
Caballé, S. [et al.] (2014) Building a software service for mobile devices to enhance awareness in web collaboration. *2014 International Conference on Intelligent Networking and Collaborative Systems: IEEE INCoS 2014: 10–12 September 2014, University of Salerno, Salerno, Italy: proceedings.* [S.l.]: IEEE, 2014. Pp. 369-376 Doi: http://dx.doi.org/10.1109/INCoS.2014.11.

Caballé, S. [et al.] (2014) Building a software service for mobile devices to enhance awareness in web collaboration. *2014 International Conference on Intelligent Networking and Collaborative Systems: IEEE INCoS 2014: 10–12 September 2014, University of Salerno, Salerno, Italy: proceedings.* [S.l.]: IEEE, 2014. Pp. 369-376 Doi: http://dx.doi.org/10.1109/INCoS.2014.11.

Building a Software Service for Mobile Devices to Enhance Awareness in Web Collaboration

Santi Caballé[1], Fatos Xhafa[1], Jordi Raya[1], Leonard Barolli[2]
Department of IT, Multimedia, and Telecommunication
Open University of Catalonia
Barcelona, Spain
{scaballe, fxhafa, jraya1}@uoc.edu
[2] Fukuoka Institute of Technology, Department of Information and Communication Engineering
Fukuoka, Japan
barolli@fit.ac.jp

***Abstract*— In this paper, we propose the construction of an effective event notification software service for mobile devices to provide anytime anywhere awareness to online work teams during Web collaboration. The software service is first designed to achieve the required flexibility to be used in different collaborative work situations, from professional work to informal communities of practice. Then, the building of our event-oriented service for mobile clients is reported from all the stages of our software engineering methodology and it is prototyped for evaluation purposes. We believe the outcomes of our approach will be very beneficial for achieving more productive and quality Web collaboration practices. The ultimate aim of this research is to provide software designers of Web collaboration tools and applications with general guidelines as for how to face common issues and challenges when incorporating a complete event management and notification system in their developments.**

Keywords- Web collaboration; software engineering methodology; mobile service; awareness; event management; collaborative work and learning.

I. INTRODUCTION AND BACKGROUND

Web collaboration provides virtual work teams with the capability to collaborate via the Internet [1], [2], [3]. It brings together many types of communications, such as on-line meetings and seminars, which facilitate organizational teamwork and workflow, and is supported by a large range of Web-based tools, from wikis for joint production of documents to chats and bulletin boards for synchronous/asynchronous discussions as well as document repositories, and monitoring and evaluation tools, among many others [3],[4].

Web collaboration is to assist in the three main areas of any collaboration, namely coordination, cooperation, communication, with communication being the base for reaching coordination and cooperation [5],[6]. Moreover, awareness is essential for any of these three areas due to its positive impact in on-line collaborative work in terms of group motivation, interaction, or problem-solving abilities [6],[7]. Indeed, the provision of effective knowledge extracted from the information collected in on-line collaborative environments is essential for any form of collaboration [8],[9]. It allows implicit coordination of collaborative work, opportunities for informal, spontaneous communication in both synchronous and asynchronous modes. Users should be aware of the current activity in the group (e.g. the contribution of other members, their location and availability, the users working on a shared document at the same time, and so on) and should know what other co-participants are doing in real time (e.g. during a multi-user editor session, who is editing and what is being shown). In an asynchronous context, users must know the activities performed by receiving deferred information of who, when, how and where others' interactions have been performed, and also why these interactions have been performed, which implies receiving complex knowledge of the interaction history [10]. As a result, participants in Web-based collaborative experience may greatly enhance their abilities by increasing their knowledge about others in terms of cognitive processes and skills of the students and the group as a whole in solving problems, individual and group effectiveness regarding participation and interaction behavior, social support and help and so on [6],[7],[11].

From this view, the success of Web collaboration tools and applications depends to a large extent on the capability of such applications to embed information and knowledge of group activity and use it to achieve a more effective group monitoring [6] as well as constantly provide group members with as much awareness and feedback as possible. Awareness [12],[13] refers to the knowledge provided to participants about both what other participants are doing at the same time and what they did in the past, whereas feedback [14] goes one step further than awareness by providing exhaustive and elaborated information and knowledge of what is going on in the group over a long period of time. Finally, the persistent storage of the knowledge extracted as group memory [15] is essential for all participants since it allows them to access not only the latest documents and data, which are commonly stored for later retrieval, but also the context in which they were

created. For the purpose of our research, we focus and center our work in the provision of awareness to Web collaboration as primitive data source for immediate presentation to participants of what is happening during the collaboration.

Finally, current powerful mobile technologies, such as latest generation of smart phones and tables, contribute to both a greater user interaction and a more immediate event notification on what is happening during the collaborative work and learning [16], [17]. In this context, standard mobile requirements are to be extended to support costly server-side functionalities, thus bearing new issues and challenges, such as memory and processing limitations of mobile devices as well as devices with a great variety of hardware and software. Hence, in this research we also explore how to deal with these limitations when developing an event-oriented and notification service for mobile devices to support effective Web collaboration.

To sum up, current attempts fail in providing appropriate support to Web collaboration from the knowledge management and mobility perspectives in order to both support mobile needs during the collaboration and provide all participants with immediate awareness about what is happening during the collaboration. In this paper, we take all the above approaches one step further in order to develop a software service for mobile devices that support anytime anywhere awareness during Web collaboration sessions and activities. To this end, the main software engineering goals when developing our mobile service are the following: (i) offer core functionalities to Web collaboration with remote access from mobile devices; (ii) provide a generic platform-independent event model to support awareness during online collaborative activities [2]; (iii) explore basic mobile requirements to provide participants with immediate awareness anytime anywhere; (iv) use of standards to interoperate in heterogeneous infrastructure and (v) use of local resources in mobile devices to support users' interactions.

This paper is structured as follows. In Section 2 we provide an extensive survey of mobile technologies to support Web collaboration as additional requirements from those discussed in the previous section. Section 3 presents a software methodology to build our mobile service for Web collaboration by the analysis of the requirements and design within a detailed software development process. Section 4 provides the prototyping and testing of our system for evaluation purposes. We conclude the paper in Section 5 by highlighting key concepts and outlining ongoing and future work.

The paper continues our previous work started in [2] with proposing a generic event model for knowledge management and notification in support for many types of Web collaboration. Based on our event model, the software service is designed to achieve the required flexibility to be used in different collaborative work situations, from professional work to informal communities of practice and social networks, thus dealing with different type of events to be collected and analyzed, and various forms to notify the participants the knowledge extracted for awareness purposes. See [2] for full information on the event model.

II. MOBILE TECHNOLOGY FOR WEB COLLABORATION

In developing Web collaboration environments that support collaborative work and learning, several issues must be taken into account in order to ensure full support to the online teams. One key issue is mobility in correspondence with the current mobility of work teams and the widespread of mobile devices and wireless technologies [17],[18]. Indeed, the proliferation of mobile phones and other handheld devices has transformed mobile collaboration from a researcher-led endeavor to an everyday activity, whereby mobile personal tools help people work and learn everywhere either formal training or informal support, collaboration and conversation [19]. As a result, by the addition of mobility and the support of mobile technologies, new paradigms, such as Mobile Computer-Supported Collaborative Learning (MCSCL), has appeared [20].

MCSCL bases the success of current and future collaborative learning applications on the capability of such applications to incorporate mobility to support the collaborative learning processes [20]. For this reason, this issue has already attracted the attention of researchers, pedagogues and developers of applications from e-learning and collaborative learning domains [17]. Current literature in collaborative learning is however rather short though intensively increasing over the last years. As a result, mobile collaborative learning is still in its infancy and many challenges are to be addressed before being fully benefited from incorporating mobility to day-to-day collaborative learning. In our work we aim to promote Web collaboration over CSCW and MCSCL by exploring the plausibility of providing system-level support and services for the provision of effective awareness to on-line work teams.

Mobility is seen by researchers as a new opportunity for work and education since it provides more chances for on-line groups to personalize their collaborative process, enhance social interactions, work and learn more effectively and more autonomously, and collaborate with other peers and facilitators at anytime and from anywhere, inside and outside the formal collaborative context. Indeed, both the capabilities of mobile devices and their wide context of use contribute to their propensity to foster collaboration. Mobile devices can easily communicate with other devices of the same or similar type, enabling learners to share data, files and messages. They can also be connected from anywhere at anytime to a shared data network, further enhancing possibilities for communication [21]. These devices are also typically used in a group setting, and so interactions and collaboration will tend to take place not just through the devices but also at and around them as well.

A great variety of challenges arise though when using mobile devices to support collaboration, ranging from technical - such as how to manage devices with very small screens and keywords, which do not facilitate easy access to text and impede input or annotation - to performance and outcomes, such as how to coordinate small groups in the workplace. And yet mobile collaboration is about supporting groups of people for formal and informal group activities in which they are willing to participate seamlessly [17].

A. *Mobile devices*

Technology is getting smaller, more personal, ubiquitous, pervasive, and powerful. Mobile devices range from the use of Personal Digital Assistants and tablet computers to context-aware devices for field trips, museums, and tourist visits [21]. This way, mobile technologies provide flexibility and ubiquity by accessing on-line resources anytime, anywhere and adapt them to users' personal features, preferences and interests, as well as pervasiveness by means of the latest wearable devices for working and learning across contexts [22].

From the literature, certain classification is made in order to organize the great variety of existing technology for work and education into personal, shared, portable and static. In this paper, we consider personal and portable devices, such as smart phones and tablets, since those shared, portable technology forces the users to move, not the device [17].

B. *Infrastructure for mobile technologies*

Modern mobile collaborative environments must provide advanced enablement for the distribution of learning activities and the provision of immediate effective awareness as well as the necessary functionalities and resources to all participants, regardless of where these participants and resources are located, and whether this location is static or dynamic. The aim of newest collaborative environments is to enable the experience in open, dynamic, large-scale and heterogeneous environments. As a result, ubiquity and pervasiveness are essential requirements to support formal and informal work and learning, and to allow all group members, from a variety of locations, to cooperate with each other by means of a large variety of technology-enhanced equipment. Therefore, non-functional requirements, such as scalability, flexibility, availability, security, interoperability are to be considered for the development and adoption of the next generation of mobile collaborative systems [21].

Powerful and pervasive technologies, such as Grid/Cloud, P2P, distributed and wireless infrastructure, enable ubiquity and pervasiveness in order to provide not only anywhere anytime learning but also full seamlessly collaboration [18]. These advanced infrastructural needs in combination with the innate business goal for lower cost and higher profit are driving key business sectors such as multimedia, engineering, e-health, gaming, m-learning, among others towards adopting new efficient and scalable solutions into their business [23], [24].

These days, it is common to find all the aforementioned technologies available in work centers, so it is necessary to manage these technologies to obtain the greatest benefits that can help in the Web collaboration process [17]. However, the use of mobile devices to support Web collaboration sets out new challenges and opens up new issues worth investigating.

Based on the above requirements and challenges presented throughout this section and conducted by these software engineering goals, next section presents a software methodological approach to build our mobile service, which can provide immediate and effective awareness, anytime and anywhere, thus yielding more quality collaborative processes whilst augmenting group performance and outcomes.

III. SOFTWARE ENGINEERING METHODOLOGY

In order to achieve this research's goals set in the previous section, we present here our Web collaboration system that includes a mobile service to support awareness-based collaboration. The whole system is reported technically in detail from all software development stages and architectural components. First, the requirements are elicited and reported, including technical, user interface and functional requirements, being the latter elicited from the main discussion in Section 1. Then, these requirements along with the non-functional requirements discussed in Section 2 as well as those elicited from our event model for Web collaboration reported in [2] are all considered for the design of our mobile knowledge-based groupware system. The design is finally implemented resulting in a system prototype that achieves the mentioned goals presented in this paper.

A. *Requirements*

In this subsection, the main requirements of our system are identified in terms of technical requirements, user interface, business model, and use cases that formally document the functional requirements elicited.

1) *Technical requirements*

In order to develop our web collaboration system, we follow the traditional client/server (C/S) paradigm usually supporting Web-based applications for many purposes. However, in order to alleviate issues typically found in C/S applications over the Web and meet relevant non-functional requirements, such as scalability, interoperability, performance and security, we include in the requirements a distributed and multi-tier architecture, based on modular components running on an application server [25] (see also previous section for further information on distributed infrastructures). In particular, we follow the well known 3-tier architecture formed by the presentation, business logis and data tiers (see Fig. 1). Each tier becomes a modular component independent from the rest of tiers in order to make the application extensible and scalable for the coming requirements and technological advances to be considered. In the particular case of the presentation tier we consider two types of client interfaces; web (standard browsers) and mobile (Android platform).

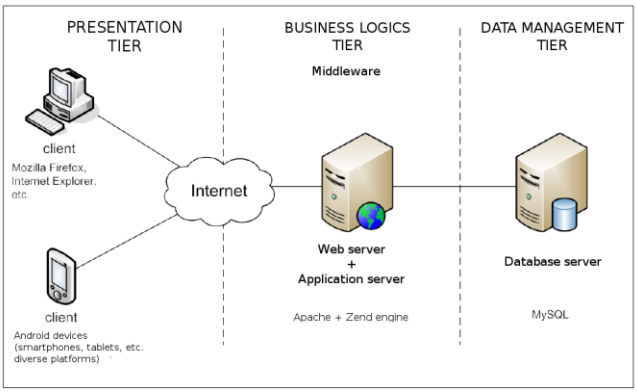

Figure 1. 3-tier architecture

In the presentation tier, Web clients, on the one hand, will use standard Web browsers to view HTML information in their desktop and laptop computers and will have access to the application server through the Internet by the HTTP protocol. By means of the Web interface, clients do not need to install any specific software and can easy operate with server-side services. On the other hand, mobile users will face a heavy client installed in their mobile devices consisting of an ad hoc application supported by Android platform. By meeting this requirement, we avoid the issues with the user interface when using a Web client in mobile devices (e.g., small screens and lack of standardization of HTML 5.0 for diverse platforms of mobile devices).

The business logics tier includes the middleware software where our actual application is found, which runs in an Apache web server with a Zend engine in charge of processing PHP code as the selected language for programming the back end [26]. Therefore, the same server side will perform both web and application server tasks. On the one hand, the application server will process PHP code from the users' requests accessing to data if necessary and returning the result as HTML files rendered by the web browsers. On the other hand, the web server will offer an entry point as a web service for mobile clients. Finally, the middleware software in the application server will simplify many common and tedious low-level tasks, such as dealing with sockets, managing stateful sessions within HTTP to identify clients and so on.

The data management tier will use a relational database supported by MySQL as a good implementation of SQL-92 and has libraries and drivers specially designed to interact with PHP [26].

2) User interface

Given that a complete and standardized graphical user interface (GUI) is out of scope of this work, the client's GUI will be as simple as possible to accomplish the required functionalities (e.g., Cascade Style Sheets templates are not considered to improve the GUI design esthetics). Therefore, our application's GUI for Web clients will be generated by the PHP engine with only those indispensable interactive features usually found in any HTML documents, such as hyperlinks, text fields and buttons. To this end, the application server will process PHP code to generate hypertext documents to be rendered in the client's browse (see Figs 7-10).

The GUI requirements for mobile clients will be designed by using Android platform's API [26], which provides a collection of generic components that offer common input controls and various layout models. Similarly to the GUI requirements for Web clients, mobile clients' GUI will be as simple as possible without any layout personalization or control detail beyond it is strictly needed for our application to work properly.

3) Business model

From the background discussed in Section 2, we extract and describe here the main business entities and relationships participating in our Web collaboration application. They also serve to set the boundaries of the system. Due to space constraints just a brief description of each entity is provided.

- User: represents any user of the application and stores the personal data of the user (e.g., user name, password, etc.)
- Group: it represents the grouping of users for a certain purpose. A User may belong to several Groups at the same time. Only administrators can create groups.
- Role: sets the relation between User and Group. For the purpose of our application we only consider "user" and "administrator" roles. The latter has more privileges than users to carry out certain tasks, such as create groups, post in special forums, and manage the system.
- Forum: represents the grouping of discussion threads, usually to address a shared topic. There are two types of forums, read-only and writable. The former forums only administrators can post whilst the latter all users can post on it.
- Post: represents the information carrier between a sender process (User) and a recipient process (User and Forum). Post implements two ways of message communication: unicast and multicast, the latter will create a separate post for each recipient. Finally, Post can be public (if sent to a forum) or private (sent to a user).
- Resource: represents any type of resource assigned to a group by a user, such as folders, files, Web links, and so on. Considering the simplicity of our application, the system will not store any physical resource but an URL link to its location in an external public file repository.
- Event: represents the events generated as a result of the actions in the system by the users, such as the management of other entities (users, groups, posts, resources, etc). An event will be identified by a code and will store its type as well as a free text field with all the information available about the event. The purpose is to simplify the generation of the event by avoiding toretrieve data from other entities, thus speeding up the notification of the event.
- Notification: relationship between Event and User, and stores whether users have been notified and when (timestamp).

4) Use cases

Use cases are part of a software engineering methodology that allow software designers to identify, clarify and organize system's requirements by capturing the interaction between stakeholders and the system, and describing the system behavior [28]. Use cases capture main functional requirements (i.e. functions with input, behavior and output) forming the observable system behavior at high level. Hence, for the purposes of our application, all clients interacting with the system, either web or mobile, will realize the same use cases.

Main requirements of our system were discussed in Sections 1 and 2, and the main entities participating in our system were presented in the previous subsection. In this subsection we elicit and describe these requirements

formally by using use cases. The following is a list of the main use cases grouped by the scope they belong to, emphasizing the knowledge management use cases for awareness purposes. Due to space constraints we report on them by a brief text-form description.

a) User and Group management use cases

These use cases capture all the behavior related to manage the system's users, which can act as a group coordinator, group member, group-entity and system administrator. They will tackle the basic user management functions in the Web collaboration environment (namely registration, deregistration, modifications, elimination, joining a group, and forum group members) as well as the user profile management, which captures the user and group models within the collaboration.

b) User authentication and authorization use cases

These use cases capture the measures and rules decided upon to resolve user login and privilege issues and so protect the system from both unknown users and the intentional or accidental ill use of its resources.

c) Collaboration use cases

These use cases mainly capture the essential functionality (i.e. create, update, list and eliminate) to manage forums, thread and messages, including public and private posts, in order to provide asynchronous communication support to the Web collaboration users and groups. This includes the management of the resources in form of files, folders, links, etc, shared during the collaboration, including download, block and internal organization of the resources.

d) Knowledge management use cases

These use cases manage all the user events in order to handle the data of user interaction as crucial information for the extraction of the essential knowledge during the collaboration as well as monitor user behavior and control system resources. They also capture the presentation of this knowledge managed with the aim of providing participants with effective awareness of what is going on in the group, including who, when (time stamp), how and where shared resources have been created, changed or read by other users. These use cases will be purposely implemented in mobile clients to offer immediate awareness by the mobile devices.

B. Design

From the analysis of the previous requirements, we introduce here the design of our application as a technological solution based on the Object-Oriented (OO) paradigm to meet the requirements. The OO paradigm results in modular software offering many advantages, such as easy for reuse and maintain, flexibility, extensibility and scalability [29], thus meeting the non-functional requirements of our application. PHP, as the selected programming language of our application, supports the OO paradigm (see [26] and subsection 3.A.1).

In this section, we first show the design of the application domain. Then, we describe the newly software objects and artifacts generated from the design that will form the implementation basis of our application. Finally, due to its crucial role in the system and in our research as whole, we develop in detail the Event entity, which is modeled to be platform-independent and adaptable to many collaborative settings and purposes.

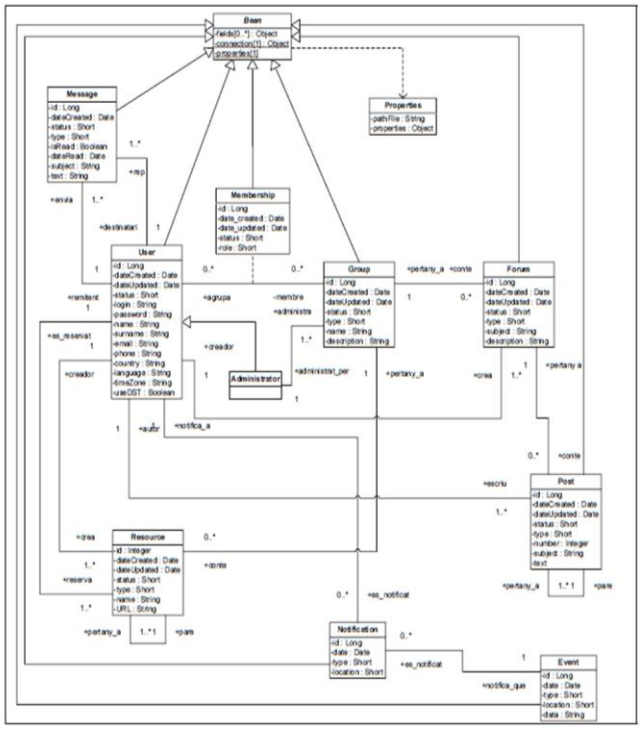

Figure 2. Class diagram modeling the application domain

1) Application domain

We present here a class diagram modeled with UML [30] with the software entities and their relationships participating in our application domain (i.e. data persistence) (see Fig. 2). For graphical clarity, only the main relationships are shown and methods are hidden. The Event class is explained in more detail in the next section.

2) Software objects and design decisions

As shown in Fig. 2, all domain classes inherit from the Bean abstract class (see Fig. 3). The attributes of this class will be managed by an associative array of PHP that will internally store the attribute values and will be retrieved by the name, thus having similar behavior to a hash table. The aim is to provide flexibility, easy to use, and better adaptation to further updates of the design.

Bean
-fields : Object -connection : Object -properties : Properties
+getField(entrada fieldName : String) : Object +setField(entrada fieldName : String, entrada fieldValue : Object) +getFieldSQLNumber(entrada fieldName : String) : String +getFieldSQLString(entrada fieldName : String) : String +getFieldSQLBoolean(entrada fieldName : String) : Short +getFieldSQLDate(entrada fieldName : String) : String +getFieldSQLDateOrNow(entrada fieldName : String) : String +getFieldUTCDate(entrada fieldName : String) : Date +getFieldLocalDate(entrada fieldName : String, entrada timeZone : String, entrada useDST : Boolean) : Date +getFieldUTCDateStr(entrada fieldName : String) : String +getFieldUTCDateWS(entrada fieldName : String) : String

Figure 3. Bean class

All domain classes inherited from the Bean class offer the essential functionality to manage the data (i.e. CRUD operations), which is realized by SQL sentences and the mysql API found in any PHP installation [26]. Due to the scarce business logics found in these operations, we did not consider the use of design patterns for data access and transmission, such as data Data Transfer Object or Value Object [25], thus also reducing the number of classes and the design complexity. Instead, the access to data is implemented in the own domain class. However, each domain class in our application has the method *toArray()* to convert attribute names and values to an associative array, and then the kSOAP library for Android is used [31] to transmit the data by a web service. This implementation could be considered as an adaptation of the DTO design pattern. Finally, each of the CRUD operations will automatically generate the corresponding Event object, without specific involvement of controller components, except for those complex events that need further data from other sources.

The user navigation scheme of our application did not follow the typical Model-View-Controller (MVC) pattern [25] due mainly to the known implementation problems caused by PHP when dealing with GET and POST methods (i.e. user data is transmitted by the URL). Despite there are some solutions to this issue, we consider that the complexities involved (e.g. use of session variables, etc.) break our main principle of simplicity (see subsection 3.A).

Finally, our mobile service is designed as a web service in the server side for the communication with the mobile client. The main purpose of this service is the management of events to efficiently notify awareness to mobile clients. The service retrieves from the server a set of events classified by user, group and date, sorted by time. The events of those users without group are also notified and it is the client's responsibility to manage and present them in an appropriate way. The web service is implemented by NuSOAP library to convert PHP associative arrays into SOAP structures [26] as well as match the web service's methods and the corresponding PHP operations to obtain and transmit the set of Event objects to be presented to the mobile clients.

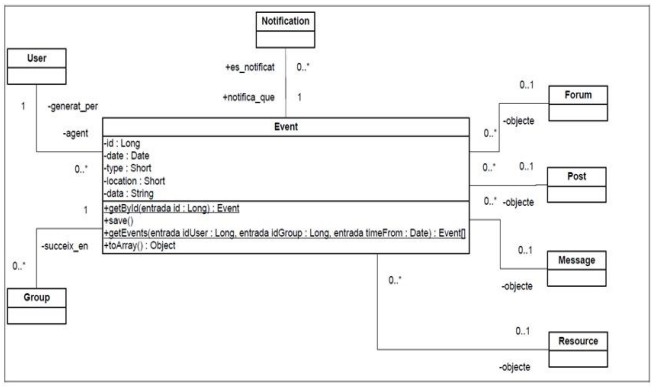

Figure 4. Event class diagram

3) Event model

Given the importance in this study of the knowledge management and provision of awareness approach to support Web collaboration, we summarize in this subsection our event model as one of the goals of our work. The model is explained in detail in [2]. This model will become the basis to design the event management system of our application, which will produce the Event software entities.

A first approach of our event model is shown in Fig. 4 with the general representation of the Event entity and the relationships with the involved domain entities, as follows (refer to [2] for a full description and diagrammatic details:

- User: the user who triggers the event. In case of system events and events without a defined user, they can take a generic user.
- Group: the group where the events take place. Most events happen in a group except for those events related to the system, user management and private messages.
- Notification: this entity keeps track of the event notification to users. Not all events are to be communicated. Hence in certain cases the event notification cannot happen while in other cases the notification is just transmitted to the client but it is not stored in the server.
- Other entities (Forum, Post, Message and Resource): the Event relationship is created where necessary. For instance, Forum events will be related to the specific forum but not to the rest of forums.

IV. EVALUATION

For evaluation purposes we implemented and technically tested the above described design of our system that supports Web collaboration with the focus on the event management capabilities for mobile clients. In this section, we show first the prototype we developed as a result of the implementation of the design presented in the previous section. Second, we discuss on the results of testing the prototype and present the most relevant issues appearing during the evaluation.

A. Prototyping

The implementation of the design involved the technologies already presented in previous sections (see Section 3.1.1), namely PHP for programming the business logics tier, MySQL as a DBMS supporting the data management tier and dynamic HTML and Android to implement the two clients forming the presentation tier of our application [26],[27]. Next, we show several snapshots of the prototype (see Figs. 5-6, text is in Catalan language).

Finally, the event notification to mobile clients was the most important aspect when developing the prototype. It was implemented in the mobile client only (see Fig. 8). After testing it, the application proved to transmit and render the events promptly in the Android client. The event information presented for awareness was exhaustive and friendly by showing clear messages about who initiated the action, when and how.

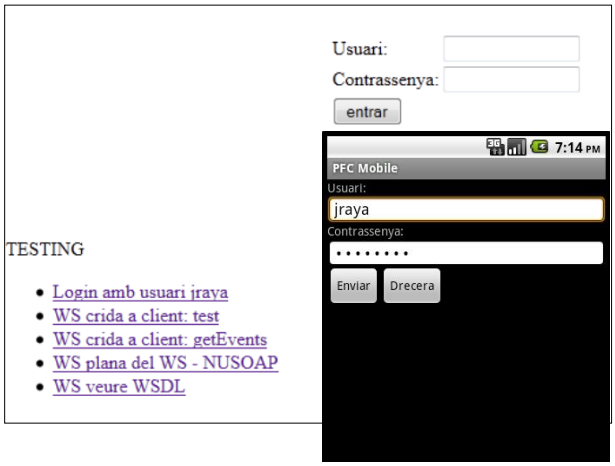

Figure 5. User login screen: back (a) login on the Web client; front (b) login on the mobile client

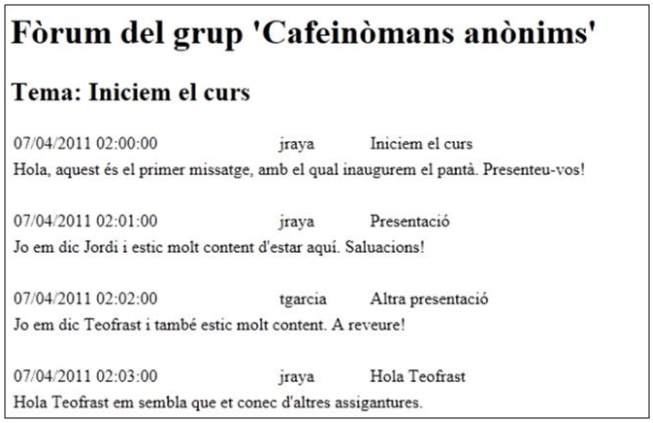

Figure 6. Web collaborative activity within a group

Figure 7. Event notification on the mobile client

B. Discussion

In this subsection we report on the main issues appeared during the development and testing of our prototype. They intend to serve as guidelines for next iteration of this work and for other similar developments.

In early stages of this development (see Section 3.A), we decided to support the communication between the back-end and mobile clients by a web service built on top of the SOAP protocol, which is a W3C standard widely accepted by the community to support machine-machine communication in the context of heterogeneous systems. However, SOAP protocol is not well supported by our implementation language PHP, which does not include SOAP in its default installation and may not be available in an external host. In the Android case, the situation is also problematic as no direct support is available to web services and SOAP. A solution to this issue was to provide SOAP support in Android by using NuSOAP for PHP i kSOAP2 for Android (see Section 3.A.1 and [27]). Despite these libraries do not offer complete implementation of the SOAP standard (e.g. they do not support SOAP time data types and they must be simulated with text string), they were enough for our development purposes.

Time-related values, such as creation and update dates, were stored in the MySQL database with time as data type. However, MySQL does not support time with finer precision of seconds. For milliseconds precision it is needed complex conversion of data [26]. However, second precision suffices above all for the event notification (i.e., awareness), which is the most important requirement of our application. Moreover, considering that the mobile clients of our application can be potentially found anywhere in the world, it was needed to manage the correct time stamp of event notification to the collaboration participants. To this end, all dates were implemented following the Universal Time Coordinated (UTC) convention supported by the UTC_TIMESTAMP function of MySQL. Then, dates are to be converted into the local time of the geographical area set up by the user. These conversions are essential for showing the correct timestamp of the event information notified to the participants anywhere they are located.

V. CONCLUSIONS AND FURTHER WORK

This paper presents a development work aimed at providing Web collaboration participants with effective and immediate event notification anytime anywhere by a specific mobile service.

We can conclude this work, first, by highlighting the fundamental requirement of having an effective event notification system to greatly enhance and improve any type of Web collaboration practices, which may become a critical feature for the success of on-line work teams (e.g. when time factor is involved during the collaboration with strict deadlines and sequential distribution of individual work). Second, latest mobile technology is able to contribute to a greater and timely data workflow. This is especially relevant when notifying the new events from others' actions so that participants equipped with mobile devices can be promptly

aware of what is happening in their workspaces and act accordingly. Third, in order for the event management and notification system to fit any Web collaborative practice, a general, flexible and extensible independent event model is required to be adapted to different scenarios and future awareness needs. Last, from the technological perspective, the system must be able to interact and transmit data between the back-end and front-end with heterogeneous client implementations by using standards and interoperable technologies, such as SOAP and web services.

In this paper, we took the above approaches one step further by developing and testing a prototype to support Web collaboration that faced the above goals and challenges, with strong emphasis in the provision of effective awareness to the participants. We believe that our simple prototype and in particular the outcomes of our event management approach give new opportunities for more effective and productive on-line collaborative work and learning.

Ongoing work includes next iterations of our prototype development and further evaluation of the event-oriented approach in a massive scale in both real work and learning contexts. Experimentation and validation activities will be conducted in several on-line workspaces with intensive group activity in order to greatly improve and enhance the collaborative processes and practices.

REFERENCES

[1] Bentley, R., Hosrtmann, T., Trevor, J. The World Wide Web as enabling technology for CSCW: The case of BSCW. Journal of Computer Supported Cooperative Work, Special issue on Groupware and the World Wide Web, 6(2/3). 111-134. Norwell, MA, USA: Kluwer Academic Publishers Norwell, MA, USA, 1997

[2] Caballé, S., Xhafa, F., Barolli, L. Towards a Platform-independent Event Management Model for Web Collaboration, 6th International Conference on Intelligent Networking and Collaborative Systems. Submitted, 2014.

[3] Ebersbach, A., Glaser, M., Heigl, R., Warta, A. Wiki: Web Collaboration. Springer, 2008.

[4] Caballé, S., Xhafa, F. CLPL: Providing Software Infrastructure for the Systematic and Effective Construction of Complex Collaborative Learning Systems. Journal of Systems and Software (JSS). Vol. 83, No. 11, pp. 2083-2097. Elsevier, 2010.

[5] Ochoa, S., Guerrero, L.A., Fuller, D. and Herrera, O. Designing the Communications Infrastructure of Groupware Systems. In Haake and J. Pino Eds., Groupware: Design, Implementation and Use. LNCS, 2440, 114-133. Springer, 2002.

[6] Caballé, S. (2011). Computational Intelligence for On-line Collaborative Learning: A Computational Model for the Construction of Knowledge-based Collaborative Learning Distributed Applications. Saarbrücken, Germany: Lambert Academic Publishing, 2011.

[7] Daradoumis, T., Martínez, A., and Xhafa, F. A Layered Framework for Evaluating Online Collaborative Learning Interactions. International Journal of Human-Computer Studies, Special Issue on Theoretical and Empirical Advances in Groupware Research, Academic Press: Elsevier Ltd, 2006.

[8] Ellis, C., Gibbs, S., and Rein, G. Groupware: some issues and experiences. Communications of the ACM, 34,1, 9-28. ACM, 1991.

[9] De Wever, B., Schellens, T., Valcke, M., and Van Keer, H. Content analysis schemes to analyze transcripts of online asynchronous discussion groups: A review. Computers and Education. 46(1), 6-28, Academic Press: Elsevier Ltd, 2006.

[10] Stahl, G. Group Cognition: Computer Support for Building Collaborative Knowledge. Acting with Technology Series, Cambridge, MA, USA: MIT Press, 2006.

[11] Schrire, S Knowledge building in asynchronous discussion groups: Going beyond quantitative analysis. Computers and Education. 46(1): 49-70, Academic Press: Elsevier Ltd, 2006.

[12] Dourish, P.; Bellotti, V. Awareness and coordination in shared workspaces. Proceedings of the 1992 ACM conference on Computer-supported cooperative workcc. ACM Press New York, NY, USA. pp. 107–114, 1992

[13] Gutwin, C. and Greenberg, S. A descriptive framework of workspace awareness for real-time groupware , Journal of Computer Supported Cooperative Work, 11(3/4), 411-446. Norwell, MA, USA: Kluwer Academic Publishers Norwell, MA, USA, 2002.

[14] Zumbach, J., Schönemann, J., and Reimann, P. Analyzing and Supporting Collaboration in Cooperative Computer-Mediated Communication. In T. Koschmann, D. Suthers, and T. W. Chan (Eds.), Computer Supported Collaborative Learning 2005: The Next 10 Years! (pp. 758-767). Mahwah, NJ: Lawrence Erlbaum, 2005.

[15] Conklin, E. J. Capturing Organizational Memory. D. Coleman (Ed.), Proceedings of GroupWare '92 (pp. 133-137). San Mateo, CA: Morgan Kaufmann, 1992.

[16] Sharples, M. Learning as conversation: Transforming education in the mobile age, Proceedings of Conference on Seeing, Understanding, Learning in the Mobile Age, 2005.

[17] Caballé, S. Xhafa, F., Barolli, L. Using mobile devices to support online collaborative learning. Mobile Information Systems, 6(1), 27-47. IOS Press, 2010.

[18] Cook, J., Pachler, N. & Bradley, C. Bridging the Gap? Mobile Phones at the Interface between Informal and Formal Learning. Journal of the Research Center for Educational Technology 4(1), 3-18, 2008.

[19] Kukulska-Hulme, A., Traxler, J.. Cognitive, Ergonomic and Affective Aspects of PDA Use for Learning, in Proceedings of MLearn, 2002.

[20] Zurita, G. and Nussbaum, M. Computer supported collaborative learning using wirelessly interconnected handheld computers. Computer & Education. 42(3), 289-314, 2004.

[21] Caballé, S., Xhafa, F., Daradoumis, Th., Juan, A.A. Architectures for Distributed and Complex M-Learning Systems: Applying Intelligent Technologies. In: IGI Global ed., Hershey, PA, USA, 2009.

[22] Dourish, P. What we talk about when we talk about context, Personal and Ubiquitous Computing 8, 19–30, 2004.

[23] Foster, I. and Kesselman, C. The Grid: Blueprint for a Future Computing Infrastructure. San Francisco, CA: Morgan Kaufmann, 1998.

[24] Litke, A., Skoutas, D. and Varvarigou, T. Mobile Grid Computing: Changes and Challenges of Resource Management in a Mobile Grid Environment. Access to Knowledge through Grid in a Mobile World, PAKM 2004 Conference, 2004.

[25] Alur, D., Crupi, J., Malks, D. Core J2EE Patterns: Best Practices and Design Strategies, Prentice Hall, 2003.

[26] Welling, L., Thomson, L. PHP and MySQL Web Development (Developer's Library). Sams, 2004.

[27] Meier, R. Professional Android 2 application development (Wrox programmer to programmer) Indianapolis: Wiley, 2010.

[28] Cockburn, A. Writing Effective Use Cases Addison-Wesle, 2001.

[29] Meyer, B. Construcción de Software Orientado a Objetos. Prentice Hall, 2002.

[30] Fowler, M. UML distilled: A brief guide to the standard object modeling language Addison-Wesley, 2004.

[31] Juntao-Yuan, M. Enterprise J2ME: Developing Mobile Java Applications. Upper Saddle River (New Jersey): Prentice Hall, 2003.