INSCO REQUISITE
A Web-Based RM-Tool to support Hybrid Software Development

F. J. Orellana, J. Cañadas, I. M. Del Águila and S. Túnez
Department of Computer Science, University of Almería
Ctra. Sacramento s/n, 04120, Almería, Spain

Keywords: Requirements Management, RM Tools, Hybrid Software, Knowledge Engineering.

Abstract: This paper presents InSCo Requisite, a Web-based Requirements Management Tool (RM-tool) supporting InSCo, a methodology based on CommonKADS and RUP, for developing software systems in which information systems coexist with knowledge-based components. A revision of similar requirement management tools is presented, as well as a review of the main requirements proposed for this kind of tools.

1 INTRODUCTION

In many cases the complexity of the domains entails software to provide solutions to problems related to decision support process, transaction processing and information management. This situation forces development process to apply both Software Engineering (SE) and Knowledge Engineering (KE) methodological approaches.

In this context, Hybrid Systems (Kendal, 2001) are software systems formed by several components or subsystems, some of which perform knowledge intensive tasks related to the decision support process, and others performing tasks related to transactions processing and information management in databases. The former are knowledge-based components and the latter are non-knowledge-based components.

The application of existing methodologies developing hybrid software presents some problems that make the intertwining between SE and KE necessary, in order to obtain the best result and a successful final product (Acuña, 1999).

The InSCo methodology (Águila, 2006) emerges as a solution for specific problems in hybrid systems software development, specifying in an integrated method, the process, activities, languages and tools used for this type of systems, based on CommonKADS (Schreiber, 1999) and RUP (Jacobson, 2000).

Requirements elicitation is a basic InSCo activity. The management of a large amount of information and requirements is a critical task in Requirements Engineering (RE) process (Leffingwell, 2000). This work presents InSCo Requisite, a web-based tool which supports requirements management in the development of hybrid software systems.

InSCo Requisite is available at the following URL: http://www.dkse.ual.es/insco/ . This tool is the final product which evolved from a prototype version described by Orellana (2005).

The rest of this work is structured as follows: section 2 explains the reasons that led us to build a Web-based tool and the alternatives that were considered. Section 3 reviews common requirements for RM tools focusing on those that were applied to InSCo Requisite, and the architecture applied in the design. Then, section 4 describes an overview of usage. Finally, the main conclusions and future work are summarized.

2 RELATED WORKS

The success of the RE process depends mainly on the collaborative work performed by the different members, since a good participation will result in a well defined system (Sinha, 2006) (Lang, 2001). For this reason, any tool supporting the RE process for the InSCo methodology should guarantee a direct collaboration among the stakeholders involved in a project.

Hybrid software projects usually need the contribution of experts in a specific domain (agriculture, medicine, architecture, etc) that do not

J. Orellana F., Cañadas J., M. Del Águila I. and Túnez S. (2008).
INSCO REQUISITE - A Web-Based RM-Tool to support Hybrid Software Development.
In *Proceedings of the Tenth International Conference on Enterprise Information Systems - ISAS*, pages 326-329
DOI: 10.5220/0001712203260329

belong to the development team. If experts cannot properly participate on the projects, the cooperative work and consequently, the project progress could be affected. Therefore, stand-alone tools were not suitable for our purpose.

A Web-based application solves this problem. It provides an easier access to the tool avoiding the need to install a software client, and ensuring that every stakeholder is able to participate actively in a project from anywhere at anytime by means of an Internet connection and a browser.

DOORS®/Net, Rational Requisite® Web and IRqA®-Net are three of the most complete web-based tools for requirement management. They are mostly considered as a complement of their corresponding stand-alone versions for some eventual situations where a stakeholder has to work from a remote place. Other web-based tools were also considered as an alternative to carry out the RM process of the InSCo methodology. However, they were designed to deal with traditional software, without regard to any aspect related to the knowledge-based components presented in a hybrid software system.

These reasons led us to build a specific web-based tool to reach our goals. InSCo Requisite deals with the characteristic aspects related to the development of hybrid software following the steps and activities defined by the InSCo methodology.

3 DESIGN APPROACH

Requirements needed for every RM-tool can mainly vary depending on the organization and project scope or the development team process maturity. Hoffmann (2004) proposes a wide range of functional requirements for RM-tools, a set of features mostly based on project experience in the area of automotive and defence systems, whose aim is helping to achieve the main goals of RM. The most important features for our tool were gathered from this catalogue, considering the main factors involved in our work: the development methodology for hybrid software that InSCo Requisite supports, the available resources and the type of stakeholders involved. These features are listed as follows:

- Management of unlimited projects.
- Support for scalable projects.
- Administration of users, user groups and rights defined for each project.
- Identifiers for each project object over its lifetime
- Support for formatting text and attachment of non-text objects (files and diagrams)
- Control of changes for each requirement.
- Traceability through links between requirements.
- Interoperability with other tools in order to continue the development process.
- Documents generation.

A web-based architecture is compulsory if we want to access applications from the Internet with a web browser. The J2EE (Java 2 Enterprise Edition) platform was chosen for the development because it offers a powerful alternative for projects of medium and high complexity. Along with this platform, a MVC (Model View Controller) pattern was applied using Struts, an open-source and multiplatform framework developed by the Apache Software Foundation. These technologies allow the enhancement of robustness, code reuse and a better organization of an application that from its initial stage was expected to become considerably complex.

The tool is structured in a three-tier architecture, which separates presentation logic from business logic and persistent storage. First, the Presentation Tier represents the client side, web browsers sending requests using the HTTP (HyperText Transfer Protocol) protocol to the Application Server in the business layer, and representing the data sent back, basically composed by HTML (HyperText Markup Language), CSS (Cascading Style Sheet) and Javascript code. The Business Tier that stores the InSCo Requisite application files and is composed by the Apache Tomcat server. The server processes those incoming requests, executing the appropriated Java servlets, and exchanging data with the Data Tier if necessary, through a JDBC (Java DataBase Connection) connection. Finally, the Data Tier, in charge of keeping the data persistence, is formed by the Oracle Database Server.

InSCo Requisite is composed by two well-differenced modules. The administration module allows the setting of actions related to the management of projects and users which involve the creation of new development projects, registration of new system users and management of user groups and right permissions.

The system can host unlimited hybrid software projects formed by several work groups. Users can take part in many projects simultaneously, adopting different roles on each one. Work teams are used as a mechanism to classify the project users, define roles and establish the rights permission for each project.

The main module guides the requirements management of hybrid software projects covering both business and services level defined by the InSCo methodology.

4 USAGE OVERVIEW

The management is basically performed using templates which are identified by means of their prefixes; business level templates start with the BSN prefix, whereas Services level templates use the SER prefix (Figure 1).

A hybrid software project starts with the analysis of the organization and the problems that can be solved, stored in the template BSN_Project1. This step is followed by the creation of other templates at the business level. BSN_Goal presents the strategic goals of the organization than can be refined in subgoals before the services level is started. BSN_Domain stores the information resources needed to reach those strategic goals. Finally, BSN_Actor describes the whole group of people or roles implied in the project.

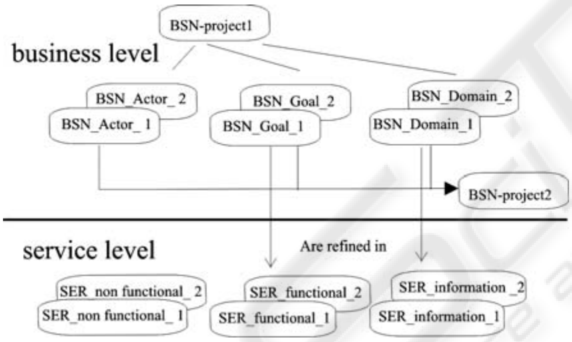
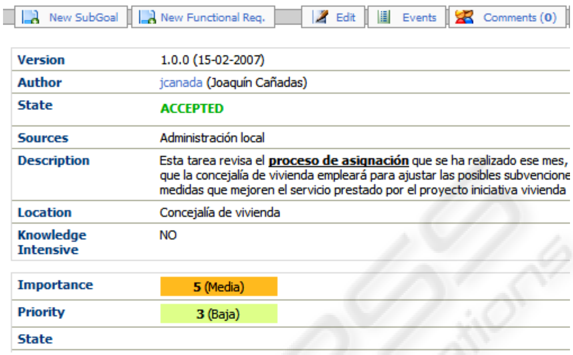


Figure 1: InSCO templates.

In order to complete the business model, InSCo defines a BSN_Project2 template used to analyse the feasibility in terms of technology and economy.

The elements of the services model are mostly refinements of the business level templates. SER_Functional template represents the functional requirements of the systems refining the goals. SER_Information describes the information requirements refining the domain information. Finally, SER_Non_Functional templates collect the system restrictions.

The management is performed using HTML forms and hyperlinks that allow the navigation among requirements. Most templates include a Rich Text Editor in text fields to guarantee a more flexible description of the information. Since requirements descriptions are performed in natural language, it was essential to avoid ambiguity offering a solution for a better representation of the information. Figure 2 presents a template of a goal specification, using the rich text feature.



Version	1.0.0 (15-02-2007)
Author	jcanada (Joaquín Cañadas)
State	ACCEPTED
Sources	Administración local
Description	Esta tarea revisa el **proceso de asignación** que se ha realizado ese mes, que la concejalía de vivienda empleará para ajustar las posibles subvencione medidas que mejoren el servicio prestado por el proyecto iniciativa vivienda
Location	Concejalía de vivienda
Knowledge Intensive	NO
Importance	5 (Media)
Priority	3 (Baja)
State	

Figure 2: InSCo Requisite template example.

In order to make navigation easier and to present a global view of the relationships between the project templates, a project explorer with a hierarchical representation of the project content is shown.

Most of the elements offer the possibility to attach files and diagrams which can be used to add important information. Templates have a validation state that informs whether a requirement is accepted or rejected. Changes on requirements are recorded in a change history which shows the author, date and the type of action performed.

The tool provides a group of reports which present the requirements classified into different types and, by doing so, allow their exportation to Microsoft Excel (XLS) files.

RM-tools must provide a way to continue the development once the requirements have been specified. InSCo Requisite allows the interoperability with other tools to deal with several aspects. On the one hand, the development of the knowledge-based section as a result of the knowledge-intensive requirements, and, on the other hand, the functionality based on the requirements related to the traditional software.

It offers an option to export the project content to Conceptual Modeling Language code. This code is generated by the whole group of requirements selected as knowledge intensive by the project stakeholders.

The second approach, the functionality based on traditional software, is performed by the use of XML Metadata Interchange (XMI), which provides a standard format for storing UML (Unified Modeling

Language) models and loading those using different UML tools without interoperability problems. Therefore, InSCo Requisite supports the generation of XMI files that incorporates the content of a development project in order to import them from other tools afterwards.

InSCo Requisite includes some facilities for informal collaboration in the requirement management environment. Discussions about requirements are available where every stakeholder can give their opinion, suggest ideas and changes. Messages, along with the author and date are stored in the database in order to keep a discussion log chronologically ordered, which can be consulted during the project lifetime.

5 CONCLUSIONS

InSCo Requisite has been presented in this paper. This tool supports a development methodology called InSCo which integrates SE and KE techniques and methods.

InSCo Requisite provides the management requirement of hybrid software projects, including mechanisms to handle knowledge-intensive requirements and which allows interoperability with other tools to continue the development process beyond requirements.

A web-based architecture has been applied in order to facilitate the participation of distributed work groups. The application design makes a multi-language support possible with Spanish and English versions.

The tool is being evaluated in the development of a hybrid software system in the agriculture domain.

The future work is related to the enhancement of the collaborative features and project management aspects. An important improvement involves the implementation of a traceability control of requirements along the whole development process. As far as the collaborative aspects is concerned, virtual meetings will improve the communication between the stakeholders. Automatic notification of requirement changes and the subscription to important requirements will be used to keep stakeholder informed about the project activity.

ACKNOWLEDGEMENTS

This work is part of the Research Project TIN2004-05694, funded by the Spanish Ministry of Education and Science.

REFERENCES

Acuña, S.T., López, M., Juristo, N. and A. M. Moreno. (1999). Process model applicable to software engineering and knowledge engineering. *International Journal of Software Engineering and Knowledge Engin*eering, 9(5), 663–687.

Águila, I.M., Cañadas, J.J, Palma, J. and Túnez, S. (2006). Towards a Methodology for Hybrid Systems Software Development. *In Proceedings of the Eighteenth International Conference on Software Engineering and Knowledge Engineering (SEKE'06).* San Francisco, USA, pp 188-195.

Hoffmann, M., Kuhn, N., Weber, M. and Bittner, M., (2004). Requirements for requirements management tools. In *12th IEEE International Requirements Engineering Conference Proceedings*., 301-308.

Jacobson, I., Booch, G. and Rumbaugh, J. (2000). *The unified software development process*. Addison Wesley.

Kendal, S. and Chen, X. (2001). Towards Hybrid Knowledge and Software Engineering. *In Proceedings of International Conference on Computing and Information Technologies,* pp. 449-454.

Lang, M. and Duggan, J., (2001). A Tool to Support Collaborative Software Requirements Management. *Requirements Engineering,* 6 (3), pp. 161-172.

Leffingwell, D. and Widrig, D. (2000) *Managing Software Requirements - A Unified Approach,* Addison Wesley Longman, Inc.

Orellana, F.J., Guil, F., del Águila, I.M. and Túnez, S., (2005). A WEB-CASE Tool Prototype for Hybrid Software Development. *Lecture Notes in Computer Science*, vol. 3643, pp. 217-222.

Schreiber, G., Akkermans, H., Anjewierden, A., deHoog, R., Shadbolt, N., VandeVelde and W., Wielinga, B., (1999). *Knowledge Engineering and Management: The CommonKADS Methodology*. The MIT Press.

Sinha, V., Sengupta, B. and Chandra, S., (2006). Enabling Collaboration in Distributed Requirements Management. *IEEE Software*, vol. 23(5), pp. 52–61.

Sommerville, I. and Sawyer P., (1997). *Requirements Engineering: A Good Practice Guide*. John Wiley & Sons, Inc. New York, USA.