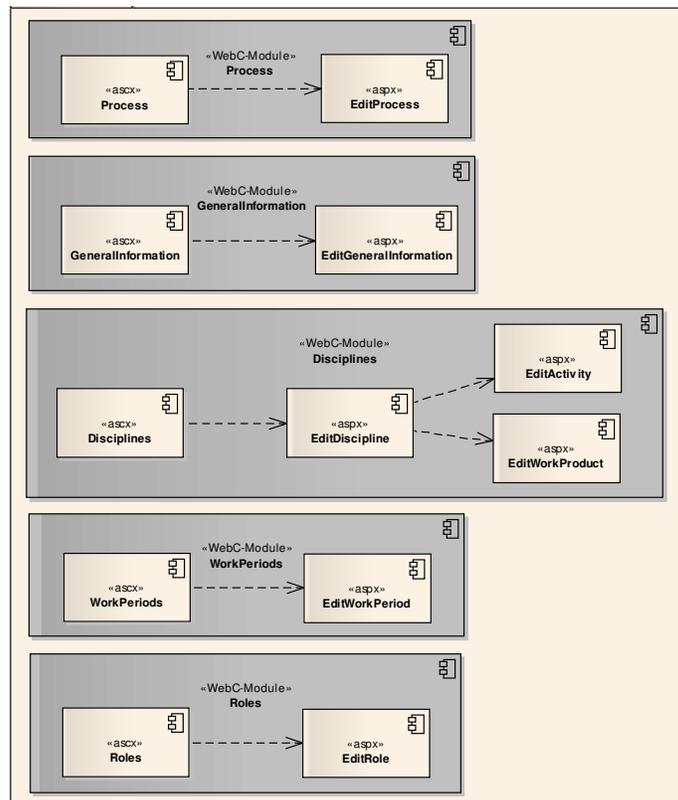


Figure 4.5: Modules developed for **Process** section

The Process module contains both name and logo process. The other modules are related to the concepts identified in domain model of processes management area (see section 3.1). All modules of this section can be viewed by any user but are managed only by Process Manager.

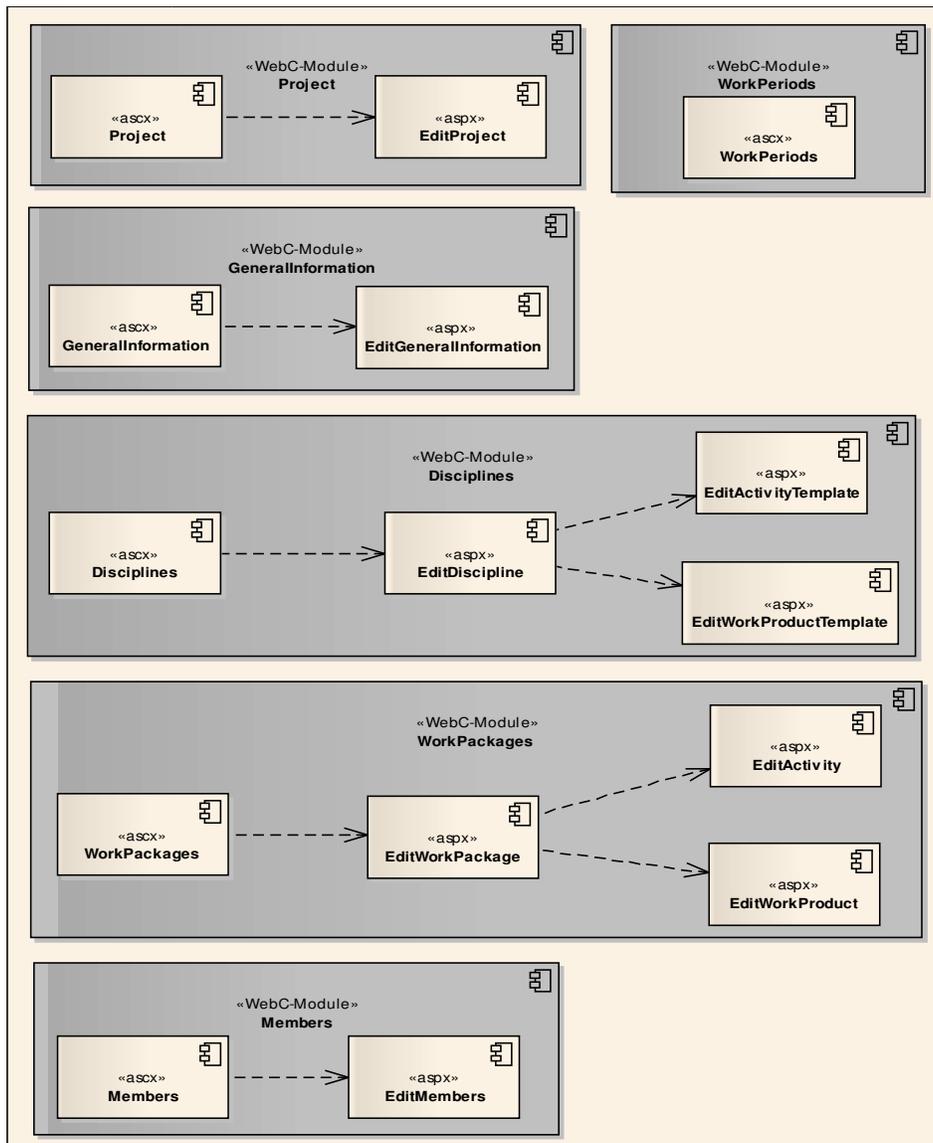


Figure 4.6: Modules developed for **Project section**

These modules are related to the concepts identified in domain model of projects management area (see section 3.1). All modules of this section are managed by Project Manager and can be viewed by Anonymous users (if privacy level of the project is “Public”), by Authenticated users (if privacy level of the project is “Registered Users”) or only by members of the project (if privacy level of the project is “Private”). The WorkPackages module can be managed by Project Manager or by Member users (if Project Manager delegates this responsibility).

5 Validation

In this chapter, the proposed solution is validated by applying a case study, based on this project. Therefore, we defined a project with the name of the current work, "Project Management in Agile Processes", associated with a process called “Dissertation MEIC / IST”. This case study can be consulted in the instance of the prototype, available at URL: <http://isg.inesc-id.pt/pit-enterprise>. The next sections illustrate the application of this case study.

5.1 Definition of process “Dissertation MEIC/IST”

Initially, a set of disciplines, activities and work products templates were defined, reproducing the various steps followed in a dissertation. The work periods defined were two: School Year (level 1; duration between 6 and 12 months) and Semester (level 2; duration between 1 and 6 months). Three roles were identified: student; supervisor and co-supervisor.

5.2 Definition of project “Project Management with Agile Processes”

This project was created and the process defined in the previous section was associated. Therefore, all the disciplines, activities, work products and work periods became available to the Project Manager. The project was created with the following information:

- Initial Date – 01/09/2009
- Final Date – 16/10/2010
- Privacy Level – Public
- Project Manager – Miguel Pinto
- Team Members – Alberto Silva (supervisor), David Ferreira and João Saraiva (co-supervisor) and Miguel Pinto (student)

The following work periods were created:

- Level 1 – School Year 2009/2010
- Level 2 – First Semester and Second Semester

Two work packages were associated at each one of the work periods of level 2:

- Course “Projecto de Mestrado”, associated at first semester.
- Course “Dissertação de Mestrado”, associated at second semester.

Each work packages has activities and work products. All the information of this project can be consult at URL prototype mentioned above.

6 Conclusions

6.1 Discussion

The case study presented in the previous chapter gives us a set of considerations about system developed:

- The concepts "Work Period" and "Work Package" fit well both in project management based both on traditional and agile processes.
- The project management is more efficient and accurate with the existing alignment mechanisms in disciplines, activities, work products and work periods.
- The fact that the system has been developed on a CMS platform introduces many advantages in terms of modularity, extensibility and the way how users interact with the system. Additionally, ProjectIT-Enterprise takes advantage from other capabilities of WebComfort, such as creating dynamic pages, related to projects and processes, and communication mechanisms between modules.
- Export a project to Microsoft Project allows a team to take advantage of both communication and collaboration tools, while the Project Manager can use Microsoft Project to produce its many reports and graphs.
- The system ProjectIT-Enterprise has a clear focus on activities related to communication and collaboration (e.g. Wiki) that was not possible to validate in this case study, because the project defined isn't very collaborative (only one element).

6.2 Future Work

The ProjectIT-Enterprise can be improved with new functionalities. Primarily, it would be interesting to define a case study with more team members to validate the ProjectIT-Enterprise in aspects related to communication and collaboration. Additionally, in order to complement these aspects would be interesting to take advantage of some emerging technologies of social computing. Additionally, many project teams use Microsoft Excel when initially intend to adapt the project management with agile processes. Thus, the ProjectIT-Enterprise would provide the possibility to import/export files from Excel.

7 References

1. Michel dos Santos Soares: Comparação entre Metodologias Ágeis e Tradicionais para o Desenvolvimento de Software (2004)
2. Thelma Hataria: The Confounding World of Process Methodologies. In: Proceedings of the Fourth Annual College of Computing, Engineering and Construction Symposium, University of North Florida, USA (2006)
3. Everette R. Keith, Agile Software Development Processes: A Different Approach to Software Design (2002)
4. Alberto Silva, Carlos Videira: UML, Metodologias e Ferramentas CASE, 2nd Edition, Vol. 2, March 2008, Centro Atlântico Editora, ISBN: 978-989-615-061-7. Chapter 3 – Metodologias Ágeis & Chapter 7 – Iniciativa ProjectIT
5. Manifesto for Agile Software Development: <http://agilemanifesto.org/>
6. Michael Hirsch: Making RUP Agile. In: Proceedings of the 2002 ACM SIGPLAN Conference on Object-Oriented Programming Systems, Languages and Applications (OOPSLA), Seattle, Washington, USA (2002)
7. Alberto Silva: O Programa de Investigação “ProjectIT” (White Paper), v1.0 (2004)
8. WebComfort: www.webcomfort.org
9. Alberto Silva, João Saraiva: The WebComfort Framework: An Extensible Platform for the Development of Web Applications. 34th EUROMICRO Conference on Software Engineering and Advanced Applications, Service and Component-Based Software Engineering Track, Italy (September 2008)
10. Alberto Silva: WebComfort – Gestor de Conteúdos e Aplicações Web: Plataforma, Extensões e Aplicações (White Paper), v1.0 (2007)
11. Kent Beck: Extreme Programming Explained: Embrace Change. Addison-Wesley, Reading, PA, Vol. 32 (1999)
12. Extreme Programming: A Gentle Introduction: <http://www.extremeprogramming.org/>
13. Ken Schwaber, Mike Beedle: Agile Project Management with Scrum. Prentice Hall, (February 2002)
14. Rui Francisco: Ferramenta de Gestão de Processos de Desenvolvimento de Software. Master thesis, Instituto Superior Técnico (2008)
15. What is Scrum?: <http://www.controlchaos.com/>
16. VersionOne: <http://www.versionone.com/>
17. Pivotal Tracker: <http://www.pivotaltracker.com/>
18. Rally Software Development: <http://www.rallydev.com/>
19. TargetProcess: <http://www.targetprocess.com/>
20. Gonçalo Marcos, Miguel Silva: ProjectIT-Enterprise – Integração com Requisitos. Relatório Trabalho Final de Curso. Instituto Superior Técnico (2007)
21. João Leonardo Carmo, Alberto Rodrigues da Silva: The WebComfort Project. Second International Conference of Innovative Views of .NET Technologies (IVNET), SBC & Microsoft, Brazil (2006)
22. Paula Martins: ProPAM – A Software Process Improvement Approach based on Process and Project Alignment. PhD Thesis, Instituto Superior Técnico, (2008)
23. Scrum Alliance: Transforming the World of Work: <http://scrumalliance.org/>
24. J.Highsmith: Adaptive Software Development: A Collaborative Approach to Managing Complex Systems, Dorset House (2000)
25. Alistair Cockburn: Agile Software Development, Addison Wesley (2002)
26. Introduction to OpenUP: <http://epf.eclipse.org/wikis/openup/>