A Tool to Manage Educational Activities on a University Campus

Antonio Sarasa-Cabezuelo[1(✉)] and Santi Caballé[2]

[1] Universidad Complutense de Madrid, Madrid, Spain
asarasa@ucm.es
[2] Universitat Oberta de Catalunya, Barcelona, Spain
scaballe@uoc.edu

Abstract. A very common type of extracurricular activities in universities are seminars and workshops. These types of events are used by instructors to introduce advanced content of the subjects they teach, to carry out practical applications, to invite speakers or experts in the field who speak about leading research topics, etc. The development of these activities requires some tool that facilitates its organization and management. Thus, it is necessary to manage relevant aspects for this purpose, such as the control of the participants in the events, the carrying out of satisfaction surveys, or the sending of notifications about unexpected changes in the event (time change, classroom change, etc.). This article presents a tool that has been developed with the aim of managing this type of events in an agile and simple way for both participating instructors and students.

1 Introduction

Workshops and seminars are widespread extracurricular activities at the university level. Normally, instructors use this type of activity to introduce advanced content [1], to hold practical sessions on some topic related to general content, to invite experts in the field to give talks on their lines of research, etc. In general, these activities are held outside official class hours, attendance is usually free, and they are intended to complement the basic training of students. To organize them, the instructor informs the students of the event, indicating its place, date and time. Another element related to this type of events is that some feedback is expected from the students to know if the training received has been useful to them as well as other data about the experience with the aim of improving or making changes in future activities. In addition, in certain occasions, the participation of the students in this type of activities is rewarded by the instructor with an increase of the final grade of the course directly or indirectly after carrying out a work or exercise on the treated contents. Therefore, the instructor requires to organize these activities by a communication mechanism with the students that allows him/her to inform when the event will take place, carry out satisfaction surveys about the event or communicate changes that take place in the realization of the event. In addition, the communication means used should be simple enough to be used by both students and instructors and also easily accessible.

L. Barolli et al. (Eds.): 3PGCIC 2020, LNNS 158, pp. 170–178, 2021.
https://doi.org/10.1007/978-3-030-61105-7_17

With the aim of offering a solution that would be suited to the above-described needs, a particular system was developed for managing extracurricular events that would offer the most common services needed to organize these activities. In addition, the design took into account the requirement that the organization process should be agile and that the use of the system should be as simple and intuitive as possible for both instructors and students. In this sense, the system developed consists of a web application and an Android app. The first application is oriented towards the instructor and offers him/her all the necessary services to create events associated to subjects, create alerts, develop surveys associated to each event or visualize information about the surveys answered by the students. On the other hand, the mobile application is oriented towards the students, so they can register for an event, receive information about the event as well as answer surveys that have been proposed by the instructor organizing the event.

The structure of the article is as follows. Section 2 presents some proposal in order to solve the problem described in this section. Then, Sect. 3 describes the architecture of the application and explains briefly the data model used. Section 4 discuss in detail the functionality of the application. Finally, Sect. 5 presents a set of conclusions and future lines of work.

2 Background

In order to meet the needs described above, there are several alternatives that can be grouped basically into official and non-official tools. The official tools refer [5] essentially to the possibilities offered by the virtual campuses that universities normally have, and which are based on some type of LMS (Learning Management System), such as Moodle, Blackboard and others. This type of system offers a large number of functionalities aimed at facilitating the digitalisation of teaching [3]. In this sense, it would be feasible for a instructor to have specific virtual spaces for workshops or seminars, where students who can attend the event would be registered, the documentation associated with the activity would be stored, or the performance of assignments or feedback surveys would be planned. The advantage of this option is the integration of these activities with official teaching, so that the instructor has all the information about the students in a single system. However, there are some disadvantages due to the unofficial nature of these activities. In general, virtual campuses are designed to support official teaching, and many universities do not consider supporting unofficial activities. Thus, the instructor does not have the possibility of creating new virtual spaces directly, having to make a request to the virtual campus managers, this makes the organization process of these activities not as agile and simple as it would be expected. In addition, the organization of these systems is designed so that access to the virtual spaces can only be done by a group of students previously registered by the instructor or by the virtual campus managers, but not in the opposite sense (i.e., that a student can register for the event). Likewise, the system for creating surveys is not simple since surveys are designed to cover a very wide spectrum of test types (normally, it is first necessary to create a bank of questions, and from the bank of questions several types of question tests can be created). Finally, instructors usually do not have a system of alerts or notifications oriented towards mobile devices [2]. Notifications are sent to the registered email address. Therefore, this option

mainly poses problems of agility in the implementation process. Eventually, it should be noted that a large number of universities have created mobile apps [4] that offer services similar to those offered by the universities' web applications, as well as a set of value-added services, such as management of academic records, management of the calendar or class schedules in which students are enrolled, and management of events held at the university or faculty. However, these applications are generalist in nature and are not designed to adapt to the organization of this type of event and the information management it requires [6]. For example, with this type of application, it would be possible to manage the advertisement of the event, but it would not be possible to manage the students who participate or the carrying out of satisfaction surveys or their subsequent exploitation. This is why they do not adequately cover the needs raised.

The second option consists of the use of non-formal tools [7, 8]. This option involves the combination of a communication application (usually a mobile app, such as WhatsApp, Twitter and others) and a survey application (e.g., Google Forms). Thus, in this option, the instructor uses the first application to communicate to the students all the aspects related to the organization of the event, notifications, such as changes in the schedule or venue, etc., while using the second type of application for the creation of surveys related to the event. This option introduces flexibility and agility into the process, as no third-party permissions are required to use these applications. However, it has some disadvantages. Firstly, the lack of an information integration system that manages the applications used. Thus, there is no system of persistence that unifies and integrates the information from each application, so it will be a task for the instructor. This issue introduces other types of problems, such as the exploitation of the individual information of each student who has participated in these events. Likewise, there are other additional problems, such as the persistence in time of the information that has been generated from an event, the organization of the events by subjects, the complexity that can be produced if several events have to be managed simultaneously, and finally the dependence with respect to third parties (in this sense the applications can be modified by changing their way of functioning, or even closed up so that the proposed organization system would fail). In sum, this option offers flexibility and agility in the organization of the process but has the disadvantage of lack of integration and dependence on third parties. In addition, as unofficial solutions, agenda tools that allow for the management and organization of events could be considered. The main problem is precisely that the only functionality offered by an agenda is that they do not allow for the development of surveys, user management or the exploitation of information.

3 Application Architecture and Data Model

The tool has been implemented as a system consisting of a web application, an Android app and a relational database that is used to perform data persistence. In this way, the information that each application manages is synchronized by sharing the same database, which serves as a means of communication between both applications. The web application is used by the instructor to manage the extracurricular events of the subjects taught, while the Android app is used by the students to participate in the events generated by the instructors. In addition, the communication of the Android app with

the database is carried out through a web service that acts as a proxy between the two. Figure 1 shows the architecture schematic.

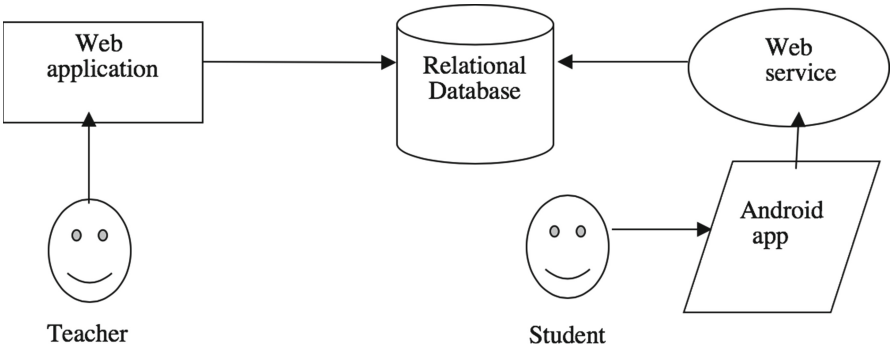

Fig. 1. Application architecture

The following technologies have been used to implement each of the components of the architecture. First, the mobile app has been implemented using Android, and the web service and web application have been implemented using PHP. In addition, the responses to the requests made by the Android app to the web service are performed using json files where the encoded information is found. This information is later used by the Android app to display it on the mobile. Finally, the relational database has been implemented using MySQL. Although a MongoDB database could have been used as the information is exchanged with the web application in json format, however, it has been decided to use a relational database since the information that is stored is regular (i.e., all data always has the same structure).

In order to implement the persistence of information, a relational database of the MySQL type has been used and a number of tables have been defined containing the information that the application needs to manage, as follows: data referring to users, data referring to the activities and the relationship with the students who participate in the activities, data about the surveys associated with the activities and data about the answers of the students to the surveys, data referring to the alerts that define the students and the relationship the activities to which they are associated, and data about the subjects defined by the instructor allowing for relating activities, students, and surveys.

4 Implementation

In this section, the functionality of the developed tool will be described. Since it consists of two different applications, the web application will be presented first and the Android app second.

4.1 Web Application

Functions have been defined in the web application for two different roles, administrator and instructors. The administrator performs 3 management functions:

- **Instructor Management:** Add, modify or remove instructors from the system.
- **Survey Management:** Create, modify or delete system surveys that will be associated later with a system activity. It is possible also to consult the statistics of the surveys associated with the activities.
- **User Management:** Consult the data of the users of the application as well as modify some of their data or delete the user from the system in case of error.

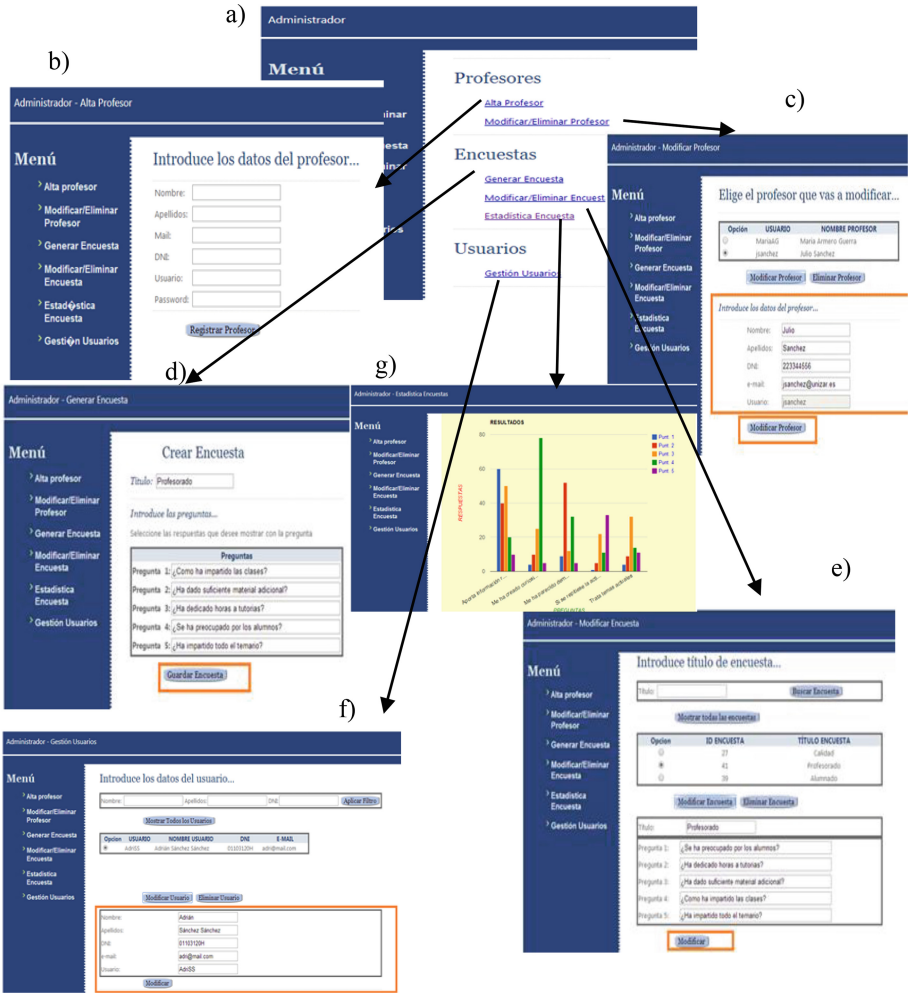

Fig. 2. Administrator functionality

Figure 2 shows the main administrator user interfaces. All of them are based on simple and intuitive forms that guide the administrator through the different functionalities implemented. The screen in Fig. 2a represents the main administrator menu where the

functions are listed. Figures 2b and 2c show the form for registering a instructor and for modifying or removing a instructor from the application. Figures 2d and 2e show the forms for generating a survey and for modifying or deleting a survey. Next, Fig. 2f shows the screen that allows users to manage the application. And finally, Fig. 2g shows an example of the statistics that the tool generates from the data collected in the surveys.

On the other hand, the functions of an instructor are as follows:

- **Private Activity Management:** A private activity is one in which it is necessary to be registered in the application in order to access the information. In addition, the student must have been validated by the instructor in the subject to which the private activity belongs. Once validated, the student will have access to all activities and surveys related to that subject. An instructor can create private activities, modify them or delete them. However, instructors can only manage those that they own.
- **Management of public activities:** An instructor can create public activities as well as modify the information of an activity created by him/her or delete it. A public activity is the one in which it is not necessary to be registered in the application in order to access the information. Instructors will only manage and view public activities of which they are owners.
- **Alert Management:** There is the possibility of creating alerts about activities. These alerts will be shown to the user in the mobile application. The alerts can also be modified or deleted by the instructor who created them.
- **Consult Surveys:** An instructor can consult all the surveys in the system in order to know the questions contained in a survey and associate it with an activity. Instructors can also find out the results of the surveys associated with the activities.
- **Student management:** An instructor must validate the students who request access to a subject created by the instructor. In addition, they have the possibility of cancelling any student who is registered in subjects created by them.

Figure 3 shows the main instructor user interfaces. The screen in Fig. 3a represents the instructor's main menu where the functions are listed. Figures 3b and 3c show the forms for registering, modifying or deleting a subject. Figures 3d and 3e show the forms to register, modify or delete an activity. Figures 3f and 3g show the forms to validate or eliminate a student. Figures 3h and 3i show the forms for displaying a survey and the statistics associated with it. Finally, Fig. 3j shows the form to manage an alert associated with an activity.

4.2 Android App

In the Android application, the functionalities for the students will be implemented. It is necessary to differentiate between registered and unregistered students:

1. **Registered student:**

- **Enrolment in Private Activities:** A registered student can consult all the information about private activities that a specific instructor has created for a subject. It is possible also to enroll in any of them. In order to access these activities, the student must have previously been validated by the instructor.

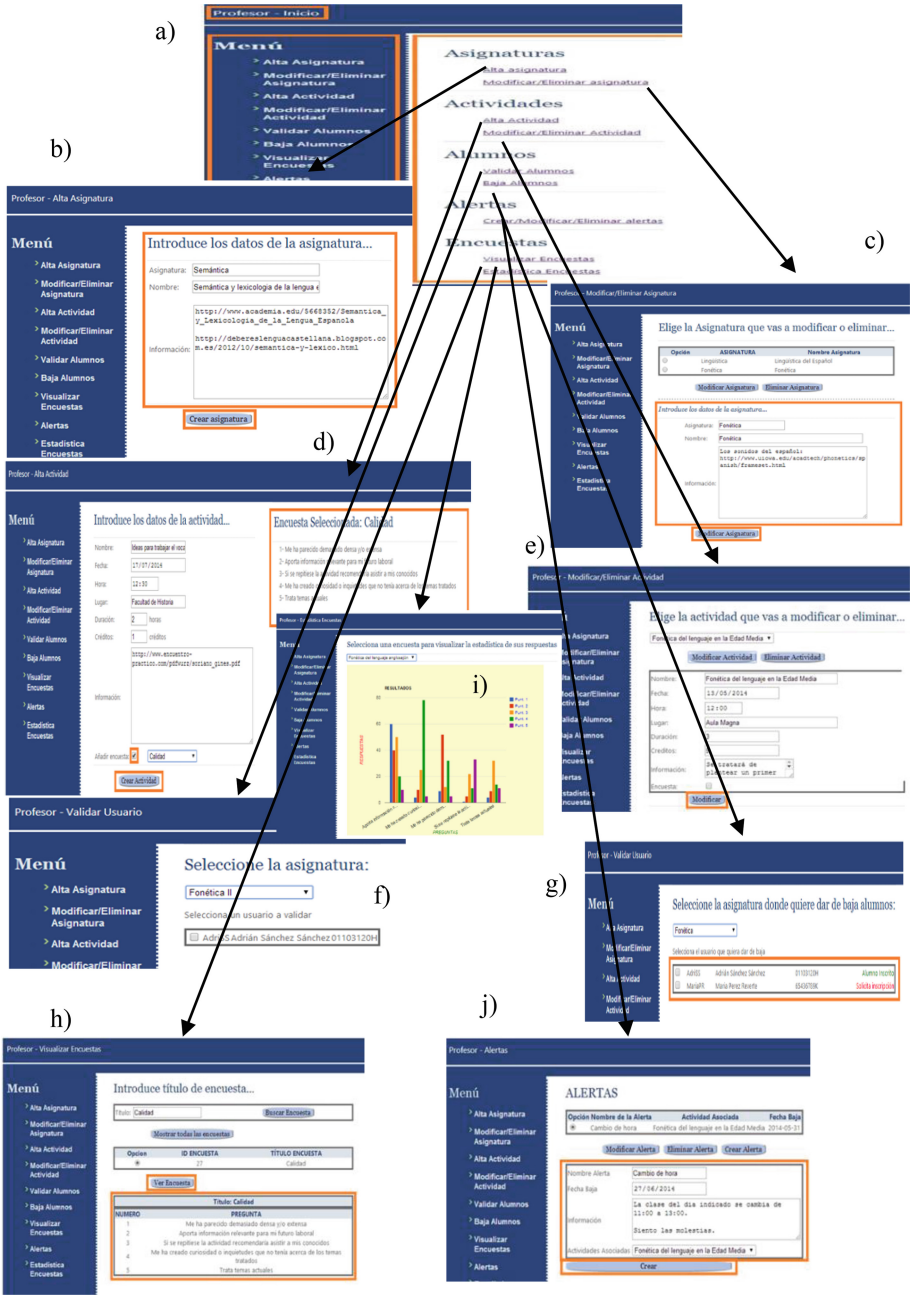

Fig. 3. Instructor functionality

- **Enrolment in Public Activities:** A registered student can enroll in public activities, which are associated a number of credits assigned to the students who carry them out. Likewise, they may have an associated satisfaction survey.
- **Credit counter.** A registered student can consult the credits that s/he adds with the public activities to which the student has been registered and can establish a goal of credits to fulfill. When the user consults the credits, if it has reached the maximum established by the student, the app will be communicated to the student through a pop-up that s/he has reached her maximum credits. However, the student could continue to carry out activities and accumulate more credits.
- **Alert notification.** A registered student will receive, by means of a notification in the form of a pop-up, the alerts generated by the instructors about the activities in which they are enrolled.

2. Unregistered student:

Any student who installs the Android app can consult all the information about public activities and the alerts that have been generated about them.

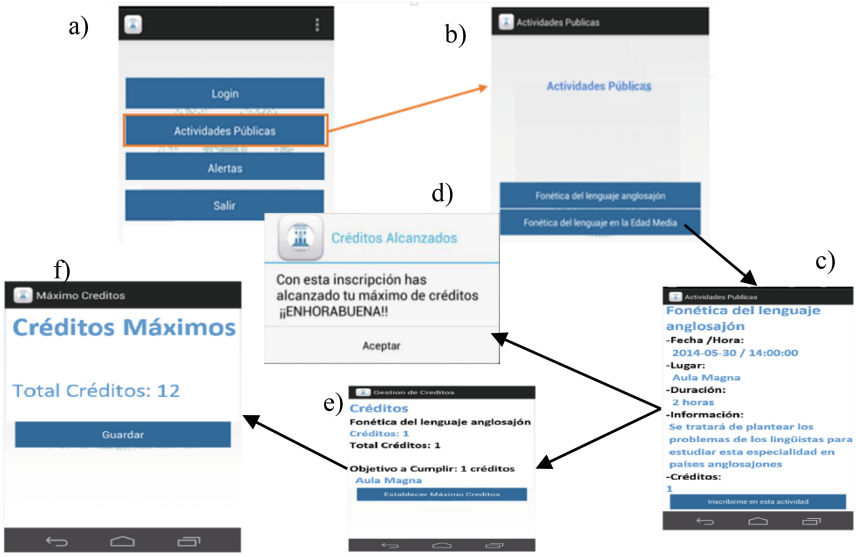

Fig. 4. Student functionality

Figure 4 shows some of the functions associated with a student. The screen in Fig. 4a represents the main menu of the app. Figure 4b shows an example of a list of public activities, and Fig. 4c shows an example of access to a public activity. Figures 4d and 4e show information about the credits associated with a public activity and the notification generated when the user reaches the maximum number of credits that they have established. Finally, Fig. 4f shows the interface that allows configuring the maximum number of credits that a user wants to obtain.

5 Conclusions and Future Work

This article has presented a tool designed to manage extracurricular training activities such as seminars or training workshops. The system includes a web application oriented to instructor tasks where it is possible to register public and private activities associated with the subjects taught, as well as design surveys associated with the activities or consult the statistics generated. In addition, an Android app has been implemented that allows students to register for activities generated by instructors with the restriction that in order to access private activities they require the validation of instructors. Likewise, students can carry out surveys associated with these activities, and keep track of the credits they obtain for participating in them. The system is currently not active and is not used in any higher education environment. Funding has been requested with the aim of integrating it into several universities that have shown interest.

As future lines of work, it is proposed: 1) Create a version for iOS; 2) Extend the functionality of the application to allow documentation to be associated with activities; 3) Allow the system to be configured in several languages; 4) Allow students to be able to send documents to the instructor associated with an activity or survey, 5) Currently, the application has not taken into account data privacy and the protection. In this sense, a future line of work is to implement the necessary mechanisms to respect the RGPD law and integrate privacy management into the mechanisms of the VLE of the universities.

Acknowledgments. This work has been partially supported by the European Commission through the project "colMOOC: Integrating Conversational Agents and Learning Analytics in MOOCs" (588438-EPP-1-2017-1-EL-EPPKA2-KA).

References

1. Claudia, C.: The role of extracurricular activities and their impact on learning process. In: The Annals of the University of Oradea, 1117 (2014)
2. Lovászová, G., Cápay, M., Michalicková, V.: Learning activities mediated by mobile technology: best practices for informatics education. In: CSEDU (2), pp. 394–401 (2016)
3. Lubis, M., Fauzi, R., Lubis, A.: Enterprise application integration for high school students using blended learning system. In: MATEC Web of Conferences, vol. 218, p. 04016. EDP Sciences (2018)
4. Molnár, G., Szűts, Z.: Advanced mobile communication and media devices and applications in the base of higher education. In: 2014 IEEE 12th International Symposium on Intelligent Systems and Informatics (SISY), pp. 169–174. IEEE (2014)
5. Sarasa-Cabezuelo, A., Sierra-Rodríguez, J.L.: An app for managing unregulated teaching activities. In: 2014 International Symposium on Computers in Education (SIIE), pp. 217–222. IEEE (2014)
6. Sarasa-Cabezuelo, A., Sierra-Rodríguez, J.L.: A system to manage non-formal higher education activities. IEEE Revista Iberoamericana de Tecnologias del Aprendizaje **11**(3), 205–212 (2016)
7. Sarasa-Cabezuelo, A., Caballe, S.: A tool for creating educational resources through content aggregation. In: International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, pp. 515–524. Springer, Cham, November 2019
8. Turani, A., Calvo, R.A.: Beehive: a software application for synchronous collaborative learning. Campus-Wide Information Systems (2006)