A Project Report on

“Travel Buddy”

By

Prithviraj Patil (501850)
Arnav Pandey (501847)
Pranav Patil (501849)

Guided by:

Prof. Poonam Bari

Department of Information Technology
Fr. Conceicao Rodrigues Institute of Technology
Sector 9A, Vashi, Navi Mumbai – 400703

University of
Mumbai 2021-2022

CERTIFICATE

This is to certify that the project entitled

“Travel Buddy”

Submitted By

Prithviraj Patil
Arnav Pandey
Pranav Patil

In partial fulfilment of the degree of **B.E**. in **Information Technology** for term work of the MAD Mini Project is approved.

External Examiner	**Internal Examiner**
____________________	____________________
External Guide	**Internal Guide**
____________________	____________________
Head of the Department	**Principal**
____________________	____________________
Date: -	**College Seal**

ABSTRACT

This project android Travel Buddy provides the tourist with a city map depending on its current location entered by the android phone user. This information helps the tourists to find the desired locations to visit. Well it consists of entire details of those locations or how to reach the location as well as other emergency amenities like hospitals, institutes, bus stops etc but it provides the basic information to decide the places to visit. This project is mainly beneficial for the tourist's having no idea about the places they want to visit. By providing a geographic based information system the tourists and people shifting to new cities can get a better guidance of the places they want to visit . By making the application GIS based, it includes many advantages as the user can view the required location on a map and accordingly estimate the time that will be required to reach the final destination. The system gives the basic details that will be required such as an image of that place along with basic details like the address, contact no etc.

INDEX

LIST OF FIGURES

1. Introduction

We use Android phones on a regular basis because of features such as GPS, GSM, GCS, computing capability, and internet connection. There are many Android applications available these days that assist users in finding solutions to a variety of challenges that they face in their daily lives. We propose an architecture for an Android-based tourist guide system that may give tourism information to mobile users in a comfortable manner. It may query information for beautiful views, restaurants, or top-rated sites, among other things, and provide many outputs, making it more useful. This tourist guide system based on Android is simple, efficient, faster, and more customizable to meet all of the tourist's needs.

1.1 Problem Statement

In this project we are creating an android app that helps the user as a guide to travel around in a city. To access the application, you must first register and then login to the system. Following that, you can choose which places to visit. After choosing a place to visit, the user will get all the categories of locations like cafe, restaurant, etc. It will then show all the locations of selected categories sorted in the order of ratings of that location. Once you click the desired location, it will automatically display the details and ratings of that location.

2. Literature Review

2.1 Existing Systems

[1]*Dynamic Tour Guide:* Context-driven mobile Travel Buddy that has been developed for Mobile operating systems. The study presents methodology, implementation and evaluation of mobile Travel Buddy.The paper presents a smart space-based tourist recommendation system that allows acquiring information about places of interests around the tourist from different internet sources (like wikipedia, wikivoyage, wikitravel, panoramio, flickr). The system implements ranking acquired attractions according to the tourist preferences and current situation in the tourist location. Tourists can rate attractions that they like or dislike. Based on these ratings a recommendation service clusters tourists into groups with similar interests and uses evaluations of tourists belonging to the same group for ranking attractions around the tourist. The paper presents a prototype service for these purposes that is based on smart space technology. The prototype has been developed for Android devices and is available for free downloaded from Google Play market.

[2]*PSiS:* PSiS is a tour planning support system that aims to provide the tourist a visit plan combining, in a tour, the most adequate tourism products, namely interesting places to visit, attractions, restaurants and accommodation, according to tourists' specific profile (which includes interests, personal values, wishes, constraints and disabilities).Functioning and transportation schedules are also considered to generate a tour planning. Before the trip a tourist interacts with the system through the special web application and during the trip it is possible to use the special mobile application for Android-based smartphones.

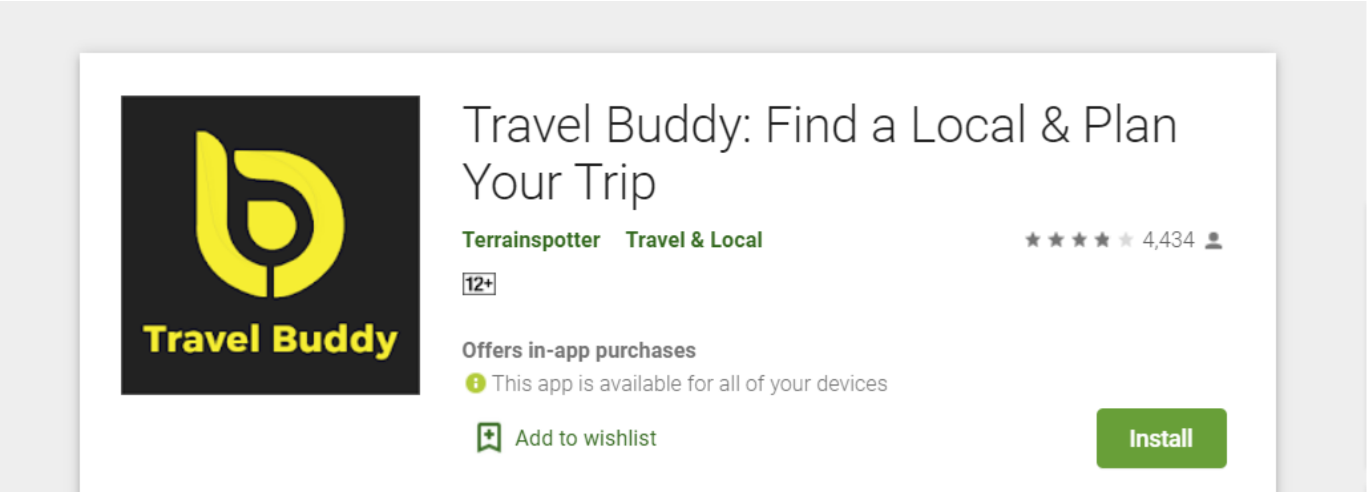
[3]*Triposo:* The travel guide Triposo is a free mobile guide service available for Apple and Android devices. A user can download the application and appropriate database (which is updated once every two months) to the mobile device beforehand and use it during the trip without Internet connection. The application supports logging of trav-elling. It includes databases from the following sources: World66, Wikitravel, Wikipe-dia, Open Street Maps, TouristEye, Dmoz, Chefmoz and Flickr. Each guide contains information on sightseeing, nightlife, restaurants and more.

[4]*Smart Travelling:* Online travel guide that supports about 30 cities world-wide including the most interesting destinations in European countries and USA. The guide includes a database of restaurants, cafes, hotels, shopping-tips and other places of interest. The mobile application for iPhone is accessible in AppStore. Integration with Google maps allows users to see the current location in the map and helps to navigate to each and every tip in destination cities. Application allows the user to down-load the content and use a guide without Internet connection.

2.2 Statistics

Travel Buddy: Find a Local & Plan Your Trip

Travel Buddy has 500 thousand active users worldwide. Travel buddy is ranked as the most used mobile traveler guide app in the world.With Travel Buddy app, you can couch-surf with your local guide. You can discover the trail of Manali with a tour companion who knows it better than any travel agent.You can use this smart social app to make travel friends and find local guides. Travel Buddy also organizes online & offline travel events.



Travel Buddy: Find a Local & Plan Your Trip

Terrainspotter Travel & Local ★★★★☆ 4,434

12+

Offers in-app purchases

This app is available for all of your devices

Figure 2.1 - Travel Buddy - Basic Application Details

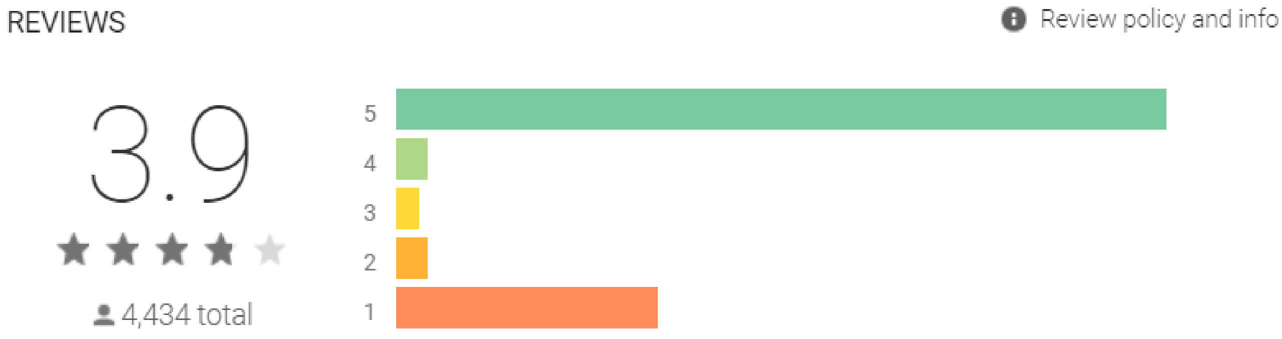


Figure 2.2 -Travel-Buddy- Review Statistics

Parindey - Find a Travel Partner & Plan Your Trip

Parindey provides you the facility to find a local, travel partner or travel buddy. Share your travel moments , video shots, short vlog videos. Get more engagement on your travel shots, plan a trip with others and connect them via chat.

Features of Parindey - Travel Partner Finder app in India

* Share travel photos and videos with others.

* Find other travellers near your location.

* Find locals when you are travelling.

* Connect with others with real time chat.

* Share short vlog videos.

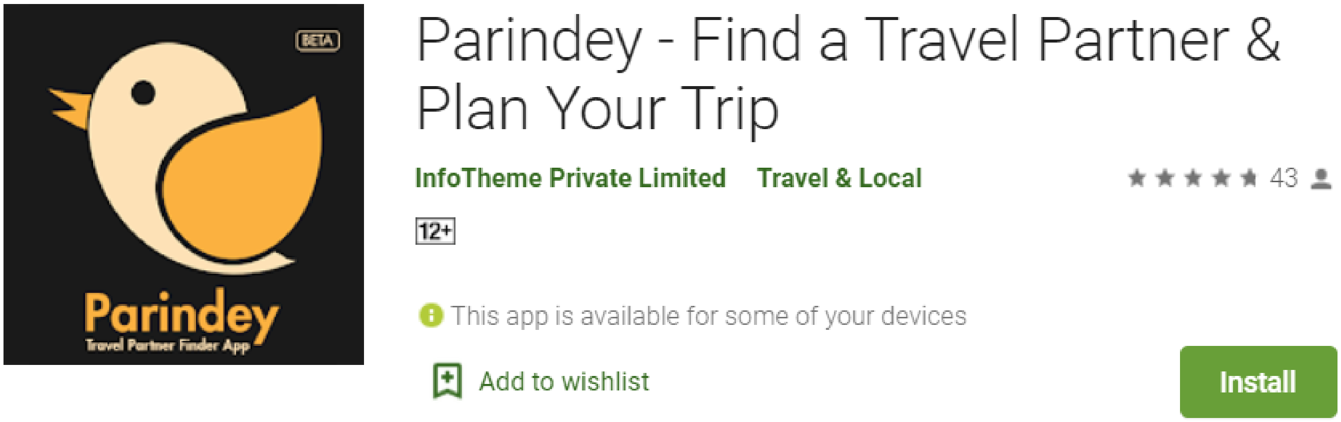
* Comment, Like and share others posts.

* Local businesses can promote their services with others.

* Traveller + Traveller = Your Trip.

* Easy to use, awesome user interface.

* No membership, Always Free to use.



Parindey - Find a Travel Partner & Plan Your Trip

InfoTheme Private Limited Travel & Local

This app is available for some of your devices

Figure 2.3 - Parindey- Basic Application Details

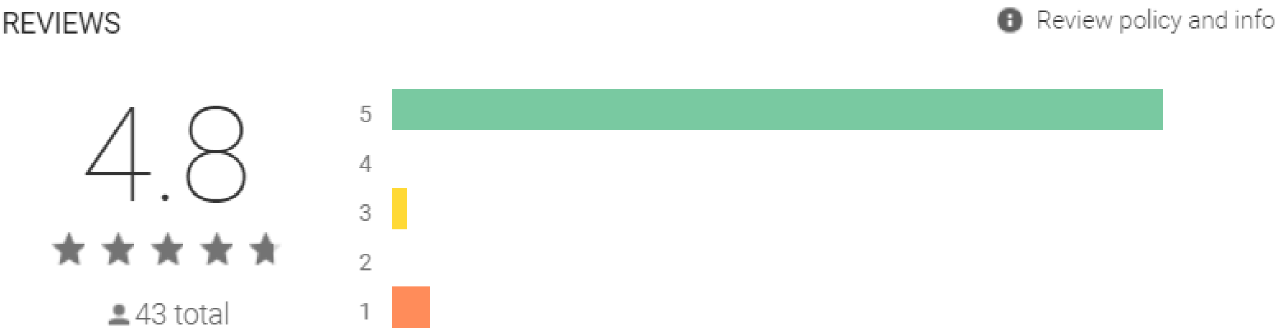


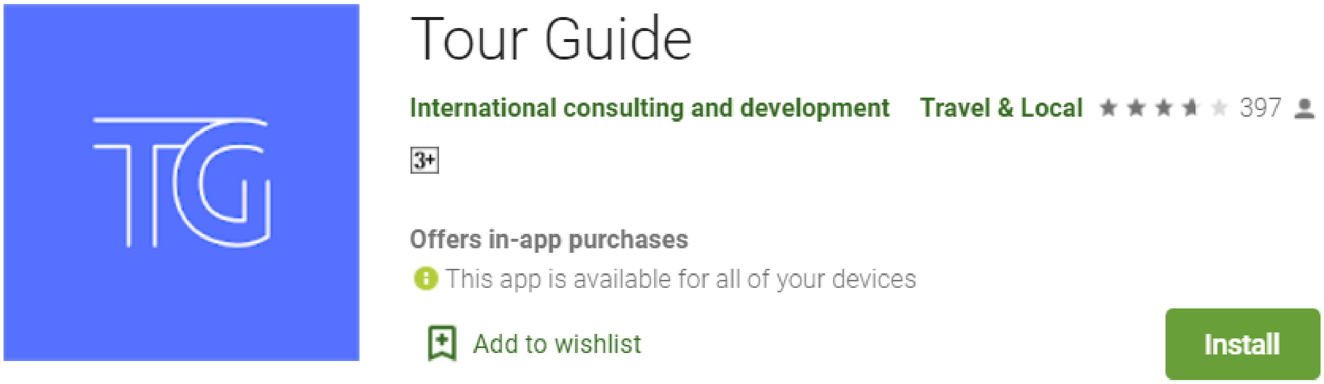
Figure 2.4 - Parindey- Review Statistics

Tour Guide

You want to see places out of the boring tourist brochures, feel domestic life and the pure emotion of living? To meet new people, have fun, learn something new and original?

Make most of every single trip you make across the globe using our TourGuide app.

TourGuide is a mobile application where tourists can find local people willing to help them discover the most interesting parts of their world.



Tour Guide

International consulting and development Travel & Local ★★★★☆ 397

3+

Offers in-app purchases

This app is available for all of your devices

Figure 2.5 - Tour Guide App - Basic Application Details

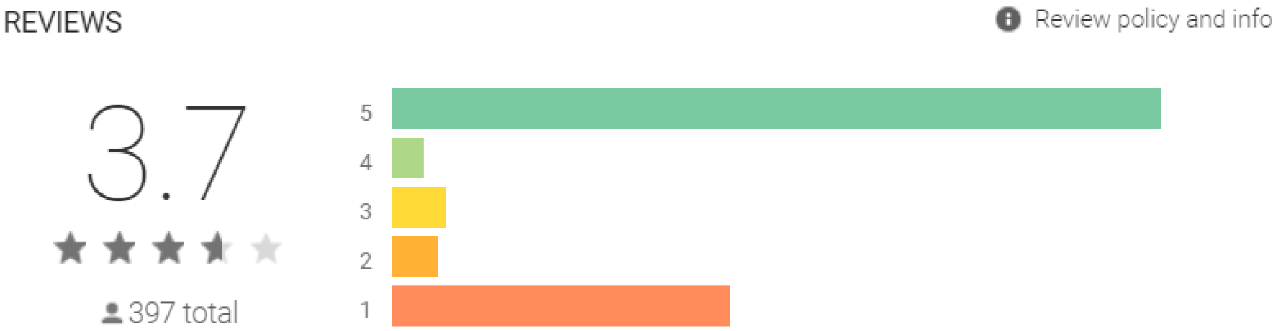


Figure 2.6 - Tour Guide App - Review Statistics

3. Architecture Flow

3.1 Block Diagram

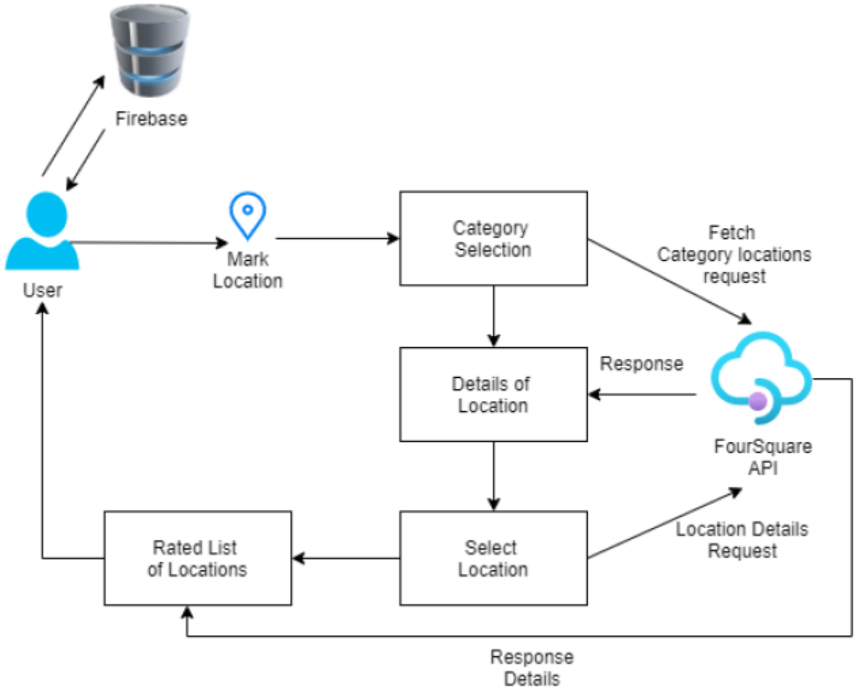


Figure 3.1 - Block Diagram of Tourist Buddy Application

As shown in the figure a new user has to select a location. For this, first, the user will be given categories of locations(restaurants, scenic views, etc). This list will be shown in the order of ratings of the location which will help the user to select the best of the locations. Now, users can select the location and the details will be shown for the same.

3.2 Flow Diagram

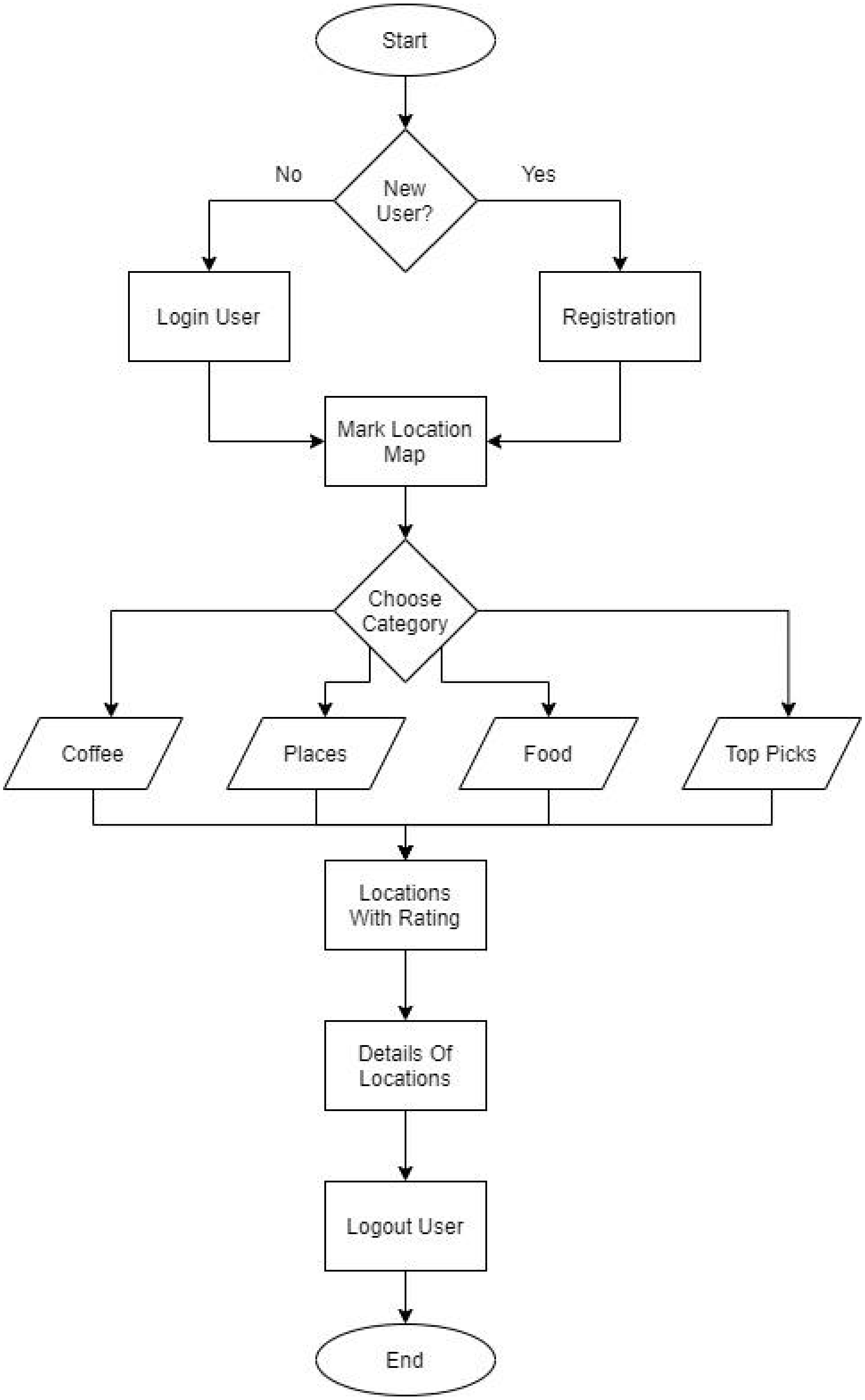


Figure 3.2 - Flow Diagram of Tourist Buddy Application

The Life cycle of application starts when user opens the application ,he/she is prompted to either login if the user is already registered else the user has to first register and then login Continuing,next the user is prompted with a map the user can search for a particular region that the user wishes to visit or the user can pinpoint the location on the map that the user wishes to visit . After selecting a location the user is prompted with the categories that the application has to offer , the user is given the options for coffee,places,food or top places. The User has to select one of the options that he/she wishes to look for ,that is coffee for the available places selling coffee in nearby places with respect to the location opted by the user,places are the popular nearby destinations for tourist to visit,food option will give users best options for restaurants or hotels . Then the users will be given nearby best locations with ratings and feedback from others along with the details of the location,then the user can log out from the application.

4. Software and Hardware Requirements

4.1 Software and Hardware Requirements

1. Android Studio
2. Processor i3 and above
3. 4GB Ram
4. Java JDK

4.2 Features of Android Application

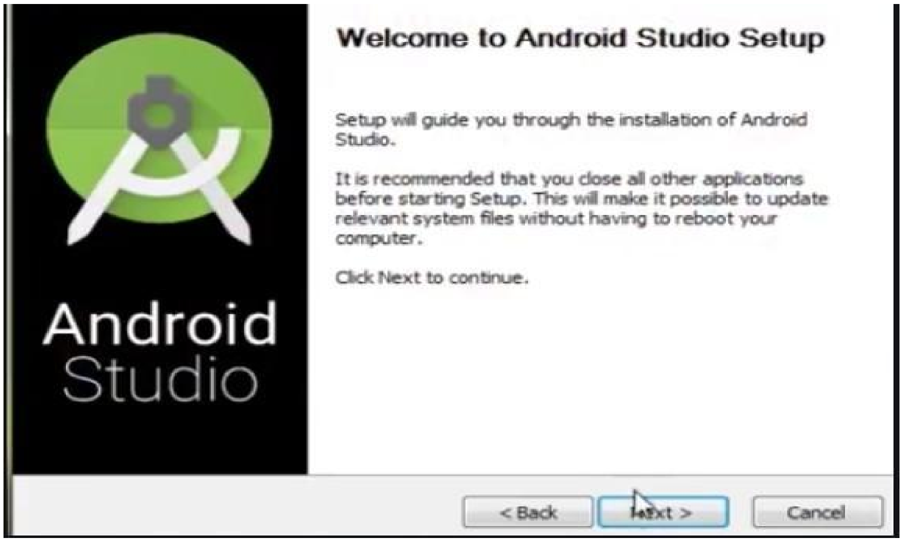
1. The user has the privilege to create an account and log in.
2. Authentication is provided to allow only registered users to log in.
3. Users can update their profile name, about them or the profile photo.
4. Since the location can be viewed on a map, the user can even zoom in and zoom out to get a better view.
5. The usage of this application greatly reduces the time required to search for a place.
6. This system can be used to view the location in a map that the user wishes to reach.
7. The user can also find the paths to follow to reach the final destination in a map which gives a better view to the users.
8. The application can prove very beneficial to the tourist who have very little or completely no idea about the places to visit.

5. Software and Hardware Requirements

5.1 Steps to install android

Download the executable file from this link.

After the downloading has finished, open the file from downloads and run it. It will prompt the following dialog box. Click on next. In the next prompt, it'll ask for a path for installation. Choose a path and hit next.



Welcome to Android Studio Setup

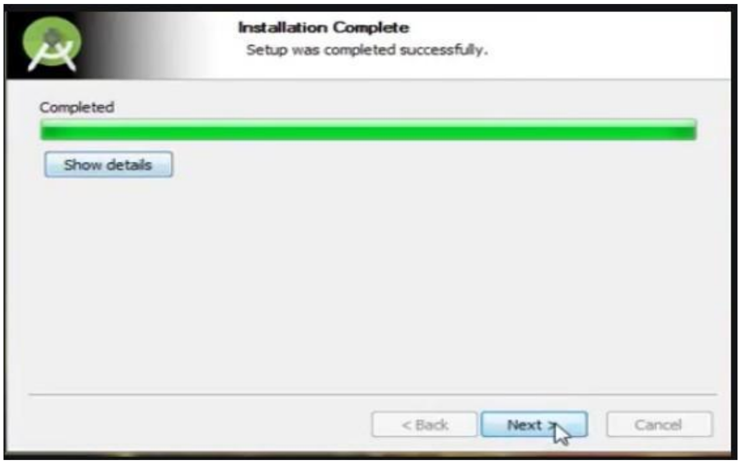
Setup will guide you through the installation of Android Studio.

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

Fig 5.1 Installation Start

It will start the installation, and once it is completed, it will be like the image shown below.



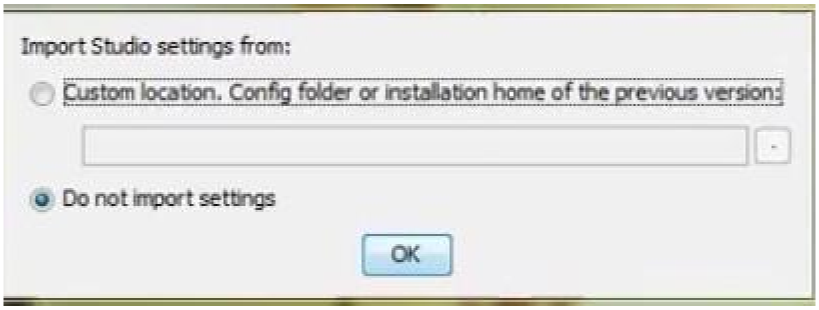
Installation Complete

Setup was completed successfully.

Completed

Fig 5.2 Installation Complete

Once "Finish" is clicked, it will ask whether the previous settings need to be imported [if the android studio had been installed earlier, or not. It is better to choose the 'Don't import Settings option'.



Import Studio settings from:

○ Custom location. Config folder or installation home of the previous version:

● Do not import settings

OK

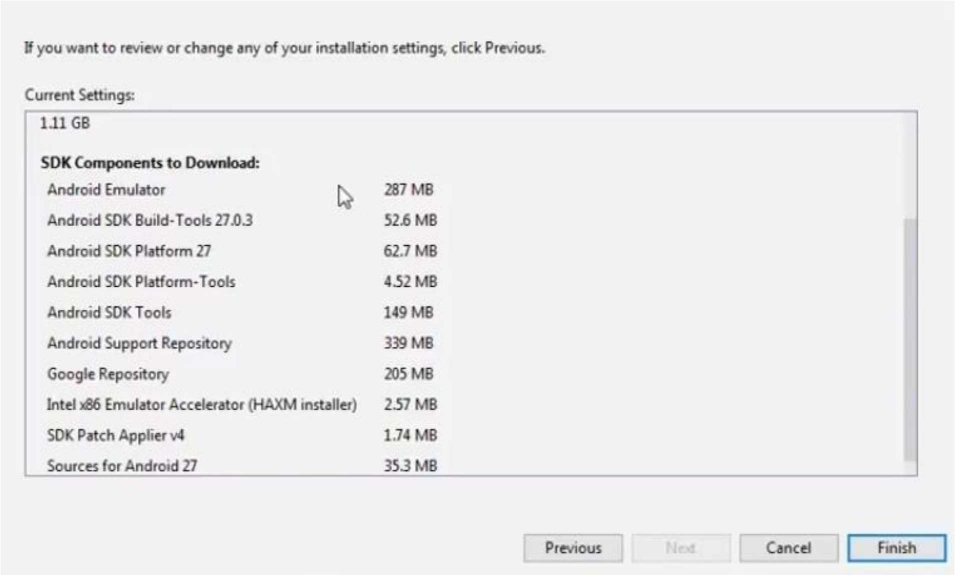
Fig 5.3 Settings

This will start the Android Studio.

FIg 5.4 Loading

After it has found the SDK components, it will redirect to the Welcome dialog box.

Now it is time to download the SDK components.



If you want to review or change any of your installation settings, click Previous.

Current Settings:

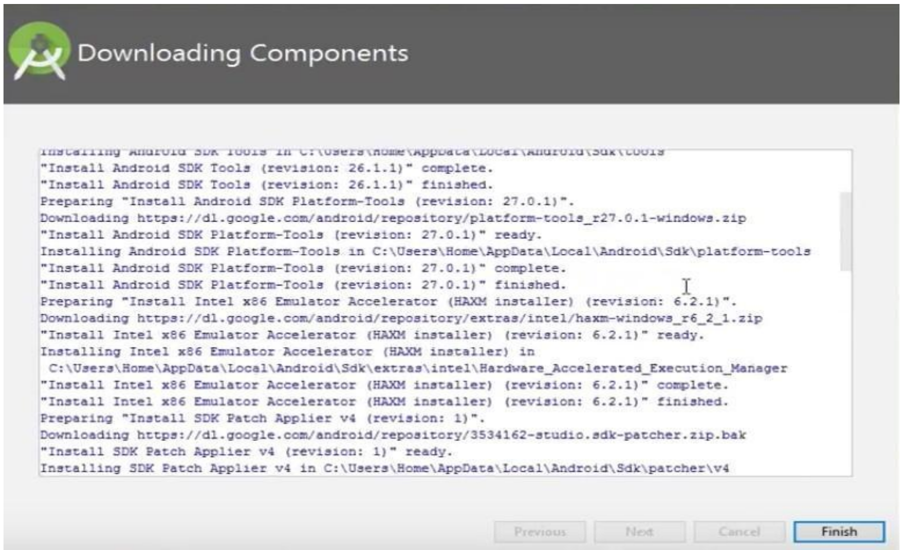
1.11 GB

SDK Components to Download:

Android Emulator	287 MB
Android SDK Build-Tools 27.0.3	52.6 MB
Android SDK Platform 27	62.7 MB
Android SDK Platform-Tools	4.52 MB
Android SDK Tools	149 MB
Android Support Repository	339 MB
Google Repository	205 MB
Intel x86 Emulator Accelerator (HAXM installer)	2.57 MB
SDK Patch Applier v4	1.74 MB
Sources for Android 27	35.3 MB

Fig 5.5 Preview Settings

Click on Finish. Components begin to download and let it complete.



Downloading Components

```
Installing Android SDK Tools in C:\Users\Home\AppData\Local\Android\Sdk\tools
"Install Android SDK Tools (revision: 26.1.1)" complete.
"Install Android SDK Tools (revision: 26.1.1)" finished.
Preparing "Install Android SDK Platform-Tools (revision: 27.0.1)".
Downloading https://dl.google.com/android/repository/platform-tools_r27.0.1-windows.zip
"Install Android SDK Platform-Tools (revision: 27.0.1)" ready.
Installing Android SDK Platform-Tools in C:\Users\Home\AppData\Local\Android\Sdk\platform-tools
"Install Android SDK Platform-Tools (revision: 27.0.1)" complete.
"Install Android SDK Platform-Tools (revision: 27.0.1)" finished.
Preparing "Install Intel x86 Emulator Accelerator (HAXM installer) (revision: 6.2.1)".
Downloading https://dl.google.com/android/repository/extras/intel/haxm-windows_r6_2_1.zip
"Install Intel x86 Emulator Accelerator (HAXM installer) (revision: 6.2.1)" ready.
Installing Intel x86 Emulator Accelerator (HAXM installer) in
 C:\Users\Home\AppData\Local\Android\Sdk\extras\intel\Hardware_Accelerated_Execution_Manager
"Install Intel x86 Emulator Accelerator (HAXM installer) (revision: 6.2.1)" complete.
"Install Intel x86 Emulator Accelerator (HAXM installer) (revision: 6.2.1)" finished.
Preparing "Install SDK Patch Applier v4 (revision: 1)".
Downloading https://dl.google.com/android/repository/3534162-studio.sdk-patcher.zip.bak
"Install SDK Patch Applier v4 (revision: 1)" ready.
Installing SDK Patch Applier v4 in C:\Users\Home\AppData\Local\Android\Sdk\patcher\v4
```

Fig 5.6 Download Components

The Android Studio has been successfully configured. Now it’s time to launch and build apps. Click on the Finish button to launch it.

Click on Start a new Android Studio project to build a new app.

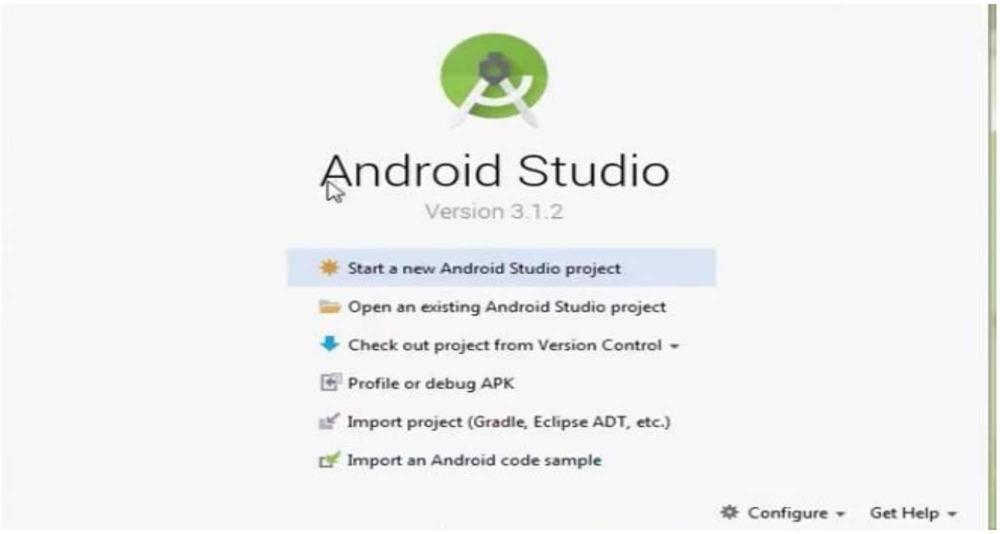

Android Studio
Version 3.1.2
Start a new Android Studio project
Open an existing Android Studio project
Check out project from Version Control
Profile or debug APK
Import project (Gradle, Eclipse ADT, etc.)
Import an Android code sample
Configure
Get Help

Fig 5.7 Open Android Studio

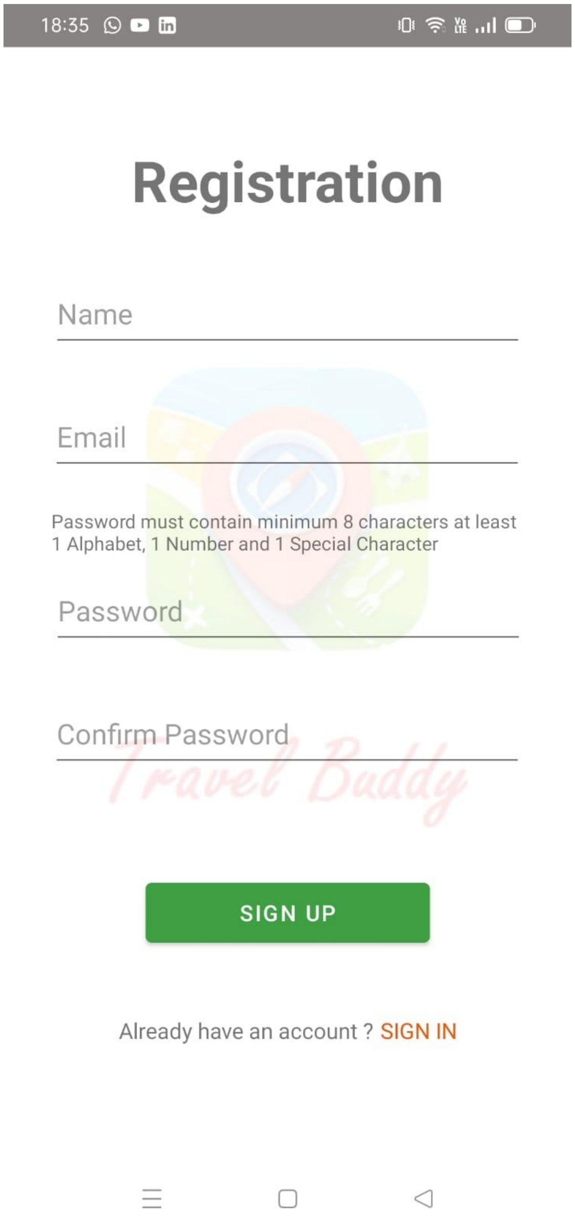
5.2 Explanation of Android manifest file of your app

List of Activities:

1. HomeActivity - Category selection and mark location for user
2. LocationListing - List of all nearby locations according to selected category
3. Details - Rating, address , description of particular location
4. SignIn - SignIn Activity contains the page for Sign in
5. SignUp - SignUp Activity contains the page for Sign up
6. MainActivity - Main Activity is used as a Page Viewer

Intent-filter
Action: android.intent.action.MAIN
Category: android.intent.category.LAUNCHER

6. Graphical User Interface



Password must contain minimum 8 characters at least 1 Alphabet, 1 Number and 1 Special Character

Fig 6.1 Sign up

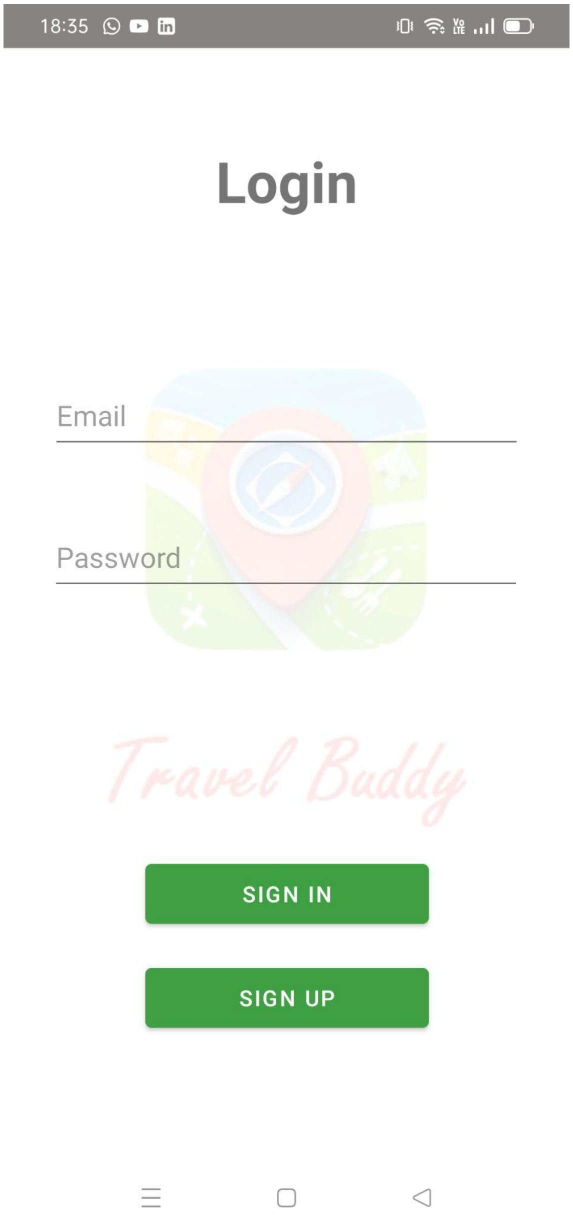


Fig 6.2 Sign In

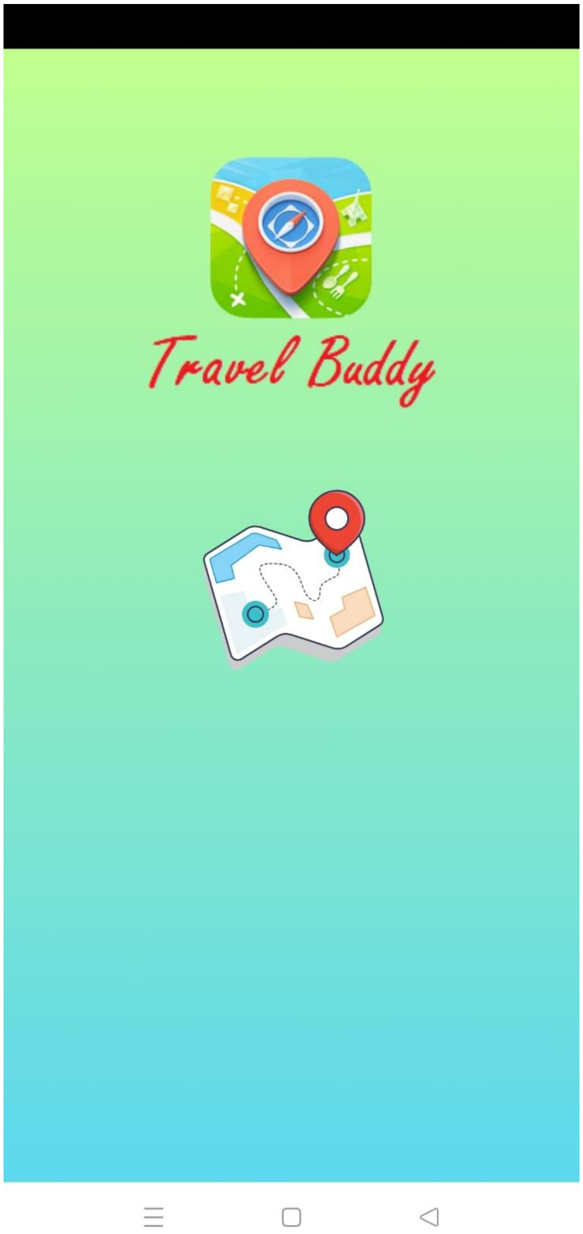


Fig 6.3 Splash Screen

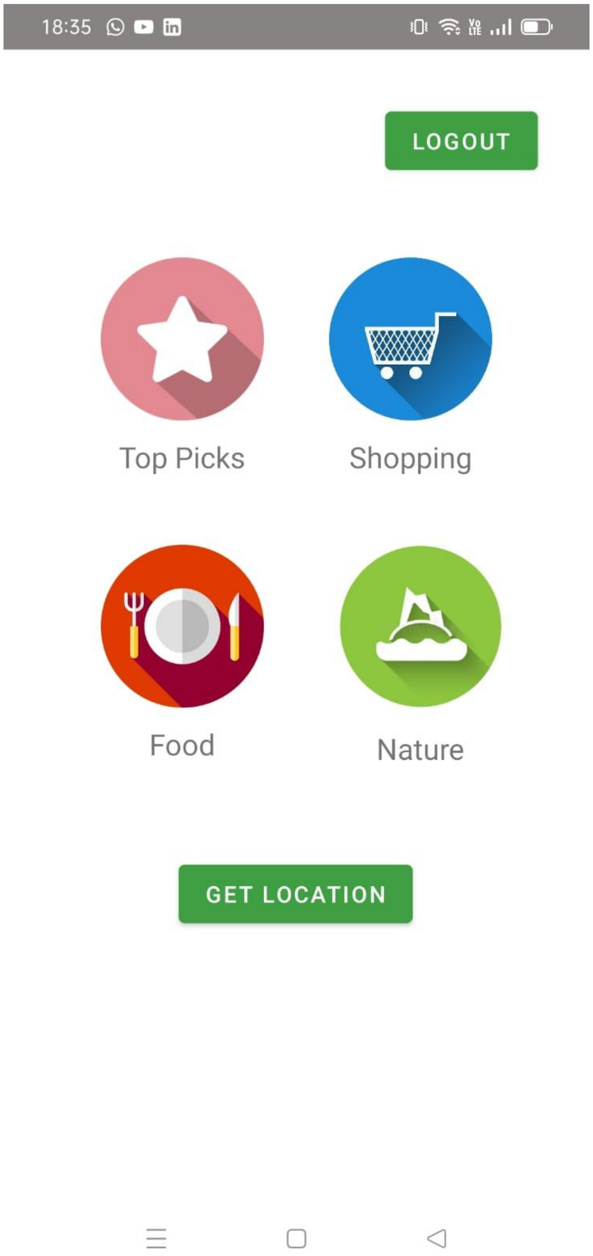
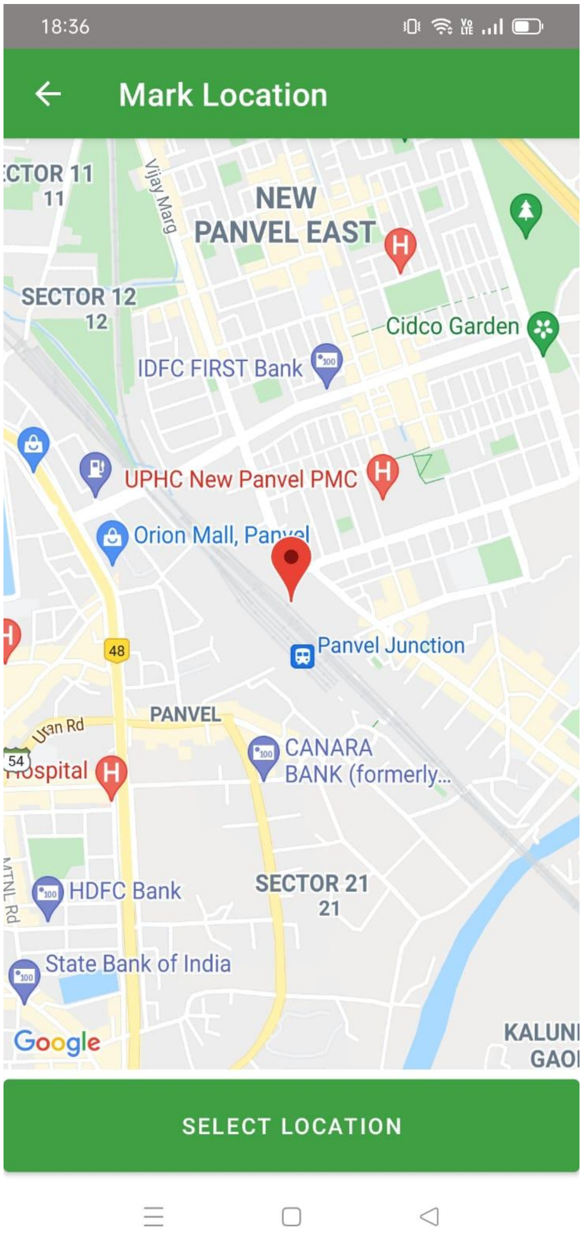
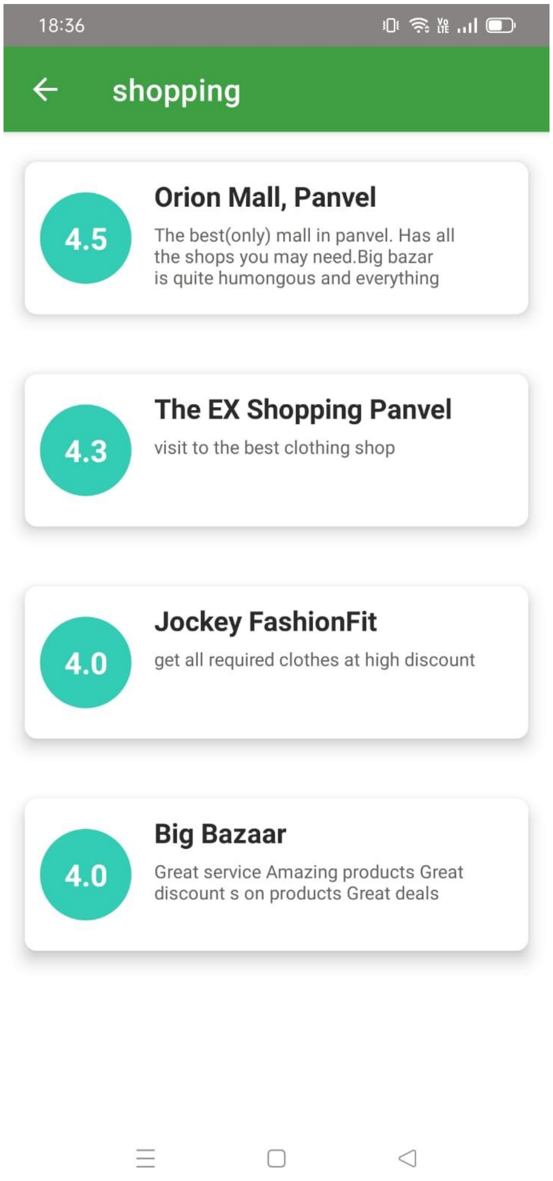


Fig 6.4 Home Screen





4.5 **Orion Mall, Panvel**
The best(only) mall in panvel. Has all the shops you may need.Big bazar is quite humongous and everything

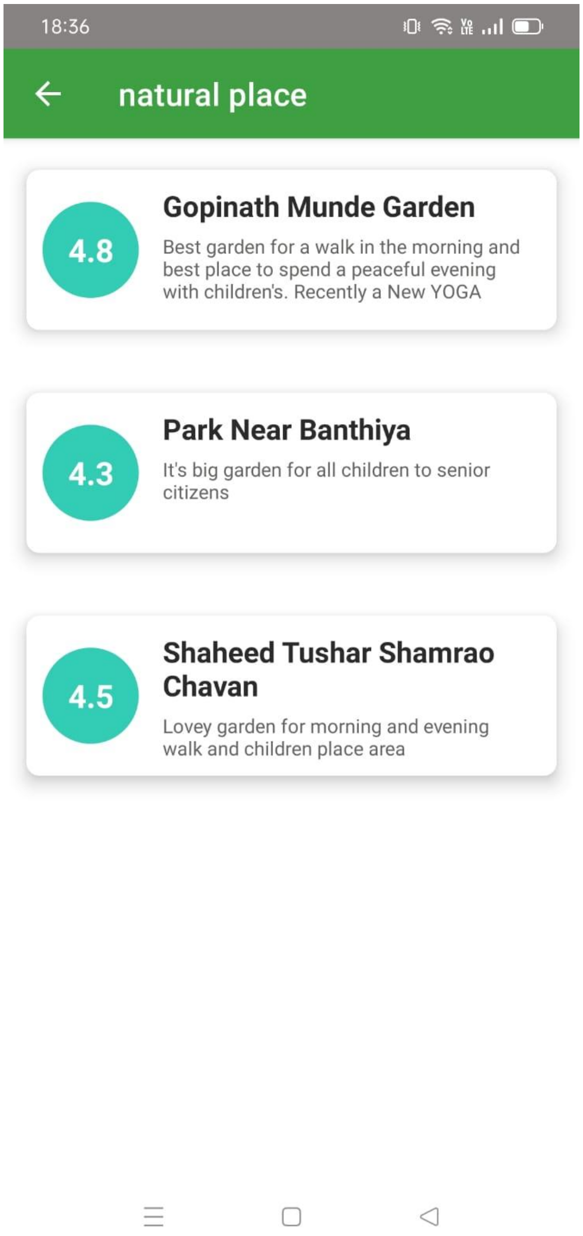
4.3 **The EX Shopping Panvel**
visit to the best clothing shop

4.0 **Jockey FashionFit**
get all required clothes at high discount

4.0 **Big Bazaar**
Great service Amazing products Great discount s on products Great deals

Fig 6.5 Mark Location Screen

Fig 6.6 Shopping listing



18:36

natural place

4.8 **Gopinath Munde Garden**

Best garden for a walk in the morning and best place to spend a peaceful evening with children's. Recently a New YOGA

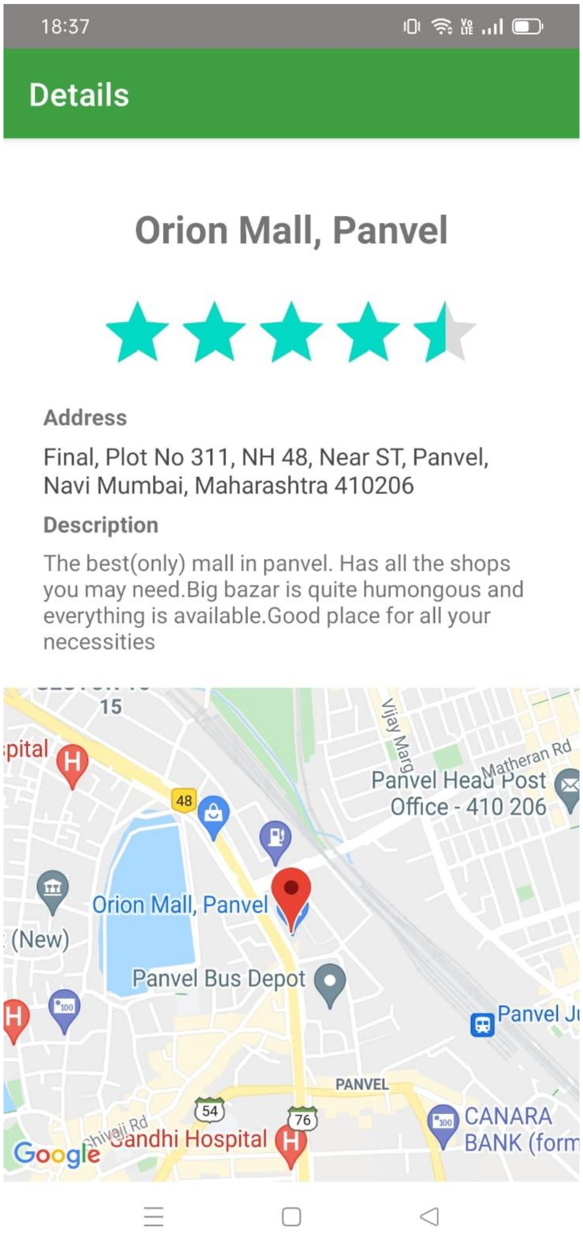
4.3 **Park Near Banthiya**

It's big garden for all children to senior citizens

4.5 **Shaheed Tushar Shamrao Chavan**

Lovey garden for morning and evening walk and children place area

Fig 6.7 Natural Place Listing



18:37

Details

Orion Mall, Panvel

★★★★½

Address

Final, Plot No 311, NH 48, Near ST, Panvel,
Navi Mumbai, Maharashtra 410206

Description

The best(only) mall in panvel. Has all the shops you may need.Big bazar is quite humongous and everything is available.Good place for all your necessities

Fig 6.8 Details Screen

7. Conclusion and Future Scope

7.1 Conclusion

Android tourist guide application based on the internet allows users to explore nearby locations to the selected location .User can filter out locations based on categories provided by Tourist guide application. Firebase is a platform and a tool that is known for its speed and reliability in terms of the time it takes for building applications that are real-time with a highly simpler platform, many of the Google features are carried forward along with other advanced features like crash reporting and thereby allowing the developers to create critical and more functional applications providing a wide variety of services. Hence Firebase is used as a database.Google maps api used for map functionality of application.

7.2 Future Scope

The application can further incorporate various features that will increase the usability of the application. Some of these features are as below:

- ➔ Expense manager to distribute and manage expenses among the group of people
- ➔ Add images of places
- ➔ Users will be able to rate that place through our app
- ➔ Users will get directions to the destination

References

[1] Smart Space-Based Tourist Recommendation System Alexander Smirnov Alexey Kashevnik,Andrew Ponomarev,Nikolay Teslya,Maksim Shchekotov,Sergey I. Balandin https://link.springer.com/chapter/10.1007/978-3-319-10353-2_4

[2] https://www.researchgate.net/publication/259143216_Mobile_application_to_provide_personalized_sightseeing_tours Mobile application to provide personalized sightseeing tours Ricardo Anacleto,Lino Figueiredo,AnaAlmeida,PauloJorgeNovais

[3] https://www.triposo.com/

[4] https://www.smart-travelling.net/en/

[5] Parindey - Find a travel Partner and plan your trip https://play.google.com/store/apps/details?id=com.travelpartner.parindey

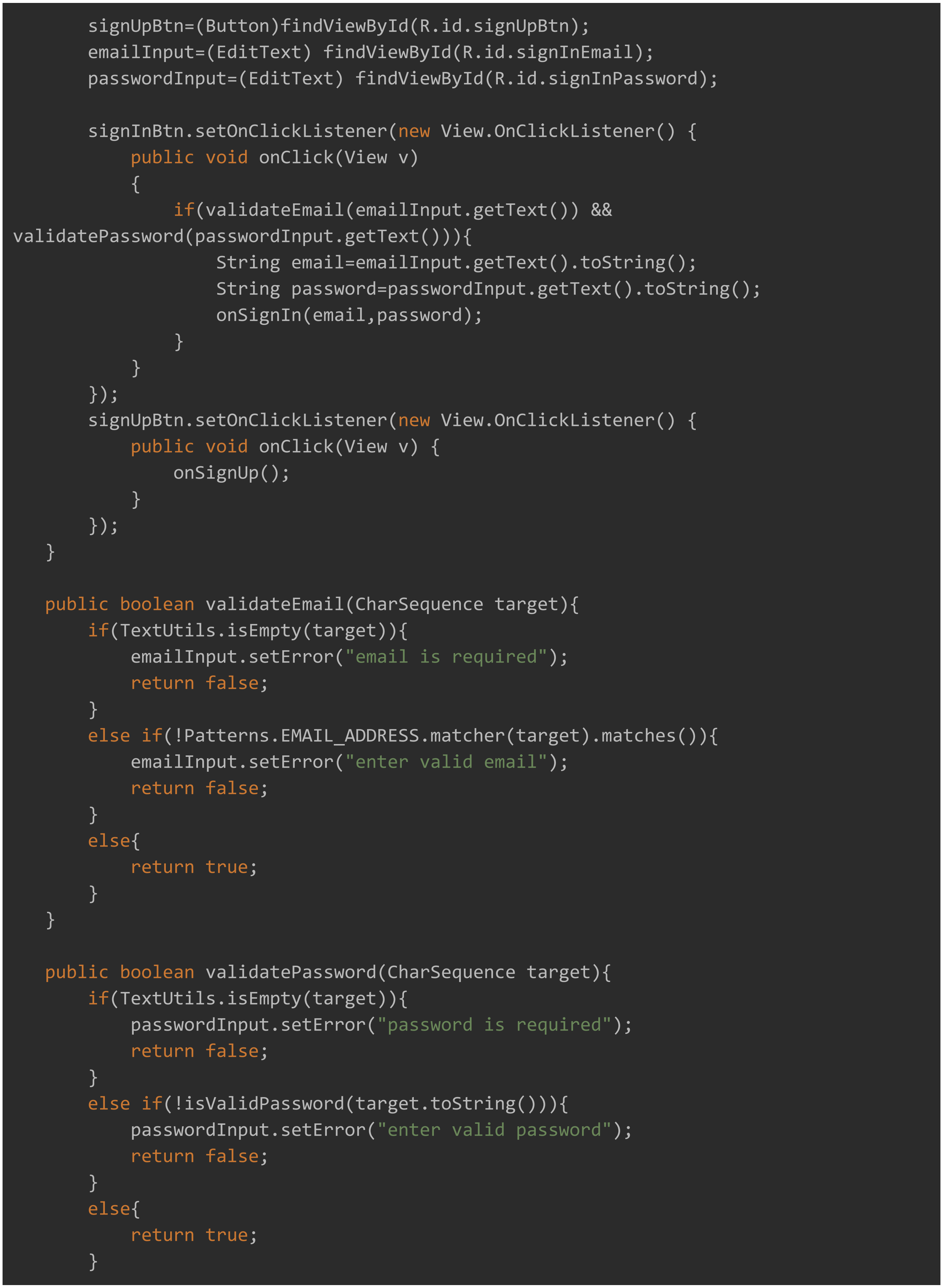
[6] Travel - Buddy https://play.google.com/store/apps/details?id=com.beatravelbuddy.travelbuddy

[7] Tour Guide https://play.google.com/store/apps/details?id=com.icd.tourguide

Appendix: Code Sample

MainActivity.java

```
package com.example.touristguide;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.content.ContextCompat;

import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.util.Patterns;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;

import java.util.regex.Matcher;
import java.util.regex.Pattern;

public class MainActivity extends AppCompatActivity {
   Button signInBtn,signUpBtn;
   EditText emailInput,passwordInput;

   private FirebaseAuth mAuth;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
       super.onCreate(savedInstanceState);
       this.getSupportActionBar().hide();

getWindow().setStatusBarColor(ContextCompat.getColor(MainActivity.this,R.color.colorAccent));
       setContentView(R.layout.activity_main);

       mAuth = FirebaseAuth.getInstance();

       signInBtn=(Button) findViewById(R.id.signInBtn);
```

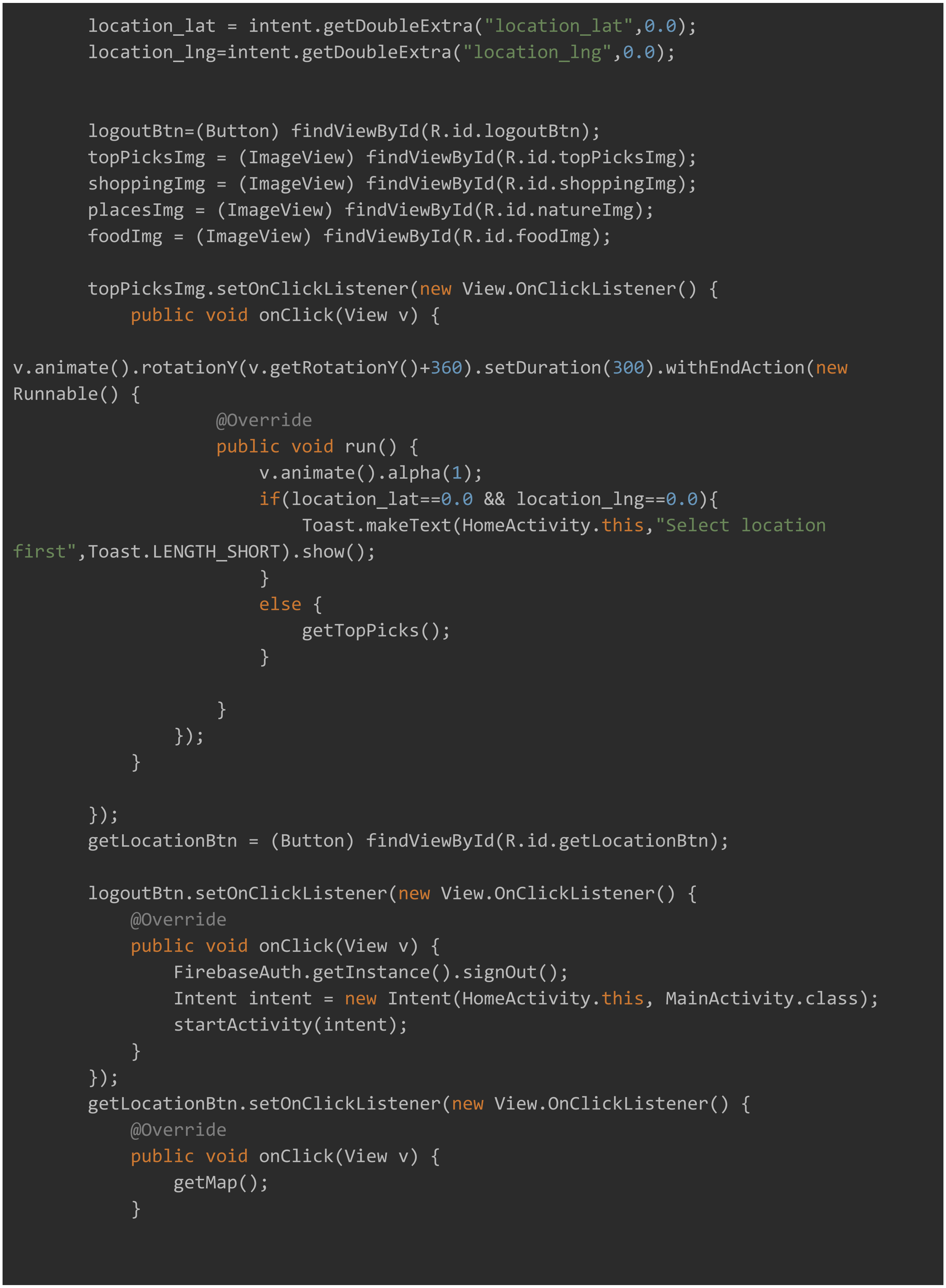


```
        signUpBtn=(Button)findViewById(R.id.signUpBtn);
        emailInput=(EditText) findViewById(R.id.signInEmail);
        passwordInput=(EditText) findViewById(R.id.signInPassword);

        signInBtn.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v)
            {
                if(validateEmail(emailInput.getText()) &&
validatePassword(passwordInput.getText())){
                    String email=emailInput.getText().toString();
                    String password=passwordInput.getText().toString();
                    onSignIn(email,password);
                }
            }
        });
        signUpBtn.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                onSignUp();
            }
        });
    }

    public boolean validateEmail(CharSequence target){
        if(TextUtils.isEmpty(target)){
            emailInput.setError("email is required");
            return false;
        }
        else if(!Patterns.EMAIL_ADDRESS.matcher(target).matches()){
            emailInput.setError("enter valid email");
            return false;
        }
        else{
            return true;
        }
    }

    public boolean validatePassword(CharSequence target){
        if(TextUtils.isEmpty(target)){
            passwordInput.setError("password is required");
            return false;
        }
        else if(!isValidPassword(target.toString())){
            passwordInput.setError("enter valid password");
            return false;
        }
        else{
            return true;
        }
    }

    public static boolean isValidPassword(final String password) {

        Pattern pattern;
        Matcher matcher;
        final String PASSWORD_PATTERN =
"^(?=.*[0-9])(?=.*[A-Z])(?=.*[@#$%^&+=!])(?=\\S+$).{4,}$";
        pattern = Pattern.compile(PASSWORD_PATTERN);
        matcher = pattern.matcher(password);

        return matcher.matches();

    }

    @Override
    public void onStart() {
        super.onStart();
        // Check if user is signed in (non-null) and update UI accordingly.
        FirebaseUser currentUser = mAuth.getCurrentUser();
        if(currentUser!=null){
            signInSuccessfull();
        }

    }

    private void onSignUp(){
        Intent intent = new Intent(this, SignUpActivity.class);
        startActivity(intent);
    }
    private void onSignIn(String email,String password){
        mAuth.signInWithEmailAndPassword(email, password)
                .addOnCompleteListener(this, new OnCompleteListener<AuthResult>() {
                    @Override
                    public void onComplete(@NonNull Task<AuthResult> task) {
                        if (task.isSuccessful()) {
                            // Sign in success, update UI with the signed-in user's
information
                            FirebaseUser user = mAuth.getCurrentUser();
                            if(user!=null){
                                signInSuccessfull();
                                return;
                            }
                            Toast.makeText(MainActivity.this, "Authentication
failed.",
                                    Toast.LENGTH_SHORT).show();
                        } else {
                            // If sign in fails, display a message to the user.
```

```
                           Toast.makeText(MainActivity.this, "Authentication
failed.",
                                   Toast.LENGTH_SHORT).show();
                       }
                   }
               });

   }

   private  void signInSuccessfull(){
       Intent intent = new Intent(this, HomeActivity.class);
       startActivity(intent);
   }
}
```

HomeActivity.java

```
package com.example.touristguide;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.content.ContextCompat;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.Toast;

import com.google.firebase.auth.FirebaseAuth;

public class HomeActivity extends AppCompatActivity {
   private Button getLocationBtn,logoutBtn;
   private ImageView topPicksImg,shoppingImg,placesImg,foodImg;
   Double location_lat=0.0;
   Double location_lng=0.0;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
       super.onCreate(savedInstanceState);
       this.getSupportActionBar().hide();

getWindow().setStatusBarColor(ContextCompat.getColor(HomeActivity.this,R.color.colorA
ccent));
       setContentView(R.layout.activity_home);

       Intent intent = getIntent();
```



```
        location_lat = intent.getDoubleExtra("location_lat",0.0);
        location_lng=intent.getDoubleExtra("location_lng",0.0);


        logoutBtn=(Button) findViewById(R.id.logoutBtn);
        topPicksImg = (ImageView) findViewById(R.id.topPicksImg);
        shoppingImg = (ImageView) findViewById(R.id.shoppingImg);
        placesImg = (ImageView) findViewById(R.id.natureImg);
        foodImg = (ImageView) findViewById(R.id.foodImg);

        topPicksImg.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {

v.animate().rotationY(v.getRotationY()+360).setDuration(300).withEndAction(new
Runnable() {
                    @Override
                    public void run() {
                        v.animate().alpha(1);
                        if(location_lat==0.0 && location_lng==0.0){
                            Toast.makeText(HomeActivity.this,"Select location
first",Toast.LENGTH_SHORT).show();
                        }
                        else {
                            getTopPicks();
                        }

                    }
                });
            }

        });
        getLocationBtn = (Button) findViewById(R.id.getLocationBtn);

        logoutBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                FirebaseAuth.getInstance().signOut();
                Intent intent = new Intent(HomeActivity.this, MainActivity.class);
                startActivity(intent);
            }
        });
        getLocationBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                getMap();
            }


```

```
       });

       shoppingImg.setOnClickListener(new View.OnClickListener() {
           public void onClick(View v) {

v.animate().rotationY(v.getRotationY()+360).setDuration(300).withEndAction(new
Runnable() {
                   @Override
                   public void run() {
                       v.animate().alpha(1);
                       if(location_lat==0.0 && location_lng==0.0){
                           Toast.makeText(HomeActivity.this,"Select location
first",Toast.LENGTH_SHORT).show();
                       }
                       else {
                           getShopping();
                       }
                   }
               });
           }
       });

       placesImg.setOnClickListener(new View.OnClickListener() {
           public void onClick(View v) {

v.animate().rotationY(v.getRotationY()+360).setDuration(300).withEndAction(new
Runnable() {
                   @Override
                   public void run() {
                       v.animate().alpha(1);
                       if(location_lat==0.0 && location_lng==0.0){
                           Toast.makeText(HomeActivity.this,"Select location
first",Toast.LENGTH_SHORT).show();
                       }
                       else {
                           getPlace();
                       }

                   }
               });
           }
       });

       foodImg.setOnClickListener(new View.OnClickListener() {
           public void onClick(View v) {

v.animate().rotationY(v.getRotationY()+360).setDuration(300).withEndAction(new
Runnable() {
```

```
                    @Override
                    public void run() {
                        v.animate().alpha(1);
                        if(location_lat==0.0 && location_lng==0.0){
                            Toast.makeText(HomeActivity.this,"Select location
first",Toast.LENGTH_SHORT).show();
                        }
                        else {
                            getFood();
                        }

                    }
                });
            }
        });

    }



    private void getTopPicks(){
        Intent intent = new Intent(this, LocationListing.class);
        intent.putExtra("location_type", "top");
        intent.putExtra("location_latitude", location_lat);
        intent.putExtra("location_longitude",location_lng);
        startActivity(intent);
    }
    private void getMap() {
        Intent intent = new Intent(this, LocationMap.class);
        startActivity(intent);
    }
    private void getShopping(){
        Intent intent = new Intent(this, LocationListing.class);
        intent.putExtra("location_type", "shopping");
        intent.putExtra("location_latitude", location_lat);
        intent.putExtra("location_longitude",location_lng);
        startActivity(intent);
    }
    private void getPlace(){
        Intent intent = new Intent(this, LocationListing.class);
        intent.putExtra("location_latitude", location_lat);
        intent.putExtra("location_longitude",location_lng);
        intent.putExtra("location_type", "natural place");
        startActivity(intent);
    }
    private void getFood(){
        Intent intent = new Intent(this, LocationListing.class);
        intent.putExtra("location_latitude", location_lat);
        intent.putExtra("location_longitude",location_lng);
        intent.putExtra("location_type", "food");
        startActivity(intent);
    }
}
```

LocationListing.java

```
package com.example.touristguide;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.content.ContextCompat;
import androidx.recyclerview.widget.RecyclerView;

import android.content.Intent;
import android.os.Bundle;

import java.util.ArrayList;
import android.util.Log;
import android.view.MenuItem;
import android.widget.Toast;

import androidx.recyclerview.widget.LinearLayoutManager;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.QueryDocumentSnapshot;
import com.google.firebase.firestore.QuerySnapshot;

import static android.content.ContentValues.TAG;

public class LocationListing extends AppCompatActivity implements
LocationAdapter.OnLocationListener {

    private RecyclerView locationRV;
    private ArrayList<LocationModel> locationModelArrayList;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        this.getSupportActionBar().setDisplayHomeAsUpEnabled(true);

getWindow().setStatusBarColor(ContextCompat.getColor(LocationListing.this,R.color.colorAccent));
```

```
        setContentView(R.layout.activity_location_listing);

        Intent intent = getIntent();
        String location_type = intent.getStringExtra("location_type");

        this.getSupportActionBar().setTitle(location_type);
        double userLat=intent.getDoubleExtra("location_latitude",0.0);
        double userLng=intent.getDoubleExtra("location_longitude",0.0);

        locationRV = findViewById(R.id.idRVLocation);
        locationModelArrayList = new ArrayList<>();
        LocationAdapter locationAdapter = new LocationAdapter(LocationListing.this,
locationModelArrayList,LocationListing.this);



        FirebaseFirestore db = FirebaseFirestore.getInstance();
        db.collection("TouristBuddy")
                .whereEqualTo("type",location_type)
                .get()
                .addOnCompleteListener(new OnCompleteListener<QuerySnapshot>() {
                    @Override
                    public void onComplete(@NonNull Task<QuerySnapshot> task) {
                        if (task.isSuccessful()) {
                            for (QueryDocumentSnapshot document : task.getResult()) {

                                String[] latlongArr =
document.getString("latlong").split(",");
                                double lat=Double.parseDouble(latlongArr[0]);
                                double lng=Double.parseDouble(latlongArr[1]);
                                if(caldistance(userLat,lat,userLng,lng,0.0,0.0)<3000){
                                    String location_id=document.getId();
                                    String location_name=document.getString("name");
                                    float
location_rating=Float.parseFloat(document.get("rating").toString());
                                    String
location_desc=document.getString("description");
                                    LocationModel lm=new
LocationModel(location_id,location_name,location_rating,location_desc);
                                    locationModelArrayList.add(lm);
                                    locationAdapter.notifyDataSetChanged();
                                }

                            }
                        } else {
                            Log.w(TAG, "Error getting documents.",
task.getException());
                        }
```

```
                   }
               });


       LinearLayoutManager linearLayoutManager = new
LinearLayoutManager(LocationListing.this, LinearLayoutManager.VERTICAL, false);
       locationRV.setLayoutManager(linearLayoutManager);
       locationRV.setAdapter(locationAdapter);
   }


   @Override
   public void onLocationClick(int pos) {
       Toast.makeText(this,"You are at
"+locationModelArrayList.get(pos).getLocation_name(),Toast.LENGTH_SHORT).show();
       Intent intent = new Intent(this, Details.class);
       intent.putExtra("place_id",locationModelArrayList.get(pos).getLocation_id());
       startActivity(intent);
   }


   @Override
   public boolean onOptionsItemSelected(MenuItem item)
   {
       // Handle item selection
       finish();
       return true;
   }

   public static double caldistance(double lat1, double lat2, double lon1,
                                    double lon2, double el1, double el2) {

       final int R = 6371; // Radius of the earth

       double latDistance = Math.toRadians(lat2 - lat1);
       double lonDistance = Math.toRadians(lon2 - lon1);
       double a = Math.sin(latDistance / 2) * Math.sin(latDistance / 2)
               + Math.cos(Math.toRadians(lat1)) * Math.cos(Math.toRadians(lat2))
               * Math.sin(lonDistance / 2) * Math.sin(lonDistance / 2);
       double c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1 - a));
       double distance = R * c * 1000; // convert to meters

       double height = el1 - el2;

       distance = Math.pow(distance, 2) + Math.pow(height, 2);

       return Math.sqrt(distance);
   }
```

```
}
```

LocationAdapter.java

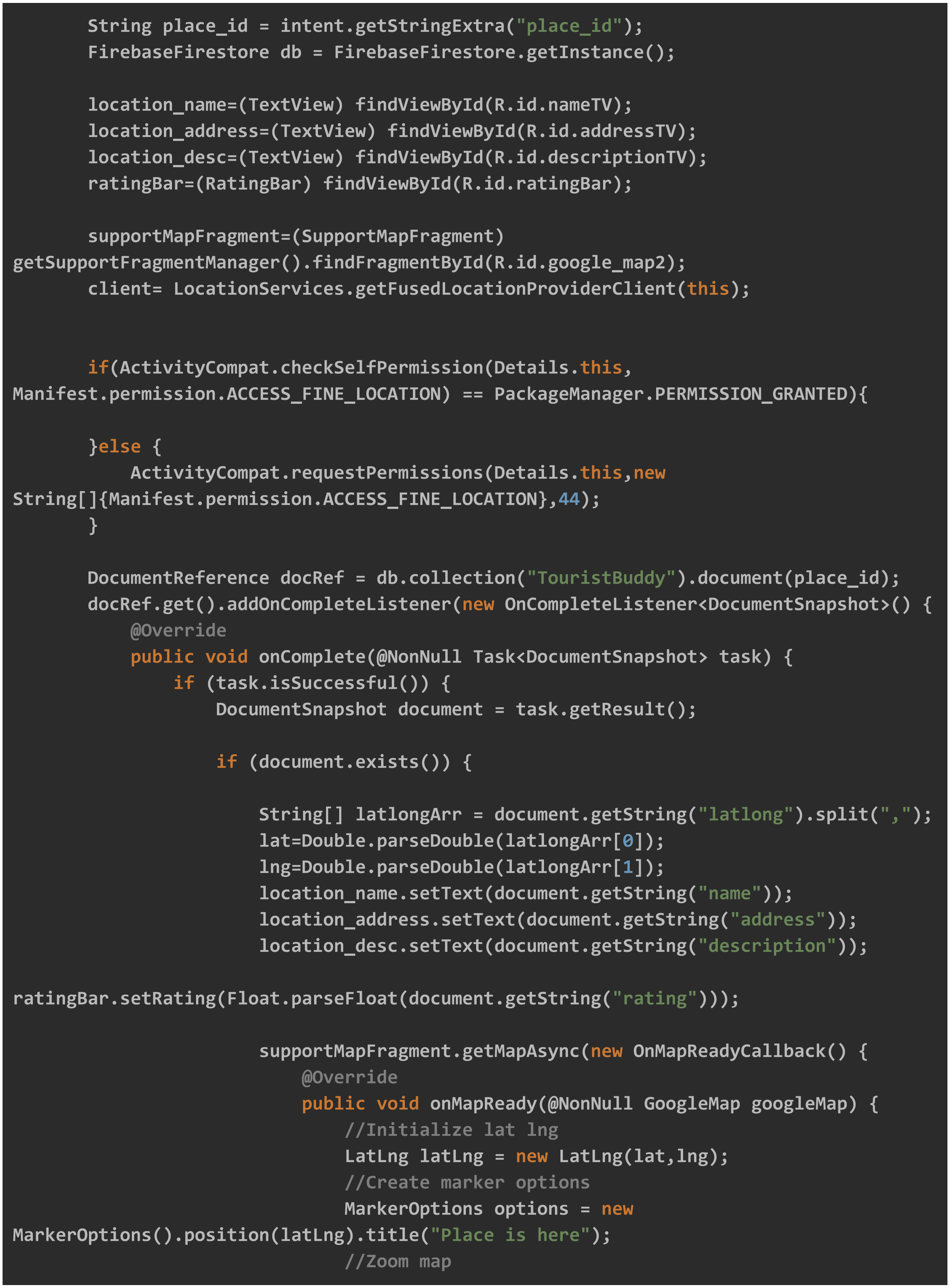
```java
package com.example.touristguide;

import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import java.util.ArrayList;

public class LocationAdapter extends RecyclerView.Adapter<LocationAdapter.Viewholder>
{

   private Context context;
   private ArrayList<LocationModel> locationModelArrayList;
   private OnLocationListener onLocationListener;

   // Constructor
   public LocationAdapter(Context context, ArrayList<LocationModel>
locationModelArrayList,OnLocationListener onLocationListener) {
       this.context = context;
       this.locationModelArrayList = locationModelArrayList;
       this.onLocationListener=onLocationListener;
   }

   @NonNull
   @Override
   public LocationAdapter.Viewholder onCreateViewHolder(@NonNull ViewGroup parent, int
viewType) {
       // to inflate the layout for each item of recycler view.
       View view =
LayoutInflater.from(parent.getContext()).inflate(R.layout.card_layout, parent, false);
       return new Viewholder(view,onLocationListener);
   }

   @Override
   public void onBindViewHolder(@NonNull LocationAdapter.Viewholder holder, int
position) {
       // to set data to textview and imageview of each card layout
       LocationModel model = locationModelArrayList.get(position);
```

```
        holder.locationNameTV.setText(model.getLocation_name());
        holder.locationRatingTV.setText("" + model.getLocation_rating());
        holder.locationDescTV.setText(model.getLocation_desc());
    }

    @Override
    public int getItemCount() {
        // this method is used for showing number
        // of card items in recycler view.
        return locationModelArrayList.size();
    }

    // View holder class for initializing of
    // your views such as TextView and Imageview.
    public class Viewholder extends RecyclerView.ViewHolder implements
View.OnClickListener {
        private TextView locationNameTV, locationRatingTV,locationDescTV;
        private OnLocationListener onLocationListener;
        public Viewholder(@NonNull View itemView,OnLocationListener onLocationListener)
{
            super(itemView);
            locationDescTV = itemView.findViewById(R.id.idTVLocationDesc);
            locationNameTV = itemView.findViewById(R.id.idTVLocationName);
            locationRatingTV = itemView.findViewById(R.id.idTVLocationRating);
            this.onLocationListener=onLocationListener;
            itemView.setOnClickListener(this);
        }

        @Override
        public void onClick(View v) {
            onLocationListener.onLocationClick(getAdapterPosition());
        }
    }

    public interface OnLocationListener{
        void onLocationClick(int pos);
    }

}
```

Details.java

```
package com.example.touristguide;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
```

```java
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;

import android.Manifest;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.location.Location;
import android.os.Bundle;
import android.util.Log;
import android.webkit.WebView;
import android.widget.RatingBar;
import android.widget.TextView;

import com.google.android.gms.location.FusedLocationProviderClient;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.FirebaseFirestore;

import static android.content.ContentValues.TAG;

public class Details extends AppCompatActivity {

    SupportMapFragment supportMapFragment;
    FusedLocationProviderClient client;
    double lat=0.0;
    double lng=0.0;
    private TextView location_name,location_desc,location_address;
    private RatingBar ratingBar;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        this.getSupportActionBar().setTitle("Details");

getWindow().setStatusBarColor(ContextCompat.getColor(Details.this,R.color.colorAccent)
);
        setContentView(R.layout.activity_details);

        Intent intent = getIntent();
```



```
        String place_id = intent.getStringExtra("place_id");
        FirebaseFirestore db = FirebaseFirestore.getInstance();

        location_name=(TextView) findViewById(R.id.nameTV);
        location_address=(TextView) findViewById(R.id.addressTV);
        location_desc=(TextView) findViewById(R.id.descriptionTV);
        ratingBar=(RatingBar) findViewById(R.id.ratingBar);

        supportMapFragment=(SupportMapFragment)
getSupportFragmentManager().findFragmentById(R.id.google_map2);
        client= LocationServices.getFusedLocationProviderClient(this);


        if(ActivityCompat.checkSelfPermission(Details.this,
Manifest.permission.ACCESS_FINE_LOCATION) == PackageManager.PERMISSION_GRANTED){

        }else {
            ActivityCompat.requestPermissions(Details.this,new
String[]{Manifest.permission.ACCESS_FINE_LOCATION},44);
        }

        DocumentReference docRef = db.collection("TouristBuddy").document(place_id);
        docRef.get().addOnCompleteListener(new OnCompleteListener<DocumentSnapshot>() {
            @Override
            public void onComplete(@NonNull Task<DocumentSnapshot> task) {
                if (task.isSuccessful()) {
                    DocumentSnapshot document = task.getResult();

                    if (document.exists()) {

                        String[] latlongArr = document.getString("latlong").split(",");
                        lat=Double.parseDouble(latlongArr[0]);
                        lng=Double.parseDouble(latlongArr[1]);
                        location_name.setText(document.getString("name"));
                        location_address.setText(document.getString("address"));
                        location_desc.setText(document.getString("description"));

ratingBar.setRating(Float.parseFloat(document.getString("rating")));

                        supportMapFragment.getMapAsync(new OnMapReadyCallback() {
                            @Override
                            public void onMapReady(@NonNull GoogleMap googleMap) {
                                //Initialize lat lng
                                LatLng latLng = new LatLng(lat,lng);
                                //Create marker options
                                MarkerOptions options = new
MarkerOptions().position(latLng).title("Place is here");
                                //Zoom map
googleMap.animateCamera(CameraUpdateFactory.newLatLngZoom(latLng,15));
                               //Add marker on map
                               googleMap.addMarker(options);
                           }
                       });
                   }
               } else {
                   Log.d(TAG, "get failed with ", task.getException());
               }
           }
       });

   }

   @Override
   public void onRequestPermissionsResult(int requestCode, @NonNull String[]
permissions, @NonNull int[] grantResults) {
       super.onRequestPermissionsResult(requestCode, permissions, grantResults);
       if (requestCode == 44) {
           if (grantResults.length > 0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
               supportMapFragment.getMapAsync(new OnMapReadyCallback() {
                   @Override
                   public void onMapReady(@NonNull GoogleMap googleMap) {
                       //Initialize lat lng
                       LatLng latLng = new LatLng(lat,lng);
                       //Create marker options
                       MarkerOptions options = new
MarkerOptions().position(latLng).title("Place is here");
                       //Zoom map

googleMap.animateCamera(CameraUpdateFactory.newLatLngZoom(latLng,15));
                       //Add marker on map
                       googleMap.addMarker(options);
                   }
               });
           }
       }
   }
}
```

Acknowledgement

We would like to express special thanks to our guide Mrs. Poonam Bari who gave us the golden opportunity to do this wonderful project on the topic of Travel Buddy application, which helped us in doing a lot of research and we came to know about so many new things. We would also like to thank our HOD Mrs. Dhanashree Hadsul for providing us the opportunity to implement our project. We are really thankful to them. Finally we would also like to thank our parents and friends who helped us a lot in finalizing this project within the limited time frame.

Prithviraj Patil (501850)
Arnav Pandey (501847)
Pranav Patil (501849)