

Mobile and Ubiquitous Computing

Summer semester 2014

Michael Weber, Bastian Könings, Marcel Walch, Tobias Sieber

Assignment 3: Context & Localization

Submission deadline: Monday, 9 June 2013, 7:00

Task 1: Reading Assignment: Context

2+2+2 points

- B. Schilit, N. Adams, and R. Want, "Context-aware computing applications," in Workshop on Mobile Computing Systems and Applications, 1994, pp. 85–90. <http://dx.doi.org/10.1109/WMCSA.1994.16>
 - A. K. Dey, "Understanding and Using Context," Personal and Ubiquitous Computing, vol. 5, no. 1, pp. 4–7, Feb. 2001. <http://dx.doi.org/10.1007/s007790170019>
- a) Schilit et al. introduce the concept of context-aware computing and describe four categories of context-aware applications. Dey distinguishes only three application categories. Explain how Schilit et al's categories are mapped onto Dey's categories. Argue why or why not it makes sense to focus on the three categories proposed by Dey.
- b) How does Dey define context and context-awareness? Explain why Dey states "When we try to apply previous definitions to established context-aware applications, we find that they do not fit."
- c) Find two examples of context-aware applications (published as research papers) that are not mentioned in the papers above. Give a summary how the applications use context information. Where does each example fit in Schilit et al's categorization? Where does it fit in Dey's categorization? Feel free to search in academic databases (Google scholar, ACM DL, IEEE Xplore), but do not forget to cite your sources!

Task 2: Mensafinder

2+3+2+2+2+2+1+(5)+(3) points

Goal of this exercise is the development of an ubiquitous computing application to help locate your friends in the Mensa at Ulm University. The application should support the following use case:

1. Alice sits down at a table inside the Mensa and places her smartphone on the table. She turns it towards the staircase in the middle of the Mensa (the top of the phone is facing towards the stair case), and sets her Mensafinder status to *discoverable*.
2. After Bob finished paying, he stands close to the staircase, and uses the Mensafinder to search for friends. The Mensafinder shows him the rough direction in which Alice is sitting.

The Mensafinder application for the Android platform should be developed in multiple steps. It consists of a client, that can request the orientation of the phone's compass sensor. This information can be exchanged with a server via a RESTful API based on HTTP requests containing json formatted data.

The Mensafinder server is online at <http://barracuda-vm8.informatik.uni-ulm.de/>. Visit the URL and read the API documentation of the Mensafinder service. Pay attention to the assumptions and restrictions that the service documentation contains. Make sure you understand all features by dispatching some mock requests with curl or similar HTTP tools.

Please contact Björn (bjoern.wiedersheim@uni-ulm.de) in case of service issues or outages.

- a) The GUI of the Mensafinder must support a *discovery mode*, a *search mode*, and a *settings dialog* to change the username. Implement the general GUI and the settings dialog.



- b) Implement the discovery mode. When the phone is oriented towards the staircase, the user can start the discovery mode. The application then sends the current compass direction and username to the Mensafinder server.
- c) If the phone is not resting level on the table, a message should be displayed asking the user to position the device properly before contacting the server. Define reasonable error margins for levelness to avoid unnecessary error messages.
- d) When the device is moved in discovery mode, a message should be displayed to ask if the user wants to recalibrate the device or if the status should be changed to undiscoverable. Once the user calibrated the device and confirmed the message, an update should be sent to the server. If the status should be changed to **undiscoverable**, the application should **logout from the server** to prevent showing up in search results.
- e) In search mode, the application should first retrieve a **list of discoverable peers** from the server. After that, the application needs to **handle push notifications** from the server.
- f) In a **basic compass view**, visualize the directions in which friends are sitting, as well as their names. Remind the user to stand close to the staircase for best results.
- g) **Adapt your visualization to a map of the Mensa.** There is a map template available on Moodle.
- h) **Bonus:** Extend your application with an augmented reality view by overlaying the live camera stream with the visualization of directions. How can you achieve calibration to make the application work even if the user is not standing at the staircase?
- i) **Bonus:** Test your application with real phones in the real setting and refine it. Document your tests and test results, including negative results.



Submission details

One group member submits the solution for the group incl. a `group.txt` with the member names.

Text tasks: Submit only PDF or plain text documents.

Programming tasks: Submit an **exported Eclipse project named:** `muc14_<assignmentnumber>_<groupname>`.
Always submit the complete project incl. source and apk files.