Tour and Travel Management System

Project Report submitted in partial fulfillment of the requirements for the award of

Bachelor's Degree in Computer Science

(University of Calicut)

Submitted by

Miss. ANJALY P A

Miss. ASHMIYA V M

Miss. SANDRA.BIJU

SREEKALA M

(Internal Guide)

2023-2024

DEPARTMENT OF COMPUTER SCIENCE

VIMALA COLLEGE (AUTONOMOUS)

THRISSUR – KERALA

DEPARTMENT OF COMPUTER SCIENCE

CERTIFICATE

This is to certify that the project work entitled

Tour and Travel Management System

is a bonafide work done by

Anjaly P A
Ashmiya V M
Sandra Biju

Under my guidance during the period of November 2023 to March 2024 in partial fulfillment of the requirements for awarding Bachelor's degree in Computer Science by the University of Calicut.

Head of the Department **Project Guide**

The presentation and viva-voce examination was conducted on ……..at Vimala College, Thrissur.

Examiners:

Register No. …… **1. ……**

Name…… **2. …….**

ACKNOWLEDGEMENT

First of all, we would like to thank the LORD ALMIGHTY for giving us zeal to complete this project within the stipulated time and for the blessing he has showered on us. Our spiritual quotient gave us more strength and motivation that helped us immensely. We thank our Principal Dr. Sr. Beena Jose for giving us this golden opportunity to undertake our course in NAAC accredited "A+" grade college. We also express our gratitude towards Dr Sareena Rose, HoD of the Computer Science department for her extensive support to students in acquiring knowledge with various systems and advanced lab facilities available in the college. We repeat our heartfelt gratitude to our guide Ms Sreekala M for her effective guidance, timely suggestions and encouragement as an internal guide. We gratefully acknowledge the timely support of Mr Albin, lab admin for his support rendered to avail the hardware and software resources needed for the project. We are also thankful to all other staff of the department for their support in developing our project named "Tour and Travel Management System". We express our deep-felt gratitude to our classmates who have helped us to complete this project.

Team members,

Anjaly P A

Ashmiya V M

Sandra Biju

SYNOPSIS

The main objective of the tour and travel management system manage the operation related to travelling, booking and arranging accommodation. The system is designed to help traveler's to get the idea about the places they are going to visit, the quality of amenities and services and the attraction at their destination.

The registered user can access the account with valid credentials and users can view and book admin added package. It also helps them to book desired tour package, bus, hotel and other necessary services with minimum effort. It also provides payment and security features to ensure a safe and secure transaction. In the Tour and Travel Management, admin can handle the functionalities like add new tour packages, edit/delete tour packages. The hotel owners can register and login and can manage information about their hotel. Bus owners can register and login and can manage information about their bus after the registration admin can accept or reject hotel owners and bus owners.

The primary aim of developing the Tour and Travel Management System project is to develop a system that helps traveler's to plan their trip more efficiently and effectively. Another important reason for developing this project is to create a centralized platform that facilitates seamless communication and efficient management of various travel related process.

Table Content

SI NO	Topic	Page No
1	**INTRODUCTION**	
2	**SYSTEM STUDY** • Existing System • Disadvantage of existing system • Proposed system • Advantages of proposed system • Problem definition	
3	**SYSTEM ANALYSIS** • Hardware/software requirement • Feasibility study • DFD	
4	**SYSTEM DESIGN** • Architectural Design • Form design	
5	**SYSTEM DEVELOPMENT** • Features of front end • Code	
6	**TESTING** • System testing • Testing strategy	
7	**SYSTEM IMPLEMENTATION**	
8	**SYSTEM MAINTENANCE**	
9	**CONCLUSION**	
10	**SCOPE FOR FURTHER ENHANCEMENT**	
11	**BIBLIOGRAPHY**	
12	**APPENDIX**	

INTRODUCTION

INTRODUCTION

1. INTRODUCTION

1.1 About the project

The main purpose of the Tour and Travel Management System is to provide a convenient way for a customer to book a hotel, bus and package for tour purpose.The objective of this project is to develop a system that automates the process and activities of a travel.

In this project we will make it an easier task of searching places and booking packages. In the present system a customer has to approach various agencies to find details of places to book packages. This often requires a lot of time and effort. This application is developed to provide best traveling services to the customer. We have developed a Tour and Travel Management System to provide a search platform where a tourist can find tour places according to their choice. This system also helps to promote responsible and interesting tourism so that people can enjoy their holidays at their favorable places.

The registered user can access the account with valid credentials. Users can view and book packages according to categories. Online payment options are available to users. Users can book their favorable places.

In Tour and Travel Management System admin can handle the functionalities like add new packages edit/delete packages. Admin have authority to view packages and accept or reject hotel owners and bus owners request and admin can view the feedback of the user. The payment transaction and user details are also viewable to admin.

SYSTEM STUDY

2. SYSTEM STUDY

2.1 Existing system

In the existing system, a customer has to approach various agencies to find details of place and to book tickets. This often requires a lot of time and effort. A customer may not get the desired information from the offices and customer may be misguided.

2.2 Disadvantages of Existing system

- Lack of real time updates
- Difficulty in customization
- Poor user experience

2.3 Proposed system

Proposed system is an android application which provides user relevant information on the places, hotels and buses. It doesn't take a lot of time or effort, also provides additional information on the places near to their destinations

2.4 Advantages of Proposed system

- Real time updates
- Cost efficiency
- Customer feedback integration

2.5 Problem definition

The main objective of tour and travel management system is to create a user friendly platform that facilitates efficient of tours and travel services.while enhancing the overall customer experience

SYSTEM ANALYSIS

3. SYSTEM ANALYSIS

The objective of the system analysis phase is the establishment of the requirement for the system to be acquired, developed and installed. Analysis of the project is to understand the intricacy and forms the vital part of the system study. Problematic areas are identified and information is collected. Fact finding or gathering is essential to any analysis of the requirement. It is necessary that the analyst familiarize with objects, activities and the functions of the device in which the system is implemented.

System analysis involves the study of determining how to make a system better.The System analyst should develop an alternative system and evaluate each in terms of cost,benefits and feasibility. The term analysis, system and development are used in the sequence, because in practice this sequence of steps is used to construct a computer-based information system.At the conclusion of the system-analysis there is a system description handset requirements for a new system. The design phase which follows the analysis phase, proposes a new system that meets these requirements. This new system may be built afresh or by changing the existing System. The system analysis of this project involves studying user requirements, identifying key functionalities such as booking packages and itinerary planning,defining data structure for storing information like customer details and package bookings.

3.1 Resource Requirements

3.1.1 Software Requirements

Front end: Java, Xml

Back end: python with Django

Database: MySQL

IDE: Android Studio

Server: Localhost

3.1.2 Hardware Requirements

Processor: Dual Core Memory: 4 GB Hard disk: 1 TB Monitor: 18.5” LED Keyboard: 104 keys Mouse: Optical Android phone: 6 GB

3.2 Feasibility Study

A feasibility study is an evaluation of proposal design to determine the difficulty in arriving at a designated task. Generally, a feasibility study precedes technical development and project implementation. In other words, a feasibility study is an evaluation or analysis of potential impact of the proposed project. This study is to establish the reason for developing the software that is acceptable to users, adaptable to change and conformable to establish standards. The feasibility study in a project development process shows whether the project creates a technical and economically feasible concept.

Our Application “Tour and Travel Management System"is feasible because it is usable for all the travelers. This can be used to manage the operation related to traveling, booking and arranging accommodation.

Feasibility analysis involves the following steps:

- Provides the searching facilities based on various factors such as packages,hotel,bus
- Prepare a system flowchart
- It provide all the information of packages
- Describe and identify characteristics of the tour and travel system.
- Weight system performance and cost data
- Prepare and report final project directives and management

The various types of feasibilities are:

- Economic feasibility
- Technical feasibility

- Operational feasibility

3.2.1 Economic Feasibility

Economic analysis is the most frequently used method for evaluating the effectiveness of a new system. More commonly known as cost/benefits analysis, the procedure is to determine the benefits and savings that are expected from a candidate system and compare it with costs.if benefits weigh more than cost, then the decision is made to design and implement the system. Cost estimating is essentially an intuitive process that attempts to predict the final outcome of a future capital expenditure. Even though it seems impossible to come up with the exact number of costs and benefits for a particular project during this initial phase of the development process, one should spend adequate time in estimating the costs and benefits of the project for comparison. Our system is economically feasible as it can be availed with less cost and less maintenance charge.

3.2.2 Technical Feasibility

The assessment is based on an outline design of system requirements in terms of input, processes, output, fields, programs and procedures. Technical feasibility is carried out to determine whether the software chosen has the capability to implement the requirements of the Tour And Travel Management system in an efficient and effective manner. In developing the new system, one has to investigate and compare technology providers, determine reliability and competitiveness of the system, and identify limitations or constraints of technology, as well as the risk of the proposed system that is dependent on the size of the system, complexity, and groups experience with the similar systems. According to the definition of technical feasibility the compatibility of front-end and back-end is very important. In this project the compatibility of front is very good as we have used Android Studio. The Android environment provides easy implementation of

code, well defined and easy to use interface and easy debugging. Java being a simple and powerful language makes the code efficient and sophisticated.

3.2.3 Operational feasibility

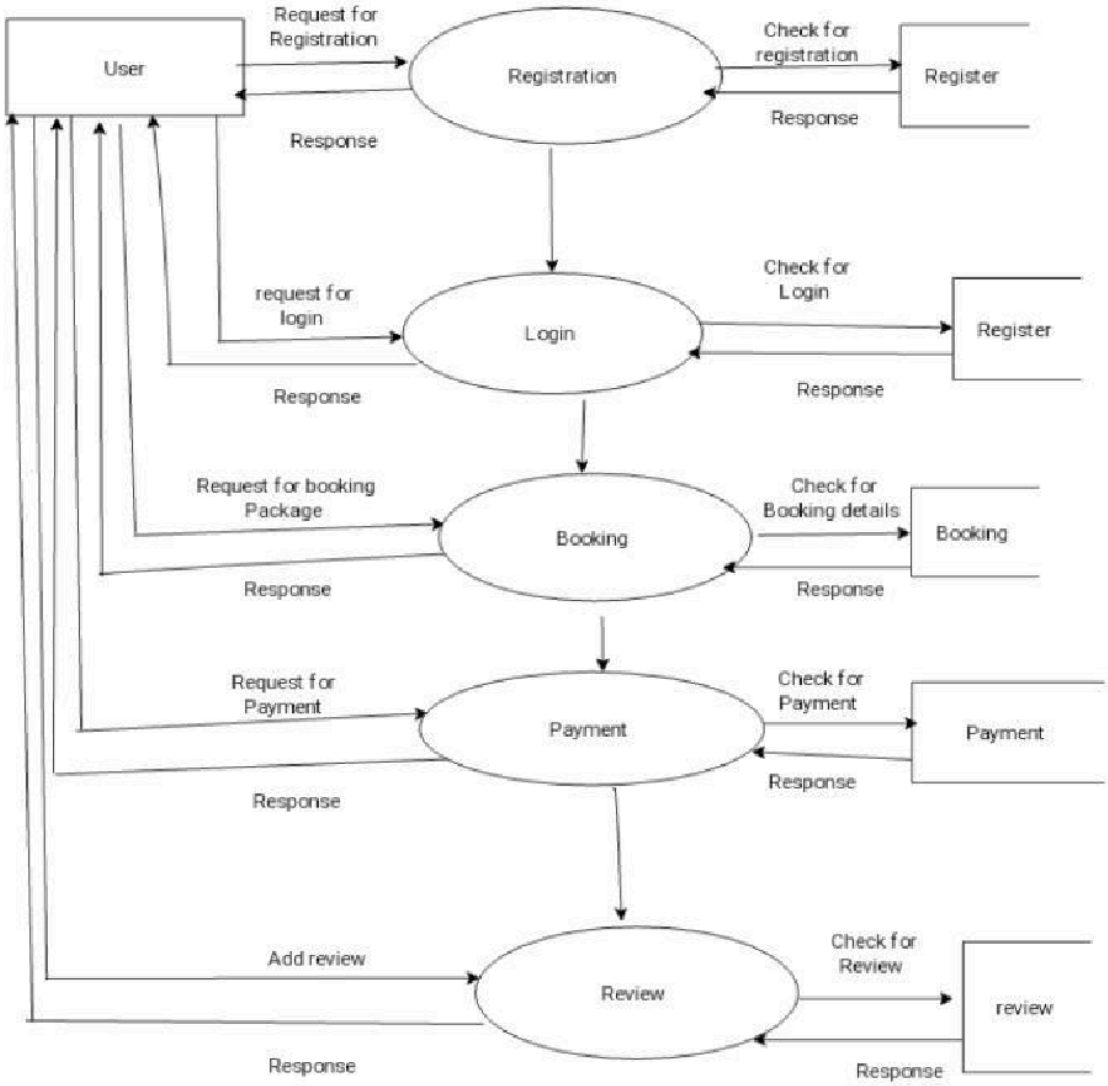
It is a measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development. This feasibility study is carried out by a small group of people who are familiar with information system technique and are skilled in system analysis and design process. Proposed projects are beneficial only if they can be turned into an information system that will meet the operating requirements of the organization. This project is operational feasible it is helpful to users by streaming booking process, managing itineraries effectively.

3.3 Data Flow Diagram

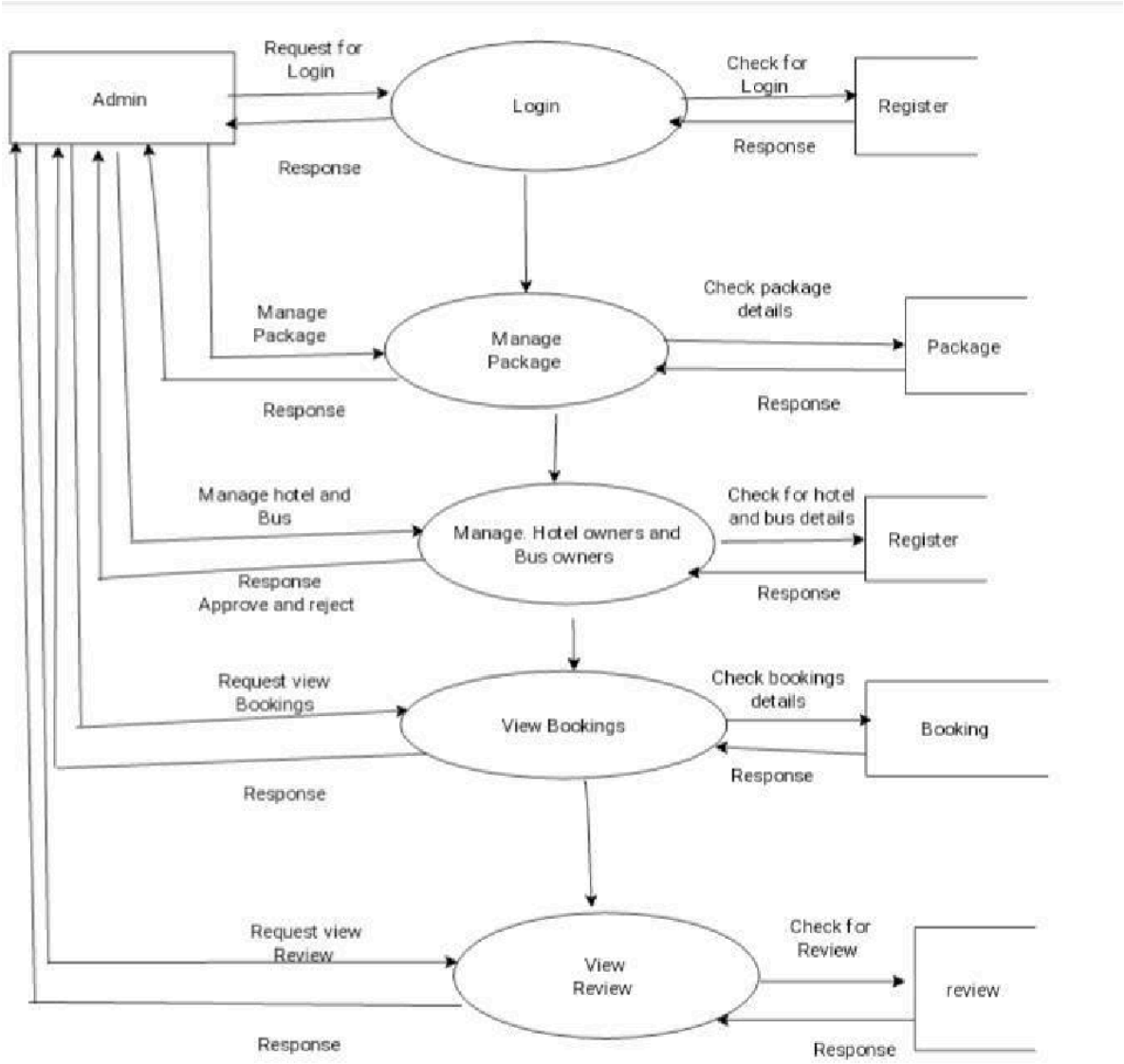
Level 0:

Level 1:

User app



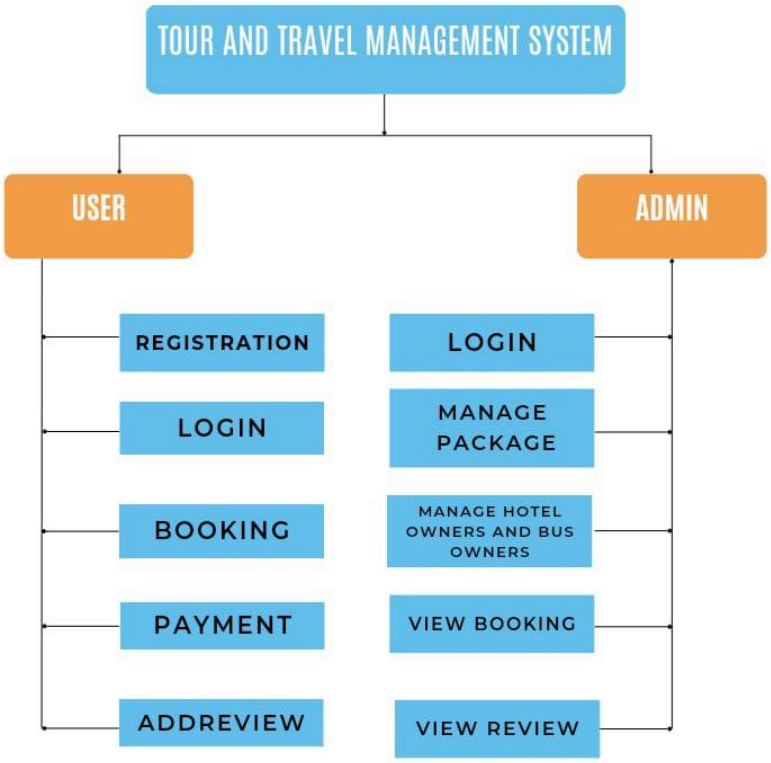
Admin app:


Admin
Request for Login
Response
Login
Check for Login
Response
Register
Manage Package
Response
Manage Package
Check package details
Response
Package
Manage hotel and Bus
Response Approve and reject
Manage. Hotel owners and Bus owners
Check for hotel and bus details
Response
Register
Request view Bookings
Response
View Bookings
Check bookings details
Response
Booking
Request view Review
Response
View Review
Check for Review
Response
review

SYSTEM DESIGN

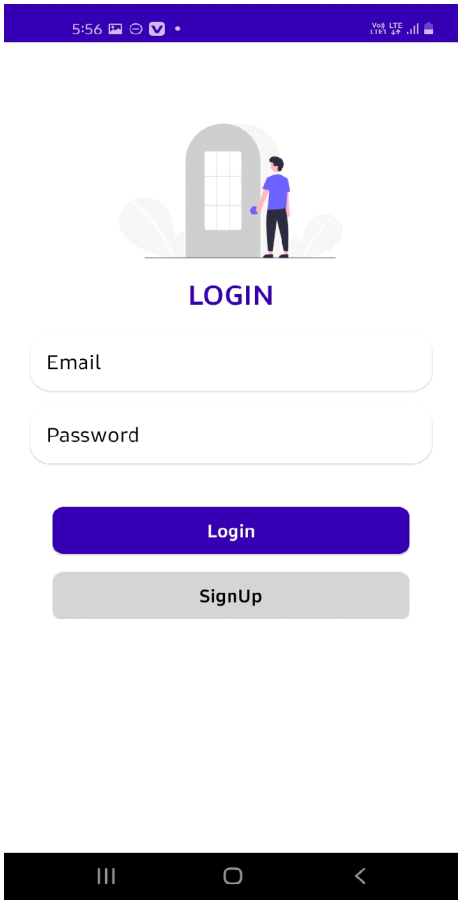
4. SYSTEM DESIGN

4.1 Architectural Design: -



4.2 Form Design

Admin and User App


5:56
LOGIN
Email
Password
Login
SignUp

SYSTEM DEVELOPMENT

5.SYSTEM DEVELOPMENT

5.1 Features of front-end

Android Studio is Android's official IDE. It is purpose-built for Android to accelerate the development and help to build the highest quality apps for every Android device. The latest release of Android brings a host of new features, like stronger protections for user privacy,new ways to engage users, better support for foldables, Vulcan extensions and more.Android Studio provides a better layout editor and includes everything that is needed to build Android apps. Android runs on devices of all shapes and sizes, providing a vast opportunity for continued user engagement.

5.2 Code

//XML CODE

//activity_login.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=".ui.activities.LoginActivity">

   <FrameLayout
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:background="@drawable/login">

      <LinearLayout
         android:layout_width="match_parent"
         android:layout_height="wrap_content"
         android:layout_gravity="center"
         android:gravity="center"
         android:orientation="vertical">
```

```
<TextView
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:layout_marginBottom="0dp"
   android:text="Login"
   android:textAllCaps="true"
   android:textColor="@color/blue"
   android:textSize="25sp"
   android:textStyle="bold" />

<EditText
   android:id="@+id/email"
   android:layout_width="match_parent"
   android:layout_height="60dp"
   android:layout_marginStart="@dimen/gap_xxl"
   android:layout_marginTop="20dp"
   android:layout_marginEnd="@dimen/gap_xxl"
   android:background="@drawable/edit_text_bg"
   android:drawablePadding="@dimen/gap_md"
   android:elevation="@dimen/gap_xsm"
   android:hint="Email"
   android:inputType="textEmailAddress"
   android:padding="@dimen/gap_md"
   android:paddingStart="@dimen/gap_lg"
   android:paddingEnd="@dimen/gap_lg"
   android:textColor="@color/black"
   android:textColorHint="@color/black" />

<EditText
   android:id="@+id/password"
   android:layout_width="match_parent"
   android:layout_height="60dp"
   android:layout_marginStart="@dimen/gap_xxl"
   android:layout_marginTop="@dimen/gap_lg"
   android:layout_marginEnd="@dimen/gap_xxl"
   android:background="@drawable/edit_text_bg"
   android:drawablePadding="@dimen/gap_sm"
   android:elevation="@dimen/gap_xsm"
   android:hint="Password"
   android:inputType="textPassword"
   android:padding="@dimen/gap_md"
   android:paddingStart="@dimen/gap_lg"
   android:paddingEnd="@dimen/gap_lg"
   android:textColor="@color/black"
   android:textColorHint="@color/black" />


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

```
            android:layout_height="match_parent"
            android:layout_marginStart="@dimen/gap_xxxl"
            android:layout_marginTop="@dimen/gap_xxxl"
            android:layout_marginEnd="@dimen/gap_xxxl"
            android:orientation="vertical">

            <androidx.appcompat.widget.AppCompatButton
               android:id="@+id/login"
               android:layout_width="match_parent"
               android:layout_height="wrap_content"
               android:background="@drawable/app_btn_bg"
               android:elevation="4dp"
               android:gravity="center"
               android:text="Login"
               android:textAllCaps="false"
               android:textColor="@color/btn_text_color"
               android:textSize="16sp"
               android:textStyle="bold" />

            <androidx.appcompat.widget.AppCompatButton
               android:id="@+id/signup"
               android:layout_width="match_parent"
               android:layout_height="wrap_content"
               android:layout_marginTop="@dimen/gap_xl"
               android:background="@drawable/app_btn_secondry_bg"
               android:elevation="4dp"
               android:gravity="center"
               android:text="SignUp"
               android:textAllCaps="false"
               android:textColor="@color/black"
               android:textSize="16sp"
               android:textStyle="bold" />

         </LinearLayout>
      </LinearLayout>
   </FrameLayout>

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

//userssignup_btm_nav.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
   <item
      android:id="@+id/Users"
      android:title="User"
      android:icon="@drawable/ic_profile"/>
   <item
```

```
      android:id="@+id/Busowners"
      android:title="Bus Owners"
      android:icon="@drawable/ic_profile"/>
   <item
      android:id="@+id/Hotelowners"
      android:title="Hotel Owners"
      android:icon="@drawable/ic_profile"/>
</menu>
```

//activity_users_sign_up.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=".ui.activities.UsersSignUp">


   <FrameLayout
      android:id="@+id/flFragment"
      android:layout_width="match_parent"
      android:layout_height="0dp"
      android:background="@drawable/signup"
      app:layout_constraintBottom_toTopOf="@+id/bottomNavigationView"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.0"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="1.0">

     <TextView
        android:id="@+id/gotologin"
        android:layout_width="200dp"
        android:layout_height="40dp"
        android:gravity="center"
        android:layout_gravity="bottom"
        android:layout_marginLeft="100dp"
        android:layout_marginBottom="40dp"
        android:text="Go To Login"
        android:textSize="20dp"
        android:textStyle="bold"
        android:textColor="@color/blue"
       />
   </FrameLayout>
```

```
    <com.google.android.material.bottomnavigation.BottomNavigationView
        android:id="@+id/bottomNavigationView"
        android:layout_width="match_parent"
        android:layout_height="75dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:menu="@menu/userssignup_btm_nav_menu"/>

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

//activity_signup.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:background="@color/blue"
    tools:context=".ui.activities.SignupActivity">

    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:gravity="center"
            android:padding="@dimen/gap_md"
            android:orientation="vertical">

            <TextView
                android:id="@+id/sign"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginBottom="@dimen/gap_xxl"
                android:text="Sign Up"
                android:textColor="@color/white"
                android:textAllCaps="true"
                android:textSize="@dimen/font_xxl"
                android:textStyle="bold" />


            <androidx.cardview.widget.CardView
                android:layout_width="match_parent"
```

```
android:layout_height="wrap_content"
android:layout_marginStart="@dimen/gap_lg"
android:layout_marginEnd="@dimen/gap_lg"
app:cardCornerRadius="@dimen/gap_xl"
android:elevation="@dimen/gap_sm">

<LinearLayout
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:orientation="vertical"
   android:background="@color/white"
   android:padding="@dimen/gap_md"
   android:gravity="center">

   <ImageView
      android:layout_width="100dp"
      android:layout_height="100dp"
      android:src="@drawable/logo_nobg"/>

   <EditText
      android:id="@+id/name"
      android:layout_width="match_parent"
      android:layout_height="50dp"
      android:layout_margin="@dimen/gap_sm"
      android:layout_marginStart="@dimen/gap_xxl"
      android:layout_marginTop="@dimen/gap_xl"
      android:layout_marginEnd="@dimen/gap_xxl"
      android:background="@drawable/edit_text_bg"
      android:drawablePadding="@dimen/gap_md"
      android:elevation="@dimen/gap_xsm"
      android:hint="Name"
      android:inputType="textCapWords"
      android:maxLines="1"
      android:padding="@dimen/gap_md"
      android:paddingStart="@dimen/gap_lg"
      android:paddingEnd="@dimen/gap_lg"
      android:textColor="@color/black"
      android:textColorHint="@color/black"
      android:textSize="@dimen/font_lg" />

   <EditText
      android:id="@+id/phone"
      android:layout_width="match_parent"
      android:layout_height="50dp"
      android:layout_margin="@dimen/gap_sm"
      android:layout_marginStart="@dimen/gap_xxl"
      android:layout_marginTop="@dimen/gap_xl"
      android:layout_marginEnd="@dimen/gap_xxl"
      android:background="@drawable/edit_text_bg"
      android:drawablePadding="@dimen/gap_md"
```

```
    android:elevation="@dimen/gap_xsm"
    android:hint="Phone"
    android:inputType="phone"

    android:maxLines="1"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/black"
    android:textColorHint="@color/black"
    android:textSize="@dimen/font_lg" />

<EditText
    android:id="@+id/email"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_margin="@dimen/gap_sm"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginTop="@dimen/gap_xl"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:hint="Email"
    android:inputType="textEmailAddress"
    android:maxLines="1"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/black"
    android:textColorHint="@color/black"
    android:textSize="@dimen/font_lg" />

<EditText
    android:id="@+id/password"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_margin="@dimen/gap_sm"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginTop="@dimen/gap_xl"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:hint="Password"
    android:inputType="textPassword"
    android:maxLines="1"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
```

```
            android:textColor="@color/black"
            android:textColorHint="@color/black"
            android:textSize="@dimen/font_lg" />

        <TextView
            android:id="@+id/login"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="@dimen/gap_md"
            android:layout_marginRight="@dimen/gap_xxl"
            android:gravity="right"
            android:text="Login"
            android:textColor="@color/blue"
            android:textSize="@dimen/font_lg"
            android:textStyle="bold" />

        <androidx.appcompat.widget.AppCompatButton
            android:id="@+id/signup"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@drawable/app_btn_bg"
            android:elevation="4dp"
            android:gravity="center"
            android:layout_marginStart="@dimen/gap_xl"
            android:layout_marginEnd="@dimen/gap_xl"
            android:layout_gravity="bottom"
            android:layout_marginTop="40dp"
            android:layout_marginBottom="20dp"
            android:text="Signup"
            android:textAllCaps="true"
            android:textColor="@color/btn_text_color"
            android:textSize="16sp"
            android:textStyle="bold" />
      </LinearLayout>
    </androidx.cardview.widget.CardView>
  </LinearLayout>
 </ScrollView>
</LinearLayout>
```

//activity_bus_sign_up.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:background="@color/blue"
```

```
tools:context=".ui.activities.BusSignUp">

<ScrollView
   android:layout_width="match_parent"
   android:layout_height="match_parent">

   <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:gravity="center"
      android:padding="@dimen/gap_md"
      android:orientation="vertical">

      <TextView
         android:id="@+id/sign"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:text="Sign Up"
         android:textStyle="bold"
         android:textColor="@color/white"
         android:textAllCaps="true"
         android:textSize="@dimen/font_xxl"
         android:layout_marginBottom="@dimen/gap_xxl"/>


      <androidx.cardview.widget.CardView
         android:layout_width="match_parent"
         android:layout_height="wrap_content"
         android:layout_marginStart="@dimen/gap_lg"
         android:layout_marginEnd="@dimen/gap_lg"
         app:cardCornerRadius="@dimen/gap_xl"
         android:elevation="@dimen/gap_sm">

         <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:background="@color/white"
            android:padding="@dimen/gap_md"
            android:gravity="center">



            <EditText
               android:id="@+id/Busowrname"
               android:layout_width="match_parent"
               android:layout_height="50dp"
               android:layout_marginTop="@dimen/gap_xl"
               android:layout_marginStart="@dimen/gap_xxl"
               android:background="@drawable/edit_text_bg"
```

```
    android:padding="@dimen/gap_md"
    android:maxLines="1"
    android:textSize="@dimen/font_lg"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:elevation="@dimen/gap_xsm"
    android:hint="Name"
    android:inputType="textCapWords"
    android:layout_margin="@dimen/gap_sm"
    android:textColor="@color/blue"
    android:drawablePadding="@dimen/gap_md"
    android:textColorHint="@color/black"
    android:layout_marginEnd="@dimen/gap_xxl"/>

<EditText
    android:id="@+id/Busowrphone"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_marginTop="@dimen/gap_xl"
    android:layout_marginStart="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:padding="@dimen/gap_md"
    android:maxLines="1"
    android:textSize="@dimen/font_lg"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:elevation="@dimen/gap_xsm"
    android:hint="Phone"
    android:inputType="phone"
    android:layout_margin="@dimen/gap_sm"
    android:textColor="@color/blue"
    android:drawablePadding="@dimen/gap_md"
    android:textColorHint="@color/black"
    android:layout_marginEnd="@dimen/gap_xxl"/>

<EditText
    android:id="@+id/Busowremail"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_marginTop="@dimen/gap_xl"
    android:layout_marginStart="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:padding="@dimen/gap_md"
    android:maxLines="1"
    android:paddingStart="@dimen/gap_lg"
    android:textSize="@dimen/font_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:elevation="@dimen/gap_xsm"
    android:hint="Email"
    android:inputType="textEmailAddress"
```

```
    android:layout_margin="@dimen/gap_sm"
    android:textColor="@color/blue"
    android:drawablePadding="@dimen/gap_md"
    android:textColorHint="@color/black"
    android:layout_marginEnd="@dimen/gap_xxl"/>

  <EditText
    android:id="@+id/Busowrpassword"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_marginTop="@dimen/gap_xl"
    android:layout_marginStart="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:padding="@dimen/gap_md"
    android:maxLines="1"
    android:paddingStart="@dimen/gap_lg"
    android:textSize="@dimen/font_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:elevation="@dimen/gap_xsm"
    android:hint="Password"
    android:inputType="textPassword"
    android:layout_margin="@dimen/gap_sm"
    android:textColor="@color/blue"
    android:drawablePadding="@dimen/gap_md"
    android:textColorHint="@color/black"
    android:layout_marginEnd="@dimen/gap_xxl"/>
  <TextView
    android:id="@+id/image"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:hint="Upload Owner Profile"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textStyle="bold"
    android:textColorHint="@color/blue"
    />


  <Button
    android:id="@+id/bus_btn_chs"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginEnd="0dp"
    android:text="CHOOSE" />

  <ImageView
    android:id="@+id/bus_gallery_view"
    android:visibility="gone"
```

```
    android:layout_marginTop="10dp"
    android:layout_gravity="center"
    android:layout_width="100dp"
    android:layout_height="100dp"/>
<EditText
    android:id="@+id/Busaddress"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginTop="@dimen/gap_lg"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:gravity="start"
    android:hint="Address of Bus Owner"
    android:inputType="textCapWords"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/blue"
    android:textColorHint="@color/black" />
<TextView
    android:id="@+id/login"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="@dimen/gap_md"
    android:textSize="@dimen/font_lg"
    android:textColor="@color/blue"
    android:textStyle="bold"
    android:gravity="right"
    android:layout_marginRight="@dimen/gap_xxl"
    android:text="Login"/>

<androidx.appcompat.widget.AppCompatButton
    android:id="@+id/signup"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@drawable/app_btn_bg"
    android:elevation="4dp"
    android:gravity="center"
    android:layout_marginStart="@dimen/gap_xl"
    android:layout_marginEnd="@dimen/gap_xl"
    android:layout_gravity="bottom"
    android:layout_marginTop="40dp"
    android:layout_marginBottom="20dp"
    android:text="Signup"
    android:textAllCaps="true"
    android:textColor="@color/btn_text_color"
    android:textSize="16sp"
```

```
                android:textStyle="bold" />
            </LinearLayout>
        </androidx.cardview.widget.CardView>
    </LinearLayout>
  </ScrollView>
</LinearLayout>
```

//activity_hotel_sign_up.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:background="@color/blue"
  tools:context=".ui.activities.SignupActivity">

  <ScrollView
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:gravity="center"
      android:padding="@dimen/gap_md"
      android:orientation="vertical">

      <TextView
        android:id="@+id/sign"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Sign Up"
        android:textAllCaps="true"
        android:textStyle="bold"
        android:textColor="@color/white"
        android:textSize="@dimen/font_xxl"
        android:layout_marginBottom="@dimen/gap_xxl"/>


      <androidx.cardview.widget.CardView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="@dimen/gap_lg"
        android:layout_marginEnd="@dimen/gap_lg"
        app:cardCornerRadius="@dimen/gap_xl"
        android:elevation="@dimen/gap_sm">
```

```
<LinearLayout
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:orientation="vertical"
   android:background="@color/white"
   android:padding="@dimen/gap_md"
   android:gravity="center">
   <EditText
      android:id="@+id/hotelownername"
      android:layout_width="match_parent"
      android:layout_height="50dp"
      android:layout_marginTop="@dimen/gap_xl"
      android:layout_marginStart="@dimen/gap_xxl"
      android:background="@drawable/edit_text_bg"
      android:padding="@dimen/gap_md"
      android:maxLines="1"
      android:textSize="@dimen/font_lg"
      android:paddingStart="@dimen/gap_lg"
      android:paddingEnd="@dimen/gap_lg"
      android:elevation="@dimen/gap_xsm"
      android:hint="Name"
      android:inputType="textCapWords"
      android:layout_margin="@dimen/gap_sm"
      android:textColor="@color/blue"
      android:drawablePadding="@dimen/gap_md"
      android:textColorHint="@color/black"
      android:layout_marginEnd="@dimen/gap_xxl"/>

   <EditText
      android:id="@+id/hotelname"
      android:layout_width="match_parent"
      android:layout_height="50dp"
      android:layout_marginTop="@dimen/gap_xl"
      android:layout_marginStart="@dimen/gap_xxl"
      android:background="@drawable/edit_text_bg"
      android:padding="@dimen/gap_md"
      android:maxLines="1"
      android:textSize="@dimen/font_lg"
      android:paddingStart="@dimen/gap_lg"
      android:paddingEnd="@dimen/gap_lg"
      android:elevation="@dimen/gap_xsm"
      android:hint="Hotel Name"
      android:inputType="textCapWords"
      android:layout_margin="@dimen/gap_sm"
      android:textColor="@color/blue"
      android:drawablePadding="@dimen/gap_md"
      android:textColorHint="@color/black"
      android:layout_marginEnd="@dimen/gap_xxl"/>
   <TextView
```

```
    android:id="@+id/image"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:text="Upload Hotel Image"
    android:inputType="textCapWords"
    android:textStyle="bold"
    android:textColor="@color/blue"
   />


<Button
    android:id="@+id/hotel_img_chs"
    android:layout_width="wrap_content"
    android:layout_gravity="center"
    android:layout_height="wrap_content"
    android:text="CHOOSE" />
<ImageView
    android:id="@+id/hotel_gallery_view"
    android:visibility="gone"
    android:layout_marginTop="10dp"
    android:layout_gravity="center"
    android:layout_width="100dp"
    android:layout_height="100dp"/>
<EditText
    android:id="@+id/hotelownerphone"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_marginTop="@dimen/gap_xl"
    android:layout_marginStart="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:padding="@dimen/gap_md"
    android:maxLines="1"
    android:textSize="@dimen/font_lg"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:elevation="@dimen/gap_xsm"
    android:hint="Phone"
    android:inputType="phone"
    android:layout_margin="@dimen/gap_sm"
    android:textColor="@color/blue"
    android:drawablePadding="@dimen/gap_md"
    android:textColorHint="@color/black"
    android:layout_marginEnd="@dimen/gap_xxl"/>

<EditText
    android:id="@+id/hotelowneremail"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_marginTop="@dimen/gap_xl"
```

```
    android:layout_marginStart="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:padding="@dimen/gap_md"
    android:maxLines="1"
    android:paddingStart="@dimen/gap_lg"
    android:textSize="@dimen/font_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:elevation="@dimen/gap_xsm"
    android:hint="Email"
    android:inputType="textEmailAddress"
    android:layout_margin="@dimen/gap_sm"
    android:textColor="@color/blue"
    android:drawablePadding="@dimen/gap_md"
    android:textColorHint="@color/black"
    android:layout_marginEnd="@dimen/gap_xxl"/>

<EditText
    android:id="@+id/hotelownerpassword"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_marginTop="@dimen/gap_xl"
    android:layout_marginStart="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:padding="@dimen/gap_md"
    android:maxLines="1"
    android:paddingStart="@dimen/gap_lg"
    android:textSize="@dimen/font_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:elevation="@dimen/gap_xsm"
    android:hint="Password"
    android:inputType="textPassword"
    android:layout_margin="@dimen/gap_sm"
    android:textColor="@color/blue"
    android:drawablePadding="@dimen/gap_md"
    android:textColorHint="@color/black"
    android:layout_marginEnd="@dimen/gap_xxl"/>



<EditText
    android:id="@+id/hoteldesc"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginTop="@dimen/gap_lg"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:gravity="start"
```

```
            android:hint="Address of hotel"
            android:inputType="textCapWords"
            android:padding="@dimen/gap_md"
            android:paddingStart="@dimen/gap_lg"
            android:paddingEnd="@dimen/gap_lg"
            android:textColor="@color/blue"
            android:textColorHint="@color/black" />
        <TextView
            android:id="@+id/login"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="@dimen/gap_md"
            android:textSize="@dimen/font_lg"
            android:textColor="@color/blue"
            android:textStyle="bold"
            android:gravity="right"
            android:layout_marginRight="@dimen/gap_xxl"
            android:text="Login"/>

        <androidx.appcompat.widget.AppCompatButton
            android:id="@+id/signup"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@drawable/app_btn_bg"
            android:elevation="4dp"
            android:gravity="center"
            android:layout_marginStart="@dimen/gap_xl"
            android:layout_marginEnd="@dimen/gap_xl"
            android:layout_gravity="bottom"
            android:layout_marginTop="40dp"
            android:layout_marginBottom="20dp"
            android:text="Signup"
            android:textAllCaps="true"
            android:textColor="@color/btn_text_color"
            android:textSize="16sp"
            android:textStyle="bold" />
      </LinearLayout>
    </androidx.cardview.widget.CardView>
  </LinearLayout>
 </ScrollView>
</LinearLayout>
```

//activity_main.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"
    tools:context=".ui.activities.MainActivity">

<TextView
    android:layout_width="match_parent"
    android:text=""
    android:gravity="center"
    android:textSize="20sp"
    android:padding="5dp"
    android:textColor="@color/black"
    android:layout_height="wrap_content"/>


    <LinearLayout
        android:layout_width="match_parent"
        android:orientation="horizontal"
        android:weightSum="2"
        android:padding="5dp"
        android:layout_height="wrap_content">
    </LinearLayout>

    <FrameLayout
        android:id="@+id/navigation_container"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:scaleType="fitXY"
            android:src="@drawable/homebg" />

        <androidx.fragment.app.FragmentContainerView
            android:id="@+id/container"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_marginTop="50dp"
            android:layout_marginBottom="60dp" />


        <FrameLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_gravity="bottom">

            <FrameLayout
                android:layout_width="match_parent"
                android:layout_height="50dp"
```

```
android:layout_gravity="top"
android:background="@color/tranparrent_white"
android:elevation="@dimen/gap_sm"
android:orientation="horizontal">

<TextView
   android:id="@+id/title"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:layout_gravity="center"
   android:text="@string/app_name"
   android:textColor="@color/blue"
   android:textSize="@dimen/font_vlg"
   android:textStyle="bold" />

<FrameLayout
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:layout_gravity="left|center_vertical"
   android:layout_marginEnd="@dimen/gap_lg"
   android:orientation="horizontal">

   <ImageView
      android:id="@+id/add_spot"
      android:layout_width="20dp"
      android:layout_height="20dp"
      android:visibility="gone"
      android:layout_gravity="left|center_vertical"
      android:layout_marginStart="@dimen/gap_lg"
      android:src="@drawable/ic_add" />

   <TextView
      android:id="@+id/item_text"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_marginLeft="40dp"
      android:text=""
      android:textColor="@color/black"
      android:textStyle="bold"
      android:visibility="gone" />

   <ImageView
      android:id="@+id/logout"
      android:layout_width="20dp"
      android:layout_height="20dp"
      android:layout_gravity="right|center_vertical"
      android:layout_marginStart="@dimen/gap_lg"
      android:onClick="LogOut"
      android:src="@drawable/ic_logout"
      app:tint="@color/blue" />
```

```
            </FrameLayout>

        </FrameLayout>

        <com.google.android.material.bottomnavigation.BottomNavigationView
            android:id="@+id/nav_view"
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:layout_gravity="bottom"
            android:background="@color/white"
            android:elevation="@dimen/gap_sm"
            app:itemIconSize="15dp"
            app:itemIconTint="@drawable/navigation_selector"
            app:itemTextColor="@color/blue"
            app:labelVisibilityMode="labeled"
            app:menu="@menu/bottom_nav_menu" />


        <Switch
            android:id="@+id/switch1"
            android:visibility="gone"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Switch" />
      </FrameLayout>
    </FrameLayout>
</LinearLayout>
```

//view_users_list.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:id="@+id/busfrag"
    >

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center"
        android:orientation="vertical"
        android:padding="@dimen/gap_md"
        android:background="@color/white">

        <androidx.cardview.widget.CardView
```

```
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_marginStart="@dimen/gap_lg"
android:layout_marginEnd="@dimen/gap_lg"
android:elevation="@dimen/gap_sm"
app:cardCornerRadius="@dimen/gap_xl">

<LinearLayout
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:background="@color/light_grey"
   android:gravity="center"
   android:layout_gravity="center"
   android:orientation="vertical"
   android:padding="@dimen/gap_md">

   <TextView
      android:id="@+id/sign"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_marginBottom="@dimen/gap_xxl"
      android:text="Profile"
      android:textColor="@color/white"
      android:textSize="@dimen/font_xxl"
      android:textStyle="bold" />

   <ImageView
      android:id="@+id/imgview"
      android:layout_width="100dp"
      android:layout_height="100dp"
      android:src="@drawable/logo"
      android:visibility="gone" />
   <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_gravity="left"
      android:text="Name"
      android:textStyle="bold"
      android:textColor="@color/white"
      android:layout_marginLeft="20dp"
      android:layout_marginTop="10dp"
      />
   <TextView
      android:id="@+id/ownername"
      android:layout_width="match_parent"
      android:layout_height="50dp"
      android:layout_margin="@dimen/gap_sm"
      android:layout_marginStart="@dimen/gap_xxl"
      android:layout_marginTop="@dimen/gap_xl"
      android:layout_marginEnd="@dimen/gap_xxl"
```

```
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:maxLines="1"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/black"
    android:textSize="@dimen/font_lg" />
  <TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="left"
    android:text="Email"
    android:textStyle="bold"
    android:textColor="@color/white"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="10dp"
    />
  <TextView
    android:id="@+id/owneremail"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_margin="@dimen/gap_sm"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginTop="@dimen/gap_xl"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:maxLines="1"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/black"
    android:textSize="@dimen/font_lg" />
  <TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="left"
    android:text="Phone"
    android:textStyle="bold"
    android:textColor="@color/white"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="10dp"
    />
  <TextView
    android:id="@+id/ownerphone"
    android:layout_width="match_parent"
    android:layout_height="50dp"
```

```
    android:layout_margin="@dimen/gap_sm"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginTop="@dimen/gap_xl"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:maxLines="1"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/black"
    android:textSize="@dimen/font_lg" />

  <LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="@dimen/gap_lg"
    android:layout_marginBottom="@dimen/gap_lg"
    android:gravity="center">

    <androidx.appcompat.widget.AppCompatButton
      android:id="@+id/bus"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_gravity="right"
      android:layout_marginRight="5dp"
      android:background="@drawable/app_btn_bg"
      android:elevation="4dp"
      android:gravity="center"
      android:text="Viewbuses"
      android:textAllCaps="false"
      android:textColor="@color/btn_text_color"
      android:textSize="16sp"
      android:visibility="gone" />

    <androidx.appcompat.widget.AppCompatButton
      android:id="@+id/room"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_gravity="center"
      android:background="@drawable/app_btn_bg"
      android:elevation="4dp"
      android:gravity="center"
      android:text="View Rooms"
      android:textAllCaps="false"
      android:textColor="@color/btn_text_color"
      android:textSize="16sp"
      android:visibility="gone" />
  </LinearLayout>
```

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

//fragment_add_bus.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
   tools:context=".ui.fragments.AddBusFragment">
   <ScrollView
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:background="@color/light_grey">

      <LinearLayout
         android:layout_width="match_parent"
         android:layout_height="match_parent"
         android:orientation="vertical"
         android:padding="@dimen/gap_sm"
         android:background="@color/light_grey">

         <TextView
            android:id="@+id/title"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="@dimen/gap_lg"
            android:layout_marginBottom="@dimen/gap_xl"
            android:gravity="center"
            android:text="Add Bus"
            android:textColor="@color/blue"
            android:textAllCaps="true"
            android:textSize="@dimen/font_xl"
            android:textStyle="bold" />

         <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:weightSum="2"
            android:layout_marginTop="15dp"
            android:orientation="horizontal">

            <EditText
               android:id="@+id/addbownerimage"
               android:layout_width="wrap_content"
```

```
        android:layout_height="60dp"
        android:layout_marginStart="@dimen/gap_xxl"
        android:layout_weight="2"
        android:layout_marginEnd="@dimen/gap_xxl"
        android:background="@drawable/edit_text_bg"
        android:drawablePadding="@dimen/gap_md"
        android:elevation="@dimen/gap_xsm"
        android:hint="Choose an image"
        android:inputType="textCapWords"
        android:paddingStart="@dimen/gap_lg"
        android:paddingEnd="@dimen/gap_lg"
        android:textColor="@color/blue"
        android:textColorHint="@color/blue" />

    <Button
        android:id="@+id/addbownerbtn_choose"
        android:layout_width="wrap_content"
        android:layout_gravity="center"
        android:layout_marginEnd="15dp"
        android:layout_height="wrap_content"
        android:text="CHOOSE" />

</LinearLayout>

<ImageView
    android:id="@+id/gallery_view"
    android:visibility="gone"
    android:layout_marginTop="10dp"
    android:layout_gravity="center"
    android:layout_width="100dp"
    android:layout_height="100dp"/>
<EditText
    android:id="@+id/addbusname"
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginTop="15dp"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:hint="Bus Name"
    android:inputType="textCapWords"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/black"
    android:textColorHint="@color/black" />
<EditText
    android:id="@+id/addbownerdesc"
```

```
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginTop="@dimen/gap_lg"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:gravity="start"
    android:hint="Description"
    android:inputType="textCapWords"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/black"
    android:textColorHint="@color/black" />

<EditText
    android:id="@+id/addbownerseat_no"
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginTop="15dp"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:hint="Total Seats"
    android:inputType="textCapWords"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/black"
    android:textColorHint="@color/black" />

<androidx.appcompat.widget.AppCompatButton
    android:id="@+id/addbowneradd_btn"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="bottom"
    android:layout_marginStart="@dimen/gap_xl"
    android:layout_marginTop="40dp"
    android:layout_marginEnd="@dimen/gap_xl"
    android:layout_marginBottom="20dp"
    android:background="@drawable/app_btn_bg"
    android:elevation="4dp"
    android:gravity="center"
    android:text="Add"
    android:textAllCaps="true"
    android:textColor="@color/btn_text_color"
```

```
            android:textSize="16sp"
            android:textStyle="bold" />
        </LinearLayout>
    </ScrollView>
</FrameLayout>
```

//fragment_add_hotel_rooms.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
    tools:context=".ui.fragments.AddBusFragment">
    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="vertical"
            android:padding="@dimen/gap_sm"
            android:background="@color/light_grey">

            <TextView
                android:id="@+id/title"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginTop="@dimen/gap_lg"
                android:layout_marginBottom="@dimen/gap_xl"
                android:gravity="center"
                android:text="Add Rooms"
                android:textColor="@color/blue"
                android:textSize="@dimen/font_xl"
                android:textStyle="bold" />

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:weightSum="2"
                android:layout_marginTop="15dp"
                android:orientation="horizontal">

                <EditText
                    android:id="@+id/addbownerimage"
                    android:layout_width="wrap_content"
                    android:layout_height="60dp"
```

```
        android:layout_marginStart="@dimen/gap_xxl"
        android:layout_weight="2"
        android:layout_marginEnd="@dimen/gap_xxl"
        android:background="@drawable/edit_text_bg"
        android:drawablePadding="@dimen/gap_md"
        android:elevation="@dimen/gap_xsm"
        android:hint="Choose an image"
        android:inputType="textCapWords"
        android:paddingStart="@dimen/gap_lg"
        android:paddingEnd="@dimen/gap_lg"
        android:textStyle="bold"
        android:textColor="@color/blue"
        android:textColorHint="@color/blue" />

    <Button
        android:id="@+id/addbownerbtn_choose"
        android:layout_width="wrap_content"
        android:layout_gravity="center"
        android:layout_marginEnd="15dp"
        android:layout_height="wrap_content"
        android:text="CHOOSE" />

</LinearLayout>

<ImageView
    android:id="@+id/gallery_view"
    android:visibility="gone"
    android:layout_marginTop="10dp"
    android:layout_gravity="center"
    android:layout_width="100dp"
    android:layout_height="100dp"/>
<EditText
    android:id="@+id/addbownerdesc"
    android:layout_width="match_parent"
    android:layout_height="120dp"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginTop="@dimen/gap_lg"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:gravity="start"
    android:hint="Description"
    android:inputType="textCapWords"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/black"
    android:textColorHint="@color/black" />
```

```
<EditText
   android:id="@+id/addbownerroom_no"
   android:layout_width="match_parent"
   android:layout_height="60dp"
   android:layout_marginStart="@dimen/gap_xxl"
   android:layout_marginTop="15dp"
   android:layout_marginEnd="@dimen/gap_xxl"
   android:background="@drawable/edit_text_bg"
   android:drawablePadding="@dimen/gap_md"
   android:elevation="@dimen/gap_xsm"
   android:hint="Room number"
   android:inputType="textCapWords"
   android:padding="@dimen/gap_md"
   android:paddingStart="@dimen/gap_lg"
   android:paddingEnd="@dimen/gap_lg"
   android:textColor="@color/black"
   android:textColorHint="@color/black" />

<EditText
   android:id="@+id/roomname"
   android:layout_width="match_parent"
   android:layout_height="60dp"
   android:layout_marginStart="@dimen/gap_xxl"
   android:layout_marginTop="15dp"
   android:layout_marginEnd="@dimen/gap_xxl"
   android:background="@drawable/edit_text_bg"
   android:drawablePadding="@dimen/gap_md"
   android:elevation="@dimen/gap_xsm"
   android:hint="Room name"
   android:inputType="textCapWords"
   android:padding="@dimen/gap_md"
   android:paddingStart="@dimen/gap_lg"
   android:paddingEnd="@dimen/gap_lg"
   android:textColor="@color/black"
   android:textColorHint="@color/black" />

<androidx.appcompat.widget.AppCompatButton
   android:id="@+id/addbowneradd_btn"
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:layout_gravity="bottom"
   android:layout_marginStart="@dimen/gap_xl"
   android:layout_marginTop="40dp"
   android:layout_marginEnd="@dimen/gap_xl"
   android:layout_marginBottom="20dp"
   android:background="@drawable/app_btn_bg"
   android:elevation="4dp"
   android:gravity="center"
   android:text="Add"
   android:textAllCaps="true"
```

```
            android:textColor="@color/btn_text_color"
            android:textSize="16sp"
            android:textStyle="bold" />
      </LinearLayout>
   </ScrollView>
</FrameLayout>
```

//activity_hotel_edit.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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="@color/app_bg"
   tools:context=".ui.fragments.AddBusFragment">
   <ScrollView
      android:layout_width="match_parent"
      android:layout_height="match_parent">

      <LinearLayout
         android:layout_width="match_parent"
         android:layout_height="match_parent"
         android:orientation="vertical"
         android:padding="@dimen/gap_sm">

         <TextView
            android:id="@+id/title"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="@dimen/gap_lg"
            android:layout_marginBottom="@dimen/gap_xl"
            android:gravity="center"
            android:text="Add Rooms"
            android:textAllCaps="true"
            android:textColor="@color/blue"
            android:textSize="@dimen/font_xl"
            android:textStyle="bold" />

         <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:weightSum="2"
            android:layout_marginTop="15dp"
            android:orientation="horizontal">

            <EditText
               android:id="@+id/addbownerimage"
```

```
        android:layout_width="wrap_content"
        android:layout_height="60dp"
        android:layout_marginStart="@dimen/gap_xxl"
        android:layout_weight="2"
        android:layout_marginEnd="@dimen/gap_xxl"
        android:background="@drawable/edit_text_bg"
        android:drawablePadding="@dimen/gap_md"
        android:hint="Choose an image"
        android:inputType="textCapWords"
        android:paddingStart="@dimen/gap_lg"
        android:paddingEnd="@dimen/gap_lg"
        android:textColor="@color/blue"
        android:textColorHint="@color/blue"
        android:textStyle="bold"/>

    <Button
        android:id="@+id/addbownerbtn_choose"
        android:layout_width="wrap_content"
        android:layout_gravity="center"
        android:layout_marginEnd="15dp"
        android:layout_height="wrap_content"
        android:text="CHOOSE" />

</LinearLayout>

<ImageView
    android:id="@+id/gallery_view"
    android:layout_marginTop="10dp"
    android:layout_gravity="center"
    android:layout_width="100dp"
    android:layout_height="100dp"/>
<EditText
    android:id="@+id/addbownerdesc"
    android:layout_width="match_parent"
    android:layout_height="120dp"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginTop="@dimen/gap_lg"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:gravity="start"
    android:hint="Description"
    android:inputType="textCapWords"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/blue"
    android:textColorHint="@color/black" />
```

```
<EditText
   android:id="@+id/addbownerroom_no"
   android:layout_width="match_parent"
   android:layout_height="60dp"
   android:layout_marginStart="@dimen/gap_xxl"
   android:layout_marginTop="15dp"
   android:layout_marginEnd="@dimen/gap_xxl"
   android:background="@drawable/edit_text_bg"
   android:drawablePadding="@dimen/gap_md"
   android:elevation="@dimen/gap_xsm"
   android:hint="Room number"
   android:inputType="textCapWords"
   android:padding="@dimen/gap_md"
   android:paddingStart="@dimen/gap_lg"
   android:paddingEnd="@dimen/gap_lg"
   android:textColor="@color/blue"
   android:textColorHint="@color/black" />

<EditText
   android:id="@+id/roomname"
   android:layout_width="match_parent"
   android:layout_height="60dp"
   android:layout_marginStart="@dimen/gap_xxl"
   android:layout_marginTop="15dp"
   android:layout_marginEnd="@dimen/gap_xxl"
   android:background="@drawable/edit_text_bg"
   android:drawablePadding="@dimen/gap_md"
   android:elevation="@dimen/gap_xsm"
   android:hint="Room name"
   android:inputType="textCapWords"
   android:padding="@dimen/gap_md"
   android:paddingStart="@dimen/gap_lg"
   android:paddingEnd="@dimen/gap_lg"
   android:textColor="@color/blue"
   android:textColorHint="@color/black" />

<androidx.appcompat.widget.AppCompatButton
   android:id="@+id/addbowneradd_btn"
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:layout_gravity="bottom"
   android:layout_marginStart="@dimen/gap_xl"
   android:layout_marginTop="40dp"
   android:layout_marginEnd="@dimen/gap_xl"
   android:layout_marginBottom="20dp"
   android:background="@drawable/app_btn_bg"
   android:elevation="4dp"
   android:gravity="center"
   android:text="Add"
   android:textAllCaps="true"
```

```
            android:textColor="@color/btn_text_color"
            android:textSize="16sp"
            android:textStyle="bold" />
      </LinearLayout>
   </ScrollView>
</FrameLayout>
```

//activity_edit_bus.xml

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

<FrameLayout 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:id="@+id/ebcontainer"
   android:background="@color/app_bg"
   >
   <ScrollView
      android:layout_width="match_parent"
      android:layout_height="match_parent">

      <LinearLayout
         android:layout_width="match_parent"
         android:layout_height="match_parent"
         android:orientation="vertical"
         android:padding="@dimen/gap_sm"
         >

         <TextView
            android:id="@+id/title"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="@dimen/gap_lg"
            android:layout_marginBottom="@dimen/gap_xl"
            android:gravity="center"
            android:text="Edit Bus Details"
            android:textColor="@color/blue"
            android:textSize="@dimen/font_xl"
            android:textStyle="bold" />

         <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:weightSum="2"
            android:layout_marginTop="15dp"
            android:orientation="horizontal">
```

```
    <EditText
        android:id="@+id/addbownerimage"
        android:layout_width="wrap_content"
        android:layout_height="60dp"
        android:layout_marginStart="@dimen/gap_xxl"
        android:layout_weight="2"
        android:layout_marginEnd="@dimen/gap_xxl"
        android:background="@drawable/edit_text_bg"
        android:drawablePadding="@dimen/gap_md"
        android:hint="Choose an image"
        android:inputType="textCapWords"
        android:paddingStart="@dimen/gap_lg"
        android:paddingEnd="@dimen/gap_lg"
        android:textColor="@color/blue"
        android:textColorHint="@color/blue" />

    <Button
        android:id="@+id/addbownerbtn_choose"
        android:layout_width="wrap_content"
        android:layout_gravity="center"
        android:layout_marginEnd="15dp"
        android:layout_height="wrap_content"
        android:text="CHOOSE" />

</LinearLayout>

<ImageView
    android:id="@+id/gallery_view"

    android:layout_marginTop="10dp"
    android:layout_gravity="center"
    android:layout_width="100dp"
    android:layout_height="100dp"/>

<EditText
    android:id="@+id/addbusname"
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginTop="15dp"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:hint="Bus Name"
    android:inputType="textCapWords"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/blue"
```

```
    android:textColorHint="@color/black" />

<EditText
    android:id="@+id/addbownerdesc"
    android:layout_width="match_parent"
    android:layout_height="120dp"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginTop="@dimen/gap_lg"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:gravity="start"
    android:hint="Description"
    android:inputType="textCapWords"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/blue"
    android:textColorHint="@color/black" />

<EditText
    android:id="@+id/addbownerseat_no"
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginTop="15dp"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:hint="Total Seats"
    android:inputType="textCapWords"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/blue"
    android:textColorHint="@color/black" />

<androidx.appcompat.widget.AppCompatButton
    android:id="@+id/addbowneradd_btn"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="bottom"
    android:layout_marginStart="@dimen/gap_xl"
    android:layout_marginTop="40dp"
    android:layout_marginEnd="@dimen/gap_xl"
    android:layout_marginBottom="20dp"
    android:background="@drawable/app_btn_bg"
    android:elevation="4dp"
```

```
            android:gravity="center"
            android:text="Edit"
            android:textAllCaps="true"
            android:textColor="@color/btn_text_color"
            android:textSize="16sp"
            android:textStyle="bold" />
        </LinearLayout>
    </ScrollView>
</FrameLayout>
```

//view_buses_busowners.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"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <androidx.cardview.widget.CardView
        android:id="@+id/parrent_layout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:elevation="@dimen/gap_sm"
        app:cardBackgroundColor="@color/app_bg"
        app:cardCornerRadius="@dimen/gap_md"
        android:layout_margin="@dimen/gap_sm"
        android:padding="@dimen/gap_md">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="vertical"
            android:gravity="center"
            android:padding="@dimen/gap_md"
            >

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="horizontal"
                android:gravity="center"
                android:background="@color/app_bg">

                <androidx.cardview.widget.CardView
                    android:layout_width="80dp"
                    android:layout_height="80dp"
                    android:background="@color/white"
                    android:elevation="@dimen/gap_sm"
```

```
    app:cardBackgroundColor="@color/white"
    app:cardCornerRadius="@dimen/gap_xxl"
    android:layout_margin="@dimen/gap_sm">

    <ImageView
        android:id="@+id/picture"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:scaleType="fitXY"/>

</androidx.cardview.widget.CardView>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">
    <TextView
        android:id="@+id/Busname"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:paddingEnd="@dimen/gap_xsm"
        android:text="Bus Name"
        android:textColor="@color/blue"
        android:textSize="25dp"
        android:textStyle="bold" />


    <TextView
        android:id="@+id/id"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textStyle="bold"
        android:gravity="center"
        android:paddingEnd="@dimen/gap_xsm"
        android:visibility="gone"
        android:text="Spot Name"
        android:textSize="@dimen/font_lg"
        android:textColor="@color/text_color" />
    <TextView
        android:id="@+id/totst"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textStyle="bold"
        android:gravity="center"
        android:paddingEnd="@dimen/gap_xsm"
        android:visibility="gone"
        android:text="Spot Name"
        android:textSize="@dimen/font_lg"
        android:textColor="@color/text_color" />
```

```
            <LinearLayout
                android:layout_width="match_parent"
                android:orientation="horizontal"
                android:weightSum="2"
                android:layout_gravity="center"
                android:layout_height="wrap_content">
                <TextView
                    android:id="@+id/available_text"
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:gravity="end"
                    android:visibility="gone"
                    android:layout_weight="1"
                    android:text="Available Seats : "
                    android:textStyle="bold"
                    android:textSize="@dimen/font_lg"
                    android:textColor="@color/recived_tr_clr" />

                <TextView
                    android:id="@+id/available"
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:textAlignment="textStart"
                    android:visibility="gone"
                    android:layout_weight="1"
                    android:text="0"
                    android:textStyle="bold"
                    android:textSize="@dimen/font_lg"
                    android:textColor="@color/recived_tr_clr" />
            </LinearLayout>
        </LinearLayout>
    </LinearLayout>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="left"
        android:text="Description"
        android:textColor="@color/blue"
        android:textStyle="bold"/>
    <TextView
        android:id="@+id/descr"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:layout_gravity="left|center_vertical"
        android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
        android:textColor="@color/text_color_light"
```

```
            android:textSize="15dp"/>
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="left"
            android:text="Seats"
            android:textStyle="bold"
            android:textColor="@color/blue"
            />
        <TextView
            android:id="@+id/availableseat"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/text_color_light"
            android:textSize="25dp"/>
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="@dimen/gap_lg"
            android:layout_marginBottom="@dimen/gap_lg"
            android:gravity="center"
            >

            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/edit"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_gravity="right"
                android:background="@drawable/app_btn_bg"
                android:elevation="4dp"
                android:layout_marginRight="5dp"
                android:gravity="center"
                android:text="Edit"
                android:textAllCaps="false"
                android:textColor="@color/btn_text_color"
                android:textSize="16sp"/>

            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/delete"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_gravity="right"
                android:background="@drawable/app_btn_bg"
                android:elevation="4dp"
                android:gravity="center"
                android:text="Delete"
```

```
            android:textAllCaps="false"
            android:textColor="@color/btn_text_color"
            android:textSize="16sp" />

        </LinearLayout>

        <LinearLayout
          android:layout_width="match_parent"
          android:layout_height="wrap_content"
          android:orientation="horizontal"
          android:gravity="center"
          android:weightSum="2"
          android:layout_marginTop="@dimen/gap_md">
        </LinearLayout>
      </LinearLayout>
    </androidx.cardview.widget.CardView>
</LinearLayout>
```

//view_hotel_roomsbyowner.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="wrap_content">

    <androidx.cardview.widget.CardView
      android:id="@+id/parrent_layout"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:elevation="@dimen/gap_sm"
      app:cardBackgroundColor="@color/white"
      app:cardCornerRadius="@dimen/gap_md"
      android:layout_margin="@dimen/gap_sm"
      android:padding="@dimen/gap_md">

      <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:gravity="center"
        android:padding="@dimen/gap_md"
        android:background="@color/app_bg"
        >

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

```
android:layout_height="wrap_content"
android:orientation="horizontal"
android:gravity="center"
android:background="@color/app_bg"
>

<androidx.cardview.widget.CardView
   android:layout_width="80dp"
   android:layout_height="80dp"
   android:background="@color/white"
   android:elevation="@dimen/gap_sm"
   app:cardBackgroundColor="@color/white"
   app:cardCornerRadius="@dimen/gap_xxl"
   android:layout_margin="@dimen/gap_sm">

   <ImageView
      android:id="@+id/pictureroom"
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:scaleType="fitXY"
      android:clickable="true"
      />

</androidx.cardview.widget.CardView>

<LinearLayout
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:orientation="vertical">
   <TextView
      android:id="@+id/roomnames"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:gravity="center"
      android:paddingEnd="@dimen/gap_xsm"
      android:text="Room Type"
      android:textColor="@color/blue"
      android:textSize="25dp"
      android:textStyle="bold" />

   <TextView
      android:id="@+id/id"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:textStyle="bold"
      android:gravity="center"
      android:paddingEnd="@dimen/gap_xsm"
      android:visibility="gone"
      android:text="Spot Name"
      android:textSize="@dimen/font_lg"
```

```
            android:textColor="@color/text_color" />
        <TextView
            android:id="@+id/totst"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textStyle="bold"
            android:gravity="center"
            android:paddingEnd="@dimen/gap_xsm"
            android:visibility="gone"
            android:text="Spot Name"
            android:textSize="@dimen/font_lg"
            android:textColor="@color/text_color" />

        <LinearLayout
            android:layout_width="match_parent"
            android:orientation="horizontal"
            android:weightSum="2"
            android:layout_gravity="center"
            android:layout_height="wrap_content">
            <TextView
                android:id="@+id/available_text"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:gravity="end"
                android:visibility="gone"
                android:layout_weight="1"
                android:text="Available Seats : "
                android:textStyle="bold"
                android:textSize="@dimen/font_lg"
                android:textColor="@color/recived_tr_clr" />

            <TextView
                android:id="@+id/available"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:textAlignment="textStart"
                android:visibility="gone"
                android:layout_weight="1"
                android:text="0"
                android:textStyle="bold"
                android:textSize="@dimen/font_lg"
                android:textColor="@color/recived_tr_clr" />
        </LinearLayout>
    </LinearLayout>
</LinearLayout>
<TextView
    android:id="@+id/descrip"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="left|center_vertical"
```

```
            android:gravity="center"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/text_color_light"
            android:textSize="15dp"
            tools:ignore="TextSizeCheck" />
        <TextView
            android:id="@+id/roomno"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/text_color_light"
            android:textSize="15dp"/>

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:gravity="center"
            android:weightSum="2"
            android:layout_marginTop="@dimen/gap_md"
            >

            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/edit"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_gravity="center"
                android:background="@drawable/app_btn_bg"
                android:elevation="4dp"
                android:layout_marginRight="5dp"
                android:gravity="center"
                android:text="Edit"
                android:textAllCaps="false"
                android:textColor="@color/btn_text_color"
                android:textSize="16sp"/>
            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/delete"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_gravity="center"
                android:background="@drawable/app_btn_bg2"
                android:elevation="4dp"
                android:layout_marginRight="5dp"
                android:gravity="center"
                android:text="Delete"
                android:textAllCaps="false"
```

```
            android:textColor="@color/btn_text_color"
            android:textSize="16sp"/>
        </LinearLayout>
      </LinearLayout>
    </androidx.cardview.widget.CardView>
</LinearLayout>
```

//fragment_admin_view_buses.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    tools:context=".ui.fragments.ProfileFragment">

    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:gravity="center"
            android:padding="@dimen/gap_md"
            android:orientation="vertical">

            <androidx.cardview.widget.CardView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginStart="@dimen/gap_lg"
                android:layout_marginEnd="@dimen/gap_lg"
                app:cardCornerRadius="@dimen/gap_xl"
                android:elevation="@dimen/gap_sm">


            </androidx.cardview.widget.CardView>

            <androidx.cardview.widget.CardView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                app:cardCornerRadius="@dimen/gap_xl"
                android:layout_marginTop="@dimen/gap_xxl"
                android:elevation="@dimen/gap_sm">

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

```
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:background="@color/white"
            android:gravity="center">

            <LinearLayout
               android:layout_width="match_parent"
               android:layout_height="wrap_content"
               android:orientation="horizontal"
               android:gravity="center"
               android:layout_margin="@dimen/gap_md">



            </LinearLayout>


            <androidx.recyclerview.widget.RecyclerView
               android:id="@+id/adminbusview"
               android:layout_width="match_parent"
               android:layout_height="wrap_content"
               android:visibility="visible"/>
            <TextView
               android:id="@+id/no_data"
               android:layout_width="match_parent"
               android:layout_height="match_parent"
               android:text="No Data"
               android:visibility="gone"
               android:textColor="@color/app_color"
               android:textSize="@dimen/font_xl"
               android:gravity="center"/>
         </LinearLayout>
      </androidx.cardview.widget.CardView>
   </LinearLayout>
  </ScrollView>
</FrameLayout>
```

//fragment_admin_view_rooms.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   tools:context=".ui.fragments.ProfileFragment">

   <ScrollView
```

```
android:layout_width="match_parent"
android:layout_height="match_parent">

<LinearLayout
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:gravity="center"
   android:padding="@dimen/gap_md"
   android:orientation="vertical">

   <androidx.cardview.widget.CardView
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:layout_marginStart="@dimen/gap_lg"
      android:layout_marginEnd="@dimen/gap_lg"
      app:cardCornerRadius="@dimen/gap_xl"
      android:elevation="@dimen/gap_sm">


   </androidx.cardview.widget.CardView>

   <androidx.cardview.widget.CardView
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      app:cardCornerRadius="@dimen/gap_xl"
      android:layout_marginTop="@dimen/gap_xxl"
      android:elevation="@dimen/gap_sm">

      <LinearLayout
         android:layout_width="match_parent"
         android:layout_height="wrap_content"
         android:orientation="vertical"
         android:background="@color/white"
         android:gravity="center">

         <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:gravity="center"
            android:layout_margin="@dimen/gap_md">

         </LinearLayout>

         <androidx.recyclerview.widget.RecyclerView
            android:id="@+id/adminroomsview"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:visibility="visible"/>
         <TextView
```

```
                android:id="@+id/no_data"
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:text="No Data"
                android:visibility="gone"
                android:textColor="@color/app_color"
                android:textSize="@dimen/font_xl"
                android:gravity="center"/>
            </LinearLayout>
          </androidx.cardview.widget.CardView>
        </LinearLayout>
    </ScrollView>
</FrameLayout>
```

//fragment_admin_package.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/container1"
    tools:context=".ui.fragments.AdminPackageFragment">

    <Button
        android:id="@+id/add_package"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="top"
        android:gravity="center"
        android:text="Add Package"
        />
    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:gravity="center"
            android:orientation="vertical"
            android:padding="@dimen/gap_md">
            <androidx.cardview.widget.CardView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginStart="@dimen/gap_lg"
                android:layout_marginEnd="@dimen/gap_lg"
                android:elevation="@dimen/gap_sm"
```

```
    app:cardCornerRadius="@dimen/gap_xl">
</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="40dp"
    android:elevation="@dimen/gap_sm"
    app:cardCornerRadius="@dimen/gap_xl">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:background="@color/white"
        android:gravity="center"
        android:orientation="vertical">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_margin="@dimen/gap_md"
            android:gravity="center"
            android:orientation="horizontal">

            <TextView
                android:id="@+id/packages"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_margin="@dimen/gap_md"
                android:background="@drawable/edit_text_bg"
                android:elevation="@dimen/gap_lg"
                android:padding="@dimen/gap_md"
                android:text="Packages"
                android:textColor="@color/blue"
                android:textSize="@dimen/font_vlg"
                android:textStyle="bold" />

            <TextView
                android:id="@+id/bookings"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_margin="@dimen/gap_md"
                android:background="@drawable/edit_text_bg"
                android:elevation="@dimen/gap_lg"
                android:padding="@dimen/gap_md"
                android:text="Bookings"
                android:textColor="@color/text_color"
                android:textSize="@dimen/font_vlg"
                android:textStyle="bold" />
```

```
            </LinearLayout>

            <androidx.recyclerview.widget.RecyclerView
               android:id="@+id/rv_package"
               android:layout_width="match_parent"
               android:layout_height="wrap_content"
               android:visibility="visible" />

            <TextView
               android:id="@+id/no_data"
               android:layout_width="match_parent"
               android:layout_height="match_parent"
               android:gravity="center"
               android:text="No Data"
               android:textColor="@color/app_color"
               android:textSize="@dimen/font_xl"
               android:visibility="gone" />
         </LinearLayout>
      </androidx.cardview.widget.CardView>
   </LinearLayout>
  </ScrollView>
</FrameLayout>
```

//fragment_admin_view_package.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
   tools:context=".ui.fragments.AdminViewPackageFragment">

   <TextView
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:text="@string/hello_blank_fragment" />

</FrameLayout>
```

//adminviewbuses.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"
   android:layout_width="match_parent"
   android:layout_height="wrap_content">
```

```
<androidx.cardview.widget.CardView
   android:id="@+id/parrent_layout"
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:elevation="@dimen/gap_sm"
   app:cardBackgroundColor="@color/white"
   app:cardCornerRadius="@dimen/gap_md"
   android:layout_margin="@dimen/gap_sm"
   android:padding="@dimen/gap_md">

   <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:orientation="vertical"
      android:gravity="center"
      android:padding="@dimen/gap_md">

      <LinearLayout
         android:layout_width="match_parent"
         android:layout_height="wrap_content"
         android:orientation="horizontal"
         android:gravity="center"
         android:background="@color/white">

         <androidx.cardview.widget.CardView
            android:layout_width="80dp"
            android:layout_height="80dp"
            android:background="@color/white"
            android:elevation="@dimen/gap_sm"
            app:cardBackgroundColor="@color/white"
            app:cardCornerRadius="@dimen/gap_xxl"
            android:layout_margin="@dimen/gap_sm">

            <ImageView
               android:id="@+id/picture"
               android:layout_width="match_parent"
               android:layout_height="match_parent"
               android:scaleType="fitXY"/>

         </androidx.cardview.widget.CardView>

         <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">
            <TextView
               android:id="@+id/Busname"
               android:layout_width="match_parent"
               android:layout_height="wrap_content"
```

```
    android:gravity="center"
    android:paddingEnd="@dimen/gap_xsm"
    android:text="Bus Name"
    android:textColor="@color/blue"
    android:textSize="25dp"
    android:textStyle="bold" />

  <TextView
    android:id="@+id/id"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textStyle="bold"
    android:gravity="center"
    android:paddingEnd="@dimen/gap_xsm"
    android:visibility="gone"
    android:text="Spot Name"
    android:textSize="@dimen/font_lg"
    android:textColor="@color/text_color" />
  <TextView
    android:id="@+id/totst"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textStyle="bold"
    android:gravity="center"
    android:paddingEnd="@dimen/gap_xsm"
    android:visibility="gone"
    android:text="Spot Name"
    android:textSize="@dimen/font_lg"
    android:textColor="@color/text_color" />

  <LinearLayout
    android:layout_width="match_parent"
    android:orientation="horizontal"
    android:weightSum="2"
    android:layout_gravity="center"
    android:layout_height="wrap_content">
    <TextView
      android:id="@+id/available_text"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:gravity="end"
      android:visibility="gone"
      android:layout_weight="1"
      android:text="Available Seats : "
      android:textStyle="bold"
      android:textSize="@dimen/font_lg"
      android:textColor="@color/recived_tr_clr" />
    <TextView
      android:id="@+id/available"
      android:layout_width="match_parent"
```

```
                android:layout_height="wrap_content"
                android:textAlignment="textStart"
                android:visibility="gone"
                android:layout_weight="1"
                android:text="0"
                android:textStyle="bold"
                android:textSize="@dimen/font_lg"
                android:textColor="@color/recived_tr_clr" />
            </LinearLayout>
          </LinearLayout>
        </LinearLayout>
        <TextView
          android:layout_width="wrap_content"
          android:layout_height="wrap_content"
          android:layout_gravity="left"
          android:text="Description"
          android:textColor="@color/blue"
          android:textStyle="bold"/>
        <TextView
          android:id="@+id/descr"
          android:layout_width="wrap_content"
          android:layout_height="wrap_content"
          android:gravity="center"
          android:layout_gravity="left|center_vertical"
          android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
          android:textColor="@color/text_color_light"
          android:textSize="15dp"/>
        <TextView
          android:layout_width="wrap_content"
          android:layout_height="wrap_content"
          android:layout_gravity="left"
          android:text="Seats"
          android:textColor="@color/blue"
          android:textStyle="bold"/>
        <TextView
          android:id="@+id/availableseat"
          android:layout_width="wrap_content"
          android:layout_height="wrap_content"
          android:gravity="center"
          android:layout_gravity="left|center_vertical"
          android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
          android:textColor="@color/text_color_light"
          android:textSize="25dp"/>

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

```
            android:gravity="center"
            android:weightSum="2"
            android:layout_marginTop="@dimen/gap_md">
        </LinearLayout>
    </LinearLayout>
  </androidx.cardview.widget.CardView>
</LinearLayout>
```

//activity_edit_package.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"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@android:color/white"
    android:orientation="vertical">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:padding="10dp"
        android:textColor="@color/blue"
        android:text="ASSIGN TRIPS"
        android:textStyle="bold" />
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:weightSum="2"
        android:layout_marginTop="15dp"
        android:orientation="horizontal">
        <Button
            android:id="@+id/img"
            android:layout_width="wrap_content"
            android:layout_gravity="center"
            android:layout_marginEnd="15dp"
            android:layout_height="wrap_content"
            android:text="CHOOSE another image"/>
        <ImageView
            android:id="@+id/gallery_view"

            android:layout_marginTop="10dp"
            android:layout_gravity="center"
            android:layout_width="100dp"
            android:layout_height="100dp"/>
    </LinearLayout>
    <com.toptoche.searchablespinnerlibrary.SearchableSpinner
```

```
    android:id="@+id/busedit"
    android:layout_width="match_parent"
    android:layout_height="40dp"
    android:soundEffectsEnabled="true"
    app:hintText="Please choose a bus" />

<com.toptoche.searchablespinnerlibrary.SearchableSpinner
    android:id="@+id/hotel"
    android:layout_width="match_parent"
    android:layout_height="40dp"
    android:layout_marginTop="10dp"
    android:soundEffectsEnabled="true"
    app:hintText="Please choose a Hotel" />
<com.toptoche.searchablespinnerlibrary.SearchableSpinner
    android:id="@+id/rooms"
    android:layout_width="match_parent"
    android:layout_height="40dp"
    android:layout_marginTop="10dp"
    android:soundEffectsEnabled="true"
    app:hintText="Please choose a room" />

<EditText
    android:id="@+id/allot_bpoint"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:hint="Enter cost per head"
    android:padding="10dp"
    android:inputType="number"
    android:textColorHint="@color/black" />
<TextView
    android:id="@+id/startdate"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:hint="Select start date"
    android:gravity="center"
    android:padding="10dp"
    android:layout_marginLeft="20dp"
    android:background="@color/app_color_secondary_light"
    android:textColorHint="@color/black" />
<TextView
    android:id="@+id/enddate"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:gravity="center"

    android:layout_marginTop="10dp"
    android:hint="Select end date"
    android:padding="10dp"
```

```
    android:layout_marginLeft="20dp"
    android:background="@color/app_color_secondary_light"
    android:textColorHint="@color/black" />
  <EditText
    android:id="@+id/editfrom"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:hint="From"
    android:padding="10dp"
    android:textColorHint="@color/black" />
  <EditText
    android:id="@+id/editto"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:hint="To"
    android:padding="10dp"
    android:textColorHint="@color/black" />
  <EditText
    android:id="@+id/pdesc"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:hint="Description"
    android:padding="10dp"
    android:textColorHint="@color/black" />
  <EditText
    android:id="@+id/avlsts"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:hint="Available "
    android:padding="10dp"
    android:textColorHint="@color/black" />
  <androidx.appcompat.widget.AppCompatButton
    android:id="@+id/submitnew"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginTop="10dp"
    android:background="@drawable/app_btn_secondry_bg"
    android:text="Update"
    android:textAllCaps="true"
    android:textColor="@color/blue" />

</LinearLayout>
```

//activity_tripassign.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"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@android:color/white"
    android:orientation="vertical">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:padding="10dp"
        android:textColor="@color/black"
        android:text="ASSIGN TRIPS"
        android:textStyle="bold" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:weightSum="2"
        android:layout_marginTop="15dp"
        android:orientation="horizontal"
        >
    <Button
        android:id="@+id/img"
        android:layout_width="300dp"
        android:layout_marginEnd="15dp"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:text="Click to choose an image" />
    <ImageView
        android:id="@+id/gallery_view"
        android:visibility="gone"
        android:layout_marginTop="10dp"
        android:layout_gravity="center"
        android:layout_width="100dp"
        android:layout_height="100dp"/>
    </LinearLayout>

    <com.toptoche.searchablespinnerlibrary.SearchableSpinner
        android:id="@+id/hj"
        android:layout_width="match_parent"
        android:layout_height="40dp"
        android:soundEffectsEnabled="true"
        app:hintText="Please choose a bus"
        android:layout_marginTop="20dp"
        />

    <com.toptoche.searchablespinnerlibrary.SearchableSpinner
```

```
    android:id="@+id/hotel"
    android:layout_width="match_parent"
    android:layout_height="40dp"
    android:layout_marginTop="10dp"
    android:soundEffectsEnabled="true"
    app:hintText="Please choose a Hotel" />
<com.toptoche.searchablespinnerlibrary.SearchableSpinner
    android:id="@+id/rooms"
    android:layout_width="match_parent"
    android:layout_height="40dp"
    android:layout_marginTop="10dp"
    android:soundEffectsEnabled="true"
    app:hintText="Please choose a room" />

<EditText
    android:id="@+id/allot_bpoint"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:hint="Enter cost per head"
    android:inputType="number"
    android:padding="10dp"
    android:textColorHint="@color/black" />

<TextView
    android:id="@+id/startdate"
    android:layout_width="200dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:layout_marginLeft="20dp"
    android:hint="Select starting date"
    android:textColor="@color/black"
    android:padding="10dp"
    android:textColorHint="@color/black"
    android:background="@color/app_color_secondary_light"/>
<TextView
    android:id="@+id/enddate"
    android:layout_width="200dp"
    android:layout_height="wrap_content"
    android:background="@color/app_color_secondary_light"
    android:layout_marginTop="20dp"
    android:layout_marginLeft="20dp"
    android:hint="Select ending date"
    android:textColor="@color/black"

    android:padding="10dp"
    android:textColorHint="@color/black" />
<EditText
    android:id="@+id/from"
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:hint="From"
    android:padding="10dp"
    android:textColorHint="@color/black" />
  <EditText
    android:id="@+id/to"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:hint="To"
    android:padding="10dp"
    android:textColorHint="@color/black" />

  <EditText
    android:id="@+id/pdesc"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:hint="Description"
    android:padding="10dp"
    android:textColorHint="@color/black" />

  <EditText
    android:id="@+id/avlsts"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:enabled="false"
    android:hint="Available seats"
    android:padding="10dp"
    android:textColorHint="@color/black" />

  <androidx.appcompat.widget.AppCompatButton
    android:id="@+id/submitnew"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginTop="10dp"
    android:background="@drawable/app_btn_bg"
    android:text="ASSIGN"
    android:textColor="@color/white" />

</LinearLayout>
```

//fragment_view_hotel_rooms.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    tools:context=".ui.fragments.ViewHotelRoomsFragment">

    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:gravity="center"
            android:padding="@dimen/gap_md"
            android:orientation="vertical">

            <androidx.cardview.widget.CardView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginStart="@dimen/gap_lg"
                android:layout_marginEnd="@dimen/gap_lg"
                app:cardCornerRadius="@dimen/gap_xl"
                android:elevation="@dimen/gap_sm">

            </androidx.cardview.widget.CardView>

            <androidx.cardview.widget.CardView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                app:cardCornerRadius="@dimen/gap_xl"
                android:layout_marginTop="@dimen/gap_xxl"
                android:elevation="@dimen/gap_sm">

                <LinearLayout
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:orientation="vertical"
                    android:background="@color/white"
                    android:gravity="center">

                    <LinearLayout
                        android:layout_width="match_parent"
                        android:layout_height="wrap_content"
                        android:orientation="horizontal"
                        android:gravity="center"
                        android:layout_margin="@dimen/gap_md">
```

```
            </LinearLayout>

            <androidx.recyclerview.widget.RecyclerView
               android:id="@+id/rv_roomsview"

               android:layout_width="match_parent"
               android:layout_height="wrap_content"
               android:visibility="visible"/>
            <TextView
               android:id="@+id/no_data"
               android:layout_width="match_parent"
               android:layout_height="match_parent"
               android:text="No Data"
               android:visibility="gone"
               android:textColor="@color/blue"
               android:textSize="@dimen/font_xl"
               android:gravity="center"/>

         </LinearLayout>
      </androidx.cardview.widget.CardView>
   </LinearLayout>
  </ScrollView>
</FrameLayout>
```

//fragment_view_buses.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   android:id="@+id/frag"
   tools:context=".ui.fragments.ViewBusesFragment">

   <ScrollView
      android:layout_width="match_parent"
      android:layout_height="match_parent">

      <LinearLayout
         android:layout_width="match_parent"
         android:layout_height="match_parent"
         android:gravity="center"
         android:padding="@dimen/gap_md"
         android:orientation="vertical">

         <androidx.cardview.widget.CardView
            android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
        android:layout_marginStart="@dimen/gap_lg"
        android:layout_marginEnd="@dimen/gap_lg"
        app:cardCornerRadius="@dimen/gap_xl"
        android:elevation="@dimen/gap_sm">

    </androidx.cardview.widget.CardView>

    <androidx.cardview.widget.CardView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:cardCornerRadius="@dimen/gap_xl"
        android:layout_marginTop="@dimen/gap_xxl"
        android:elevation="@dimen/gap_sm">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:background="@color/white"
            android:gravity="center">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="horizontal"
                android:gravity="center"
                android:layout_margin="@dimen/gap_md">

            </LinearLayout>

            <androidx.recyclerview.widget.RecyclerView
                android:id="@+id/rv_busesview"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:visibility="visible"/>
            <TextView
                android:id="@+id/no_data"
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:text="No Data"
                android:visibility="gone"
                android:textColor="@color/blue"
                android:textSize="@dimen/font_xl"
                android:gravity="center"/>
        </LinearLayout>
    </androidx.cardview.widget.CardView>
</LinearLayout>

</ScrollView>
```

```
</FrameLayout>
```

//fragment_users_view.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   android:id="@+id/busfrag"
   >

   <ScrollView
      android:layout_width="match_parent"
      android:layout_height="match_parent">

      <LinearLayout
         android:layout_width="match_parent"
         android:layout_height="match_parent"
         android:gravity="center"
         android:padding="@dimen/gap_md"
         android:orientation="vertical">

         <androidx.cardview.widget.CardView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginStart="@dimen/gap_lg"
            android:layout_marginEnd="@dimen/gap_lg"
            app:cardCornerRadius="@dimen/gap_xl"
            android:elevation="@dimen/gap_sm">

         </androidx.cardview.widget.CardView>

         <androidx.cardview.widget.CardView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            app:cardCornerRadius="@dimen/gap_xl"
            android:layout_marginTop="@dimen/gap_xxl"
            android:elevation="@dimen/gap_sm">

            <LinearLayout
               android:layout_width="match_parent"
               android:layout_height="wrap_content"
               android:orientation="vertical"
               android:background="@color/white"
               android:gravity="center">
```

```
<LinearLayout
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:orientation="horizontal"
   android:gravity="center"
   android:layout_margin="@dimen/gap_md">

   <TextView
      android:id="@+id/Bus_Owners"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Bus Owners"
      android:textStyle="bold"
      android:textColor="@color/blue"
      android:textSize="@dimen/font_vlg"
      android:background="@drawable/edit_text_bg"
      android:padding="@dimen/gap_md"
      android:layout_margin="@dimen/gap_md"
      android:elevation="@dimen/gap_lg"/>

   <TextView
      android:id="@+id/Hotel_Owners"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Hotel Owners"
      android:textStyle="bold"
      android:textColor="@color/black"
      android:textSize="@dimen/font_vlg"
      android:background="@drawable/edit_text_bg"
      android:padding="@dimen/gap_md"
      android:layout_margin="@dimen/gap_md"
      android:elevation="@dimen/gap_lg"/>
   <TextView
      android:id="@+id/Users"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Users"
      android:textStyle="bold"
      android:textColor="@color/black"
      android:textSize="@dimen/font_vlg"
      android:background="@drawable/edit_text_bg"
      android:padding="@dimen/gap_md"
      android:layout_margin="@dimen/gap_md"
      android:elevation="@dimen/gap_lg"/>

</LinearLayout>

<androidx.recyclerview.widget.RecyclerView
   android:id="@+id/rv_users"
```

```
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:visibility="visible"/>
        <TextView
            android:id="@+id/no_data"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:text="No Data"
            android:visibility="gone"
            android:textColor="@color/black"
            android:textSize="@dimen/font_xl"
            android:gravity="center"/>
      </LinearLayout>
    </androidx.cardview.widget.CardView>
  </LinearLayout>

 </ScrollView>

</FrameLayout>
```

//fragment_users_request.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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:id="@+id/requestcontainer"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   tools:context=".ui.fragments.ProfileFragment">
   <TextView
      android:id="@+id/History"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="VIEW REJECTION HISTORY"
      android:textStyle="bold"
      android:textColor="#ffffff"
      android:textSize="@dimen/font_vlg"
      android:background="@color/app_color"
      android:padding="@dimen/gap_md"
      android:layout_margin="@dimen/gap_md"
      android:elevation="@dimen/gap_lg"
    />


   <ScrollView
      android:layout_width="match_parent"
      android:layout_height="match_parent">
```

```
<LinearLayout
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:gravity="center"
   android:padding="@dimen/gap_md"
   android:orientation="vertical">

   <androidx.cardview.widget.CardView
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:layout_marginStart="@dimen/gap_lg"
      android:layout_marginEnd="@dimen/gap_lg"
      app:cardCornerRadius="@dimen/gap_xl"
      android:elevation="@dimen/gap_sm">

   </androidx.cardview.widget.CardView>

   <androidx.cardview.widget.CardView
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      app:cardCornerRadius="@dimen/gap_xl"
      android:layout_marginTop="40dp"
      android:elevation="@dimen/gap_sm">

      <LinearLayout
         android:layout_width="match_parent"
         android:layout_height="wrap_content"
         android:orientation="vertical"
         android:background="@color/white"
         android:gravity="center">

         <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:gravity="center"
            android:layout_margin="@dimen/gap_md">

            <TextView
               android:id="@+id/Bus_Owners"
               android:layout_width="wrap_content"
               android:layout_height="wrap_content"
               android:text="Bus Owners"
               android:textStyle="bold"
               android:textColor="@color/blue"
               android:textSize="@dimen/font_vlg"
               android:background="@drawable/edit_text_bg"
               android:padding="@dimen/gap_md"
               android:layout_margin="@dimen/gap_md"
```

```
            android:elevation="@dimen/gap_lg"/>

        <TextView
            android:id="@+id/Hotel_Owners"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Hotel Owners"
            android:textStyle="bold"
            android:textColor="@color/black"
            android:textSize="@dimen/font_vlg"
            android:background="@drawable/edit_text_bg"
            android:padding="@dimen/gap_md"
            android:layout_margin="@dimen/gap_md"
            android:elevation="@dimen/gap_lg"/>

    </LinearLayout>

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/rv_request"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:visibility="visible"/>
    <TextView
        android:id="@+id/no_data"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:text="No Data"
        android:visibility="gone"
        android:textColor="@color/black"
        android:textSize="@dimen/font_xl"
        android:gravity="center"/>
  </LinearLayout>
 </androidx.cardview.widget.CardView>
 </LinearLayout>
 </ScrollView>
</FrameLayout>
```

//user_view_bookings.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"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <androidx.cardview.widget.CardView
        android:id="@+id/parrent_layout"
        android:layout_width="match_parent"
```

```
android:layout_height="wrap_content"
android:elevation="@dimen/gap_sm"
app:cardBackgroundColor="@color/white"
app:cardCornerRadius="@dimen/gap_md"
android:layout_margin="@dimen/gap_sm"
android:padding="@dimen/gap_md">

<LinearLayout
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  android:gravity="center"
  android:orientation="vertical"
  android:padding="@dimen/gap_md"
  android:background="@color/light_grey">
  <TextView
    android:id="@+id/spotname"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:paddingEnd="@dimen/gap_xsm"
    android:text="Spot Name"
    android:textAllCaps="true"
    android:textColor="@color/white"
    android:textSize="25dp"
    android:textStyle="bold" />

  <LinearLayout
    android:layout_width="match_parent"
    android:layout_height="208dp"
    android:gravity="center"
    android:orientation="horizontal">

    <ImageView
      android:id="@+id/picture"
      android:layout_width="match_parent"
      android:layout_height="200dp"
      android:background="@color/light_grey"
      android:scaleType="fitXY"
      android:layout_marginTop="10dp"/>

  </LinearLayout>

  <TextView
    android:id="@+id/frmto"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_margin="@dimen/gap_sm"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
```

```
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:maxLines="1"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/black"
    android:textSize="15dp" />

  <TextView
    android:id="@+id/sdateedate"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_margin="@dimen/gap_sm"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:maxLines="1"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/black"
    android:textSize="15dp" />

  <TextView
    android:id="@+id/bushotel"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_margin="@dimen/gap_sm"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:maxLines="1"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/black"
    android:textSize="15dp" />
  <TextView
    android:id="@+id/HotelBus"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_margin="@dimen/gap_sm"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginTop="@dimen/gap_xl"
    android:layout_marginEnd="@dimen/gap_xxl"
```

```
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:maxLines="1"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/black"
    android:visibility="visible"
    android:textSize="15dp" />

  <TextView
    android:id="@+id/amt"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_margin="@dimen/gap_sm"
    android:layout_marginStart="@dimen/gap_xxl"
    android:layout_marginTop="@dimen/gap_xl"
    android:layout_marginEnd="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:drawablePadding="@dimen/gap_md"
    android:elevation="@dimen/gap_xsm"
    android:maxLines="1"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:visibility="visible"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/black"
    android:textSize="15dp" />

  <LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="@dimen/gap_lg"
    android:layout_marginBottom="@dimen/gap_lg"
    android:gravity="center"
    >

    <androidx.appcompat.widget.AppCompatButton
      android:id="@+id/cancel"
      android:layout_width="200dp"
      android:layout_height="60dp"
      android:layout_gravity="right"
      android:layout_marginRight="5dp"
      android:background="@drawable/app_btn_bg2"
      android:elevation="4dp"
      android:gravity="center"
      android:text="Cancel"
      android:textAllCaps="false"
      android:textColor="@color/btn_text_color"
```

```
            android:textSize="16sp" />

        </LinearLayout>

        <TextView
            android:id="@+id/Tripexpired"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:visibility="gone"
            android:paddingEnd="@dimen/gap_xsm"
            android:text="Trip Expired"
            android:textColor="@color/app_color"
            android:textSize="25dp"
            android:textStyle="bold" />

    </LinearLayout>

  </androidx.cardview.widget.CardView>

</LinearLayout>
```

//view_bus_request.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
  xmlns:app="http://schemas.android.com/apk/res-auto"
  tools:context=".ui.fragments.UsersRequestFragment">

  <LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:padding="@dimen/gap_md"
    android:orientation="vertical"
    >

    <androidx.cardview.widget.CardView
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      app:cardCornerRadius="@dimen/gap_xl"
      android:elevation="@dimen/gap_sm">

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

```
android:layout_height="wrap_content"
android:orientation="vertical"
android:background="@color/light_grey"
android:padding="@dimen/gap_md"
android:gravity="center">

<TextView
   android:id="@+id/sign"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:text="Profile"
   android:textStyle="bold"
   android:textColor="@color/blue"
   android:textSize="@dimen/font_xxl"
   android:layout_marginBottom="@dimen/gap_xxl" />
<androidx.cardview.widget.CardView
   android:layout_width="80dp"
   android:layout_height="80dp"
   android:background="@color/white"
   android:elevation="@dimen/gap_sm"
   app:cardBackgroundColor="@color/white"
   app:cardCornerRadius="@dimen/gap_xxl"
   android:layout_margin="@dimen/gap_sm">

   <ImageView
      android:id="@+id/profile"
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:scaleType="fitXY"/>

</androidx.cardview.widget.CardView>
<TextView
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:layout_gravity="left"
   android:layout_marginLeft="10dp"
   android:text="Name"
   android:textStyle="bold"
   android:textColor="@color/blue"/>
<TextView
   android:id="@+id/Bownername"
   android:layout_width="match_parent"
   android:layout_height="50dp"
   android:layout_marginTop="@dimen/gap_xl"
   android:background="@drawable/edit_text_bg"
   android:padding="@dimen/gap_md"
   android:maxLines="1"
   android:textSize="@dimen/font_lg"
   android:paddingStart="@dimen/gap_lg"
   android:paddingEnd="@dimen/gap_lg"
```

```
    android:elevation="@dimen/gap_xsm"
    android:textColor="@color/blue"
    android:drawablePadding="@dimen/gap_md" />
  <TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="left"
    android:layout_marginLeft="10dp"
    android:text="Email"
    android:textStyle="bold"
    android:textColor="@color/blue"/>

  <TextView
    android:id="@+id/Bowneremail"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_marginTop="@dimen/gap_xl"
    android:background="@drawable/edit_text_bg"
    android:padding="@dimen/gap_md"
    android:maxLines="1"
    android:textSize="@dimen/font_lg"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:elevation="@dimen/gap_xsm"
    android:textColor="@color/blue"
    android:drawablePadding="@dimen/gap_md" />
  <TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="left"
    android:layout_marginTop="10dp"
    android:layout_marginLeft="10dp"
    android:text="Phone"
    android:textStyle="bold"
    android:textColor="@color/blue"/>
  <TextView
    android:id="@+id/Bownerphone"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_marginTop="@dimen/gap_xl"
    android:background="@drawable/edit_text_bg"
    android:padding="@dimen/gap_md"
    android:maxLines="1"
    android:textSize="@dimen/font_lg"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:elevation="@dimen/gap_xsm"
    android:textColor="@color/blue"
    android:drawablePadding="@dimen/gap_md" />
```

```
        <TextView
            android:id="@+id/hotelname"
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:layout_marginTop="@dimen/gap_xl"
            android:background="@drawable/edit_text_bg"
            android:padding="@dimen/gap_md"
            android:maxLines="1"
            android:visibility="gone"
            android:textSize="@dimen/font_lg"
            android:paddingStart="@dimen/gap_lg"
            android:paddingEnd="@dimen/gap_lg"
            android:elevation="@dimen/gap_xsm"
            android:textColor="@color/app_color"
            android:drawablePadding="@dimen/gap_md" />

        <TextView
            android:id="@+id/desc"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:visibility="gone"
            android:layout_gravity="left|center_vertical"
            android:gravity="center"
            android:text="All Description comes here, Don't worry it's a temporary one All
Description comes here, Don't worry it's a temporary one"
            android:textColor="@color/text_color_light"
            android:textSize="18dp" />

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:gravity="center"
            android:padding="@dimen/gap_md"
            android:orientation="horizontal">
            <Button
                android:id="@+id/Delete"
                android:layout_width="120dp"
                android:text="Delete"
                android:visibility="gone"
                android:layout_margin="10dp"
                android:layout_height="50dp" />

            <Button
                android:id="@+id/approve"
                android:layout_width="120dp"
                android:text="Approve"
                android:layout_marginLeft="6dp"
                android:layout_height="50dp" />

            <Button
```

```
                android:id="@+id/reject"
                android:layout_width="120dp"
                android:text="Reject"
                android:layout_margin="10dp"
                android:layout_height="50dp"
                android:shadowColor="@color/app_color"
                />
            </LinearLayout>

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

//fragment_profile.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
  xmlns:app="http://schemas.android.com/apk/res-auto"
  tools:context=".ui.fragments.ProfileFragment">

  <ScrollView
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:gravity="center"
      android:padding="@dimen/gap_md"
      android:orientation="vertical">

      <androidx.cardview.widget.CardView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="@dimen/gap_lg"
        android:layout_marginEnd="@dimen/gap_lg"
        app:cardCornerRadius="@dimen/gap_xl"
        android:elevation="@dimen/gap_sm">

        <LinearLayout
          android:layout_width="match_parent"
          android:layout_height="wrap_content"
          android:orientation="vertical"
          android:background="@color/light_grey"
```

```
android:padding="@dimen/gap_md"
android:gravity="center">

<TextView
   android:id="@+id/sign"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:text="Profile"
   android:textStyle="bold"
   android:textColor="@color/blue"
   android:textSize="@dimen/font_xxl"
   android:layout_marginBottom="@dimen/gap_xxl"/>
<TextView
   android:id="@+id/hname"
   android:layout_width="match_parent"
   android:layout_height="50dp"
   android:layout_marginTop="@dimen/gap_xl"
   android:layout_marginStart="@dimen/gap_xxl"
   android:background="@drawable/edit_text_bg"
   android:padding="@dimen/gap_md"
   android:visibility="gone"
   android:maxLines="1"
   android:textSize="@dimen/font_lg"
   android:paddingStart="@dimen/gap_lg"
   android:paddingEnd="@dimen/gap_lg"
   android:elevation="@dimen/gap_xsm"
   android:layout_margin="@dimen/gap_sm"
   android:textColor="@color/blue"
   android:drawablePadding="@dimen/gap_md"
   android:layout_marginEnd="@dimen/gap_xxl"/>

<TextView
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:layout_gravity="left"
   android:layout_marginLeft="10dp"
   android:text="Name"
   android:textStyle="bold"
   android:textColor="@color/blue"/>
<TextView
   android:id="@+id/name"
   android:layout_width="match_parent"
   android:layout_height="50dp"
   android:layout_marginTop="@dimen/gap_xl"
   android:layout_marginStart="@dimen/gap_xxl"
   android:background="@drawable/edit_text_bg"
   android:padding="@dimen/gap_md"
   android:maxLines="1"
   android:textSize="@dimen/font_lg"
   android:paddingStart="@dimen/gap_lg"
```

```
    android:paddingEnd="@dimen/gap_lg"
    android:elevation="@dimen/gap_xsm"
    android:layout_margin="@dimen/gap_sm"
    android:textColor="@color/black"
    android:drawablePadding="@dimen/gap_md"
    android:layout_marginEnd="@dimen/gap_xxl"/>
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="left"
    android:layout_marginLeft="10dp"
    android:text="Email"
    android:textStyle="bold"
    android:textColor="@color/blue"/>
<TextView
    android:id="@+id/email"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:inputType="textEmailAddress"
    android:layout_marginTop="@dimen/gap_xl"
    android:layout_marginStart="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:padding="@dimen/gap_md"
    android:maxLines="1"
    android:textSize="@dimen/font_lg"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:elevation="@dimen/gap_xsm"
    android:layout_margin="@dimen/gap_sm"
    android:textColor="@color/black"
    android:drawablePadding="@dimen/gap_md"
    android:layout_marginEnd="@dimen/gap_xxl"/>
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="left"
    android:layout_marginLeft="10dp"
    android:text="Phone"
    android:textStyle="bold"
    android:textColor="@color/blue"/>
<TextView
    android:id="@+id/phone"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_marginTop="@dimen/gap_xl"
    android:layout_marginStart="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:padding="@dimen/gap_md"
    android:maxLines="1"
    android:textSize="@dimen/font_lg"
```

```
            android:paddingStart="@dimen/gap_lg"
            android:paddingEnd="@dimen/gap_lg"
            android:elevation="@dimen/gap_xsm"
            android:inputType="phone"
            android:layout_margin="@dimen/gap_sm"
            android:textColor="@color/black"
            android:drawablePadding="@dimen/gap_md"
            android:layout_marginEnd="@dimen/gap_xxl"/>

        <Button
            android:id="@+id/Editprofile"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Edit Profile"
            />
            <TextView
                android:id="@+id/dlaccount"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginTop="10dp"
                android:text="Delete my acccount?"
                android:textStyle="bold"
                android:textColor="@color/app_color"
                android:layout_marginBottom="@dimen/gap_xxl"/>
        </LinearLayout>
    </androidx.cardview.widget.CardView>

    <androidx.cardview.widget.CardView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:cardCornerRadius="@dimen/gap_xl"
        android:layout_marginTop="@dimen/gap_xxl"
        android:elevation="@dimen/gap_sm">

        <LinearLayout
            android:id="@+id/li"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:background="@color/white"
            android:gravity="center"
            android:orientation="vertical">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_margin="@dimen/gap_md"
                android:gravity="center"
                android:orientation="horizontal">

                <TextView
```

```
            android:id="@+id/my_bookings"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_margin="@dimen/gap_md"
            android:background="@drawable/edit_text_bg"
            android:elevation="@dimen/gap_lg"
            android:padding="@dimen/gap_md"
            android:text="Bookings"
            android:textColor="@color/blue"
            android:textSize="@dimen/font_vlg"
            android:textStyle="bold" />

        <TextView
            android:id="@+id/apkg"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_margin="@dimen/gap_md"
            android:background="@drawable/edit_text_bg"
            android:elevation="@dimen/gap_lg"
            android:padding="@dimen/gap_md"
            android:text="Assigned Packages"
            android:textColor="@color/black"
            android:textSize="@dimen/font_vlg"
            android:textStyle="bold" />

    </LinearLayout>

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/rv_hbdata"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:visibility="visible" />

    <TextView
        android:id="@+id/no_data"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center"
        android:text="No Data"
        android:textColor="@color/black"
        android:textSize="@dimen/font_xl"
        android:visibility="gone" />
  </LinearLayout>
 </androidx.cardview.widget.CardView>
</LinearLayout>
</ScrollView>
</FrameLayout>
```

//fragment_package_admin.xml

//fragment_home.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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=".ui.fragments.HomeFragment">

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:src="@drawable/home" />


        </LinearLayout>

    </FrameLayout>
```

//fragment_edit_profile.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    tools:context=".ui.fragments.ProfileFragment">

    <ScrollView
        android:layout_marginTop="80dp"
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:gravity="center"
            android:padding="@dimen/gap_md"
            android:orientation="vertical">

            <androidx.cardview.widget.CardView
```

```
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_marginStart="@dimen/gap_lg"
android:layout_marginEnd="@dimen/gap_lg"
app:cardCornerRadius="@dimen/gap_xl"
android:elevation="@dimen/gap_sm">

<LinearLayout
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:orientation="vertical"
   android:background="@color/light_grey"
   android:padding="@dimen/gap_md"
   android:gravity="center">

   <TextView
      android:id="@+id/sign"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Edit Profile"
      android:textStyle="bold"
      android:textColor="@color/blue"
      android:textSize="@dimen/font_xxl"
      android:layout_marginBottom="@dimen/gap_xxl"/>

   <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_gravity="left"
      android:layout_marginLeft="10dp"
      android:text="Name"
      android:textStyle="bold"
      android:textColor="@color/blue"/>
   <EditText
      android:id="@+id/name"
      android:layout_width="match_parent"
      android:layout_height="50dp"
      android:layout_marginTop="@dimen/gap_xl"
      android:layout_marginStart="@dimen/gap_xxl"
      android:background="@drawable/edit_text_bg"
      android:padding="@dimen/gap_md"
      android:maxLines="1"
      android:textSize="@dimen/font_lg"
      android:paddingStart="@dimen/gap_lg"
      android:paddingEnd="@dimen/gap_lg"
      android:elevation="@dimen/gap_xsm"
      android:layout_margin="@dimen/gap_sm"
      android:textColor="@color/black"
      android:drawablePadding="@dimen/gap_md"
      android:layout_marginEnd="@dimen/gap_xxl"/>
```

```
<TextView
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:layout_gravity="left"
   android:layout_marginLeft="10dp"
   android:text="Email"
   android:textStyle="bold"
   android:textColor="@color/blue"/>
<EditText
   android:id="@+id/email"
   android:layout_width="match_parent"
   android:layout_height="50dp"
   android:layout_marginTop="@dimen/gap_xl"
   android:layout_marginStart="@dimen/gap_xxl"
   android:background="@drawable/edit_text_bg"
   android:padding="@dimen/gap_md"
   android:maxLines="1"
   android:textSize="@dimen/font_lg"
   android:paddingStart="@dimen/gap_lg"
   android:paddingEnd="@dimen/gap_lg"
   android:elevation="@dimen/gap_xsm"
   android:layout_margin="@dimen/gap_sm"
   android:textColor="@color/black"
   android:inputType="textEmailAddress"
   android:drawablePadding="@dimen/gap_md"
   android:layout_marginEnd="@dimen/gap_xxl"/>
<TextView
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:layout_gravity="left"
   android:layout_marginLeft="10dp"
   android:text="Phone"
   android:textStyle="bold"
   android:textColor="@color/blue"/>
<EditText
   android:id="@+id/phone"
   android:layout_width="match_parent"
   android:layout_height="50dp"
   android:layout_marginTop="@dimen/gap_xl"
   android:layout_marginStart="@dimen/gap_xxl"
   android:inputType="phone"
   android:background="@drawable/edit_text_bg"
   android:padding="@dimen/gap_md"
   android:maxLines="1"
   android:textSize="@dimen/font_lg"
   android:paddingStart="@dimen/gap_lg"
   android:paddingEnd="@dimen/gap_lg"
   android:elevation="@dimen/gap_xsm"
   android:layout_margin="@dimen/gap_sm"
   android:textColor="@color/black"
```

```
            android:drawablePadding="@dimen/gap_md"
            android:layout_marginEnd="@dimen/gap_xxl"/>

        <Button
            android:id="@+id/edit"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Edit"
            />
      </LinearLayout>
    </androidx.cardview.widget.CardView>

  </LinearLayout>

</ScrollView>

</FrameLayout>
```

//fragment_edit_package.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
    tools:context=".ui.fragments.EditPackageFragment">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:text="@string/hello_blank_fragment" />

</FrameLayout>
```

//fragment_edit_buses_by_owner.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
    tools:context=".ui.fragments.EditBusesByOwnerFragment">

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

```
    android:text="@string/hello_blank_fragment" />

</FrameLayout>
```

//fragment_admin_view_bookings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
   tools:context=".ui.fragments.AdminViewBookingsFragment">

   <TextView
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:text="@string/hello_blank_fragment" />

</FrameLayout>
```

//expense_item.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"
   android:layout_width="match_parent"
   android:layout_height="wrap_content">

   <androidx.cardview.widget.CardView
      android:id="@+id/parrent_layout"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:elevation="@dimen/gap_sm"
      app:cardBackgroundColor="@color/white"
      app:cardCornerRadius="@dimen/gap_md"
      android:layout_margin="@dimen/gap_sm"
      android:padding="@dimen/gap_md">

      <LinearLayout
         android:layout_width="match_parent"
         android:layout_height="match_parent"
         android:gravity="center"
         android:orientation="vertical"
         android:padding="@dimen/gap_md">

         <LinearLayout
```

```
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:gravity="center"
android:orientation="horizontal">

<androidx.cardview.widget.CardView
   android:layout_width="80dp"
   android:layout_height="80dp"
   android:layout_margin="@dimen/gap_sm"
   android:background="@color/white"
   android:elevation="@dimen/gap_sm"
   app:cardBackgroundColor="@color/white"
   app:cardCornerRadius="@dimen/gap_xxl">

   <ImageView
      android:id="@+id/image"
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:scaleType="fitXY" />

</androidx.cardview.widget.CardView>

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

   <TextView
      android:id="@+id/Busname"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:gravity="center"
      android:paddingEnd="@dimen/gap_xsm"
      android:text="bus Name"
      android:textColor="@color/app_color"
      android:textSize="25dp"
      android:textStyle="bold" />
   <TextView
      android:id="@+id/user"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:gravity="center"
      android:paddingEnd="@dimen/gap_xsm"
      android:text="bus Name"
      android:textColor="@color/app_color"
      android:textSize="25dp"
      android:textStyle="bold" />


   <LinearLayout
```

```
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:orientation="horizontal"
            android:weightSum="2">

            <TextView
               android:id="@+id/available_text"
               android:layout_width="match_parent"
               android:layout_height="wrap_content"
               android:layout_weight="1"
               android:gravity="end"
               android:text="Available Seats : "
               android:textColor="@color/recived_tr_clr"
               android:textSize="@dimen/font_lg"
               android:textStyle="bold"
               android:visibility="gone" />

            <TextView
               android:id="@+id/available"
               android:layout_width="match_parent"
               android:layout_height="wrap_content"
               android:layout_weight="1"
               android:text="0"
               android:textAlignment="textStart"
               android:textColor="@color/recived_tr_clr"
               android:textSize="@dimen/font_lg"
               android:textStyle="bold"
               android:visibility="gone" />

         </LinearLayout>
      </LinearLayout>

   </LinearLayout>

   <TextView
      android:id="@+id/frmto"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:gravity="center"
      android:layout_gravity="left|center_vertical"
      android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
      android:textColor="@color/text_color_light"
      android:textSize="20dp"/>

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

```
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/text_color_light"
            android:textSize="20dp"/>

        <TextView
            android:id="@+id/totst"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/text_color_light"
            android:textSize="20dp"/>
        <TextView
            android:id="@+id/desc"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="left|center_vertical"
            android:gravity="center"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/text_color_light"
            android:textSize="18dp"
            />

        <TextView
            android:id="@+id/amt"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/text_color_light"
            android:textSize="20dp"/>

    </LinearLayout>

  </androidx.cardview.widget.CardView>

</LinearLayout>
```

//assigned_trips.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"
   android:layout_width="match_parent"
   android:layout_height="wrap_content">

   <androidx.cardview.widget.CardView
      android:id="@+id/parrent_layout"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:elevation="@dimen/gap_sm"
      app:cardBackgroundColor="@color/white"
      app:cardCornerRadius="@dimen/gap_md"
      android:layout_margin="@dimen/gap_sm"
      android:padding="@dimen/gap_md">

      <LinearLayout
         android:layout_width="match_parent"
         android:layout_height="match_parent"
         android:gravity="center"
         android:orientation="vertical"
         android:padding="@dimen/gap_md">

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

            <androidx.cardview.widget.CardView
               android:layout_width="80dp"
               android:layout_height="80dp"
               android:layout_margin="@dimen/gap_sm"
               android:background="@color/white"
               android:elevation="@dimen/gap_sm"
               app:cardBackgroundColor="@color/white"
               app:cardCornerRadius="@dimen/gap_xxl">

               <ImageView
                  android:id="@+id/picture"
                  android:layout_width="match_parent"
                  android:layout_height="match_parent"
                  android:scaleType="fitXY" />

            </androidx.cardview.widget.CardView>

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

```
<TextView
   android:id="@+id/Busname"
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:gravity="center"
   android:paddingEnd="@dimen/gap_xsm"
   android:text="bus Name"
   android:textColor="@color/app_color"
   android:textSize="25dp"
   android:textStyle="bold" />


<LinearLayout
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:layout_gravity="center"
   android:orientation="horizontal"
   android:weightSum="2">

   <TextView
      android:id="@+id/available_text"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:layout_weight="1"
      android:gravity="end"
      android:text="Available Seats : "
      android:textColor="@color/recived_tr_clr"
      android:textSize="@dimen/font_lg"
      android:textStyle="bold"
      android:visibility="gone" />

   <TextView
      android:id="@+id/available"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:layout_weight="1"
      android:text="0"
      android:textAlignment="textStart"
      android:textColor="@color/recived_tr_clr"
      android:textSize="@dimen/font_lg"
      android:textStyle="bold"
      android:visibility="gone" />

   </LinearLayout>
  </LinearLayout>

</LinearLayout>

<TextView
```

```
            android:id="@+id/frmto"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/text_color_light"
            android:textSize="20dp"/>

        <TextView
            android:id="@+id/sdateedate"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/text_color_light"
            android:textSize="20dp"/>

        <TextView
            android:id="@+id/totst"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/text_color_light"
            android:textSize="20dp"/>
        <TextView
            android:id="@+id/desc"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="left|center_vertical"
            android:gravity="center"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/text_color_light"
            android:textSize="18dp"
            />

        <TextView
            android:id="@+id/amt"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
```

```
comes here, DOnt worry its a temprory one"
            android:textColor="@color/text_color_light"
            android:textSize="20dp"/>

        </LinearLayout>

    </androidx.cardview.widget.CardView>

</LinearLayout>
```

//admin_view_package_bookings.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"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <androidx.cardview.widget.CardView
        android:id="@+id/parrent_layout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:elevation="@dimen/gap_sm"
        app:cardBackgroundColor="@color/white"
        app:cardCornerRadius="@dimen/gap_md"
        android:layout_margin="@dimen/gap_sm"
        android:padding="@dimen/gap_md">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:gravity="center"
            android:orientation="vertical"
            android:padding="@dimen/gap_md"
            android:background="@color/app_bg">
            <TextView
                android:id="@+id/spotname"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:gravity="center"
                android:paddingEnd="@dimen/gap_xsm"
                android:text="Spot Name"
                android:textColor="@color/app_color"
                android:textSize="25dp"
                android:textStyle="bold"
                android:textAllCaps="true"/>
                    <ImageView
                      android:id="@+id/picture"
```

```
            android:layout_width="match_parent"
            android:layout_height="200dp"
            android:scaleType="fitXY"
            android:background="@color/white"
             android:layout_marginTop="10dp"/>

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


            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_gravity="center"
                android:orientation="horizontal"
                android:weightSum="2">

                <TextView
                    android:id="@+id/available_text"
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:layout_weight="1"
                    android:gravity="end"
                    android:text="Available Seats : "
                    android:textColor="@color/recived_tr_clr"
                    android:textSize="@dimen/font_lg"
                    android:textStyle="bold"
                    android:visibility="gone" />

                <TextView
                    android:id="@+id/available"
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:layout_weight="1"
                    android:text="0"
                    android:textAlignment="textStart"
                    android:textColor="@color/recived_tr_clr"
                    android:textSize="@dimen/font_lg"
                    android:textStyle="bold"
                    android:visibility="gone" />

            </LinearLayout>

        </LinearLayout>

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

```
            android:layout_height="wrap_content"
            android:layout_gravity="left|center_vertical"
            android:gravity="center"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/black"
            android:textSize="18dp"
            />

        <TextView
            android:id="@+id/frmto"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:layout_marginTop="10dp"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/black"
            android:textSize="20dp"/>

        <TextView
            android:id="@+id/sdateedate"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:layout_marginTop="10dp"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/black"
            android:textSize="20dp"/>

        <TextView
            android:id="@+id/bushotel"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/black"
            android:layout_marginTop="10dp"
            android:textSize="20dp"/>
        <TextView
            android:id="@+id/roomtype"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
```

```
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/black"
            android:layout_marginTop="10dp"
            android:textSize="20dp"/>
          <TextView
            android:id="@+id/HotelBus"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/black"
            android:layout_marginTop="10dp"
            android:textSize="20dp"/>

         <TextView
            android:id="@+id/totst"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/black"
            android:layout_marginTop="10dp"
            android:textSize="20dp"/>

         <TextView
            android:id="@+id/amt"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/black"
            android:layout_marginTop="10dp"
            android:textSize="20dp"/>
         <EditText
            android:id="@+id/ttlp"
            android:layout_width="match_parent"
            android:layout_height="60dp"
            android:layout_marginStart="@dimen/gap_xxl"
            android:layout_marginEnd="@dimen/gap_xxl"
            android:background="@drawable/edit_text_bg"
            android:drawablePadding="@dimen/gap_md"
            android:elevation="@dimen/gap_xsm"
            android:hint="Enter number of peoples"
```

```
            android:inputType="textCapWords"
            android:visibility="gone"
            android:padding="@dimen/gap_md"
            android:paddingStart="@dimen/gap_lg"
            android:paddingEnd="@dimen/gap_lg"
            android:textColor="@color/black"
            android:textColorHint="@color/black" />

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="@dimen/gap_lg"
            android:layout_marginBottom="@dimen/gap_lg"
            android:gravity="center"
            >

            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/edit"
                android:layout_width="200dp"
                android:layout_height="wrap_content"
                android:layout_gravity="right"
                android:background="@drawable/app_btn_bg"
                android:elevation="4dp"
                android:layout_marginRight="5dp"
                android:gravity="center"
                android:text="View Bookings"
                android:textAllCaps="false"
                android:textColor="@color/btn_text_color"
                android:textSize="16sp"/>

        </LinearLayout>

    </LinearLayout>

  </androidx.cardview.widget.CardView>

</LinearLayout>
```

//admin_view_package.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"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <androidx.cardview.widget.CardView
        android:id="@+id/parrent_layout"
```

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:elevation="@dimen/gap_sm"
    app:cardBackgroundColor="@color/white"
    app:cardCornerRadius="@dimen/gap_md"
    android:layout_margin="@dimen/gap_sm"
    android:padding="@dimen/gap_md">

    <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:gravity="center"
      android:orientation="vertical"
      android:padding="@dimen/gap_md"
      android:background="@color/app_bg">

      <TextView
        android:id="@+id/spotname"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:paddingEnd="@dimen/gap_xsm"
        android:text="Spot Name"
        android:textColor="@color/app_color"
        android:textSize="25dp"
        android:textStyle="bold" />

          <ImageView
            android:id="@+id/picture"
            android:layout_width="match_parent"
            android:layout_height="200dp"
            android:scaleType="fitXY"
            android:background="@color/light_grey"/>
      <TextView
        android:id="@+id/desc"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="left|center_vertical"
        android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
        android:textColor="@color/black"
        android:layout_marginTop="10dp"
        android:textSize="18dp"
        android:inputType="textMultiLine"
        />
        <LinearLayout
          android:layout_width="match_parent"
          android:layout_height="wrap_content"
          android:orientation="vertical">
          <TextView
```

```
        android:id="@+id/id"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textStyle="bold"
        android:gravity="center"
        android:paddingEnd="@dimen/gap_xsm"
        android:visibility="gone"
        android:text="Spot Name"
        android:textSize="@dimen/font_lg"
        android:textColor="@color/text_color" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:orientation="horizontal"
        android:weightSum="2">

        <TextView
            android:id="@+id/available_text"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:gravity="end"
            android:text="Available Seats : "
            android:textColor="@color/recived_tr_clr"
            android:textSize="@dimen/font_lg"
            android:textStyle="bold"
            android:visibility="gone" />

        <TextView
            android:id="@+id/available"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="0"
            android:textAlignment="textStart"
            android:textColor="@color/recived_tr_clr"
            android:textSize="@dimen/font_lg"
            android:textStyle="bold"
            android:visibility="gone" />

    </LinearLayout>

</LinearLayout>

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

```
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/black"
            android:layout_marginTop="10dp"
            android:textSize="20dp"/>
        <TextView
            android:id="@+id/sdateedate"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/black"
            android:layout_marginTop="10dp"
            android:textSize="20dp"/>

        <TextView
            android:id="@+id/bushotel"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/black"
            android:layout_marginTop="10dp"
            android:textSize="20dp"/>
        <ImageView
            android:id="@+id/roompro"
            android:layout_width="wrap_content"
            android:layout_height="100dp"
            android:scaleType="fitXY"
            android:background="@color/light_grey"
            android:layout_marginTop="10dp"/>

        <TextView
            android:id="@+id/roomtype"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/black"
            android:layout_marginTop="10dp"
            android:textSize="20dp"
          />
```

```
        <TextView
            android:id="@+id/HotelBus"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/black"
            android:layout_marginTop="10dp"
            android:textSize="20dp"
           />

        <ImageView
            android:id="@+id/buspro"
            android:layout_width="wrap_content"
            android:layout_height="100dp"
            android:scaleType="fitXY"
            android:background="@color/light_grey"
            android:layout_marginTop="10dp"/>
        <TextView
            android:id="@+id/totst"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/black"
            android:layout_marginTop="10dp"
            android:textSize="20dp"/>
        <TextView
            android:id="@+id/amt"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_gravity="left|center_vertical"
            android:text="All Description comes here, DOnt worry its a temprory one All Description
comes here, DOnt worry its a temprory one"
            android:textColor="@color/black"
            android:layout_marginTop="10dp"
            android:textSize="20dp"/>
        <EditText
            android:id="@+id/ttlp"
            android:layout_width="match_parent"
            android:layout_height="60dp"
            android:layout_marginStart="@dimen/gap_xxl"
            android:layout_marginEnd="@dimen/gap_xxl"
            android:background="@drawable/edit_text_bg"
            android:drawablePadding="@dimen/gap_md"
```

```
    android:elevation="@dimen/gap_xsm"
    android:hint="Enter number of peoples"
    android:inputType="textCapWords"
    android:visibility="gone"
    android:padding="@dimen/gap_md"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:textColor="@color/black"
    android:textColorHint="@color/black"
    android:layout_marginTop="10dp"
    />

  <LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="@dimen/gap_lg"
    android:layout_marginBottom="@dimen/gap_lg"
    android:gravity="center"
   >

    <androidx.appcompat.widget.AppCompatButton
      android:id="@+id/edit"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_gravity="right"
      android:background="@drawable/app_btn_bg"
      android:elevation="4dp"
      android:layout_marginRight="5dp"
      android:gravity="center"
      android:text="Edit"
      android:textAllCaps="false"
      android:textColor="@color/white"
      android:textSize="16sp"/>

    <androidx.appcompat.widget.AppCompatButton
      android:id="@+id/delete"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_gravity="right"
      android:background="@drawable/app_btn_bg2"
      android:elevation="4dp"
      android:gravity="center"
      android:text="Delete"
      android:textAllCaps="false"
      android:textColor="@color/white"
      android:textSize="16sp" />

  </LinearLayout>

  <TextView
```

```
            android:id="@+id/expired"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:paddingEnd="@dimen/gap_xsm"
            android:text="Expired"
            android:visibility="gone"
            android:textColor="@color/black"
            android:textSize="25dp"
            android:textStyle="bold" />

      </LinearLayout>

   </androidx.cardview.widget.CardView>

</LinearLayout>
```

//activity_admin_view_bookings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:tools="http://schemas.android.com/tools"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:id="@+id/container1"
   tools:context=".ui.fragments.AdminPackageFragment">

   <ScrollView
      android:layout_width="match_parent"
      android:layout_height="match_parent">

      <LinearLayout
         android:layout_width="match_parent"
         android:layout_height="match_parent"
         android:gravity="center"
         android:padding="@dimen/gap_md"
         android:orientation="vertical">

         <androidx.cardview.widget.CardView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginStart="@dimen/gap_lg"
            android:layout_marginEnd="@dimen/gap_lg"
            app:cardCornerRadius="@dimen/gap_xl"
            android:elevation="@dimen/gap_sm">

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

```
        <androidx.cardview.widget.CardView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            app:cardCornerRadius="@dimen/gap_xl"
            android:layout_marginTop="40dp"
            android:elevation="@dimen/gap_sm">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="vertical"
                android:layout_marginTop="30dp"
                android:background="@color/white"
                android:gravity="center">

                <LinearLayout
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:orientation="horizontal"
                    android:gravity="center"
                    android:layout_margin="@dimen/gap_md">

                </LinearLayout>

                <androidx.recyclerview.widget.RecyclerView
                    android:id="@+id/rv_apackage"
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:visibility="visible"/>
                <TextView
                    android:id="@+id/no_data"
                    android:layout_width="match_parent"
                    android:layout_height="match_parent"
                    android:text="No Data"
                    android:visibility="gone"
                    android:textColor="@color/black"
                    android:textSize="@dimen/font_xl"
                    android:gravity="center"/>
            </LinearLayout>
        </androidx.cardview.widget.CardView>

    </LinearLayout>

  </ScrollView>

</FrameLayout>
```

//admin_view_bookings_one.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   tools:context=".ui.fragments.UsersRequestFragment">

   <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:gravity="center"
      android:padding="@dimen/gap_md"
      android:orientation="vertical"
      android:background="@color/white">

      <androidx.cardview.widget.CardView
         android:layout_width="match_parent"
         android:layout_height="wrap_content"
         android:layout_marginStart="@dimen/gap_lg"
         android:layout_marginEnd="@dimen/gap_lg"
         app:cardCornerRadius="@dimen/gap_xl"
         android:elevation="@dimen/gap_sm">

         <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:background="@color/light_grey"
            android:padding="@dimen/gap_md"
            android:gravity="center">

            <TextView
               android:id="@+id/sign"
               android:layout_width="wrap_content"
               android:layout_height="wrap_content"
               android:text="Profile"
               android:textStyle="bold"
               android:textColor="@color/blue"
               android:textSize="@dimen/font_xxl"
               android:layout_marginBottom="@dimen/gap_xxl"/>
            <TextView
               android:layout_width="wrap_content"
               android:layout_height="wrap_content"
               android:layout_gravity="left"
               android:layout_marginLeft="10dp"
               android:text="Name"
               android:textColor="@color/blue"
               android:textStyle="bold"/>
            <TextView
```

```
    android:id="@+id/name"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_marginTop="@dimen/gap_xl"
    android:layout_marginStart="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:padding="@dimen/gap_md"
    android:maxLines="1"
    android:textSize="@dimen/font_lg"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:elevation="@dimen/gap_xsm"
    android:layout_margin="@dimen/gap_sm"
    android:textColor="@color/black"
    android:drawablePadding="@dimen/gap_md"
    android:layout_marginEnd="@dimen/gap_xxl"/>
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="left"
    android:layout_marginLeft="10dp"
    android:text="Email"
    android:textColor="@color/blue"
    android:textStyle="bold"/>
<TextView
    android:id="@+id/email"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_marginTop="@dimen/gap_xl"
    android:layout_marginStart="@dimen/gap_xxl"
    android:background="@drawable/edit_text_bg"
    android:padding="@dimen/gap_md"
    android:maxLines="1"
    android:textSize="@dimen/font_lg"
    android:paddingStart="@dimen/gap_lg"
    android:paddingEnd="@dimen/gap_lg"
    android:elevation="@dimen/gap_xsm"
    android:layout_margin="@dimen/gap_sm"
    android:textColor="@color/black"
    android:drawablePadding="@dimen/gap_md"
    android:layout_marginEnd="@dimen/gap_xxl"/>
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="left"
    android:layout_marginLeft="10dp"
    android:text="Phone"
    android:textColor="@color/blue"
    android:textStyle="bold"/>
<TextView
```

```
            android:id="@+id/phone"
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:layout_marginTop="@dimen/gap_xl"
            android:layout_marginStart="@dimen/gap_xxl"
            android:background="@drawable/edit_text_bg"
            android:padding="@dimen/gap_md"
            android:maxLines="1"
            android:textSize="@dimen/font_lg"
            android:paddingStart="@dimen/gap_lg"
            android:paddingEnd="@dimen/gap_lg"
            android:elevation="@dimen/gap_xsm"
            android:layout_margin="@dimen/gap_sm"
            android:textColor="@color/black"
            android:drawablePadding="@dimen/gap_md"
            android:layout_marginEnd="@dimen/gap_xxl"/>
        </LinearLayout>
      </androidx.cardview.widget.CardView>
    </LinearLayout>
</FrameLayout>
```

//activity_splash_screen.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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="@color/blue"
    android:theme="@style/ThemeOverlay.travel_mate.FullscreenContainer"
    tools:context=".ui.activities.SplashScreenActivity">


    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="Tour And Travel"
        android:textColor="@color/white"
        android:textSize="40sp"
        android:textStyle="bold"
        android:visibility="visible" />
</FrameLayout>
```

//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=".ui.activities.Booking_Activity">

   <FrameLayout
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:background="@color/app_bg">

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

         <EditText
            android:id="@+id/cnum"
            android:layout_width="match_parent"
            android:layout_height="60dp"
            android:layout_marginTop="@dimen/gap_lg"
            android:layout_marginStart="@dimen/gap_xxl"
            android:background="@drawable/edit_text_bg"
            android:padding="@dimen/gap_md"
            android:paddingStart="@dimen/gap_lg"
            android:paddingEnd="@dimen/gap_lg"
            android:elevation="@dimen/gap_xsm"

            android:hint="Card Number"
            android:inputType="number"
            android:textColor="@color/blue"
            android:drawablePadding="@dimen/gap_md"
            android:textColorHint="@color/light_grey"
            android:layout_marginEnd="@dimen/gap_xxl"/>

         <EditText
            android:id="@+id/cvv"
            android:layout_width="match_parent"
            android:layout_height="60dp"
            android:layout_marginTop="@dimen/gap_lg"
            android:layout_marginStart="@dimen/gap_xxl"
            android:background="@drawable/edit_text_bg"
            android:padding="@dimen/gap_md"
            android:paddingStart="@dimen/gap_lg"
            android:paddingEnd="@dimen/gap_lg"
```

```
            android:elevation="@dimen/gap_xsm"
            android:hint="CVV"
            android:inputType="number"
            android:textColor="@color/blue"
            android:drawablePadding="@dimen/gap_sm"
            android:textColorHint="@color/light_grey"
            android:layout_marginEnd="@dimen/gap_xxl"/>

        <EditText
            android:id="@+id/vlddate"
            android:layout_width="match_parent"
            android:layout_height="60dp"
            android:layout_marginTop="@dimen/gap_lg"
            android:layout_marginStart="@dimen/gap_xxl"
            android:background="@drawable/edit_text_bg"
            android:padding="@dimen/gap_md"
            android:paddingStart="@dimen/gap_lg"
            android:paddingEnd="@dimen/gap_lg"
            android:elevation="@dimen/gap_xsm"
            android:hint="Valid Date"
            android:inputType="text"
            android:textColor="@color/blue"
            android:drawablePadding="@dimen/gap_sm"
            android:textColorHint="@color/light_grey"
            android:layout_marginEnd="@dimen/gap_xxl"/>
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="vertical"
            android:layout_marginTop="@dimen/gap_xxxl"
            android:layout_marginStart="@dimen/gap_xxxl"
            android:layout_marginEnd="@dimen/gap_xxxl">

            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/book"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:background="@drawable/app_btn_bg"
                android:elevation="4dp"
                android:gravity="center"
                android:text="Confirm Booking"
                android:textAllCaps="false"
                android:textColor="@color/btn_text_color"
                android:textSize="16sp"
                android:textStyle="bold" />

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

//fragment_review.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    tools:context=".ui.fragments.ReviewFragment">

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

        <androidx.recyclerview.widget.RecyclerView
            android:id="@+id/rv_ideas"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"/>

        <androidx.cardview.widget.CardView
            android:id="@+id/card_add_ideas"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginStart="@dimen/gap_lg"
            android:layout_marginEnd="@dimen/gap_lg"
            app:cardBackgroundColor="@color/white"
            app:cardCornerRadius="@dimen/gap_xl"
            android:elevation="@dimen/gap_sm"
            android:visibility="gone"
            android:layout_gravity="center">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="vertical"
                android:background="@color/white"
                android:padding="@dimen/gap_md"
                android:gravity="center">

                <EditText
                    android:id="@+id/title"
                    android:layout_width="match_parent"
                    android:layout_height="150dp"
                    android:layout_marginTop="@dimen/gap_xl"
                    android:layout_marginStart="@dimen/gap_xxl"
                    android:background="@drawable/edit_text_bg"
                    android:padding="@dimen/gap_md"
                    android:textSize="@dimen/font_lg"
```

```
            android:paddingStart="@dimen/gap_lg"
            android:paddingEnd="@dimen/gap_lg"
            android:elevation="@dimen/gap_xsm"
            android:hint="Write here..."
            android:layout_margin="@dimen/gap_sm"
            android:textColor="@color/black"
            android:drawablePadding="@dimen/gap_md"
            android:textColorHint="@color/light_grey"
            android:inputType="textMultiLine"
            android:layout_marginEnd="@dimen/gap_xxl"/>

        <androidx.appcompat.widget.AppCompatButton
            android:id="@+id/add_btn"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@drawable/app_btn_bg"
            android:elevation="4dp"
            android:gravity="center"
            android:layout_marginStart="@dimen/gap_xl"
            android:layout_marginEnd="@dimen/gap_xl"
            android:layout_gravity="bottom"
            android:layout_marginTop="40dp"
            android:layout_marginBottom="20dp"
            android:text="Submit"
            android:textAllCaps="true"
            android:textColor="@color/white"
            android:textSize="16sp"
            android:textStyle="bold" />

        <androidx.appcompat.widget.AppCompatButton
            android:id="@+id/cancel_btn"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@drawable/app_btn_bg2"
            android:elevation="4dp"
            android:gravity="center"
            android:layout_marginStart="@dimen/gap_xl"
            android:layout_marginEnd="@dimen/gap_xl"
            android:layout_gravity="bottom"
            android:layout_marginTop="20dp"
            android:layout_marginBottom="20dp"
            android:text="Cancel"
            android:textAllCaps="true"
            android:textColor="@color/white"
            android:textSize="16sp"
            android:textStyle="bold" />
    </LinearLayout>
</androidx.cardview.widget.CardView>

<ImageView
```

```
        android:id="@+id/add_review"
        android:layout_width="50dp"
        android:layout_height="50dp"
        android:src="@drawable/add_button"
        android:layout_gravity="bottom|right"
        android:layout_margin="@dimen/gap_xl"/>
  </FrameLayout>
</FrameLayout>
```

//item_review.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="wrap_content"
   xmlns:app="http://schemas.android.com/apk/res-auto">

   <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:orientation="vertical"
      android:background="@color/transparent">

      <androidx.cardview.widget.CardView
         android:layout_width="match_parent"
         android:layout_height="wrap_content"
         android:layout_marginStart="@dimen/gap_lg"
         android:layout_marginEnd="@dimen/gap_lg"
         app:cardCornerRadius="@dimen/gap_lg"
         android:elevation="@dimen/gap_lg"
         android:layout_marginTop="@dimen/gap_xl">

         <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:padding="@dimen/gap_lg"
            android:gravity="center"
            android:background="@color/app_bg"
            >
            <TextView
               android:id="@+id/email"
               android:layout_width="match_parent"
               android:layout_height="wrap_content"
               android:textStyle="bold"
               android:textSize="@dimen/font_lg"
               android:textColor="@color/black"
               android:text="email"/>
```

```
            <TextView
               android:id="@+id/title"
               android:layout_width="match_parent"
               android:layout_height="wrap_content"
               android:textStyle="bold"
               android:textSize="@dimen/font_lg"
               android:textColor="@color/black"
               android:text="Review"/>

            <TextView
               android:id="@+id/delete"
               android:layout_width="wrap_content"
               android:layout_height="wrap_content"
               android:textStyle="bold"
               android:textSize="@dimen/font_lg"
               android:layout_marginTop="@dimen/gap_sm"
               android:textColor="@color/white"
               android:text="Delete"
               android:visibility="gone"
               android:padding="@dimen/gap_sm"
               android:paddingStart="@dimen/gap_sm"
               android:paddingEnd="@dimen/gap_sm"
               android:background="@drawable/app_btn_bg2"/>

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

//fragment_rejection_history.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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:id="@+id/requestcontainer"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   tools:context=".ui.fragments.ProfileFragment">

   <ScrollView
      android:layout_width="match_parent"
      android:layout_height="match_parent">

      <LinearLayout
         android:layout_width="match_parent"
         android:layout_height="match_parent"
         android:gravity="center"
```

```
android:padding="@dimen/gap_md"
android:orientation="vertical">
<TextView
   android:id="@+id/History"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:text="History"
   android:textStyle="bold"
   android:textColor="@color/app_color"
   android:textSize="30dp"
   android:background="@drawable/edit_text_bg"
   android:padding="@dimen/gap_md"
   android:layout_margin="@dimen/gap_md"
   android:elevation="@dimen/gap_lg"/>
<androidx.cardview.widget.CardView
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:layout_marginStart="@dimen/gap_lg"
   android:layout_marginEnd="@dimen/gap_lg"
   app:cardCornerRadius="@dimen/gap_xl"
   android:elevation="@dimen/gap_sm">

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   app:cardCornerRadius="@dimen/gap_xl"
   android:layout_marginTop="@dimen/gap_xxl"
   android:elevation="@dimen/gap_sm">

   <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:orientation="vertical"
      android:background="@color/white"
      android:gravity="center">

      <LinearLayout
         android:layout_width="match_parent"
         android:layout_height="wrap_content"
         android:orientation="horizontal"
         android:gravity="center"
         android:layout_margin="@dimen/gap_md">

         <TextView
            android:id="@+id/Bus_Owners"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Bus Owners"
```

```
                android:textStyle="bold"
                android:textColor="@color/app_color"
                android:textSize="@dimen/font_vlg"
                android:background="@drawable/edit_text_bg"
                android:padding="@dimen/gap_md"
                android:layout_margin="@dimen/gap_md"
                android:elevation="@dimen/gap_lg"/>

            <TextView
                android:id="@+id/Hotel_Owners"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Hotel Owners"
                android:textStyle="bold"
                android:textColor="@color/text_color"
                android:textSize="@dimen/font_vlg"
                android:background="@drawable/edit_text_bg"
                android:padding="@dimen/gap_md"
                android:layout_margin="@dimen/gap_md"
                android:elevation="@dimen/gap_lg"/>

        </LinearLayout>

        <androidx.recyclerview.widget.RecyclerView
            android:id="@+id/rv_history"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:visibility="visible"/>
        <TextView
            android:id="@+id/no_data"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:text="No Data"
            android:visibility="gone"
            android:textColor="@color/black"
            android:textSize="@dimen/font_xl"
            android:gravity="center"/>
      </LinearLayout>
    </androidx.cardview.widget.CardView>
  </LinearLayout>

 </ScrollView>

</FrameLayout>
```

//history_page.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    tools:context=".ui.fragments.Rejection_History_Fragment">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center"
        android:padding="@dimen/gap_md"
        android:orientation="vertical">

        <androidx.cardview.widget.CardView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            app:cardCornerRadius="@dimen/gap_xl"
            android:elevation="@dimen/gap_sm">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="vertical"
                android:background="@color/app_bg"
                android:padding="@dimen/gap_md"
                android:gravity="center">

                <TextView
                    android:id="@+id/sign"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:text="Profile"
                    android:textStyle="bold"
                    android:textColor="@color/blue"
                    android:textSize="@dimen/font_xxl"
                    android:textAllCaps="true"
                    android:layout_marginBottom="@dimen/gap_xxl" />
                <androidx.cardview.widget.CardView
                    android:layout_width="80dp"
                    android:layout_height="80dp"
                    android:background="@color/white"
                    android:elevation="@dimen/gap_sm"
                    app:cardBackgroundColor="@color/white"
                    app:cardCornerRadius="@dimen/gap_xxl"
                    android:layout_margin="@dimen/gap_sm">

                    <ImageView
                        android:id="@+id/profile"
```

```
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:scaleType="fitXY"/>

    </androidx.cardview.widget.CardView>
    <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_gravity="left"
      android:layout_marginLeft="10dp"
      android:text="Name"
      android:textColor="@color/blue"
      android:textStyle="bold"/>
    <TextView
      android:id="@+id/Bownername"
      android:layout_width="match_parent"
      android:layout_height="50dp"
      android:layout_marginTop="@dimen/gap_xl"
      android:background="@drawable/edit_text_bg"
      android:padding="@dimen/gap_md"
      android:maxLines="1"
      android:textSize="@dimen/font_lg"
      android:paddingStart="@dimen/gap_lg"
      android:paddingEnd="@dimen/gap_lg"
      android:elevation="@dimen/gap_xsm"
      android:textColor="@color/blue"
      android:drawablePadding="@dimen/gap_md" />
    <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_gravity="left"
      android:layout_marginLeft="10dp"
      android:text="Email"
      android:textColor="@color/blue"
      android:textStyle="bold"/>
    <TextView
      android:id="@+id/Bowneremail"
      android:layout_width="match_parent"
      android:layout_height="50dp"
      android:layout_marginTop="@dimen/gap_xl"
      android:background="@drawable/edit_text_bg"
      android:padding="@dimen/gap_md"
      android:maxLines="1"
      android:textSize="@dimen/font_lg"
      android:paddingStart="@dimen/gap_lg"
      android:paddingEnd="@dimen/gap_lg"
      android:elevation="@dimen/gap_xsm"
      android:textColor="@color/blue"
      android:drawablePadding="@dimen/gap_md" />
    <TextView
```

```
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="left"
            android:layout_marginLeft="10dp"
            android:text="Phone"
            android:textColor="@color/blue"
            android:textStyle="bold"/>
        <TextView
            android:id="@+id/Bownerphone"
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:layout_marginTop="@dimen/gap_xl"
            android:background="@drawable/edit_text_bg"
            android:padding="@dimen/gap_md"
            android:maxLines="1"
            android:textSize="@dimen/font_lg"
            android:paddingStart="@dimen/gap_lg"
            android:paddingEnd="@dimen/gap_lg"
            android:elevation="@dimen/gap_xsm"
            android:textColor="@color/blue"
            android:drawablePadding="@dimen/gap_md" />

        <TextView
            android:id="@+id/hotelname"
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:layout_marginTop="@dimen/gap_xl"
            android:background="@drawable/edit_text_bg"
            android:padding="@dimen/gap_md"
            android:maxLines="1"
            android:visibility="gone"
            android:textSize="@dimen/font_lg"
            android:paddingStart="@dimen/gap_lg"
            android:paddingEnd="@dimen/gap_lg"
            android:elevation="@dimen/gap_xsm"
            android:textColor="@color/app_color"
            android:drawablePadding="@dimen/gap_md" />

        <TextView
            android:id="@+id/desc"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:visibility="gone"
            android:layout_gravity="left|center_vertical"
            android:gravity="center"
            android:text="All Description comes here, Don't worry it's a temporary one All
Description comes here, Don't worry it's a temporary one"
            android:textColor="@color/text_color_light"
            android:textSize="18dp" />
```

```
            <LinearLayout
               android:layout_width="match_parent"
               android:layout_height="match_parent"
               android:gravity="center"
               android:padding="@dimen/gap_md"
               android:orientation="horizontal">
               <Button
                  android:id="@+id/Delete"
                  android:layout_width="120dp"
                  android:text="Delete"
                  android:layout_margin="10dp"
                  android:layout_height="50dp" />
            </LinearLayout>

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

//activity_razor_pay.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=".PGIntegration.RazorPayActivity">
</androidx.constraintlayout.widget.ConstraintLayout>
```

//JAVA CODE

//MainActivity.java

```
package com.dzniox.travel_mate.ui.activities;

import android.Manifest;
import android.content.Intent;
import android.os.Build;
import android.os.Bundle;
import android.os.StrictMode;
import android.util.Log;
import android.view.MenuItem;
```

```
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.ImageView;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;
import androidx.annotation.RequiresApi;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.AppCompatSpinner;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentTransaction;
import com.dzniox.travel_mate.Model.AllCategory;
import com.dzniox.travel_mate.Model.Example;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.NetworkUtilsCustom.GetResult;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.databinding.ActivityMainBinding;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import com.dzniox.travel_mate.ui.fragments.AddBusFragment;
import com.dzniox.travel_mate.ui.fragments.AddHotelRoomsFragment;
import com.dzniox.travel_mate.ui.fragments.AddSpotFragment;
import com.dzniox.travel_mate.ui.fragments.AdminPackageFragment;
import com.dzniox.travel_mate.ui.fragments.BookingFragment;
import com.dzniox.travel_mate.ui.fragments.HomeFragment;
import com.dzniox.travel_mate.ui.fragments.ProfileFragment;
import com.dzniox.travel_mate.ui.fragments.UsersRequestFragment;
import com.dzniox.travel_mate.ui.fragments.ReviewFragment;
import com.dzniox.travel_mate.ui.fragments.UsersViewFragment;
import com.dzniox.travel_mate.ui.fragments.ViewBusesFragment;
import com.dzniox.travel_mate.ui.fragments.ViewHotelRoomsFragment;
import com.dzniox.travel_mate.utils.CustPrograssbar;
import com.google.android.material.bottomnavigation.BottomNavigationView;
import com.google.android.material.navigation.NavigationBarView;
import com.google.gson.Gson;
import com.google.gson.JsonObject;
import com.google.gson.JsonParser;
import com.karumi.dexter.Dexter;
import com.karumi.dexter.MultiplePermissionsReport;
import com.karumi.dexter.PermissionToken;
import com.karumi.dexter.listener.DexterError;
import com.karumi.dexter.listener.PermissionRequest;
import com.karumi.dexter.listener.PermissionRequestErrorListener;
import com.karumi.dexter.listener.multi.MultiplePermissionsListener;
import com.toptoche.searchablespinnerlibrary.SearchableSpinner;
import org.json.JSONException;
import org.json.JSONObject;
import java.util.ArrayList;
import java.util.List;
import retrofit2.Call;

public class MainActivity extends AppCompatActivity implements View.OnClickListener,
GetResult.MyListener {

    ActivityMainBinding binding;
    private BottomNavigationView nav_view;
```

```
    PrefUtils prefrences;
    List<AllCategory> ShopList;
    ArrayList<AllCategory> shopLIST = new ArrayList<>();
    private ImageView add_spot;
    private HomeFragment homeFragment;
    TextView item_text;
    private BookingFragment bookingFragment;
    private ProfileFragment profileFragment;
    private AddSpotFragment addSpotFragment;
    private UsersRequestFragment usersrequestfragment;
    private UsersViewFragment usersviewfragment;
    private AdminPackageFragment adminpackagefragment;
    private ViewBusesFragment viewbusesfragment;
    private AddBusFragment addbusfragment;
    private AddHotelRoomsFragment addhotelroomsfragment;
    private ViewHotelRoomsFragment viewhotelroomsfragment;
    private ArrayList<String> DISTRICTS = new ArrayList<>();
    CustPrograssbar custPrograssbar;
    AppCompatSpinner spinner;

    Spinner spnCity;
    private ReviewFragment reviewFragment;
    MenuItem menuItem;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        prefrences = PrefUtils.getInstance(this);
        add_spot = findViewById(R.id.add_spot);
        item_text = findViewById(R.id.item_text);
        custPrograssbar = new CustPrograssbar();
        requestPermissions();
        StrictMode.ThreadPolicy policy = new StrictMode.ThreadPolicy.Builder().permitAll().build();
        StrictMode.setThreadPolicy(policy);
        homeFragment = new HomeFragment();
        bookingFragment = new BookingFragment();
        profileFragment = new ProfileFragment();
        addSpotFragment = new AddSpotFragment();
        reviewFragment = new ReviewFragment();
        usersrequestfragment=new UsersRequestFragment();
        usersviewfragment=new UsersViewFragment();
        adminpackagefragment=new AdminPackageFragment();
        addbusfragment=new AddBusFragment();
        viewbusesfragment=new ViewBusesFragment();
        addhotelroomsfragment=new AddHotelRoomsFragment();
        viewhotelroomsfragment=new ViewHotelRoomsFragment();
        LoadFragment(homeFragment);

        nav_view = findViewById(R.id.nav_view);

        if (prefrences.getStringValue("email", "").equalsIgnoreCase("admin@gmail.com") &&
prefrences.getStringValue("usertype","").equalsIgnoreCase("1")) {
            add_spot.setVisibility(View.VISIBLE);
            item_text.setVisibility(View.VISIBLE);
            nav_view.getMenu().findItem(R.id.navigation_booking).setTitle("Packages");
```

```
            nav_view.getMenu().findItem(R.id.navigation_profile).setTitle("Requests");
        }
        else if ( prefrences.getStringValue("usertype","").equalsIgnoreCase("0")) {
            add_spot.setVisibility(View.GONE);
            item_text.setVisibility(View.GONE);
            nav_view.getMenu().removeItem(R.id.navigation_Users);
        }
        else if (prefrences.getStringValue("usertype", "").equalsIgnoreCase("2")) {
            add_spot.setVisibility(View.GONE);
            item_text.setVisibility(View.GONE);
            nav_view.getMenu().findItem(R.id.navigation_booking).setTitle("Add Bus");
            nav_view.getMenu().findItem(R.id.navigation_review).setTitle("View Buses");
            nav_view.getMenu().removeItem(R.id.navigation_Users);
        }
       else if (prefrences.getStringValue("usertype", "").equalsIgnoreCase("3")) {
            add_spot.setVisibility(View.GONE);
            item_text.setVisibility(View.GONE);
            nav_view.getMenu().findItem(R.id.navigation_booking).setTitle("Add Rooms");
            nav_view.getMenu().findItem(R.id.navigation_review).setTitle("View Rooms");
            nav_view.getMenu().removeItem(R.id.navigation_Users);
        }

        nav_view.setOnNavigationItemSelectedListener(item -> {
            switch (item.getItemId()) {
                case R.id.navigation_home:

                    LoadFragment(homeFragment);
                    break;

                case R.id.navigation_booking:

                    if(prefrences.getStringValue("email", "").equalsIgnoreCase("admin@gmail.com") ||
prefrences.getStringValue("usertype", "").equalsIgnoreCase("0")){
                        LoadFragment(adminpackagefragment);
                    }else if(prefrences.getStringValue("usertype", "").equalsIgnoreCase("2")){
                        LoadFragment(addbusfragment);
                    }
                    else if(prefrences.getStringValue("usertype", "").equalsIgnoreCase("3")){
                        LoadFragment(addhotelroomsfragment);
                    }
                    else{
                        LoadFragment(bookingFragment);
                    }
                    break;
                case R.id.navigation_review:

                    if(prefrences.getStringValue("usertype", "").equalsIgnoreCase("2")){
                        LoadFragment(viewbusesfragment);
                    }
                    else if(prefrences.getStringValue("usertype", "").equalsIgnoreCase("3")) {
                        LoadFragment(viewhotelroomsfragment);
                     }
                    else {
                        LoadFragment(reviewFragment);
                    }
                    break;
```

```
            case R.id.navigation_profile:

               if (prefrences.getStringValue("email", "").equalsIgnoreCase("admin@gmail.com")) {
                  LoadFragment(usersrequestfragment);

Log.d("ket","ddhvdafudsgfsdfhgsdj88888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888gvf");
               }
               else {
                  LoadFragment(profileFragment);

Log.d("ket","ddhvdafudsgfsdfhgsdj88888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888gvf");
               }
               break;
            case R.id.navigation_Users:
                  LoadFragment(usersviewfragment);
                  Log.d("ket",
"ddhvdafudsgfsdfhgsdj88888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888gvf");
         }
         return true;
      });

      add_spot.setOnClickListener(view -> {
      });
   }

   private void getAvailableCategory(String toString) {

   }
   private void requestPermissions() {
      Dexter.withActivity(this)
            .withPermissions(

                  android.Manifest.permission.WRITE_EXTERNAL_STORAGE,
                  Manifest.permission.CAMERA,
                  Manifest.permission.CALL_PHONE,
                  Manifest.permission.READ_EXTERNAL_STORAGE)
            .withListener(new MultiplePermissionsListener() {
               @Override
               public void onPermissionsChecked(MultiplePermissionsReport report) {
                  if (report.areAllPermissionsGranted()) {
                  }

                  if (report.isAnyPermissionPermanentlyDenied()) {

                  }
               }

               @Override
               public void onPermissionRationaleShouldBeShown(List<PermissionRequest>
permissions, PermissionToken token) {
                  token.continuePermissionRequest();
               }
            }).
```

```
            withErrorListener(new PermissionRequestErrorListener() {
               @Override
               public void onError(DexterError error) {
                  Toast.makeText(getApplicationContext(), "Some Error! ",
Toast.LENGTH_SHORT).show();
               }
            })
            .onSameThread()
            .check();
   }

   public void GoToback(View view) {
   }

   public void LogOut(View view) {
      prefrences.clear();
      Intent intent = new Intent(this, SplashScreenActivity.class);
      startActivity(intent);
      finish();
   }

   public void LoadFragment(Fragment fragment) {
      final FragmentTransaction transaction = getSupportFragmentManager().beginTransaction();
      transaction.replace(R.id.container, fragment);
      transaction.addToBackStack(null);
      transaction.commit();
   }

   @Override
   public void onClick(View v) {
      switch (v.getId()) {
         case R.id.add_btn:
            break;
      }
   }

   @Override
   public void callback(JsonObject result, String callNo) {

   }
}
```

//MainActivity2.java

```
package com.dzniox.travel_mate.ui.activities;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.widget.EditText;
import com.dzniox.travel_mate.R;
public class MainActivity2 extends AppCompatActivity {

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

```
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main2);
    String desc = getIntent().getStringExtra("desc");
    String busname = getIntent().getStringExtra("busname");
    String seatno = getIntent().getStringExtra("seatno");
    String id=getIntent().getStringExtra("id");
    EditText editTextDesc = findViewById(R.id.addbownerdesc);
    EditText editTextBusName = findViewById(R.id.addbusname);
    EditText seat = findViewById(R.id.addbownerseat_no);
    editTextDesc.setText(desc);
    editTextBusName.setText(busname);
    seat.setText(seatno);
  }
}
```

//MainActivity3.java

```
package com.dzniox.travel_mate.ui.activities;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.EditText;
import com.dzniox.travel_mate.R;

public class MainActivity3 extends AppCompatActivity {

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main3);
    String desc = getIntent().getStringExtra("desc");
    String busname = getIntent().getStringExtra("roomname");
    String seatno = getIntent().getStringExtra("roomno");
    String id=getIntent().getStringExtra("id");
    EditText editTextDesc = findViewById(R.id.rdesc);
    EditText editTextroomName = findViewById(R.id.roomname);
    EditText roomno = findViewById(R.id.roomno);
    editTextDesc.setText(desc);
    editTextroomName.setText(busname);
    roomno.setText(seatno);
  }
}
```

//SplashScreenActivity.java

```
package com.dzniox.travel_mate.ui.activities;

import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;
import android.widget.Toast;
import com.dzniox.travel_mate.databinding.ActivitySplashScreenBinding;

public class SplashScreenActivity extends AppCompatActivity {
```

```
  private ActivitySplashScreenBinding binding;
  final Handler handler = new Handler();
  @Override
  protected void onCreate(Bundle savedInstanceState) {
     super.onCreate(savedInstanceState);
     binding = ActivitySplashScreenBinding.inflate(getLayoutInflater());
     setContentView(binding.getRoot());
     handler.postDelayed(new Runnable() {
        @Override
        public void run() {
           Intent intent = new Intent(SplashScreenActivity.this, LoginActivity.class);
           startActivity(intent);
   }
     }, 1000);

  }
}
```

//UsersSignUp.java

```
package com.dzniox.travel_mate.ui.activities;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import com.dzniox.travel_mate.R;
import com.google.android.material.navigation.NavigationBarView;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.MenuItem;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import com.google.android.material.bottomnavigation.BottomNavigationView;
import com.google.android.material.navigation.NavigationBarView;

public class UsersSignUp extends AppCompatActivity {

  BottomNavigationView bottomNavigationView;
  TextView goToLoginTextView;

  @Override
  protected void onCreate(Bundle savedInstanceState) {
     super.onCreate(savedInstanceState);
     setContentView(R.layout.activity_users_sign_up);
     bottomNavigationView=findViewById(R.id.bottomNavigationView);
     bottomNavigationView.setOnItemSelectedListener(new
NavigationBarView.OnItemSelectedListener() {
        @Override
        public boolean onNavigationItemSelected(@NonNull MenuItem item) {

           int id=item.getItemId();
           if (id == R.id.Users) {
```

```
            startActivity(new Intent(UsersSignUp.this, SignupActivity.class));
        } else if (id == R.id.Busowners) {
            startActivity(new Intent(UsersSignUp.this, BusSignUp.class));
        }
        else if (id == R.id.Hotelowners) {
            startActivity(new Intent(UsersSignUp.this, HotelSignUp.class));
        }
        return false;

      }

    });
    TextView goToLoginTextView = findViewById(R.id.gotologin);

    goToLoginTextView.setOnClickListener(v -> {
      // Intent to start the Login activity
      Intent intent = new Intent(UsersSignUp.this, LoginActivity.class);
      startActivity(intent);
    });

  }
}
```

//BusSignUp.java

```
package com.dzniox.travel_mate.ui.activities;

import androidx.appcompat.app.AppCompatActivity;
import android.content.pm.PackageManager;
import android.Manifest;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.net.Uri;
import android.os.Bundle;
import android.os.Environment;
import android.provider.MediaStore;
import android.provider.Settings;
import android.util.Base64;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;
import android.content.Context;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import java.io.ByteArrayOutputStream;
import java.io.File;
```

```
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.Locale;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;



public class BusSignUp extends AppCompatActivity {

  private Button SignUp;
  private Context context;
  private File destination = null;
  private ImageView gallery_view;
  private Bitmap bitmap;
  private String imgPath = null;
  private final int CGALLERY = 1;
  private final int CAMERA = 2;
  String eventencodedImage = "", imgname = "";
  private EditText Name,Email,Password,Phone,busaddress;
  private TextView Login;
   private Button ImageBtn;
  APIInterface apiInterface;

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_bus_sign_up);
    apiInterface = APIClient.getClient().create(APIInterface.class);
    SignUp = findViewById(R.id.signup);
    Name = findViewById(R.id.Busowrname);
    Email = findViewById(R.id.Busowremail);
    Phone = findViewById(R.id.Busowrphone);
    busaddress = findViewById(R.id.Busaddress);
    Password = findViewById(R.id.Busowrpassword);
    Login = findViewById(R.id.login);
    ImageBtn=findViewById(R.id.bus_btn_chs);
    gallery_view =findViewById(R.id.bus_gallery_view);

    Login.setOnClickListener(view -> {
      Intent intent = new Intent(this, LoginActivity.class);
      startActivity(intent);
      finish();
    });

    SignUp.setOnClickListener(view -> {
      DoSignUp(Name.getText().toString().trim(),
          Email.getText().toString().trim(),
          Password.getText().toString().trim(),
          Phone.getText().toString().trim(),
          busaddress.getText().toString().trim(),
```

```
            eventencodedImage,
            imgname
        );
    });

    ImageBtn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Log.d("key","btn clicked");
            selectImage();
        }
    });
}
@Override
public void onActivityResult(int requestCode, int resultCode, Intent data) {

    super.onActivityResult(requestCode, resultCode, data);
    if (resultCode == BusSignUp.this.RESULT_CANCELED) {
        return;
    }
    if (requestCode == CGALLERY) {
        if (data != null) {
            Uri contentURI = data.getData();
            try {

                Bitmap bitmap =
MediaStore.Images.Media.getBitmap(BusSignUp.this.getContentResolver(), contentURI);
                ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();
                bitmap.compress(Bitmap.CompressFormat.JPEG, 5, byteArrayOutputStream);
                gallery_view.setVisibility(View.VISIBLE);
                gallery_view.setImageBitmap(bitmap);
                eventencodedImage = Base64.encodeToString(byteArrayOutputStream.toByteArray(),
Base64.DEFAULT);
                imgname = String.valueOf(Calendar.getInstance().getTimeInMillis());


            } catch (IOException e) {
                e.printStackTrace();
                Toast.makeText(BusSignUp.this, "Failed!", Toast.LENGTH_SHORT).show();
            }
        }
    } else if (requestCode == CAMERA) {


        try {
            Uri selectedImage = data.getData();
            bitmap = (Bitmap) data.getExtras().get("data");
            ByteArrayOutputStream bytes = new ByteArrayOutputStream();
            bitmap.compress(Bitmap.CompressFormat.JPEG, 5, bytes);
            String timeStamp = new SimpleDateFormat("yyyyMMdd_HHmmss",
Locale.getDefault()).format(new Date());
            destination = new File(Environment.getExternalStorageDirectory() + "/" +
                    getString(R.string.app_name), "IMG_" + timeStamp + ".jpg");
            FileOutputStream fo;
            try {
                destination.createNewFile();
```

```
            fo = new FileOutputStream(destination);
            fo.write(bytes.toByteArray());
            fo.close();
          } catch (FileNotFoundException e) {
            e.printStackTrace();
          } catch (IOException e) {
            e.printStackTrace();
          }
          eventencodedImage = Base64.encodeToString(bytes.toByteArray(), Base64.DEFAULT);
          imgPath = destination.getAbsolutePath();
          gallery_view.setVisibility(View.VISIBLE);
          gallery_view.setImageBitmap(bitmap);

        } catch (Exception e) {
          e.printStackTrace();
        }

      } else {

      }
    }

    private void DoSignUp(String name, String email, String password, String phone,String
busaddress, String eventencodedImage,String imgname) {
      if (name.equals("") || email.equals("") || password.equals("") || phone.equals("") ||
busaddress.equals("") ||  eventencodedImage.equals("") ){
        Toast.makeText(this,"Check Fields",Toast.LENGTH_LONG).show();
      }else {
        Call<String> call =
apiInterface.signUpBus(email,password,name,phone,busaddress,eventencodedImage,imgname);
        call.enqueue(new Callback<String>() {
          @Override
          public void onResponse(Call<String> call, Response<String> response) {
            if (response.code() == 200){
              GoToLoginActivity();
            }
          }



          @Override
          public void onFailure(Call<String> call, Throwable t) {
          }
        });
      }
    }

    private void GoToLoginActivity() {
        Toast.makeText(this,"SignUp Completed",Toast.LENGTH_LONG).show();
        Intent intent = new Intent(this, LoginActivity.class);
        startActivity(intent);
        finish();
    }

    private void selectImage() {
      try {
```

```
        PackageManager pm = BusSignUp.this.getPackageManager();
        int hasPerm = pm.checkPermission(Manifest.permission.CAMERA,
BusSignUp.this.getPackageName());
        if (hasPerm == PackageManager.PERMISSION_GRANTED) {
           final CharSequence[] options = {"Take Photo", "Choose From Gallery", "Cancel"};
           AlertDialog.Builder builder = new AlertDialog.Builder(BusSignUp.this);
           builder.setTitle("Select Option");
           builder.setItems(options, new DialogInterface.OnClickListener() {
              @Override
              public void onClick(DialogInterface dialog, int item) {
                 if (options[item].equals("Take Photo")) {
                    dialog.dismiss();
                    Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
                    startActivityForResult(intent, CAMERA);
                 } else if (options[item].equals("Choose From Gallery")) {
                    dialog.dismiss();
                    Intent pickPhoto = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
                    startActivityForResult(pickPhoto, CGALLERY);
                 } else if (options[item].equals("Cancel")) {
                    dialog.dismiss();
                 }
              }
           });
           builder.show();
        } else
           Toast.makeText(BusSignUp.this, "Camera Permission error",
Toast.LENGTH_SHORT).show();
     } catch (Exception e) {
        Toast.makeText(BusSignUp.this, "Camera Permission error",
Toast.LENGTH_SHORT).show();
        e.printStackTrace();
     }
  }
}
```

//HotelSignUp.java

```
package com.dzniox.travel_mate.ui.activities;

import androidx.appcompat.app.AppCompatActivity;
import android.content.pm.PackageManager;
import android.Manifest;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.net.Uri;
import android.os.Bundle;
import android.os.Environment;
import android.provider.MediaStore;
import android.provider.Settings;
import android.util.Base64;
import android.util.Log;
```

```
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;
import android.content.Context;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import java.io.ByteArrayOutputStream;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.Locale;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class HotelSignUp extends AppCompatActivity {

  private Button SignUp;
  private Context context;
  private File destination = null;
  private ImageView gallery_view;
  private Bitmap bitmap;
  private String imgPath = null;
  private final int CGALLERY = 1;
  private final int CAMERA = 2;
  String eventencodedImage = "", imgname = "";
  private EditText Name,Email,Password,Phone,Hotelname,Hoteladdress;
  private TextView Login;
  private Button ImageBtn;
  APIInterface apiInterface;

  @Override
  protected void onCreate(Bundle savedInstanceState) {
     super.onCreate(savedInstanceState);
     setContentView(R.layout.activity_hotel_sign_up);
     apiInterface = APIClient.getClient().create(APIInterface.class);

     SignUp = findViewById(R.id.signup);
     Name = findViewById(R.id.hotelownername);
     Hotelname = findViewById(R.id.hotelname);
     Hoteladdress = findViewById(R.id.hoteldesc);
     Email = findViewById(R.id.hotelowneremail);
     Phone = findViewById(R.id.hotelownerphone);
     Password = findViewById(R.id.hotelownerpassword);
     Login = findViewById(R.id.login);
     ImageBtn=findViewById(R.id.hotel_img_chs);
     gallery_view =findViewById(R.id.hotel_gallery_view);
```

```
    Login.setOnClickListener(view -> {
        Intent intent = new Intent(this, LoginActivity.class);
        startActivity(intent);
        finish();
    });

    SignUp.setOnClickListener(view -> {
        DoSignUp(Name.getText().toString().trim(),
                Email.getText().toString().trim(),
                Password.getText().toString().trim(),
                Phone.getText().toString().trim(),
                Hotelname.getText().toString().trim(),
                Hoteladdress.getText().toString().trim(),
                eventencodedImage,
                imgname
        );
    });

    ImageBtn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Log.d("key","btn clicked");
            selectImage();
        }
    });
}

  @Override
  public void onActivityResult(int requestCode, int resultCode, Intent data) {
    Log.d("key","called onstartactivivty");
    super.onActivityResult(requestCode, resultCode, data);
    if (resultCode ==HotelSignUp.this.RESULT_CANCELED) {
        return;
    }
    if (requestCode == CGALLERY) {
        if (data != null) {
            Uri contentURI = data.getData();
            try {

                Bitmap bitmap =
MediaStore.Images.Media.getBitmap(HotelSignUp.this.getContentResolver(), contentURI);
                ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();
                bitmap.compress(Bitmap.CompressFormat.JPEG, 5, byteArrayOutputStream);
                gallery_view.setVisibility(View.VISIBLE);
                gallery_view.setImageBitmap(bitmap);
                eventencodedImage = Base64.encodeToString(byteArrayOutputStream.toByteArray(),
Base64.DEFAULT);
                imgname = String.valueOf(Calendar.getInstance().getTimeInMillis());
            } catch (IOException e) {
                e.printStackTrace();
                Toast.makeText(HotelSignUp.this, "Failed!", Toast.LENGTH_SHORT).show();
            }
        }
    } else if (requestCode == CAMERA) {
        try {
            Uri selectedImage = data.getData();
```

```
            bitmap = (Bitmap) data.getExtras().get("data");
            ByteArrayOutputStream bytes = new ByteArrayOutputStream();
            bitmap.compress(Bitmap.CompressFormat.JPEG, 5, bytes);
            String timeStamp = new SimpleDateFormat("yyyyMMdd_HHmmss",
Locale.getDefault()).format(new Date());
            destination = new File(Environment.getExternalStorageDirectory() + "/" +
                  getString(R.string.app_name), "IMG_" + timeStamp + ".jpg");
            FileOutputStream fo;
            try {
               destination.createNewFile();
               fo = new FileOutputStream(destination);
               fo.write(bytes.toByteArray());
               fo.close();
            } catch (FileNotFoundException e) {
               e.printStackTrace();
            } catch (IOException e) {
               e.printStackTrace();
            }
            eventencodedImage = Base64.encodeToString(bytes.toByteArray(), Base64.DEFAULT);
            imgPath = destination.getAbsolutePath();
            gallery_view.setVisibility(View.VISIBLE);
            gallery_view.setImageBitmap(bitmap);

         } catch (Exception e) {
            e.printStackTrace();
         }

      } else {
      }
   }
   private void DoSignUp(String name, String email, String password, String phone,String
hotelname,String hoteladdress,String eventencodedImage,String imgname ) {
      if (name.equals("") || email.equals("") || password.equals("") || phone.equals("") ||
hoteladdress.equals("") || eventencodedImage.equals("")){
         Toast.makeText(this,"Check Fields",Toast.LENGTH_LONG).show();
      }else {
         Call<String> call =
apiInterface.signUpHotel(email,password,name,phone,hotelname,hoteladdress,eventencodedImage,i
mgname);
         call.enqueue(new Callback<String>() {
            @Override
            public void onResponse(Call<String> call, Response<String> response) {
               if (response.code() == 200){
                  GoToLoginActivity();
               }
            }

            @Override
            public void onFailure(Call<String> call, Throwable t) {
            }
         });
      }
   }
   private void GoToLoginActivity() {
      Toast.makeText(this,"SignUp Completed",Toast.LENGTH_LONG).show();
      Intent intent = new Intent(this, LoginActivity.class);
```

```
    startActivity(intent);
    finish();
  }
  private void selectImage() {

    try {
      PackageManager pm = HotelSignUp.this.getPackageManager();
      int hasPerm = pm.checkPermission(Manifest.permission.CAMERA,
HotelSignUp.this.getPackageName());
      if (hasPerm == PackageManager.PERMISSION_GRANTED) {
        final CharSequence[] options = {"Take Photo", "Choose From Gallery", "Cancel"};
        AlertDialog.Builder builder = new AlertDialog.Builder(HotelSignUp.this);
        builder.setTitle("Select Option");
        builder.setItems(options, new DialogInterface.OnClickListener() {
          @Override
          public void onClick(DialogInterface dialog, int item) {
            if (options[item].equals("Take Photo")) {
              dialog.dismiss();
              Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
              startActivityForResult(intent, CAMERA);
            } else if (options[item].equals("Choose From Gallery")) {
              dialog.dismiss();
              Intent pickPhoto = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
              startActivityForResult(pickPhoto, CGALLERY);
            } else if (options[item].equals("Cancel")) {
              dialog.dismiss();
            }
          }
        });
        builder.show();
      } else
        Toast.makeText(HotelSignUp.this, "Camera Permission error",
Toast.LENGTH_SHORT).show();
    } catch (Exception e) {
      Toast.makeText(HotelSignUp.this, "Camera Permission error",
Toast.LENGTH_SHORT).show();
      e.printStackTrace();
    }
  }
}
```

//AdminViewBookingsActivity.java

```
package com.dzniox.travel_mate.ui.activities;

import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;
import android.content.Context;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.TextView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.Model.Example;
```

```
import com.dzniox.travel_mate.Model.GetPackageData;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import com.dzniox.travel_mate.ui.adapters.AdminViewBookingByOneAdapter;
import com.dzniox.travel_mate.ui.adapters.BusOwnersListAdapter;
import com.google.gson.Gson;
import com.google.gson.JsonObject;
import com.google.gson.JsonParser;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import java.util.ArrayList;
import java.util.List;

import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class AdminViewBookingsActivity extends AppCompatActivity {
  private Context context;
  APIInterface apiInterface;
  PrefUtils prefrences;
  RecyclerView rv_apackage;
  TextView no_data;

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_admin_view_bookings);
    context = this;
    apiInterface = APIClient.getClient().create(APIInterface.class);
    rv_apackage=findViewById(R.id.rv_apackage);
    no_data=findViewById(R.id.no_data);
    GetPackagesByOne();

Log.d("key","@@@
```

```
                }
                Log.d("key",response.body());


Log.d("key","HEYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY");
                if (data.getJSONArray("data").length() > 0) {

Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");


                    no_data.setVisibility(View.GONE);

Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");
                    rv_apackage.setVisibility(View.VISIBLE);

Log.d("key","adStart**********************************************************************");
//
                    AdminViewBookingByOneAdapter adapter = new
AdminViewBookingByOneAdapter(context, data.getJSONArray("data"));

Log.d("key","adEnd************************************************************************");
                    rv_apackage.setLayoutManager(new LinearLayoutManager(context));
                    rv_apackage.setAdapter(adapter);
                } else {
                    no_data.setVisibility(View.VISIBLE);
                    rv_apackage.setVisibility(View.GONE);
                }
            } catch (Exception e) {
                no_data.setVisibility(View.VISIBLE);
                rv_apackage.setVisibility(View.GONE);
            }

        } else {
            no_data.setVisibility(View.VISIBLE);
            rv_apackage.setVisibility(View.GONE);
        }
    }

    @Override
    public void onFailure(Call<String> call, Throwable t) {
        no_data.setVisibility(View.VISIBLE);
        rv_apackage.setVisibility(View.GONE);
    }
  });
 }
}
```

//Booking_Activity.java

```
package com.dzniox.travel_mate.ui.activities;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.widget.Button;
```

```
import android.widget.EditText;
import android.widget.Toast;
import com.dzniox.travel_mate.Model.ResponseCommon;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.utils.CustPrograssbar;
import com.google.gson.Gson;
import com.google.gson.JsonObject;
import com.google.gson.JsonParser;
import com.dzniox.travel_mate.utils.CustPrograssbar;
import org.json.JSONException;
import org.json.JSONObject;
import com.dzniox.travel_mate.databinding.ActivityBookingBinding;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.Locale;
public class Booking_Activity extends AppCompatActivity {
private Button book;
  ActivityBookingBinding binding;
private EditText Cvv,Vlddate,Cnum;
  @Override
  protected void onCreate(Bundle savedInstanceState) {
     super.onCreate(savedInstanceState);
     setContentView(R.layout.activity_booking);
     book=findViewById(R.id.book);
     Cvv=findViewById(R.id.cvv);
     Vlddate=findViewById(R.id.vlddate);
     Cnum=findViewById(R.id.cnum);
     CustPrograssbar custPrograssbar;
     custPrograssbar = new CustPrograssbar();
     SimpleDateFormat sdf = new SimpleDateFormat("yyyy.MM.dd G 'at' HH:mm:ss z");
     String currentDateandTime = sdf.format(new Date());
     // Get the current date
     Date currentDate = Calendar.getInstance().getTime();

     // Format the date as a string in Django date format (YYYY-MM-DD)
     SimpleDateFormat dateFormat = new SimpleDateFormat("yyyy-MM-dd", Locale.getDefault());
     String formattedDate = dateFormat.format(currentDate);

     String packageid = getIntent().getStringExtra("packageid");
     String uid=getIntent().getStringExtra("uid");
     String amount=getIntent().getStringExtra("amount");
     String num=getIntent().getStringExtra("numppl");
     Log.d("key",packageid+"  "+uid+"  "+amount+"  "+"  "+num);
     int pamount=Integer.parseInt(amount);
     int pnum=Integer.parseInt(num);
     int total=pamount*pnum;

     book.setOnClickListener(view -> {
        if (Cnum.getText().toString().trim().equals("") ||
```

```
            Cvv.getText().toString().trim().equals("") ||
Vlddate.getText().toString().trim().equals("")) {
            Toast.makeText(this, "Check Fields!!!!!", Toast.LENGTH_LONG).show();
        } else {
            custPrograssbar.progressCreate(this);
            custPrograssbar.setCancel(false);
            JSONObject jsonObject = new JSONObject();
            try {
                jsonObject.put("pkgid", packageid);
                jsonObject.put("uid", uid);
                jsonObject.put("date", formattedDate);
                Log.d("date",currentDateandTime);
                jsonObject.put("totalppl", num);
                jsonObject.put("amount", total);

                JsonParser jsonParser = new JsonParser();
                Call<JsonObject> call = ApiClientCustom.getInterface().addbooking((JsonObject)
jsonParser.parse(jsonObject.toString()));
                call.enqueue(new Callback<JsonObject>() {
                    @Override
                    public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
                        custPrograssbar.close();
                        JsonObject object = response.body();
                        Gson gson = new Gson();
                        ResponseCommon example = gson.fromJson(object.toString(),
ResponseCommon.class);
                        Toast.makeText(Booking_Activity.this, example.getResponseMsg(),
Toast.LENGTH_SHORT).show();
                    }

                    @Override
                    public void onFailure(Call<JsonObject> call, Throwable t) {

                    }
                });
            } catch (JSONException e) {
                e.printStackTrace();
            }


        }
    });
  }
}
```

//EditBusActivity.java

```
package com.dzniox.travel_mate.ui.activities;

import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentTransaction;
import android.Manifest;
```

```
import android.app.Activity;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.graphics.drawable.BitmapDrawable;
import android.net.Uri;
import android.os.Bundle;
import android.os.Environment;
import android.provider.MediaStore;
import android.util.Base64;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Toast;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.Model.ResponseCommon;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.ui.fragments.ViewBusesFragment;
import com.dzniox.travel_mate.utils.CustPrograssbar;
import com.google.gson.Gson;
import com.google.gson.JsonObject;
import com.google.gson.JsonParser;
import org.json.JSONException;
import org.json.JSONObject;
import java.io.ByteArrayOutputStream;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.Locale;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class EditBusActivity extends AppCompatActivity {
  EditText addbusname,addbownerdesc,addbownerseat_no;
  ImageView gallery_view;
  Button addbownerbtn_choose,addbowneradd_btn;
  CustPrograssbar custPrograssbar;
  EditText FROM,To;
  String seats = "";
  Button ImageBtn;
  private File destination = null;
  private Bitmap bitmap;
  private String imgPath = null;
  private final int CGALLERY = 1;
  private final int CAMERA = 2;
```

```
String eventencodedImage = "", imgname = "";
private Context context=this;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_edit_bus);
    custProgressbar = new CustProgressbar();
    String desc = getIntent().getStringExtra("desc");
    String busname = getIntent().getStringExtra("busname");
    String seatno = getIntent().getStringExtra("seatno");
    String id=getIntent().getStringExtra("id");
    String image=getIntent().getStringExtra("image");
    addbusname = findViewById(R.id.addbusname);
    addbownerdesc = findViewById(R.id.addbownerdesc);
    addbownerseat_no = findViewById(R.id.addbownerseat_no);
    addbownerbtn_choose =findViewById(R.id.addbownerbtn_choose);
    addbowneradd_btn = findViewById(R.id.addbowneradd_btn);
    gallery_view = findViewById(R.id.gallery_view);
    addbusname.setText(busname);
    addbownerdesc.setText(desc);
    addbownerseat_no.setText(seatno);
    Log.d("key","Here is
Fragment@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@");
    Glide.with(context)
            .load(ApiClientCustom.baseUrl + "uploads/" +image)
            .into(gallery_view);
    addbownerbtn_choose.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Log.d("key","btn clicked");
            selectImage();
        }
    });

    addbowneradd_btn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Log.d("key","btn clicked");
            addToserver();
        }
    });
}

@Override
public void onActivityResult(int requestCode, int resultCode, Intent data) {

    super.onActivityResult(requestCode, resultCode, data);
    if (resultCode == Activity.RESULT_CANCELED) {
        return;
    }
    if (requestCode == CGALLERY) {
        if (data != null) {
            Uri contentURI = data.getData();
            try {
```

```
            Bitmap bitmap = MediaStore.Images.Media.getBitmap(this.getContentResolver(),
contentURI);
            ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();
            bitmap.compress(Bitmap.CompressFormat.JPEG, 5, byteArrayOutputStream);
            gallery_view.setVisibility(View.VISIBLE);
            gallery_view.setImageBitmap(bitmap);
            eventencodedImage = Base64.encodeToString(byteArrayOutputStream.toByteArray(),
Base64.DEFAULT);
            imgname = String.valueOf(Calendar.getInstance().getTimeInMillis());
          } catch (IOException e) {
            e.printStackTrace();
            Toast.makeText(this, "Failed!", Toast.LENGTH_SHORT).show();
          }
        }
    } else if (requestCode == CAMERA) {
        try {
          Uri selectedImage = data.getData();
          bitmap = (Bitmap) data.getExtras().get("data");
          ByteArrayOutputStream bytes = new ByteArrayOutputStream();
          bitmap.compress(Bitmap.CompressFormat.JPEG, 5, bytes);
          String timeStamp = new SimpleDateFormat("yyyyMMdd_HHmmss",
Locale.getDefault()).format(new Date());
          destination = new File(Environment.getExternalStorageDirectory() + "/" +
              getString(R.string.app_name), "IMG_" + timeStamp + ".jpg");
          FileOutputStream fo;
          try {
            destination.createNewFile();
            fo = new FileOutputStream(destination);
            fo.write(bytes.toByteArray());
            fo.close();
          } catch (FileNotFoundException e) {
            e.printStackTrace();
          } catch (IOException e) {
            e.printStackTrace();
          }
          eventencodedImage = Base64.encodeToString(bytes.toByteArray(), Base64.DEFAULT);
          imgPath = destination.getAbsolutePath();
          gallery_view.setVisibility(View.VISIBLE);
          gallery_view.setImageBitmap(bitmap);

        } catch (Exception e) {
          e.printStackTrace();
        }

    } else {

    }
  }

  private void selectImage() {
    try {
      PackageManager pm = this.getPackageManager();
      int hasPerm = pm.checkPermission(Manifest.permission.CAMERA, this.getPackageName());
      if (hasPerm == PackageManager.PERMISSION_GRANTED) {
        final CharSequence[] options = {"Take Photo", "Choose From Gallery", "Cancel"};
        AlertDialog.Builder builder = new AlertDialog.Builder(this);
```

```
            builder.setTitle("Select Option");
            builder.setItems(options, new DialogInterface.OnClickListener() {
               @Override
               public void onClick(DialogInterface dialog, int item) {
                  if (options[item].equals("Take Photo")) {
                     dialog.dismiss();
                     Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
                     startActivityForResult(intent, CAMERA);
                  } else if (options[item].equals("Choose From Gallery")) {
                     dialog.dismiss();
                     Intent pickPhoto = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
                     startActivityForResult(pickPhoto, CGALLERY);
                  } else if (options[item].equals("Cancel")) {
                     dialog.dismiss();
                  }
               }
            });
            builder.show();
         } else
            Toast.makeText(this, "Camera Permission error", Toast.LENGTH_SHORT).show();
      } catch (Exception e) {
         Toast.makeText(this, "Camera Permission error", Toast.LENGTH_SHORT).show();
         e.printStackTrace();
      }
   }




   private void addToserver() {
      Log.d("key","helloo called!!!!!!!!!!!!!!!!!!!");
      custPrograssbar.progressCreate(this);
      custPrograssbar.setCancel(false);
      JSONObject jsonObject = new JSONObject();
      Log.d("key","helloo called!!!!!!!!!!!!!!!!!!!");

      try {
         Log.d("key","_____________________________");
         Log.d("key",eventencodedImage);
         Log.d("key","_____________________________");

         jsonObject.put("id",getIntent().getStringExtra("id"));
         jsonObject.put("busname",addbusname.getText().toString() );
         jsonObject.put("desc", addbownerdesc.getText().toString());
         jsonObject.put("seats", addbownerseat_no.getText().toString());


         if (eventencodedImage != "") {
            jsonObject.put("image", eventencodedImage);
            jsonObject.put("imagename", imgname);
            Log.d("key","it is  not empty!!!!!!!!!!!!!!!!!!!");


         } else {
```

```
            Log.d("key","it is   empty!!!!!!!!!!!!!!!!!!");
            Bitmap bitmap = ((BitmapDrawable) gallery_view.getDrawable()).getBitmap();
            ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();
            bitmap.compress(Bitmap.CompressFormat.JPEG, 5, byteArrayOutputStream);
            String imageInGalleryView =
Base64.encodeToString(byteArrayOutputStream.toByteArray(), Base64.DEFAULT);
            jsonObject.put("image", imageInGalleryView);
            String galleryViewImgName = "gallery_view_img_" +
Calendar.getInstance().getTimeInMillis();
            jsonObject.put("imagename", galleryViewImgName);
        }

        JsonParser jsonParser = new JsonParser();
        Call<JsonObject> call = ApiClientCustom.getInterface().Editbuses((JsonObject)
jsonParser.parse(jsonObject.toString()));
        call.enqueue(new Callback<JsonObject>() {
            @Override
            public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
                custPrograssbar.close();
                JsonObject object = response.body();
                Gson gson = new Gson();
                ResponseCommon example = gson.fromJson(object.toString(), ResponseCommon.class);
                Toast.makeText(EditBusActivity.this, example.getResponseMsg(),
Toast.LENGTH_SHORT).show();
            }
            @Override
            public void onFailure(Call<JsonObject> call, Throwable t) {

            }
        });
    } catch (JSONException e) {
        e.printStackTrace();
    }
  }
}
```

//EditPackageActivity.java

```
package com.dzniox.travel_mate.ui.activities;

import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import android.Manifest;
import android.app.Activity;
import android.app.AlertDialog;
import android.app.DatePickerDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.graphics.drawable.BitmapDrawable;
```

```
import android.net.Uri;
import android.os.Bundle;
import android.os.Environment;
import android.provider.MediaStore;
import android.util.Base64;
import android.util.Log;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Spinner;
import android.widget.Toast;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.Model.AllBusbyadmin;
import com.dzniox.travel_mate.Model.AllHotelsbyAdmin;
import com.dzniox.travel_mate.Model.AllroomsbyOwner;
import com.dzniox.travel_mate.Model.Example;
import com.dzniox.travel_mate.Model.GetPackageData;
import com.dzniox.travel_mate.Model.GetbusSeats;
import com.dzniox.travel_mate.Model.ResponseCommon;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.NetworkUtilsCustom.GetResult;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.databinding.ActivityEditPackageBinding;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import com.dzniox.travel_mate.ui.adapters.BusOwnersListAdapter;
import com.dzniox.travel_mate.utils.CustPrograssbar;
import com.google.gson.Gson;
import com.google.gson.JsonObject;
import com.google.gson.JsonParser;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import java.io.ByteArrayOutputStream;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.Date;
import java.util.List;
import java.util.Locale;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class EditPackageActivity extends AppCompatActivity implements View.OnClickListener,
GetResult.MyListener{

  ActivityEditPackageBinding binding;
```

```
List<AllBusbyadmin> ShopList;
APIInterface apiInterface;
PrefUtils prefrences;
List<AllroomsbyOwner> RoomList;
ArrayList<AllroomsbyOwner> ROOMLIST = new ArrayList<>();
ArrayList<AllBusbyadmin> shopLIST = new ArrayList<>();
List<AllHotelsbyAdmin> ShopList1;
ArrayList<AllHotelsbyAdmin> shopLIST1 = new ArrayList<>();
List<GetPackageData> pkg;
ArrayList<GetPackageData> PKG = new ArrayList<>();
String hotel_id, bus_id,room_id;
final Calendar myCalendar = Calendar.getInstance();
CustPrograssbar custPrograssbar;
EditText FROM,To;
String seats = "";
Button ImageBtn;
ImageView gallery_view;
private File destination = null;
private Bitmap bitmap;
private String imgPath = null;
private final int CGALLERY = 1;
private final int CAMERA = 2;
String eventencodedImage = "", imgname = "";
String y;
private Context context=this;

@Override
protected void onCreate(Bundle savedInstanceState) {
   super.onCreate(savedInstanceState);
   Log.d("key","!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!");
   setContentView(R.layout.activity_edit_package);
   FROM =findViewById(R.id.editfrom);
   To=findViewById(R.id.editto);
   Log.d("key","!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!");
   binding = ActivityEditPackageBinding.inflate(getLayoutInflater());

   View view = binding.getRoot();
   setContentView(view);

   custPrograssbar = new CustPrograssbar();
   ImageBtn=findViewById(R.id.img);
   gallery_view =findViewById(R.id.gallery_view);
   binding.rooms.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
      @Override
      public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {
         AllroomsbyOwner b1 = (AllroomsbyOwner) parent.getSelectedItem();
         room_id = b1.getRoom_id();
      }

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

      }
   });
```

```
    binding.busedit.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
        @Override
        public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {
            AllBusbyadmin b1 = (AllBusbyadmin) parent.getSelectedItem();
            Log.d("key","this is bi"+b1);
            JSONObject jsonObject = new JSONObject();

            bus_id = b1.getBus_id();
            Log.d("key","#####################################################");
            try {
                jsonObject.put("busid",bus_id);
                JsonParser jsonParser = new JsonParser();
                Call<JsonObject> call = ApiClientCustom.getInterface().getbusseats((JsonObject)
jsonParser.parse(jsonObject.toString()));

                call.enqueue(new Callback<JsonObject>() {
                    @Override
                    public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
                        JsonObject object = response.body();
                        Gson gson = new Gson();
                        Example example = gson.fromJson(object.toString(), Example.class);
                        pkg = new ArrayList<>();
                        if (example.getResultData().getbusSeats() != null){
                            List<GetbusSeats> pakg = example.getResultData().getbusSeats();
                            Log.d("key","HEAREIS THE DATA_________________!!!!!!!!!!!!!!_____________");

                            for (GetbusSeats pkg : pakg) {
                                seats = pkg.getSeats();

Log.d("key","data@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@"+seats
);
                                binding.avlsts.setText(seats);
                            }

                        }
                    }
                    @Override
                    public void onFailure(Call<JsonObject> call, Throwable t) {

                    }
                });
            } catch (JSONException e) {
                e.printStackTrace();
            }
        }

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

        }
    });
    binding.hotel.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
        @Override
        public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {
            AllHotelsbyAdmin s1 = (AllHotelsbyAdmin) parent.getSelectedItem();
```

```
            hotel_id = s1.getHotel_id();

            JSONObject jsonObject = new JSONObject();
            try {
               jsonObject.put("hid", hotel_id);
               JsonParser jsonParser = new JsonParser();
               Call<JsonObject> call = ApiClientCustom.getInterface().getAllroomsbyid((JsonObject)
jsonParser.parse(jsonObject.toString()));
               call.enqueue(new Callback<JsonObject>() {
                  @Override
                  public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
                     JsonObject object = response.body();
                     Gson gson = new Gson();
                     Example example = gson.fromJson(object.toString(), Example.class);
                     RoomList = new ArrayList<>();
                     if(example.getResultData().getAllroomsbyowner() == null){

Log.d("key","@@@
```

```
        }
    });

    ImageBtn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Log.d("key","btn clicked");
            selectImage();

        }
    });
    binding.startdate.setOnClickListener(this);
    binding.enddate.setOnClickListener(this);
    binding.submitnew.setOnClickListener(this);
    getpackageId();
    getAllBus();
    getAllSpots();
    FROM.setText("helloooo");
    To.setText("heloooo");
  }

  private void getpackageId(){
    JSONObject jsonObject = new JSONObject();
    String packageid = getIntent().getStringExtra("packageid");
    try {
        jsonObject.put("packageid",packageid);
        JsonParser jsonParser = new JsonParser();
        Call<JsonObject> call = ApiClientCustom.getInterface().getPackageById((JsonObject)
jsonParser.parse(jsonObject.toString()));

        call.enqueue(new Callback<JsonObject>() {
            @Override
            public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
                JsonObject object = response.body();
                Gson gson = new Gson();
                Example example = gson.fromJson(object.toString(), Example.class);
                pkg = new ArrayList<>();
                if (example.getResultData().getpackagebyid() != null) {
                    List<GetPackageData> pakg = example.getResultData().getpackagebyid();
                   Log.d("key","HEAREIS THE DATA_________________!!!!!!!!!!!!!!_____________");
                    for (GetPackageData pkg : pakg) {

                        String busname = pkg.getBus_Name();
                        String hotelName = pkg.getHotel_Name();
                        String frm=pkg.getFrom_place();
                        String to= pkg.getTo_place();
                        String srtd=pkg.getStart_date();
                        String endd=pkg.getEnd_date();
                        String desc=pkg.getDesc();
                        String amount=pkg.getamount();
                        String img=pkg.getImage();
                        binding.editfrom.setText(frm);
                        binding.editto.setText(to);
                        binding.startdate.setText(srtd);
                        binding.enddate.setText(endd);
                        binding.pdesc.setText(desc);
```

```
                    binding.allotBpoint.setText(amount);
                    if (context != null) {
                        Glide.with(context)
                                .load(ApiClientCustom.baseUrl + "uploads/" + img)
                                .into(gallery_view);
                    }
                }
            }
        }
        @Override
        public void onFailure(Call<JsonObject> call, Throwable t) {
        }
      });
    } catch (JSONException e) {
      e.printStackTrace();
    }
  }
  private void getAllSpots() {
    JSONObject jsonObject = new JSONObject();
    try {
      jsonObject.put("ClientID", "0");
      JsonParser jsonParser = new JsonParser();
      Call<JsonObject> call = ApiClientCustom.getInterface().getAllhotelssid((JsonObject)
jsonParser.parse(jsonObject.toString()));
      call.enqueue(new Callback<JsonObject>() {
        @Override
        public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
          JsonObject object = response.body();
          Gson gson = new Gson();
          Example example = gson.fromJson(object.toString(), Example.class);
          ShopList1 = new ArrayList<>();
          pkg = new ArrayList<>();
          if (example.getResultData().getAllhotelsbyadmin() != null) {
            List<AllHotelsbyAdmin> hotelsList = example.getResultData().getAllhotelsbyadmin();
            Log.d("key","Hear is data !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!");
            ShopList1 = example.getResultData().getAllhotelsbyadmin();
            int i = 1;
            for (AllHotelsbyAdmin sn : ShopList1) {
              shopLIST1.add(new AllHotelsbyAdmin(String.valueOf(sn.getHotel_id()),
sn.getHotel_Name()));
              i = i + 1;
            }
            ArrayAdapter<AllHotelsbyAdmin> adapter = new
ArrayAdapter<AllHotelsbyAdmin>(EditPackageActivity.this, R.layout.spinner_def, shopLIST1);
            binding.hotel.setAdapter(adapter);
            Log.d("key","$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$");
            binding.hotel.setTitle("Select Item");
            binding.hotel.setPositiveButton("OK");
          }
        }

        @Override
        public void onFailure(Call<JsonObject> call, Throwable t) {
        }
      });
    } catch (JSONException e) {
```

```
            e.printStackTrace();
        }
    }

    private void getAllBus() {
        JSONObject jsonObject = new JSONObject();
        try {
            jsonObject.put("ClientID", "0");
            JsonParser jsonParser = new JsonParser();
            Call<JsonObject> call = ApiClientCustom.getInterface().getAllBusbyid((JsonObject)
jsonParser.parse(jsonObject.toString()));
            call.enqueue(new Callback<JsonObject>() {
                @Override
                public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
                    JsonObject object = response.body();
                    Gson gson = new Gson();
                    Example example = gson.fromJson(object.toString(), Example.class);
                    ShopList = new ArrayList<>();
                    pkg = new ArrayList<>();
                    if (example.getResultData().getAllBusbyadmins() != null) {

                        ShopList = example.getResultData().getAllBusbyadmins();
                        int i = 1;
                        for (AllBusbyadmin sn : ShopList) {
 shopLIST.add(new AllBusbyadmin(String.valueOf(sn.getBus_id()), sn.getBus_Name() + " | " +
sn.getSeats(), sn.getSeats()));
                            i = i + 1;

                        }
                        ArrayAdapter<AllBusbyadmin> adapter = new
ArrayAdapter<AllBusbyadmin>(EditPackageActivity.this, R.layout.spinner_def, shopLIST);
                        binding.busedit.setAdapter(adapter);
                        binding.busedit.setTitle("Select Item");
                        binding.busedit.setPositiveButton("OK");
                        Log.d("key", "Size of shopLIST: " + shopLIST.size());
                        Log.d("key", "Before setting selection");
                        binding.busedit.setSelection(0);
                        Log.d("key", "After setting selection");
                        Log.d("key", "Size of shopLIST: " +    shopLIST.indexOf(0));
                    }

                    JSONObject jsonObject = new JSONObject();
                    String packageid = getIntent().getStringExtra("packageid");
                   try {

                        jsonObject.put("packageid",packageid);
                        JsonParser jsonParser = new JsonParser();
                        Call<JsonObject> call2 =
ApiClientCustom.getInterface().getPackageById((JsonObject)
jsonParser.parse(jsonObject.toString()));
                        call2.enqueue(new Callback<JsonObject>() {
                            @Override
                            public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
                                JsonObject object = response.body();
                                Gson gson = new Gson();
                                Example example = gson.fromJson(object.toString(), Example.class);
```

```
                    pkg = new ArrayList<>();
                    if (example.getResultData().getpackagebyid() != null) {
                        List<GetPackageData> pakg = example.getResultData().getpackagebyid();
                        Log.d("key","HEAREIS THE
DATA_____!!!!!!!!!!!!!!!_____");
                        for (GetPackageData pkg : pakg) {
                            String id = pkg.getBus_Name();
                            String hotelName = pkg.getHotel_Name();
                            Log.d("key", "Size of !!!!!!!!!!!!!JKKKKKKKKK!!!!!!!!!!!!!!!!!!: " +
shopLIST.indexOf(id));
                            Log.d("key","This is Bus name"+id);
                            String frm=pkg.getFrom_place();
                            String to= pkg.getTo_place();
                            String srtd=pkg.getStart_date();
                            String endd=pkg.getEnd_date();
                            String desc=pkg.getDesc();
                            String amt=pkg.getamount();
//                              String img=pkg.getImage();
                            binding.editfrom.setText(frm);
                            binding.editto.setText(to);
                            binding.startdate.setText(srtd);
                            binding.enddate.setText(endd);
                            binding.pdesc.setText(desc);
                        }
                    }
                }
                @Override
                public void onFailure(Call<JsonObject> call, Throwable t) {

                }
            });
        } catch (JSONException e) {
            e.printStackTrace();
        }
    }

    @Override
    public void onFailure(Call<JsonObject> call, Throwable t) {

    }
  });
 } catch (JSONException e) {
  e.printStackTrace();
 }
}

@Override
public void onClick(View v) {
 switch (v.getId()) {
  case R.id.startdate:
   showDatepopupFromstart();
   break;
  case R.id.enddate:
   showDatepopupFromend();
   Log.d("key","###############!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!");
   break;
```

```
        case R.id.submitnew:
              addToserver();
            break;
      }
  }

  private void showDatepopupFromstart() {
      DatePickerDialog.OnDateSetListener date = new DatePickerDialog.OnDateSetListener() {
        @Override
        public void onDateSet(DatePicker view, int year, int monthOfYear,
                      int dayOfMonth) {
          // TODO Auto-generated method stub
          myCalendar.set(Calendar.YEAR, year);
          myCalendar.set(Calendar.MONTH, monthOfYear);
          myCalendar.set(Calendar.DAY_OF_MONTH, dayOfMonth);
          updateLabelstart();
        }
      };

      DatePickerDialog dialog = new DatePickerDialog(this, date, myCalendar
            .get(Calendar.YEAR), myCalendar.get(Calendar.MONTH),
            myCalendar.get(Calendar.DAY_OF_MONTH));
      dialog.getDatePicker().setMinDate(System.currentTimeMillis());
      dialog.show();
  }
  private void showDatepopupFromend() {

Log.d("key","########################!!!!!!!!!!!!!!!!!$$$$$$##################################
#");
      DatePickerDialog.OnDateSetListener date = new DatePickerDialog.OnDateSetListener() {
        @Override
        public void onDateSet(DatePicker view, int year, int monthOfYear,
                      int dayOfMonth) {
          myCalendar.set(Calendar.YEAR, year);
          myCalendar.set(Calendar.MONTH, monthOfYear);
          myCalendar.set(Calendar.DAY_OF_MONTH, dayOfMonth);
          updateLabelend();
        }
      };

      DatePickerDialog dialog = new DatePickerDialog(this, date, myCalendar
            .get(Calendar.YEAR), myCalendar.get(Calendar.MONTH),
            myCalendar.get(Calendar.DAY_OF_MONTH));
      dialog.getDatePicker().setMinDate(System.currentTimeMillis());
      dialog.show();
  }
  private void updateLabelstart() {
      String myFormat = "dd-MM-yyyy";
      SimpleDateFormat sdf = new SimpleDateFormat(myFormat, Locale.US);
      binding.startdate.setText(sdf.format(myCalendar.getTime()));
  }
  private void updateLabelend() {

Log.d("key","############################&&&&&&&&&&&&&&&&&&&&&&&&##########
##############");
```

```
    String myFormat = "dd-MM-yyyy";
    SimpleDateFormat sdf = new SimpleDateFormat(myFormat, Locale.US);
    binding.enddate.setText(sdf.format(myCalendar.getTime()));
  }

  @Override
  public void callback(JsonObject result, String callNo) {
  }

  @Override
  public void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if (resultCode == Activity.RESULT_CANCELED) {
      return;
    }
    if (requestCode == CGALLERY) {
      if (data != null) {
        Uri contentURI = data.getData();
        try {
          Bitmap bitmap = MediaStore.Images.Media.getBitmap(this.getContentResolver(),
contentURI);
          ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();
          bitmap.compress(Bitmap.CompressFormat.JPEG, 5, byteArrayOutputStream);
          gallery_view.setVisibility(View.VISIBLE);
          gallery_view.setImageBitmap(bitmap);
          eventencodedImage = Base64.encodeToString(byteArrayOutputStream.toByteArray(),
Base64.DEFAULT);
          imgname = String.valueOf(Calendar.getInstance().getTimeInMillis());
        } catch (IOException e) {
          e.printStackTrace();
          Toast.makeText(this, "Failed!", Toast.LENGTH_SHORT).show();
        }
      }
    } else if (requestCode == CAMERA) {

      try {
        Uri selectedImage = data.getData();
        bitmap = (Bitmap) data.getExtras().get("data");
        ByteArrayOutputStream bytes = new ByteArrayOutputStream();
        bitmap.compress(Bitmap.CompressFormat.JPEG, 5, bytes);
        String timeStamp = new SimpleDateFormat("yyyyMMdd_HHmmss",
Locale.getDefault()).format(new Date());
        destination = new File(Environment.getExternalStorageDirectory() + "/" +
            getString(R.string.app_name), "IMG_" + timeStamp + ".jpg");
        FileOutputStream fo;
        try {
          destination.createNewFile();
          fo = new FileOutputStream(destination);
          fo.write(bytes.toByteArray());
          fo.close();
        } catch (FileNotFoundException e) {
          e.printStackTrace();
        } catch (IOException e) {
          e.printStackTrace();
        }
        eventencodedImage = Base64.encodeToString(bytes.toByteArray(), Base64.DEFAULT);
```

```
            imgPath = destination.getAbsolutePath();
            gallery_view.setVisibility(View.VISIBLE);
            gallery_view.setImageBitmap(bitmap);

        } catch (Exception e) {
            e.printStackTrace();
        }

    } else {
    }
}

private void selectImage() {
    try {
        PackageManager pm = this.getPackageManager();
        int hasPerm = pm.checkPermission(Manifest.permission.CAMERA, this.getPackageName());
        if (hasPerm == PackageManager.PERMISSION_GRANTED) {
            final CharSequence[] options = {"Take Photo", "Choose From Gallery", "Cancel"};
            AlertDialog.Builder builder = new AlertDialog.Builder(this);
            builder.setTitle("Select Option");
            builder.setItems(options, new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int item) {
                    if (options[item].equals("Take Photo")) {
                        dialog.dismiss();
                        Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
                        startActivityForResult(intent, CAMERA);
                    } else if (options[item].equals("Choose From Gallery")) {
                        dialog.dismiss();
                        Intent pickPhoto = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
                        startActivityForResult(pickPhoto, CGALLERY);
                    } else if (options[item].equals("Cancel")) {
                        dialog.dismiss();
                    }
                }
            });
            builder.show();
        } else
            Toast.makeText(this, "Camera Permission error", Toast.LENGTH_SHORT).show();
    } catch (Exception e) {
        Toast.makeText(this, "Camera Permission error", Toast.LENGTH_SHORT).show();
        e.printStackTrace();
    }
}

private void addToserver() {
    custPrograssbar.progressCreate(this);
    custPrograssbar.setCancel(false);
    JSONObject jsonObject = new JSONObject();
    String pkg = getIntent().getStringExtra("packageid");
    Log.d("key",pkg);
    try {
        jsonObject.put("packageid",pkg);
        jsonObject.put("hotelid", hotel_id);
        jsonObject.put("bus_id", bus_id);
```

```
        jsonObject.put("room_id", room_id);
        jsonObject.put("seats", binding.avlsts.getText().toString());
        jsonObject.put("sday", binding.startdate.getText().toString());
        jsonObject.put("eday", binding.enddate.getText().toString());
//          jsonObject.put("city", binding.spnCity.getSelectedItem());
        jsonObject.put("amount", binding.allotBpoint.getText().toString().trim());
        jsonObject.put("from", binding.editfrom.getText().toString().trim());
        jsonObject.put("to", binding.editto.getText().toString().trim());
        jsonObject.put("desc", binding.pdesc.getText().toString().trim());
        if (eventencodedImage != "") {
            jsonObject.put("image", eventencodedImage);
            jsonObject.put("imagename", imgname);
            Log.d("key","it is  not empty!!!!!!!!!!!!!!!!!!!");


        } else {
            Log.d("key","it is   empty!!!!!!!!!!!!!!!!!!!");
            Bitmap bitmap = ((BitmapDrawable) gallery_view.getDrawable()).getBitmap();
            ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();
            bitmap.compress(Bitmap.CompressFormat.JPEG, 5, byteArrayOutputStream);
            String imageInGalleryView =
Base64.encodeToString(byteArrayOutputStream.toByteArray(), Base64.DEFAULT);
            jsonObject.put("image", imageInGalleryView);
            String galleryViewImgName = "gallery_view_img_" +
Calendar.getInstance().getTimeInMillis();
            jsonObject.put("imagename", galleryViewImgName);
        }

        Log.d("key",binding.pdesc.getText().toString().trim());

        JsonParser jsonParser = new JsonParser();
        Call<JsonObject> call = ApiClientCustom.getInterface().EditaddTripassign((JsonObject)
jsonParser.parse(jsonObject.toString()));
        call.enqueue(new Callback<JsonObject>() {
            @Override
            public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
                custPrograssbar.close();
                JsonObject object = response.body();
                Gson gson = new Gson();
                ResponseCommon example = gson.fromJson(object.toString(), ResponseCommon.class);
                Toast.makeText(EditPackageActivity.this, example.getResponseMsg(),
Toast.LENGTH_SHORT).show();
            }

            @Override
            public void onFailure(Call<JsonObject> call, Throwable t) {

            }
        });
    } catch (JSONException e) {
        e.printStackTrace();
    }
  }
```

//HotelEditActivity.java

```
package com.dzniox.travel_mate.ui.activities;

import androidx.appcompat.app.AppCompatActivity;

import android.Manifest;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.graphics.drawable.BitmapDrawable;
import android.net.Uri;
import android.os.Bundle;
import android.os.Environment;
import android.provider.MediaStore;
import android.util.Base64;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Toast;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.Model.ResponseCommon;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.utils.CustPrograssbar;
import com.google.gson.Gson;
import com.google.gson.JsonObject;
import com.google.gson.JsonParser;
import org.json.JSONException;
import org.json.JSONObject;
import java.io.ByteArrayOutputStream;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.Locale;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class HotelEditActivity extends AppCompatActivity {
  EditText roomname,addbownerdesc,addbownerroom_no;
  ImageView gallery_view;
  Button addbownerbtn_choose,addbowneradd_btn;
  CustPrograssbar custPrograssbar;
  EditText FROM,To;
  String seats = "";
  Button ImageBtn;
```

```
  private File destination = null;

  private Bitmap bitmap;
  private String imgPath = null;
  private final int CGALLERY = 1;
  private final int CAMERA = 2;
  String eventencodedImage = "", imgname = "";
  private Context context=this;

  @Override
  protected void onCreate(Bundle savedInstanceState) {
     super.onCreate(savedInstanceState);
     setContentView(R.layout.activity_hotel_edit);
     custPrograssbar = new CustPrograssbar();
     String desc = getIntent().getStringExtra("desc");
     String room_name = getIntent().getStringExtra("roomname");
     String roomno = getIntent().getStringExtra("roomno");
     String id=getIntent().getStringExtra("id");
     String image=getIntent().getStringExtra("image");
     roomname = findViewById(R.id.roomname);
     addbownerdesc = findViewById(R.id.addbownerdesc);
     addbownerroom_no = findViewById(R.id.addbownerroom_no);
     addbownerbtn_choose =findViewById(R.id.addbownerbtn_choose);
     addbowneradd_btn = findViewById(R.id.addbowneradd_btn);
     gallery_view = findViewById(R.id.gallery_view);
     roomname.setText(room_name);
     addbownerdesc.setText(desc);
     addbownerroom_no.setText(roomno);
     Log.d("key","Here is
Fragment@@@
```

```
    }
    if (requestCode == CGALLERY) {
      if (data != null) {
        Uri contentURI = data.getData();
        try {

          Bitmap bitmap = MediaStore.Images.Media.getBitmap(this.getContentResolver(),
contentURI);
          ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();
          bitmap.compress(Bitmap.CompressFormat.JPEG, 5, byteArrayOutputStream);
          gallery_view.setVisibility(View.VISIBLE);
          gallery_view.setImageBitmap(bitmap);
          eventencodedImage = Base64.encodeToString(byteArrayOutputStream.toByteArray(),
Base64.DEFAULT);
          imgname = String.valueOf(Calendar.getInstance().getTimeInMillis());
        } catch (IOException e) {
          e.printStackTrace();
          Toast.makeText(this, "Failed!", Toast.LENGTH_SHORT).show();
        }
      }
    } else if (requestCode == CAMERA) {
      try {
        Uri selectedImage = data.getData();
        bitmap = (Bitmap) data.getExtras().get("data");
        ByteArrayOutputStream bytes = new ByteArrayOutputStream();
        bitmap.compress(Bitmap.CompressFormat.JPEG, 5, bytes);
        String timeStamp = new SimpleDateFormat("yyyyMMdd_HHmmss",
Locale.getDefault()).format(new Date());
        destination = new File(Environment.getExternalStorageDirectory() + "/" +
            getString(R.string.app_name), "IMG_" + timeStamp + ".jpg");
        FileOutputStream fo;
        try {
          destination.createNewFile();
          fo = new FileOutputStream(destination);
          fo.write(bytes.toByteArray());
          fo.close();
        } catch (FileNotFoundException e) {
          e.printStackTrace();
        } catch (IOException e) {
          e.printStackTrace();
        }
        eventencodedImage = Base64.encodeToString(bytes.toByteArray(), Base64.DEFAULT);
        imgPath = destination.getAbsolutePath();
        gallery_view.setVisibility(View.VISIBLE);
        gallery_view.setImageBitmap(bitmap);
      } catch (Exception e) {
        e.printStackTrace();
      }
    } else {

    }
  }
  private void selectImage() {
    try {
      PackageManager pm = this.getPackageManager();
      int hasPerm = pm.checkPermission(Manifest.permission.CAMERA, this.getPackageName());
```

```
        if (hasPerm == PackageManager.PERMISSION_GRANTED) {
           final CharSequence[] options = {"Take Photo", "Choose From Gallery", "Cancel"};
           AlertDialog.Builder builder = new AlertDialog.Builder(this);
           builder.setTitle("Select Option");
           builder.setItems(options, new DialogInterface.OnClickListener() {
              @Override
              public void onClick(DialogInterface dialog, int item) {
                 if (options[item].equals("Take Photo")) {
                    dialog.dismiss();
                    Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
                    startActivityForResult(intent, CAMERA);
                 } else if (options[item].equals("Choose From Gallery")) {
                    dialog.dismiss();
                    Intent pickPhoto = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
                    startActivityForResult(pickPhoto, CGALLERY);
                 } else if (options[item].equals("Cancel")) {
                    dialog.dismiss();
                 }
              }
           });
           builder.show();
        } else
           Toast.makeText(this, "Camera Permission error", Toast.LENGTH_SHORT).show();
     } catch (Exception e) {
        Toast.makeText(this, "Camera Permission error", Toast.LENGTH_SHORT).show();
        e.printStackTrace();
     }
  }
  private void addToserver() {
     custPrograssbar.progressCreate(this);
     custPrograssbar.setCancel(false);
     JSONObject jsonObject = new JSONObject();
     String h=roomname.getText().toString();
     Log.d("key",h);

     try {
        jsonObject.put("id",getIntent().getStringExtra("id"));
        jsonObject.put("roomname",roomname.getText().toString() );
        jsonObject.put("desc", addbownerdesc.getText().toString());
        jsonObject.put("roomno", addbownerroom_no.getText().toString());
        if (eventencodedImage != "") {
           jsonObject.put("image", eventencodedImage);
           jsonObject.put("imagename", imgname);
           Log.d("key","it is  not empty!!!!!!!!!!!!!!!!!!!");
        } else {
           Log.d("key","it is   empty!!!!!!!!!!!!!!!!!!!");
           Bitmap bitmap = ((BitmapDrawable) gallery_view.getDrawable()).getBitmap();
           ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();
           bitmap.compress(Bitmap.CompressFormat.JPEG, 5, byteArrayOutputStream);
           String imageInGalleryView =
Base64.encodeToString(byteArrayOutputStream.toByteArray(), Base64.DEFAULT);
           jsonObject.put("image", imageInGalleryView);
           String galleryViewImgName = "gallery_view_img_" +
Calendar.getInstance().getTimeInMillis();
           jsonObject.put("imagename", galleryViewImgName);
```

```
        }

        JsonParser jsonParser = new JsonParser();
        Call<JsonObject> call = ApiClientCustom.getInterface().Editrooms((JsonObject)
jsonParser.parse(jsonObject.toString()));
        call.enqueue(new Callback<JsonObject>() {
           @Override
           public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
              custPrograssbar.close();
              JsonObject object = response.body();
              Gson gson = new Gson();
              ResponseCommon example = gson.fromJson(object.toString(), ResponseCommon.class);
              Toast.makeText(HotelEditActivity.this, example.getResponseMsg(),
Toast.LENGTH_SHORT).show();

           }
           @Override
           public void onFailure(Call<JsonObject> call, Throwable t) {
           }
        });
     } catch (JSONException e) {
        e.printStackTrace();
     }
  }
}
```

//LoginActivity.java

```
package com.dzniox.travel_mate.ui.activities;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefKeys;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import com.dzniox.travel_mate.utils.CustPrograssbar;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class LoginActivity extends AppCompatActivity {

  private Button Login, SignUp;
  private EditText Email, Password;
  APIInterface apiInterface;
```

```
PrefUtils prefrences;
CustPrograssbar custPrograssbar;

@Override
protected void onCreate(Bundle savedInstanceState) {
   super.onCreate(savedInstanceState);
   setContentView(R.layout.activity_login);
   apiInterface = APIClient.getClient().create(APIInterface.class);
   prefrences = PrefUtils.getInstance(this);
   custPrograssbar = new CustPrograssbar();
   Login = findViewById(R.id.login);
   SignUp = findViewById(R.id.signup);
   Email = findViewById(R.id.email);
   Password = findViewById(R.id.password);



   if (prefrences.getBooleanValue(PrefKeys.IS_LOGGED_IN, false)) {
      GoToMainActivity();
   } else {
   }

   SignUp.setOnClickListener(view -> {
      Intent intent = new Intent(this, UsersSignUp.class);
      startActivity(intent);
      finish();
   });

   Login.setOnClickListener(view -> {
      if (Email.getText().toString().trim().equals("") ||
            Password.getText().toString().trim().equals("")) {
         Toast.makeText(this, "Check Credentials", Toast.LENGTH_LONG).show();
      } else {
         DoLogin();
      }
   });
}
@Override
public void onBackPressed() {
   super.onBackPressed();
   finishAffinity();
}
private void DoLogin() {
   custPrograssbar.progressCreate(this);
   custPrograssbar.setCancel(false);
   try {
      Call<String> call = apiInterface.loginUser(
            Email.getText().toString().trim(),
            Password.getText().toString().trim()
      );
      call.enqueue(new Callback<String>() {
         @Override
         public void onResponse(Call<String> call, Response<String> response) {
            if (response.code() == 200) {
               try {
```

```
                    JSONObject data = new JSONObject(response.body());
                    Log.d("key","THIs IS RESPONSE");

                    if (data.getJSONArray("data").length() > 0) {
                        prefrences.setValue(PrefKeys.IS_LOGGED_IN, true);
                        prefrences.setValue("name", data.getJSONArray("data").getJSONObject(0)
                                .getString("name"));
                        prefrences.setValue("phone", data.getJSONArray("data").getJSONObject(0)
                                .getString("phone"));
                        prefrences.setValue("email", data.getJSONArray("data").getJSONObject(0)
                                .getString("email"));
                        prefrences.setValue("usertype", data.getJSONArray("data").getJSONObject(0)
                                .getString("usertype"));
                        prefrences.setValue("id", data.getJSONArray("data").getJSONObject(0)
                                .getString("id"));

                        GoToMainActivity();
                    }
                } catch (Exception e) {

                    custPrograssbar.close();
                    Toast.makeText(getBaseContext(), "Check Credentials",
Toast.LENGTH_SHORT).show();
                }
            } else {
                custPrograssbar.close();
                Toast.makeText(LoginActivity.this, "no", Toast.LENGTH_SHORT).show();
            }
        }

        @Override
        public void onFailure(Call<String> call, Throwable t) {
            String a = t.toString();
            custPrograssbar.close();
        }
      });
    }
    catch (Exception e) {
        Toast.makeText(this, e.toString(), Toast.LENGTH_SHORT).show();
    }
  }

  private void GoToMainActivity() {
    Log.d("key","THIs IS
RESPONSE8888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888
88888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888888");
    custPrograssbar.close();
    Intent intent = new Intent(this, MainActivity.class);
    intent.putExtra("admin",Email.getText().toString().trim());
    startActivity(intent);
    finish();
  }
}
```

//SignupActivity.java

```
package com.dzniox.travel_mate.ui.activities;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class SignupActivity extends AppCompatActivity {
  private Button SignUp;
  private EditText Name,Email,Password,Phone;
  private TextView Login;
  APIInterface apiInterface;

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_signup);
    apiInterface = APIClient.getClient().create(APIInterface.class);
    SignUp = findViewById(R.id.signup);
    Name = findViewById(R.id.name);
    Email = findViewById(R.id.email);
    Phone = findViewById(R.id.phone);
    Password = findViewById(R.id.password);
    Login = findViewById(R.id.login);

    Login.setOnClickListener(view -> {
      Intent intent = new Intent(this, LoginActivity.class);
      startActivity(intent);
      finish();
    });

    SignUp.setOnClickListener(view -> {
      DoSignUp(Name.getText().toString().trim(),
          Email.getText().toString().trim(),
          Password.getText().toString().trim(),
          Phone.getText().toString().trim()
          );
    });
  }

  private void DoSignUp(String name, String email, String password, String phone) {
    if (name.equals("") || email.equals("") || password.equals("") || phone.equals("")){
```

```
            Toast.makeText(this,"Check Fields",Toast.LENGTH_LONG).show();
        }else {
            Call<String> call = apiInterface.signUpUser(email,password,name,phone);
            call.enqueue(new Callback<String>() {
                @Override
                public void onResponse(Call<String> call, Response<String> response) {
                    if (response.code() == 200){
                        GoToLoginActivity();
                    }
                }
                @Override
                public void onFailure(Call<String> call, Throwable t) {
                }
            });
        }
    }

    private void GoToLoginActivity() {
        Toast.makeText(this,"SignUp Completed",Toast.LENGTH_LONG).show();
        Intent intent = new Intent(this, LoginActivity.class);
        startActivity(intent);
        finish();
    }
}
```

//TripassignActivity.java

```
package com.dzniox.travel_mate.ui.activities;

import android.Manifest;
import android.app.Activity;
import android.app.AlertDialog;
import android.app.DatePickerDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.net.Uri;
import android.os.Bundle;
import android.os.Environment;
import android.provider.MediaStore;
import android.util.Base64;
import android.util.Log;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import com.dzniox.travel_mate.Model.AllBusbyadmin;
import com.dzniox.travel_mate.Model.AllHotelsbyAdmin;
import com.dzniox.travel_mate.Model.AllroomsbyOwner;
import com.dzniox.travel_mate.Model.Example;
import com.dzniox.travel_mate.Model.GetbusSeats;
```

```
import com.dzniox.travel_mate.Model.ResponseCommon;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.NetworkUtilsCustom.GetResult;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.databinding.ActivityTripassignBinding;
import com.dzniox.travel_mate.utils.CustPrograssbar;
import com.google.gson.Gson;
import com.google.gson.JsonObject;
import com.google.gson.JsonParser;
import org.json.JSONException;
import org.json.JSONObject;
import java.io.ByteArrayOutputStream;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.Date;
import java.util.List;
import java.util.Locale;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class TripassignActivity extends AppCompatActivity implements View.OnClickListener,
GetResult.MyListener {

  ActivityTripassignBinding binding;
  List<AllroomsbyOwner> RoomList;
  ArrayList<AllroomsbyOwner> ROOMLIST = new ArrayList<>();
  List<AllBusbyadmin> ShopList;
  ArrayList<AllBusbyadmin> shopLIST = new ArrayList<>();
  List<AllHotelsbyAdmin> ShopList1;
  ArrayList<AllHotelsbyAdmin> shopLIST1 = new ArrayList<>();
  String hotel_id, bus_id,room_id;
  final Calendar myCalendar = Calendar.getInstance();
  CustPrograssbar custPrograssbar;
  List<GetbusSeats> pkg;
  String seats = "";
  Button ImageBtn;
  ImageView gallery_view;
  private File destination = null;

  private Bitmap bitmap;
  private String imgPath = null;
  private final int CGALLERY = 1;
  private final int CAMERA = 2;
  String eventencodedImage = "", imgname = "";
  EditText a,b;
  @Override

  protected void onCreate(Bundle savedInstanceState) {
     super.onCreate(savedInstanceState);
     a=findViewById(R.id.from);
     binding = ActivityTripassignBinding.inflate(getLayoutInflater());
```

```
        View view = binding.getRoot();
        setContentView(view);
        custPrograssbar = new CustPrograssbar();
        ImageBtn=findViewById(R.id.img);
        gallery_view =findViewById(R.id.gallery_view);
        binding.rooms.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
            @Override
            public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {
                AllroomsbyOwner b1 = (AllroomsbyOwner) parent.getSelectedItem();
                room_id = b1.getRoom_id();
            }
            @Override
            public void onNothingSelected(AdapterView<?> parent) {

            }
        });

        binding.hj.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
            @Override
            public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {
                AllBusbyadmin b1 = (AllBusbyadmin) parent.getSelectedItem();
                bus_id = b1.getBus_id();
                Log.d("key","##################################################");
                JSONObject jsonObject = new JSONObject();
                String packageid = getIntent().getStringExtra("packageid");
                try {
                    jsonObject.put("busid",bus_id);
                    JsonParser jsonParser = new JsonParser();
                    Call<JsonObject> call = ApiClientCustom.getInterface().getbusseats((JsonObject)
jsonParser.parse(jsonObject.toString()));

                    call.enqueue(new Callback<JsonObject>() {
                        @Override
                        public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
                            JsonObject object = response.body();
                            Gson gson = new Gson();
                            Example example = gson.fromJson(object.toString(), Example.class);
                            pkg = new ArrayList<>();
                            if (example.getResultData().getbusSeats() != null){
                                List<GetbusSeats> pakg = example.getResultData().getbusSeats();
                                Log.d("key","HEAREIS THE DATA_________________!!!!!!!!!!!!!!_____________");
                                for (GetbusSeats pkg : pakg) {
                                    seats = pkg.getSeats();

Log.d("key","data@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@"+seats
);
                                    binding.avlsts.setText(seats);
                                }
                            }
                        }
                        @Override
                        public void onFailure(Call<JsonObject> call, Throwable t) {

                        }
                    });
                } catch (JSONException e) {
```

```
            e.printStackTrace();
          }

          Log.d("key","##################################################");
        }

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

        }
      });
      binding.hotel.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
        @Override
        public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {
          AllHotelsbyAdmin s1 = (AllHotelsbyAdmin) parent.getSelectedItem();
          hotel_id = s1.getHotel_id();
          JSONObject jsonObject = new JSONObject();
          try {
            jsonObject.put("hid", hotel_id);
            JsonParser jsonParser = new JsonParser();
            Call<JsonObject> call = ApiClientCustom.getInterface().getAllroomsbyid((JsonObject)
jsonParser.parse(jsonObject.toString()));
            call.enqueue(new Callback<JsonObject>() {
              @Override
              public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
                JsonObject object = response.body();
                Gson gson = new Gson();
                Example example = gson.fromJson(object.toString(), Example.class);
                RoomList = new ArrayList<>();
                if(example.getResultData().getAllroomsbyowner() == null){

Log.d("key","@@@
```

```
                }
            });
        } catch (JSONException e) {
            e.printStackTrace();
        }

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

    }
  });
  ImageBtn.setOnClickListener(new View.OnClickListener() {
   @Override
    public void onClick(View view) {
      Log.d("key","btn clicked");
      selectImage();

    }
  });
  binding.startdate.setOnClickListener(this);
  binding.enddate.setOnClickListener(this);
  binding.submitnew.setOnClickListener(this);
  getAllBus();
  getAllSpots();
 }
 private void getAllSpots() {
  JSONObject jsonObject = new JSONObject();
  try {
    jsonObject.put("ClientID", "0");
    JsonParser jsonParser = new JsonParser();
    Call<JsonObject> call = ApiClientCustom.getInterface().getAllhotelssid((JsonObject)
jsonParser.parse(jsonObject.toString()));
    call.enqueue(new Callback<JsonObject>() {
      @Override
      public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
        JsonObject object = response.body();
        Gson gson = new Gson();
        Example example = gson.fromJson(object.toString(), Example.class);
        ShopList1 = new ArrayList<>();
        if (example.getResultData().getAllhotelsbyadmin() != null) {

          ShopList1 = example.getResultData().getAllhotelsbyadmin();
          int i = 1;
          for (AllHotelsbyAdmin sn : ShopList1) {
            shopLIST1.add(new AllHotelsbyAdmin(String.valueOf(sn.getHotel_id()),
sn.getHotel_Name()));
            i = i + 1;
          }
          ArrayAdapter<AllHotelsbyAdmin> adapter = new
ArrayAdapter<AllHotelsbyAdmin>(TripassignActivity.this, R.layout.spinner_def, shopLIST1);
          binding.hotel.setAdapter(adapter);
          Log.d("key","$$$
```

```
            }
          }
          @Override
          public void onFailure(Call<JsonObject> call, Throwable t) {
          }
        });
    } catch (JSONException e) {
      e.printStackTrace();
    }
  }
  private void getAllBus() {
    JSONObject jsonObject = new JSONObject();
    try {
      jsonObject.put("ClientID", "0");
      JsonParser jsonParser = new JsonParser();
      Call<JsonObject> call = ApiClientCustom.getInterface().getAllBusbyid((JsonObject)
jsonParser.parse(jsonObject.toString()));
      call.enqueue(new Callback<JsonObject>() {
        @Override
        public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
          JsonObject object = response.body();
          Gson gson = new Gson();
          Example example = gson.fromJson(object.toString(), Example.class);
          ShopList = new ArrayList<>();
          if (example.getResultData().getAllBusbyadmins() != null) {

            ShopList = example.getResultData().getAllBusbyadmins();
            int i = 1;
            for (AllBusbyadmin sn : ShopList) {
              shopLIST.add(new AllBusbyadmin(String.valueOf(sn.getBus_id()),
sn.getBus_Name() + " | " + sn.getSeats(), sn.getSeats()));
              i = i + 1;

            }
            ArrayAdapter<AllBusbyadmin> adapter = new
ArrayAdapter<AllBusbyadmin>(TripassignActivity.this, R.layout.spinner_def, shopLIST);
            binding.hj.setAdapter(adapter);
            binding.hj.setTitle("Select Item");
            binding.hj.setPositiveButton("OK");
          }
        }

        @Override
        public void onFailure(Call<JsonObject> call, Throwable t) {
        }
      });
    } catch (JSONException e) {
      e.printStackTrace();
    }
  }
  @Override
  public void onClick(View v) {
    switch (v.getId()) {

      case R.id.startdate:
```

```
            showDatepopupFromstart();
            break;
        case R.id.enddate:
            showDatepopupFromend();
            Log.d("key","###############!!!
```

```
    dialog.getDatePicker().setMinDate(System.currentTimeMillis());
    dialog.show();
  }
  private void updateLabelstart() {
    String myFormat = "dd-MM-yyyy";
    SimpleDateFormat sdf = new SimpleDateFormat(myFormat, Locale.US);
    binding.startdate.setText(sdf.format(myCalendar.getTime()));
  }

  private void updateLabelend() {

Log.d("key","##########################&&&
```

```
            String timeStamp = new SimpleDateFormat("yyyyMMdd_HHmmss",
Locale.getDefault()).format(new Date());
            destination = new File(Environment.getExternalStorageDirectory() + "/" +
                  getString(R.string.app_name), "IMG_" + timeStamp + ".jpg");
            FileOutputStream fo;
            try {
               destination.createNewFile();
               fo = new FileOutputStream(destination);
               fo.write(bytes.toByteArray());
               fo.close();
            } catch (FileNotFoundException e) {
               e.printStackTrace();
            } catch (IOException e) {
               e.printStackTrace();
            }
            eventencodedImage = Base64.encodeToString(bytes.toByteArray(), Base64.DEFAULT);
            imgPath = destination.getAbsolutePath();
            gallery_view.setVisibility(View.VISIBLE);
            gallery_view.setImageBitmap(bitmap);
         } catch (Exception e) {
            e.printStackTrace();
         }
      } else {

      }
   }

   private void selectImage() {

      try {
         PackageManager pm = this.getPackageManager();
         int hasPerm = pm.checkPermission(Manifest.permission.CAMERA, this.getPackageName());
         if (hasPerm == PackageManager.PERMISSION_GRANTED) {
            final CharSequence[] options = {"Take Photo", "Choose From Gallery", "Cancel"};
            AlertDialog.Builder builder = new AlertDialog.Builder(this);
            builder.setTitle("Select Option");
            builder.setItems(options, new DialogInterface.OnClickListener() {
               @Override
               public void onClick(DialogInterface dialog, int item) {
                  if (options[item].equals("Take Photo")) {
                     dialog.dismiss();
                     Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
                     startActivityForResult(intent, CAMERA);
                  } else if (options[item].equals("Choose From Gallery")) {
                     dialog.dismiss();
                     Intent pickPhoto = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
                     startActivityForResult(pickPhoto, CGALLERY);
                  } else if (options[item].equals("Cancel")) {
                     dialog.dismiss();
                  }
               }
            });
            builder.show();
         } else
            Toast.makeText(this, "Camera Permission error", Toast.LENGTH_SHORT).show();
```

```
    } catch (Exception e) {
        Toast.makeText(this, "Camera Permission error", Toast.LENGTH_SHORT).show();
        e.printStackTrace();
    }
  }

  private void addToserver() {
    custPrograssbar.progressCreate(this);
    custPrograssbar.setCancel(false);
    JSONObject jsonObject = new JSONObject();
    try {
        jsonObject.put("hotelid", hotel_id);
        jsonObject.put("bus_id", bus_id);
        jsonObject.put("room_id",room_id);
        jsonObject.put("image",eventencodedImage);
        jsonObject.put("imagename",imgname);
        jsonObject.put("bus", binding.hj.getSelectedItem());
        jsonObject.put("hotelname", binding.hotel.getSelectedItem());
        jsonObject.put("roomname", binding.rooms.getSelectedItem());
        jsonObject.put("sday", binding.startdate.getText().toString());
        jsonObject.put("eday", binding.enddate.getText().toString());
        jsonObject.put("amount", binding.allotBpoint.getText().toString().trim());
        jsonObject.put("from", binding.from.getText().toString().trim());
        jsonObject.put("to", binding.to.getText().toString().trim());
        jsonObject.put("desc", binding.pdesc.getText().toString().trim());
        jsonObject.put("avls", binding.avlsts.getText().toString().trim());
        Log.d("key",binding.pdesc.getText().toString().trim());
        JsonParser jsonParser = new JsonParser();
        Call<JsonObject> call = ApiClientCustom.getInterface().addTripassign((JsonObject)
jsonParser.parse(jsonObject.toString()));
        call.enqueue(new Callback<JsonObject>() {
            @Override
            public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
                custPrograssbar.close();
                if (response.isSuccessful() && response.body() != null) {
                    JsonObject object = response.body();
                    Gson gson = new Gson();
                    ResponseCommon example = gson.fromJson(object.toString(),
ResponseCommon.class);
                    Toast.makeText(TripassignActivity.this, example.getResponseMsg(),
Toast.LENGTH_SHORT).show();
                } else {
                    // Handle the case where the response is not successful or the body is null.
                    Toast.makeText(TripassignActivity.this, "Error in the response",
Toast.LENGTH_SHORT).show();
                }
            }

            @Override
            public void onFailure(Call<JsonObject> call, Throwable t) {

            }
        });
    } catch (JSONException e) {
        e.printStackTrace();
    }
```

```
  }
}
```

//AdminViewBookingByOneAdapter.java

```
package com.dzniox.travel_mate.ui.adapters;
import android.content.Context;
import android.content.SharedPreferences;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import com.dzniox.travel_mate.ui.fragments.AdminViewBookingsFragment;
import com.dzniox.travel_mate.ui.activities.AdminViewBookingsActivity;
import com.dzniox.travel_mate.ui.activities.MainActivity3;
import androidx.annotation.NonNull;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentTransaction;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class AdminViewBookingByOneAdapter extends
RecyclerView.Adapter<AdminViewBookingByOneAdapter.ViewHolder> {

  LayoutInflater inflater;
```

```
  private Context context;
  private JSONArray data;
  private LayoutInflater layoutInflater;
  APIInterface apiInterface;
  PrefUtils prefrences;
  RecyclerView recyclerView;
  AdminViewBookingsFragment editFragment = new AdminViewBookingsFragment();

  public AdminViewBookingByOneAdapter(Context context, JSONArray data) {

Log.d("key","adCalled3**********************************************************************");

    this.context = context;
    this.data = data;
    this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
  }

  @NonNull
  @Override
  public ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {

Log.d("key","adCalled4**********************************************************************");

    View view = layoutInflater.inflate(R.layout.admin_view_bookings_one, viewGroup, false);
    apiInterface = APIClient.getClient().create(APIInterface.class);
    prefrences = PrefUtils.getInstance(context);
    return new ViewHolder(view);
  }
  @Override
  public void onBindViewHolder(ViewHolder holder, int position) {
    try {


Log.d("key","adCalled1**********************************************************************");

      JSONObject currentObject = data.getJSONObject(position);

      holder.name.setText(currentObject.getString("name"));

      holder.email.setText(currentObject.getString("email"));
      holder.phone.setText("From "+currentObject.getString("phone"));
      String aid=currentObject.getString("id");
      String rid=currentObject.getString("id");

    }catch (Exception e){}
  }
  @Override
  public int getItemCount() {
    return data.length();
  }

  public static class ViewHolder extends RecyclerView.ViewHolder {
    private TextView name,phone,email;
    private  ImageView pkgpro;
    private Button Edit,Delete;
```

```
        private EditText ttlppl;
        public ViewHolder(View itemView) {
            super(itemView);

Log.d("key","adCalled2**********************************************************************");
            name = itemView.findViewById(R.id.name);
            email = itemView.findViewById(R.id.email);
            phone = itemView.findViewById(R.id.phone);
            ttlppl = itemView.findViewById(R.id.ttlp);
        }
    }
}
```

//AdminViewBookingsAdapter.java

```
package com.dzniox.travel_mate.ui.adapters;

import android.content.Context;
import android.content.SharedPreferences;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;
import com.dzniox.travel_mate.ui.fragments.AdminViewBookingsFragment;
import com.dzniox.travel_mate.ui.activities.AdminViewBookingsActivity;
import com.dzniox.travel_mate.ui.activities.MainActivity3;
import androidx.annotation.NonNull;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentTransaction;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
```

```
import com.dzniox.travel_mate.network.APIInterface;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class AdminViewBookingsAdapter extends
RecyclerView.Adapter<AdminViewBookingsAdapter.ViewHolder> {

  LayoutInflater inflater;
  private Context context;
  private JSONArray data;
  private LayoutInflater layoutInflater;
  APIInterface apiInterface;
  PrefUtils prefrences;
  RecyclerView recyclerView;
  AdminViewBookingsFragment editFragment = new AdminViewBookingsFragment();

  public AdminViewBookingsAdapter(Context context, JSONArray data) {

Log.d("key","adCalled3**********************************************************************");
    this.context = context;
    this.data = data;
    this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
  }
  @NonNull
  @Override
  public ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {

Log.d("key","adCalled4**********************************************************************");
    View view = layoutInflater.inflate(R.layout.admin_view_package_bookings, viewGroup, false);
    apiInterface = APIClient.getClient().create(APIInterface.class);
    prefrences = PrefUtils.getInstance(context);
    return new ViewHolder(view);
  }
  @Override
  public void onBindViewHolder(ViewHolder holder, int position) {
    try {


Log.d("key","adCalled1**********************************************************************");
      JSONObject currentObject = data.getJSONObject(position);
      holder.place.setText(currentObject.getString("toplace"));
      Log.d("key",prefrences.getStringValue("email", ""));
      holder.fromto.setText("From: "+currentObject.getString("fromplace")+" To:
"+currentObject.getString("toplace"));
      holder.sdateenddate.setText("From "+currentObject.getString("startdate")+" to
"+currentObject.getString("enddate"));
      holder.bushotel.setText("Hotel: "+currentObject.getString("hotelname"));
      holder.decrip.setText(currentObject.getString("desc"));
      holder.Bushotel.setText("Bus "+currentObject.getString("busname"));
      holder.amount.setText("Amount(per person):"+currentObject.getString("amount")+"Rs");
      holder.totseats.setText("Available Seats:"+currentObject.getString("seats"));
      holder.roomtype.setText("Room Type: "+currentObject.getString("roomname"));
      Log.d("key",prefrences.getStringValue("email", ""));
      Glide.with(context)
```

```
                .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("img"))
                .into(holder.pkgpro);

        String aid=currentObject.getString("id");
        String rid=currentObject.getString("id");
Log.d("key","**************@@@@@@@@@@@@@@@@*******************************
********");
        if (prefrences.getStringValue("email", "").equalsIgnoreCase("admin@gmail.com") &&
prefrences.getStringValue("usertype","").equalsIgnoreCase("1")) {
            Log.d("key","Admin");
        } else if (prefrences.getStringValue("usertype","").equalsIgnoreCase("0")) {
            Log.d("key","User");
            holder.Delete.setVisibility(View.GONE);
            holder.ttlppl.setVisibility(View.VISIBLE);
            holder.Edit.setText("Book");
        }
        holder.Edit.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Log.d("key",aid);
           Log.d("key","dfjghjkjdfhgkjfghjdf");
                Intent intent = new Intent(context, AdminViewBookingsActivity.class);
                Log.d("key","hellooadmin!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!");
                intent.putExtra("packageid",aid); // Replace with the actual user ID
                Log.d("key","hellooadmin!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!");
                context.startActivity(intent);
            }
        });

    }catch (Exception e){}
  }
  @Override
  public int getItemCount() {
    return data.length();
  }

  public static class ViewHolder extends RecyclerView.ViewHolder {

    private TextView place,fromto,to,sdateenddate,bushotel,roomtype,decrip,amount,totseats,Bushotel;
    private  ImageView pkgpro,buspro,hotelpro;
    private Button Edit,Delete;
    private EditText ttlppl;
    public ViewHolder(View itemView) {
      super(itemView);

Log.d("key","adCalled2*****************************************************************");
        place = itemView.findViewById(R.id.spotname);
        fromto = itemView.findViewById(R.id.frmto);
        sdateenddate = itemView.findViewById(R.id.sdateedate);
        bushotel = itemView.findViewById(R.id.bushotel);
        decrip = itemView.findViewById(R.id.desc);
        amount = itemView.findViewById(R.id.amt);
        totseats = itemView.findViewById(R.id.totst);
        Bushotel = itemView.findViewById(R.id.HotelBus);
        pkgpro = itemView.findViewById(R.id.picture);
        Edit = itemView.findViewById(R.id.edit);
```

```
        Delete = itemView.findViewById(R.id.delete);
        ttlppl = itemView.findViewById(R.id.ttlp);
        roomtype = itemView.findViewById(R.id.roomtype);
      }
    }
}
```

//AdminViewBusesAdapter.java

```
package com.dzniox.travel_mate.ui.adapters;
import android.content.Context;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentTransaction;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import com.dzniox.travel_mate.ui.fragments.AdminViewBusesFragment;
import com.dzniox.travel_mate.ui.fragments.AdminViewRoomsFragment;
import org.json.JSONArray;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class AdminViewBusesAdapter extends
RecyclerView.Adapter<AdminViewBusesAdapter.ViewHolder>  {
  LayoutInflater inflater;
  private Context context;
  private JSONArray data;
  private LayoutInflater layoutInflater;
  APIInterface apiInterface;
  PrefUtils prefrences;
  AdminViewBusesFragment adminbus = new AdminViewBusesFragment();
  public AdminViewBusesAdapter(Context context, JSONArray data) {

Log.d("key","adCalled3**********************************************************************");

    this.context = context;
    this.data = data;
    this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
  }

  @NonNull
  @Override
```

```
  public AdminViewBusesAdapter.ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup,
int viewType) {

Log.d("key","adCalled4**********************************************************************");
    View view = layoutInflater.inflate(R.layout.adminviewbuses, viewGroup, false);
    apiInterface = APIClient.getClient().create(APIInterface.class);
    prefrences = PrefUtils.getInstance(context);
    return new AdminViewBusesAdapter.ViewHolder(view);
  }
  @Override
  public void onBindViewHolder(AdminViewBusesAdapter.ViewHolder holder, int position) {
    try {

Log.d("key","adCalled1**********************************************************************");
      JSONObject currentObject = data.getJSONObject(position);
      holder.Busname.setText(currentObject.getString("name"));
      holder.descr.setText(currentObject.getString("desc"));
      holder.availableseat.setText(currentObject.getString("seat"));
      holder.imageView.setVisibility(View.VISIBLE);
      Glide.with(context)
          .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("image"))
          .into(holder.imageView);
      Log.d("key",currentObject.getString("id"));
      final String aid=currentObject.getString("id");

    }catch (Exception e){}
  }
  @Override
  public int getItemCount() {
    return data.length();
  }
  public static class ViewHolder extends RecyclerView.ViewHolder {
    private TextView Busname,descr,availableseat,delete;
    ImageView imageView;
    public ViewHolder(View itemView) {
      super(itemView);

Log.d("key","adCalled2**********************************************************************");
      Busname = itemView.findViewById(R.id.Busname);
      descr = itemView.findViewById(R.id.descr);
      availableseat = itemView.findViewById(R.id.availableseat);
      imageView = itemView.findViewById(R.id.picture);
    }
  }
}
```

//AdminViewRoomsAdapter.java

```
package com.dzniox.travel_mate.ui.adapters;
import android.content.Context;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;
import androidx.annotation.NonNull;
```

```
import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentTransaction;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import com.dzniox.travel_mate.ui.fragments.AdminViewBusesFragment;
import com.dzniox.travel_mate.ui.fragments.AdminViewRoomsFragment;
import org.json.JSONArray;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class AdminViewRoomsAdapter extends
RecyclerView.Adapter<AdminViewRoomsAdapter.ViewHolder>  {
  LayoutInflater inflater;
  private Context context;
  private JSONArray data;
  private LayoutInflater layoutInflater;
  APIInterface apiInterface;
  PrefUtils prefrences;
  AdminViewBusesFragment adminbus = new AdminViewBusesFragment();
  public AdminViewRoomsAdapter(Context context, JSONArray data) {

Log.d("key","adCalled3**********************************************************************");
    this.context = context;
    this.data = data;
    this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
  }
  @NonNull
  @Override
  public AdminViewRoomsAdapter.ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup,
int viewType) {

Log.d("key","adCalled4**********************************************************************");
    View view = layoutInflater.inflate(R.layout.adminviewbuses, viewGroup, false);
    apiInterface = APIClient.getClient().create(APIInterface.class);
    prefrences = PrefUtils.getInstance(context);
    return new AdminViewRoomsAdapter.ViewHolder(view);
  }
  @Override
  public void onBindViewHolder(AdminViewRoomsAdapter.ViewHolder holder, int position) {
    try {

Log.d("key","adCalled1**********************************************************************");
      JSONObject currentObject = data.getJSONObject(position);
      holder.Busname.setText(currentObject.getString("roomname"));
      holder.descr.setText(currentObject.getString("desc"));
      holder.availableseat.setText(currentObject.getString("roomno"));
      holder.imageView.setVisibility(View.VISIBLE);
      Glide.with(context)
```

```
                .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("image"))
                .into(holder.imageView);
            Log.d("key",currentObject.getString("id"));
            final String aid=currentObject.getString("id");
        }catch (Exception e){}
    }
    @Override
    public int getItemCount() {
        return data.length();
    }
    public static class ViewHolder extends RecyclerView.ViewHolder {
        private TextView Busname,descr,availableseat,delete;
        ImageView imageView;
        public ViewHolder(View itemView) {
            super(itemView);

Log.d("key","adCalled2**********************************************************************");
            Busname = itemView.findViewById(R.id.Busname);
            descr = itemView.findViewById(R.id.descr);
            availableseat = itemView.findViewById(R.id.availableseat);
            imageView = itemView.findViewById(R.id.picture);
        }
    }
}
```

//AdmiViewPackageAdapter.java

```
package com.dzniox.travel_mate.ui.adapters;
import android.content.Context;
import android.content.SharedPreferences;
import android.os.Handler;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;
import android.content.Intent;
import com.dzniox.travel_mate.ui.activities.Booking_Activity;
import com.dzniox.travel_mate.ui.activities.EditPackageActivity;
import com.dzniox.travel_mate.ui.fragments.AdminPackageFragment;
import com.dzniox.travel_mate.ui.activities.MainActivity3;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
```

```
import retrofit2.Response;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.ui.activities.TripassignActivity;
import com.dzniox.travel_mate.ui.fragments.AdminViewPackageFragment;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Locale;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class AdmiViewPackageAdapter extends
RecyclerView.Adapter<AdmiViewPackageAdapter.ViewHolder> {
  private void showToastAndDelete() {
  }
  public boolean flag=true;
  String rid;
  LayoutInflater inflater;
  private Context context;
  private JSONArray data;
  private LayoutInflater layoutInflater;
  APIInterface apiInterface;
  PrefUtils prefrences;
  RecyclerView recyclerView;
  private AdminViewPackageFragment adminViewPackageFragment;
  public AdmiViewPackageAdapter(Context context, JSONArray data) {

Log.d("key","adCalled3**********************************************************************");
    this.context = context;
    this.data = data;
    this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
  }
  @NonNull
  @Override
  public ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {

Log.d("key","adCalled4**********************************************************************");
    View view = layoutInflater.inflate(R.layout.admin_view_package, viewGroup, false);
    apiInterface = APIClient.getClient().create(APIInterface.class);
    prefrences = PrefUtils.getInstance(context);
    return new ViewHolder(view);
  }
  @Override
  public void onBindViewHolder(ViewHolder holder, int position) {
    try {

Log.d("key","adCalled1**********************************************************************");
```

```
        JSONObject currentObject = data.getJSONObject(position);
        String endDateString = currentObject.getString("enddate");
        SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd", Locale.getDefault());
        Date endDate = sdf.parse(endDateString);
        adminViewPackageFragment=new AdminViewPackageFragment();
        Date currentDate = new Date();
        if (endDate != null &&  (endDate.before(currentDate) || endDate.equals(currentDate))) {
           flag=false;
           Log.d("key","called called called ");
           holder.expired.setVisibility(View.VISIBLE);
           holder.Edit.setVisibility(View.GONE);
           holder.ttlppl.setVisibility(View.GONE);
           holder.Delete.setVisibility(View.GONE);
        } else {
           holder.expired.setVisibility(View.GONE);
           holder.Edit.setVisibility(View.VISIBLE);
           holder.Delete.setVisibility(View.VISIBLE);
        }
        holder.place.setText(currentObject.getString("toplace"));
        Log.d("key",prefrences.getStringValue("email", ""));
        holder.fromto.setText("Travel: "+currentObject.getString("fromplace")+" to
"+currentObject.getString("toplace"));
        holder.sdateenddate.setText("From: "+currentObject.getString("startdate")+" To:
"+currentObject.getString("enddate"));
        holder.bushotel.setText("Hotel: "+currentObject.getString("hotelname"));
        holder.decrip.setText(currentObject.getString("desc"));
        holder.Bushotel.setText("Bus: "+currentObject.getString("busname"));
        holder.amount.setText("Amount(Per person): "+currentObject.getString("amount")+"Rs");
        holder.totseats.setText("Available Seats: "+currentObject.getString("seats"));
        holder.roomtype.setText("Room Type: "+currentObject.getString("roomname"));
        Log.d("key",prefrences.getStringValue("email", ""));
        Glide.with(context)
              .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("img"))
              .into(holder.pkgpro);
        Glide.with(context)
              .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("roomimage"))
              .into(holder.roompro);
        Glide.with(context)
              .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("busimage"))
              .into(holder.buspro);
        AdminViewPackageFragment fragment = new AdminViewPackageFragment();
        String aid=currentObject.getString("id");
         rid=currentObject.getString("id");

         if (prefrences.getStringValue("email", "").equalsIgnoreCase("admin@gmail.com") &&
prefrences.getStringValue("usertype","").equalsIgnoreCase("1")) {
              Log.d("key","Admin");
        } else if (prefrences.getStringValue("usertype","").equalsIgnoreCase("0")) {
           Log.d("key","User");
           holder.Delete.setVisibility(View.GONE);
           holder.ttlppl.setVisibility(View.VISIBLE);
           holder.Edit.setText("Book");
        }
        holder.Edit.setOnClickListener(new View.OnClickListener() {
              @Override
              public void onClick(View view) {
```

```
                if (prefrences.getStringValue("email", "").equalsIgnoreCase("admin@gmail.com") &&
prefrences.getStringValue("usertype","").equalsIgnoreCase("1")) {
                    Log.d("key","hellooadmin!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!");
                    Intent intent = new Intent(context, EditPackageActivity.class);
                    Log.d("key","hellooadmin!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!");
                    /                       Log.d("key","hellooadmin!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!");
                    context.startActivity(intent);
                } else if (prefrences.getStringValue("usertype","").equalsIgnoreCase("0")) {
                    Log.d("key", "helloouser!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!");
                    String amount;
                    String edate;
                    try {
                        amount = currentObject.getString("amount");
                        edate = currentObject.getString("enddate");
                        Log.d("key",amount+"          "+edate);
                    } catch (JSONException e) {
                        throw new RuntimeException(e);
                    }
                    Intent intent = new Intent(context, Booking_Activity.class);
                    Log.d("key", "hellooadmin!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!");
                    intent.putExtra("packageid", aid); // Replace with the actual user ID
                    intent.putExtra("uid", prefrences.getStringValue("id", ""));
                    String numberOfPeople = holder.ttlppl.getText().toString();
                    intent.putExtra("amount",amount);
                    intent.putExtra("numppl",numberOfPeople);
                    Log.d("key", "hellooadmin!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!");
                    if((numberOfPeople.isEmpty())){
                        Toast.makeText(context, "Please enter number of peoples",
Toast.LENGTH_SHORT).show();
                    }else {
                        context.startActivity(intent);
                    }
                }
            }
        });
        holder.Delete.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Call<String> call = apiInterface.getpkgdltId(rid);
                call.enqueue(new Callback<String>() {
                    @Override
                    public void onResponse(Call<String> call, Response<String> response) {
                        if (response.code() == 200) {
                            Toast.makeText(view.getContext(), "Package Deleted Successfully",
Toast.LENGTH_SHORT).show();
                        }
                    }
                    @Override
                    public void onFailure(Call<String> call, Throwable t) {
                    }
                });
            }
        });
    }catch (Exception e){}
  }
  @Override
```

```
  public int getItemCount() {
     return data.length();
  }
  public static class ViewHolder extends RecyclerView.ViewHolder {
     private TextView
place,fromto,to,sdateenddate,bushotel,expired,decrip,amount,totseats,Bushotel,roomtype;
     private  ImageView pkgpro, roompro,buspro;
     private Button Edit,Delete;
     private EditText ttlppl;
     public ViewHolder(View itemView) {
        super(itemView);
Log.d("key","adCalled2**********************************************************************");
        place = itemView.findViewById(R.id.spotname);
        fromto = itemView.findViewById(R.id.frmto);
        sdateenddate = itemView.findViewById(R.id.sdateedate);
        bushotel = itemView.findViewById(R.id.bushotel);
        decrip = itemView.findViewById(R.id.desc);
        amount = itemView.findViewById(R.id.amt);
        totseats = itemView.findViewById(R.id.totst);
        Bushotel = itemView.findViewById(R.id.HotelBus);
        roompro = itemView.findViewById(R.id.roompro);
        buspro = itemView.findViewById(R.id.buspro);
        pkgpro = itemView.findViewById(R.id.picture);
        Edit = itemView.findViewById(R.id.edit);
        Delete = itemView.findViewById(R.id.delete);
        ttlppl = itemView.findViewById(R.id.ttlp);
        roomtype = itemView.findViewById(R.id.roomtype);
        expired = itemView.findViewById(R.id.expired);
     }
  }
}
```

//BookingAdapter.java

```
package com.dzniox.travel_mate.ui.adapters;
import android.annotation.SuppressLint;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
```

```
import retrofit2.Response;
public class BookingAdapter extends RecyclerView.Adapter<BookingAdapter.ViewHolder> {
  LayoutInflater inflater;
  private Context context;
  private JSONArray data;
  private LayoutInflater layoutInflater;
  APIInterface apiInterface;
  PrefUtils prefrences;
  public BookingAdapter(Context context, JSONArray data) {
    this.context = context;
    this.data = data;
    this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
  }
  @NonNull
  @Override
  public ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {
    View view = layoutInflater.inflate(R.layout.hotel_item, viewGroup, false);
    apiInterface = APIClient.getClient().create(APIInterface.class);
    prefrences = PrefUtils.getInstance(context);
    return new ViewHolder(view);
  }
  @Override
  public void onBindViewHolder(ViewHolder holder, @SuppressLint("RecyclerView") int position) {
    try {
      JSONObject currentObject = data.getJSONObject(position);
      holder.submit.setText("Cancel");
      holder.peoples.setEnabled(false);
      holder.peoples.setText(currentObject.getString("day") + " Peoples");
      holder.date.setText(currentObject.getString("from_date"));
      holder.submit.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
          try {
            CancelBooking(position, currentObject.getString("id"));
          } catch (JSONException e) {
            e.printStackTrace();
          }
        }
      });
    } catch (Exception e) {
    }
  }
  private void CancelBooking(int position, String id) {
    Call<String> call = apiInterface.deleteBooking(id);
    call.enqueue(new Callback<String>() {
      @Override
      public void onResponse(Call<String> call, Response<String> response) {
        try {
          JSONObject datal = new JSONObject(response.body());
          data.remove(position);
          notifyItemRemoved(position);
        } catch (JSONException e) {
        }
      }
      @Override
```

```
            public void onFailure(Call<String> call, Throwable t) {
            }
        });
    }
    @Override
    public int getItemCount() {
        return data.length();
    }
    public static class ViewHolder extends RecyclerView.ViewHolder {
        private TextView busname, fromto, date, totseat,desc;
        private ImageView picture;
        private Button submit;
        private EditText peoples;
        public ViewHolder(View itemView) {
            super(itemView);
            date = itemView.findViewById(R.id.sdateedate);
            busname = itemView.findViewById(R.id.Busname);
            fromto = itemView.findViewById(R.id.frmto);
            totseat = itemView.findViewById(R.id.totst);
            submit = itemView.findViewById(R.id.submit);
            desc = itemView.findViewById(R.id.desc);
            peoples = itemView.findViewById(R.id.peoples);
        }
    }
}
```

//Bus_History_Adapter.java

```
package com.dzniox.travel_mate.ui.adapters;
        import android.content.Context;
        import android.content.SharedPreferences;
        import android.util.Log;
        import android.view.LayoutInflater;
        import android.view.View;
        import android.view.ViewGroup;
        import android.widget.Button;
        import android.widget.EditText;
        import android.widget.ImageView;
        import android.widget.TextView;
        import android.widget.Toast;
        import androidx.annotation.NonNull;
        import androidx.recyclerview.widget.RecyclerView;
        import com.bumptech.glide.Glide;
        import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
        import com.dzniox.travel_mate.R;
        import com.dzniox.travel_mate.network.APIClient;
        import com.dzniox.travel_mate.network.APIInterface;
        import com.dzniox.travel_mate.sharedPref.PrefUtils;
        import org.json.JSONArray;
        import org.json.JSONException;
        import org.json.JSONObject;
        import retrofit2.Call;
        import retrofit2.Callback;
        import retrofit2.Response;
        import androidx.appcompat.app.AppCompatActivity;
        import android.content.Intent;
        import android.os.Bundle;
```

```
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class Bus_History_Adapter extends RecyclerView.Adapter<Bus_History_Adapter.ViewHolder> {
  LayoutInflater inflater;
  private Context context;
  private JSONArray data;
  private LayoutInflater layoutInflater;
  APIInterface apiInterface;
  PrefUtils prefrences;
  public Bus_History_Adapter(Context context, JSONArray data) {

Log.d("key","adCalled3**********************************************************************");
    this.context = context;
    this.data = data;
    this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
  }
  @NonNull
  @Override
  public ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {

Log.d("key","adCalled4**********************************************************************");
    View view = layoutInflater.inflate(R.layout.view_bus_request, viewGroup, false);
    apiInterface = APIClient.getClient().create(APIInterface.class);
    prefrences = PrefUtils.getInstance(context);
    return new ViewHolder(view);
  }
  @Override
  public void onBindViewHolder(ViewHolder holder, int position) {
    try {

Log.d("key","adCalled1**********************************************************************");
      JSONObject currentObject = data.getJSONObject(position);
      holder.approve.setVisibility(View.GONE);
      holder.reject.setVisibility(View.GONE);
      holder.Delete.setVisibility(View.VISIBLE);
      holder.bownername.setText(currentObject.getString("name"));
      holder.bowneremail.setText(currentObject.getString("email"));
      holder.bownerphone.setText(currentObject.getString("phone"));
      Log.d("key",currentObject.getString("busprofile"));
      Glide.with(context)
          .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("busprofile"))
          .into(holder.profile);
      String aid=currentObject.getString("id");
      String rid=currentObject.getString("id");
      holder.Delete.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
```

```
                data.remove(position);
                notifyItemRemoved(position);
                Call<String> call = apiInterface.hid(rid);
                call.enqueue(new Callback<String>() {
                    @Override
                    public void onResponse(Call<String> call, Response<String> response) {
                        if (response.code() == 200){
                            Toast.makeText(context, "Bus Owner Request Rejected",
Toast.LENGTH_SHORT).show();
                            data.remove(position);
                            notifyItemRemoved(position);
                        }
                    }
                    @Override
                    public void onFailure(Call<String> call, Throwable t) {
                    }
                });
            }
        });
    }catch (Exception e){}
  }
  @Override
  public int getItemCount() {
    return data.length();
  }
  public static class ViewHolder extends RecyclerView.ViewHolder {

    private TextView bowneremail,bownerphone,bownername,approve,reject,Delete;
    private ImageView profile;
    public ViewHolder(View itemView) {
      super(itemView);

Log.d("key","adCalled2**********************************************************************");
      bownername = itemView.findViewById(R.id.Bownername);
      bowneremail = itemView.findViewById(R.id.Bowneremail);
      bownerphone = itemView.findViewById(R.id.Bownerphone);
      reject = itemView.findViewById(R.id.reject);
      approve = itemView.findViewById(R.id.approve);
      Delete = itemView.findViewById(R.id.Delete);
      profile = itemView.findViewById(R.id.profile);
    }
  }
}
```

//BusAdapter.java

```
package com.dzniox.travel_mate.ui.adapters;
import android.app.DatePickerDialog;
import android.content.Context;
import android.content.Intent;
import android.text.Editable;
import android.text.TextWatcher;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.EditText;
```

```
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.Model.ResponseCommon;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.NetworkUtilsCustom.GetResult;
import com.dzniox.travel_mate.PGIntegration.RazorPayActivity;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.network.MessageClient;
import com.dzniox.travel_mate.network.MessageInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import com.dzniox.travel_mate.ui.activities.MainActivity;
import com.dzniox.travel_mate.ui.fragments.HomeFragment;
import com.dzniox.travel_mate.utils.CustPrograssbar;
import com.google.gson.Gson;
import com.google.gson.JsonObject;
import com.google.gson.JsonParser;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.net.URL;
import java.net.URLConnection;
import java.net.URLEncoder;
import java.util.Calendar;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class BusAdapter extends RecyclerView.Adapter<BusAdapter.ViewHolder> {
   LayoutInflater inflater;
   private Context context;
   private JSONArray data;
   private LayoutInflater layoutInflater;
   APIInterface apiInterface;
   MessageInterface messageInterface;
   PrefUtils prefrences;
   CustPrograssbar custPrograssbar;
   public BusAdapter(Context context, JSONArray data) {
      this.context = context;
      this.data = data;
      this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
   }

   @NonNull
   @Override
   public ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {
      View view = layoutInflater.inflate(R.layout.hotel_item, viewGroup, false);
      apiInterface = APIClient.getClient().create(APIInterface.class);
      messageInterface = MessageClient.getClient().create(MessageInterface.class);
```

```
        prefrences = PrefUtils.getInstance(context);
        return new ViewHolder(view);
    }
    @Override
    public void onBindViewHolder(ViewHolder holder, int position) {
        try {
            JSONObject currentObject = data.getJSONObject(position);
            holder.name.setText(currentObject.getString("name"));
            holder.desc.setText(currentObject.getString("description"));
            holder.id.setText(currentObject.getString("id"));
            holder.tot_st.setText(currentObject.getString("tot_seats"));
            holder.link.setVisibility(View.GONE);
            holder.submit.setText("Check Availability");
            custPrograssbar = new CustPrograssbar();
            Glide.with(context)
                    .load(ApiClientCustom.baseUrl + "travelmate/uploads/" +
currentObject.getString("image") + ".jpg")
                    .into(holder.picture);
            holder.date.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    final Calendar c = Calendar.getInstance();
                    int mYear = c.get(Calendar.YEAR); // current year
                    int mMonth = c.get(Calendar.MONTH); // current month
                    int mDay = c.get(Calendar.DAY_OF_MONTH); // current day
                    DatePickerDialog datePickerDialog = new DatePickerDialog(context,
                            new DatePickerDialog.OnDateSetListener() {
                                @Override
                                public void onDateSet(DatePicker view, int year, int monthOfYear, int
dayOfMonth) {
                                    holder.date.setText(dayOfMonth + "/" + (monthOfYear + 1) + "/" + year);
                                }
                            }, mYear, mMonth, mDay);
                    datePickerDialog.show();
                }
            });
            holder.submit.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    if (holder.date.getText().toString().trim().equals("Select date") ||
                            holder.peoples.getText().toString().trim().equals("")) {
                        Toast.makeText(context, "Date and No of peoples required",
Toast.LENGTH_SHORT).show();
                    } else {

                        BookNow(holder.date.getText().toString().trim(),
holder.peoples.getText().toString().trim(), currentObject, holder);
                    }
                }
            });
        } catch (Exception e) {
        }
    }
    private void BookNow(String date, String peoples, JSONObject currentObject, ViewHolder
holder) {
        if (holder.submit.getText().toString().equalsIgnoreCase("Check Availability")) {
```

```
        custPrograssbar.progressCreate(context);
        custPrograssbar.setCancel(false);
        JSONObject jsonObject = new JSONObject();
        try {
            jsonObject.put("day", date);
            jsonObject.put("book_id", holder.id.getText().toString());
            JsonParser jsonParser = new JsonParser();
            Call<JsonObject> call = ApiClientCustom.getInterface().getTotBookAvailable((JsonObject)
jsonParser.parse(jsonObject.toString()));
            call.enqueue(new Callback<JsonObject>() {
                @Override
                public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
                    custPrograssbar.close();
                    JsonObject object = response.body();
                    Gson gson = new Gson();
                    ResponseCommon responseCommon = gson.fromJson(object.toString(),
ResponseCommon.class);
                    holder.available.setVisibility(View.VISIBLE);
                    holder.available_text.setVisibility(View.VISIBLE);
                    if (responseCommon.getResponseMsg() != null) {
                        int totseats = Integer.parseInt(holder.tot_st.getText().toString());
                        int cur = Integer.parseInt(responseCommon.getResponseMsg().toString());
                        if (cur > totseats) {
                            int rem = totseats - cur;
                            holder.available_text.setText("Available Seats : ");
                            holder.available.setText(String.valueOf(rem));
                        } else {
                            int rem = totseats - cur;
                            holder.available_text.setText("Available Seats : ");
                            holder.available.setText(String.valueOf(rem));
                            holder.submit.setText("Book Now");
                        }
                    } else {
                        holder.available_text.setText("Available");
                        holder.available.setText(String.valueOf(holder.tot_st.getText().toString()));
                        holder.submit.setText("Book Now");
                    }
                }
                @Override
                public void onFailure(Call<JsonObject> call, Throwable t) {
                }
            });
        } catch (JSONException e) {
            e.printStackTrace();
        }
    } else {
    }
  }

  private void SendMessage() {
    Call<String> call = messageInterface.sendMessage(
            "sreerag",
            "bhagyasree",
            /*     prefrences.getStringValue("phone",""),*/
            "9961424428",
            "Thankyou",
```

```
            "TESTKK",
            "3",
            "1507161717524942102"
        );
        call.enqueue(new Callback<String>() {
            @Override
            public void onResponse(Call<String> call, Response<String> response) {
                if (response.code() == 200) {
                    try {
                        JSONObject data = new JSONObject(response.body());
                        if (data.getString("status").equals("success")) {
                            Toast.makeText(context, "Confirmation Message send",
Toast.LENGTH_SHORT).show();
                        } else {
                            Toast.makeText(context, "Confirmation Message not send",
Toast.LENGTH_SHORT).show();
                        }
                    } catch (JSONException e) {
                    }
                }
            }
            @Override
            public void onFailure(Call<String> call, Throwable t) {
            }
        });
    }
    public String sendSms() {
        try {
            String apiKey = "apikey=" +
URLEncoder.encode("NmU3NjY4NDgzMDZhNGQ3YTc4NDg1MzMxNTQ3OTc2MzY=",
"UTF-8");
            String message = "&message=" + URLEncoder.encode("This is your message", "UTF-8");
            String sender = "&sender=" + URLEncoder.encode("TXTLCL", "UTF-8");
            String numbers = "&numbers=" + URLEncoder.encode("919961424428", "UTF-8");
            String data = "https://api.textlocal.in/send/?" + apiKey + numbers + message + sender;
            URL url = new URL(data);
            URLConnection conn = url.openConnection();
            conn.setDoOutput(true);
            BufferedReader rd = new BufferedReader(new InputStreamReader(conn.getInputStream()));
            String line;
            String sResult = "";
            while ((line = rd.readLine()) != null) {
                sResult = sResult + line + " ";
            }
            rd.close();
            Toast.makeText(context, "Confirmation Message send" + sResult,
Toast.LENGTH_SHORT).show();
            return sResult;
        } catch (Exception e) {
            Toast.makeText(context, "Confirmation Message Send Error" + e,
Toast.LENGTH_SHORT).show();
            return e.toString();
        }
    }
    @Override
    public int getItemCount() {
```

```
        return data.length();
    }
    public static class ViewHolder extends RecyclerView.ViewHolder {
        private TextView name, desc, link, date, available, id, tot_st, available_text;
        private ImageView picture;
        private Button submit;
        private EditText peoples;
        public ViewHolder(View itemView) {
            super(itemView);
            picture = itemView.findViewById(R.id.picture);
            name = itemView.findViewById(R.id.name);
            desc = itemView.findViewById(R.id.desc);
            link = itemView.findViewById(R.id.link);
            id = itemView.findViewById(R.id.id);
            available = itemView.findViewById(R.id.available);
            available_text = itemView.findViewById(R.id.available_text);
            tot_st = itemView.findViewById(R.id.totst);
            submit = itemView.findViewById(R.id.submit);
            date = itemView.findViewById(R.id.date);
            peoples = itemView.findViewById(R.id.peoples);
        }
    }
}
//BusAssignedAdapter.java
package com.dzniox.travel_mate.ui.adapters;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class BusAssignedAdapter extends RecyclerView.Adapter<BusAssignedAdapter.ViewHolder>
{
    LayoutInflater inflater;
    private Context context;
    private JSONArray data;
    private LayoutInflater layoutInflater;
    APIInterface apiInterface;
    PrefUtils prefrences;
    public BusAssignedAdapter(Context context, JSONArray data) {
```

```
        this.context = context;
        this.data = data;
        this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
    }
    @NonNull
    @Override
    public ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {
        View view = layoutInflater.inflate(R.layout.assigned_trips, viewGroup, false);
        apiInterface = APIClient.getClient().create(APIInterface.class);
        prefrences = PrefUtils.getInstance(context);
        return new ViewHolder(view);
    }
    @Override
    public void onBindViewHolder(ViewHolder holder, int position) {
        try {
            JSONObject currentObject = data.getJSONObject(position);
            holder.Busname.setText(currentObject.getString("busname"));
            holder.amount.setText("Amount"+currentObject.getString("amount") + " Rs");
            holder.desc.setText(currentObject.getString("desc"));
            holder.frmto.setText(currentObject.getString("fromplace") +" to
"+currentObject.getString("toplace"));
            holder.sdateedate.setText(currentObject.getString("startdate") + " to "
+currentObject.getString("enddate"));
            holder.totst.setText("Seats:"+currentObject.getString("avls"));
            Glide.with(context)
                    .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("image"))
                    .into(holder.image);
        }catch (Exception e){}
    }
    @Override
    public int getItemCount() {
        return data.length();
    }
    public static class ViewHolder extends RecyclerView.ViewHolder {
        private TextView Busname,desc,amount,frmto,sdateedate,totst,user;
        ImageView image;
        public ViewHolder(View itemView) {
            super(itemView);
            amount = itemView.findViewById(R.id.amt);
            Busname = itemView.findViewById(R.id.Busname);
            desc = itemView.findViewById(R.id.desc);
            frmto = itemView.findViewById(R.id.frmto);
            sdateedate = itemView.findViewById(R.id.sdateedate);
            totst = itemView.findViewById(R.id.totst);
            image = itemView.findViewById(R.id.picture);
        }
    }
}
//BusOwnersListAdapter.java
package com.dzniox.travel_mate.ui.adapters;
import android.content.Context;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
```

```
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentTransaction;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import com.dzniox.travel_mate.ui.fragments.AdminViewBusesFragment;
import com.dzniox.travel_mate.ui.fragments.AdminViewRoomsFragment;
import org.json.JSONArray;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class BusOwnersListAdapter extends
RecyclerView.Adapter<BusOwnersListAdapter.ViewHolder>  {
   LayoutInflater inflater;
   private Context context;
   private JSONArray data;
   private LayoutInflater layoutInflater;
   APIInterface apiInterface;
   PrefUtils prefrences;
   AdminViewBusesFragment adminbus = new AdminViewBusesFragment();
   public BusOwnersListAdapter(Context context, JSONArray data) {

Log.d("key","adCalled3**********************************************************************"
);
      this.context = context;
      this.data = data;
      this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
   }
   @NonNull
   @Override
   public BusOwnersListAdapter.ViewHolder onCreateViewHolder(@NonNull ViewGroup
viewGroup, int viewType) {

Log.d("key","adCalled4**********************************************************************"
);
      View view = layoutInflater.inflate(R.layout.view_users_list, viewGroup, false);
      apiInterface = APIClient.getClient().create(APIInterface.class);
      prefrences = PrefUtils.getInstance(context);
      return new BusOwnersListAdapter.ViewHolder(view);
   }
   @Override
   public void onBindViewHolder(BusOwnersListAdapter.ViewHolder holder, int position) {
      try {
```

```
Log.d("key","adCalled1**********************************************************************"
);
        JSONObject currentObject = data.getJSONObject(position);
        holder.bownername.setText(currentObject.getString("name"));
        holder.bowneremail.setText(currentObject.getString("email"));
        holder.bownerphone.setText(currentObject.getString("phone"));
        holder.viewbus.setVisibility(View.VISIBLE);
        holder.imageView.setVisibility(View.VISIBLE);
        Glide.with(context)
              .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("busprofile"))
              .into(holder.imageView);
        Log.d("key",currentObject.getString("id"));
        final String aid=currentObject.getString("id");
        holder.viewbus.setOnClickListener(new View.OnClickListener() {
          @Override
          public void onClick(View view) {
             FragmentManager fragmentManager = ((AppCompatActivity)
context).getSupportFragmentManager();
              FragmentTransaction fragmentTransaction = fragmentManager.beginTransaction();
              AdminViewBusesFragment adminbus = new AdminViewBusesFragment(aid);
              fragmentTransaction.replace(R.id.container, adminbus);
              fragmentTransaction.addToBackStack(null);
              fragmentTransaction.commit();
          }
        });
    }catch (Exception e){}
  }
  @Override
  public int getItemCount() {
    return data.length();
  }
  public static class ViewHolder extends RecyclerView.ViewHolder {
    private TextView bowneremail,bownerphone,bownername,delete;
    ImageView imageView;
    Button viewrooms,viewbus;
    public ViewHolder(View itemView) {
      super(itemView);

Log.d("key","adCalled2**********************************************************************"
);
        bownername = itemView.findViewById(R.id.ownername);
        bowneremail = itemView.findViewById(R.id.owneremail);
        bownerphone = itemView.findViewById(R.id.ownerphone);
        imageView = itemView.findViewById(R.id.imgview);
        viewrooms = itemView.findViewById(R.id.room);
        viewbus = itemView.findViewById(R.id.bus);
    }
  }
}
//BusRequestAdapter.java
package com.dzniox.travel_mate.ui.adapters;
import android.content.Context;
import android.content.SharedPreferences;
import android.util.Log;
import android.view.LayoutInflater;
```

```
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class BusRequestAdapter extends RecyclerView.Adapter<BusRequestAdapter.ViewHolder> {
   LayoutInflater inflater;
   private Context context;
   private JSONArray data;
   private LayoutInflater layoutInflater;
   APIInterface apiInterface;
   PrefUtils prefrences;
   public BusRequestAdapter(Context context, JSONArray data) {

Log.d("key","adCalled3**********************************************************************"
);
      this.context = context;
      this.data = data;
      this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
   }
   @NonNull
   @Override
   public ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {

Log.d("key","adCalled4**********************************************************************"
);
```

```
        View view = layoutInflater.inflate(R.layout.view_bus_request, viewGroup, false);
        apiInterface = APIClient.getClient().create(APIInterface.class);
        prefrences = PrefUtils.getInstance(context);
        return new ViewHolder(view);
    }
    @Override
    public void onBindViewHolder(ViewHolder holder, int position) {
        try {

Log.d("key","adCalled1**********************************************************************"
);
            JSONObject currentObject = data.getJSONObject(position);
            holder.bownername.setText(currentObject.getString("name"));
            holder.bowneremail.setText(currentObject.getString("email"));
            holder.bownerphone.setText(currentObject.getString("phone"));
            Log.d("key",currentObject.getString("busprofile"));
            Glide.with(context)
                    .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("busprofile"))
                    .into(holder.profile);
            String aid=currentObject.getString("id");
            String rid=currentObject.getString("id");
            holder.approve.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    Call<String> call = apiInterface.getUserId(aid);
                    call.enqueue(new Callback<String>() {
                        @Override
                        public void onResponse(Call<String> call, Response<String> response) {
                            if (response.code() == 200){
                                Toast.makeText(context, "bus Owner Request Accepted",
Toast.LENGTH_SHORT).show();
                                data.remove(position);
                                notifyItemRemoved(position);
                            }
                        }
                        @Override
                        public void onFailure(Call<String> call, Throwable t) {
                        }
                    });
                }
            });
            holder.reject.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    data.remove(position);
                    notifyItemRemoved(position);
                    Call<String> call = apiInterface.getRid(rid);
                    call.enqueue(new Callback<String>() {
                        @Override
                        public void onResponse(Call<String> call, Response<String> response) {
                            if (response.code() == 200){
                                Toast.makeText(context, "Bus Owner Request Rejected",
Toast.LENGTH_SHORT).show();
                                data.remove(position);
                                notifyItemRemoved(position);
                            }
```

```
                }
                @Override
                public void onFailure(Call<String> call, Throwable t) {
                }
            });
        }
      });
    }catch (Exception e){}
  }
  @Override
  public int getItemCount() {
    return data.length();
  }
  public static class ViewHolder extends RecyclerView.ViewHolder {
    private TextView bowneremail,bownerphone,bownername,approve,reject;
    private  ImageView profile;
    public ViewHolder(View itemView) {
      super(itemView);

Log.d("key","adCalled2**********************************************************************"
);
      bownername = itemView.findViewById(R.id.Bownername);
      bowneremail = itemView.findViewById(R.id.Bowneremail);
      bownerphone = itemView.findViewById(R.id.Bownerphone);
      reject = itemView.findViewById(R.id.reject);
      approve = itemView.findViewById(R.id.approve);
      profile = itemView.findViewById(R.id.profile);
    }
  }
}
//ExpenseAdapter.java
package com.dzniox.travel_mate.ui.adapters;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class ExpenseAdapter extends RecyclerView.Adapter<ExpenseAdapter.ViewHolder> {
  LayoutInflater inflater;
```

```
    private Context context;
    private JSONArray data;
    private LayoutInflater layoutInflater;
    APIInterface apiInterface;
    PrefUtils prefrences;
    public ExpenseAdapter(Context context, JSONArray data) {
        this.context = context;
        this.data = data;
        this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
    }
    @NonNull
    @Override
    public ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {
        View view = layoutInflater.inflate(R.layout.expense_item, viewGroup, false);
        apiInterface = APIClient.getClient().create(APIInterface.class);
        prefrences = PrefUtils.getInstance(context);
        return new ViewHolder(view);
    }
    @Override
    public void onBindViewHolder(ViewHolder holder, int position) {
        try {
            JSONObject currentObject = data.getJSONObject(position);
            holder.Busname.setText("Bus:"+currentObject.getString("busname"));
            holder.user.setText("User:"+currentObject.getString("uname"));
            holder.amount.setText("Amount:"+currentObject.getString("amount") + " Rs");
            holder.desc.setText(currentObject.getString("desc"));
            holder.frmto.setText(currentObject.getString("fromplace") +" to
"+currentObject.getString("toplace"));
            holder.sdateedate.setText(currentObject.getString("startdate") + " to "
+currentObject.getString("enddate"));
            holder.totst.setText("Seats:"+currentObject.getString("seats"));
            Glide.with(context)
                    .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("image"))
                    .into(holder.img);
        }catch (Exception e){}
    }
    @Override
    public int getItemCount() {
        return data.length();
    }
    public static class ViewHolder extends RecyclerView.ViewHolder {
        private TextView Busname,desc,amount,frmto,sdateedate,totst,user;
        ImageView img;
        public ViewHolder(View itemView) {
            super(itemView);
            amount = itemView.findViewById(R.id.amt);
            Busname = itemView.findViewById(R.id.Busname);
            desc = itemView.findViewById(R.id.desc);
            frmto = itemView.findViewById(R.id.frmto);
            sdateedate = itemView.findViewById(R.id.sdateedate);
            totst = itemView.findViewById(R.id.totst);
            user = itemView.findViewById(R.id.user);
            img = itemView.findViewById(R.id.image);
        }
    }
```

```
}
//Hotel_History_Adapter .java
package com.dzniox.travel_mate.ui.adapters;
      import android.content.Context;
      import android.content.SharedPreferences;
      import android.util.Log;
      import android.view.LayoutInflater;
      import android.view.View;
      import android.view.ViewGroup;
      import android.widget.Button;
      import android.widget.EditText;
      import android.widget.ImageView;
      import android.widget.TextView;
      import android.widget.Toast;
      import androidx.annotation.NonNull;
      import androidx.recyclerview.widget.RecyclerView;
      import com.bumptech.glide.Glide;
      import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
      import com.dzniox.travel_mate.R;
      import com.dzniox.travel_mate.network.APIClient;
      import com.dzniox.travel_mate.network.APIInterface;
      import com.dzniox.travel_mate.sharedPref.PrefUtils;
      import org.json.JSONArray;
      import org.json.JSONException;
      import org.json.JSONObject;
      import retrofit2.Call;
      import retrofit2.Callback;
      import retrofit2.Response;
      import androidx.appcompat.app.AppCompatActivity;
      import android.content.Intent;
      import android.os.Bundle;
      import android.widget.Button;
      import android.widget.EditText;
      import android.widget.TextView;
      import android.widget.Toast;
      import com.dzniox.travel_mate.R;
      import com.dzniox.travel_mate.network.APIClient;
      import com.dzniox.travel_mate.network.APIInterface;
      import retrofit2.Call;
      import retrofit2.Callback;
      import retrofit2.Response;
public class Hotel_History_Adapter extends
RecyclerView.Adapter<Hotel_History_Adapter.ViewHolder> {
   LayoutInflater inflater;
   private Context context;
   private JSONArray data;
   private LayoutInflater layoutInflater;
   APIInterface apiInterface;
   PrefUtils prefrences;
   public Hotel_History_Adapter(Context context, JSONArray data) {

Log.d("key","adCalled3**********************************************************************"
);

      this.context = context;
      this.data = data;
```

```
      this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
   }
   @NonNull
   @Override
   public ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {

Log.d("key","adCalled4**********************************************************************"
);
      View view = layoutInflater.inflate(R.layout.view_bus_request, viewGroup, false);
      apiInterface = APIClient.getClient().create(APIInterface.class);
      prefrences = PrefUtils.getInstance(context);
      return new ViewHolder(view);
   }
   @Override
   public void onBindViewHolder(ViewHolder holder, int position) {
      try {

Log.d("key","adCalled1**********************************************************************"
);
         JSONObject currentObject = data.getJSONObject(position);
         holder.approve.setVisibility(View.GONE);
         holder.reject.setVisibility(View.GONE);
         holder.Delete.setVisibility(View.VISIBLE);
         holder.bownername.setText(currentObject.getString("name"));
         holder.bowneremail.setText(currentObject.getString("email"));
         holder.bownerphone.setText(currentObject.getString("phone"));
         Log.d("key",currentObject.getString("busprofile"));
         Glide.with(context)
               .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("busprofile"))
               .into(holder.profile);
         String aid=currentObject.getString("id");
         String rid=currentObject.getString("id");
         holder.Delete.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
               data.remove(position);
               notifyItemRemoved(position);
               Call<String> call = apiInterface.hid(rid);
               call.enqueue(new Callback<String>() {
                  @Override
                  public void onResponse(Call<String> call, Response<String> response) {
                     if (response.code() == 200){
                        Toast.makeText(context, "Bus Owner Request Rejected",
Toast.LENGTH_SHORT).show();
                        data.remove(position);
                        notifyItemRemoved(position);
                     }
                  }
                  @Override
                  public void onFailure(Call<String> call, Throwable t) {
                  }
               });
            }
         });
      }catch (Exception e){}
```

```
    }
    @Override
    public int getItemCount() {
        return data.length();
    }
    public static class ViewHolder extends RecyclerView.ViewHolder {
        private TextView bowneremail,bownerphone,bownername,approve,reject,Delete;
        private ImageView profile;
        public ViewHolder(View itemView) {
            super(itemView);

Log.d("key","adCalled2**********************************************************************"
);
            bownername = itemView.findViewById(R.id.Bownername);
            bowneremail = itemView.findViewById(R.id.Bowneremail);
            bownerphone = itemView.findViewById(R.id.Bownerphone);
            reject = itemView.findViewById(R.id.reject);
            approve = itemView.findViewById(R.id.approve);
            Delete = itemView.findViewById(R.id.Delete);
            profile = itemView.findViewById(R.id.profile);
        }
    }
}
//HotelAdapter.java
package com.dzniox.travel_mate.ui.adapters;
import android.app.DatePickerDialog;
import android.content.Context;
import android.content.Intent;
import android.net.Uri;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.LinearLayout;
import android.widget.TextView;
import android.widget.Toast;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.PGIntegration.RazorPayActivity;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.network.MessageClient;
import com.dzniox.travel_mate.network.MessageInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import com.dzniox.travel_mate.ui.activities.WebViewActivity;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Locale;
```

```
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class HotelAdapter extends RecyclerView.Adapter<HotelAdapter.ViewHolder> {
   LayoutInflater inflater;
   private Context context;
   private JSONArray data;
   private LayoutInflater layoutInflater;
   APIInterface apiInterface;
   PrefUtils prefrences;
   final Calendar myCalendar = Calendar.getInstance();
   MessageInterface messageInterface;

   public HotelAdapter(Context context, JSONArray data) {
      this.context = context;
      this.data = data;
      this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
   }
   @NonNull
   @Override
   public ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {
      View view = layoutInflater.inflate(R.layout.hotel_item, viewGroup, false);
      apiInterface = APIClient.getClient().create(APIInterface.class);
      prefrences = PrefUtils.getInstance(context);
      messageInterface = MessageClient.getClient().create(MessageInterface.class);



      return new ViewHolder(view);
   }
   @Override
   public void onBindViewHolder(ViewHolder holder, int position) {
      try {
         JSONObject currentObject = data.getJSONObject(position);
         holder.name.setText(currentObject.getString("name"));
         holder.desc.setText(currentObject.getString("description"));
         Glide.with(context)
               .load(currentObject.getString("image"))
               .into(holder.picture);

         holder.link.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
               String url = null;
               try {
                  url = currentObject.getString("map_link");
               } catch (JSONException e) {
                  e.printStackTrace();
               }
               Intent i = new Intent(Intent.ACTION_VIEW);
               i.setData(Uri.parse(url));
               context.startActivity(i);
            }
         });
         holder.date.setOnClickListener(new View.OnClickListener() {
```

```
            @Override
            public void onClick(View view) {
               DatePickerDialog.OnDateSetListener date = new DatePickerDialog.OnDateSetListener()
{
                  @Override
                  public void onDateSet(DatePicker view, int year, int monthOfYear,
                                 int dayOfMonth) {
                     myCalendar.set(Calendar.YEAR, year);
                     myCalendar.set(Calendar.MONTH, monthOfYear);
                     myCalendar.set(Calendar.DAY_OF_MONTH, dayOfMonth);
                     updateLabel();
                  }
               };
               DatePickerDialog dialog = new DatePickerDialog(context, date, myCalendar
                     .get(Calendar.YEAR), myCalendar.get(Calendar.MONTH),
                     myCalendar.get(Calendar.DAY_OF_MONTH));
               dialog.getDatePicker().setMinDate(System.currentTimeMillis());
               dialog.show();
            }

            private void updateLabel() {
               String myFormat = "yyyy-MM-dd";
               SimpleDateFormat sdf = new SimpleDateFormat(myFormat, Locale.US);
               holder.date.setText(sdf.format(myCalendar.getTime()));
            }
         });
         holder.submit.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
               if (holder.date.getText().toString().trim().equals("Select date") ||
                     holder.peoples.getText().toString().trim().equals("")) {
                  Toast.makeText(context, "Date and No of peoples required",
Toast.LENGTH_SHORT).show();
               } else {
                  BookNow(holder.date.getText().toString(), holder.peoples.getText().toString().trim(),
currentObject, holder);
               }
            }
         });

      } catch (Exception e) {
      }
   }
   private void BookNow(String date, String peoples, JSONObject currentObject, ViewHolder
holder) {
      try {
         context.startActivity(new Intent(context, RazorPayActivity.class)
               .putExtra("day", peoples)
               .putExtra("date", date)
               .putExtra("isAction", "HOTEL")
               .putExtra("id", currentObject.getString("id")));
      } catch (JSONException e) {
         e.printStackTrace();
      }
   }
   @Override
```

```
    public int getItemCount() {
        return data.length();
    }
    public static class ViewHolder extends RecyclerView.ViewHolder {
        private TextView name, desc, link, date;
        private ImageView picture;
        private Button submit;
        private EditText peoples;
        public ViewHolder(View itemView) {
            super(itemView);
            picture = itemView.findViewById(R.id.picture);
            name = itemView.findViewById(R.id.name);
            desc = itemView.findViewById(R.id.desc);
            link = itemView.findViewById(R.id.link);
            submit = itemView.findViewById(R.id.submit);
            date = itemView.findViewById(R.id.date);
            peoples = itemView.findViewById(R.id.peoples);
        }
    }
}
//HotelAssignedAdapter.java
package com.dzniox.travel_mate.ui.adapters;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class HotelAssignedAdapter extends
RecyclerView.Adapter<HotelAssignedAdapter.ViewHolder> {
    LayoutInflater inflater;
    private Context context;
    private JSONArray data;
    private LayoutInflater layoutInflater;
    APIInterface apiInterface;
    PrefUtils prefrences;
    public HotelAssignedAdapter(Context context, JSONArray data) {
        this.context = context;
        this.data = data;
```

```
        this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
    }
    @NonNull
    @Override
    public ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {
        View view = layoutInflater.inflate(R.layout.assigned_trips, viewGroup, false);
        apiInterface = APIClient.getClient().create(APIInterface.class);
        prefrences = PrefUtils.getInstance(context);
        return new ViewHolder(view);
    }
    @Override
    public void onBindViewHolder(ViewHolder holder, int position) {
        try {
            JSONObject currentObject = data.getJSONObject(position);
            holder.Busname.setText(currentObject.getString("hotelname"));
            holder.amount.setText("Amount"+currentObject.getString("amount") + " Rs");
            holder.desc.setText(currentObject.getString("desc"));
            holder.frmto.setText(currentObject.getString("fromplace") +" to
"+currentObject.getString("toplace"));
            holder.sdateedate.setText(currentObject.getString("startdate") + " to "
+currentObject.getString("enddate"));
            holder.totst.setText("Seats:"+currentObject.getString("avls"));
            Glide.with(context)
                    .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("image"))
                    .into(holder.image);
        }catch (Exception e){}
    }
    @Override
    public int getItemCount() {
        return data.length();
    }
    public static class ViewHolder extends RecyclerView.ViewHolder {
        private TextView Busname,desc,amount,frmto,sdateedate,totst,user;
        ImageView image;
        public ViewHolder(View itemView) {
            super(itemView);
            amount = itemView.findViewById(R.id.amt);
            Busname = itemView.findViewById(R.id.Busname);
            desc = itemView.findViewById(R.id.desc);
            frmto = itemView.findViewById(R.id.frmto);
            sdateedate = itemView.findViewById(R.id.sdateedate);
            totst = itemView.findViewById(R.id.totst);
            image = itemView.findViewById(R.id.picture);
        }
    }
}
//HotelBookingsAdapter.java
package com.dzniox.travel_mate.ui.adapters;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
```

```
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class HotelBookingsAdapter extends
RecyclerView.Adapter<HotelBookingsAdapter.ViewHolder> {
   LayoutInflater inflater;
   private Context context;
   private JSONArray data;
   private LayoutInflater layoutInflater;
   APIInterface apiInterface;
   PrefUtils prefrences;
   public HotelBookingsAdapter(Context context, JSONArray data) {
      this.context = context;
      this.data = data;
      this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
   }
   @NonNull
   @Override
   public ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {
      View view = layoutInflater.inflate(R.layout.expense_item, viewGroup, false);
      apiInterface = APIClient.getClient().create(APIInterface.class);
      prefrences = PrefUtils.getInstance(context);
      return new ViewHolder(view);
   }
   @Override
   public void onBindViewHolder(ViewHolder holder, int position) {
      try {
         JSONObject currentObject = data.getJSONObject(position);
         holder.Busname.setText("Room:"+currentObject.getString("roomname"));
         holder.user.setText("User:"+currentObject.getString("uname"));
         holder.amount.setText("Amount:"+currentObject.getString("amount") + " Rs");
         holder.desc.setText(currentObject.getString("desc"));
         holder.frmto.setText(currentObject.getString("fromplace") +" to
"+currentObject.getString("toplace"));
         holder.sdateedate.setText(currentObject.getString("startdate") + " to "
+currentObject.getString("enddate"));
         holder.totst.setText("Seats:"+currentObject.getString("seats"));
         Glide.with(context)
               .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("image"))
               .into(holder.img);
      }catch (Exception e){}
   }
   @Override
```

```
    public int getItemCount() {
        return data.length();
    }
    public static class ViewHolder extends RecyclerView.ViewHolder {
        private TextView Busname,desc,amount,frmto,sdateedate,totst,user;
        ImageView img;
        public ViewHolder(View itemView) {
            super(itemView);
            amount = itemView.findViewById(R.id.amt);
            Busname = itemView.findViewById(R.id.Busname);
            desc = itemView.findViewById(R.id.desc);
            frmto = itemView.findViewById(R.id.frmto);
            sdateedate = itemView.findViewById(R.id.sdateedate);
            totst = itemView.findViewById(R.id.totst);
            user = itemView.findViewById(R.id.user);
            img = itemView.findViewById(R.id.image);
        }
    }
}
//HotelOwnersListAdapter.java
package com.dzniox.travel_mate.ui.adapters;
import android.content.Context;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentTransaction;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import com.dzniox.travel_mate.ui.fragments.AdminViewBusesFragment;
import com.dzniox.travel_mate.ui.fragments.AdminViewRoomsFragment;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class HotelOwnersListAdapter extends
RecyclerView.Adapter<HotelOwnersListAdapter.ViewHolder>  {
    LayoutInflater inflater;
    private Context context;
    private JSONArray data;
    private LayoutInflater layoutInflater;
    APIInterface apiInterface;
    PrefUtils prefrences;
```

```
   private FragmentManager fragmentManager;
   public HotelOwnersListAdapter(Context context, JSONArray data) {

Log.d("key","adCalled3**********************************************************************"
);
      this.context = context;
      this.data = data;
      this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
   }
   @NonNull
   @Override
   public HotelOwnersListAdapter.ViewHolder onCreateViewHolder(@NonNull ViewGroup
viewGroup, int viewType) {

Log.d("key","adCalled4**********************************************************************"
);
      View view = layoutInflater.inflate(R.layout.view_users_list, viewGroup, false);
      apiInterface = APIClient.getClient().create(APIInterface.class);
      prefrences = PrefUtils.getInstance(context);
      return new HotelOwnersListAdapter.ViewHolder(view);
   }
   @Override
   public void onBindViewHolder(HotelOwnersListAdapter.ViewHolder holder, int position) {
      try {

Log.d("key","adCalled1**********************************************************************"
);
         JSONObject currentObject = data.getJSONObject(position);
         holder.bownername.setText(currentObject.getString("name"));
         holder.bowneremail.setText(currentObject.getString("email"));
         holder.bownerphone.setText(currentObject.getString("phone"));
         holder.viewrooms.setVisibility(View.VISIBLE);
         holder.imageView.setVisibility(View.VISIBLE);
         Glide.with(context)
               .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("busprofile"))
               .into(holder.imageView);
         Log.d("key",currentObject.getString("id"));
         final String aid = currentObject.getString("id");
         holder.viewrooms.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
               FragmentManager fragmentManager = ((AppCompatActivity)
context).getSupportFragmentManager();
               FragmentTransaction fragmentTransaction = fragmentManager.beginTransaction();
               AdminViewRoomsFragment adminroom = new AdminViewRoomsFragment(aid);
               fragmentTransaction.replace(R.id.container, adminroom);
               fragmentTransaction.addToBackStack(null);
               fragmentTransaction.commit();
            }
         });
      }catch (Exception e){}
   }
   @Override
   public int getItemCount() {
      return data.length();
```

```
    }
    public static class ViewHolder extends RecyclerView.ViewHolder {
        private TextView bowneremail,bownerphone,bownername,delete;
        ImageView imageView;
        Button viewrooms,viewbus;
        public ViewHolder(View itemView) {
            super(itemView);

Log.d("key","adCalled2**********************************************************************"
);
            bownername = itemView.findViewById(R.id.ownername);
            bowneremail = itemView.findViewById(R.id.owneremail);
            bownerphone = itemView.findViewById(R.id.ownerphone);
            imageView = itemView.findViewById(R.id.imgview);
            viewrooms = itemView.findViewById(R.id.room);
            viewbus = itemView.findViewById(R.id.bus);
        }
    }
}
//HotelRequestAdapter .java
package com.dzniox.travel_mate.ui.adapters;
import android.content.Context;
import android.content.SharedPreferences;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import com.dzniox.travel_mate.ui.fragments.UsersViewFragment;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
import androidx.fragment.app.FragmentTransaction;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentManager;
public class HotelRequestAdapter extends
RecyclerView.Adapter<HotelRequestAdapter.ViewHolder> {
    LayoutInflater inflater;
    private Context context;
    private JSONArray data;
    private LayoutInflater layoutInflater;
```

```
   APIInterface apiInterface;
   PrefUtils prefrences;
   UsersViewFragment usersviewfragment;
   private FragmentManager fragmentManager;
   public HotelRequestAdapter(Context context, JSONArray data) {

Log.d("key","adCalled3**********************************************************************"
);
      this.context = context;
      this.data = data;
      this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
      this.fragmentManager = fragmentManager;
   }
   @NonNull
   @Override
   public ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {

Log.d("key","adCalled4**********************************************************************"
);
      View view = layoutInflater.inflate(R.layout.view_bus_request, viewGroup, false);
      apiInterface = APIClient.getClient().create(APIInterface.class);
      prefrences = PrefUtils.getInstance(context);
      return new ViewHolder(view);
   }
   @Override
   public void onBindViewHolder(ViewHolder holder, int position) {
      try {

Log.d("key","adCalled1**********************************************************************"
);
         JSONObject currentObject = data.getJSONObject(position);
         holder.bownername.setText(currentObject.getString("name"));
         holder.bowneremail.setText(currentObject.getString("email"));
         holder.bownerphone.setText(currentObject.getString("phone"));
         holder.hname.setVisibility(View.VISIBLE);
         holder.add.setVisibility(View.VISIBLE);
         holder.hname.setText(currentObject.getString("hotelname"));
         holder.add.setText(currentObject.getString("hoteladdress"));
         Glide.with(context)
               .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("busprofile"))
               .into(holder.profile);
         String aid=currentObject.getString("id");
         String rid=currentObject.getString("id");
         holder.approve.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
               Call<String> call = apiInterface.getUserId(aid);
               call.enqueue(new Callback<String>() {
                  @Override
                  public void onResponse(Call<String> call, Response<String> response) {
                     if (response.code() == 200){
                        Toast.makeText(context, "Hotel Owner Request Accepted",
Toast.LENGTH_SHORT).show();
                        data.remove(position);
                        notifyItemRemoved(position);
```

```
                }
            }
            @Override
            public void onFailure(Call<String> call, Throwable t) {
            }
        });
      }
    });
    holder.reject.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View view) {
        data.remove(position);
        notifyItemRemoved(position);
        Call<String> call = apiInterface.getRid(rid);
        call.enqueue(new Callback<String>() {
          @Override
          public void onResponse(Call<String> call, Response<String> response) {
            if (response.code() == 200){
              Toast.makeText(context, "Hotel Owner Request Rejected",
Toast.LENGTH_SHORT).show();
              data.remove(position);
              notifyItemRemoved(position);
            }
          }
          @Override
          public void onFailure(Call<String> call, Throwable t) {
          }
        });
      }
    });
  }catch (Exception e){}
 }
 @Override
 public int getItemCount() {
  return data.length();
 }
 public static class ViewHolder extends RecyclerView.ViewHolder {
  private TextView bowneremail,bownerphone,bownername,approve,hname,add,reject;
  private  ImageView profile;
  public ViewHolder(View itemView) {
   super(itemView);

Log.d("key","adCalled2**********************************************************************"
);
   bownername = itemView.findViewById(R.id.Bownername);
   hname = itemView.findViewById(R.id.hotelname);
   add = itemView.findViewById(R.id.desc);
   bowneremail = itemView.findViewById(R.id.Bowneremail);
   bownerphone = itemView.findViewById(R.id.Bownerphone);
   reject = itemView.findViewById(R.id.reject);
   approve = itemView.findViewById(R.id.approve);
   profile = itemView.findViewById(R.id.profile);
  }
 }
}
//ReviewsAdapter .java
```

```
package com.dzniox.travel_mate.ui.adapters;
import android.annotation.SuppressLint;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;
import android.widget.Toast;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class ReviewsAdapter extends RecyclerView.Adapter<ReviewsAdapter.ViewHolder> {
   LayoutInflater inflater;
   private Context context;
   private JSONArray data;
   private LayoutInflater layoutInflater;
   APIInterface apiInterface;
   PrefUtils prefrences;
   public ReviewsAdapter(Context context, JSONArray data) {
      this.context = context;
      this.data = data;
      this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
   }
   @NonNull
   @Override
   public ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {
      View view = layoutInflater.inflate(R.layout.item_review, viewGroup, false);
      apiInterface = APIClient.getClient().create(APIInterface.class);
      prefrences = PrefUtils.getInstance(context);
      return new ViewHolder(view);
   }
   @Override
   public void onBindViewHolder(ViewHolder holder, @SuppressLint("RecyclerView") int position)
{
      try {
         JSONObject currentObject = data.getJSONObject(position);
         if (prefrences.getStringValue("id","").equals(currentObject.getString("uid"))){
            holder.delete.setVisibility(View.VISIBLE);
         }else  {}
         holder.email.setText(currentObject.getString("uname"));
         holder.title.setText(currentObject.getString("review"));
         holder.delete.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
               data.remove(position);
               notifyItemRemoved(position);
```

```
                try {
                    DeleteReview(currentObject.getString("id"));
                } catch (JSONException e) {
                    e.printStackTrace();
                }
            }
        });
    }catch (Exception e){}
}
private void DeleteReview(String id) {
    try {
        Call<String> call = apiInterface.deleteReview(id);
        call.enqueue(new Callback<String>() {
            @Override
            public void onResponse(Call<String> call, Response<String> response) {
                if (response.code() == 200) {
                    try {
                        JSONObject data = new JSONObject(response.body());
                        if (data.getBoolean("data")){
                            Toast.makeText(context,"Review Deleted",Toast.LENGTH_SHORT).show();
                        }else {
                            Toast.makeText(context,"Try Again",Toast.LENGTH_SHORT).show();
                        }
                    }catch (Exception e){
                    }
                }
            }
            @Override
            public void onFailure(Call<String> call, Throwable t) {
                String a = t.toString();
            }
        });
    }catch (Exception e){
        Toast.makeText(context,e.toString(),Toast.LENGTH_SHORT).show();
    }
}
@Override
public int getItemCount() {
    return data.length();
}
public static class ViewHolder extends RecyclerView.ViewHolder {
    private TextView title,delete,email;
    public ViewHolder(View itemView) {
        super(itemView);
        title = itemView.findViewById(R.id.title);
        delete = itemView.findViewById(R.id.delete);
        email = itemView.findViewById(R.id.email);
    }
}
}


//RoomsViewByOwnerAdapter.java
package com.dzniox.travel_mate.ui.adapters;
import android.content.Context;
```

```
import android.content.Intent;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentTransaction;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import com.dzniox.travel_mate.ui.activities.EditBusActivity;
import com.dzniox.travel_mate.ui.activities.HotelEditActivity;
import com.dzniox.travel_mate.ui.activities.MainActivity2;
import com.dzniox.travel_mate.ui.activities.MainActivity3;
import com.dzniox.travel_mate.ui.fragments.AdminViewBusesFragment;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class RoomsViewByOwnerAdapter extends
RecyclerView.Adapter<RoomsViewByOwnerAdapter.ViewHolder>  {
   LayoutInflater inflater;
   private Context context;
   private JSONArray data;
   private LayoutInflater layoutInflater;
   APIInterface apiInterface;
   PrefUtils prefrences;
   AdminViewBusesFragment adminbus = new AdminViewBusesFragment();
   public RoomsViewByOwnerAdapter(Context context, JSONArray data) {

Log.d("key","adCalled3**********************************************************************"
);
      this.context = context;
      this.data = data;
      this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
   }
   @NonNull
   @Override
   public RoomsViewByOwnerAdapter.ViewHolder onCreateViewHolder(@NonNull ViewGroup
viewGroup, int viewType) {

Log.d("key","adCalled4**********************************************************************"
);
```

```
    View view = layoutInflater.inflate(R.layout.view_hotel_roomsbyowner, viewGroup, false);
    apiInterface = APIClient.getClient().create(APIInterface.class);
    prefrences = PrefUtils.getInstance(context);
    return new RoomsViewByOwnerAdapter.ViewHolder(view);
  }
  @Override
  public void onBindViewHolder(RoomsViewByOwnerAdapter.ViewHolder holder, int position) {
    try {

Log.d("key","adCalled1***********9090*****************************************************
***");
      JSONObject currentObject = data.getJSONObject(position);
      holder.descrip.setText("Description :"+currentObject.getString("room_desc"));
      holder.roomno.setText("Room No. :"+currentObject.getString("room_no"));
      Log.d("key",currentObject.getString("roomname"));
      holder.rooms.setText(currentObject.getString("roomname"));
      Glide.with(context)
          .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("roompic"))
          .into(holder.img);
      String aid=currentObject.getString("id");
       Log.d("key","This is the Button THis is THe BTTon1234");
      holder.edits.setOnClickListener(new View.OnClickListener() {
        @Override
        pubic void onClick(View view) {

          Log.d("key","fghbjfkjfkhjfdfkjhkdfghkdgfjlhkdgjhkldg");
          String desc;
          String roomname;
          String roomno;
          String id;
          String image;
          try {
            desc =currentObject.getString("room_desc");
            roomname = currentObject.getString("roomname");
            roomno =currentObject.getString("room_no");
            id = currentObject.getString("id");
            image=currentObject.getString("roompic");
          } catch (JSONException e) {
            throw new RuntimeException(e);
          }
          Log.d("key","fghbjfkjfkhjfdfkjhkdfghkdgfjlhkdgjhkldg");
          Intent intent = new Intent(context,HotelEditActivity.class);
          intent.putExtra("id", id); // Replace with the actual user ID
          intent.putExtra("roomname",roomname );
          intent.putExtra("roomno",roomno);
          intent.putExtra("desc",desc);
          intent.putExtra("image",image);
          context.startActivity(intent);
        }
      });
      holder.delete.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
          Toast.makeText(context, "delete", Toast.LENGTH_SHORT).show();

          Call<String> call = apiInterface.roomdlt(aid);
```

```
            call.enqueue(new Callback<String>() {
                @Override
                public void onResponse(Call<String> call, Response<String> response) {
                    if (response.code() == 200) {
                        data.remove(position);
                        notifyItemRemoved(position);
                    }
                }
                @Override
                public void onFailure(Call<String> call, Throwable t) {
                }
            });
        }
    });
  }
  catch (Exception e){}
 }
 @Override
 public int getItemCount() {
  return data.length();
 }
 public static class ViewHolder extends RecyclerView.ViewHolder {
  private TextView descrip,roomno,imageview,rooms;
  public ImageView img;
  private Button edits,delete;
  public ViewHolder(View itemView) {
   super(itemView);

Log.d("key","adCalled2**********************************************************************"
);
   descrip = itemView.findViewById(R.id.descrip);
   roomno = itemView.findViewById(R.id.roomno);
   rooms= itemView.findViewById(R.id.roomnames);
   img= itemView.findViewById(R.id.pictureroom);
   edits= itemView.findViewById(R.id.edit);
   delete= itemView.findViewById(R.id.delete);
  }
 }
}
//UserBookingsviewAdapter .java
package com.dzniox.travel_mate.ui.adapters;


import android.content.Context;
import android.content.SharedPreferences;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;
import com.dzniox.travel_mate.ui.activities.Booking_Activity;
import com.dzniox.travel_mate.ui.activities.EditPackageActivity;
```

```
import com.dzniox.travel_mate.ui.activities.MainActivity3;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Context;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;

import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Context;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Locale;
public class UserBookingsviewAdapter extends
RecyclerView.Adapter<UserBookingsviewAdapter.ViewHolder> {

    LayoutInflater inflater;
    private Context context;
    private JSONArray data;
    private LayoutInflater layoutInflater;
    APIInterface apiInterface;
    PrefUtils prefrences;
```

```
   RecyclerView recyclerView;
   public UserBookingsviewAdapter(Context context, JSONArray data) {

Log.d("key","adCalled3**********************************************************************"
);
      this.context = context;
      this.data = data;
      this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
   }
   @NonNull
   @Override
   public ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {
Log.d("key","adCalled4**********************************************************************"
);
      View view = layoutInflater.inflate(R.layout.user_view_bookings, viewGroup, false);
      apiInterface = APIClient.getClient().create(APIInterface.class);
      prefrences = PrefUtils.getInstance(context);
      return new ViewHolder(view);
   }
   @Override
   public void onBindViewHolder(ViewHolder holder, int position) {
      try {

Log.d("key","adCalled1**********************************************************************"
);
         JSONObject currentObject = data.getJSONObject(position);
         String endDateString = currentObject.getString("enddate");
         SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd", Locale.getDefault());
         Date endDate = sdf.parse(endDateString);
         Date currentDate = new Date();
         if (endDate != null && endDate.before(currentDate)) {

         } else {

         }
         holder.place.setText(currentObject.getString("toplace"));
         holder.fromto.setText("Travel: "+currentObject.getString("fromplace")+" to
"+currentObject.getString("toplace"));
         holder.sdateenddate.setText("From: "+currentObject.getString("startdate")+" To:
"+currentObject.getString("enddate"));
         holder.bushotel.setText("Hotel: "+currentObject.getString("hotelname") +" Room Type:
"+currentObject.getString("room"));
         holder.Bushotel.setText("Bus: "+currentObject.getString("busname"));
         holder.amount.setText("Amount: "+currentObject.getString("amount"));
         Glide.with(context)
               .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("profile"))
               .into(holder.profile);
         String aid=currentObject.getString("id");
         String rid=currentObject.getString("id");
         holder.Delete.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

Log.d("key","**************@@@@@@@1234567890@@@@@@@@**********************
*******************");
```

```
                Call<String> call = apiInterface.cancelbooking(rid);
                call.enqueue(new Callback<String>() {
                    @Override
                    public void onResponse(Call<String> call, Response<String> response) {
                        if (response.code() == 200) {
                            data.remove(position);
                            notifyItemRemoved(position);
                        }
                    }
                    @Override
                    public void onFailure(Call<String> call, Throwable t) {
                    }
                });
            }
        });
    }catch (Exception e){}
  }
  @Override
  public int getItemCount() {
    return data.length();
  }
  private void showDeleteConfirmationDialog() {
    AlertDialog.Builder alertDialogBuilder = new AlertDialog.Builder(context);
    alertDialogBuilder.setMessage("Are you sure you want to delete this item?");
    alertDialogBuilder.setPositiveButton("Yes", new DialogInterface.OnClickListener() {
      @Override
      public void onClick(DialogInterface dialog, int which) {

      }
    });
    alertDialogBuilder.setNegativeButton("No", new DialogInterface.OnClickListener() {
      @Override
      public void onClick(DialogInterface dialog, int which) {
        dialog.dismiss(); // Dismiss the dialog if the user clicks "No"
      }
    });
    AlertDialog alertDialog = alertDialogBuilder.create();
    alertDialog.show();
  }
  public static class ViewHolder extends RecyclerView.ViewHolder {
    private TextView
place,fromto,to,sdateenddate,bushotel,Tripexpired,decrip,amount,totseats,Bushotel,useramount;
    private  ImageView profile;
    private Button Edit,Delete;
    private EditText ttlppl;
    public ViewHolder(View itemView) {
      super(itemView);

Log.d("key","adCalled2**********************************************************************"
);
      place = itemView.findViewById(R.id.spotname);
      fromto = itemView.findViewById(R.id.frmto);
      sdateenddate = itemView.findViewById(R.id.sdateedate);
      bushotel = itemView.findViewById(R.id.bushotel);
      decrip = itemView.findViewById(R.id.desc);
      amount = itemView.findViewById(R.id.amt);
```

```
            totseats = itemView.findViewById(R.id.totst);
            Bushotel = itemView.findViewById(R.id.HotelBus);
            profile = itemView.findViewById(R.id.picture);
            Edit = itemView.findViewById(R.id.edit);
            Delete = itemView.findViewById(R.id.cancel);
            ttlppl = itemView.findViewById(R.id.ttlp);
            Tripexpired = itemView.findViewById(R.id.Tripexpired);
        }
    }
}

//UsersListAdapter .java
package com.dzniox.travel_mate.ui.adapters;

import android.content.Context;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;

import org.json.JSONArray;
import org.json.JSONObject;

import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class UsersListAdapter extends RecyclerView.Adapter<UsersListAdapter.ViewHolder>  {

    LayoutInflater inflater;
    private Context context;
    private JSONArray data;
    private LayoutInflater layoutInflater;
    APIInterface apiInterface;
    PrefUtils prefrences;

    public UsersListAdapter(Context context, JSONArray data) {

Log.d("key","adCalled3**********************************************************************"
);

        this.context = context;
        this.data = data;
        this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
```

```
    }

    @NonNull
    @Override
    public UsersListAdapter.ViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int
viewType) {

Log.d("key","adCalled4**********************************************************************"
);

        View view = layoutInflater.inflate(R.layout.view_users_list, viewGroup, false);
        apiInterface = APIClient.getClient().create(APIInterface.class);
        prefrences = PrefUtils.getInstance(context);

        return new UsersListAdapter.ViewHolder(view);
    }
    @Override
    public void onBindViewHolder(UsersListAdapter.ViewHolder holder, int position) {
        try {

Log.d("key","adCalled1**********************************************************************"
);
            JSONObject currentObject = data.getJSONObject(position);
            holder.bownername.setText(currentObject.getString("name"));
            holder.bowneremail.setText(currentObject.getString("email"));
            holder.bownerphone.setText(currentObject.getString("phone"));
            Log.d("key",currentObject.getString("id"));
            String aid=currentObject.getString("id");

        }catch (Exception e){}
    }

    @Override
    public int getItemCount() {
        return data.length();
    }

    public static class ViewHolder extends RecyclerView.ViewHolder {

        private TextView bowneremail,bownerphone,bownername,delete;
        ImageView imageView;
        Button viewrooms,viewbus;
        public ViewHolder(View itemView) {
            super(itemView);
Log.d("key","adCalled2**********************************************************************"
);
            bownername = itemView.findViewById(R.id.ownername);
            bowneremail = itemView.findViewById(R.id.owneremail);
            bownerphone = itemView.findViewById(R.id.ownerphone);
            imageView = itemView.findViewById(R.id.imgview);
            viewrooms = itemView.findViewById(R.id.room);
            viewbus = itemView.findViewById(R.id.bus);
        }
    }
}
```

```
//ViewBusesByOwnersAdapter .java
package com.dzniox.travel_mate.ui.adapters;

import android.content.Context;
import android.content.Intent;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentTransaction;
import androidx.recyclerview.widget.RecyclerView;

import com.bumptech.glide.Glide;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import com.dzniox.travel_mate.ui.activities.Booking_Activity;
import com.dzniox.travel_mate.ui.activities.EditBusActivity;
import com.dzniox.travel_mate.ui.activities.MainActivity2;
import com.dzniox.travel_mate.ui.fragments.AdminViewRoomsFragment;
import com.dzniox.travel_mate.ui.fragments.EditBusFragment;

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

import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class ViewBusesByOwnersAdapter extends
RecyclerView.Adapter<ViewBusesByOwnersAdapter.ViewHolder>  {

   LayoutInflater inflater;
   private Context context;
   private JSONArray data;
   private LayoutInflater layoutInflater;
   APIInterface apiInterface;
   PrefUtils prefrences;

   public ViewBusesByOwnersAdapter(Context context, JSONArray data) {

Log.d("key","adCalled3**********************************************************************"
);

      this.context = context;
      this.data = data;
```

```
      this.layoutInflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
   }

   @NonNull
   @Override
   public ViewBusesByOwnersAdapter.ViewHolder onCreateViewHolder(@NonNull ViewGroup
viewGroup, int viewType) {

Log.d("key","adCalled4**********************************************************************"
);

      View view = layoutInflater.inflate(R.layout.view_buses_busowners, viewGroup, false);
      apiInterface = APIClient.getClient().create(APIInterface.class);
      prefrences = PrefUtils.getInstance(context);

      return new ViewBusesByOwnersAdapter.ViewHolder(view);
   }
   @Override
   public void onBindViewHolder(ViewBusesByOwnersAdapter.ViewHolder holder, int position) {
      try {

Log.d("key","adCalled1**********************************************************************"
);
         JSONObject currentObject = data.getJSONObject(position);
         holder.descrip.setText(currentObject.getString("desc"));
         holder.Busname.setText(currentObject.getString("busname"));
         holder.seats.setText("Total Seats:"+currentObject.getString("seat_no"));
         Glide.with(context)
               .load(ApiClientCustom.baseUrl + "uploads/" +currentObject.getString("busprofile"))
               .into(holder.img);

         Log.d("key",currentObject.getString("desc"));
         String aid=currentObject.getString("id");

         holder.edit.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

               String desc;
               String busname;
               String seatno;
               String id;
               String image;
               try {
              desc =currentObject.getString("desc");
               busname = currentObject.getString("busname");
                 seatno =currentObject.getString("seat_no");
                  id = currentObject.getString("id");
                  image=currentObject.getString("busprofile");
               } catch (JSONException e) {
                  throw new RuntimeException(e);
               }
               Log.d("key","fghbjfkjfkhjfdfkjhkdfghkdgfjlhkdgjhkldg");
               Intent intent = new Intent(context, EditBusActivity.class);
```

```
            intent.putExtra("id", id); // Replace with the actual user ID
            intent.putExtra("busname",busname );
            intent.putExtra("seatno",seatno);
            intent.putExtra("desc",desc);
            intent.putExtra("image",image);
            context.startActivity(intent);


          }
        });

        holder.del.setOnClickListener(new View.OnClickListener() {
          @Override
          public void onClick(View view) {

            Call<String> call = apiInterface.busdlt(aid);
            call.enqueue(new Callback<String>() {
              @Override
              public void onResponse(Call<String> call, Response<String> response) {
                if (response.code() == 200) {
                  data.remove(position);
                  notifyItemRemoved(position);

                }
              }

              @Override
              public void onFailure(Call<String> call, Throwable t) {
              }
            });
          }
        });
      }catch (Exception e){}
    }

    @Override
    public int getItemCount() {
      return data.length();
    }

    public static class ViewHolder extends RecyclerView.ViewHolder {

      private TextView descrip,seats,Busname,user;
      private Button edit,del;
      ImageView img;

      public ViewHolder(View itemView) {
        super(itemView);

Log.d("key","adCalled2**********************************************************************"
);
        descrip = itemView.findViewById(R.id.descr);
        seats = itemView.findViewById(R.id.availableseat);
        Busname = itemView.findViewById(R.id.Busname);
        edit = itemView.findViewById(R.id.edit);
        img = itemView.findViewById(R.id.picture);
```

```
            del = itemView.findViewById(R.id.delete);

        }
    }
}

//AddBusFragment .java
package com.dzniox.travel_mate.ui.fragments;

import android.Manifest;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.net.Uri;
import android.os.Bundle;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentTransaction;

import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.RecyclerView;

import android.os.Environment;
import android.provider.MediaStore;
import android.util.Base64;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;
import android.app.Activity;

import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import com.dzniox.travel_mate.ui.activities.BusSignUp;

import org.json.JSONObject;

import java.io.ByteArrayOutputStream;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.Locale;
```

```
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class AddBusFragment extends Fragment {

   private static final String ARG_PARAM1 = "param1";
   private static final String ARG_PARAM2 = "param2";

   private String mParam1;
   private String mParam2;
   APIInterface apiInterface;
   private Context context;
   private View rootView;
   private RecyclerView rv_users;
   private  String id;

   PrefUtils prefrences;
   private Button addbus;
   private File destination = null;
   private ImageView gallery_view;
   private Bitmap bitmap;
   private String imgPath = null;
   private final int CGALLERY = 1;
   private final int CAMERA = 2;
   String eventencodedImage = "", imgname = "";
   private EditText ownername, ownerdesc, ownerseatno, busname, ownermaplink;
   private TextView Login;
   private Button ImageBtn;


   public AddBusFragment() {
      // Required empty public constructor
   }


   public static AddBusFragment newInstance(String param1, String param2) {
      AddBusFragment fragment = new AddBusFragment();
      Bundle args = new Bundle();
      args.putString(ARG_PARAM1, param1);
      args.putString(ARG_PARAM2, param2);
      fragment.setArguments(args);
      return fragment;
   }

   @Override
   public void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      if (getArguments() != null) {
         mParam1 = getArguments().getString(ARG_PARAM1);
         mParam2 = getArguments().getString(ARG_PARAM2);
      }
   }

   @Override
```

```
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    rootView = inflater.inflate(R.layout.fragment_add_bus, container, false);
    context = getContext();
    prefrences = PrefUtils.getInstance(context);
    apiInterface = APIClient.getClient().create(APIInterface.class);
    addbus = rootView.findViewById(R.id.addbowneradd_btn);
    busname = rootView.findViewById(R.id.addbusname);
    ownerdesc = rootView.findViewById(R.id.addbownerdesc);
    ownerseatno = rootView.findViewById(R.id.addbownerseat_no);
    Login = rootView.findViewById(R.id.login);
    ImageBtn=rootView.findViewById(R.id.addbownerbtn_choose);
    gallery_view =rootView.findViewById(R.id.gallery_view);
     id=prefrences.getStringValue("id", "");
    Log.d("key",prefrences.getStringValue("id", ""));
    onResume();
    addbus.setOnClickListener(view -> {
        DoAddBus(
                ownerdesc.getText().toString().trim(),
                ownerseatno.getText().toString().trim(),
                busname.getText().toString().trim(),
                eventencodedImage,
                imgname,
                id
        );
    });
    ImageBtn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Log.d("key","btn clicked");
            selectImage();
        }
    });
    return rootView;
}
@Override
public void onResume() {
    super.onResume();

    busname.setText("");
    ownerdesc.setText("");
    ownerseatno.setText("");
}

@Override
public void onActivityResult(int requestCode, int resultCode, Intent data) {

    super.onActivityResult(requestCode, resultCode, data);
    if (resultCode == Activity.RESULT_CANCELED) {
        return;
    }
    if (requestCode == CGALLERY) {
        if (data != null) {
            Uri contentURI = data.getData();
            try {
```

```
                Bitmap bitmap =
MediaStore.Images.Media.getBitmap(this.context.getContentResolver(), contentURI);
                ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();
                bitmap.compress(Bitmap.CompressFormat.JPEG, 5, byteArrayOutputStream);
                gallery_view.setVisibility(View.VISIBLE);
                gallery_view.setImageBitmap(bitmap);
                eventencodedImage = Base64.encodeToString(byteArrayOutputStream.toByteArray(),
Base64.DEFAULT);
                imgname = String.valueOf(Calendar.getInstance().getTimeInMillis());


            } catch (IOException e) {
                e.printStackTrace();
                Toast.makeText(this.context, "Failed!", Toast.LENGTH_SHORT).show();
            }
        }
    } else if (requestCode == CAMERA) {
        try {
            Uri selectedImage = data.getData();
            bitmap = (Bitmap) data.getExtras().get("data");
            ByteArrayOutputStream bytes = new ByteArrayOutputStream();
            bitmap.compress(Bitmap.CompressFormat.JPEG, 5, bytes);
            String timeStamp = new SimpleDateFormat("yyyyMMdd_HHmmss",
Locale.getDefault()).format(new Date());
            destination = new File(Environment.getExternalStorageDirectory() + "/" +
                    getString(R.string.app_name), "IMG_" + timeStamp + ".jpg");
            FileOutputStream fo;
            try {
                destination.createNewFile();
                fo = new FileOutputStream(destination);
                fo.write(bytes.toByteArray());
                fo.close();
            } catch (FileNotFoundException e) {
                e.printStackTrace();
            } catch (IOException e) {
                e.printStackTrace();
            }
            eventencodedImage = Base64.encodeToString(bytes.toByteArray(), Base64.DEFAULT);
            imgPath = destination.getAbsolutePath();
            gallery_view.setVisibility(View.VISIBLE);
            gallery_view.setImageBitmap(bitmap);

        } catch (Exception e) {
            e.printStackTrace();
        }

    } else {

    }
  }


   private void DoAddBus( String ownerdesc, String ownerseatno,String busname,String
eventencodedImage,String imgname,String id) {
      if ( ownerdesc.equals("") || ownerseatno.equals("") ||  eventencodedImage.equals("") ||
busname.equals("")){
```

```
        Toast.makeText(this.context,"Check Fields",Toast.LENGTH_LONG).show();
      }else {
        Call<String> call =
apiInterface.AddbusDet(ownerdesc,ownerseatno,busname,eventencodedImage,imgname,id);
        call.enqueue(new Callback<String>() {
          @Override
          public void onResponse(Call<String> call, Response<String> response) {
            if (response.code() == 200){


              ViewBusesFragment nextFragment = new ViewBusesFragment(); // Replace with the
actual fragment class you want to navigate to
              FragmentManager fragmentManager = getParentFragmentManager(); // Use
getActivity().getSupportFragmentManager() if you are in an Activity
              FragmentTransaction transaction = fragmentManager.beginTransaction();
              transaction.replace(R.id.container, nextFragment); // Replace R.id.fragment_container
with the ID of your fragment container
              transaction.addToBackStack(null); // Add to back stack if you want to enable the back
button to navigate back
              transaction.commit();
              Toast.makeText(context, "Bus added successfully", Toast.LENGTH_SHORT).show();
            } else {
              Toast.makeText(context, "Failed to add bus. Please try again.",
Toast.LENGTH_SHORT).show();
            }
          }
          @Override
          public void onFailure(Call<String> call, Throwable t) {
          }
        });
      }
    }

    private void selectImage() {

      try {
        PackageManager pm = this.context.getPackageManager();
        int hasPerm = pm.checkPermission(Manifest.permission.CAMERA,
this.context.getPackageName());
        if (hasPerm == PackageManager.PERMISSION_GRANTED) {
          final CharSequence[] options = {"Take Photo", "Choose From Gallery", "Cancel"};
          AlertDialog.Builder builder = new AlertDialog.Builder(this.context);
          builder.setTitle("Select Option");
          builder.setItems(options, new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int item) {
              if (options[item].equals("Take Photo")) {
                dialog.dismiss();
                Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
                startActivityForResult(intent, CAMERA);
              } else if (options[item].equals("Choose From Gallery")) {
                dialog.dismiss();
                Intent pickPhoto = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
                startActivityForResult(pickPhoto, CGALLERY);
              } else if (options[item].equals("Cancel")) {
```

```
                    dialog.dismiss();
                }
            }
        });
        builder.show();
    } else
        Toast.makeText(this.context, "Camera Permission error",
Toast.LENGTH_SHORT).show();
    } catch (Exception e) {
        Toast.makeText(this.context, "Camera Permission error", Toast.LENGTH_SHORT).show();
        e.printStackTrace();
    }
  }
}

//AddBusHotelFragment .java
package com.dzniox.travel_mate.ui.fragments;

import android.Manifest;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.net.Uri;
import android.os.Bundle;

import androidx.annotation.Nullable;
import androidx.fragment.app.Fragment;

import android.os.Environment;
import android.provider.MediaStore;
import android.util.Base64;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

import com.dzniox.travel_mate.Model.ResponseCommon;
import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;
import com.dzniox.travel_mate.NetworkUtilsCustom.GetResult;
import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
import com.dzniox.travel_mate.utils.CustPrograssbar;
import com.google.gson.Gson;
import com.google.gson.JsonObject;
import com.google.gson.JsonParser;
```

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

import java.io.ByteArrayOutputStream;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.Locale;
import java.util.Random;

import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;


public class AddBusHotelFragment extends Fragment implements
View.OnClickListener,GetResult.MyListener {


    private static final String ARG_PARAM1 = "param1";
    private static final String ARG_PARAM2 = "param2";

    private String mParam1;
    private String mParam2;

    private Context context;
    private View rootView;
    APIInterface apiInterface;
    PrefUtils prefrences;
    private Bitmap bitmap;
    private File destination = null;
    private InputStream inputStreamImg;
    private String imgPath = null;
    Random rand;
    private final int CGALLERY = 1;
    private final int CAMERA = 2;
    int click = 0;
    String eventencodedImage = "", imgname = "";
    private String Title;
    CustPrograssbar custPrograssbar;
    private TextView title;
    private EditText name, desc, image, location, tot_seats;
    private Spinner city;
    private Button add_btn, btn_choose;
    private ImageView gallery_view;

    public AddBusHotelFragment() {
    }


    public static AddBusHotelFragment newInstance(String param1, String param2) {
```

```
    AddBusHotelFragment fragment = new AddBusHotelFragment();
    Bundle args = new Bundle();
    args.putString(ARG_PARAM1, param1);
    args.putString(ARG_PARAM2, param2);
    fragment.setArguments(args);
    return fragment;
}

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    // Inflate the layout for this fragment
    rootView = inflater.inflate(R.layout.fragment_add_bus_hotel, container, false);
    context = getContext();
    apiInterface = APIClient.getClient().create(APIInterface.class);
    prefrences = PrefUtils.getInstance(context);
    custPrograssbar = new CustPrograssbar();
    title = rootView.findViewById(R.id.title);
    name = rootView.findViewById(R.id.name);
    desc = rootView.findViewById(R.id.desc);
    tot_seats = rootView.findViewById(R.id.seat_no);
    image = rootView.findViewById(R.id.image);
    location = rootView.findViewById(R.id.location);
    gallery_view = rootView.findViewById(R.id.gallery_view);
    add_btn = rootView.findViewById(R.id.add_btn);
    btn_choose = rootView.findViewById(R.id.btn_choose);
    city = rootView.findViewById(R.id.sp_district);
    btn_choose.setOnClickListener(this);
    Bundle bundle = getArguments();
    Title = bundle.getString("title");

    title.setText(Title);

    if (Title.equals("Add Bus")) {
        location.setVisibility(View.GONE);
        city.setVisibility(View.GONE);

    } else {
        location.setVisibility(View.VISIBLE);
        tot_seats.setHint("Total rooms");
    }

    add_btn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            if (Title.equals("Add Bus")) {
                if (name.getText().toString().trim().equals("") ||
```

```
                    name.getText().toString().trim().equals("") ||
                    desc.getText().toString().trim().equals("") ||
                    image.getText().toString().trim().equals("") ||
                    tot_seats.getText().toString().trim().equals("")

            ) {
                Toast.makeText(context, "Check all the fields", Toast.LENGTH_SHORT).show();
            } else {
                //AddBus();
                AddBusCustom();
            }

        } else {
            if (name.getText().toString().trim().equals("") ||
                    name.getText().toString().trim().equals("") ||
                    desc.getText().toString().trim().equals("") ||
                    image.getText().toString().trim().equals("") ||
                    location.getText().toString().trim().equals("") ||
                    city.getSelectedItem().toString().equals("Choose City")) {
                Toast.makeText(context, "Check all the fields", Toast.LENGTH_SHORT).show();
            } else {
                AddHotel();
            }
        }
    }
});

    return rootView;
}

private void AddBusCustom() {
    custPrograssbar.progressCreate(getContext());
    custPrograssbar.setCancel(false);
    JSONObject jsonObject = new JSONObject();
    try {

        jsonObject.put("name", name.getText().toString().trim());
        jsonObject.put("description", desc.getText().toString());
        jsonObject.put("image", eventencodedImage);
        jsonObject.put("image_name", imgname);
        jsonObject.put("tot_seats", tot_seats.getText().toString());
        JsonParser jsonParser = new JsonParser();
        Call<JsonObject> call = ApiClientCustom.getInterface().addBus((JsonObject)
jsonParser.parse(jsonObject.toString()));


        GetResult getResult = new GetResult();
        getResult.setMyListener(this);
        getResult.onNCHandle(call, "addBus");

    } catch (JSONException e) {
        e.printStackTrace();
    }
}

private void AddHotel() {
```

```
        Call<String> call = null;
        try {
            call = apiInterface.addHotel(name.getText().toString().trim(), desc.getText().toString().trim(),
                    image.getText().toString().trim(), location.getText().toString().trim(),
city.getSelectedItem().toString());
        } catch (Exception e) {
        }
        call.enqueue(new Callback<String>() {
            @Override
            public void onResponse(Call<String> call, Response<String> response) {
                if (response.code() == 200) {
                    Toast.makeText(context, "Hotel Added", Toast.LENGTH_SHORT).show();
                    BookingFragment bookingFragment = new BookingFragment();
                    getActivity().getSupportFragmentManager().beginTransaction()
                            .replace(R.id.container, bookingFragment, "findThisFragment")
                            .addToBackStack(null)
                            .commit();
                }
            }

            @Override
            public void onFailure(Call<String> call, Throwable t) {
            }
        });
    }

    private void AddBus() {
        custPrograssbar.progressCreate(getContext());
        custPrograssbar.setCancel(false);
        Call<String> call = null;
        try {
            call = apiInterface.addBus(name.getText().toString().trim(), desc.getText().toString().trim(),
                    eventencodedImage, imgname, tot_seats.getText().toString().trim());
        } catch (Exception e) {
        }
        call.enqueue(new Callback<String>() {
            @Override
            public void onResponse(Call<String> call, Response<String> response) {
                if (response.code() == 200) {
                    Toast.makeText(context, "Bus Added", Toast.LENGTH_SHORT).show();
                    BookingFragment bookingFragment = new BookingFragment();
                    getActivity().getSupportFragmentManager().beginTransaction()
                            .replace(R.id.container, bookingFragment, "findThisFragment")
                            .addToBackStack(null)
                            .commit();
                }
            }

            @Override
            public void onFailure(Call<String> call, Throwable t) {
            }
        });
    }

    @Override
    public void onActivityResult(int requestCode, int resultCode, Intent data) {
```

```
    super.onActivityResult(requestCode, resultCode, data);
    if (resultCode == getActivity().RESULT_CANCELED) {
       return;
    }
    if (requestCode == CGALLERY) {
       if (data != null) {
          Uri contentURI = data.getData();
          try {

             Bitmap bitmap =
MediaStore.Images.Media.getBitmap(getContext().getContentResolver(), contentURI);
             ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();
             bitmap.compress(Bitmap.CompressFormat.JPEG, 5, byteArrayOutputStream);
             gallery_view.setVisibility(View.VISIBLE);
             gallery_view.setImageBitmap(bitmap);
             eventencodedImage = Base64.encodeToString(byteArrayOutputStream.toByteArray(),
Base64.DEFAULT);
             imgname = String.valueOf(Calendar.getInstance().getTimeInMillis());


          } catch (IOException e) {
             e.printStackTrace();
             Toast.makeText(getContext(), "Failed!", Toast.LENGTH_SHORT).show();
          }
       }
    } else if (requestCode == CAMERA) {


       try {
          Uri selectedImage = data.getData();
          bitmap = (Bitmap) data.getExtras().get("data");
          ByteArrayOutputStream bytes = new ByteArrayOutputStream();
          bitmap.compress(Bitmap.CompressFormat.JPEG, 5, bytes);
          String timeStamp = new SimpleDateFormat("yyyyMMdd_HHmmss",
Locale.getDefault()).format(new Date());
          destination = new File(Environment.getExternalStorageDirectory() + "/" +
                getString(R.string.app_name), "IMG_" + timeStamp + ".jpg");
          FileOutputStream fo;
          try {
             destination.createNewFile();
             fo = new FileOutputStream(destination);
             fo.write(bytes.toByteArray());
             fo.close();
          } catch (FileNotFoundException e) {
             e.printStackTrace();
          } catch (IOException e) {
             e.printStackTrace();
          }
          eventencodedImage = Base64.encodeToString(bytes.toByteArray(), Base64.DEFAULT);
          imgPath = destination.getAbsolutePath();
          gallery_view.setVisibility(View.VISIBLE);
          gallery_view.setImageBitmap(bitmap);

       } catch (Exception e) {
          e.printStackTrace();
```

```
        }

    } else {

    }

  }

  @Override
  public void onClick(View v) {
    switch (v.getId()) {

      case R.id.btn_choose:

        selectImage();

        break;
    }

  }

  private void selectImage() {

    try {
      PackageManager pm = getContext().getPackageManager();
      int hasPerm = pm.checkPermission(Manifest.permission.CAMERA,
getContext().getPackageName());
      if (hasPerm == PackageManager.PERMISSION_GRANTED) {
        final CharSequence[] options = {"Take Photo", "Choose From Gallery", "Cancel"};
        AlertDialog.Builder builder = new AlertDialog.Builder(getContext());
        builder.setTitle("Select Option");
        builder.setItems(options, new DialogInterface.OnClickListener() {
          @Override
          public void onClick(DialogInterface dialog, int item) {
            if (options[item].equals("Take Photo")) {
              dialog.dismiss();
              Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
              startActivityForResult(intent, CAMERA);
            } else if (options[item].equals("Choose From Gallery")) {
              dialog.dismiss();
              Intent pickPhoto = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
              startActivityForResult(pickPhoto, CGALLERY);
            } else if (options[item].equals("Cancel")) {
              dialog.dismiss();
            }
          }
        });
        builder.show();
      } else
        Toast.makeText(getContext(), "Camera Permission error",
Toast.LENGTH_SHORT).show();
    } catch (Exception e) {
      Toast.makeText(getContext(), "Camera Permission error", Toast.LENGTH_SHORT).show();
      e.printStackTrace();
    }
```

```
    }

    @Override
    public void callback(JsonObject result, String callNo) {
        if(callNo.equalsIgnoreCase("addBus")){
            custPrograssbar.close();
            Gson gson=new Gson();
            ResponseCommon
responseCommon=gson.fromJson(result.toString(),ResponseCommon.class);
            Toast.makeText(context, responseCommon.getResponseMsg(),
Toast.LENGTH_SHORT).show();
        }
        else if(callNo.equalsIgnoreCase("bjkdf")){


        }

    }
}//AddExpenseFragmennt.java

package com.dzniox.travel_mate.ui.fragments;



import android.content.Context;

import android.os.Bundle;

import androidx.fragment.app.Fragment;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

import android.widget.Toast;



import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.network.APIClient;
```

```
import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;

import retrofit2.Call;

import retrofit2.Callback;

import retrofit2.Response;

public class AddExpenseFragment extends Fragment {

  private static final String ARG_PARAM1 = "param1";

   private static final String ARG_PARAM2 = "param2";

   private String mParam1;

   private String mParam2;



   private Context context;

   private View rootView;

   APIInterface apiInterface;

   PrefUtils prefrences;

   private EditText amount,name,desc;

   private Button add_btn;









   public AddExpenseFragment() {

}

   public static AddExpenseFragment newInstance(String param1, String param2) {

      AddExpenseFragment fragment = new AddExpenseFragment();
```

```
    Bundle args = new Bundle();
    args.putString(ARG_PARAM1, param1);
    args.putString(ARG_PARAM2, param2);
    fragment.setArguments(args);
    return fragment;
}

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    // Inflate the layout for this fragment
    rootView = inflater.inflate(R.layout.fragment_add_expense, container, false);
    context = getContext();
    apiInterface = APIClient.getClient().create(APIInterface.class);
    prefrences = PrefUtils.getInstance(context);
```

```
    name = rootView.findViewById(R.id.name);
    amount = rootView.findViewById(R.id.amount);
    desc = rootView.findViewById(R.id.desc);
    add_btn = rootView.findViewById(R.id.add_btn);

    add_btn.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View view) {
        if (amount.getText().toString().trim().equals("") ||
            name.getText().toString().trim().equals("") ||
            desc.getText().toString().trim().equals("")) {
          Toast.makeText(context, "Check all fields.", Toast.LENGTH_SHORT).show();
        }else {
          AddExpense();
        }
      }
    });

    return rootView;
  }

  private void AddExpense() {
    Call<String> call = null;
```

```
try {
    call = apiInterface.addExpense(
            prefrences.getStringValue("id",""),
            name.getText().toString().trim(),
            amount.getText().toString().trim(),
            desc.getText().toString().trim()
    );
} catch (Exception e) {}
call.enqueue(new Callback<String>() {
    @Override
    public void onResponse(Call<String> call, Response<String> response) {
        if (response.code() == 200){
            Toast.makeText(context,"Expense Added",Toast.LENGTH_SHORT).show();
            ProfileFragment profileFragment = new ProfileFragment();
            getActivity().getSupportFragmentManager().beginTransaction()
                    .replace(R.id.container, profileFragment, "findThisFragment")
                    .addToBackStack(null)
                    .commit();
        }
    }
    @Override
    public void onFailure(Call<String> call, Throwable t) {
    }
});
```

```
    }
}
//AddHotelRoomsFragment .java
package com.dzniox.travel_mate.ui.fragments;
import android.Manifest;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.net.Uri;
import android.os.Bundle;
import androidx.fragment.app.FragmentManager;
import androidx.recyclerview.widget.RecyclerView;
import android.os.Environment;
import android.provider.MediaStore;
import android.util.Base64;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
```

```
import android.widget.ImageView;

import android.widget.TextView;

import android.widget.Toast;

import android.app.Activity;

import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.network.APIClient;

import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;

import com.dzniox.travel_mate.ui.activities.BusSignUp;

import org.json.JSONObject;

import java.io.ByteArrayOutputStream;

import java.io.File;

import java.io.FileNotFoundException;

import java.io.FileOutputStream;

import java.io.IOException;

import java.text.SimpleDateFormat;

import java.util.Calendar;

import java.util.Date;

import java.util.Locale;

import retrofit2.Call;

import retrofit2.Callback;

import retrofit2.Response;

public class AddHotelRoomsFragment extends Fragment {

   private static final String ARG_PARAM1 = "param1";
```

```
private static final String ARG_PARAM2 = "param2";


private String mParam1;

private String mParam2;

APIInterface apiInterface;

private Context context;

private View rootView;

private RecyclerView rv_users;

private  String id;

PrefUtils prefrences;

private Button addbus;

private File destination = null;

private ImageView gallery_view;

private Bitmap bitmap;

private String imgPath = null;

private final int CGALLERY = 1;

private final int CAMERA = 2;

String eventencodedImage = "", imgname = "";

private EditText ownername, ownerdesc, ownerroomno, ownerPhone, ownermaplink,roomname;

private TextView Login;

private Button ImageBtn;

public AddHotelRoomsFragment() {

}

public static AddBusFragment newInstance(String param1, String param2) {
```

```
    AddBusFragment fragment = new AddBusFragment();
    Bundle args = new Bundle();
    args.putString(ARG_PARAM1, param1);
    args.putString(ARG_PARAM2, param2);
    fragment.setArguments(args);
    return fragment;
}
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}
@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    rootView = inflater.inflate(R.layout.fragment_add_hotel_rooms, container, false);
    context = getContext();
    prefrences = PrefUtils.getInstance(context);
    apiInterface = APIClient.getClient().create(APIInterface.class);
    addbus = rootView.findViewById(R.id.addbowneradd_btn);
    ownerdesc = rootView.findViewById(R.id.addbownerdesc);
```

```
ownerroomno = rootView.findViewById(R.id.addbownerroom_no);
roomname = rootView.findViewById(R.id.roomname);
Login = rootView.findViewById(R.id.login);
ImageBtn=rootView.findViewById(R.id.addbownerbtn_choose);
gallery_view =rootView.findViewById(R.id.gallery_view);
id=prefrences.getStringValue("id", "");
onResume();
Log.d("key",prefrences.getStringValue("id", ""));
addbus.setOnClickListener(view -> {
   DoAddRooms(
         ownerdesc.getText().toString().trim(),
         ownerroomno.getText().toString().trim(),
         roomname.getText().toString().trim(),
         eventencodedImage,
         imgname,
         id
   );
});
ImageBtn.setOnClickListener(new View.OnClickListener() {
   @Override
   public void onClick(View view) {
      Log.d("key","btn clicked");
      selectImage();
   }
```

```
        });
        return rootView;
    }
    @Override
    public void onResume() {
        super.onResume();
        roomname.setText("");
        ownerdesc.setText("");
        ownerroomno.setText("");
    }
    @Override
    public void onActivityResult(int requestCode, int resultCode, Intent data) {
        super.onActivityResult(requestCode, resultCode, data);
        if (resultCode == Activity.RESULT_CANCELED) {
            return;
        }
        if (requestCode == CGALLERY) {
            if (data != null) {
                Uri contentURI = data.getData();
                try {
MediaStore.Images.Media.getBitmap(this.context.getContentResolver(), contentURI);
                    Bitmap bitmap =
                    ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();
                    bitmap.compress(Bitmap.CompressFormat.JPEG, 5, byteArrayOutputStream);
                    gallery_view.setVisibility(View.VISIBLE);
```

```
                gallery_view.setImageBitmap(bitmap);
Base64.DEFAULT);encodedImage = Base64.encodeToString(byteArrayOutputStream.toByteArray(),
                imgname = String.valueOf(Calendar.getInstance().getTimeInMillis());
            } catch (IOException e) {
                e.printStackTrace();
                Toast.makeText(this.context, "Failed!", Toast.LENGTH_SHORT).show();
            }
        }
    } else if (requestCode == CAMERA) {
        try {
            Uri selectedImage = data.getData();
            bitmap = (Bitmap) data.getExtras().get("data");
            ByteArrayOutputStream bytes = new ByteArrayOutputStream();
            bitmap.compress(Bitmap.CompressFormat.JPEG, 5, bytes);
Locale.getDefault()).format(new Date());SimpleDateFormat("yyyyMMdd_HHmmss",
            destination = new File(Environment.getExternalStorageDirectory() + "/" +
                    getString(R.string.app_name), "IMG_" + timeStamp + ".jpg");
            FileOutputStream fo;
            try {
                destination.createNewFile();
                fo = new FileOutputStream(destination);
                fo.write(bytes.toByteArray());
                fo.close();
            } catch (FileNotFoundException e) {
```

```
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        }
        eventencodedImage = Base64.encodeToString(bytes.toByteArray(), Base64.DEFAULT);
        imgPath = destination.getAbsolutePath();
        gallery_view.setVisibility(View.VISIBLE);
        gallery_view.setImageBitmap(bitmap);

      } catch (Exception e) {
        e.printStackTrace();
      }

    } else {
    }
  }
  r ARooS,r ownerroomno,String roomname,String eventencodedImage,String imgname,String id) {
roomname.equals("")){
    if(ownerdesc.equals("") || ownerroomno.equals("") ||  eventencodedImage.equals("") ||
      Toast.makeText(this.context,"Check Fields",Toast.LENGTH_LONG).show();
    }else {
apiInterface.addgoodDet(ownerdesc,ownerroomno,roomname,eventencodedImage,imgname,id);
      Call<String> call =
      call.enqueue(new Callback<String>() {
        @Override
        public void onResponse(Call<String> call, Response<String> response) {
```

```
                if (response.code() == 200){
Replace with the actual fragment class you want to navigate to ViewHotelRoomsFragment(); //
getActivity().getSupportFragmentManager() if you are in an activity
                    FragmentManager fragmentManager = getParentFragmentManager(); // Use
                    FragmentTransaction transaction = fragmentManager.beginTransaction();
with the ID of your fragment container (R.id.
                    transaction.replace(R.id.container, nextFragment); // Replace R.id.fragment_container
button to navigate back
                    transaction.addToBackStack(null); // Add to back stack if you want to enable the back
                    transaction.commit();
Toast.LENGTH_SHORT).show();
                    Toast.makeText(context, "Room added successfully",
                } else {
Toast.LENGTH_SHORT).show();
                    Toast.makeText(context, "Failed to add Room. Please try again.",
                }
            }
            @Override
            public void onFailure(Call<String> call, Throwable t) {
            }
        });
    }
}




  private void selectImage() {


    try {
```

```
        PackageManager pm = this.context.getPackageManager();
int hasPerm = pm.checkPermission(Manifest.permission.CAMERA,
this.context.getPackageName());
        if (hasPerm == PackageManager.PERMISSION_GRANTED) {
           final CharSequence[] options = {"Take Photo", "Choose From Gallery", "Cancel"};
           AlertDialog.Builder builder = new AlertDialog.Builder(this.context);
           builder.setTitle("Select Option");
           builder.setItems(options, new DialogInterface.OnClickListener() {
              @Override
              public void onClick(DialogInterface dialog, int item) {
                 if (options[item].equals("Take Photo")) {
                    dialog.dismiss();
                    Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
                    startActivityForResult(intent, CAMERA);
                 } else if (options[item].equals("Choose From Gallery")) {
                    dialog.dismiss();
Intent pickPhoto = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
                    startActivityForResult(pickPhoto, CGALLERY);
                 } else if (options[item].equals("Cancel")) {
                    dialog.dismiss();
                 }
              }
           });
           builder.show();
        } else
```

```
Toast.LENGTH_SHORT).show();
Toast.makeText(this.context, "Camera Permission error",
        } catch (Exception e) {
            Toast.makeText(this.context, "Camera Permission error", Toast.LENGTH_SHORT).show();
            e.printStackTrace();
        }
    }
}
```

/AdminAddPackageFragment.java/

```
package com.dzniox.travel_mate.ui.fragments;

import android.app.DatePickerDialog;

import android.content.Context;

import android.os.Bundle;


import androidx.appcompat.app.AppCompatActivity;

import androidx.fragment.app.Fragment;

import androidx.recyclerview.widget.RecyclerView;


import android.util.Log;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.DatePicker;

import android.widget.ImageView;

import android.widget.TextView;
```

```
import com.dzniox.travel_mate.Model.AllBusesByAdmin;

import com.dzniox.travel_mate.NetworkUtilsCustom.GetResult;

import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.databinding.FragmentAdminAddPackageBinding;

import com.dzniox.travel_mate.network.APIClient;

import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;


import java.text.SimpleDateFormat;

import java.util.Calendar;

import java.util.Locale;

import android.app.DatePickerDialog;

import android.os.Bundle;

import android.view.View;

import android.widget.AdapterView;

import android.widget.ArrayAdapter;

import android.widget.DatePicker;

import android.widget.Toast;


import androidx.appcompat.app.AppCompatActivity;


import com.dzniox.travel_mate.Model.AllBusbyadmin;

import com.dzniox.travel_mate.Model.AllBusesByAdmin;
```

```
import com.dzniox.travel_mate.Model.Example;

import com.dzniox.travel_mate.Model.ResponseCommon;

import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;

import com.dzniox.travel_mate.NetworkUtilsCustom.GetResult;

import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.databinding.ActivityTripassignBinding;

//import com.dzniox.travel_mate.ui.activities.TripassignActivity;

import com.dzniox.travel_mate.utils.CustPrograssbar;

import com.google.gson.Gson;

import com.google.gson.JsonObject;

import com.google.gson.JsonParser;

import org.json.JSONException;

import org.json.JSONObject;

import java.text.SimpleDateFormat;

import java.util.ArrayList;

import java.util.Calendar;

import java.util.List;

import java.util.Locale;

import com.dzniox.travel_mate.databinding.FragmentAdminAddPackageBinding;

import retrofit2.Call;

import retrofit2.Callback;

import retrofit2.Response;

import android.app.DatePickerDialog;

import android.content.Context;
```

```
import android.content.Intent;

import android.net.Uri;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.Button;

import android.widget.DatePicker;

import android.widget.EditText;

import android.widget.ImageView;

import android.widget.LinearLayout;

import android.widget.TextView;

import android.widget.Toast;


import androidx.annotation.NonNull;

import androidx.recyclerview.widget.RecyclerView;

import com.bumptech.glide.Glide;

import com.dzniox.travel_mate.PGIntegration.RazorPayActivity;

import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.network.APIClient;

import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.network.MessageClient;

import com.dzniox.travel_mate.network.MessageInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;

import com.dzniox.travel_mate.ui.activities.WebViewActivity;
```

```
import org.json.JSONArray;

import org.json.JSONException;

import org.json.JSONObject;


import java.text.SimpleDateFormat;

import java.util.Calendar;

import java.util.Locale;


public class AdminAddPackageFragment extends Fragment {



    private static final String ARG_PARAM1 = "param1";

    private static final String ARG_PARAM2 = "param2";


    private String mParam1;

    private String mParam2;

    private TextView fdate, ldate;

    APIInterface apiInterface;

    private Context context;

    private View rootView;

    private RecyclerView rv_users;

    final Calendar myCalendar = Calendar.getInstance();

    PrefUtils prefrences;
```

```
private AdminAddPackageFragment adminaddPackagefragment;

private FragmentAdminAddPackageBinding binding;

List<AllBusesByAdmin> Buslist;

ArrayList<AllBusesByAdmin> Buslists = new ArrayList<>();


private ArrayList<String> DISTRICTS = new ArrayList<>();

String spot_id, bus_id;


CustPrograssbar custPrograssbar;

public AdminAddPackageFragment() {

}


public static AdminAddPackageFragment newInstance(String param1, String param2) {

    AdminAddPackageFragment fragment = new AdminAddPackageFragment();

    Bundle args = new Bundle();

    args.putString(ARG_PARAM1, param1);

    args.putString(ARG_PARAM2, param2);

    fragment.setArguments(args);

    return fragment;

}


@Override

public void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);
```

```
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    rootView = inflater.inflate(R.layout.fragment_admin_add_package, container, false);
    context = getContext();
    prefrences = PrefUtils.getInstance(context);
    apiInterface = APIClient.getClient().create(APIInterface.class);
    fdate = rootView.findViewById(R.id.fdate);
    ldate = rootView.findViewById(R.id.ldate);
    binding = FragmentAdminAddPackageBinding.inflate(getLayoutInflater());

    fdate.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view){

{           DatePickerDialog.OnDateSetListener date = new DatePickerDialog.OnDateSetListener()
                @Override
```

```
            public void onDateSet(DatePicker view, int year, int monthOfYear,
                                  int dayOfMonth) {
                myCalendar.set(Calendar.YEAR, year);
                myCalendar.set(Calendar.MONTH, monthOfYear);
                myCalendar.set(Calendar.DAY_OF_MONTH, dayOfMonth);
                updateLabel();
            }
        };

        DatePickerDialog dialog = new DatePickerDialog(context, date, myCalendar
            .get(Calendar.YEAR), myCalendar.get(Calendar.MONTH),
            myCalendar.get(Calendar.DAY_OF_MONTH));
        dialog.getDatePicker().setMinDate(System.currentTimeMillis());
        dialog.show();
    }
});
ldate.setOnClickListener(new View.OnClickListener() {
  @Override
  public void onClick(View view) {
    DatePickerDialog.OnDateSetListener date = new DatePickerDialog.OnDateSetListener() {
      @Override
      public void onDateSet(DatePicker view, int year, int monthOfYear,
                            int dayOfMonth) {
        myCalendar.set(Calendar.YEAR, year);
```

```
            myCalendar.set(Calendar.MONTH, monthOfYear);
            myCalendar.set(Calendar.DAY_OF_MONTH, dayOfMonth);
            updateLabel2();
        }
      };

      DatePickerDialog dialog = new DatePickerDialog(context, date, myCalendar
            .get(Calendar.YEAR), myCalendar.get(Calendar.MONTH),
            myCalendar.get(Calendar.DAY_OF_MONTH));
      dialog.getDatePicker().setMinDate(System.currentTimeMillis());
      dialog.show();
    }
  });
  binding.spnSpot.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
    @Override
    public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {
      AllBusesByAdmin s1 = (AllBusesByAdmin) parent.getSelectedItem();
      spot_id = s1.getBus_id();
    }
    @Override
    public void onNothingSelected(AdapterView<?> parent) {
    }
  });
  getAllSpots();
```

```
        return rootView;
    }
    private void updateLabel() {
        String myFormat = "yyyy-MM-dd";
        SimpleDateFormat sdf = new SimpleDateFormat(myFormat, Locale.US);
        fdate.setText(sdf.format(myCalendar.getTime()));
    }

    private void updateLabel2() {
        String myFormat = "yyyy-MM-dd";
        SimpleDateFormat sdf = new SimpleDateFormat(myFormat, Locale.US);
        ldate.setText(sdf.format(myCalendar.getTime()));
    }
    private void getAllSpots() {
        JSONObject jsonObject = new JSONObject();
        try {
            jsonObject.put("ClientID", "0");
            JsonParser jsonParser = new JsonParser();
jsonParser.parse(jsonObject.toString()));Call<JsonObject> call = ApiClientCustom.getInterface().getAllBusesid((JsonObject)
            call.enqueue(new Callback<JsonObject>() {
                @Override
                public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {
response.body().toString(),"$$$
```

```
                JsonObject object = response.body();

Log.d("Key","$$$
```

```
                binding.spnSpot.setAdapter(adapter);
                binding.spnSpot.setTitle("Select Item");
                binding.spnSpot.setPositiveButton("OK");
            }
        }
        @Override
        public void onFailure(Call<JsonObject> call, Throwable t) {
        }
      });
    } catch (JSONException e) {
      e.printStackTrace();
    }
  }
}
//AdminPackageFragment.java
package com.dzniox.travel_mate.ui.fragments;

import android.annotation.SuppressLint;
import android.content.Context;
import android.content.Intent;
import android.os.Bundle;

import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentManager;
```

```
import androidx.fragment.app.FragmentTransaction;

import androidx.recyclerview.widget.RecyclerView;


import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.Button;

import android.widget.ImageView;

import android.widget.TextView;


import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.network.APIClient;

import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;

import android.annotation.SuppressLint;

import android.content.Context;

import android.os.Build;

import android.os.Bundle;

import androidx.annotation.RequiresApi;

import androidx.fragment.app.Fragment;

import androidx.recyclerview.widget.LinearLayoutManager;

import androidx.recyclerview.widget.RecyclerView;

import android.util.Log;

import android.view.LayoutInflater;
```

```
import android.view.View;

import android.view.ViewGroup;

import android.widget.EditText;

import android.widget.TextView;

import android.widget.Toast;

import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.network.APIClient;

import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;

import com.dzniox.travel_mate.ui.activities.TripassignActivity;

import com.dzniox.travel_mate.ui.adapters.AdminViewBookingsAdapter;

import com.dzniox.travel_mate.ui.adapters.BookingAdapter;

import com.dzniox.travel_mate.ui.adapters.BusOwnersListAdapter;

import com.dzniox.travel_mate.ui.adapters.AdmiViewPackageAdapter;

import com.dzniox.travel_mate.ui.adapters.HotelOwnersListAdapter;

import com.dzniox.travel_mate.ui.adapters.HotelRequestAdapter;

import com.dzniox.travel_mate.ui.adapters.ExpenseAdapter;

import com.dzniox.travel_mate.ui.adapters.SpotAdapter;

import com.dzniox.travel_mate.ui.adapters.UserBookingsviewAdapter;

import org.json.JSONObject;

import retrofit2.Call;

import retrofit2.Callback;

import retrofit2.Response;
```

```
public class AdminPackageFragment extends Fragment {


    private static final String ARG_PARAM1 = "param1";

    private static final String ARG_PARAM2 = "param2";


    private String mParam1;

    private String mParam2;

    private TextView Name, Email, no_data, Phone, bus_Owners, hotel_Owners, users;
private ImageView add_package;
private Button btn;
    APIInterface apiInterface;

    private Context context;

    private View rootView;

    private RecyclerView rv_package;


    PrefUtils prefrences;

    private TextView  Packages, Bookings, mark_exp;


    private AdminAddPackageFragment adminaddPackagefragment;


    public AdminPackageFragment() {

    }


    public static AdminPackageFragment newInstance(String param1, String param2) {
```

```
    AdminPackageFragment fragment = new AdminPackageFragment();
    Bundle args = new Bundle();
    args.putString(ARG_PARAM1, param1);
    args.putString(ARG_PARAM2, param2);
    fragment.setArguments(args);
    return fragment;
}

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}
@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    // Inflate the layout for this fragment
    rootView = inflater.inflate(R.layout.fragment_admin_package, container, false);
    context = getContext();
    prefrences = PrefUtils.getInstance(context);
    Packages = rootView.findViewById(R.id.packages);
```

```
        Bookings = rootView.findViewById(R.id.bookings);
        no_data = rootView.findViewById(R.id.no_data);
        rv_package = rootView.findViewById(R.id.rv_package);
        apiInterface = APIClient.getClient().create(APIInterface.class);
        btn= rootView.findViewById(R.id.add_package);
        if (prefrences.getStringValue("usertype","").equalsIgnoreCase("0")) {
            btn.setVisibility(View.GONE);
        }

        adminaddPackagefragment=new AdminAddPackageFragment();
        getPackageList();

        btn.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View view) {
Log.d("key","*************************************************************************");
                startActivity(new Intent(getContext(), TripassignActivity.class));



            }
        });
        Packages.setOnClickListener(new View.OnClickListener() {
            @RequiresApi(api = Build.VERSION_CODES.M)
```

```
        @Override
        public void onClick(View view) {
            getPackageList();

            Packages.setTextColor(context.getColor(R.color.blue));
            Bookings.setTextColor(context.getColor(R.color.text_color));

        }
    });
    Bookings.setOnClickListener(new View.OnClickListener() {
        @RequiresApi(api = Build.VERSION_CODES.M)
        @Override
        public void onClick(View view) {
            if (prefrences.getStringValue("usertype","").equalsIgnoreCase("0")) {
                getBookings();
            }
            if (prefrences.getStringValue("email","").equalsIgnoreCase("admin@gmail.com") &&
                getpackageBookings();

            }
            Packages.setTextColor(context.getColor(R.color.text_color));
            Bookings.setTextColor(context.getColor(R.color.blue));
```

```
        }
      });
return  rootView;

  }
   private void getPackageList() {

     Call<String> call = apiInterface.getallpackages("packages");

     call.enqueue(new Callback<String>() {
       @Override
       public void onResponse(Call<String> call, Response<String> response) {
         if (response.code() == 200) {
           rv_package.setVisibility(View.VISIBLE);
           try {
             JSONObject data = new JSONObject(response.body());
             Log.d("key",response.body());
Log.d("key","HEYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY");
             if (data.getJSONArray("data").length() > 0) {
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");


               no_data.setVisibility(View.GONE);
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");
```

```
                    rv_package.setVisibility(View.VISIBLE);

Log.d("key","adStart**********************************************************************");


data.getJSONArray("data"));                AdmiViewPackageAdapter adapter = new AdmiViewPackageAdapter(context,

Log.d("key","adEnd************************************************************************");
                    rv_package.setLayoutManager(new LinearLayoutManager(context));
                    rv_package.setAdapter(adapter);
                } else {
                    no_data.setVisibility(View.VISIBLE);
                    rv_package.setVisibility(View.GONE);
                }
            } catch (Exception e) {
                no_data.setVisibility(View.VISIBLE);
                rv_package.setVisibility(View.GONE);
            }

        } else {
            no_data.setVisibility(View.VISIBLE);
            rv_package.setVisibility(View.GONE);
        }
    }

    @Override
    public void onFailure(Call<String> call, Throwable t) {
```

```
            no_data.setVisibility(View.VISIBLE);
            rv_package.setVisibility(View.GONE);
        }
    });
}
private void getBookings() {

    Call<String> call = apiInterface.getallbookings(prefrences.getStringValue("id", ""));

    call.enqueue(new Callback<String>() {
        @Override
        public void onResponse(Call<String> call, Response<String> response) {
            if (response.code() == 200) {
                rv_package.setVisibility(View.VISIBLE);
                try {
                    JSONObject data = new JSONObject(response.body());
                    Log.d("key",response.body());
Log.d("key","HEYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY");
                    if (data.getJSONArray("data").length() > 0) {
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");
                        no_data.setVisibility(View.GONE);
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");
                        rv_package.setVisibility(View.VISIBLE);
Log.d("key","adStart************************************************************");
```

```
data.getJSONArray("data"),UserBookingsviewAdapter adapter = new UserBookingsviewAdapter(context,

Log.d("key","adEnd**********************************************************************");
                    rv_package.setLayoutManager(new LinearLayoutManager(context));
                    rv_package.setAdapter(adapter);
                } else {
                    no_data.setVisibility(View.VISIBLE);
                    rv_package.setVisibility(View.GONE);
                }
            } catch (Exception e) {
                no_data.setVisibility(View.VISIBLE);
                rv_package.setVisibility(View.GONE);
            }

        } else {
            no_data.setVisibility(View.VISIBLE);
            rv_package.setVisibility(View.GONE);
        }
    }

    @Override
    public void onFailure(Call<String> call, Throwable t) {
        no_data.setVisibility(View.VISIBLE);
        rv_package.setVisibility(View.GONE);
```

```
        }
    });
}

private void getpackageBookings(){

    Call<String> call = apiInterface.getallpackages("packages");

    call.enqueue(new Callback<String>() {
        @Override
        public void onResponse(Call<String> call, Response<String> response) {
            if (response.code() == 200) {
                rv_package.setVisibility(View.VISIBLE);
                try {
                    JSONObject data = new JSONObject(response.body());
                    Log.d("key",response.body());
Log.d("key","HEYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY");
                    if (data.getJSONArray("data").length() > 0) {
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");


                        no_data.setVisibility(View.GONE);
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");
                        rv_package.setVisibility(View.VISIBLE);
```

```
Log.d("key","adStart**********************************************************************");


data.getJSONArray("data"));                AdminViewBookingsAdapter adapter = new AdminViewBookingsAdapter(context,

Log.d("key","adEnd************************************************************************");
                    rv_package.setLayoutManager(new LinearLayoutManager(context));
                    rv_package.setAdapter(adapter);
                } else {
                    no_data.setVisibility(View.VISIBLE);
                    rv_package.setVisibility(View.GONE);
                }
            } catch (Exception e) {
                no_data.setVisibility(View.VISIBLE);
                rv_package.setVisibility(View.GONE);
            }


        } else {
            no_data.setVisibility(View.VISIBLE);
            rv_package.setVisibility(View.GONE);
        }
    }


    @Override
    public void onFailure(Call<String> call, Throwable t) {
        no_data.setVisibility(View.VISIBLE);
```

```
            rv_package.setVisibility(View.GONE);
        }
    });
  }
}
//AdminViewBookingsFragment .java
package com.dzniox.travel_mate.ui.fragments;
import android.os.Bundle;
import androidx.fragment.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import com.dzniox.travel_mate.R;
public class AdminViewBookingsFragment extends Fragment {
  private static final String ARG_PARAM1 = "param1";
  private static final String ARG_PARAM2 = "param2";
  private String mParam1;
  private String mParam2;
  public AdminViewBookingsFragment() {
  }

  public static AdminViewBookingsFragment newInstance(String param1, String param2) {
    AdminViewBookingsFragment fragment = new AdminViewBookingsFragment();
```

```
        Bundle args = new Bundle();
        args.putString(ARG_PARAM1, param1);
        args.putString(ARG_PARAM2, param2);
        fragment.setArguments(args);
        return fragment;
    }

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            mParam1 = getArguments().getString(ARG_PARAM1);
            mParam2 = getArguments().getString(ARG_PARAM2);
        }
    }
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        return inflater.inflate(R.layout.fragment_admin_view_bookings, container, false);
    }
}
//AdminViewBusesFragment.java
package com.dzniox.travel_mate.ui.fragments;
```

```
import android.content.Context;

import android.os.Bundle;


import androidx.fragment.app.Fragment;

import androidx.recyclerview.widget.LinearLayoutManager;

import androidx.recyclerview.widget.RecyclerView;


import android.util.Log;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.TextView;


import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.network.APIClient;

import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;

import com.dzniox.travel_mate.ui.adapters.AdminViewBusesAdapter;

import com.dzniox.travel_mate.ui.adapters.BusOwnersListAdapter;


import org.json.JSONArray;

import org.json.JSONObject;


import retrofit2.Call;
```

```
import retrofit2.Callback;
import retrofit2.Response;
public class AdminViewBusesFragment extends Fragment {
   private String ownerId;
   public AdminViewBusesFragment(String ownerId) {
      this.ownerId = ownerId;
   }

   private static final String ARG_PARAM1 = "param1";
   private static final String ARG_PARAM2 = "param2";
   private String mParam1;
   private String mParam2;
   APIInterface apiInterface;
   private Context context;
   private TextView no_data;
   private View rootView;
   private RecyclerView adminbusview;
   PrefUtils prefrences;
   public AdminViewBusesFragment() {
   }


   public static AdminViewBusesFragment newInstance(String param1, String param2) {
      AdminViewBusesFragment fragment = new AdminViewBusesFragment();
```

```
    Bundle args = new Bundle();
    args.putString(ARG_PARAM1, param1);
    args.putString(ARG_PARAM2, param2);
    fragment.setArguments(args);
    return fragment;
}

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    // Inflate the layout for this fragment
    rootView= inflater.inflate(R.layout.fragment_admin_view_buses, container, false);
    context = getContext();
    prefrences = PrefUtils.getInstance(context);
    apiInterface = APIClient.getClient().create(APIInterface.class);
```

```
    adminbusview = rootView.findViewById(R.id.adminbusview);
    no_data = rootView.findViewById(R.id.no_data);
    getBusesList();
    return rootView;
}

private void getBusesList() {

    Call<String> call = apiInterface.getbustoadmin(ownerId);

    call.enqueue(new Callback<String>() {
        @Override
        public void onResponse(Call<String> call, Response<String> response) {
            if (response.code() == 200) {
                adminbusview.setVisibility(View.VISIBLE);
                try {
                    JSONObject data = new JSONObject(response.body());
                    JSONArray dataArray = data.getJSONArray("data");

                    Log.d("key",response.body());
```

```
Log.d("key","HEYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY");
                if (data.getJSONArray("data").length() > 0) {
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");



                    no_data.setVisibility(View.GONE);
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");
                    adminbusview.setVisibility(View.VISIBLE);
Log.d("key","adStart*****************************************************************");


data.getJSONArray("data"));                AdminViewBusesAdapter adapter = new AdminViewBusesAdapter(context,
Log.d("key","adEnd*******************************************************************");
                    adminbusview.setLayoutManager(new LinearLayoutManager(context));
                    adminbusview.setAdapter(adapter);
                } else {
                    no_data.setVisibility(View.VISIBLE);
                    adminbusview.setVisibility(View.GONE);
                }
            } catch (Exception e) {
                no_data.setVisibility(View.VISIBLE);
                adminbusview.setVisibility(View.GONE);
            }
```

```
            } else {
                no_data.setVisibility(View.VISIBLE);
                adminbusview.setVisibility(View.GONE);
            }
        }

        @Override
        public void onFailure(Call<String> call, Throwable t) {
            no_data.setVisibility(View.VISIBLE);
            adminbusview.setVisibility(View.GONE);
        }
    });
  }

}
```

//AdminViewPackageFragment.java

```
package com.dzniox.travel_mate.ui.fragments;

import android.os.Bundle;

import androidx.fragment.app.Fragment;

import android.view.LayoutInflater;
```

```
import android.view.View;

import android.view.ViewGroup;


import com.dzniox.travel_mate.R;


public class AdminViewPackageFragment extends Fragment {


    private static final String ARG_PARAM1 = "param1";

    private static final String ARG_PARAM2 = "param2";


    private String mParam1;

    private String mParam2;


    public AdminViewPackageFragment() {

    }



    public static AdminViewPackageFragment newInstance(String param1, String param2) {

        AdminViewPackageFragment fragment = new AdminViewPackageFragment();

        Bundle args = new Bundle();

        args.putString(ARG_PARAM1, param1);

        args.putString(ARG_PARAM2, param2);

        fragment.setArguments(args);

        return fragment;
```

```
    }

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            mParam1 = getArguments().getString(ARG_PARAM1);
            mParam2 = getArguments().getString(ARG_PARAM2);
        }
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        return inflater.inflate(R.layout.fragment_admin_view_package, container, false);
    }
}

//AdminViewRoomsFragment .java
package com.dzniox.travel_mate.ui.fragments;

import android.content.Context;
import android.os.Bundle;
```

```
import androidx.fragment.app.Fragment;

import androidx.recyclerview.widget.LinearLayoutManager;

import androidx.recyclerview.widget.RecyclerView;


import android.util.Log;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.TextView;


import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.network.APIClient;

import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;

import com.dzniox.travel_mate.ui.adapters.AdminViewBusesAdapter;

import com.dzniox.travel_mate.ui.adapters.AdminViewRoomsAdapter;

import org.json.JSONArray;

import org.json.JSONObject;

import retrofit2.Call;

import retrofit2.Callback;

import retrofit2.Response;

public class AdminViewRoomsFragment extends Fragment {

   private String ownerId;
```

```
public AdminViewRoomsFragment(String ownerId) {
    this.ownerId = ownerId;
}
private static final String ARG_PARAM1 = "param1";
private static final String ARG_PARAM2 = "param2";

private String mParam1;
private String mParam2;
APIInterface apiInterface;
private Context context;
private TextView no_data;
private View rootView;
private RecyclerView adminroomsview;
PrefUtils prefrences;
public AdminViewRoomsFragment() {

}

public static AdminViewRoomsFragment newInstance(String param1, String param2) {
    AdminViewRoomsFragment fragment = new AdminViewRoomsFragment();
    Bundle args = new Bundle();
    args.putString(ARG_PARAM1, param1);
    args.putString(ARG_PARAM2, param2);
    fragment.setArguments(args);
```

```
    return fragment;
}

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    rootView= inflater.inflate(R.layout.fragment_admin_view_rooms, container, false);

    context = getContext();
    prefrences = PrefUtils.getInstance(context);
    apiInterface = APIClient.getClient().create(APIInterface.class);

    adminroomsview = rootView.findViewById(R.id.adminroomsview);
    no_data = rootView.findViewById(R.id.no_data);
    getroomsList();
```

```
    return  rootView;
  }
  private void getroomsList() {
    Call<String> call = apiInterface.getroomstoadmin(ownerId);
    call.enqueue(new Callback<String>() {
      @Override
      public void onResponse(Call<String> call, Response<String> response) {
        if (response.code() == 200) {
          adminroomsview.setVisibility(View.VISIBLE);
          try {
            JSONObject data = new JSONObject(response.body());
            JSONArray dataArray = data.getJSONArray("data");

            Log.d("key",response.body());

Log.d("key","HEYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY");
            if (data.getJSONArray("data").length() > 0) {
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");

              no_data.setVisibility(View.GONE);
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");
              adminroomsview.setVisibility(View.VISIBLE);
Log.d("key","adStart*****************************************************************");
AdminViewRoomsAdapter adapter = new AdminViewRoomsAdapter
```

```
(context, data.getJSONArray("data"));

Log.d("key","adEnd**********************************************************************");
                    adminroomsview.setLayoutManager(new LinearLayoutManager(context));
                    adminroomsview.setAdapter(adapter);
                } else {
                    no_data.setVisibility(View.VISIBLE);
                    adminroomsview.setVisibility(View.GONE);
                }
            } catch (Exception e) {
                no_data.setVisibility(View.VISIBLE);
                adminroomsview.setVisibility(View.GONE);
            }

        } else {
            no_data.setVisibility(View.VISIBLE);
            adminroomsview.setVisibility(View.GONE);
        }
    }

    @Override
    public void onFailure(Call<String> call, Throwable t) {
        no_data.setVisibility(View.VISIBLE);
        adminroomsview.setVisibility(View.GONE);
    }
```

```
        });
    }

}
```

//BookingFragment .java

```
package com.dzniox.travel_mate.ui.fragments;

import android.app.DatePickerDialog;

import android.content.Context;

import android.content.Intent;

import android.os.Build;

import android.os.Bundle;


import androidx.annotation.RequiresApi;

import androidx.appcompat.widget.AppCompatButton;

import androidx.fragment.app.Fragment;

import androidx.recyclerview.widget.LinearLayoutManager;

import androidx.recyclerview.widget.RecyclerView;


import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.DatePicker;

import android.widget.EditText;
```

```
import android.widget.LinearLayout;

import android.widget.Spinner;

import android.widget.TextView;

import android.widget.Toast;


import com.dzniox.travel_mate.Model.Example;

import com.dzniox.travel_mate.Model.TripChartList;

import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;

import com.dzniox.travel_mate.NetworkUtilsCustom.GetResult;

import com.dzniox.travel_mate.PGIntegration.RazorPayActivity;

import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.databinding.FragmentBookingBinding;

import com.dzniox.travel_mate.network.APIClient;

import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;

import com.dzniox.travel_mate.ui.adapters.BusAdapter;

import com.dzniox.travel_mate.ui.adapters.HotelAdapter;

import com.dzniox.travel_mate.ui.adapters.SpotAdapter;

import com.dzniox.travel_mate.ui.adapters.TripChartAdapter;

import com.dzniox.travel_mate.utils.CustPrograssbar;

import com.dzniox.travel_mate.utils.RecyclerViewClickInterface;

import com.google.android.material.textfield.TextInputEditText;

import com.google.gson.Gson;

import com.google.gson.JsonObject;
```

```
import com.google.gson.JsonParser;


import org.json.JSONArray;

import org.json.JSONException;

import org.json.JSONObject;

import java.text.ParseException;

import java.text.SimpleDateFormat;

import java.util.ArrayList;

import java.util.Calendar;

import java.util.Date;

import java.util.List;

import java.util.Locale;


import retrofit2.Call;

import retrofit2.Callback;

import retrofit2.Response;

public class BookingFragment extends Fragment implements

 GetResult.MyListener, RecyclerViewClickInterface {

    FragmentBookingBinding binding;

    final Calendar myCalendar = Calendar.getInstance();

    final Calendar myCalendarnew = Calendar.getInstance();

    List<TripChartList> tripChartLists;


    private static final String ARG_PARAM1 = "param1";
```

```
private static final String ARG_PARAM2 = "param2";

CustPrograssbar custPrograssbar;


private String mParam1;

private String mParam2;


private Context context;

private View rootView;


APIInterface apiInterface;

PrefUtils prefrences;

private RecyclerView rv_data;


private LinearLayout add_bus, add_hotel, lt1, lt2;


private TextView bus_booking, hotel_booking;


EditText edt_from, edt_to, edt_passenger;

String edt_from1, edt_to1;

Spinner city;

AppCompatButton button;

private String SELECTED_PAGE = "bus";


public BookingFragment() {
```

```
}



public static BookingFragment newInstance(String param1, String param2) {
    BookingFragment fragment = new BookingFragment();
    Bundle args = new Bundle();
    args.putString(ARG_PARAM1, param1);
    args.putString(ARG_PARAM2, param2);
    fragment.setArguments(args);
    return fragment;
}

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}



@Override
public void onResume() {
```

```
    super.onResume();


}


@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {


    rootView = inflater.inflate(R.layout.fragment_booking, container, false);
    context = getContext();
    apiInterface = APIClient.getClient().create(APIInterface.class);
    prefrences = PrefUtils.getInstance(context);
    custPrograssbar = new CustPrograssbar();
    hotel_booking = rootView.findViewById(R.id.hotel_booking);
    lt1 = rootView.findViewById(R.id.lt1);
    lt2 = rootView.findViewById(R.id.lt2);
    bus_booking = rootView.findViewById(R.id.bus_booking);
    rv_data = rootView.findViewById(R.id.rv_data);
    add_bus = rootView.findViewById(R.id.add_bus);
    add_hotel = rootView.findViewById(R.id.add_hotel);
    edt_from = rootView.findViewById(R.id.edt_fromdate);
    edt_to = rootView.findViewById(R.id.edt_todate);
    edt_passenger = rootView.findViewById(R.id.edt_passenger);
    button = rootView.findViewById(R.id.search);
```

```
        city = rootView.findViewById(R.id.sp_city);
        lt1.setVisibility(View.VISIBLE);
        lt2.setVisibility(View.VISIBLE);
        button.setVisibility(View.VISIBLE);
        if (prefrences.getStringValue("id", "").equals("1")) {
            add_bus.setVisibility(View.VISIBLE);
            add_hotel.setVisibility(View.VISIBLE);

        } else {
            add_bus.setVisibility(View.GONE);
            add_hotel.setVisibility(View.GONE);
        }
        edt_to.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                showDatepopupTo();
            }
        });
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
edt_passenger.getText().toString().isEmpty()) {
                if (edt_from.getText().toString().isEmpty() || edt_to.getText().toString().isEmpty() ||

                    Toast.makeText(context, "Please fill all the fields", Toast.LENGTH_SHORT).show();
```

```
        } else {
            SimpleDateFormat sdformat = new SimpleDateFormat("dd-MM-yyyy");
            try {
                Date d1 = sdformat.parse(edt_from.getText().toString());
                Date d2 = sdformat.parse(edt_to.getText().toString());
                if (d1.compareTo(d2) > 0) {
                    Toast.makeText(context, "From date  < To date", Toast.LENGTH_SHORT).show();
                } else {
                    getTripCharts(edt_from.getText().toString(), edt_to.getText().toString());
                }
            } catch (ParseException e) {
                e.printStackTrace();
            }
        }
    }
});
edt_from.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        showDatepopupFrom();
    }
});
add_bus.setOnClickListener(new View.OnClickListener() {
    @Override
```

```
    public void onClick(View view) {
        AddBusHotelFragment addBusHotelFragment = new AddBusHotelFragment();
        Bundle bundle = new Bundle();
        bundle.putString("title", "Add Bus");
        addBusHotelFragment.setArguments(bundle);
        getActivity().getSupportFragmentManager().beginTransaction()
                .replace(R.id.container, addBusHotelFragment, "findThisFragment")
                .addToBackStack(null)
                .commit();
    }
});
add_hotel.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        AddBusHotelFragment addBusHotelFragment = new AddBusHotelFragment();
        Bundle bundle = new Bundle();
        bundle.putString("title", "Add Hotel");
        addBusHotelFragment.setArguments(bundle);
        getActivity().getSupportFragmentManager().beginTransaction()
                .replace(R.id.container, addBusHotelFragment, "findThisFragment")
                .addToBackStack(null)
                .commit();
    }
});
```

```
GetBus();
hotel_booking.setOnClickListener(new View.OnClickListener() {
    @RequiresApi(api = Build.VERSION_CODES.M)
    @Override
    public void onClick(View view) {
        lt1.setVisibility(View.GONE);
        lt2.setVisibility(View.GONE);
        button.setVisibility(View.GONE);
        SELECTED_PAGE = "hotel";
        hotel_booking.setTextColor(context.getColor(R.color.blue));
        bus_booking.setTextColor(context.getColor(R.color.text_color));
        GetHotels();
    }
});

bus_booking.setOnClickListener(new View.OnClickListener() {
    @RequiresApi(api = Build.VERSION_CODES.M)
    @Override
    public void onClick(View view) {
        lt1.setVisibility(View.VISIBLE);
        lt2.setVisibility(View.VISIBLE);
        button.setVisibility(View.VISIBLE);
        SELECTED_PAGE = "bus";
        hotel_booking.setTextColor(context.getColor(R.color.text_color));
```

```
            bus_booking.setTextColor(context.getColor(R.color.blue));
            GetBus();
        }
    });

    return rootView;
}

private void getTripCharts(String toString, String toString1) {
    try {
        custPrograssbar.progressCreate(getContext());
        custPrograssbar.setCancel(false);
        JSONObject jsonObject = new JSONObject();
        jsonObject.put("From_date", toString);
        jsonObject.put("To_Range", toString1);
        jsonObject.put("city", city.getSelectedItem().toString());
        JsonParser jsonParser = new JsonParser();
        Custom.getInterface().getTripCharts((JsonObject) jsonParser.parse(jsonObject.toString()));
        GetResult getResult = new GetResult();
        getResult.setMyListener(this);
        getResult.onNCHandle(call, "getTripCharts");
    } catch (JSONException e) {
        e.printStackTrace();
    }
```

```
}
private void showDatepopupTo() {

   DatePickerDialog.OnDateSetListener date = new DatePickerDialog.OnDateSetListener() {

      @Override
      public void onDateSet(DatePicker view, int year, int monthOfYear,
                        int dayOfMonth) {
         // TODO Auto-generated method stub
         myCalendarnew.set(Calendar.YEAR, year);
         myCalendarnew.set(Calendar.MONTH, monthOfYear);
         myCalendarnew.set(Calendar.DAY_OF_MONTH, dayOfMonth);
         updateLabelTo();
      }
   };
   DatePickerDialog dialog = new DatePickerDialog(getContext(), date, myCalendarnew
         .get(Calendar.YEAR), myCalendarnew.get(Calendar.MONTH),
         myCalendarnew.get(Calendar.DAY_OF_MONTH));
   dialog.getDatePicker().setMinDate(System.currentTimeMillis());
   dialog.show();
}
private void updateLabelTo() {
   String myFormat = "dd-MM-yyyy";
   String myFormat1 = "yyyy-MM-dd";
```

```
    SimpleDateFormat sdf = new SimpleDateFormat(myFormat, Locale.US);
    SimpleDateFormat sdf1 = new SimpleDateFormat(myFormat1, Locale.US);
    edt_to.setText(sdf.format(myCalendarnew.getTime()));
    edt_to1 = sdf1.format(myCalendarnew.getTime());
}
private void showDatepopupFrom() {
    DatePickerDialog.OnDateSetListener date = new DatePickerDialog.OnDateSetListener() {
        @Override
        public void onDateSet(DatePicker view, int year, int monthOfYear, int dayOfMonth) {
            myCalendar.set(Calendar.YEAR, year);
            myCalendar.set(Calendar.MONTH, monthOfYear);
            myCalendar.set(Calendar.DAY_OF_MONTH, dayOfMonth);
            updateLabel();
        }
    };

    DatePickerDialog dialog = new DatePickerDialog(getContext(), date, myCalendar
            .get(Calendar.YEAR), myCalendar.get(Calendar.MONTH),
            myCalendar.get(Calendar.DAY_OF_MONTH));
    dialog.getDatePicker().setMinDate(System.currentTimeMillis());
    dialog.show();
}
private void updateLabel() {
```

```
    String myFormat = "dd-MM-yyyy";
    SimpleDateFormat sdf = new SimpleDateFormat(myFormat, Locale.US);


    edt_from.setText(sdf.format(myCalendar.getTime()));
}
private void GetBus() {
    rv_data.setVisibility(View.GONE);
}
private void GetHotels() {
    Call<String> call = apiInterface.getHotel();
    call.enqueue(new Callback<String>() {
        @Override
        public void onResponse(Call<String> call, Response<String> response) {
            if (response.code() == 200) {
                try {
                    JSONObject data = new JSONObject(response.body());
                    if (data.getJSONArray("data").length() > 0) {
                        HotelAdapter adapter = new HotelAdapter(context, data.getJSONArray("data"));


                        rv_data.setLayoutManager(new LinearLayoutManager(context));
                        rv_data.setAdapter(adapter);
                        rv_data.setVisibility(View.VISIBLE);
                    }
                } catch (Exception e) {
```

```
                Toast.makeText(context, "No Data Found", Toast.LENGTH_SHORT).show();
            }
        }
    }

    @Override
    public void onFailure(Call<String> call, Throwable t) {
    }
  });
}

@Override
public void callback(JsonObject result, String callNo) {
  if (callNo.equalsIgnoreCase("getTripCharts")) {
    custPrograssbar.close();
    Gson gson = new Gson();
    Example example = gson.fromJson(result.toString(), Example.class);
    tripChartLists = new ArrayList<>();
    if (example.getResultData().getTripChartLists() != null) {
      tripChartLists = example.getResultData().getTripChartLists();
      TripChartAdapter adapter = new TripChartAdapter(getContext(), tripChartLists, this);
      rv_data.setLayoutManager(new LinearLayoutManager(getContext()));
      rv_data.setAdapter(adapter);
      rv_data.setVisibility(View.VISIBLE);
```

```
            adapter.notifyDataSetChanged();


        } else {

            Toast.makeText(context, "No data found", Toast.LENGTH_SHORT).show();

        }

    }

}


@Override

public void onItemClick(int position, String chk) {

    int plist = Integer.parseInt(edt_passenger.getText().toString());

    int av = Integer.parseInt(String.valueOf(chk));

    if (av >= plist) {

        startActivity(new Intent(context, RazorPayActivity.class)

                .putExtra("day", edt_passenger.getText().toString())

                .putExtra("date", tripChartLists.get(position).getTrip_date())

                .putExtra("isAction", "BUS")

                .putExtra("id", tripChartLists.get(position).getBusid()));


    } else {

        Toast.makeText(context, "Please enter Passenger to available scheme", Toast.
LENGTH_SHORT).show();


    }

}
```

```
}
```

//EditBusesByOwnerFragment.java

```
package com.dzniox.travel_mate.ui.fragments;

import android.os.Bundle;

import androidx.fragment.app.Fragment;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import com.dzniox.travel_mate.R;

public class EditBusesByOwnerFragment extends Fragment {

   private static final String ARG_PARAM1 = "param1";

   private static final String ARG_PARAM2 = "param2";

   private String mParam1;

   private String mParam2;

   public EditBusesByOwnerFragment() {

   }

   public static EditBusesByOwnerFragment newInstance(String param1, String param2) {

      EditBusesByOwnerFragment fragment = new EditBusesByOwnerFragment();

      Bundle args = new Bundle();

      args.putString(ARG_PARAM1, param1);

      args.putString(ARG_PARAM2, param2);

      fragment.setArguments(args);

      return fragment;

   }
```

```
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            mParam1 = getArguments().getString(ARG_PARAM1);
            mParam2 = getArguments().getString(ARG_PARAM2);
        }
    }
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container
    Bundle savedInstanceState) {
        return inflater.inflate(R.layout.fragment_edit_buses_by_owner, container, false);
    }
}
//EditBusFragment .java
package com.dzniox.travel_mate.ui.fragments;


import android.content.Context;
import android.os.Bundle;
import androidx.fragment.app.Fragment;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
```

```
import android.widget.Button;

import android.widget.EditText;

import android.widget.ImageView;

import com.bumptech.glide.Glide;

import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;

import com.dzniox.travel_mate.R;

public class EditBusFragment extends Fragment {

    private View rootView;

    private String desc;

    private String busname;

    private String seatno;

    private String id;

    private String image;

    private Context context;

EditText addbusname,addbownerdesc,addbownerseat_no;

ImageView gallery_view;

Button addbownerbtn_choose,addbowneradd_btn;

String image){
public EditBusFragment(Context context,String desc, String busname, String seatno, String id,

        this.context = context;

        this.desc = desc;

        this.busname = busname;

        this.seatno = seatno;

        this.id = id;

        this.image = image;
```

```
}
private static final String ARG_PARAM1 = "param1";
private static final String ARG_PARAM2 = "param2";

private String mParam1;
private String mParam2;

public EditBusFragment() {
}
public static EditBusFragment newInstance(String param1, String param2) {
   EditBusFragment fragment = new EditBusFragment();
   Bundle args = new Bundle();
   args.putString(ARG_PARAM1, param1);
   args.putString(ARG_PARAM2, param2);
   fragment.setArguments(args);
   return fragment;
}
@Override
public void onCreate(Bundle savedInstanceState) {
   super.onCreate(savedInstanceState);
   if (getArguments() != null) {
      mParam1 = getArguments().getString(ARG_PARAM1);
      mParam2 = getArguments().getString(ARG_PARAM2);
   }
```

```
    }

    @Override

    public View onCreateView(LayoutInflater inflater, ViewGroup container,


Bundle savedInstanceState) {

        Log.d("key","Here is Fragment@@@
```

```
        Log.d("key","Here is Fragment@@@
```

```
        args.putString(ARG_PARAM2, param2);
        fragment.setArguments(args);
        return fragment;
    }

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            mParam1 = getArguments().getString(ARG_PARAM1);
            mParam2 = getArguments().getString(ARG_PARAM2);
        }
    }
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        return inflater.inflate(R.layout.fragment_edit_package, container, false);
    }
}
```

/EditProfileFragment.java/

```
package com.dzniox.travel_mate.ui.fragments;
import android.content.Context;
import android.os.Bundle;
import androidx.fragment.app.Fragment;
```

```
import android.util.Log;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

import android.widget.Toast;


import com.dzniox.travel_mate.Model.Example;

import com.dzniox.travel_mate.Model.GetProfile;

import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;

import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.network.APIClient;

import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;

import com.google.gson.Gson;

import com.google.gson.JsonObject;

import com.google.gson.JsonParser;


import org.json.JSONException;

import org.json.JSONObject;


import java.util.ArrayList;
```

```
import java.util.List;


import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class EditProfileFragment extends Fragment {


   private static final String ARG_PARAM1 = "param1";
   private static final String ARG_PARAM2 = "param2";
   private String mParam1;
   private String mParam2;
   private View rootView;
   APIInterface apiInterface;
   private Button edit;
   PrefUtils prefrences;
   String ename,ephone,eemail;
   List<GetProfile> pkg;
   private EditText Name, Email, no_data, Phone, liked_spot, my_bookings, expences, mark_exp;
   public EditProfileFragment() {
   }
   public static EditProfileFragment newInstance(String param1, String param2) {
      EditProfileFragment fragment = new EditProfileFragment();
      Bundle args = new Bundle();
      args.putString(ARG_PARAM1, param1);
```

```
    args.putString(ARG_PARAM2, param2);
    fragment.setArguments(args);
    return fragment;
}
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
    prefrences = PrefUtils.getInstance(getContext());
}
@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    rootView= inflater.inflate(R.layout.fragment_edit_profile, container, false);
    apiInterface = APIClient.getClient().create(APIInterface.class);
    Name = rootView.findViewById(R.id.name);
    Email = rootView.findViewById(R.id.email);
    Phone = rootView.findViewById(R.id.phone);
    edit = rootView.findViewById(R.id.edit);
    Getprofile();
    edit.setOnClickListener(new View.OnClickListener() {
```

```
        @Override
        public void onClick(View view) {
           eemail=Email.getText().toString();
           ephone=Phone.getText().toString();
           ename=Name.getText().toString();
           if(eemail.isEmpty() || eemail.equals("")){
Toast.makeText(getContext(), "Please fill phone field name should not be null",
Toast.LENGTH_SHORT).show();
           } else if (ephone.isEmpty() || ephone.equals("")) {
Toast.makeText(getContext(), "Please fill phone field name should not be null",
Toast.LENGTH_SHORT).show();

           } else if (ename.isEmpty() || ename.equals("")) {
Toast.makeText(getContext(), "Please fill name field name should not be null",
Toast.LENGTH_SHORT).show();
           } else {
Call<String> call = apiInterface.editprofile(prefrences.getStringValue("id", ""), eemail,
 ename, ephone);
              call.enqueue(new Callback<String>() {
                 @Override
                 public void onResponse(Call<String> call, Response<String> response) {
                    if (response.code() == 200) {
Toast.makeText(getContext(), "Profile edited successfully",Toast.LENGTH_SHORT).show();
                    }
```

```
                }

                @Override

                public void onFailure(Call<String> call, Throwable t) {

                }

            });

        }

    }

    });

    return  rootView;

  }

  private void Getprofile(){

    JSONObject jsonObject = new JSONObject();



    try {

      jsonObject.put("id",prefrences.getStringValue("id", ""));

      JsonParser jsonParser = new JsonParser();

Call<JsonObject> call = ApiClientCustom.getInterface().getprofile((JsonObject)

jsonParser.parse(jsonObject.toString()));

      call.enqueue(new Callback<JsonObject>() {

        @Override

        public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {

          JsonObject object = response.body();

          Gson gson = new Gson();

          Example example = gson.fromJson(object.toString(), Example.class);
```

```
                pkg = new ArrayList<>();
                if (example.getResultData().getProfiles() != null) {
                   List<GetProfile> pakg = example.getResultData().getProfiles();
                   Log.d("key","HEAREIS THE DATA_____!!!!!!!!!!!!!!_____");
                   for (GetProfile pkg : pakg) {
                      String email=pkg.getEmail().toString();
                      String name=pkg.getName().toString();
                      String phone=pkg.getPhone().toString();
                      Log.d("key",email+"   "+name+"   "+phone);
                      Name.setText(name);
                      Email.setText(email);
                      Phone.setText(phone);
                   }
                }
             }
             @Override
             public void onFailure(Call<JsonObject> call, Throwable t) {
             }
          });
       } catch (JSONException e) {
          e.printStackTrace();
       }
    }
}
```

```
// PackageAdminFragment.java

package com.dzniox.travel_mate.ui.fragments;

import android.annotation.SuppressLint;

import android.content.Context;

import android.os.Build;

import android.os.Bundle;

import androidx.annotation.RequiresApi;

import androidx.fragment.app.Fragment;

import androidx.recyclerview.widget.LinearLayoutManager;

import androidx.recyclerview.widget.RecyclerView;

import android.util.Log;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.TextView;

import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.network.APIClient;

import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;

import com.dzniox.travel_mate.ui.adapters.BusRequestAdapter;

import org.json.JSONObject;

import retrofit2.Call;

import retrofit2.Callback;

import retrofit2.Response;
```

```
public class PackageAdminFragment extends Fragment {

    private static final String ARG_PARAM1 = "param1";

    private static final String ARG_PARAM2 = "param2";


    private String mParam1;

    private String mParam2;

    private View rootView

    private TextView Name, Email, no_data, Phone, Packages, Bookings, mark_exp;

    APIInterface apiInterface;

    private Context context;

    private RecyclerView rv_package;

    PrefUtils prefrences;

    public PackageAdminFragment() {

    }


    public static PackageAdminFragment newInstance(String param1, String param2) {

        PackageAdminFragment fragment = new PackageAdminFragment();

        Bundle args = new Bundle();

        args.putString(ARG_PARAM1, param1);

        args.putString(ARG_PARAM2, param2);

        fragment.setArguments(args);

        return fragment;

    }
```

```
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}
@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    rootView=inflater.inflate(R.layout.fragment_package_admin, container, false);


    context = getContext();
    prefrences = PrefUtils.getInstance(context);
    apiInterface = APIClient.getClient().create(APIInterface.class);


    rv_package = rootView.findViewById(R.id.rv_request);
    Packages = rootView.findViewById(R.id.packages);
    Bookings = rootView.findViewById(R.id.bookings);
    no_data = rootView.findViewById(R.id.no_data);
    Log.d("key","$$$
```

```
        @RequiresApi(api = Build.VERSION_CODES.M)
        @Override
        public void onClick(View view) {
          getBusRequestList();
          Packages.setTextColor(context.getColor(R.color.blue));
          Bookings.setTextColor(context.getColor(R.color.text_color));
        }
      });
      return rootView;
    }
    private void getBusRequestList() {
      Call<String> call = apiInterface.getBusRequest(prefrences.getStringValue("usertype", ""));
      call.enqueue(new Callback<String>() {
        @Override
        public void onResponse(Call<String> call, Response<String> response) {
          if (response.code() == 200) {
            rv_package.setVisibility(View.VISIBLE);
            try {
              JSONObject data = new JSONObject(response.body());
              Log.d("key",response.body());
Log.d("key","HEYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY");
              if (data.getJSONArray("data").length() > 0) {
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");
```

```
                    no_data.setVisibility(View.GONE);
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");
                    rv_package.setVisibility(View.VISIBLE);
Log.d("key","adStart************************************************************");
BusRequestAdapter adapter = new BusRequestAdapter(context, data.getJSONArray
("data"));
Log.d("key","adEnd**************************************************************");
                    rv_package.setLayoutManager(new LinearLayoutManager(context));
                    rv_package.setAdapter(adapter);
                  } else {
                    no_data.setVisibility(View.VISIBLE);
                    rv_package.setVisibility(View.GONE);
                  }
                } catch (Exception e) {
                  no_data.setVisibility(View.VISIBLE);
                  rv_package.setVisibility(View.GONE);
                }
              } else {
                no_data.setVisibility(View.VISIBLE);
                rv_package.setVisibility(View.GONE);
              }
            }
            @Override
```

```
        public void onFailure(Call<String> call, Throwable t) {
            no_data.setVisibility(View.VISIBLE);
            rv_package.setVisibility(View.GONE);
        }
    });
  }
}
```

//ProfileFragment.java

```
package com.dzniox.travel_mate.ui.fragments;

import android.annotation.SuppressLint;
import android.content.Context;
import android.os.Build;
import android.os.Bundle;

import androidx.annotation.RequiresApi;
import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
```

```
import android.widget.Button;

import android.widget.EditText;

import android.widget.LinearLayout;

import android.widget.TextView;

import android.widget.Toast;


import com.bumptech.glide.Glide;

import com.dzniox.travel_mate.Model.Example;

import com.dzniox.travel_mate.Model.GetPackageData;

import com.dzniox.travel_mate.Model.GetProfile;

import com.dzniox.travel_mate.NetworkUtilsCustom.ApiClientCustom;

import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.network.APIClient;

import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;

import com.dzniox.travel_mate.ui.adapters.BookingAdapter;

import com.dzniox.travel_mate.ui.adapters.BusAssignedAdapter;

import com.dzniox.travel_mate.ui.adapters.ExpenseAdapter;

import com.dzniox.travel_mate.ui.adapters.HotelAssignedAdapter;

import com.dzniox.travel_mate.ui.adapters.HotelBookingsAdapter;

import com.dzniox.travel_mate.ui.adapters.SpotAdapter;

import com.google.gson.Gson;

import com.google.gson.JsonObject;

import com.google.gson.JsonParser;
```

```
import org.json.JSONException;

import org.json.JSONObject;


import java.util.ArrayList;

import java.util.List;


import retrofit2.Call;

import retrofit2.Callback;

import retrofit2.Response;




public class ProfileFragment extends Fragment {

    private static final String ARG_PARAM1 = "param1";

    private static final String ARG_PARAM2 = "param2";


    private String mParam1;

    private String mParam2;


    private TextView Name, Email, no_data, Phone, liked_spot, my_bookings, assignedpkg,
dlname,lname,desc;

    private  Button Edit;

    APIInterface apiInterface;

    private Context context;

    private View rootView;
```

```
    private RecyclerView rv_hbdata;
    List<GetProfile> pkg;
    PrefUtils prefrences;
    LinearLayout lin;
ProfileFragment   binding;
    public ProfileFragment() {
    }
    public static ProfileFragment newInstance(String param1, String param2) {
        ProfileFragment fragment = new ProfileFragment();
        Bundle args = new Bundle();
        args.putString(ARG_PARAM1, param1);
        args.putString(ARG_PARAM2, param2);
        fragment.setArguments(args);
        return fragment;
    }
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            mParam1 = getArguments().getString(ARG_PARAM1);
            mParam2 = getArguments().getString(ARG_PARAM2);
        }
    }
    @Override
```

```
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    rootView=inflater.inflate(R.layout.fragment_profile, container, false);
    context = getContext();
    prefrences = PrefUtils.getInstance(context);
    apiInterface = APIClient.getClient().create(APIInterface.class);


    Name = rootView.findViewById(R.id.name);
    assignedpkg = rootView.findViewById(R.id.apkg);
    Email = rootView.findViewById(R.id.email);
    Phone = rootView.findViewById(R.id.phone);
    rv_hbdata = rootView.findViewById(R.id.rv_hbdata);
    my_bookings = rootView.findViewById(R.id.my_bookings);
    no_data = rootView.findViewById(R.id.no_data);
    hname = rootView.findViewById(R.id.hname);
    Edit = rootView.findViewById(R.id.Editprofile);
    dlaccount = rootView.findViewById(R.id.dlaccount);
    lin = rootView.findViewById(R.id.li);
    Getprofile();
    if (prefrences.getStringValue("usertype","").equalsIgnoreCase("0")){
       lin.setVisibility(View.GONE);


    }
    Edit.setOnClickListener(new View.OnClickListener() {
```

```
        @Override
        public void onClick(View view) {
            EditProfileFragment editprofile = new EditProfileFragment();
            getActivity().getSupportFragmentManager().beginTransaction()
                    .replace(R.id.container, editprofile, "findThisFragment")
                    .addToBackStack(null)
                    .commit();
        }
    });
    dlaccount.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
Call<String> call = apiInterface.deleteaccount(prefrences.getStringValue("id", ""), prefrences
.getStringValue("usertype", ""));
            call.enqueue(new Callback<String>() {
                @Override
                public void onResponse(Call<String> call, Response<String> response) {
                    if (response.code() == 200) {

Toast.LENGTH_SHORT).show();Toast.makeText(getContext(), "Profile Deleted successfully",

                    }
                }
                @Override
                public void onFailure(Call<String> call, Throwable t) {
```

```
            }
        });
    }
});

if (prefrences.getStringValue("usertype","").equalsIgnoreCase("2")) {
    getMybusBookings();
}else if(prefrences.getStringValue("usertype","").equalsIgnoreCase("3")){
    getMyHotelBookings();
}


my_bookings.setOnClickListener(new View.OnClickListener() {
    @SuppressLint("NewApi")
    @Override
    public void onClick(View view) {
        if (prefrences.getStringValue("usertype","").equalsIgnoreCase("2")) {
            getMybusBookings();
        }else if(prefrences.getStringValue("usertype","").equalsIgnoreCase("3")){
            getMyHotelBookings();
        }

    }
});
```

```
    assignedpkg.setOnClickListener(new View.OnClickListener() {
        @SuppressLint("NewApi")
        @Override
        public void onClick(View view) {
            if (prefrences.getStringValue("usertype","").equalsIgnoreCase("2")) {
                BusAssignedpackages();
            }else if(prefrences.getStringValue("usertype","").equalsIgnoreCase("3")){
                HotelAssignedpackages();
            }

        }
    });

    return rootView;
}

private void  getMyHotelBookings() {
    Call<String> call = apiInterface.gethotelbookings(prefrences.getStringValue("id", ""));
    call.enqueue(new Callback<String>() {
        @Override
        public void onResponse(Call<String> call, Response<String> response) {
            if (response.code() == 200) {
                rv_hbdata.setVisibility(View.VISIBLE);
                try {
```

```
                JSONObject data = new JSONObject(response.body());


                if (data.getJSONArray("data").length() > 0) {
                    no_data.setVisibility(View.GONE);
                    rv_hbdata.setVisibility(View.VISIBLE);
                    HotelBookingsAdapter adapter = new HotelBookingsAdapter(context,
data.getJSONArray("data"));
                    rv_hbdata.setLayoutManager(new LinearLayoutManager(context));
                    rv_hbdata.setAdapter(adapter);
                } else {
                    no_data.setVisibility(View.VISIBLE);
                    rv_hbdata.setVisibility(View.GONE);
                }
            } catch (Exception e) {
                no_data.setVisibility(View.VISIBLE);
                rv_hbdata.setVisibility(View.GONE);
            }


        } else {
            no_data.setVisibility(View.VISIBLE);
            rv_hbdata.setVisibility(View.GONE);
        }
    }


    @Override
```

```
        public void onFailure(Call<String> call, Throwable t) {
            no_data.setVisibility(View.VISIBLE);
            rv_hbdata.setVisibility(View.GONE);
        }
    });
}

private void getMybusBookings() {
    Call<String> call = apiInterface.getMyBookings(prefrences.getStringValue("id", ""));
    call.enqueue(new Callback<String>() {
        @Override
        public void onResponse(Call<String> call, Response<String> response) {
            if (response.code() == 200) {
                rv_hbdata.setVisibility(View.VISIBLE);
                try {
                    JSONObject data = new JSONObject(response.body());

                    if (data.getJSONArray("data").length() > 0) {
                        no_data.setVisibility(View.GONE);
                        rv_hbdata.setVisibility(View.VISIBLE);
data.getJSONArray("data"));                        ExpenseAdapter adapter = new ExpenseAdapter(context,
                        rv_hbdata.setLayoutManager(new LinearLayoutManager(context));
                        rv_hbdata.setAdapter(adapter);
                    } else {
```

```
                no_data.setVisibility(View.VISIBLE);
                rv_hbdata.setVisibility(View.GONE);
              }
            } catch (Exception e) {
              no_data.setVisibility(View.VISIBLE);
              rv_hbdata.setVisibility(View.GONE);
            }

          } else {
            no_data.setVisibility(View.VISIBLE);
            rv_hbdata.setVisibility(View.GONE);
          }
        }

        @Override
        public void onFailure(Call<String> call, Throwable t) {
          no_data.setVisibility(View.VISIBLE);
          rv_hbdata.setVisibility(View.GONE);
        }
      });
    }
```

```
    private void Getprofile(){

        JSONObject jsonObject = new JSONObject();


        try {

            jsonObject.put("id",prefrences.getStringValue("id", ""));

            JsonParser jsonParser = new JsonParser();

Call<JsonObject> call = ApiClientCustom.getInterface().getprofile((JsonObject) jsonParser

.parse(jsonObject.toString()));

            call.enqueue(new Callback<JsonObject>() {

                @Override

                public void onResponse(Call<JsonObject> call, Response<JsonObject> response) {

                    JsonObject object = response.body();

                    Gson gson = new Gson();

                    Example example = gson.fromJson(object.toString(), Example.class);

                    pkg = new ArrayList<>();

                    if (example.getResultData().getProfiles() != null) {

                        List<GetProfile> pakg = example.getResultData().getProfiles();

                        Log.d("key","HEAREIS THE DATA_____!!!!!!!!!!!!!!_____");

                        for (GetProfile pkg : pakg) {

                            String email=pkg.getEmail().toString();

                            String name=pkg.getName().toString();

                            String phone=pkg.getPhone().toString();

                            Log.d("key",email+"   "+name+"   "+phone);

                            if (prefrences.getStringValue("usertype","").equalsIgnoreCase("3")) {
```

```
                String hotel=pkg.getHotel().toString();
                hname.setVisibility(View.VISIBLE);
                hname.setText(hotel);
                Name.setText(name);
                Email.setText(email);
                Phone.setText(phone);

            }else{
                Name.setText(name);
                Email.setText(email);
                Phone.setText(phone);
            }
          }
        }

      @Override
      public void onFailure(Call<JsonObject> call, Throwable t) {

      }
    });
  } catch (JSONException e) {
    e.printStackTrace();
  }
```

```
}



private void BusAssignedpackages() {
   try {
      Call<String> call = apiInterface.busassignedtrip(prefrences.getStringValue("id", ""));
      call.enqueue(new Callback<String>() {
         @Override
         public void onResponse(Call<String> call, Response<String> response) {
            if (response.code() == 200) {
               rv_hbdata.setVisibility(View.VISIBLE);
               try {
                  JSONObject data = new JSONObject(response.body());
                  if (data.getJSONArray("data").length() > 0) {
                     no_data.setVisibility(View.GONE);
                     rv_hbdata.setVisibility(View.VISIBLE);
                     BusAssignedAdapter adapter = new BusAssignedAdapter(context,
                           data.getJSONArray("data"));
                     rv_hbdata.setLayoutManager(new LinearLayoutManager(context));
                     rv_hbdata.setAdapter(adapter);
                  }
               } catch (Exception e) {
                  no_data.setVisibility(View.VISIBLE);
                  rv_hbdata.setVisibility(View.GONE);
```

```
                }
            } else {
                no_data.setVisibility(View.VISIBLE);
                rv_hbdata.setVisibility(View.GONE);
            }
        }

        @Override
        public void onFailure(Call<String> call, Throwable t) {
            no_data.setVisibility(View.VISIBLE);
            rv_hbdata.setVisibility(View.GONE);
        }
    });
  } catch (Exception e) {
    rv_hbdata.setVisibility(View.GONE);
    Toast.makeText(requireContext(), e.toString(), Toast.LENGTH_SHORT).show();
  }
}
private void HotelAssignedpackages() {
  try {
    Call<String> call = apiInterface.hotelassignedtrip(prefrences.getStringValue("id", ""));
    call.enqueue(new Callback<String>() {
      @Override
      public void onResponse(Call<String> call, Response<String> response) {
```

```
            if (response.code() == 200) {
               rv_hbdata.setVisibility(View.VISIBLE);
               try {
                  JSONObject data = new JSONObject(response.body());
                  if (data.getJSONArray("data").length() > 0) {
                     no_data.setVisibility(View.GONE);
                     rv_hbdata.setVisibility(View.VISIBLE);
ExpenseAdapter adapter = new ExpenseAdapter(context, data.getJSONArray
          ("data"));
                     rv_hbdata.setLayoutManager(new LinearLayoutManager(context));
                     rv_hbdata.setAdapter(adapter);
                  }
               } catch (Exception e) {
                  no_data.setVisibility(View.VISIBLE);
                  rv_hbdata.setVisibility(View.GONE);
               }
            } else {
               no_data.setVisibility(View.VISIBLE);
               rv_hbdata.setVisibility(View.GONE);
            }
         }
         @Override
         public void onFailure(Call<String> call, Throwable t) {
            no_data.setVisibility(View.VISIBLE);
```

```
                rv_hbdata.setVisibility(View.GONE);
            }
        });
    } catch (Exception e) {
        rv_hbdata.setVisibility(View.GONE);
        Toast.makeText(requireContext(), e.toString(), Toast.LENGTH_SHORT).show();
    }
  }
}
//Rejection_History_Fragment .java
package com.dzniox.travel_mate.ui.fragments;

import android.annotation.SuppressLint;
import android.content.Context;
import android.content.Intent;
import android.os.Build;
import android.os.Bundle;

import androidx.annotation.RequiresApi;
import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.util.Log;
```

```
import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.EditText;

import android.widget.TextView;

import android.widget.Toast;


import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.network.APIClient;

import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;

import com.dzniox.travel_mate.ui.adapters.BookingAdapter;

import com.dzniox.travel_mate.ui.adapters.BusRequestAdapter;

import com.dzniox.travel_mate.ui.adapters.Bus_History_Adapter;

import com.dzniox.travel_mate.ui.adapters.HotelRequestAdapter;

import com.dzniox.travel_mate.ui.adapters.ExpenseAdapter;

import com.dzniox.travel_mate.ui.adapters.Hotel_History_Adapter;

import com.dzniox.travel_mate.ui.adapters.SpotAdapter;


import org.json.JSONObject;


import retrofit2.Call;

import retrofit2.Callback;

import retrofit2.Response;
```

```
public class Rejection_History_Fragment extends Fragment {


    private static final String ARG_PARAM1 = "param1";

    private static final String ARG_PARAM2 = "param2";

    private String mParam1;

    private String mParam2;


    private TextView Name, Email, no_data, Phone, bus_Owners, hotel_Owners, history;

    APIInterface apiInterface;

    private Context context;

    private View rootView;

    private RecyclerView rv_request;


    PrefUtils prefrences;


    public Rejection_History_Fragment() {

    }



    public static UsersRequestFragment newInstance(String param1, String param2) {

        UsersRequestFragment fragment = new UsersRequestFragment();

        Bundle args = new Bundle();

        args.putString(ARG_PARAM1, param1);
```

```
    args.putString(ARG_PARAM2, param2);
    fragment.setArguments(args);
    return fragment;
}

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    rootView = inflater.inflate(R.layout.fragment_rejection__history_, container, false);
    context = getContext();
    prefrences = PrefUtils.getInstance(context);
    apiInterface = APIClient.getClient().create(APIInterface.class);

    rv_request = rootView.findViewById(R.id.rv_history);
    bus_Owners = rootView.findViewById(R.id.Bus_Owners);
```

```
hotel_Owners = rootView.findViewById(R.id.Hotel_Owners);
no_data = rootView.findViewById(R.id.no_data);



Log.d("key","$$$
```

```
            bus_Owners.setTextColor(context.getColor(R.color.text_color));


        }
    });
    return rootView;
}
private void getBusRequestList() {


    Call<String> call = apiInterface.getbushistory();


    call.enqueue(new Callback<String>() {
        @Override
        public void onResponse(Call<String> call, Response<String> response) {
            if (response.code() == 200) {
                rv_request.setVisibility(View.VISIBLE);
                try {
                    JSONObject data = new JSONObject(response.body());
                    Log.d("key",response.body());
Log.d("key","HEYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY");
                    if (data.getJSONArray("data").length() > 0) {
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");


                        no_data.setVisibility(View.GONE);
```

```
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");
                    rv_request.setVisibility(View.VISIBLE);
                    Log.d("key","adStart*********************************************");

                    Bus_History_Adapter adapter = new Bus_History_Adapter(context,
                          data.getJSONArray("data"));
                    Log.d("key","adEnd***********************************************");
                    rv_request.setLayoutManager(new LinearLayoutManager(context));
                    rv_request.setAdapter(adapter);
                  } else {
                    no_data.setVisibility(View.VISIBLE);
                    rv_request.setVisibility(View.GONE);
                  }
                } catch (Exception e) {
                  no_data.setVisibility(View.VISIBLE);
                  rv_request.setVisibility(View.GONE);
                }

              } else {
                no_data.setVisibility(View.VISIBLE);
                rv_request.setVisibility(View.GONE);
              }
            }
```

```
        @Override
        public void onFailure(Call<String> call, Throwable t) {
            no_data.setVisibility(View.VISIBLE);
            rv_request.setVisibility(View.GONE);
        }
    });
  }

  private void getHotelRequestList() {
    try {
        Call<String> call = apiInterface.gethotelhistory();
        call.enqueue(new Callback<String>() {
            @Override
            public void onResponse(Call<String> call, Response<String> response) {
                if (response.code() == 200) {
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK1");
                    rv_request.setVisibility(View.VISIBLE);
                    try {

                        JSONObject data = new JSONObject(response.body());
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK2");
                        if (data.getJSONArray("data").length() > 0) {
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK3");
```

```
                    no_data.setVisibility(View.GONE);

                    rv_request.setVisibility(View.VISIBLE);

                   Hotel_History_Adapter adapter = new Hotel_History_Adapter(context,

                            data.getJSONArray("data"));

                    rv_request.setLayoutManager(new LinearLayoutManager(context));

                    rv_request.setAdapter(adapter);

                  }else {

                    Log.d("key","!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!4");


                    no_data.setVisibility(View.VISIBLE);

                    rv_request.setVisibility(View.GONE);

                  }

                } catch (Exception e) {

Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK5");

                    no_data.setVisibility(View.VISIBLE);

                    rv_request.setVisibility(View.GONE);

                  }

              } else {

Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK6");


                  no_data.setVisibility(View.VISIBLE);

                  rv_request.setVisibility(View.GONE);

                }

            }
```

```
            @Override
            public void onFailure(Call<String> call, Throwable t) {
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK");

                no_data.setVisibility(View.VISIBLE);
                rv_request.setVisibility(View.GONE);
            }
        });
    } catch (Exception e) {
        rv_request.setVisibility(View.GONE);
        Toast.makeText(requireContext(), e.toString(), Toast.LENGTH_SHORT).show();
    }
  }
}
//ReviewFragment .java
package com.dzniox.travel_mate.ui.fragments;

import android.content.Context;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
```

```
import android.widget.EditText;

import android.widget.ImageView;

import android.widget.Toast;


import androidx.cardview.widget.CardView;

import androidx.fragment.app.Fragment;

import androidx.recyclerview.widget.LinearLayoutManager;

import androidx.recyclerview.widget.RecyclerView;

import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.network.APIClient;

import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;

import com.dzniox.travel_mate.ui.adapters.ReviewsAdapter;

import org.json.JSONObject;

import retrofit2.Call;

import retrofit2.Callback;

import retrofit2.Response;


public class ReviewFragment extends Fragment {


    private static final String ARG_PARAM1 = "param1";

    private static final String ARG_PARAM2 = "param2";

    private String mParam1;

    private String mParam2;
```

```
private View rootView;

private Context context;


APIInterface apiInterface;

PrefUtils prefrences;

private RecyclerView rv_ideas;

private CardView card_add;


private Button add_btn,cancel_btn;

private EditText title;

private ImageView add_review;


public ReviewFragment() {

}


public static ReviewFragment newInstance(String param1, String param2) {

   ReviewFragment fragment = new ReviewFragment();

   Bundle args = new Bundle();

   args.putString(ARG_PARAM1, param1);

   args.putString(ARG_PARAM2, param2);

   fragment.setArguments(args);

   return fragment;

}

@Override
```

```
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}
@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    rootView = inflater.inflate(R.layout.fragment_review, container, false);
    context = getContext();


    apiInterface = APIClient.getClient().create(APIInterface.class);
    prefrences = PrefUtils.getInstance(context);


    rv_ideas = rootView.findViewById(R.id.rv_ideas);
    add_btn = rootView.findViewById(R.id.add_btn);
    cancel_btn = rootView.findViewById(R.id.cancel_btn);
    title = rootView.findViewById(R.id.title);
    add_review = rootView.findViewById(R.id.add_review);
    card_add = rootView.findViewById(R.id.card_add_ideas);
    add_review.setOnClickListener(view -> card_add.setVisibility(View.VISIBLE));
    cancel_btn.setOnClickListener(view -> card_add.setVisibility(View.GONE));
```

```
    GetReview();
    add_btn.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View view) {
        if (title.getText().toString().trim().equals("")) {
          Toast.makeText(context,"Fill All the fields",Toast.LENGTH_SHORT).show();
        }else {
          AddReview(title.getText().toString().trim());
        }
      }
    });

    return rootView;
  }

  private void GetReview() {
    try {
      Call<String> call = apiInterface.getReviews();
      call.enqueue(new Callback<String>() {
        @Override
        public void onResponse(Call<String> call, Response<String> response) {
          if (response.code() == 200){
            try {
              JSONObject data = new JSONObject(response.body());
```

```
                if (data.getJSONArray("data").length() > 0){
                    ReviewsAdapter adapter = new ReviewsAdapter(context,
                            data.getJSONArray("data"));
                    rv_ideas.setLayoutManager(new LinearLayoutManager(context));
                    rv_ideas.setAdapter(adapter);
                }
            }catch (Exception e){}
        }
    }
    @Override
    public void onFailure(Call<String> call, Throwable t) {
        String a = t.toString();
    }
  });
 }catch (Exception e){
    Toast.makeText(context,e.toString(),Toast.LENGTH_SHORT).show();
 }
}


private void AddReview(String review) {
 try {
    Call<String> call = apiInterface.insertReview(review,
            prefrences.getStringValue("id",""));
    call.enqueue(new Callback<String>() {
```

```
        @Override
        public void onResponse(Call<String> call, Response<String> response) {
          if (response.code() == 200){
            try {
              JSONObject data = new JSONObject(response.body());
              if (data.getBoolean("data")){
                card_add.setVisibility(View.GONE);
                Toast.makeText(context,"Review Added",Toast.LENGTH_SHORT).show();
                GetReview();
              }else {
                Toast.makeText(context,"Try Again",Toast.LENGTH_SHORT).show();
              }
            }catch (Exception e){


            }
          }
        }
        @Override
        public void onFailure(Call<String> call, Throwable t) {
          String a = t.toString();
        }
      });
    }catch (Exception e){
      Toast.makeText(context,e.toString(),Toast.LENGTH_SHORT).show();
```

```
        }
    }
}
//UsersRequestFragment .java
package com.dzniox.travel_mate.ui.fragments;


import android.annotation.SuppressLint;
import android.content.Context;
import android.content.Intent;
import android.os.Build;
import android.os.Bundle;


import androidx.annotation.RequiresApi;
import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;


import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
```

```
import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.network.APIClient;

import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;

import com.dzniox.travel_mate.ui.adapters.BookingAdapter;

import com.dzniox.travel_mate.ui.adapters.BusRequestAdapter;

import com.dzniox.travel_mate.ui.adapters.HotelRequestAdapter;

import com.dzniox.travel_mate.ui.adapters.ExpenseAdapter;

import com.dzniox.travel_mate.ui.adapters.SpotAdapter;


import org.json.JSONObject;

import retrofit2.Call;

import retrofit2.Callback;

import retrofit2.Response;

public class UsersRequestFragment extends Fragment {

    private static final String ARG_PARAM1 = "param1";

    private static final String ARG_PARAM2 = "param2";


    private String mParam1;

    private String mParam2;


    private TextView Name, Email, no_data, Phone, bus_Owners, hotel_Owners, history;
```

```
APIInterface apiInterface;
private Context context;
private View rootView;
private RecyclerView rv_request;
PrefUtils prefrences;
public UsersRequestFragment() {
}
public static UsersRequestFragment newInstance(String param1, String param2) {
    UsersRequestFragment fragment = new UsersRequestFragment();
    Bundle args = new Bundle();
    args.putString(ARG_PARAM1, param1);
    args.putString(ARG_PARAM2, param2);
    fragment.setArguments(args);
    return fragment;
}

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}
```

```
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {


        rootView = inflater.inflate(R.layout.fragment_users_request, container, false);
        context = getContext();
        prefrences = PrefUtils.getInstance(context);
        apiInterface = APIClient.getClient().create(APIInterface.class);


        rv_request = rootView.findViewById(R.id.rv_request);
        bus_Owners = rootView.findViewById(R.id.Bus_Owners);
        hotel_Owners = rootView.findViewById(R.id.Hotel_Owners);
        no_data = rootView.findViewById(R.id.no_data);
        history = rootView.findViewById(R.id.History);


Log.d("key","$$$
```

```
        bus_Owners.setTextColor(context.getColor(R.color.blue));
        hotel_Owners.setTextColor(context.getColor(R.color.text_color));

    }
});
hotel_Owners.setOnClickListener(new View.OnClickListener() {
    @SuppressLint("NewApi")
    @Override
    public void onClick(View view) {
        getHotelRequestList();
        hotel_Owners.setTextColor(context.getColor(R.color.blue));
        bus_Owners.setTextColor(context.getColor(R.color.text_color));

    }
});
history.setOnClickListener(new View.OnClickListener() {
    @SuppressLint("NewApi")
    @Override
    public void onClick(View view) {
        Rejection_History_Fragment newFragment = new Rejection_History_Fragment();
        getActivity().getSupportFragmentManager().beginTransaction()
                .replace(R.id.container, newFragment)
                .addToBackStack(null)
                .commit();
```

```
        }
    });
    return rootView;
}
private void getBusRequestList() {

    Call<String> call = apiInterface.getBusRequest(prefrences.getStringValue("usertype", ""));

    call.enqueue(new Callback<String>() {
        @Override
        public void onResponse(Call<String> call, Response<String> response) {
            if (response.code() == 200) {
                rv_request.setVisibility(View.VISIBLE);
                try {
                    JSONObject data = new JSONObject(response.body());
                    Log.d("key",response.body());
Log.d("key","HEYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY");
                    if (data.getJSONArray("data").length() > 0) {
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");

                        no_data.setVisibility(View.GONE);
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");
```

```
                    rv_request.setVisibility(View.VISIBLE);

Log.d("key","adStart*****************************************************");


                    BusRequestAdapter adapter = new BusRequestAdapter(context,

                            data.getJSONArray("data"));

Log.d("key","adEnd*******************************************************");
                    rv_request.setLayoutManager(new LinearLayoutManager(context));

                    rv_request.setAdapter(adapter);

                } else {

                    no_data.setVisibility(View.VISIBLE);

                    rv_request.setVisibility(View.GONE);

                }

            } catch (Exception e) {

                no_data.setVisibility(View.VISIBLE);

                rv_request.setVisibility(View.GONE);

            }


        } else {

            no_data.setVisibility(View.VISIBLE);

            rv_request.setVisibility(View.GONE);

        }

    }


    @Override
```

```
        public void onFailure(Call<String> call, Throwable t) {
            no_data.setVisibility(View.VISIBLE);
            rv_request.setVisibility(View.GONE);
        }
    });
}

private void getHotelRequestList() {
    try {
        Call<String> call = apiInterface.getHotelRequest(prefrences.getStringValue("id", ""));
        call.enqueue(new Callback<String>() {
            @Override
            public void onResponse(Call<String> call, Response<String> response) {
                if (response.code() == 200) {
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK1");
                    rv_request.setVisibility(View.VISIBLE);
                    try {

                        JSONObject data = new JSONObject(response.body());
Log.d(KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK2");
                        if (data.getJSONArray("data").length() > 0) {
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK3");

                            no_data.setVisibility(View.GONE);
```

```
                    rv_request.setVisibility(View.VISIBLE);
                    HotelRequestAdapter adapter = new HotelRequestAdapter(context,
                            data.getJSONArray("data"));
                    rv_request.setLayoutManager(new LinearLayoutManager(context));
                    rv_request.setAdapter(adapter);
                }else {
                    Log.d("key","!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!4");

                    no_data.setVisibility(View.VISIBLE);
                    rv_request.setVisibility(View.GONE);
                }
            } catch (Exception e) {
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK5");

                    no_data.setVisibility(View.VISIBLE);
                    rv_request.setVisibility(View.GONE);
                }
            } else {
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK6");

                no_data.setVisibility(View.VISIBLE);
                rv_request.setVisibility(View.GONE);
            }
        }
```

```
            @Override
            public void onFailure(Call<String> call, Throwable t) {
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK");

                no_data.setVisibility(View.VISIBLE);
                rv_request.setVisibility(View.GONE);
            }
        });
    } catch (Exception e) {
        rv_request.setVisibility(View.GONE);
        Toast.makeText(requireContext(), e.toString(), Toast.LENGTH_SHORT).show();
    }
  }
}
```

//UsersViewFragment .java

```
package com.dzniox.travel_mate.ui.fragments;

import android.annotation.SuppressLint;
import android.content.Context;
import android.os.Build;
import android.os.Bundle;
import androidx.annotation.RequiresApi;
import androidx.fragment.app.Fragment;
```

```
import androidx.recyclerview.widget.LinearLayoutManager;

import androidx.recyclerview.widget.RecyclerView;

import android.util.Log;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.EditText;

import android.widget.TextView;

import android.widget.Toast;

import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.network.APIClient;

import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;

import com.dzniox.travel_mate.ui.adapters.BookingAdapter;

import com.dzniox.travel_mate.ui.adapters.BusOwnersListAdapter;

import com.dzniox.travel_mate.ui.adapters.BusRequestAdapter;

import com.dzniox.travel_mate.ui.adapters.HotelOwnersListAdapter;

import com.dzniox.travel_mate.ui.adapters.HotelRequestAdapter;

import com.dzniox.travel_mate.ui.adapters.ExpenseAdapter;

import com.dzniox.travel_mate.ui.adapters.SpotAdapter;

import com.dzniox.travel_mate.ui.adapters.UsersListAdapter;

import org.json.JSONArray;

import org.json.JSONObject;

import retrofit2.Call;
```

```
import retrofit2.Callback;

import retrofit2.Response;

public class UsersViewFragment extends Fragment {

   private static final String ARG_PARAM1 = "param1";

   private static final String ARG_PARAM2 = "param2";

   private String mParam1;

   private String mParam2;

   private TextView Name, Email, no_data, Phone, bus_Owners, hotel_Owners, users;

   APIInterface apiInterface;

   private Context context;

   private View rootView;

   private RecyclerView rv_users;

   PrefUtils prefrences;

   public UsersViewFragment() {

   }

   public static UsersRequestFragment newInstance(String param1, String param2) {

      UsersRequestFragment fragment = new UsersRequestFragment();

      Bundle args = new Bundle();

      args.putString(ARG_PARAM1, param1);

      args.putString(ARG_PARAM2, param2);

      fragment.setArguments(args);

      return fragment;

   }
```

```
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    rootView = inflater.inflate(R.layout.fragment_users_view, container, false);
    context = getContext();
    prefrences = PrefUtils.getInstance(context);
    apiInterface = APIClient.getClient().create(APIInterface.class);

    rv_users = rootView.findViewById(R.id.rv_users);
    bus_Owners = rootView.findViewById(R.id.Bus_Owners);
    hotel_Owners = rootView.findViewById(R.id.Hotel_Owners);
    no_data = rootView.findViewById(R.id.no_data);
    users= rootView.findViewById(R.id.Users);
    getBusOwnersList();
```

```
bus_Owners.setOnClickListener(new View.OnClickListener() {
    @RequiresApi(api = Build.VERSION_CODES.M)
    @Override
    public void onClick(View view) {
        getBusOwnersList();
        Log.d("key","heloooooooooooooooooooooooooooooohiiiiiiiiiiiiiiiiiiiiiii");
        bus_Owners.setTextColor(context.getColor(R.color.blue));
        hotel_Owners.setTextColor(context.getColor(R.color.text_color));
        users.setTextColor(context.getColor(R.color.text_color));

    }
});
hotel_Owners.setOnClickListener(new View.OnClickListener() {
    @SuppressLint("NewApi")
    @Override
    public void onClick(View view) {
        getHotelOwnersList();
        hotel_Owners.setTextColor(context.getColor(R.color.blue));
        bus_Owners.setTextColor(context.getColor(R.color.text_color));
        users.setTextColor(context.getColor(R.color.text_color));

    }
});
users.setOnClickListener(new View.OnClickListener() {
```

```
        @SuppressLint("NewApi")
        @Override
        public void onClick(View view) {
            getUsersList();
            hotel_Owners.setTextColor(context.getColor(R.color.text_color));
            bus_Owners.setTextColor(context.getColor(R.color.text_color));
            users.setTextColor(context.getColor(R.color.blue));

        }
    });
    return rootView;
}
private void getBusOwnersList() {

                Call<String> call = apiInterface.getBusOwners(prefrences
                    .getStringValue("usertype", ""));

                call.enqueue(new Callback<String>() {
                    @Override
                    public void onResponse(Call<String> call, Response<String> response) {
                        if (response.code() == 200) {
                            rv_users.setVisibility(View.VISIBLE);
                            try {
                                JSONObject data = new JSONObject(response.body());
```

```
JSONArray dataArray = data.getJSONArray("data");
if (dataArray.length() > 0) {

   JSONObject firstPerson = dataArray.getJSONObject(0);



   String firstName = firstPerson.getString("name");



   System.out.println("Name of the first person: " + firstName);
}
Log.d("key",response.body());
Log.d("key","HEYYYYYYYYYYYYYYYYYYYYYYYYYYYYY");
if (data.getJSONArray("data").length() > 0) {
   no_data.setVisibility(View.GONE);
   rv_users.setVisibility(View.VISIBLE);
   Log.d("key","adStart**********************");
   BusOwnersListAdapter adapter = new BusOwnersListAdapter
            (context, data.getJSONArray("data"));
   Log.d("key","adEnd******************************");
   rv_users.setLayoutManager(new LinearLayoutManager(context));
   rv_users.setAdapter(adapter);
} else {
   no_data.setVisibility(View.VISIBLE);
```

```
                    rv_users.setVisibility(View.GONE);
                }
            } catch (Exception e) {
                no_data.setVisibility(View.VISIBLE);
                rv_users.setVisibility(View.GONE);
            }

        } else {
            no_data.setVisibility(View.VISIBLE);
            rv_users.setVisibility(View.GONE);
        }
    }

    @Override
    public void onFailure(Call<String> call, Throwable t) {
        no_data.setVisibility(View.VISIBLE);
        rv_users.setVisibility(View.GONE);
    }
  });
}

private void getHotelOwnersList() {
  try {
    Call<String> call = apiInterface.getHotelOwners(prefrences.getStringValue("id", ""));
```

```
        call.enqueue(new Callback<String>() {
            @Override
            public void onResponse(Call<String> call, Response<String> response) {
                if (response.code() == 200) {
                    rv_users.setVisibility(View.VISIBLE);
                    try {
                        JSONObject data = new JSONObject(response.body());
                        if (data.getJSONArray("data").length() > 0) {
                            no_data.setVisibility(View.GONE);
                            rv_users.setVisibility(View.VISIBLE);
                            HotelOwnersListAdapter adapter = new HotelOwnersListAdapter(context,
                                    data.getJSONArray("data"));
                            rv_users.setLayoutManager(new LinearLayoutManager(context));
                            rv_users.setAdapter(adapter);
                        }else {
                            Log.d("key","!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!4");


                            no_data.setVisibility(View.VISIBLE);
                            rv_users.setVisibility(View.GONE);
                        }
                    } catch (Exception e) {
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK5");


                        no_data.setVisibility(View.VISIBLE);
```

```
                    rv_users.setVisibility(View.GONE);
                }
            } else {
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK6");
                no_data.setVisibility(View.VISIBLE);
                rv_users.setVisibility(View.GONE);
            }
        }
        @Override
        public void onFailure(Call<String> call, Throwable t) {
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK");
            no_data.setVisibility(View.VISIBLE);
            rv_users.setVisibility(View.GONE);
        }
    });
  } catch (Exception e) {
    rv_users.setVisibility(View.GONE);
    Toast.makeText(requireContext(), e.toString(), Toast.LENGTH_SHORT).show();
  }
}
private void getUsersList() {
  try {
    Call<String> call = apiInterface.getUsers(prefrences.getStringValue("id", ""));
    call.enqueue(new Callback<String>() {
```

```
            @Override
            public void onResponse(Call<String> call, Response<String> response) {
               if (response.code() == 200) {
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK1");
                  rv_users.setVisibility(View.VISIBLE);
                  try {

                     JSONObject data = new JSONObject(response.body());
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK2");
                     if (data.getJSONArray("data").length() > 0) {
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK3");

                        no_data.setVisibility(View.GONE);
                        rv_users.setVisibility(View.VISIBLE);
                        UsersListAdapter adapter = new UsersListAdapter(context,
                              data.getJSONArray("data"));
                        rv_users.setLayoutManager(new LinearLayoutManager(context));
                        rv_users.setAdapter(adapter);
                     }else {
                        Log.d("key","!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!4");

                        no_data.setVisibility(View.VISIBLE);
                        rv_users.setVisibility(View.GONE);
                     }
```

```
                } catch (Exception e) {
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK5");

                    no_data.setVisibility(View.VISIBLE);
                    rv_users.setVisibility(View.GONE);
                }
            } else {
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK6");

                no_data.setVisibility(View.VISIBLE);
                rv_users.setVisibility(View.GONE);
            }
        }

        @Override
        public void onFailure(Call<String> call, Throwable t) {
Log.d("key","KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK");

            no_data.setVisibility(View.VISIBLE);
            rv_users.setVisibility(View.GONE);
        }
    });
} catch (Exception e) {
    rv_users.setVisibility(View.GONE);
```

```
            Toast.makeText(requireContext(), e.toString(), Toast.LENGTH_SHORT).show();
        }
    }
}
//ViewBusesFragment .java
package com.dzniox.travel_mate.ui.fragments;

import android.content.Context;
import android.os.Bundle;

import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;

import com.dzniox.travel_mate.R;
import com.dzniox.travel_mate.network.APIClient;
import com.dzniox.travel_mate.network.APIInterface;
import com.dzniox.travel_mate.sharedPref.PrefUtils;
```

```
import com.dzniox.travel_mate.ui.adapters.ViewBusesByOwnersAdapter;


import org.json.JSONObject;


import retrofit2.Call;

import retrofit2.Callback;

import retrofit2.Response;

public class ViewBusesFragment extends Fragment {

    private static final String ARG_PARAM1 = "param1";

    private static final String ARG_PARAM2 = "param2";


    private String mParam1;

    private String mParam2;


    private TextView Name, Email, no_data, Phone, bus_Owners, hotel_Owners, users;


    APIInterface apiInterface;

    private Context context;

    private View rootView;

    private RecyclerView rv_busesview;


    PrefUtils prefrences;

    public ViewBusesFragment() {

    }
```

```
public static ViewBusesFragment newInstance(String param1, String param2) {
    ViewBusesFragment fragment = new ViewBusesFragment();
    Bundle args = new Bundle();
    args.putString(ARG_PARAM1, param1);
    args.putString(ARG_PARAM2, param2);
    fragment.setArguments(args);
    return fragment;
}

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    rootView = inflater.inflate(R.layout.fragment_view_buses, container, false);
    context = getContext();
    prefrences = PrefUtils.getInstance(context);
```

```
apiInterface = APIClient.getClient().create(APIInterface.class);
rv_busesview = rootView.findViewById(R.id.rv_busesview);
no_data = rootView.findViewById(R.id.no_data);
Call<String> call = apiInterface.getBuseslist(prefrences.getStringValue("id", ""));

call.enqueue(new Callback<String>() {
  @Override
  public void onResponse(Call<String> call, Response<String> response) {
    if (response.code() == 200) {
      rv_busesview.setVisibility(View.VISIBLE);
      try {
        JSONObject data = new JSONObject(response.body());
        Log.d("key",response.body());
Log.d("key","HEYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY");
        if (data.getJSONArray("data").length() > 0) {
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");
          no_data.setVisibility(View.GONE);
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");
          rv_busesview.setVisibility(View.VISIBLE);
          ViewBusesByOwnersAdapter adapter = new ViewBusesByOwnersAdapter(context,
              data.getJSONArray("data"));

          rv_busesview.setLayoutManager(new LinearLayoutManager(context));
          rv_busesview.setAdapter(adapter);
```

```
                } else {
                    no_data.setVisibility(View.VISIBLE);
                    rv_busesview.setVisibility(View.GONE);
                }
            } catch (Exception e) {
                no_data.setVisibility(View.VISIBLE);
                rv_busesview.setVisibility(View.GONE);
            }

        } else {
            no_data.setVisibility(View.VISIBLE);
            rv_busesview.setVisibility(View.GONE);
        }
      }
      @Override
      public void onFailure(Call<String> call, Throwable t) {
        no_data.setVisibility(View.VISIBLE);
        rv_busesview.setVisibility(View.GONE);
      }
    });
    return rootView;
  }
}
```

//ViewHotelRoomsFragment .java

```
package com.dzniox.travel_mate.ui.fragments;


import android.content.Context;

import android.os.Bundle;


import androidx.fragment.app.Fragment;

import androidx.recyclerview.widget.LinearLayoutManager;

import androidx.recyclerview.widget.RecyclerView;


import android.util.Log;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.Button;

import android.widget.TextView;

import android.widget.Toast;


import com.dzniox.travel_mate.R;

import com.dzniox.travel_mate.network.APIClient;

import com.dzniox.travel_mate.network.APIInterface;

import com.dzniox.travel_mate.sharedPref.PrefUtils;

import com.dzniox.travel_mate.ui.adapters.RoomsViewByOwnerAdapter;

import com.dzniox.travel_mate.ui.adapters.ViewBusesByOwnersAdapter;
```

```
import org.json.JSONObject;


import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
public class ViewHotelRoomsFragment extends Fragment {
   private static final String ARG_PARAM1 = "param1";
   private static final String ARG_PARAM2 = "param2";
   private String mParam1;
   private String mParam2;
   private TextView Name, Email, no_data, Phone, bus_Owners, hotel_Owners, users;
   APIInterface apiInterface;
   private Context context;
   private View rootView;
   private RecyclerView rv_roomsview;
   PrefUtils prefrences;
Button btn;
   public ViewHotelRoomsFragment() {
   }
   public static ViewBusesFragment newInstance(String param1, String param2) {
      ViewBusesFragment fragment = new ViewBusesFragment();
      Bundle args = new Bundle();
      args.putString(ARG_PARAM1, param1);
      args.putString(ARG_PARAM2, param2);
```

```
    fragment.setArguments(args);
    return fragment;
}

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    rootView = inflater.inflate(R.layout.fragment_view_hotel_rooms, container, false);
    context = getContext();
    prefrences = PrefUtils.getInstance(context);
    apiInterface = APIClient.getClient().create(APIInterface.class);
    rv_roomsview = rootView.findViewById(R.id.rv_roomsview);
    no_data = rootView.findViewById(R.id.no_data);

    Call<String> call = apiInterface.getRoomslist(prefrences.getStringValue("id", ""));
```

```
        call.enqueue(new Callback<String>() {
            @Override
            public void onResponse(Call<String> call, Response<String> response) {
                if (response.code() == 200) {
                    rv_roomsview.setVisibility(View.VISIBLE);
                    try {
                        JSONObject data = new JSONObject(response.body());
                        Log.d("key",response.body());
Log.d("key","HEYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY");
                        if (data.getJSONArray("data").length() > 0) {
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");


                            no_data.setVisibility(View.GONE);
Log.d("key","HEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE");
                            rv_roomsview.setVisibility(View.VISIBLE);
                            Log.d("key","adStart****************************************");


                            RoomsViewByOwnerAdapter adapter = new RoomsViewByOwnerAdapter(context,
                                    data.getJSONArray("data"));
                            Log.d("key","adEnd****************************************");
                            rv_roomsview.setLayoutManager(new LinearLayoutManager(context));
                            rv_roomsview.setAdapter(adapter);
```

```
                } else {
                    no_data.setVisibility(View.VISIBLE);
                    rv_roomsview.setVisibility(View.GONE);
                }
            } catch (Exception e) {
                no_data.setVisibility(View.VISIBLE);
                rv_roomsview.setVisibility(View.GONE);
            }

        } else {
            no_data.setVisibility(View.VISIBLE);
            rv_roomsview.setVisibility(View.GONE);
        }
    }

    @Override
    public void onFailure(Call<String> call, Throwable t) {
        no_data.setVisibility(View.VISIBLE);
        rv_roomsview.setVisibility(View.GONE);
    }
  });
  return rootView;
 }
}
```

SYSTEM IMPLEMENTATION

7.SYSTEM IMPLEMENTATION

The implementation of a tour and travel management system involves several key steps to ensure its functionality and effectiveness.Firstly, the system architecture needs to be designed, outlining the various components such as user interface, database management, booking systems, payment processing, and administrative functionalities. This phase involves deciding on the technology stack, programming languages, and frameworks to be used.Next, the development process begins with coding the different modules of the system based on the outlined architecture. This includes creating interfaces for users to search and book tours, managing accounts, processing payments securely, and generating reports for administrative purposes.During the development phase, thorough testing is conducted to identify and fix any bugs or issues. This includes unit testing of individual components, integration testing to ensure smooth interaction between different modules, and system testing to validate the system as a whole.Once development and testing are complete, the system is deployed onto servers or cloud platforms, making it accessible to users over the internet. Security measures such as encryption and access control are implemented to protect user data and transactions.Training sessions may be conducted for administrators and staff members to familiarize them with the new system and its functionalities. User feedback is also collected during this phase to identify any areas for improvement or additional features that may be required.Finally, ongoing maintenance and updates are essential to keep the system running smoothly and to address any new requirements or issues that may arise over time. Regular monitoring of system performance and user satisfaction helps to ensure the system continues to meet the needs of its users effectively.

SYSTEM MAINTENANCE

8. SYSTEM MAINTENANCE

Maintaining a tour and travel management system involves ongoing efforts to ensure its reliability, security, and efficiency. This includes regular updates to incorporate software enhancements, security patches, and bug fixes. Database management is essential for optimizing data storage and retrieval, while implementing robust security measures such as encryption and access control safeguards sensitive information. Backup and recovery procedures are crucial to prevent data loss in case of system failures or cyberattacks. Providing user support and training helps users navigate the system effectively, while performance monitoring identifies and addresses any performance issues promptly. Gathering feedback from users and stakeholders informs future updates and enhancements, ensuring that the system remains compliant with industry regulations and standards. Through these measures, a well-maintained tour and travel management system can deliver a seamless experience for both customers and operators alike.

CONCLUSION

9.CONCLUSION

In conclusion, the implementation of a tour and travel management system is a comprehensive process that involves careful planning, development, testing, deployment, training, and ongoing maintenance. By leveraging technology, businesses in the travel industry can streamline operations, enhance customer experiences, and improve overall efficiency.Through the systematic design and development of user-friendly interfaces, robust booking systems, secure payment processing, and efficient administrative functionalities, tour and travel management systems enable businesses to manage their operations more effectively while providing customers with convenient and personalized booking experiences.Moreover, the continuous monitoring and adaptation of the system based on user feedback and evolving industry trends ensure its relevance and effectiveness in meeting the changing needs of both businesses and travelers. Overall, tour and travel management systems play a vital role in driving innovation and competitiveness in the travel industry, ultimately leading to improved customer satisfaction and business

SCOPE FOR FURTHER ENHANCEMENT

10.SCOPE FOR FURTHER ENHANCEMENT

BIBLIOGRAPHY

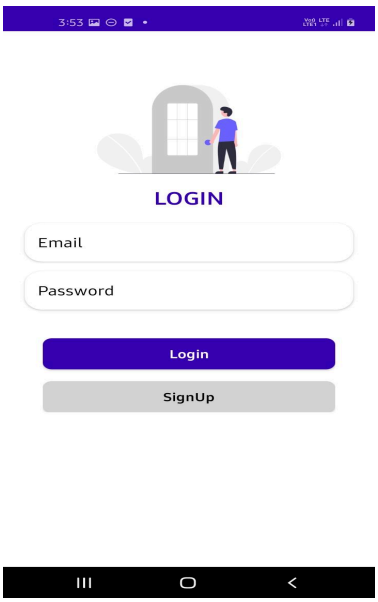
11. BIBLIOGRAPHY

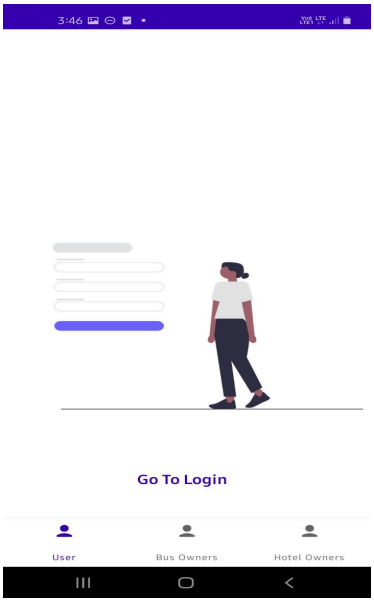
https://nevonprojects.com/android-tour-travel-agency-project/

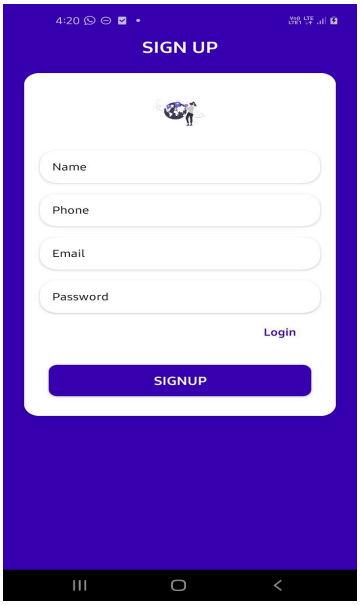
APPENDIX

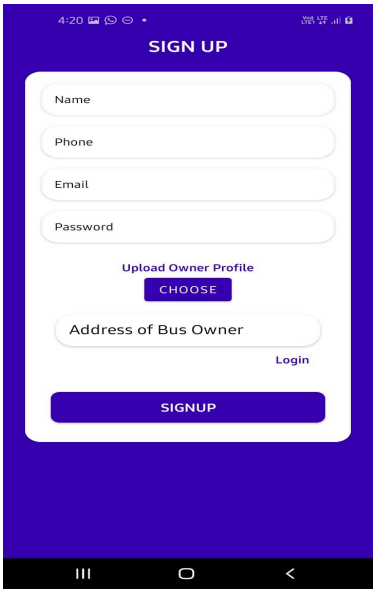
12. APPENDIX

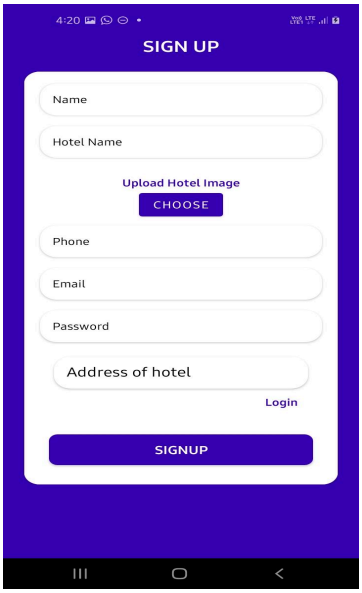
In the appendix of a tour and travel management system, additional details and resources related to the system can be provided. This may include technical documentation such as system architecture diagrams, database schemas, API specifications, and user manuals.Moreover, the appendix can include supplementary materials such as sample booking forms, payment processing flows, and customer communication templates. These materials help stakeholders, including developers, administrators, and end-users, to better understand the functionalities and usage of the system.Additionally, the appendix may contain information on third-party integrations, regulatory compliance requirements, and security protocols implemented within the system. This ensures transparency and accountability in the system's operation and ensures that all relevant standards and regulations are adhered to.Furthermore, the appendix can include references to relevant research papers, case studies, and industry reports that informed the design and development of the system. This provides additional context and background information for stakeholders interested in the system's development process and its impact on the travel industry.Overall, the appendix serves as a comprehensive resource for stakeholders to reference and access additional information related to the tour and travel management system, enhancing their understanding and facilitating their engagement with the system.


LOGIN
Email
Password
Login
SignUp


Go To Login
User
Bus Owners
Hotel Owners


SIGN UP
Name
Phone
Email
Password
Login
SIGNUP


SIGN UP
Name
Phone
Email
Password
Upload Owner Profile
CHOOSE
Address of Bus Owner
Login
SIGNUP


SIGN UP
Name
Hotel Name
Upload Hotel Image
CHOOSE
Phone
Email
Password
Address of hotel
Login
SIGNUP