A Software Reference Architecture for the Design and Development of Mobile Workflow Learning Applications

E. Castelán, M.A. Brigos, J. Fernández
Polytechnic University Of Cataluña, Department of Graphic Expression
edatreyu@gmail.com, miguel.brigos@upc.edu, jfernandez@ege.upc.edu

Abstract

The aim of this research is the design, implementation and evaluation of a Software Reference Architecture for WfLMS (Workflow Learning Management Systems) with Mobile, Cloud and Collaborative functionalities. The Reference Architecture was the result of applying the Workflow Management System Reference Model in conjunction with Mobile, Cloud and Collaborative Architectural pattern solutions. The Design Science Research methodology was applied in this research as the principal goal of the research is the same as in the methodology, the creation of artifacts for a practical purpose. The implementation of the Reference Architecture resulted in a mobile WfLMS for the iOS platform. A Delphi Study with a group of 12 experts was carried out in order to evaluate if the Mobile, Cloud and Collaborative architectural pattern solutions applied in the Reference Architecture enhanced the functionalities of a WILMS. 14 factors were identified in the pattern solutions that were implemented, the experts ranked each one of the factors with a 5 point Likert scale in a 3 round questionnaire. 12 of the factors achieved more than 75% of agreement with a strong level of group consensus.

Keywords
m-learning, workflow learning, collaborative, cloud, reference architecture, mobile

1. INTRODUCTION

In learning environments team collaboration face the same challenges as in business scenarios and without proper management team collaboration can become more an obstacle than a solution. Workflow Learning is the adaptation of WfMS (Workflow Management Systems) from the business domain in the learning domain (DePree et al. 2011). Cesarini et al. (2004) support the e-learning process with a WfMS and makes a perfect adaptation of the learning process as a Workflow, explains in a simple way that a Workflow which is made by participants, actions, documents and a set of procedural rules, can be adapted in a learning environment, the actors involved in the learning process are the participants, the learning activities are the actions, the artifacts students can produce are the documents and the relationship among the learning topics are the procedural rules. The aim of this research is the design, implementation and evaluation of a Software Reference Architecture for the design and development of Workflow Learning Management Systems with Mobile, Cloud and Collaborative functionalities. The Reference Architecture is the result of the application of the Workflow Management Systems Reference Model in conjunction with Mobile, Cloud and Collaborative Architectural pattern solutions.

2. METHODOLOGY

Because the result of this research is the design, development and evaluation of an artifact, the Design Science Research Methodology has been chosen. This methodology has as a principal goal: the creation of artifacts for a practical purpose with two important characteristics, relevance and novelty (Nicolaou & Geerts 2011). This methodology should be used to address unsolved problems in a unique and innovative way or for trying to enhance a solved problem in a more effective or efficient way (Hevner et al. 2004). Nicolaou & Geerts (2011) mention that the four possible artifacts that can result from the application of this methodology are: concepts, models, methods and instantiations.

Hevner et al. (2004) provides an understanding on how to conduct, evaluate and present Design Science Research. In (Hevner & Chatterjee 2010) the research activities are described in a conceptual framework and a set of guidelines are prescribed for conducting and evaluating good design science research. In Table 1 the seven design science research guidelines are presented, these guidelines are
followed in this research in order to validate that the research is following the design science approach.

<table>
<thead>
<tr>
<th>Guideline Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: Design as an Artifact</td>
<td>Design science research must produce a viable artifact in the form of a construct, a model, a method, or an instantiation.</td>
</tr>
<tr>
<td>2: Problem relevance</td>
<td>The objective of design science research is to develop technology-based solutions to important and relevant business problems.</td>
</tr>
<tr>
<td>3: Design evaluation</td>
<td>The utility, quality, and efficacy of a design artifact must be rigorously demonstrated via well-executed evaluation methods.</td>
</tr>
<tr>
<td>4: Research contributions</td>
<td>Effective design science research must provide clear and verifiable contributions in the areas of the design artifact, design foundations, and/or design methodologies.</td>
</tr>
<tr>
<td>5: Research rigor</td>
<td>Design science research relies upon the application of rigorous methods in both the construction and evaluation of the design artifact.</td>
</tr>
<tr>
<td>6: Design as a search process</td>
<td>The search for an effective artifact requires utilizing available means to reach desired ends while satisfying laws in the problem environment.</td>
</tr>
<tr>
<td>7: Communication of research</td>
<td>Design science research must be presented effectively to both technology-oriented and management-oriented audiences.</td>
</tr>
</tbody>
</table>

Table 1. Design Science Research Guidelines, source (Hevner & Chatterjee 2010).

3. A REFERENCE ARCHITECTURE FOR WORKFLOW LEARNING MANAGEMENT SYSTEMS

A Reference Architecture is the result of applying a Reference Model together with Architectural Patterns (see Fig. 1) in order to solve the problem defined in the Reference Model (Bass et al. 2012).


A Reference Architecture is the template of how a software architecture for a specific domain should be designed, describes the software elements that solve a problem in a specific domain just as was stated in the Reference Model (Bass et al. 2012), a Reference Architecture can be designed with different levels of abstraction, but should be generic enough in order to be used for the design of different software architectures in the same family of software systems (Clements et al. 2010). Most of the times a Reference Architecture is related to the best practices that have been useful for solving architectural problems in an specific field (Greethorst & Proper 2011).

The Reference Model applied in this research is the WiMS Reference Model proposed by the Workflow Management Coalition (Hollingsworth 1995). The model (see Fig. 2) is the general description of the architecture for a WiMS, it describes the functionalities of the software components that are part of a WiMS.
The Architecture Patterns applied for the creation of the Reference architecture were related to mobile architectures (Friese 2012; Flautero 2012), cloud computing (Liu et al. 2011; Mikkonen & Taivalsaari 2013; Dinh & Lee 2011) and collaboration (Hansen et al. 2011; Neyem et al. 2012; Herskovic et al. 2009), all these patterns are based on the most common and well known architecture styles explained by Microsoft Patterns & Practices (2009).

4. A SOFTWARE ARCHITECTURE FOR A WORKFLOW LEARNING MANAGEMENT SYSTEM

The implementation of the Reference Architecture for a specific kind of Software Architecture system in a specific context is known as Concrete Architecture (Stoitsev 2012; Angelov et al. 2012) (see Fig. 1). The Software Architecture in this research is for the design and development of a WILMS with mobile, cloud and collaborative functionalities (see Fig. 3). The Reference Architecture has been adapted in order to provide the best WILMS solution, taking advantage of the cloud, mobile and collaborative functionalities for the learning environment. The yellow functionalities of the architecture in the Figure 3 are implemented all together as a mobile application with any of the mobile architecture patterns (native, hybrid, web, etc.) proposed in (Friese 2012). Next we describe each of the software modules of the Software Architecture and their functionalities.

4.1. Workflow Enactment Service

The Workflow Enactment Service is the central part of the WIMS (Aalst & Hee 2004; Hollingsworth 1995), this software module through the Workflow Engine functionality is in charge of the creation, management and execution of the Workflow instances. In this module is also the Data Base Management System (DBMS) in order to store information related to each of the Workflows instances managed by the Workflow Engine. It also contains the API’s for the correct communication and operation with the other software modules of the system. This software module was implemented with cloud computing patterns through the private implementation model (Subashini & Kavitha 2011; Mell & Grance 2011; Esayas 2012; Aceto et al. 2013) together with the Software as Servide model (Xu 2012; Harman et al. 2013; Park & Ryoo 2013; Fernando et al. 2013). The API’s were implemented as RESTful services (Arroqui et al. 2012; Jamal & Deters 2011; Mohamed & Wijesekera 2012).
4.2. Notification Tools

This module contains the notification tools for keeping the user updated about the state of the Workflow Learning instances, tasks and activities where the user is collaborating. Push notifications and Twitter were used as the principal notification tools. This module was implemented with cloud computing patterns using communications as a service (Khan et al. 2013) and collaboration patterns implementing social media as communication and notification services (Hansen et al. 2011; Kim et al. 2010), the notification tools are necessary for the collaboration and interaction between users and are an essential functionality for communication and coordination in a mobile collaborative architecture (Neyem et al. 2012).

4.3. Collaboration and Storage Applications in the Cloud

This module is the repository of the cloud collaborative applications and the storage management applications in the cloud. Google Drive, Dropbox and Evernote are the mobile cloud applications for supporting the collaboration in the WILMS. This module was implemented with cloud computing patterns using software as a service and storage as a service (Harman et al. 2013; Park & Ryoo 2013; Fernando et al. 2013), collaboration patterns were implemented according to the layered architecture for mobile collaboration applications proposed by (Neyem et al. 2012) where mobile collaborative applications are the principal support for the collaborative interaction between the users in order to reach the goals of the group. The collaboration and notification module fulfill the requirements proposed by (Herskovic et al. 2011) for the WILMS to be a mobile collaboration system.

4.4. Mobile Services Tool

This module contains the API’s for accessing the mobile services available in the cloud, in that way the mobile workflow learning application can extend its functionalities. The functionalities that social media services offer are key for the collaboration in the WILMS. The Facebook Connect service is implemented as an authentication method, in that way the user can Login using its credentials from Facebook and collaborate with other users in the WILMS with the same user identity that has on Facebook. The open social service is also implemented and allows Facebook users to access its content and contacts inside a third party application, in this way the user avoids the necessity of
having to make new connections again inside the third party application with the people that already is collaborating with him in Facebook. In the WfLMS the user can use their contacts from Facebook as human resources in order to assign them roles and tasks inside the learning activities of the Workflows. This module was implemented with collaboration patterns using social media as collaboration services (Kim et al. 2010) and Single Sign On (Riesner et al. 2013).

4.5. Process Definition Tool

This module provides the functionality for creating the process definition of the Workflows, a process definition contains all the information related to the Workflow in a format that can be interpreted by the Enactment Service. With this tool the user creates the Workflows and configure the conditions for when to start and finish the Workflow, creates the activities and tasks that the users must execute, selects the kind of user that must execute the tasks, adds relevant information related to the Workflow, models the logical order in which the tasks are going to be executed. The user has a resource classification tool for the selection of the users that are going to be part of the Workflow, this tool makes use of the mobile services module.

4.6. Work List Handler Tool

This module provides the functionality for the administration and management of the Workflows, also for the presentation of the work items that the user has to execute in each one of the Workflows. In the task module the user can select work items in order to load the detailed information of the Workflow and the current activity that is being executed. From the task module the user can launch the mobile collaborative applications and the cloud content management system tool.

4.7. Cloud Content Management System Tool

This module contains the API’s for connecting with the storage services in the cloud (Dropbox, Drive and Evernote), in order to provide the functionality of a content management system (CMS) in the WfLMS, all the content is stored in the cloud and the user utilize the CMS functionality for associating the content stored in the cloud with the tasks and activities of the Workflows. The functionalities of the cloud storage services like session management, data synchronization, access rights and sharing of the content extends the functionalities of the CMS module. This module was implemented with cloud computing patterns, using software and storage as a service (Xu 2012; Harman et al. 2013; Park & Ryoo 2013; Fernando et al. 2013), and collaboration patterns for the coordination of the mobile users when accessing the content in the cloud (Neyem et al. 2012).

5. IMPLEMENTATION OF THE SOFTWARE ARCHITECTURE

The implementation of the software architecture was the development of a native application for the iOS platform. The application was published in the Apple Store with the name of WOLF (Work Linear Flow). In the figure 4 the Work List Handler Tool and a Workflow instance is displayed (task module). In the WOLF application, the users can create projects and associate Workflows to them, a Twitter hashtag is created for each project so in that way the users can post tweets about the learning activities that are related to the Workflows, the project and communicate with other users for collaboration. The users can open Twitter inside WOLF in order to see the list of tweets related to the hashtags of the projects.

The instantiation of a Workflow contains all the information the user needs to know about it, like the project it belongs to, the starting and the finishing date, the owner, the documents that are shared, the number of steps (activities) and comments the users make. The steps are executed in a sequential flow, all the steps related to the Workflow are displayed in an item list in the order they should be executed, each one of the steps can be evaluated, in that way the user can reject the activity (deny button) that was executed in the previous step, if this is the case, the flow goes back one step and the user in charge of the rejected activity is notified with a push notification. When the user has finished the activity that was in charge of, he has to forward (validate button) the Workflow to the next step, then the user in charge of the next step is notified with a push notification.
6. A DELPHI STUDY FOR THE EVALUATION OF THE WfLMS FUNCTIONALITIES

In order to evaluate if the mobile, cloud and collaborative pattern solutions that have been implemented in the software architecture, enhance the functionalities of a WILMS, a Delphi study was executed. In a Delphi study a group of experts with similar training and general understanding in a field of interest are questioned about their opinion in a series of surveys with controlled feedback (Meijering et al. 2013). The achievement of consensus in the area of interest which is uncertain or lack of empirical evidence is the principal advantage of a Delphi (McGinn et al. 2012).

The guidelines for how to design a Delphi study proposed by (Worrell et al. 2013) were followed. A Delphi study of three rounds with a group of 12 experts was implemented. The experts had a background experience with WILMS, WfMS, mobile collaboration, cloud computing and became also expert users of the WOLF application in order to participate in the Delphi study. 14 factors were identified in the pattern solutions that were implemented for the design of the Software Architecture (see table 2). A questionnaire was designed with statements of how each one of the factors enhanced the functionalities of a WILMS. Using a five point Likert scale (1 indicated strong disagreement and 5, strong agreement) the experts ranked each one of the factors and were also invited to leave comments for each question.

The entire Delphi study was carried out with the WOLF mobile application. For each questionnaire a Google Drive Form was created, each form was linked to a Spreadsheet in order to save the responses of all the experts. Within the WOLF application, for each one of the questionnaires a sequential Workflow with 3 steps (activities) was created by the moderator of the Delphi study (see figure 4), only two participants were part of each Workflow, the moderator and one expert. In the first step of the Workflow the moderator attached the questionnaire to the Workflow using the CMS functionality, by forwarding the Workflow to the next step (pushing the validate button), the expert received a push notification that the questionnaire was open, in the second step of the Workflow the
expert opened and responded the questionnaire and forwarded the Workflow to the next step, the moderator received a push notification that the user had responded the questionnaire, in the third step, if everything was correct the moderator closed the Workflow validating the last task, if something was wrong with the questionnaire the moderator could go back one step (pushing the deny button), then the expert received a push notification indicating that the task that was assigned to him (answer the questionnaire) was denied and had to be done again.

7. RESULTS AND DISCUSSION

The statistical analysis method was based on (McGinn et al. 2012) where consensus on a factor was considered strong when 75% or more of the experts reached an agreement, moderate consensus when the agreement was between 60% and 74% of the experts, and the absence of consensus when less than 60% of the experts agree. To calculate the level of consensus McGinn et al. calculated percentile scores (10th and 25th) where the 10th percentile indicated the lowest number on the Likert scale upon which at least 90% of the participants agreed and 25th percentile scores indicated the lowest number on the Likert scale upon which at least 75% participants agreed. The statistical dispersion function known as interquartile range was used for measuring the strength of the consensus, a result close to 0 indicates a strong group consensus and close to 2 indicates the responses of the experts are disperse.

<table>
<thead>
<tr>
<th>Factor</th>
<th>Consensus (agreement in %)</th>
<th>5-point Likert score</th>
<th>Percentile score 10th</th>
<th>25th</th>
<th>Inter-quartile range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collaborative</td>
<td>&gt; 75%</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Collaboration</td>
<td>&gt; 75%</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Communication</td>
<td>&gt; 75%</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Coordination</td>
<td>&gt; 75%</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Open social</td>
<td>&lt; 60%</td>
<td>-</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Single Sign On</td>
<td>&gt; 75%</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Cloud</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Availability</td>
<td>&gt; 75%</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Elasticity</td>
<td>&gt; 60%</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Cloud Services</td>
<td>&gt; 75%</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Consistency</td>
<td>&gt; 75%</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Synchronization</td>
<td>&gt; 75%</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Mobile</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mobility</td>
<td>&gt; 75%</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Ubiquity</td>
<td>&gt; 75%</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Communication</td>
<td>&gt; 75%</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Mobile Services</td>
<td>&gt; 75%</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 2: Delphi study factors and consensus agreement.

The results in Table 2 indicates that the group of experts reached strong consensus for most of the factors except for the Elasticity Factor (reached moderate consensus) and the Open Social Factor (absence of consensus). For example the consensus agreement for the coordination factor reached more than 75% (strong consensus), according with the 10th percentile scores 90% of the experts responded 4 or 5 on the 5 point Likert scale for this factor and according to the 25th the 75% of the experts responded 5, the interquartile range on this factor was 0 indicating that the strength of the consensus was strong. Based on the results in Table 2, the mobile, cloud and most of the
collaborative architectural pattern solutions implemented in the software architecture enhance the functionalities of a WfLMS (See Fig. 3).

8. CONCLUSIONS AND FUTURE WORK

This research has been done following the design science research methodology, as a result two artifacts were designed: a software reference architecture for the design and development of WfLMS and a mobile application as a result of the implementation of the software architecture. This research was based on rigorous methods for the construction and design phase of the software architecture and makes the use of architectural pattern solutions that are well established and recognized in their own fields of research. This research also provides the empirical experience of carrying a Delphi study with a cloud collaborative application (Google Drive) and WfLMS as part of the research methodology evaluation phase. Future evaluation for the WfLMS iOS application has to be done, other implementations of the reference architecture with the architectural pattern solutions in different mobile architectures and platforms have to be done in order to evaluate the functionalities of the software architecture and support the results found in this research.

9. BIBLIOGRAPHY


