5COSC005W MOBILE APPLICATION DEVELOPMENT
Lecture 11: Maps, Location and Runtime Permissions

Dr Dimitris C. Dracopoulos

Module Web page:

https://dracopd.users.ecs.westminster.ac.uk/DOCUM/courses/5cosc005w/5cosc005w.html

Get the Last Known Location

Steps (set up):

1. Obtain an Google Maps API (for usage with Maps) and insert it in the `res/values/` (follow the instructions in the same file for how to obtain the API key).

2. Add a **dependency** of Google Play location services by adding the following line in the `build.gradle` module file (where `XX.X.X` is the latest version for Google Play Services:

   ```
   implementation 'com.google.android.gms:play-services-location:XX.X.X
   ```

3. Add the `ACCESS_FINE_LOCATION` permission in the manifest file of the project:

   ```
   <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
   ```

Get the Last Known Location (cont'ed)

Steps (Java code):

1. Check if the permission is granted by the user, otherwise request the permission by calling ActivityCompat.requestPermissions.

2. Implement in your activity the onRequestPemissionsResult() callback method which will receive the permissions result.

3. Create a FusedLocationProviderClient object:
   `mFusedLocationClient = LocationServices.getFusedLocationProviderClient(this);`

4. Call getLocation() on the FusedLocationProviderClient object returning a Task object.

5. Call addOnSuccessListener() method on the task and pass it an object which implements the OnSuccessListener<Location> interface.

How to Receive Location Updates

Steps (Java code):

1. Create a `LocationRequest` object containing the requirements of the request (update frequency, accuracy).

2. Create a `LocationCallback` as part of the activity and override its `onLocationResult()` method which is called periodically with the location updates.

3. Call `requestLocationUpdates()` on the `FusedLocationProviderClient` object and pass it the `LocationRequest` and the `LocationCallBack` objects.