INTELLIGENT MOBILE TRAVELLING GUIDE

Submitted by:
Khushi Mishra- 4333
Mohini Dave-4344
Abhishek Tiwari-4342

Group number G225

BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY
2023-24

UNDER THE GUIDANCE OF
Dr. S. K. Singh

THAKUR
TRUSTS

Thakur College of Science and Commerce, Thakur Village, Kandivali (E),
Mumbai - 400101

THAKUR TRUSTS

Thakur Educational Trust's (Regd.)

THAKUR COLLEGE OF SCIENCE & COMMERCE

AUTONOMOUS COLLEGE, PERMANENTLY AFFILIATED TO UNIVERSITY OF MUMBAI

NAAC Accredited Grade 'A' (3rd Cycle) & ISO 9001: 2015 (Certified)

Best College Award by University of Mumbai for the Year 2018-2019

tcsc

CERTIFICATE

This is to certify that project entitled "**Intelligent Mobile Travelling Guide**" is undertaken at the Thakur College of Science and Commerce by

Mr. Abhishek Tiwari Roll No. **4342**, **Ms. Mohini Dave** Roll no. **4344**,

Ms. Khushi Mishra Roll No. **4333** of Class **TY B.Sc. IT** Div. **A** has satisfactorily completed the Final year Project in the department of Information Technology during the academic year **2023-24**.

____________________ ____________________

Internal Examiner External Examiner

____________________ ____________________

Project Guide Head of Department

____

College stamp

Date ______________

Acknowledgement

We take much pride in presenting our project. During the development of the project, if we would like to mention the name of certain individuals without whose assistance, our project would have been difficult undertaking indeed, we hereby pleased to have this opportunity to express our deep sense of gratitude for my project on "**INTELLIGENT MOBILE TRAVELLING GUIDE**", further we are very thankful to our Head of Department "**Dr Santosh Kumar Singh Sir**", whose valuable guidance and suggestions helped us in accomplishing our project. Last but not the least, we would like to thank all our friends, family member for individual help.

INDEX

Diagram Index

1. Introduction

In today's fast-moving world, travel has become a big part of our lives. It's all about making memories and experiencing new things. That's where our Intelligent Mobile Traveling Guide comes in. It's not just another travel app it's a smart and easy-to-use tool that combines technology and your personal preferences to make your travels special.

The primary objective of our application is to provide travellers the lists of destinations to explore following their visits to specific locations. Additionally, our project aims to provide users with estimated budgets required to visit these suggested destinations. Imagine an app that plans a special tour just for you in the city you want to visit. It considers how many days you'll be there. You can even get precise information about the place you're visiting, including its history, things to do, and more

So, our Intelligent Mobile Travel Guide is here to change how people travel. Our app becomes your virtual Guide on your trip by using technology and your choices. This virtual tour guide helps better understand your preferences, personal expectations, and behaviour as a tourist. Based on all these criteria, it recommends the perfect tourist places for you to visit.

Our Intelligent Travel Guide is exclusively designed for travellers exploring within India. Our app is like a friendly helper, especially when you're in a new city. We've added a special area where travellers can share their own experiences and tips. People can post what they did, what they loved, and useful info about the places they visited. Travellers can get awesome suggestions from real people who've been there and done that. Plus, tourists can share their own adventures and tips to inspire others.

Statistic-

The travel app market has been growing steadily, with Statista reporting that in 2021, the global travel and tourism industry contributed approximately 7.8 trillion U.S. dollars to the global economy.

According to App Annie's State of Mobile 2022 report, travel apps saw a surge in usage, with average monthly active users (MAUs) increasing by 15% year-over-year.

A survey conducted by Travelport Digital found that 85% of travellers use a smartphone while traveling, and 60% have at least one travel app installed on their device.

Research by eMarketer indicates that mobile travel booking is on the rise, with mobile bookings accounting for more than 40% of digital travel sales in 2021.

Travel app users are highly engaged, with data from eMarketer showing that the average time spent per day on travel apps increased by 40% in 2021 compared to the previous year.

India is a significant market for travel apps, with a growing number of smartphone users and increasing internet penetration. According to a report by Statista, India is expected to have over 800 million smartphone users by 2025.

Travelers increasingly value personalized experiences, with a study by Accenture finding that 83% of consumers are willing to share their data to enable a personalized travel experience.

Platforms that facilitate user-generated content, such as reviews and tips, are gaining popularity. According to a survey by TripAdvisor, 76% of travellers used travel review websites before booking accommodations.

Advancements in technology, including artificial intelligence and machine learning, are enhancing the capabilities of travel apps to provide personalized recommendations and seamless user experiences.

The travel industry is rebounding from the effects of the COVID-19 pandemic, with travel app usage expected to continue growing as restrictions ease and confidence in travel increases.

i. Objective and Scope

1. **Optimized Travel Planning**:
   - Our project is designed to offer users a simpler and user-friendly platform for planning their trips. Users will have the ability to select destinations, set travel dates, and define their preferences. Traditional travel planning is often a time-consuming and error-prone process Coordinating various aspects of a trip, such as selecting destinations, accommodations, and activities, can lead to missed opportunities and logistical challenges.

2. **Personalization and Recommendations**:
   - The application's intelligence will offer personalized recommendations. User preferences collected during registration, such as favourite travel categories (e.g., historical, beaches, mountains), will be integrated to create travel suggestions aligned with individual interests. Understanding the primary interests of users, the app suggests. personalized recommendations on the dashboard. Continuous analysis of user engagement helps identify categories explored more deeply, refining recommendations further to consistently prioritize user interests.

3. **Destination Information and Exploration**:
   - Users are equipped with all necessary details, eliminating the need to seek assistance from locals or pay high charges for guides. The app provides comprehensive information about the best tourist places in the city, including opening hours, closing times, entry fees, suggested activities, and the optimal times to visit.

4. **Budget Management Tools**:
   - Within this feature, users can input their destination, the duration of their stay, and their budget for their visit. The system then checks whether the selected transportation, accommodation, and restaurant options fit within their budget. In cases where the initial choices exceed the set budget, the system generates alternative solutions by calculating costs that best align with the specified duration and budget constraints.

5. **User Feedback and Continuous Improvement**:
   - Our system actively collects user feedback, allowing users to provide reviews, suggestions, and report issues. This feedback will be used to improve the application and enhance the overall travel experience for all users. Additionally, the system analyses popularly booked trip categories, prioritizing these categories in our monthly Bestsellers feature. This ensures on going refinement and optimization based on user preferences and trends.

Scope:

The scope of Intelligent Mobile Travelling Guide extends to providing users with user-friendly platform that provides to their travel preferences, ensuring a well-informed, personalized, and budget-conscious travel experience within India:

1. **Geographical Coverage**:

   - Intelligent Mobile Travelling Guide aims to cover a wide range of cities and tourist destinations across India. It aspires to provide travel planning and assistance for both popular tourist spots and lesser-known destinations. Our application aims to be your go-to travel companion across the diverse landscape of India. Whether you're planning to explore popular tourist spots or discover hidden gems, our platform ensures comprehensive travel planning and assistance. We're here to guide you through the beauty of various cities and tourist destinations across the country, making your travel experience both enriching and memorable.

2. **User Based**:

   - The application will provide a diverse user base of travellers, including tourists, adventure seekers, families, solo travellers, and explorers. The features and recommendations provided will be adaptable to different user preferences and travel styles. The Intelligent Mobile Travelling Guide caters to a wide spectrum of travellers. It's not just for tourists; it's for everyone. Whether you're a family on a vacation, a solo adventurer, or someone seeking thrilling experiences, our features and recommendations are adaptable to different preferences and travel styles. Your journey is unique, and we're here to tailor our guidance accordingly.

3. **Content Coverage**:

   - The application will provide extensive destination information, covering various types of travel experiences, including historical, natural, cultural, and leisurely destinations. It will cover a wide range of a travel locations within India. Our application is not just about destinations; it's about experiences. We provide in-depth destination information covering a variety of travel experiences – be it historical wonders, natural landscapes, cultural treasures, or leisurely getaways. Our goal is to showcase the rich diversity of travel locations within India, ensuring there's something for every type of traveller.

4. **Security and Privacy**:

   - The project scope includes the implementation of security measures, such as password hashing, to protect user data. We take your security seriously. The project scope includes robust security measures, such as password hashing, to safeguard your personal information. Your data privacy is a priority, and we want you to feel confident and secure while using our application to plan your travels.

Out of Scope:

1. **Booking and Reservation System**:

   - The project will not include a direct booking or reservation system for flights, accommodations, or other travel-related services. It will primarily focus on information and guidance. While our focus is on providing information and guidance, we won't be incorporating a direct booking or reservation system for flights, accommodations, or other travel-related services within the project scope. Our primary aim is to assist and inform, ensuring you have the knowledge needed to make your own bookings.

2. **Language Localization**:

   - Extensive language localization to support multiple languages will not be within the initial project scope but can be considered for future enhancements. Initially, our project may not extensively support multiple languages through language localization. However, this can be considered for future enhancements. We're committed to continuous improvement, and the addition of language support is on our radar for providing an even more inclusive and accessible experience in the future.

   .

ii. Problem Statement

1. **Vast Choices:**

- Problem: Planning a trip can feel like navigating a maze with countless options for destinations, flights, accommodations, and activities. This abundance of choices often overwhelms travellers, making it difficult to make well-informed decisions that align with their preferences and interests.
- Our Solution: To address this, our travel application introduces a Recommendation System. By understanding users' preferences during registration, the app narrows down choices and provides personalized suggestions. This simplifies decision-making, ensuring users receive recommendations tailored to their individual tastes.

2. **Budget Constraints:**

- Problem: Financial concerns are a common challenge during travel. Many travellers find it challenging to stick to a budget, leading to stress, overspending, or missing out on experiences they desire.
- Our Solution: The Budget Finder feature in our application is designed to empower users in managing their finances effectively. Users can set their travel budget, and the app may offer recommendations based on typical costs, ensuring users can enjoy their journey without worrying about financial constraints.

3. **Inefficient Travel Planning**:

- Problem: Traditional travel planning is often a time-consuming and error-prone process Coordinating various aspects of a trip, such as selecting destinations, accommodations, and activities, can lead to missed opportunities and logistical challenges.
- Our Solution: The Personalized Trip Creation feature uses the Traveling Salesman Algorithm to optimize travel itineraries. This not only reduces the time required for planning but also minimizes the chances of errors, providing users with efficient and well-organized travel plans.

4. **Lack of Personalization:**

- Problem: Existing travel applications tend to provide generic recommendations that do not cater to individual preferences and constraints. This lack of personalization results in less satisfying and memorable travel experiences.
- Our Solution: Our application addresses this issue through the Recommendation System. By integrating user preferences gathered during registration, the app ensures that recommendations align closely with each user's interests, providing a more personalized and enjoyable travel experience.

5. Solo travellers:

- Problem: Solo travellers and solo family travellers often face challenges when exploring new cities independently. The absence of a personalized guide makes it difficult for them to discover suitable activities and attractions.

- Our Solution: Our user-friendly application acts as a virtual assistant, offering personalized recommendations for solo travellers. This ensures they have a more enriching and enjoyable solo travel experience, empowering them to explore and discover with confidence.

iii. Software Requirement Specification

Functional Requirement

1. **User Registration and Authentication:**

The User Registration Module is a fundamental component of any application that involves user accounts. This module encompasses user registration, login functionality, email verification through OTP (One-Time Password), and a forgot password feature.

1. **User Registration:**

- The user registration process allows new users to create an account within the application.
- Users provide their basic information, including their name, email address, and a chosen password. The password is securely hashed and stored in the database. User input is validated for accuracy and completeness.

2. **Login:**

- The login process allows registered users to access their accounts and the features of the application.
- Users provide their registered email and password. The system validates these credentials against the stored data in the database. If the credentials are correct and the email is verified, the user gains access to their account.

3. **Forgot Password:**

- The Forgot Password feature provides a means for users to reset their password if they forget it.
- Users initiate the process by clicking on the Forgot Password link on the login page. The system then prompts the user to enter their registered email address. An email is sent to the user's email address, containing OTP. The user clicks the link or enters the OTP to verify their identity. After successful verification, the user is allowed to set a new password for their account.

2. Recommendation System:

The Recommendation Module is an important component of the application, designed to enhance the user experience by providing personalized and relevant recommendations for travel destinations and experiences. This module incorporates user preferences gathered during registration to present users with tailored recommendations.

1. User Preference Integration:

- The module integrates user preferences collected during registration to offer recommendations that align with the user's interests.
- Preferences gathered during registration (e.g., mountains, beaches, historical sites) are stored in the user's profile.
- The preferences may be weighted to ensure that recommendations align closely with the user's stated interests
- Historical, Religious, Beaches, Hill Stations, Landmarks, Wonders, Architectural, Scenic Beauty this are the Category which will be Integrated in the application.

2. Personalized Dashboard:

- The personalized dashboard on the user's home screen offers a customized view of travel recommendations.
- The user's first preference, based on their registration data (e.g., mountains), is prominently featured on the dashboard.
- Additionally, the module identifies and displays what the user has recently been exploring or viewing within the application.
- Seasonal recommendations are also showcased, offering destinations and activities tailored to the current season or time of year.

3. Seasonal Recommendations:

- Our app finds out the weather conditions at the selected destination. Based on this information, the system provides users with tailored suggestions on what to carry for their travel.
- This proactive approach ensures that users are well-prepared for the specific weather conditions at their destination, enhancing their overall travel experience and minimizing any potential challenges associated with climate variations.

3. Destination Search and Description:

The Destination Search and Description Module is a basic component of the travel application. It allows users to search for specific travel destinations and provides comprehensive descriptions, information, and insights about those destinations.

1. Destination Search:

- The primary purpose of this component is to enable users to search for specific travel destinations, whether it's a city, state, or tourist spot.
- Users enter the name of their desired destination into the search bar.
- The application retrieves destination data from its database.
- Search results are displayed, including destination names, images, and brief descriptions.
- Users can apply filters to narrow down search results, such as location, popularity, or type of destination (e.g., historical, natural, religious).

2. Destination Description:

- This component provides users with detailed information about a selected destination to help them make informed travel decisions.
- When a user selects a destination from the search results, they are directed to a dedicated destination description page.
- General information about the destination (e.g., location, climate, nearest Railway station).
- Historical and cultural context, including notable landmarks and events.
- Tourist attractions and activities available at the destination.
- Travel tips and recommendations for travellers (e.g., best time to visit).
- High-quality images and multimedia content to showcase the destination's beauty.

4. Category based Sorting and Filtering:

Category-based sorting and filtering on the dashboard is a user-friendly feature that enhances the travel application's usability and helps users discover and manage content more efficiently. This feature is primarily located on the user's dashboard, allowing them to refine and organize the content they see based on predefined categories.

The primary purpose of category-based sorting and filtering on the dashboard is to simplify the user's navigation through the application's content, making it easier and more relevant to their interests.

1. **User Dashboard**:

   - Users access the application's dashboard upon logging in, where they can view various content related to travel destinations, activities, and other features.

2. **Category Selection**:

   - Within the dashboard, users are presented with a set of predefined categories that help them filter and sort the content. These categories can encompass destination types e.g., beaches, mountains, historical sites and more.

3. **Updated Content Display:**

   - The content on the user's dashboard is dynamically updated based on the selected categories and sorting preferences. Users can see a list or grid of destinations, activities, or recommendations that match their criteria.

5. Personalized Trip Creation:

The Personalized Trip Creation feature is a key component of the travel application, empowering users to plan and customize their travel experiences according to their preferences, interests, and constraints.

The primary purpose of Personalized Trip Creation is to allow users to design and organize their travel itineraries to make the most of their trips. It enables users to plan their journeys in a way that aligns with their unique preferences and requirements.

1. **User Profile and Preferences**:
   - Before creating a trip, users are encouraged to complete their profiles, providing information such as their travel preferences, budget, the number of travellers, and the duration of the trip.

2. **Destination Selection**:
   - Users can search for and select the destination(s) they intend to visit. This may involve entering the name of a city or choosing from a list of recommended destinations.
   - The application provides a brief overview of each destination to help users make informed choices.

3. **Traveling Salesman Algorithm:**
   - Incorporates the Traveling Salesman Algorithm to solve the problem of finding the most efficient route between multiple destinations.
   - Ensures the generated trip plan follows to the algorithm's optimized path, providing users with a practical and efficient travel itinerary.

6. Budget Finder:

The Budget Finder feature is a valuable component of the travel application designed to help users identify and manage their travel budgets effectively. It assists travellers in planning and tracking their expenses throughout their journey. Here is a detailed explanation of how the Budget Finder feature works:

The primary purpose of the Budget Finder feature is to provide users with the tools and information they need to set, manage, and stay within their travel budgets, ensuring a financially stress-free and enjoyable trip.

1. **Budget Setup:**
   - Users initiate the process by setting their travel budget for the entire trip or for specific categories such as accommodation, transportation, dining, and activities.
   - The application may offer recommendations based on typical costs for the chosen destination and trip duration.

2. **Expense Categories:**
   - Users categorize their expected expenses, detailing items such as flights, hotel stays, meals, transportation, and various costs.
   - The application may offer predefined categories, but users can also create custom categories based on their specific needs.

7. Feedback Module:

The Feedback Module is a vibrant component of the travel application that facilitates user engagement, satisfaction, and continuous improvement. It enables users to provide feedback, reviews, and suggestions about their travel experiences, destinations, and the application itself.

The primary purpose of the Feedback Module is to collect and manage user feedback, offering travellers a platform to express their opinions, rate their experiences, and make recommendations. This feedback contributes to improving the overall travel experience and application functionality.

1. **User-Initiated Feedback**:
   - Users have the option to provide feedback within the application. This can include after visiting a destination, completing an activity, or using a feature of the app.
   - Users can initiate the feedback process through a dedicated Provide Feedback Section.

8. Admin Module:

The Admin Module serves as control centre for managing and optimizing the travel application.

1. **User Analysis**:
   - In the admin module, the system administrator, an intelligent entity, utilizes advanced analytics to analyse the ratios and percentages derived from user-personalized trips.
   - The purpose is to identify which categories of trips are more frequently created during the current month. Subsequently, the intelligent admin strategically displays these preferred categories on the user dashboard, specifically within the "Bestseller" section.
   - This not only informs users about popular choices but also highlights the categories that are currently trending and favoured by tourists within India.

2. **Error Reports and Feedback Report:**
   - To give Admin a report of all the error occurred During any operation of functionality so that Admin can resolve it and make it efficient.
   - This Include Storing all the Exception occur in database and showing in interface to the Admin.

3. **User Demography Report:**
   - To scale our application the intelligent admin conducts geographical analysis to find the concentration of users in specific locations. By counting and visualizing user presence on a map, the admin gains insights into the distribution of user activity across different regions.

9. Community Post:

The Community Post module serves as a platform for users to share their travel experiences, photos, and updates with the wider community. It encourages user engagement and interaction within the travel application.

1. **Accessing the Community Page:**
   - Users navigate to the Community Page from the bottom menu on dashboard.
   - Upon entering the Community Page, users are presented with a feed of posts from other users.
2. **Posting a Community Update:**
   - To share their experiences, users start the posting process by clicking on the Add or Create Post button.
   - The application prompts users to upload images.
3. **Image and Description Selection:**
   - Users can select the images from their device gallery.
   - Additionally, users have the option to add a descriptive caption context to their post.
4. **Uploading the Post:**
   - After selecting images and adding a description, users click the Upload button to submit their post to the Community Page.
   - The system verifies that both images and descriptions are provided before allowing the upload to proceed.
5. **Live Display on Community Page:**
   - Successfully uploaded posts are immediately displayed on the Community Page in real-time.
   - Other logged-in users can view these posts, for community engagement.

v. Hardware and Software Requirement

Hardware:

- RAM – 8 GB
- SSD – 1 TB
- Processor – i5 12 generation

Software:

- Operating System – Window 10/11
- Front End – React Native
- Back End – Python (Flask)
- Database – MongoDB Compass
- Software for Configuration – Visual Studio Code & Android Studio

2. System analysis and design:

i. Detailed Life Cycle

Our approach to Develop our project Intelligent Mobile Traveling Guide will be incremental development model. Incremental development is a project management and software development approach where the project is divided into smaller, manageable parts (increments). Each increment is developed and tested separately, and the project evolves incrementally with each new functionality or feature added.

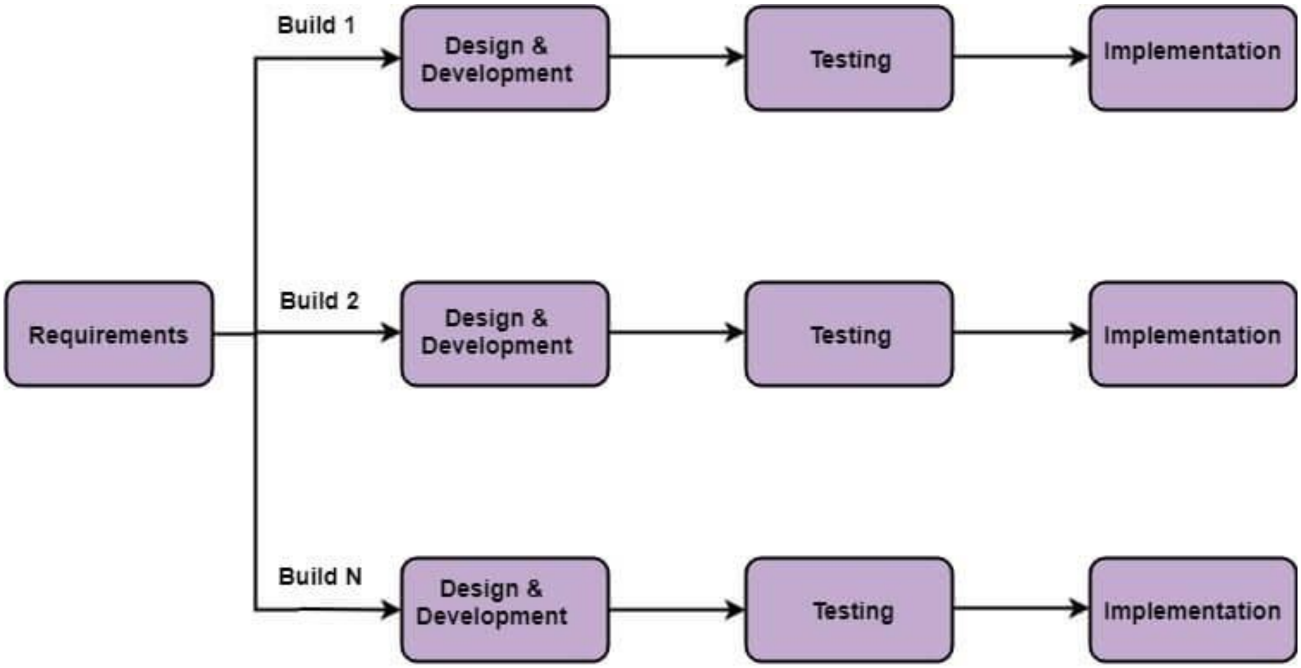


Fig: Incremental Model

Project Initiation:

The project is initiated to create an intelligent mobile traveling guide that assists users in planning their trips.

The project team is formed, consisting of Abhishek, Khushi, and Mohini

The project's scope, objectives, and expected outcomes are clearly defined.

Increment 1 - Requirement Gathering (April):

The first increment focuses on gathering project requirements. This is a critical phase where we will collect and analyses user needs and system functionalities. We created Software requirement specification and got the improvement from our guide

Increment 2 - Analysis (April to May):

In this increment, we will perform a detailed analysis of the gathered requirements to create a comprehensive understanding of the system's structure and functionality.

Create use cases, system architecture, and design specifications.

Increment 3 - Data Set Creation (May to September):

This increment focuses on creating the dataset that will be used by the mobile traveling guide to provide recommendations and information.

Data collection from various sources using web scraping.

Data cleaning and formatting to make it suitable for use.

Increment 4 - Documentation (May to October):

Comprehensive project documentation is essential for reference and future development. This increment involves creating detailed documentation.

Develop system documentation, including architecture, design, and user manuals.

Ensure all team members have access to up-to-date project documentation.

Increment 5 - User Registration and Authentication (May to July):

This increment involves implementing and thoroughly testing the user registration and authentication system, ensuring the security and privacy of user data.

Develop user registration and authentication functionalities.

Increment 6 - Dashboard and Detail (August to November):

In this increment, the team works on the implementation and testing of the dashboard and detailed view features, allowing users to access travel information easily.

Develop and refine the dashboard and detailed view.

Perform usability and functionality testing.

Increment 7 - Budget Finder Module (August to October):

This increment involves implementing and testing the budget finder module, which helps users plan their trips within their budget.

Develop budget calculation and recommendation features.

Conduct extensive testing for accuracy and reliability.

Increment 8 - Trip Planning and Itinerary Generation (July to October):

The team works on the implementation and testing of the trip planning and itinerary generation features, enabling users to create detailed travel plans. Develop itinerary creation and trip planning functionalities.

Test the system's ability to generate effective itineraries.

Increment 9 - Category-Based Sorting and Filtering (November):

In this increment, category-based sorting and filtering functionalities are implemented and tested to improve the user experience.

Develop sorting and filtering options based on categories.

Ensure the system effectively filters and displays relevant information.

Increment 10 - Recommendation Module (November to January):

The team implements and tests the recommendation module, which provides personalized recommendations for users.

Develop recommendation algorithms and user preference tracking.

Increment 11 - Admin Reporting and Analytics (December to January):

In this increment, the admin reporting and analytics features are implemented, allowing administrators to gain insights into user activity and system performance.

Test the system's reporting capabilities.

Increment 12 – Community Post (February):

Development of Community Post module completely which will allow user to save and share their tour memory.

Test the posting functionality and whether the uploaded post is visible to all other user with the uploaded image or not.

Increment 13 - Integration and Final Testing (March):

This increment involves integrating all previously developed modules to create a complete and cohesive system.

A thorough and rigorous testing phase is conducted to ensure the entire system works seamlessly as a whole.

ii. Context Diagram

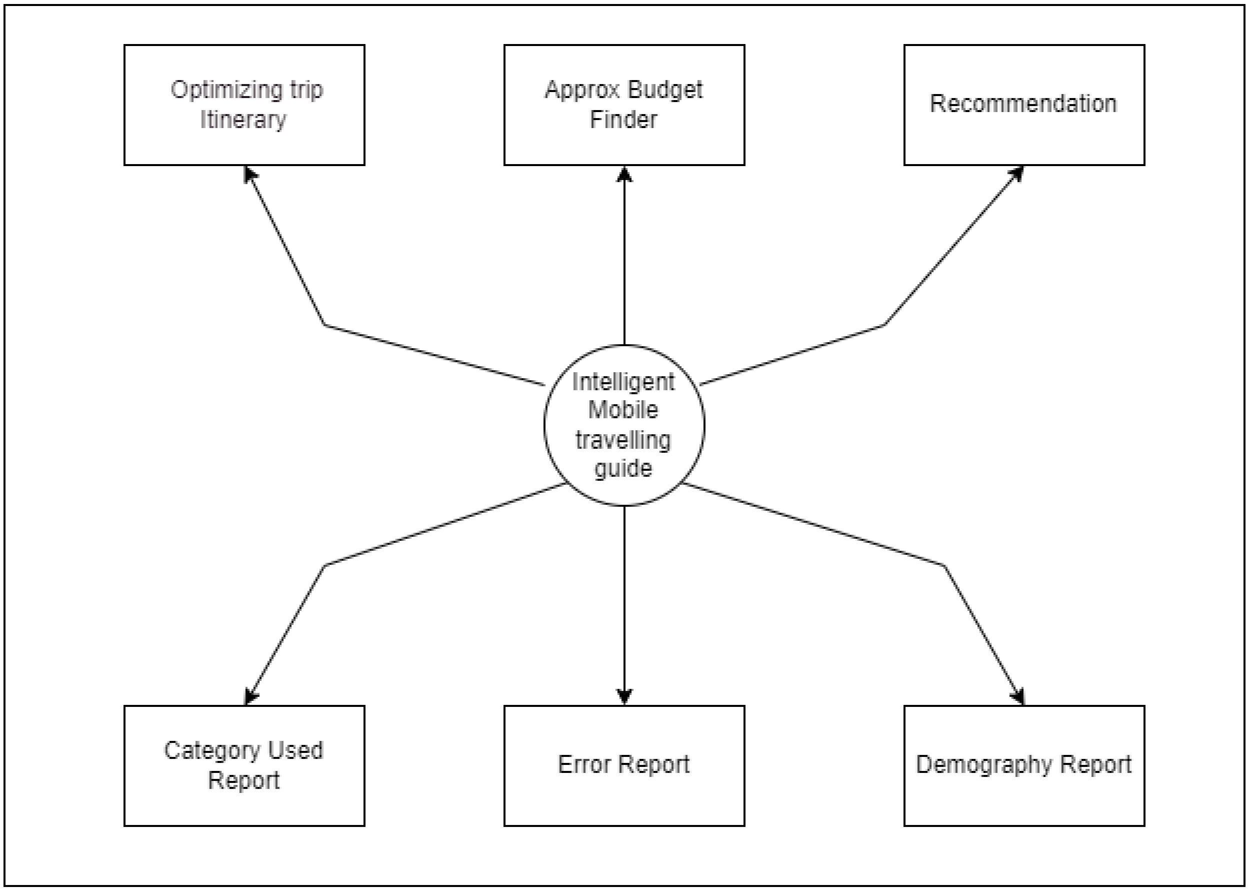


Fig 2.1 Context diagram

The context diagram for the Intelligent Mobile Traveling Guide illustrates how various components interact with users and the system. In this scenario, travellers (users) communicate their destination preferences, travel duration, and interests to the mobile app. The app, in turn, collaborates with a recommendation system to suggest places to visit and authenticates users for access. Additionally, it receives valuable assistance from a Best Tour Plan Suggestion component to create personalized travel plans and gathers place information to enrich the user's experience. The app, acting as a responsive travel guide, then provides users with customized trip plans, budget management tools, and comprehensive place details. Simultaneously, it reports user statistics and feedback to administrators while also offering users the ability to report errors or provide feedback directly. This context diagram highlights the communication flow within the Intelligent Mobile Traveling Guide, emphasizing its user-centric and interactive nature.

iii. Data Flow Diagram

Data Flow Diagram of Overall process

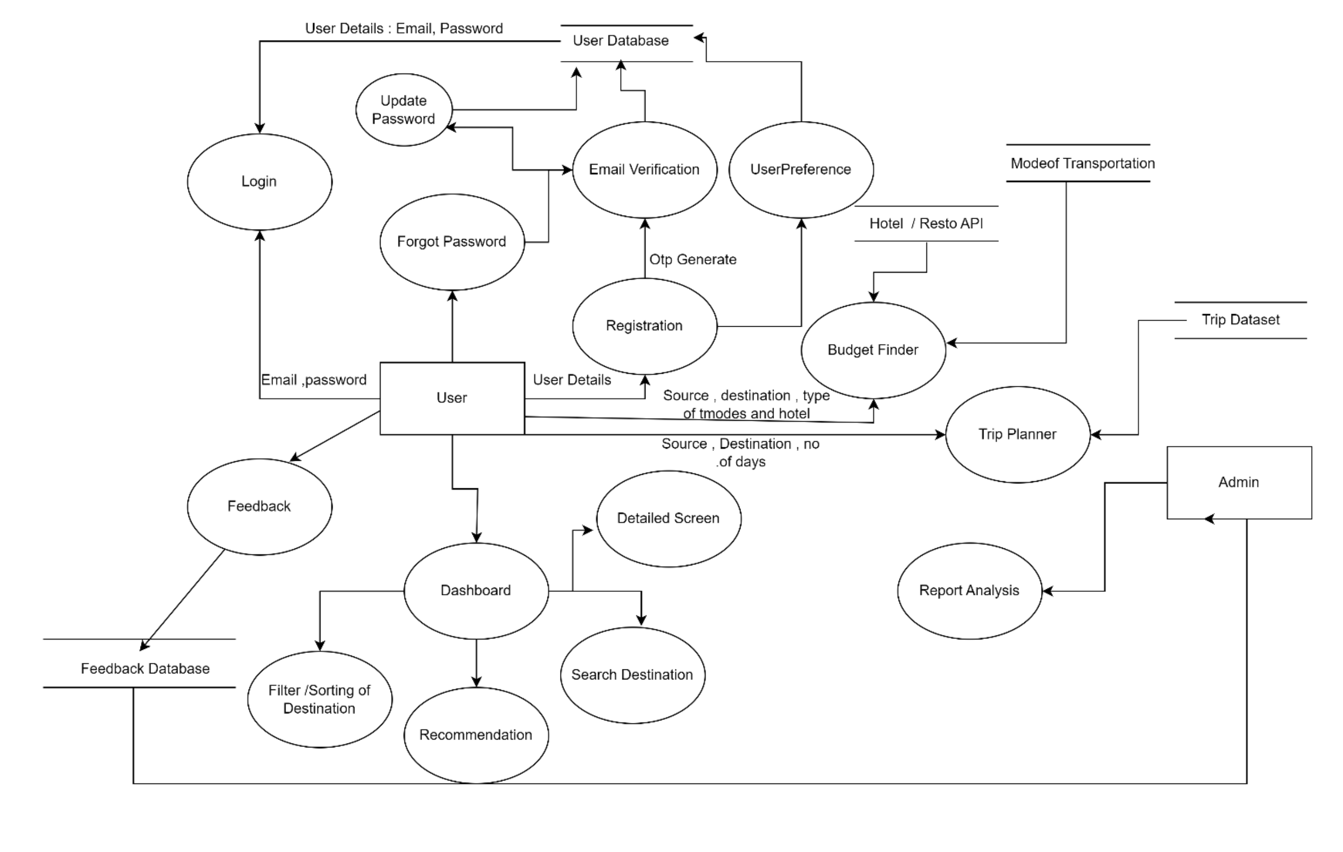


Fig 2.2 Data Flow Diagram of Overall process

The data flow diagram (DFD) you sent shows the flow of data between the different processes in a travel app. The DFD shows the following processes:

- User/Traveller Authentication: This process represents the user's login and authentication. When a user interacts with the system, they provide login credentials (e.g., username and password). The system verifies these credentials, ensuring secure access to their account.
- Trip Planning Process: The "Trip Planning" process takes user input and generates trip itineraries, provides information on destinations. Users can input their no. of days and destination, and the system processes this information to create a travel plan.
- Feedback: After the user completes a trip or uses the app, they can submit feedback regarding their experiences, which may include reviews, ratings, and comments. The feedback data is collected and processed to improve the system and services.
- Budgets Finder for the Trip: This process assists users in budgeting their trips. Users provide information on their budget constraints and preferences, and the system generates budget recommendations. It may consider factors like accommodation costs, transportation and food to create a cost-effective travel plan.

Dataflow Diagram for User Authentication:

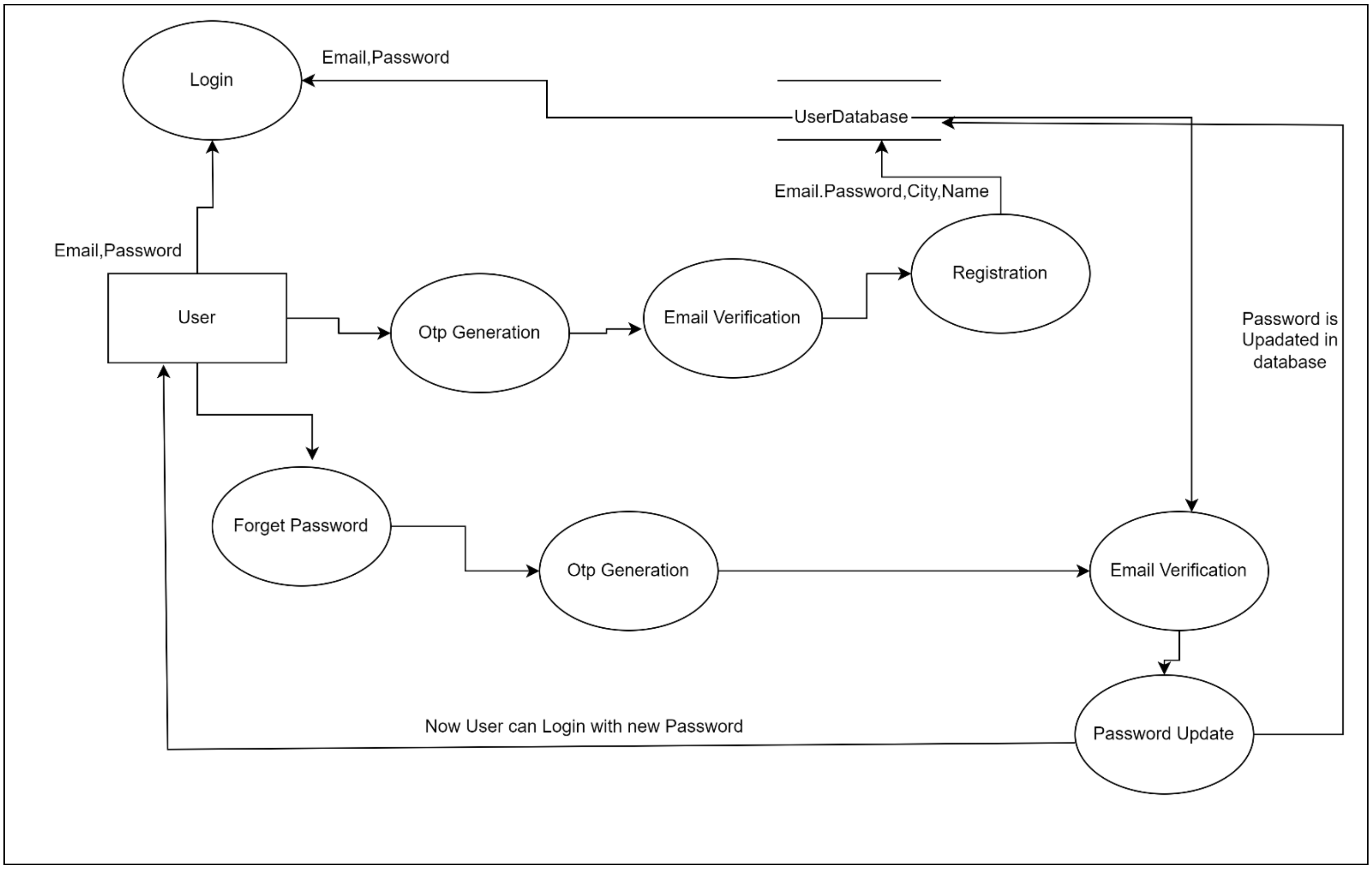


Fig 2.3 Dataflow Diagram for User Authentication

- User/Traveller Input Data: Users provide input data during the registration process, including their name, email, city, password, and password confirmation. These details are essential for creating and verifying user accounts.
- User/Traveller Registration (Email Verification via OTP): The "User Registration" process involves users entering their registration details. After submitting the registration form, an OTP (One-Time Password) is generated by the system and sent to the user's provided email address for verification. This OTP serves as a security measure to confirm the user's identity.
- OTP Verification: The OTP generated is sent to the user's email. Users are required to enter this OTP in a dedicated field to verify their email address. If the OTP is correct, the system proceeds with storing the user's information in the database.
- Database: The database stores the user's information, including their name, email, city, and hashed password. This database is crucial for user account management and authentication.
- Login: Users can log in by providing their email and password. The system validates the login credentials against the information stored in the database. If the login is successful, the user gains access to their account.
- Forgot Password: The "Forgot Password" process allows users to request a password reset. Users provide their registered email address, and the system sends an OTP to the user's email. After OTP verification, users can reset their password.

Dataflow Diagram for Feedback Generation:

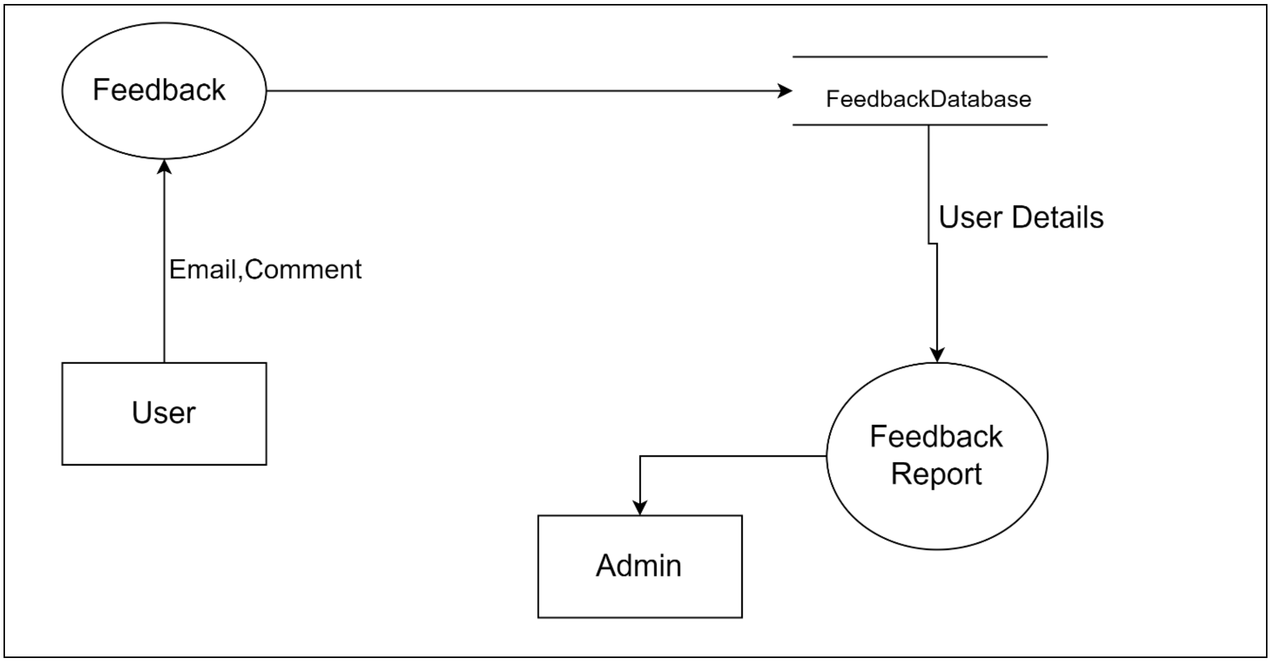


Fig 2.4 Dataflow Diagram for Feedback Generation

The DFD also shows the following data flows:

- User/Traveller feedback: This data flow carries the user's feedback from the User provides feedback process to the Store feedback process.
- Feedback: This data flow carries the user's feedback from the Store feedback process to the Analyse feedback process and the Share feedback report process.
- Feedback analysis: This data flow carries the results of the feedback analysis from the Analyse feedback process to the Generate feedback report process.
- Feedback report: This data flow carries the feedback report from the Generate feedback report process to the Share feedback report process.

Dataflow Diagram for Budgets Finder:

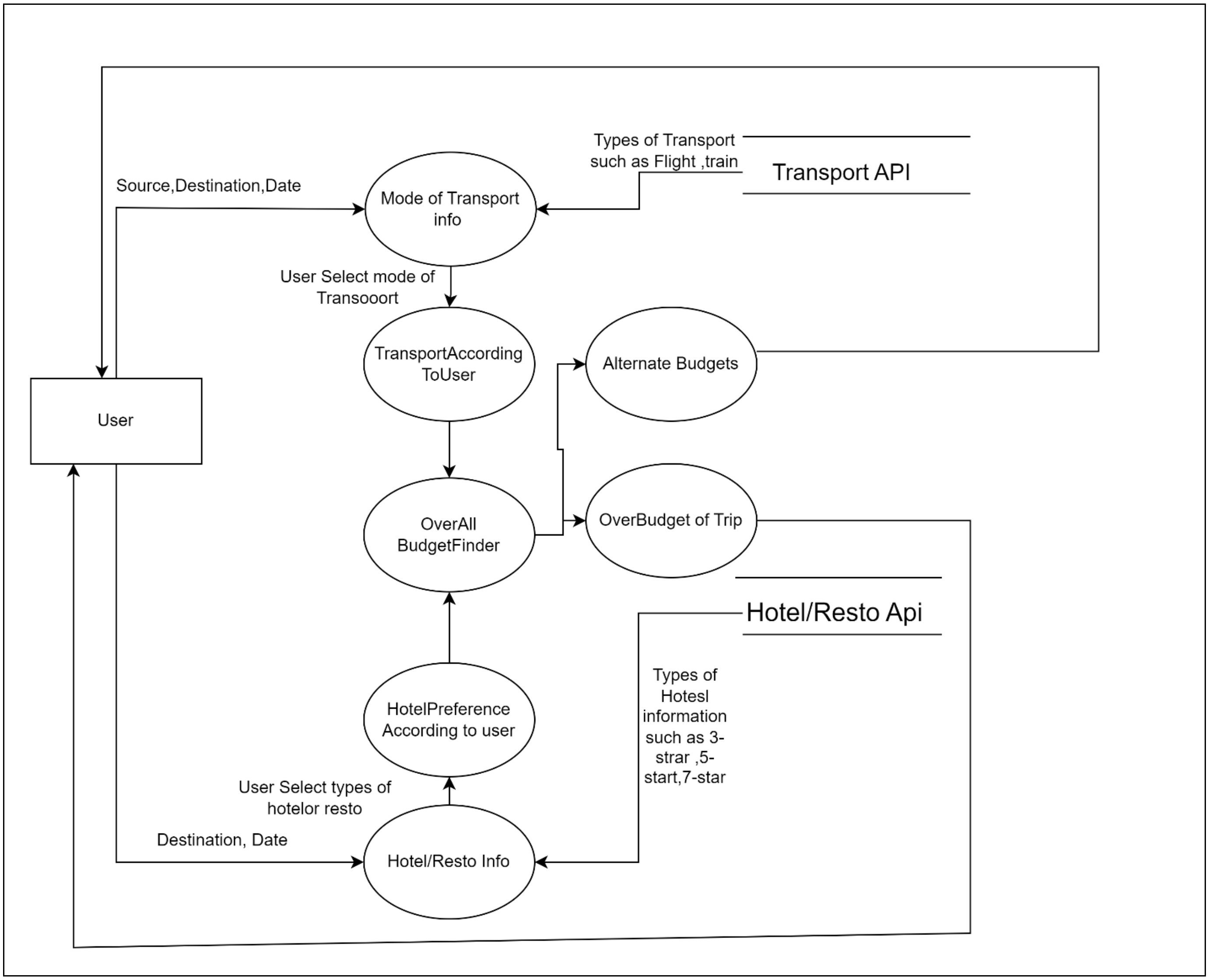

Types of Transport such as Flight ,train
Transport API
Source,Destination,Date
Mode of Transport info
User Select mode of Transooort
TransportAccording ToUser
Alternate Budgets
User
OverAll BudgetFinder
OverBudget of Trip
Hotel/Resto Api
Types of Hotesl information such as 3-strar ,5-start,7-star
HotelPreference According to user
User Select types of hotelor resto
Destination, Date
Hotel/Resto Info

Fig 2.5 Dataflow Diagram for Budgets Finder

- Budgets Finder: The "Budgets Finder" process is at the centre of the DFD. It is responsible for helping users plan their travel budgets based on their preferences.
- External APIs: The DFD shows two external entities, "Hotel/Resto API" and "Transport API," representing Application Programming Interfaces that provide data related to hotels and transportation options. These APIs are essential for obtaining information that helps in budget planning.
- Budget Generation: The "Budget Generation" process takes the user's hotel and transportation selections and dynamically calculates a budget based on the costs of these selections. It considers the user's preferences and choices when creating the budget.
- User/Traveller Preferences: The DFD illustrates that users have the flexibility to alter their budgets according to their preferences. They can make changes to their choices, such as selecting different hotels or modes of transport, and the system updates the budget accordingly.

Dataflow Diagram for Trip Planar:

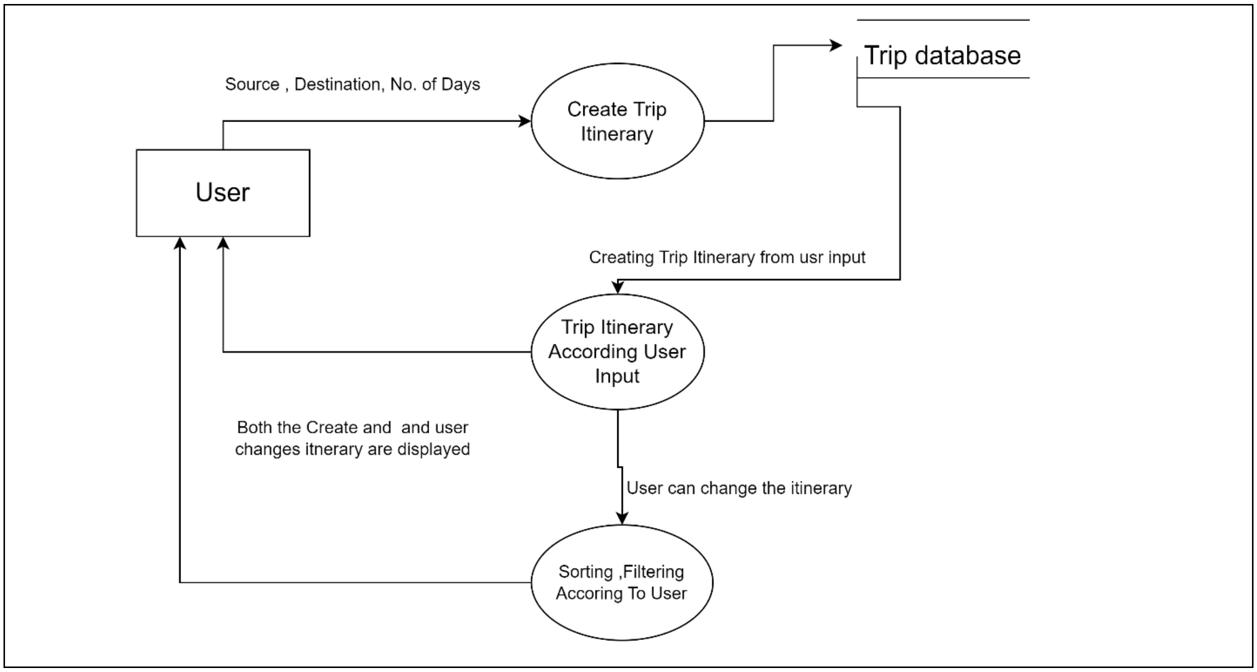

Trip database
Source , Destination, No. of Days
Create Trip Itinerary
User
Creating Trip Itinerary from usr input
Trip Itinerary According User Input
Both the Create and and user changes itnerary are displayed
User can change the itinerary
Sorting ,Filtering Accoring To User

Fig 2.6 Dataflow Diagram for Trip Planar

- User/Traveller: This represents the person interacting with the system. They provide two key inputs: the number of days they plan to spend on their trip and the name of the destination.
- Trip Planning Process: This is the core component of the system. It takes the user's input, processes it, and generates a customized itinerary that matches the number of days and the chosen destination. The generated itinerary is based on various factors like available attractions, travel time, and user preferences.
- Itinerary: The output of the system, which is the trip itinerary. It includes a day-by-day plan, activities, places to visit, and other relevant information tailored to the user's input.

iv. Entity Relationship Diagram

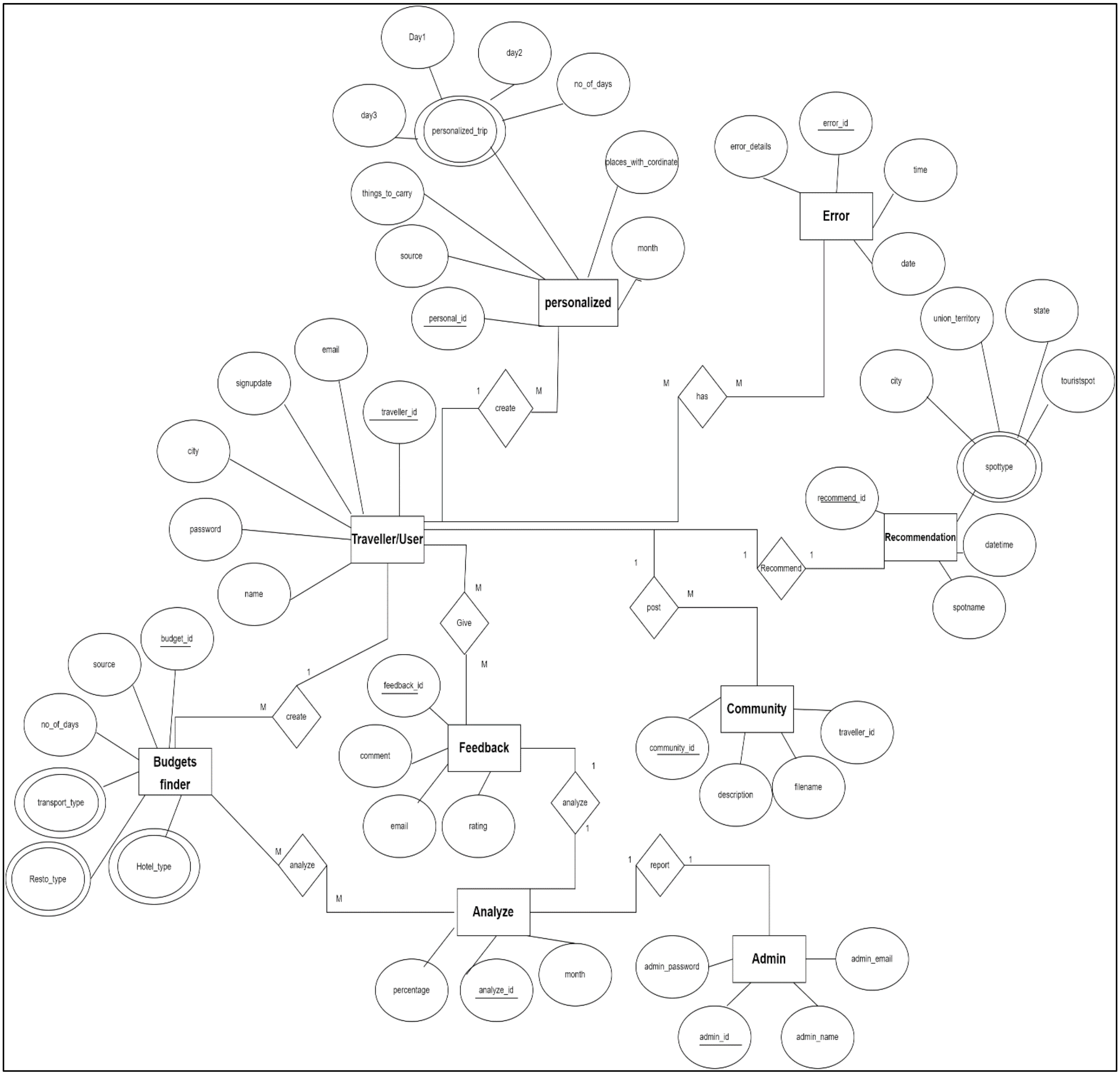


Fig 2.7 ER Diagram

The ER diagram represents an entity such as Traveller, Admin, Budget Finder, Feedback, Personalized Trip, Error, Analyse Collection, Community, and Recommendation.

- Traveller entity: Stores information about users/traveller, including their name, ID, city, email, password, and signup date.
- Admin entity: Manages administrative aspects, with attributes including admin ID, name, email, and password.
- Budget Finder entity: Facilitates budget-based travel planning, with attributes for budget ID, source, duration, transport, restaurant, and hotel preferences.
- Feedback entity: Records user feedback, including feedback ID, rating, and comments.
- Personalized Trip entity: Allows users to plan personalized trips, capturing details such as trip ID, source, duration, trip itinerary (potentially multivalued), items to carry, and trip month.
- Error entity: Logs system errors, tracking error ID, time, date, and error details.
- Analyse Collection entity: Analyses data, storing analysis ID, month, and percentage.
- Community entity: Encourages user interaction, containing community ID, filename, and description.

V. Class Diagram

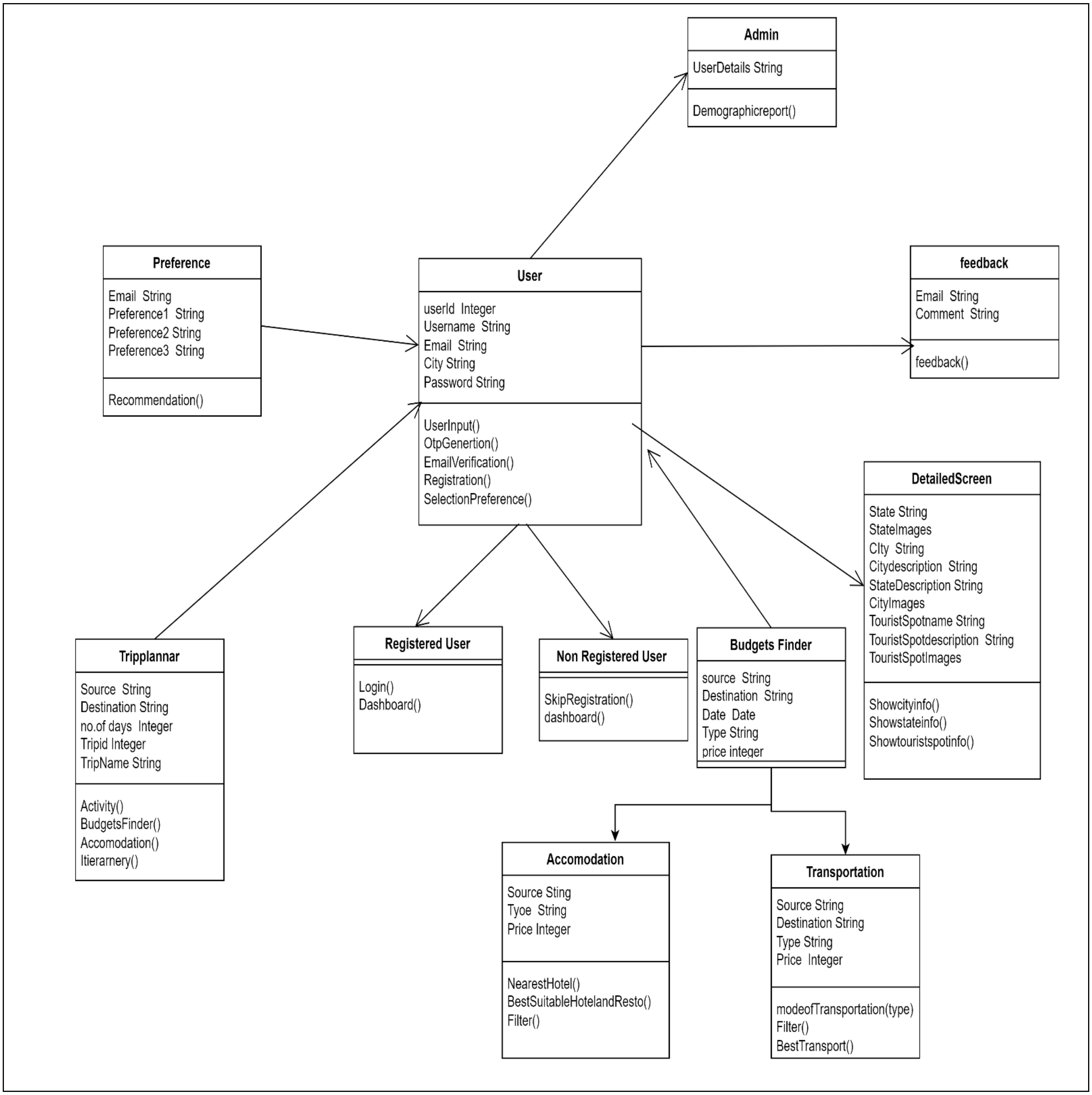


Fig 2.8 Class Diagram

vi. Use case Diagram (User)

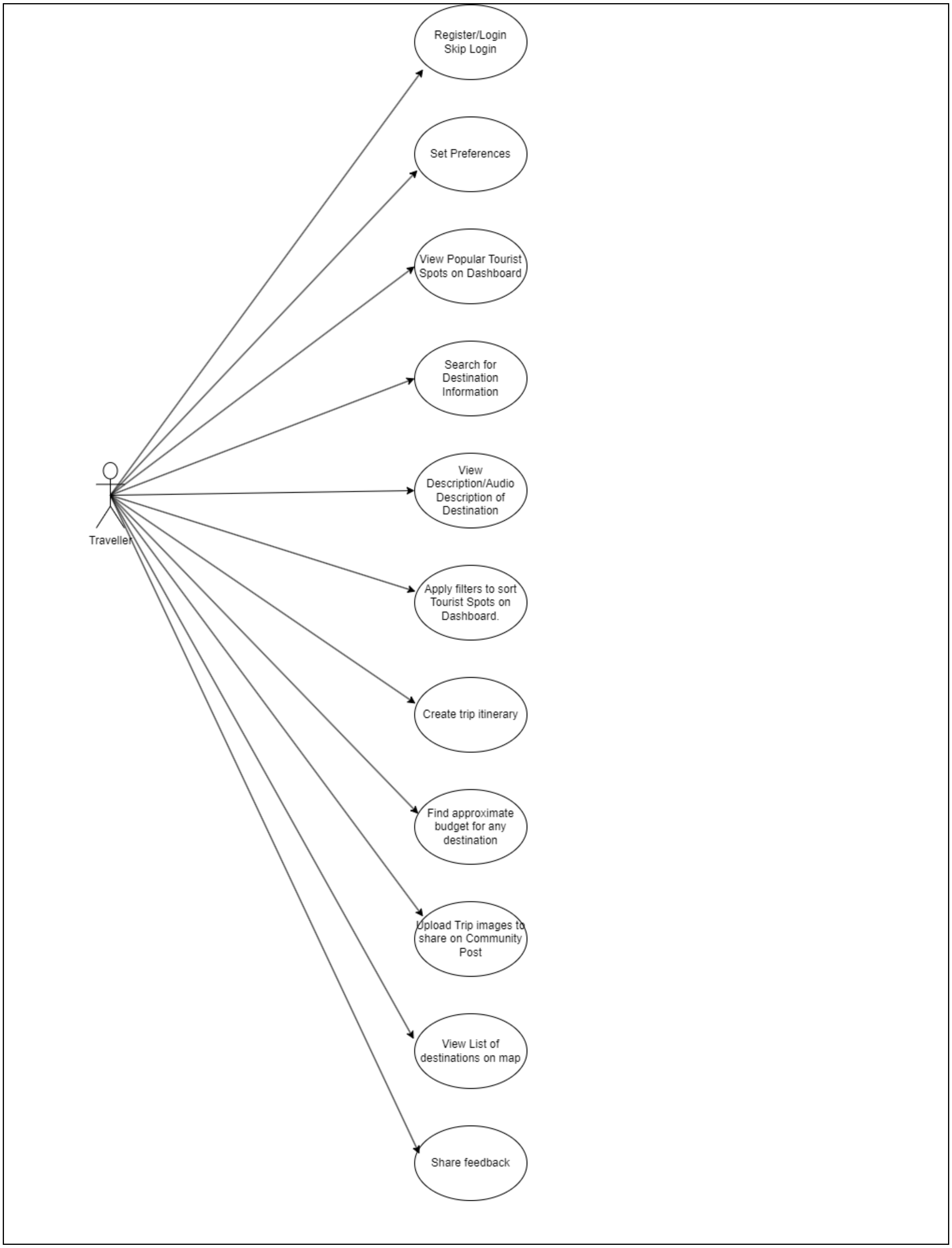

Register/Login
Skip Login
Set Preferences
View Popular Tourist Spots on Dashboard
Search for Destination Information
View Description/Audio Description of Destination
Traveller
Apply filters to sort Tourist Spots on Dashboard.
Create trip itinerary
Find approximate budget for any destination
Upload Trip images to share on Community Post
View List of destinations on map
Share feedback

Fig 2.9 Use Case Diagram (User)

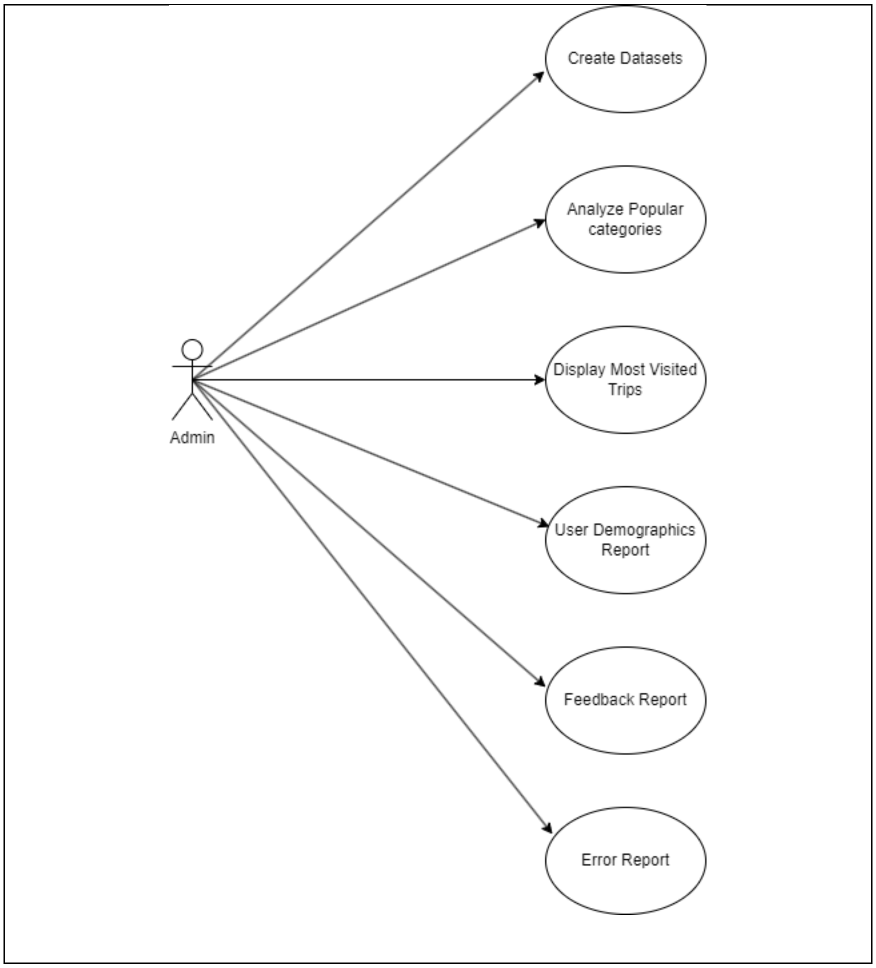


Fig 2.10 Use Case Diagram (Admin)

The use case diagram represents the interactions between two main players, the user and the admin, in the context of the application. Users have several capabilities, including the ability to sign up or log in, select their desired travel destinations, access detailed descriptions of these places, plan their trips, mark favourite locations, sort places by various categories, calculate trip costs, share valuable feedback about their experiences, and even subscribe for special features or services offered by the app. On the other side, the admin plays a crucial role in managing the app's operation. They monitor and analyse user feedback to enhance the user experience, and address any error reports submitted by users to ensure the smooth functioning of the application. In essence, users enjoy the ability to plan and personalize their travel experiences, while the admin ensures that the app runs seamlessly and efficiently, benefiting both travellers and the app's continued improvement.

vii. Collaboration Diagram

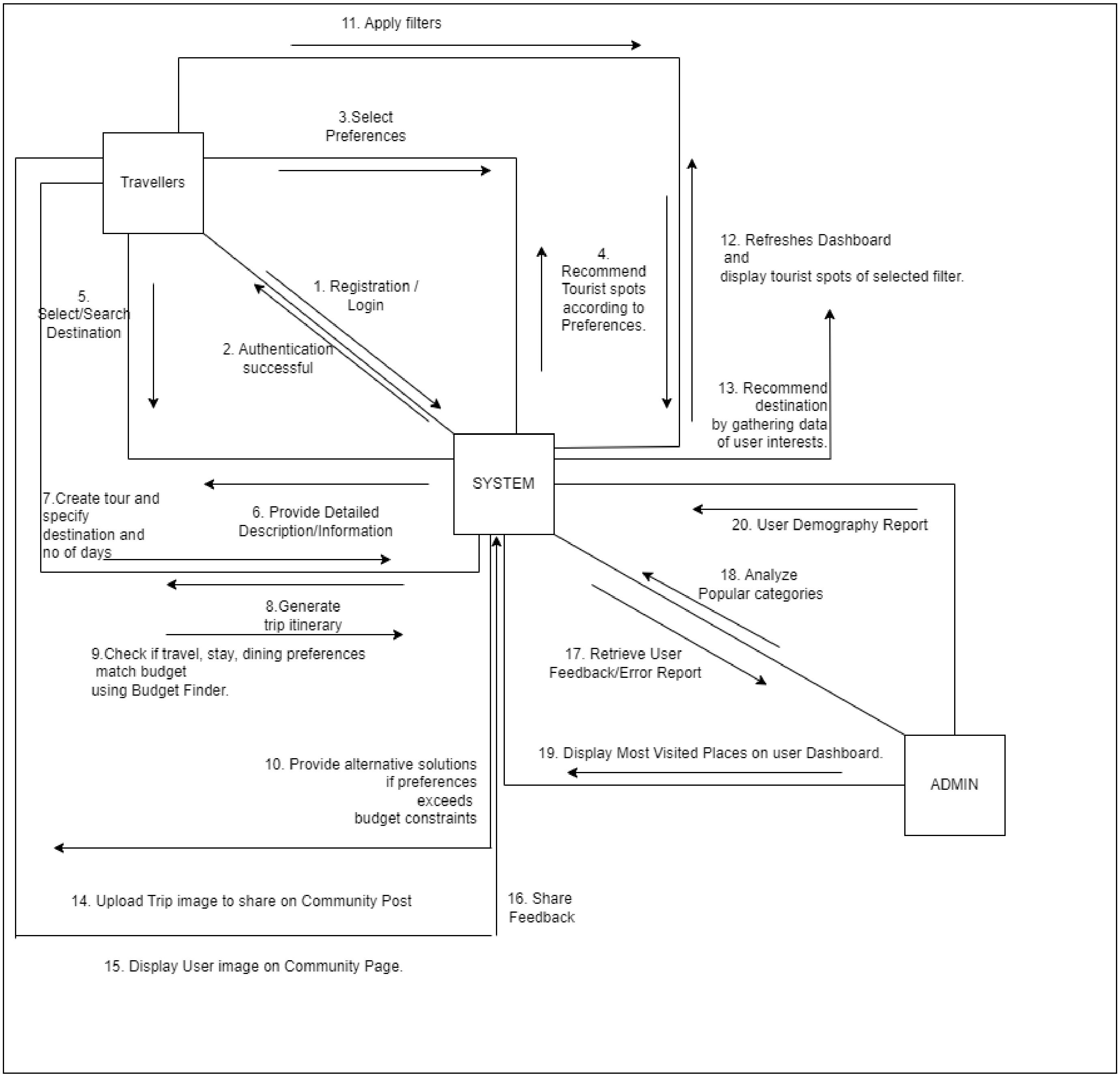

11. Apply filters
3.Select Preferences
Travellers
1. Registration / Login
2. Authentication successful
4. Recommend Tourist spots according to Preferences.
12. Refreshes Dashboard and display tourist spots of selected filter.
5. Select/Search Destination
13. Recommend destination by gathering data of user interests.
SYSTEM
7.Create tour and specify destination and no of days
6. Provide Detailed Description/Information
20. User Demography Report
18. Analyze Popular categories
8.Generate trip itinerary
9.Check if travel, stay, dining preferences match budget using Budget Finder.
17. Retrieve User Feedback/Error Report
19. Display Most Visited Places on user Dashboard.
10. Provide alternative solutions if preferences exceeds budget constraints
ADMIN
14. Upload Trip image to share on Community Post
16. Share Feedback
15. Display User image on Community Page.

Fig 2.11 Collaboration Diagram

This diagram explains the collaboration of event happening between the system user and admin. The user utilizing the intelligent tour guide application to explore a particular location or receive tour recommendations. The admin represents the management and controlling functionality of application. The admin interacts with the system by making requests, such as asking for information about a specific location, requesting a guided tour, or seeking recommendations for nearby attractions. Admins might monitor user interactions and system performance to gather data on user behaviour and system usage

viii. Component Diagram

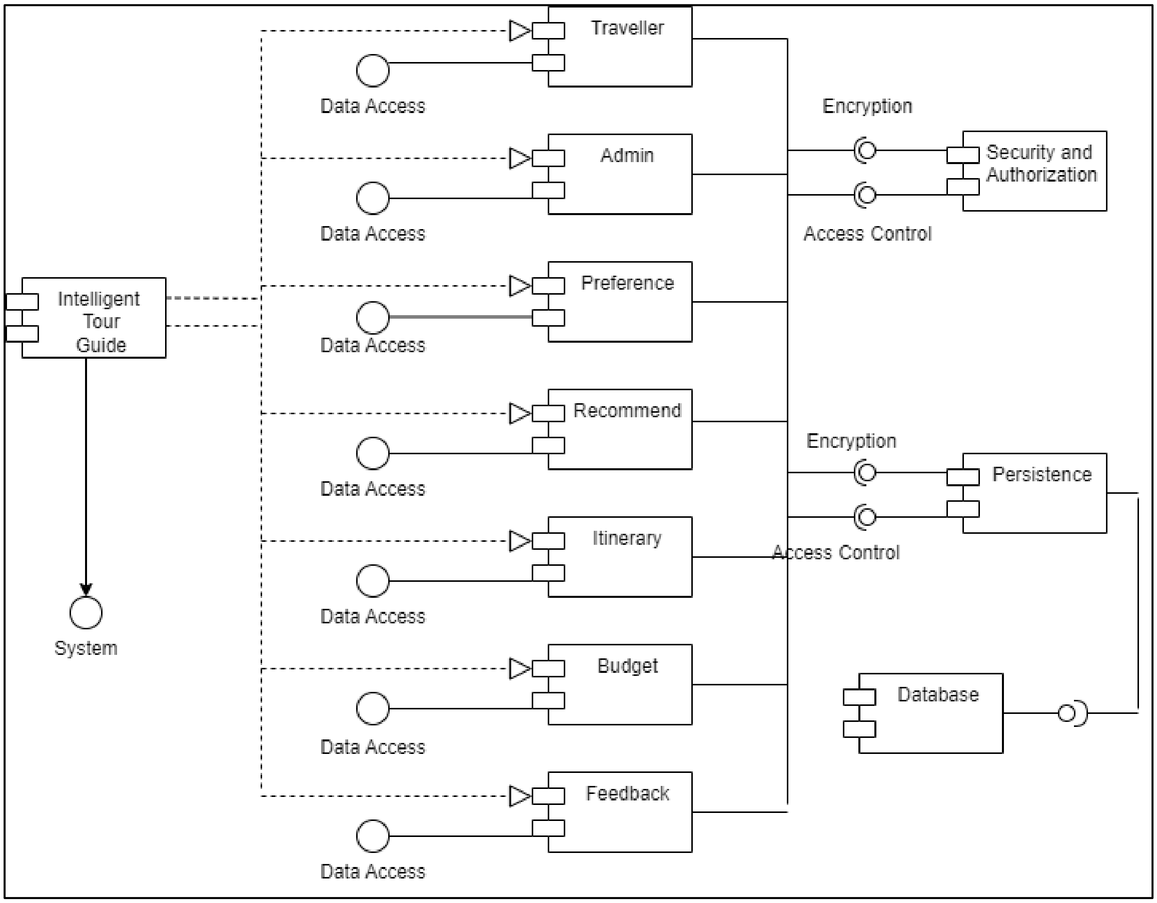


Fig 2.12 Component Diagram

The system is composed of several essential components that work together to facilitate trip planning. At the core, there's the Dashboard, which provides users with a high-level view of their trip planning process, showing upcoming trips, budget status, and itineraries. For more detailed information on specific trips, users can access the Detail Screen, which displays destination details, trip dates, the full itinerary, and the allocated budget. The Trip Creation component enables users to create new trips by entering destination, dates, and budget constraints, with the added benefit of generating a personalized itinerary. The Budget Finder is a useful tool for estimating trip costs, as users can input destination, budget limits, and trip duration to receive cost estimates. Additionally, the Recommendation System suggests cities and tourist attractions based on user preferences and popularity, making trip planning more efficient. Lastly, the Admin Interface serves as a control centre for administrators to oversee user feedback, analyse subscriptions, and monitor error reports, ensuring the system operates smoothly.

ix. Activity Diagram

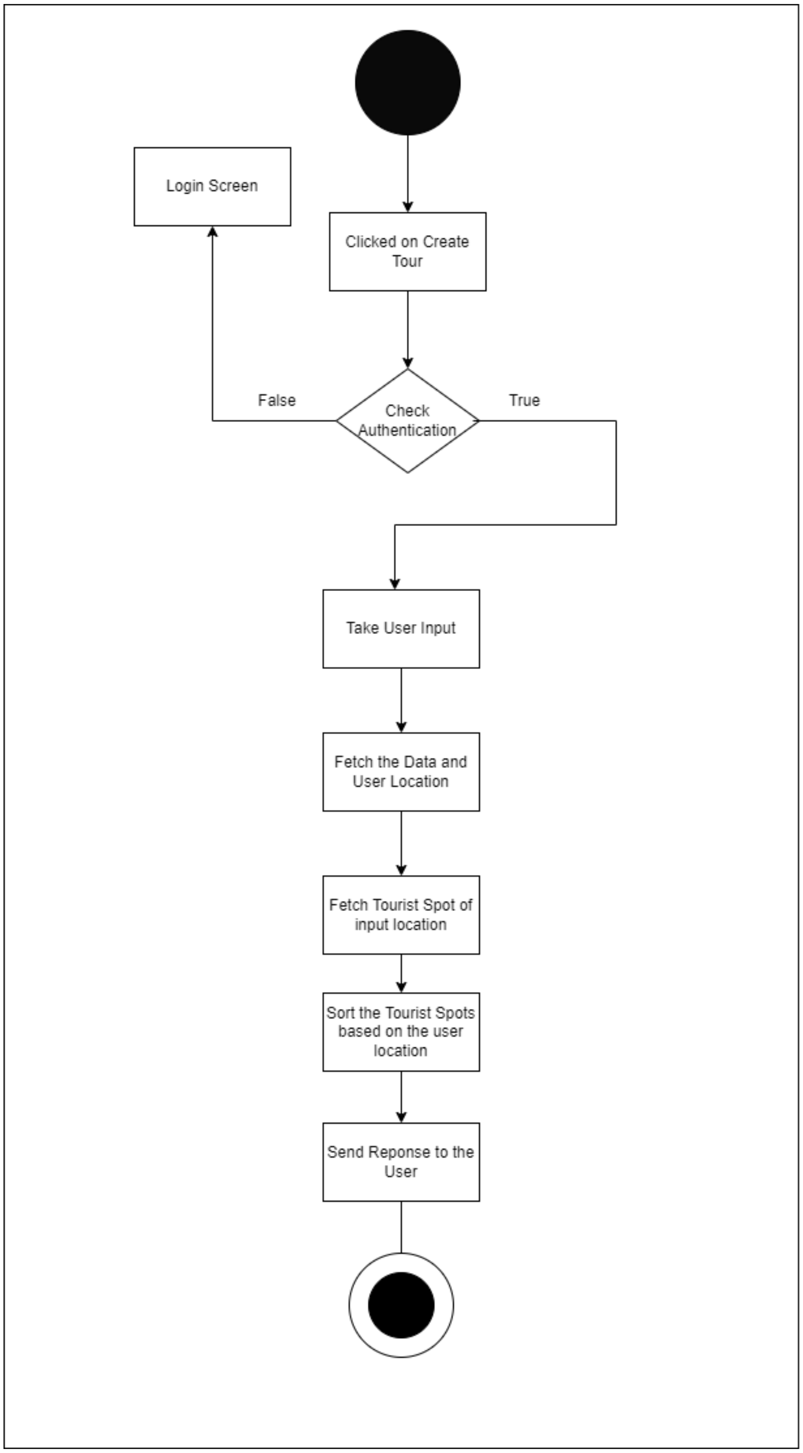

Login Screen
Clicked on Create Tour
False
Check Authentication
True
Take User Input
Fetch the Data and User Location
Fetch Tourist Spot of input location
Sort the Tourist Spots based on the user location
Send Reponse to the User

Fig 2.13 Activity Diagram for create personalised tour

The Create Tour process initiates when a user clicks on the Create Tour button within the app. This action prompts the app to engage the user in a step-by-step journey. First, the user is asked to select their desired destination and specify the number of days they intend to spend there. With this information in hand, the app's intelligence comes into play, generating a personalized trip plan tailored to the user's specific interests and preferences. This plan typically comprises a curated list of places to visit, a suggested itinerary for their travel, and an estimated budget. Once the user is content with the generated trip plan, they have the option to save it for future reference. Moreover, the app provides a sharing feature, allowing users to seamlessly share their trip plan with friends, family, or fellow travellers through email, social media, or various messaging platforms. This process streamlines trip planning, making it more accessible and enjoyable for users while facilitating effortless sharing with others.

Activity Diagram for Budgets Finder:

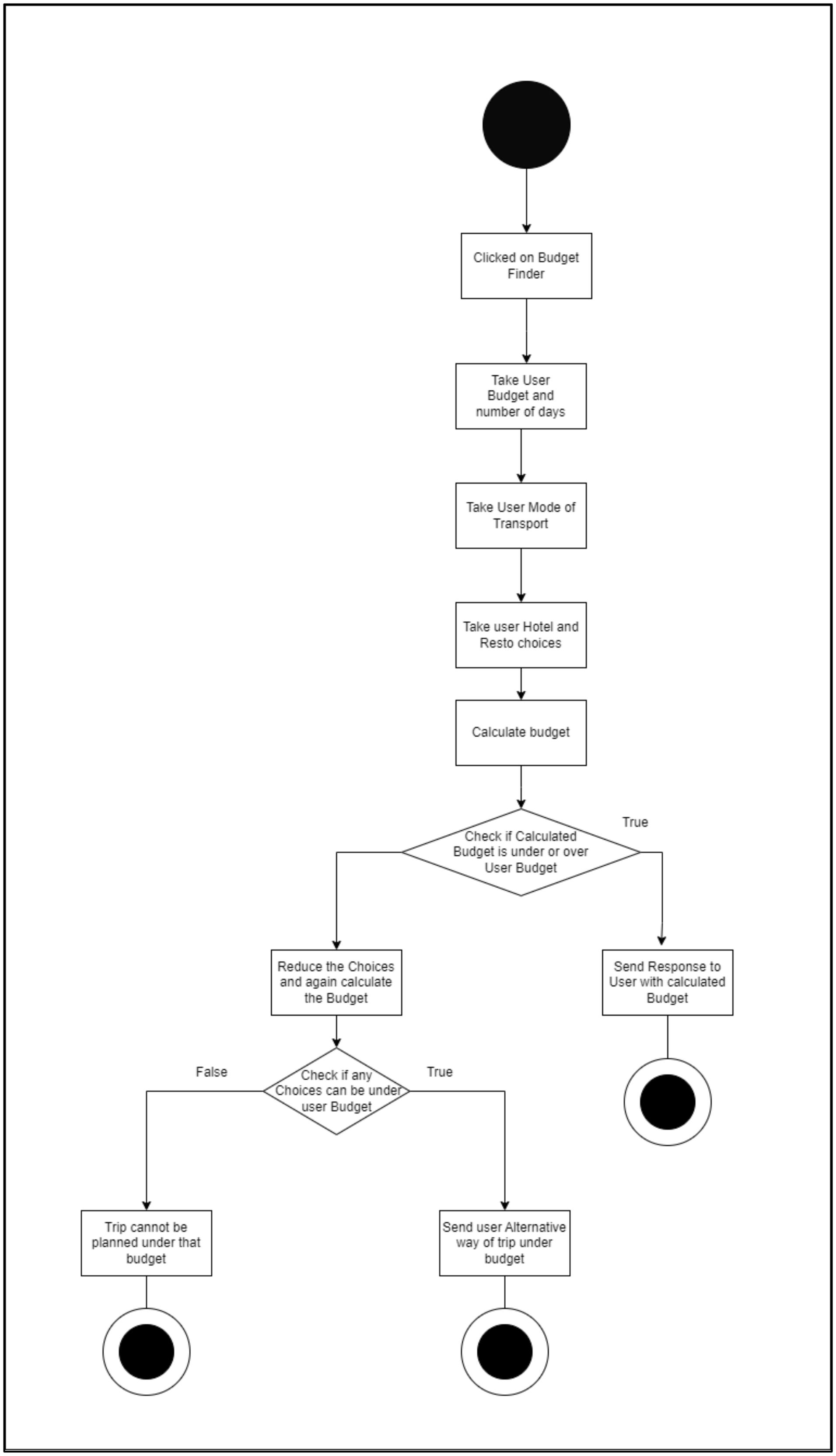

Clicked on Budget Finder
Take User Budget and number of days
Take User Mode of Transport
Take user Hotel and Resto choices
Calculate budget
Check if Calculated Budget is under or over User Budget
True
Reduce the Choices and again calculate the Budget
Send Response to User with calculated Budget
False
Check if any Choices can be under user Budget
True
Trip cannot be planned under that budget
Send user Alternative way of trip under budget

Fig 2.14 Activity Diagram for Budgets Finder

The activity diagram for the Budget Finder feature outlines a sequence of steps to help users determine whether their budget aligns with their travel preferences. It commences when the user clicks on the Budget Finder button, prompting them to input their budget and the number of days they plan to travel. The app then proceeds to calculate the budget, comparing it to the user's specified budget. If the calculated budget falls within the user's budget constraints, the app proceeds to provide a list of recommended places to visit, ensuring the plan aligns with the financial plan.

However, if the calculated budget exceeds the user's budget, the app offers a solution. It suggests changes to the user's preferences and then recalculates the budget based on these adjustments. If the user agrees to modify their preferences, the process loops back to the budget calculation step, offering a revised plan. In contrast, if the user opts not to change their preferences, the app kindly informs them that their initial budget isn't sufficient for their desired travel preferences.

Activity Diagram for Dashboard

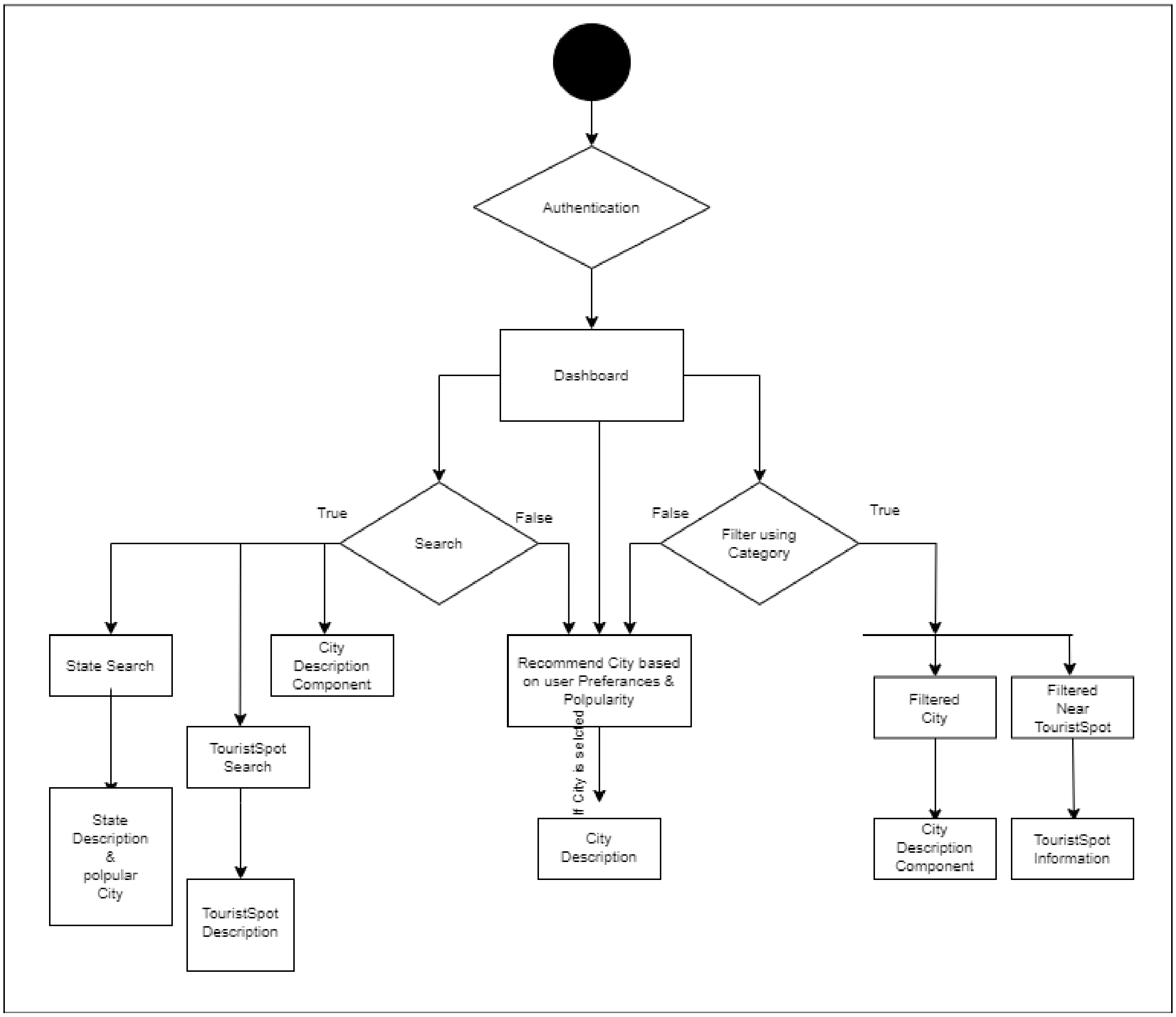


Fig 2.15 Activity Diagram for Dashboard

The process starts with the user authenticating themselves by entering their email and password. If the authentication is successful, the user is taken to the dashboard. From the dashboard, the user can select to search for a city. Once the user selects or search for a city, they can enter the name of the city or state and state description. If the user enters the name of a state, the system will display a list of popular cities in that state. The user can then select the city they want to search for. Once the user selects a city, the system will display a description of the city, as well as a list of popular tourist spots in the city. The user can then select a tourist spot to learn more about it. If the user wants to filter their search results, they can do so by selecting a category for the city or tourist spot. For example, they can filter the results to show only cities with beaches, or tourist spots that are family-friendly. The system will then display a list of filtered cities or tourist spots. The user can then select the city or tourist spot they want to learn more about. The activity diagram also shows a recommendation feature. The system can recommend cities or tourist spots to the user based on their preferences and the popularity of the city or tourist spot.

Activity Diagram for Community Post

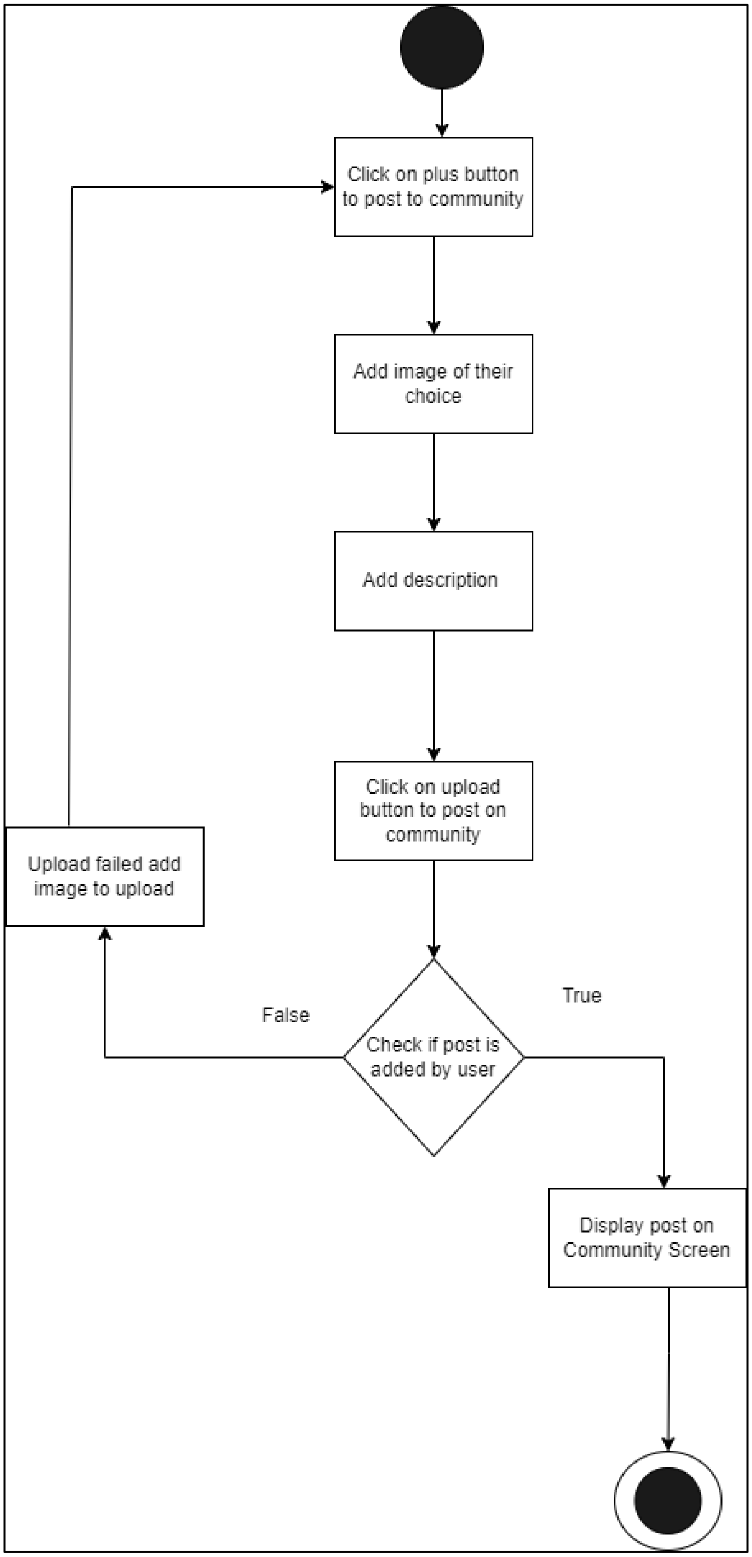
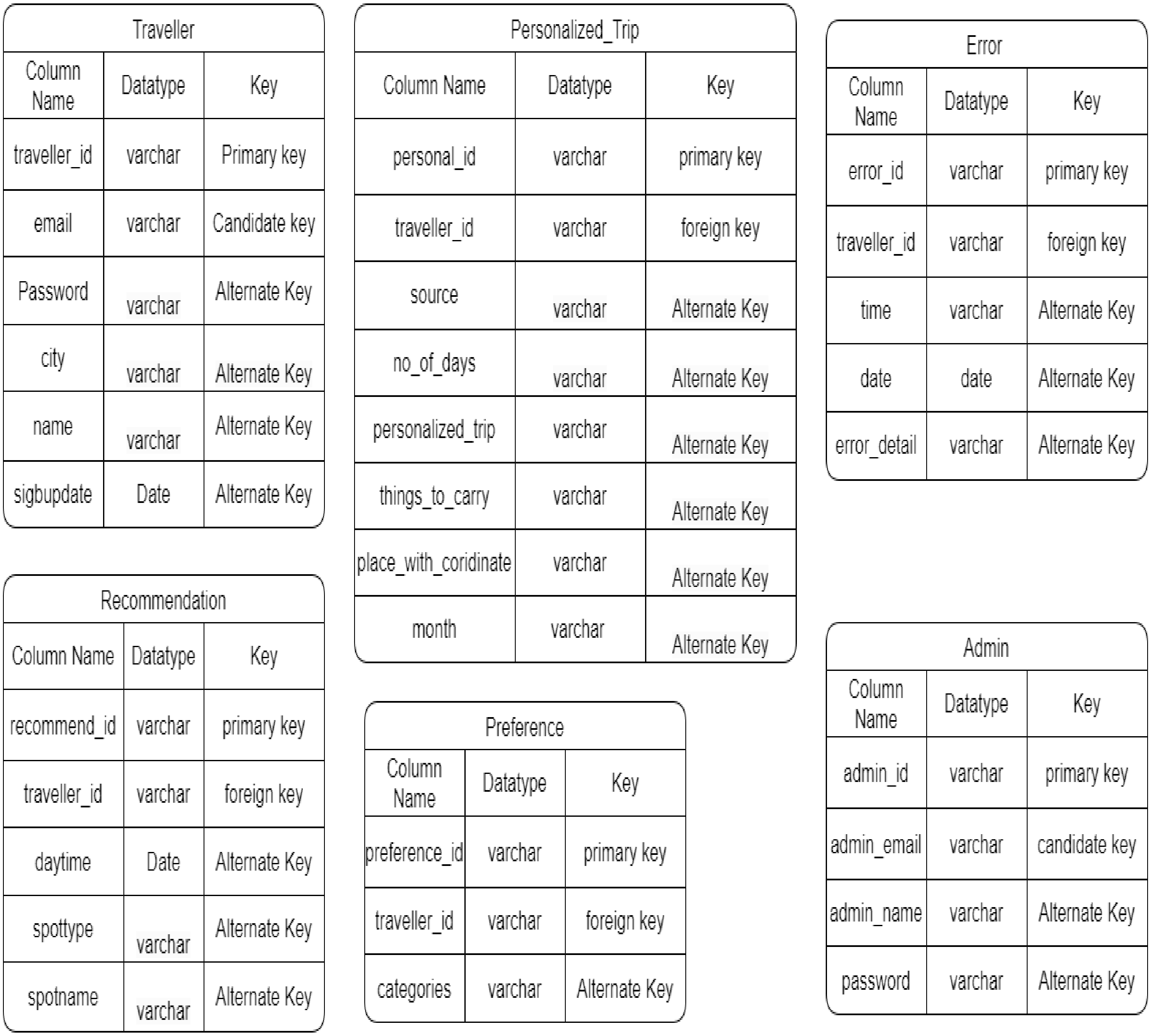

Click on plus button to post to community
Add image of their choice
Add description
Click on upload button to post on community
Upload failed add image to upload
False
True
Check if post is added by user
Display post on Community Screen

Fig 2.16 Activity Diagram for Community Post

The above activity diagram displays the flow of actions taken by user while uploading their post on the Community page. The users can upload pictures throughout their journey and share their live experiences with everyone through Community page. First, they select add button to upload post then they can select the image they want to post and can also add description they want to attach with the respective post. After adding image and description they can click on upload button to finally post on Community Screen to display their post live which other logged in users in our application can see. And if they click on upload button without selecting the image the upload will fail and they have to first select the image to upload either nothing will be uploaded.

x. Table Design



Traveller

Column Name	Datatype	Key
traveller_id	varchar	Primary key
email	varchar	Candidate key
Password	varchar	Alternate Key
city	varchar	Alternate Key
name	varchar	Alternate Key
sigbupdate	Date	Alternate Key

Personalized_Trip

Column Name	Datatype	Key
personal_id	varchar	primary key
traveller_id	varchar	foreign key
source	varchar	Alternate Key
no_of_days	varchar	Alternate Key
personalized_trip	varchar	Alternate Key
things_to_carry	varchar	Alternate Key
place_with_coridinate	varchar	Alternate Key
month	varchar	Alternate Key

Error

Column Name	Datatype	Key
error_id	varchar	primary key
traveller_id	varchar	foreign key
time	varchar	Alternate Key
date	date	Alternate Key
error_detail	varchar	Alternate Key

Recommendation

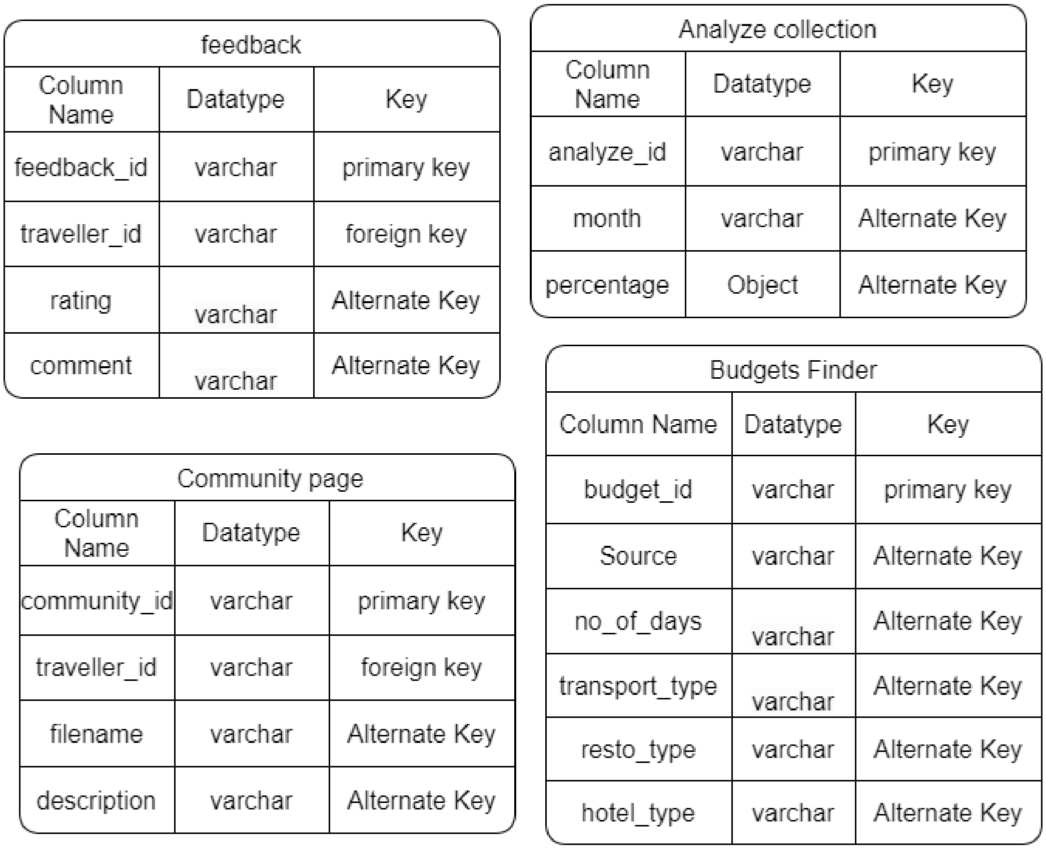
Column Name	Datatype	Key
recommend_id	varchar	primary key
traveller_id	varchar	foreign key
daytime	Date	Alternate Key
spottype	varchar	Alternate Key
spotname	varchar	Alternate Key

Preference

Column Name	Datatype	Key
preference_id	varchar	primary key
traveller_id	varchar	foreign key
categories	varchar	Alternate Key

Admin

Column Name	Datatype	Key
admin_id	varchar	primary key
admin_email	varchar	candidate key
admin_name	varchar	Alternate Key
password	varchar	Alternate Key



feedback

Column Name	Datatype	Key
feedback_id	varchar	primary key
traveller_id	varchar	foreign key
rating	varchar	Alternate Key
comment	varchar	Alternate Key

Community page

Column Name	Datatype	Key
community_id	varchar	primary key
traveller_id	varchar	foreign key
filename	varchar	Alternate Key
description	varchar	Alternate Key

Analyze collection

Column Name	Datatype	Key
analyze_id	varchar	primary key
month	varchar	Alternate Key
percentage	Object	Alternate Key

Budgets Finder

Column Name	Datatype	Key
budget_id	varchar	primary key
Source	varchar	Alternate Key
no_of_days	varchar	Alternate Key
transport_type	varchar	Alternate Key
resto_type	varchar	Alternate Key
hotel_type	varchar	Alternate Key

Fig 2.17 Table Design

xi. Deployment Diagram

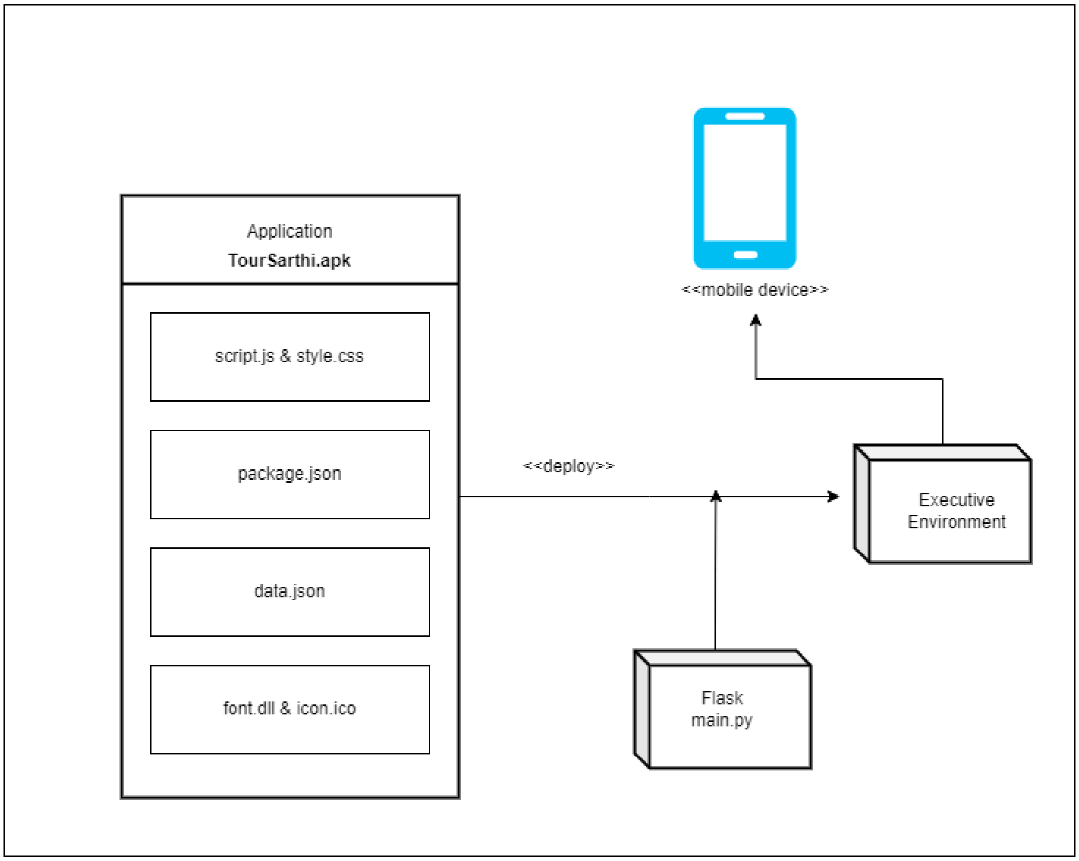


Fig 2.18 Deployment Diagram

A deployment diagram for a React Native and Flask application illustrates the physical deployment of the software components and their interaction with hardware and network infrastructure.

Nodes: Nodes represent physical or virtual devices where various parts of the application are deployed. In this case, we have nodes for the mobile devices running the React Native app and nodes for the server infrastructure where the Flask backend is hosted.

Components: Components represent the software modules or units that make up the system. In this context, you'd have two main components:

React Native App Component: This component represents the mobile application running on users devices.

Flask Backend Component: This component represents the backend server built with Flask, which provides APIs and handles business logic as well fetch the response front-end which is react native and with the help of routing corresponding function will be executed

Package. Json file contain all the dependency which is used and require to execute the application

xii. Architecture Diagram

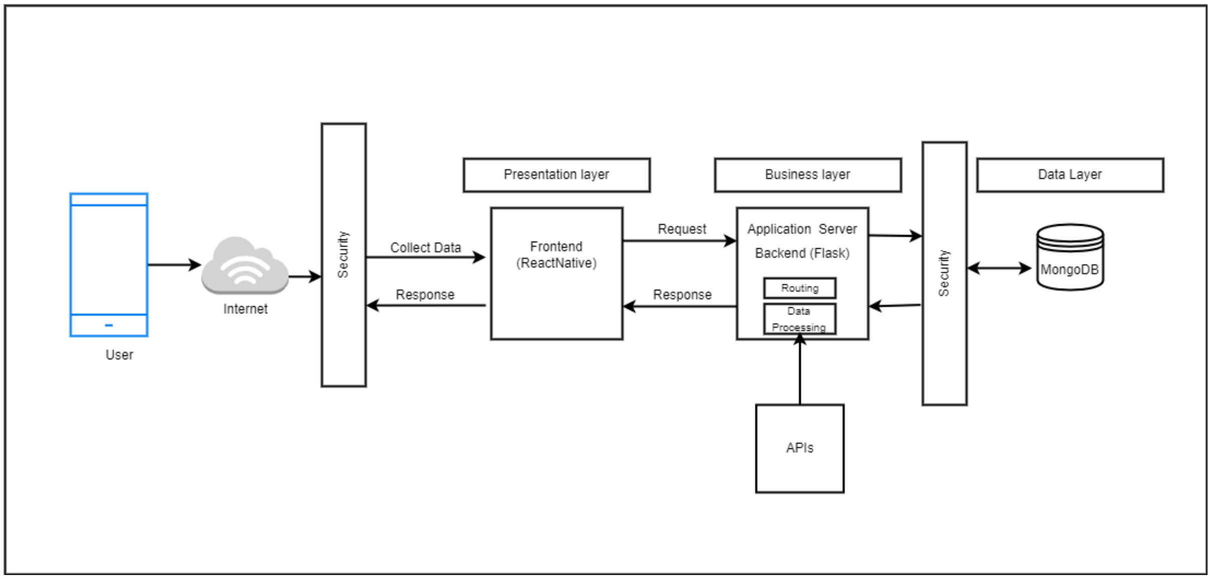


Fig 2.19 Architecture Diagram

The architecture diagram of the Intelligent Mobile Travelling Guide app shows a three-tier architecture:

Presentation Layer: The presentation layer is responsible for displaying the user interface and interacting with the user. It is implemented using React Native, a framework for building mobile apps using JavaScript.

Business Layer: The business layer contains the core logic of the app, such as generating personalized trip plans, budget plans, and providing information about places. It is implemented using Fiasco, a Python framework for building web applications.

Data Layer: The data layer stores and retrieves data from the database. It is implemented using MongoDB, a NoSQL database.

The three layers communicate with each other using APIs. The presentation layer calls the business layer APIs to get personalized trip plans, budget plans, and information about places. The business layer calls the data layer APIs to store and retrieve data from the database.

The architecture diagram also shows the security layer, which is responsible for authenticating users and protecting the app from unauthorized access. The security layer is implemented using a combination of authentication and authorization mechanisms.

3. System planning

i. Gantt Chart

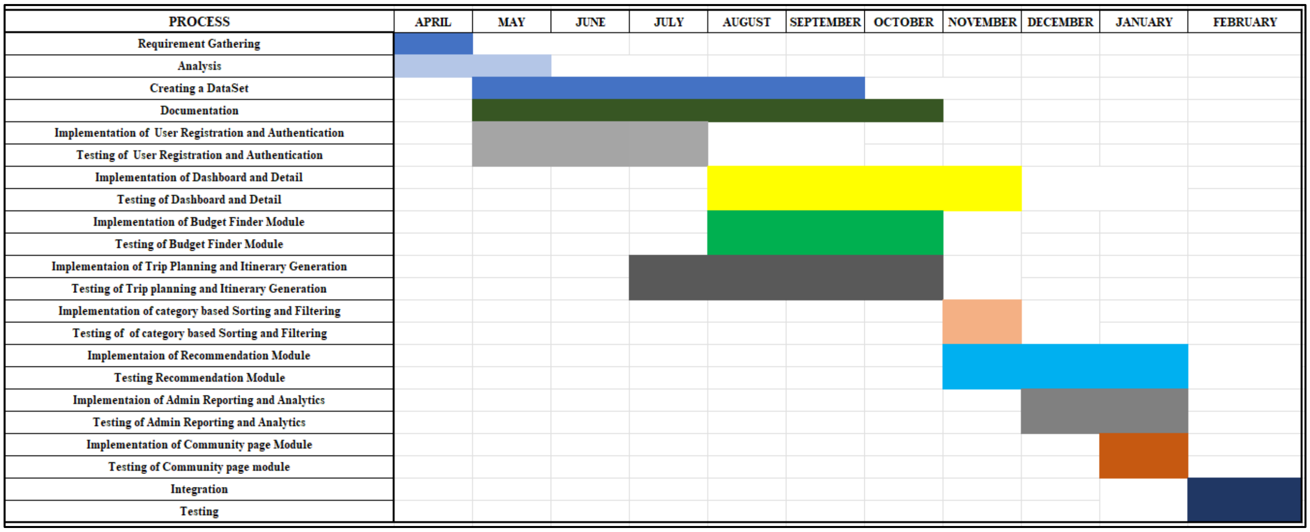


Fig 2.20 Gantt Chart

The Gantt chart estimates that the project will be completed in 11 months, which is approximately 44 weeks. Initially, we will begin with project requirement gathering. After one week, we will simultaneously commence the analysis phase. Following this, we will allocate 3 weeks to the creation of a dataset containing details of tourist spots in Indian cities. In parallel, we will develop the project's design and documentation.

Subsequently, we will initiate the implementation of 3 modules while concurrently conducting testing. Once these modules are completed, we will proceed to implement and test the remaining 3 modules. Afterward, we will begin implementing 2 more modules and conduct testing for each. Finally, we will integrate all the modules and perform comprehensive testing.

ii. Activity diagram

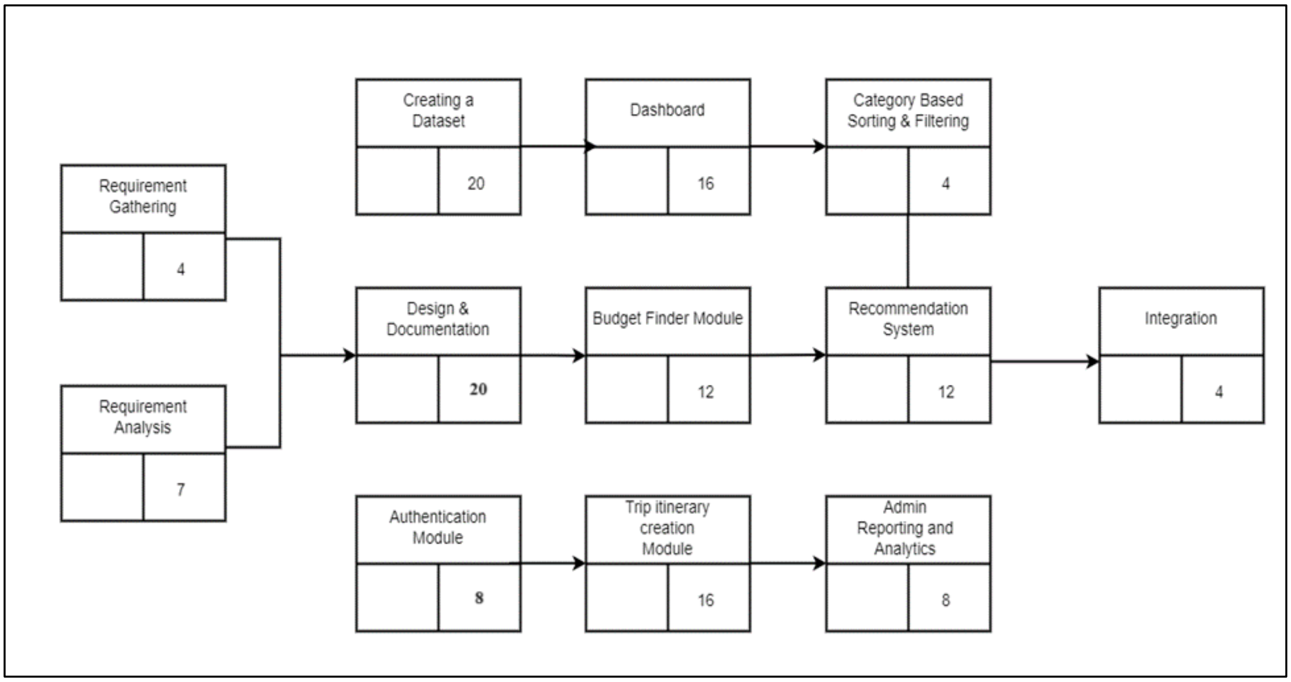
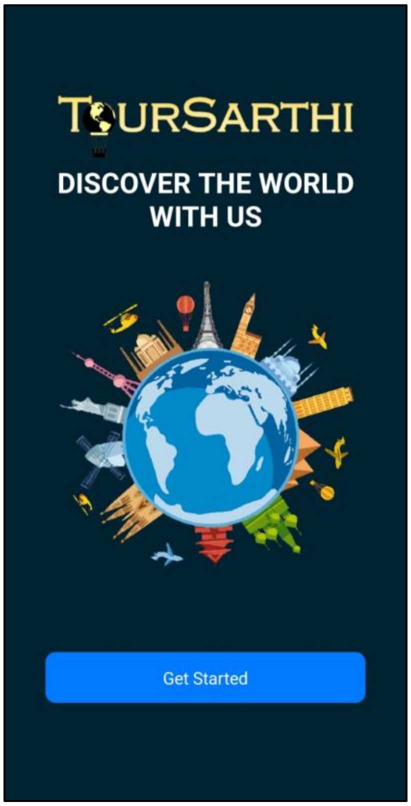


Fig 2.21 Activity diagram

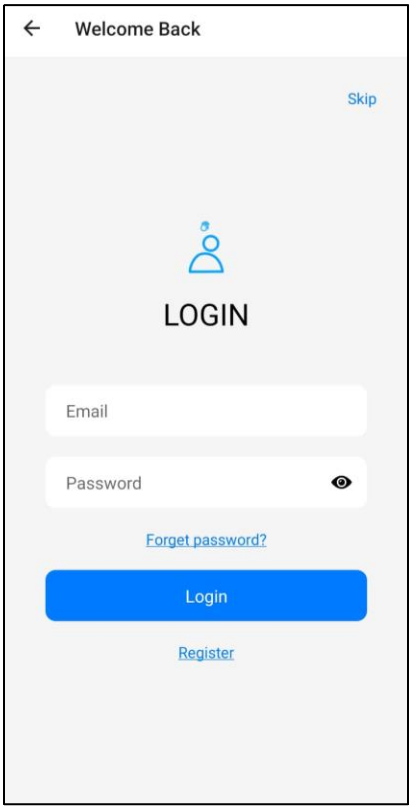
A PERT chart, which stands for Program Evaluation and Review Technique, is a visual project management tool used to plan and track project tasks and their timelines. It differs from a Gantt chart, which is a bar chart that schedules tasks, by its structure and how it estimates task durations. There are 3 processes running in parallel, and in the end, we will be integrating all the modules. If the creation of a dataset, dashboard, and category-based sorting and filtering gets delayed for 4 weeks, it will not affect the overall submission date. Similarly, the Authentication module, trip itinerary creation module, and admin reporting and analytics get delayed for 10 weeks, it will not affect the overall submission date

4. Implementation / User Interface



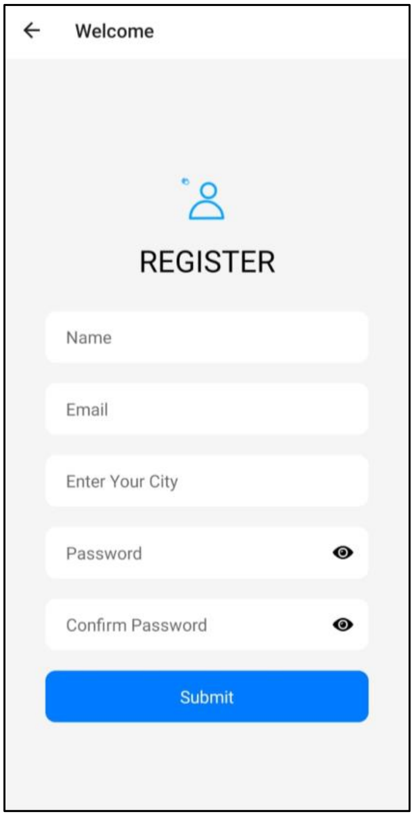
Onboarding Page

The initial screen that greets users upon downloading the intelligent tour guide app for first time will be this captivating Onboarding screen. Its purpose is to introduce users to the app's name, "TourSarthi" and there will be button which indicate user to get started with the app on their journey of exploration and interaction



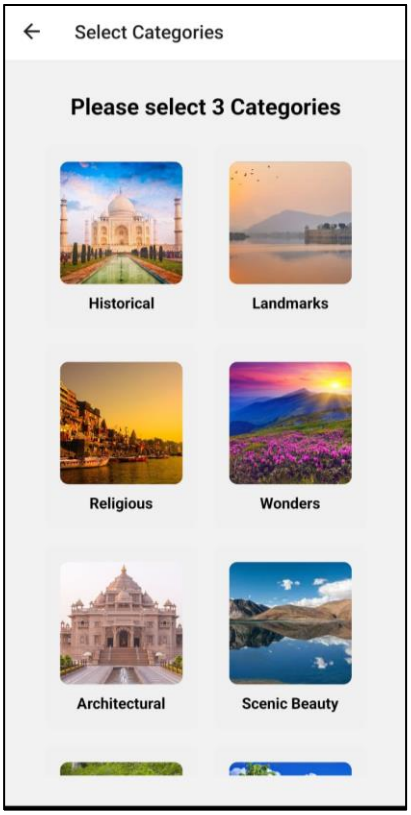
Login Page

After the Onboarding page, users will be directed to the Registration and Login page. This page serves two main functions: registration for new users and login for existing users. For new users who have not registered before, they can create their account by clicking on the "Register" button. This action will navigate them to the Register page where they can provide the necessary information to create their account. On the other hand, if users are already registered and returning to the app, they can log in by entering their email and password in the designated fields. Upon entering the correct login credentials, they will be directed to their dashboard, which provides access to various features and functionalities of the app.



Register Page

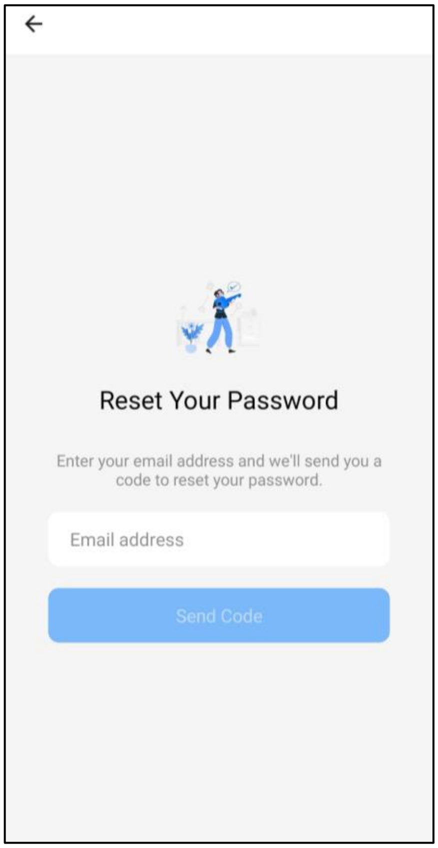
During the initial registration process, users will be required to provide their username, email address, city of residence, and a password to create their unique user ID within the application. Upon tapping the submit button, the user ID will be successfully created and stored in the database. Subsequently, when users wish to log in to the application, they can simply enter their registered email address and password to gain access to their personalized dashboard.



Please select 3 Categories

Preference Page

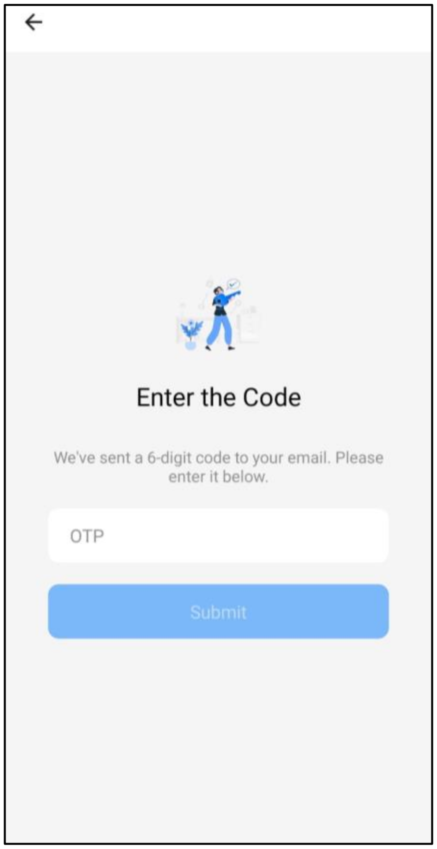
After the successful registration process, the user will be prompted to select three categories of their preference from a range of options, including Historical, Religious, Wonders, Scenic Beauty, and more. These selected categories will serve as a basis for filtering and displaying relevant trip options and recommendations to the user based on their preferences



Reset Your Password

Enter your email address and we'll send you a code to reset your password.

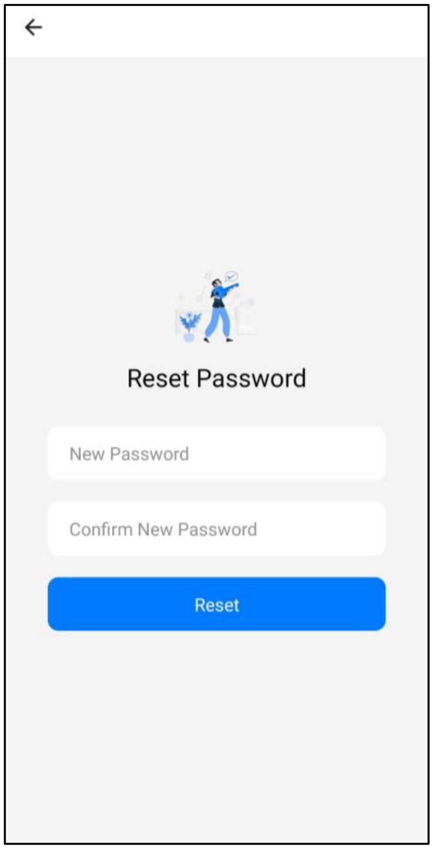
Email address



Enter the Code

We've sent a 6-digit code to your email. Please enter it below.

OTP



Reset Password

New Password

Confirm New Password

Forget Password Page

The Forgot Password functionality allows existing users who have forgotten their login password to reset it. By clicking on the Forgot Password option, the user can initiate the password reset process. A verification code will be sent to their registered email address. Upon entering the correct verification code, the user can set a new password and regain access to their existing account, eliminating the need to create a new ID.

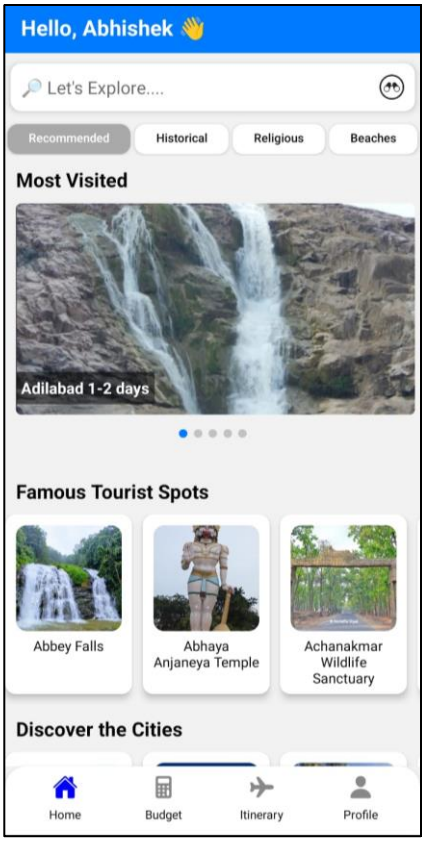
Verification Code

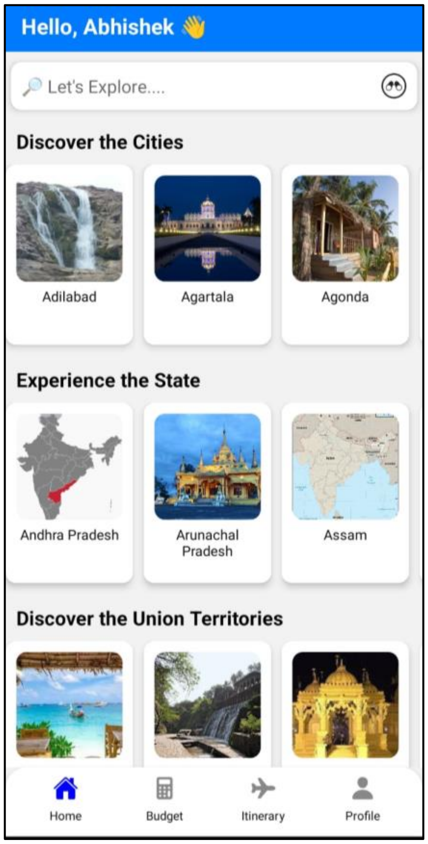
When the user clicks on the Send Code button, a verification code will be generated and sent to their registered email address. In the backend, the system will verify the user's identity based on their email and generate the code accordingly. The user will receive a notification indicating that the code has been sent to their email. They can then check their email, retrieve the code, and enter it into the app to proceed with the password reset process.

Reset Password

After entering the verification code, the app will perform a validation check to ensure that the entered code matches the one previously sent to the user's email. If the code is correct, the user will be redirected to the reset password page. On this page, the user can enter a new password and confirm it. Once the password is successfully reset, the user will be able to log in to the app using their updated credentials without any further difficulties.

Dashboard





The UI design of the Intelligent Mobile Travelling Guide app is simple, easy to use, and visually appealing. The app uses a blue and green colour scheme, which is both calming and inviting. The app also uses a lot of white space, which makes the interface look clean and organized.

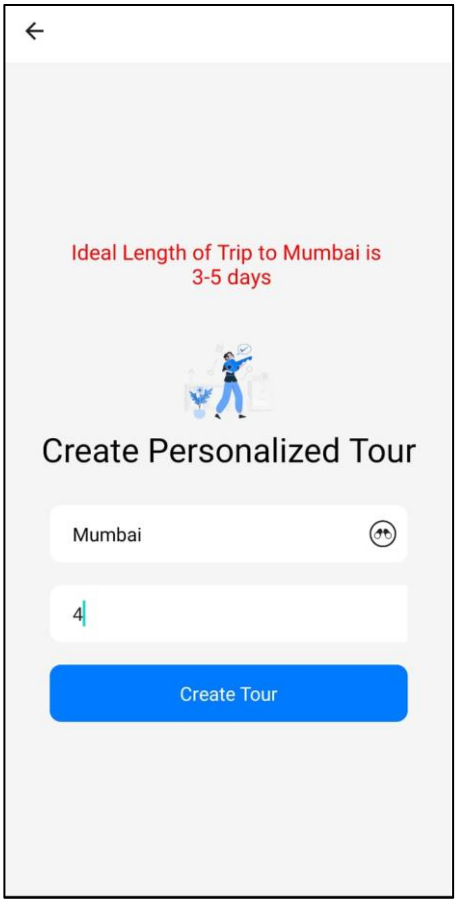
Search bar: A search bar where users can search for places to visit.

Recommended: A section that displays recommended places to visit based on the user's interests and preferences.

Most Visited: A section that displays the most visited places in the city.

City Highlights: A section that displays highlights of the city, such as its temples, palaces, and gardens.

Personalised Trip Creation Pages

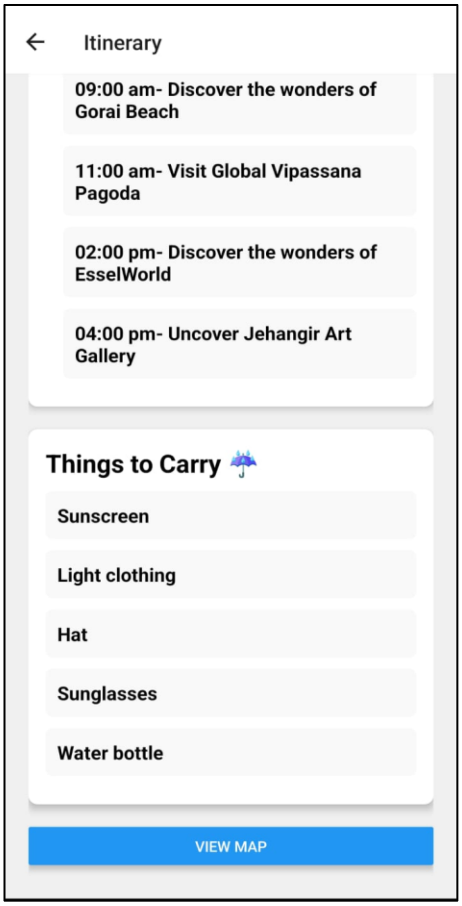


Ideal Length of Trip to Mumbai is 3-5 days

Create Personalized Tour

Mumbai

4



Itinerary

09:00 am- Discover the wonders of Gorai Beach

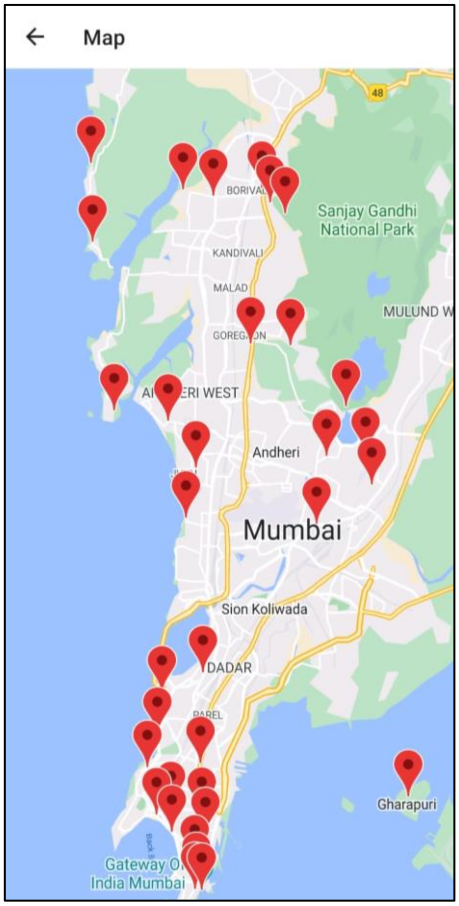
11:00 am- Visit Global Vipassana Pagoda

02:00 pm- Discover the wonders of EsselWorld

04:00 pm- Uncover Jehangir Art Gallery

Things to Carry ☔

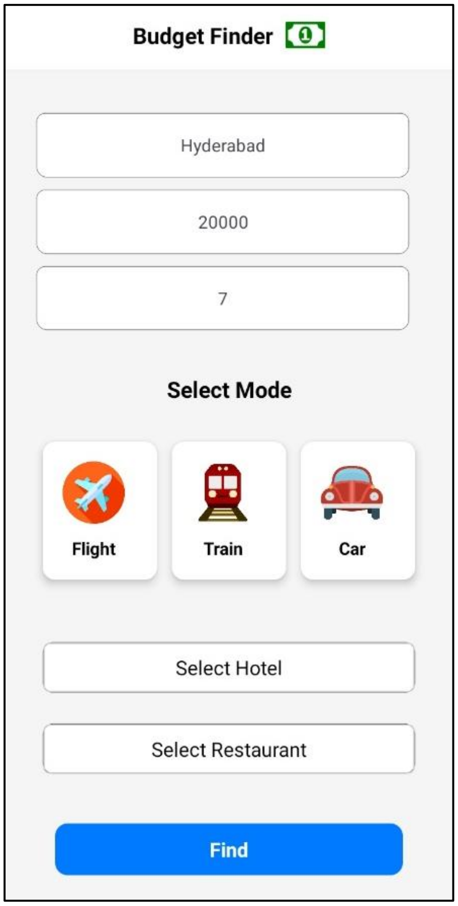
- Sunscreen
- Light clothing
- Hat
- Sunglasses
- Water bottle



Map

The initial page features two input boxes prompting users to input their destination and the number of days for their trip. Upon clicking the "Create Tour" button, the application generates a personalized trip plan tailored to the user's preferences. Additionally, it offers suggestions on items to bring based on the destination's temperature. Below this, a map section is displayed, allowing users to navigate to the map page. Here, the tour destination is highlighted, enhancing the user's understanding of their trip's geographic context. This seamless integration of input, planning, and visualization enhances the user experience, making trip planning both efficient and enjoyable.

Budget Finder



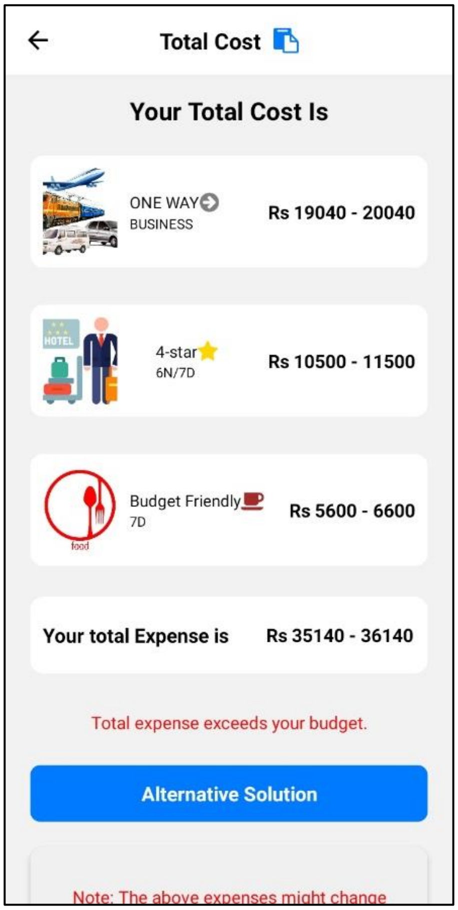
Budget Finder

Hyderabad

20000

7

Select Mode



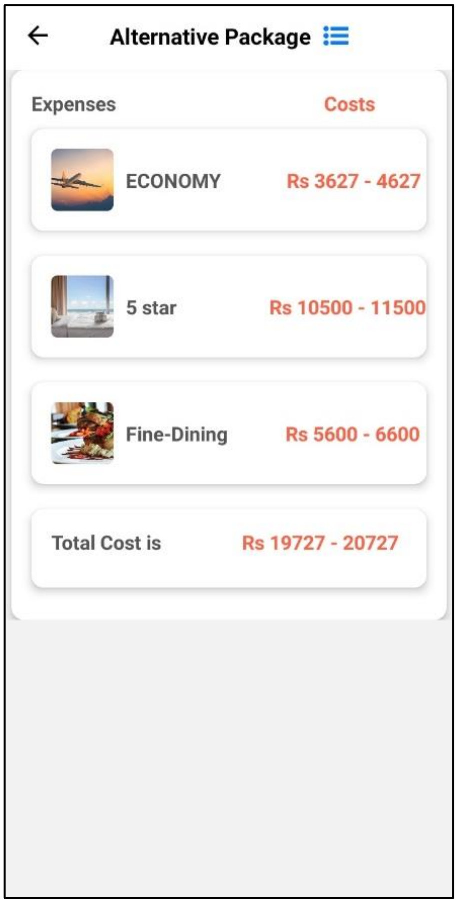
Total Cost

Your Total Cost Is

ONE WAY BUSINESS	Rs 19040 - 20040
4-star 6N/7D	Rs 10500 - 11500
Budget Friendly 7D	Rs 5600 - 6600
Your total Expense is	Rs 35140 - 36140

Total expense exceeds your budget.

Note: The above expenses might change

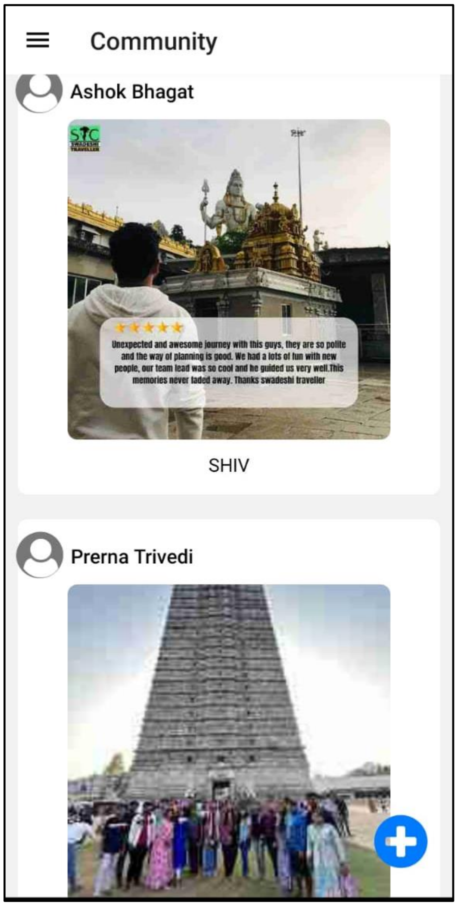


Alternative Package

Expenses	Costs
ECONOMY	Rs 3627 - 4627
5 star	Rs 10500 - 11500
Fine-Dining	Rs 5600 - 6600
Total Cost is	Rs 19727 - 20727

In this module, User will select the destination, enter a budget and number of days they want to go for the selected destination. User can select mode from three types i.e., flights, train and car. User can select the flight class, train class and can select hotel considering their rating that is three-star, four-star, five star and can select restaurant from budget-friendly, fine-dining and then can click process button to proceed further that if the total cost is under budget or not. Then budget-finder will provide alternative solution if the selected option is not under the budget of the user.

Community Post


Community

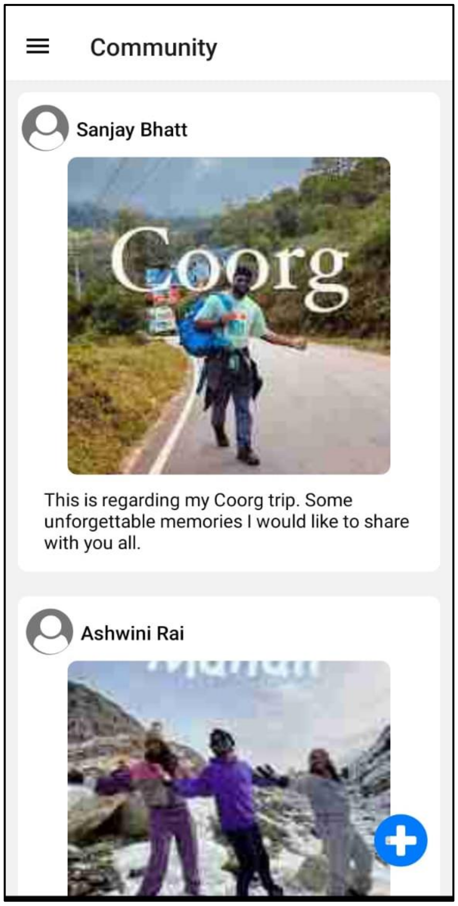
Ashok Bhagat

★★★★★

Unexpected and awesome journey with this guys, they are so polite and the way of planning is good. We had a lots of fun with new people, our team lead was so cool and he guided us very well.This memories never faded away. Thanks swadeshi traveller

SHIV

Prerna Trivedi


Community

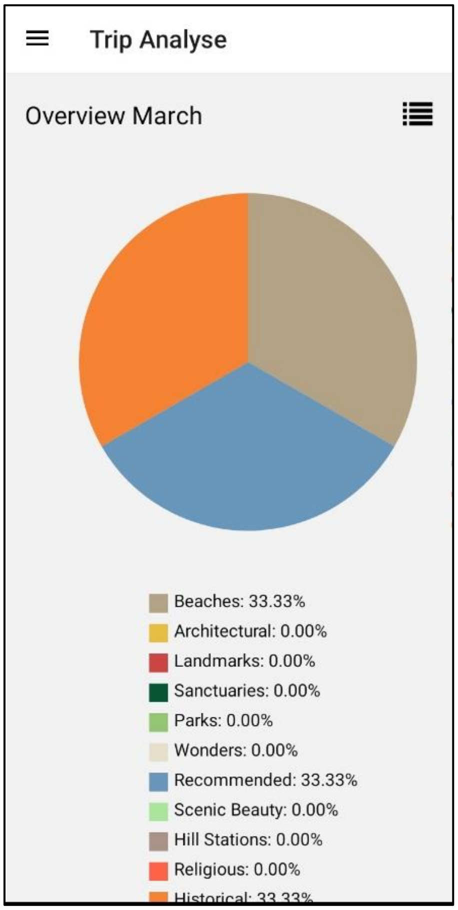
Sanjay Bhatt

This is regarding my Coorg trip. Some unforgettable memories I would like to share with you all.

Ashwini Rai

In this module, User can upload the image of their trips with description for that image, the uploaded image and description will be displayed on Community Post with all the users in our application. User can see other users trip posts and their experiences through Community Page.

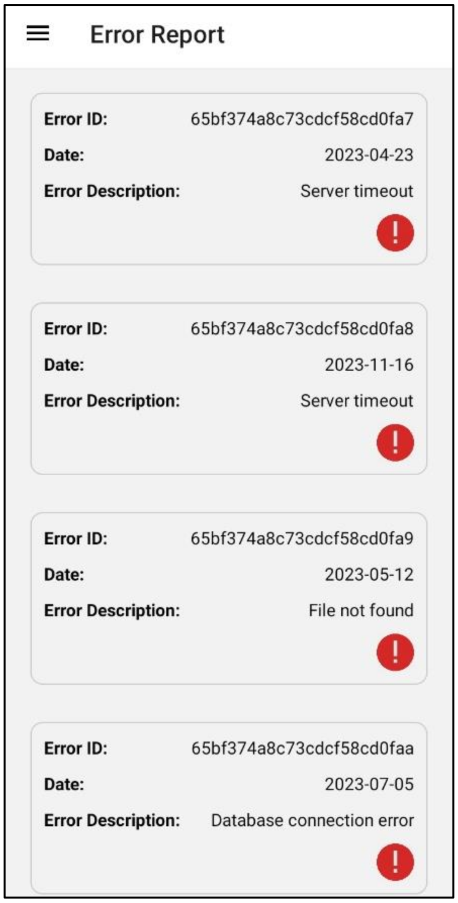
Admin Dashboard



Trip Analyse

Overview March

■ Beaches: 33.33%
■ Architectural: 0.00%
■ Landmarks: 0.00%
■ Sanctuaries: 0.00%
■ Parks: 0.00%
■ Wonders: 0.00%
■ Recommended: 33.33%
■ Scenic Beauty: 0.00%
■ Hill Stations: 0.00%
■ Religious: 0.00%
■ Historical: 33.33%



Error Report

Error ID: 65bf374a8c73cdcf58cd0fa7
Date: 2023-04-23
Error Description: Server timeout

Error ID: 65bf374a8c73cdcf58cd0fa8
Date: 2023-11-16
Error Description: Server timeout

Error ID: 65bf374a8c73cdcf58cd0fa9
Date: 2023-05-12
Error Description: File not found

Error ID: 65bf374a8c73cdcf58cd0faa
Date: 2023-07-05
Error Description: Database connection error

In this module, Admin will be displayed with the user most preferred created itinerary. It will display the percentage of which category is booked more and what is the percentage of that category. Based on this it visualizes people are preferring which category more and the best category trips are displayed on User Dashboard on Bestseller.

The error report is displayed to admin so that they can made the improvement contacting to developers and error and recurring issues in our system are easy to identify.

4. Testing

Software testing is the process of evaluating and verifying that a software application or system meets its specified requirements and works correctly. It involves systematically examining and assessing the various components, functionalities, and behaviors of the software to identify defects, errors, or bugs.

The main objectives of software testing include:

- Validation: Ensuring that the software meets the intended requirements and functions as expected by the end-users.
- Verification: Confirming that the software is implemented correctly according to its design and specifications.
- Error Detection: Identifying defects, bugs, or inconsistencies in the software that may lead to incorrect behavior, performance issues, or security vulnerabilities.
- Quality Assurance: Ensuring the quality, reliability, and usability of the software to meet the needs and expectations of stakeholders.

Software testing is typically performed by skilled professionals known as testers or quality assurance (QA) engineers. They employ various testing techniques, methodologies, and tools to assess different aspects of the software, including functionality, performance, security, usability, and compatibility across different platforms and environments.

The testing methodologies used for Intelligent mobile Travelling Guide includes ‘Unit Testing’ for individual units of code.

Unit Testing (White Box)

Unit Testing refers to a method by which individual units of source code are tested to determine if they are fit for use. A unit is the smallest testable part of an application. The goal of unit testing is to isolate each part of the program and show that the individual parts are correct. For Intelligent Tour Guide, this implies testing textboxes with both valid and invalid input and determining whether the outcome is satisfactory.

1. Sign In

I/P Parameter	Value	I/P Type	Result of Test
Username	"_$"	Invalid (Non-Alphanumeric Characters)	Successful
	"Mira"	Valid	Successful
	"So…"	Invalid	Unsuccessful
Email Address	"abc@gmail.com"	Valid	Successful
	"123"	Invalid	Successful
	"mari.com"	Invalid	Successful
	"a123@gmail.com"	Valid	Successful
City	"Mumbai"	Valid	Successful
	"Jaipur"	Valid	Successful

2. Feedback

I/P Parameter	Value	I/P Type	Result of Test
Email	"abc@gmail.com"	Valid	Successful
	"abcgmail.com"	Invalid	Successful
	"ab@gcm"	Invalid	Successful
Comment	"no Comments"	Valid	Successful
	"Poor"	Valid	Successful
	"Excellent Services"	Valid	Successful
Ratings	"1 Rating"	Valid	Successful
	"2 Rating"	Valid	Successful
	"3 Rating"	Valid	Successful
	"4 Rating"	Valid	Successful
	"5 Rating"	Valid	Successful

3. Budget Finder

I/P Parameter	Value	I/P Type	Result of Test
Source	"Mumbai"	valid	Successful
	"123"	Invalid	Successful
	"Juhu Beach"	Invalid	Successful
Destination	"Raipur"	valid	Successful
	"123"	Invalid	Successful
	"Kohima"	Valid	Successful
No of Days	10	Valid	Successful

Test Case ID: 1

Test Scenario: Finding Estimated Budget for a Destination.

Test Steps:

1. Navigate to the Budget Finder module in the application.
2. Enter the destination "Destination Name" in the input field.
3. Specify your budget.
4. Specify the “No of Days”.
5. Select Any one from Transportation mode, select Hotel type, select Dining type.
6. Click on the "Process" button.
7. Total Budget displayed addition of selected preferences.

Prerequisites:

- Ensure the user is logged in.

Test Data:

- Destination: Varanasi
- Budget Limit: 15000

Expected Results:

- The application should accept the user's choice of Varanasi as the destination.
- The input field should allow the entry of the budget limit as 15000.
- After submitting, the system should display the estimated budget for the trip to Varanasi.
- If the estimated budget exceeds 15000, the system should provide alternative solutions that fit within the budget.

Actual Results:

- The application accepted Destination Name, User budget.
- The application found the total cost for the entered destination and adding selected preferences.

Test Status:

- Pass

Parameter	**Value**	**Result**
Destination Name (City)	Hyderabad	Pass
	Virar	Fail
	Varanasi	Pass
	Marine Lines	Fail
	Ghatkopar	Fail
	Udaipur	Pass

Parameter	**Value**	**Result**
No of days	10	Pass
	100	No is too large
	“Fff”	Invalid

Parameter	**Value**	**Result**
Budget	30,000	Pass
	Thirty Thousand	Fail

Test Case ID: 2

Test Scenario: Providing Alternative Solutions within User specified Budget.

Test Steps:

1. Navigate to the Budget Finder module in the application.
2. Enter the departure city and the destination city.
3. Specify the budget.
4. Choose the transport type, hotel type, restaurant type.
5. Click on the "Process" button.
6. Display the total calculated cost.
7. If the total cost exceeds the specified budget, Application should display alternative solution button. Otherwise, the test is failed.

Prerequisites:

- Ensure the user is logged in.

Test Data:

- Departure City: Mumbai
- Destination City: Hyderabad
- Transport Type: Flight (Business Class)
- Hotel Type: 4 Star
- Restaurant Type: Fine dining
- Budget Limit: ₹20,000

Expected Results:

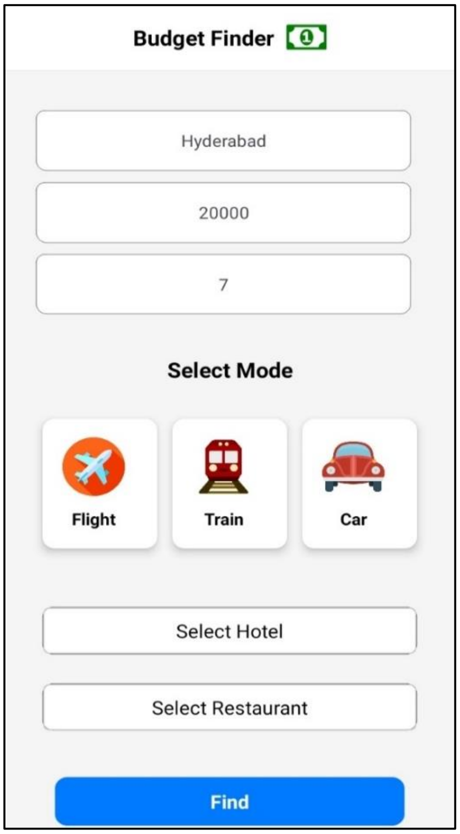
- If the total cost exceeds the specified budget limit, the application should provide alternative solutions to fit within the budget.
- The alternative options provided by the application should have a total cost that does not exceed the specified budget of ₹20,000.

Actual Results:

- Total Cost exceeds the limit 20,000 to 34,000.
- Application displayed the alternative solution button.
- Alternative solution provided new options that fits under specified limit budget that is 20,000.

Test Status:

- Pass.



Hyderabad

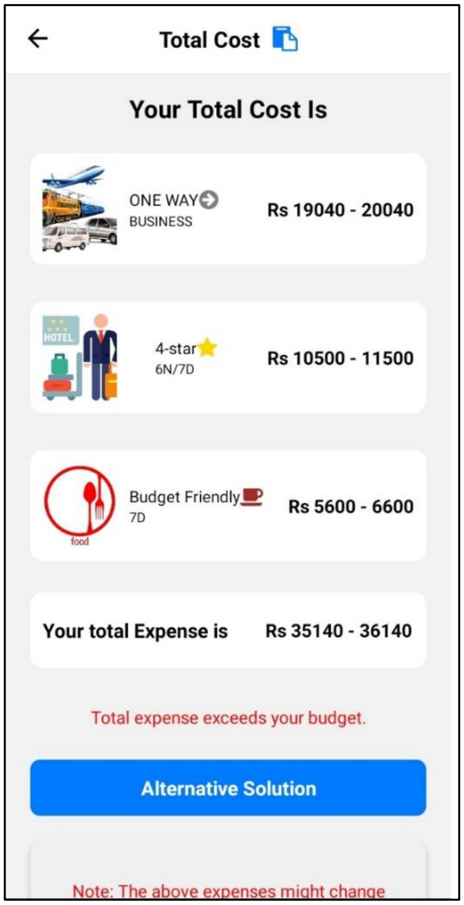
20000

7

This is the manual testing done on the above data are as follows:

Test Input:

1. Destination: Hyderabad
2. Budget: 20,000
3. No of days: 7
4. Mode: Flight
5. Class: Business Class
6. Hotel: 4 Star
7. Restaurant: Fine Dining



Your Total Cost Is

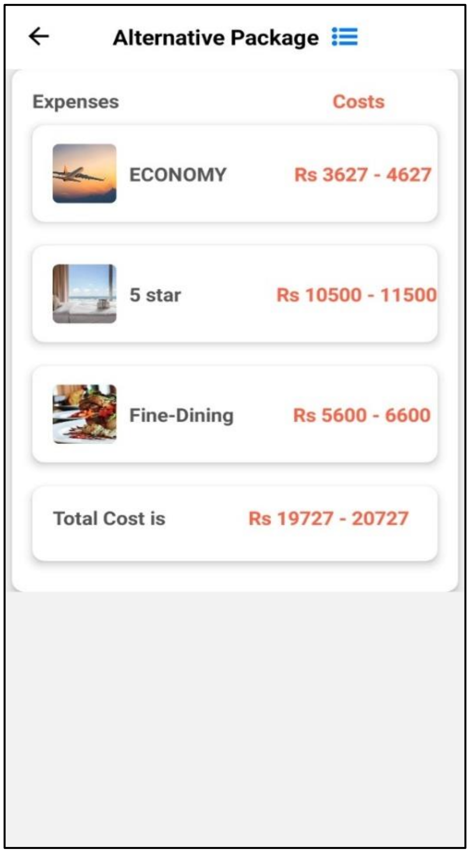
ONE WAY BUSINESS	Rs 19040 - 20040
4-star 6N/7D	Rs 10500 - 11500
Budget Friendly 7D	Rs 5600 - 6600
Your total Expense is	Rs 35140 - 36140

Total expense exceeds your budget.

Note: The above expenses might change

After clicking on Process, Total Cost is displayed finding the cost of each.

1. Total cost exceeded budget.
2. Alternative Solution Button got displayed.
3. Clicked on Alternative solution.



Alternative Package

Expenses	Costs
ECONOMY	Rs 3627 - 4627
5 star	Rs 10500 - 11500
Fine-Dining	Rs 5600 - 6600
Total Cost is	Rs 19727 - 20727

Alternative solution is provided by application.

1. Alternative solution provided by application that fits under user set budget.
2. Alternative package is less than or equal to the user budget that is 20,000.
3. Test Case Successful.

Test Case ID: 3

Test Scenario: Comparing Flight Prices with an External Application to check accuracy.

Test Steps:

1. Navigate to the Budget Finder module in the application.
2. Enter the departure city as "Mumbai" and the destination city as "Hyderabad."
3. Specify the budget limit as ₹20,000.
4. Choose the flight type as "Business Class."
5. Record the total cost of the flight options displayed in the application.
6. Open an external web browser (e.g., Google Chrome, Mozilla Firefox).
7. Visit a flight booking website.
8. Search for flights from Mumbai to Hyderabad with similar criteria (business class, similar travel dates).
9. Note down the total cost of the flights listed on the external website.
10. Compare the total cost obtained from the application with the prices from the external website to ensure accuracy.

Prerequisites:

- Ensure the user is logged in.

Test Data:

- Departure City: Mumbai
- Destination City: Hyderabad
- Budget Limit: ₹20,000
- Mode: Flight
- Class: Business Class

Expected Results:

- The total cost of flights displayed within the application should closely match the prices obtained from the external flight booking website.

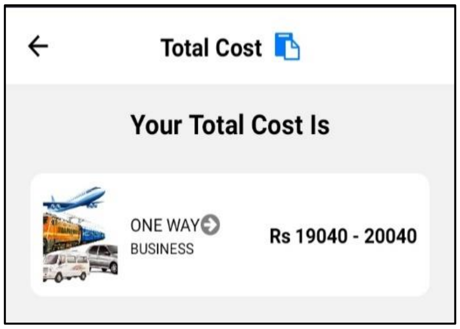
Actual Results:

- The prices matched closely with the external website.
- Price of application is close to actual price.

Test Status:

- Pass

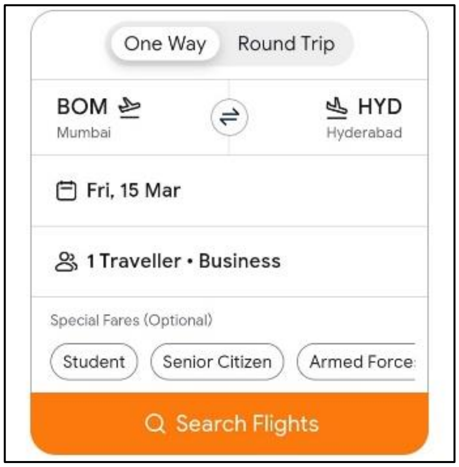
The price of Business Class in our Application Vs in External Application.


Total Cost

Your Total Cost Is

ONE WAY
BUSINESS

Rs 19040 - 20040


One Way | Round Trip

BOM Mumbai → HYD Hyderabad

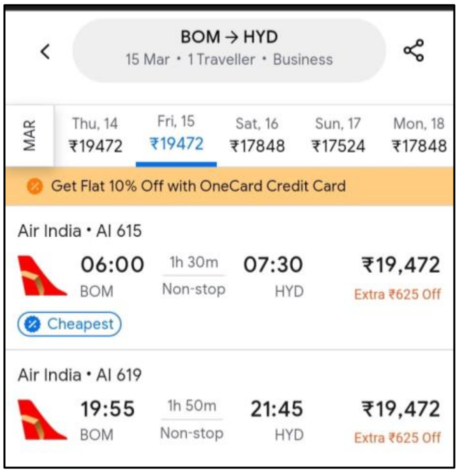
Fri, 15 Mar

1 Traveller • Business

Special Fares (Optional)

Student | Senior Citizen | Armed Force

Search Flights


BOM → HYD
15 Mar • 1 Traveller • Business

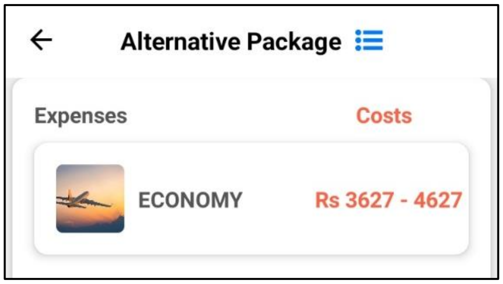
MAR	Thu, 14	Fri, 15	Sat, 16	Sun, 17	Mon, 18
	₹19472	₹19472	₹17848	₹17524	₹17848

Get Flat 10% Off with OneCard Credit Card

Air India • AI 615
06:00 BOM | 1h 30m Non-stop | 07:30 HYD | ₹19,472 Extra ₹625 Off
Cheapest

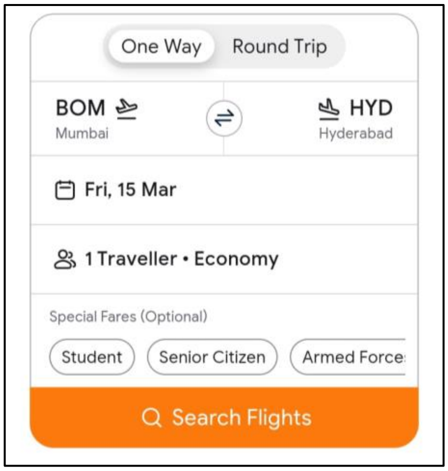
Air India • AI 619
19:55 BOM | 1h 50m Non-stop | 21:45 HYD | ₹19,472 Extra ₹625 Off

Price of Economy Class in our Application Vs External Application.


Alternative Package

Expenses | Costs

ECONOMY | Rs 3627 - 4627


One Way | Round Trip

BOM Mumbai → HYD Hyderabad

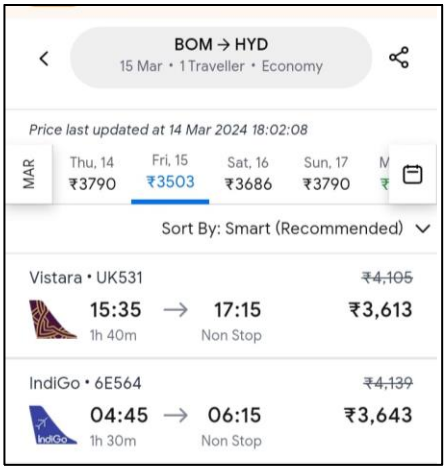
Fri, 15 Mar

1 Traveller • Economy

Special Fares (Optional)

Student | Senior Citizen | Armed Force

Search Flights


BOM → HYD
15 Mar • 1 Traveller • Economy

Price last updated at 14 Mar 2024 18:02:08

MAR	Thu, 14	Fri, 15	Sat, 16	Sun, 17	M
	₹3790	₹3503	₹3686	₹3790	₹

Sort By: Smart (Recommended)

Vistara • UK531
15:35 → 17:15 | 1h 40m Non Stop | ~~₹4,105~~ ₹3,613

IndiGo • 6E564
04:45 → 06:15 | 1h 30m Non Stop | ~~₹4,139~~ ₹3,643

Test Case for Login

Test Case ID: Test Case 1

Test Scenario: User Login Process

Test Steps:

1. User navigates to the login page.
2. User enters their registered email address in the "Email" field.
3. User enters their password in the "Password" field.
4. User clicks on the login button.

Prerequisites: User must have a registered email address and password.

Physical Device: Android, Emulator.

Test Data: Valid registered email address and corresponding password.

Expected/Intended Results:

- Enter the correct email and password combination.
- Successfully log in.
- Redirect to the dashboard.

Actual Results: As Expected,

Test Status: Pass.

Test Case for Registration

Test Case ID: Test Case 2

Test Scenario: To register user details in the database.

Test Steps:

1. User must fill the following details such name, email, city, password and confirm password,
2. User enter valid email in which it may initiates with letters or alpha-numeric character but must ends with "@gmail.com".
3. User must enter same password and confirm password. And Password must be more than 6 digits.
4. After filling all the detail at register screen user must click on submit.

Prerequisites: The user must have valid Email id.

Physical Device: Android, Emulator.

Test Data: Legitimate username, Email id and Password.

Expected/Intended Results:

- User filled all required details.
- System redirects to the OTP verification page upon completion.

Actual Results: As Expected,

Test Status: Pass.

Test Case for OTP-Verification

Test Case ID: Test Case 3

Test Scenario: To verify OTP during user registration.

Test Steps:

1. User receives OTP on the registered email address.
2. User enters the received OTP in the designated field.
3. User clicks on the verify button.

Prerequisites: User must have received the OTP on the registered email address.

Test Data: Valid OTP received on the registered email address.

Expected/Intended Results:

- Enter the correct OTP.
- Click the "Verify" button.
- Upon successful verification, proceed to the next step of registration.

Actual Results: As Expected,

Test Status: Pass

Test Case for Admin Dashboard

Test Case ID: Test Case 4

Test Scenario: To analyze user reports for trips created by users and determine the distribution among different categories.

Test Steps:

1. The admin selects the User Analysis tab.
2. The admin chooses the month to generate an overview report.

Prerequisites:

- Admin must specify the target month for analysis.
- They need to adjust strategies based on the chosen month.
- Analyze data and make decisions accordingly for the specified month.

Physical Device: Android, Emulator.

Test Data: Overview analysis of user-personalized trips.

Expected/Intended Results:

- Admin selects the desired month for overview.
- System generates a pie chart.
- Pie chart displays trip distribution among various categories.
- The representation aims for clarity and visual appeal to facilitate admin understanding.

Actual Results: As Expected,

Test Status: Pass.

Test Case for Personalized Itinerary

Test Case ID: Test Case 5

Test Scenario: Creating a Personalized Itinerary

Test Steps:

1. Navigate to the application's homepage.
2. Click on the "Itinerary" or "Plan Trip" section to start creating a personalized itinerary.
3. From the dropdown list of cities, choose "Varanasi."
4. In the input box for the number of days, enter "5."
5. Click on the "Submit" or "Create Itinerary" button to proceed.

Prerequisites:

- Ensure the user is logged in.
- Verify that the application is in a stable state with no known issues.

Test Data:

- Destination: Varanasi
- Number of Days: 5

Expected/Intended Results:

- The application should accept the user's choice of Varanasi as the destination.
- The input box should allow the entry of "5" for the number of days.
- After submitting, the system should generate a personalized itinerary for Varanasi for 5 days without any errors.

Actual Results: As Expected,

Test Status: Pass

Test Case for Forgot Password

Test Case ID: **Test Case 6**

Test Scenario: Forgot Password Process

Test Steps:

1. Forgot Password
   - User initiates the forgot password process by clicking on the "Forgot Password" link.
   - User enters their registered email address.
   - User clicks on the submit button.
2. OTP Verification
   - User receives OTP on the registered email address.
   - User enters the received OTP in the designated field.
   - User clicks on the verify button.
3. Reset Password
   - Upon successful OTP verification, user enters a new password in the "New Password" field.
   - User confirms the new password by re-entering it in the "Confirm Password" field.
   - User clicks on the submit button.

Prerequisites:

- User must possess a registered email address.
- User must have access to the registered email inbox.

Test Data:

- User provides a valid registered email address.
- User receives a valid OTP on the registered email address.
- User sets a new valid password.

Expected/Intended Results:

- User successfully verifies OTP.
- User resets password.
- User can log in using the new password.

Actual Results: As Expected,

Test Status: Pass.

5. Cost Estimation and Analysis

COCOMO (Constructive Cost Model) is a regression model based on LOC, i.e., number of Lines of Code. It is a procedural cost estimate model for software projects and is often used as a process of reliably predicting the various parameters associated with making a project such as size, effort, cost, time, and quality. It was proposed by Barry Boehm in 1981 and is based on the study of 63 projects, which makes it one of the best-documented models.

The key parameters which define the quality of any software products, which are also an outcome of the COCOMO are primarily Effort & Schedule:

- Effort**:** Amount of labour that will be required to complete a task. It is measured in person-months units.
- Schedule**:** Simply means the amount of time required for the completion of the job, which is, of course, proportional to the effort put in. It is measured in the units of time such as weeks, and months.

Different models of COCOMO have been proposed to predict the cost estimation at different levels, based on the amount of accuracy and correctness required. All of these models can be applied to a variety of projects, whose characteristics determine the value of the constant to be used in subsequent calculations.

These factors are incorporated into the Basic COCOMO model to provide an initial estimate of the effort required to complete a software project.

In COCOMO, projects are categorized into three types;

1. Organic**:** A development project can be treated of the organic type, if the project deals with developing a well-understood application program, the size of the development team is reasonably small, and the team members are experienced in developing similar methods of projects.
2. Semidetached: A development project can be treated with semidetached type if the development consists of a mixture of experienced and inexperienced staff. Team members may have finite experience in related systems but may be unfamiliar with some aspects of the order being developed
3. Embedded: A development project is treated to be of an embedded type, if the software being developed is strongly coupled to complex hardware, or if the stringent regulations on the operational method exist.

This project lies under Semidetached. The value of constant of ai , bi , ci and di are different for the different project. So, The value for these constants is as follows:

Mode	Ai	bi	ci	di
Organic	2.5	1.12	2.5	0.35

Step1-COCOMO Basic Model

Formula for Calculation:

E=ai*(KLOC)^bi

Here:

E: Effort applied in terms of person-months.

KLOC: Kilo lines of code of the project.

ai and bi are constant but they are different for various class software project.

Tdev = ci*(E)^di

Tdev: Estimated time to develop the software expressed in Months.

According to the project, KLOC is different for different project. Here is Estimated KLOC for this project:

1. Authentication:3KLOC, 2.3KSLOC
2. User dashboard: 3KLOC, 2KSLOC
3. Admin dashboard:2KLOC, 1.2KSLOC
4. Detailed screen: 2KLOC, 1.2KSLOC
5. Budgets finder: 3KLOC, 2.3KSLOC
6. Feedback of trip: 1 KLOC, 0.8KSLOC
7. Profile of user: 1 KLOC, 0.7KSLOC
8. Trip Itinerary: 4KLOC, 3KSLOC
9. Recommendation: 1KLOC, 0.8KSLOC
10. Community Post:1KLOC, 0.7KSLOC

Total estimated KLOC: 21KLOC,15KSLOC

Calculation:

Effort = ai * (KLOC) ^ bi

= 2.5 * (15) ^ 1.12

= 62.2793 Person-month

Tdev = ci * (KLOC) ^ di

= 2.5 * (62.2793) ^ 0.35

= 15.6160 month

Average Staff Size = E/Tdev

= 62.2793 /15.6160

= 3.9882 person

In above calculation using the Constructive Cost Model (COCOMO), the estimated effort for a software project of 15 KSLOC (Kilo Source Lines of Code) is 62.2793 person-months, with a development time of approximately 15.6160 months. Based on these figures, an average staff size of around 4 persons is recommended for the project.

6. Future work

Future work for Intelligent mobile travelling guide will be:

- **Geographical Coverage Expansion:**

  Increasing the dataset to include global destinations means acquiring and organizing information about various places of interest, accommodations, restaurants, transportation options, and activities worldwide.

  This expansion requires extensive research, data collection, and verification to ensure accuracy and relevance.

  It involves incorporating diverse cultural, historical, and geographical information to provide users with comprehensive insights into destinations worldwide.

  The expansion also entails developing algorithms or systems to efficiently manage and present information from a wide range of geographical locations.

- **Social Media Integration:**

  Integrating the platform with social media networks enhances user engagement and facilitates marketing.

  Enabling users to share their travel experiences on platforms like Facebook, Twitter, and Instagram can app's reach and attract new users.

  Social media integration may involve implementing features such as sharing photos, writing reviews, tagging locations, and inviting friends to join the platform.

  Social media analytics can provide valuable insights into user behaviour, preferences, and trends, for further app development and marketing strategies.

- **Language Localization**:

  Extensive language localization involves translating the app's interface, content, and user communication into multiple languages to provide to a diverse global audience.

  It requires careful consideration of cultural nuances, idiomatic expressions, and linguistic conventions to ensure accurate and culturally appropriate communication.

  Language localization enhances accessibility and user experience for non-English speaking users, potentially expanding the app's user base and market penetration.

  Implementing language preferences and localization settings within the app allows users to customize their language experience, further personalizing their interaction with the platform.

- **Booking and Payment Integration:**

  Integrating the platform with travel booking and payment systems streamlines the travel planning and booking process for users.

  Partnering with established booking platforms allows users to search, compare, and book flights, accommodations, tours, and activities directly within the app.

  Secure payment integration ensures a seamless and trustworthy transaction process, enhancing user confidence and satisfaction.

  Implementing features such as real-time availability, price alerts, and flexible payment options enhances the user experience and encourages booking conversions.

- **Subscription Model:**

  Introducing a subscription model provides a sustainable revenue stream for the app's continued development and maintenance.

  Subscription plans may offer additional benefits such as access to premium content, exclusive deals, priority customer support, or ad-free browsing.

  Offering tiered subscription options allows users to choose a plan that best suits their needs and budget, maximizing revenue potential.

  Implementing free trial periods or discounted introductory offers can incentivize users to subscribe and experience the full benefits of the app.

  Regularly updating and expanding subscription offerings keeps the app competitive and encourages long-term user retention.

7. Conclusion

In conclusion, the Intelligent Mobile Travelling Guide is an innovative and comprehensive solution for the travel industry. The project aims to provide travellers with a seamless and personalized travel experience through a mobile application. By offering users the ability to plan their own tours based on their preferences, the application provides a unique and customized experience that is not found in traditional travel packages.

The project is designed with a user-friendly interface that is easy to navigate, allowing users to easily access the features of the application.

The use of an incremental model for the project life cycle ensures that the development process is flexible and allows for changes to be made throughout the project. This approach ensures that the final product is aligned with the needs of the users and meets their expectations.

Overall, the Intelligent Mobile Travelling Guide project is an exciting and promising endeavour that has the potential to revolutionize the travel industry. By providing a personalized and intuitive travel experience, the application can help travellers create memorable and unique experiences. The success of this project will depend on the commitment and collaboration of the project team.

8. Reference

Books:

- Sommerville, I. (2011). "Software Engineering." Addison-Wesley.
- Pressman, R. S. (2014). "Software Engineering: A Practitioner's Approach." McGraw-Hill Education.
- Ambler, S. W. (2004). "Introduction to UML 2 Class Diagrams." Agile Modelling.
- Myers, G. J., Sandler, C., & Badgett, T. (2011). "The Art of Software Testing." John Wiley & Sons.
- Kaner, C., Falk, J., & Nguyen, H. Q. (1999). "Testing Computer Software." Wiley.

Website:

- Flask Documentation
https://flask.palletsprojects.com/en/3.0.x/

- React Native Documentation
https://reactnative.dev/docs/getting-started

- PyMongo Documentation
https://pymongo.readthedocs.io/en/stable/

- Tourist Spot Information
https://tourism.gov.in/
https://www.incredibleindia.org/content/incredible-india-v2/en.html
https://www.india.gov.in/topics/travel-tourism
https://en.wikipedia.org/wiki/Tourism_in_India