The purpose of this lab is to look at what Android provides for letting apps detect the location of the device. i.e. to help look at making apps **location aware.**

**Please refer to the lecture for the concept of Location Provider, Permissions, Location Listener, minTime, maxTime.**

1. **Understanding GPS coordinates**

Every location on earth is identifiable by a GPS coordinate (latitude, longitude). Use Google Maps web page to find the GPS coordinates of **Kevin st.**

1. **Implement location tracking**

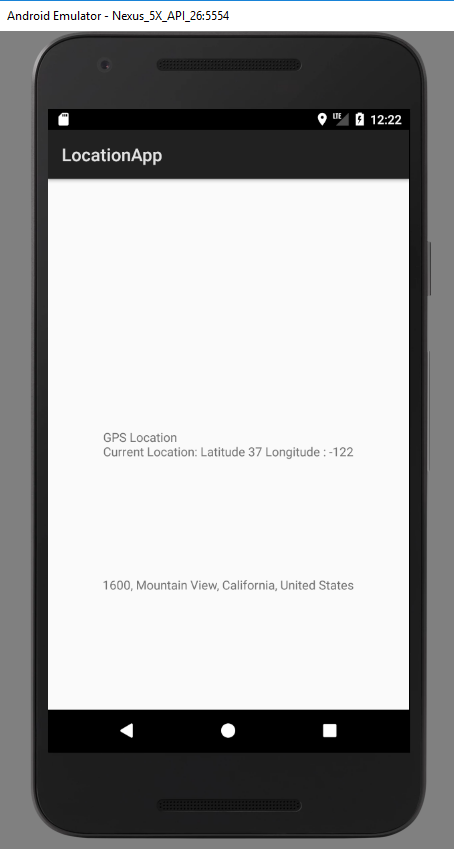
Implement the activity code from class (locationTest1) to implement a screen that uses location tracking to display the current latitude and longitude in a textview on the Screen. You need to write your own screen layout.

Watch out for:

* Permissions – run time (as per code) AND manifest file
* importing ActivityCompat/ ContextCompat classes - may have to add these from your project
* Mocking up location if you are using an emulator. See diagram at end of lab on how to do this . If you are using a window, go near the window!

**3 Use GeoCoding to “translate” Lat, long into an address.**

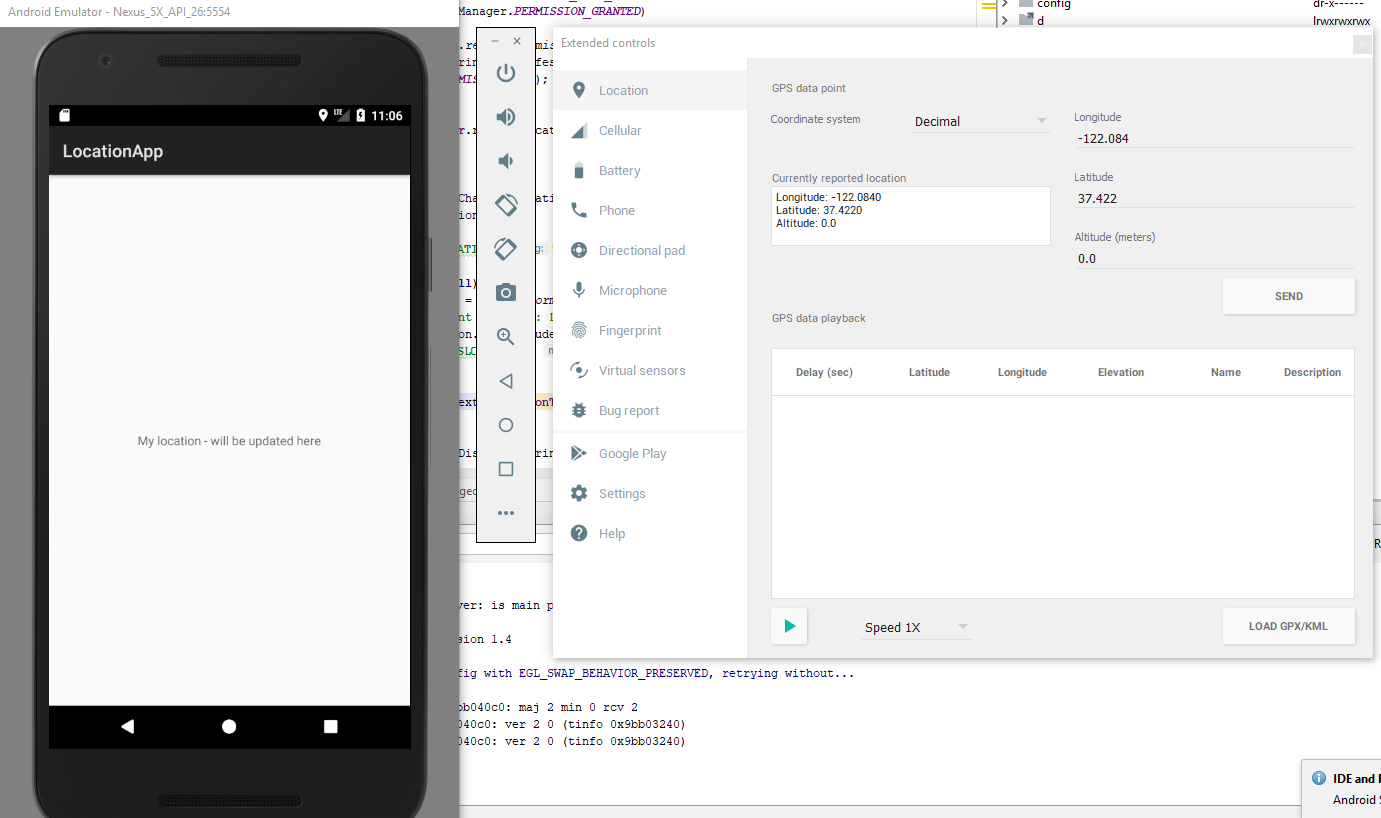
Add a second textview to your layout. Use the GeoCode class to translate the lat, long into address and display that on the screen in the 2nd textview, as shown.



**4 Saving on the battery using Lifecycle methods**

You can save on the battery usage by (1) minimising the frequency of checking for location changes to the minimum needed for your app AND (2) using the activity lifecycle methods to switch the location checking on and off.

Add code to the relevant lifecycle (onPause(), onResume() methods so that you switch location checking on and off if your activity goes from and back to the foreground. Add appropriate LOG message (e.g. “Just dropped location checking when activity gone into background”) to go into the LogCat file so that you can check it’s working. Test by receiving a call on your phone OR on the AVD, sending a fake call to the emulator.



Mocking up location on the emulator