

wanderLUST - A Travel App

A Project Report

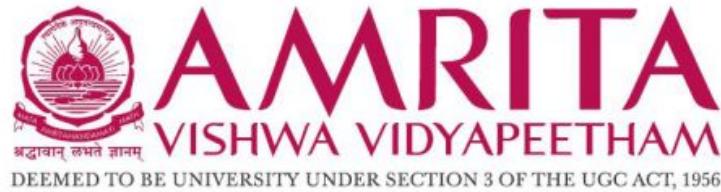
submitted as a part of the internal evaluation process for the course
15CSE375 Android Application Development

By

GROUP V

RAHUL SANI (AM.EN.U4CSE17057)

YAMUNA K. (AM.EN.U4CSE17358)



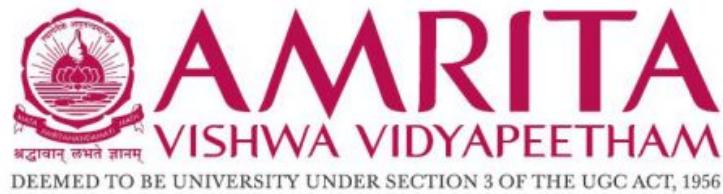
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

AMRITA VISHWA VIDYAPEETHAM

AMRITAPURI CAMPUS (INDIA)

MAY 2020

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
AMRITA VISHWA VIDYAPEETHAM
AMRITAPURI CAMPUS



BONAFIDE CERTIFICATE

This is to certify that the project report entitled **WANDERLUST - A TRAVEL APP** submitted by **Rahul Sani (AM.EN.U4CSE17057)**, **Yamuna K (AM.EN.U4CSE17358)** as a part of the internal evaluation process for the course **15CSE375 ANDROID APPLICATION DEVELOPMENT** is a bonafide record of the work carried out by them under the guidance supervision of me at the Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amritapuri Campus.

Mr. Rahul Varma U.
Faculty Associate,
Department of Computer Science And Engineering,
Amrita School of Engineering,
Amritapuri

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

AMRITA VISHWA VIDYAPEETHAM

AMRITAPURI CAMPUS

DECLARATION

We, **Rahul Sani** (**AM.EN.U4CSE17057**) and **Yamuna K** (**AM.EN.U4CSE17358**) hereby declare that this project report entitled **WANDERLUST - A TRAVEL APP** is the record of the original work done by us under the guidance of **Mr. Rahul Varma U.**, Faculty Associate, Department of Computer Science And Engineering, Amrita School of Engineering, Amritapuri.

Place: Amritapuri

Date: 15-05-2020

RAHUL SANI **(AM.EN.U4CSE17057)**

YAMUNA K. **(AM.EN.U4CSE17358)**

ACKNOWLEDGEMENT

We have given our full efforts in the completion of this project. However, it would not have been possible without the kind support and help of many individuals. We would like to extend our sincere thanks to all of them. We are highly indebted to Mr. Rahul Varma U. for his guidance and constant supervision as well as for providing necessary information regarding the project & also for his support in completing the project. All two members of the group have put their efforts equally in the successful completion of this project. We are also thankful to the Institution and CSE Department for providing us with the opportunity to work on a project to enhance our skills in the field of Android Application Development.

RAHUL SANI (AM.EN.U4CSE17057)

YAMUNA K. (AM.EN.U4CSE17358)

CONTENTS

1.	EXECUTIVE SUMMARY.....	8-9
1.1.	ABSTRACT.....	
1.2.	PROJECT OBJECTIVE.....	
1.3.	BACKGROUND.....	
2.	SYSTEM REQUIREMENTS.....	10-14
2.1.	REGISTRATION & PROFILE MANAGEMENT.....	
2.2.	HOME PAGE.....	
2.3.	SEARCH & FILTERS, LISTING INFORMATION.....	
2.3.1.	Images.....	
2.3.2.	Weather at the place.....	
2.3.3.	Facts about the location.....	
2.3.4.	Hangouts, Destinations, Shopping, Restaurants & City map.....	
2.4.	BOOKING.....	
2.5.	PLANNING OF TRIP.....	
2.6.	NEED FOR UTILITIES.....	
3.	SYSTEM ARCHITECTURE.....	15-16
3.1.	BLOCK DIAGRAM.....	
3.2.	PROGRAM MODULE DESIGN.....	
4.	USER INTERFACE DESIGN.....	17-29
4.1.	LOGIN PAGE.....	
4.2.	REGISTER PAGE.....	
4.3.	HOME PAGE.....	
4.4.	DESTINATION.....	
4.4.1.	BANGALORE.....	
4.4.1.1.	Check Weather.....	
4.4.1.2.	Fun Facts about the City.....	
4.4.1.3.	Hangout.....	
4.4.1.4.	Destination.....	
4.4.1.5.	Shopping.....	
4.4.1.6.	Restaurants.....	
4.4.1.7.	City Maps.....	
4.5.	HOTEL BOOKING.....	
4.6.	MY TRIP.....	

4.6.1.	ADD TRIP.....	
4.6.2.	VIEW DETAILS.....	
4.6.3.	UPDATE TRIP.....	
4.7.	UTILITY.....	
4.7.1.	CHECKLIST.....	
4.7.2.	COMPASS.....	
4.7.3.	WORLD CLOCK.....	
5.	ANDROID COMPONENTS/SERVICES/API USED.....	30-35
5.1.	LOGIN PAGE.....	
5.2.	REGISTER PAGE.....	
5.3.	HOME PAGE.....	
5.4.	DESTINATION.....	
5.4.1.	BANGALORE.....	
5.4.1.1.	Check Weather.....	
5.4.1.2.	Fun Facts about the City.....	
5.4.1.3.	Hangout.....	
5.4.1.4.	Destination.....	
5.4.1.5.	Shopping.....	
5.4.1.6.	Restaurants.....	
5.4.1.7.	City Maps.....	
5.5.	HOTEL BOOKING.....	
5.6.	MY TRIP.....	
5.6.1.	ADD TRIP.....	
5.6.2.	VIEW DETAILS.....	
5.6.3.	UPDATE TRIP & CANCEL TRIP.....	
5.7.	UTILITY.....	
5.7.1.	CHECKLIST.....	
5.7.2.	COMPASS.....	
5.7.3.	WORLD CLOCK.....	
6.	DATA STORAGE.....	36-37
6.1.	DATABASE DESIGN.....	
6.1.1.	PACKAGE.....	
6.1.2.	HELPER.....	
7.	PROGRAM DEVELOPMENT.....	38-152
7.1.	CODE OF wanderLUST APPLICATION.....	
7.1.1.	LOGIN PAGE.....	

7.1.2.	REGISTER PAGE	
7.1.3.	HOME PAGE	
7.1.4.	DESTINATION	
7.1.5.	HOTEL BOOKING	
7.1.6.	MY TRIP	
7.1.7.	UTILITY	
7.1.7.1.	CHECKLIST	
7.1.7.2.	COMPASS	
7.1.7.3.	WEATHER FORECAST	
7.1.7.4.	WORLD CLOCK	
8.	RESULT	153-160
9.	CONCLUSION	161

CHAPTER 1

EXECUTIVE SUMMARY

1.1. ABSTRACT

In recent years, the development of worldwide travel has been outstanding. At the same time, due to the fast development of the travel business, it is important to offer tourists an enhanced travel platform. Nationally, a customer has to approach various agencies to find details of places and to book travel tickets. This often requires a lot of time and effort. A customer may not get the desired information from these offices and often the customer may be misguided. Sometimes, they might have trouble when touring and less aware of what to find and expect and also locating places at different sites. Moreover, it is tedious for a customer to plan a particular journey and have it executed properly. This project is an effort to maintain a centralized repository to make necessary travel arrangements and to retrieve information easily. As part of the project, a mobile travel guide application is developed entitled “wanderLUST”. Nowadays people prefer to use android applications due to easy, conveyance usage. In existing applications bookings can be made but there is no feature for planning the trip. *wanderLUST* is an android app for travelers. It provides features like choosing the correct destination, making bookings, and organizing the trip. It provides solutions for every problem a traveler might face during their journey. It also provides nearby tourist places and suggests nearby hotels, restaurants and helps them to find places through GPS. It provides a comprehensive list of information about a destination: current weather, best places to hang out, the city's current trends. The application provides places with pictures and then people can add the trip and can give a name to the trip and select the type of the trip from the provided types. Then we can set a start date and save. Once they save they can edit as many times they can. The provided data by the users are stored in the SQLite database provided by the android studio.

1.2. PROJECT OBJECTIVE

- To develop a system that automates the processes and activities of travel.
- To design a system using which one can perform all operations related to traveling.
- To design a product that contains a possible integration of a number of features.
- To facilitate the tourism industry by using modern technologies to boost tourism.
- To change the traditional techniques that might create problems while decision making due to lack of information.

1.3. BACKGROUND

Today, mobile apps play an important role in every business domain as well as in personal life. The travel & tourism industry is no exception since a huge part of its success relies on these mobile apps. The days of paper maps, guidebooks, and other stuff are gone and replaced by interactive & amazing mobile apps for the travel & tourism industry that come with many functionalities.

According to **Statista**, the travel & tourism industry is projected to generate 17.3 billion U.S. dollars in revenue by 2020. In fact, it is one of the world's largest industries with a global economic contribution of over 7.6 trillion U.S. dollars in 2016. Almost all travelers carry smartphones with them and prefer apps for various tasks like flight booking, hotel room booking, etc. So, these mobile apps bring happiness with multi-benefits for both its users as well as for the company that owns these apps. **Travel application development** has become a one-stop solution for its users as you don't have to have different portals for car rentals, ticket booking, insurance, travel-related shopping, etc. You get everything related to traveling at a single place.

CHAPTER 2

SYSTEM REQUIREMENTS

App Features

- Login/Register
- Destination
- Hotel Booking
- My Trip
- Utility
 - Checklist
 - Compass
 - Weather Forecast
 - World Clock

2.1. REGISTRATION & PROFILE MANAGEMENT

- The login screen allows registered users to login to the app to access all of the features that their account gives them access to.
- The app shall require users to enter **Email & Password** as the input in the **TextInputLayout** on the screen.
- The password shall not be displayed on-screen when a user types it in. We can make it visible on clicking on the **password visibility toggle**.
- After entering the required details and on clicking the **Login Button** the user credentials are validated by fetching the data from **SQLite Database for Users**. If the data entered matches and is correct the user gets logged in. If they are incorrect they get an error message stating **Please enter valid email!** Or **Please enter valid password!** (if the password field is empty) Or **Password is too short!** (if the entered password doesn't match with the original password stored in the database).
- If the user does not have an account then they click the **create new account TextView** written below the Login button. On clicking, they are taken to the registration screen.
- The app shall require users to enter **Username, Email & Password** as the input in the **TextInputLayout** on the screen.

- The user types in all of the information and clicks **Register Button**. The data is then validated to make sure that there is not an existing user with those credentials. If there is an existing user then a message is shown stating **User already exists with same email**. The user is asked to enter new credentials.
- If there is no conflict with the credentials then the user is registered and the data gets stored in the SQLite database for future purposes and is redirected to the Login page to login.

2.2. HOME PAGE

- On clicking the login page the Home Page is loaded (**Circular Progress Bar** is used).
- This page contains the main features of the app listing **Destination, Hotel Booking, My Trip & Utility** as **ImageButton**.
- There is an **OptionMenu** on the toolbar which contains an item **Logout**.
- On clicking Logout, An **AlertDialog box** pops up stating **You sure, that you want to logout?** With two options **No & Yes**.
- On clicking **No** the user returns to the same page (Home page) while on clicking **Yes** the user is redirected to the Login page asking to enter the credentials.

2.3. SEARCH & FILTERS, LISTING INFORMATION

Any travel app should help users to find places to travel. The more complex filters users have the more precise results they get.

Here are the primary filters users have access to:

- Images
- Weather at the place
- Facts about the Location
- Hangouts
- Destinations
- Shopping
- Restaurants
- City Map

2.3.1. Images

- The users could experience the place virtually through the pictures of the various places around the country and decide upon their choice of travel destination.
- Various images of and related to the city are displayed using *Image Slider*.

2.3.2. Weather at the place

- The app provides accurate forecasts to enable users to make informed decisions before planning the trip.
- The user can get hourly, daily and weekly predictions as per their location (*Weather API*).
- The users get additional functionality like weather comparison by entering the location they would like to view in addition to the selected location in the *EditText* that is added as an additional feature. The user could type any location and click the *search button* to get the real-time weather updates.
- In addition to weather, *icons* of the corresponding weather, *humidity*, *pressure* are also displayed as additional information so that the user gets better insights.

2.3.3. Facts about the location

- The user is provided with real-time information about the location fetched from Wikipedia using *Wikipedia API*.

2.3.4. Hangouts, Destinations, Shopping, Restaurants & City map

- *Google Maps & Places API* is used as the base to depict locations of interest.
- Various overlays provided by Google Maps API are leveraged to merge data from various sources and present it on the screen.
- The user interface consists of multiple tabs(*CardViews*) which *intents* to multiple screens depending on the functionality we want to access (like destinations, restaurants, etc.), for example - In the Map mode, the screen will just contain a map with the various overlays depicting the location of the searched utility.
- When the users open the application the user can see a map on their screen with a pointer on the current location. This gives the user an idea of his current location.

- Initially, the user must click the desired utility for which the user wants to search for the locations. Touching the Map overlay of a location shows detailed information about that utility.

2.4. BOOKING

- A *WebView* is used to load the website of the hotel booking website like *Booking.com*.
- The user could use the website just like they use them by retrieving the result from google search.
- The user could check for availability and make hotel bookings through this website after deciding upon the destination they would like to visit from wanderLUST.

2.5. PLANNING OF TRIP

- The user adds a new trip by clicking on the *FloatingActionButton* on the My trip screen which lists down the trips planned as *ListView* by retrieving the data from *SQLite Database for Trips*.
- On clicking the add button the users are taken to a screen where they are required to enter the Date when the trip is planned by clicking on the *calendar button* which pops up *DatePickerDialog*. The user is required to pick a date and click **Ok**. The selected date appears on the *Select Trip Date EditText field* and enters the location by selecting a location from the *CustomAutoCompleteTextView* that drops down on entering “0” characters. On clicking on of the items from the drop-down list the corresponding location name appears on the *Enter a place field*.
- The user types in all information and on clicking the *Add Trip button*, the user is diverted to the My trip screen where the newly added trip appears in the list with the *Location, Date & Image* of the corresponding place.
- The provided data by the users is stored in the SQLite database.
- The user clicking the item could view the trip details where they can edit the details as many times they want by clicking the *Update Trip button*.
- The user is required to enter the new data and on clicking the Update Trip button the data gets updated in the database and correspondingly on the screen and list.
- The users cancel the trip by clicking the *Cancel Trip button*.
- On clicking Cancel Trip, an *AlertDialog box* pops up stating *You sure, that you want to cancel?* With two options *No & Yes*.

- On clicking **No** the user returns to the My Trip page with no changes while on clicking **Yes** the user is redirected to the My Trip page with the list item removed from the list and from the database.
- The users can share the information about the trip with their friends by clicking the **share button**. This feature uses **Implicit Intent** to send the message to another person through external apps.

2.6. NEED FOR UTILITIES

- The user can use the **Checklist** to organize the trip. Before the user starts the trip, they can make sure they've checked off all the required things by clicking the **checkbox** corresponding to the items and if anything missed could be typed in **EditText** and added to the **List** clicking **FloatingActionButton**.
- The **Compass** can be used for navigation, location, and direction. The user can use it to find their way, whether it is on a hiking trail or on a trip to a new location. The Compass activity extends **SensorEventListener**.
- The user can use the **World Clock** to compare specific times, to customize local times from around the world to meet the needs of business meetings if any scheduled, organize favorite cities so that they can easily find out the correct local time as well as get the most relevant time zone information about the city.

CHAPTER 3

SYSTEM ARCHITECTURE

Architecture is the fundamental and unifying system structure defined in terms of system elements, interfaces, process, constraints, and behaviors and it is the structure of components, their relationships, principles, and the guidelines governing their evolution over time. A System Architecture is the link between needs analysis, project scoping and functional analysis, and the first descriptions of the system structure.

3.1. BLOCK DIAGRAM

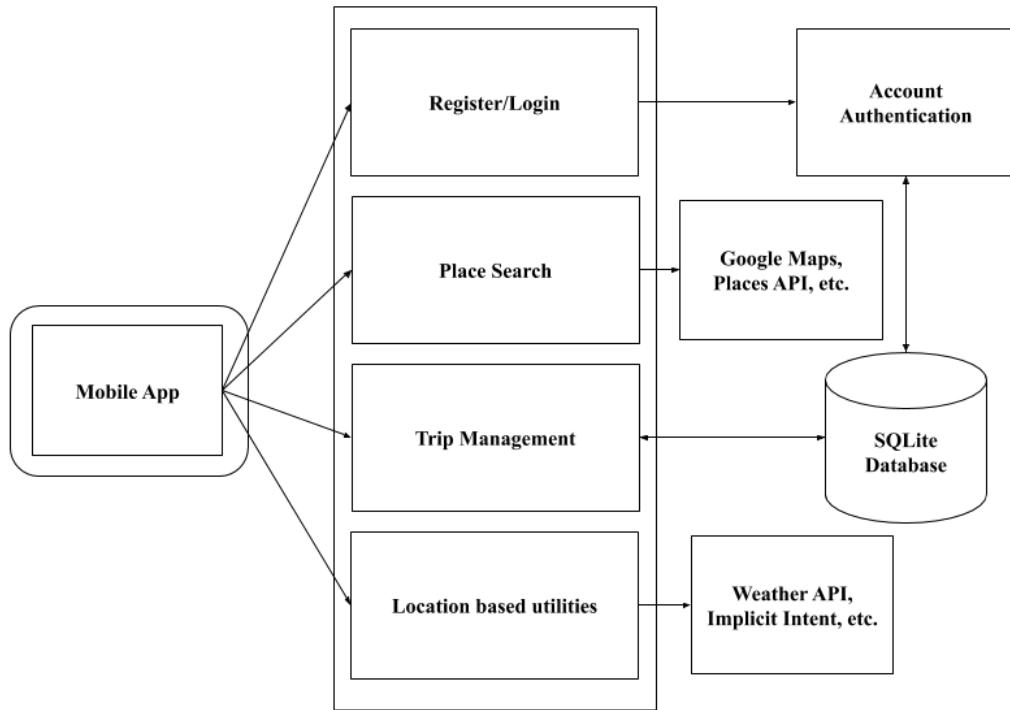


Fig. Block Diagram

3.2. PROGRAM MODULE DESIGN

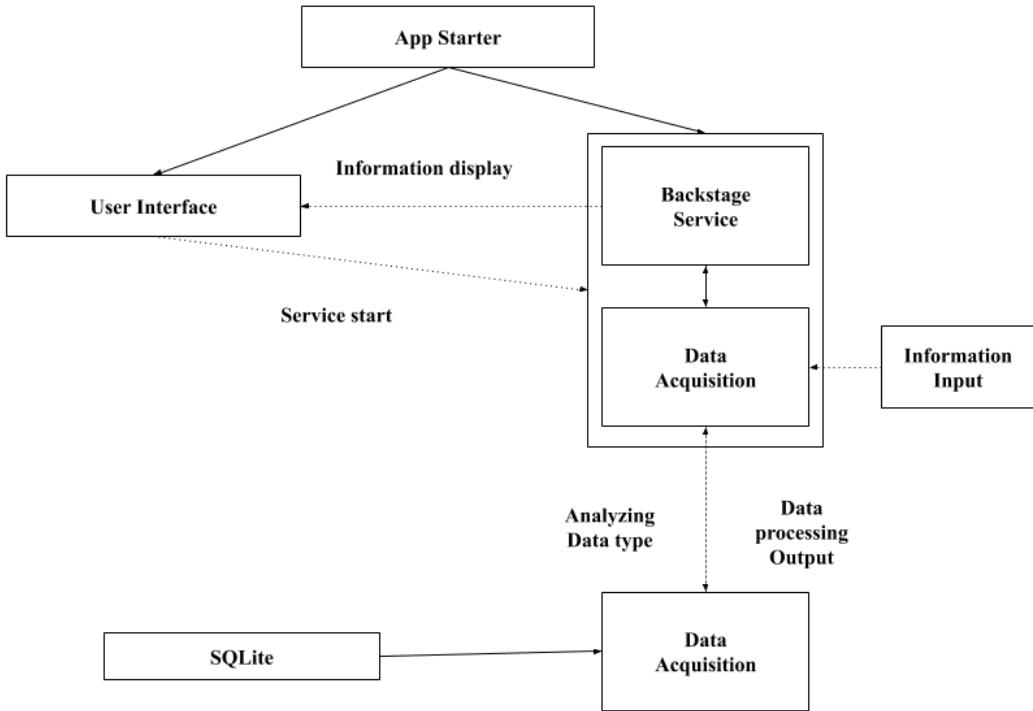


Fig. Module Relationship Diagram

CHAPTER 4

USER INTERFACE DESIGN

4.1. LOGIN PAGE

The login page interface is the first page that the user sees in the application. It provides two fields- one for entering the Email and one for entering a password. If either of the text fields is left blank it is an error that must be reported to the user. If both fields are filled in but there is no record of the Email or the Password is incorrect that is also reported to the user.



4.2. REGISTER PAGE

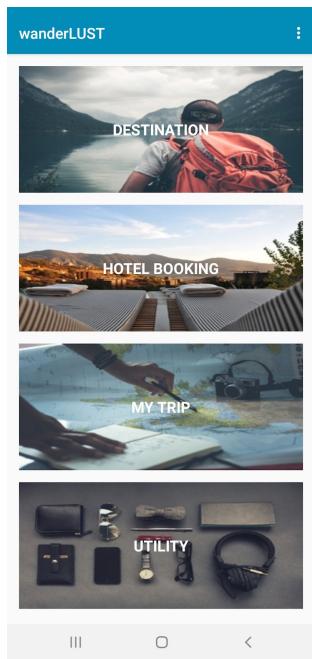
The registration page should be displayed when a user clicks on the register command button on the login page. It should provide three text fields - one for entering a Username, one for Email, and one for entering Password. If any of the text fields is left blank it is an error that must be reported to the user. If all fields

are filled in but there is already a record for the user name then that must be reported as an error.



4.3. HOME PAGE

The home page is the main page a user navigates to an app from the login page. It serves as the landing page to attract visitors. The home page is used to facilitate navigation to other pages on the app by clicking on various images and tabs present.



4.4. DESTINATION

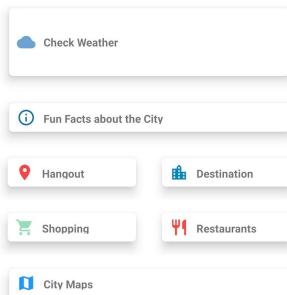
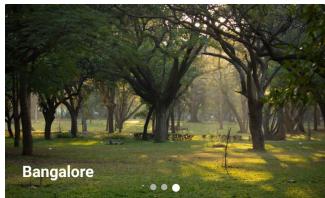
It helps decide the destination by providing real-time information about tourist spots, interesting trivia, local trends, weather, restaurants, monuments, restaurants, and much more for travelers to explore unfamiliar cities.



4.4.1. BANGALORE

Bangalore is a famous Silicon Valley of India, Karnataka's capital city is a well known IT hub and one of the world's major IT corporations operate out of the city. Many people like the city because of its greenery and interesting buildings.

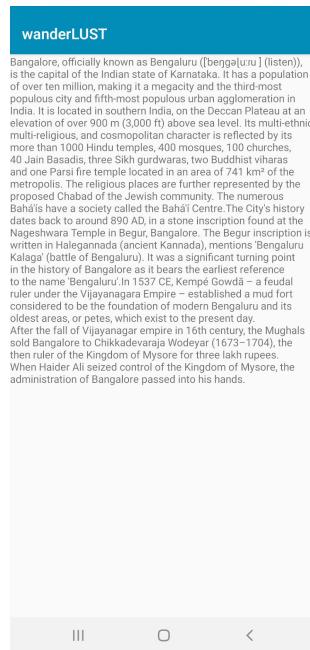
The top attractions to visit in Bengaluru are ISKCON Temple, Cubbon Park, Wonderla Amusement Park, Lalbagh Botanical Garden, etc. The top shopping spots are Phoenix Marketcity, UB City, Mantri Square, Orion Mall, etc. Eat like there's no tomorrow as Bangalore has some of the best restaurants in India like Truffles Koramangala, Smally's Resto Cafe, Treat Restaurant, Indiranagar.



4.4.1.1. Check Weather



4.4.1.2. Fun Facts about the City



4.4.1.3. Hangouts



4.4.1.4. Destination



4.4.1.5. Shopping



4.4.1.6. Restaurants



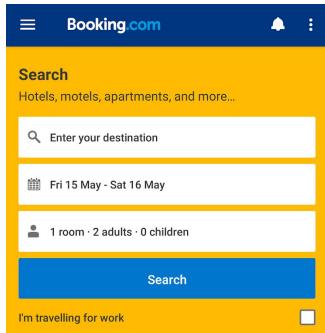
4.4.1.7 City Maps



4.5. HOTEL BOOKING

Booking.com's mission is to make it easier for everyone to experience the world. Booking.com seamlessly connects millions of travelers to memorable experiences, a variety of transportation options, and incredible places to stay – from homes to hotels, and much more.

Booking.com is available in 43 languages and offers more than 28 million reported accommodation listings, including over 6.2 million homes, apartments, and other unique places to stay. Wherever you want to go and whatever you want to do, Booking.com makes it easy and supports you with 24/7 customer support.

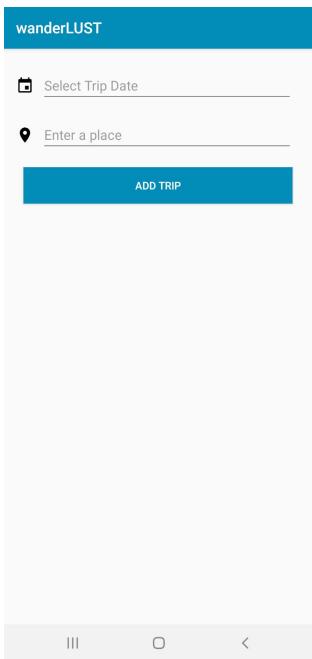


4.6. MY TRIP

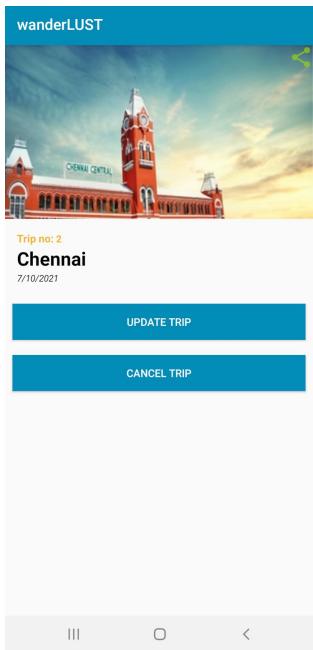
It allows users to store their travel history and future plans. They have the option to keep their trips private or share with others.



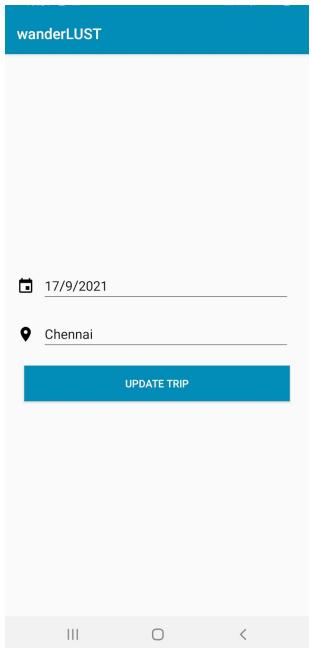
4.6.1. ADD TRIP



4.6.2. VIEW DETAILS



4.6.3. UPDATE TRIP

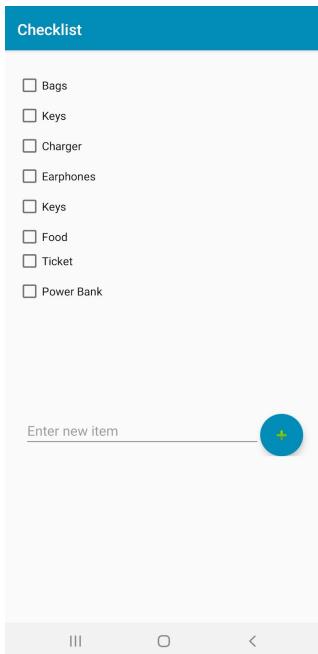


4.7. UTILITY

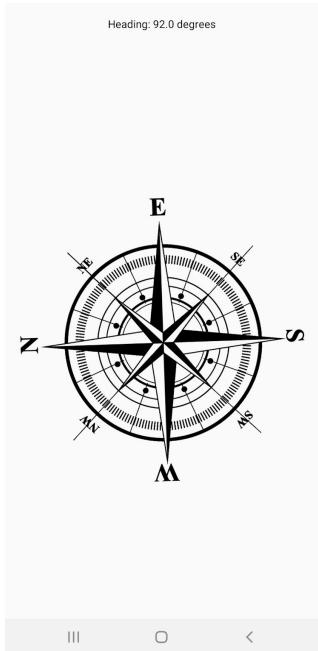
It gives a bunch of helpful utilities in one app including World Clock, checklists, compass, weather forecast.



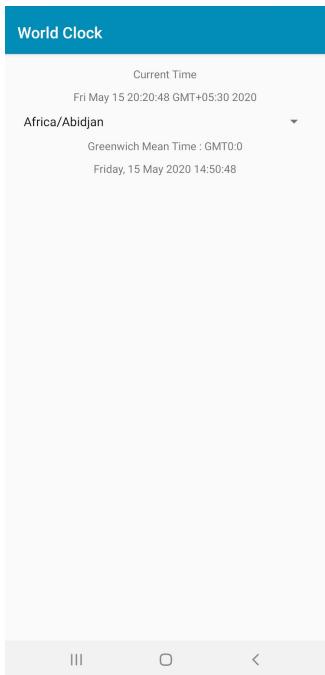
4.7.1. CHECKLIST



4.7.2 COMPASS



4.7.3 WORLD CLOCK



CHAPTER 5

ANDROID COMPONENTS/SERVICES/API USED

5.1. LOGIN PAGE

- ScrollView - Activity layout
- LinearLayout - Child elements of the palette
- ImageView - the logo of the app
- TextInputLayout
 - TextInputEditText - Email & Password field
- Button - Login
- TextView
 - Html - to Intent to Register page

5.2. REGISTER PAGE

- ScrollView - Activity layout
- LinearLayout - Child elements of the palette
- ImageView - the logo of the app
- TextInputLayout
 - TextInputEditText - Username, Email & Password field
- Button - Register
- TextView
 - Html - to Intent to Login page

5.3. HOME PAGE

- ScrollView - Activity layout
- LinearLayout - Child elements of the palette
- FrameLayout - to block out an area on the screen to display a single item
 - ImageButton - to Intent to various functionalities provided by the app (Destination, Hotel Booking, My Trip, Utility)
 - TextView - to display the names of functionality the ImageButton would navigate to (Destination, Hotel Booking, My Trip, Utility)

- Options Menu - to logout from the app and return to the login page
- AlertDialog box - to confirm the logout

5.4. DESTINATION

- ScrollView - Activity layout
- LinearLayout - Child elements of the palette
- FrameLayout - to block out an area on the screen to display a single item
 - ImageButton - to Intent to various functionalities provided by the page (Bangalore, Chennai, Delhi, etc.)
 - TextView - to display the names of functionality the ImageButton would navigate to (Bangalore, Chennai, Delhi, etc.)

5.4.1 BANGALORE

- ScrollView - Activity layout
- LinearLayout - Child elements of the palette
- FrameLayout - to block out an area on the screen to display a single item
 - ImageSlider - to slide from one screen to another
 - to implement an image slider, you need to inherit ViewPager class which extends PagerAdapter
 - TextView - to display the names of the destination
- CardView - a card is a sheet of material that serves as an entry point to more detailed information. Cards may contain a photo, text, and a link about a single subject. They may display content containing elements of varying size, such as photos with captions of variable length
 - TextView - Check Weather, Fun facts about the City, Hangout, Destination, Shopping, Restaurants, City Maps

5.4.1.1. Check Weather

- RelativeLayout - Activity layout
- LinearLayout - Child elements of the palette
- EditText - to enter the place for search
- ImageButton - to initiate the search process

- TextView - City name, Date & Time, Weather icon, current temperature status, humidity, pressure
- ImageView - to display icons for humidity and pressure
- **API:** Weather API (OpenWeatherMap)
 - JSON Parsing

5.4.1.2. Fun facts about the CIty

- RelativeLayout - Activity layout
- Scroll View - Child elements of the palette
- TextView - To show the data fetched

5.4.1.3. Hangout

- MapView- To show the Map
- **API:**
 - Weather API (OpenWeatherMap)
 - Google Places API
 - Google Maps API
 - JSON Parsing

5.4.1.4. Destination

- MapView- To show the Map
- **API:**
 - Weather API (OpenWeatherMap)
 - Google Places API
 - Google Maps API
 - JSON Parsing

5.4.1.5. Shopping

- MapView- To show the Map
- **API:**
 - Weather API (OpenWeatherMap)
 - Google Places API

- Google Maps API
- JSON Parsing

5.4.1.6. Restaurants

- MapView- To show the Map
- **API:**
 - Weather API (OpenWeatherMap)
 - Google Places API
 - Google Maps API
 - JSON Parsing

5.4.1.7. City Maps

- MapView- To show the Map
- **API:**
 - Google Places API
 - Google Maps API

5.5. HOTEL BOOKING

- ConstraintLayout - Activity layout
- WebView - to load the URL of the website

5.6. MY TRIP

- RelativeLayout - Activity layout
- ListView - to display the SQLite database data
 - TimeLineView - simple implementation used to display view like tracking of events
 - CardView - a card is a sheet of material that serves as an entry point to more detailed information. Cards may contain a photo, text, and a link about a single subject. They may display content containing elements of varying size, such as photos with captions of variable length

- ImageView - to display the image of the location user planned the trip & display arrow showing the navigation possible to view the contents and update
- TextView - to display the Place and Date of the trip
- FloatingActionButton - to add new trips

5.6.1. ADD TRIP

- LinearLayout - Activity layout
- ImageView - Calendar
 - DatePickerDialog - to pick the date of the trip
- Custom AutoCompleteTextView (extends ArrayAdapter<>)
 - Filter - to customize the autocomplete predictions
 - ListView
 - ImageView - to display the image of the location user planned the trip
 - TextView - to display the name of the location user planned the trip

5.6.2. VIEW DETAILS

- LinearLayout - Activity layout
- FrameLayout - to block out an area on the screen to display a single item
 - ImageView - to display the image of the location user planned the trip
 - ImageButton - to share the information about the trip with others
 - Implicit Intent (Intent.ACTION_SEND)
- TextView - to display Trip no, Place and Date of trip
- Button - Update Trip, Cancel Trip

5.6.3. UPDATE TRIP & CANCEL TRIP

- LinearLayout - Activity layout
- ImageView - Calendar
 - DatePickerDialog - to pick the date of the trip
- EditText - displays the location the trip was planned initially to.
- Button - Update the Database with newly entered data.
- AlertDialog box - to confirm the cancelation of the trip

5.7. UTILITY

- ScrollView - Activity layout
- LinearLayout - Child elements of the palette
- FrameLayout - to block out an area on the screen to display a single item
 - ImageButton - to Intent to various functionalities provided by the page (Checklist, Compass, Weather Forecast, World Clock)
 - TextView - to display the names of functionality the ImageButton would navigate to (Checklist, Compass, Weather Forecast, World Clock)

5.7.1. CHECKLIST

- LinearLayout - Activity layout
- Checkbox - to list the items
- ListView - to add more items to list
- EditText - to enter the item to be added to the list
- FloatingActionButton - to initiate the above-mentioned task

5.7.2. COMPASS (implements SensorEventListener)

- RelativeLayout - Activity layout
- TextView - to display the degree of rotation
- ImageView - Compass
- SensorManager - to make the compass show the correct direction.

5.7.3. WORLD CLOCK

- LinearLayout - Activity layout
- TextView - to display the current time, updated time of various Time Zones
- Spinner - list of Time Zones
- Calendar
- SimpleDateFormat
- TimeZone

CHAPTER 6

DATA STORAGE

6.1. DATABASE DESIGN

SQLite is an open source SQL database that stores data to a text file on a device. Android comes in with built-in SQLite database implementation. SQLite Database has methods to create, delete, execute SQL commands, and perform other common database management tasks. In order to access this database, you don't need to establish any kind of connections for it like JDBC, ODBC e.t.c

6.1.1. PACKAGE

The main package is ***android.database.sqlite*** that contains the classes to manage your own databases.

6.1.2. HELPER

For managing all the operations related to the database, a helper class has been given and is called ***SQLiteOpenHelper***. It automatically manages the creation and update of the database.

Two databases are used in the project.

In Table_1. **User**, there are primary key id, component id userName, email & password.

Variable Name	Type	Description
id	STRING	Auto-incrementing primary key
username	STRING	Username for the account
email	STRING	Email address for the account. This will also be used as the username for the login of the user account
password	STRING	Password for the account

User.db Structure

id	userName	email	password
1	Yamuna K	yamuna.pixie@gmail.com	yamuna
2	Test	test@mail.com	test

In Table_2. **Trips**, there are primary key id, component id placeName & date.

Variable Name	Type	Description
id	STRING	Auto-incrementing primary key
placeName	STRING	Name of the place the user is planning the trip
date	STRING	Date when the trip commences

Trips.db Structure

id	placeName	date
1	Kolkata	20-01-2020

CHAPTER 7

PROGRAM DEVELOPMENT

7.1. CODE OF wanderLUST APPLICATION

7.1.1. LOGIN PAGE

JavaFiles

Login Activity.java

```
package com.example.wanderlust;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.text.Html;
import android.text.Spanned;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

import com.google.android.material.snackbar.Snackbar;
import com.google.android.material.textfield.TextInputLayout;

public class LoginActivity extends AppCompatActivity {

    EditText editTextEmail;
    EditText editTextPassword;

    TextInputLayout textInputLayoutEmail;
    TextInputLayout textInputLayoutPassword;

    Button buttonLogin;

    SqliteHelper sqliteHelper;

    @Override
```

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_login);
    sqliteHelper = new SqliteHelper(this);
    initCreateAccountTextView();
    initViews();

    buttonLogin.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            if (validate()) {
                String Email = editTextEmail.getText().toString();
                String Password = editTextPassword.getText().toString();
                User currentUser = sqliteHelper.Authenticate(new
User(null, null, Email, Password));

                if (currentUser != null) {
                    Snackbar.make(buttonLogin, "Successfully Logged
in!", Snackbar.LENGTH_LONG).show();
                    Intent intent = new Intent(LoginActivity.this,
SplashActivity.class);
                    startActivity(intent);
                } else {
                    Snackbar.make(buttonLogin, "Failed to log in ,
please try again", Snackbar.LENGTH_LONG).show();
                }
            }
        });
    });
}

private void initCreateAccountTextView() {
    TextView textViewCreateAccount = (TextView)
findViewById(R.id.textViewCreateAccount);
    textViewCreateAccount.setText(fromHtml("<font color='#000000'>I
don't have account yet. </font><font color='#018db7'>create one</font>"));
    textViewCreateAccount.setOnClickListener(new View.OnClickListener()
{
    @Override

```

```

        public void onClick(View view) {
            Intent intent = new Intent(LoginActivity.this,
RegisterActivity.class);
            startActivity(intent);
        }
    });

}

private void initViews() {
    editTextEmail = (EditText) findViewById(R.id.editTextEmail);
    editTextPassword = (EditText) findViewById(R.id.editTextPassword);
    TextInputLayoutEmail = (TextInputLayout)
findViewById(R.id.textInputLayoutEmail);
    TextInputLayoutPassword = (TextInputLayout)
findViewById(R.id.textInputLayoutPassword);
    buttonLogin = (Button) findViewById(R.id.buttonLogin);

}

@SuppressWarnings("deprecation")
public static Spanned fromHtml(String html) {
    Spanned result;
    if (android.os.Build.VERSION.SDK_INT >=
android.os.Build.VERSION_CODES.N) {
        result = Html.fromHtml(html, Html.FROM_HTML_MODE_LEGACY);
    } else {
        result = Html.fromHtml(html);
    }
    return result;
}

public boolean validate() {
    boolean valid = false;

    String Email = editTextEmail.getText().toString();
    String Password = editTextPassword.getText().toString();

    if (!android.util.Patterns.EMAIL_ADDRESS.matcher(Email).matches()) {
        valid = false;
        TextInputLayoutEmail.setError("Please enter valid email!");
    } else {
        valid = true;
    }
}

```

```

       textInputLayoutEmail.setError(null);
    }

    if (Password.isEmpty()) {
        valid = false;
        TextInputLayoutPassword.setError("Please enter valid
password!");
    } else {
        if (Password.length() > 5) {
            valid = true;
            TextInputLayoutPassword.setError(null);
        } else {
            valid = false;
            TextInputLayoutPassword.setError("Password is too short!");
        }
    }
    return valid;
}

}

```

Sqlite Helper.java

```

package com.example.wanderlust;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class SqliteHelper extends SQLiteOpenHelper {

    public static final String DATABASE_NAME = "loopwiki.com";
    public static final int DATABASE_VERSION = 1;
    public static final String TABLE_USERS = "users";
    public static final String KEY_ID = "id";
    public static final String KEY_USER_NAME = "username";
    public static final String KEY_EMAIL = "email";
    public static final String KEY_PASSWORD = "password";
    public static final String SQL_TABLE_USERS = " CREATE TABLE " +

```

```

TABLE_USERS
    + " ( "
    + KEY_ID + " INTEGER PRIMARY KEY, "
    + KEY_USER_NAME + " TEXT, "
    + KEY_EMAIL + " TEXT, "
    + KEY_PASSWORD + " TEXT"
    + " ) ";

public SqliteHelper(Context context) {
    super(context, DATABASE_NAME, null, DATABASE_VERSION);
}

@Override
public void onCreate(SQLiteDatabase sqLiteDatabase) {
    sqLiteDatabase.execSQL(SQL_TABLE_USERS);
}

@Override
public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {
    sqLiteDatabase.execSQL(" DROP TABLE IF EXISTS " + TABLE_USERS);
}

public void addUser(User user) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(KEY_USER_NAME, user.userName);
    values.put(KEY_EMAIL, user.email);
    values.put(KEY_PASSWORD, user.password);
    long todo_id = db.insert(TABLE_USERS, null, values);
}

public User Authenticate(User user) {
    SQLiteDatabase db = this.getReadableDatabase();
    Cursor cursor = db.query(TABLE_USERS,
        new String[]{KEY_ID, KEY_USER_NAME, KEY_EMAIL,
        KEY_PASSWORD},
        KEY_EMAIL + "=?",
        new String[]{user.email},
        null, null, null);

    if (cursor != null && cursor.moveToFirst()&& cursor.getCount()>0) {
        User user1 = new User(cursor.getString(0), cursor.getString(1),

```

```

        cursor.getString(2), cursor.getString(3));

        if (user.password.equalsIgnoreCase(user1.password)) {
            return user1;
        }
    }
    return null;
}

public boolean isEmailExists(String email) {
    SQLiteDatabase db = this.getReadableDatabase();
    Cursor cursor = db.query(TABLE_USERS,
        new String[]{KEY_ID, KEY_USER_NAME, KEY_EMAIL,
KEY_PASSWORD},
        KEY_EMAIL + "=?",
        new String[]{email},
        null, null, null);

    if (cursor != null && cursor.moveToFirst()&& cursor.getCount()>0) {
        return true;
    }
    return false;
}
}

```

User.java

```

package com.example.wanderlust;

public class User {
    public String id;
    public String userName;
    public String email;
    public String password;

    public User(String id, String userName, String email, String password) {
        this.id = id;
        this.userName = userName;
        this.email = email;
        this.password = password;
    }
}

```

XML Files

activity_login.xml

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/white"
    android:fitsSystemWindows="true">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:layout_marginLeft="16dp"
        android:layout_marginRight="16dp"
        android:orientation="vertical">

        <ImageView
            android:id="@+id/imageView"
            android:layout_width="wrap_content"
            android:layout_height="293dp"
            android:layout_gravity="center_horizontal"
            android:layout_margin="16dp"
            android:src="@drawable/logo" />

        <com.google.android.material.textfield.TextInputLayout
            android:id="@+id/textInputLayoutEmail"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="16dp"
            app:errorEnabled="true">

            <com.google.android.material.textfield.TextInputEditText
                android:id="@+id/editTextEmail"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:hint="@string/email"
                android:inputType="textEmailAddress" />
    
```

```
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:id="@+id/textInputLayoutPassword"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    app:errorEnabled="true"
    app:passwordToggleEnabled="true"
    app:passwordToggleTint="@color/colorAccent">

    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/editTextPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/password"
        android:inputType="textPassword" />
</com.google.android.material.textfield.TextInputLayout>

<Button
    android:id="@+id/buttonLogin"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center_horizontal"
    android:layout_marginTop="16dp"
    android:background="@drawable/button_background"
    android:text="@string/login"
    android:textColor="@android:color/white" />

<TextView
    android:id="@+id/textViewCreateAccount"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:layout_marginTop="16dp"
    android:gravity="center_horizontal" />
</LinearLayout>
</ScrollView>
```

7.1.2. REGISTER PAGE

JavaFiles

RegisterActivity.java

```
package com.example.wanderlust;

import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

import com.google.android.material.snackbar.Snackbar;
import com.google.android.material.textfield.TextInputLayout;

import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;

public class RegisterActivity extends AppCompatActivity {

    //Declaration EditTexts
    EditText editTextUserName;
    EditText editTextEmail;
    EditText editTextPassword;

    //Declaration TextInputLayout
    TextInputLayout textInputLayoutUserName;
    TextInputLayout textInputLayoutEmail;
    TextInputLayout textInputLayoutPassword;

    //Declaration Button
    Button buttonRegister;

    //Declaration SqliteHelper
    SqliteHelper sqliteHelper;

    @Override
    protected void onCreate(@Nullable Bundle savedInstanceState) {
```

```

super.onCreate(savedInstanceState);
setContentView(R.layout.activity_register);

sqliteHelper = new SqliteHelper(this);
initTextViewLogin();
initViews();
buttonRegister.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (validate()) {
            String UserName = editTextUserName.getText().toString();
            String Email = editTextEmail.getText().toString();
            String Password = editTextPassword.getText().toString();
            if (!sqliteHelper.isEmailExists(Email)) {
                sqliteHelper.addUser(new User(null, UserName, Email,
Password));
                Snackbar.make(buttonRegister, "User created
successfully! Please Login ", Snackbar.LENGTH_LONG).show();
                new Handler().postDelayed(new Runnable() {
                    @Override
                    public void run() {
                        finish();
                    }
                }, Snackbar.LENGTH_LONG);
            }else {
                Snackbar.make(buttonRegister, "User already exists
with same email ", Snackbar.LENGTH_LONG).show();
            }
        }
    }
});

private void initTextViewLogin() {
    TextView textViewLogin = (TextView)
findViewById(R.id.textViewLogin);
    textViewLogin.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            finish();
        }
    });
}

```

```

        }
    });
}

private void initViews() {
    editTextEmail = (EditText) findViewById(R.id.editTextEmail);
    editTextPassword = (EditText) findViewById(R.id.editTextPassword);
    editTextUserName = (EditText) findViewById(R.id.editTextUserName);
    textInputLayoutEmail = (TextInputLayout)
            findViewById(R.id.textInputLayoutEmail);
    textInputLayoutPassword = (TextInputLayout)
            findViewById(R.id.textInputLayoutPassword);
    textInputLayoutUserName = (TextInputLayout)
            findViewById(R.id.textInputLayoutUserName);
    buttonRegister = (Button) findViewById(R.id.buttonRegister);

}

public boolean validate() {
    boolean valid = false;

    String UserName = editTextUserName.getText().toString();
    String Email = editTextEmail.getText().toString();
    String Password = editTextPassword.getText().toString();

    if (UserName.isEmpty()) {
        valid = false;
        textInputLayoutUserName.setError("Please enter valid
username!");
    } else {
        if (UserName.length() > 5) {
            valid = true;
            textInputLayoutUserName.setError(null);
        } else {
            valid = false;
            textInputLayoutUserName.setError("Username is too short!");
        }
    }

    //Handling validation for Email field
    if (!android.util.Patterns.EMAIL_ADDRESS.matcher(Email).matches()) {
        valid = false;
    }
}

```

```

        TextInputLayoutEmail.setError("Please enter valid email!");
    } else {
        valid = true;
        TextInputLayoutEmail.setError(null);
    }

    //Handling validation for Password field
    if (Password.isEmpty()) {
        valid = false;
        TextInputLayoutPassword.setError("Please enter valid
password!");
    } else {
        if (Password.length() > 5) {
            valid = true;
            TextInputLayoutPassword.setError(null);
        } else {
            valid = false;
            TextInputLayoutPassword.setError("Password is too short!");
        }
    }

    return valid;
}
}

```

Sqlite Helper.java

```

package com.example.wanderlust;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class SqliteHelper extends SQLiteOpenHelper {

    public static final String DATABASE_NAME = "loopwiki.com";
    public static final int DATABASE_VERSION = 1;
    public static final String TABLE_USERS = "users";
    public static final String KEY_ID = "id";
}

```

```

public static final String KEY_USER_NAME = "username";
public static final String KEY_EMAIL = "email";
public static final String KEY_PASSWORD = "password";
public static final String SQL_TABLE_USERS = " CREATE TABLE " +
TABLE_USERS
    + " ( "
    + KEY_ID + " INTEGER PRIMARY KEY, "
    + KEY_USER_NAME + " TEXT, "
    + KEY_EMAIL + " TEXT, "
    + KEY_PASSWORD + " TEXT"
    + " ) ";
}

public SqliteHelper(Context context) {
    super(context, DATABASE_NAME, null, DATABASE_VERSION);
}

@Override
public void onCreate(SQLiteDatabase sqLiteDatabase) {
    sqLiteDatabase.execSQL(SQL_TABLE_USERS);
}

@Override
public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {
    sqLiteDatabase.execSQL(" DROP TABLE IF EXISTS " + TABLE_USERS);
}

public void addUser(User user) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(KEY_USER_NAME, user.userName);
    values.put(KEY_EMAIL, user.email);
    values.put(KEY_PASSWORD, user.password);
    long todo_id = db.insert(TABLE_USERS, null, values);
}

public User Authenticate(User user) {
    SQLiteDatabase db = this.getReadableDatabase();
    Cursor cursor = db.query(TABLE_USERS,
        new String[]{KEY_ID, KEY_USER_NAME, KEY_EMAIL,
KEY_PASSWORD},
        KEY_EMAIL + "=?", 
        new String[]{user.email},
```

```

        null, null, null);

    if (cursor != null && cursor.moveToFirst()&& cursor.getCount()>0) {
        User user1 = new User(cursor.getString(0), cursor.getString(1),
cursor.getString(2), cursor.getString(3));

        if (user.password.equalsIgnoreCase(user1.password)) {
            return user1;
        }
    }
    return null;
}

public boolean isEmailExists(String email) {
    SQLiteDatabase db = this.getReadableDatabase();
    Cursor cursor = db.query(TABLE_USERS,
            new String[]{KEY_ID, KEY_USER_NAME, KEY_EMAIL,
KEY_PASSWORD},
            KEY_EMAIL + "=?",
            new String[]{email},
            null, null, null);

    if (cursor != null && cursor.moveToFirst()&& cursor.getCount()>0) {
        return true;
    }
    return false;
}
}

```

User.java

```

package com.example.wanderlust;

public class User {
    public String id;
    public String userName;
    public String email;
    public String password;

    public User(String id, String userName, String email, String password) {
        this.id = id;
        this.userName = userName;
    }
}

```

```

        this.email = email;
        this.password = password;
    }
}

```

XML Files

activity_register.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/white"
    android:fitsSystemWindows="true">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:layout_marginLeft="16dp"
        android:layout_marginRight="16dp"
        android:orientation="vertical">

        <ImageView
            android:id="@+id/imageView"
            android:layout_width="wrap_content"
            android:layout_height="298dp"
            android:layout_gravity="center_horizontal"
            android:layout_margin="16dp"
            android:src="@drawable/logo" />

        <com.google.android.material.textfield.TextInputLayout
            app:errorEnabled="true"
            android:id="@+id/textInputLayoutUserName"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="16dp">

            <com.google.android.material.textfield.TextInputEditText
                android:id="@+id/editTextUserName"

```

```
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/username"
        android:inputType="textPersonName" />
    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:id="@+id/textInputLayoutEmail"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        app:errorEnabled="true">

        <com.google.android.material.textfield.TextInputEditText
            android:id="@+id/editTextEmail"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="@string/email"
            android:inputType="textEmailAddress" />
    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:id="@+id/textInputLayoutPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        app:errorEnabled="true"
        app:passwordToggleEnabled="true"
        app:passwordToggleTint="@color/colorAccent">

        <com.google.android.material.textfield.TextInputEditText
            android:id="@+id/editTextPassword"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="@string/password"
            android:inputType="textPassword" />
    </com.google.android.material.textfield.TextInputLayout>

    <Button
        android:id="@+id/buttonRegister"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

```
        android:textColor="@android:color/white"
        android:layout_gravity="center_horizontal"
        android:layout_marginTop="16dp"
        android:layout_weight="1"
        android:background="@drawable/button_background"
        android:text="@string/register" />

    <TextView
        android:id="@+id/textViewLogin"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_marginTop="16dp"
        android:gravity="center"
        android:text="@string/back_to_login"
        android:textColor="@color/colorAccent" />
</LinearLayout>
</ScrollView>
```

7.1.3. HOME PAGE

JavaFiles

HomeActivity.java

```
package com.example.wanderlust;

import android.content.DialogInterface;
import android.content.Intent;
import android.graphics.Color;
import android.graphics.drawable.ColorDrawable;
import android.media.Image;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.ImageButton;

import androidx.appcompat.app.ActionBar;
```

```
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;

public class HomeActivity extends AppCompatActivity {

    ImageButton destination;
    ImageButton booking;
    ImageButton trips;
    ImageButton utility;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_home);

        ActionBar actionBar;
        actionBar = getSupportActionBar();
        ColorDrawable colorDrawable
            = new ColorDrawable(Color.parseColor("#018db7"));
        actionBar.setBackgroundDrawable(colorDrawable);

        utility = (ImageButton) findViewById(R.id.utility);
        utility.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new
                    Intent(getApplicationContext(), UtilityActivity.class);
                startActivity(intent);
            }
        });

        destination = (ImageButton) findViewById(R.id.destination);
        destination.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new
                    Intent(getApplicationContext(), Destination.class);
                startActivity(intent);
            }
        });

        booking = (ImageButton) findViewById(R.id.booking);
    }
}
```

```

        booking.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new
Intent(getApplicationContext(),BookingActivity.class);
                startActivity(intent);
            }
        });

        trips = (ImageButton) findViewById(R.id.trips);
        trips.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new
Intent(getApplicationContext(),DisplaySQLiteDataActivity.class);
                startActivity(intent);
            }
        });
    }

    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.options_menu, menu);
        return true;
    }

    public boolean onOptionsItemSelected(MenuItem item) {
        switch (item.getItemId()) {
            case R.id.logout:
                AlertDialog.Builder builder = new
AlertDialog.Builder(HomeActivity.this);
                builder.setTitle("Confirmation!");
                setMessage("You sure, that you want to logout?");
                builder.setPositiveButton("Yes",
                        new DialogInterface.OnClickListener() {
                            public void onClick(DialogInterface dialog, int
id) {
                                Intent i = new
Intent(getApplicationContext(),
                                LoginActivity.class);
                                startActivity(i);
                            }
                        });
        }
    }
}

```

```

        builder.setNegativeButton("No",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog, int
id) {
                    dialog.cancel();
                }
            });
        AlertDialog alertDialog = builder.create();
        alertDialog.show();
        return true;
    default:
        return super.onOptionsItemSelected(item);
    }
}
}

```

XML Files

activity_home.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:fitsSystemWindows="true">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="16dp"
        android:layout_marginRight="16dp"
        android:layout_marginBottom="16dp"
        android:layout_marginTop="16dp"
        android:orientation="vertical">

        <FrameLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="center_vertical"
            android:orientation="vertical">

```

```
<ImageButton
    android:id="@+id/destination"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@drawable/destination"/>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:text="DESTINATION"
    android:textColor="#FFFFFF"
    android:textSize="20dp"
    android:textStyle="bold" />
</FrameLayout>

<FrameLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center_vertical"
    android:layout_marginTop="16dp"
    android:orientation="vertical">

<ImageButton
    android:id="@+id/booking"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@drawable/hotels"/>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:text="HOTEL BOOKING"
    android:textSize="20dp"
    android:textColor="#FFFFFF"
    android:textStyle="bold" />
</FrameLayout>

<FrameLayout
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
    android:layout_gravity="center_vertical"
    android:layout_marginTop="16dp"
    android:orientation="vertical">

    <ImageButton
        android:id="@+id/trips"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@drawable/trips" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="MY TRIP"
        android:textSize="20dp"
        android:textColor="#FFFFFF"
        android:textStyle="bold" />
</FrameLayout>

<FrameLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center_vertical"
    android:layout_marginTop="16dp"
    android:orientation="vertical">

    <ImageButton
        android:id="@+id/utility"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@drawable/utilities"/>

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="UTILITY"
        android:textSize="20dp"
        android:textColor="#FFFFFF"
        android:textStyle="bold" />
```

```
</FrameLayout>

</LinearLayout>

</ScrollView>
```

7.1.4. DESTINATION

JavaFiles

Destination

```
package com.example.wanderlust;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.ImageButton;

public class Destination extends AppCompatActivity {

    ImageButton
banglore,mumbai,delhi,chennai,pune,kolkata,hyderabad,panaji,ahmedabad,jaipur;
    String city_name;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_destination);

        banglore = (ImageButton) findViewById(R.id.banglore);
        banglore.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                city_name="Bangalore";
                Intent intent = new Intent(getApplicationContext(),
```

```
CityActivity.class);
        intent.putExtra("city",city_name);
        startActivity(intent);
    }
});

mumbai = (ImageButton) findViewById(R.id.mumbai);
mumbai.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        city_name="Mumbai";
        Intent intent = new Intent(getApplicationContext(),
CityActivity.class);
        intent.putExtra("city",city_name);
        startActivity(intent);
    }
});

delhi = (ImageButton) findViewById(R.id.delhi);
delhi.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        city_name="Delhi";
        Intent intent = new Intent(getApplicationContext(),
CityActivity.class);
        intent.putExtra("city",city_name);
        startActivity(intent);
    }
});

chennai = (ImageButton) findViewById(R.id.chennai);
chennai.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        city_name="Chennai";
        Intent intent = new Intent(getApplicationContext(),
CityActivity.class);
        intent.putExtra("city",city_name);
        startActivity(intent);
    }
});
});
```

```

pune = (ImageButton) findViewById(R.id.pune);
pune.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        city_name="Pune";
        Intent intent = new Intent(getApplicationContext(),
CityActivity.class);
        intent.putExtra("city",city_name);
        startActivity(intent);
    }
});

kolkata = (ImageButton) findViewById(R.id.kolkata);
kolkata.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        city_name="Kolkata";
        Intent intent = new Intent(getApplicationContext(),
CityActivity.class);
        intent.putExtra("city",city_name);
        startActivity(intent);
    }
});

hyderabad = (ImageButton) findViewById(R.id.hydrabad);
hyderabad.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        city_name="Hyderabad";
        Intent intent = new Intent(getApplicationContext(),
CityActivity.class);
        intent.putExtra("city",city_name);
        startActivity(intent);
    }
});

panaji = (ImageButton) findViewById(R.id.panaji);
panaji.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        city_name="Panaji";
        Intent intent = new Intent(getApplicationContext(),

```

```
CityActivity.class);
        intent.putExtra("city",city_name);
        startActivity(intent);
    }
});

ahmedabad = (ImageButton) findViewById(R.id.ahmedabad);
ahmedabad.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        city_name="Ahmedabad";
        Intent intent = new Intent(getApplicationContext(),
CityActivity.class);
        intent.putExtra("city",city_name);
        startActivity(intent);
    }
});

jaipur = (ImageButton) findViewById(R.id.jaipur);
jaipur.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        city_name="Jaipur";
        Intent intent = new Intent(getApplicationContext(),
CityActivity.class);
        intent.putExtra("city",city_name);
        startActivity(intent);
    }
});

}
}
```

CityActivity.java

```
package com.example.wanderlust;

import androidx.appcompat.app.AppCompatActivity;
import androidx.cardview.widget.CardView;
```

```
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.TextView;

import com.denzcoskun.imageslider.ImageSlider;
import com.denzcoskun.imageslider.models.SlideModel;
import com.google.android.gms.maps.model.LatLng;

import org.w3c.dom.Text;

import java.util.ArrayList;
import java.util.List;

public class CityActivity extends AppCompatActivity {
    CardView funfacts;
    CardView hangouts;
    CardView destinations;
    CardView shopping;
    CardView restaurant;
    CardView citymap;
    CardView weather;
    ImageSlider is;
    String city;
    public LatLng location;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_city);
        Intent intent = getIntent();
        city=intent.getStringExtra("city");

        TextView imagetext = (TextView) findViewById(R.id.imagetext);
        imagetext.setText(city);

        //Funfacts
        funfacts=findViewById(R.id.fetchdata);
        funfacts.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
```

```

        Intent intent = new Intent(CityActivity.this,
Funfacts.class);
        intent.putExtra("city",city);
        startActivity(intent); }
});

//Set Slides and LatLng

switch(city){
    case "Bangalore":

slides(R.drawable.blr_slide_1,R.drawable.blr_slide_2,R.drawable.blr_slides_
3);
    location = new LatLng(12.9716, 77.5946);
    break;
    case "Mumbai":

slides(R.drawable.mumbai_slide_1,R.drawable.mumbai_slide_2,R.drawable.mumba
i_slide_3);
    location = new LatLng(19.0760,72.8777);
    break;
    case "Delhi":

slides(R.drawable.delhi_slide_2,R.drawable.delhi_slide_1,R.drawable.delhi_s
lide_3);
    location = new LatLng(28.7041,77.1025);
    break;
    case "Chennai":

slides(R.drawable.chennai_slide_1,R.drawable.chennai_slide_2,R.drawable.che
nnai_slide_3);
    location = new LatLng(13.0827,80.2707);
    break;
    case "Pune":

slides(R.drawable.pune_slides_1,R.drawable.pune_slides_2,R.drawable.pune_sl
ides_3);
    location = new LatLng(18.5204,73.8567);
    break;
    case "Kolkata":
```

```

        slides(R.drawable.kolkata_slides_1,R.drawable.kolkata_slides_2,R.drawable.k
olkata_slides_3);
            location = new LatLng(22.5726,88.3639);
            break;
        case "Hyderabad":

        slides(R.drawable.hyderabad_slides_1,R.drawable.hyderabad_slides_2,R.drawab
le.hyderabad_slides_3);
            location = new LatLng(17.3850,78.4867);
            break;
        case "Panaji":

        slides(R.drawable.panaji_slides_1,R.drawable.panaji_slides_2,R.drawable.pan
aji_slides_3);
            location = new LatLng(15.4909,73.8278);
            break;
        case "Ahmedabad":

        slides(R.drawable.ahmedabad_slides_1,R.drawable.ahmedabad_slides_2,R.drawab
le.ahmedabad_slides_3);
            location = new LatLng(23.0225,72.5714);
            break;
        case "Jaipur":

        slides(R.drawable.jaipur_slides_1,R.drawable.jaipur_slides_2,R.drawable.jai
pur_slides_3);
            location = new LatLng(26.9124,75.7873);
            break;

    }

    // Hangouts

    hangouts=findViewById(R.id.hangout);
    hangouts.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            String type="cafe|park";
            Bundle args = new Bundle();
            args.putString("type",type);

```

```

        args.putParcelable("location", location);
        Intent intent = new Intent(getApplicationContext(),
PermissionsActivity.class);
        intent.putExtra("bundle",args);
        startActivity(intent);
    }
});

//Destination
destinations = (CardView) findViewById(R.id.destinations);
destinations.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String type="tourist_attraction";
        Bundle args = new Bundle();
        args.putString("type",type);
        args.putParcelable("location", location);
        Intent intent = new Intent(getApplicationContext(),
PermissionsActivity.class);
        intent.putExtra("bundle",args);
        startActivity(intent);

    }
});

//shopping
shopping = (CardView) findViewById(R.id.shopping);
shopping.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String type="shopping_mall";
        Bundle args = new Bundle();
        args.putString("type",type);
        args.putParcelable("location", location);
        Intent intent = new Intent(getApplicationContext(),
PermissionsActivity.class);
        intent.putExtra("bundle",args);
        startActivity(intent);
    }
});

//Restaurant

```

```

restaurant = (CardView) findViewById(R.id.restaurant);
restaurant.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String type="restaurant";
        Bundle args = new Bundle();
        args.putString("type",type);
        args.putParcelable("location", location);
        Intent intent = new Intent(getApplicationContext(),
PermissionsActivity.class);
        intent.putExtra("bundle",args);
        startActivity(intent);
    }
});

//CityMap
citymap=findViewById(R.id.citymap);
citymap.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Bundle args = new Bundle();
        args.putParcelable("location", location);
        Intent intent = new
Intent(getApplicationContext(),CityMap.class);
        intent.putExtra("bundle",args);
        startActivity(intent);

    }
});

weather = findViewById(R.id.check_weather);
weather.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new
Intent(getApplicationContext(),WeatherActivity.class);
        intent.putExtra("city",city);
        startActivity(intent);
    }
});

```

```
}

private void slides(int img_1,int img_2,int img_3) {
    is= findViewById(R.id.slides);
    List<SlideModel> sm = new ArrayList<>();
    sm.add(new SlideModel(img_1));
    sm.add(new SlideModel(img_2));
    sm.add(new SlideModel(img_3));
    is.setImageList(sm,true);
}
}
```

FunFacts.java

```
package com.example.wanderlust;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.widget.TextView;

public class Funfacts extends AppCompatActivity {

    public static TextView fetchdata;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_funfacts);

        Intent intent = getIntent();
        String city=intent.getStringExtra("city");
    }
}
```

```
        Log.d("TAG", "onCreate: "+city);
        fetchdata=(TextView) findViewById(R.id.funfactstv);
        fetchwiki fetchprocess= new fetchwiki(city);
        fetchprocess.execute();

    }
}
```

PermissionActivity.java

```
package com.example.wanderlust;

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

import android.Manifest;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.net.Uri;
import android.os.Bundle;
import android.provider.Settings;
import android.util.Log;
import android.view.View;
import android.webkit.PermissionRequest;
import android.widget.Button;
import android.widget.Toast;

import com.google.android.gms.maps.model.LatLng;
import com.karumi.dexter.Dexter;
import com.karumi.dexter.PermissionToken;
import com.karumi.dexter.listener.PermissionDeniedResponse;
import com.karumi.dexter.listener.PermissionGrantedResponse;
import com.karumi.dexter.listener.single.PermissionListener;

public class PermissionsActivity extends AppCompatActivity {
```

```

private Button btnGrant;
private LatLng place1;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_permissions_activity);
    btnGrant = findViewById(R.id.btn_grant);
    if(ContextCompat.checkSelfPermission(PermissionsActivity.this,
Manifest.permission.ACCESS_FINE_LOCATION) ==
PackageManager.PERMISSION_GRANTED){

        Bundle bundle = getIntent().getParcelableExtra("bundle");
        place1 = bundle.getParcelable("location");
        String type = bundle.getString("type");

        Bundle args = new Bundle();
        args.putParcelable("location", place1);
        args.putString("type",type);
        Intent intent = new
Intent(getApplicationContext(),Nearby.class);
        intent.putExtra("bundles",args);
        startActivity(intent);
        finish();
        return;
    }

    //If the permission is not available
    btnGrant.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {

            Dexter.withContext(PermissionsActivity.this)

```

```

.withPermission(Manifest.permission.ACCESS_FINE_LOCATION)
        .withListener(new PermissionListener() {
            @Override
            public void
onPermissionGranted(PermissionGrantedResponse response) {
                startActivity(new
Intent(PermissionsActivity.this, CityActivity.class));
                finish();
            }

            @Override
            public void
onPermissionDenied(PermissionDeniedResponse response) {
                if(response.isPermanentlyDenied()){
                    AlertDialog.Builder builder = new
AlertDialog.Builder(PermissionsActivity.this);
                    builder.setTitle("Permission Denied")
                        .setMessage("Permission to
access device location is permanently denied. you need to go to setting to
allow the permission.")
                        .setNegativeButton("Cancel",
null)
                        .setPositiveButton("OK", new
DialogInterface.OnClickListener() {
                            @Override
                            public void
onClick(DialogInterface dialog, int which) {
                                Intent intent = new
Intent();
                                intent.setAction(Settings.ACTION_APPLICATION_DETAILS_SETTINGS);
                                intent.setData(Uri.fromParts("package",
getPackageName(), null));
                            }
                        })
                        .show();
                } else {
                    Toast.makeText(PermissionsActivity.this,
"Permission Denied", Toast.LENGTH_SHORT).show();
                }
            }
        }
    }
}

```

```
        @Override
        public void
onPermissionRationaleShouldBeShown(com.karumi.dexter.listener.PermissionReq
uest permissionRequest, PermissionToken permissionToken) {
            permissionToken.continuePermissionRequest();
        }

    })
    .check();
}

);
}

}
```

Nearby.java

```
package com.example.wanderlust;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;
import androidx.fragment.app.FragmentActivity;

import android.Manifest;
import android.content.IntentSender;
import android.content.pm.PackageManager;
import android.location.Location;
import android.os.Bundle;

import com.google.android.gms.common.ConnectionResult;
import com.google.android.gms.common.api.GoogleApiClient;
import com.google.android.gms.common.api.PendingResult;
import com.google.android.gms.common.api.ResultCallback;
import com.google.android.gms.common.api.Status;
import com.google.android.gms.location.LocationListener;
```

```
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.location.LocationSettingsRequest;
import com.google.android.gms.location.LocationSettingsResult;
import com.google.android.gms.location.LocationSettingsStatusCodes;
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.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;

import java.util.ArrayList;
import java.util.List;

public class Nearby extends FragmentActivity implements OnMapReadyCallback,
GoogleApiClient.ConnectionCallbacks,GoogleApiClient.OnConnectionFailedListe
ner,LocationListener {

    private GoogleMap mMap;
    public LatLng place;

    private final float DEFAULT_ZOOM = 15;
    private GoogleApiClient mgoogleApiClient;
    double currentLatitude,currentLongitude;
    Location mylocation;
    public String type;

    private final static int REQUEST_CHECK_SETTINGS_GPS=0x1;
    private final static int REQUEST_ID_MULTIPLE_PERMISSION=0x2;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_nearby);
        // Obtain the SupportMapFragment and get notified when the map is
        ready to be used.
        SupportMapFragment mapFragment = (SupportMapFragment)
getSupportFragmentManager()
```

```

        .findFragmentById(R.id.map);
mapFragment.getMapAsync(this);

Bundle bundle = getIntent().getParcelableExtra("bundles");
this.place = bundle.getParcelable("location");
type = bundle.getString("type");

setClient();
}

private void setClient() {
    mgoogleApiClient = new GoogleApiClient.Builder(this)
        .enableAutoManage(this,0,this)
        .addConnectionCallbacks(this)
        .addOnConnectionFailedListener(this)
        .addApi(LocationServices.API)
        .build();
    mgoogleApiClient.connect();
}

@Override
public void onMapReady(GoogleMap googleMap) {
    mMap = googleMap;

    mMap.addMarker(new
MarkerOptions().position(place).title("Marker").icon(BitmapDescriptorFactor
y.fromResource(R.drawable.gps)));
    mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(place,DEFAULT_ZOOM
));
}

private void getNearBy(String type) {
    StringBuilder stringBuilder =
        new

```

```

StringBuilder("https://maps.googleapis.com/maps/api/place/nearbysearch/json
?");

stringBuilder.append("location="+String.valueOf(place.latitude)+","+String.
valueOf(place.longitude));
    stringBuilder.append("&radius=3000");
    stringBuilder.append("&type="+type);

stringBuilder.append("&key="+getResources().getString(R.string.google_maps_
key));

    String url = stringBuilder.toString();

    Object dataTransfer[] = new Object[2];
    dataTransfer[0] = mMap;
    dataTransfer[1] = url;

    GetNearByPlacesData getNearByPlacesData = new
GetNearByPlacesData();
    getNearByPlacesData.execute(dataTransfer);
}

@Override
public void onConnected(@Nullable Bundle bundle) {
    checkPermission();

}

private void checkPermission() {
    int permissionLocation =
ContextCompat.checkSelfPermission(Nearby.this,
        Manifest.permission.ACCESS_FINE_LOCATION);

    List<String> listpermission = new ArrayList<>();
    if(permissionLocation != PackageManager.PERMISSION_GRANTED){
        listpermission.add( Manifest.permission.ACCESS_FINE_LOCATION);

        if(!listpermission.isEmpty()) {

ActivityCompat.requestPermissions(this,listpermission.toArray(new
String[listpermission.size()]),REQUEST_ID_MULTIPLE_PERMISSION);

```

```
        }
    }
    else {
        getMyLocation();
    }
}

@Override
public void onRequestPermissionsResult(int requestCode, @NonNull
String[] permissions, @NonNull int[] grantResults) {
    int permissionLocation =
ContextCompat.checkSelfPermission(this,Manifest.permission.ACCESS_FINE_LOCATION);

    if(permissionLocation == PackageManager.PERMISSION_GRANTED){
        getMyLocation();

    } else {
        checkPermission();
    }
}

@Override
public void onConnectionSuspended(int i) {
}

@Override
public void onConnectionFailed(@NonNull ConnectionResult
connectionResult) {

}

@Override
public void onLocationChanged(Location location) {

    getNearBy(type);
}

private void getMyLocation(){
```

```

        if(mgoogleApiClient!=null) {
            if (mgoogleApiClient.isConnected()) {
                int permissionLocation =
ContextCompat.checkSelfPermission(Nearby.this,
                                Manifest.permission.ACCESS_FINE_LOCATION);
                if (permissionLocation == PackageManager.PERMISSION_GRANTED)
{
                    mylocation =
LocationServices.FusedLocationApi.getLastLocation(mgoogleApiClient);
                    LocationRequest locationRequest = new LocationRequest();

locationRequest.setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);
                    LocationSettingsRequest.Builder builder = new
LocationSettingsRequest.Builder()
                        .addLocationRequest(locationRequest);
                    builder.setAlwaysShow(true);
                    LocationServices.FusedLocationApi
                        .requestLocationUpdates(mgoogleApiClient,
locationRequest, (LocationListener) this);
                    PendingResult<LocationSettingsResult> result =
                        LocationServices.SettingsApi
                            .checkLocationSettings(mgoogleApiClient,
builder.build());
                    result.setResultCallback(new
ResultCallback<LocationSettingsResult>() {

                        @Override
                        public void onResult(LocationSettingsResult result)
{
                            final Status status = result.getStatus();
                            switch (status.getStatusCode()) {
                                case LocationSettingsStatusCodes.SUCCESS:

int permissionLocation = ContextCompat

.checkSelfPermission(Nearby.this,
                                Manifest.permission.ACCESS_FINE_LOCATION);
                                if (permissionLocation ==
PackageManager.PERMISSION_GRANTED) {

```


XML Files

activity_destination.xml

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#ffff"
    android:fitsSystemWindows="true">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="16dp"
        android:layout_marginRight="16dp"
        android:layout_marginBottom="16dp"
        android:layout_marginTop="16dp"
        android:orientation="vertical">

        <FrameLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="center_vertical"

            android:orientation="vertical">

            <ImageButton
                android:id="@+id/bangalore"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:background="@drawable/blore"/>

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_gravity="center"
                android:text="BANGLORE"
                android:textColor="#FFFFFF"
```

```
        android:textSize="20dp"
        android:textStyle="bold" />
    </FrameLayout>

    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:layout_marginTop="16dp"
        android:orientation="vertical">

        <ImageButton
            android:id="@+id/mumbai"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@drawable/mumbai"/>

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:text="MUMBAI"
            android:textSize="20dp"
            android:textColor="#FFFFFF"
            android:textStyle="bold" />
    </FrameLayout>

    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:layout_marginTop="16dp"
        android:orientation="vertical">

        <ImageButton
            android:id="@+id/delhi"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@drawable/delhi" />

        <TextView
            android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="DELHI"
        android:textSize="20dp"
        android:textColor="#FFFFFF"
        android:textStyle="bold" />
    </FrameLayout>

    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:layout_marginTop="16dp"
        android:orientation="vertical">

        <ImageButton
            android:id="@+id/chennai"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@drawable/chennai"/>

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:text="CHENNAI"
            android:textSize="20dp"
            android:textColor="#FFFFFF"
            android:textStyle="bold" />
    </FrameLayout>

    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:layout_marginTop="16dp"
        android:orientation="vertical">

        <ImageButton
            android:id="@+id/pune"
            android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
        android:background="@drawable/pune" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="PUNE"
        android:textSize="20dp"
        android:textColor="#FFFFFF"
        android:textStyle="bold" />
    </FrameLayout>

    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:layout_marginTop="16dp"
        android:orientation="vertical">

        <ImageButton
            android:id="@+id/kolkata"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@drawable/kolkata" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:text="KOLKATA"
            android:textSize="20dp"
            android:textColor="#FFFFFF"
            android:textStyle="bold" />
    </FrameLayout>

    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:layout_marginTop="16dp"
        android:orientation="vertical">
```

```
<ImageButton
    android:id="@+id/hyderabad"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@drawable/hyderabad" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:text="HYDERABAD"
    android:textSize="20dp"
    android:textColor="#FFFFFF"
    android:textStyle="bold" />
</FrameLayout>

<FrameLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center_vertical"
    android:layout_marginTop="16dp"
    android:orientation="vertical">

<ImageButton
    android:id="@+id/panaji"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@drawable/panaji" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:text="PANAJI"
    android:textSize="20dp"
    android:textColor="#FFFFFF"
    android:textStyle="bold" />
</FrameLayout>

<FrameLayout
    android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:layout_marginTop="16dp"
        android:orientation="vertical">

    <ImageButton
        android:id="@+id/ahmedabad"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@drawable/ahmedabad" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="AHMEDABAD"
        android:textSize="20dp"
        android:textColor="#FFFFFF"
        android:textStyle="bold" />
</FrameLayout>

<FrameLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center_vertical"
    android:layout_marginTop="16dp"
    android:orientation="vertical">

    <ImageButton
        android:id="@+id/jaipur"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@drawable/jaipur" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="JAIPUR"
        android:textSize="20dp"
        android:textColor="#FFFFFF"
        android:textStyle="bold" />
```

```
</FrameLayout>

</LinearLayout>
</ScrollView>
```

activity_city.xml

```
<?xml version="1.0" encoding="utf-8"?>

<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:card_view="http://schemas.android.com/tools"
    android:background="#ffff"
    android:fitsSystemWindows="true"
    android:scrollbars="none">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <FrameLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">

            <com.denzcoskun.imageslider.ImageSlider
                android:layout_width="match_parent"
                android:layout_height="250dp"
                android:layout_gravity="center_horizontal"
                android:id="@+id/slides"
                app:auto_cycle="true"
                app:placeholder="@color/white"
                app:corner_radius="5"
                app:period="2000"
                />
        
```

```
<TextView
    android:id="@+id/imagetext"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text=""
    android:layout_marginTop="200dp"
    android:layout_marginLeft="22dp"
    android:textColor="#FFFFFF"
    android:textSize="20dp"
    android:textStyle="bold" />

</FrameLayout>

<androidx.cardview.widget.CardView
    android:id="@+id/check_weather"
    android:layout_width="match_parent"
    android:layout_height="120dp"
    android:layout_gravity="center_vertical"
    android:layout_marginTop="25dp"
    android:layout_marginLeft="14dp"
    android:layout_marginRight="14dp"
    android:orientation="vertical"
    app:cardCornerRadius="5dp"
    app:cardUseCompatPadding="true"
    app:cardElevation="10dp"
    app:cardBackgroundColor="@color/white">

    <TextView
        android:id="@+id/destination"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:drawableLeft="@drawable/check_weather"
        android:backgroundTint="@color/cardview_light_background"
        android:padding="10dp"
        android:textStyle="bold"
        android:gravity="center_vertical"
        android:text="Check Weather"/>

</androidx.cardview.widget.CardView>
```

```
<androidx.cardview.widget.CardView
    android:id="@+id/fetchdata"
    android:layout_width="match_parent"
    android:layout_height="70dp"
    android:layout_marginLeft="14dp"
    android:layout_marginTop="1dp"
    android:layout_marginRight="14dp"
    android:orientation="horizontal"
    app:cardCornerRadius="5dp"
    app:cardUseCompatPadding="true"
    app:cardElevation="10dp">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:drawableLeft="@drawable/ic_info_outline_black_24dp"
        android:text="    Fun Facts about the City"
        android:layout_gravity="center_horizontal"
        android:padding="10dp"
        android:textStyle="bold"
        android:background="@color/cardview_light_background"
        />
</androidx.cardview.widget.CardView>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="12dp"
    android:layout_marginRight="12dp"
    android:orientation="horizontal"
    android:weightSum="2">

    <androidx.cardview.widget.CardView
        android:id="@+id/hangout"
        android:layout_width="match_parent"
        android:layout_weight="1"
        android:layout_height="70dp"
        android:layout_marginRight="5dp"
        android:orientation="horizontal"
        app:cardCornerRadius="5dp"
        app:cardUseCompatPadding="true"
```

```
    app:cardElevation="10dp">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:drawableLeft="@drawable/location"
            android:text=" Hangout"
            android:layout_gravity="center_horizontal"
            android:padding="10dp"
            android:textStyle="bold"
            android:background="@color/cardview_light_background"/>
    </androidx.cardview.widget.CardView>

    <androidx.cardview.widget.CardView
        android:id="@+id/destinations"
        android:layout_width="match_parent"
        android:layout_height="70dp"
        android:layout_marginLeft="5dp"
        android:layout_weight="1"
        android:orientation="horizontal"
        app:cardCornerRadius="5dp"
        app:cardUseCompatPadding="true"
        app:cardElevation="10dp">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:drawableLeft="@drawable/cityloc"
            android:text=" Destination"
            android:layout_gravity="center_horizontal"
            android:padding="10dp"
            android:textStyle="bold"
            android:background="@color/cardview_light_background"/>
    </androidx.cardview.widget.CardView>

</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
```

```
        android:layout_marginLeft="12dp"
        android:layout_marginRight="12dp"
        android:orientation="horizontal"
        android:weightSum="2">

    <androidx.cardview.widget.CardView
        android:id="@+id/shopping"
        android:layout_width="match_parent"
        android:layout_weight="1"
        android:layout_height="70dp"
        android:layout_marginRight="5dp"
        android:orientation="horizontal"
        app:cardCornerRadius="5dp"
        app:cardUseCompatPadding="true"
        app:cardElevation="10dp">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:drawableLeft="@drawable/shoppingcart"
            android:text=" Shopping"
            android:layout_gravity="center_horizontal"
            android:padding="10dp"
            android:textStyle="bold"
            android:background="@color/cardview_light_background"/>
    </androidx.cardview.widget.CardView>

    <androidx.cardview.widget.CardView
        android:id="@+id/restaurant"
        android:layout_width="match_parent"
        android:layout_height="70dp"
        android:layout_marginLeft="5dp"
        android:layout_weight="1"
        android:orientation="horizontal"
        app:cardCornerRadius="5dp"
        app:cardUseCompatPadding="true"
        app:cardElevation="10dp">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
```

```
        android:drawableLeft="@drawable/ic_restaurant_black_24dp"
            android:text=" Restaurants"
            android:layout_gravity="center_horizontal"
            android:padding="10dp"
            android:textStyle="bold"
            android:background="@color/cardview_light_background"/>
    
```

```
        </androidx.cardview.widget.CardView>

    
```

```
    </LinearLayout>

    <androidx.cardview.widget.CardView
        android:id="@+id/citymap"
        android:layout_width="match_parent"
        android:layout_height="70dp"
        android:layout_marginLeft="14dp"
        android:layout_marginTop="1dp"
        android:layout_marginRight="14dp"
        android:orientation="horizontal"
        app:cardCornerRadius="5dp"
        app:cardUseCompatPadding="true"
        app:cardElevation="10dp">

```

```
        <TextView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:drawableLeft="@drawable/mapsc"
            android:text=" City Maps"
            android:layout_gravity="center_horizontal"
            android:padding="10dp"
            android:textStyle="bold"
            android:background="@color/cardview_light_background"/>
    
```

```
        </androidx.cardview.widget.CardView>
    
```

```
    </LinearLayout>
</ScrollView>
```

activity_funfacts.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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=".Funfacts">

    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <TextView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:id="@+id/funfactstv"
            android:textColor="#000"
            android:textSize="24dp"
            android:hint="Fetching Data "/>
    </ScrollView>

</RelativeLayout>
```

activity_permission_activity.xml

```
<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout 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=".PermissionsActivity"
    android:padding="32dp">

    <LinearLayout
        android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:gravity="center_horizontal">

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@mipmap/ic_launcher" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Permission Request"
        android:textSize="24sp"
        android:layout_marginTop="12dp"
        android:fontFamily="sans-serif-condensed"
        android:textStyle="bold"/>

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:layout_marginTop="36dp"
        android:text="Places Project needs permission to acces device
location to provide required services. Please allow the permission"/>

</LinearLayout>

<Button
    android:id="@+id/btn_grant"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:background="@color/colorPrimary"
    android:textColor="#FFFFFF"
    android:text="Grant Permission"
    android:textAllCaps="false" />

</RelativeLayout>
```

activity_nearby.xml

```
<?xml version="1.0" encoding="utf-8"?>
<fragment xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:map="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/map"
    android:name="com.google.android.gms.maps.SupportMapFragment"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".Nearby" />
```

7.1.5. HOTEL BOOKING

JavaFiles

BookingActivity.java

```
package com.example.wanderlust;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.webkit.WebView;

public class BookingActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_booking);

        WebView mywebview = (WebView) findViewById(R.id.webview);
        mywebview.loadUrl("https://www.booking.com/");
        finish();

    }
}
```

XML Files

activity_booking.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=".BookingActivity">

    <WebView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        android:id="@+id/webview"/>

</androidx.constraintlayout.widget.ConstraintLayout>
```

7.1.6. MY TRIP

JavaFiles

AutoCompletePlaceHolder.java

```
package com.example.wanderlust;

import android.content.Context;
import android.graphics.drawable.Drawable;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ArrayAdapter;
```

```
import android.widget.Filter;
import android.widget.ImageView;
import android.widget.TextView;

import java.util.ArrayList;
import java.util.List;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;

public class AutoCompletePlaceAdapter extends ArrayAdapter<PlaceItem> {
    private List<PlaceItem> placeListFull;
    public static Drawable imageDrawable;

    public AutoCompletePlaceAdapter(@NonNull Context context, @NonNull
List<PlaceItem> placeList) {
        super(context, 0, placeList);
        placeListFull = new ArrayList<>(placeList);
    }

    @NonNull
    @Override
    public Filter getFilter() {
        return placeFilter;
    }

    @NonNull
    @Override
    public View getView(int position, @Nullable View convertView, @NonNull
ViewGroup parent) {
        if(convertView == null){
            convertView =
LayoutInflater.from(getContext()).inflate(R.layout.place_autocomplete_row,p
arent,false);
        }

        TextView textViewName =
convertView.findViewById(R.id.text_view_name);
        ImageView imageViewPlace =
convertView.findViewById(R.id.image_view_place);
        PlaceItem placeItem = getItem(position);
    }
}
```

```

        if(placeItem != null){
            textViewName.setText(placeItem.getPlaceName());
            imageViewPlace.setImageResource(placeItem.getPlaceImage());
        }
        return convertView;
    }

private Filter placeFilter = new Filter() {
    @Override
    protected FilterResults performFiltering(CharSequence charSequence)
    {
        FilterResults results = new FilterResults();
        List<PlaceItem> suggestions = new ArrayList<>();

        if(charSequence == null||charSequence.length() == 0) {
            suggestions.addAll(placeListFull);
        }else {
            String filterPattern =
charSequence.toString().toLowerCase().trim();

            for (PlaceItem item : placeListFull) {
                if
(item.getPlaceName().toLowerCase().contains(filterPattern)) {
                    suggestions.add(item);
                }
            }
        }
        results.values = suggestions;
        results.count = suggestions.size();
        return results;
    }

    @Override
    protected void publishResults(CharSequence charSequence,
FilterResults results) {
        clear();
        addAll((List)results.values);
        notifyDataSetChanged();
    }

    @Override
    public CharSequence convertResultToString(Object resultValue) {

```

```
        return ((PlaceItem) resultValue).getPlaceName();
    }
};

}
```

DisplaySQLiteDataActivity.java

```
package com.example.wanderlust;

import android.content.ClipData;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ListView;

import com.google.android.material.floatingactionbutton.FloatingActionButton;

import java.util.ArrayList;

import androidx.appcompat.app.AppCompatActivity;

public class DisplaySQLiteDataActivity extends AppCompatActivity {

    QLiteHelper qLiteHelper;
    SQLiteDatabase sqLiteDatabase;
    Cursor cursor;
    ListAdapter listAdapter ;
    ListView LISTVIEW;
    FloatingActionButton floatingActionButton;
    ArrayList<String> ID_Array;
    ArrayList<String> NAME_Array;
    ArrayList<String> PHONE_NUMBER_Array;
    ArrayList<String> ListViewClickItemArray = new ArrayList<String>();

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

```

setContentView(R.layout.activity_display_sqlite_data);

LISTVIEW = (ListView) findViewById(R.id.listView1);
floatingActionButton = (FloatingActionButton)
findViewById(R.id.add_trip);
ID_Array = new ArrayList<String>();
NAME_Array = new ArrayList<String>();
PHONE_NUMBER_Array = new ArrayList<String>();
qLiteHelper = new QLiteHelper(this);

LISTVIEW.setOnItemClickListener(new
AdapterView.OnItemClickListener()
{
    @Override
    public void onItemClick(AdapterView<?> parent, View view, int
position, long id) {
        ClipData.Item item =
(ClipData.Item)listAdapter.getItem(position);
        Intent intent = new Intent(getApplicationContext(),
ShowSingleRecordActivity.class);
        intent.putExtra("ListViewClickedItemValue",
ListViewClickItemArray.get(position).toString());
        startActivity(intent);
    }
});

floatingActionButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(DisplaySQLiteDataActivity.this,
TripActivity.class);
        startActivity(intent);
    }
});

@Override
protected void onResume() {
    ShowSQLiteDBdata() ;
    super.onResume();
}

```

```

private void ShowSQLiteDBdata() {
    sqLiteDatabase = qLiteHelper.getWritableDatabase();
    cursor = sqLiteDatabase.rawQuery("SELECT * FROM " +
QLiteHelper.TABLE_NAME+"", null);
    ID_Array.clear();
    NAME_Array.clear();
    PHONE_NUMBER_Array.clear();

    if (cursor.moveToFirst()) {
        do {

ID_Array.add(cursor.getString(cursor.getColumnIndex(QLiteHelper.Table_Colum
n_ID)));
ListVieClickItemArray.add(cursor.getString(cursor.getColumnIndex(QLiteHelp
er.Table_Column_ID)));

NAME_Array.add(cursor.getString(cursor.getColumnIndex(QLiteHelper.Table_Col
umn_1_Name)));

PHONE_NUMBER_Array.add(cursor.getString(cursor.getColumnIndex(QLiteHelper.T
able_Column_2_PhoneNumber)));
        } while (cursor.moveToNext());
    }
    listAdapter = new ListAdapter(DisplaySQLiteDataActivity.this,
        ID_Array,
        NAME_Array,
        PHONE_NUMBER_Array
    );
    LISTVIEW.setAdapter(listAdapter);
    cursor.close();
}
}

```

EditSingleRecordActivity.java

```
package com.example.wanderlust;

import android.app.DatePickerDialog;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Toast;

import java.util.Calendar;

import androidx.appcompat.app.AppCompatActivity;

public class EditSingleRecordActivity extends AppCompatActivity {

    EditText name, phone_number;
    Button update;
    SQLiteDatabase sqLiteDatabase;
    QLITEHelper qLiteHelper;
    Cursor cursor;
    String IDholder;
    String SQLiteDataBaseQueryHolder ;
    SQLiteDatabase sqLiteDatabaseObj;
    Calendar c;
    DatePickerDialog dpd;
    ImageView date;
    ImageView location;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_edit_single_record);

        name = (EditText) findViewById(R.id.EditTextName);
```

```

phone_number = (EditText) findViewById(R.id.editText3);
update = (Button) findViewById(R.id.buttonUpdate);
date = (ImageView) findViewById(R.id.date);
location = (ImageView) findViewById(R.id.location);
qLiteHelper = new QLiteHelper(this);

date.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        c = Calendar.getInstance();
        int day = c.get(Calendar.DAY_OF_MONTH);
        int month = c.get(Calendar.MONTH);
        int year = c.get(Calendar.YEAR);
        dpd = new DatePickerDialog(EditSingleRecordActivity.this,
new DatePickerDialog.OnDateSetListener() {
        @Override
        public void onDateSet(DatePicker datePicker, int mYear,
int mMonth, int mDay)
        {
            name.setText(mDay + "/" + (mMonth+1) + "/" + mYear);
        }
},day,month,year);
dpd.show();
}
});
update.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {

        String GetName = name.getText().toString();
        String GetPhoneNumber = phone_number.getText().toString();

        OpenSQLiteDataBase();

        SQLiteDataBaseQueryHolder = "UPDATE " +
QLiteHelper.TABLE_NAME + " SET "+ QLiteHelper.Table_Column_1_Name+" =
"+GetName+" , "+ QLiteHelper.Table_Column_2_PhoneNumber+" =
"+GetPhoneNumber+" WHERE id = " + IDholder + "";

        sqLiteDatabaseObj.execSQL(SQLiteDataBaseQueryHolder);

        sqLiteDatabase.close();
    }
});

```



```
public void OpenSQLiteDataBase(){

    sqLiteDatabaseObj = openOrCreateDatabase(QLiteHelper.DATABASE_NAME,
Context.MODE_PRIVATE, null);

}

}
```

ListAdapter.java

```
package com.example.wanderlust;

import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.BaseAdapter;
import android.widget.ImageView;
import android.widget.TextView;

import java.util.ArrayList;

public class ListAdapter extends BaseAdapter {

    Context context;
    ArrayList<String> ID;
    ArrayList<String> Name;
    ArrayList<String> PhoneNumber;

    public ListAdapter(
        Context context2,
        ArrayList<String> id,
        ArrayList<String> name,
        ArrayList<String> phone
    ) {

        this.context = context2;
        this.ID = id;
    }
}
```

```

        this.Name = name;
        this.PhoneNumber = phone;
    }

    public int getCount() {
        // TODO Auto-generated method stub
        return ID.size();
    }

    public Object getItem(int position) {
        // TODO Auto-generated method stub
        return null;
    }

    public long getItemId(int position) {
        // TODO Auto-generated method stub
        return 0;
    }

    public View getView(int position, View child, ViewGroup parent) {
        Holder holder;
        LayoutInflator layoutInflater;

        if (child == null) {
            layoutInflater = (LayoutInflator)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
            child = layoutInflater.inflate(R.layout.items, null);

            holder = new Holder();
            holder.Name_TextView = (TextView)
child.findViewById(R.id.textViewNAME);
            holder.PhoneNumberTextView = (TextView)
child.findViewById(R.id.textViewPHONE_NUMBER);
            holder.PlaceImage = (ImageView)
child.findViewById(R.id.list_image);
            child.setTag(holder);
        }

        holder = (Holder) child.getTag();
    }

    if(PhoneNumber.get(position).equals("Delhi")) {
        holder.PlaceImage.setImageResource(R.drawable.delhi);
    }
}

```

```

    }
    else if(PhoneNumber.get(position).equals("Ahmedabad")){
        holder.PlaceImage.setImageResource(R.drawable.ahmedabad);
    }
    else if(PhoneNumber.get(position).equals("Bangalore")){
        holder.PlaceImage.setImageResource(R.drawable.blr_slide_1);
    }
    else if(PhoneNumber.get(position).equals("Chennai")){
        holder.PlaceImage.setImageResource(R.drawable.chennai);
    }
    else if(PhoneNumber.get(position).equals("Hyderabad")){
        holder.PlaceImage.setImageResource(R.drawable.hydrabad);
    }
    else if(PhoneNumber.get(position).equals("Jaipur")){
        holder.PlaceImage.setImageResource(R.drawable.jaipur);
    }
    else if(PhoneNumber.get(position).equals("Kolkata")){
        holder.PlaceImage.setImageResource(R.drawable.kolkata);
    }
    else if(PhoneNumber.get(position).equals("Mumbai")){
        holder.PlaceImage.setImageResource(R.drawable.mumbai);
    }
    else if(PhoneNumber.get(position).equals("Panjim")){
        holder.PlaceImage.setImageResource(R.drawable.panaji);
    }
    else if(PhoneNumber.get(position).equals("Pune")){
        holder.PlaceImage.setImageResource(R.drawable.pune);
    }
    holder.Name_TextView.setText(Name.get(position));
    holder.PhoneNumberTextView.setText(PhoneNumber.get(position));

    return child;
}

public class Holder {
    TextView Name_TextView;
    TextView PhoneNumberTextView;
    ImageView PlaceImage;
}
}

```

PlaceItem.java

```
package com.example.wanderlust;

public class PlaceItem {

    private String placeName;
    private int placeImage;

    public PlaceItem(String placeName, int placeImage){
        this.placeName = placeName;
        this.placeImage = placeImage;
    }

    public String getPlaceName(){
        return placeName;
    }

    public int getPlaceImage(){
        return placeImage;
    }
}
```

QLiteHelper.java

```
package com.example.wanderlust;

import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class QLiteHelper extends SQLiteOpenHelper {

    static String DATABASE_NAME="AndroidJSonDataBase";

    public static final String TABLE_NAME="AndroidJSonTable";

    public static final String Table_Column_ID="id";

    public static final String Table_Column_1_Name="name";

    public static final String Table_Column_2_PhoneNumber="phone_number";

    public QLiteHelper(Context context) {
```

```
        super(context, DATABASE_NAME, null, 1);

    }

    @Override
    public void onCreate(SQLiteDatabase database) {

        String CREATE_TABLE="CREATE TABLE IF NOT EXISTS "+TABLE_NAME+
        ("+Table_Column_ID+" INTEGER PRIMARY KEY, "+Table_Column_1_Name+" VARCHAR,
        "+Table_Column_2_PhoneNumber+" VARCHAR)";
        database.execSQL(CREATE_TABLE);

    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion)
    {
        db.execSQL("DROP TABLE IF EXISTS "+TABLE_NAME);
        onCreate(db);
    }
}
```

ShowSingleRecordActivity.java

```
package com.example.wanderlust;

import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageButton;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
```

```

public class ShowSingleRecordActivity extends AppCompatActivity {

    String IDholder;
    TextView id, name, phone_number;
    SQLiteDatabase sqLiteDatabase;
    QLiteHelper qLiteHelper;
    Cursor cursor;
    Button Delete, Edit;
    SQLiteDatabase sqLiteDatabaseObj;
    String SQLiteDataBaseQueryHolder ;
    ImageView img_place;
    ImageButton share;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_show_single_record);

        id = (TextView) findViewById(R.id.textViewID);
        name = (TextView) findViewById(R.id.textViewName);
        phone_number = (TextView) findViewById(R.id.textViewPhoneNumber);
        Delete = (Button) findViewById(R.id.buttonDelete);
        Edit = (Button) findViewById(R.id.buttonEdit);
        img_place = (ImageView) findViewById(R.id.img_place);
        share = (ImageButton) findViewById(R.id.share);
        qLiteHelper = new QLiteHelper(this);
        Delete.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                AlertDialog.Builder alertDialogBuilder = new
                AlertDialog.Builder(ShowSingleRecordActivity.this);
                alertDialogBuilder.setMessage("You sure, that you want to
cancel");
                alertDialogBuilder.setPositiveButton("Yes", new
                DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface arg0, int arg1)
{
                        OpenSQLiteDataBase();
                        SQLiteDataBaseQueryHolder = "DELETE FROM "+
QLiteHelper.TABLE_NAME+" WHERE id = "+IDholder+"";

```

```

sqLiteDatabase.execSQL(SQLiteDataBaseQueryHolder);
        sqLiteDatabase.close();
        finish();
    }
});
AlertDialogBuilder alertDialogBuilder = new AlertDialog.Builder(this);
alertDialogBuilder.setNegativeButton("No", new
DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialog, int which) {
        finish();
    }
});
AlertDialog alertDialog = alertDialogBuilder.create();
alertDialog.show();
}
});

share.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent sendIntent = new Intent();
        sendIntent.setAction(Intent.ACTION_SEND);
        sendIntent.putExtra(Intent.EXTRA_TEXT, "Planning a trip to
"+phone_number.getText().toString()+" on "+name.getText().toString()+
"Would you like to join?");
        sendIntent.setType("text/plain");
        Intent shareIntent = Intent.createChooser(sendIntent, null);
        startActivity(shareIntent);
    }
});
Edit.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent(getApplicationContext(),
EditSingleRecordActivity.class);
        intent.putExtra("EditID", IDholder);
        startActivity(intent);
    }
});
}

@Override

```

```

protected void onResume() {
    ShowSingleRecordInTextView();
    super.onResume();
}

public void ShowSingleRecordInTextView() {
    SQLiteDatabase = qLiteHelper.getReadableDatabase();
    IDholder = getIntent().getStringExtra("ListViewClickedItemValue");
    cursor = SQLiteDatabase.rawQuery("SELECT * FROM " +
    QLiteHelper.TABLE_NAME + " WHERE id = " + IDholder + "", null);
    if (cursor.moveToFirst()) {
        do {

id.setText(cursor.getString(cursor.getColumnIndex(QLiteHelper.Table_Column_ID)));
name.setText(cursor.getString(cursor.getColumnIndex(QLiteHelper.Table_Column_1_Name)));

phone_number.setText(cursor.getString(cursor.getColumnIndex(QLiteHelper.Table_Column_2_PhoneNumber)));
        setImage();
    }
    while (cursor.moveToNext());
    cursor.close();
}
}

public void setImage(){
    if(phone_number.getText().toString().equals("Delhi")){
        img_place.setImageResource(R.drawable.delhi);
    }
    else if(phone_number.getText().toString().equals("Ahmedabad")){
        img_place.setImageResource(R.drawable.ahmedabad);
    }
    else if(phone_number.getText().toString().equals("Bangalore")){
        img_place.setImageResource(R.drawable.blr_slide_1);
    }
    else if(phone_number.getText().toString().equals("Chennai")){
        img_place.setImageResource(R.drawable.chennai);
    }
    else if(phone_number.getText().toString().equals("Hyderabad")){

```

```

        img_place.setImageResource(R.drawable.hydrabad);
    }
    else if(phone_number.getText().toString().equals("Jaipur")){
        img_place.setImageResource(R.drawable.jaipur);
    }
    else if(phone_number.getText().toString().equals("Kolkata")){
        img_place.setImageResource(R.drawable.kolkata);
    }
    else if(phone_number.getText().toString().equals("Mumbai")){
        img_place.setImageResource(R.drawable.mumbai);
    }
    else if(phone_number.getText().toString().equals("Panjim")){
        img_place.setImageResource(R.drawable.panaji);
    }
    else if(phone_number.getText().toString().equals("Pune")){
        img_place.setImageResource(R.drawable.pune);
    }
}

public void OpenSQLiteDataBase(){
    SQLiteDatabaseObj = openOrCreateDatabase(QLiteHelper.DATABASE_NAME,
Context.MODE_PRIVATE, null);
}
}

```

TripActivity.java

```

package com.example.wanderlust;

import android.app.DatePickerDialog;
import android.content.Context;
import android.content.Intent;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.AutoCompleteTextView;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Toast;

```

```
import java.util.ArrayList;
import java.util.Calendar;
import java.util.List;

import androidx.appcompat.app.AppCompatActivity;

public class TripActivity extends AppCompatActivity {

    SQLiteDatabase sqLiteDatabaseObj;
    EditText editTextName;
    AutoCompleteTextView editTextPhoneNumber;
    String NameHolder, NumberHolder, SQLiteDataBaseQueryHolder;
    Button EnterData;
    Boolean EditTextEmptyHold;
    ImageView date;
    ImageView location;
    Calendar c;
    DatePickerDialog dpd;
    private List<PlaceItem> placeList;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_trip);

        fillPlaceList();

        AutoCompleteTextView editText = findViewById(R.id.actv);
        AutoCompletePlaceAdapter adapter = new
        AutoCompletePlaceAdapter(this, placeList);
        editText.setAdapter(adapter);

        EnterData = (Button)findViewById(R.id.button);
        editTextName = (EditText)findViewById(R.id.editText);
        editTextPhoneNumber = (AutoCompleteTextView)
        findViewById(R.id.actv);
        date = (ImageView)findViewById(R.id.date);
        location = (ImageView)findViewById(R.id.location);

        date.setOnClickListener(new View.OnClickListener() {
            @Override
```

```

        public void onClick(View view) {
            c = Calendar.getInstance();
            int day = c.get(Calendar.DAY_OF_MONTH);
            int month = c.get(Calendar.MONTH);
            int year = c.get(Calendar.YEAR);
            dpd = new DatePickerDialog(TripActivity.this, new
DatePickerDialog.OnDateSetListener() {
                @Override
                public void onDateSet(DatePicker datePicker, int mYear,
int mMonth, int mDay)
                {
                    editTextName.setText(mDay + "/" + (mMonth+1) + "/" +
mYear);
                }
            },day,month,year);
            dpd.show();
        }
    });

EnterData.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        SQLiteDataBaseBuild();
        SQLiteTableBuild();
        CheckEditTextStatus();
        InsertDataIntoSQLiteDatabase();
        EmptyEditTextAfterDataInsert();
        Intent intent = new Intent(TripActivity.this,
DisplaySQLiteDataActivity.class);
        startActivity(intent);

    }
});

private void fillPlaceList(){
    placeList = new ArrayList<>();
    placeList.add(new PlaceItem("Delhi", R.drawable.delhi));
    placeList.add(new PlaceItem("Bangalore", R.drawable.blr_slide_1));
    placeList.add(new PlaceItem("Chennai", R.drawable.chennai));
    placeList.add(new PlaceItem("Hyderabad", R.drawable.hydrabad));
}

```

```

placeList.add(new PlaceItem("Ahmedabad", R.drawable.ahmedabad));
placeList.add(new PlaceItem("Kolkata", R.drawable.kolkata));
placeList.add(new PlaceItem("Jaipur", R.drawable.jaipur));
placeList.add(new PlaceItem("Mumbai", R.drawable.mumbai));
placeList.add(new PlaceItem("Pune", R.drawable.pune));
placeList.add(new PlaceItem("Panjim", R.drawable.panaji));

}

public void SQLiteDataBaseBuild(){
    sqLiteDatabaseObj = openOrCreateDatabase(QLiteHelper.DATABASE_NAME,
Context.MODE_PRIVATE, null);
}

public void SQLiteTableBuild(){
    String CREATE_TABLE="CREATE TABLE IF NOT EXISTS "+
QLiteHelper.TABLE_NAME+" ("+ QLiteHelper.Table_Column_ID+" INTEGER PRIMARY
KEY, "+ QLiteHelper.Table_Column_1_Name+" VARCHAR, "+
QLiteHelper.Table_Column_2_PhoneNumber+" VARCHAR)";
    sqLiteDatabaseObj.execSQL(CREATE_TABLE);
}

public void CheckEditTextStatus(){
    NameHolder = editTextName.getText().toString() ;
    NumberHolder = editTextPhoneNumber.getText().toString();

    if(TextUtils.isEmpty(NameHolder) ||
TextUtils.isEmpty(NumberHolder)){
        EditTextEmptyHold = false ;
    }
    else {
        EditTextEmptyHold = true ;
    }
}

public void InsertDataIntoSQLiteDatabase(){

    if(EditTextEmptyHold == true)
    {
        SQLiteDataBaseQueryHolder = "INSERT INTO "+
QLiteHelper.TABLE_NAME+" (name,phone_number) VALUES('"++NameHolder+"',
 '"++NumberHolder+"');";
        sqLiteDatabaseObj.execSQL(SQLiteDataBaseQueryHolder);
    }
}

```

```

        sqLiteDatabaseObj.close();
        Toast.makeText(TripActivity.this,"Data Inserted Successfully",
Toast.LENGTH_LONG).show();
    }
    else {
        Toast.makeText(TripActivity.this,"Please Fill All The Required
Fields.", Toast.LENGTH_LONG).show();
    }
}
public void EmptyEditTextAfterDataInsert(){
    editTextName.getText().clear();
    editTextPhoneNumber.getText().clear();
}
}

```

XML Files

activity_display_sqlite_data.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/activity_display_sqlite_data"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="16dp"
    android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    android:background="#ffffffff">

    <ListView
        android:id="@+id/listView1"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_alignParentStart="true"
        android:divider="#ffffffff">
    </ListView>

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/add_trip"

```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_alignParentEnd="true"
        android:layout_margin="16dp"
        android:src="@drawable/ic_add_circle_white_24dp" />
    </RelativeLayout>
```

activity_edit_single_record.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_edit_single_record"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_gravity="center"
    android:gravity="center"
    android:orientation="vertical"
    android:paddingBottom="16dp"
    android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp">

    <LinearLayout
        android:orientation="horizontal"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <ImageView
            android:id="@+id/date"
            android:layout_width="wrap_content"
            android:layout_height="match_parent"
            android:src="@drawable/ic_event"
            android:layout_weight="0"/>
        <EditText
            android:id="@+id/EditTextName"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_margin="10dp"
            android:layout_weight="1"
```

```
        android:textSize="18sp"
        android:layout_marginTop="10dp"
        android:hint="Select Trip Date"
        android:textColor="?attr/editTextColor"
        android:background="?attr/editTextBackground" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <ImageView
            android:id="@+id/location"
            android:layout_width="wrap_content"
            android:layout_height="match_parent"
            android:src="@drawable/ic_location"
            android:layout_weight="0"/>
        <EditText
            android:layout_weight="1"
            android:id="@+id/editText3"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_margin="10dp"
            android:textSize="18sp"
            android:layout_marginTop="10dp"
            android:hint="Select City"
            android:textColor="?attr/editTextColor"
            android:background="?attr/editTextBackground" />
    </LinearLayout>

    <Button
        android:id="@+id/buttonUpdate"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="10dp"
        android:layout_marginTop="10dp"
        android:background="@color/colorPrimary"
        android:text="Update Trip"
        android:textColor="#ffffff" />

```

activity_show_single_record.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:id="@+id/activity_show_single_record"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <ImageView
            android:id="@+id/img_place"
            android:layout_width="match_parent"
            android:layout_height="230dp"
            android:scaleType="centerCrop"
            android:background="#000000"/>

        <ImageButton
            android:id="@+id/share"
            android:paddingRight="16dp"
            android:paddingBottom="16dp"
            android:paddingLeft="16dp"
            android:paddingTop="16dp"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="right"
            android:background="@drawable/share" />
    </FrameLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:paddingLeft="16dp"
        android:paddingTop="16dp">
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Trip no: "
    android:textColor="@color/saffron"
    android:textStyle="bold"/>

<TextView
    android:text="Id Show Here"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/textViewID"
    android:textColor="@color/saffron"
    android:textSize="13dp"
    android:textStyle="bold"
    android:layout_alignParentTop="true" />
</LinearLayout>

<TextView
    android:text="Name Show Here"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/textViewPhoneNumber"
    android:textColor="#040404"
    android:typeface="sans"
    android:textSize="25dp"
    android:textStyle="bold"
    android:paddingLeft="16dp"/>

<TextView
    android:text="Phone Number Here"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/textViewName"
    android:textColor="#343434"
    android:textSize="12dp"
    android:textStyle="italic"
    android:layout_marginTop="1dp"
    android:paddingLeft="16dp"
    android:paddingBottom="16dp"/>

<Button
```

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:layout_marginTop="10dp"
    android:background="@color/colorPrimary"
    android:text="Update Trip"
    android:textColor="#ffffffff"
    android:id="@+id/buttonEdit" />

<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:layout_marginTop="10dp"
    android:background="@color/colorPrimary"
    android:text="Cancel Trip"
    android:textColor="#ffffffff"
    android:id="@+id/buttonDelete" />

</LinearLayout>
```

activity_trip.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:paddingBottom="16dp"
    android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp">

    <LinearLayout
        android:orientation="horizontal"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <ImageView
            android:id="@+id/date"
            android:layout_width="wrap_content"
```

```
        android:layout_height="match_parent"
        android:src="@drawable/ic_event"
        android:layout_weight="0"/>
    <EditText
        android:id="@+id/editText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="10dp"
        android:layout_weight="1"
        android:textSize="18sp"
        android:layout_marginTop="10dp"
        android:hint="Select Trip Date"
        android:textColor="?attr/editTextColor"
        android:background="?attr/editTextBackground" />

    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <ImageView
            android:id="@+id/location"
            android:layout_width="wrap_content"
            android:layout_height="match_parent"
            android:src="@drawable/ic_location"
            android:layout_weight="0"/>

        <AutoCompleteTextView
            android:id="@+id/actv"
            android:layout_height="wrap_content"
            android:layout_width="match_parent"
            android:completionThreshold="0"
            android:layout_margin="10dp"
            android:textSize="18sp"
            android:layout_marginTop="10dp"
            android:hint="Enter a place"/>
    </LinearLayout>

    <Button
        android:id="@+id/button"
```

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:layout_marginTop="10dp"
    android:background="@color/colorPrimary"
    android:text="Add Trip"
    android:textColor="#ffffffff" />

</LinearLayout>
```

Items.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:card_view="http://schemas.android.com/tools"
    android:id="@+id/trip_linear_layout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#ffffffff"
    android:orientation="horizontal"
    android:layout_marginLeft="10dp">

    <com.github.vipulasri.timelineview.TimelineView
        android:id="@+id/timeline_view"
        app:marker="@drawable/marker"
        android:layout_width="18dp"
        android:layout_height="match_parent"
        app:line="@color/colorPrimary"
        app:lineSize="2dp"
        app:markerInCenter="false"
        app:markerSize="20dp" />

    <androidx.cardview.widget.CardView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:layout_margin="5dp"
        card_view:contentPadding="100dp"
        card_view:cardCornerRadius="2dp">

        <RelativeLayout
```

```
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="5dip" >

    <LinearLayout android:id="@+id/thumbnail"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:padding="3dip"
        android:layout_alignParentLeft="true"
        android:background="@drawable/image_bg"
        android:layout_marginRight="5dip">

        <ImageView
            android:id="@+id/list_image"
            android:layout_width="60dp"
            android:layout_height="40dp"/>

    </LinearLayout>

    <TextView
        android:id="@+id/textViewPHONE_NUMBER"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignTop="@+id/thumbnail"
        android:layout_toRightOf="@+id/thumbnail"
        android:textColor="#040404"
        android:typeface="sans"
        android:text="Chennai"
        android:textSize="15dip"
        android:textStyle="bold"/>

    <TextView
        android:id="@+id/textViewNAME"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/textViewPHONE_NUMBER"
        android:textColor="#343434"
        android:textSize="10dip"
        android:layout_marginTop="1dip"
        android:layout_toRightOf="@+id/thumbnail"
```

```
        android:text="12/02/2021" />

    <!-- Rightend Arrow -->
    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/arrow"
        android:layout_alignParentRight="true"
        android:layout_centerVertical="true"/>

</RelativeLayout>
</androidx.cardview.widget.CardView>
</LinearLayout>
```

place_autocomplete_row.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:padding="4dp">

    <ImageView
        android:id="@+id/image_view_place"
        android:layout_width="60dp"
        android:layout_height="40dp"
        tools:src="@drawable/delhi" />

    <TextView
        android:id="@+id/text_view_name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:paddingStart="8dp"
        android:textAppearance="?android:attr/textAppearanceListItemSmall"
        tools:text="Delhi" />

</LinearLayout>
```

7.1.7. UTILITY

JavaFiles

UtilityActivity.java

```
package com.example.wanderlust;

import android.content.Intent;
import android.graphics.Color;
import android.graphics.drawable.ColorDrawable;
import android.os.Bundle;
import android.view.View;
import android.widget.ImageButton;

import androidx.appcompat.app.ActionBar;
import androidx.appcompat.app.AppCompatActivity;

public class UtilityActivity extends AppCompatActivity {

    ImageButton checklist;
    ImageButton compass;
    ImageButton weather;
    ImageButton world_clock;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_utility);

        ActionBar actionBar;
        actionBar = getSupportActionBar();
        ColorDrawable colorDrawable
                = new ColorDrawable(Color.parseColor("#018db7"));
        actionBar.setBackgroundDrawable(colorDrawable);

        checklist = (ImageButton) findViewById(R.id.checklist);
        checklist.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new
```

```
Intent(getApplicationContext(),ChecklistActivity.class);
        startActivity(intent);
    }
});

compass = (ImageButton)findViewById(R.id.compass);
compass.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new
Intent(getApplicationContext(),CompassActivity.class);
        startActivity(intent);
    }
});

weather = (ImageButton)findViewById(R.id.weather);
weather.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new
Intent(getApplicationContext(),WeatherActivity.class);
        startActivity(intent);
    }
});

world_clock = (ImageButton)findViewById(R.id.clock);
world_clock.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new
Intent(getApplicationContext(),WorldClockActivity.class);
        startActivity(intent);
    }
});
}
}
```

XML Files

activity_utility.xml

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:fitsSystemWindows="true">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="16dp"
        android:layout_marginRight="16dp"
        android:layout_marginBottom="16dp"
        android:layout_marginTop="16dp"
        android:orientation="vertical">

        <FrameLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="center_vertical"
            android:orientation="vertical">

            <ImageButton
                android:id="@+id/checklist"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:background="@drawable/checklist"/>

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_gravity="center"
                android:text="CHECK LIST"
                android:textColor="#FFFFFF"
                android:textSize="20dp"
                android:textStyle="bold" />
        </FrameLayout>
    
```

```
<FrameLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center_vertical"
    android:layout_marginTop="16dp"
    android:orientation="vertical">

    <ImageButton
        android:id="@+id/compass"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@drawable/compass"/>

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="COMPASS"
        android:textSize="20dp"
        android:textColor="#FFFFFF"
        android:textStyle="bold" />
</FrameLayout>

<FrameLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center_vertical"
    android:layout_marginTop="16dp"
    android:orientation="vertical">

    <ImageButton
        android:id="@+id/weather"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@drawable/weather"/>

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="WEATHER FORECAST"
        android:textSize="20dp"
        android:textColor="#FFFFFF"
        android:textStyle="bold" />
</FrameLayout>
```

```

        android:textSize="20dp"
        android:textColor="#FFFFFF"
        android:textStyle="bold" />
    </FrameLayout>

    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:layout_marginTop="16dp"
        android:orientation="vertical">

        <ImageButton
            android:id="@+id/clock"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@drawable/clock"/>

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:text="WORLD CLOCK"
            android:textSize="20dp"
            android:textColor="#FFFFFF"
            android:textStyle="bold" />
    </FrameLayout>
</LinearLayout>
</ScrollView>
```

7.1.7.1. CHECKLIST

JavaFiles

ChecklistActivity.java

```

package com.example.wanderlust;

import android.graphics.Color;
import android.graphics.drawable.ColorDrawable;
```

```
import android.os.Bundle;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.EditText;
import android.widget.ListView;

import com.google.android.material.floatingactionbutton.FloatingActionButton;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;

import androidx.appcompat.app.ActionBar;
import androidx.appcompat.app.AppCompatActivity;

public class ChecklistActivity extends AppCompatActivity {

    EditText edt_item;
    ListView itemlist;
    String[] ListElements = new String[] {
    };

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_checklist);

        ActionBar actionBar;
        actionBar = getSupportActionBar();
        ColorDrawable colorDrawable
                = new ColorDrawable(Color.parseColor("#018db7"));
        actionBar.setBackgroundDrawable(colorDrawable);

        edt_item = (EditText)findViewById(R.id.edt_item);
        itemlist = (ListView)findViewById(R.id.itemlist);

        final List<String> ListElementsArrayList = new
        ArrayList<String>(Arrays.asList(ListElements));
        final ArrayAdapter<String> adapter = new
        ArrayAdapter<String>(ChecklistActivity.this,
        android.R.layout.simple_list_item_1, ListElementsArrayList);
```

```
itemlist.setAdapter(adapter);

FloatingActionButton floatingActionButton =
        (FloatingActionButton)
findViewById(R.id.floating_action_button);
floatingActionButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {

        ListElementsArrayList.add(edt_item.getText().toString());
        adapter.notifyDataSetChanged();
        edt_item.setText("");
    }
});
}
}
```

XML Files

activity_checklist.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:orientation="vertical"
    android:fitsSystemWindows="true">

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:layout_marginLeft="16dp"
        android:orientation="vertical">

        <CheckBox
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textColor="#000000"
        android:textSize="15dp"
        android:text="Bags"/>

<CheckBox
    android:layout_marginTop="8dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textColor="#000000"
    android:textSize="15dp"
    android:text="Keys" />

<CheckBox
    android:layout_marginTop="8dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textColor="#000000"
    android:textSize="15dp"
    android:text="Charger" />

<CheckBox
    android:layout_marginTop="8dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textColor="#000000"
    android:textSize="15dp"
    android:text="Earphones" />

<CheckBox
    android:layout_marginTop="8dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textColor="#000000"
    android:textSize="15dp"
    android:text="Keys" />

<CheckBox
    android:layout_marginTop="8dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
```

```
        android:textColor="#000000"
        android:textSize="15dp"
        android:text="Food" />

<CheckBox
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textColor="#000000"
    android:textSize="15dp"
    android:text="Ticket" />

<CheckBox
    android:layout_marginTop="8dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textColor="#000000"
    android:textSize="15dp"
    android:text="Power Bank" />

<ListView
    android:layout_marginLeft="10dp"
    android:id="@+id/itemlist"
    android:layout_width="match_parent"
    android:layout_height="138dp"
    android:layout_marginTop="8dp"
    tools:layout_editor_absoluteX="4dp"
    tools:layout_editor_absoluteY="8dp" />

</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="25dp">

    <EditText
        android:id="@+id/edt_item"
        android:layout_width="307dp"
        android:layout_height="wrap_content"
        android:hint="Enter new item"
        android:textColor="#000000" />

```

```
<com.google.android.material.floatingactionbutton.FloatingActionButton
    android:id="@+id/floating_action_button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:srcCompat="@android:drawable/ic_input_add" />

</LinearLayout>
</LinearLayout>
```

7.1.7.2. COMPASS

JavaFiles

CompassActivity.java

```
package com.example.wanderlust;

import android.app.Activity;
import android.hardware.Sensor;
import android.hardware.SensorEvent;
import android.hardware.SensorEventListener;
import android.hardware.SensorManager;
import android.os.Bundle;
import android.view.animation.Animation;
import android.view.animation.RotateAnimation;
import android.widget.ImageView;
import android.widget.TextView;

public class CompassActivity extends Activity implements
SensorEventListener {

    private ImageView image;
    private float currentDegree = 0f;
    private SensorManager mSensorManager;

    TextView tvHeading;

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

```

        setContentView(R.layout.activity_compass);

        image = (ImageView) findViewById(R.id.main_iv);
        tvHeading = (TextView) findViewById(R.id.tvHeading);
        mSensorManager = (SensorManager) getSystemService(SENSOR_SERVICE);
    }

    @Override
    protected void onResume() {
        super.onResume();
        mSensorManager.registerListener(this,
            mSensorManager.getDefaultSensor(Sensor.TYPE_ORIENTATION),
            SensorManager.SENSOR_DELAY_GAME);
    }

    @Override
    protected void onPause() {
        super.onPause();
        mSensorManager.unregisterListener(this);
    }

    @Override
    public void onSensorChanged(SensorEvent event) {
        float degree = Math.round(event.values[0]);

        tvHeading.setText("Heading: " + Float.toString(degree) + " "
degrees);
        RotateAnimation ra = new RotateAnimation(
            currentDegree,
            -degree,
            Animation.RELATIVE_TO_SELF, 0.5f,
            Animation.RELATIVE_TO_SELF,
            0.5f);

        ra.setDuration(210);
        ra.setFillAfter(true);
        image.startAnimation(ra);
        currentDegree = -degree;
    }

    @Override

```

```
    public void onAccuracyChanged(Sensor sensor, int accuracy) {  
    }  
}
```

XML Files

activity_compass.xml

```
<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent">  
  
    <TextView  
        android:id="@+id/tvHeading"  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:layout_centerHorizontal="true"  
        android:layout_marginBottom="40dp"  
        android:layout_marginTop="20dp"  
        android:textColor="#000000"  
        android:text="Heading: 0.0" />  
  
    <ImageView  
        android:id="@+id/main_iv"  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:layout_below="@+id/tvHeading"  
        android:layout_centerHorizontal="true"  
        android:src="@drawable/compass_img" />  
  
</RelativeLayout>
```

7.1.7.3. WEATHER FORECAST

JavaFiles

WeatherActivity.java

```
package com.example.wanderlust;

import android.content.Intent;
import android.graphics.Typeface;
import android.os.Bundle;
import android.text.Html;
import android.view.View;
import android.widget.EditText;
import android.widget.ImageButton;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class WeatherActivity extends AppCompatActivity {
    TextView cityField, detailsField, currentTemperatureField,
humidity_field, pressure_field, weatherIcon, updatedField;
    EditText edt_place;
    Typeface weatherFont;
    ImageButton imgbtn_location;
    String place;
    String city;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_weather);
        Intent intent = getIntent();
        city=intent.getStringExtra("city");

        weatherFont =
Typeface.createFromAsset(getApplicationContext().getAssets(),
"fonts/weathericons-regular-webfont.ttf");

        cityField = (TextView)findViewById(R.id.city_field);
        updatedField = (TextView)findViewById(R.id.updated_field);
```

```

detailsField = (TextView)findViewById(R.id.details_field);
currentTemperatureField =
(TextView)findViewById(R.id.current_temperature_field);
humidity_field = (TextView)findViewById(R.id.humidity_field);
pressure_field = (TextView)findViewById(R.id.pressure_field);
edt_place = (EditText)findViewById(R.id.edt_place);
imgbtn_location = (ImageButton)findViewById(R.id.imgbtn_location);
weatherIcon = (TextView)findViewById(R.id.weather_icon);
weatherIcon.setTypeface(weatherFont);

final Function.placeIdTask asyncTask =new Function.placeIdTask(new
Function.AsyncResponse() {
    public void processFinish(String weather_city, String
weather_description, String weather_temperature, String weather_humidity,
String weather_pressure, String weather_updatedOn, String weather_iconText,
String sun_rise) {

        cityField.setText(weather_city);
        updatedField.setText(weather_updatedOn);
        detailsField.setText(weather_description);
        currentTemperatureField.setText(weather_temperature);
        humidity_field.setText("Humidity: "+weather_humidity);
        pressure_field.setText("Pressure: "+weather_pressure);
        weatherIcon.setText(Html.fromHtml(weather_iconText,1));

    }
});

Function.placeIdTask asynctask =new Function.placeIdTask(new
Function.AsyncResponse() {
    public void processFinish(String weather_city, String
weather_description, String weather_temperature, String weather_humidity,
String weather_pressure, String weather_updatedOn, String weather_iconText,
String sun_rise) {

        cityField.setText(weather_city);
        updatedField.setText(weather_updatedOn);
        detailsField.setText(weather_description);
        currentTemperatureField.setText(weather_temperature);
        humidity_field.setText("Humidity: "+weather_humidity);
        pressure_field.setText("Pressure: "+weather_pressure);
    }
});

```

```

        weatherIcon.setText(Html.fromHtml(weather_iconText,1));

    }

});;

asynctask.execute(city);
imgbtn_location.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        place = edt_place.getText().toString();
        asyncTask.execute(place);
    }
});
}
}
}

```

Function.java

```

package com.example.wanderlust;

import android.os.AsyncTask;
import android.util.Log;

import org.json.JSONException;
import org.json.JSONObject;

import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;
import java.text.DateFormat;
import java.util.Date;
import java.util.Locale;

public class Function {

    private static final String OPEN_WEATHER_MAP_URL =
    "http://api.openweathermap.org/data/2.5/weather?q=%s&units=metric";
    private static final String OPEN_WEATHER_MAP_API =
    "8798ffcb1e81c0e94157bab4f8dad937";
}

```

```

    public static String setWeatherIcon(int actualId, long sunrise, long
sunset) {
        int id = actualId / 100;
        String icon = "";
        if (actualId == 800) {
            long currentTime = new Date().getTime();
            if (currentTime >= sunrise && currentTime < sunset) {
                icon = "\uf00d";
            } else {
                icon = "\uf02e";
            }
        } else {
            switch (id) {
                case 2:
                    icon = "\uf01e";
                    break;
                case 3:
                    icon = "\uf01c";
                    break;
                case 7:
                    icon = "\uf014";
                    break;
                case 8:
                    icon = "\uf013";
                    break;
                case 6:
                    icon = "\uf01b";
                    break;
                case 5:
                    icon = "\uf019";
                    break;
            }
        }
        return icon;
    }

    public interface AsyncResponse {
        void processFinish(String output1, String output2, String output3,
String output4, String output5, String output6, String output7, String
output8);
    }
}

```

```

    public static class placeIdTask extends AsyncTask<String, Void,
JSONObject> {
        public AsyncResponse delegate = null;

        public placeIdTask(AsyncResponse asyncResponse) {
            delegate = asyncResponse;
        }

        @Override
        protected JSONObject doInBackground(String... params) {
            JSONObject jsonWeather = null;
            try {
                jsonWeather = getWeatherJSON(params[0]);
            } catch (Exception e) {
                Log.d("Error", "Cannot process JSON results", e);
            }
            return jsonWeather;
        }

        @Override
        protected void onPostExecute(JSONObject json) {
            try {
                if (json != null) {
                    JSONObject details =
json.getJSONArray("weather").getJSONObject(0);
                    JSONObject main = json.getJSONObject("main");
                    DateFormat df = DateFormat.getDateInstance();

                    String city =
json.getString("name").toUpperCase(Locale.US) + ", " +
json.getJSONObject("sys").getString("country");
                    String description =
details.getString("description").toUpperCase(Locale.US);
                    String temperature = String.format("%.2f",
main.getDouble("temp")) + "°";
                    String humidity = main.getString("humidity") + "%";
                    String pressure = main.getString("pressure") + " hPa";
                    String updatedOn = df.format(new Date(json.getLong("dt") *
1000));
                    String iconText = setWeatherIcon(details.getInt("id"),
json.getJSONObject("sys").getLong("sunrise") *

```

```

1000,
                json.getJSONObject("sys").getLong("sunset") *
1000);
            delegate.processFinish(city, description, temperature,
humidity, pressure, updatedOn, iconText, "" +
(json.getJSONObject("sys").getLong("sunrise") * 1000));
        }
    } catch (JSONException e) {
        }
    }
}

public static JSONObject getWeatherJSON(String city) {
    try {
        URL url = new URL(String.format(OPEN_WEATHER_MAP_URL, city));
        HttpURLConnection connection = (HttpURLConnection)
url.openConnection();
        connection.addRequestProperty("x-api-key",
OPEN_WEATHER_MAP_API);

        BufferedReader reader = new BufferedReader(new
InputStreamReader(connection.getInputStream()));

        StringBuffer json = new StringBuffer(1024);
        String tmp = "";
        while ((tmp = reader.readLine()) != null)
            json.append(tmp).append("\n");
        reader.close();

        JSONObject data = new JSONObject(json.toString());
        if (data.getInt("cod") != 200) {
            return null;
        }
        return data;
    } catch (Exception e) {
        return null;
    }
}
}

```

XML Files

activity_weather.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/bg_gradient"
    android:padding="20dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="16dp"
        android:layout_marginRight="16dp">

        <EditText
            android:id="@+id/edt_place"
            android:layout_width="308dp"
            android:layout_height="wrap_content"
            android:layout_alignParentTop="true"
            android:layout_centerHorizontal="true"
            android:hint="Enter place" />

        <ImageButton
            android:id="@+id/imgbtn_location"
            android:layout_width="wrap_content"
            android:layout_height="match_parent"
            android:background="@drawable/ic_action_place" />

    </LinearLayout>

    <TextView
        android:id="@+id/city_field"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="77dp"
```

```
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:textColor="#000000"
        android:textSize="24dp" />

<TextView
    android:id="@+id/updated_field"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/city_field"
    android:layout_centerHorizontal="true"
    android:textColor="#000000"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:textSize="14dp" />

<TextView
    android:id="@+id/weather_icon"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerVertical="true"
    android:layout_centerHorizontal="true"
    android:textColor="#FFFFFF"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:textSize="90sp"
    />

<TextView
    android:id="@+id/current_temperature_field"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:layout_gravity="center"
    android:layout_marginTop="179dp"
    android:fontFamily="sans-serif-thin"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:textColor="#000000"
    android:textSize="90dp" />

<TextView
    android:id="@+id/details_field"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
```

```
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:layout_marginBottom="280dp"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textColor="#FFFFFF" />

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="127dp"
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:layout_marginBottom="70dp"
        android:orientation="horizontal">

        <LinearLayout
            android:layout_width="153dp"
            android:layout_height="133dp"
            android:layout_alignParentBottom="true"
            android:layout_alignParentEnd="true"
            android:layout_marginBottom="58dp"
            android:layout_marginEnd="8dp"
            android:layout_weight="1"
            android:background="#3CF1EBF1"
            android:gravity="center"
            android:orientation="vertical"
            android:padding="8dp">

            <ImageView
                android:layout_width="25dp"
                android:layout_height="25dp"
                android:src="@drawable/humidity"
                android:tint="#FFFFFF" />

            <Space
                android:layout_width="wrap_content"
                android:layout_height="5dp" />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Humidity"
                android:textSize="12dp" />
    
```

```
<TextView
    android:id="@+id/humidity_field"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/details_field"
    android:layout_centerHorizontal="true"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:textColor="#FFFFFF"
    android:textSize="14dp" />
</LinearLayout>

<LinearLayout
    android:layout_width="153dp"
    android:layout_height="133dp"
    android:layout_alignParentBottom="true"
    android:layout_alignParentStart="true"
    android:layout_marginBottom="109dp"
    android:layout_marginStart="40dp"
    android:layout_weight="1"
    android:background="#3CF1EBF1"
    android:gravity="center"
    android:orientation="vertical"
    android:padding="8dp">

    <ImageView
        android:layout_width="25dp"
        android:layout_height="25dp"
        android:src="@drawable/pressure"
        android:tint="#FFFFFF" />

    <Space
        android:layout_width="wrap_content"
        android:layout_height="5dp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Pressure"
        android:textSize="12dp" />

    <TextView
```

```
        android:id="@+id/pressure_field"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textColor="#FFFFFF"
        android:textSize="14dp" />
    </LinearLayout>
</LinearLayout>
</RelativeLayout>
```

7.1.7.4. WORLD CLOCK

JavaFiles

WorldClockActivity.java

```
package com.example.wanderlust;

import android.graphics.Color;
import android.graphics.drawable.ColorDrawable;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Spinner;
import android.widget.TextView;

import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.TimeZone;

import androidx.appcompat.app.ActionBar;
import androidx.appcompat.app.AppCompatActivity;

public class WorldClockActivity extends AppCompatActivity {

    Calendar current;
    Spinner spinner;
```

```
TextView timezone, txtCurrentTime,txtTimeZoneTime;
long milliseconds;
ArrayAdapter<String> idAdapter;
SimpleDateFormat sdf;
Date resultDate;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_clock);

    ActionBar actionBar;
    actionBar = getSupportActionBar();
    ColorDrawable colorDrawable
        = new ColorDrawable(Color.parseColor("#018db7"));
    actionBar.setBackgroundDrawable(colorDrawable);

    spinner = (Spinner)findViewById(R.id.spinner);
    timezone = (TextView)findViewById(R.id.timezone);
    txtCurrentTime = (TextView)findViewById(R.id.txtCurrentTime);
    txtTimeZoneTime = (TextView)findViewById(R.id.txtTimeZoneTime);

    String[] idArray = TimeZone.getAvailableIDs();
    sdf = new SimpleDateFormat("EEEE, dd MMMM yyyy HH:mm:ss");

    idAdapter = new ArrayAdapter<String>(this,
    android.R.layout.simple_spinner_item,idArray);

    idAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_
item);
    spinner.setAdapter(idAdapter);
    getGMTtime();
    spinner.setOnItemSelectedListener(new
AdapterView.OnItemSelectedListener() {
        @Override
        public void onItemSelected(AdapterView<?> parent, View view, int
position, long l) {
            getGMTtime();
            String selectedID = (String)
(parent.getItemAtPosition(position));
```

```

        TimeZone timeZone = TimeZone.getTimeZone(selectedID);
        String TimeZoneName = timeZone.getDisplayName();

        int TimeZoneOffset = timeZone.getRawOffset() / (60 * 1000);
        int hrs = TimeZoneOffset / 60;
        int mins = TimeZoneOffset % 60;

        milliseconds = milliseconds + timeZone.getRawOffset();
        resultDate = new Date(milliseconds);
        System.out.println(sdf.format(resultDate));

        timezone.setText(TimeZoneName + " : GMT" + hrs + ":" +
mins);
        txtTimeZoneTime.setText(" " + sdf.format(resultDate));
        milliseconds = 0;
    }

    @Override
    public void onNothingSelected(AdapterView<?> adapterView) {

    }
});
```

}

```

private void getGMTtime(){
    current = Calendar.getInstance();
    txtCurrentTime.setText(" " + current.getTime());

    milliseconds = current.getTimeInMillis();

    TimeZone txtCurrent = current.getTimeZone();
    int offset = txtCurrent.getRawOffset();

    if(txtCurrent.inDaylightTime(new Date())){
        offset = offset + txtCurrent.getDSTSavings();
    }

    milliseconds = milliseconds - offset;
    resultDate = new Date(milliseconds);
    System.out.println(sdf.format(resultDate));
}
```

}

XML Files

activity_clock.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    android:orientation="vertical"
    android:paddingTop="16dp"
    android:paddingBottom="16dp"
    android:paddingRight="16dp"
    android:paddingLeft="16dp">

    <TextView
        android:id="@+id/textView"
        android:gravity="center"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:text="Current Time" />

    <TextView
        android:layout_marginTop="10dp"
        android:id="@+id/txtCurrentTime"
        android:gravity="center"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_below="@+id/textView"
        android:layout_marginStart="0dp"
        android:text="" />

    <Spinner
        android:id="@+id/spinner"
        android:layout_marginTop="10dp"
        android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"/>

    <TextView
        android:id="@+id/timezone"
        android:layout_marginTop="10dp"
        android:gravity="center"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:text="" />

    <TextView
        android:id="@+id/txtTimeZoneTime"
        android:layout_marginTop="10dp"
        android:gravity="center"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_below="@+id/textView"
        android:layout_marginStart="0dp"
        android:text="" />

</LinearLayout>
```

CHAPTER 8 - RESULT

The figure displays three screenshots of a mobile application interface for 'wanderLUST'. The screenshots are arranged in a grid-like fashion.

Screenshot 1 (Left): Login Screen

- Time: 19:38
- Header: wanderLUST
- Form:
 - Username:
 - Email:
 - Password:
 - Buttons: **LOGIN**, [I don't have account yet. *create one*](#)
- Bottom: Navigation icons (three horizontal bars, square, back arrow)

Screenshot 2 (Middle): Register Screen

- Time: 19:38
- Header: wanderLUST
- Form:
 - Username:
 - Email:
 - Password:
 - Buttons: **REGISTER**, [Back to Login](#)
- Bottom: Navigation icons (three horizontal bars, square, back arrow)

Screenshot 3 (Right): Register Screen with Error

- Time: 19:32
- Header: wanderLUST
- Form:
 - Username: Yamuna K
 - Email: yamuna.pixie@gmail.com
 - Password:
 - Buttons: **REGISTER**, [Back to Login](#)
- Message: User already exists with same email
- Bottom: Navigation icons (three horizontal bars, square, back arrow)

Screenshot 4 (Bottom Left): Register Screen

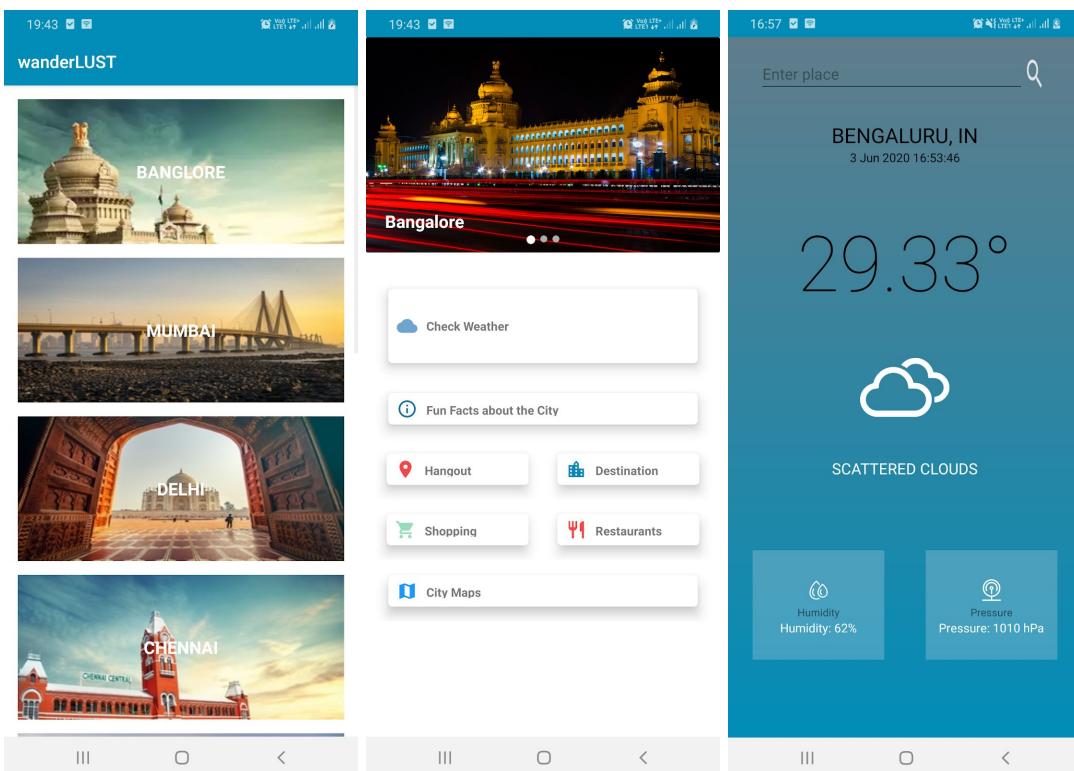
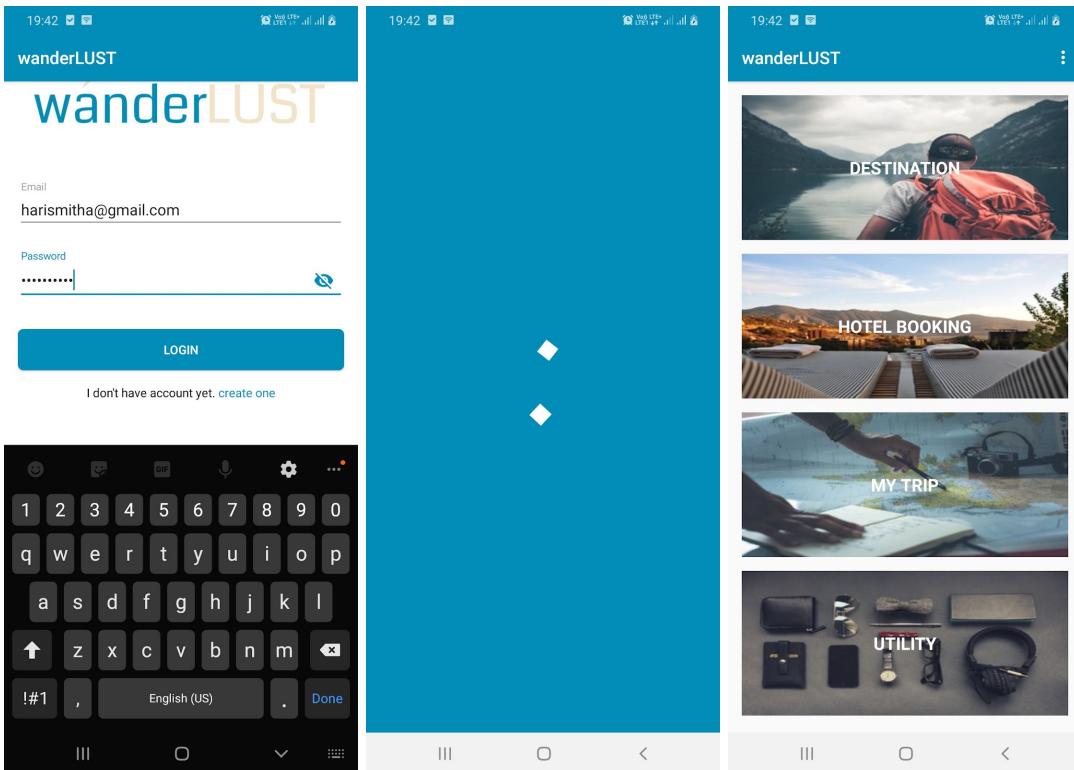
- Time: 19:41
- Header: wanderLUST
- Form:
 - Username: Harismitha
 - Email: harismitha@gmail.com
 - Password:
 - Buttons: **REGISTER**, [Back to Login](#)
- Bottom: Navigation icons (three horizontal bars, square, back arrow)

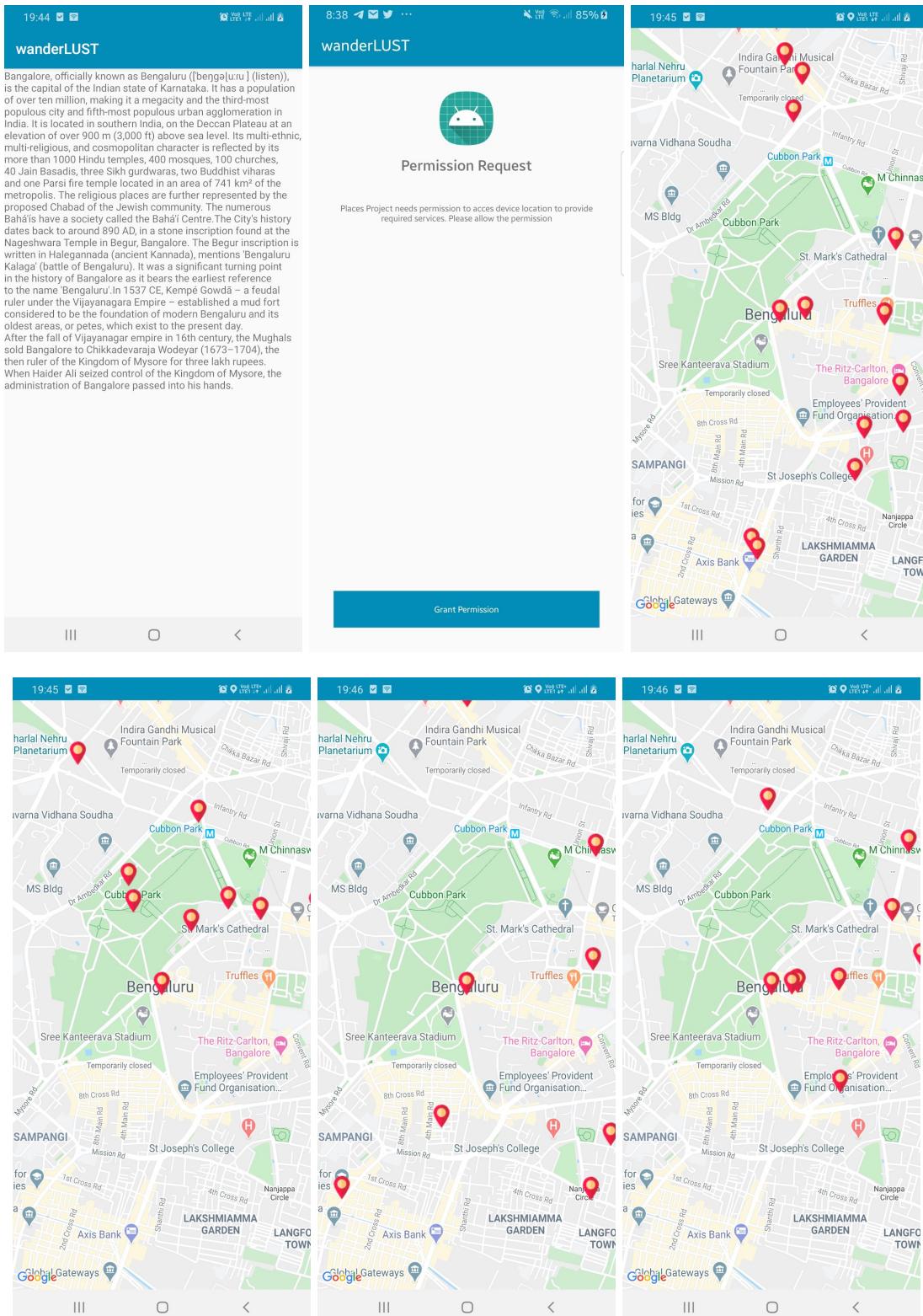
Screenshot 5 (Bottom Middle): Login Screen

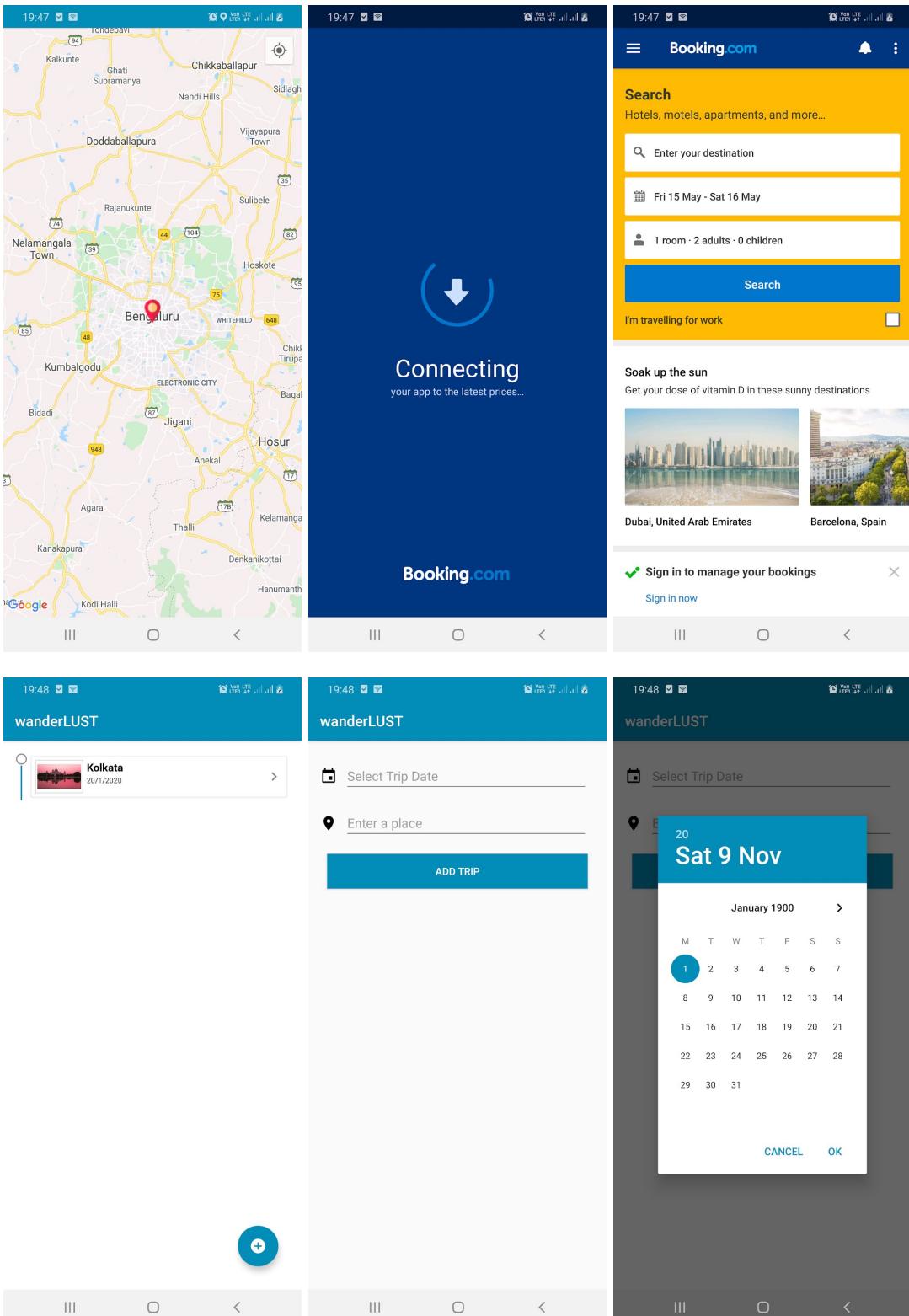
- Time: 20:07
- Header: wanderLUST
- Form:
 - Email: harismitha@gmail.com
 - Password: |
 - Message: Please enter valid password!
 - Buttons: **LOGIN**, [I don't have account yet. *create one*](#)
- Bottom: Navigation icons (three horizontal bars, square, back arrow)

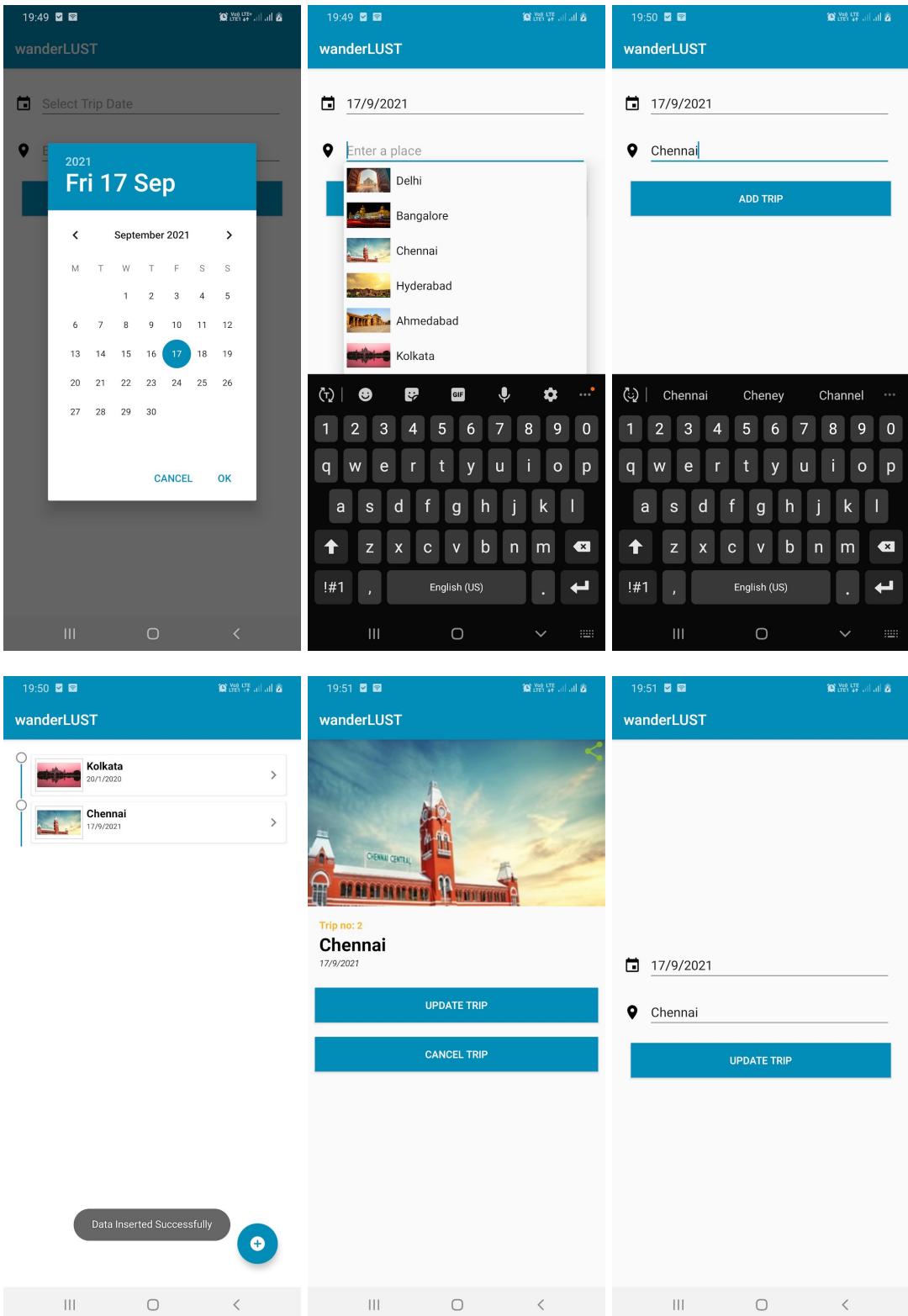
Screenshot 6 (Bottom Right): Login Screen with Error

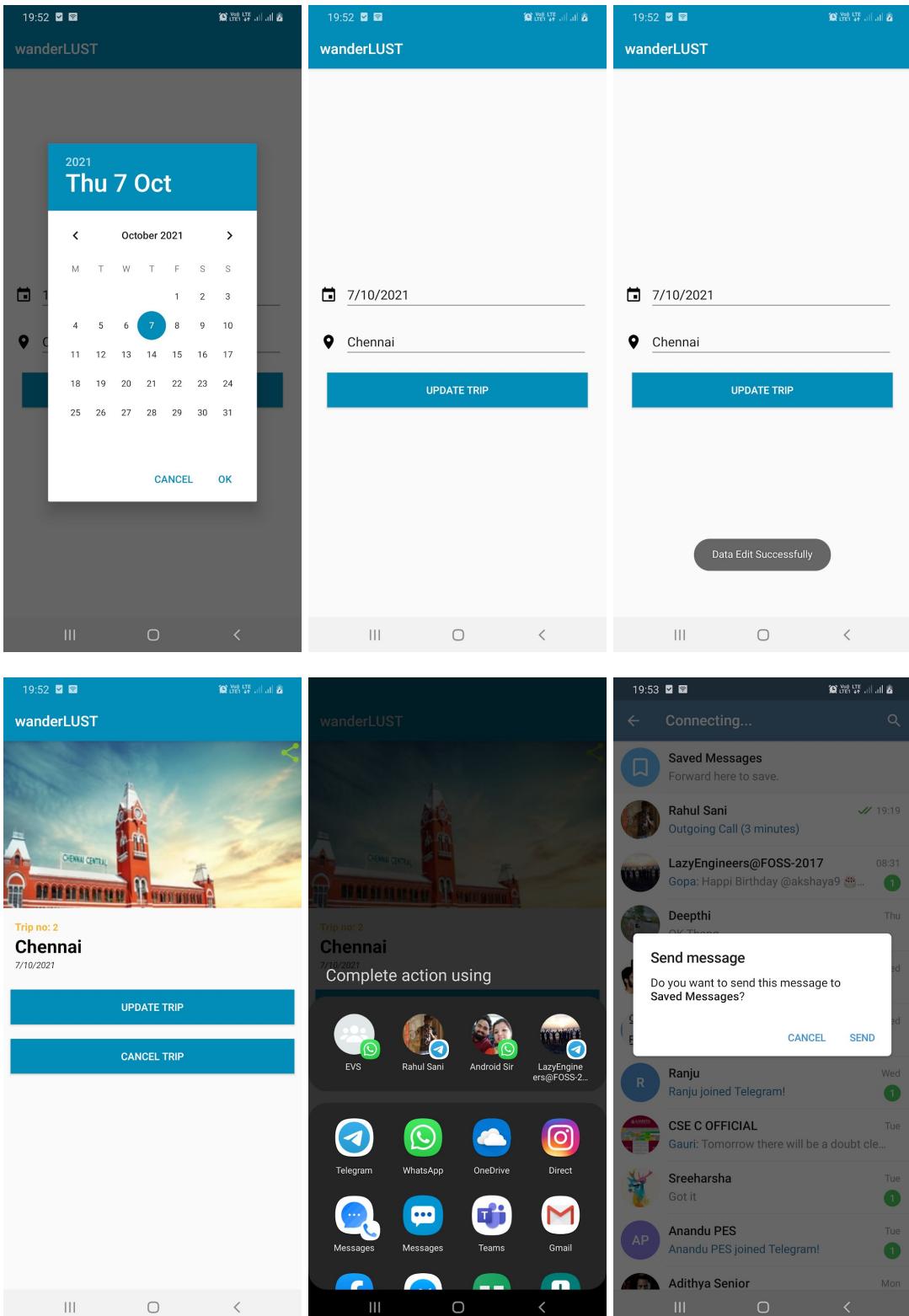
- Time: 20:06
- Header: wanderLUST
- Form:
 - Email: harismitha@gmail.com
 - Password:
 - Message: Password is to short!
 - Buttons: **LOGIN**, [I don't have account yet. *create one*](#)
- Bottom: Navigation icons (three horizontal bars, square, back arrow)











Saved Messages

Change on the Multigenerational Migration Journey of Monarch Butterflies.pdf 13:26 ✓

Impact of Climate Change on Monarch... 1.5 MB PDF 13:31 ✓

Impact of Climate Change on Monarch Butterflies-plagiarism.pdf 13:31 ✓

May 14 https://forms.office.com/Pages/ResponsePage.aspx?id=D4IULeMYcEyXzbweXyH5yxxX0zQlBSJAmrRy5cY5a0IUNVZDVFg1NTBLQ0pZNFHJN1ZZWkVXMDY2Ry4u

Office Fill | Grace Marks Request Microsoft Forms 09:22 ✓

May 15 Planning a trip to Jaipur on 11/6/2021. Would you like to join? 01:47 ✓

Planning a trip to Chennai on 7/10/2021. Would you like to join? 19:53 ✓

Message

wanderLUST

Kolkata 20/1/2020

You sure, that you want to cancel?

NO YES

CANCEL TRIP

Utility

CHECK LIST

COMPASS

WEATHER FORECAST

WORLD CLOCK

Checklist

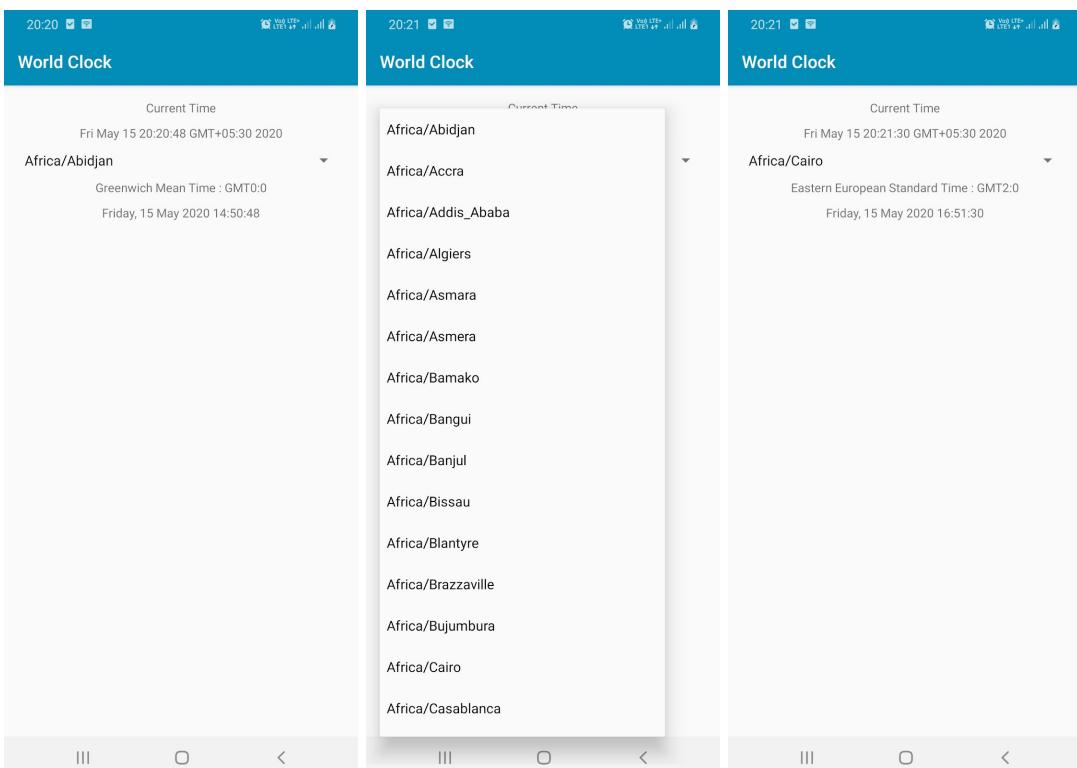
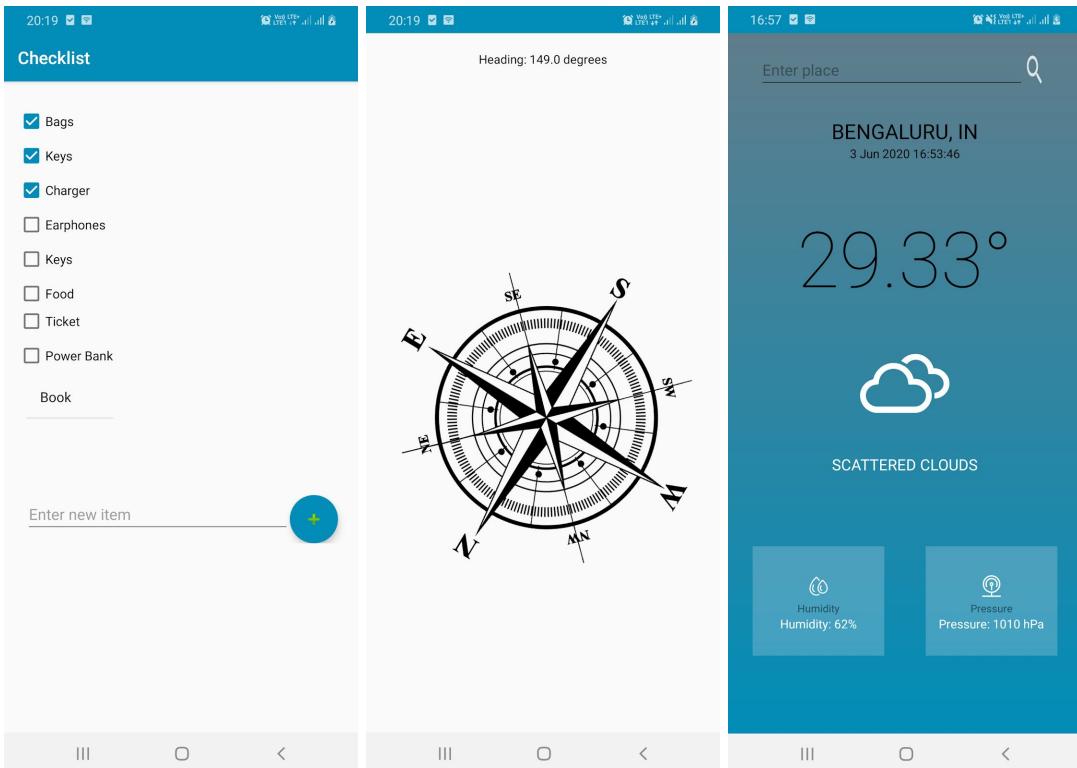
- Bags
- Keys
- Charger
- Earphones
- Food
- Ticket
- Power Bank

Enter new item

Checklist

- Bags
- Keys
- Charger
- Earphones
- Food
- Ticket
- Power Bank

Book



CHAPTER 9

CONCLUSION

Every person will travel in their life to different cities and this application is helpful to people who want to travel long journeys with their friends and make an organized journey. The google maps help viewing the cities and help to find nearby places and provides a link for lodging through Booking.com. It is helpful for people who feel alone in new cities and it provides nearby travelers to mingle and make friends. It also provides additional features like Checklist, Compass, etc. to make the trip more convenient and memorable.