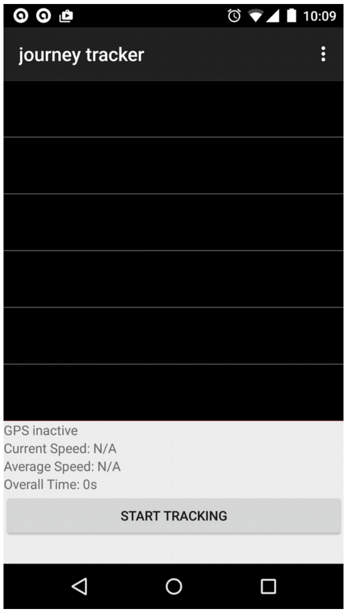
Assignment 03: building a GPS tracker that will analyse user speed

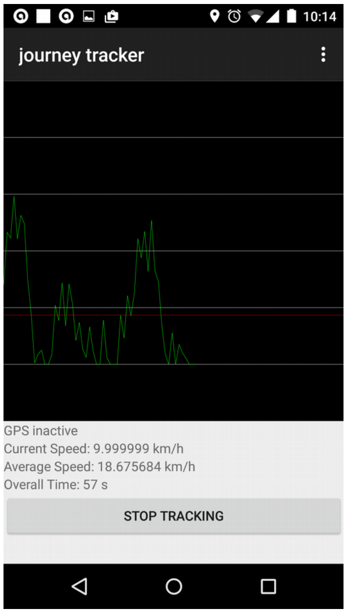
introduction:

Note: read the whole assignment brief first before implementing it contains very important information.

In this assignment you will be tasked with building a GPS tracker application that will require you to master the use of a custom view and also the GPS facilities of an android device. This application will be used to track the user's speed as they travel on a journey. In this case the scale we will use is representative of a car travelling through a city. The speed scale will cover from 0 km/h to 60 km/h. In the screen shots below you will see the initial state on the left and the running state on the right.



GPS inactive
Current Speed: N/A
Average Speed: N/A
Overall Time: 0s



GPS inactive
Current Speed: 9.999999 km/h
Average Speed: 18.675684 km/h
Overall Time: 57 s

Notes:

In this application you will be required to have a single activity with a custom view, three textfields and also a button to start and stop tracking. The three textfields contain: the current speed, average speed, and the overall time for the user. You will be required to keep track of the last 30 gps locations of the user in order to plot the graph.

As for the graph itself you will be required to draw three things. First the white lines you see denote grades of 10 km/h so you should have lines for {10, 20, 30, 40, 50, 60} km/h. Second you will need a red line that will denote the average speed for the whole journey. This will be a straight line across the graph. Finally you will need a green line that will be a trace detailing the speed of the user over the last 30 samples. Note that the line at 30 samples must cover the entire width of the graph. NOTE that in the screenshots above my graph is configured for 100 samples but I can easily change this to 30

If you are using an Android device to develop this application I would suggest that you enable mock locations in your developer settings also install this mock locations app to generate a route with varing speeds for you https://play.google.com/store/apps/details?id=ru.gavrikov.mocklocations you will find this particularly useful because it means you will not have to go and do your testing outside.

As you will be using Android Studio for development you will be required to use a local git repository for your code. You will be required to have at least two commits in your repository: one for project creation, and one for the finished project and at least four commits in between to make a minimum total of 6 commits. This is beneficial for two reasons: 1) it provides a backup of your code 2) if you mess up your code you can rollback to a previous working version and start again. The name and email you use for the author must match the name and email you have on the moodle.

Take note with the milestones. If you skip a milestone or have serious errors with a milestone, further milestones will not be considered for correction. e.g. if you skip milestone 03 or have bad errors with it then milestones 04,05,06,…. will not be considered for correction. With regards to errors, depending on their severity I will either deduct 5, 10, or 15% for each bug. Thus it's possible that if you implement all milestones but have 20 small bugs you can end up with a score of 0%. This is to encourage you to build robust software free from bugs.

No structure will be provided for the application so it is up to you to determine what datastructures and algorithms you will need to complete this assignment.

For submission you are required to upload an archive of your complete android studio project to the moodle by 2016-05-01 at 23:55:00. The accepted archive formats are the following six: 7z/zip/rar/tar.gz/tar.bz2/tar.xz.

Penalties:

There will be penalties applied to your score if any of the following conditions are not met. These are silly things that slow down my correction of your work. Thus if you want your result back in the quickest possible time make sure you do not fall foul of these penalties.

- - 40% for a missing git repository. Make sure when you go to submit your work that there is a .git/ directory in your project before archiving. I like to see development history on assignments to see how it was made. You will also lose marks for missing commits.
- - 30% for non compiling code. I should be able to open your project have it compile and run immediately. You are final year students non compiling code is a grave sin at your level.

- - 10% for a non supported archive format. I have a python script that can unpack and sort all projects downloaded from moodle. Anything not in those archive formats I have to unpack and sort by hand.
- Standard late penalties: submission even a second late after 23:55:00 on submission day will have the late submission penalties applied.

Milestones:

As noted before make sure that you complete milestones properly before moving onto the next one. The percentage in brackets indicates the maximum score you can achieve if you pass that milestone.

01) Generate a layout and an activity that contains the custom view of the GPS tracker. (5%)

02) Hook up the button to a listener that will start and stop GPS tracking. The text on the button must change to reflect if it is tracking or not. (10%)

03) Add in a LocationListener that will sample the GPS every second. For each sample it collects it should add it into a datastructure of your choosing. If there are more than 100 samples after the most recent was added then remove the oldest sample (20%)

04) Add in a custom view for the graph, force it to be square in size and to have the while lines running across it. (30%)

05) Draw the trace on the graph. The speed for each point must match up with the graph on the vertical axis. i.e. if the point is 10 km/h that point should be drawn on the 10 km/h line. Also when you have 100 samples they should create a line that takes up the entire width of the graph. (80%)

06) calculate the current time, average speed and current speed and update the text fields after every sample. (100%)