Emergency Ambulance Booking App

Submitted in partial fulfillment of the requirements

of the syllabus of

Android Apps Development Lab

in

Information Technology

by

MELVINA MICHAEL	118A3027
S ANANTHASELVI A SHANKARAN	118A3046

Under the Guidance of:

Ms. Bushra Shaikh

Department of Information Technology

SIES Graduate School of Technology

2021-22

CERTIFICATE

This is to certify that the project entitled **“**Emergency Ambulance Booking App**”** is a bonafide work of the following students, submitted to the University of Mumbai in partial fulfillment of the requirement of the syllabus of **Android Apps Development Lab** in **Information Technology.**

MELVINA MICHAEL **118A3027**

S ANANTHASELVI A SHANKARAN **118A3046**

Ms. Bushra Shaikh	Dr. Lakshmi Sudha	Dr. Atul N Kemkar
Internal Guide	Head of Department	Principal

PROJECT REPORT APPROVAL

This project report entitled ***Emergency Ambulance Booking App*** by following students is approved for the requirement of the syllabus of ***Android Apps Development Lab*** in ***Information Technology.***

MELVINA MICHAEL 118A3027

S ANANTHASELVI A SHANKARAN 118A3046

Name of External Examiner: ----------

Signature:----------

Name of Internal Examiner: ----------

Signature:----------

Date:

Place:

DECLARATION

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

MELVINA MICHAEL 118A3027 _____

S ANANTHASELVI A SHANKARAN 118A3046 _____

Signature

Date:

ACKNOWLEDGEMENT

It gives us immense pleasure to thank Dr. Atul N Kemkar, our Principal for extending his support to carry out and develop the project. We also thank our Head of Department Dr. Lakshmi Sudha for her support in completing the project. We wish to express our deep sense of gratitude and thank to our Internal Guide, Ms. Bushra Shaikh for her guidance, help and useful suggestions, which helped in completing our project work in time.

We would like to thank the entire faculty of Information Technology Department for their valuable ideas and timely assistance in this project, last but not least, we would like to thank our non-teaching staff members of our college for their support, in facilitating timely completion of this project.

Project Team

Melvina Michael

S Ananthaselvi A Shankaran

ABSTRACT

Studies show absence of public transportation is one of the reasons for failing healthcare system even in places where basic health infrastructure exists. The study found lack of transport was a major reason for high maternal mortality rate (MMR) in the region. The goal of an ambulance is to reach any place within 15-20 minutes after the distress call and transport the patient to a health facility within 20 minutes thereafter. But this rarely happens. There is a lot of delay in contacting the ambulance and if contacted, when the ambulance will reach the destination. It happens that even after repeated number of calls, the ambulance don't reach the destination on time. Therefore, an android application is proposed where any user can contact or book an ambulance nearby his location and who is currently available so that the ambulance can reach the destination within short span of time. By this method, many human lives can be saved.

Keywords: Emergency, app, ambulance, health, user.

Contents

Introduction

The goal of an ambulance is to reach any place within 15-20 minutes after the distress call and transport the patient to a health facility within 20 minutes thereafter. But this rarely happens. There is a lot of delay in contacting the ambulance and if contacted, when the ambulance will reach the destination. It happens that even after repeated number of calls, the ambulance don't reach the destination on time. Therefore, an android application is proposed where any user can contact or book an ambulance nearby his location and who is currently available so that the ambulance can reach the destination within short span of time. By this method, many human lives can be saved. The application is made in such a way that it can be used by both driver and user efficiently. The user can book the app from anywhere within fraction of seconds as the proposed project automatically takes the user's current location and only that driver is contacted who is available at that moment of time which makes it even more faster to function.

Survey on Existing Apps

1. AmbiPalm

With the AmbiPalm ambulance service, you can book an ambulance immediately or schedule a ride to your hospital.

Features

- Their advanced Algorithms work with a wide network of partners to help you get an Ambulance in the shortest span of time at a very reasonable cost. All you need is to select your desired location and hospital and the ambulance nearest to you will be notified.
- With AmbiPalm you can Request or Donate blood and find a donor at the quickest time possible. You can request blood, plasma, or platelets through AmbiPalm and we connect you to the donor at zero cost.

2. Meddco

Meddco is a similar Ambulance Booking App.

Features

- Meddco ambulance Assistance (MAA) is India's First Emergency ambulance booking application based on live location tracking system.
- Meddco ambulance assistance is an aggregator of the emergency patient transport services. Their app integrates city emergency ambulance transportation for patients and ambulance drivers as their partners onto a mobile technology platform convenient, transpoarent and quick emercgency service fulfilment.

Report on Present Investigation

3.1) Problem Statement:

To build an emergency ambulance booking android application, which will book the driver who is available at that moment from the nearby hospital by automatically taking the user's current location.

3.2) Source of Problem Statement:

Increase in medical ailments has led to the rise in hospitals and emergency medical services assisting patients who require emergency medical assistance at critical moments helping them to reach the hospital on time thus saving their life. Thus we have designed an emergency ambulance booking mobile app to help you book & track the nearest ambulance via mobile. This system would allow the user to book an ambulance easily. In case of emergency the system will automatically book the nearest ambulance and the hospital by fetching the user's current location and add that particular location as pick up point. After fetching of current location, the driver will be allocated according to the status of the driver, if he is free then that particular driver would be assigned to that user. Then the user will be taken to the nearby hospital under that location

Design and Implementation of Android Apps Components

4.1) Layouts

Layout basically refers to the arrangement of elements on a page these elements are likely to be images, texts or styles. These are a part of Android Jetpack**.** They define the structure of android user interface in the app, like in an activity. All elements in the layout are built with the help of Views and ViewGroups. These layouts can have various widgets like buttons, labels, textboxes, and many others.

Some of the Layouts in Android are

- Linear Layout
- Relative Layout
- Constraint Layout
- Table Layout
- Frame Layout
- Absolute Layout

You can declare a layout in two ways:

- **Declare UI elements in XML**. Android provides a straightforward XML vocabulary that corresponds to the View classes and subclasses, such as those for widgets and layouts.

  You can also use Android Studio's Layout Editor to build your XML layout using a drag-and-drop interface.

- **Instantiate layout elements at runtime**. Your app can create View and ViewGroup objects (and manipulate their properties) programmatically.

For our application we used Linear layout, Table layout and Relative layout for customization of inner components wherever needed.

4.2) Intents

Android Intent is the *message* that is passed between components such as activities, content providers, broadcast receivers, services etc. It is generally used with startActivity() method to invoke activity, broadcast receivers etc.

There are two types of intents:

- **Explicit intents** specify which application will satisfy the intent, by supplying either the target app's package name or a fully-qualified component class name. You'll typically use an explicit intent to start a component in your own app, because you know the class name of the activity or service you want to start. For example, you might start a new activity within your app in response to a user action, or start a service to download a file in the background.

  We have used Explicit intent to connect various activities like going to home page from login page.

```
Intent intent = new Intent(LoginActivity.this, HomePage.class);
startActivity(intent);
```

- **Implicit intents** do not name a specific component, but instead declare a general action to perform, which allows a component from another app to handle it. For example, if you want to show the user a location on a map, you can use an implicit intent to request that another capable app show a specified location on a map.

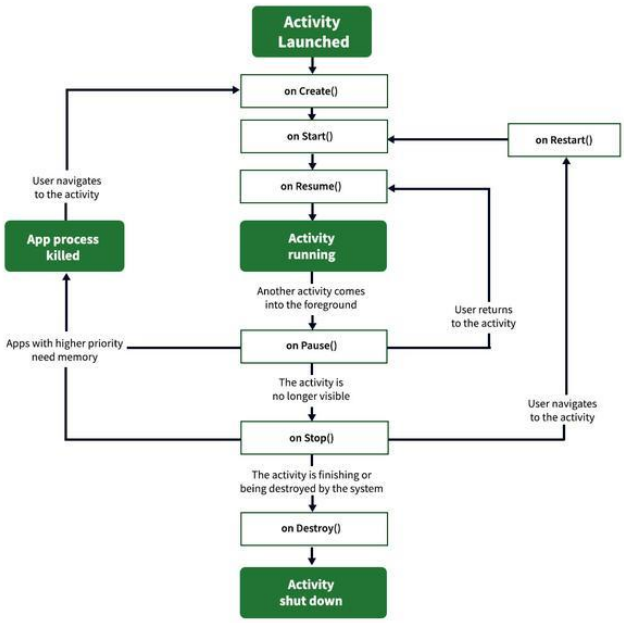
  We have used Implicit intent to access camera.

```
Intent takePictureIntent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
```

4.3) Activity

To navigate transitions between stages of the activity lifecycle, the Activity class provides a core set of six callbacks: onCreate(), onStart(), onResume(), onPause(), onStop(), and onDestroy(). The system invokes each of these callbacks as an activity enters a new state.

Activity Lifecycle :



Activity Lifecycle in Android

We have created many activities like for Registration Page and Login Page. We have also created separate activities for Home Page and My Account Page.

4.4) Database

SQLite is an in-process library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine. The code for SQLite is in the public domain and is thus free for use for any purpose, commercial or private. SQLite is the most widely deployed database in the world with more applications than we can count, including several high-profile projects.

SQLite is an embedded SQL database engine. Unlike most other SQL databases, SQLite does not have a separate server process. SQLite reads and writes directly to ordinary disk files. A complete SQL database with multiple tables, indices, triggers, and views, is contained in a single disk file. The database file format is cross-platform - you can freely copy a database between 32-bit and 64-bit systems or between big-endian and little-endian architectures. These features make SQLite a popular choice as an Application File Format. SQLite database files are a recommended storage format by the US Library of Congress. Think of SQLite not as a replacement for Oracle but as a replacement for fopen(). SQLite is a compact library. With all features enabled, the library size can be less than 600KiB, depending on the target platform and compiler optimization settings. (64-bit code is larger. And some compiler optimizations such as

aggressive function inlining and loop unrolling can cause the object code to be much larger.) There is a tradeoff between memory usage and speed. SQLite generally runs faster the more memory you give it. Nevertheless, performance is usually quite good even in low-memory environments. Depending on how it is used, SQLite can be faster than direct filesystem I/O.

We have used SQLlite to store the user details entered at the time of registration.

4.5) Camera

Camera is mainly used to capture picture and video. We can control the camera by using methods of camera API.

Android provides the facility to work on camera by 2 ways:

1. By Camera Intent
2. By Camera API

We have used camera to allow user take a picture for the profile image.

```
Intent takePictureIntent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
```

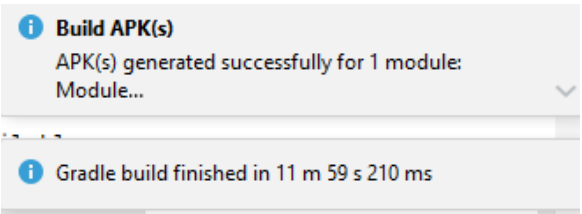
4.6) Location API

The location APIs available in Google Play services facilitate adding location awareness to your app with automated location tracking, wrong-side-of-the-street detection, geofencing, and activity recognition.

We have used location API to fetch the current location of user and driver.

4.7) Generate APK

Open the project >> Go to Build >> Build APKs/ Bundles >> Build APKs



Report on Proposed System and its Implementation

Block Diagram:

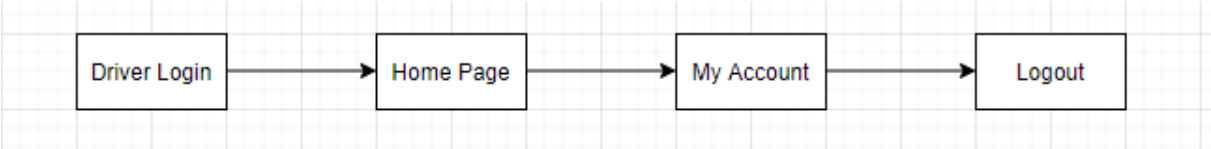


Fig.1: Driver Module

Fig.1. shows the flow of the driver module which mainly consists of four pages. The driver can enter his username and password in login page to login into the app. Then comes the home page where the driver can see a map showing his current location on it. There is also an Account Page where the driver can update his personal information, update his profile picture, etc. Then can successfully logout using logout button.

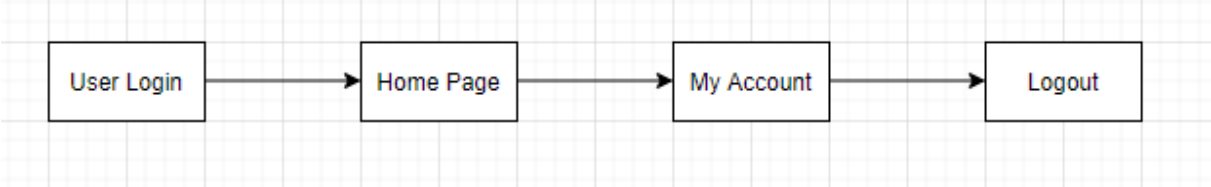


Fig.2: User Module

Fig.2. shows the flow of the user module which mainly consists of four pages. The user can enter his username and password in login page to login into the app. Then comes the home page where the user can see a map showing his current location on it. A button is also available on the home page to book the driver. There is also an Account Page where the user can update his personal information, update his profile picture, etc. Then can successfully logout using logout button.

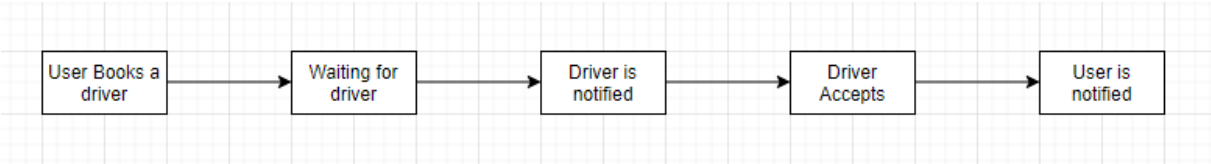


Fig.3: App Flow

Fig.3. shows the basic flow of the application. The user can start booking a driver using the button provided in his home page. Then the driver is notified about the request by the user. And only that driver is notified who is available at that moment. Till then the user needs to wait. Once the driver accepts the user's request, the user is then notified that the driver is booked and the location and the route between both of them is shown to both.

Flowchart:

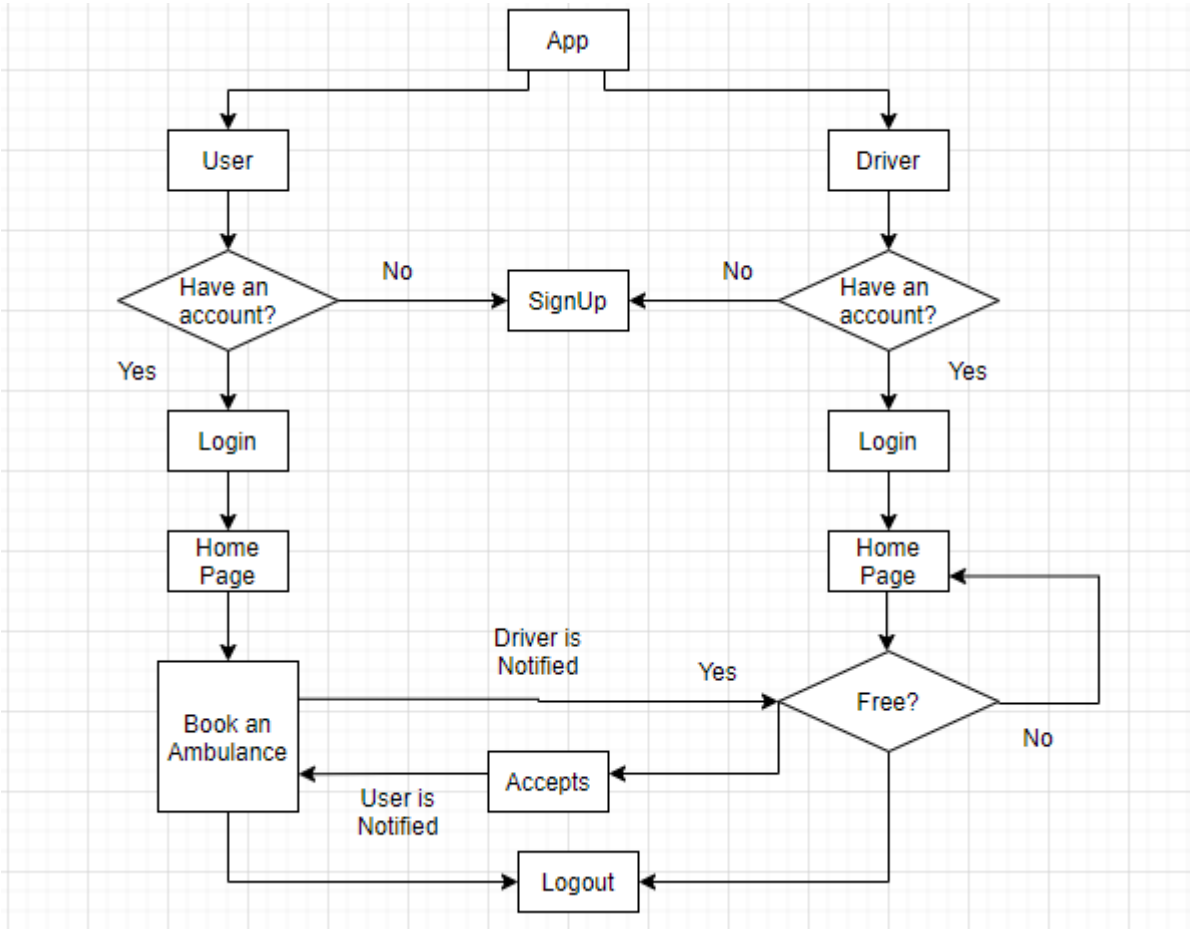


Fig.4

Hardware –

- Android Device
- GPS
- Internet
- Camera (For Updating Profile Picture)

Results and Discussions :

Module A: Driver Modules

Fig.5

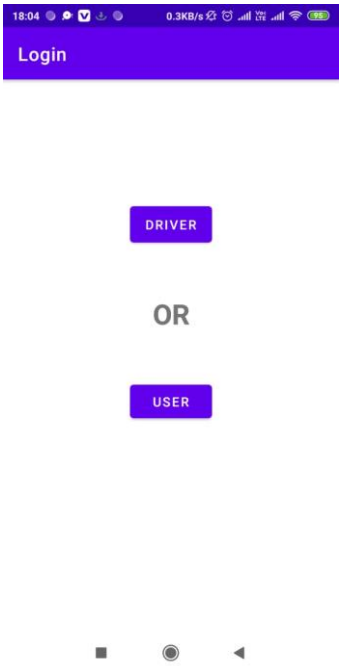


Fig.6

Fig.7

New user (driver) can register himself using Sign Up Page. Validation is done for every input field in the Sign Up Page.

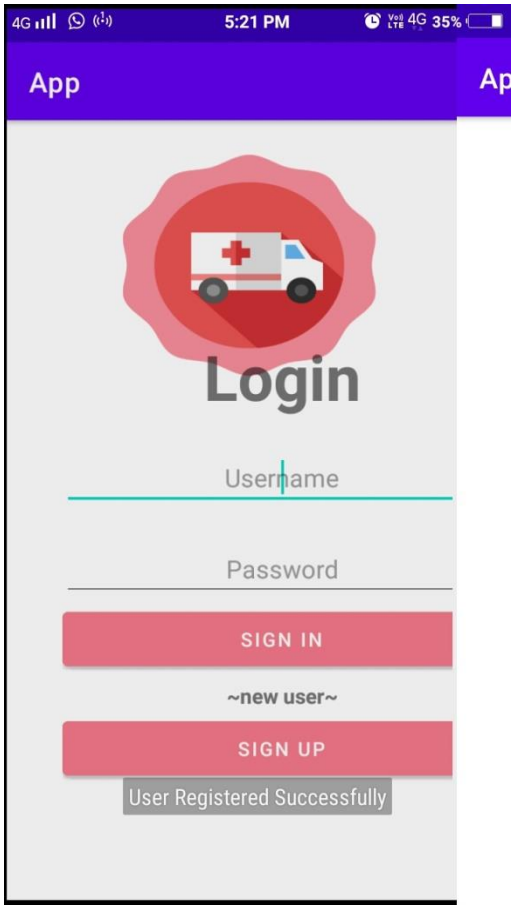


Fig.8

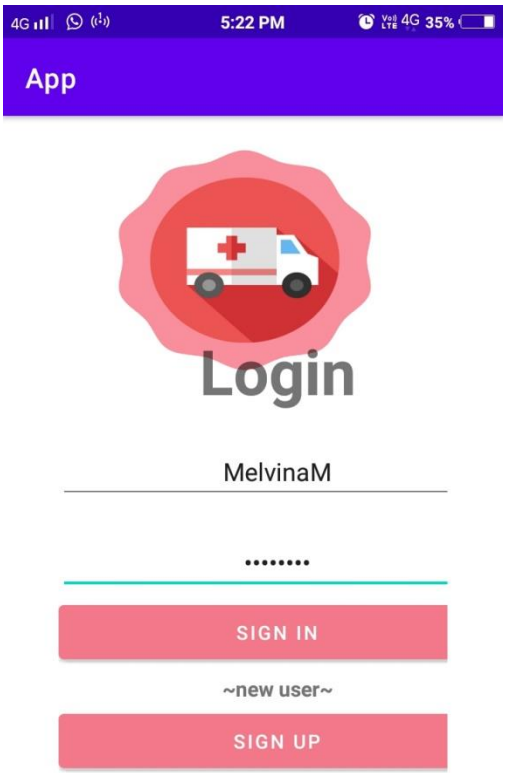


Fig.9

Then he can enter his registration details in order to login to the app using the Login Page.

Module B: User Modules

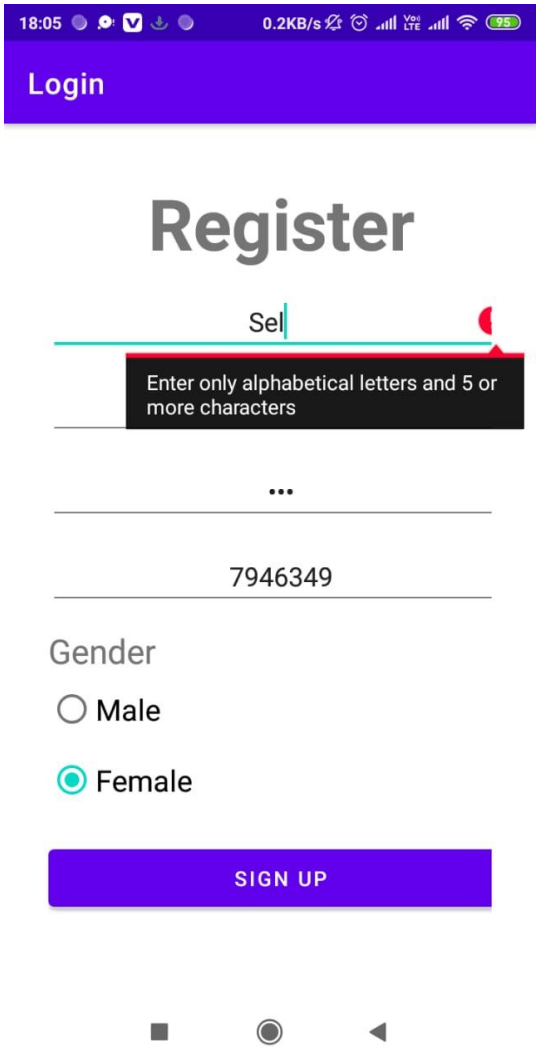


Fig. 10

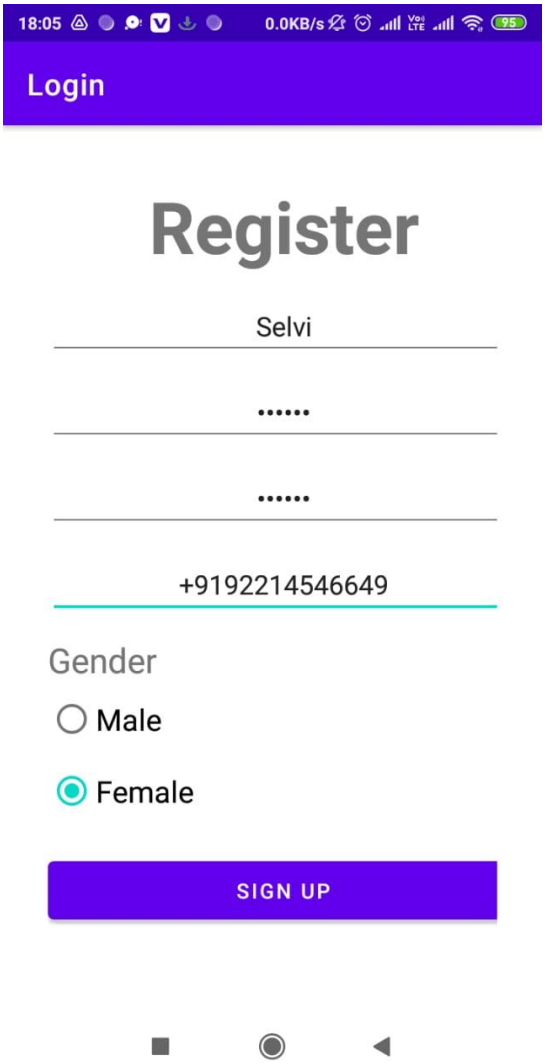


Fig.11

The user can use this Sign Up Page to register himself to use the App efficiently. Strict Validations has been applied on every input field.

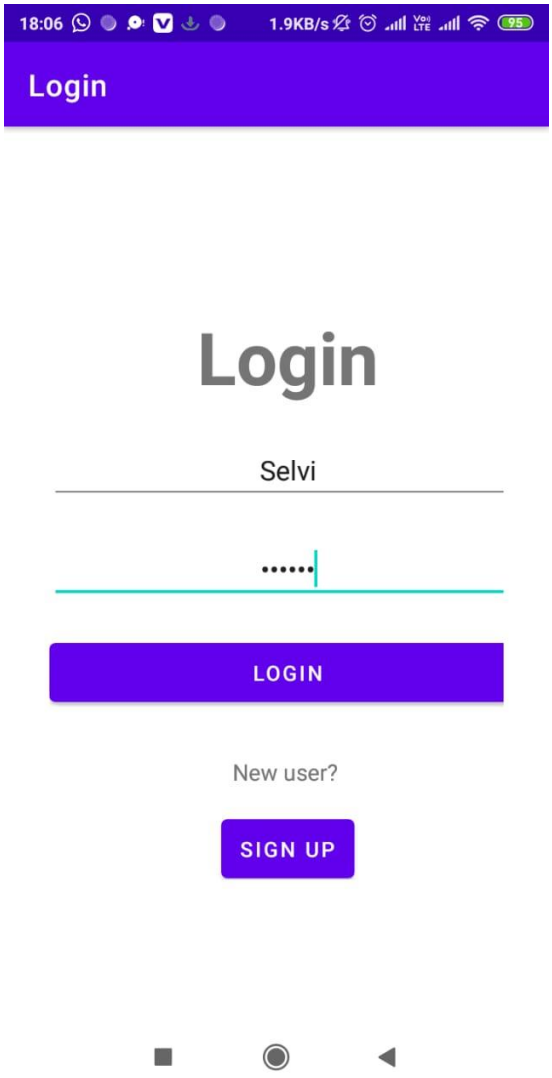


Fig. 12

Fig. 13

The user can login with the registered credentials using login Page. Once logged in, the user can able to see the home page which displays the map on which current location of the user is fetched and a button is given to book the driver.

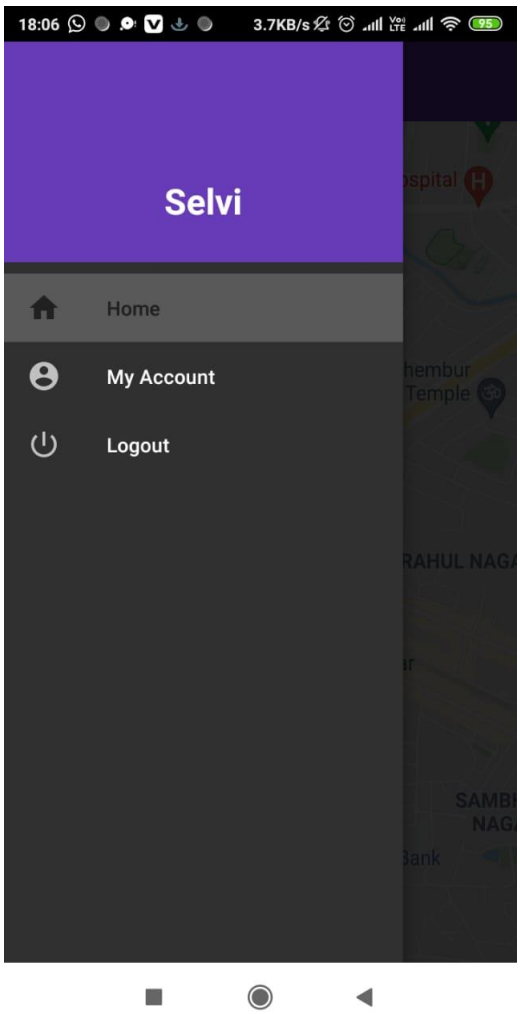


Fig. 14

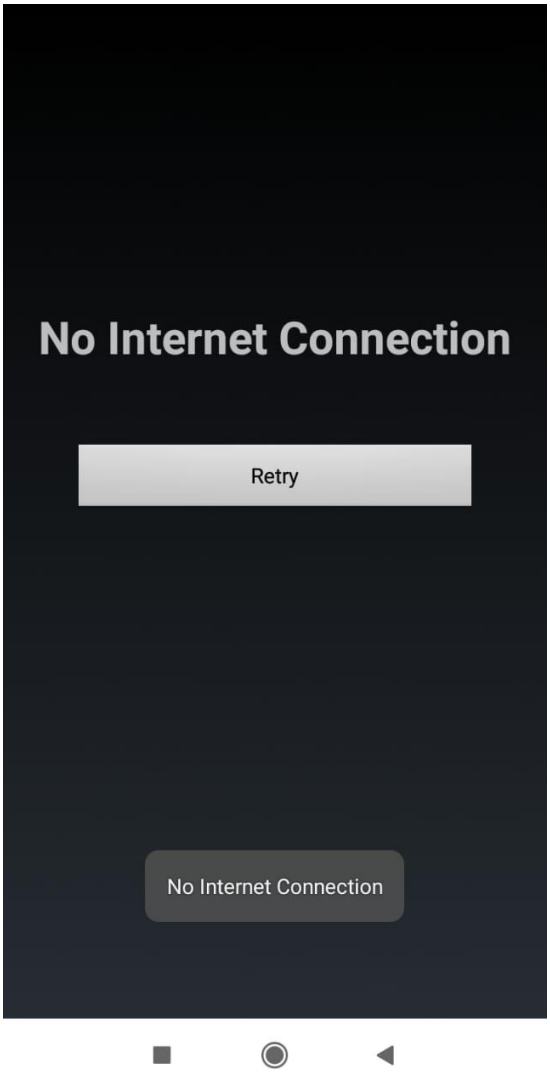


Fig. 15

Navigational drawer is implemented to move from one fragment to another fragment. Home page, My account and logout are the three options available for the user. On clicking My account, user will be taken to that page. And on clicking logout option, user will be taken to the login page again. And in the header the username is retrieved. Broadcast receiver is used to detect the network change. Whenever the Internet connectivity is lost, the above page is shown to the user. And home page is displayed again when the internet connection is stable.

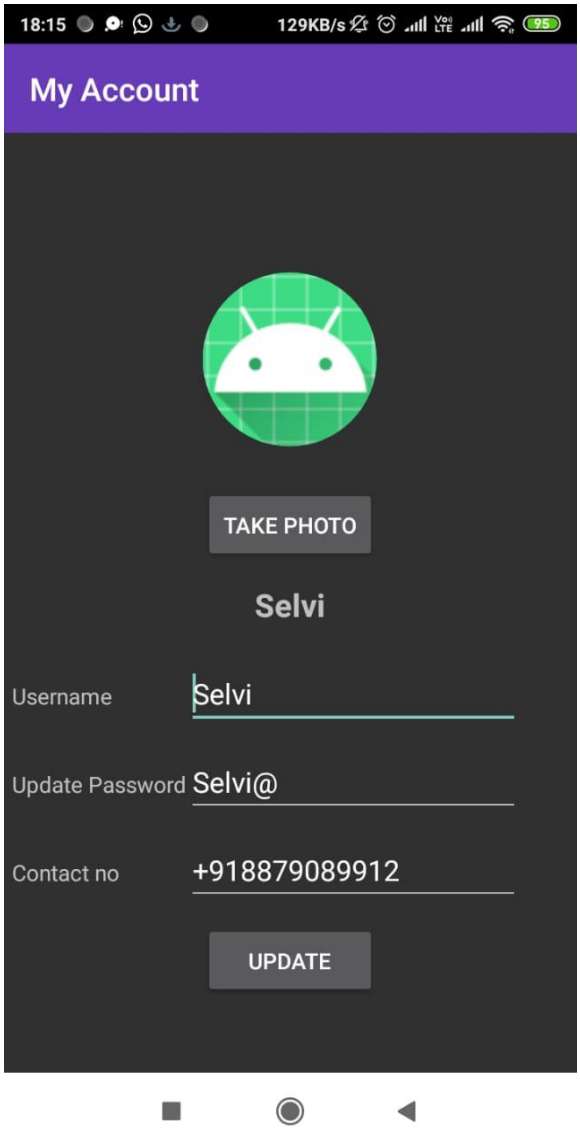


Fig.16

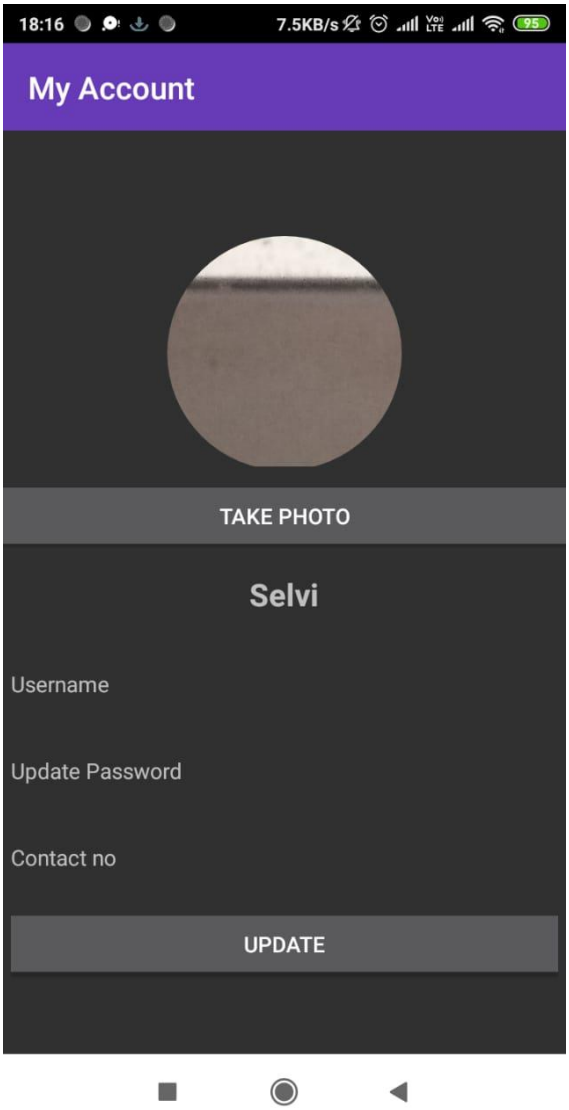


Fig.17

Inside my account fragement, all information of user is retrieved and can be updated by the user whenever required. User can set or update his profile picture using camera.

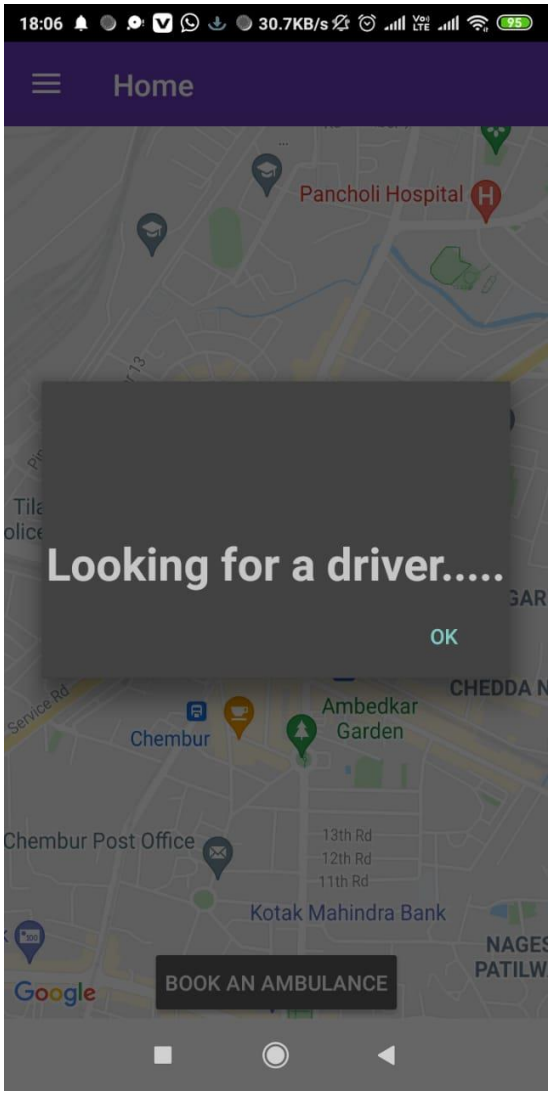


Fig.18

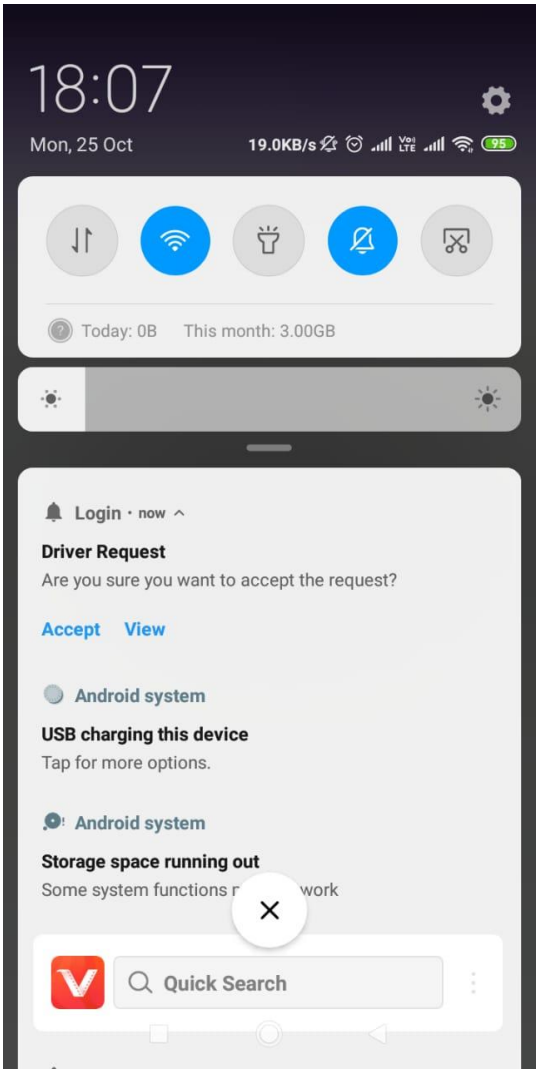


Fig. 19

Whenever the user clicks on Book an Ambulance button, an alert is shown. And at the same time, the driver who is available at that point of time will get a notification (request) which he can accept or view accordingly.

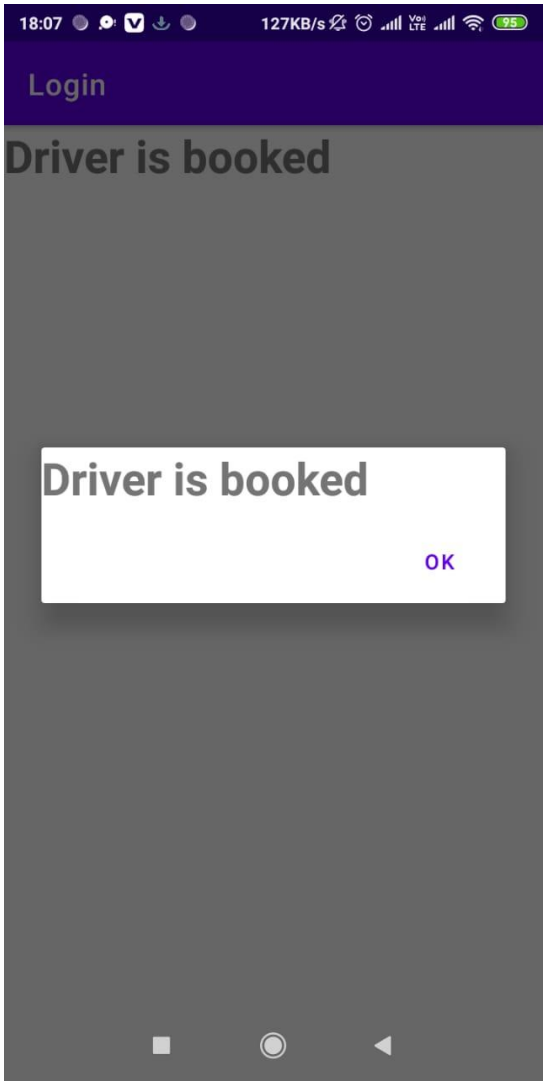


Fig.20

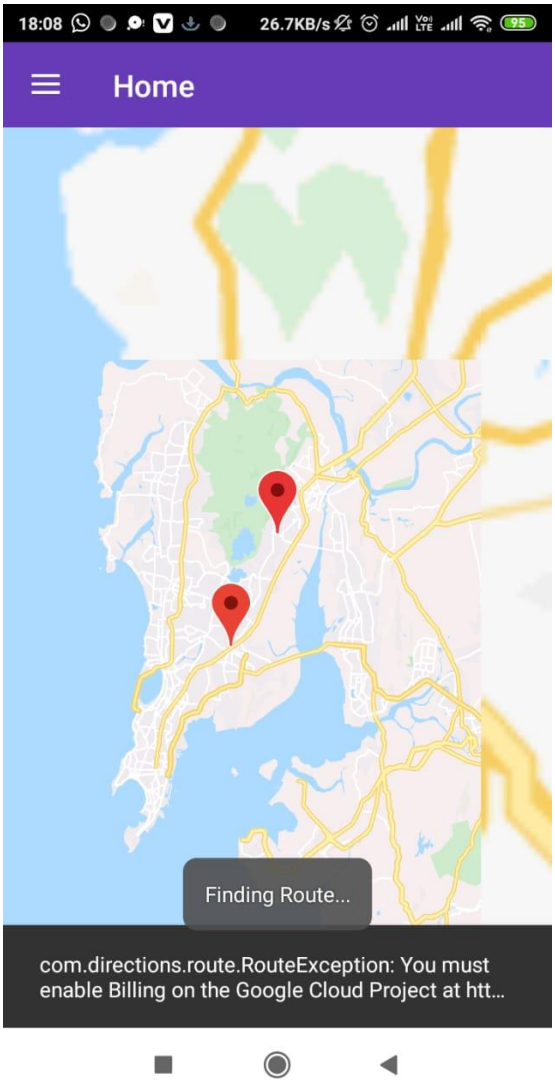


Fig. 21

If the driver accepts the request the user will be shown an alert that driver is booked and when the user clicks on OK button, it will take him to the map where the route between the user location and driver location will be shown.

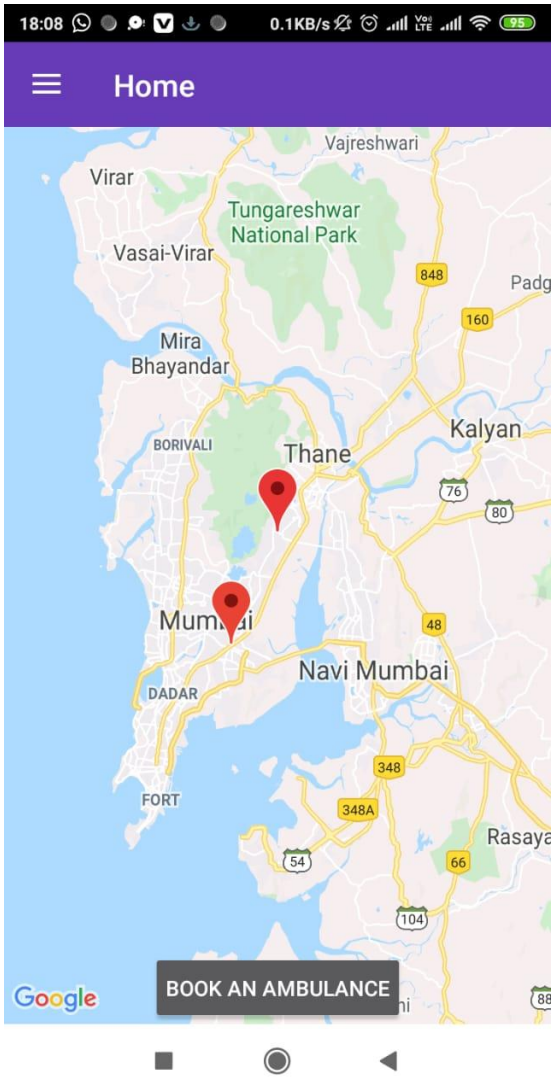

18:08
0.1KB/s
Home
Vajreshwari
Virar
Tungareshwar
National Park
Vasai-Virar
848
Padg
160
Mira
Bhayandar
Kalyan
BORIVALI
Thane
76
80
48
Navi Mumbai
DADAR
348
FORT
348A
Rasaya
54
66
104
Google
BOOK AN AMBULANCE

Fig.22

Conclusion

Thus created an Emergency Ambulance Booking android application which helps the common people to book an ambulance immediately whenever necessary so that no delay occurs in saving a life. The project overall consists of login page, registration page, home page, my account page. The user can book an ambulance anytime and anywhere. The driver will get an immediate notification which when he accepts will be notified to the user. In this way, the application works in an efficient manner.

Advantages:

1) Automatically takes user's current location, hence quick and easy.

2) Nearby hospital's driver is allocated.

Disadvantages:

1) The driver needs to set his status again and again.

Future Scope:

- Request or Donate blood and find a donor at quickest time possible.

References

[1] https://developers.google.com/maps/documentation/android-sdk/start
[2] https://youtu.be/bpRpQTykGfg - SQLite Database
[3] https://youtu.be/lO1PfP96V6c - Camera
[4] https://youtu.be/gNFhT-WUHv0 - Broadcast Receiver
[5] https://youtu.be/M8YuFjoGUp8 - Internet Connectivity
[6] https://youtu.be/2bW0X1nNvX0 - Notification