SMART TRAVEL SUPER-APP FOR PERSONALIZED, ADAPTIVE AND COLLABORATIVE TRAVEL PLANNING

A MINIPROJECT REPORT FOR THE COURSE DESIGN THINKING AND INNOVATION

SUBMITTED BY

KAMALESH P.	**(230701137)**
MONIC AUDITYA A.	**(230701194)**
MONISH D.Y.	**(230701195)**

IN PARTIAL FULFILMENT FOR THE AWARD OF THE DEGREE OF

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

MAY 2025

RAJALAKSHMI ENGINEERING COLLEGE (AUTONOMOUS)

RAJALAKSHMI NAGAR, THANDALAM – 602 105

BONAFIDE CERTIFICATE

Certified that this project titled **“Smart Travel Super-App for Personalized, Adaptive and Collaborative Travel Planning”** is the Bonafide work of **Kamalesh P. (230701137)**, **Monic Auditya A. (230701194), Monish D.Y. (230701195),** who carried out the project work under our supervision during the year **2024–2025**.

Student Signature with name

Kamalesh P. -

Monic Auditya A. -

Monish D.Y. -

Signature of the Supervisor

Signature Examiner-1

Signature Examiner-2

ANNEXURE III

TABLE OF CONTENTS

ABSTRACT

Traveling around the world can be a fun experience, but planning a trip can be tedious - selecting destinations, arranging transportation, finding food, and booking hotels are scattered across different apps. Most travel applications fail to provide adaptive recommendations in real-time, as they do not integrate with live data from sources such as Google Places or Foursquare. This forces customers to switch between several apps, leading to fragmented planning and outdated recommendations.

ExploreEase is a travel super-app that utilizes artificial intelligence to integrate all the features of trip planning on one platform. It offers individual and group-based recommendations by combining structured data (place attributes) with unstructured data (user reviews). Unsupervised learning groups users according to their preferences, neural matrix factorization (NMF) maps user-item interactions, and reinforcement learning supports adaptive planning.

The core functionality includes dynamic trip packages themed to the season, three route suggestions optimized for cost savings, time savings, and scenic views, and live travel and accommodation bookings. Group functionality includes trip conversation through chat, polls to make shared decisions such as cancellations, shared expense accounting, and real-time support for emergencies. User reviews are stored in MongoDB to support scalability.

In general, ExploreEase transforms challenging travel planning into a seamless and smart experience. challenging travel planning into a seamless and smart experience.

LIST OF ABBREVIATIONS

Abbreviation	Full Form
AI	Artificial Intelligence
NMF	Neural Matrix Factorization
RL	Reinforcement Learning
URS	Unified Recommendation Score
API	Application Programming Interface
UI/UX	User Interface / User Experience
PCA	Principal Component Analysis
ML	Machine Learning
JSON	JavaScript Object Notation
EXIF	Exchangeable Image File Format
S3	Amazon Simple Storage Service
REST	Representational State Transfer
OTAs	Online Travel Agencies
CF	Collaborative Filtering
RMSE	Root Mean Squared Error
MAE	Mean Absolute Error
R^2	Coefficient of Determination (Statistical Measure)
EC2	Elastic Compute Cloud (AWS)

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 – INTRODUCTION

Traveling can be personally rewarding, however, it usually involves a tremendous amount of stressful planning. This can include coordinating information from others, booking flights, booking hotels, managing budgets and costs, and the overall planning can feel messy and overwhelming. Each of the introductory travel spaces we used and evaluated offered pieces of the puzzle, but none provided a clear, complete process for either groups or individual travelers.

As part of our mini-project for the Design Thinking and Innovation class, we developed the idea of ExploreEase, a super app that simplifies travel planning. In its basic form, the concept was simple: we felt that it would be useful to develop one simple, easy-to-use platform that allowed users to plan trips, receive tailored recommendations and itineraries, make reservations, keep a budget, foster a sense of fellowship, and allow for openness and flexibility. ExploreEase will eliminate the hassle from trip planning, making it an enjoyable part of the adventure.

1.1 DESIGN THINKING APPROACH

Understanding the User and Problem Definition

In the ExploreEase project, design thinking was a key factor in understanding more deeply the issues facing travelers, particularly when organizing travel as a group. We interviewed and surveyed solo and group travelers to understand their needs and barriers. We identified four pain points that travelers faced when planning group trips: Conflicting opinions of group members, managing combined budgets, managing multiple bookings separately managing cancellations.

Based on these examples, we defined the problem as:

How do we design a travel app that provides support for planning, considers group members' preferences, routes are flexible, and can provide transparency on cost control?

This then informed our design solution to include a travel app with AI recommendations, route optimization, flexible cancellation terms, integrated communication, and tracking expenses.

Ideation, Prototyping, and Iterative Testing

During the ideation phase, we evaluated numerous ways to address the user, starting with basic filters and ultimately with selecting AI algorithms that learned people's preferences as a group. The prototype included AI-supported destination and accommodation recommendations, route suggestions that were customizable according to budget and time, group chat abilities, expense tracking, and polls to vote on cancellations.

Testing happened in a controlled environment, with users creating hypothetical trips. We received feedback that most people generally liked the recommendations and communications, but some thought we could improve on making it more customizable and user-friendly. Iteration led to increasing the AI's accuracy while using the tools and as well as expense tracking, which helped facilitate better performance of the platform for both individuals and groups.

1.2 STANFORD DESIGN THINKING MODEL AND ITS PHASES

The Stanford Design Thinking Model served as our guiding framework throughout five iterative stages.

Empathize

We conducted user interviews, surveys, and observations and attained rich insights about the pain points associated with planning travel, including challenges that are related to group decision-making and personalized recommendations.

Define

We have synthesized user insights to frame the key problem statement.

"In what ways can we design a simple, intelligent super-app that streamlines travel planning while providing an intelligent financial and communications application?"

We identified essential design goals such as AI-based group recommendations, flexible routing, clear and transparent cancellation policies, chat capability, and automatic expense tracking.

Ideate

We developed a wide range of ideas through brainstorming and some competitor analysis. We considered a number of options before choosing AI recommendation systems and cancellation mechanisms based on democratic polling.

Prototype

We have built a cross-platform app using UI technologies like React Native, Python frameworks for back end and AI capabilities, MongoDB for storage, AWS S3 for hosting, and web sockets for real-time communications. We developed features that include AI recommendations, group chat function, expense tracker, and cancellation voting polls.

Test

We have conducted usability testing with real users simulating trips in the app and collected feedback to improve AI accuracy, UI responsiveness, offline use, and data management. Through this circular process, we made several iterations to arrive at a viable and scalable platform for users who need solutions for travel planning.

CHAPTER 2 – LITERATURE REVIEW

Several research works were conducted to create a user-friendly travel app that meets all user needs. The major contributions were as follows:

A mobile app called '*Travigate'* was developed by **Parikh et al.** [1] that employs K-modes clustering along with CNN image recognition capabilities and provides interest-based and image-matching recommendations. Similar alternatives utilize unsupervised learning and personalized suggestions, but they lack group coordination and real-time data.

A group recommender system named '*GRec Tr',* developed by **Kim et al.** [2], utilizes collaborative filtering and constraint satisfaction to accommodate a variety of group member preferences, contributing to the investigation of group-focused recommendations and polls for consensus that integrate trip packaging.

Kong et al. [3] offered *RPMTD*, a multi-agent reinforcement learning system for over tourism that distributes tourists over routes with dual-congestion reward definitions. This is indicative of an adaptive, sustainability-responsible reinforcement learning approach for dynamic and equitable trip recommendations.

A tourist routing planning mechanism that utilizes a "comprehensive attractiveness index" was proposed by **Zhang et al.** [4] that takes into consideration cost, duration, and popularity, and is optimized with genetic algorithms. This work is similar in its aims of personalizing routing; however, it adds the ability for real-time flexibility with reinforcement learning.

Sharaff et al. [5] developed a hybrid LMF and popularity model to deal with personalization and cold-start issues, using standard neural matrix factorization

(NMF) and a fallback of popularity-based recommendation to improve user-item engagement.

Alenezi and Hirtle [6] proposed the paper on '*Normalized Attraction Travel Personality (NATP)'* representation, a topic model-based method that involves creating attraction embeddings based on crowd-sourced review semantics. Their model improves attraction modeling using implicit user travel personalities derived from review analysis and regularized normalization, achieving considerable advances in rating prediction and ranking quality.

Travel destinations are recommended by analyzing Instagram photos using the object detection capabilities of pixel-based clustering and SOMs. This was the idea given by **Stefanović and Ramanauskaitė** [7]**.** Other examples of organizing trip photos and subsequently inferring preferences from media content will use similar techniques.

Qin et al. [8] developed *DCSGR*, which divides large groups into subgroups based on interests and uses collaborative filtering to aggregate individual recommendations. Subgroup dynamics and group decision-making features are aided by unsupervised learning.

The integration of travel survey data and Amap API data was done by **Tang et al.** [9] to improve travel mode prediction using XGBoost and SHAP, and they worked with other sources of live feed data, including Google Places, to improve recommendation accuracy and responsiveness.

Jia-Xiang et al. [10] developed a multimodal travel route system to calculate trips using taxi, subway, and walking elements. This aligns with support for multiple mixed transport modes and group route customization preferences, such as trust of savings time, savings money, or scenic routes.

Chang et al. [11] improved hotel recommendation performance using heterogeneous social media data and leveraged signal information available to refine suggestions across social platforms.

Overall, these systems exemplify the higher-order sophistication developing in travel recommender systems. Yet, none accounted for an integrated mobile experience encompassing personalized, real-time, and group-based planning as a single challenge. The table below shows the major features that distinguishes our super app from other existing apps.

Features / Platforms	Explore Ease	Squad Trip	Travefy	Viavii	Road Trippers	Guide Geek
Personalized Trip Recommendations (ML-based)	☑	✕	✕	☑	✕	☑
Real-Time Group Collaboration (Chat + Polls)	☑	☑	☑	✕	✕	✕
Reinforcement Learning-Based Routing	☑	✕	✕	✕	✕	✕
Time/Cost/Scenic Route Optimization	☑	✕	✕	✕	☑	✕
Dynamic Trip Packages via Clustering	☑	✕	✕	☑	✕	✕
Hotel & Transport Booking Integration (APIs)	☑	☑	☑	☑	☑	☑

Trip Album Organizer (EXIF + clustering)	☑	✕	✕	✕	✕	✕
Expense Tracker (Individual & Group)	☑	☑	☑	✕	✕	✕
Cold-Start Handling (Hybrid NMF + Popularity)	☑	✕	✕	☑	✕	☑
Real-Time Traffic & Live Routing	☑	✕	✕	✕	☑	✕
AI-Powered Decisions	☑	✕	✕	☑	✕	☑
Platform Support (Mobile/Web)	☑	☑	☑	☑	☑	☑
API / Microservice Architecture	☑	✕	✕	✕	✕	✕

Table 1. Comparison of ExploreEase with other apps

ExploreEase uniquely combines reinforcement learning, clustering, real-time collaboration, smart routing, expense tracking, and trip album organization in a single AI-driven platform, unlike manually curated alternatives.

CHAPTER 3 – DOMAIN AREA

This domain emphasizes **Travel Planning & Super Apps**, which pulls together various travel services into an integrated platform designed to enhance the user's experience in trip planning.

3.1 KEY ASPECTS OF THE DOMAIN

Travel planning super apps employ technology to integrate a range of trip management functionalities into a single interface. Key goals include - reduce the confusion caused by fragmented travel planning, improve coordination and group decision-making and provide accurate travel recommendations with easy-to-understand financial reports. The journey of travel tools reflects this ideal growing togetherness:

1st Generation: Manual booking through an agent or on the phone, limited sourcing, and no analytics

2nd Generation: Online Travel Agencies (OTAs), but no group plans or recommendations

3rd Generation: Itinerary builders, but did not have booking, group communication, and expense management

4th Generation: AI travel planner apps, but not actual intelligence, generic AI support without group preferences, and equitable financial balance

5th Generation: ExploreEase, a super app, web-based, that integrates AI-powered recommendations, group focus and planning, fair cancellation policies, and shared expense tracking management.

3.2 CHALLENGES IN THE PROJECT

Developing a full-featured travel planning super app like ExploreEase will face essential hurdles in both the functional and technical worlds. Fragmentation means creating a single platform to book, communicate, and track expenses. Group behavior means being considerate of personal preferences while also allowing collectively fair decisions.

Financial transparency means providing clear, simple, user-friendly policies on budgets, cancellations, and refunds in order to build trust. From a technical perspective, cross-platform development will be focused on React Native and Flask/Django, and AI company libraries (like TensorFlow and Pytorch) for general and group recommendations.

Real-time status updates and messaging functions will happen through web sockets hosted through AWS EC2 and from Objects in S3. Managing data will be done with a mix of MongoDB for unstructured data and PostgreSQL for structured travel and user data. Overcoming those challenges will be fundamental in developing a trustworthy, scalable, and valuable travel planning solution.

CHAPTER 4 – EMPATHIZE STAGE

The empathize stage is the base stage when we take the time to immerse ourselves in our users to know what they're feeling, where they struggle, who they worry about, and what their ambitions are. If we really understand our users, we can be confident that every solution we design meets their real needs and provides real value.

4.1 ACTIVITIES

At this stage, we used numerous methods to gain authentic user perspectives:

We have conducted surveys with several participants, representing solo travelers, family groups, and group travelers

We have interviewed 10 travelers who regularly and occasionally travel to gain insights into their planning journeys

We have observed some groups independently planning trips for their group and were able to see how they communicate, how they work through any conflicts and decisions, who is responsible for what, how they manage bookings, and any costs

These activities provided us with a rich view of user behavior, pain points, and motivations.

4.2 PRIMARY RESEARCH

The main method of research we undertook was to learn about user challenges using direct user interaction:

Surveys and Interviews: Questions included methods of organizing travel, issues with group planning, where they decide where to go to visit, comfort level with online booking, and cancellation experiences

User Observation: Let groups plan their trips without intervention to observe their real-time problem-solving and communication patterns.

User Personas: Built out detailed personas. Fig.1(a) and 1(b) show the user persona of two persons.

Persona 1:

Laksheta S.V. - An 18-year-old college student who organizes group trips but dislikes the complexity involved.

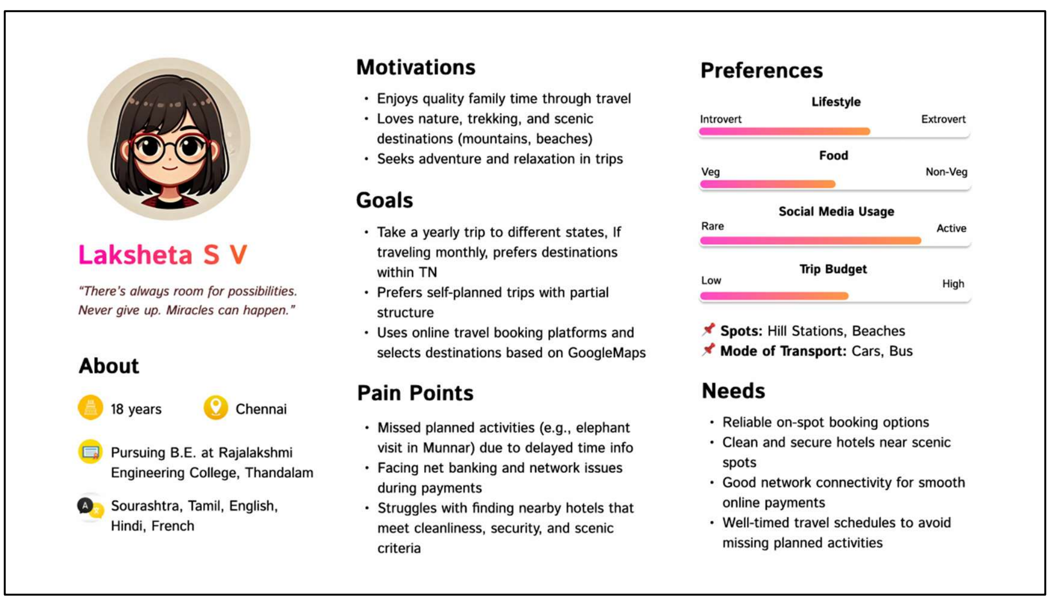


Fig.1(a). User Persona 1

Persona 2:

Sabitha R - A family-focused traveler seeking safe, budget-friendly hill station getaways.

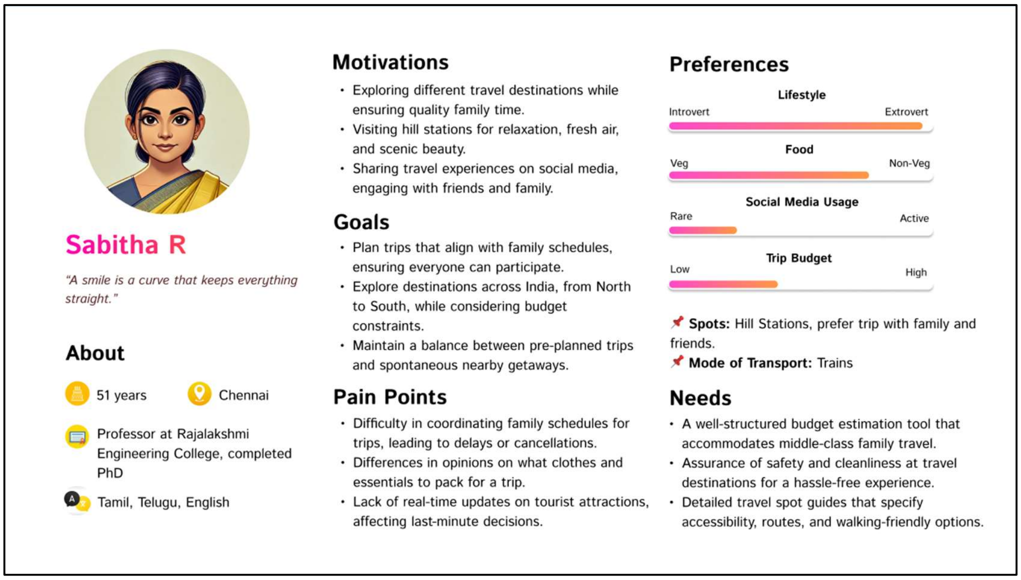


Fig.1(b). User Persona 2

4.3 SECONDARY RESEARCH

We have also conducted secondary research by consulting travel industry professionals and IT experts:

Travel Industry Professionals: Identified patterns in customer pain points, matching preferences, and gaps in product delivery.

IT Experts: Clarified what to include in a scalable app, how to build recommendations using AI/ML, considerations for real-time communication and accounts monitoring, data privacy issues, and optimizing performance across devices and networks.

4.4 OUTCOME OF THE EMPATHIZE STAGE

The empathy stage resulted in a complete synthesis of all the research findings, allowing us to identify the main pain points and desires of the users to develop an informed understanding of the key travel-related challenges, and develop problem statements to better direct design and development decisions moving forward in a clear and manageable way.

This stage prioritizes features that address real-life issues when it comes to travel planning, including group coordination, expense coordination, and flexible booking policies.

We have created Empathy Maps (refer to Fig.2) from user interviews and survey results, identifying what users think, feel, say, and do when planning their travel.

We have also created User Journey Maps (refer to Fig.3) outlining the end-to-end travel planning process and highlighting touchpoints, pain points, and areas of opportunity for innovation.

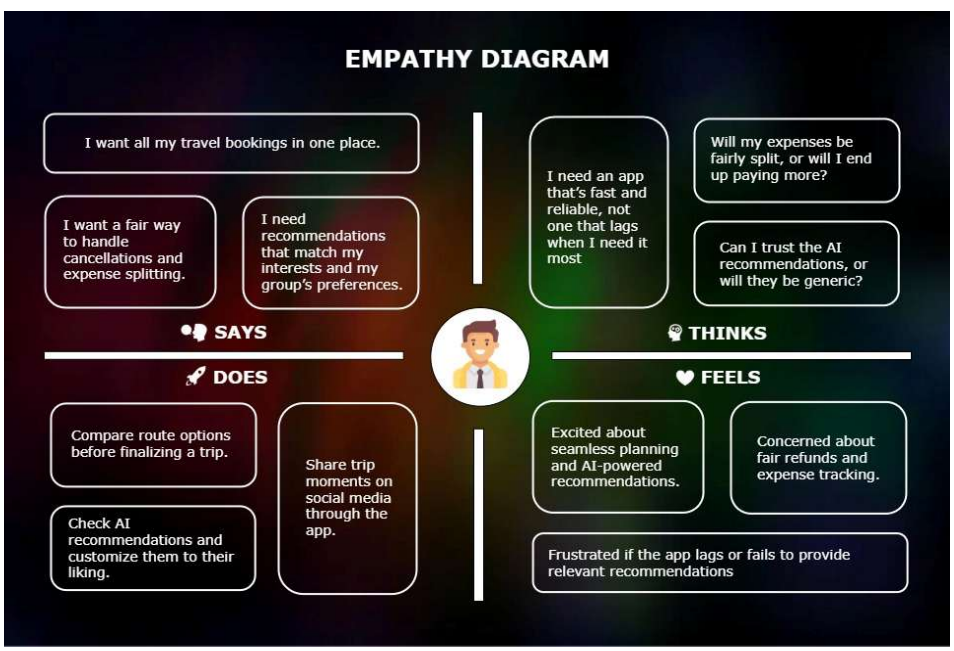


Fig.2. Empathy Map

	Phase 1 (Trip Ideation)	Phase 2 (Trip Planning)	Phase 3 (Booking)	Phase 4 (During Travel)	Phase 5 (Expense Tracker)	Phase 6 (Post Trip)
Touchpoints	Social media, Friend's Recommendation	Group chat, AI suggestions	App booking system, payment portals	In-app navigation, notifications	Expense tracker, refund or cancellation system	Social sharing, reviews, memories
Actions	Discuss trip ideas, search for destinations	Trip accomodation	Book hotels and transport, manage trip costs	Navigate routes, visit attractions	Track expenses, manage trip cancellations	Share experiences, leave reviews
Emotions	Excitement, curiosity	Overwhelmed	Confident, satisfied	Enjoyment	Relief	Inspired for next trip
User Needs	Curiosity for destinations	A well organized platform with real-time suggestions	A seamless booking experience	Flexibility	Transparent expense tracking	Share experiences
Pain Points	Too many options to choose	Biased suggestions	Complex refund policies	Last minute issues	Refund disputes	Limited Social Interactions
Opportunities	AI-driven personalized recommendations	Smart group planning	Budget friendly, fair refund system	Enjoyable route suggestions	Automated Refunds	Trip history for future planning

Fig.3. Journey Map

CHAPTER 5 – DEFINE STAGE

In this stage, we analyzed all the information collected during the empathize phase and consolidated it into clear problem statements to guide our design efforts.

5.1 USER NEEDS IDENTIFIED

The Empathize phase indicated user needs as follows: A single platform for planning, booking, and communication, AI-suggested options based on the interests of the group, easy, transparent expense tracking for the group, a fair approach to cancellations and refunds, and a built-in group chat for sharing updates and media.

5.2 BRAINSTORMING TO DEFINE PROBLEM STATEMENTS

We held a series of brainstorming sessions by our team, using "How Might We" prompts and journey mapping. This process revealed key issues in planning group trips:

- Difficulty managing trip planning across platforms.
- Conflicts with group preferences and no real clear plan for managing finances.
- The communication by the group was scattered across platforms.
- There was no simple way to sort through last-minute cancellations.

These insights were parlayed into our final problem statement.

5.3 FINALIZED PROBLEM STATEMENT

Travelers who plan group trips have trouble managing the separate parts of the trip, adjusting to everyone's interests, handling group finances and facing unexpected cancellations. A well-designed super app should guide trip planning, make useful suggestions, assist in group communication, and manage finances easily for everyone.

5.4 KEY USER NEEDS

Users' value is to be able to use a single platform for every phase of trip planning, from researching places to making reservations and messaging.

People prefer AI to suggest recommendations that are based on their group's preferences, highlight shared interests, and allow room for personalization.

Users should be able to monitor group expenses openly and get a fair and simple system for sorting out cancellations and requesting refunds.

Group members should be able to use an in-group chat to exchange details, pictures, and information easily.

5.5 DESIGN GOALS

In order to solve these problems, we set the following goals: We have come up with an AI recommendation system that learns from individual and group choices. We allow users to customize their trip by choosing between budget, time, or scenery goals. We make canceling an order and requesting a refund simple with a poll process. We link a complete chat system to an interface that supports sharing videos and pictures. We make an interface that allows people to stay within their trip budget easily.

CHAPTER 6 – IDEATION STAGE

6.1 ANALYSIS OF THE PROBLEM STATEMENT

The major problems are related to separated trip planning technologies, the inability to aggregate preferences from different group members, the uncertainty of cancellation processes, and the lack of transparency in tracking shared receipts. Overall, these represent barriers to ideating functional, impactful and human-centered solutions.

6.2 MIND MAPPING

We used the "How Might We (HMW)" prompt approach to think of ideas:

HMW make trip planning easy for groups?

HMW personalize travel recommendations for different users?

HMW make cancellations equitable for groups?

This catalyzed our ideation beyond traditional thought.

6.3 RESULTS OF THE BRAINSTORMING SESSION

There were a lot of outstanding ideas, including the following: AI-driven group reasoning based on interests, Continuous routes based on budget, time, or scenery, Use of blockchain-based smart contracts for refunding payouts, In-app polling for group decision-making with cancellations, Automatic expense tracking without manual inputs, Mystery trip mode to gamify the travel planning experience. The ideas were fleshed out and analyzed for feasibility and impact.

6.4 SELECTED BEST IDEA

The final concept selected was a travel super-app including AI-fueled group and individual recommendations, dynamic route options, streamlined booking options for transportation and accommodation, group chat and coordination capabilities, clear tracking of expenses, polling mechanism for refunds and cancellations, and smart trip album creation

The solution would be the only one that addressed all of the core user pain points as a whole and in an intuitive design.

6.5 VALUE PROPOSITION STATEMENT

"Our travel super app changes the way group trips come together by making the entire process simple, smart, and personalized—combining AI-designed recommendations, on-the-fly routing, bookings built in, expense tracking capabilities, and collaboration in one platform."

CHAPTER 7 – PROTOTYPE STAGE

The prototype stage was all about creating an interactive model of our platform to simulate real user experience and gather actionable feedback.

7.1 TECHSTACK USED

The app was programmed with React Native and optimized for top responsiveness with CSS, so it is simple and suitable for all devices. For the backend, we used Python (Flask/Django) to handle AI and Node.js/Express.js for anything real-time and API. It uses MongoDB to store trip information and user choices, and PostgreSQL for keeping financial and other organized booking information.

For its recommendation engine, AI/ML relies on scikit-learn and NLTK Python libraries. By relying on WebSockets, the system provides fast updates and chats that happen in real-time during group planning. Media (including photos and documents) from users is safely stored with AWS S3. Route visualization and optimization are made possible using mapping APIs (like Google Maps API).

7.2 DATASET DESCRIPTION

A. Data Sources

The data that we used is a hybrid of real-world and synthetic data sources. We used APIs (Application Programming Interfaces) from **Google Places, Foursquare, and Eventbrite** to collect location and event information in 64 Indian cities. User profile information was obtained from the Kaggle **"Traveller"** dataset, and behavioral interaction information was synthetically generated using Python libraries such as Faker and random.

B. Data Collection

The location and event information were aggregated in a single-day API run and stored in JSON files. The user interaction data collectively consisted of connection strength and visit frequency, intentionally modeled synthetically based on realistic social and travel experience and behavior.

C. Dataset Composition

Places: We obtained over **2000 records**, covering **15 categories**, improved by applying rule-based tagging to data and BERT-based classification on unmatched entries.

Users: Profiles contained base location, age group (teenagers, middle aged, elderly persons), as well as sentiment score.

Interaction: The interaction information constituted simulated social graphs, where connections were made up of three strength based levels (0-2) and numeric frequencies of visits.

D. Data Cleaning and Feature Engineering

We removed irrelevant fields (e.g., postal codes, etc.), coded the connection strength as numerically 0-2, and engineered features including **Average number of visits per location, User activity levels, Distinct locations visited, etc.**

Users were clustered by **K-means** (with the elbow method returning four clusters as shown in Fig.4), and we plotted dimensionally reduced data using **Principal Component Analysis (PCA)**.

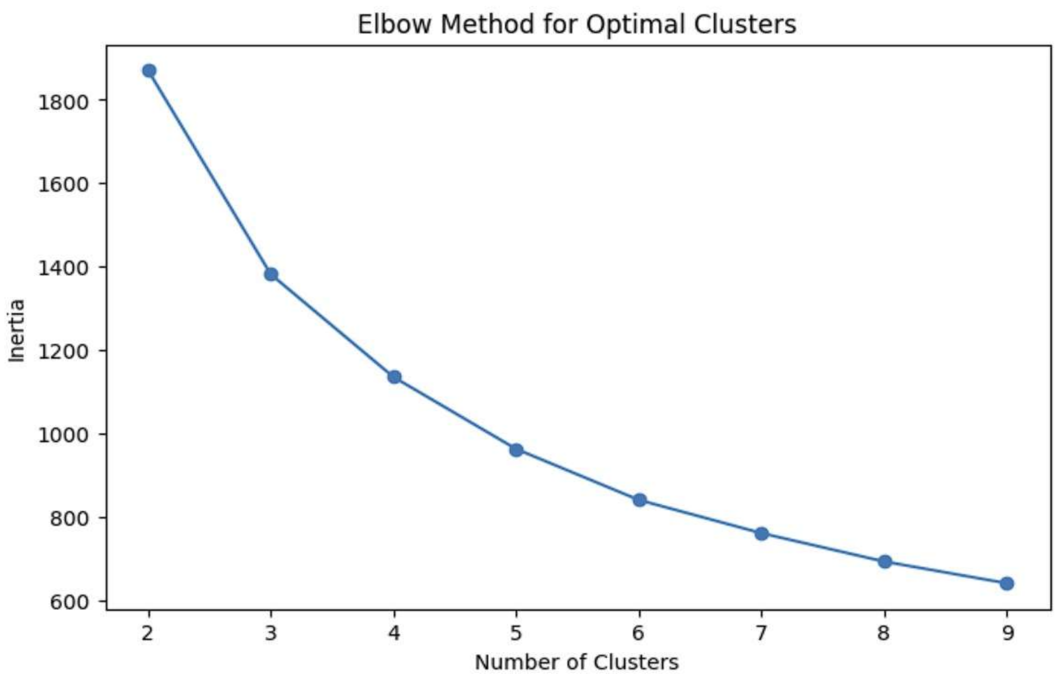


Fig.4. Elbow method to find optimal cluster (here k=4 is selected)

The entire dataset was anonymized and extracted from publicly available or synthetically generated sources.

7.3 METHODOLOGY

ExploreEase is built with a modular microservice architecture designed for scalability and real-time integration (which is clearly shown in Fig.5). ExploreEase uses RESTful APIs and MongoDB, which will allow seamless communications between services and reliable data persistence.

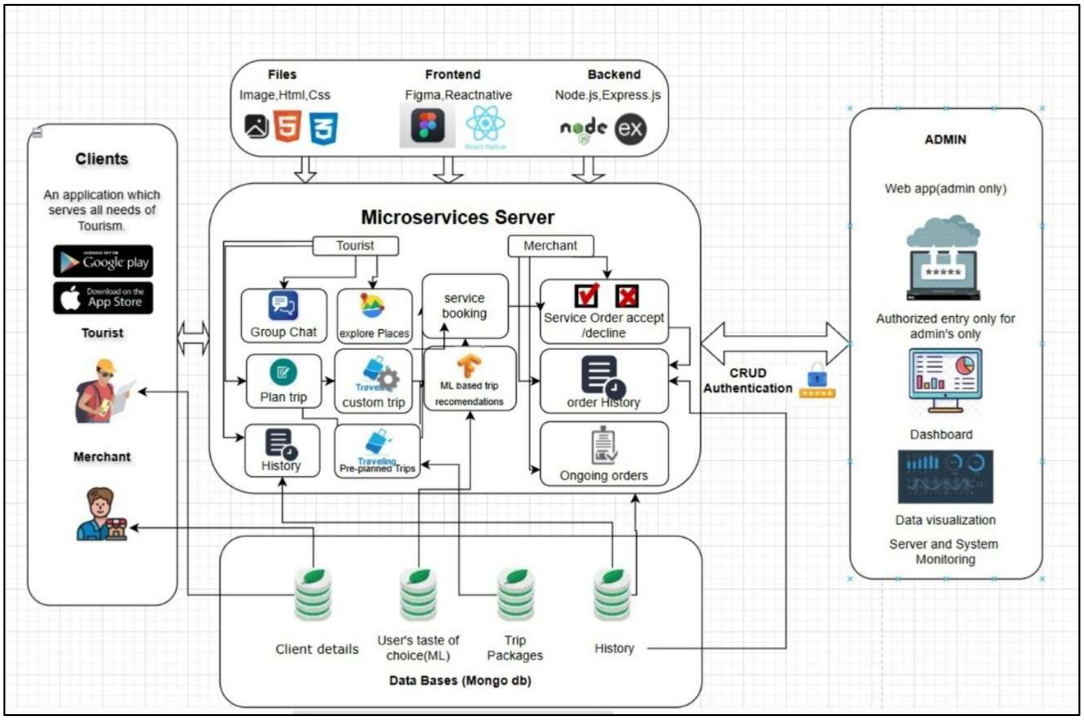


Fig.5. System Architecture Diagram

We're purposefully following a modular approach so we can monitor the modules independently for maintenance and improvement. The system is organized into core functional modules as follows:

A. <u>Personalized/Group-Based Recommendations Module</u>

This module utilizes several machine learning methods to build a recommendation pipeline:

K-Means clustering is used to create user behavioral groups utilizing a clustering objective based on trip data, trip frequency, and the depth of exploration through the social nature of the user behavior model.

Principal Component Analysis (PCA) is used to visualize K-means clustering and to reduce the high-dimensional data to create a meaningful representation of groupings. This is shown in Fig.6.

Neural Matrix Factorization (NMF) is used to learn the latent embeddings to identify representations of user-place relationships to accurately predict the relative travel preferences and desires of a user. The 3D PCA view of all embeddings is clearly shown in Fig.7.

Reinforcement learning is used to update user preferences to adjust for drift using highly iterative engagement feedback.

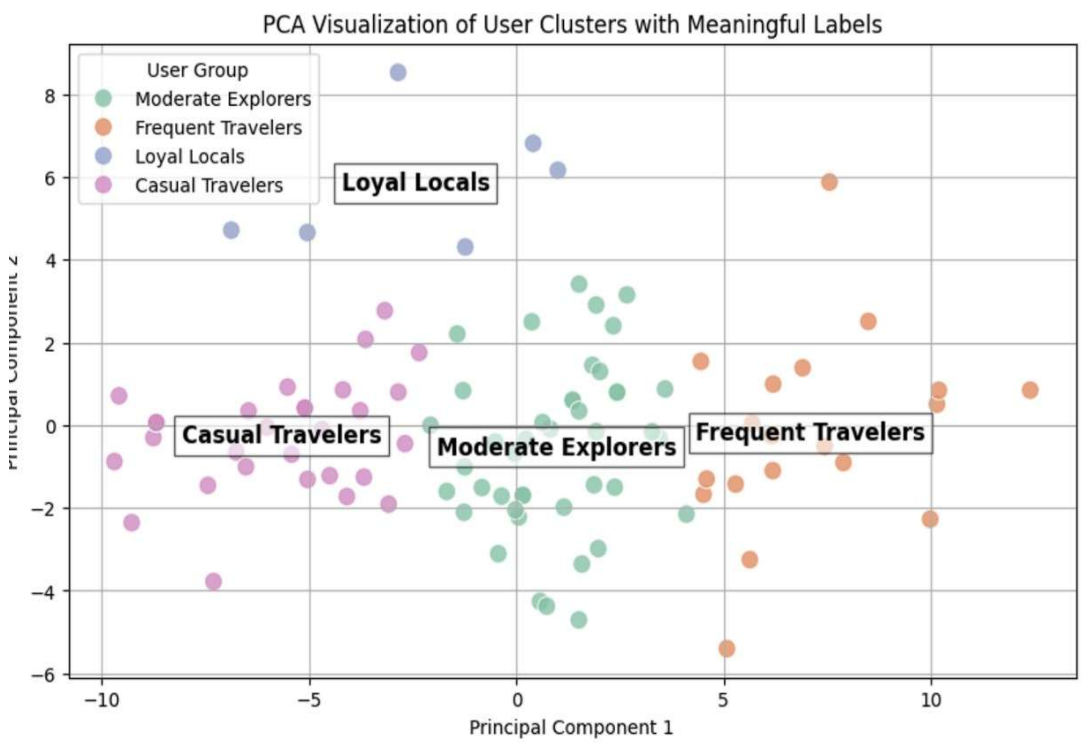


Fig.6. PCA Visualization of User Clusters with labels

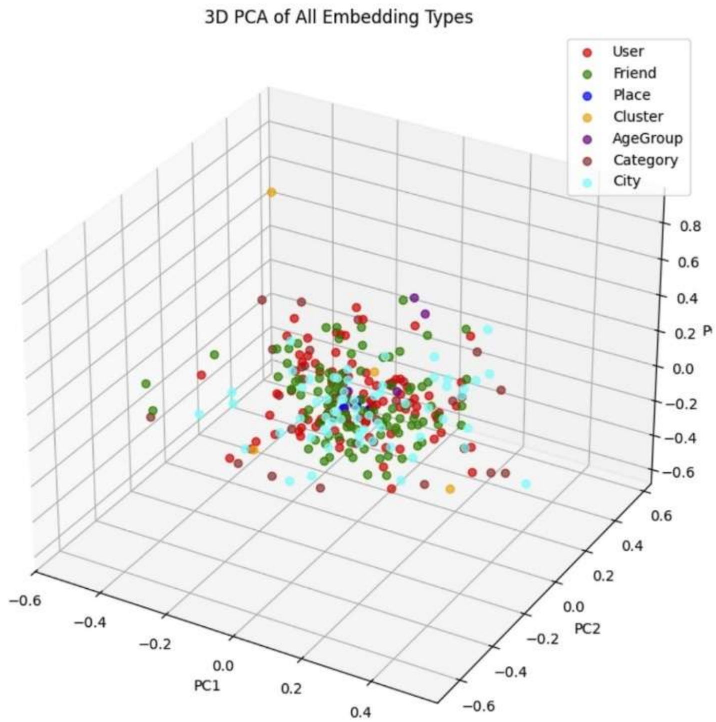


Fig.7. 3D PCA of all Embeddings

A **Unified Recommendation Score (URS)** is averaged using a weight assigned to clustering, NMF-based score output, and reinforcement update based on weighting options.

$$\mathbf{URS\ (u, p) = \alpha \cdot Cu\ (p) + \beta \cdot Nu\ (p) + \gamma \cdot Ru\ (p)} \qquad \textbf{[1]}$$

Where Cu(p) is the cluster consistency score, Nu(p) is the NMF-based preference score, and Ru(p) is the reinforcement learning update score.

The parameters α, β, and γ are adjustable weights to tune the contributions of the components in the final recommendation score.

B. Route Optimization Module

The route optimization module gives the user three options for travel routes:

Time-Optimized: The Google Maps API provides the fastest route based on real-time traffic and distance.

Cost-Optimized: The Travel plan will be calculated based on the overall cheapest travel plan by looking at fuel costs, transportation fares, and ride share options.

Scenic Route: A modified Dijkstra's or A* algorithm, where the weights on each leg of the travel route are based on the formula,

$$\mathbf{W_{edge} = \alpha \cdot T - \beta \cdot S} \qquad \textbf{[2]}$$

Where T is travel time, and S is the scenic score calculated based on photo metadata, elevation data, and user-generated content.

Scenic routes have limits so their travel time does not exceed 30% greater than the minimum travel time to travel between the origins and destination. The map shows the three routes – cost-efficient, time-efficient, and scenic route from Chennai to Bangalore is shown in Fig.8.

Implementation options include:

- Google Maps API for time and cost route.
- NetworkX (Python) for scenic route path finding.
- OpenStreetMap and Google Places to get map metadata and user input.

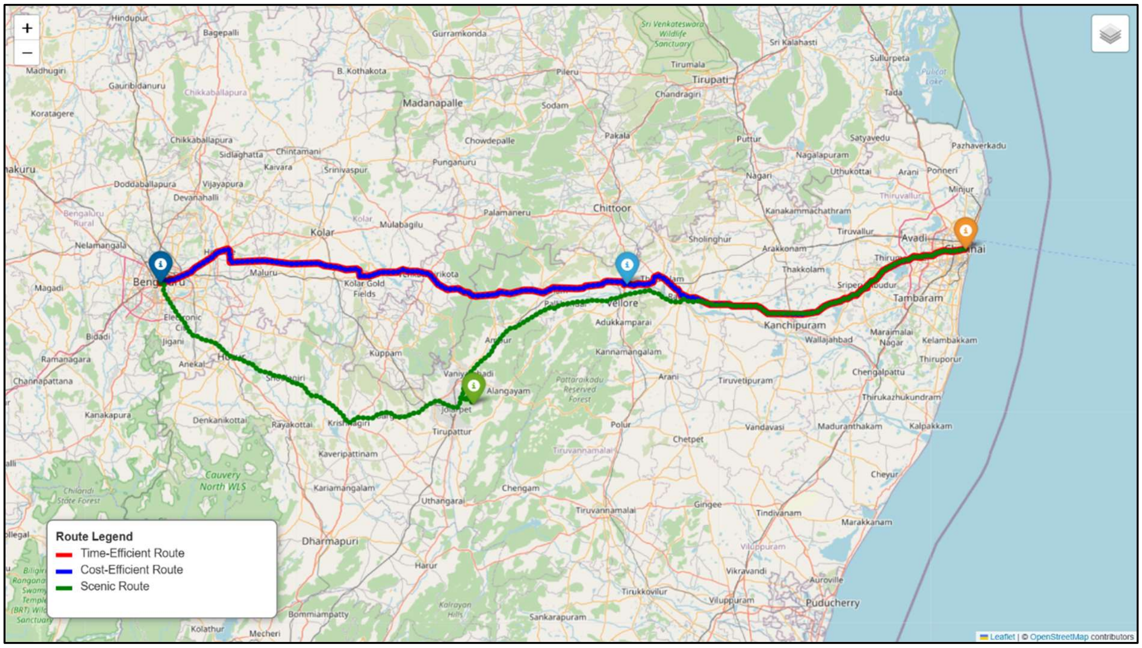


Fig.8. Map showing three route suggestions:

time-efficient (red), cost-efficient (blue), and scenic route (green)

C. <u>Supporting Modules</u>

Dynamic Trip Packages: Created through the analysis of historical travel data, including seasonal trends and user interest clusters.

Hotel and Transport Booking: Developed as microservices consuming third-party API's e.g., Skyscanner, Redbus, IRCTC.

Group Chat & Polling System: A simple real-time messaging and decision-making tool developed using socket programming and a custom event-driven structure as a part of the Trips application.

Expense Tracker: Using MongoDB as the structure to store categorized expenses. The expenses can be personal or part of a group tracking total expenses.

Trip Album Manager: Auto-manages uploading user-generated content using EXIF Metadata and time-stamp, and groups around like geo-locations that enable easy digital trip album creation.

D. Implementation & Integration

Front-End: Developed from the ground up to work as a mobile-first application to provide cross-platform performance for mobile and user interface.

Back-End: Built from the ground up using Node.js for REST API endpoints and business logic, asynchronous usage of functions.

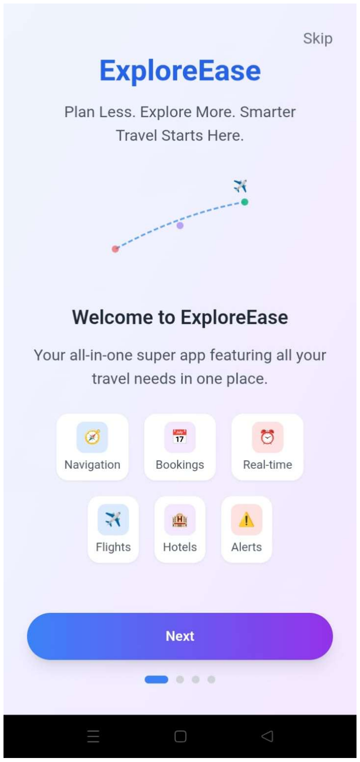
Machine Learning Models: Built in Python and exposed to the back-end as RESTful API's allowing for the model training and inference to not respond with blocking behavior to any of the user operations.

System Architecture: Required a modular orchestration of services using REST API endpoints to take care of session, security, authentication, and scalable micro-services.

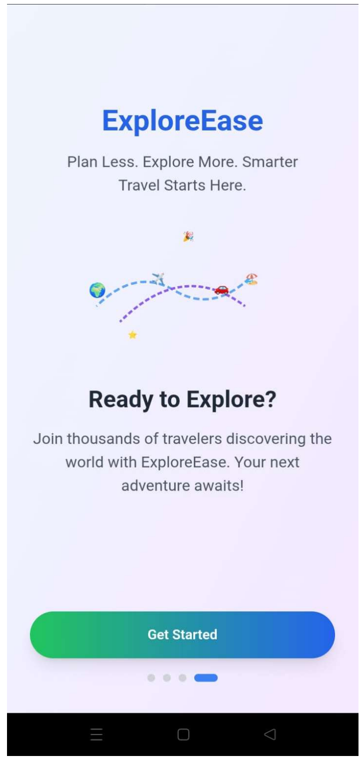
7.4 PROTOTYPE UI VISUALS

The **Trip Planning Interface** allows users to set the trip date, choose what activities they want. The **Route Selection Page** provides ways to look at routes that are budget-friendly, quick, or loaded with beautiful places to see. With the **Group Chat Interface**, communicating, sharing booking links, photos, and polls for group selection is easy. The system shows **poll choices for a refund** or sharing the costs, so others in the group can decide. Shows **shared expenses** and your expenses properly sorted with charts in the Dashboard.

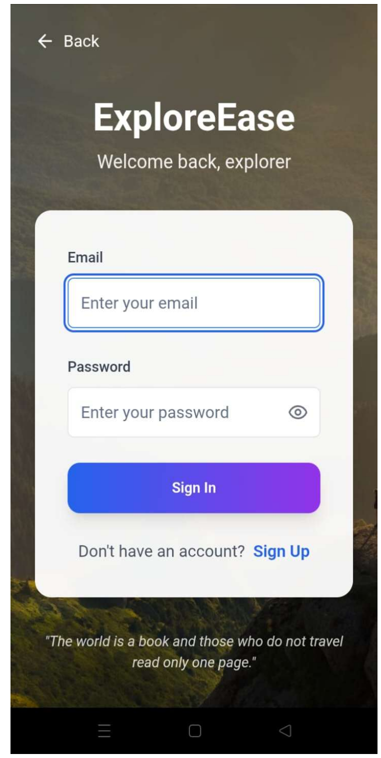
The UI of ExploreEase is shown on the following pages – Fig. 9(a) to 9(m).



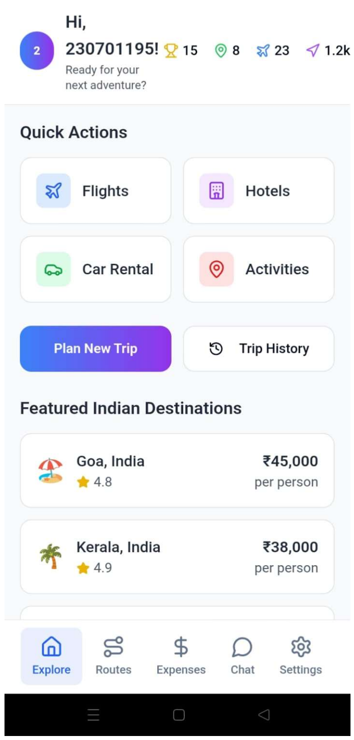
9 (a)



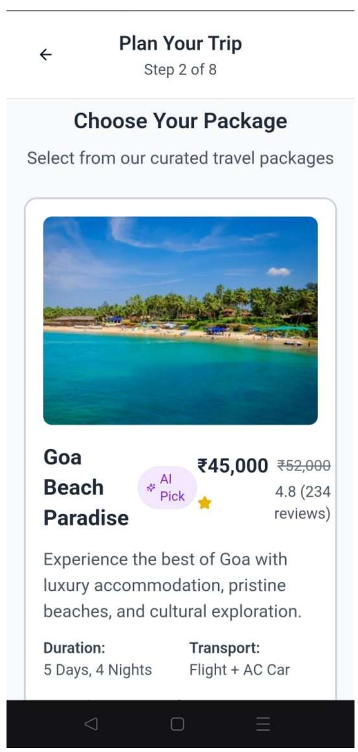
9 (b)



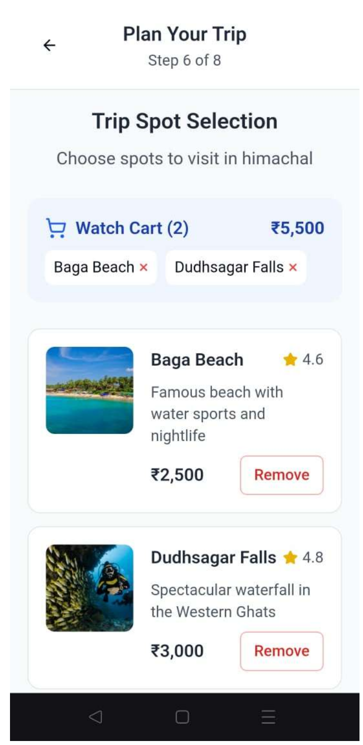
9 (c)



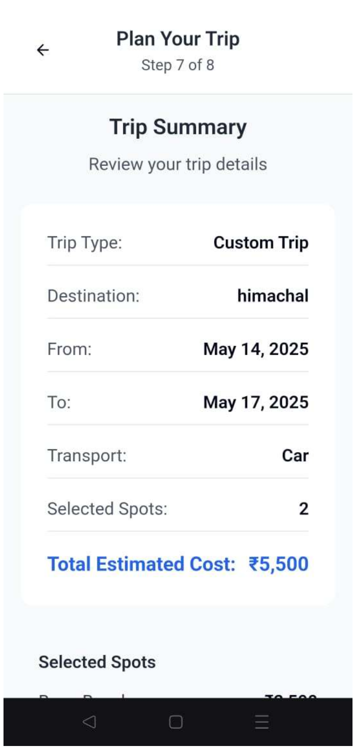
9 (d)



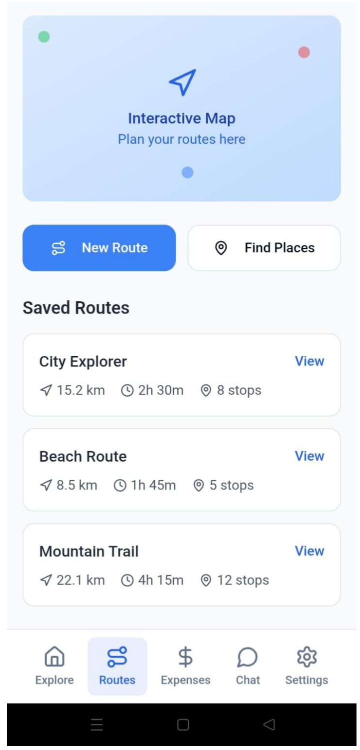
9 (e)



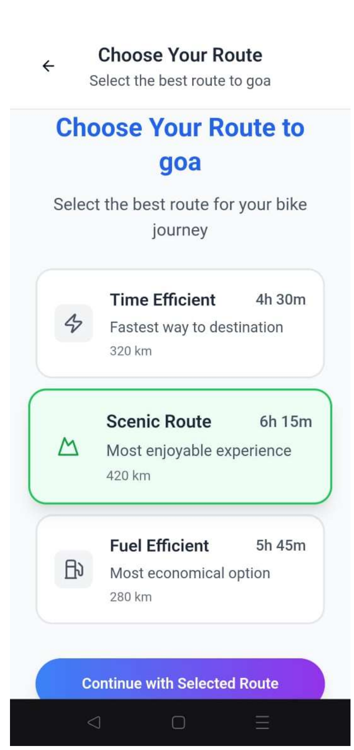
9 (f)



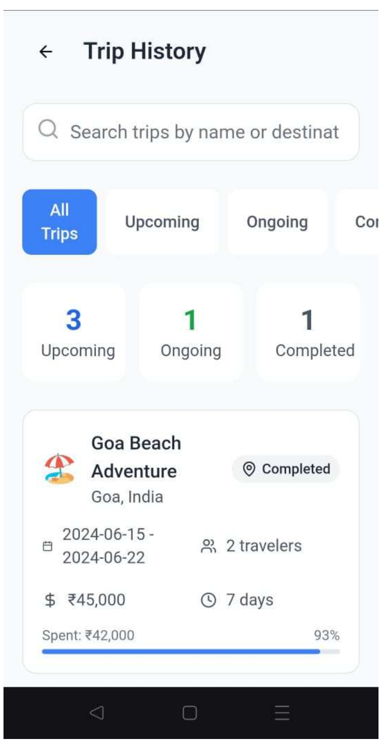
9 (g)



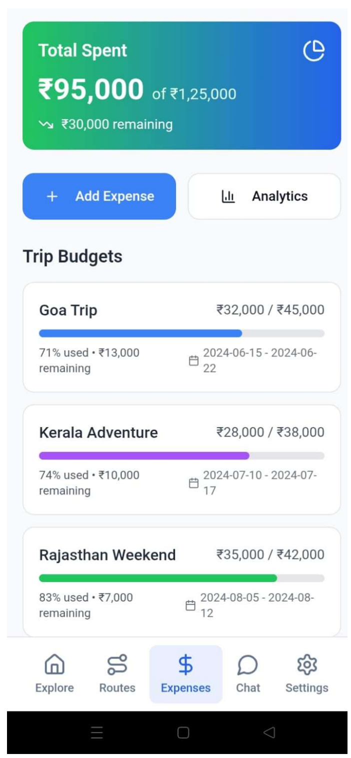
9 (h)



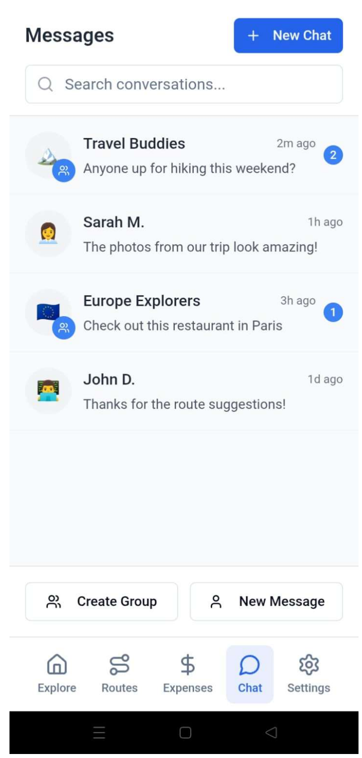
9 (i)



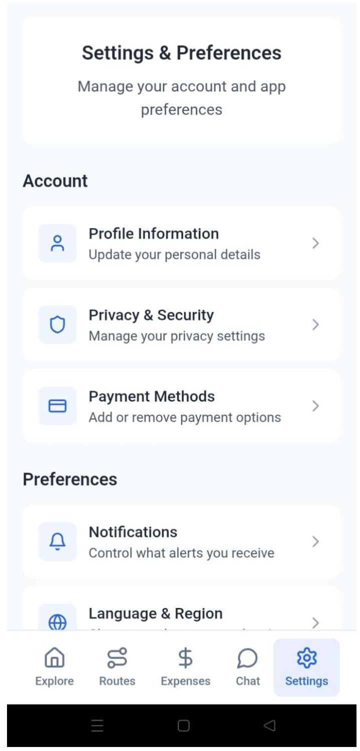
9 (j)



9 (k)



9 (l)



9 (m)

Fig.9. UI screenshots of ExploreEase super-app

- ***9(a)*** ***Starting Page***
- ***9(b)*** ***Launch Page***
- ***9(c)*** ***Login Page***
- ***9(d)*** ***Homepage Dashboard***
- ***9(e)*** ***Plan Your Trip – Trip Packages Page***
- ***9(f)*** ***Plan Your Trip – Trip Spot Selection Page***
- ***9(g)*** ***Plan Your Trip – Trip Summary Page***
- ***9(h)*** ***Routes Page***
- ***9(i)*** ***Choose Your Route – Three Routes Selection Page***
- ***9(j)*** ***Trip History Page***
- ***9(k)*** ***Expenses Page***
- ***9(l)*** ***Chat Page***
- ***9(m)*** ***Settings Page***

CHAPTER 8 – EXPERIMENTAL RESULTS

8.1 TEST

A. Performance of the Trip Recommendation Module

The performance of the recommendation engine has been computed using the standard regression metrics, and data shows good accuracy, as shown in Table 2.

Metric	Value
Test Loss	0.0275
RMSE (Root Mean Squared Error)	0.0008
MAE (Mean Absolute Error)	0.1344
R^2 Score	0.9711

Table 2. Recommendation System Performance Metrics

The data indicates that the overall approach of integrating K-Means Clustering, Neural Matrix Factorization, and Reinforcement Learning to probabilistically capture where individuals choose to move has achieved 97.11% predictive accuracy in situational contexts.

B. Route Optimization Test Case

A practical test was carried out and focused on the Chennai to Bangalore route. The trips generated three optimized routes, which are seen in Table 3.

Route Type	Time	Distance	Key Feature
Time-Efficient	5 hr 30 min	347 km	Fastest via NH48 with real-time traffic data
Cost-Efficient	6 hr 15 min	325 km	Public transport (train + bus) for low cost
Scenic Route	6 hr 45 min	360 km	Through Yelagiri Hills and forest regions

Table 3. Chennai to Bangalore Route Optimization Outputs

The scenic aware routing algorithm was built using a variation of the Dijkstra/A* algorithm that includes travel time restrictions, estimated visual appeal/beauty scores which may take up to 30% added time beyond the shortest path.

C. Baseline Comparison

ExploreEase was compared with two standard models as shown in Table 4:

Model	RMSE	MAE	R^2 Score
ExploreEase (Proposed Model)	0.0008	0.1344	0.9711
Naive Average Recommender	0.1510	0.2329	0.6124
Traditional CF (KNN-based)	0.0923	0.1871	0.7432

Table 4. Baseline Comparison of Recommended Models

The ExploreEase model has consistently outperformed baseline models to provide personalized and context-aware recommendations.

8.2 SUMMARY RESULTS

- High values of R^2 with low values of RMSE confirm the predictive accuracy of the model.
- Route optimization achieves a balance/enhanced perspectives of time, cost, and scenery.
- ExploreEase outperforms baseline models on all performance metrics.
- Supports individuals and groups planning travel scenarios.

CHAPTER 9 – REDESIGN AND IMPLEMENTATION

9.1 REDESIGNED ACTIONS BASED ON FEEDBACK

During evaluations and early stage testing, user and stakeholder feedback was collected. Here are the salient actions of revision taken from their feedback:

Improved Cold-Start Handling:

Users were unable to use the app when they lacked historical interaction. Although, there were many ways to fix this, the team decided to integrate the user's sentiment and demographics into the recommender heuristics in the calculated recommendations for the user.

Improved Scenic Scoring Logic:

In previous versions there was a stronger emphasis on scenic features some of the time than others. The rationale was changed to set a 30% travel-time threshold, and in a more normalized way use some scores from both user generated content (e.g., images, videos, and reviews) and elevation data to improve the scenic ratings.

Improved Group Recommendation Logic:

Feedback suggested that there was misalignment in preferences in groups. Scenarios suggested that the group must get confirmation from all group members prior to agreeing on a recommendation. Thus, the aggregation function was improved to be sensitive to dominant preferences, but also be agnostic to minority users by using adaptive weighting.

Changes to UI/UX:

While conducting usability reviews, changes to the UI of the mobile app was made based on testing. Functions that had change were focused on displaying cleaner tab-based navigation, user interface changes to the parking polling during real-time, and an improved display of alternative routes compared with time.

API Handling of Microservices:

During the testing of the real-time chat service and the booking service, there was a considerable amount of slow responses in API calls to the service. In order to rectify this, we made small changes to use an asynchronous style of handling, along with batching API requests - which collectively sped the response time to around 40%.

9.2 IMPLEMENTATION

The new design architecture was designed around a modular and scalable approach. The front-end was built with React Native, allowing responsive design and performance-optimized screens for multi-platform deployment.

The back-end was built on Node.js, where business logic was created, data routing and service orchestration were handled through REST APIs. Key integrated modules included: User & Session Management, Trip

Recommendation Engine, Route Optimization, Booking & Polling Microservices.

The Python-based machine learning layer was built using TensorFlow, scikit-learn, and PyTorch libraries and hosted as a REST service using Flask. Core models included; K-Means Clustering, Neural Matrix Factorization, and Reinforcement Learning Engine.

The database was managed with MongoDB for APIs controlling user profiles, user interactions, and trip history, while Redis was used to manage sessions and to speed up lookups. Real-time communication features, like group chat and polls, used Socket.io and inter-module communication was kept clean with RESTful endpoint calls.

CHAPTER 10 – CONCLUSION

Our "ExploreEase" super app demonstrates how Design Thinking can be used to effectively address real-world travel planning issues. We created a platform that addresses common problems that travelers encounter, like disjointed planning, coordinating a variety of preferences, and equitably managing trip finances, by concentrating on understanding user needs, testing in small steps, and listening to feedback.

Users can enjoy AI-driven trip planning, choose the best routes, interact with groups easily, manage shared expenses, and fairly settle cancellation disputes in this app's extensive and user-friendly space. To make it reliable and user-friendly, we used technologies like React Native, Python, MongoDB, and WebSockets. By including mapping APIs and cloud services, users were able to organize and visualize their trips easily, increasing their awareness of their methods. Additionally, we conducted pilot testing with actual users, and we improved based on their input. Users appreciated the cancellation policy's unique fairness and how easy it was to receive insightful recommendations.

This project demonstrated the effectiveness of Design Thinking in developing solutions that genuinely prioritize people, in addition to assisting users in making travel plans in a helpful and integrated manner. We can't wait to add more features like global accessibility, integration with additional service providers, and sophisticated AI for individualized experiences. Ultimately, this app is more than just an app; it's a means of offering everyone accessible, entertaining, and adventure-focused travel planning.

CHAPTER 11 – FUTURE WORKS

For maximum scalability and richer user experiences, several future enhancements can be considered for further versions of ExploreEase:

Better Destination Accuracy:

Through development and access to an expanded world place database and advanced processing power, the application can identify and suggest destinations with much higher accuracy. This will allow for the broadening of travel locations worldwide.

Blockchain Integration:

Using blockchain for the storing of travel records and handling user identification offers upgraded security and reliability of the platform. This will increase security and guarantee that user data and travel history cannot be altered or manipulated, and require verification.

Voice-Based Virtual Assistant:

A future direction that has the greatest development potential is a voice-based virtual assistant. This would allow users to talk to the application in common vernacular and more easily create, edit, and manage trips while quickly traveling.

These enhancements are focused on improving ExploreEase to not only be more powerful and secure, but also more user-friendly, intuitive, and globally accessible.

CHAPTER 12 – LEARNING OUTCOMES OF DESIGN THINKING

This section briefly highlights the learning outcomes of Design Thinking, and presents some key skills and mindsets that you develop during the process.

Empathizing with Users

The team began by understanding the real frustrations of travelers—traveling is too fragmented while trying to utilize too many apps and options, no personalization or customization, and no cohesiveness when planning in a group. This helped identify features such as group chat, decision polls, and sharing expenses that addressed the user's pain points.

Reframing the Problem

The team reframed the task to identify as building a smart travel super-app rather than just a simple travel app. This helped expand their scope from building and booking to planning, recommended itineraries, and collaboration.

Encouraging Innovation

In our ideation phase, the team was excited to play with new technologies such as K-Means clustering, Neural Matrix factorization (NMF), and Reinforcement Learning. Our team ended up building a single recommendation engine that could adjust based on real-time user preferences.

Prototyping with Purpose

Our team was able to move quickly to build prototype methods and combine them together using various development technologies from React Native for front-end and REST APIs for services in the background. We used fake user data algorithmically searched using our APIs for location-based data to loop through our testing patterns as quickly as possible.

Running Tests Using Metrics

We generated clear metrics to evaluate our success: RMSE, MAE, and R^2 Score to measure model accuracy. This kind of quantitative feedback also assured that our design decisions were user-centered and technically feasible.

Designing for Collaboration

ExploreEase was guided by social collaboration through design experience and development. It developed features such as group travel planning and in-app communication with shared decision-making in mind, connecting users to each other and establishing an inclusive travel process.

Designing for the Future

From the very first stage of development, future possibilities for improvements were considered, ranging from exploring using blockchain for security and also incorporating voice-based assistants for accessibility. This shows an example of a design thinking mindset that aimed to focus on scalability, innovation, and most importantly, long-term value.

CHAPTER 13 – REFERENCES

A total of 15 research papers were reviewed, 11 contributed to idea development and feature design, and 4 lent support by developing machine learning models defined in the project.

13.1 MAIN JOURNAL PAPERS – FOR THE IDEA

1. V. Parikh, M. Keskar, D. Dharia, and P. Gotmare, "A Tourist Place Recommendation and Recognition System," *Proc. 2018 Int. Conf. Inventive Commun. Comput. Technol. (ICICCT)*, Coimbatore, India, 2018, pp. 218–222

doi: 10.1109/ICICCT.2018.8473077.

2. J. K. Kim, W. C. Kwon, I. Y. Choi, H. Heo, and H. S. Moon, "A Group Travel Recommender System Based on Group Approximate Constraint Satisfaction," *IEEE Access*, vol. 12, pp. 96113–96125, 2024

doi: 10.1109/ACCESS.2024.3427122.

3. Y. Kong, K. Yi, L. Wang, C. Peng, L.-M. Nguyen, and Q. Ma, "RPMTD: A Route Planning Model With Consideration of Tourists' Distribution," *IEEE Access*, vol. 12, pp. 69488–69504, 2024

doi: 10.1109/ACCESS.2024.3400373.

4. Y. Zhang, L. Jiao, Z. Yu, Z. Lin, and M. Gan, "A Tourism Route-Planning Approach Based on Comprehensive Attractiveness," *IEEE Access*, 2020

doi: 10.1109/ACCESS.2020.2967060.

5. A. Sharaff, S. Khurana, K. Cheepurupalli, and T. Sahu, “Personalized Recommendation System with User Interaction based on LMF and Popularity Model,” *Proc. 2020 Int. Conf. Syst. Comput. Autom. Netw. (ICSCAN)*, Pondicherry, India, 2020, pp. 1–6

doi: 10.1109/ICSCAN49426.2020.9262442.

6. T. Alenezi and S. Hirtle, “Normalized Attraction Travel Personality Representation for Improving Travel Recommender Systems,” *IEEE Access*, vol. 10, pp. 56493–56503, 2022

doi: 10.1109/ACCESS.2022.3178439.

7. P. Stefanović and S. Ramanauskaitė, “Travel Direction Recommendation Model Based on Photos of User Social Network Profile,” *IEEE Access*, 2023

doi: 10.1109/ACCESS.2023.3260103.

8. D. Qin, X. Zhou, L. Chen, G. Huang, and Y. Zhang, “Dynamic Connection-Based Social Group Recommendation,” *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 3, pp. 453–467, Mar. 2020

doi: 10.1109/TKDE.2018.2879658.

9. L. Tang, X. Lin, J. Yu, and C. Tang, “Integrating Travel Survey and Amap API Data Into Travel Mode Choice Analysis With Interpretable Machine Learning Models: A Case Study in China,” *IEEE Access*, vol. 13, pp. 27852–27867, 2025

doi: 10.1109/ACCESS.2025.3540082.

10. Z. Jia-Xiang, M. Zhan-Xiao, and G. Yan, “Mixed Traffic Mode Based Travel Route Recommendation,” *Proc. 2018 Int. Conf. Netw. Inf. Syst. Comput. (ICNISC)*, Wuhan, China, 2018, pp. 372–376

doi: 10.1109/ICNISC.2018.00082.

11. J.-H. Chang, C.-E. Tsai, and J.-H. Chiang, “Using Heterogeneous Social Media as Auxiliary Information to Improve Hotel Recommendation Performance,” *IEEE Access*, vol. 6, pp. 42647–42660, 2018

doi: 10.1109/ACCESS.2018.2855690.