
Android Smartphone Programming

<http://proglang.informatik.uni-freiburg.de/teaching/androidpracticum/2015/>

Exercise Sheet 6

Telephone Manager (20 points)

In this exercise you will create an application to manage incoming phone calls. Its core functionality should be to decide whether the telephone ring tone should be muted on an incoming call or not. This will be decided by looking through the built-in calendar to check if there is currently a meeting/event taking place. If so, the telephone gets muted. If the application mutes an incoming call, the caller should be notified by an SMS with some explanation. The user should also be notified of all muted calls. *You have to take efficiency and usability into account.*

First, the following main features should be supported in your application.

- Mute incoming telephone calls if an event whose availability is *busy* happens. Check the built-in calendar. *Note, after muting, the cellphone has to be in the state as before.*
- Notify the user about muted calls. The notification has to include date and time of the incoming call in combination with the caller's phone number, *undefined* if the phone number is unknown, or the caller's name if the phone number is present in the contact list.
- Send an SMS back to the caller to inform him that his call was omitted.

Develop a possibility to activate/deactivate the *Telephone Manager*. *Feel free to choose your own solution. This can be done by an Activity, an AppWidget, an extension to the preferences menu, etc..*

Further, develop a possibility to recognise if the *Telephone Manager* is active or inactive. This should be possible without starting an extra activity! *Choose a suitable solution. For example, a Status Bar Icon, a permanent Notification, an AppWidget, etc..*

Second, implement a *White-Listing* functionality. The *White List* is a simple list that contains contact (only known contact from the address book). It should be possible to add and remove contacts from the list. Incoming calls of listed contacts should not be muted, even if the application is active.

Third, provide an installable *.apk* file with your application and provide a QR-Code to install your application. *Hint: Use your university web directory /~user/ to publish your application. Your QR-Code has to include a link to the .apk file.* Your submission has to include the QR-Code (as *.pdf* file) and the *.apk* file in addition to the *Eclipse Project*.

Extra task. Extend you app so that incoming call are not noticeable, this includes also to deactivate vibration and the display.

Submission

Deadline The submission deadline is **01.02.2016, 12:00 (noon)**. Late submissions will not be accepted.

Project Create an *Android Studio Project* (e.g. **exercise6**) for each exercise. Use **androidlab.<user>.exercise6** as package name. Make sure that your project include all source files.

Report Your solution will consist of a *pdf file* **<user>_report6.pdf** with a description. The description must be limited to one page per exercise. Submitting more than one page will lead to reduction in points. The description may be either in German or in English. Clear and understandable style is required.

Submission Submit your solution to the subversion repository. Your solution will consist of one *folder* (e.g. **exercise6**) for each exercise which include the project and the report.

You are strongly encouraged to test your solution. Provide your source code with comments to understand the intention. Clear and understandable style is required.