A

Mini Project

On

STAY ALERT FOR EVER(SAFE)

(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

B. KAWSHIK(197R1A05K0)

G. SAI MAHESH(197R1A05K9)

M.SAI KUMAR(197R1A05M8)

Under the Guidance of

DR.RAJ KUMAR PATRA

(Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR TECHNICAL CAMPUS

UGC AUTONOMOUS

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi) Recognized Under Section 2(f) & 12(B) of the UGCAct.1956, Kandlakoya (V), Medchal Road, Hyderabad-501401.

2019-2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

This is to certify that the Project entitled "**STAY ALERT FOR EVER(SAFE)**" being submitted by B.KAWSHIK(197R1A05K0),G.SAI MAHESH(197R1A05K9),M.SAI KUMAR(197R1A05M8) in partial fulfillment of the requirements for the award of degree B.TECH Computer science and engineering to the Jawaharlal Nehru Technological University Hyderabad ,is a record of bondafide work carried out by them under our guidance and supervision during the year 2022-23.

The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

DR. RAJ KUMAR PATRA
(Professor)
INTERNAL GUIDE

DR.A.RAJI REDDY
DIRECTOR

Dr. K. Srujan Raju
HOD

EXTERNAL EXAMINER

Submitted for viva voice Examination held on _____

ACKNOWLEDGEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard to my guide **DR.RAJ KUMAR PATRA,** Associate Professor for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by him shall carry us a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to the Project Review Committee (PRC) **Dr.Punyaban Patel, Ms.Shilpa, Dr.M.Subha Mastan Rao & J. Narasimharao** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to **Dr. K. Srujan Raju,** Head, Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

We are obliged to **Dr.A.Raji Reddy,** Director for being cooperative throughout the course of this project. We also express our sincere gratitude to Sri. **Ch. Gopal Reddy,**Chairman for providing excellent infrastructure and a nice atmosphere throughout the course Of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

B. KAWSHIK (197R1A05K0)

G.SAI MAHESH (197R1A05K9)

M.SAI KUMAR (197R1A05M8)

ABSTRACT

In today's world, safety is the major concern for everyone while driving. There is one death for every 4 minutes due to road accidents in India.The fatalities by road accidents is more than that of the fatalities caused by other aspects in India. Our government is trying to prevent the road accidents in many ways by creating awareness on traffic rules, safety measures etc. As a responsible citizen, we would like to contribute our best by creating an APP to prevent the accidents.This app will monitor the coordinates of the user based on the GPS and when the user is close (~1 KM) to the accident-prone area the app will raise a notification to alert the user. This help the users to be more cautious about the upcoming black spots and the speed limit of vehicle get reduced which prevents the possible accident by alerting the user and save lives of our beloved ones.

LIST OF FIGURES/TABLES

TABLE OF CONTENTS

1. INTRODUCTION

1.1 PROJECT SCOPE

More than 1.2 Millions people are killed in Road Accidents.Nearly for Every 3 minutes in the World one person is getting killed in Road Accident. Road accidents are very common now days. This can be reduced to some extent by providing the information of road head. There is a need to have an effective road accident spot detection system. A system that provide alert about the accident location. This application will provide a specific path from source to destination and provide an alert while reaching close to the accident spot. Our application AS pointer: Accident Prone Spot Locator will give information about road where to put sign boards and alert.

1.2 PROJECT PURPOSE

When a vehicle enters an area comprising of the accident prone location a notification is made that is vehicle has entered the area comprising of one of the accident prone location. When we are 1 km or 500 m close to the accident location the application will give a voice alert. We are also using speed governor device. This device is used to limit the speed of vehicle.

1.3 PROJECT FEATURES

Now a days Road Accidents are major problem in our day to day life.It Reduces the road accidents and Every one is aware of using the app so that by using this app we can prevent more no.of accidents .By using this App we can also aware of Black spots.It should be the priority in all aspects for everyone in present day life

2.SYSTEM ANALYSIS

2.SYSTEM ANALYSIS

SYSTEM ANALYSIS

System Analysis is the important phase in the system development process.The System is studied to the minute details and analyzed. The system analyst plays an important role of an interrogator and dwells deep into the working of the present system. In analysis, a detailed study of these operations performed by the system and their relationships within and outside the system is done. A key question considered here is, “what must be done to solve the problem?” The system is viewed as a whole and the inputs to the system are identified. Once analysis is completed the analyst has a firm understanding of what is to be done.

2.1 PROBLEM DEFINITION

A general statement of medicine repository can be formulated as the given database of information which needs to be informed at regular intervals with accurate data and no misinformation needs to be given to patients or health care workers relying on this software.

2.2 EXISTING SYSTEM

Government identifies the black spots and spread the awareness to the public with the help of hoardings, traffic signals and media etc. Traffic rules are imposed by the government but still lot of citizens don’t obey them due to various factors There are lot of challenges with the current system as it purely depends on the individual to make sure to follow the rules and at the same time it will be very difficult even for the government to identify each and every black spot throughout the country.

2.2.1 DRABACKS OF EXISTING SYSTEM

Following are the disadvantages of existing system:

- •Not following traffic rules
- •Distraction to the user by different reasons
- •Unaware of the accident spots
- •Unknown of the existing speed of the vehicle

2.3 PROPOSED SYSTEM

Today almost everyone is using smart phone, internet and every person is aware of using the apps in mobile. This APP will identify the position of users using GPS and fetches the black spot coordinates from the system. Accident prone areas or Black spots can be identified and obtained by the different sources.It compares the distance between the current position of the user and the black spots, if the distance between both is approximately close to 1 km will raise a notification in mobile.The system will be updated on periodic basis with authentic data about black spots. Rest API is used to pull the data from database and compare it against the coordinates of the user using GPS. It measures the speed of the vehicle and when the vehicle reaches the limit, then it will raise the voice message.

2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM

- •Helps proactively to save the life of an individual.
- •Helps to control the damage if happens in case of an accident.
- •User can access the information easily using mobile.
- •Aware of the accident spots well in hand and more accidents can be controlled.
- •Speed of the vehicle can be maintained within limits

2.4 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and a business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. Three key considerations involved in the feasibility analysis:

- Economic Feasibility
- Technical Feasibility
- Social Feasibility

2.4.1 ECONOMIC FEASIBILITY

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on a project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it give an indication that the system is economically possible for development.

2.4.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

2.4.3 BEHAVIORAL FEASIBILITY

This includes the following questions:

- Is there sufficient support for the users?
- Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible

2.5 HARDWARE & SOFTWARE REQUIREMENTS

2.5.1 HARDWARE REQUIREMENTS:

Hardware interfaces specify the logical characteristics of each interface between the software product and the hardware components of the system. The following are some hardware requirements.

•Smart Phone android with latest version.

•RAM: 6GB and above

•Space on Hard Disk: minimum 5GB

•Processor: AMD Ryzen 3 2500U and above

2.5.2 SOFTWARE REQUIREMENTS:

Software Requirements specifies the logical characteristics of each interface and software components of the system. The following are some software requirements,

•Android Studio – 2021.2.1.15

•Java, XML

•Rest API

•MySQL – Database

3.ARCHITECTURE

3.ARCHITECTURE

3.1 PROJECT ARCHITECTURE

This project architecture shows the procedure followed for classification, starting from input to final output.

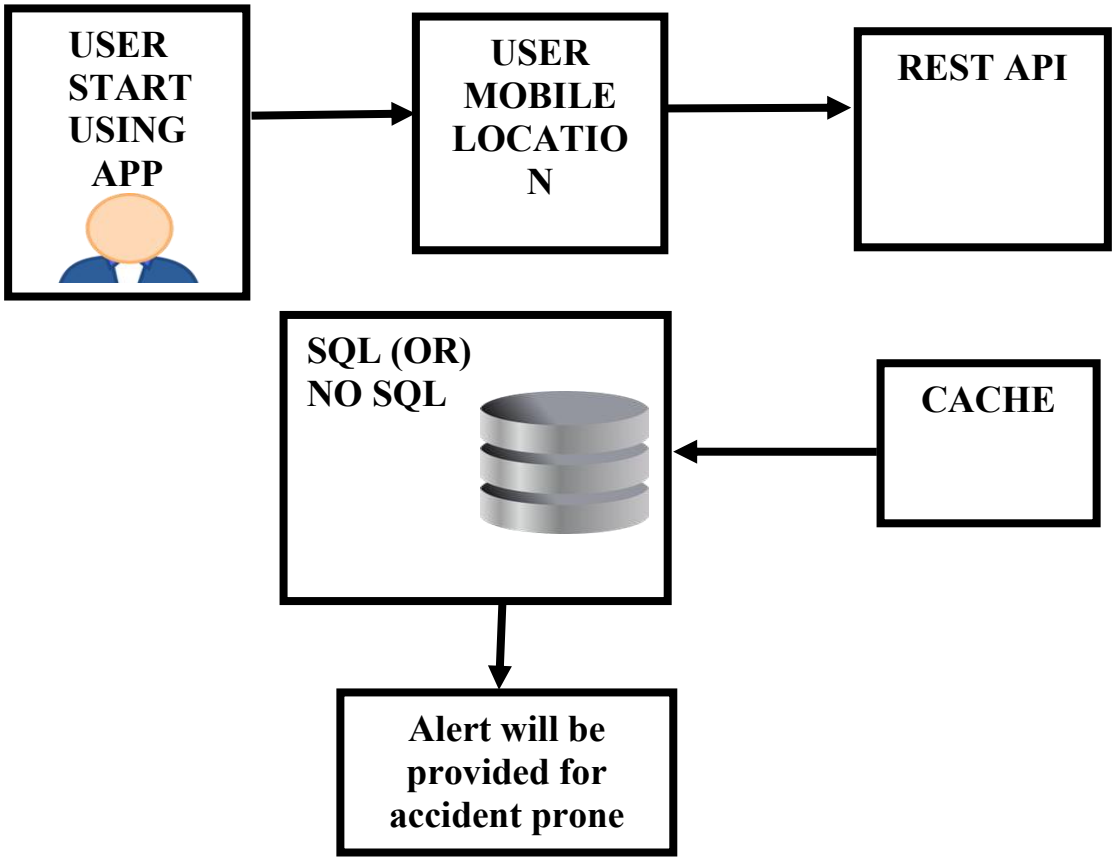

FIG 3.1: PROJECT ARCHITECURE FOR STAY ALERT FOR EVER

This APP is useful to make everyone comfortable to stay safe by proactively raising the voice message.It also reminds the exceeding speed of the vehicle beyond safe speed on that particular road to avoid accidents.This helps to overcome the current challenges to maintain safety measures in the society.This APP definitely plays a key role in decreasing the percentage of road accidents and death rates.

3.2 Proposed Solution Architecture

-

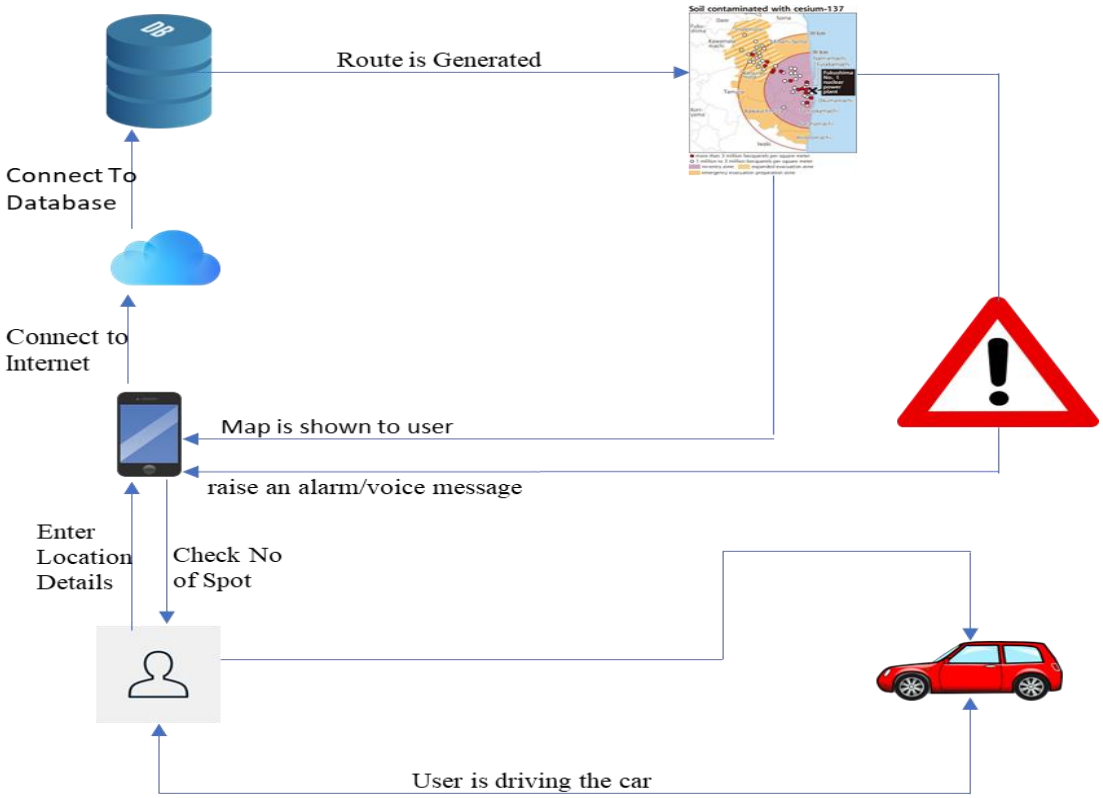

Figure 3.3: Solution Architecture

When the user opens the app automatically his location is tracked when he enters the destination in between his destination if any accident prone zones are identified by our database the user gets notification.So that user can go safe with limited speed when the accident prone zone is arrived.so that we can find the accident zones.

3.3 USE CASE DIAGRAM

In the use case diagram, we have basically one actor who is the user in the trained model. A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures.

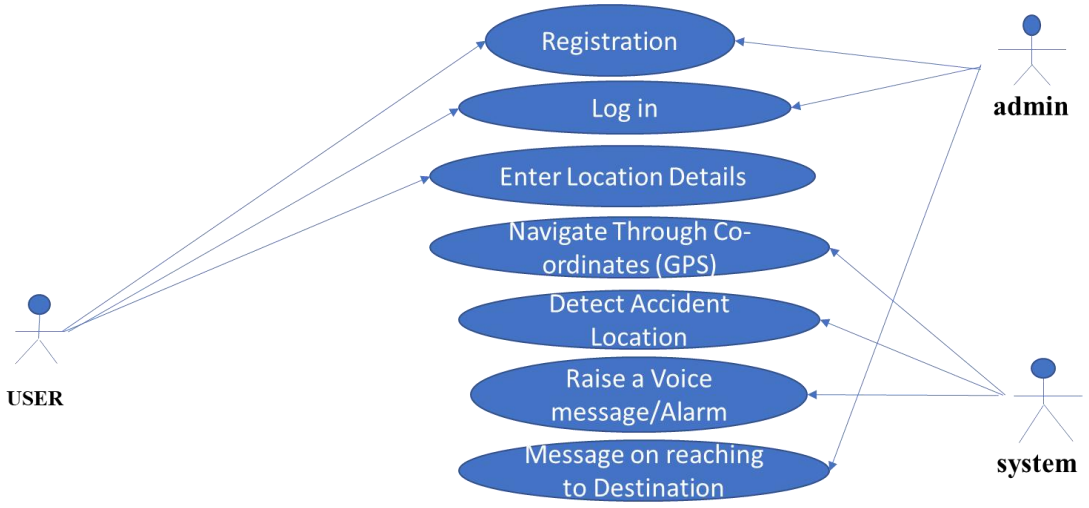

Figure 3.4: Use Case Diagram for SAFE ALERT FOR EVER

3.4 CLASS DIAGRAM

Class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods),and the relationships among objects.

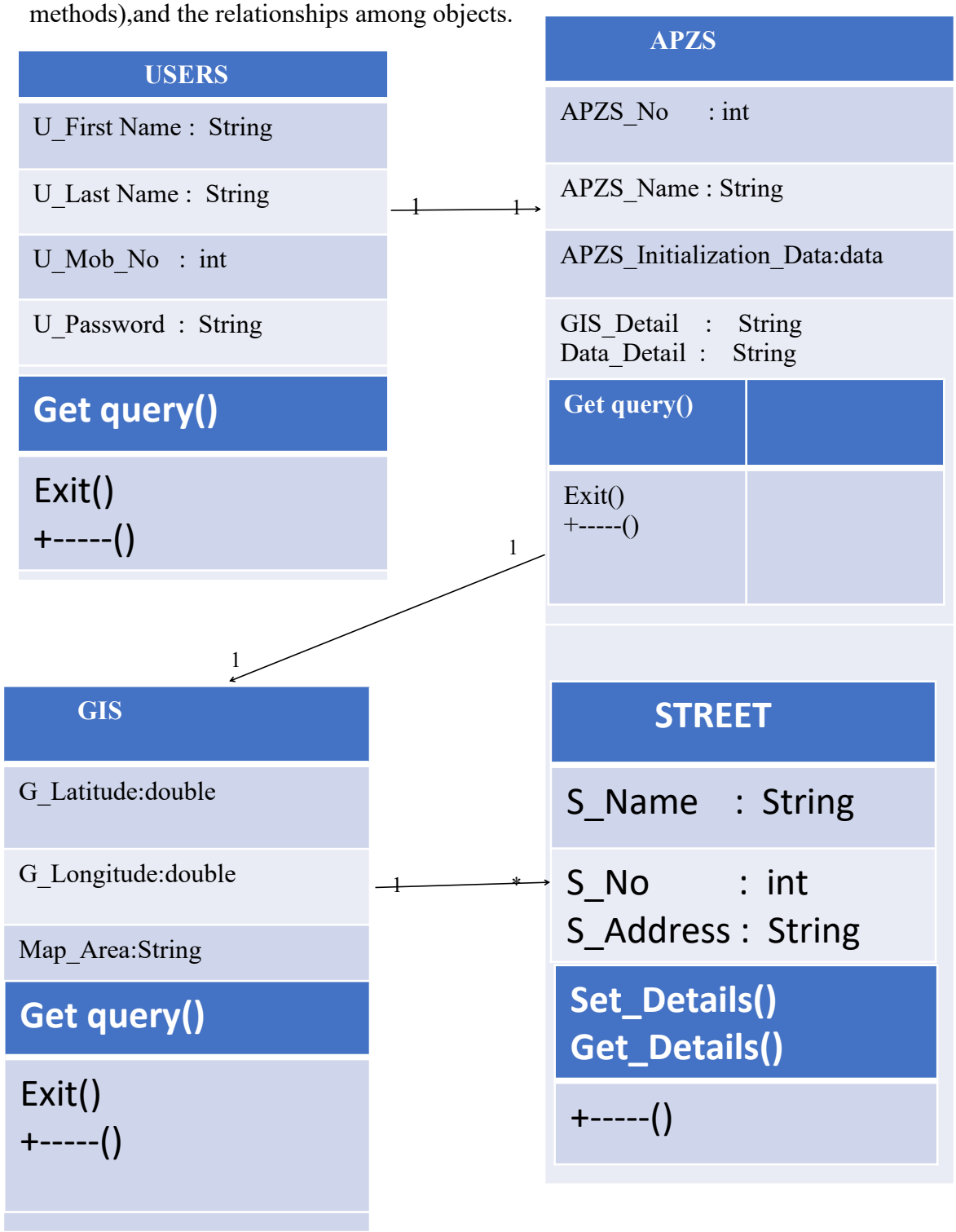

Figure 3.5: Class Diagram for SAFE ALERT FOR EVER

3.5 ACTIVITY DIAGRAM

Activity diagrams are graphical representations of work flows of stepwise activities and actions with support for choice, iteration and concurrency. They can also include elements showing the flow of data between activities through one or more data stores.

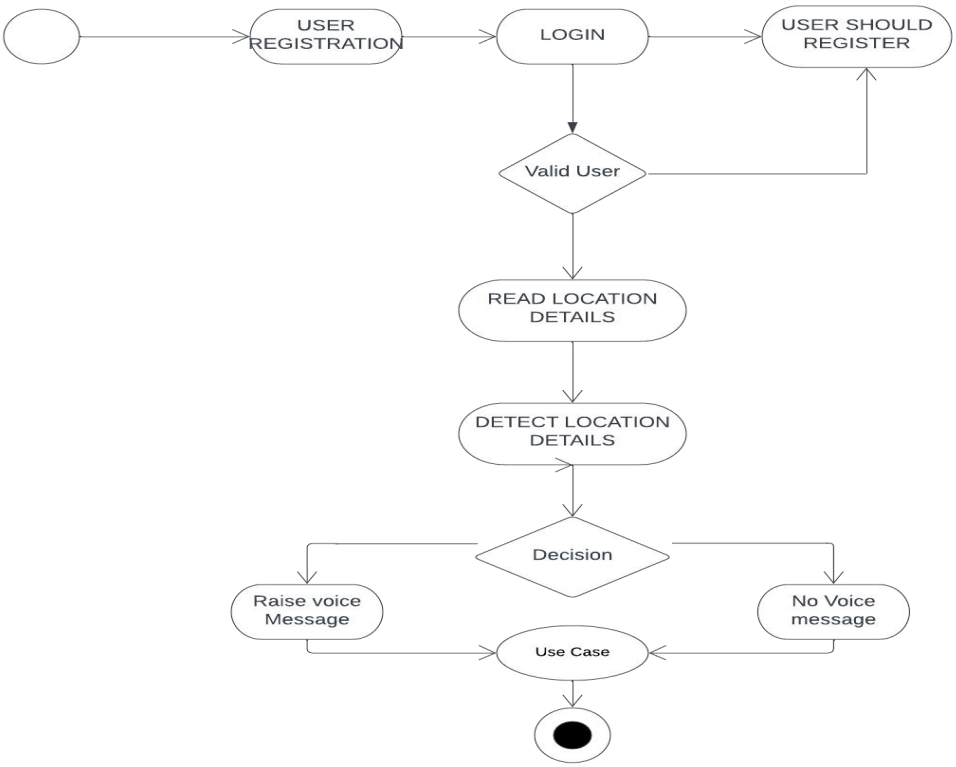

Figure 3.6: Activity Diagram for STAY ALERT FOR EVER

3.6 SEQUENCE DIAGRAM

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the logical view of the system under development.

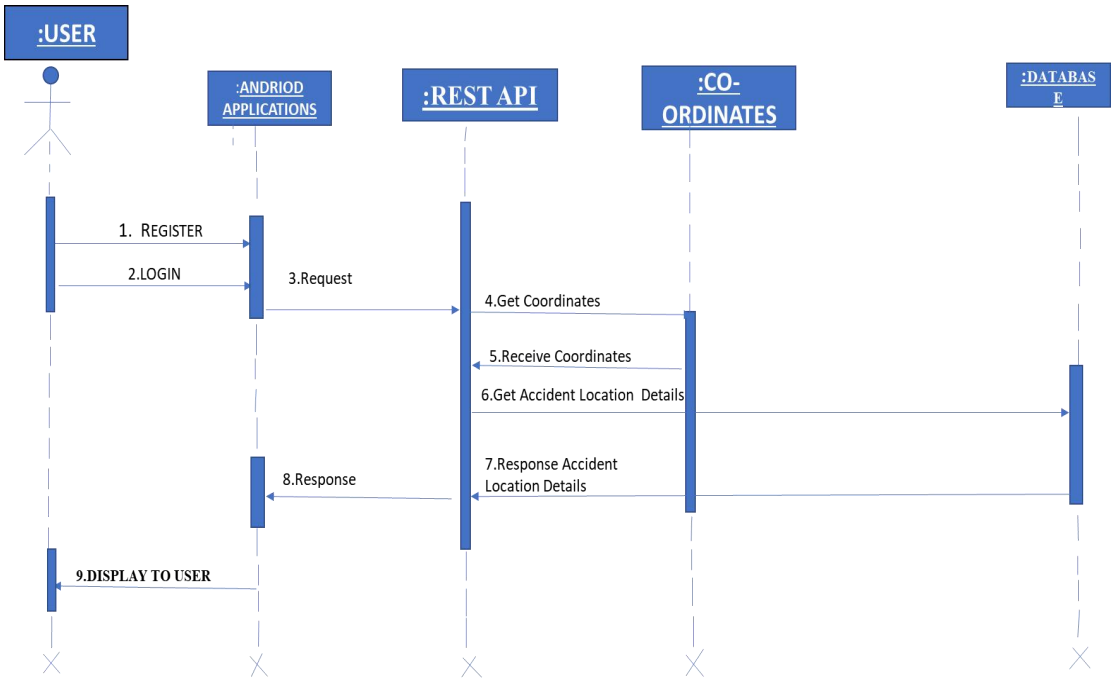

Figure 3.7: Sequence Diagram for STAY ALERT FOR EVER

4.IMPLEMENTATION

4.1 SAMPLE CODE

Main Activity

```
package edu.gcu.gpstrackingdemo;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import android.Manifest;
import android.annotation.SuppressLint;
import android.content.pm.PackageManager;
import android.location.Address;
import android.location.Geocoder;
import android.location.Location;
import android.os.Build;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Switch;
import android.widget.TextView;
import android.widget.Toast;
import com.google.android.gms.location.FusedLocationProviderClient;
import com.google.android.gms.location.LocationCallback;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.location.LocationResult;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.tasks.OnSuccessListener;
import java.util.List;
public class MainActivity extends AppCompatActivity {
    public static final int DEFAULT_UPDATE_INTERVAL = 30;
    public static final int FAST_UPDATE_INVTERVAL = 5;
    private static final int PERMISSIONS_FINE_LOCATION = 99;
```

```
    TextView tv_lat, tv_lon, tv_altitude, tv_accuracy, tv_speed, tv_sensor,
tv_updates, tv_address, tv_wayPointCounts;
    Button btn_newWaypoint, btn_showWayPointList;
    Switch sw_locationupdates, sw_gps;
    FusedLocationProviderClient fusedLocationProviderClient;
    boolean updateOn = false;
    LocationCallback locationCallback;
    LocationRequest locationRequest;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Location currentLocation;
        tv_lat = findViewById(R.id.tv_lat);
        tv_lon = findViewById(R.id.tv_lon);
        tv_altitude = findViewById(R.id.tv_altitude);
        tv_accuracy = findViewById(R.id.tv_accuracy);
        tv_speed = findViewById(R.id.tv_speed);
        tv_sensor = findViewById(R.id.tv_sensor);
        tv_updates = findViewById(R.id.tv_updates);
        tv_address = findViewById(R.id.tv_address);
        sw_gps = findViewById(R.id.sw_gps);
        sw_locationupdates = findViewById(R.id.sw_locationsupdates);
        locationRequest = new LocationRequest();
        locationRequest.setInterval(1000 * DEFAULT_UPDATE_INTERVAL);
            locationRequest.setFastestInterval(1000                    *
    FAST_UPDATE_INVTERVAL);
    locationRequest.setPriority(LocationRequest.PRIORITY_BALANCED_PO
    WER_ACCURACY);
        locationCallback =new LocationCallback() {
            @Override
                public      void      onLocationResult(@NonNull      LocationResult
    locationResult) {    super.onLocationResult(locationResult);
```

```
 updateUIValues(locationResult.getLastLocation());
            }
        };
        sw_gps.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if (sw_gps.isChecked()) {

locationRequest.setPriority(LocationRequest.PRIORITY_HIGH_ACCURAC
Y);
                    tv_sensor.setText("Using GPS sensors");
                }
                else{

locationRequest.setPriority(locationRequest.PRIORITY_BALANCED_POW
ER_ACCURACY);
                    tv_sensor.setText("Using Towers + WIFI");
                }
            }
        });
        sw_locationupdates.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if (sw_locationupdates.isChecked()) {
                    startLocationUpdates();
                }
                else {
                    stopLocationUpdates();
                }
            }
        });
  updateGPS();
 }
```

```
    private void stopLocationUpdates() {
        tv_updates.setText("Location is NOT being tracked");
        tv_lat.setText("Not tracking location");
        tv_lon.setText("Not tracking location");
        tv_speed.setText("not tracking location");
        tv_address.setText("Not tracking location");
        tv_accuracy.setText("Not tracking location");
        tv_altitude.setText("Not tracking location");
        tv_sensor.setText("Not tracking location");
        fusedLocationProviderClient.removeLocationUpdates(locationCallback);
    }
    private void startLocationUpdates() {
        tv_updates.setText(" Location is being tracked");
        FusedLocationProviderClient fusedLocationProviderClient =
this.fusedLocationProviderClient;
        updateGPS();
    }
    @Override
    public void onRequestPermissionsResult(int requestCode, @NonNull
String[] permissions, @NonNull int[] grantResults) {
        super.onRequestPermissionsResult(requestCode, permissions,
grantResults);
        switch (requestCode){
            case PERMISSIONS_FINE_LOCATION:
                if (grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
                    updateGPS();
                }
                else {
                    Toast.makeText(this,"this apps requires permission to be granted
to work properly",Toast.LENGTH_SHORT).show();
                    finish();
                } break;
        } }
```

```
    private void updateGPS(){
        fusedLocationProviderClient =
LocationServices.getFusedLocationProviderClient(MainActivity.this);
        if(ActivityCompat.checkSelfPermission( this,
Manifest.permission.ACCESS_FINE_LOCATION) ==
PackageManager.PERMISSION_GRANTED){


fusedLocationProviderClient.getLastLocation().addOnSuccessListener(this,
new OnSuccessListener<Location>() {
                @Override
                public void onSuccess(Location location) {
                    updateUIValues(location);
                }
            });
        }
        else {
            if(Build.VERSION.SDK_INT >=Build.VERSION_CODES.M){
                    requestPermissions(newString[]
{Manifest.permission.ACCESS_FINE_LOCATION},
PERMISSIONS_FINE_LOCATION);
            }
        }
    }
    private void updateUIValues(Location location) {
        tv_lat.setText(String.valueOf(location.getLatitude()));
        tv_lon.setText(String.valueOf(location.getLongitude()));
        tv_accuracy.setText(String.valueOf(location.getAccuracy()));
        if(location.hasAltitude()){
            tv_altitude.setText(String.valueOf(location.getAltitude()));
        }
        else{
            tv_altitude.setText(" not available");}
```

```
        if(location.hasSpeed()){
            tv_speed.setText(String.valueOf(location.getSpeed()));
        }
        else{
            tv_speed.setText(" not available");
        }

        Geocoder geocoder = new Geocoder(MainActivity.this);
        try {
            List<Address> addresses =
geocoder.getFromLocation(location.getLatitude(),location.getLongitude(), 1);
            tv_address.setText(addresses.get(0).getAddressLine( 0));
        }
        catch (Exception e){
            tv_address.setText("unable to get street address");
        }
    }
}
```

Android Manifest.XML

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
```

```
    android:id="@+id/tv_labellat"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="32dp"
    android:text="Lat:"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

<TextView
    android:id="@+id/tv_lat"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="32dp"
    android:text="0.00"
    app:layout_constraintStart_toEndOf="@+id/tv_labellat"
    app:layout_constraintTop_toTopOf="parent" />

<TextView
    android:id="@+id/tv_labellon"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="16dp"
    android:text="lon:"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/tv_labellat" />

<TextView
    android:id="@+id/tv_lon"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="16dp"
    android:text="0.00"
    app:layout_constraintStart_toEndOf="@+id/tv_labellon"
    app:layout_constraintTop_toBottomOf="@+id/tv_lat" />
```

```
<TextView
    android:id="@+id/tv_labelaltitude"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="16dp"
    android:text="Altitude:"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/tv_labellon" />

<TextView
    android:id="@+id/tv_altitude"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="0.00"
    app:layout_constraintStart_toEndOf="@+id/tv_labelaltitude"
    app:layout_constraintTop_toTopOf="@+id/tv_labelaltitude" />

<TextView
    android:id="@+id/tv_labelaccuracy"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="16dp"
    android:text="Accuracy:"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/tv_labelaltitude" />

<TextView
    android:id="@+id/tv_accuracy"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="0.00"
    app:layout_constraintStart_toEndOf="@+id/tv_labelaccuracy"
    app:layout_constraintTop_toTopOf="@+id/tv_labelaccuracy" />
```

```
<TextView
    android:id="@+id/tv_labelspeed"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="16dp"
    android:text="Speed:"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/tv_labelaccuracy" />

<TextView
    android:id="@+id/tv_speed"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="0.00"
    app:layout_constraintStart_toEndOf="@+id/tv_labelspeed"
    app:layout_constraintTop_toTopOf="@+id/tv_labelspeed" />

<TextView
    android:id="@+id/tv_labelsensor"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="16dp"
    android:text="Sensor:"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/sw_gps" />

<TextView
    android:id="@+id/tv_sensor"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Cell Tower + Wifi"
    app:layout_constraintStart_toEndOf="@+id/tv_labelsensor"
    app:layout_constraintTop_toTopOf="@+id/tv_labelsensor" />
```

```
<TextView
    android:id="@+id/tv_labelupdates"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:text="Updates:"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/sw_locationsupdates" />

<TextView
    android:id="@+id/tv_updates"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Off"
    app:layout_constraintStart_toEndOf="@+id/tv_labelupdates"
    app:layout_constraintTop_toTopOf="@+id/tv_labelupdates" />

<Switch
    android:id="@+id/sw_locationsupdates"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="16dp"
    android:checked="true"
    android:minHeight="48dp"
    android:text="Location Updates"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/divider" />

<Switch
    android:id="@+id/sw_gps"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="16dp"
```

```
        android:minHeight="48dp"
        android:text="GPS/SavePower"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/tv_labelupdates" />

    <TextView
        android:id="@+id/tv_address"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        app:layout_constraintStart_toEndOf="@+id/tv_lbladdress"
        app:layout_constraintTop_toTopOf="@+id/tv_lbladdress" />

    <TextView
        android:id="@+id/tv_lbladdress"
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:text="Address:"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/tv_labelspeed" />
    <View
        android:id="@+id/divider"
        android:layout_width="409dp"
        android:layout_height="1dp"
        android:layout_marginStart="16dp"
        android:layout_marginTop="32dp"
        android:background="?android:attr/listDivider"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLa
```

5.RESULTS

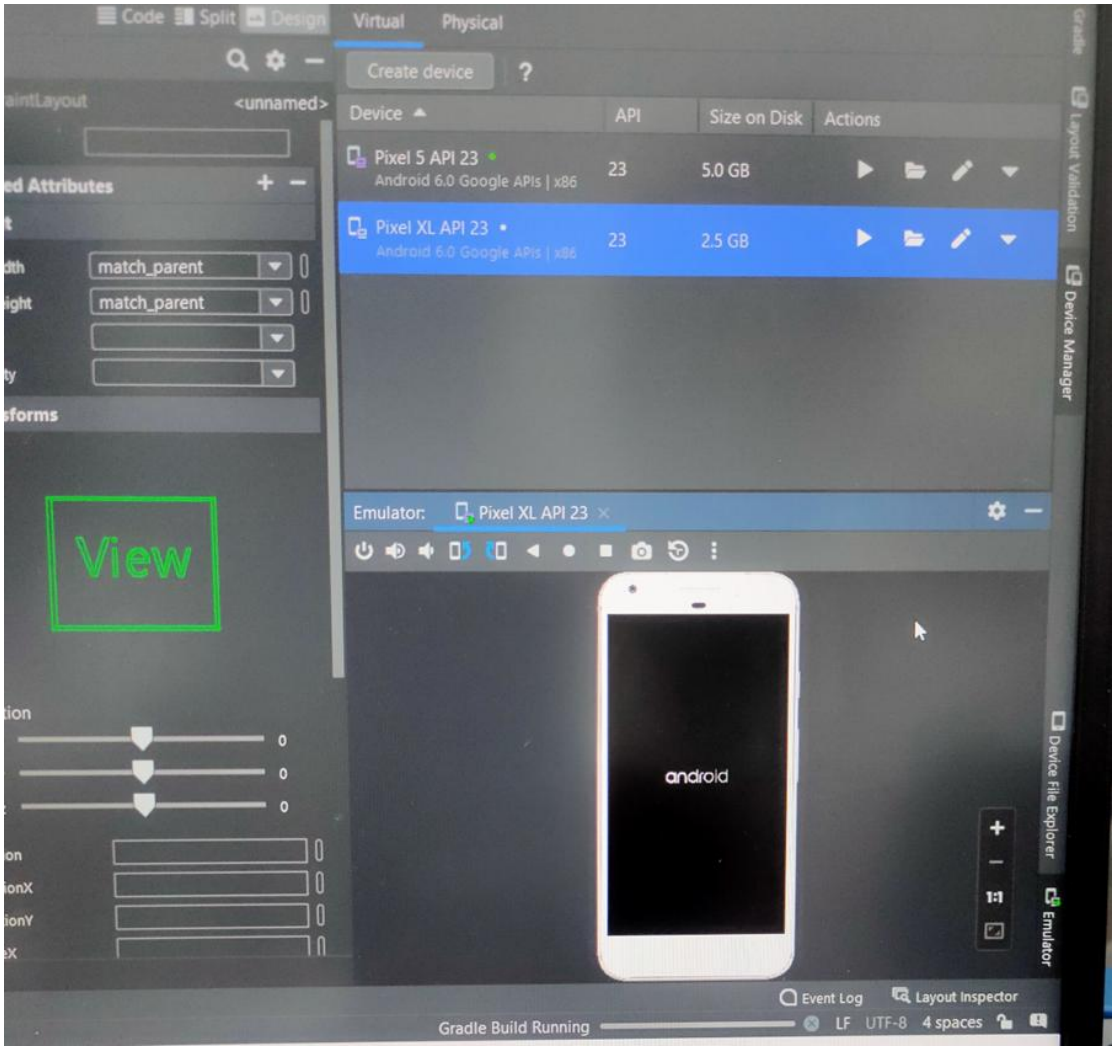

FIGURE 5.1: EMULATOR IN ANDROID STUDIOS

In the above **FIGURE 5.1** the Emulator in Android Studios is build and a device is created named as **PIXEL 6 PRO API 33 .** In this process we are using the advanced Emulator , so that process will be easy to find the location.

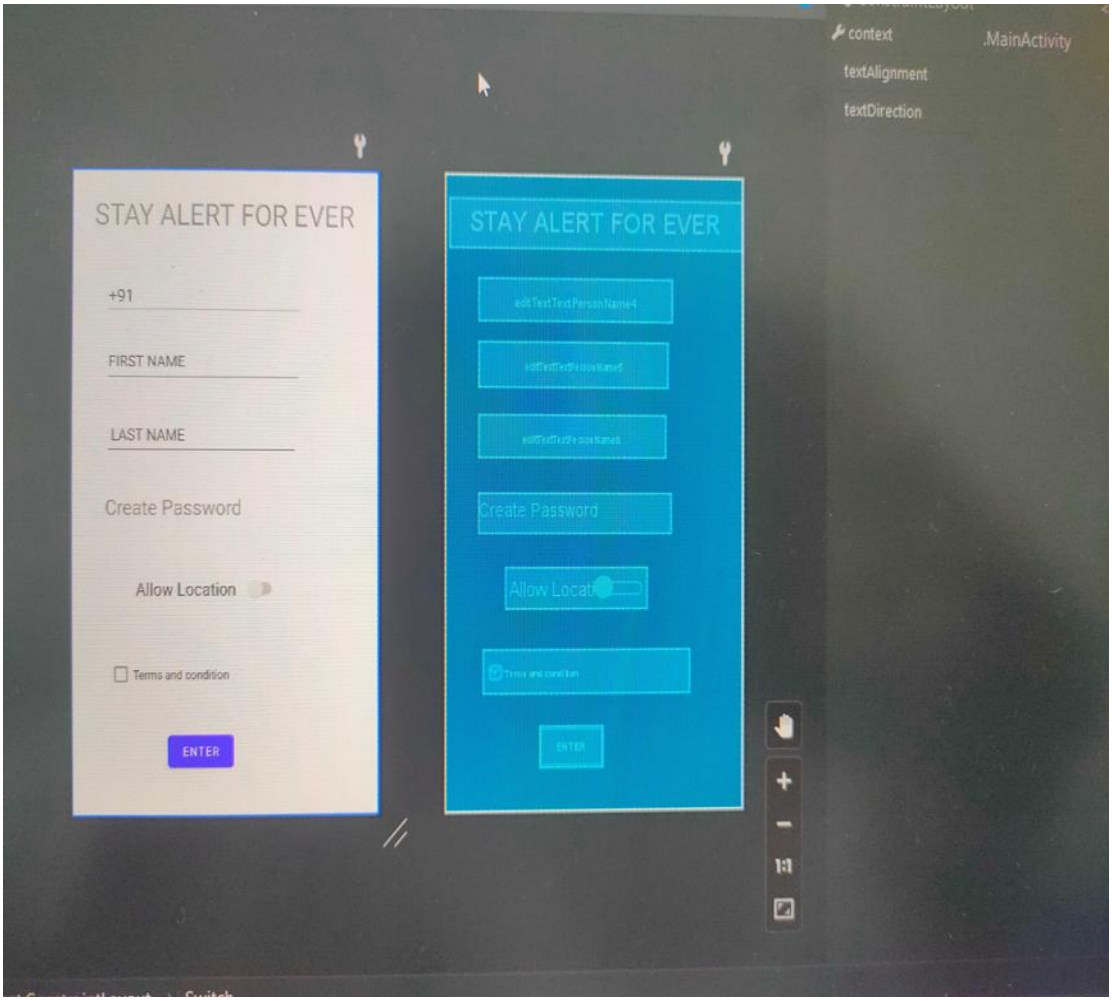

FIGURE 5.2: REGISTRATION PAGE

In the above FIGURE 5.2 is the Login page of the App. Here user can Register and user should enter all the details and should allow location and automatically location will be on by accepting all the permissions.

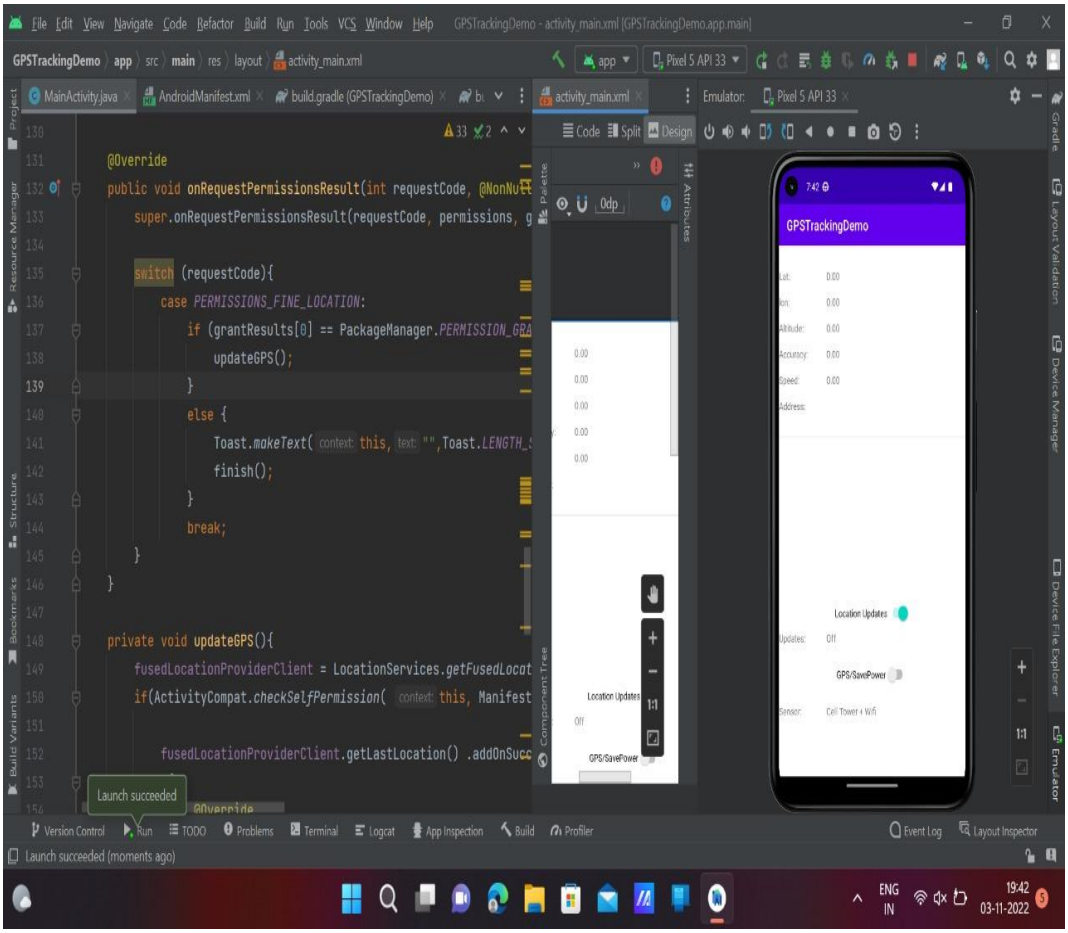

FIGURE 5.3: GPS Activated

In the above FIGURE 5.3 the user enter the destination Location then GPS TRACKING is activated with respect to the user.So that the user able to access the Location.

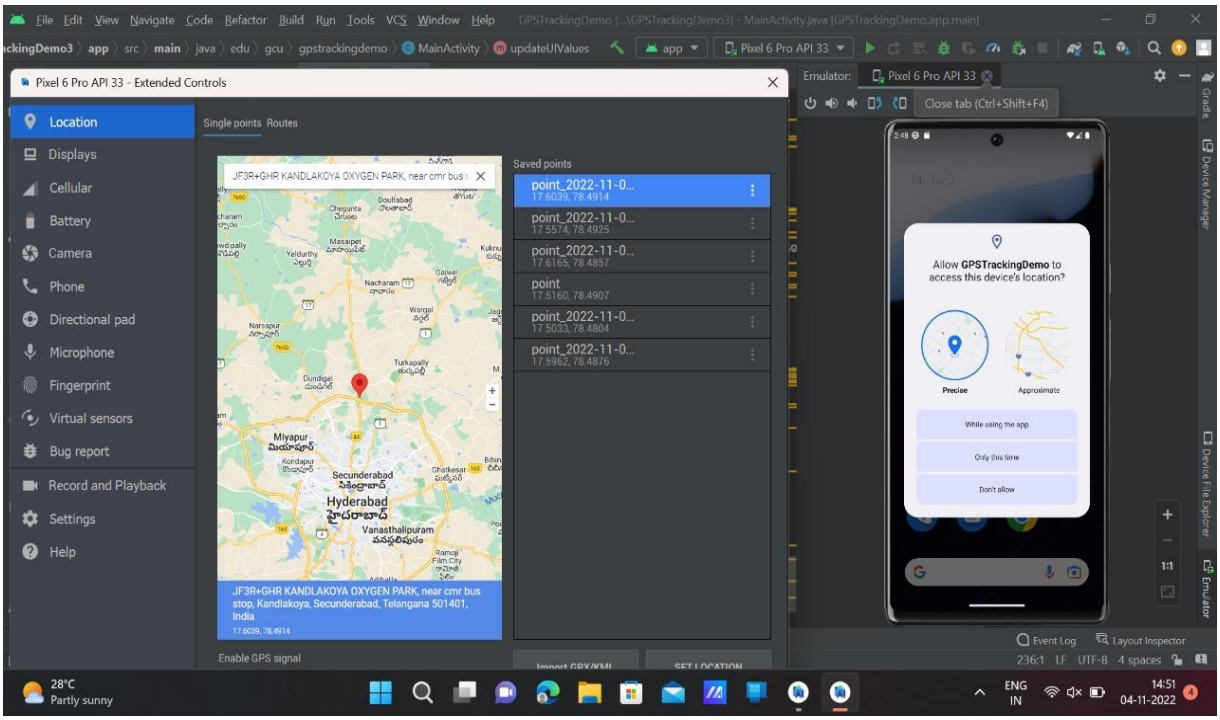

FIGURE 5.4:GPS LOCATION

In the above FIGURE 5.4 User entered location is generated with respect to the database,so that user can view the accident zones points.

6.TESTING

6. TESTING

6.1 INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

6.2 TYPES OF TESTING

6.2.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .It is done after the completion of an individual unit before integration. This is a structural testing that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

6.2.2 INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Integration tests demonstrate that although the components were individually satisfactory, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

6.2.3 FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : Identified classes of valid input must be accepted.
Invalid Input : Identified classes of invalid input must be rejected.
Functions : Identified functions must be exercised.
Output : Identified classes of application outputs Must be exercised.

Systems/Procedures:interfacing systems or procedures must be invoked. Organization and preparation of functional tests is focused on requirements, key functions, or special test cases.

6.3 TEST CASES

6.3.1 CLASSIFICATION

Test case ID	Test case name	Purpose	Input	output
1	ACCIDENT PRONE ZONE	To detect Accident prone zones	The user gives the input in the form of a location through App	An output is Raises voice message
2	Accident zones	To detect Black spots	The user go on a vehicle search for black spots through app	An output Raise voice Message

7.CONCLUSION

7.CONCLUSION & FUTURE SCOPE

7.1 CONCLUSION

This APP is useful to make everyone comfortable to stay safe by proactively raising the voice message. It also reminds the exceeding speed of the vehicle beyond safe speed on that particular road to avoid accidents .This helps to overcome the current challenges to maintain safety measures in the society.This APP definitely plays a key role in decreasing the percentage of road accidents and death rates.

7.2 FUTURE SCOPE

Currently, we have incorporated chosen and selected areas, places all over Telangana /India can be incorporated with the help of Govt. or tie ups with NGO or Organizations .Tie up with Google Maps for wide range of usage and provide safety even without downloading the APP.

8.BIBLIOGRAPHY

8.BIBLIOGRAPHY

REFERENCES

[1] Deeksha Gour , Amit Kanskar, Optimised YOLO:Algorithm for CPU to Detect Road Traffic Accident and Alert System, International Journal of Engineering Research & Technology (IJERT), September 16, 2019.

[2] Xi Jianfeng, Guo Hongyu, Tian Jian, Lisa Liu, Liu Haizhu, A Classification and Recognition Model for the Severity of Road Traffic Accidents , SAGE Journal , May 22 ,2019.

[3] Jabar H. Yousif, Al Rababba Mamoun S, Neural Technique for Predicting Traffic Accidents in Jordan, Journal of American Sciences, November, 2013.

[4] Ajinkya Patankar, Prof S.G Walunj VJER-Vishwakarma Journal of Engineering Research “Blackspot Warning System Identification and Analysis”, Volume1 Issue 4, December2017.

[5] Hah Dhruvit, Shah Pranay “Road Accident Analysis and Identify the black spot location on state highway-5(HalolGodhra Section).

GITHUB LINK

https://github.com/kawshikbhyroju/SAFE