

**CS 5551 Advance Software Engineering**

**FIRST INCREMENT REPORT**

**Locus**

**Team#7**

**Venkatesh Pallay**

**Sowmith Reddy Pentaparthu**

**Rohit Nagulapati**

**Abhilash Reddy Gaddam**

# **Project Proposal**

## **Project Title: Locus**

### **PROJECT GOAL AND OBJECTIVES:**

#### **MOVITATION:**

How important is your vehicle to you? Do you treat it as your family member, if so what if your friend requests for the car to go for a ride with his girlfriend? Don't worry if you have "Locus" app with you anymore; you can track your vehicle anytime, anywhere wherever you are. Also, when you are traveling, the app reminds your favorite locations near to you.

#### **SIGNIFICANCE/UNIQUENESS:**

Our investigation for applications which provides location based services like reminders for favorite spots near current location, tracking vehicle, didn't fetch great results. This made us thinking about creating an application with advanced technologies that brings in real time and accurate information available to users all the time.

#### **OBJECTIVES:**

The objective of our web/ mobile application is to track their loved vehicle anytime, anywhere if shared to anyone till it reaches to him. Also, notifying his/her favorite spots available near to his current location when in travel. Our best efforts appear in presenting the most accurate and realtime data available to users all the time.

#### **SYSTEM FEATURES:**

- The user can track where the vehicle is moving.
- The user can estimate the arrival time of the vehicle.
- The user can find the distance traveled by the vehicle.
- The user can receive notifications about his favorite spots near his current location.

As a part of our project's 1<sup>st</sup> increment, we have implemented the Login page, a register page and on successful login redirects to the Home page. At first, a new user needs to register and can login using his name email and the password. Here we will be using the Google Maps location API to get the current location of the user.

## FEATURES:

1. Google Location Services: Google's location services is implemented as two features i.e Location Reporting, and Location History:
  - (1) Location Reporting is the feature that shows you places nearby, suggests local businesses, or helps you find the favorite spots by tracking the device.
  - (2) Location History is the feature that keeps track of where you've been, tracks the moving vehicle and for estimating the distance traveled by car using the smartphone.
2. Push Notification: Once the user sets his favorite spots, in case if there is any favorite spot nearby then a notification will be given to his device.
3. Map Locations: We are integrating the google maps to our application so that the user can view the exact location on the maps and locate the favorite spots nearby and also for tracking the vehicle.
4. Gesture Recognition: The gesture recognition API allows to register callback functions to be called when the user performs meaningful gestures like shaking the device which helps you display your favorite spots.

## SERVICE DESCRIPTION:

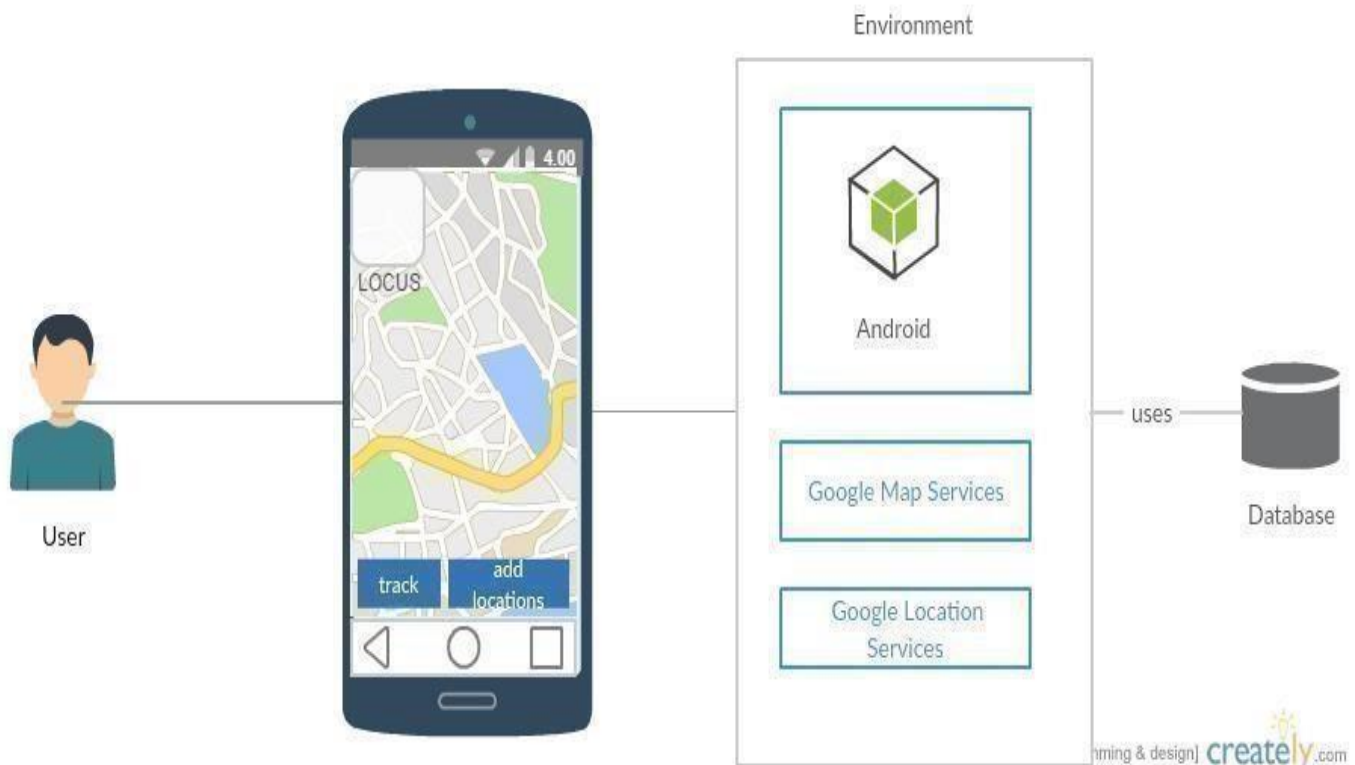
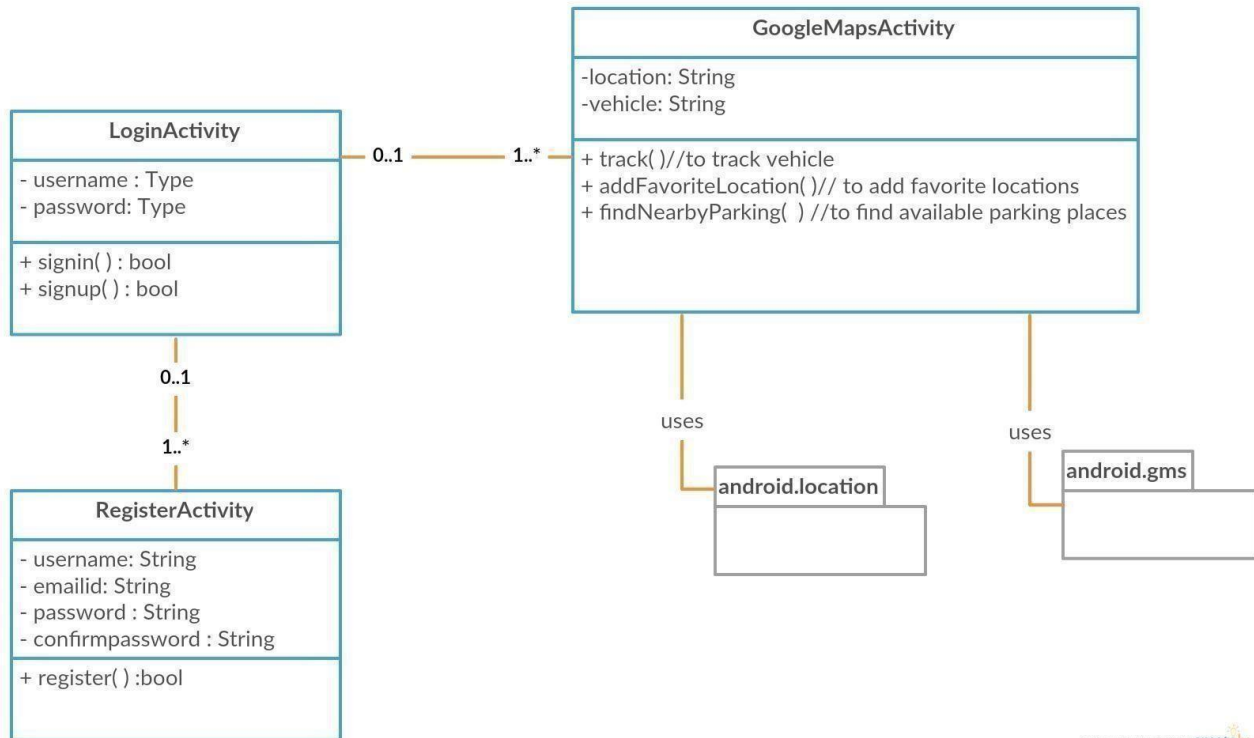
1. Maps: Maps information is displayed using the Google Maps API which gives us the information about the houses in a locality.
2. Gesture Recognition: The gesture Recognition API is used to pin the favorite spots of the user using the Gesture Recognition Toolkit(GRT).
3. Locations: To get the auto locations fill we will be using the Auto Location fill API.

## GITHUB URL:

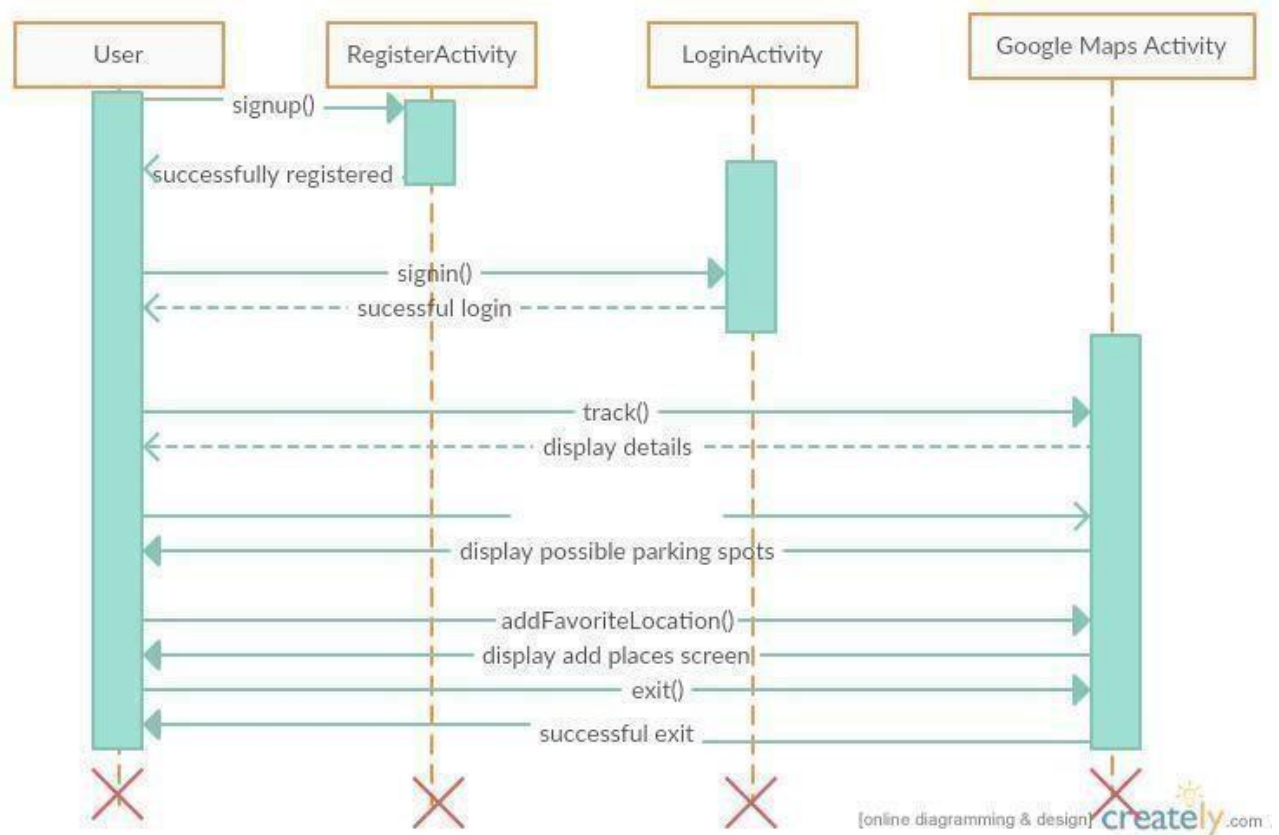
[https://github.com/PallayVenkatesh/ASE\\_Project](https://github.com/PallayVenkatesh/ASE_Project)

## ARCHITECTURE DIAGRAM:

# CLASS DIAGRAM:



# SEQUENCE DIAGRAM:



# ISSUES TOOLBAR:

PallayVenkatesh / ASE\_Project

Watch 2Star 0Fork 0

<> Code

Issues 3

Pull requests 0

Boards

Reports

Projects 0

Wiki

View

Repos (1/1)

Show one

Labels

Milestones

Assignees

Epics

0

0

In Progress 1

0

Review/QA 0

0

Done 0

0

Closed 4+

ASE\_Project #1

Project report

Increment - 1

ASE\_Project #3

Code and Documentation

Increment - 1

ASE\_Project #5

Test Cases for Login and Registration Page

Increment - 1

ASE\_Project #2

UML Diagrams

ASE\_Project #4

Create Wire Frames

Load more issues...

Add a Pipeline ...

This repository

Search

Pull requests

Issues

Gist

ToDo

PallayVenkatesh / ASE\_Project

Watch 2Star 0Fork 0

<> Code

Issues 0

Pull requests 0

Boards

Reports

Projects 0

Wiki

Filters

is:issue is:closed

Labels

Milestones

New issue

Clear current search query, filters, and sorts

0 Open

5 Closed

Author

Labels

Milestones

Assignee

Sort

Test Cases for Login and Registration Page

#5 by PallayVenkatesh was closed a minute ago

Increment - 1

1

Create Wire Frames

#4 by PallayVenkatesh was closed 2 hours ago

1

Code and Documentation

#3 by PallayVenkatesh was closed a minute ago

Increment - 1

1

UML Diagrams

#2 by PallayVenkatesh was closed an hour ago

1

Project report

#1 by PallayVenkatesh was closed 12 seconds ago

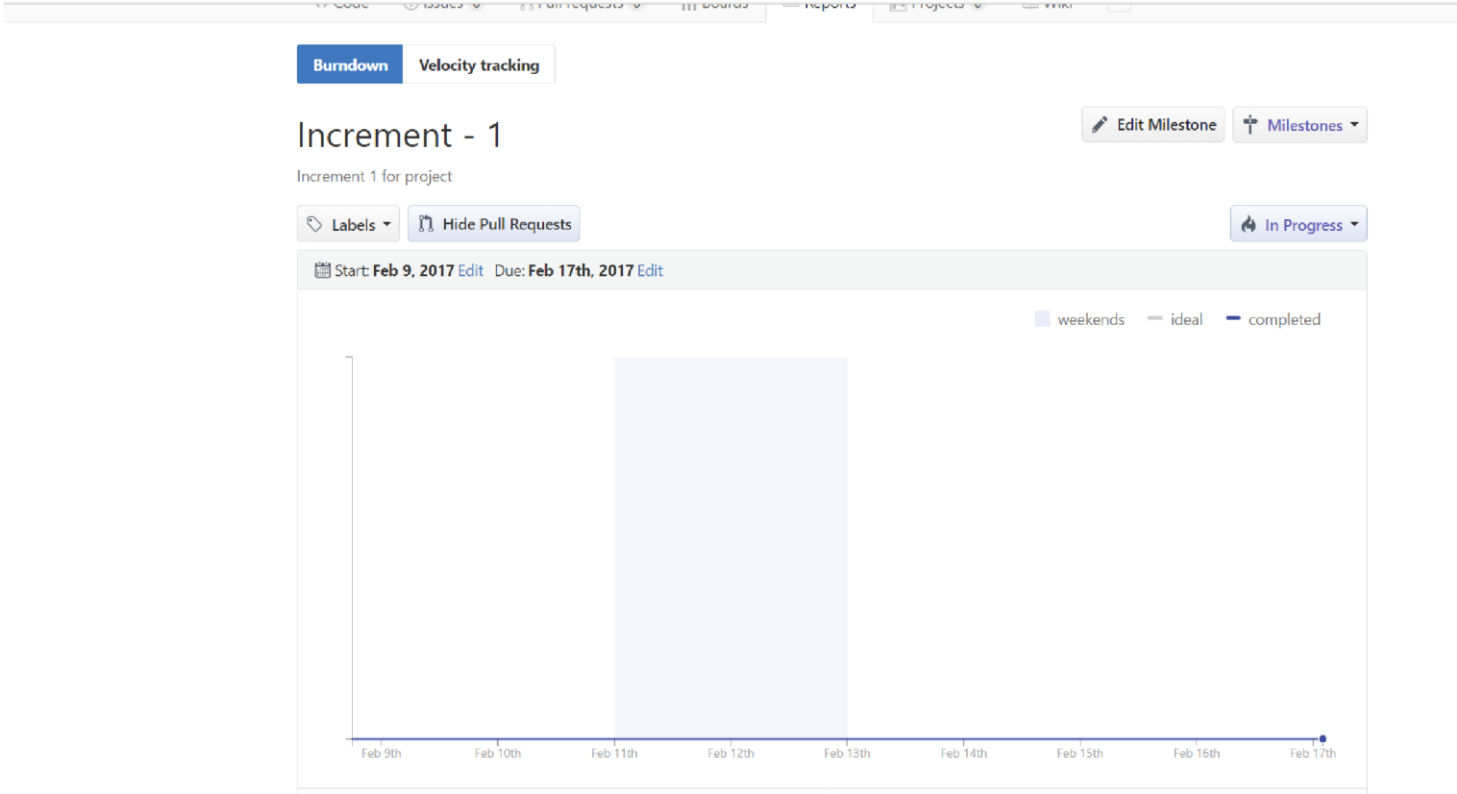
Increment - 1

1

ProTip!

Adding no:label will show everything without a label.

# BURNDOWN GRAPH:





# WIREFRAME

## LOGIN PAGE



## REGISTRATION PAGE:

LOCUS

Register Here...!

Please enter Username

Please enter Email Address

Please enter Password

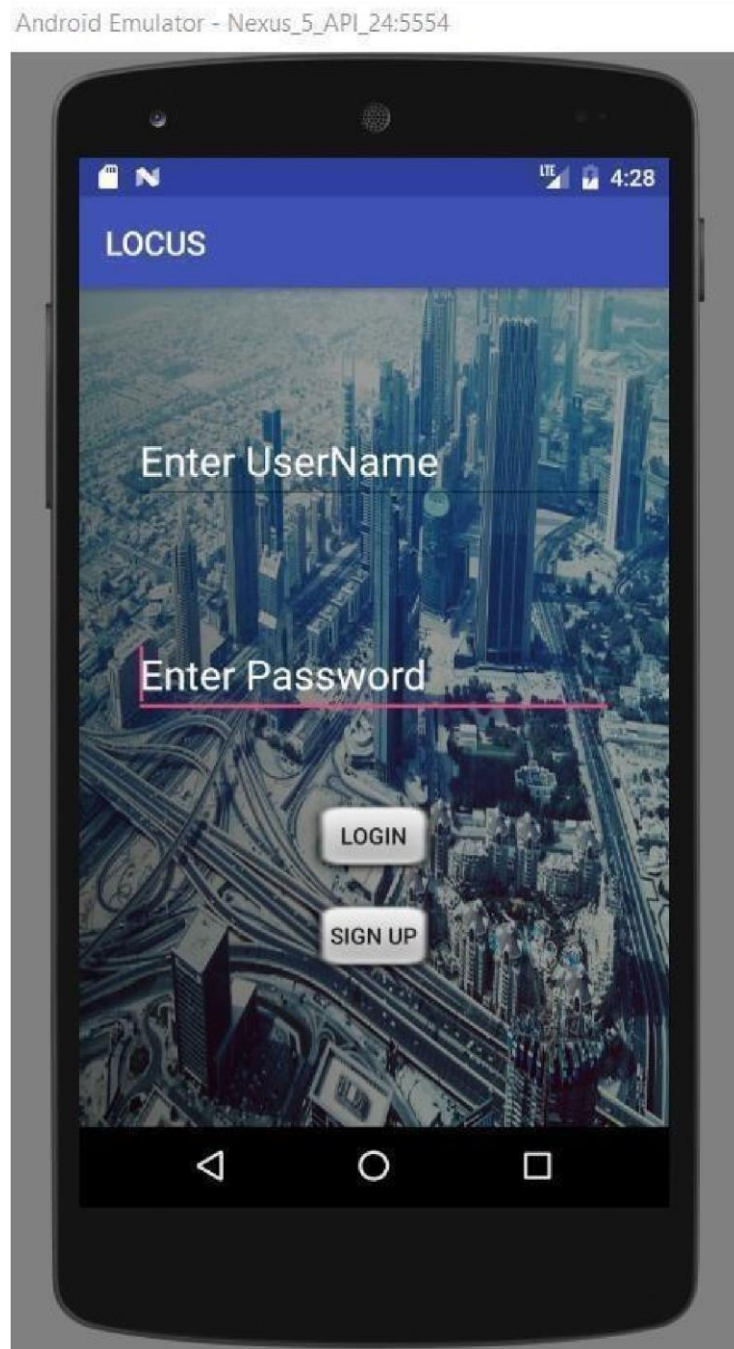
Confirm Password

☐ I agree to the terms and conditions

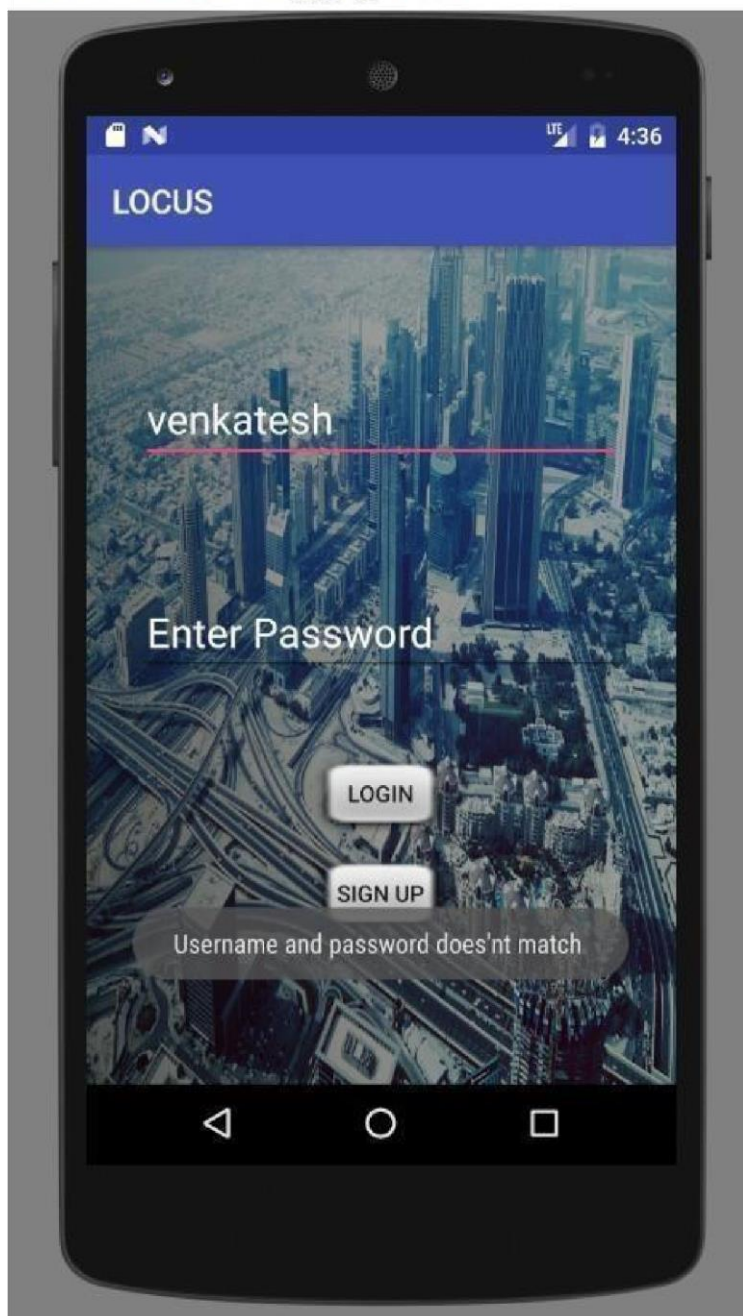
register

# MOCK-UPS:

## LOGIN PAGE:



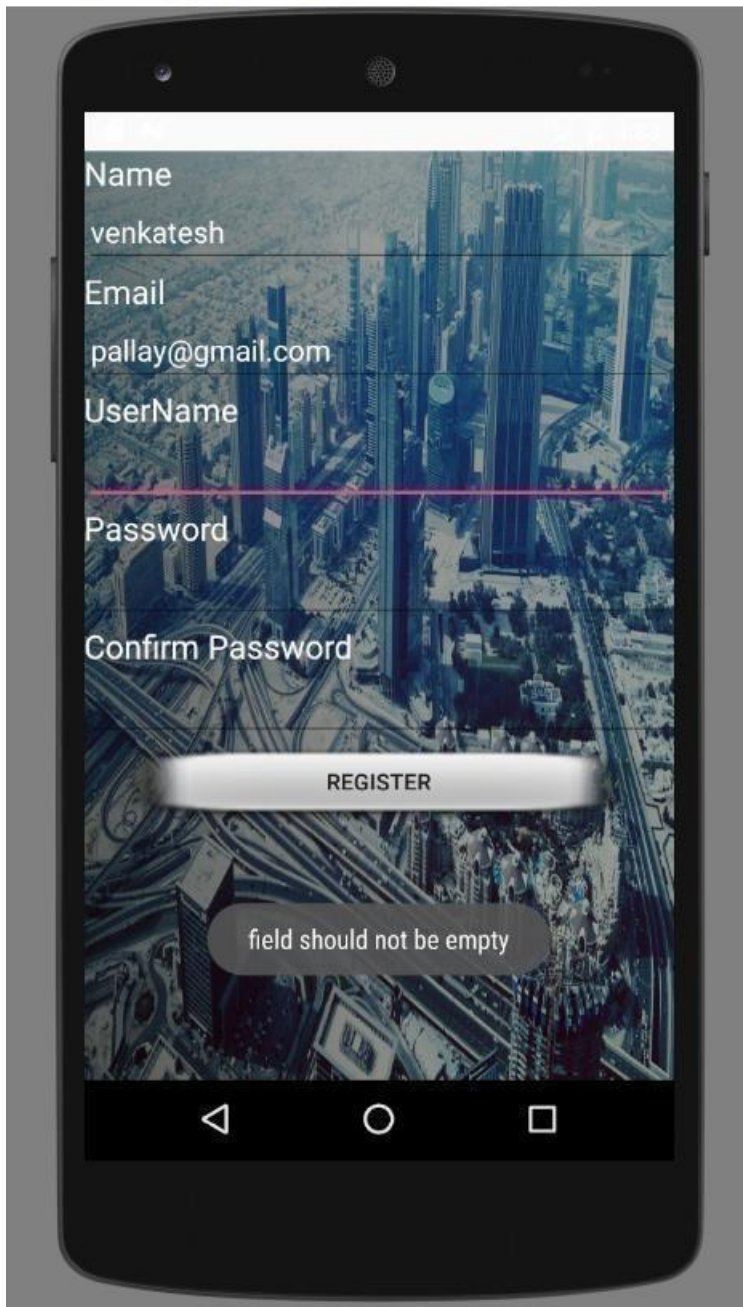
## LOGIN PAGE VALIDATION:



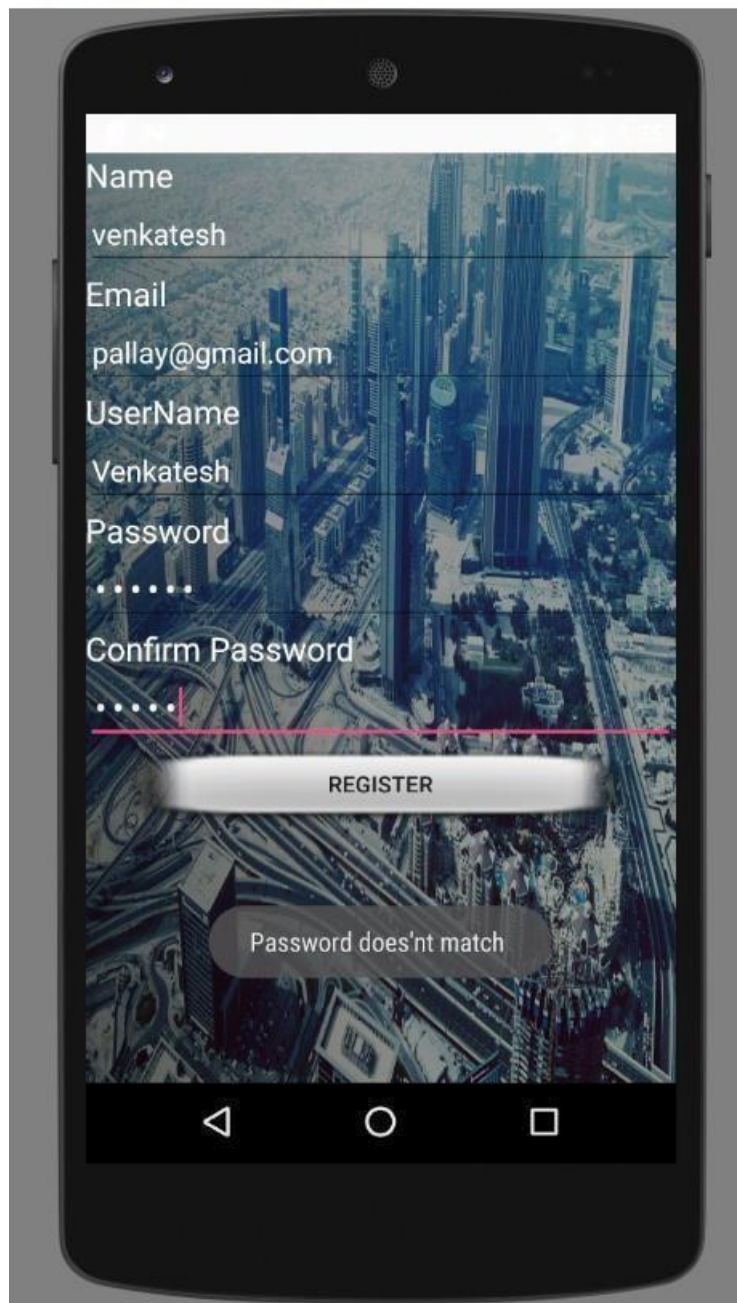
REGISTER PAGE:



REGISTER PAGE VALIDATION:







## TEST CASES:

Test Case			
Name	Test Description	Expected Results	Pass/Fail

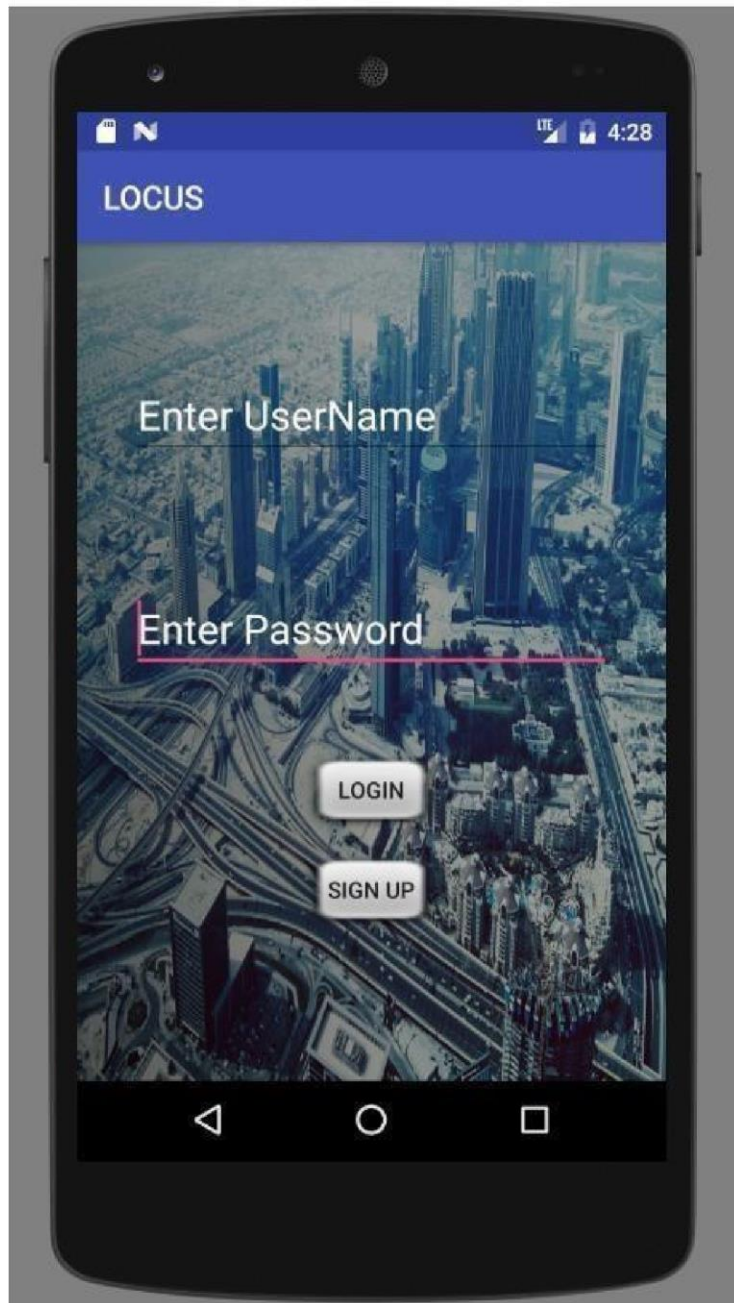
Login	Enter Invalid Userid and Invalid Password	Invalid Login Error Message should be displayed	Pass
	Enter Valid Userid and Invalid Password	Invalid Login Error Message should be displayed	Pass
	Enter Valid Userid and Valid Password	Application Should Be Redirected to Home page	Pass
Sign Up	Enter Email Id without @	Invalid Email id should be displayed	Pass
	Enter different confirm password	Invalid Error Message should be displayed	Pass
	Blank Spaces	Invalid Message should be displayed	Pass

**FINAL SCREEN SHOTS:**

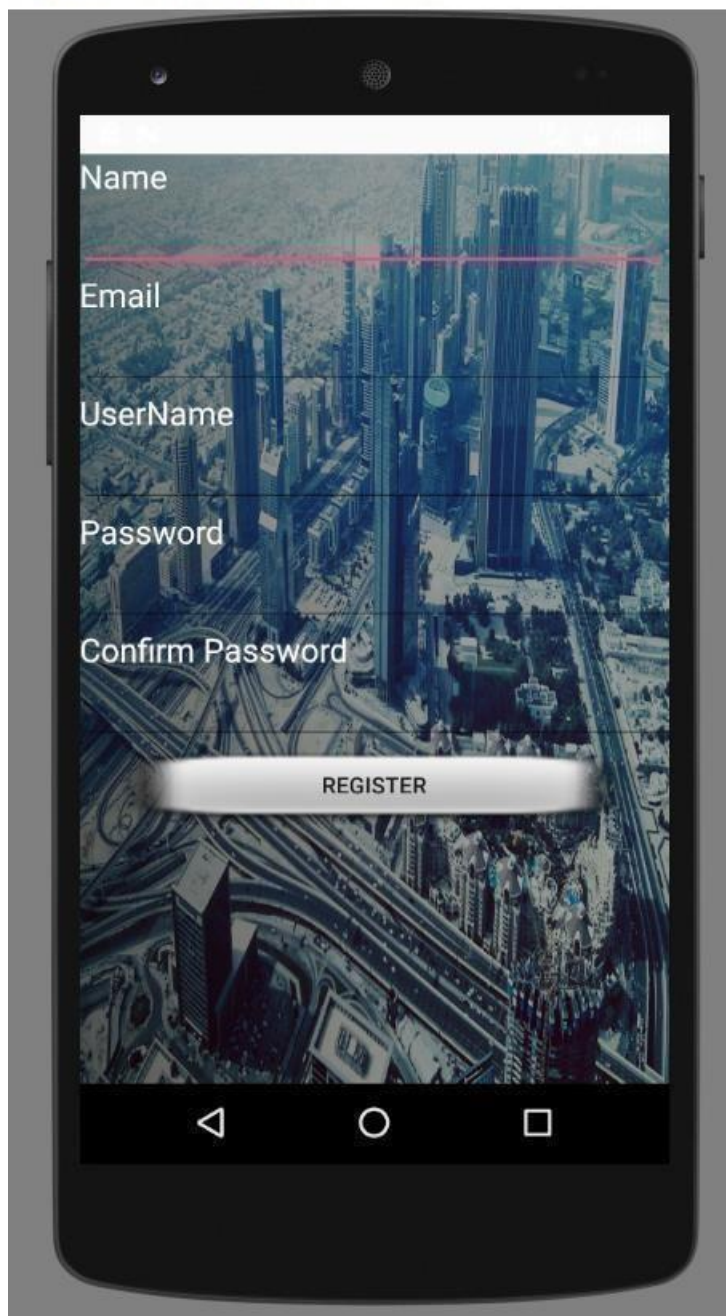


## LOGIN PAGE:

Android Emulator - Nexus\_5\_API\_24:5554



## REGISTER PAGE:



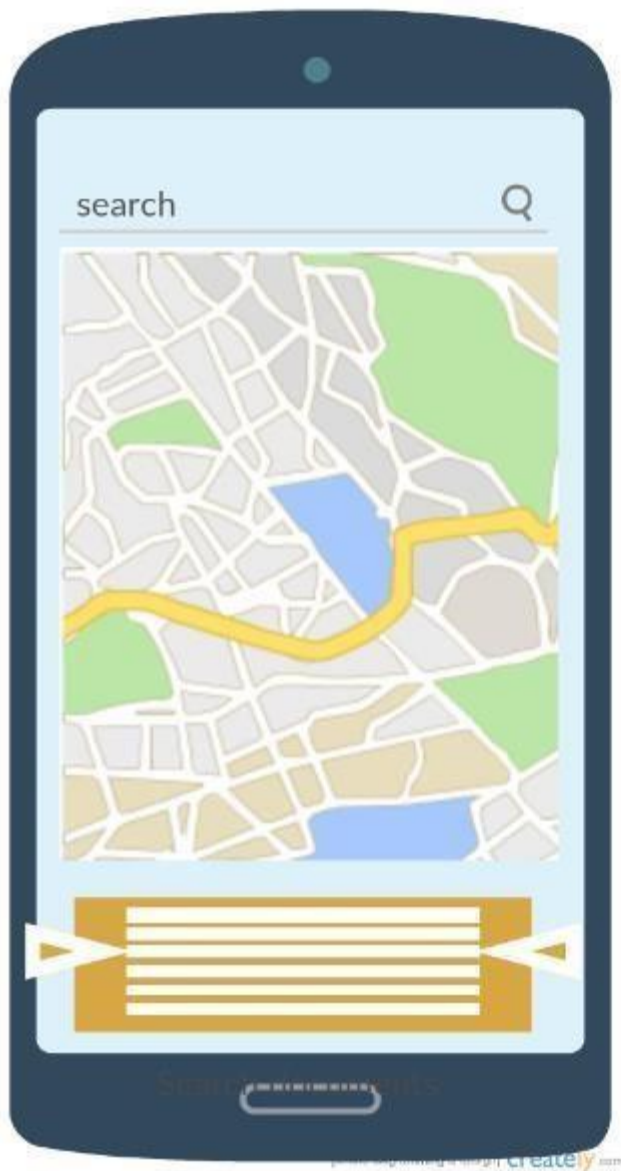
# INCREMENT 2(FAVORITE LOCATION MANAGER and Current Location Updation)

PallayVenkatesh edited this page 21 minutes ago · 3 revisions

## Location Manager Module

### WireFrame Design

#### Map intent Design



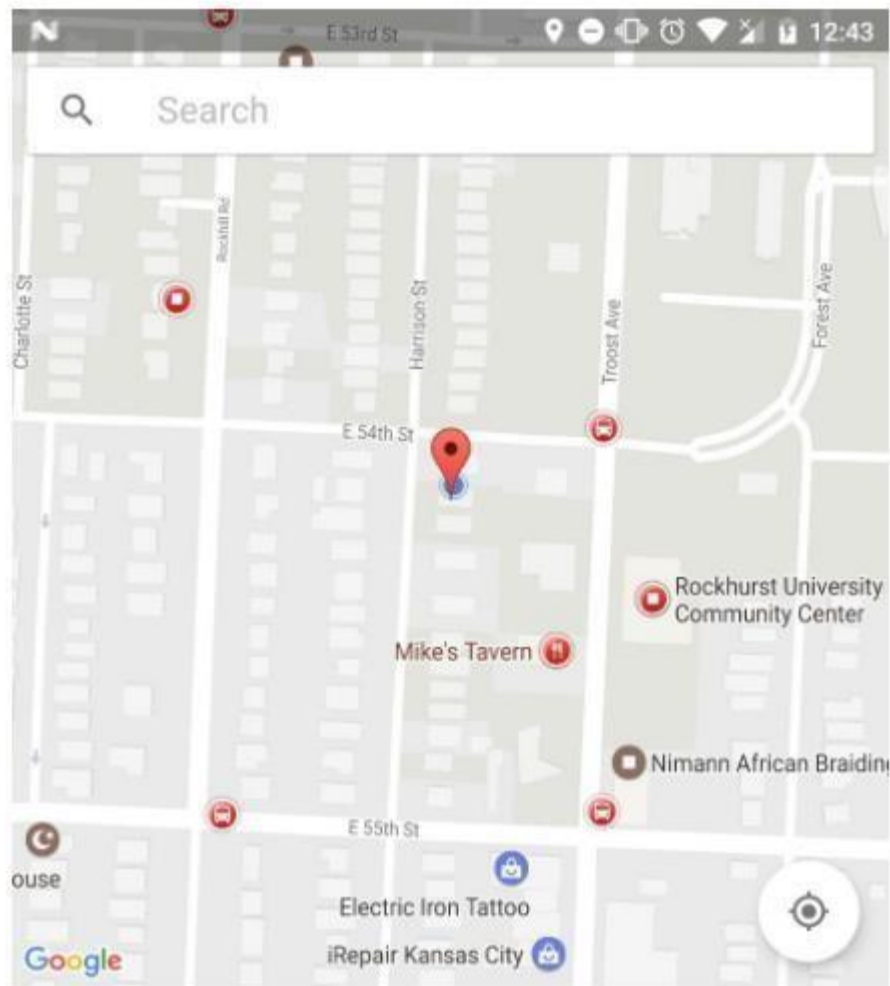


## Favorite Location List Design



## Output Screens

### Place Marker Intent



Select this location

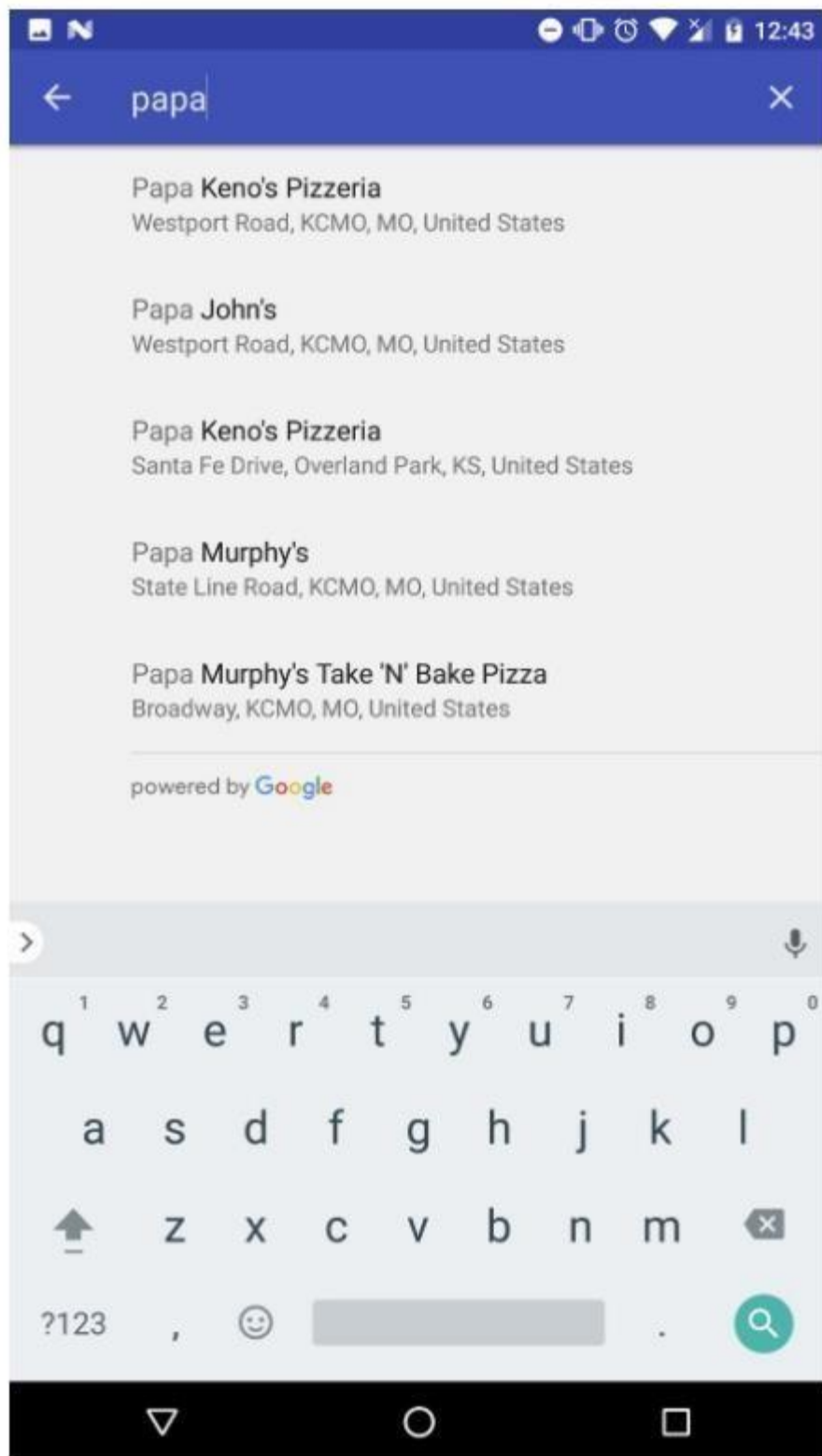
Or choose a nearby place

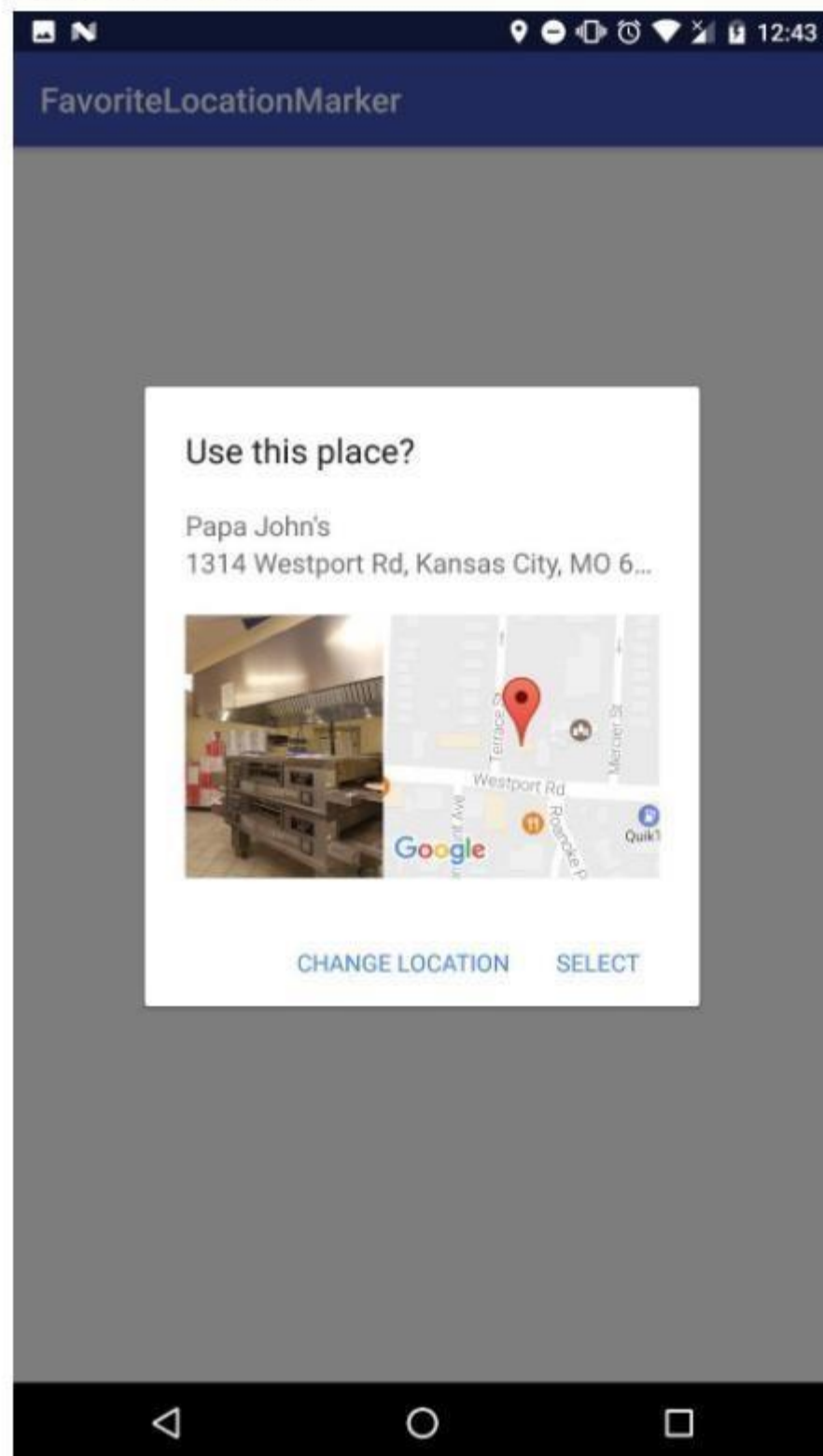


**University of Missouri-Kansas City**  
5100 Rockhill Rd, Kansas City, MO 64110, USA

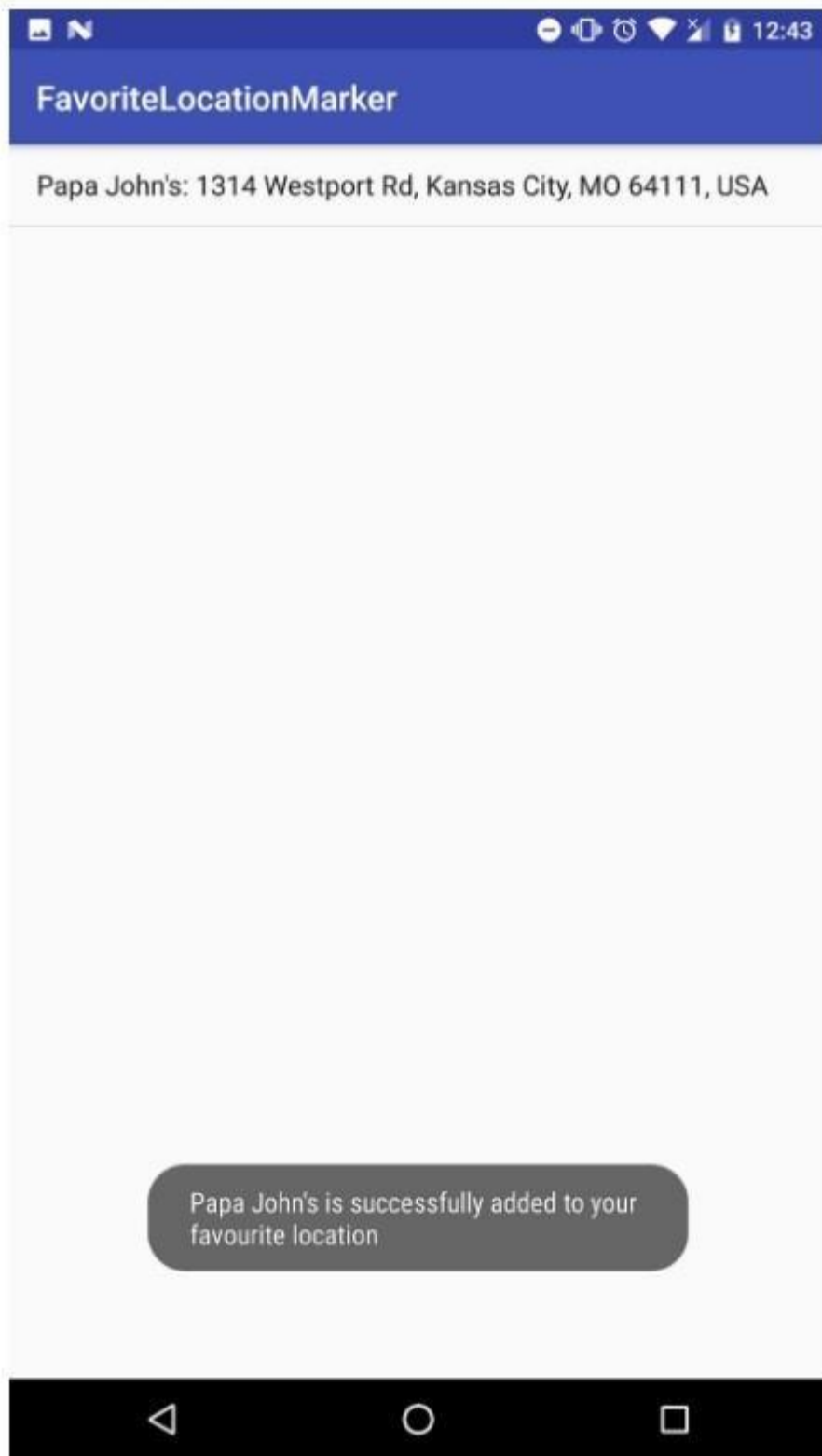


**Rockhurst University**  
1100 Rockhurst Rd, Kansas City, MO 64110, USA





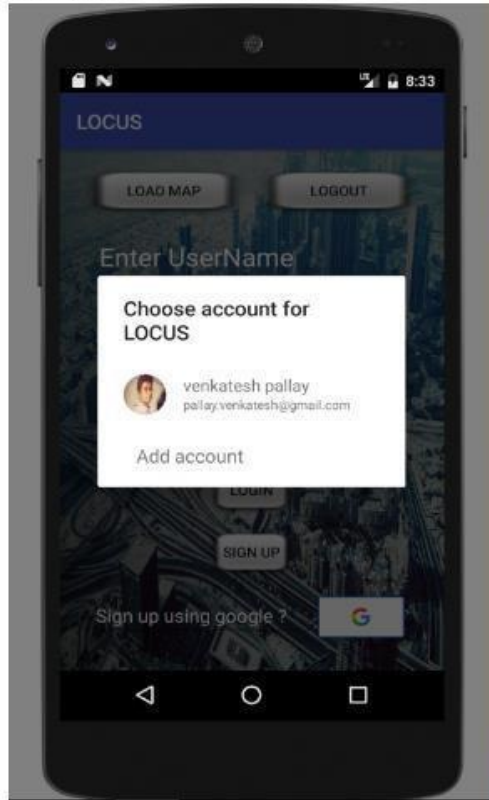




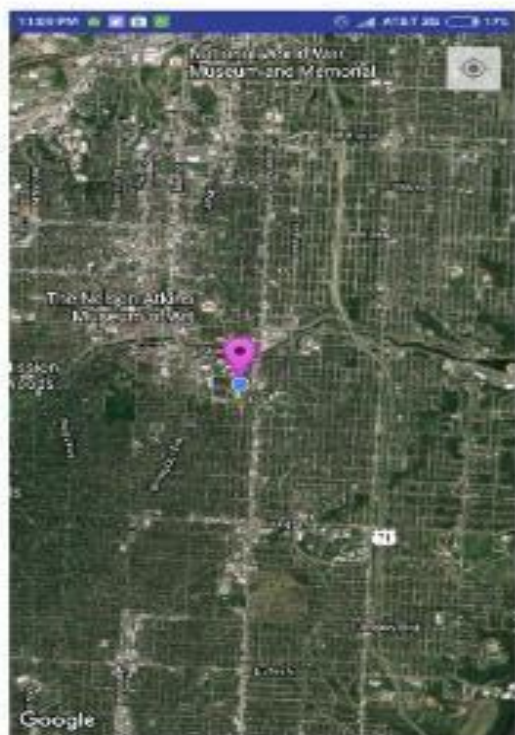
## getting Current Location Module

### OAuth Screen

Android Emulator - Copy\_Nexus\_5\_API\_24:5554

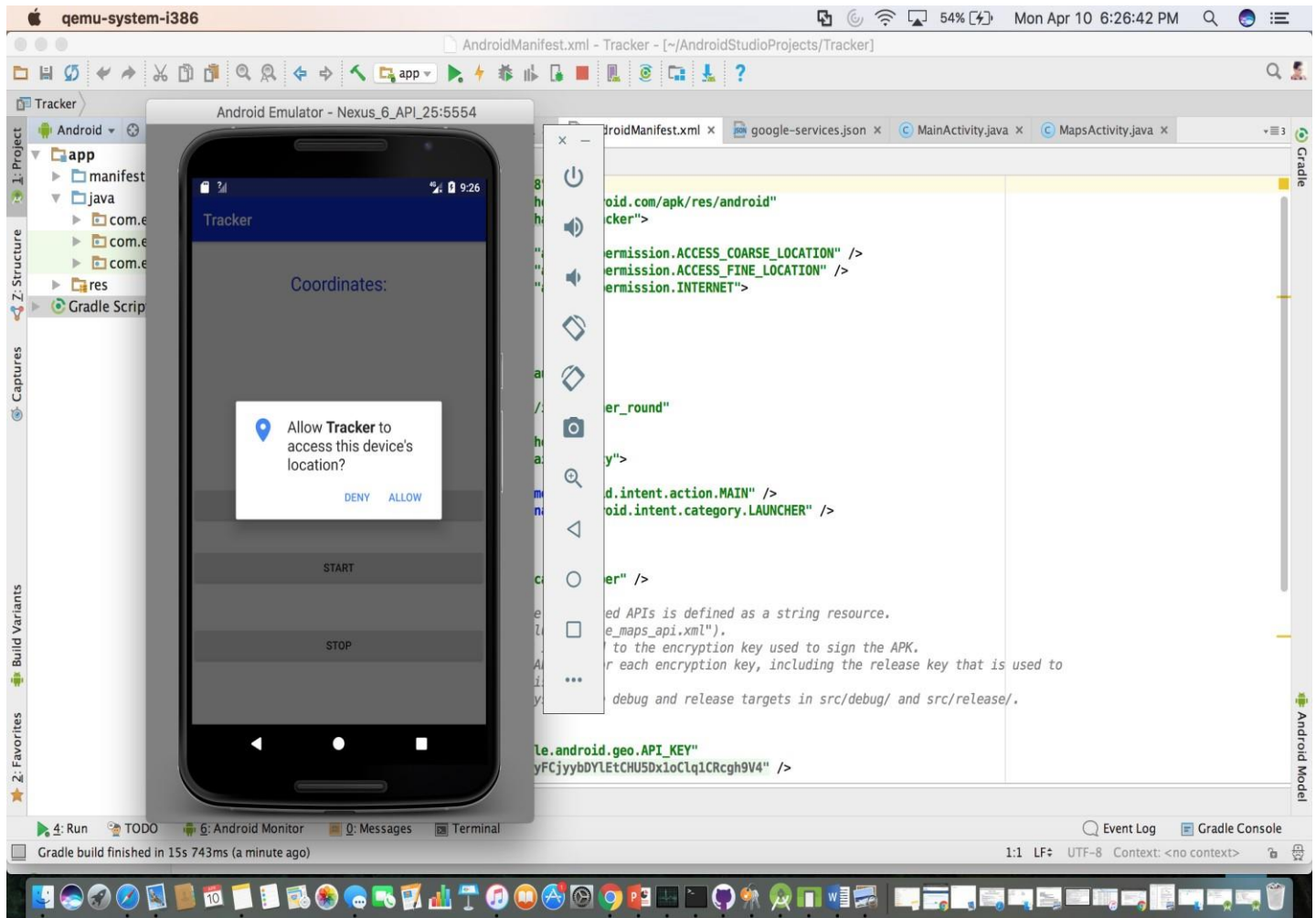


### Getting Current Location (Home Screen)



## INCREMENT 3

Getting the co-ordinates of the current location using the device's location:

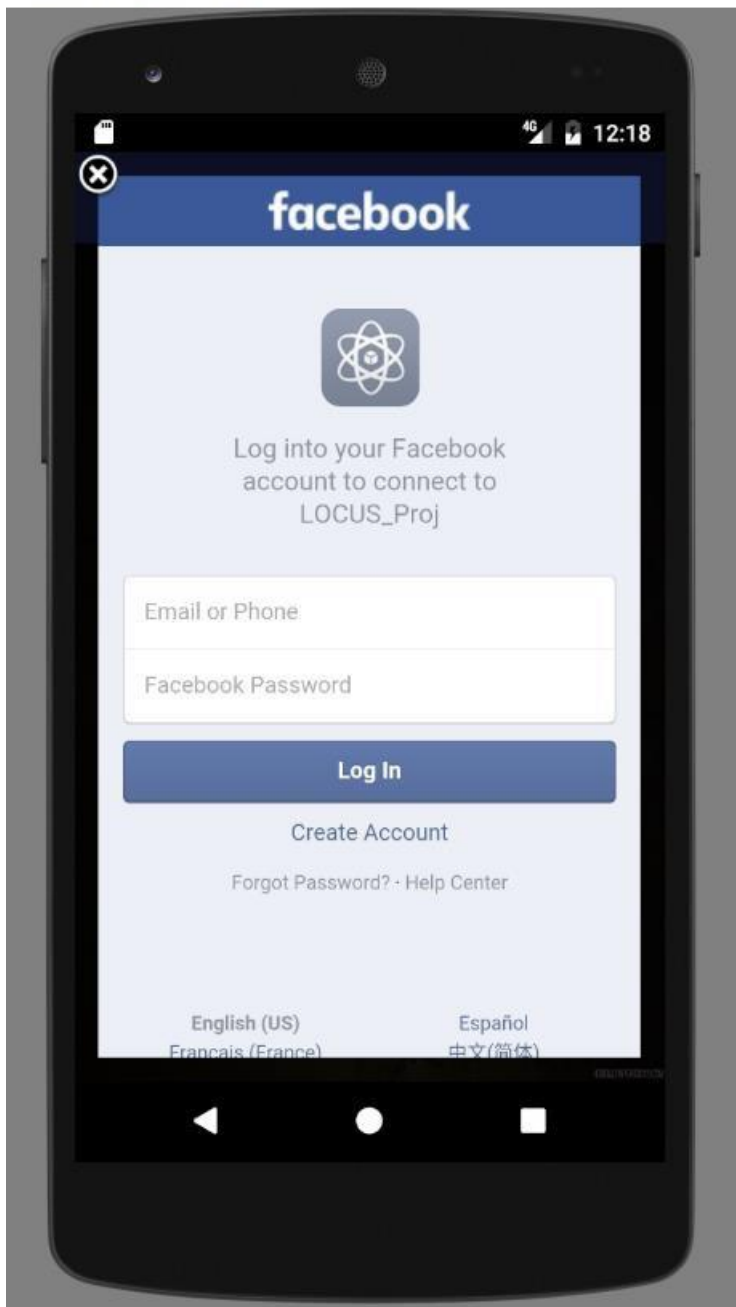


Main page with OAuth:



## Facebook OAuth

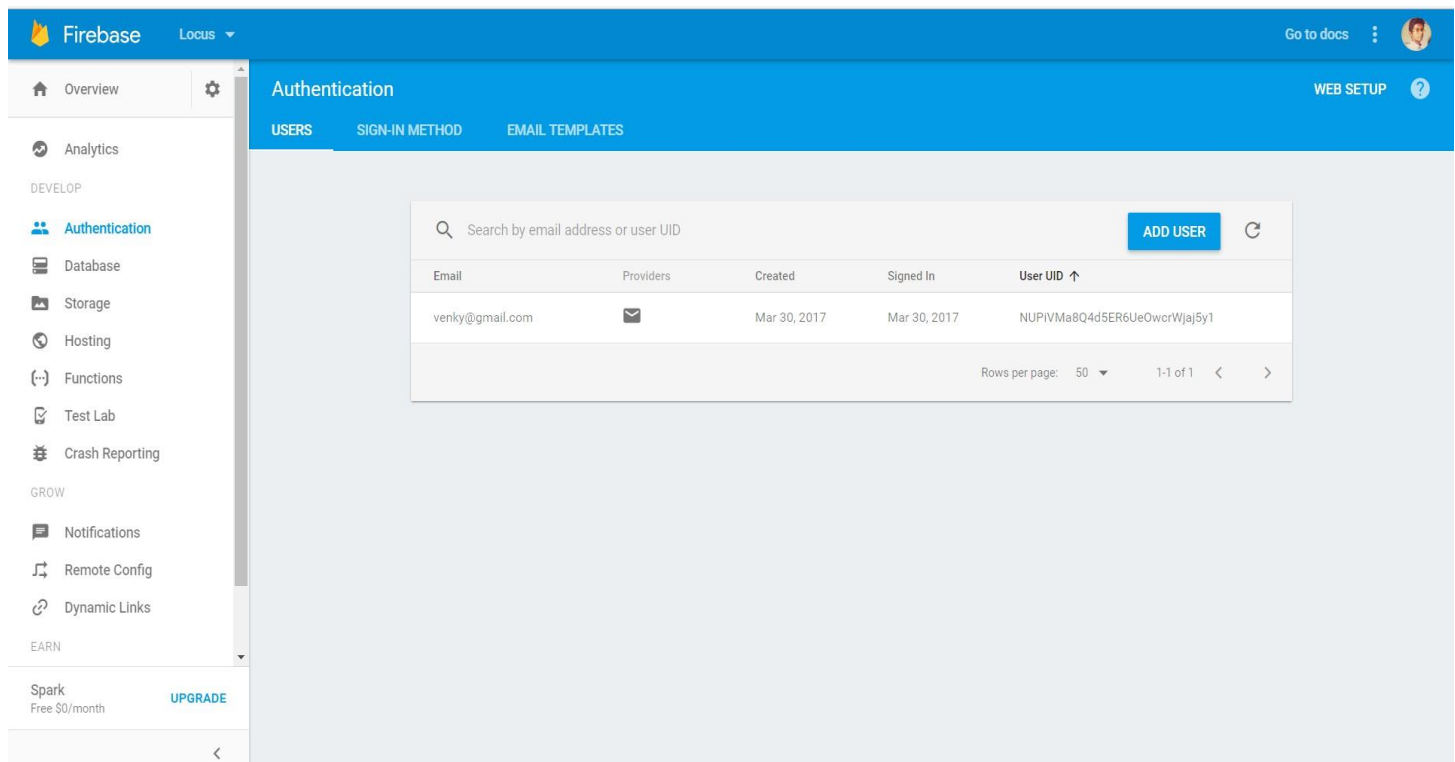
Android Emulator Nexus 5



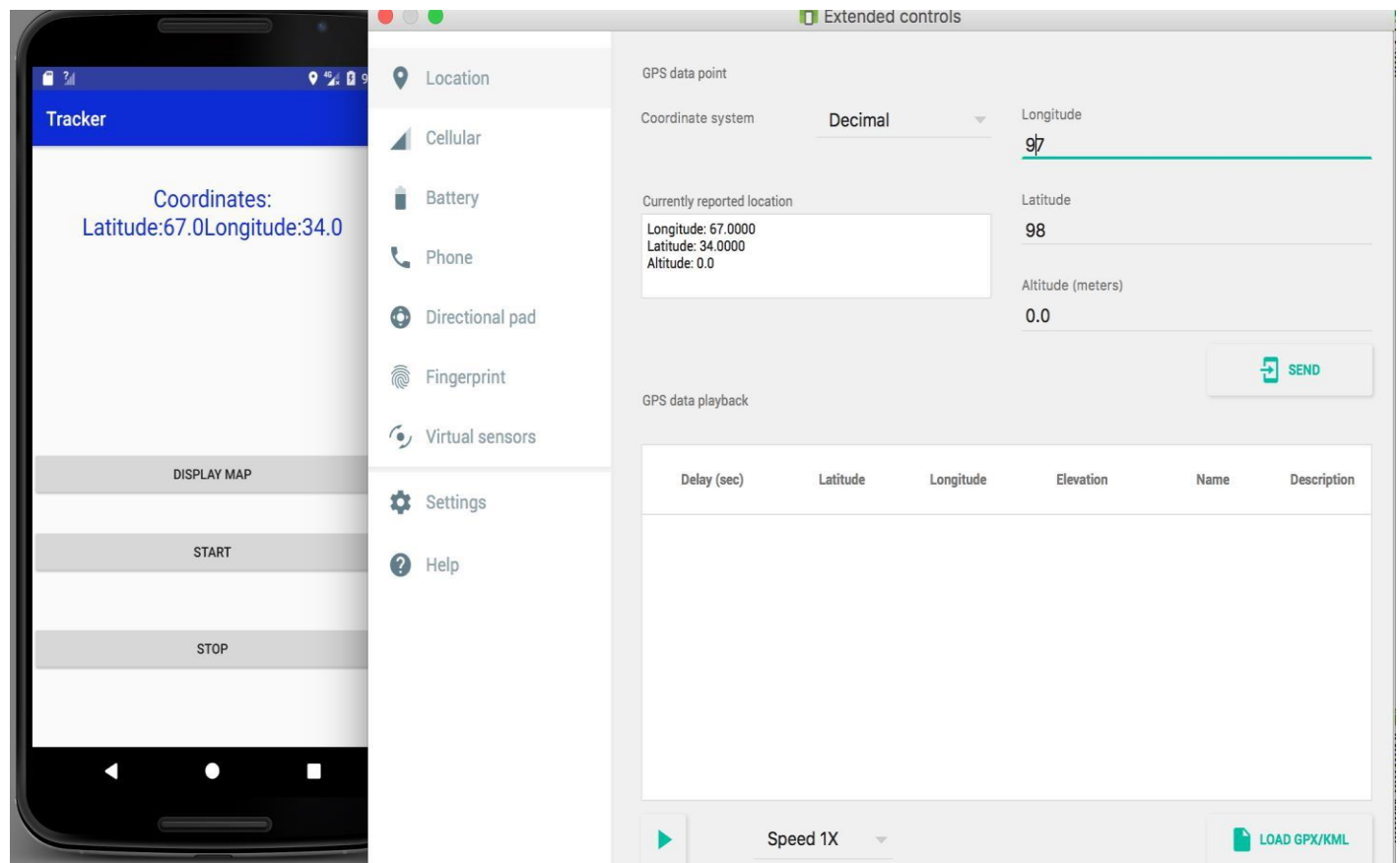
Main page with two Activities(Car Tracking and Favorite Location)



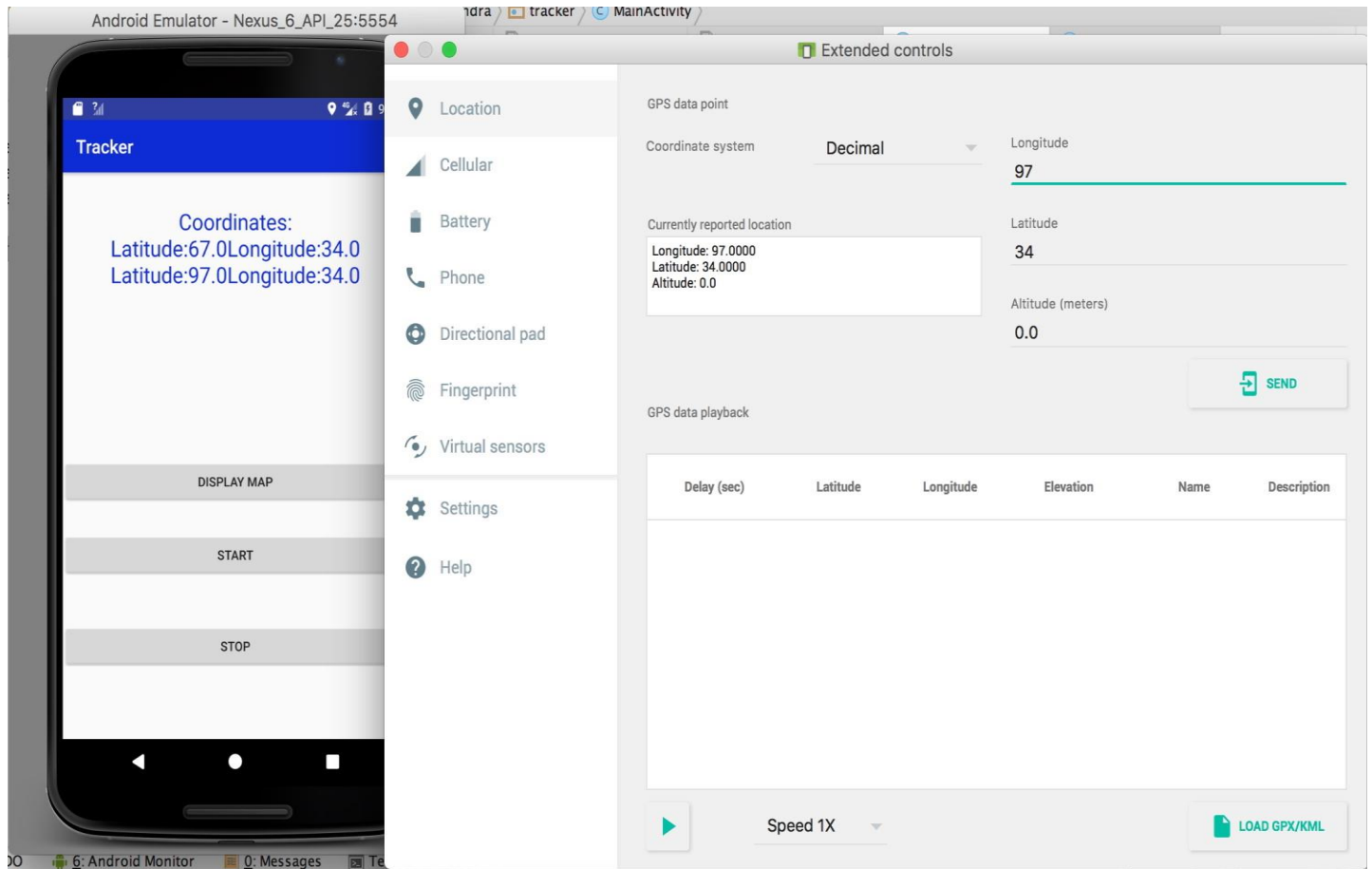
Creating a Firebase account and linking it to the Android Application.



Getting the co-ordinates of the device location and storing it in the Firebase



Tracking the device location and updating the new co-ordinates into the database.





Keeping track of the co-ordinates from the device and storing it in the database.

The screenshot shows the Firebase console interface. The top navigation bar includes the Firebase logo, the project name 'tracker', and a 'Go to docs' link. The left sidebar contains a list of services: Overview, Analytics, Authentication, Database (highlighted), Storage, Hosting, Functions, Test Lab, Crash Reporting, Notifications, Remote Config, and Dynamic Links. The main content area displays the 'DATA' tab for the Realtime Database. The URL bar shows 'https://tracker-30e22.firebaseio.com/'. The database structure is visualized as a tree under the node 'tracker-30e22'. It contains five child nodes, each with 'latitude' and 'longitude' properties.

Node ID	latitude	longitude
-KggG_rU-Ua36Hnu2A8	87	58
-KggulGfdpbkoe1GEyHT	88	88
-Kggx1pozZ3cMVRMb6uy	88	88
-Kgy20C1arlqo5_MQ8A	76	88
-KggzLBxgwmy8TCcklrZ	77	89