App Development for Smart Devices
CS 441/541
Assignment #3: Localization

Published: Thursday, Mar 29 2018
Deadline: 11:59 PM, Monday, April 9, 2018

All projects are required to be submitted with up to a one-page report that specifies what was implemented, and until what point. You need to also justify your design decisions. Graduate students need to submit additional report of the graduate student requirements.

In this assignment, you will build an app to acquire and map device locations. The purpose is to become familiar with the location and the Google Maps APIs. The required API level for this assignment is 23 (Android 6.0). This assignment requires using Google Play services location APIs and Google Maps Android API.

1) 50 points maximum

Build an app to display and remember locations in terms of latitude and longitude coordinates as well as postal address. Review the location and maps section in the online Android Developer Guide (https://developer.android.com/training/location/index.html) before getting started.

- Create an app that obtains the phone's current location and displays the latitude longitude values in a TextView. (5 points)
- Use Android Geocoder class to get your current ADDRESS and display next to the longitude and latitude. (5 points)
- Use callbacks so that when a new location becomes available, the coordinates in the TextView are updated. (5 points)
- Add a button to "check in" to your current location. This should store the current location and time in an SQLite database. Also, all stored locations (longitude, latitude, time, address) should be shown in a list in the app. (10 points)

- Allow users to add a custom name for a location when checking in. Store the names in SQLite database. Check in at five locations on ODU campus and name them. Add the custom name to the check-in list from the previous bullet (leave blank if no name assigned) (5 points)

- Any check-in within a radius of 30m of an existing check-in should be automatically associated with the prior check-in name, if any, and postal address. Design your database so that one location can be associated with several check-ins. Store the exact latitude, longitude and time of each check-in. Normalize your tables so that no redundant information is stored. Design a single SQL query to retrieve all information for updating the check-in list (the list from the previous bullet). Explain your query and how you designed the database in your report. (15 points)

2) 25 points maximum

Add another activity to the same app to show a separate map view of the locations using the Google Maps Android API. Follow the directions at (https://developers.google.com/maps/documentation/android-api/) to install the Google Play services SDK, acquire a Google Maps API key, and display it within your app.

- The map should be centered on your own location. It should include standard UI controls: zoom in/ out buttons, Compass, "My location" button. (10 points)
- Show all of the check-in locations on the map with markers (such as small dots). (5 points)
- When a user clicks on the map, enable the user to add a new named location with a marker on the map. The marker should be draggable. The user should be able to name it. The new named location should be added to the database (without recording a check-in at this location) (10 points)

4) 30 points maximum

Quality of the report and UI design. (10 points)

Graduate course additional requirements (up to -20 for graduate students who do not complete this):

1) Install your app on a smartphone. Modify it to log the delay and accuracy of the location updates from GPS and network location. You can do this, for example, by manually enabling either GPS or network localization. Experiment with the app in at least 5 indoors and 5 outdoors locations. Estimate the accuracy in meters based on a map of the area and plot the results in a graph. Prepare a report about your findings (this is in addition to the 1 page report required for all students). The report should include the accuracy reported by the location API, your estimated accuracy, and the measured delay. Think about the delay and accuracy with each localization methods and explain your findings. (20 points)

Upload your solution source code, apk package and report in the same zip or tar.gz file named as firstname_lastname. The apk named as the name defined in manifest for the main activity. The apk file should be copied to the root directory of the submission folder, also keep the original apk in the bin folder.