A Tool to Support End-User Development of Web Applications Based on a Use Case Model

Buddhima De Silva, Athula Ginige, Simi Bajaj, Ashini Ekanayake,
Richa Shirodkar, and Markus Santa

University of Western Sydney, Locked Bag 1797, Penrith South DC, 1719, NSW, Australia
bdesilva@scm.uws.edu.au, a.ginige@uws.edu.au

Abstract. Many Small to Medium Enterprises (SMEs) tend to gradually adopt Web based business applications to enhance their business processes. To support this gradual adoption we need a framework that supports iterative development. Further processes that have been supported by web based business applications can change and evolve requiring applications to be changed accordingly. One way to facilitate these requirements is to empower SMEs to make changes to the web application to accommodate the required changes while using that web application. This paper presents a requirement specification tool and a Use Case model of a Web based application which can be used to specify and create the Web applications. This approach will bring the formalism to ad-hoc end-user development by including requirement specification phase. In this approach an application is viewed as a collection of packages. A package consists of related set of use cases. A scope list is developed at the package level. This will assure the required functionality of the application is completely specified. This will also provide a framework to validate the requirements. We have developed this tool in Component Based Development/ Deployment System (CBEADS). Now we are in a process of testing it.

Keywords: Web application, Requirement specification, Use Case model.

1 Introduction

AeIMS research group at University of Western Sydney has been working with SMEs in Western Sydney region to investigate how Information and Communication Technologies (ICT) can be used to enhance their business processes to become competitive in a global economy [1, 2]. In this research the challenge we had was to find a way to develop web based business applications that can evolve with changing business needs [3]. Also we had to develop these applications rapidly as well as in a cost effective manner [4]. The development approach should also needs to reduce the gap between what the users actually wanted and what is being implemented in terms of functionality [5].

To address above mentioned issues it is very important to empower end users to get involved in the original design of the application during design time and be able to

M. Gaedke, M. Grossniklaus, and O. Díaz (Eds.): ICWE 2009, LNCS 5648, pp. 527 – 530, 2009.

modify the application as a result of evolving requirements during the use time [6]. If we are to empower end users to actively participate during design time and to be able to modify the application during use time rather than developing a specific application we need to provide a set of tools and a framework that they can use to develop and change the application in response to changing needs. However, such tools need to focus on analysis and design process as well as the construction process of an application for the success of end-user development [7]. Therefore, the fundamental challenges to end-user development have gradually shifted from basic syntactic adjustment required to help during construction towards semantic challenges including the need to convey an understanding of design and engineering principles relevant to end-users [8]. We are developing a use case driven approach to face the semantic challenge with the specification and the design of the requirements. All the functional requirements are specified as use cases or functions. We have developed a use case model and a form based tool to capture the different aspects of the use cases/ functions. These are discussed in section 2 and section 3. Using this captured information most of the application can be auto generated.

2 Use Case Model of an Application

Fig.1. shows the high level model of an application used in the use case driven approach. Application consists of one or many packages. A package groups a set of related functions of an application. A package has a list of scopes. Scope list is a list of processes that must be carried out to accomplish the goal. This defines what the project will deliver and what it will not deliver and its boundary (organisational, functional, technical, system or individual user) by specifying them for individual processes. This helps to identify the use cases, actors, workflow, some of the data objects and the relationships between the objects required in the functions. Use cases are specified using user interfaces and user actions associated with the user interfaces. We also define the business rules associated with a package which is used in the implementation of use cases.

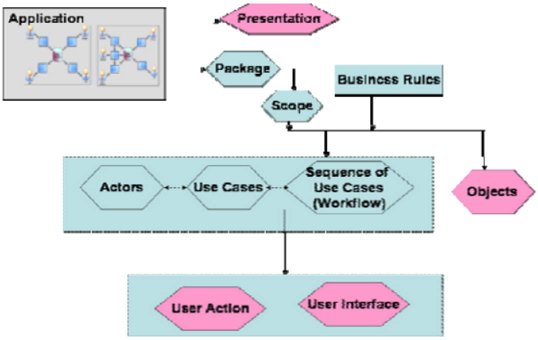
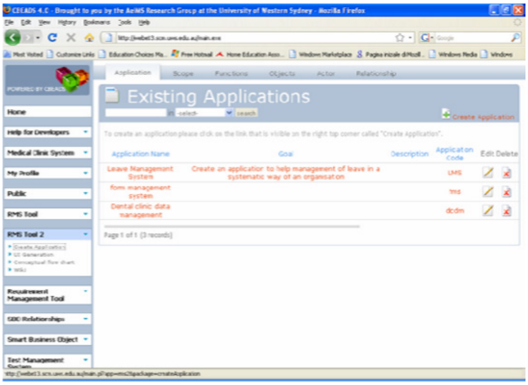


Fig. 1. High level model of an application

3 Requirement Specification Tool

Based on the above use case model we have developed a form based tool to specify the Requirements. This tool also includes some guidelines and examples for the users to specify the requirements of a Web based application. This tool is shown in fig. 2. This tool provides form based interfaces to specify the packages with in an application, scope list of a package, use cases/ functions, actors and objects. A form based interface is also provided to specify the interfaces related to the use cases. Using above information the application is semi-automatically generated. Table 1 shows how the captured information is mapped into the physical implementation.



Existing Applications

To create an application please click on the link that is visible on the right top corner called "Create Application".

Application Name	Goal	Description	Application Code	Edit	Delete
Leave Management System	Create an application to help management of leave in a systematic way of an organisation		LMS		
form management system			fms		
Dental clinic data management			dcdm		

Page 1 of 1 (3 records)

Fig. 2. Requirement Specification tool

Table 1. Purpose and corresponding realization of the inputs

Form Input		**Purpose**	**Corresponding Realisation**
Form Name	Input		
Application	Name, description, acronym	Acronym is used for namespace and folder name to store files for the application.	Folder structure for the application
Scope	Name and description	Scope defines the list of processes to be carried out in the system.	
Function/ use case	name, description	Function name used as the primary navigation to the function.	Menu link
Actor	Actor name, description	Used to conceptualise the use cases.	
Objects and Object Relationships	Application, Object Name, Attributes, relationships	Objects with the attributes are created within the application namespace. Data about Relationship is used to create the relationships.	Database, Table and Table columns and Foreign keys, Associative tables.
User Interface	Name,UI attributes, actions	UI for function is created with UI elements and actions.	file

This tool is implemented in Component based E Application Development and Deployment System (CBEADS) [3, 9]. Components and engines of CBEADS framework are used to map the specification of the application into the implementation. For example, this tool depends on CBEADS User Management component for binding the users with the created application. Also SBO [10] (Smart Business Objects) is used to map the objects and relationships to the data base tables.

4 Conclusion

We have developed a tool to specify the requirements of a Web based Business Applications based on use case driven approach. This approach is well suited to specify and create web based business applications for SMEs as it supports iterative development. We are now in the process of evaluating the tool.

References

1. Ginige, A.: Collaborating to Win - Creating an Effective Virtual Organisation. In: International Workshop on Business and Information, Taipei, Taiwan (2004)
2. Ginige, A.: From eTransformation to eCollaboration: Issues and Solutions. In: 2nd International Conference on Information Management and Business (IMB 2006), Sydney, Australia (2006)
3. Ginige, A.: New Paradigm for Developing Software for E-Business. In: Proceedings of the IEEE 2001 Symposia on Human Centric Computing Languages and Environments (HCC 2001). IEEE Computer Society, Stresa (2001)
4. Ginige, A.: Re Engineering Software Development Process for eBusiness Application Development. In: Software Engineering and Knowledge Engineering Conference -SEKE 2002, San Francisco Bay, USA (2002)
5. Epner, M.: Poor Project Management Number-One Problem of Outsourced E-Projects, in Research Briefs, Cutter Consortium (2000)
6. Fischer, G., et al.: Meta Design: A Manifesto for End -User Development. Communications of the ACM 47(9), 33–37 (2004)
7. Rosson, M.B., et al.: Design planning in end-user web development. In: IEEE Symposium on Visual Languages and Human-Centric Computing, Couer d'Alene, Idaho (2007)
8. Repenning, A.: End-User Design. In: Workshop on End-User Software Engineering (WEUSEIII), Dagstuhl Seminar Proceedings 07081 (2007)
9. Ginige, A., et al.: Smart Tools to support Meta-Design Paradigm for Developing Web based Business Applications. In: Baresi, L., Fraternali, P., Houben, G.-J. (eds.) ICWE 2007. LNCS, vol. 4607, pp. 521–525. Springer, Heidelberg (2007)
10. Liang, X., Ginige, A.: Smart Business Objects: A new Approach to Model Business Objects for Web Applications. In: 1st International Conference on Software and Data Technologies, Setubal, Portugal (2006)