**Software Requirements** **Specification**

**For**

**Travel India & Tourism**

**guide App**

**Version 1.0 approved**

**Prepared by :**

**Vrushbh Mochi**

**Navaz Sherasiya**

**Akash Parmar**

**Fenil Gondaliya**

# Index : 1. Introduction

1.1 Purpose

1.2 Document Conversation

1.3 Intended audience and reading suggestions

1.4 Definitions

1.5 Project Scope

1.6 References

# Overall Product Description

2.1 Product Perspective

2.2 Product Features

2.3 User Classes and Characteristics

2.4 Operating Environment

2.5 Design and Implementation Constraints

2.6 User Documentation

2.7 Assumptions and Dependencies

# Specific Requirements

3.1 Functional Requirements

3.2 External interface requirements

3.3 Performance Requirements

3.4 Other Non-functional Requirements

3.5 Constraints

# Design

4.1. Class diagram

4.2. Data flow diagram (DFD)

4.3. Use-case diagram

# Testing

5.1. Test cases

5.2. Snapshots

# Wireframes

1. **Introduction : 1.1 Purpose**

* + - * This document proposes the software functionalities and requirements for version 1.0 of a road trip advisor web service capable of giving the user detailed information about the trip they are planning and give the near best hotel. It will also explain system constraints, interface, and interactions with other external applications

* + - * The tourist guide application **enables users to locate distinct places to visit**. They get a detailed description of the site along with the nearby places that they can see and also they can see the near best hotel.



* 1. Document Conventions :

Text Format :

Font : Times New Roman (Body), Constantia (Heading)

Font Size : 16 for Heading, 14 for Subheading, 12 for Body

Document conventions :

|  |  |
| --- | --- |
| MNEMONICS | MEANING |
| RDB | Real-Time Database |
| DFD | Data Flow Diagram |
| SS | Snap Shot |

* 1. Intended audience and reading suggestions
     + - While the software requirement specification (SRS) document is written for a more general audience, this document is intended for individuals directly involved in the development of SplitPay.
       - This includes software developers, project consultants, and team managers. This document need not be read sequentially; users are encouraged to jump to any section they find relevant. Below is a brief overview of each part of the document.

* + - * + Part 1 (Introduction)

This section offers a summary of the SplitPay project, including goals and objectives, project scope, general system details, and some major constraints associated with the intended platform.

* + - * + Part 2 (Data Design)

Readers interested in how SplitPay organizes and handles data should consult this section, which covers data structures and flow patterns utilized by the system.

* + - * + Part 3 (Architectural and Component-Level Design)

This section describes the SplitPay system class by class, including interface details, class hierarchies, performance/design constraints, process details, and algorithmic models.

* + - * + Part 4 (User Interface Design)

This section covers all of the details related to the structure of the graphical user interface (GUI), including some preliminary mockups of the SplitPay Android application. Readers can view this section for a tentative glimpse of what the final product will look like.

* + - * + Part 5 (Restrictions, Limitations, and Constraints)

This section discusses the general constraints imposed upon the project

* + - * + Part 6 (Testing Issues)

Readers interested in the software testing process should consult this section, which offers a list of test cases, expected responses, and other pertinent information.

* + - * + Part 7 (Appendices)

This section includes any additional information which may be helpful to readers.

* 1. Definitions : 1.4.1 User Account : • A user account is a location on a network server used to store a computer username, password, and other information.

* + 1. Mobile Operating System :

• A mobile operating system, also called a mobile OS, is an operating system that is specifically designed to run on mobile devices such as mobile phones, smartphones, PDAs, tablet computers and other handheld devices.

* + 1. Database :

A database is a collection of information that is organized so that it can easily be accessed, managed, and updated.

In one view, databases can be classified according to types of content.

* + 1. Data-Flow Diagram (DFD) :

It is a graphical representation of the "flow" of data through an information system, modeling its process aspects.

A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated.

* + 1. Use Case Diagram :

In software and systems engineering, a use case is a list of actions or event steps, typically defining the interactions between a role (known in the Unified Modeling Language as an actor) and a system, to achieve a goal.

The actor can be a human or other external system.

* 1. Project Scope :
     + - This whole project helps the user as a guide to travel around in a new city.
       - To get to the application, you have to sign up first and then have to log in to the system. After that, you can select the places to visit.
       - Once you click the desired city, it will automatically display the famous sites to visit in that area.
       - However, being foreigners, the travelers face different types of problem including limited transportation information, problem in understanding Bengali language and so on. Based on travelers.

* 1. References :

 https://www.getyourguide.com

1. **Overall Product Description:** 
   1. Product Perspective :

The Trip Advisor (TA) is a application for planning road trips. Enter a starting point and destination, and it will provide potential routes with pit stops selected along the way. To allow for easy navigation, users can transfer the routes to their cell phone for turn by turn directions. The goal of this web service is to minimize the number of different applications or services a user would need to bounce between in order to have a successful road trip. We give the user an option to pre plan their trips before they hit the road.The application will not only provide routing options for the user but in fact prepare a whole trip itinerary for them on a day to day basis for the duration of their vacation. The application will be built upon several different APIs including but not limited to Navigation APIs, Location API, Reviews API and Current Events API. The application will also be relying on some AI technology to enable planning more personalized itineraries based on user’s input

* 1. Product Features :

Registrations

Login/logout

Emergency call

Location map

Live Maps

App service integration

Trip Reviews From Other Travelers

* 1. User Classes and Characteristics :

There are essentially two classes of users for the Road Trip Advisor: the user of the internal application and an on-site technician. Each of these two users have a different use of the system as well as their own requirements. As for now, we will only refer to the user.

User: Must be older than 13 years old and must possess the technical know-how to use the interface for creating trips. The user must have a login to be able to save the selected trip. Users can use any other functionalities without having a login . Once the authentication is a success, the user can plan, modify, save, delete or oversee the trip by picking from three different trip alternatives. Different suggestions will be available to give the user more options. In order to get the optimized paths based on user preferences, the user will need to input preferences under the profile page.

* 1. Operating Environment :
     + - * Only for Android Device
         * Internet Connectivity
         * RAM 2 GB
         * ROM 16 GB
         * Android v5.0 or Higher

* 1. Design and Implementation Constraints :

 Localisation must be implemented using Navigation APIs  The Google API can only be used for a specific number of time per day.

* 1. User Documentation :
     + - * User Documents will have an image or video of how to use the app.
         * It will also have written steps on how to use the app.

* 1. Assumptions and Dependencies : Assumption

:

* + - * + There will be bugs in the application.
        + It will work perfectly on all devices.

Dependencies :

* + - * + System hardware should be Good enough to run the app.
        + App should be made on the basis of requirement.
        + Admin must have a proper understanding of the app.
        + Internet connectivity for exchanging data from server.

1. **Specific Requirement** 
   1. Function Requirements :
      1. View Places:

Description : • Users can view the product in the list at the dashboard section in that product description will be there and price of product will be there, image of product will also be there etc.

* + 1. Search Places :

Description :

Users can search the product from a specific name and get a list of Places.

Input :

Users have to give the name of the Places as input in the search bar.

Processing State :

When a user searches the Places, if any product has a keyword which is searched by the user, if there is a product in the database then it will show that Places.

Output :

List of Places having keywords which are searched by the users.

3.1.3 Private ChatBot :

Description :

If any user has any query about the product or any other queries then they can chat with ChatBot.

Input :

ChatBot option enables chatting with Bot for any queries about the product.

Output :

Users will get answers from bot.

3.1.4 Offer Suggester

Description

• User will get notification if there is any offer is going on

* 1. External Interface: 3.2.1 User Interface:

• The user interface should be intuitive, such that 99.9% of all users are able to use the app without any assistance.

* + 1. Hardware Interface:

Internet Connection.

RAM 2 GB.

ROM 16 GB.

It should be android devices.

* + 1. Software Interface:

Android OS.

Version 5.0 or greater(lollipop).

* 1. Performance Requirements :

3.3.1 Scalability :

• Applications should be able to provide instant messaging services to users in sufficient time.

* + 1. Performance :

• Application must be lightweight and must show product instantly.

* + 1. Speed : • Application’s processing speed should be so high that there should be no delay in executing the user's instructions.

• Also, the application should not crash repeatedly.

* + 1. Cache Memory :

The app shall not consume more cache memory.

Even if it does, it must provide a choice to the user to clear the app cache manually.

* 1. Other Non-Functional Requirement : 3.4.1 Privacy :

The users are provided with the benefit of customizing their privacy settings.

Hence, they shall make the best use of these settings.

* + 1. Security and Safety : • Keep your password safe and don't share it with any other people, application or website.

• We also suggest using a different password for every service you use.

* + 1. Reliability : • It is very important that the app is reliable as many users use this application simultaneously.

• All data collected by app shall be preserved safely and should follow data hiding.

* + 1. Portability :

• Application can be used on any android phones and tablets.

* + 1. Maintainability :

Travel India Beta releases all new updates first and looks for acceptance from its customers.

Travel India always surprises its users by releasing fresh updates.

* + 1. User Friendly :

• This application is user-friendly, meaning to say even if one just installs the app and uses it for the first time, he’d find it easy to operate the application.

* + 1. Permissions :

• Location :

o Use your location for features like Geofilters and for other services that improve your experience.

• Full network access :

o Send and receive notifications, offers, and other data.

• Clipboard :

o Allow users to access clipboard so users can attach links quickly with the Paperclip tool.

Change network connectivity : o Connect directly to Spectacles via Wi-Fi.

Read phone status and identity :

o Autofill your phone number when you register for Travel India, for your convenience.

* + 1. Support : • A good software is one which listens to its customer’s feedback and helps them whenever they need something.
    2. Report Spam :

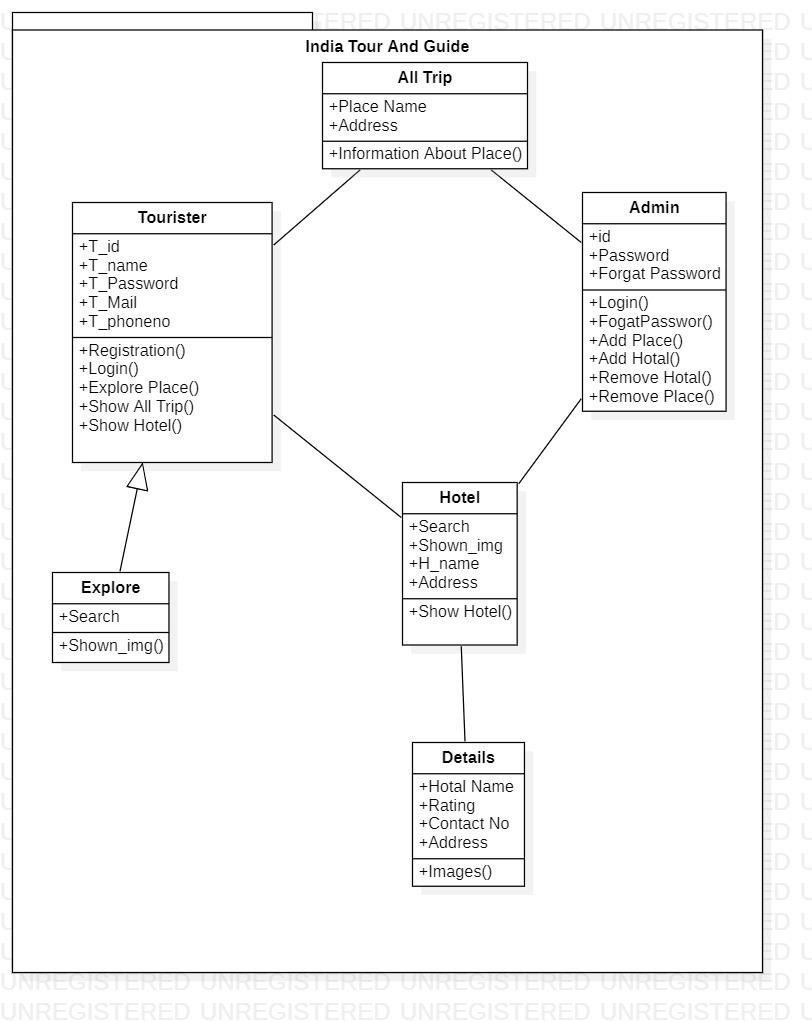
• In case of an irreverent content in the app and the user wants to report then they can report in mail.

* + 1. Feedback / Write a Review : • Once the app has been installed, the customer/user can write his valuable feedback.

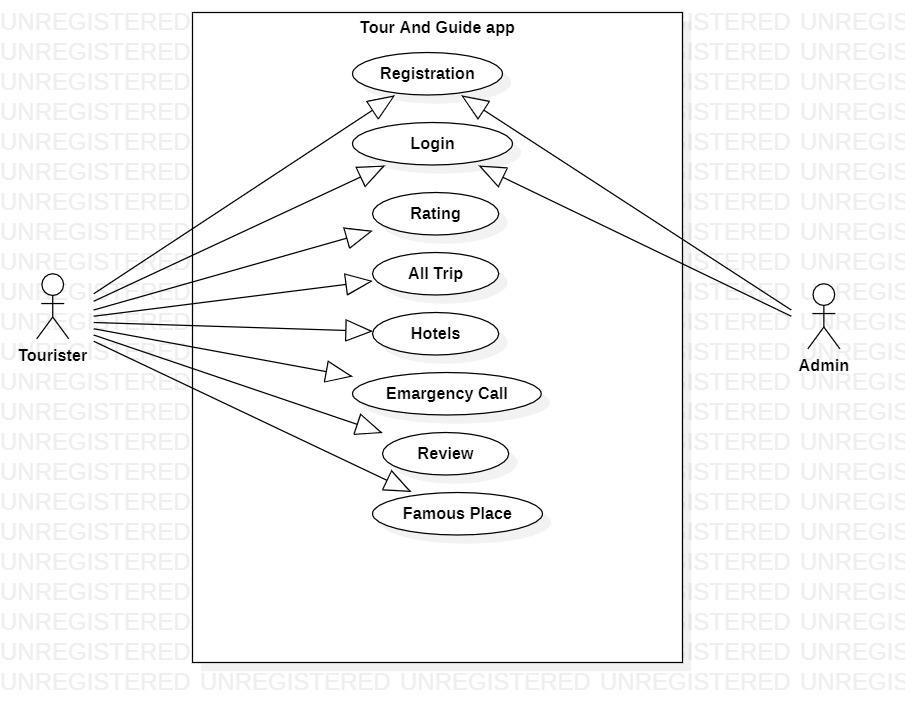
• He can rate the app accordingly and can openly share his thoughts.

* 1. Constraints :
     + - * Login id & password of user must be valid
         * Signup details – mandatory valid details must be provided by the user.
         * New account created
         * location of a user is located in unusual places within frequent intervals, verifying the user’s presence and use of the app.
         * OS required: Android (5.0), Mobile/Tablet platform

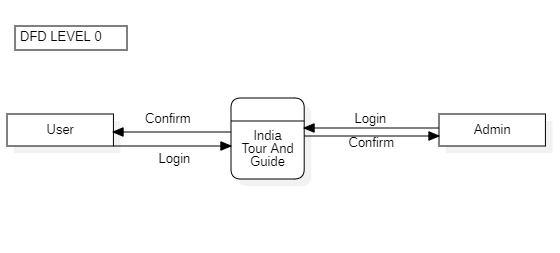
1. **Design :**  Class Diagram



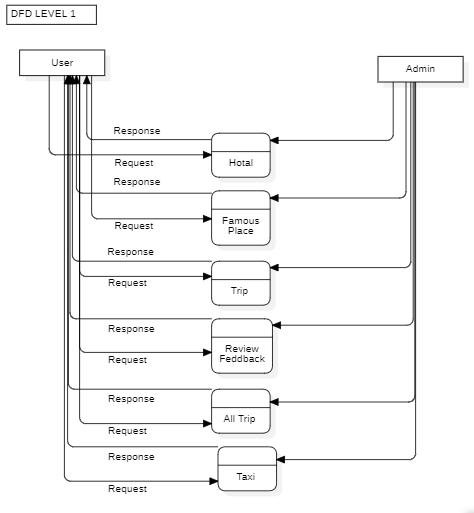
* 1. Use Case Diagram :



* 1. DFD Level 0



* 1. DFD Level 1



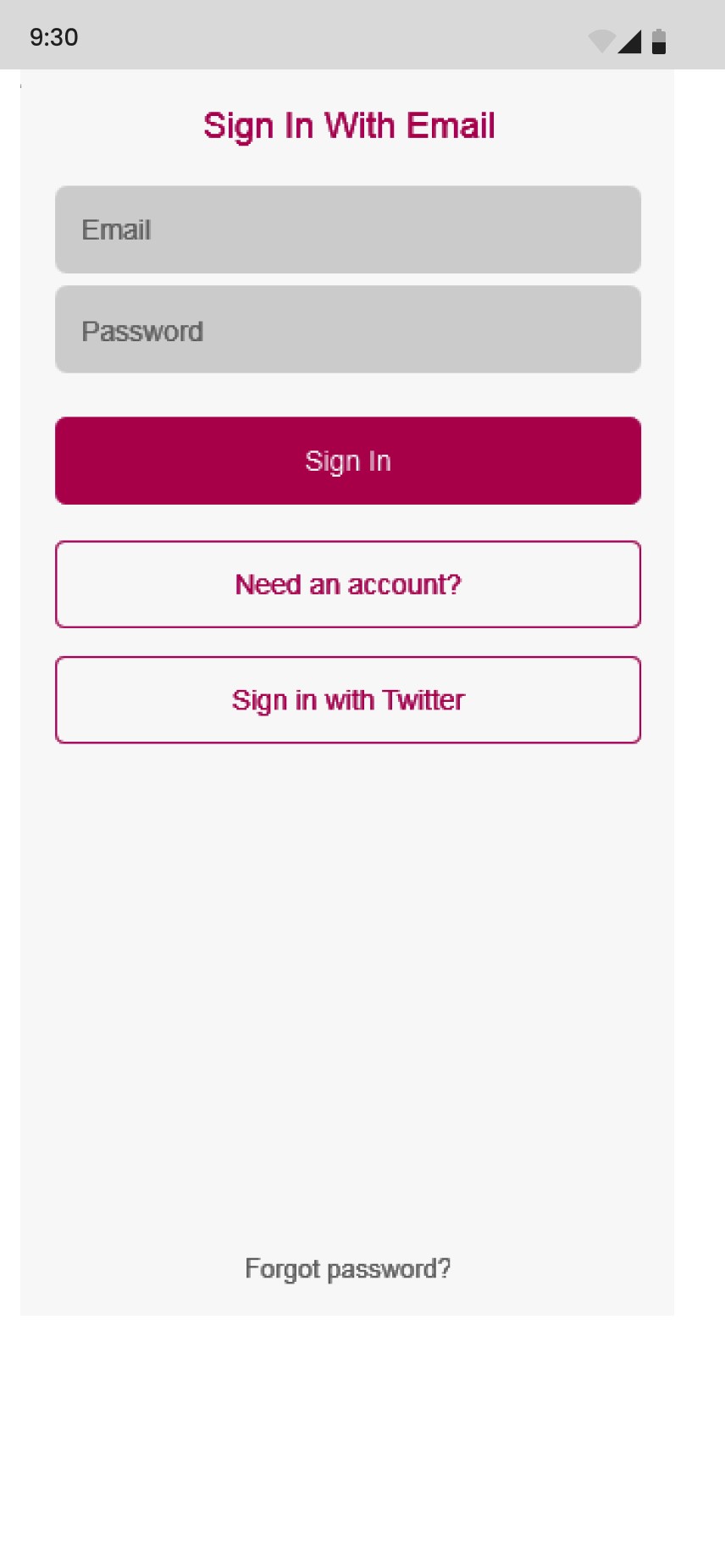
1. **Testing :** 
   1. Black Box Testing :
      * + - Black-box testing is a method of software testing that examines the functionality of an application without peering into its internal structures or workings.
          - This method of test can be applied virtually to every level of software testing:

unit, integration, system and acceptance.

5.1 White Box Testing :

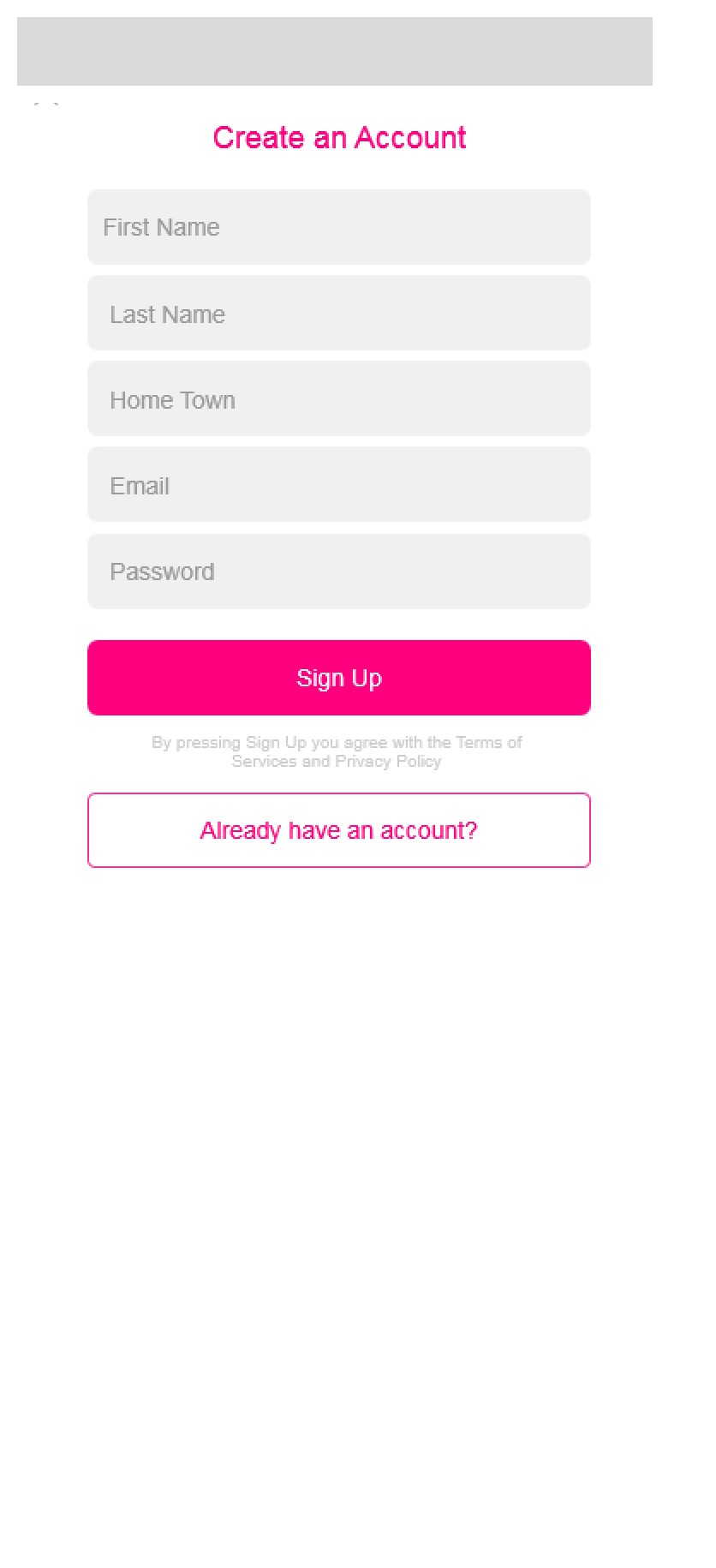
* + - * + White-box testing is a method of software testing that tests internal structures or workings of an application, as opposed to its functionality.
        + In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases.

1. **Wireframes :**



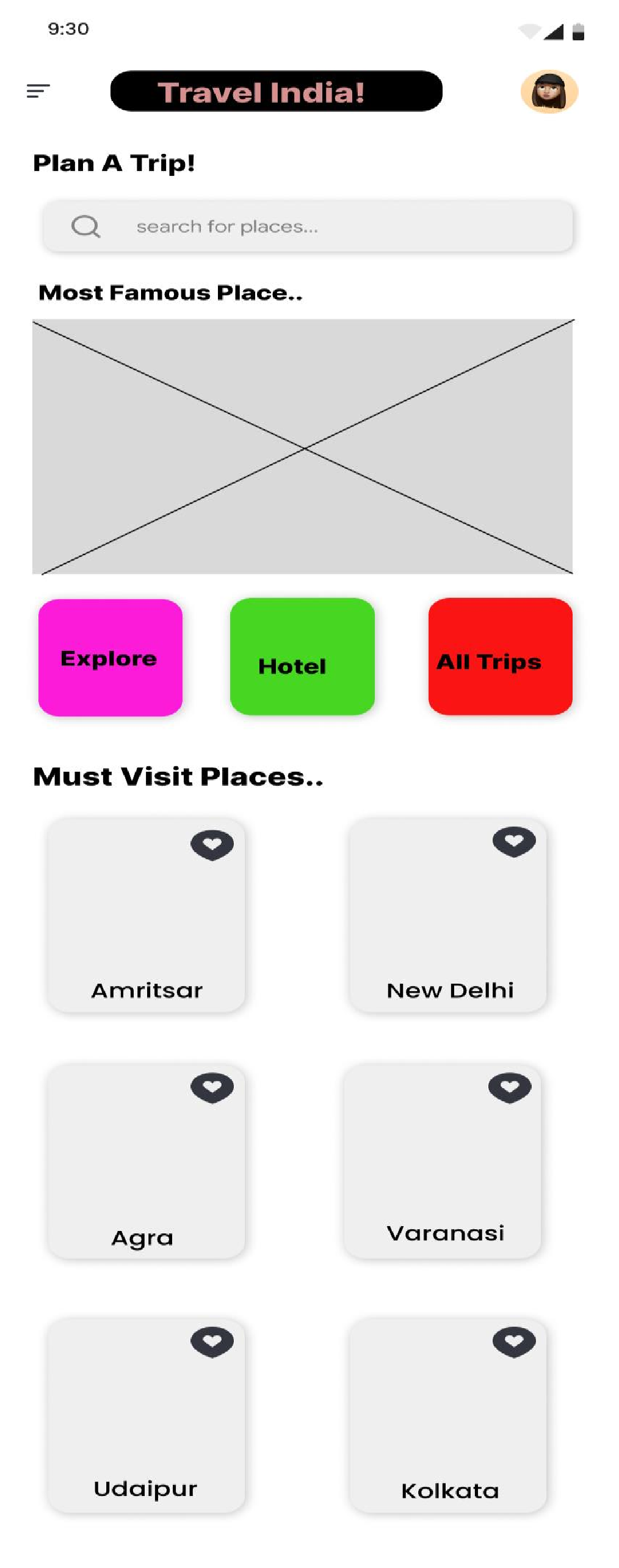
**Fig. Login Screen.**

1. This is Our Login Screen . By This The app Can Validate User.



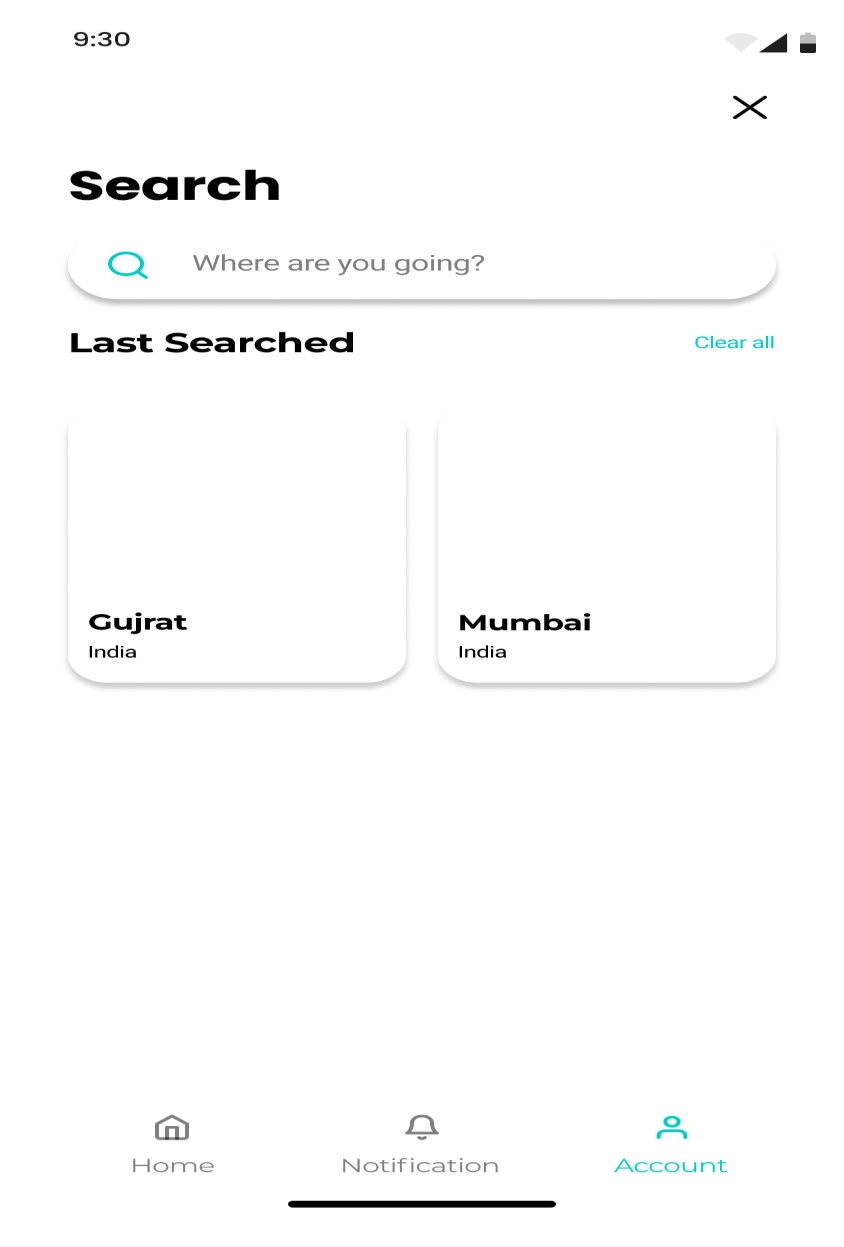
**Fig. Register Screen.**

1. This Our Registration Screen Form This Our App Can Register The user.



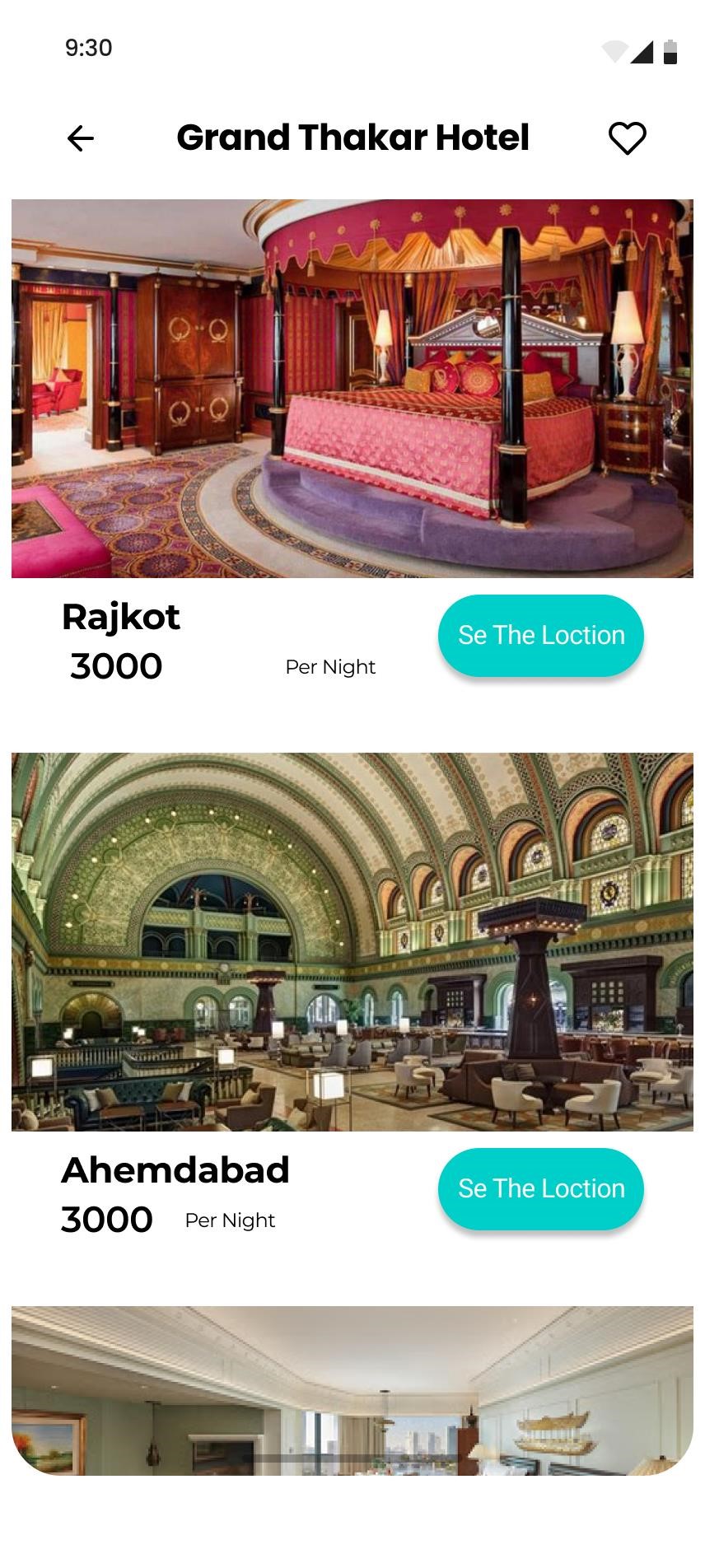
**Fig. Main Dashboard.**

1. This Is our Dashboard From This The app User can Use Many Functionality Like Explore ,Hotel,All Trip Functionality And Other Functionality



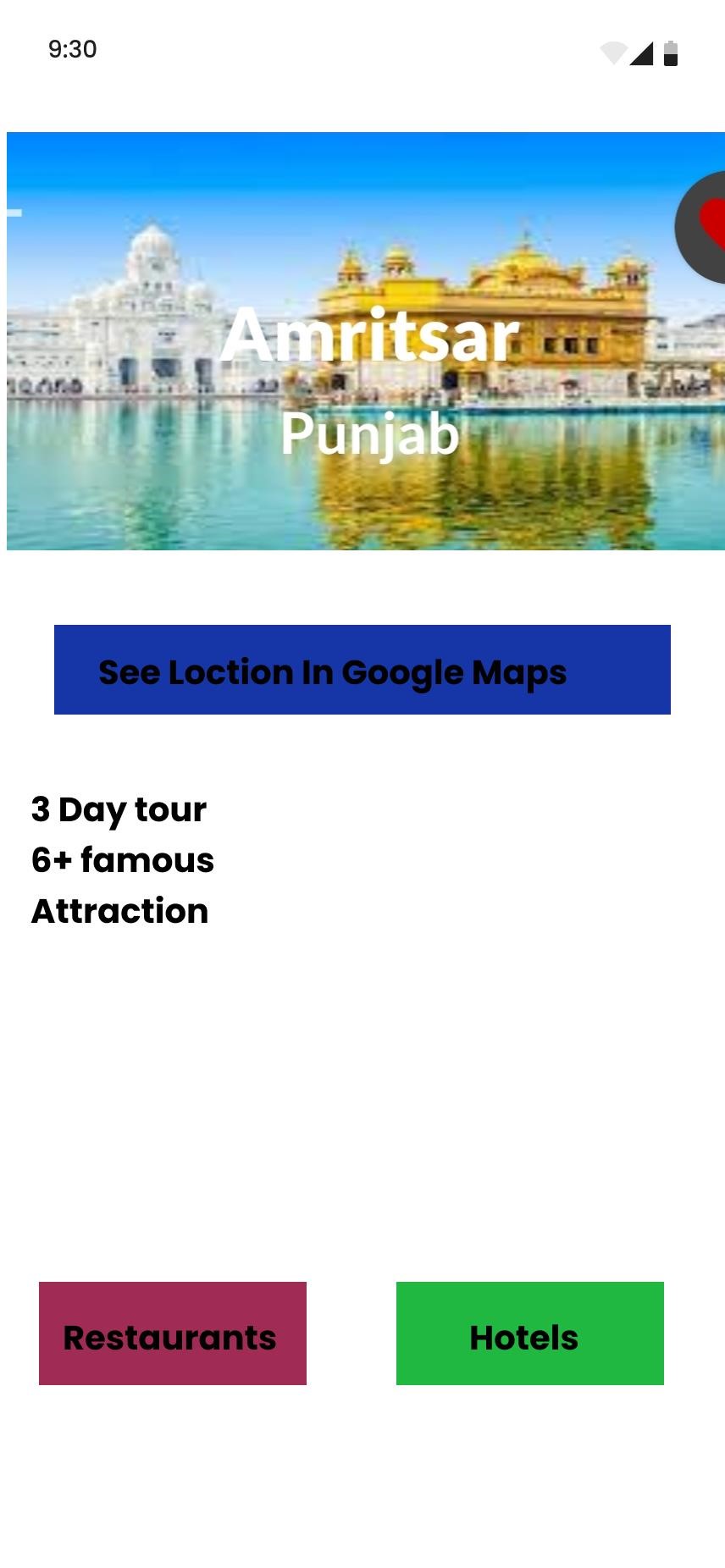
**Fig. Explorer**

1. Exlore Function By using This The user can use Map Functionality



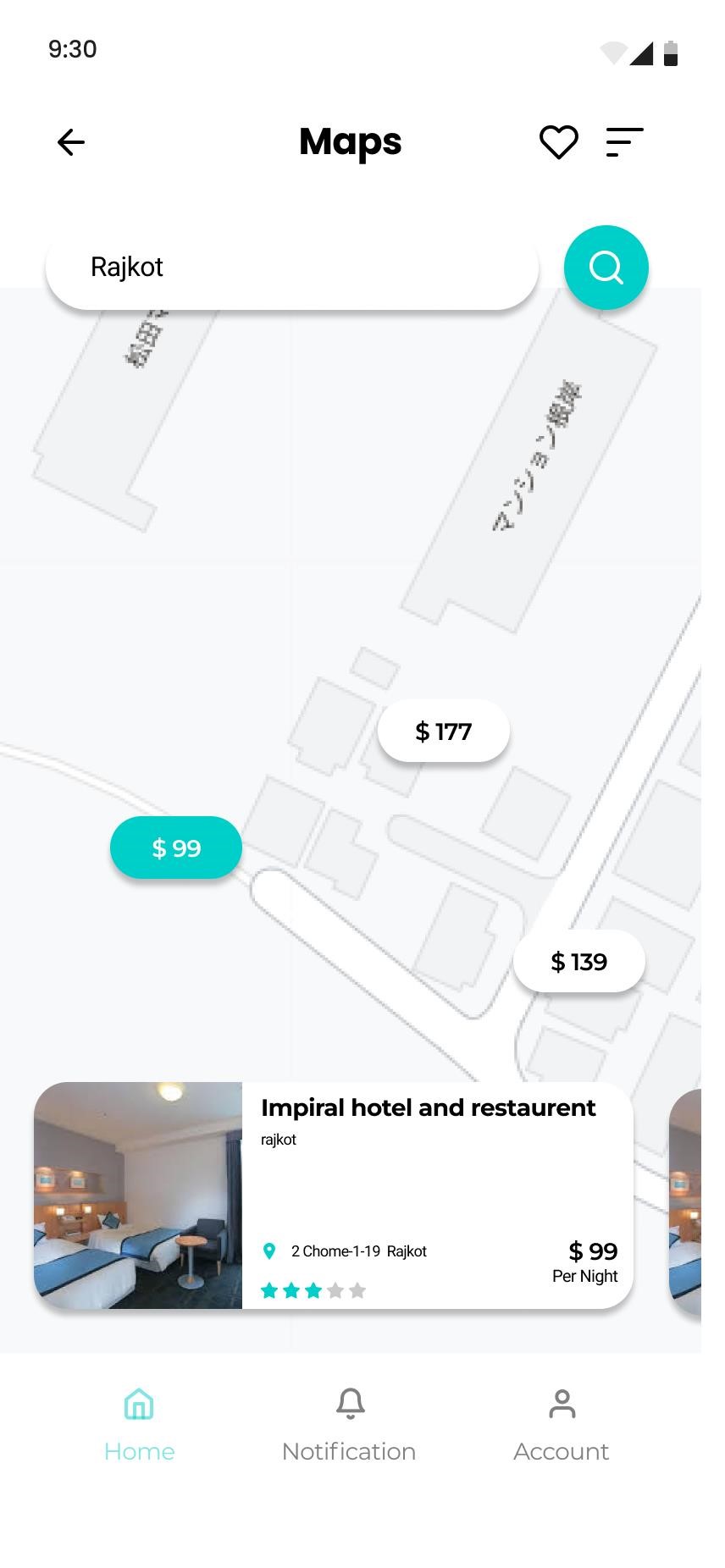
**Fig. Hotel**

1. This is Hotel Functionality By Using this User can see near best Hotel.



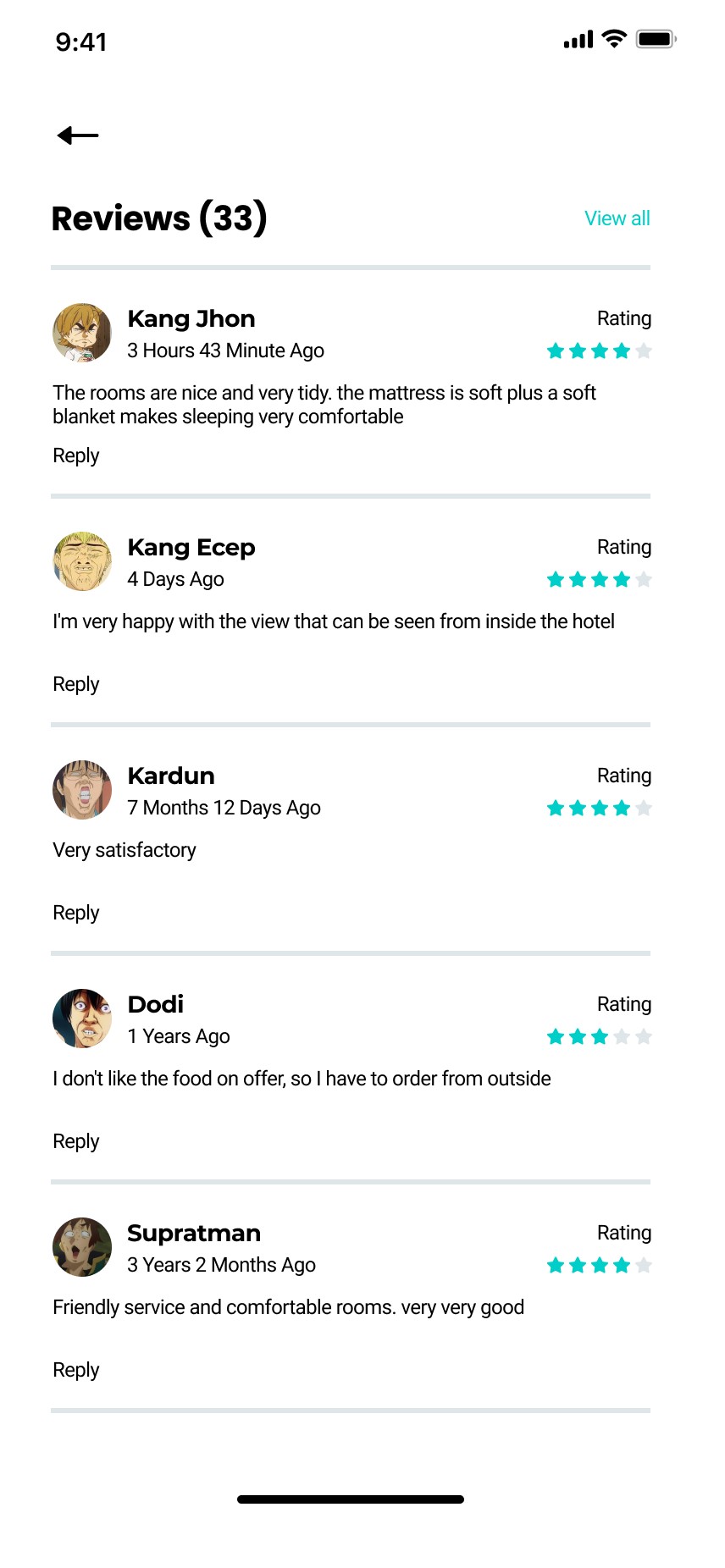
**Fig. Must visit Places**

1. This is our Must Visit place screen By using this user can see The Most Visit Place



**Fig. Map**

1. This Is Our Map Functionality By Using This User Dirct The famous place



**Fig. Reviews**

1. This Is our Review and Feedback Screen From This User can See The Feedback And Give Feedback.

**{Note : this wireframes are from prototype version: 1.2.4.36, the actual wireframe might have several difference in color or structure.}**

**Thank you**