

# ECE 150/251 – Fall 2021

## Mobile Embedded Systems

### Homework Assignment 4 – Essay / LocationPlus

---

Deadline: **Friday, Nov 19, 2021 @ 9:00 PM**

## I. Introduction

For the final homework, ECE 150 students have a choice between writing a research-based essay or working on another Android application. **ECE 251 students are required to do both.**

## II. Essay Requirements

Write a technical / survey / summary paper about **one** of the following topics:

- Wifi 802.11ax
- Changes in ARMv8-A *vs.* ARMv7-A
- Compare Bluetooth 5.0 and BLE
- Low Power DRAM (LPDDR)
- 5G Cellular Standard
- Assisted GPS in GNSS

Find relevant information online or in textbooks and summarize your understanding in **your own words**. The requirements for the essay are as follows.

- Have a minimum of 3 pages with 10pt font size.
- Begin with an abstract, include an introduction followed by sections of your choice, and end with a conclusion. Cite all references used.
- **Only** PDF format will be accepted.

## III. Android App – LocationPlus

For this homework, you will be implementing LocationPlus, an application built using Google Maps and Location Services to track the user's current position, receive destination arrival notifications, and view information about satellites that were used in determining the last position fix.

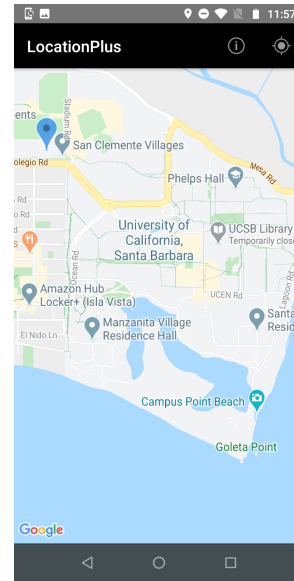
### Task Specifications

Your app must be able to perform the following:

1. The app must display the user's current location on the Google Maps fragment at all times.
  - (a) There must be a marker that indicates your current position on the map. For example, the blue marker in Figure 1 indicates the user's current location.
  - (b) You must implement a button (on the Action Bar) that toggles auto-centering.
    - When enabled, the camera should automatically move (on each location update) such that your current location is centered on the screen. See Figure 1a.
    - When disabled, the user should be able to move freely around on the map. However, *the user's current location should still be updating*. See Figure 1b.



(a) Auto-Centering Enabled



(b) Auto-Centering Disabled

Figure 1: User Location Tracking

2. Several satellites are used for calculating a position fix on the surface of the earth. The app must provide details about the satellites that were used in computing the most recent position fix.

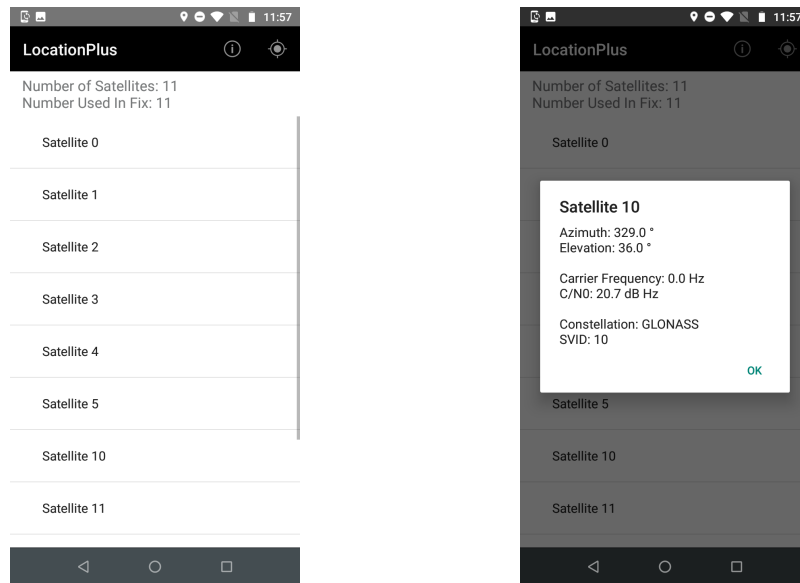
- (a) Implement a button (on the Action Bar) that reveals another view which shows the number of satellites detected, the number of satellites that were used in the last fix, and the list of satellites. The list of satellites should be maintained using an adapter (as in Homework 3 – GauchoPay) because the list of satellites may change with each location update. See Figure 2a.
- (b) Clicking on a satellite in the list should display a dialog box with detailed information about that specific satellite. The dialog box should display the following information:
  - Azimuth (in  $^{\circ}$ )
  - Elevation (in  $^{\circ}$ )
  - Carrier Frequency (in Hz)
  - Carrier-Noise Density  $C/N_0$  (in dB Hz)
  - Constellation Name
  - SVID

See Figure 2b.

3. The app must be able to keep track of a user-set destination and alert the user upon arrival.

- (a) The user can set a destination by performing a long click (i.e. click and hold) on *any* location on the map. Upon a long click, the user is asked to confirm whether they want to set the selected location as the destination. See Figure 3a.
- (b) When the destination is set, a Geofence should be set up that listens for certain transitions. Furthermore, a Floating Action Button should become visible at the bottom, which provides a method of canceling the current destination (and Geofence) with a prompt asking the user for confirmation. See Figure 3b.
- (c) When an entry into the Geofence is detected, we can say that the user has arrived at the destination. Upon arrival, the destination (and Geofence) are cleared and a notification is sent to the user. See Figure 3c.

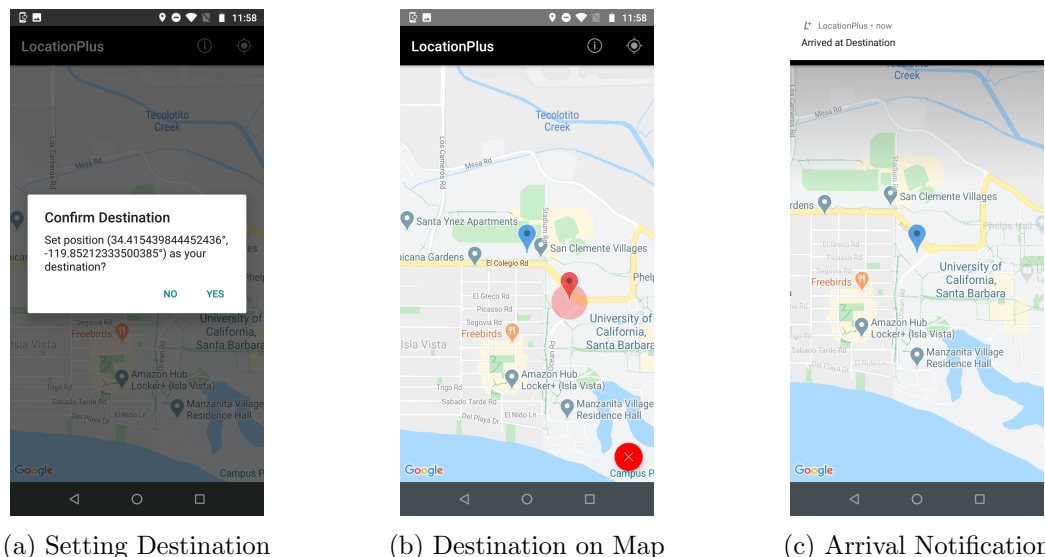
- (d) When the destination is set, the destination should have a marker (with a different color from the marker that indicates the user's location). In addition, the region encircled by the Geofence should also be colored. For example, the red marker in Figure 3 displays the destination position and the Geofence.
- The app must operate correctly in both portrait and landscape mode.
  - The app should not have data loss.



(a) Satellite List

(b) Detailed Satellite Information

Figure 2: Viewing Information About Satellites



(a) Setting Destination

(b) Destination on Map

(c) Arrival Notification

Figure 3: Destination Tracking

## Mocking GPS Location

To ensure that your app works for any arbitrary location, we will ask for you to mock locations during the demo. To mock GPS locations, you can use several different apps that are available on the Play Store. A recommended app is [Fake GPS Location](#).

## IV. Submission

Upload your Android project (as a .zip file) using the turn-in link on Gauchospace. There will be individual (online) check-offs in the week after code submission. The TAs will email out a sign-up sheet for the check-offs.

## V. Grading

The homework will be graded according to the following rubric.

- [10%] Successful integration of Google Maps into your application.
- [30%] Continuous updates to reflect the user's current location.
- [30%] Implementation of a satellite list and displaying individual satellite information.
- [30%] Implementation of destination tracking.

## VI. Resources

1. [Android Location Package](#)
2. [Creating and Monitoring Geofences](#)