Location Lab

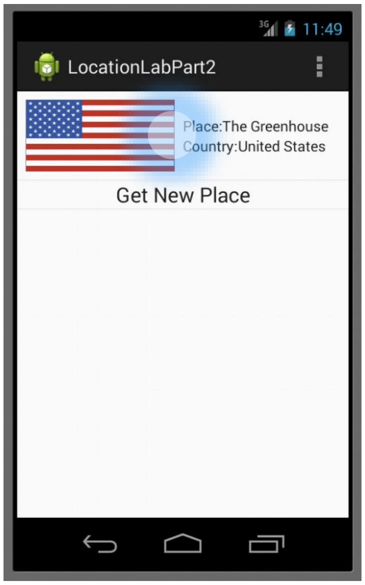
Use Location information within your app.

This week's lab will help you learn how to use location information in your Android applications. Upon completion of this lab, you should have a better understanding of how to listen for and respond to Location measurements.

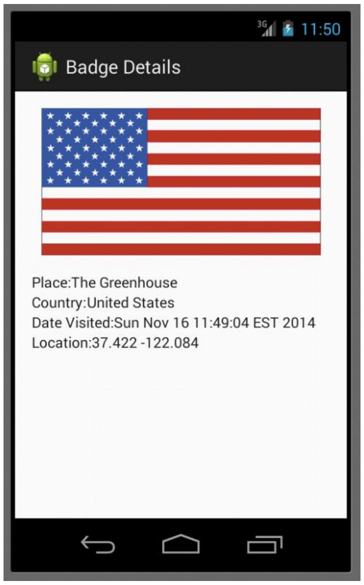
This application displays a ListView containing a set of Place Badges and you get the place's coordinates from the phone's Location measurements.

This week you will make it so that the application receives locations from the device itself. To implement this functionality, you will need to acquire location readings from Android. Exactly how you implement this is up to you, however, your app will need to listen for location updates from the NETWORK_PROVIDER (which we will control for testing purposes) and the GPS_PROVIDER (so you can enter locations via the emulator). You can also inject locations into the app, by clicking on the menu and selecting one of the menu items, specifically "Place One", "Place Two," or "Place No Country."

In addition, when the user clicks on a Place Badge, the app will open a detail view showing additional information about the place.



Place:The Greenhouse
Country:United States



Place:The Greenhouse
Country:United States
Date Visited:Sun Nov 16 11:49:04 EST 2014
Location:37.422 -122.084

Testing:

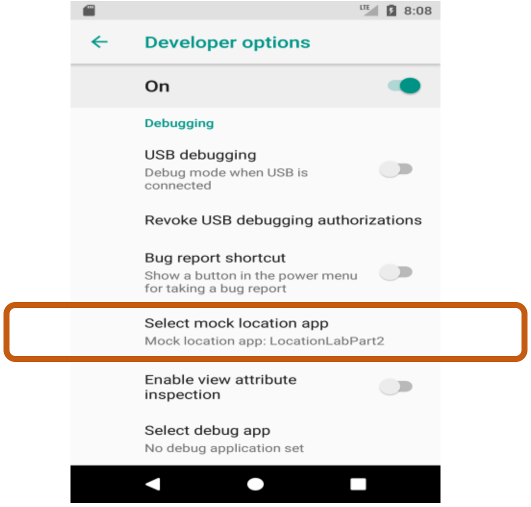
The test cases for this Lab are in the Lab10_LocationLab project. You can run the test cases either all at once, by right clicking the test package and then selecting Run ‘Tests in ‘course.lab…’, or one at a time, by right-clicking on an individual test case class (e.g., TestNoCountryLocation.java) and then continuing as before. The test classes are Robotium test cases.

Warnings:

1. These test cases have been tested on a Galaxy Nexus AVD emulator with API level 26. To limit configuration problems, you should test your app against a similar AVD.

2. You will need to go into your device's Developer Options. On Android 4.1 and lower, the Developer options screen is available by default. On Android 4.2 and higher, you must enable this screen as follows:
    a. Open the Settings app.
    b. (Only on Android 8.0 or higher) Select System.
    c. Scroll to the bottom and select About phone.
    d. Scroll to the bottom and tap Build number 7 times.
    e. Return to the previous screen to find Developer options near the bottom.

3. You will need to make sure that you've chosen "Select mock location app" as the testing app from the Developer Options.



Developer options

On

Debugging

USB debugging
Debug mode when USB is connected

Revoke USB debugging authorizations

Bug report shortcut
Show a button in the power menu for taking a bug report

Select mock location app
Mock location app: LocationLabPart2

Enable view attribute inspection

Select debug app
No debug application set

4. The application screenshots shown above require a working network connection. You will also need to create an account at http://www.geonames.org/login. Your username will need to be updated in PlaceDownloaderTask.java. If you do not have a working network connection, you can still use the app, but the flag images will use a simple placeholder.

Submission

To submit your work, you will need to commit your solution to your repo on GitLab by running the following command: git push origin master.
Note: if you have not already pushed this branch to your repo on GitLab you will need to make a slight modification for this first time and run this instead: git push –u origin master. This sets up tracking between your local branch and a branch with the same name on your repo in GitLab.