**“Travel Hunt”**

A project report submitted in partial fulfillment

Of requirement for the award of the degree of

**BACHELOR OF COMPUTER APPLICATIONS**



**RANI CHANNAMMA UNIVERSITY, BELAGAVI.**

**Submitted by:**

**Mr. Rohit Patil**

**Mr. Saiprasad Patil**

**Under the guidance of**

**Prof. Shruti Kulkarni**



**Karnatak Law Society’s**

**GOGTE COLLEGE OF COMMERCE**

**Tilakwadi, Belgaum-590006**

**2019-2020**

**“Travel Hunt”**

A project report submitted in partial fulfillment

Of requirement for the award of the degree of

**BACHELOR OF COMPUTER APPLICATIONS**



**RANI CHANNAMMA UNIVERSITY, BELAGAVI.**

**Submitted by:**

**Mr. Rohit Patil**

**Mr. Saiprasad Patil**

**Under the guidance of**

**Prof. Shruti Kulkarni**



**Karnatak Law Society’s**

**GOGTE COLLEGE OF COMMERCE**

**Tilakwadi, Belgaum-590006**

**2019-2020**

**Karnatak Law Society’s**

**GOGTE COLLEGE OF COMMERCE**

**Tilakwadi, Belgaum.**

**Bachelor of Computer Applications**

****

**CERTIFICATE**

**This is to certify that**

**Mr.Saiprasad Patil Reg No: M1710508**

**Has satisfactorily completed the project work entitled**

**“Travel Hunt”**

**FOR THE FULFILLMENT OF BACHELOR OF COMPUTER APPLICATIONS OF RANI CHANNAMMA UNIVERSITY, BELAGAVI, FOR THE YEAR 2018-2019**

**Guide Coordinator Director, BCA section**

**Prof.Shruti Kulkarni Prof.V.S. Jalihal Dr.S. G. Sugur**

**EXAMINERS**

**1. 2.**

**Karnatak Law Society’s**

**GOGTE COLLEGE OF COMMERCE**

**Tilakwadi, Belgaum.**

**Bachelor of Computer Applications**

****

**CERTIFICATE**

**This is to certify that**

**Mr. Rohit Patil Reg No: M1710505**

**Has satisfactorily completed the project work entitled**

**“Travel Hunt”**

**FOR THE FULFILLMENT OF BACHELOR OF COMPUTER APPLICATIONS OF RANI CHANNAMMA UNIVERSITY, BELAGAVI, FOR THE YEAR 2018-2019**

**Guide Coordinator Director, BCA section**

**Prof. Shruti Kulkarni Prof.V.S. Jalihal Dr. S. G. Sugur**

**EXAMINERS**

**1. 2.**

**ACKNOWLEDGEMENT**

***“A helping hand, kind soul these are the ingredients that help to make a success out of any effort.”***

We take this opportunity to acknowledge the contribution of each individual who has in some way or the other helped me in completing this project successfully. We express our gratitude to our institute, **Karnatak Law Society’s Gogte College of Commerce (BCA Section), Belgaum** and our Principal **Dr. S.G. Sugur**for being the source of encouragement.

We also enhance our gratitude to our beloved **Prof.VenugopalJalihal**, coordinator BCA Dept. for his constant inspiration and necessary resources and working environment in the college.

With pleasure, we use this occasion to concede our heartfelt thanks to**Prof.Shruti Kulkarni**our internal guide who has guided us with her valuable suggestions & guidance.

Our expressions extend unbounded to thank our most beloved parents & family members who have always been a moral support & strong pillars at every stage of our life with cheer enthusiasm. We dedicate our work to them. Last but not least, we are thankful to Almighty moral support, which helped us during the successful completion of the project.

***With heartfelt thanks to One & All,***

***Rohit Patil and Saiprasad Patil***

**INDEX**

|  |  |  |
| --- | --- | --- |
| **SL.NO** | **TOPIC** | **PAGE** |
| 1. | Introduction   1. Synopsis 2. Abstract 3. Objectives and Scope of Project | 1 |
| 2. | System Analysis   1. Analysis of Existing system 2. Analysis of Proposed system 3. System Specification 4. Functional And Non Functional Requirements 5. Feasibility Study 6. Methadology | 10 |
| 3 | System Design   1. Pert Chart 2. Gantt Chart 3. Data Flow Diagram 4. Use Case Diagram | 22 |
| 4. | System Implementation   1. Tools and languages 2. Technologies used 3. Coding 4. ScreenShots | 32 |
| 5. | System Evaluation   1. Software Testing 2. Test Cases | 91 |
| 6. | Conclusion | 105 |
| 7. | Annexure | 107 |
| 8. | Future Enhancement | 109 |
| 9. | References | 111 |

**INTRODUCTION**

**INTRODUCTION:**

**“Travel Hunt”** is an application designed to make it easier for people to book tour packages online. User may login and view various tour and travel destinations. Post login user sees different destinations to travel. User can select different types of travelling packages i.e. Domestic, International, Religious, Popular and Dream Destinations. Each ones amenities type is shown in the system with images, description and respective costing. The user may select desired packages as needed and then the payment is calculated accordingly. The user may then pay via Credit/Debit Cards, Payment Apps or by any other payments method. User may also submit a feedback form that will be delivered to the admin.

**SYNOPSIS**

**INTRODUCTION:**

**“Travel Hunt”** is an application designed to make it easier for people to book tour packages online. User may login and view various tour and travel destinations. Post login user sees different destinations to travel. User can select different types of travelling packages i.e. Domestic, International, Religious, Popular and Dream Destinations. Each ones amenities type is shown in the system with images, description and respective costing. The user may select desired packages as needed and then the payment is calculated accordingly. The user may then pay via Credit/Debit Cards, Payment Apps or by any other payments method. User may also submit a feedback form that will be delivered to the admin.

**MODULES:**

1. **Admin Login**

* Here, the admin of the travel hunt application adds his credentials in the Login page.

1. **User login/registration**

* Here, the user of the Real Estate application put his credentials in the Login page and also new user can register on this application.

1. **Tour Packages**

* Here, user can view various tour packages by selecting the destination location.

1. **Booking**

* Here, the user can book their desired package.

1. **Payment**

* Here, the Application calculates the cost and accordingly user pays the bill using online payment methods.

**System Requirement:**

**Software Requirement Specification:**

Operating System -- Windows

Coding Language -- Java

Toolkit -- Android SDK

IDE -- Android Studio

Backend Services -- Firebase

**Hardware Requirement Specification:**

System -- i3 Processor or Above

Hard Disk -- 200 GB or Above

RAM -- 8 GB or Above

**Runtime Environment:**

Device -- Smartphone /Tablet

Operating System -- Android

Minimum Version -- 4.4Android KitKat

**Advantages:**

* The Travel Hunt applications have made it possible to book a trip online and to compare prices with other companies so that user can be sure that they are getting a good deal.
* It is easy to use, secured application.
* This System has integration of the most popular payment gateways, as well as can process Credit/Debit card payments and other payments methods.

**Disadvantages:**

* Requires an active internet connection.
* May provide inaccurate results if data not feed properly.

**ABSTRACT**

**ABSTRACT:**

This report consists of a concise insight of the project undertaken by me. The project could be brought into existance through the cooperation and coordation offered by my partner.

The entire report is organized in a friendly manner so as to gain easy access to the required information. The report consists of lot of information as to use the software to enable beginners to quickly get acquired with the usage of the project. The report facilitates a better understanding of the concepts and the methodologies involved in the conception, creation and implementation of the project.

As tourism is one of the fastest growing industries today, thus within the tourism industry events are getting more and more important. Tourism is a way of utilizing resoures, which would otherwise remain idle for the instance Landscapes and cultural sites.

**Objectives**

**And**

**Scope**

**Objective**

**And**

**Scope**

**OBJECTIVE:**

To include the detailed description about the particular place this helps in knowing the information about the particular place.

* Maintains a centralized repository of all related information.
* To give a variety of travel services that will sure to match all your priorities.
* To display attractive tourist places.
* To provide the facility of giving their likes and dislikes about the place.
* To select best packages according to their budget and choice.

**SCOPE:**

This application can be widely used by various users such as,

* Tourist
* Tours and Travel Agencies.
* Students
* Every State tourism department.
* Schools and Colleges for tours.
* Photographers for Shooting

**System**

**Analysis**

## SYSTEM ANALYSIS:

It is a process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components.

System analysis is conducted for the purpose of studying a system or its parts in order to identify its objectives. It is a problem-solving technique that improves the system and ensures that all the components of the system work efficiently to accomplish their purpose.

Analysis specifies what the system should do.

* **Existing System:**

The existing system does not have reservation facilities of buses, flights and railways. And the list and information about the restaurant in a particular place are not available in a single application. It is a tedious work for tourists to search for the reservation and restaurants outside.

* **Proposed System:**

To overcome the drawback of existing system which basically does not have reservation facility, restaurants details and package details; proposed system provides this all facilities. The tourists get every facility in a single application. This application will provide to connect tourist and agents directly.

* **Feature of Proposed System.**
  + Very easy to book Tour places.
  + Beautiful places and with various activities.
  + Various category of packages
  + And special addition in fort view and Seven wonders of the world.
  + Safe tour and tour guides are present ineach of the tours.
  + Resort provides best services.
  + Payment method is secure and safe.
  + Booking cancellation.
  + Money will be refunded within one day.

**MODULE DESCRIPTION:**

* **User Module:**

In this the user needs to run the application and then register into it. Registering into the application requires some fields such as Email, Name, Phone no., Password, Profile image, after registering in this application they have to verify the email for security purpose. Then they have to login into the app.After that they can see the tourist places and the related details of packages, and thenthey can choose packages and book a tour and add some security details. After adding the details the user need to select the payments methodthat includes Cards, UPI payments. If the user is successfully booked for particular packages they will get a notification and a mail.

* **Registration/Login Module:**
* In registration module the user should register into the application. Registering into the application requires some fields such as name, email, phone no, password, Profile Image.
* In login module user should login their account using registered email and password.
* **OTP Verification Module:**
* In this module after register the OTP is sent to your registered email and phone number. If the OTP is verified and entered correctly the user is successfully registered. Otherwise registration is unsuccessful.
* **Forgot Password Module:**
* In this module if the user forgot his/her password then user can reset the password through email.
* **Book Tour Module:**
* In this module users can book their tour as per his/her choice.
* **Payment Module:**
* **Online Payment:-**This form payment includes UPI Payment like Google pay, Phone pe, Amazon Pay, Paytm and cards.
* **Notification Module:**
* In this module a notification is sent when booking is confirmed. Booking confirmation is also sent to your registered e-mail id.
* **Admin Module:**

In this Module the Admin must register using admin’s name, email, Phone number and password .If the admin is already registered then just login into the app using email and password .The Admin can add Tour Packages indiffernent categories and the details of packages inside this app. The Admin receives the Booking details from the user’s .Then Admin will be able to accept or cancel of the tour packages. Admin can also edit and delete specific packages.

* **Registration/Login Module:**
* In this module admin should register in the app by entering Admin name, admin email, phone no. and password.
* In this module admin should login by using registered email and password.
* **Forgot Password Module:**
* In this module if the adminforgots the password then to reset the password an email password reset link is sent on the registered mail.
* **Add New Package Module:**
* In this module the admin can also add the new tour packages with details into a specific category.
* **Delete/Edit Module:**

In this module the Admincan delete and edit details of existing tour packages.

**SYSTEM SPECIFICATION**

**Software Requirement Specification:**

* Operating System Windows
* Coding Language              Java
* Toolkit Android SDK
* IDE Android Studio
* Backend Services              Firebase

**Hardware requirements:**

* Processor i3 or Above
* RAM 8GB or Above
* Hard disk 200GB

**Runtime Environment:**

* Device Smartphone/Tablet
* Operating System Android
* Minimum Version Android 4.4 (Kit Kat)

**FUNCTIONAL REQUIREMENT:**

Functional requirements define the capabilities and functions that an apllication must be able to perform successfully. The functional requirements of this Travel Hunt App include:

**USERS:**

* The application shall enable the customer to create an account, login to the app, view the packages and book the specific packages.
* The application shall display the booked packages, the individual packages prices and the payment amount is calculated.
* The application shall provide visual confirmation of the package booking.
* The application also provides the online payment method to the customer.

**ADMIN:**

* The application shall enable the Admin to create an account, login to the app, add, delete and update of the packages.
* The application shall enable the Admin to view, create, edit and delete package categories and details.
* The appliction shall allow the admin to update price for a specific package.
* The application will prompt with notification to admin, if user is booked a tour package.

**NON-FUNCTIONAL REQUIREMENTS:**

A non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of an app, rather than specific behaviors.Some of the non-functional requirements includes:

**1. Performance**

- Initial load time should not exceed one second

**2. Scalability**

- The increasing number of users should not affect the performance of the application

**3. Extensibility**

- New features shall be easy to implement with separation of concern.

**4. Security**

**-**Each user data will secure and safe.

**a) Confidentiality**

-Traffic confidentiality shall be protected, all operations performed by tourist and admin must be preserved.

**b) Integrity**

- The integrity of all operations performed by users must be preserved.

**c) Availability**

- No single point of failure shall be tolerated.

**FEASIBILITY STUDY:**

The feasibility study is an evaluation and analysis of the potential of a proposed project which is based on exclusive investigation and research to support the process of decision making. The main objective of the feasibility study is to test the technical and economical feasibility of developing the computer system. A system is feasible system if it is feasible within limited resources and time.

**The key Consideration in Feasibility Study Are:**

1. Technical Feasibility.
2. Economical Feasibility.
3. Operational Feasibility.
4. **Technical Feasibility:**

“Technical feasibility is concerned with availability and capability of Software, Hardware and people”. Technical Feasibility centers on the existing android application (software) and to what extent it can support the proposed addition. The proposed system can be implemented with the existing technology with few changes easily.

* Since the project is designed in **Android Studio**. It can be easy to install in all the smart phones whenever needed.
* It is more efficient, easy and **user-friendly** to understand by almost everyone.
* The backend will be implemented using **Firebase**, its purpose is to handle database queries, authentication, and to serve an API (Application Programming Interface)
* Hence the proposed system is technically feasible.

1. **Economical Feasibility:**

“Economical analysis is the most used method for evaluating the effectiveness of the proposed software”.

In the present application, the development of new product greatly enhances the accuracy of the application. The errors can be greatly reduced and at the same time providing a great level of security. There is no need of any additional equipment expect memory of required capacity Hence the proposed system is economical feasible i.e. efficient, reliable and also quickly responding.

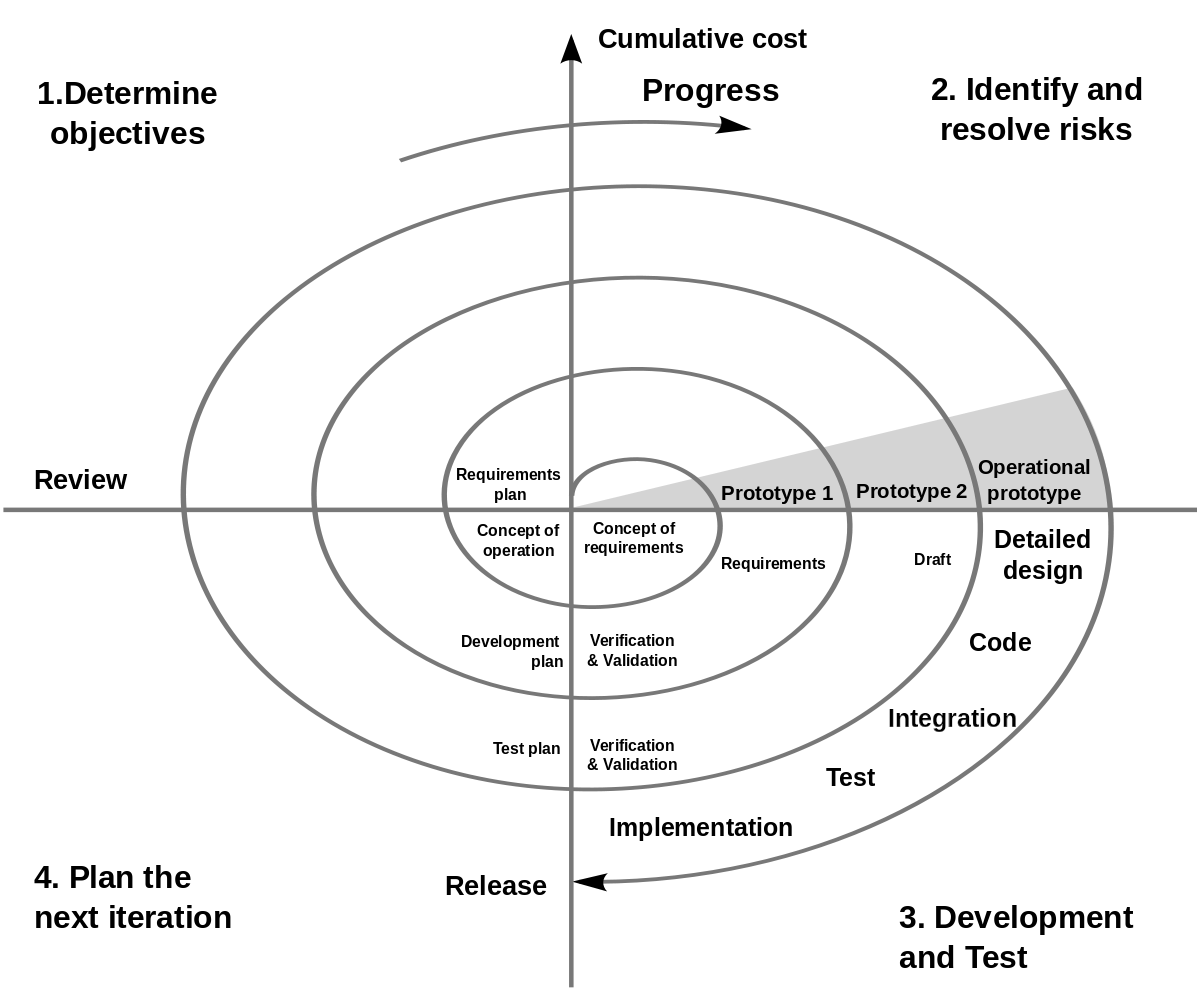
1. **Operational Feasibility:**

Operational feasibility is the measure of how well the project will support the customer and the service provider during the operational phase.It is dependent on human resources available for the project and involves projecting whether the system will be used if it is developed and implemented.The essential questions that help in testing the technical feasibility of a system include the following:

* Is the project is feasible to operate or not?
* Does the current mode of operation provide adequate throughput and response time?
* Does the current mode of operation make maximum use of available resources including people, time, and flow of forms?
* If the app is developed, will it be in used?

**Methodology Adopted**

**SPIRAL MODEL**



The spiral model, initially proposed by **Boehm**, is an evolutionary software process model that couples the iterative feature of prototyping with the controlled and systematic aspects of the **linear sequential model.** It implements the potential for rapid development of new versions of the software. Using the spiral model, the software is developed in a **series of incremental releases**. During the early iterations, the additional release may be a paper model or prototype. During later iterations, more and more complete versions of the engineered system are produced.

**Determine Objectives**

* Each cycle in the spiral starts with the identification of purpose for that cycle, the various alternatives that are possible for **achieving the targets**, and the constraints that exists.

1. **Identify and resolve risks**

The next phase in the cycle is to calculate these various alternatives based on the goals and constraints. The focus of evaluation in this stage is located on the **risk perception** for the project.

1. **Development and test**

The next phase is to develop strategies that resolve **uncertainties and risks**. This process may include activities such as **benchmarking, simulation, and prototyping**.

1. **Plan the next iteration**

Finally, the next step is planned. The project is reviewed, and a choice made whether to continue with **a further period** of the spiral. If it is determined to keep, plans are drawn up for the next step of the project.

The development phase depends on the **remaining risks**. For example, if performance or user-interface risks are treated more essential than the program development risks, the next phase may be an evolutionary development that includes developing a more detailed prototype for solving the risks.

The **risk-driven** feature of the spiral model allows it to accommodate any mixture of a specification-oriented, prototype-oriented, simulation-oriented, or another type of approach. An essential element of the model is that each period of the spiral is completed by a review that includes all the products developed during that cycle, including plans for the next cycle. The spiral model works for development as well as enhancement projects.

**Advantages of Spiral model**

* High amount of risk analysis hence, avoidance of Risk is enhanced.
* Strong approval and documentation control.
* Additional Functionality can be added at a later date.
* Software is produced early in the software life cycle.

**System Design**

**PERT CHART**

A PERT chart is a graphic representation of a project’s schedule, showing the sequence of tasks, which tasks can be performed simultaneously, and the critical path of tasks that must be completed on time in order for the project to meet its completion deadline. The chart can be constructed with a variety of attributes, such as earliest and latest start dates for each task, earliest and latest finish dates for each task, and slack time between tasks. A PERT chart can document an entire project or a key phase of a project. The chart allows a team to avoid unrealistic timetables and schedule expectations, to help identify and shorten tasks that are bottlenecks, and to focus attention on most critical tasks.

Because it is primarily a project-management tools, a PERT chart is most useful for planning and tracking entire projects or for scheduling and tracking the implementation phase of planning or improvement effort.

|  |
| --- |
| SYSTEM ANALYSIS AND DESIGN |
| 0 8days |
| 26/12/19 02/01/20 |

|  |
| --- |
| TECHNICAL STUDY |
| 1 8days |
| 03/01/20 10/01/20 |

**Fig (1)**

|  |
| --- |
| SYSTEM ANALYSIS AND DESIGN |
| 3 8days |
| 19/01/20 25/01/20 |

|  |
| --- |
| DATABASE STUDY |
| 2 8days |
| 11/12/19 18/01/20 |

|  |
| --- |
| CODING |
| 4 40days |
| 26/01/20 04/03/20 |

|  |
| --- |
| TESTING |
| 5 15days |
| 05/03/20 20/03/20 |

|  |
| --- |
| FINAL DOCUMENTATION |
| 6 8days |
| 05/04/20 12/04/20 |

**GANTT CHART**

A Gantt chart is a popular tool in project management. It basically drills down activities which need to be done by a fixed time period. It is commonly used for tracking project schedules. It schedules the start and finish dates of the terminal elements and summary elements of a project.

On the chart, tasks are shown on the vertical axis while they scheduled time-spend is laid out on the horizontal axis. Each task is represented by a bar that shows the time required for the project.

Gantt chart is useful tool in planning and scheduling the projects. It keeps the management updated as to when the project will get completed. It also keeps the management informed about any additional resources that are required and manage dependencies between tasks

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| NO | **TASK** | Duration in days | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 1 | **Problem**  **Requirement** | 10 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | **Analysis** | 10 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | **Designing** | 15 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | **Coding** | 50 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | **Testing** | 10 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | **Implementation** | 15 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 | **Report** | 5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**Fig (2)**

**DATA FLOW DIAGRAM**

1. **DATA FLOW DIAGRAMS (DFD)**

A Data Flow Diagram is graphical aid for defining systems input, process and outputs. It represents increasing information flow and function details.

A DFD shows the flow of data through a system. A system may be an organization, a manual procedure, software system, mechanical system, a hardware system or any combinations of these. A DFD shows the movement of data through different processes in the system DFD’s are made of a number of symbols, which represents system components like process, data flow and external entities.

1. **Process:**

Process show that systems do each process can have one or more inputs or outputs. A process in DFD is represented as hollow cylinder. And each process has unique name and number.

1. **Data Store:**

A file or data store is repository of data. Each data store is represented by an open sided rectangle and each store and has unique name. The symbol of data store is.

1. **External Entities:**

External entities are outside the system but they either supply input into system or use the system output. They may be an organization’s customer or others which system interacts. External entities, which supply data to the system, are sometimes called as ‘sinks’. These are represented by a rectangle in the DFD.

1. **Data Flow**

Data flow indicates the passage of data in the system, from where the data flows. It is indicated by an arrow which indicates the direction of flow. The arrow is labeled by the name of the data flow.

Flow of data in the system could be any one of the following:

* From a data store to a process.
* From source to process.
* From a process to a sink.

**CONTEXT LEVEL DIAGRAM:**

Package details,

Booking

Manage

Packages

Tourist

Admin

**Fig (3)**

Tour Package DB

Admin

1

Tourist DB

2

Identity Proof

Login DB

Tourist

Details

3

Username & Password

Username & Password

4

Email

Package Details

Package Details

5

Details

6

**USECASE DIAGRAM:**

A [Use case](https://searchsoftwarequality.techtarget.com/definition/use-case) is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system" refers to something being developed or operated, such as a mail-order product sales and service [Web site](https://whatis.techtarget.com/definition/Web-site). Use case diagrams are employed in [UML](https://searchsoftwarequality.techtarget.com/definition/Unified-Modeling-Language) (Unified Modeling Language), a standard notation for the modeling of real-world objects and systems.

System objectives can include planning overall requirements, validating a hardware design, testing and [debugging](https://searchsoftwarequality.techtarget.com/definition/debugging) a software product under development, creating an online help reference, or performing a consumer-service-oriented task. For example, use cases in a product sales environment would include item ordering, catalog updating, payment processing, and customer relations. A use case diagram contains four components.

* The boundary, which defines the system of interest in relation to the world around it.
* The actors, usually individuals involved with the system defined according to their roles.
* The use cases, which the specific roles are played by the actors within and around the system.
* The relationships between and among the actors and the use cases.

**DESIGN:**

ADMIN

TOURIST

**System Implementation**

**SYSTEM IMPLEMENTATION:**

**Tools and Langauges:**

To develop this software, we have used the following tools and languages.

* IDE
  + Android Studio
* Front End
  + Xml
* Middleware
  + Java
  + Android SDK
* Backend Services
  + Firebase

**TECHNOLOGY USED**

1. **Extensible Markup Language(XML):-**

XML stands for **E**xtensible **M**arkup **L**anguage. It is a text-based markup language derived from Standard Generalized Markup Language (SGML).XML tags identify the data and are used to store and organize the data, rather than specifying how to display it like HTML tags, which are used to display the data. XML is not going to replace HTML in the near future, but it introduces new possibilities by adopting many successful features of HTML.There are three important characteristics of XML that make it useful in a variety of systems and solutions −

* **XML is extensible** − XML allows you to create your own self-descriptive tags, or language, that suits your application.
* **XML carries the data, does not present it** − XML allows you to store the data irrespective of how it will be presented.
* **XML is a public standard** − XML was developed by an organization called the World Wide Web Consortium (W3C) and is available as an open standard.

XML is a markup language that defines set of rules for encoding documents in a format that is both human-readable and machine-readable. So *what exactly is a markup language?* Markup is information added to a document that enhances its meaning in certain ways, in that it identifies the parts and how they relate to each other. More specifically, a markup language is a set of symbols that can be placed in the text of a document to demarcate and label the parts of that document.

Following example shows how XML markup looks, when embedded in a piece of text −

<message>

<text>Hello, world!</text>

</message>

This snippet includes the markup symbols, or the tags such as <message>...</message> and <text>... </text>. The tags <message> and </message> mark the start and the end of the XML code fragment. The tags <text> and </text> surround the text Hello, world!.

1. **Java:-**

Java is a general-purpose computer **programming language** that is [concurrent](https://howtodoinjava.com/java-concurrency-tutorial/), class-based, [object-oriented](https://howtodoinjava.com/oops/object-oriented-principles/), and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers **“write once, run anywhere” (WORA)**, meaning that compiled Java code can run on all platforms that support Java without the need for recompilation.

For example, you can write and compile a Java program on UNIX and run it on Microsoft Windows, Macintosh, or UNIX machine without any modifications to the source code. WORA is achieved by compiling a Java program into an intermediate language called **byte code**. The format of byte code is platform-independent. A virtual machine, called the [Java Virtual Machine (JVM)](https://howtodoinjava.com/java/basics/jdk-jre-jvm/), is used to run the byte code on each platform.

1. **Firebase:**

**Firebase** is a Backend-as-a-Service (Baas). It provides developers with a variety of tools and services to help them develop quality apps, grow their user base, and earn profit. It is built on Google’s infrastructure. Firebase is categorized as a [NoSQL](https://www.educative.io/edpresso/whats-the-difference-betweensql-and-nosql) database program, which stores data in JSON-like documents.

In Firebase, a document is a set of key-value pairs defined by a schema. A group of documents makes up a collection.

**Key Features**

1. **Authentication:**

It supports authentication using passwords, phone numbers, Google, Facebook, Twitter, and more. The Firebase Authentication (SDK) can be used to manually integrate one or more sign-in methods into an app.

### Real-time database:

Data is synced across all clients in real-time and remains available even when an app goes offline. It uses JSON tree to save the data.

1. **Cloud Firestore:**

Cloud Firestore is a cloud-hosted, NoSQL database that your ios, android, and web app can access directly via native SDKs. It uses document/collection model to save the data.

### Hosting:

Firebase Hosting provides fast hosting for a web app; content is cached into content delivery networks worldwide.

### Test lab:

The application is tested on virtual and physical devices located in Google’s data centers.

### Cloud Messaging:

Firebase Cloud Messaging, formerly known as Google Cloud Messaging, is a cross-platform cloud solution for messages and notification for Android, iOS, and Web application, which currently can be used at no cost.

1. **Android SDK:**

Every time Google releases a new version of Android, a corresponding SDK is also released. To be able to write programs with the latest features, developers must download and install each version’s SDK for the particular phone.

The development platforms that are compatible with SDK include operating systems like Windows (XP or later), Linux (any recent Linux distribution) and Mac OS X (10.4.9 or later). The components of Android SDK can be downloaded separately. Third party add-ons are also available for download.

Although the SDK can be used to write Android programs in the command prompt, the most common method is by using an integrated development environment (IDE). The recommended IDE is Eclipse with the Android Development Tools (ADT) plug-in. However, other IDEs, such as NetBeans or IntelliJ, will also work. Most of these IDEs provide a graphical interface enabling developers to perform development tasks faster. Since Android applications are written in Java code, a user should have the Java Development Kit (JDK) installed.

**CODING**

**CODING**

* **Login Page**

package com.example.exploreworld.authentication;

import androidx.annotation.NonNull;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.graphics.Color;

import android.os.Bundle;

import android.text.InputType;

import android.util.Patterns;

import android.view.View;

import android.view.WindowManager;

import android.widget.Button;

import android.widget.CheckBox;

import android.widget.CompoundButton;

import android.widget.EditText;

import android.widget.ImageView;

import android.widget.ProgressBar;

import android.widget.ScrollView;

import android.widget.TextView;

import com.bumptech.glide.Glide;

import com.example.exploreworld.R;

import com.example.exploreworld.main.Customer\_drawer;

import com.google.android.gms.tasks.OnCompleteListener;

import com.google.android.gms.tasks.Task;

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

import com.google.firebase.auth.AuthResult;

import com.google.firebase.auth.FirebaseAuth;

import com.google.firebase.auth.FirebaseAuthInvalidCredentialsException;

import com.google.firebase.auth.FirebaseAuthInvalidUserException;

import com.google.firebase.auth.FirebaseUser;

public class Customer\_login extends AppCompatActivity {

ImageView imageView;

EditText edtusername, edtpassword;

TextView txtsignup, txtForgot;

Button btnlog;

FirebaseAuth mAuth;

CheckBox checkBox;

ProgressBar progressBar;

ScrollView scrollView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_customer\_login);

getWindow().setFlags(WindowManager.LayoutParams.FLAG\_FULLSCREEN, WindowManager.LayoutParams.FLAG\_FULLSCREEN);

imageView = findViewById(R.id.loginimage);

String uri = "https://img.freepik.com/free-vector/account-log-page\_41910-263.jpg?size=626&amp;ext=jpg";

Glide.with(this).load(uri).into(imageView);

edtusername = findViewById(R.id.ctusername);

edtpassword = findViewById(R.id.ctpassword);

btnlog = findViewById(R.id.btnlogin);

txtsignup = findViewById(R.id.txtsignup);

txtForgot = findViewById(R.id.txtforgot);

checkBox=findViewById(R.id.show);

progressBar=findViewById(R.id.ctprogressbar);

scrollView=findViewById(R.id.loginview);

mAuth = FirebaseAuth.getInstance();

checkBox.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {

@Override

public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {

if(isChecked)

{

edtpassword.setInputType(InputType.TYPE\_TEXT\_VARIATION\_VISIBLE\_PASSWORD);

}

else

{

edtpassword.setInputType(InputType.TYPE\_CLASS\_TEXT | InputType.TYPE\_TEXT\_VARIATION\_PASSWORD);

}

}

});

txtForgot.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

startActivity(new Intent(Customer\_login.this,Forgot\_Password.class));

}

});

btnlog.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

final String Customerusername = edtusername.getText().toString();

final String Customerpassword = edtpassword.getText().toString();

if (Customerusername.isEmpty()) {

edtusername.setError("Please enter username");

}

else if (Customerpassword.isEmpty()) {

edtpassword.setError("Please enter email address");

}

else if(!Patterns.EMAIL\_ADDRESS.matcher(Customerusername).matches())

{

Snackbar snackbar=Snackbar.make(scrollView,"Such user does not exist!",Snackbar.LENGTH\_INDEFINITE);

snackbar.setDuration(5000);

snackbar.setTextColor(Color.WHITE);

snackbar.show();

snackbar.setBackgroundTint(Color.RED);

progressBar.setVisibility(View.INVISIBLE);

}

else

{

progressBar.setVisibility(View.VISIBLE);

mAuth.signInWithEmailAndPassword(Customerusername, Customerpassword)

.addOnCompleteListener(Customer\_login.this, new OnCompleteListener<AuthResult>() {

@Override

public void onComplete(@NonNull Task<AuthResult> task) {

if (task.isSuccessful()) {

if(mAuth.getCurrentUser().isEmailVerified())

{

FirebaseUser user = mAuth.getCurrentUser();

updateUI(user);

}

else

{

Snackbar snackbar=Snackbar.make(scrollView,"Please verify email",Snackbar.LENGTH\_INDEFINITE);

snackbar.setDuration(5000);

snackbar.setTextColor(Color.WHITE);

snackbar.show();

snackbar.setBackgroundTint(Color.RED);

progressBar.setVisibility(View.INVISIBLE); progressBar.setVisibility(View.INVISIBLE);

}

}

else

{

if(task.getException() instanceof FirebaseAuthInvalidUserException)

{

Snackbar snackbar=Snackbar.make(scrollView,"Invalid Username!",Snackbar.LENGTH\_INDEFINITE);

snackbar.setDuration(5000);

snackbar.setTextColor(Color.WHITE);

snackbar.show();

snackbar.setBackgroundTint(Color.RED);

progressBar.setVisibility(View.INVISIBLE);

}

else if(task.getException() instanceof FirebaseAuthInvalidCredentialsException)

{

Snackbar snackbar2=Snackbar.make(scrollView,"Invalid Password!",Snackbar.LENGTH\_INDEFINITE);

snackbar2.setDuration(5000);

snackbar2.setTextColor(Color.WHITE);

snackbar2.show();

snackbar2.setBackgroundTint(Color.RED);

progressBar.setVisibility(View.INVISIBLE);

}

}

}

});

}

}

});

txtsignup.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

edtusername.setText("");

edtpassword.setText("");

startActivity(new Intent(getApplicationContext(), Customer\_Registration.class));

finish();

}

});

}

private void updateUI(FirebaseUser user) {

Snackbar snackbar=Snackbar.make(scrollView,"Login successful",Snackbar.LENGTH\_INDEFINITE);

snackbar.setDuration(5000);

snackbar.setTextColor(Color.WHITE);

snackbar.show();

snackbar.setBackgroundTint(Color.GREEN);

progressBar.setVisibility(View.INVISIBLE);

edtusername.setText("");

edtpassword.setText("");

startActivity(new Intent(Customer\_login.this, Customer\_drawer.class));

finish();

}

}

* **Home Page**

package com.example.exploreworld.main;

import android.app.AlertDialog;

import android.app.Dialog;

import android.app.ProgressDialog;

import android.content.DialogInterface;

import android.content.Intent;

import android.graphics.Color;

import android.graphics.drawable.ColorDrawable;

import android.os.Bundle;

import android.os.Handler;

import android.view.Menu;

import android.view.MenuItem;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.ImageView;

import android.widget.RelativeLayout;

import android.widget.TextView;

import android.widget.Toast;

import com.bumptech.glide.Glide;

import com.example.exploreworld.Model.Customerdata;

import com.example.exploreworld.Model.PackageData;

import com.example.exploreworld.Model.Slider;

import com.example.exploreworld.R;

import com.example.exploreworld.adapters.DomesticAdapter;

import com.example.exploreworld.adapters.InternationalAdapter;

import com.example.exploreworld.adapters.PopularAdapter;

import com.example.exploreworld.adapters.ReligiousAdapter;

import com.example.exploreworld.adapters.Slideradapter;

import com.example.exploreworld.authentication.Admin\_Register;

import com.example.exploreworld.authentication.Admin\_login;

import com.example.exploreworld.authentication.Customer\_Registration;

import com.example.exploreworld.authentication.Customer\_login;

import com.example.exploreworld.authentication.Forgot\_Password;

import com.example.exploreworld.menu.Booking\_Detail;

import com.example.exploreworld.menu.Category\_Activity;

import com.example.exploreworld.menu.ContactUs\_Activity;

import com.example.exploreworld.menu.Facilities\_Activity;

import com.example.exploreworld.viewall.Domestic\_Viewall;

import com.example.exploreworld.viewall.Fort\_Load;

import com.example.exploreworld.viewall.International\_viewall;

import com.example.exploreworld.viewall.Popular\_viewall;

import com.example.exploreworld.viewall.Religious\_viewall;

import com.example.exploreworld.viewall.Wonders\_Load;

import com.google.android.gms.tasks.OnCompleteListener;

import com.google.android.gms.tasks.OnFailureListener;

import com.google.android.gms.tasks.Task;

import com.google.android.material.navigation.NavigationView;

import com.google.android.material.tabs.TabLayout;

import com.google.firebase.auth.AuthCredential;

import com.google.firebase.auth.EmailAuthCredential;

import com.google.firebase.auth.EmailAuthProvider;

import com.google.firebase.auth.FirebaseAuth;

import com.google.firebase.auth.FirebaseUser;

import com.google.firebase.database.DataSnapshot;

import com.google.firebase.database.DatabaseError;

import com.google.firebase.database.DatabaseReference;

import com.google.firebase.database.FirebaseDatabase;

import com.google.firebase.database.Query;

import com.google.firebase.database.ValueEventListener;

import com.squareup.picasso.Picasso;

import androidx.annotation.NonNull;

import androidx.appcompat.app.ActionBarDrawerToggle;

import androidx.appcompat.widget.SearchView;

import androidx.appcompat.widget.Toolbar;

import androidx.core.view.GravityCompat;

import androidx.core.view.MenuItemCompat;

import androidx.drawerlayout.widget.DrawerLayout;

import androidx.appcompat.app.AppCompatActivity;

import androidx.recyclerview.widget.GridLayoutManager;

import androidx.recyclerview.widget.LinearLayoutManager;

import androidx.recyclerview.widget.RecyclerView;

import androidx.viewpager.widget.ViewPager;

import java.util.ArrayList;

import java.util.List;

import java.util.Timer;

import java.util.TimerTask;

import de.hdodenhof.circleimageview.CircleImageView;

import dmax.dialog.SpotsDialog;

public class Customer\_drawer extends AppCompatActivity implements NavigationView.OnNavigationItemSelectedListener {

DrawerLayout drawer;

NavigationView navigationView;

private ViewPager mainPager;

Slideradapter slideradapter;

TabLayout tabIndicator;

private List<Slider> mList = new ArrayList<>();

private int currentposition = 0;

private Timer timer;

TextView textView1,textView2,textView3,textView4;

Dialog dialog,dialog2;

List<PackageData> mydomesticlist,myinternationallist,mypopularlist,myreligiouslist;

PackageData packageData;

DomesticAdapter domesticAdapter;

InternationalAdapter internationalAdapter;

PopularAdapter popularAdapter;

ReligiousAdapter religiousAdapter;

RecyclerView recyclerView,recyclerView2,recyclerView3,recyclerView4;

private DatabaseReference databaseReference,databaseReference2,databaseReference3,databaseReference4;

private ValueEventListener valueEventListener,valueEventListener2;

ProgressDialog alertDialog;

ImageView wonders,fort,ladakh;

private long backPressedTime;

private Toast backToast;

FirebaseAuth firebaseAuth;

FirebaseUser firebaseUser;

FirebaseUser user;

FirebaseDatabase firebaseDatabase;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_customer\_drawer);

wonders=findViewById(R.id.imagepic1);

fort=findViewById(R.id.imagepic3);

textView1=findViewById(R.id.viewall);

textView2=findViewById(R.id.viewall2);

textView3=findViewById(R.id.viewall3);

textView4=findViewById(R.id.viewall4);

dialog=new Dialog(Customer\_drawer.this);

dialog2=new Dialog(Customer\_drawer.this);

Toolbar toolbar = findViewById(R.id.toolbar);

setSupportActionBar(toolbar);

drawer= findViewById(R.id.drawer\_layout);

navigationView = findViewById(R.id.nav\_view);

navigationView.setNavigationItemSelectedListener(Customer\_drawer.this);

ActionBarDrawerToggle toggle=new ActionBarDrawerToggle(Customer\_drawer.this,drawer,toolbar

,R.string.navigation\_drawer\_open,R.string.navigation\_drawer\_close);

drawer.addDrawerListener(toggle);

toggle.syncState();

if(savedInstanceState==null)

{

navigationView.setCheckedItem(R.id.nav\_home);

}

recyclerView=findViewById(R.id.recycle\_view);

recyclerView2=findViewById(R.id.recycle\_view2);

recyclerView3=findViewById(R.id.recycle\_view3);

recyclerView4=findViewById(R.id.recycle\_view4);

//slider

mList.add(new Slider("Best Tourist Places", "Travel makes one modest, you see what a tiny place you occupy in the world.", "https://imagesvc.meredithcorp.io/v3/mm/image?url=https%3A%2F%2Fstatic.onecms.io%2Fwp-content%2Fuploads%2Fsites%2F28%2F2018%2F10%2Fhot-air-balloons-festival-cappadocia-turkey-HOTAIR0605.jpg&amp;q=85"));

mList.add(new Slider("Deligious food", "More than 250 restaurant around the tourist places, ", "https://discover.luxury/wp-content/uploads/2017/09/The-World%E2%80%99s-Most-Exclusive-Events-for-Food-and-Wine-Lovers-feature-1024x581.jpg"));

mList.add(new Slider("Best activities", "Extraordinary five-star outdoor activities","https://wildhawk.in/wp-content/uploads/2019/05/Rafting\_in\_rishikesh-850x487.jpg"));

mList.add(new Slider("Great luxurious hotels", "Book cheapest and most convenient hotels around you.", "https://rimghtlak.mmtcdn.com/CollectionImages/201609091139127239/3\_woobar.jpg"));

// setup viewpager

mainPager = findViewById(R.id.imageSlider);

tabIndicator = findViewById(R.id.Tab);

slideradapter = new Slideradapter(Customer\_drawer.this, mList);

mainPager.setAdapter(slideradapter);

createSlideShow();

String uri="https://images.memphistours.com/large/a3681e58ff971daf48ea05e717a21dca.jpg";

String uri2="https://i.pinimg.com/originals/d7/fe/f4/d7fef4f1954a984f999996ff5f9aaf45.jpg";

Glide.with(this).load(uri).into(wonders);

Glide.with(this).load(uri2).into(fort);

// setup tablayout with viewpager

alertDialog=new ProgressDialog(Customer\_drawer.this);

alertDialog.setMessage("Loading Packages...");

alertDialog.setCanceledOnTouchOutside(false);

tabIndicator.setupWithViewPager(mainPager);

mydomesticlist=new ArrayList<>();

myinternationallist=new ArrayList<>();

mypopularlist=new ArrayList<>();

myreligiouslist=new ArrayList<>();

domesticAdapter=new DomesticAdapter(this,mydomesticlist);

internationalAdapter=new InternationalAdapter(this,myinternationallist);

popularAdapter=new PopularAdapter(this,mypopularlist);

religiousAdapter=new ReligiousAdapter(this,myreligiouslist);

recyclerView.setAdapter(domesticAdapter);

recyclerView2.setAdapter(internationalAdapter);

recyclerView3.setAdapter(popularAdapter);

recyclerView4.setAdapter(religiousAdapter);

LinearLayoutManager gridLayoutManager = new LinearLayoutManager(this, LinearLayoutManager.HORIZONTAL, false);

recyclerView.setLayoutManager(gridLayoutManager);

LinearLayoutManager gridLayoutManager2 = new LinearLayoutManager(this, LinearLayoutManager.HORIZONTAL, false);

recyclerView2.setLayoutManager(gridLayoutManager2);

LinearLayoutManager gridLayoutManager3= new LinearLayoutManager(this, LinearLayoutManager.HORIZONTAL, false);

recyclerView3.setLayoutManager(gridLayoutManager3);

LinearLayoutManager gridLayoutManager4 = new LinearLayoutManager(this, LinearLayoutManager.HORIZONTAL, false);

recyclerView4.setLayoutManager(gridLayoutManager4);

//domestic package

alertDialog.show();

databaseReference= FirebaseDatabase.getInstance().getReference("tourpackages").child("domesticpackage");

Query query1=databaseReference.limitToFirst(2);

valueEventListener=query1.addValueEventListener(new ValueEventListener() {

@Override

public void onDataChange(@NonNull DataSnapshot dataSnapshot) {

mydomesticlist.clear();

for (DataSnapshot snapshot : dataSnapshot.getChildren())

{

PackageData packageData=snapshot.getValue(PackageData.class);

mydomesticlist.add(packageData);

}

domesticAdapter.notifyDataSetChanged();

}

@Override

public void onCancelled(@NonNull DatabaseError databaseError) {

alertDialog.dismiss();

}

});

databaseReference2=FirebaseDatabase.getInstance().getReference("tourpackages").child("internationalpackage");

Query query2=databaseReference2.limitToFirst(1);

valueEventListener=query2.addValueEventListener(new ValueEventListener() {

@Override

public void onDataChange(@NonNull DataSnapshot dataSnapshot) {

myinternationallist.clear();

for(DataSnapshot snapshot:dataSnapshot.getChildren())

{

PackageData packageData=snapshot.getValue(PackageData.class);

myinternationallist.add(packageData);

}

internationalAdapter.notifyDataSetChanged();

}

@Override

public void onCancelled(@NonNull DatabaseError databaseError) {

alertDialog.dismiss();

}

});

databaseReference3=FirebaseDatabase.getInstance().getReference("tourpackages").child("popularpackage");

Query query3=databaseReference3.limitToFirst(1);

valueEventListener=query3.addValueEventListener(new ValueEventListener() {

@Override

public void onDataChange(@NonNull DataSnapshot dataSnapshot) {

mypopularlist.clear();

for(DataSnapshot snapshot:dataSnapshot.getChildren())

{

PackageData packageData=snapshot.getValue(PackageData.class);

mypopularlist.add(packageData);

}

popularAdapter.notifyDataSetChanged();

}

@Override

public void onCancelled(@NonNull DatabaseError databaseError) {

alertDialog.dismiss();

}

});

databaseReference4=FirebaseDatabase.getInstance().getReference("tourpackages").child("religiouspackage");

Query query4=databaseReference4.limitToFirst(1);

valueEventListener=query4.addValueEventListener(new ValueEventListener() {

@Override

public void onDataChange(@NonNull DataSnapshot dataSnapshot) {

myreligiouslist.clear();

for(DataSnapshot snapshot:dataSnapshot.getChildren())

{

PackageData packageData=snapshot.getValue(PackageData.class);

myreligiouslist.add(packageData);

}

religiousAdapter.notifyDataSetChanged();

alertDialog.dismiss();

}

@Override

public void onCancelled(@NonNull DatabaseError databaseError) {

alertDialog.dismiss();

}

});

textView1.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

startActivity(new Intent(getApplicationContext(), Domestic\_Viewall.class));

}

});

textView2.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

startActivity(new Intent(getApplicationContext(), International\_viewall.class));

}

});

textView3.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

startActivity(new Intent(getApplicationContext(), Popular\_viewall.class));

}

});

textView4.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

startActivity(new Intent(getApplicationContext(), Religious\_viewall.class));

}

});

wonders.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

startActivity(new Intent(getApplicationContext(), Wonders\_Load.class));

}

});

fort.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

startActivity(new Intent(getApplicationContext(), Fort\_Load.class));

}

});

}

private void createSlideShow()

{

final Handler handler=new Handler();

final Runnable runnable=new Runnable()

{

@Override

public void run() {

if(currentposition==mList.size())

currentposition=0;

mainPager.setCurrentItem(currentposition++,true);

}

};

timer=new Timer();

timer.schedule(new TimerTask() {

@Override

public void run()

{

handler.post(runnable);

}

},900,4000);

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

getMenuInflater().inflate(R.menu.search\_view,menu);

return super.onCreateOptionsMenu(menu);

}

@Override

public boolean onOptionsItemSelected(@NonNull MenuItem item) {

switch (item.getItemId())

{

case R.id.item1:

showpopup();

break;

case R.id.item2:

FirebaseAuth.getInstance().signOut();

finish();

startActivity(new Intent(getApplicationContext(),Admin\_login.class));

break;

case R.id.item3:

showreport();

break;

}

return super.onOptionsItemSelected(item);

}

private void showreport() {

TextView closetxt;

final EditText editText;

Button btnsend;

dialog2.setContentView(R.layout.report\_layout);

closetxt=(TextView)dialog2.findViewById(R.id.reportclose);

editText=(EditText)dialog2.findViewById(R.id.typereport);

btnsend=(Button) dialog2.findViewById(R.id.reportbtn);

dialog2.getWindow().setBackgroundDrawable(new ColorDrawable(Color.TRANSPARENT));

dialog2.show();

closetxt.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

dialog2.dismiss();

}

});

btnsend.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

String str=editText.getText().toString();

if(str.isEmpty())

{

editText.setText("Type something");

}

else {

Intent i=new Intent(Intent.ACTION\_SEND);

i.putExtra(Intent.EXTRA\_EMAIL,new String[]{"rsprojects3470@gmail.com"});

i.putExtra(Intent.EXTRA\_SUBJECT,"Explore App report");

i.putExtra(Intent.EXTRA\_TEXT,str);

i.setType("message/rfc822");

startActivity(Intent.createChooser(i,"Select Email App "));

dialog2.dismiss();

}

}

});

}

private void showpopup() {

final TextView txtclose, txtname, txtemail, txtphone, txtreset, txtdelete;

final CircleImageView circleImageView;

Button btnsignout;

dialog.setContentView(R.layout.profile\_layout);

txtclose = (TextView) dialog.findViewById(R.id.closetxt);

txtdelete = (TextView) dialog.findViewById(R.id.deleteact);

txtreset = (TextView) dialog.findViewById(R.id.resetpassword);

txtemail = (TextView) dialog.findViewById(R.id.custemail);

txtname = (TextView) dialog.findViewById(R.id.custname);

txtphone = (TextView) dialog.findViewById(R.id.custphone);

circleImageView = (CircleImageView) dialog.findViewById(R.id.custprofile);

btnsignout = (Button) dialog.findViewById(R.id.signoutact);

firebaseDatabase = FirebaseDatabase.getInstance();

user = FirebaseAuth.getInstance().getCurrentUser();

final String link=user.getEmail();

if (user != null) {

DatabaseReference reference = firebaseDatabase.getReference("User").child("customer").child(user.getUid());

reference.addValueEventListener(new ValueEventListener() {

@Override

public void onDataChange(@NonNull DataSnapshot dataSnapshot) {

Customerdata customerdata = dataSnapshot.getValue(Customerdata.class);

txtname.setText(customerdata.getFullname());

txtemail.setText(customerdata.getUsername());

txtphone.setText(customerdata.getPhone());

Picasso.get().load(customerdata.getImageuri()).into(circleImageView);

}

@Override

public void onCancelled(@NonNull DatabaseError databaseError) {

Toast.makeText(Customer\_drawer.this, databaseError.getCode(), Toast.LENGTH\_SHORT).show();

}

});

}

else

{

Toast.makeText(Customer\_drawer.this, "Please register", Toast.LENGTH\_SHORT).show();

}

txtclose.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

dialog.dismiss();

}

});

txtreset.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

if(user!=null) {

dialog.dismiss();

final EditText reset=new EditText(view.getContext());

final AlertDialog.Builder builder=new AlertDialog.Builder(view.getContext());

builder.setTitle("Reset Password");

builder.setMessage("Enter a New password");

builder.setView(reset);

builder.setPositiveButton("Yes", new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialogInterface, int i) {

String pass=reset.getText().toString();

user.updatePassword(pass).addOnCompleteListener(new OnCompleteListener<Void>() {

@Override

public void onComplete(@NonNull Task<Void> task) {

Toast.makeText(getApplicationContext(), "Reset Link send to your email", Toast.LENGTH\_SHORT).show();

startActivity(new Intent(getApplicationContext(), Customer\_login.class));

finish();

}

}).addOnFailureListener(new OnFailureListener() {

@Override

public void onFailure(@NonNull Exception e) {

Toast.makeText(getApplicationContext(), "Error ! mail is not sent" + e.getMessage(), Toast.LENGTH\_SHORT).show();

}

});

}

});

builder.setNegativeButton("No", new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialogInterface, int i) {

dialogInterface.dismiss();

}

});

builder.create().show();

}

}

});

btnsignout.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

dialog.dismiss();

FirebaseAuth.getInstance().signOut();

finish();

startActivity(new Intent(getApplicationContext(), MainActivity.class));

}

});

dialog.getWindow().setBackgroundDrawable(new ColorDrawable(Color.TRANSPARENT));

dialog.show();

}

@Override

public void onBackPressed() {

if(drawer.isDrawerOpen(GravityCompat.START))

{

drawer.closeDrawer(GravityCompat.START);

}

else {

if(backPressedTime+1000>System.currentTimeMillis()) {

backToast.cancel();

super.onBackPressed();

return;

}

else

{

backToast=Toast.makeText(Customer\_drawer.this, "Press again to exit", Toast.LENGTH\_SHORT);

backToast.show();

}

backPressedTime = System.currentTimeMillis();

}

}

@Override

public boolean onNavigationItemSelected(@NonNull MenuItem menuItem) {

switch (menuItem.getItemId())

{

case R.id.nav\_flight:

startActivity(new Intent(Customer\_drawer.this, Admin\_drawer.class));

finish();

break;

case R.id.nav\_facilities:

startActivity(new Intent(Customer\_drawer.this, Facilities\_Activity.class));

finish();

break;

case R.id.nav\_Booking:

startActivity(new Intent(Customer\_drawer.this, Booking\_Detail.class));

finish();

break;

case R.id.nav\_contactus:

startActivity(new Intent(Customer\_drawer.this, ContactUs\_Activity.class));

finish();

break;

}

drawer.closeDrawer(GravityCompat.START);

return true;

}

}

* **Package Details**

package com.example.exploreworld.firebasepackageload;

import androidx.annotation.NonNull;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.ImageView;

import android.widget.RatingBar;

import android.widget.TextView;

import android.widget.Toast;

import com.bumptech.glide.Glide;;

import com.example.exploreworld.Model.PackageRating;

import com.example.exploreworld.Model.PackageFeedback;

import com.example.exploreworld.R;

import com.google.android.gms.tasks.OnCompleteListener;

import com.google.android.gms.tasks.OnFailureListener;

import com.google.android.gms.tasks.Task;

import com.google.firebase.auth.FirebaseAuth;

import com.google.firebase.auth.FirebaseUser;

import com.google.firebase.database.DataSnapshot;

import com.google.firebase.database.DatabaseError;

import com.google.firebase.database.DatabaseReference;

import com.google.firebase.database.FirebaseDatabase;

import com.google.firebase.database.Query;

import com.google.firebase.database.ValueEventListener;

import com.stepstone.apprating.AppRatingDialog;

import com.stepstone.apprating.listener.RatingDialogListener;

import net.cachapa.expandablelayout.ExpandableLayout;

import java.text.DecimalFormat;

import java.util.Arrays;

public class Package\_details extends AppCompatActivity implements RatingDialogListener {

TextView tvtitle,tvsubtitle,tvdescription,tvtips,tvday,tvprice,tvratetxt,tvact1,tvact2,tvact3,tvact4;

private Button btnbook;

ImageView img1,img2,img3,img4,img5,img6,img7,img8,img9;

RatingBar ratingBar;

public static String Title="title";

public static String Location="location";

public static String Price="price";

public static String Daynight="dayngt";

public static String Description="description";

public static String Tips="Tips";

public static String Image="image";

public static String Image1="image1";

public static String Image2="image2";

public static String Image3="image3";

public static String Image4="image4";

public static String Activity1="activity1";

public static String Activity2="activity2";

public static String Activity3="activity3";

public static String Activity4="activity4";

public static String Activitytitle1="activitytitle1";

public static String Activitytitle2="activitytitle2";

public static String Activitytitle3="activitytitle3";

public static String Activitytitle4="activitytitle4";

public static String Resortimage="resortimage";

public static String Resorttitle="resorttitle";

public static String Resortlocation="resortlocation";

public static String Resortdescription="resortdescription";

public static String Resortprice="resortprice";

public static String semiResortimage="semiresortimage";

public static String semiResorttitle="semiresorttitle";

public static String semiResortlocation="semiresortlocation";

public static String semiResortdescription="semiresortdescription";

public static String semiResortprice="semiresortprice";

TextView expand;

ExpandableLayout expandableLayout;

FirebaseAuth firebaseAuth;

FirebaseUser firebaseUser;

DatabaseReference databaseReference,ratingdatabase,feedbackdatabase;

FirebaseDatabase database;

String strrating;

float avg,sum=0;

String valueitem;

String title,subtitle,price,dayngt,act1,act2,act3,act4,image,image1,image2,image3,image4,actimg1,actimg2,actimg3,actimg4,description,tips,strdulexprice,strsemiprice,strresortimage,strresorttitle,strresortlocation,strresortdescription,strsemiresortimage,strsemiresorttitle,strsemiresortlocation,strsemiresortdescription;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_package\_details);

expand=findViewById(R.id.expand\_tourtips);

expandableLayout=findViewById(R.id.myexpandable);

ratingBar=findViewById(R.id.detailrate);

tvtitle=findViewById(R.id.detailtitle);

tvsubtitle=findViewById(R.id.detailsubtitle);

tvdescription=findViewById(R.id.detaildescription);

tvprice=findViewById(R.id.detailprice);

tvday=findViewById(R.id.detaildn);

tvtips=findViewById(R.id.detailtips);

tvratetxt=findViewById(R.id.detailratetxt);

tvact1=findViewById(R.id.activity1);

tvact2=findViewById(R.id.activity2);

tvact3=findViewById(R.id.activity3);

tvact4=findViewById(R.id.activity4);

img1=findViewById(R.id.detailimage1);

img2=findViewById(R.id.detailimage2);

img3=findViewById(R.id.detailimage3);

img4=findViewById(R.id.detailimage4);

img5=findViewById(R.id.detailimage5);

img6=findViewById(R.id.detailimage6);

img7=findViewById(R.id.detailimage7);

img8=findViewById(R.id.detailimage8);

img9=findViewById(R.id.detailimage9);

btnbook=findViewById(R.id.btnBook);

firebaseAuth=FirebaseAuth.getInstance();

firebaseUser=firebaseAuth.getCurrentUser();

database=FirebaseDatabase.getInstance();

ratingdatabase=database.getReference("Rating").child("packagerating");

feedbackdatabase=database.getReference("Feedbacks");

title=getIntent().getStringExtra(Title);

subtitle=getIntent().getStringExtra(Location);

price=getIntent().getStringExtra(Price);

dayngt=getIntent().getStringExtra(Daynight);

description=getIntent().getStringExtra(Description);

act1=getIntent().getStringExtra(Activitytitle1);

act2=getIntent().getStringExtra(Activitytitle2);

act3=getIntent().getStringExtra(Activitytitle3);

act4=getIntent().getStringExtra(Activitytitle4);

tips=getIntent().getStringExtra(Tips);

image=getIntent().getStringExtra(Image);

image1=getIntent().getStringExtra(Image1);

image2=getIntent().getStringExtra(Image2);

image3=getIntent().getStringExtra(Image3);

image4=getIntent().getStringExtra(Image4);

actimg1=getIntent().getStringExtra(Activity1);

actimg2=getIntent().getStringExtra(Activity2);

actimg3=getIntent().getStringExtra(Activity3);

actimg4=getIntent().getStringExtra(Activity4);

strresortimage=getIntent().getStringExtra(Resortimage);

strresorttitle=getIntent().getStringExtra(Resorttitle);

strresortlocation=getIntent().getStringExtra(Resortlocation);

strresortdescription=getIntent().getStringExtra(Resortdescription);

strdulexprice=getIntent().getStringExtra(Resortprice);

strsemiresorttitle=getIntent().getStringExtra(semiResorttitle);

strsemiresortimage=getIntent().getStringExtra(semiResortimage);

strsemiresortlocation=getIntent().getStringExtra(semiResortlocation);

strsemiresortdescription=getIntent().getStringExtra(semiResortdescription);

strsemiprice=getIntent().getStringExtra(semiResortprice);

tvtitle.setText(title);

tvsubtitle.setText(subtitle);

tvprice.setText(price);

tvday.setText(dayngt);

tvdescription.setText(description);

tvtips.setText(tips);

tvact1.setText(act1);

tvact2.setText(act2);

tvact3.setText(act3);

tvact4.setText(act4);

Query query=ratingdatabase.orderByChild("packageId").equalTo(title);

query.addValueEventListener(new ValueEventListener() {

@Override

public void onDataChange(@NonNull DataSnapshot dataSnapshot) {

for(DataSnapshot snapshot:dataSnapshot.getChildren())

{

PackageRating item=snapshot.getValue(PackageRating.class);

valueitem=item.getRatevalue();

tvratetxt.setText(valueitem);

ratingBar.setRating(Float.parseFloat(valueitem));

}

}

@Override

public void onCancelled(@NonNull DatabaseError databaseError) {

}

});

Glide.with(this).load(image).into(img1);

Glide.with(this).load(image1).into(img2);

Glide.with(this).load(image2).into(img3);

Glide.with(this).load(image3).into(img4);

Glide.with(this).load(image4).into(img5);

Glide.with(this).load(actimg1).into(img6);

Glide.with(this).load(actimg2).into(img7);

Glide.with(this).load(actimg3).into(img8);

Glide.with(this).load(actimg4).into(img9);

expand.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

expandableLayout.toggle();

}

});

tvratetxt.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

showRatingDialog();

}

});

btnbook.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

Intent i=new Intent(getApplicationContext(), Choose\_resort.class);

i.putExtra("package title",title);

i.putExtra("title image",image);

i.putExtra("Price",price);

i.putExtra("resort Title",strresorttitle);

i.putExtra("Resort location",strresortlocation);

i.putExtra("Resort description",strresortdescription);

i.putExtra("Resort price",strdulexprice);

i.putExtra("Resort Image",strresortimage);

i.putExtra("Resort semi name",strsemiresorttitle);

i.putExtra("Resort semi location",strsemiresortlocation);

i.putExtra("Resort semi description",strsemiresortdescription);

i.putExtra("Resort semi price",strsemiprice);

i.putExtra("Resort semi Image",strsemiresortimage);

i.putExtra("Day night",dayngt);

startActivity(i);

}

});

}

private void showRatingDialog() {

new AppRatingDialog.Builder()

.setPositiveButtonText("Submit")

.setNegativeButtonText("Cancel")

.setNoteDescriptions(Arrays.asList("Very Bad", "Not good", "Quite ok", "Very Good", "Excellent !!!"))

.setDefaultRating(1)

.setTitle("Rate this Package")

.setDescription("Please select some stars and give your feedback")

.setCommentInputEnabled(true)

.setNoteDescriptionTextColor(R.color.noteDescriptionTextColor)

.setTitleTextColor(R.color.titleTextColor)

.setDescriptionTextColor(R.color.contentTextColor)

.setHint("Please write your comment here ...")

.setHintTextColor(R.color.hintTextColor)

.setCommentTextColor(R.color.commentTextColor)

.setCommentBackgroundColor(R.color.commentbg)

.setWindowAnimation(R.style.RatingDialogFadeAnim)

.setCancelable(false)

.setCanceledOnTouchOutside(false)

.create(Package\_details.this)// only if listener is implemented by fragment

.show();

}

@Override

public void onNegativeButtonClicked() {

}

@Override

public void onNeutralButtonClicked() {

}

@Override

public void onPositiveButtonClicked(int i, String s) {

sum= i + Float.parseFloat(valueitem);

avg=sum/2;

final PackageRating packageRating = new PackageRating(firebaseUser.getEmail(),

title, String.valueOf(new DecimalFormat("#.#").format(avg)));

PackageFeedback packageFeedback=new PackageFeedback(firebaseUser.getEmail(),title,s);

ratingdatabase.child("packagerating").child(title).addListenerForSingleValueEvent(new ValueEventListener() {

@Override

public void onDataChange(@NonNull DataSnapshot dataSnapshot) {

if (dataSnapshot.child(title).exists()) {

//Remove old value

ratingdatabase.child(title).removeValue();

//update new value

ratingdatabase.child(title).setValue(packageRating);

} else {

//update new Value

ratingdatabase.child(title).setValue(packageRating);

}

}

@Override

public void onCancelled(@NonNull DatabaseError databaseError) {

}

});

feedbackdatabase.child(title).child(firebaseUser.getUid()).setValue(packageFeedback)

.addOnCompleteListener(new OnCompleteListener<Void>() {

@Override

public void onComplete(@NonNull Task<Void> task) {

if (task.isSuccessful()) {

Toast.makeText(Package\_details.this, "Thank you for rating & feedback", Toast.LENGTH\_SHORT).show();

} else {

Toast.makeText(Package\_details.this, "sorry for rating & feedback is not submitted", Toast.LENGTH\_SHORT).show();

}

}

}).addOnFailureListener(new OnFailureListener() {

@Override

public void onFailure(@NonNull Exception e) {

Toast.makeText(Package\_details.this, e.getMessage(), Toast.LENGTH\_SHORT).show();

}

});

}

**XML CODE:**

* **Uploading**

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

<ScrollView

xmlns:android="http://schemas.android.com/apk/res/android"

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="170dp"

android:background="@drawable/pager1">

<androidx.appcompat.widget.Toolbar

android:layout\_width="match\_parent"

android:layout\_height="?attr/actionBarSize"

android:background="@drawable/toolbar\_bg"

app:title="Travel Hunt"

app:titleMarginStart="10dp"

app:titleTextColor="#134FE4"

android:layout\_margin="5dp"

android:id="@+id/toolbar"/>

</RelativeLayout>

<androidx.viewpager.widget.ViewPager

android:id="@+id/imageSlider"

android:layout\_width="match\_parent"

android:layout\_marginTop="70dp"

android:layout\_marginRight="10dp"

android:layout\_marginLeft="10dp"

android:layout\_height="180dp" />

<com.google.android.material.tabs.TabLayout

android:id="@+id/Tab"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="200dp"

app:tabBackground="@drawable/circle\_holder"

app:tabGravity="center"

app:tabIndicatorHeight="0dp"

app:tabMaxWidth="10dp" />

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

android:layout\_below="@+id/imageSlider">

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

>

<TextView

android:id="@+id/domestic"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentStart="true"

android:layout\_gravity="center\_vertical"

android:layout\_marginStart="5dp"

android:layout\_marginEnd="12dp"

android:layout\_toStartOf="@+id/viewall"

android:text="Domestic Tours"

android:textColor="#2A2A2B"

android:textSize="18sp"/>

<TextView

android:id="@+id/viewall"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentEnd="true"

android:layout\_marginEnd="8dp"

android:text="View all"

android:textColor="#1271BC"

android:textSize="15sp"/>

</RelativeLayout>

<androidx.recyclerview.widget.RecyclerView

android:id="@+id/recycle\_view"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:orientation="horizontal" />

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

>

<TextView

android:id="@+id/International"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentStart="true"

android:layout\_gravity="center\_vertical"

android:layout\_marginEnd="12dp"

android:layout\_toStartOf="@+id/viewall2"

android:text="International Tours"

android:layout\_marginStart="5dp"

android:textColor="#2A2A2B"

android:textSize="18sp"/>

<TextView

android:id="@+id/viewall2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentEnd="true"

android:layout\_marginEnd="8dp"

android:text="View all"

android:textColor="#1271BC"

android:textSize="15sp"/>

</RelativeLayout>

<androidx.recyclerview.widget.RecyclerView

android:id="@+id/recycle\_view2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:orientation="horizontal" />

<HorizontalScrollView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:scrollbars="none">

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:orientation="horizontal">

<androidx.cardview.widget.CardView

android:layout\_width="260dp"

android:layout\_height="170dp"

android:id="@+id/cardview2"

android:layout\_marginStart="5dp"

android:layout\_marginTop="10dp"

android:layout\_marginBottom="10dp"

android:layout\_marginEnd="10dp"

app:cardElevation="5dp"

app:cardCornerRadius="20dp">

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<ImageView

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:scaleType="fitXY"

android:id="@+id/imagepic1"

/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="15dp"

android:layout\_marginTop="135dp"

android:text="Seven Wonders"

android:padding="4dp"

android:textColor="#FFFFFF"

android:textSize="16sp"

android:textStyle="bold" />

</RelativeLayout>

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView

android:layout\_width="260dp"

android:layout\_height="170dp"

android:id="@+id/cardview1"

android:layout\_margin="10dp"

app:cardElevation="5dp"

app:cardCornerRadius="20dp">

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<ImageView

android:id="@+id/imagepic2"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:scaleType="fitXY"

/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="15dp"

android:layout\_marginTop="135dp"

android:text="Ladakh"

android:padding="4dp"

android:textColor="#FFFFFF"

android:textSize="16sp"

android:textStyle="bold" />

</RelativeLayout>

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView

android:layout\_width="260dp"

android:layout\_height="170dp"

android:id="@+id/cardview3"

android:layout\_margin="10dp"

app:cardElevation="5dp"

app:cardCornerRadius="20dp">

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<ImageView

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:scaleType="fitXY"

android:id="@+id/imagepic3"

/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="15dp"

android:layout\_marginTop="135dp"

android:text="Forts View"

android:padding="4dp"

android:textColor="#FFFFFF"

android:textSize="16sp"

android:textStyle="bold" />

</RelativeLayout>

</androidx.cardview.widget.CardView>

</LinearLayout>

</HorizontalScrollView>

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content">

<TextView

android:id="@+id/Popular"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentStart="true"

android:layout\_gravity="center\_vertical"

android:layout\_marginEnd="12dp"

android:layout\_toStartOf="@+id/viewall3"

android:text="Popular Tours"

android:layout\_marginStart="5dp"

android:textColor="#2A2A2B"

android:textSize="18sp" />

<TextView

android:id="@+id/viewall3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentEnd="true"

android:layout\_marginEnd="8dp"

android:text="View all"

android:textColor="#1271BC"

android:textSize="15sp"/>

</RelativeLayout>

<androidx.recyclerview.widget.RecyclerView

android:id="@+id/recycle\_view3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:orientation="horizontal"/>

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

>

<TextView

android:id="@+id/Other"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentStart="true"

android:layout\_gravity="center\_vertical"

android:layout\_marginEnd="12dp"

android:layout\_toStartOf="@+id/viewall4"

android:text="Religious Tours"

android:layout\_marginStart="5dp"

android:textColor="#2A2A2B"

android:textSize="18sp" />

<TextView

android:id="@+id/viewall4"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentEnd="true"

android:layout\_marginEnd="8dp"

android:text="View all"

android:textColor="#1271BC"

android:textSize="15sp"/>

</RelativeLayout>

<androidx.recyclerview.widget.RecyclerView

android:id="@+id/recycle\_view4"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:orientation="horizontal"/>

</LinearLayout>

</RelativeLayout>

</ScrollView>

* **Package Details:**

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

<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:scrollbars="none"

android:background="#FFF"

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

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<androidx.cardview.widget.CardView

android:id="@+id/linear"

android:layout\_width="match\_parent"

android:layout\_height="300dp"

android:layout\_marginTop="-20dp"

android:background="@drawable/detail\_img\_bg"

android:orientation="vertical"

app:cardCornerRadius="18dp">

<ImageView

android:id="@+id/detailimage1"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:scaleType="fitXY"

android:textAlignment="center" />

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView

android:layout\_width="60dp"

android:layout\_height="30dp"

android:layout\_alignParentEnd="true"

android:layout\_marginTop="262dp"

android:layout\_marginEnd="40dp"

android:padding="5dp"

app:cardCornerRadius="8dp">

<TextView

android:id="@+id/detaildn"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="#4359FA"

android:gravity="center"

android:text="3D/4N"

android:textAlignment="center"

android:textColor="#FFFFFF"

android:textSize="14sp"

android:textStyle="bold" />

</androidx.cardview.widget.CardView>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_below="@+id/linear"

android:background="#FFF"

android:orientation="vertical"

android:paddingStart="20dp">

<RelativeLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content">

<TextView

android:id="@+id/detailtitle"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:padding="5dp"

android:text="South Island"

android:textColor="#151515"

android:textSize="18sp"

android:textStyle="bold" />

<TextView

android:id="@+id/detailsubtitle"

android:layout\_width="200dp"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/detailtitle"

android:drawableStart="@drawable/location\_icon"

android:text="South Africa"

android:textColor="#4776e6"

android:textSize="14sp" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/detailsubtitle"

android:layout\_marginRight="2dp"

android:layout\_toLeftOf="@+id/detailprice"

android:text="₹"

android:textAlignment="center"

android:textColor="#4776e6"

android:textSize="18sp"

android:textStyle="bold" />

<TextView

android:id="@+id/detailprice"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/detailsubtitle"

android:layout\_toLeftOf="@id/pricetag"

android:text="250"

android:textAlignment="center"

android:textColor="#4776e6"

android:textSize="18sp"

android:textStyle="bold" />

<TextView

android:id="@+id/pricetag"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/detailsubtitle"

android:layout\_alignParentEnd="true"

android:layout\_marginRight="15dp"

android:text="/-"

android:textAlignment="center"

android:textColor="#4776e6"

android:textSize="18sp"

android:textStyle="bold" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/detailprice"

android:layout\_alignParentEnd="true"

android:layout\_marginTop="-4dp"

android:text="Per person"

android:textColor="#C3C4C5"

android:textSize="10sp"

android:textStyle="italic" />

<RatingBar

android:id="@+id/detailrate"

style="?android:attr/ratingBarStyleSmall"

android:layout\_width="91dp"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/detailsubtitle"

android:isIndicator="true"

android:numStars="5"

android:padding="5dp"

android:rating="0"

android:stepSize="0.5"

android:theme="@style/Ratingbar" />

<TextView

android:id="@+id/detailratetxt"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/detailsubtitle"

android:layout\_toRightOf="@+id/detailrate"

android:padding="5dp"

android:text="(4.8)"

android:textColor="#4776e6"

android:textSize="12sp"

android:textStyle="bold" />

</RelativeLayout>

<TextView

android:id="@+id/detaildescription"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:elevation="5dp"

android:lineSpacingExtra="2dp"

android:paddingEnd="5dp"

android:text=" From the heights of Table Mountain to the dramatic stretches of coastline, Cape Town's wonderful weather year-round makes it a popular destination for beach lovers and wine aficionados alike. Its world-famous Kirstenbosch Botanical Gardens and Green Point Park compliment these natural blessings wonderfully."

android:textColor="#B0AFB1"

android:textSize="12sp"

android:textStyle="bold" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:padding="5dp"

android:text="Photos"

android:textColor="#171717"

android:textSize="16sp"

android:textStyle="bold" />

<HorizontalScrollView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:scrollbars="none">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5dp"

android:gravity="center"

android:orientation="horizontal">

<androidx.cardview.widget.CardView

android:layout\_width="80dp"

android:layout\_height="80dp"

android:elevation="3dp"

android:padding="5dp"

app:cardCornerRadius="12dp">

<ImageView

android:id="@+id/detailimage2"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:scaleType="fitXY" />

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView

android:layout\_width="80dp"

android:layout\_height="80dp"

android:layout\_marginStart="20dp"

android:elevation="3dp"

android:padding="5dp"

app:cardCornerRadius="12dp">

<ImageView

android:id="@+id/detailimage3"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:scaleType="fitXY" />

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView

android:layout\_width="80dp"

android:layout\_height="80dp"

android:layout\_marginStart="20dp"

android:elevation="3dp"

android:padding="5dp"

app:cardCornerRadius="12dp">

<ImageView

android:id="@+id/detailimage4"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:scaleType="fitXY" />

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView

android:layout\_width="80dp"

android:layout\_height="80dp"

android:layout\_marginStart="20dp"

android:elevation="3dp"

android:padding="5dp"

app:cardCornerRadius="12dp">

<ImageView

android:id="@+id/detailimage5"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:scaleType="fitXY" />

</androidx.cardview.widget.CardView>

</LinearLayout>

</HorizontalScrollView>

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content">

<TextView

android:id="@+id/expand\_tourtips"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentEnd="true"

android:fontFamily="@font/mulir"

android:padding="5dp"

android:text="More"

android:textColor="#2196F3"

android:textSize="16dp"

android:textStyle="bold" />

<net.cachapa.expandablelayout.ExpandableLayout

android:id="@+id/myexpandable"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical"

app:el\_duration="400"

app:el\_expanded="false">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:padding="5dp"

android:text="Activities"

android:textColor="#3A3A3A"

android:textSize="18sp"

android:textStyle="bold" />

<HorizontalScrollView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:scrollbars="none">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_marginTop="5dp"

android:orientation="horizontal">

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:orientation="vertical">

<androidx.cardview.widget.CardView

android:layout\_width="80dp"

android:layout\_height="80dp"

android:elevation="3dp"

android:padding="5dp"

app:cardCornerRadius="12dp">

<ImageView

android:id="@+id/detailimage6"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:scaleType="fitXY" />

</androidx.cardview.widget.CardView>

<TextView

android:id="@+id/activity1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:padding="5dp"

android:text="Scuba diving"

android:textColor="#3A3A3A"

android:textSize="12sp" />

</LinearLayout>

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:orientation="vertical">

<androidx.cardview.widget.CardView

android:layout\_width="80dp"

android:layout\_height="80dp"

android:layout\_marginStart="20dp"

android:elevation="3dp"

android:padding="5dp"

app:cardCornerRadius="12dp">

<ImageView

android:id="@+id/detailimage7"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:scaleType="fitXY" />

</androidx.cardview.widget.CardView>

<TextView

android:id="@+id/activity2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:layout\_marginStart="20dp"

android:padding="5dp"

android:text="Scuba diving"

android:textColor="#3A3A3A"

android:textSize="12sp" />

</LinearLayout>

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:orientation="vertical">

<androidx.cardview.widget.CardView

android:layout\_width="80dp"

android:layout\_height="80dp"

android:layout\_marginStart="20dp"

android:elevation="3dp"

android:padding="5dp"

app:cardCornerRadius="12dp">

<ImageView

android:id="@+id/detailimage8"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:scaleType="fitXY" />

</androidx.cardview.widget.CardView>

<TextView

android:id="@+id/activity3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:layout\_marginStart="20dp"

android:padding="5dp"

android:text="Scuba diving"

android:textColor="#3A3A3A"

android:textSize="12sp" />

</LinearLayout>

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:orientation="vertical">

<androidx.cardview.widget.CardView

android:layout\_width="80dp"

android:layout\_height="80dp"

android:layout\_marginStart="20dp"

android:elevation="3dp"

android:padding="5dp"

app:cardCornerRadius="12dp">

<ImageView

android:id="@+id/detailimage9"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:scaleType="fitXY" />

</androidx.cardview.widget.CardView>

<TextView

android:id="@+id/activity4"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:layout\_marginStart="20dp"

android:padding="5dp"

android:text="Scuba diving"

android:textColor="#3A3A3A"

android:textSize="12sp" />

</LinearLayout>

</LinearLayout>

</HorizontalScrollView>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:padding="3dp"

android:text="Tour Tips"

android:textColor="#171717"

android:textSize="16dp"

android:textStyle="bold" />

<TextView

android:id="@+id/detailtips"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:lineSpacingExtra="2dp"

android:text="Day1:\n Explore all new activities with your friends and family under the guidance.\nDay2:\n Explore all new activities with your friends and family under the guidance.\nDay3:\n Explore all new activities with your friends and family under the guidance.\nDay4:\n Explore all new activities with your friends and family under the guidance.\n"

android:textColor="#B0AFB1"

android:textSize="12sp"

android:textStyle="bold" />

</LinearLayout>

</net.cachapa.expandablelayout.ExpandableLayout>

</RelativeLayout>

<Button

android:id="@+id/btnBook"

android:layout\_width="200dp"

android:layout\_height="50dp"

android:layout\_gravity="center"

android:layout\_marginTop="10dp"

android:layout\_marginRight="15dp"

android:layout\_marginBottom="20dp"

android:background="@drawable/book\_btn\_bg"

android:text="Book Now "

android:textAllCaps="false"

android:textColor="#FFF"

android:textSize="18sp" />

</LinearLayout>

</RelativeLayout>

</ScrollView>

* **Login Page:**

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

<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"

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

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="#FFF"

android:id="@+id/loginview"

tools:context=".authentication.Customer\_login">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

android:layout\_gravity="center">

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:padding="20dp">

<ImageView

android:id="@+id/loginimage"

android:layout\_width="match\_parent"

android:layout\_height="220dp" />

<TextView

android:id="@+id/welcome"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/loginimage"

android:layout\_centerHorizontal="true"

android:fontFamily="@font/mulie"

android:includeFontPadding="false"

android:text="Welcome Back !"

android:textAllCaps="true"

android:textColor="#000"

android:textSize="22sp" />

<TextView

android:id="@+id/existing"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/welcome"

android:layout\_centerHorizontal="true"

android:fontFamily="@font/mulir"

android:text="Log in to your existing account of explore"

android:textAlignment="center"

android:textColor="#B3B2B2"

android:textStyle="bold"

android:textSize="13sp" />

<EditText

android:id="@+id/ctusername"

android:layout\_width="300dp"

android:layout\_height="50dp"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="25dp"

android:layout\_below="@+id/existing"

android:background="@drawable/bginput"

android:drawableStart="@drawable/custom\_email"

android:fontFamily="@font/mulir"

android:hint="Enter Username"

android:inputType="textEmailAddress"

android:paddingStart="10dp"

android:drawablePadding="5dp"

android:paddingEnd="10dp"

android:textColor="#0D7BD3"

android:textSize="16dp" />

<EditText

android:id="@+id/ctpassword"

android:layout\_width="300dp"

android:layout\_height="50dp"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="15dp"

android:background="@drawable/bginput"

android:drawableStart="@drawable/custom\_lock"

android:drawablePadding="5dp"

android:layout\_below="@+id/ctusername"

android:fontFamily="@font/mulir"

android:hint="Enter Password"

android:inputType="textPassword"

android:paddingStart="10dp"

android:paddingEnd="10dp"

android:textColor="#0D7BD3"

android:textSize="16dp" />

<CheckBox

android:id="@+id/show"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/ctpassword"

android:layout\_marginTop="5dp"

android:text="show password" />

<TextView

android:id="@+id/txtforgot"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/ctpassword"

android:layout\_alignParentEnd="true"

android:layout\_gravity="center"

android:layout\_marginTop="11dp"

android:fontFamily="@font/mulir"

android:text="Forgot password"

android:textColor="#474646"

android:textSize="14sp"

android:textStyle="bold" />

<Button

android:id="@+id/btnlogin"

android:layout\_width="290dp"

android:layout\_height="45dp"

android:layout\_below="@+id/txtforgot"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="35dp"

android:background="@drawable/round\_btn\_bg"

android:elevation="5dp"

android:text="Sign in"

android:textAllCaps="true"

android:textColor="#FFFBF8" />

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/btnlogin"

android:gravity="center"

android:layout\_marginTop="8dp">

<TextView

android:id="@+id/Dont"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:fontFamily="@font/mulir"

android:text="Don't have an account?"

android:textColor="#1C1C1C"

android:textSize="12sp"

android:textStyle="bold" />

<TextView

android:id="@+id/txtsignup"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_toRightOf="@+id/Dont"

android:text="Register Now"

android:textColor="#1E53F3"

android:layout\_marginStart="3dp"

android:textSize="12sp"

android:layout\_toEndOf="@id/Dont"

android:textStyle="bold" />

</RelativeLayout>

<ProgressBar

android:id="@+id/ctprogressbar"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerVertical="true"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="5dp"

android:indeterminateTint="#1A45E3"

android:visibility="gone"

/>

</RelativeLayout>

</LinearLayout>

</ScrollView>

* **Registration**

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

<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"

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

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="#FFF"

android:id="@+id/signupview"

tools:context=".authentication.Customer\_Registration">

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

>

<ImageView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:src="@drawable/backward"

android:layout\_marginStart="10dp"

android:layout\_marginTop="15dp"

android:id="@+id/ctrback"/>

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_below="@+id/ctrback"

android:layout\_centerVertical="true"

android:layout\_marginTop="10dp"

android:padding="20dp">

<de.hdodenhof.circleimageview.CircleImageView

android:id="@+id/ctrprofile"

android:layout\_width="100dp"

android:layout\_height="100dp"

android:layout\_centerHorizontal="true"

android:background="@drawable/custprofile"

app:civ\_border\_color="#B5B3B3"

app:civ\_border\_width="1dp" />

<TextView

android:id="@+id/lets"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@id/ctrprofile"

android:layout\_centerHorizontal="true"

android:fontFamily="@font/mulie"

android:includeFontPadding="false"

android:paddingTop="10dp"

android:text="Let's Get Started!"

android:textColor="#000"

android:textSize="20sp" />

<TextView

android:id="@+id/create"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/lets"

android:layout\_centerHorizontal="true"

android:fontFamily="@font/mulir"

android:includeFontPadding="false"

android:text="Create an account to explore app to get all features"

android:textColor="#BAB7B7"

android:textSize="13sp"

android:textStyle="bold" />

<EditText

android:id="@+id/ctrfullname"

android:layout\_width="300dp"

android:layout\_height="50dp"

android:layout\_below="@+id/create"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="15dp"

android:background="@drawable/bginput"

android:drawableStart="@drawable/custom\_person"

android:drawablePadding="5dp"

android:fontFamily="@font/mulir"

android:hint="Enter full name"

android:inputType="textPersonName"

android:paddingStart="10dp"

android:paddingEnd="10dp"

android:textColor="#0D7BD3"

android:textSize="16dp" />

<EditText

android:id="@+id/ctremail"

android:layout\_width="300dp"

android:layout\_height="50dp"

android:layout\_below="@+id/ctrfullname"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="15dp"

android:background="@drawable/bginput"

android:drawableStart="@drawable/custom\_email"

android:drawablePadding="5dp"

android:fontFamily="@font/mulir"

android:hint="Enter Username"

android:inputType="textEmailAddress"

android:paddingStart="10dp"

android:paddingEnd="10dp"

android:textColor="#0D7BD3"

android:textSize="16dp" />

<EditText

android:id="@+id/ctrphone"

android:layout\_width="300dp"

android:layout\_height="50dp"

android:layout\_below="@+id/ctremail"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="15dp"

android:background="@drawable/bginput"

android:drawableStart="@drawable/custom\_phone"

android:drawablePadding="5dp"

android:fontFamily="@font/mulir"

android:hint="Enter Phone No."

android:inputType="phone"

android:paddingStart="10dp"

android:paddingEnd="10dp"

android:textColor="#0D7BD3"

android:textSize="16dp" />

<EditText

android:id="@+id/ctrpassword"

android:layout\_width="300dp"

android:layout\_height="50dp"

android:layout\_below="@+id/ctrphone"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="15dp"

android:background="@drawable/bginput"

android:drawableStart="@drawable/custom\_lock"

android:drawablePadding="5dp"

android:fontFamily="@font/mulir"

android:hint="Enter Password"

android:inputType="textPassword"

android:paddingStart="10dp"

android:paddingEnd="10dp"

android:textColor="#0D7BD3"

android:textSize="16dp" />

<CheckBox

android:id="@+id/ctrshow"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/ctrpassword"

android:layout\_marginStart="5dp"

android:layout\_marginTop="5dp"

android:text="show password" />

<Button

android:id="@+id/btnregister"

android:layout\_width="290dp"

android:layout\_height="45dp"

android:layout\_below="@+id/ctrshow"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="35dp"

android:background="@drawable/round\_btn\_bg"

android:elevation="5dp"

android:text="join now"

android:textAllCaps="true"

android:textColor="#FFFBF8" />

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/btnregister"

android:layout\_marginTop="8dp"

android:gravity="center">

<TextView

android:id="@+id/Dont"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:fontFamily="@font/mulir"

android:text="Don't have an account?"

android:textColor="#1C1C1C"

android:textSize="12sp"

android:textStyle="bold" />

<TextView

android:id="@+id/ctrsignin"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="3dp"

android:layout\_toEndOf="@id/Dont"

android:layout\_toRightOf="@+id/Dont"

android:text="Sign in"

android:textColor="#1E53F3"

android:textSize="12sp"

android:textStyle="bold" />

</RelativeLayout>

<ProgressBar

android:id="@+id/ctrprogressbar"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:layout\_centerVertical="true"

android:layout\_marginTop="5dp"

android:indeterminateTint="#1A45E3"

android:visibility="gone" />

</RelativeLayout>

</RelativeLayout>

</ScrollView>

**System**

**Evaluation**

**SOFTWARE TESTING**

The primary and larger objective of testing is to deliver quality software. Quality software is one that is devoid of error and meets with customer’s stated requirements.

If errors are found, then the software must be debugged to locate these errors in the various parts of the program. Corrections are then made. The program/system must be tested once again after corrections have been implemented – this time with additional objective of finding out whether or not corrections in one part of the system have introduced any new errors elsewhere in the system.

Once all errors are found, then another objective must be accomplished that is to check whether or not the system is doing what it is supposed to do. So another aspect of testing is that it must also ensure that the system meets with user requirements.

**Here are some of the testing techniques.**

**1.** Unit Testing

**2.** Integration Testing

**3.** Module Testing

**4.** Subsystem Testing

**5.** Black Box Testing

**6.** White Box Testing

**7.** Equivalence Partitioning

**8.** Ad-hoc Testing

**9.** Boundary Value Analysis

The Testing Technique used in this project is as follows.

**1. Unit Testing:**

Unit testing is performed by the developers before the setup is handed over to the testing team to formally execute the test cases. Unit testing is performed by the respective developers on the individual units of sources code assigned areas. The developers use test data of the quality assurance team.

The goal of unit testing is to isolate each part of the program and show that individual parts are correct in terms of requirements and functionality.

**2. Integration Testing:**

The testing of combined parts of an application to determine if they function correctly together is Integration testing. There are two methods of doing Integration Testing bottom-up Integration testing and Top down Integration testing.

**Bottom-up:**

1. This testing begins with unit testing, followed by tests of progressively higher-level combinations of units called modules or builds.

**Top-Down:**

1. This testing, the highest-level modules are tested first and progressively lower-level modules are tested after that.
2. In a comprehensive software development environment, bottom-up testing is usually done first, followed by top-down testing.

**3. Module Testing:**

Module testing is the testing of complete code objects as produced by the compiler when built from source. It is a collection of independent components such as an object class, an abstract data-type or some loser collection of procedures and functions. A module encapsulates related components so can be tested without other modules.

**4. Sub System Testing:**

This phase of testing involves collecting of modules, which have been integrated into subsystems. Subsystems may be independently designed and implemented. The most common problems, which arise in the large software systems, are sub system interface mismatches. The subsystem test process should therefore concentrate on the detection of interface errors by rigorously exercising these interfaces.

**5. System testing:**

System testing is the testing of behavior of a complete and fully integrated software product based on the software requirements specification (SRS) document. In maim focus of this testing is to evaluate functional end user requirements.

This is black box type of testing where external working of the software is evaluated with the help of requirement documents & it is totally based on users’ point of view. For this type of resting do not required knowledge of internal design or structure or code.

`**POSITIVE TESTING OF LOGIN FUNCTIONALITY:**

Testing for **Login** Functionality with valid email and password.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test**  **Case ID** | **Test**  **Title** | **Description** | **Input Data** | **TestCase Steps** | **Expected Result** | **Actual Result** | **Status** |
| 1 | User/  Restaurant  Login | Testing Login functionality of User/  Admin  for successful login | Valid Email  Valid Password | 1.Open the app and enter  2.Type valid email id in email section  3.Enter Valid Password  4.Click on login | User/  Admin (Home Page)  Verificationa mail received | Logged on to app | Test Case Passed |

Table (1)

Testing under **Login** functionality with Valid Email and Incorrect Password.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test**  **Case ID** | **Test**  **Title** | **Description** | **Input Data** | **TestCase Steps** | **Expected Result** | **Actual Result** | **Status** |
| 2 | User /Restaurant  Login | Testing Login functionality of User/ Admin  for unsuccessful login | Valid Email  Invalid Password | 1.Open the app  2.Type valid email id in email section  3.Enter Valid Password  4.Click on login | Display Error Invalid Password | Display Error | Test Case Passed |

Table (2)

**NEGATIVE TESTING OF LOGIN FUNCTIONALITY**

Testing under **Login** functionality with Valid Email and Password.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test**  **Case ID** | **Test**  **Title** | **Description** | **Input Data** | **Test Case Steps** | **Expected Result** | **Actual Result** | **Status** |
| 3 | User/  Restaurant  Login | Testing Login functionality of User and Admin for unsuccessful login with blank Email and blank Password | Blank  Email  Blank Password | 1.Open the app  2.Type blank email id in email section | Display Error  Enter Valid Id | Display Error | Test Case Passed |

Table (3)

**POSITIVE TESTING OF REGISTRATION FUNCTIONALITY:**

Testing under **Registration** functionality with following details in **User**.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test**  **Case ID** | **Test**  **Title** | **Description** | **Input Data** | **TestCase Steps** | **Expected Result** | **Actual Result** | **Status** |
| 4 | User registration | Testing Register  functionality of User  for successful register | Valid Email, Username, Phone no, Password | 1. Openthe app .  2.On the Login Page click on the “Not have an account create on”  3.Type valid email,  Name,  Phone no,  Password.  4. OTP is sent to email id. | OTP is sent to Users email Id and verified | User registered inside the app | Test Case Passed |

Table (4)

Testing under **Registration** functionality with following details in **Admin**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test**  **Case ID** | **Test**  **Title** | **Description** | **Input Data** | **TestCase Steps** | **Expected Result** | **Actual Result** | **Status** |
| 5. | 1.Admin  registration | Testing Register  functionality of Admin  for successful register | Valid Email, Admin name, Phone no, Password | 1. Openthe app.  2.On the Login Page click on the “Not have an account create on”  3.Type valid email,  Admin Name,  Phone no,  Password. | Verificationa mail receivedAdmin email Id and verified | Admin registered inside the app | Test Case Passed |

Table (5)

**NEGATIVE TESTING OF REGISTRATION FUNCTIONALITY:**

Testing under **Registration** functionality with Valid Email and Password in **User**.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test**  **Case ID** | **Test**  **Title** | **Description** | **Input Data** | **Test Case Steps** | **Expected Result** | **Actual Result** | **Status** |
| 6. | User Registration | Testing Registration functionality of User for unsuccessful register with invalid Email | Invalid  Email | 1.Open The app  2.Type invalid email id in email section | 1.Invalid Email  2.OTP is not sent | Display Error | Test Case Passed |

Table (6)

**NEGATIVE TESTING FOR REGISTRATION FUNCTIONALITY**

Testing **Registration** with functionality invalid Email and Password in **Admin.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test**  **Case ID** | **Test**  **Title** | **Description** | **Input Data** | **Test Case Steps** | **Expected Result** | **Actual Result** | **Status** |
| 7 | Admin  Registration | Testing Registration functionality of Admin  for unsuccessful register with invalid Email | Invalid  Email | 1.Open The app  2.Type invalid email id in email section | 1.Invalid Email  2.OTP is not sent | Display Error | Test Case Passed |

**Table (7)**

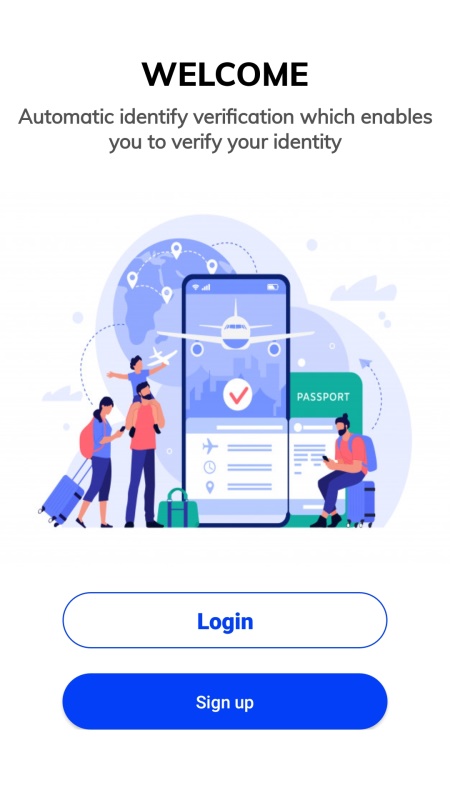
**Test Cases for Travel Hunt App**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Module** | **Sub Module** | **Input Data** | **Expected Result** | **Actual Result** | **Positive**  **Status** | **Negative status** |
| 1. | Login Module | Forgot Password | Valid Email | Email is sent for reset password | Password is reset | Pass | Fail |
|  |  | Create Account | User details | All details should be compulsorily filled | User is created | Pass | Fail |
| 2. | Register Module | User register | Name, Email, Phone no., Password | After click on “register” OTP is sent on Email. | User is registered | Pass | Fail |
|  |  | OTP verification | OTP number | Enter number and email OTP number is same. | Valid OTP number | Pass | Fail |
|  |  | Sign in | Click on “Already have an account” | Directed to login page | Login page | Pass | Fail |
| 3. | Customer  drawer | Customer navigation | Click on Home at the top  navigation | menus | Menus displayed | Pass | Fail |
|  |  | customer Profile | Click on  Open profile | Name,phone,  email | Display  Customer details | Pass | Fail |
|  |  | Contact us | Click on Home at the top  navigation | Directed to contact us activity | Display contact us | Pass | Fail |
|  |  | Facility | Click on Home at the top  navigation | Directed  To Facillity Activity | Display  All facilities | Pass | Fail |
|  |  | User  Module | Category of package | View all | Display Specific packages | Pass | Fail |
| 4. | Admin Module | Add  Module | Add  Packages to particular category | Design of adding new packge | Displayed  design | Pass | Fail |
| 5. | Package Detail Module | Book module | Click book now | Intent next activity | Opened choose resort | Pass | Fail |
|  |  | Rating | Give rating | Open Rating design and calculate | Displaying accurate  rating | Pass | Fail |
| 6. | Payment Module | Online payment Module | Select on online payment radio button  And click confirm payment | Directed to Payment Activity | Display Payment  method | Pass | Fail |
| 7. | Payment Module | Card | Insert card number, Card holder name, CVV and check in if to remember the card and click Pay | Display toast order is placed Successfully | Order is placed Successfully | Pass | Fail |
|  |  | UPI | Select Google pay /Phone Pay | Display toast order is placed Successfully | Order is placed Successfully | Pass | Fail |
| 8. | Customer  drawer | Logout | Select Logout | Logout from the current account and directed at login page | Logout is Successful | Pass | Fail |
| 9. | Booking Details Module | Cancel Order | Click on cancel Booking | Booking is cancelled | Booking is cancelled | Pass | Fail |

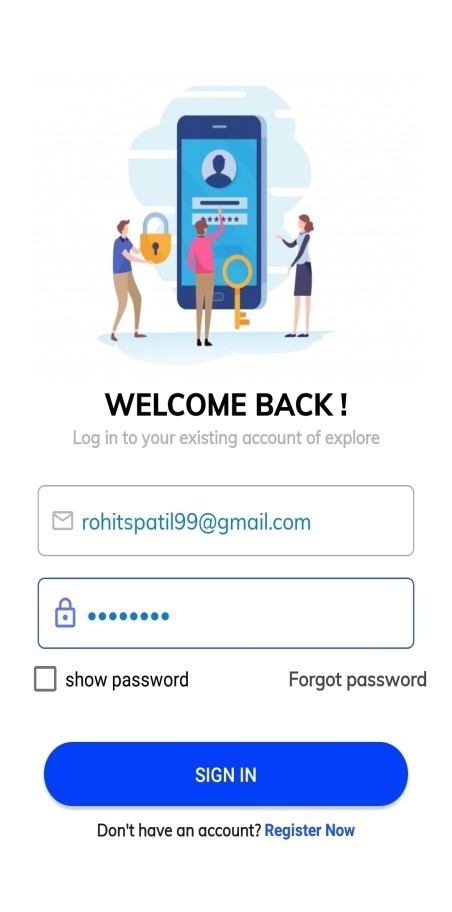
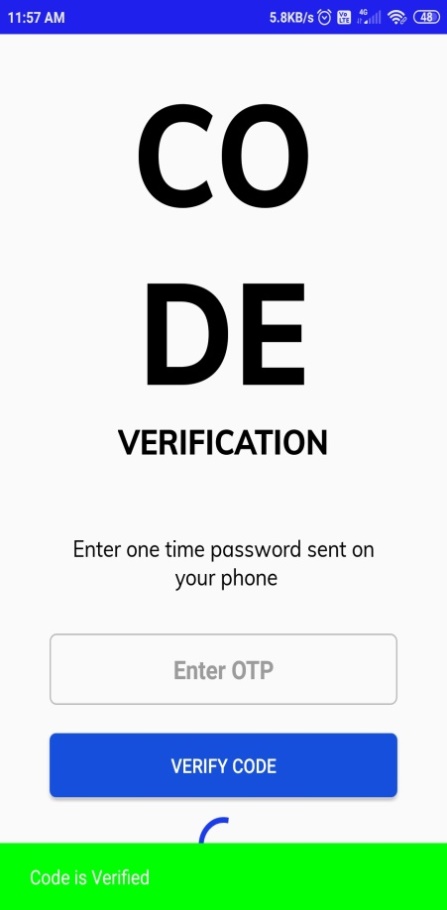
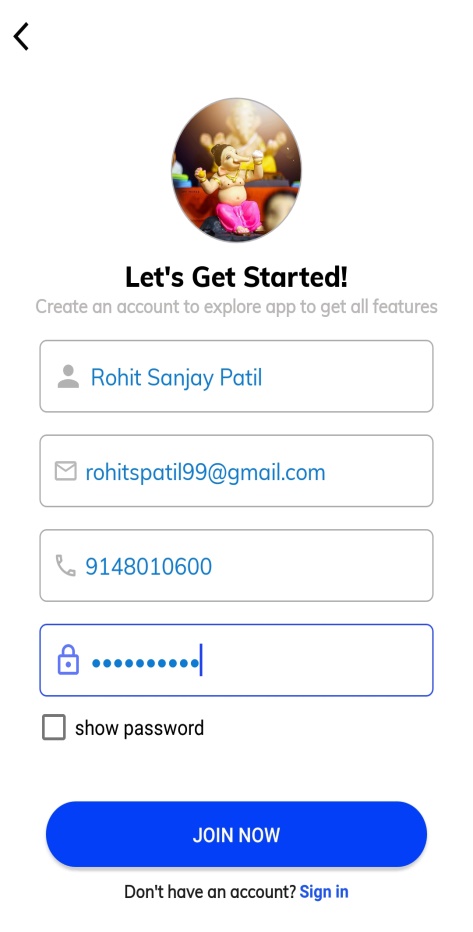
**Table (8)**

**SCREEN LAYOUTS**

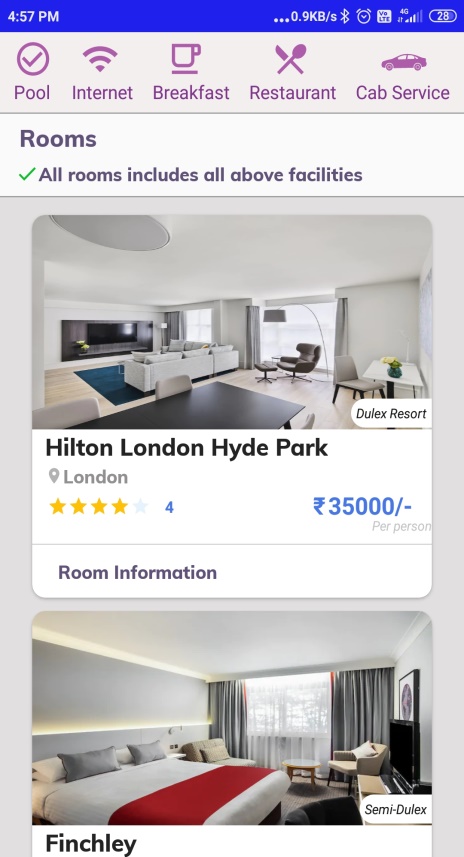
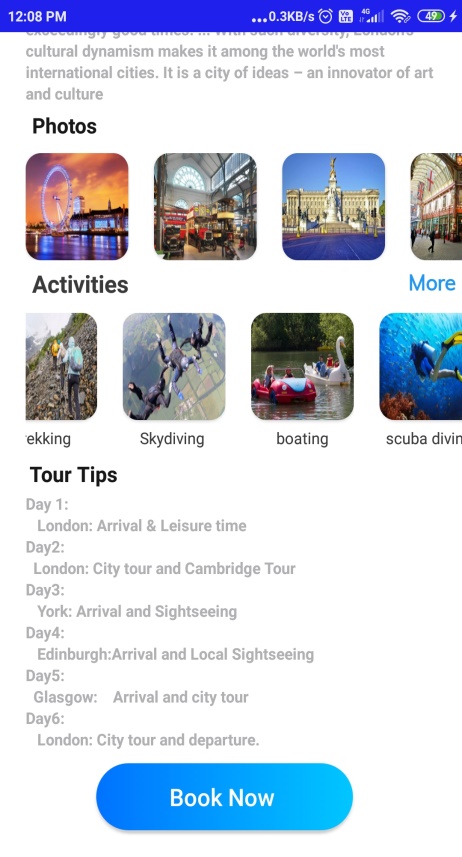
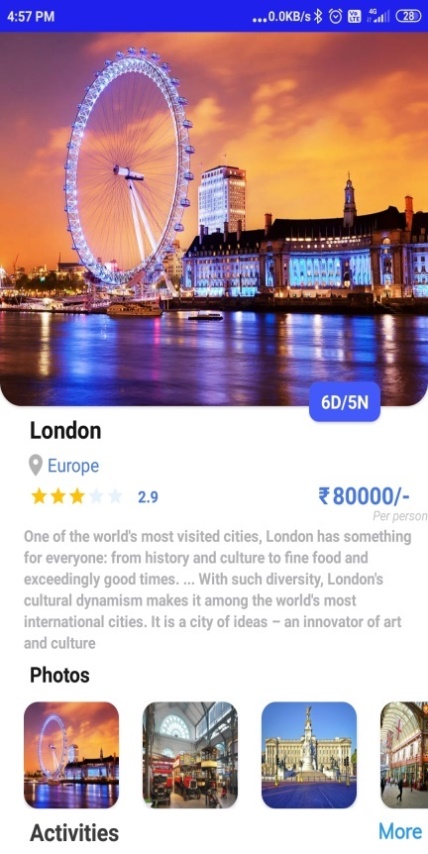
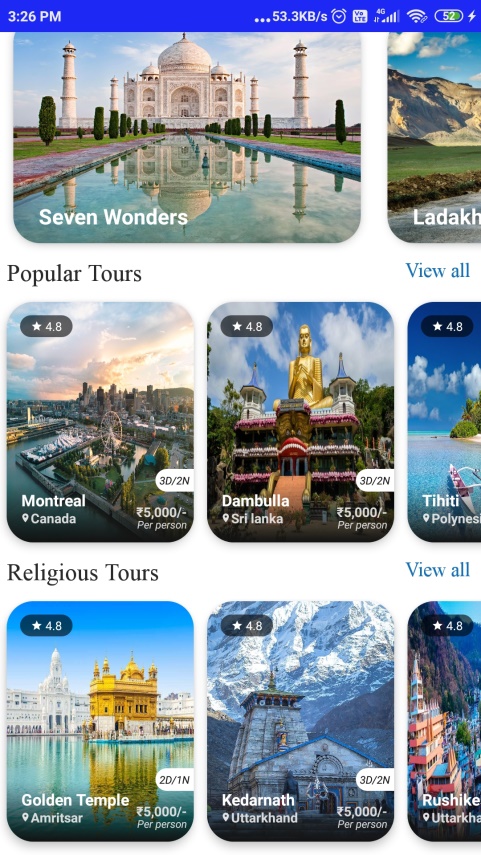
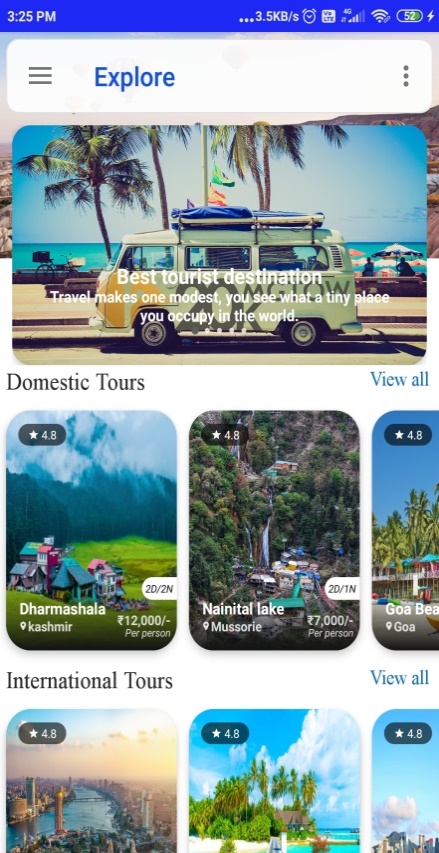
**Home Page**

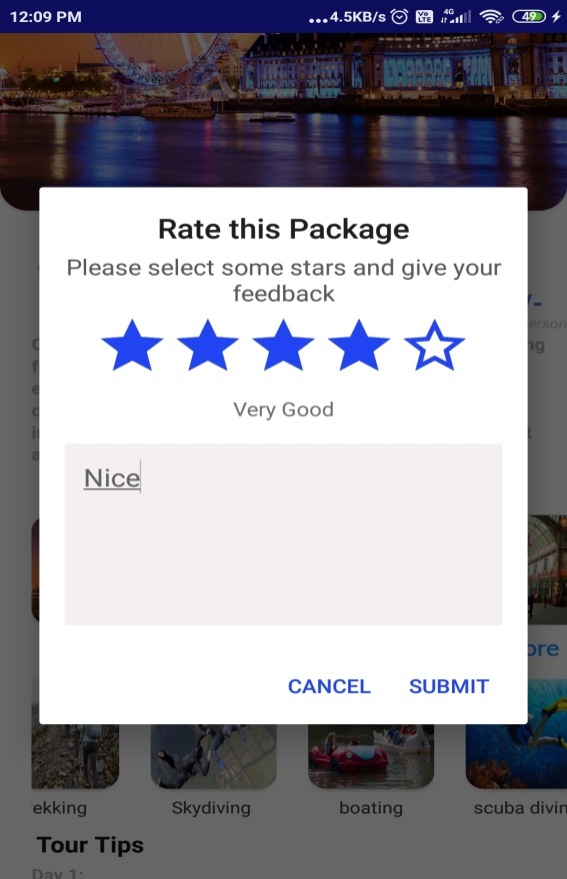
**Login and Registration Page**



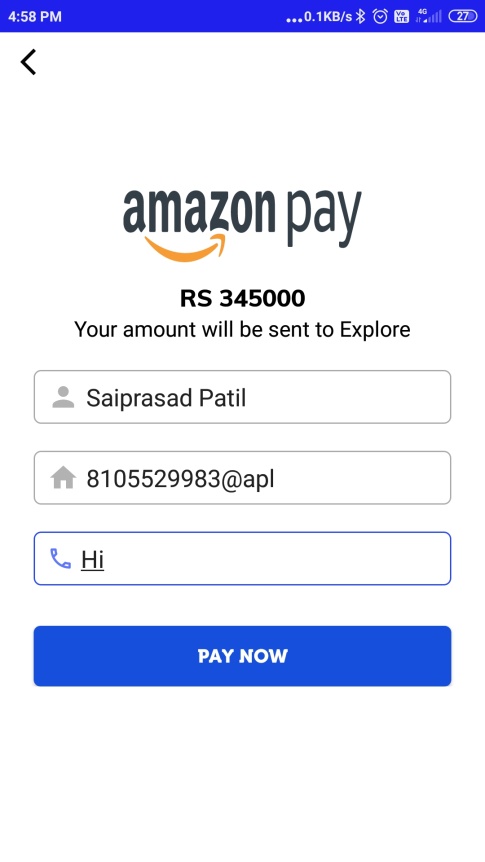
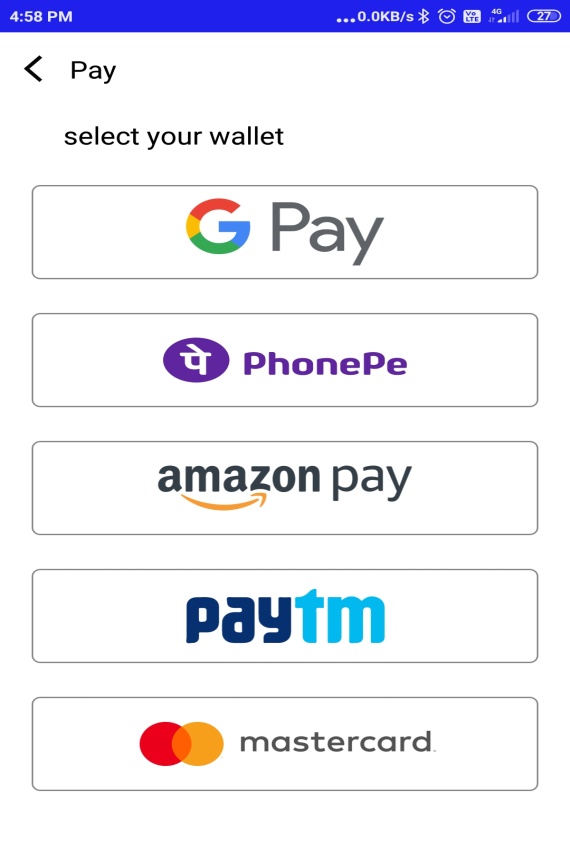
**Tour Packages**



**Rating**

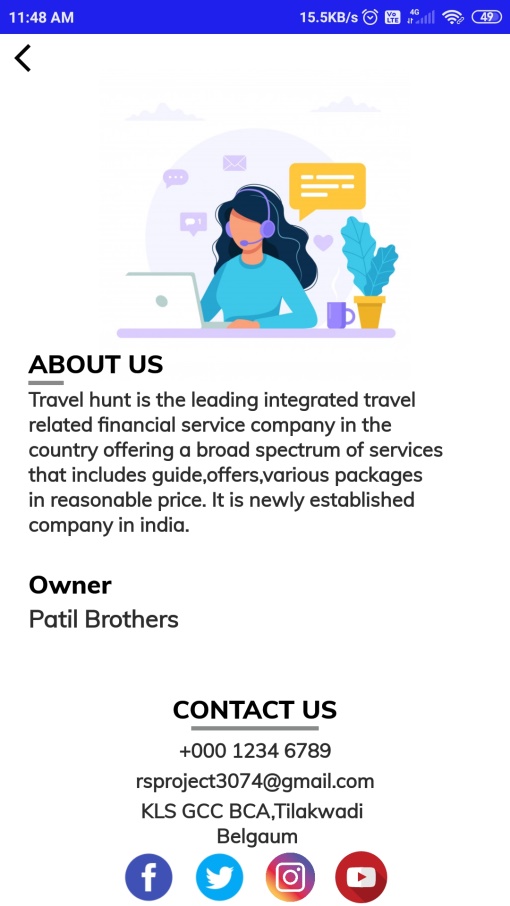
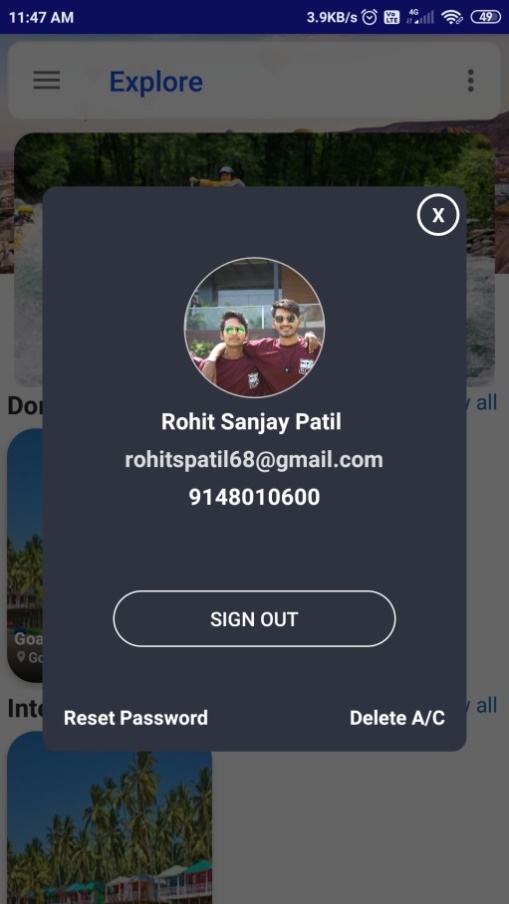


**Payment Method**

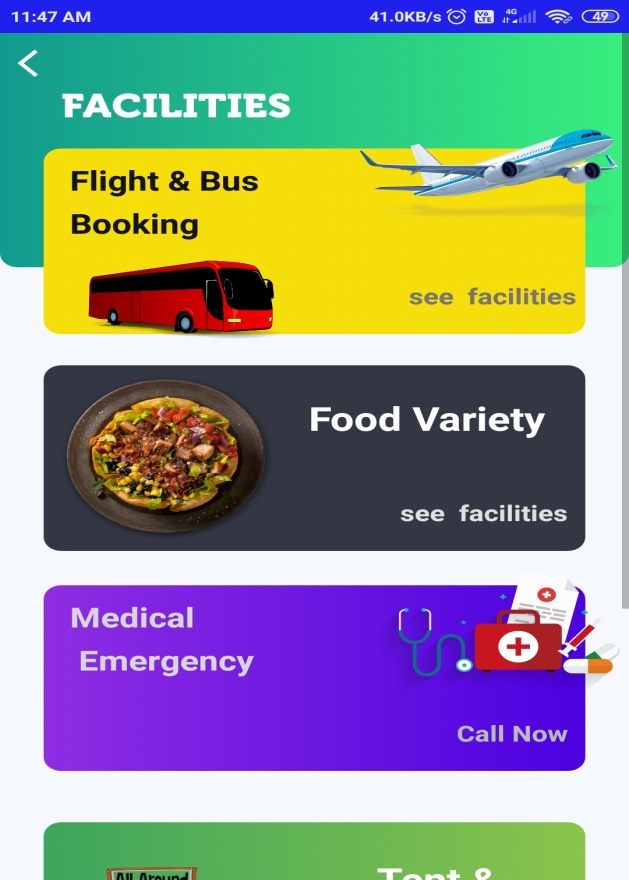
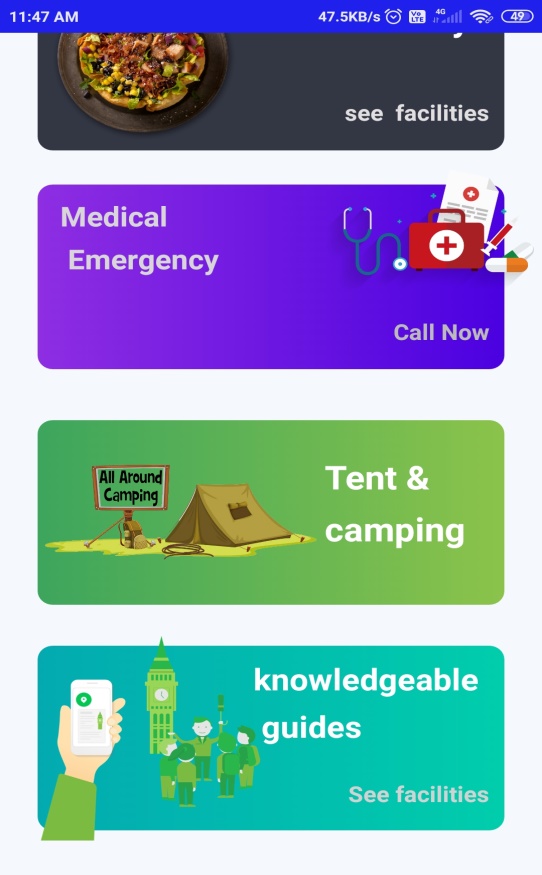


**Contact Us**

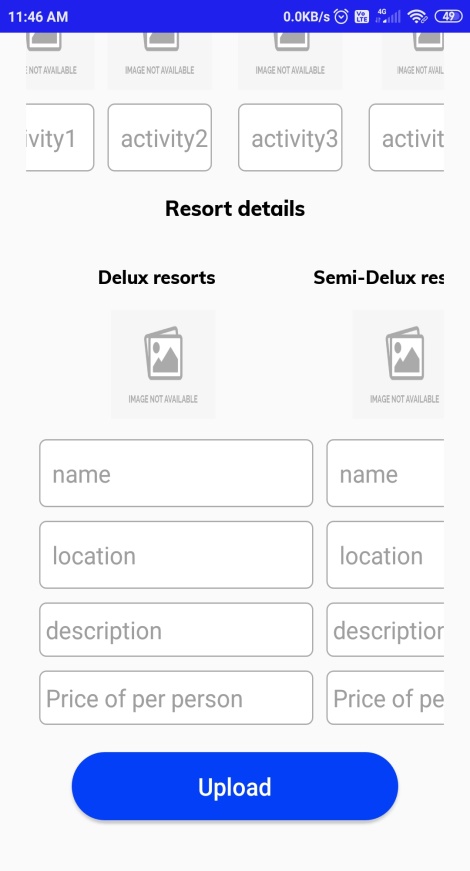
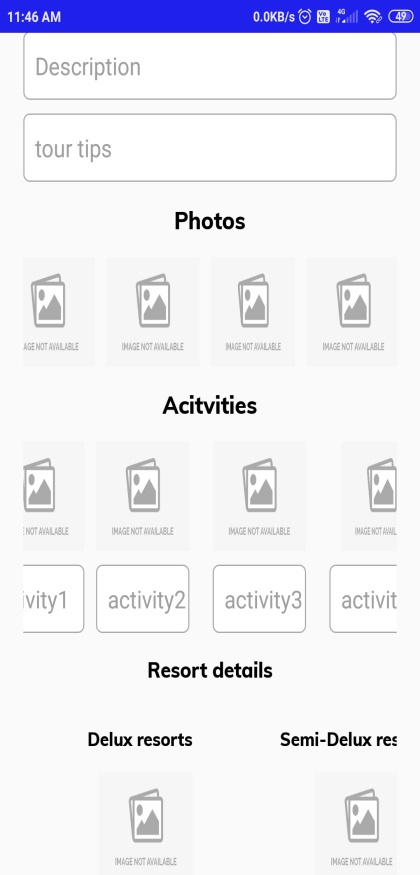
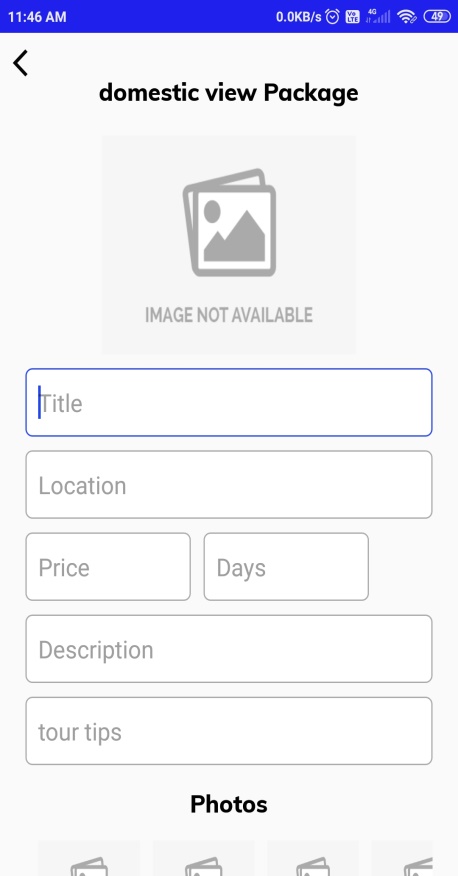
**Tourist Profile**

**Facilities**

**New Tour Package Upload**



**Conclusion**

**CONCLUSION**

This project’s objective is to develop an android application for tourists, which helps in finding out India’s and World’s tourist places and related details. It also user friendly and can be easily accessed by any user. This project is designed to make the work of admin or client much easier and faster since it is computerized.

**Annexure**

1. **LIST OF TABLES:**

Table1: Positive Testing for Login Functionality with valid input.

Table 2: Postive Testing for login functionality with invalid input.

Table 3: Negative Testing for Login Functionality.

Table 4: Postive Testing for Registration Functionality (user)

Table 5: Postive Testing for Registration Functionality (Admin).

Table 6: Negative Testing for Registration Functionality (user)

Table 7: Negative Testing for Registration Functionality (Admin).

Table 8: Test Cases for all modules in Travel Hunt App.

1. **LIST OF FIGURES:**

Figure 1: Pert Chart.

Figure 2: Gantt chart.

Figure 3: Context Level Data Flow Diagram.

Figure 4: Top Level Data Flow Diagram.

Figure 5: Use Case Diagram.

**Limitations And**

**Future Enhancement**

**LIMITATIONS**

* It needs internet connection.
* Currently Works only on android platform.
* Minimum android version 5.0.

**FUTURE ENHANCEMENT**

* Users can give the review about the particular place.
* Many more places will be available.
* Book flights via this Application.
* Launching this application on Google Play Store.
* Building an ios version.

**References**

**BIBLIOGRAPHY**

**SITES:**

* Firebase reference:-[https://firebase.google.com/docs/android](https://firebase.google.com/docs/android/setup?gclid=CjwKCAjw7e_0BRB7EiwAlH-goG8jaksLh93eLkYf1-8OBbik5gciX87k-XHfsYSOnPGAU55s0Z6k2xoCAwkQAvD_BwE)
* Android Studio:-<https://codinginflow.com/>
* Android SDK and JAVA:-[https://www.tutorialspoint.com/android/](https://www.tutorialspoint.com/android/android_fragments.htm)
* Android with Firebase tutorials :<https://www.javatpoint.com/firebase>

**BOOKS:**

* Android Studio Development Essential By Neil Smith
* Software engineering seventh edition by Ian Sommerville