Exercise 2: Android Cell Infos

183.286 (Mobile) Network Services and Applications

Group 17

Measurements report

For the measurements, we used a phone with **Android 6** and set the prefered network type to be **2G**.

During the measurements, the cell with id **19368** of type **GSM** was identified as being the **active** cell.

The following **neighboring** cells were found during the measurement:

- 24842
- 19443
- 4285
- 25148
- 7350

Since the prefered network has been set to **2G**, all the found cells were of type **GSM**. Please see screenshots for more details.

Having the preferred network option disabled the ***getAllCellInfo()*** call also returned cells of type **UMTS**, but with unknown value for the cell id, mcc, mnc.

Project report

Setting up the Android project worked for us quite fast and easy. However, we had some problems with usb debugging on Windows platform with an LG phone, because the adb tool was not automatically installed by Android Studio. On Linux platform, we had to have the corresponding permissions for the adb server, for the process to be able to communicate with the phone.

Furthermore, in Linux we also had to complete the following steps, in order to be able to run the application, both on emulator and on a real phone:

- make sure your processor supports virtualization: `egrep --color=auto 'vmx|svm|0xc0f' /proc/cpuinfo`
- make sure virtualization is enabled: `zgrep CONFIG_KVM /proc/config.gz`, otherwise enable it in BIOS
- rename `Android/Sdk/tools/lib64/libstdc++` to some backup folder and link /usr/lib/libstdc++.so to ~/Android/Sdk/tools/lib64/libstdc++/

We used *butterknife* as a resource injector for our activities, which sometimes caused issues, such as NullPointerExceptions. The problems were always fixed by a cleaning and building again (*gradlew clean build*).

Additionally, we also sensed a lack of support for free Android mockup creation software. In the end we decided to use the trial version of the **UXPin**.

Other problems we stumbled upon were with the Map View. First of all, we spend quite some time searching for the best way to find out the location for the cells, eventually deciding for the services offered by the *opencellid.org* (limited to 1000 API calls per day). Secondly, we observed that the Google Map View did not load, only after interacting with it (e.g. click on it).