
Software Requirements Specification

for

Route Mate

Version 1.3

Prepared by

Group 8:

Palagiri Tousif Ahamed	220744
Tamidala Venkata Sai Pawan	221120
Chanukya Reddy	
Vetcha Pankaj Nath	221188
Sudhamandu Sai Nikhil	221095
Mohammed Anas	220654
Himanshu Shekhar	220454
Swayamsidh Pradhan	221117
Kawaljeet Singh	220514
Anya Rajan	220191
Daksh Kumar Singh	220322

Group Name: Access_denied

atousif22@iitk.ac.in
venkatasai22@iitk.ac.in
vpankaj22@iitk.ac.in
ssain22@iitk.ac.in
mohdanas22@iitk.ac.in
shimanshu22@iitk.ac.in
spradhan22@iitk.ac.in
kawaljeets22@iitk.ac.in
anyaraian22@iitk.ac.in
dakshks22@iitk.ac.in

Course: CS253

Mentor TA: Sumit

Date: 26/01/2024

CONTENTS.....	2
REVISIONS.....	3
1 INTRODUCTION.....	4
1.1 PRODUCT SCOPE.....	4
1.2 INTENDED AUDIENCE AND DOCUMENT OVERVIEW.....	4
1.3 DEFINITIONS, ACRONYMS AND ABBREVIATIONS.....	6
1.4 DOCUMENT CONVENTIONS.....	6
1.5 REFERENCES AND ACKNOWLEDGMENTS.....	7
2 OVERALL DESCRIPTION.....	8
2.1 PRODUCT OVERVIEW.....	8
2.2 PRODUCT FUNCTIONALITY.....	11
2.3 DESIGN AND IMPLEMENTATION CONSTRAINTS.....	12
2.4 ASSUMPTIONS AND DEPENDENCIES.....	12
3 SPECIFIC REQUIREMENTS.....	15
3.1 EXTERNAL INTERFACE REQUIREMENTS.....	15
3.2 FUNCTIONAL REQUIREMENTS.....	26
3.3 USE CASE MODEL.....	28
4 OTHER NON-FUNCTIONAL REQUIREMENTS.....	36
4.1 PERFORMANCE REQUIREMENTS.....	36
4.2 SAFETY AND SECURITY REQUIREMENTS.....	36
4.3 SOFTWARE QUALITY ATTRIBUTES.....	36
APPENDIX A – DATA DICTIONARY.....	38
APPENDIX B - GROUP LOG.	

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
1.0	Group 8- Access denied	- An initial version of the document was drafted. - Sections 1.1, 2.1, 2.2, 2.3, 3.1 were added.	21/01/2024
1.1	Group 8- Access denied	-Section 1.2, 2.2, 2.3, 3.2, 3.3 was added to the document.	23/01/2024
1.2	Group 8- Access denied	-Sections 1.3, 1.5, 2.4, 4.1, 4.2, and 4.3 were added.	25/01/2024
1.3	Group 8- Access denied	- Few changes in the format of document - Document was finalized.	26/01/2024

1 Introduction

1.1 Product Scope

Our website, RouteMate, aims to organise shared rides and planned trips among the vibrant community of students from IIT Kanpur during holiday breaks. Designed to foster camaraderie and make travel more convenient, RouteMate connects students looking to journey home during vacations and visit tourist places.

Our user-friendly website allows IIT Kanpur students to create and join groups based on their destination cities and travel modes. Simply sign up, input your travel details, and find fellow students heading in the same direction. Whether you're headed north, south, east, or west, RouteMate aims to make your holiday travel a seamless and enjoyable experience.

Key Features:

1. **Destination-Based Matching:** Users input their travel destination, and the software matches individuals with similar itineraries, creating an instant connection between travelers heading to the same place.
2. **Safety and Verification:** Prioritize safety with verified profiles within the IIT Kanpur community. Get to know your potential travel buddies through user profiles and reviews.
3. **Flexible Scheduling:** Coordinate travel plans with ease by discussing departure times, meeting points, and other details within the group chat. Stay flexible and accommodate each other's schedules.
4. **Group Trip Planning:** Users can create or join travel groups based on their interests or activities, allowing for seamless coordination and planning of group activities at the destination.
5. **Cost-Sharing:** Save on travel expenses by sharing costs with your co-travelers. RouteMate fosters a cost-effective and eco-friendly approach to holiday travel.
6. **Combined Trip Blog:** Collaborate on a shared trip blog with all members of your travel group. Capture and document your collective experiences, memories, and photos in one place, creating a digital journal that each member can contribute to.
7. **Real-time Chat and Messaging:** A secure and user-friendly chat feature enables travelers to communicate in real-time.
8. **Expense Splitting:** Facilitates easy expense splitting between travel companions, making it convenient for users to divide costs for accommodation, transportation, and other shared expenses.

1.2 Intended Audience and Document Overview

This section outlines the specific audience for whom this System Requirements Specification (SRS) is crafted and provides a brief overview of the document's structure and contents.

1.2.1 Intended Audience:

The primary audience for this SRS encompasses various stakeholders involved in the conception, development, and deployment of the proposed website for the IIT Kanpur community with shared travel preferences. The key stakeholders include:

- **Project Team:** Developers, designers, and project managers responsible for the website's creation and implementation.
- **Community Members:** Users of the website who are part of the IIT Kanpur community and have a shared interest in coordinating group travels to their respective hometowns.
- **Administrators:** Those tasked with managing and maintaining the website, overseeing user activities, and ensuring the platform's smooth operation.
- **Decision-makers and Sponsors:** Individuals responsible for overseeing the project at the institutional level.

1.2.2 Document Overview:

This SRS serves as a comprehensive guide that outlines the specific requirements and expectations for the RouteMate system. The document is structured to provide a clear and detailed understanding of the project's scope, functionalities, and technical specifications. The main sections of the document include:

- **Introduction:** Presents a high-level overview of the project, introducing the purpose, background, and key features.
- **Overall Description:** This encompasses the product overview, functionality, design constraints, and assumptions and dependencies that set the context for the detailed requirements.
- **Specific Requirements:** Details the external interface requirements, functional requirements, and the use case model to define the system's behavior.
- **Other Non-functional Requirements:** Explores performance, safety, security requirements, and software quality attributes that govern the overall system quality..
- **Appendix A: Data Dictionary:** Provides a comprehensive guide to the data elements used in the system, ensuring a shared understanding of the system's data.
- **Appendix B: [Additional Appendices]:** Includes supplementary documents or logs that offer further insights, such as group logs or additional technical documentation.

1.3 Definitions, Acronyms and Abbreviations

API	Application Programming Interface
CSS	Cascading Style Sheets
DB	Database
HTML	Hypertext Markup Language
IIT	Indian Institute of Technology
OTP	One-Time Password
SQL	Structured Query Language
SRS	Software Requirements Specifications
UI	User Interface
UX	User Experience

1.4 Document Conventions

- **General Text** : Font-Arial, Size-11
- **minor subheading (h3)** : Font-Arial, Size-12
- **Subheadings (h2)** : Font-Arial, Size-14, Style-Bold
- **Heading (h1)** : Font-Arial, Size-18, white text inside grey rectangles.
- **Margin** : 1

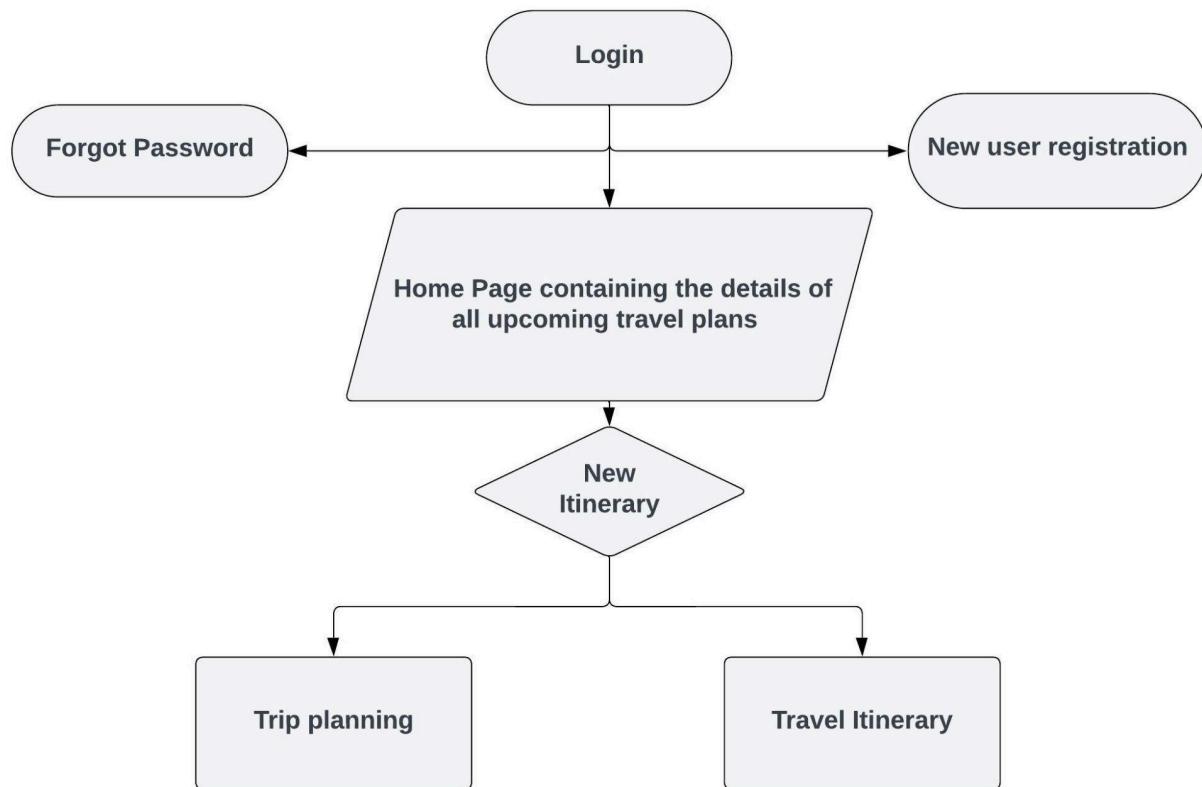
1.5 References and Acknowledgments

1. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.
2. [Visual Paradigm](#) for making the use case diagrams.
3. [Figma](#) for making the user interface designs.
4. [INDIAN RAIL API](#)
5. [Aviation Edge API](#)
6. Lucidchart for creating flow charts.
7. credits for Images : [Freepik.com](#)

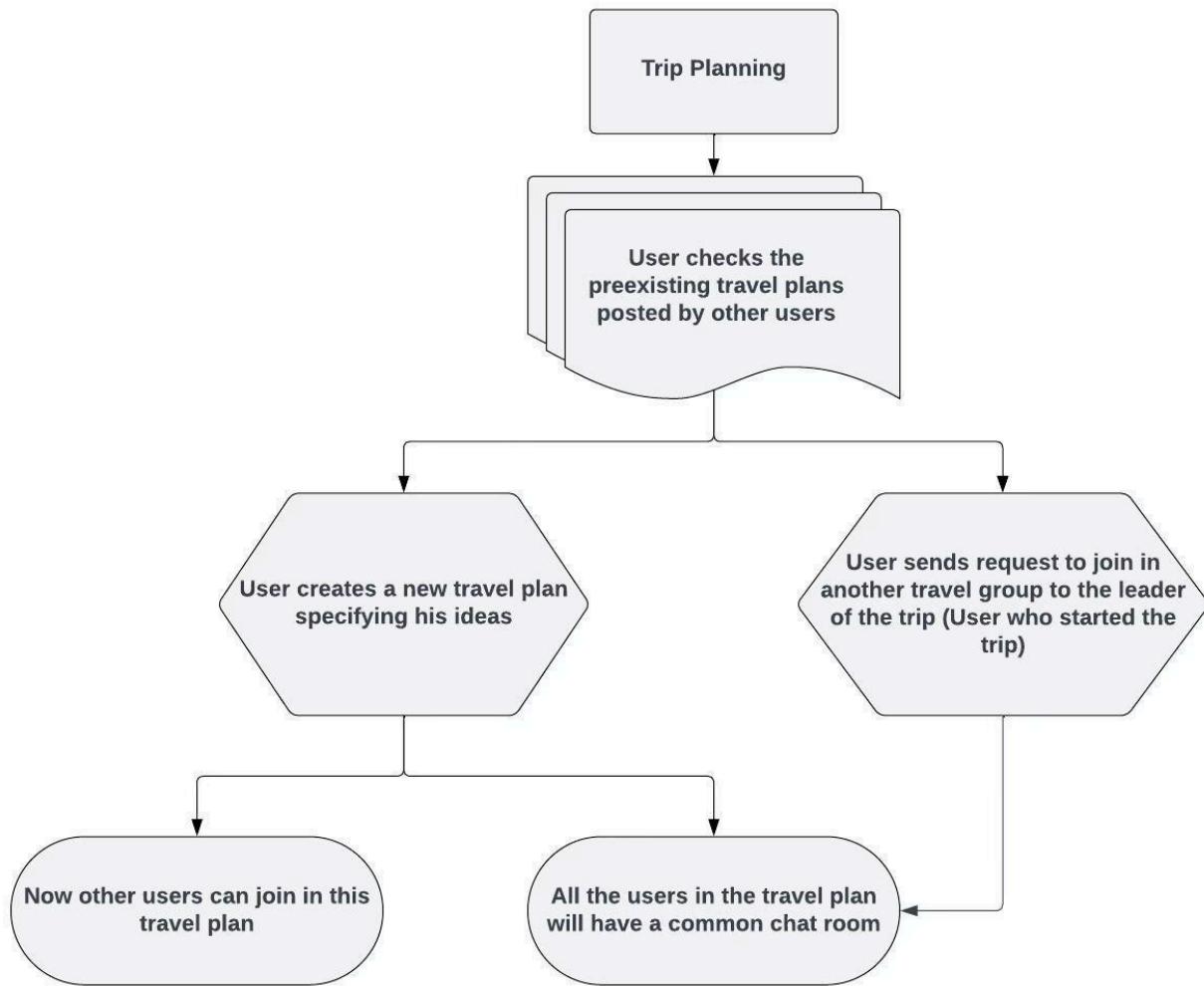
2 Overall Description

2.1 Product Overview

The main objective of RouteMate is to link people with similar travel plans in order to foster community engagement and improve the travel experience. Users enter their travel information into the platform, and RouteMate will match them with other travelers who are going to the same place on the same day. It also offers different options depending on the reason for the trip, such as returning home, on vacation, or to a place of higher education.

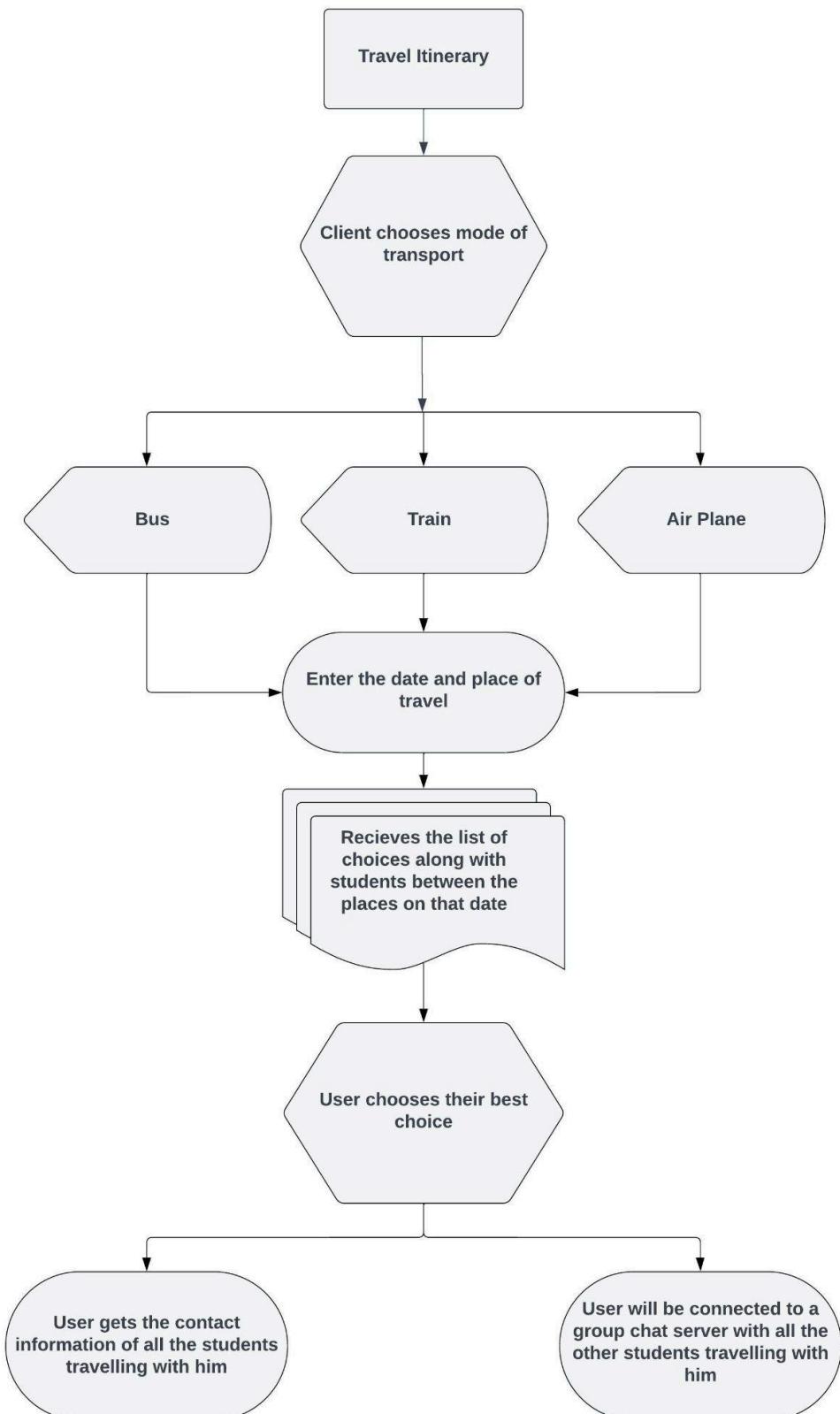


Users can create group trips to tourist destinations with friends and solo trips for commuting to work or home, where a travel companion is sought. Flexible options cater to diverse travel needs.



For planning a trip, users can explore existing travel plans or create new ones. Joining a trip involves requesting access from the plan's creator. For those initiating a new trip, others can express interest. A shared chat room facilitates seamless communication among all trip members, enhancing coordination and fostering camaraderie throughout the planning process.

For individual commutes to either the workplace or residence, users can access a specialized itinerary. Here, users can search for available trains or flights between specified locations on a chosen date. The itinerary provides a comprehensive list of transportation options and fellow users traveling on the same route. Users can seamlessly book tickets based on this information and engage with co-travelers through a dedicated chat room, fostering communication and facilitating coordination for a more efficient travel experience.



2.2 Product Functionality

The software is exclusively designed for a singular user classification with the following features:

2.2.1 User Authentication and Profiles

- Allow students to create profiles with relevant information such as info, preferred travel destinations, and home locations.
- Users have the autonomy to control the visibility of their information (excluding the username) across the application.

2.2.2 Vacation/Tour Planning

- Enable users to create and join events for joint vacations or tours.
- Include features for users to propose and plan trips, specifying details like destination, duration, and activities.

2.2.3 Matching Algorithm

- Implement a matching algorithm to suggest potential travel companions/groups based on travel schedules and locations.
- Allows users to search and connect with other people having similar travel plans.

2.2.4 Group Formation for Home Travel

- Facilitate the creation of groups for students travelling to same or nearby hometowns who share a common mode of transport.
- Include users to find and join existing groups or create new ones based on their travel routes and schedules.

2.2.5 Communication Platform

- Integrate a messaging system to enable users to communicate within the platform and coordinate travel details.
- Provide real-time chat features and discussion forums for group planning and coordination.
- Allows one-to-one messaging system to facilitate private communication while looking for companions.
- Facilitates dedicated group servers for each tour, train, or flight, enabling the dissemination of essential announcements and information related to the journey.

2.2.6 Blog Feature

- Allow users to create and publish blog posts related to their travel experiences, tips, and recommendations.

2.2.7 Flagging Inappropriate Users

- It enables users to report instances of inappropriate behavior, which are subsequently flagged for further inspection and may result in the user being permanently banned from the application.

2.3 Design and Implementation Constraints

2.3.1 User Type Restriction:

- Allows only IITK email-based authentication to ensure only legitimate users access the application.

2.3.2 Cross-Platform Compatibility:

- Accessible on browsers across diverse operating systems and device types, accommodating varied aspect ratios.
- Features a responsive design throughout the application, ensuring a seamless and optimal user experience on all devices.

2.3.3 Supported Languages:

- The primary language throughout the application is English.
- All UTF-8 characters are supported in blogs.
- Some characters are subjected to sanitizing for security reasons.

2.3.4 Memory Requirements:

- The quantity of data generated by users (blogs/groups) and the number of users is constrained by the available storage capacity on the server.

2.3.5 Network Requirements:

- The concurrent accessibility of the application is limited by the server's capacity to handle traffic at any given time.

2.4 Assumptions and Dependencies

2.4.1 Assumptions:

- Accuracy of user provided input:

—RouteMate assumes that information i.e. the train / bus / flight details furnished while querying for the contact details of others, provided by the user regarding the itinerary is correct without verifying it.

- Security assumption – Absence of middleman attacks:
 - RouteMate assumes that all the requests sent from user to server are not subjected to middleman attacks. The platform's integrity and the confidentiality of user data are contingent upon the absence of unauthorized interception or manipulation of data during the communication between the user and the server.

2.4.2 Software Dependencies:

- **Front-End:**

1. HTML
 - For providing the essential structure and template for the website.
2. CSS
 - Enhances the visual presentation of the website.
3. JavaScript – ES6
 - Enriches website interactivity by enabling dynamic content, user engagement etc.
4. ReactJS
 - Empowers the creation of dynamic and responsive web pages.
5. Tailwind CSS
 - Utilized for efficient and responsive styling of web pages.
6. framer-motion
 - Implements captivating animations, elevating the aesthetic appeal.

- **Back-End:**

1. NodeJS
 - Utilized for server-side scripting and server-side logic.
2. ExpressJS
 - Facilitates the setup and management of the server.
3. PassportJS
 - Implements authentication mechanisms
4. Bcrypt
 - Encrypts the user entered password before storing it in database

- **Database:**

1. PostgreSQL
 - Manages and manipulates data, serving as the robust relational database system

- **Hosting:**

1. Heroku
 - Heroku would host the backend services.

- **Network Communication:**

1. **HTTP/HTTPS**

- Facilitates efficient and secure exchange of requests between user and server.

2. **WebSocket using Socket.io**

- It enables real-time, bidirectional communication, enhancing the responsiveness and interactivity of the platform

3. **REST API**

- Supports the development of a scalable and interoperable architecture for communication between different components.

4. **INDIAN RAIL API**

- To fetch detailed information about trains based on user-provided details

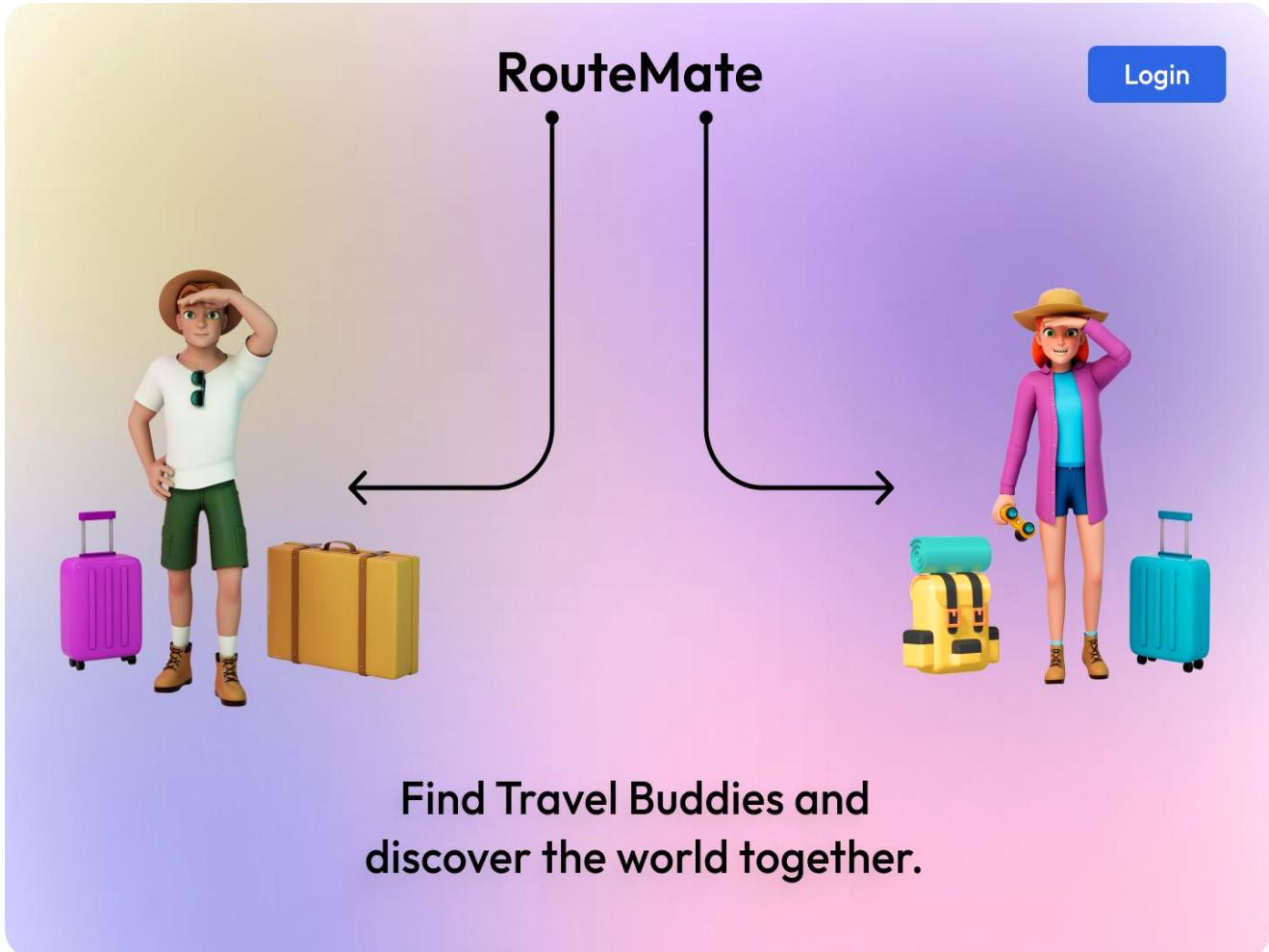
5. **Aviation Edge API**

- To fetch detailed information about flights based on user-provided details

3 Specific Requirements

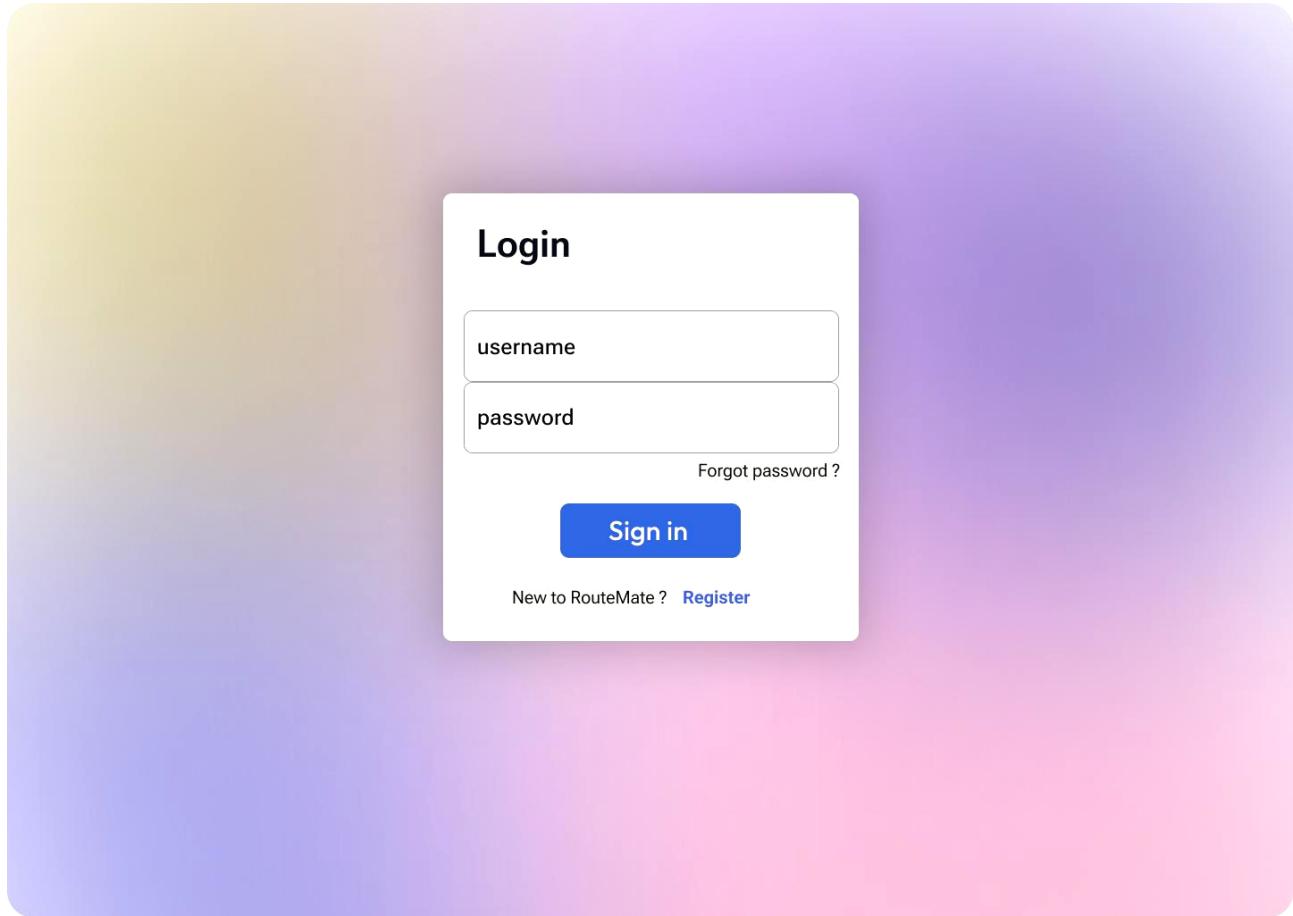
3.1 External Interface Requirements

3.1.1 User Interfaces



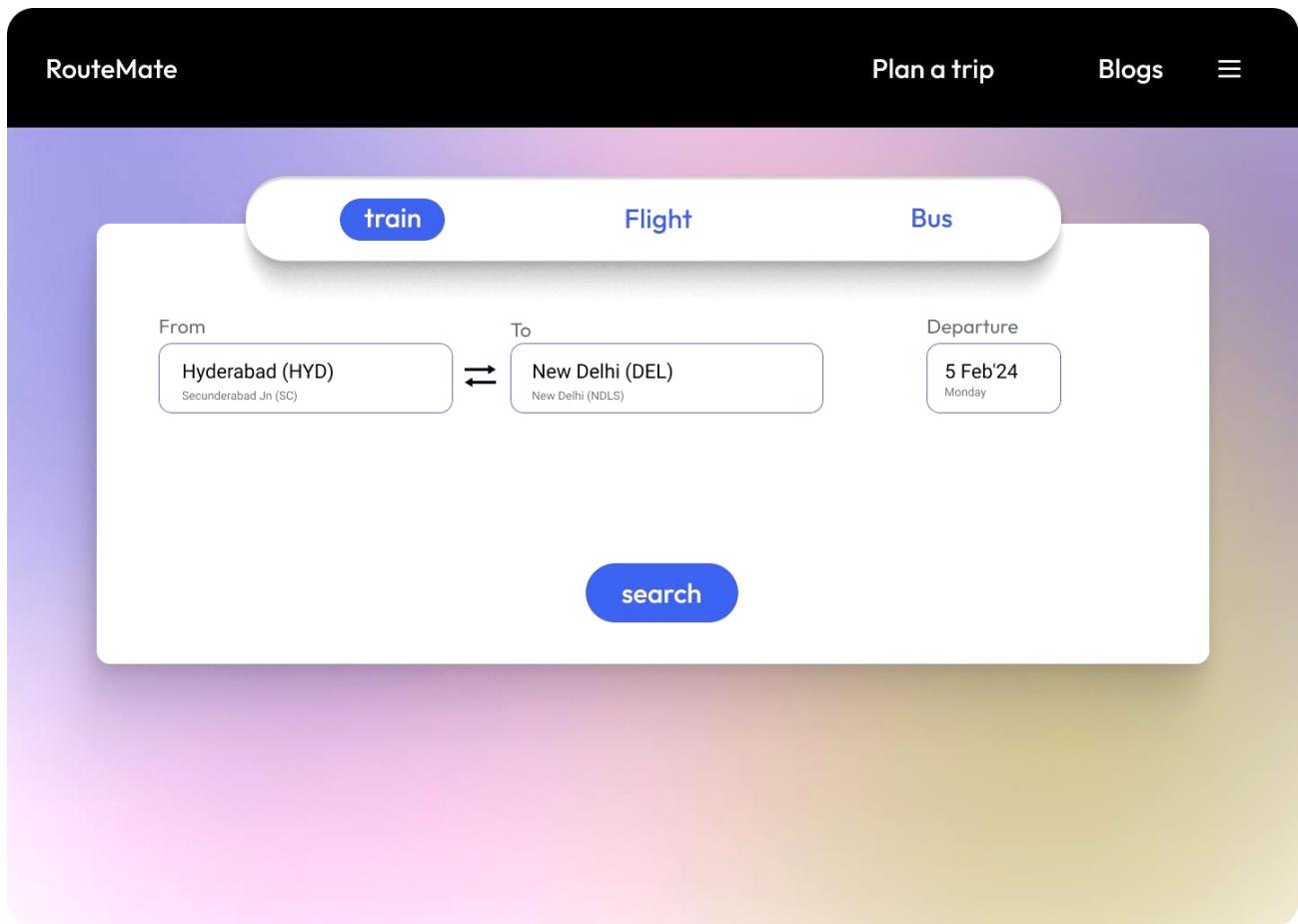
Starting page

The starting page of our website serves as a gateway, featuring our mission to connect students traveling to the same destination. It prominently showcases our **platform's motto**, along with a convenient **login option**.



Login page

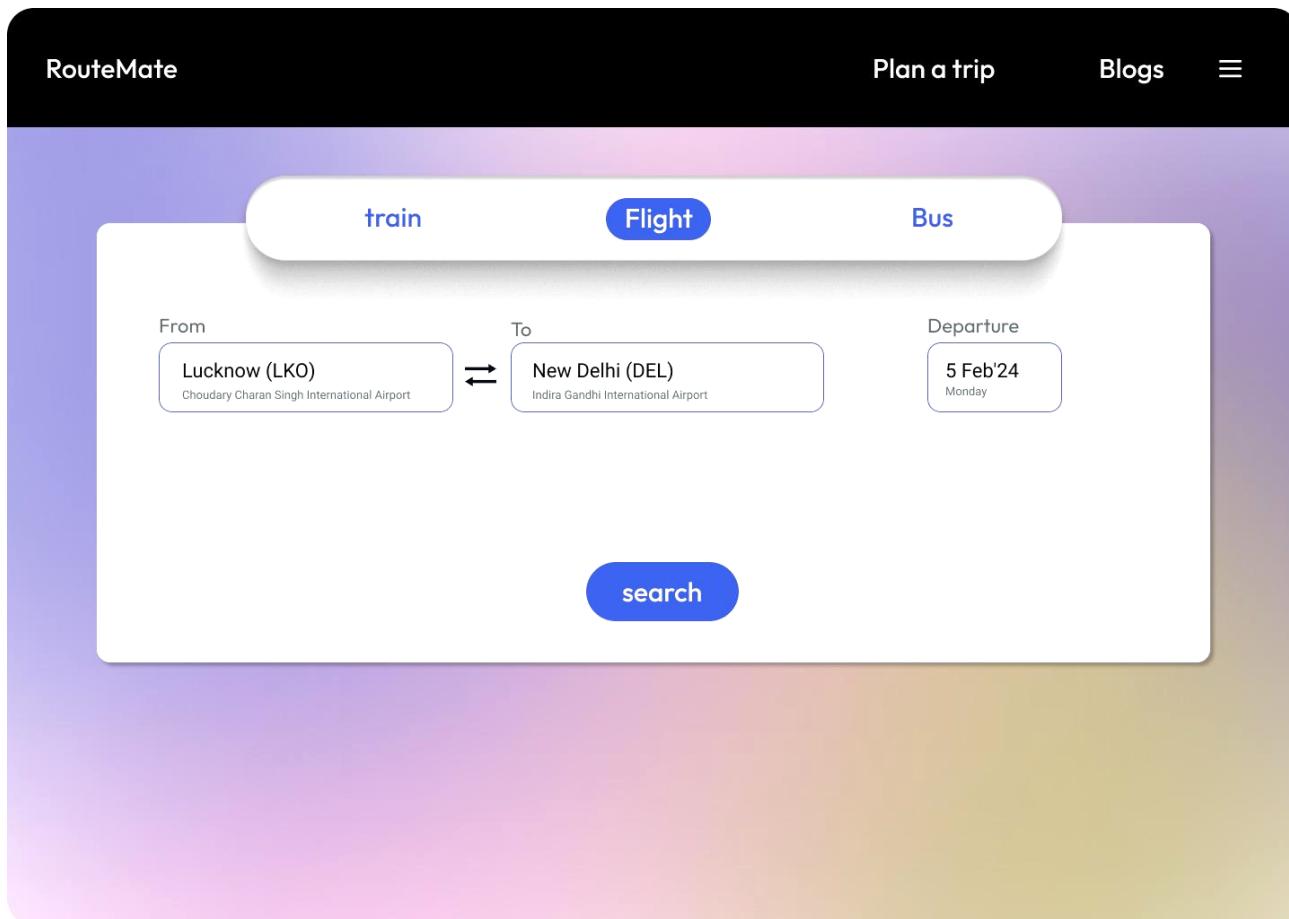
This section contains **login**, **register** options and in the event that users forget their existing password, they also have **forgot password** option to reset their password (through email verification)



Home page:

This is the main page of our website. It has two sections:

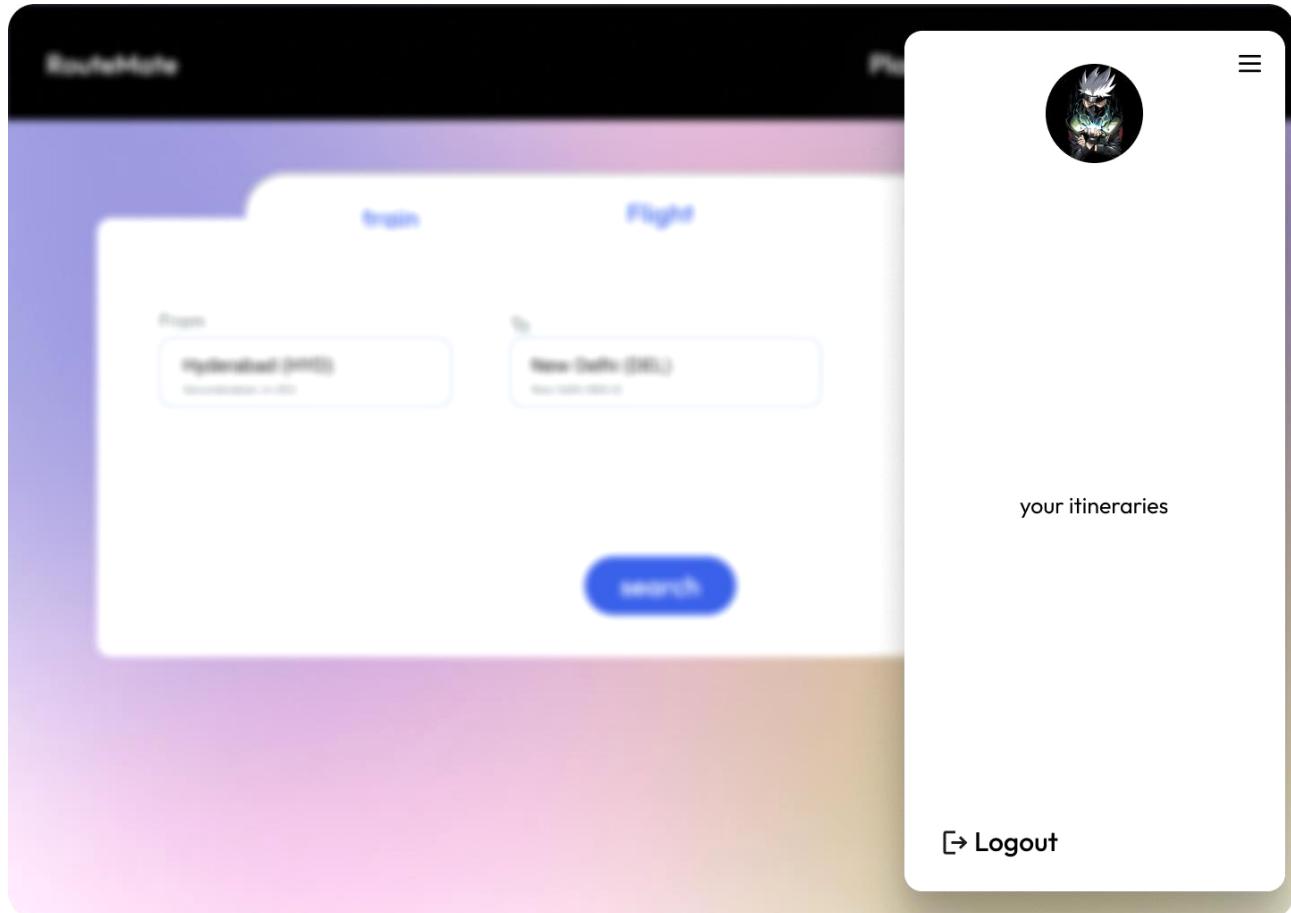
- Header at the top, which provides options like **Plan a trip**, **Blogs** and menu toggle button
- **Search box** in the middle, when filled out with enough information, allows us to look up itineraries linked to that information.



Search box :

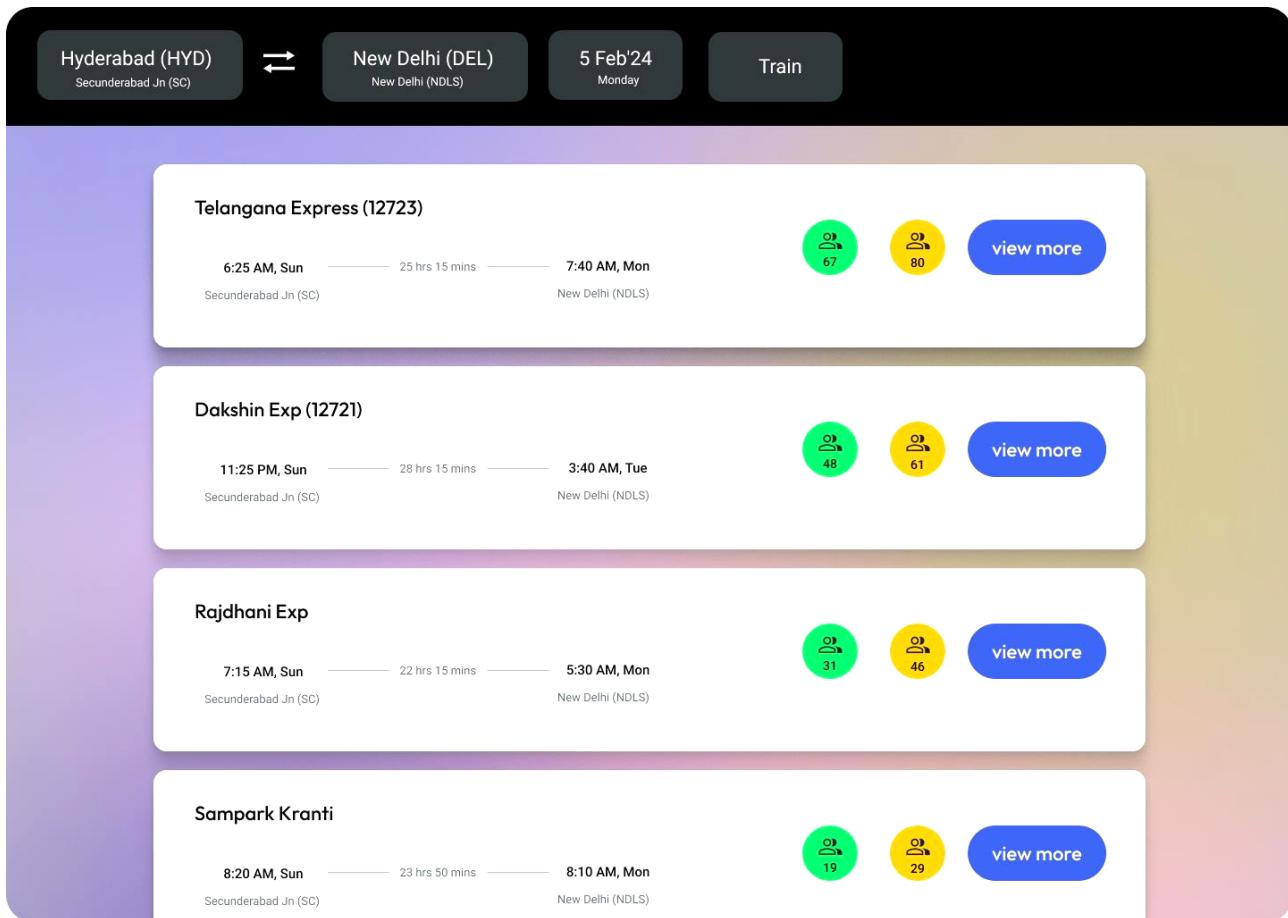
This box contains three sections :

- **Travel mode**
- **From and To places**
- **Departure date**



Menu :

After selecting the menu button in the header, the menu appears as above. It includes **user trips**, **itineraries**, **profile preferences**, **settings** and **logout** options.

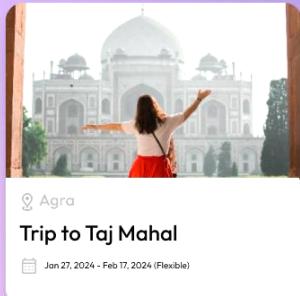


Trains list :

After you enter itinerary details in the search box, this page appears. All the trains that match your criteria are displayed here.

- Number of IITK students who have **confirmed** their tickets (Green)
- Number of IITK students **planning** to travel in that train (Yellow)
- Basic information like the name of the train, departure time, arrival time, and journey time.

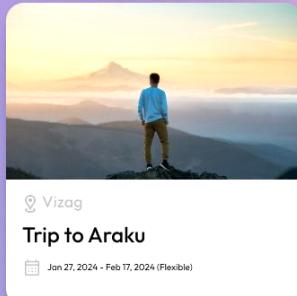
Find a Travel Buddy



📍 Agra

Trip to Taj Mahal

📅 Jan 27, 2024 - Feb 17, 2024 (Flexible)



📍 Vizag

Trip to Araku

📅 Jan 27, 2024 - Feb 17, 2024 (Flexible)



📍 Nagpur

Field Trip

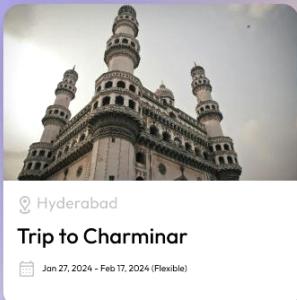
📅 Jan 27, 2024 - Feb 17, 2024 (Flexible)



📍 Delhi

Trip to Delhi

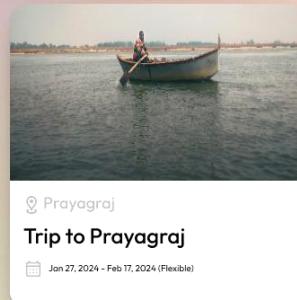
📅 Jan 27, 2024 - Feb 17, 2024 (Flexible)



📍 Hyderabad

Trip to Charminar

📅 Jan 27, 2024 - Feb 17, 2024 (Flexible)



📍 Prayagraj

Trip to Prayagraj

📅 Jan 27, 2024 - Feb 17, 2024 (Flexible)

On this page, you'll **discover travel plans** created by fellow IITK students

Where to?

Let us know your preferences

Tentative dates

Start date	End date
25th March 2024	27th March 2024

Approximate Budget

Description...

We are a group of 10 friends who are planning to visit Ayodhya for our upcoming winter holidays.

dates are flexible
please join group before 1st march so we can book train tickets

Add places to visit

- Agra fort
- Taj Mahal
- Mughal heritage walk
- Taj nature walk
- Fatehpur sikri
- Mehtab Bagh

Submit

Plan a trip :

This section will assist you in getting started with trip planning. The user will need to supply some information, such as the destination, flexible travel dates, an estimated budget, and the locations they wish to see during their journey.

After submitting user trip details, our software creates a trip and adds it to the discovery section and to user trips.

Trip to Agra

Kanpur , Uttar pradesh

Agra , Uttar pradesh

25th March 2024 - 27th March 2024 (Flexible)

Trip details

We are a group of 10 friends who are planning to visit Ayodhya for our upcoming winter holidays.

dates are flexible
please join group before 1st march so we can book train tickets

Places to Visit

- Agra fort
- Taj Mahal
- Mehtab Bagh
- Mughal Heritage Walk
- Taj Nature Walk
- Fatehpur Sikri

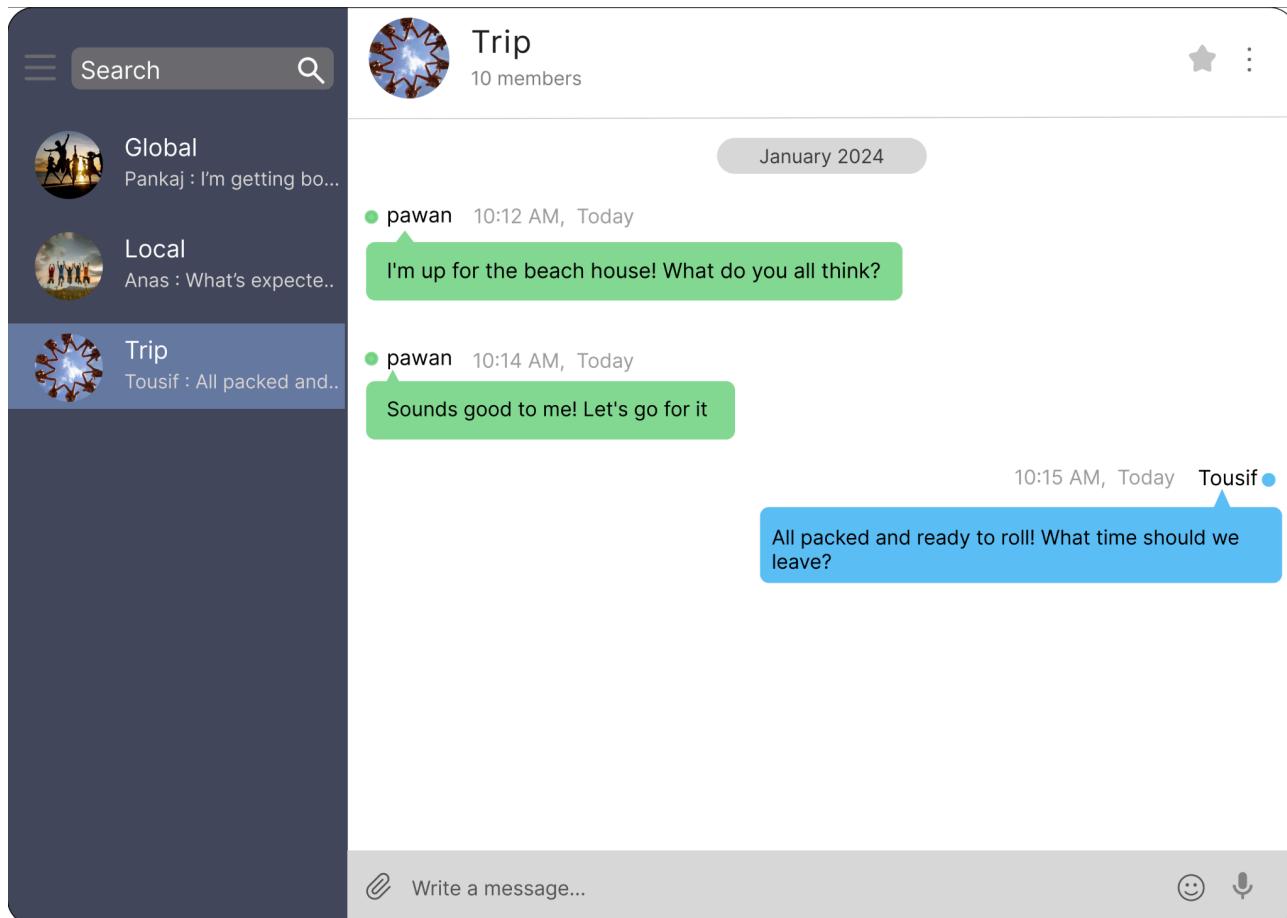
Budget Around 2500 rupees

Trip created by



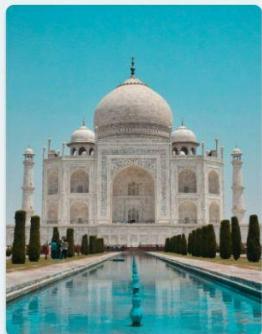
Join Trip

This is how the user-created trip appears. Any user who is willing to be part of this trip clicks the "Join Trip" button, and the trip creator receives a join request; if accepted, the new user joins the group and has the ability to amend the plan.



Chat Room

The Chat Room invites travelers to connect, share experiences, and seek advice. All messages are visible to everyone. Users can engage or exit at any time for a collaborative journey. This page contains all the messages posted by various users travelling along with you.



Day 1: The Majestic Taj Mahal

No visit to Agra is complete without witnessing the ethereal beauty of the Taj Mahal. Built by the Mughal Emperor Shah Jahan in memory of his beloved wife Mumtaz Mahal, this UNESCO World Heritage Site is an architectural masterpiece. As the first light of dawn bathes the ivory-white mausoleum, the marble comes alive with hues of pink and gold, creating a mesmerizing spectacle.

Pro Tip: To avoid the crowds, plan an early morning visit to witness the Taj Mahal at sunrise. Don't forget to capture the magical moments as the sun-kissed dome reflects in the pristine waters of the Yamuna River.

by Anya

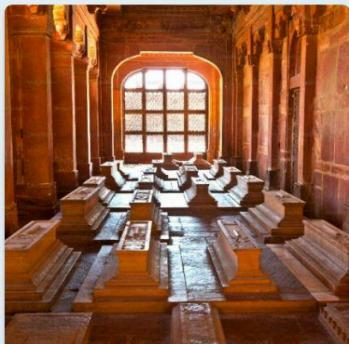
by Swayam

Day 2: A Walk Through Agra Fort

The Agra Fort, another UNESCO World Heritage Site, stands tall as a testament to the grandeur of the Mughal Empire. Built by Emperor Akbar, this red sandstone fortress houses palaces, mosques, and audience halls that provide a glimpse into the opulent lifestyle of the Mughal rulers.

A photograph showing the massive red sandstone walls and towers of the Agra Fort. A central arched gateway is visible, with several people walking in front of it under a clear blue sky.

Suggestion: Engage a local guide to unravel the stories behind the intricately designed Diwan-i-Khas, Jahangir Mahal, and the Sheesh Mahal within the Agra Fort. The historical anecdotes and architectural insights will enrich your experience.



Day 3: Exploring Fatehpur Sikri

A short drive from Agra takes you to Fatehpur Sikri, an abandoned city that served as the capital of the Mughal Empire for a short period. The well-preserved structures, including the Buland Darwaza and the Jama Masjid, showcase the architectural prowess of the Mughals.

Recommendation: Plan a day trip to Fatehpur Sikri to explore its unique blend of Mughal and Persian architecture. The serene ambiance and historical significance make it a compelling addition to your Agra itinerary.

by pankaj

3.1.2 Hardware Interfaces

Since the web-application must run over the Internet, all the hardware will be required to connect to the internet. Any operating system (Windows, Linux, MacOS, Android, etc.) with a proper internet connection will serve the purpose.

3.1.3 Software Interfaces

Any operating system with a modern mainstream browser, such as Brave, Edge, Firefox, Google Chrome, etc. is required. On the server side, databases are used to store the data and the web-application sends requests/instructions to the server via API calls. The server also calls third-party API(s) for certain functions.

3.2 Functional Requirements

3.2.1 User registration

- The system allows users to create their profiles.
- Collects basic information from users like Name, IITK Roll number, IITK Email Id and Password.
- The user has the option to set additional personal information on their profile.

3.2.2 User login

- The system authenticates users based on their credentials (IITK Roll number and IITK Email Id).
- Passwords are securely stored using encryption.
- The system prevents common attacks like SQL injection and cross-site scripting (XSS).

3.2.3 Forgot Password

- A “Forgot Password” link is available on the login page.
- Generates a secure, time-limited token for password reset.
- Deliver the password reset token to the user via email.

3.2.4 Automatic log-out

- The system tracks user sessions to monitor their activity on the website which starts after the most recent login.
- The system maintains the inactivity period using a token and timestamps of user logins and logouts.
- After a certain inactivity period (say 30 days) the user will be logged out of the system.

3.2.5 Show/hide Profile

- At the time of registration it is optional for the user to add their contact info.
- Users have the ability to change the visibility of their contact info. at any point of time.
- Users can add their contact details on the website at any point of time if they have not done so already.

3.2.6 Chat-room Privacy

- Chat-rooms dedicated to trips planned by users can be set to private by the Admin of the group (the user who created the group).
- Additional members can be added only via invites.

3.2.7 Maintain user profile

- The system enables users to upload and set a profile picture to personalize their profile.
- Users can choose from a set of default avatars provided by the system.
- Users can upload a custom profile picture from their device.
- The system allows users to hide their profile pic from some users.
- Users are able to set different privacy levels for their profile picture, including public,friends only , private.

3.2.8 Users can delete their posts

- The system allows users to delete their post/blog at any time.
- The system allows users to delete their comments to any posts/blogs.
- The system prompts users for confirmation before permanently deleting a post or comment to prevent accidental deletion.
- Users have the option to delete multiple posts or comments simultaneously using a bulk deletion feature.

3.2.8 Email Authentication

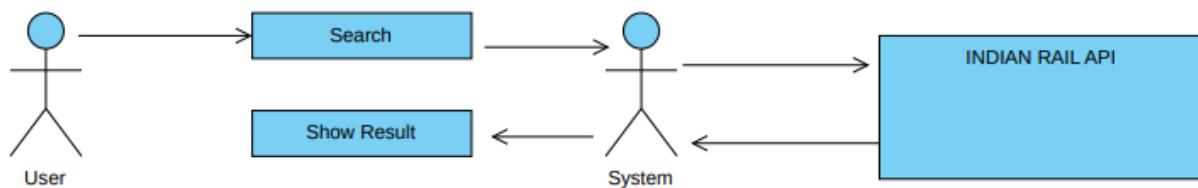
- New users must provide a valid email address during the registration process.
- Upon registration, the system sends a verification email to the provided email address.
- Users are required to click on a unique verification link within the email to confirm their registration.
- The system checks that the entered email address follows a valid format (e.g., user@iitk.ac.in); if not, it shows error messages for users.
- System ensures that each email address used for registration is unique within the system.

3.2.9 Checking Trains/flights

- Users must provide details about their travel itinerary to the system which is sent to the server
- Server gets details of all the trains/trips that meet the requirements of the user and sends it to the user.

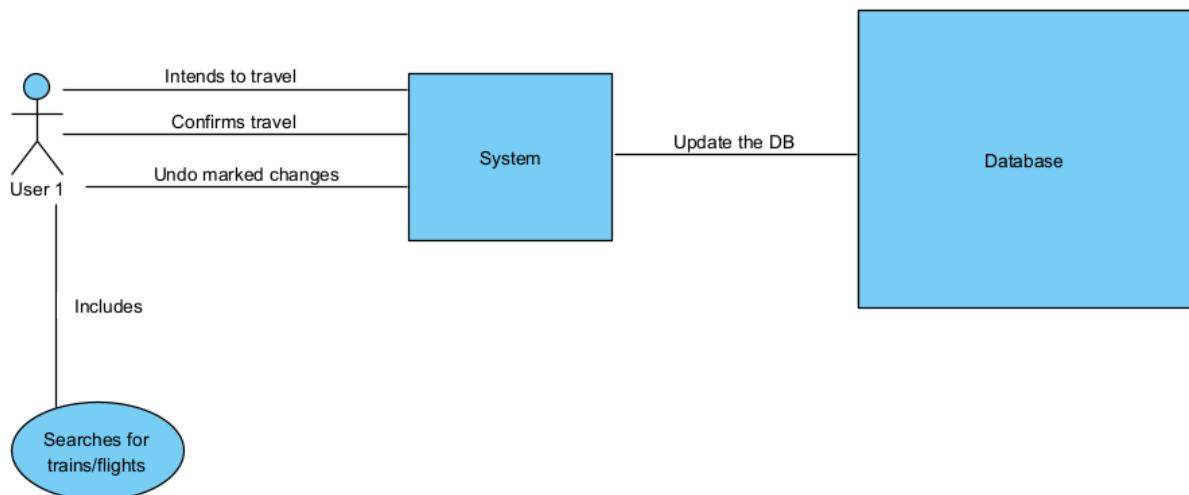
3.3 Use Case Model

3.3.1 SEARCH : Use Case - 1



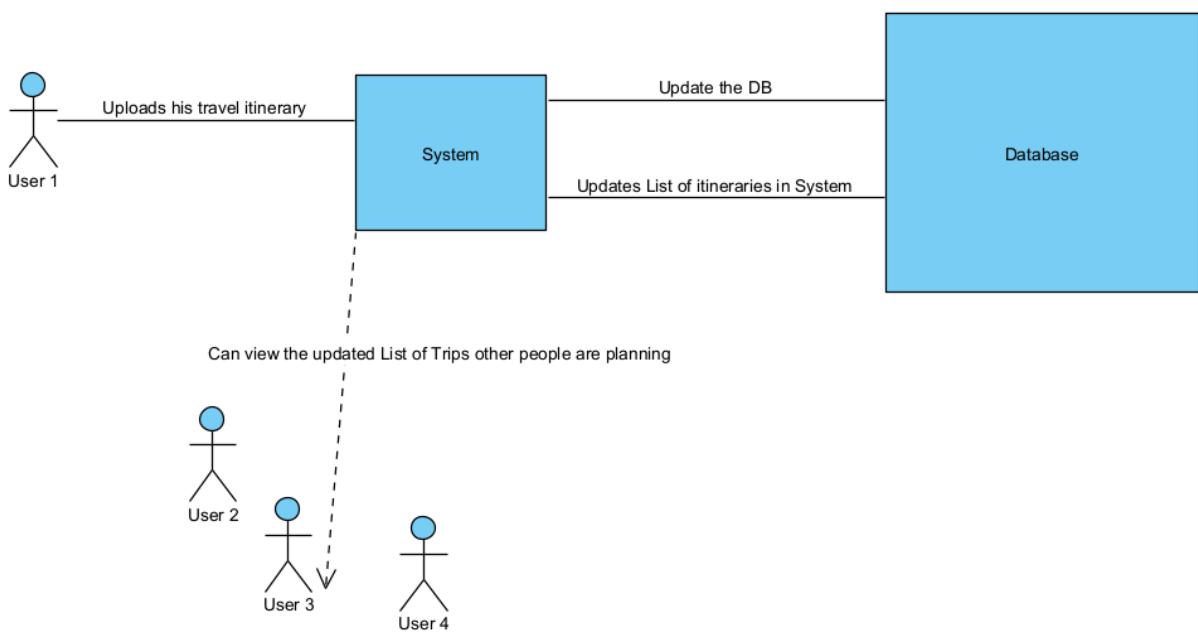
Author	Himanshu Shekhar
purpose	To enable the user to search for various trains/flights based on their preferences.
Requirements Traceability	Search interface, INDIAN RAIL API, Aviation Edge API
Priority	High, as it constitutes the foundational backbone of our application.
Preconditions	Users must be logged in to the web-application.
Post conditions	Ideally, the user should get a list of all the trains/flights with relevant information corresponding to their search preferences.
Actors	User, System
Exceptions	The API fails to work
Includes	Functional requirement 3.2.2

3.3.2 Confirm Booking : Use Case -2



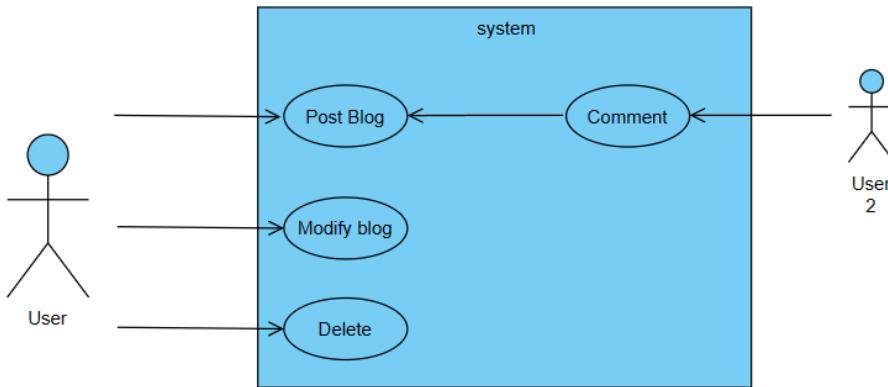
Author	Swayamsidh Pradhan
purpose	The user selects a train or flight and can either confirm their travel on the chosen option or express their intention to travel on it. Users also have an option to remove their marked plans.
Requirements Traceability	Search Interface, Database
Priority	Due to the critical nature of this feature for the proper functioning of our web application, its priority is designated as HIGH.
Preconditions	Users should be logged in to the web-application and can view various trains/flights he wishes to travel on.
Post conditions	User entered information is stored in the database. Also, the count of the number of people confirmed/intending to travel on a train/flight on a certain day gets updated.
Actors	Users, System, Database.
Exceptions	None
Includes	Use Case - 1

3.3.3 TRAVEL ITINERARY : Use Case - 3



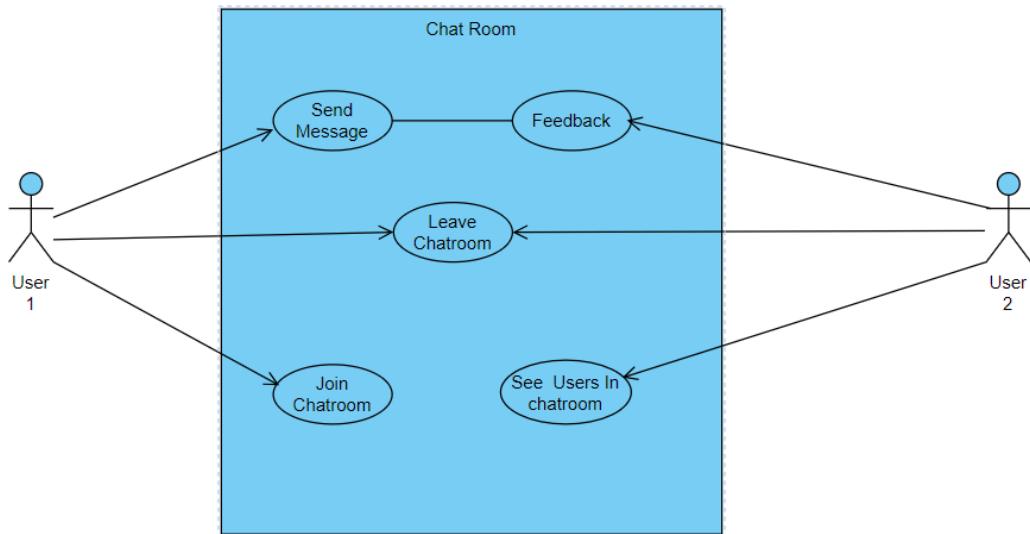
Author	Swayamsidh Pradhan
purpose	If a user intends to organize a trip and wishes to invite others to join, they can upload their itinerary on the website.
Requirements Traceability	User Interface, Database
Priority	This is one of the key features offered by the website. Hence its priority is HIGH.
Preconditions	Users must be logged into the web application and should have a preliminary idea about the trip, including tentative dates and locations to visit.
Post conditions	The itinerary uploaded by the user is visible to other users who can access the website.
Actors	Users, System, Database.
Exceptions	None
Includes	None

3.3.4 Manage and View Blogs : Use Case - 4



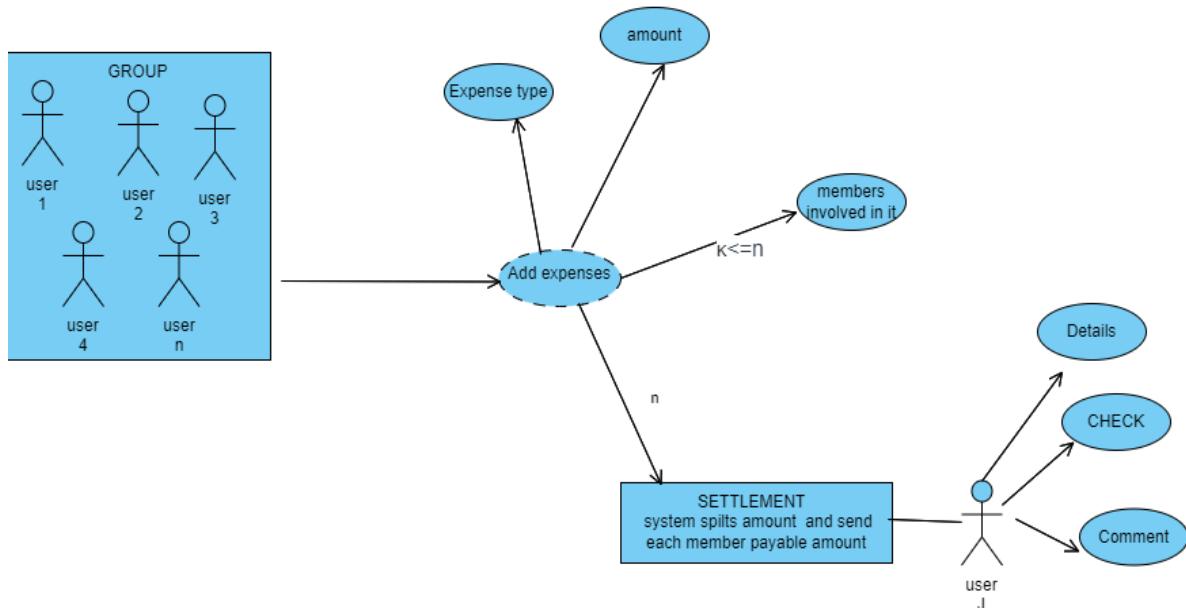
Author	Anya Rajan
purpose	The blog feature enables users to create, publish, and explore travel-related content within the travel app. Admins have moderation capabilities to ensure content adherence to site policies.
Requirements Traceability	The user must be logged into the travel site.
Priority	The priority of the use case is Medium as if this function fails the rest of the system can still operate independently without any error.
Preconditions	To enter the blog, users must have traveled to the specified place.
Post conditions	The user's published blog is visible to other registered users and can be edited in the future by the author.
Actors	<ul style="list-style-type: none"> • User: Travel app users who create and view travel blogs. • Admin: Administrators responsible for moderating and managing the blog platform
Exceptions	None
Includes	None

3.3.5 Chat room : Use Case - 5



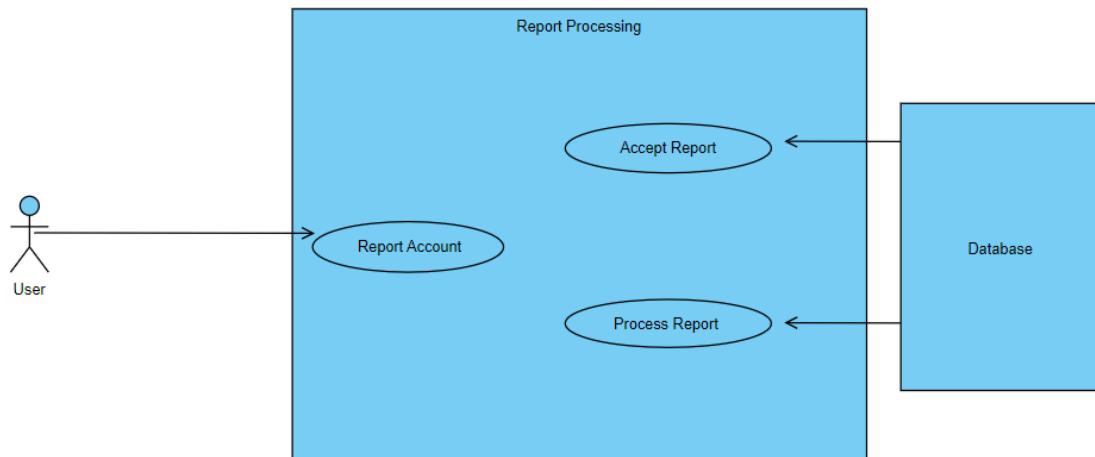
Author	Anya Rajan
purpose	This use case is triggered when two or more people connect to go on a journey together and choose to form a chat room for the same on the site.
Requirements Traceability	The users involved must be willing to travel together in the journey.
Priority	The priority of the use case is Medium as if this function fails the rest of the system can still operate independently without any error.
Preconditions	The users involved in the chat must be connected for a journey
Post conditions	The users will have a group chat for communicating various aspects of the journey with each other.
Actors	Users, System
Exceptions	None
Includes	None

3.3.6 GROUP EXPENSE MANAGER : Use Case - 6



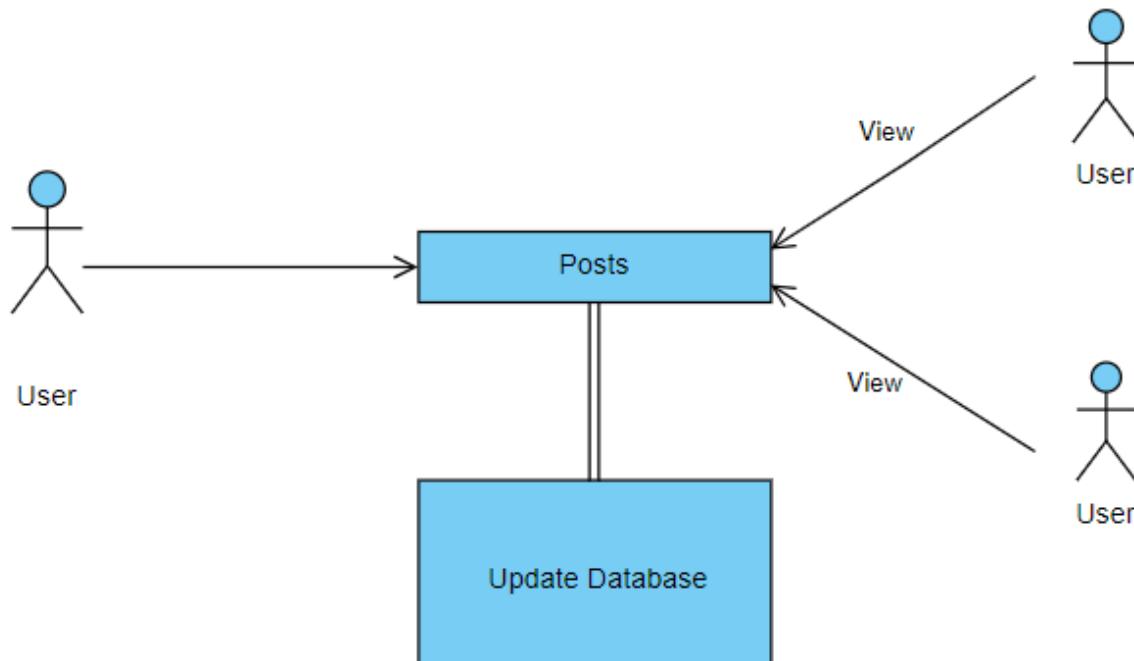
Author	Daksh Kumar Singh
purpose	The Group Expense Manager allows users to efficiently split all the expenses of their journey equally and maintain a comprehensive record of their expenses.
Requirements Traceability	For security and accountability, all users participating in the expense splitting must be logged into the system. Additionally, users must identify themselves as a group and register the group in the system.
Priority	Low
Preconditions	Users must log in to the system and should identify themselves as a group and register the group in the system.
Post conditions	All members can see the splitted expenses and pay their dues.
Actors	Members of group
Exceptions	None
Includes	None

3.3.7 REPORT : Use Case - 7



Author	Daksh Kumar Singh
purpose	Users can report other users .
Requirements Traceability	Users should be logged in and profiles should be registered in the system.
Priority	Medium
Preconditions	Users should be logged in.
Post conditions	After processing the report request, if the minimum no. of reports reached for a reported user system sends a warning to the user to maintain his/her conduct.
Actors	Users, database
Exceptions	None
Includes	3.3.4

3.3.8 CABS : Use Case - 8



Author	Himanshu Shekhar
purpose	To enable the user to post/view other users going to the same airport/station
Requirements Traceability	Cab share interface, Chat
Priority	Low
Preconditions	Users must be logged in to the web-application.
Post conditions	Ideally, users should receive a comprehensive list of posts from others.
Actors	User, System
Exceptions	None
Includes	Functional requirement 3.2.2

4 Other Non-functional Requirements

4.1 Performance Requirements

- The load time for the website must not exceed 5s.
- The software must be able to handle at least 3000 concurrent users during peak hours (considering the strength of IITK is around 10,000).
- The website must work as expected in all modern browsers irrespective of operating system or the type of device.

Possible Overhead:

APIs: The Indian-rail-api must be able to handle the above traffic.

Database: The database used must be able to handle the traffic without crashing.

4.2 Safety and Security Requirements

- Only users with IITK email must be able to register in the software to avoid spam accounts.
- The email authenticity is verified using an OTP.
- Authentication of the User must be sufficiently secure.
- The software must not accept weak passwords.
- Multiple failed login attempts from the same IP address must not be allowed.
- Passwords are never stored in the database, only hashes or similar data related to the password are stored.
- All the authentication cookies like session cookies MUST be HTTP only to avoid possible session hijacking when a Cross Site Scripting bug is found.
- “Forget Password” functionality shouldn’t leak any information from the database nor the session cookies.
- Session cookies must expire after a fixed time (30 days).
- The Back end must maintain a backup to counter any potential data loss.
- Any kind of brute force from a potential attacker must not crash the server or database.

4.3 Software Quality Attributes

1. Functionality

- The software must have the expected functionality , such as Authentication, Chat Servers and showing relevant data about the travel itineraries.

2. Reliability

- The website must be reliable, ensuring minimal downtime and consistent performance.
- Users should be able to access the platform without disruptions, enhancing their trust in the service.

3. Usability

- It should be easy for users to navigate the platform, reducing the learning curve and improving overall user satisfaction.

4. Efficiency

- The software should operate efficiently, minimizing response times and resource usage.
- This ensures a smooth and responsive user experience, even during peak usage periods.

5. Maintainability

- The codebase should be well-organized and documented, facilitating ease of maintenance for developers.
- Regular updates and improvements can be implemented more efficiently with a maintainable codebase.

6. Portability

- The software should be designed to work seamlessly across different devices and platforms, providing users with flexibility and accessibility.

7. Scalability

- The ability to scale the software infrastructure to accommodate an increasing number of users or growing data is crucial.
- This ensures that the system can handle increased load and demand without sacrificing performance.

8. Security

- Robust security measures should be in place to protect user data, ensuring confidentiality, integrity, and availability.
- This includes encryption, secure transmission, and protection against common cyber threats.

Appendix A – Data Dictionary

- User Class

Element Name	User
Description	Corresponds to the person who is using the software. Each user registers to the website using IITK email and password which is used later
Attributes	<ol style="list-style-type: none"> 1. Email : string 2. username : string 3. password : string
Operations	<ul style="list-style-type: none"> • Login() : called when user wants to login • Register() : called when user wants to create new account

- Train Class

Element Name	Train
Description	The train which is displayed based on the constraints provided by the user
Attributes	<ol style="list-style-type: none"> 1. Train Number : string 2. Origin : string 3. Destination : string 4. Travel time : string 5. Confirmed Users : Integer 6. Not confirmed users : Integer
Operations	<ul style="list-style-type: none"> • Confirm() : Called when user confirms to travel in that particular train • waiting() : called when user is not sure to travel in that train • ChatRoom() : Redirects to the chat room of train

- **Vacation Class**

Element Name	Vacation	
Description	Corresponds to the trip plan posted by a user to which other users can join.	
Attributes	1. Place : string 2. Trip duration : Integer 3. Tentative budget : Integer	
Operations	<ul style="list-style-type: none"> • sendRequest() : called when an user wants to join a pre-existing plan • AcceptRequest() : called when user accepts another user's request • ChatRoom() : Redirects to the chat room of trip 	

- **ChatRoom Class**

Element Name	Chat Room	
Description	Corresponds to the chat room containing all the users in a particular train/trip.	
Attributes	1. UserName : string 2. chats : string 3. users : objects	
Operations	<ul style="list-style-type: none"> • postmessage() : called when user posts a message • exitChat() : called when user wishes to exit a chat room • joinRoom() : called when user confirms their train/trip 	

- **Blog Class**

Element Name	blogs
Description	Corresponds to the blogs posted by the users after their trip to express their views.
Attributes	<ol style="list-style-type: none">1. Title : string2. Photos : URL3. body : string4. author : object
Operations	<ul style="list-style-type: none">• postBlog() : called when a user posts blogs to after his/her trip

Appendix B - Group Log

SNo	Date	Agenda	Duration	Members Attended
1	07-01-2024	Discussing Project Ideas	90 min	10
2	09-01-2024	Finalizing Project idea	60 min	10
3	14-01-2024	Discussed the format and content to write in the SRS document. Distributed work among the team members	120 min	10
4	17-01-2024	Met TA and asked for guidelines to complete the document	60 min	10
5	21-01-2024	Prepared first draft of the document and proposed changes to the work	120 min	10
6	23-01-2024	Prepared second draft without UI images and discussed about the functional requirements that has to be displayed in the interface	120 min	10
7	25-01-2024	Added the UI images and prepared the final draft	90 min	10
8	26-01-2024	Made a few changes in the format of the document to make it visually appealing	120 min	10