Mobile Personal Environment for Higher Education

Hendrik Thüs
RWTH Aachen University
Germany
thues@informatik.rwth-aachen.de

Mostafa Akbari
RWTH Aachen University
Germany
akbari@informatik.rwth-aachen.de

Ulrik Schroeder
RWTH Aachen University
Germany
schroeder@informatik.rwth-aachen.de

ABSTRACT

By using currently popular tools like blogs or wikis and services like social networks, the users are becoming more and more involved into the production of content instead of just consuming it. This style of working should not only be used for pleasure but can also be embedded into higher education. Students should not only passively listen to lectures but also participate actively and help fellow students solving certain tasks. This can be done by using modern hand-held devices.

Due to the ability of those devices to determine its current position, the collaboration and networking factor can be extended by location-based services. It would be easier to communicate or to collaborate with other people if one knows where these persons are. Some situations can be handled more easily and effective if there is such a location-based PLE system that allows communication and collaboration among its peers. The proposed application for the Android operating system will be focused on location-based scenarios for higher education. Students will have the possibility to offer help by providing skills and to receive help by searching for close-by people, to store and to retrieve information location-based, and to participate more active in lectures. This improves communication between students and helps them to improve their learning.

1. INTRODUCTION

Many of today's students own smart mobile devices. In case of the RWTH Aachen University, 58% of the students use devices, like Android Phones, iPhones, iPod Touches, or other smartphones [Media Usage Report of the RWTH Aachen, not yet published]. In contrast to the previous year, the amount of students, who owned such a smart mobile device, rose from 36% up to 58%. With this change in equipment, we as one part of the student's personal learning environment should provide new ways of communication to the students to support collaboration in their digital personal learning environments. In general, no other device is as personal for a student as the own mobile phone.

Due to the ability of modern hand-held devices to determine their current position, the collaboration and networking factor can be extended by location-based services. It is easier to give feedback and to communicate or to collaborate with other people, if one knows where they are. Some situations can be handled more easily and effectively with a location-based Mobile Personal Environment (MPE) system, which allows communication and collaboration (see chapter 4). In the course of this project, an application for the Android operating system has been implemented, which focused on location based scenarios for higher education. The application gives students the opportunity to offer help by providing skills and to receive help by searching for people close-by, to store and to retrieve information location-based, and to actively participate in lectures by asking questions or by giving feedback. The MPE application improves communication between students and helps them to enhance their learning. All the students can inform fellows about their skills by providing them with some keywords using the MPE system. Due to the mobility of the devices, the current participant's position is known to the system. This way, each participant of the MPE system can now search for such keywords to find the appropriate person that is currently nearby, who is not occupied at the moment and might be able to solve certain problems.

By giving students the possibility to gather in a virtual room while being close by in the real world, they are able to ask questions about a lecture, to take notes what might be of importance, or to exchange information. This data is enriched with the context information of time and location. So, the MPE system enables students to communicate with each other and to work collaboratively in order to improve a lecture by giving suggestions, to help others answering questions or to solve given tasks.

The following chapter 2 gives a short introduction to Personal Learning Environments while the focus of chapter 3 is mainly on extending a PLE with a mobile context. Chapter 4 depicts three scenarios to show the capabilities of a mobile PLE. Those scenarios are the basis for the application proposed in chapter 5. To test and to improve the usability of the application, four sets of user-tests with a total of 17 students have been conducted. The results are presented in chapter 6. This paper concludes with chapter 7.

2. PERSONAL LEARNING ENVIRONMENT

A Personal Learning Environment (PLE) is designed to enable its users to communicate with each other, to gather information, share data with colleagues or fellow students,

.

and to keep track of the current state of learning. According to Wilson [7], PLEs are concerned with the coordination of the connections made by the learner with units and agents across a wide range of systems and are envisaged primarily as open systems.

Van Harmelen [5] defines PLEs as systems that help the learner to manage their own learning and to take control over it. They are able to set their own learning goals, to be responsible for the content and the process of how it is gathered, and to communicate with others in the process of learning.

The learner is responsible for documenting both, the current state of learning and the objectives that have to be achieved. A PLE should give its users the possibility to control their own speed of learning and the contents they want to learn.

3. MOBILE PLE

The drawback of a PLE running on a desktop PC is the lack of mobility and context information. The current position of the learner using a mobile device can generate such context information. As mentioned, 58% of the students of the RWTH Aachen University own and use mobile smartphone devices. This shows that there is a great potential in utilizing such devices to generate context information in order to support the students in their everyday-life at the university.

The Mobile Personal Environment (MPE) presented here enables students to be aware of people and events in their proximity. While being on campus, one does not know who the people are that are nearby or what they are capable of. Therefore, it is not known in which fields of study they can provide help. Thus, there is a great but rather unused amount of knowledge that could be utilized to give feedback to other students to help them understanding certain topics. The combination of skills in certain topics and location-based services enables users of the MPE system to search for certain skills, offered by people in the proximity. This eases the process of getting to know other students and this helps utilizing unused knowledge to improve learning and to give feedback.

3.1 Requirements

The MPE enable its users to request location-based information about other participants. In detail, those participants are students or other people as well as Points of Interest (POI), latter ones are only virtually existent. The decision, which participants are presented to a certain user is done by comparing positions. These are the current dynamic positions of the devices owned by the users or the static positions of POIs, respectively, which are compared to the actual position of the user who requested this selection.

The students are able to find other people with specified skills by searching for keywords which describe such skills. This corresponds to the function that is used for the second scenario, described in chapter 4.2. This enables the users to get to know other students that might be of help and to work collaboratively.

Special channels can be associated with certain events (lectures, tutorials, ...) so that the students are able to exchange information according to that event, these are the previously mentioned Points of Interest. They can ask and answer questions, give feedback, take notes, or make annotations. This might encourage the students to participate more actively while attending a lecture instead of just consuming knowledge.

4. SCENARIOS

The following three scenarios represent the main aspects of this Mobile Personal Environment. The first scenario is about how students can ask questions during a lecture or during an exercise without feeling embarrassed because they might think that the question is stupid [1]. The second scenario discusses how students can find other fellow students with specific skills that are nearby to be helpful regarding a certain problem. The last of these three scenarios is about how achievements that are made during a class can be stored location-based, so they can be accessed by others.

4.1 Asking Questions

The first scenario is about Anna, an MBA student, who has some problems understanding some aspects of a certain lecture she is currently attending. The following order of events shows what might be the case in most classes:

1. Anna finds out that the lecturer put the current slides of the lecture on a website.
2. She downloads them and opens the file with the appropriate program.
3. While reading, she does not understand some facts, so she asks her neighbors if someone can explain it to her.
4. Since no one knows an answer and she does not have the heart to ask the lecturer [1], she writes her question in an email to the assistant of the lecturer.
5. Sometime later, she will receive an answer. But this question is only answered to her, not to the other students. Since the lecturer does not know about the problem, he does not explain it again. The assistant does not know that Anna is not the only student with that certain problem.

Now this first scenario with the location-based MPE:

1. Anna attends a lecture and when she logs in into the MPE system, she sees that the current slides to the lecture are uploaded to the channel that is associated to that lecture. This channel is emphasized in the system because it is currently taking place and Anna is attending it.
2. She opens the uploaded slides and while reading, she does not understand some facts.
3. Due to her problems, she asks her question directly in the channel with a reference to the slides.
4. After a short time, a discussion about the problem has arisen. Not only the students are participating, the assistants are also writing their opinion.
5. Since many students are interested in this question, the lecturer decides to talk about it in the next lecture.

The channel that is attached to a certain POI such as a lecture or a tutorial can be used to exchange information, to ask questions, and to discuss problems. Every participant can ask questions without feeling embarrassed because there is no need to use real names in this system and one does not have to stand up in front of everyone to ask the question.

4.2 Searching for People

The second scenario is about Alex who attends a lecture and also has some problems understanding certain aspects while learning for a testate:

1. Alex is sitting in a learning room at the university and is learning for a testate that is taking place in one week.
2. He does not understand one aspect and even after some time thinking about it, he still has no clue.
3. Alex starts to write emails to his fellow students with the question and hopes that one of them can help him with his problem.
4. After a couple of hours he receives an email with a poorly explained answer to his problem.

Now the second scenario with the location-based MPE:

1. Alex is sitting in a learning room at the university and is learning for a testate that is taking place in one week.
2. He does not understand one aspect and even after some time thinking about it, he still has no clue.
3. Alex searches for some keywords in his MPE system to find those people nearby, who have the ability to help him with his problem.
4. He contacts one of them and asks to meet him in a couple of minutes.

Every student can provide some keywords that inform others, which skills they offer. Each participant of the system can search for such keywords to find the appropriate person that is currently nearby, who is not occupied at the moment and might know the answer to certain problems.

4.3 Saving and Retrieving Data

The last of the three scenarios is about Claudia who is studying computer science and minors in electrical engineering.

1. Claudia is participating in a practical course about electrical engineering.
2. The task of the participants is to implement a program for an integrated circuit.
3. Since she knows a bit about integrated circuits, she is asked by her neighbor for some help.
4. She gives him some hints on how to solve his problem.
5. After some time, another fellow student of her asks the same thing, so she repeats her answer.

Now the third scenario with the location-based MPE system:

1. Claudia is participating in a practical course about electrical engineering.
2. The task of the participants is to implement a program for an integrated circuit.
3. Since she knows a bit about integrated circuits, she is asked by her neighbor for some help.
4. While thinking about an answer, she notices that this might be interesting to other students, as well.
5. She writes the answer to the question into the localized channel of the practical course.
6. Every student takes a look at that channel periodically.

The information that might be interesting to the whole course is stored in there. The back-end-channel of a certain course can also be used as a centralized pool of information. The stored information can not only be accessed at that certain location, but it is emphasized there. That information can be obtained globally by just adding this POI to the MPE's favorite-list of of Points of Interest.

5. ANDROID-APPLICATION

The proposed application has been implemented for the Android operating system. Due to some data security issues and to keep as many calculations as possible away from the clients, it employs a client-server architecture. To enable the clients to communicate with the server, an Application Programming Interface (API) has been implemented.

5.1 Client / Server

The server of this system maintains the data of all the participants, e.g. their current positions or their provided skills. The client-application requests a certain set of data by sending a command to the server which then responds with an XML (Extensible Markup Language) document that contains the requested data. This data is calculated in real-time after a request is received and then transmitted to the enquirer.

In order to implement client applications for different devices employing different operating systems, the communication with the API has been standardized. The only requirement for a system is that it is able to access the Internet and that it is able to parse XML files. Thus, one can implement applications for different kinds of smartphones or operating systems using the API of this system.

5.2 Locations

The most crucial part of a location-based application is to determine the current position of the device it is running on. The whereabouts of a person is private information. But according to Smailagic and Kogan [4], the users are likely to transmit their position; especially students seem to have only few problems with this [6].

The best method to query the current position is provided by the Global Positioning System (GPS). But since there is no possibility to receive GPS signals inside of buildings, the WiFiLoc positioning system proposed by Göndör [2] has been integrated. This system requires some measurements

before it can be put into operation. Therefore, it is only available at some special locations.

The MPE system tries to employ the best method that is available at a certain point in time. The smartphone determines on its own, whether GPS signals can be received or the WiFiLoc positioning system is available. As a fall-back, the GSM signal can be used to determine the position, but this is not as accurate as the previously mentioned systems.

The MPE system displays those participants that are currently close to the user (inside a specified radius). Especially the Points of Interest might be of current interest to the user of this application because at those locations, publicly available data can be stored. The application also allows the user to search for any participant in the system. No matter where the POI or the user is located. Here, the radius does not influence the search result. There is also the mentioned possibility to search for users inside the specified radius who have special skills that might help solving a certain problem. To communicate with others, the integrated back-end (e.g. a chat-room) is used.

5.3 The Client-Application

The exact functionalities and the possibilities of how to handle this application has been developed by creating paper prototypes. These prototypes have not been evaluated by conducting user-tests. They were used to get an overview of how to embed the whole set of functionalities to the application without overstraining the users. These prototypes have helped to develop the first version of the User Interface (UI) which has been evaluated later on by conducting some user-tests (see chapter 6).

When the application is started, the user is presented the start-screen. Here, the integrated back-end is displayed that is associated with this user. Every use of the back-key of the device on other screens of this application brings the user back to this first screen. Besides this first one, the application has five main screens that represent different functionalities: a list of users nearby, a list of POIs nearby, the search function, the list of friends, and the list of favorite POIs. The buttons to switch from one to another are located at the top of each screen (see figure 1).

Figure 1: The menu that is used to switch from one screen to another. These buttons are used to switch to: Users, POIs, Keywords, Friends, Favorites (f.l.t.r)

On the right side of each screen, there is another button that indicates the current status of the application (see figure 2). The status can be:

Online The application is connected to the server and periodically transmits the current position. Data can be retrieved from the server.

Offline The application cannot establish a connection to the server. No data can be retrieved from the server.

Pending changes Since the device periodically updates its position, the displayed data is not always up-to-date. There is new data available.

Figure 2: Status-Button for online, offline and pending changes (f.l.t.r.)

The user can click on this button to refresh the list of participants that is displayed on the current screen.

Every functionality offers a list of participants which are selected according to the functionality itself. To save traffic, the application does not display every item that the server can provide. In the preferences of the MPE, the user can choose a value x that determines the maximum amount of items that are displayed after a request has been sent. If there are more items available, the user has the option to extend the list that is currently shown. At the bottom of such a list, a *Get More* button is located if there are more items that can be displayed. With a click on this button, the next x items are fetched from the server.

6. USABILITY-TESTS

To test and to improve the usabiity of the application, three sets of user-tests have been conducted. The first two sets with five users, the third one with three users, and the last one with four users. According to Nielsen [3], it is sufficient to conduct user-tests with five users. Thus, 17 user-tests with different students are more than required. As stated, the number of usability problems that are found in a test with n users and a total number of N problems is

$$N(1-(1-L)^n)$$

where L denotes the proportion of usability problems discovered while testing a single user with a typical value of 31%. According to this formula, about 84% of the N problems can be found if five users are tested. The second set of user-tests with five persons should reveal another 13% of the original N problems. Thus, conducting user-tests with ten persons should result in finding about 97-98% of the problems concerning the usability. The objective is that the participants consider the application as intuitively operable.

6.1 Conducted User-Tests

The students who took part had different foreknowledge concerning Android smartphones (see table 1). This was the first question, the students were asked prior to the tasks. Everybody has been told not to hurry and to talk about what he or she is doing, if something unexpected happens, or if there are possible improvements. They have also been told that there are four important hardware buttons integrated in Android devices. These buttons are the back-button that is used to go to the previous screen, the menu-button to display the menu that is assigned to the current screen, the search-button to display the search dialogue of the current screen, and the home-button to switch from the application to the Home-Screen of the Android OS. The students were told not to use the latter one because it exits the application. The device that was utilized for these user-tests was

a Motorola Milestone, running the Android OS version 2.1-update1 (Eclair).

The content of the screen of the device has been captured by using the tool androidscreencast[1], a Java-based tool. It allows recording the content of the screen as a video. Unfortunately, it only has a frame rate of 4-5 frames per second which is not that smooth. Also, it does not display the positions of clicks or the swipes, a user conducts. To compensate these disadvantages and to capture audio, the device itself was also filmed by using a JVC MiniDV camera.

The students have been told that the items of lists - such as users or POIs - can be clicked once to display the item and that they can be long-clicked to display a context dialogue with additional functions concerning that certain item. This has been necessary since some of the students have never worked with an Android device, thus, they do not know how to handle it.

The students had to solve four tasks. These tasks were read to each of them in German. In the following, they are also translated to English:

1. Schaue dir die Hilfe an und lies diese durch.
   Take a look at the help-dialogue and read through it.

2. Suche nach dem User Claudia und füge diese zu deiner Freundesliste hinzu.
   Search for the user Claudia and add her to your list of friends.

3. Schaue dir an, welche User gerade die Übung "Übung abc" besuchen.
   Get to know which users are currently attending the "Übung abc"[2].

4. Suche nach Usern, die dir beim Thema "analysis" helfen können und schreibe einem eine kurze Nachricht.
   Search for those users which can help you with the topic "analysis" and send a message to one of them.

After each of the first three sets of user tests, the proposed improvements have been implemented. Those changes have been tested in the following set of the user-tests, respectively. The setup was the same as in the previous set. This was necessary to be able to compare the results with the previous ones.

The result of the first set of user-tests was that the help-dialogue was not found that easily as it should be. Also, the menu-items were not that intuitively (figure 1 contains the final ones).

The second set resulted in increasing the sizes of various labels and buttons and also in adding some explanatory texts.

The results of the third set was that some functionalities were made accessible not only by performing a long click on an item but also on some other screens. Even though the users were told that long-clicks are possible, they were not intuitively. This iteration also included adding some further help-dialogues and some changes in sizes of labels and buttons.

The fourth and last set of user-tests resulted in some minor changes of the UI. There were far less and serious suggestions made by the students.

[1]http://code.google.com/p/androidscreencast/

[2]"Übung abc" is a certain tutorial

	Gender	Duration	Foreknowledge
Student 1	Male	4:00 min	N/A
Student 2	Male	4:13 min	N/A
Student 3	Male	7:28 min	owns an iPod Touch
Student 4	Male	6:19 min	N/A
Student 5	Male	3:40 min	owns an HTC Desire (Android-based)
Student 6	Male	6:31 min	N/A
Student 7	Male	5:28 min	owns an iPod Touch and a Motorola Milestone
Student 8	Male	7:56 min	N/A
Student 9	Female	5:26 min	N/A
Student 10	Male	6:13 min	N/A
Student 11	Female	8:14 min	N/A
Student 12	Female	8:42 min	does not own a smart phone
Student 13	Female	3:36 min	does not own a smart phone
Student 14	Male	4:52 min	owns a Motorola Milestone
Student 15	Female	4:59 min	owns an HTC Desire
Student 16	Female	5:55 min	N/A
Student 17	Female	3:58 min	owns an HTC Desire

Table 1: List of participants of the four user tests

7. CONCLUSION

The MPE system enables its users to connect with each other and to find others that are nearby. These functionalities can be utilized to offer and to gain help among students and to give each other feedback.

The application has been implemented to be used on devices that base on the Android OS. The implementation of the application and the API that is used to retrieve and to store data took roughly about 10 weeks. The result is a system that is based on two parts: the application running on the client's devices and the API running as scripts on a web server. Latter can be utilized to implement applications for different kinds of mobile devices such as iPhones, Blackberrys, or Windows Phone 7 devices, and even Microsoft Windows/Linux/MacOS based notebooks (if they are able to determine their position). Thus, there is the possibility to use the same service on various kinds of devices. The proposed application running on the client's devices fulfills the requirements according to the scenarios, presented in chapter 4. Its usability has been improved by conducting four sets of user-tests with a total amount of 17 students. After the first three sets, the suggested changes to the application have been implemented to improve the usability. The students of the last set stated that the handling of the MPE system is intuitive, especially if one knows how to handle Android devices. But even those students with no experience of handling such devices were able to solve the given tasks in a proper time frame.

8. REFERENCES

[1] G. Böhm. Integration von spontanem feedback in große blended learning szenarien mittels microblogs. Diploma thesis, RWTH Aachen University, Aachen, Germany, 2009.

[2] S. Göndör. Building a location sensing and positioning service to enable indoor location based services.

Diploma thesis, RWTH Aachen University, Aachen, Germany, 2010.

[3] J. Nielsen. Usability testing with 5 users.

[4] A. Smailagic and D. Kogan. Location sensing and privacy in a context-aware computing environment. *IEEE Wireless Communications*, 9(5):10–17, 2002.

[5] M. Van Harmelen. Personal learning environments - jitt.

[6] S. Vihavainen, A. Oulasvirta, and R. Sarvas. "i can't lie anymore!": The implications of location automation for mobile social applications, 2009.

[7] S. Wilson. Architecture of virtual spaces and the future of vles. *Retrieved October*, 7:2005, 2005.