GETSETGO – SMART WEB-BASED TRAVEL MANAGEMENT SYSTEM

Thakur Shyamnarayan Degree College
Ms.Preeti Pandey, Sakshi Jaiswaal, Shruti Gupta
Assistant professor, Undergraduate Student, Undergraduate Student
Department of Information Technology
University of Mumbai, Mumbai, India

Abstract : Travel planning for long-distance journeys often involves difficulties such as finding reliable drivers, managing routes, and ensuring safety during travel. To address these challenges, this project introduces GetSetGo, a smart web-based travel management system. The platform allows users to book verified drivers and vehicles, customize travel routes, track journeys through real-time GPS, and communicate using a multilingual chat feature. Secure INR-based digital payments are also supported within the system. The application is developed using HTML, CSS, and JavaScript for the frontend, while Node.js and Express.js manage backend operations with MongoDB for database storage and Firebase for real-time services. The system aims to improve travel convenience, transparency, and safety.

IndexTerms - **Smart Travel System, Web Application, GPS Tracking, Travel Booking, Node.js, MongoDB**

I. INTRODUCTION

Planning long-distance or cross-state travel can often be complex due to difficulties in finding trustworthy drivers, managing travel routes, and ensuring passenger safety during the journey. Many traditional travel booking systems offer limited features and do not provide real-time updates or flexible travel options. Travelers may also face challenges in communicating effectively with drivers or tracking their journey progress. To overcome these challenges, this project introduces GetSetGo, a web-based travel management system that simplifies the process of booking and managing travel services. The platform enables users to book verified drivers and vehicles, customize their travel routes, and track their journey using real-time GPS tracking. Additionally, the system includes a multilingual chat feature that allows seamless communication between travelers and drivers. By integrating these services into a single platform, the system provides a safer, more convenient, and efficient travel experience for users.

II. PURPOSE

The main purpose of the GetSetGo system is to provide a reliable and user-friendly platform for managing long-distance travel bookings. The system aims to reduce the complexity involved in planning travel by integrating multiple features such as driver booking, route customization, real-time tracking, and communication within a single application. Another objective of the system is to enhance safety and transparency in travel services by allowing users to book verified drivers and monitor their journey through GPS tracking. The platform also simplifies the booking and payment process by providing secure digital payment options. Overall, the system is designed to improve the travel experience while ensuring efficiency, reliability, and ease of use.

III. SCOPE

The scope of the GetSetGo project focuses on developing a web-based platform that supports efficient management of long-distance travel bookings. The system allows users to register on the platform, browse available drivers and vehicles, select travel routes, and confirm bookings through a simple interface. The application also supports real-time GPS tracking, enabling users to monitor their travel progress and ensure safety during the journey. Communication between drivers and travelers is supported through an in-app multilingual chat feature, which improves coordination and reduces misunderstandings. All user details, booking information, and travel records are stored in a MongoDB database, ensuring organized data management and reliable system performance.

IV.EXISTING ALGORITHM

1. User Authentication and Verification

The system uses a user authentication algorithm to verify registered users during login. User credentials such as email and

password are validated against records stored in the MongoDB database. Passwords are securely stored using encryption techniques to prevent unauthorized access and ensure data security within the system.

2. Route Selection and Customization

The route management process allows users to select their starting location, destination, and preferred travel route. The system processes the route details provided by the user and stores them in the database. This feature helps travelers plan their journeys more efficiently and according to their preferences.

3. Booking Management Algorithm

The booking management algorithm manages the process of requesting and confirming travel bookings. When a user selects a driver and vehicle, the system checks availability and records the booking information in the database. This algorithm ensures that booking records are maintained properly and prevents duplicate reservations.

4. Travel Scheduling Algorithm

The system includes a scheduling mechanism that allows users to pre-book their rides by selecting the date and time of travel. The algorithm processes the selected schedule and stores it in the booking database so that drivers and travelers can manage travel plans effectively.

5. Communication and Chat Processing

The system supports communication between drivers and travelers through a multilingual in-app chat feature. Messages are processed and stored using backend services, allowing users to communicate easily regarding travel details, route changes, or other queries.

6. Database Management

The system uses MongoDB as the primary database to store user details, booking information, route preferences, and travel schedules. Database queries are handled through backend services to ensure accurate storage and retrieval of information for efficient system performance.
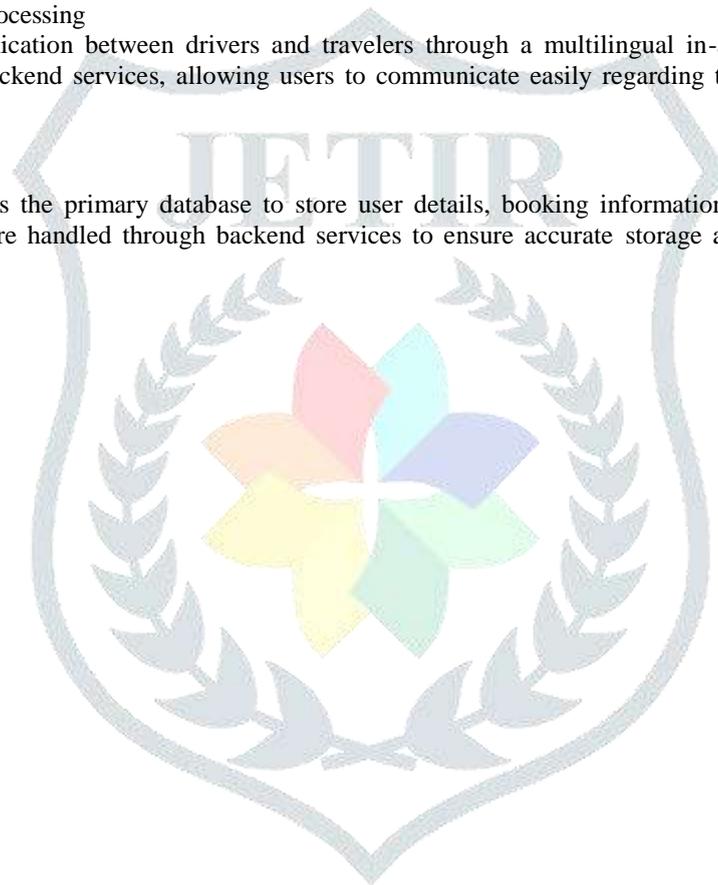
IV. FLOW OF PROJECT
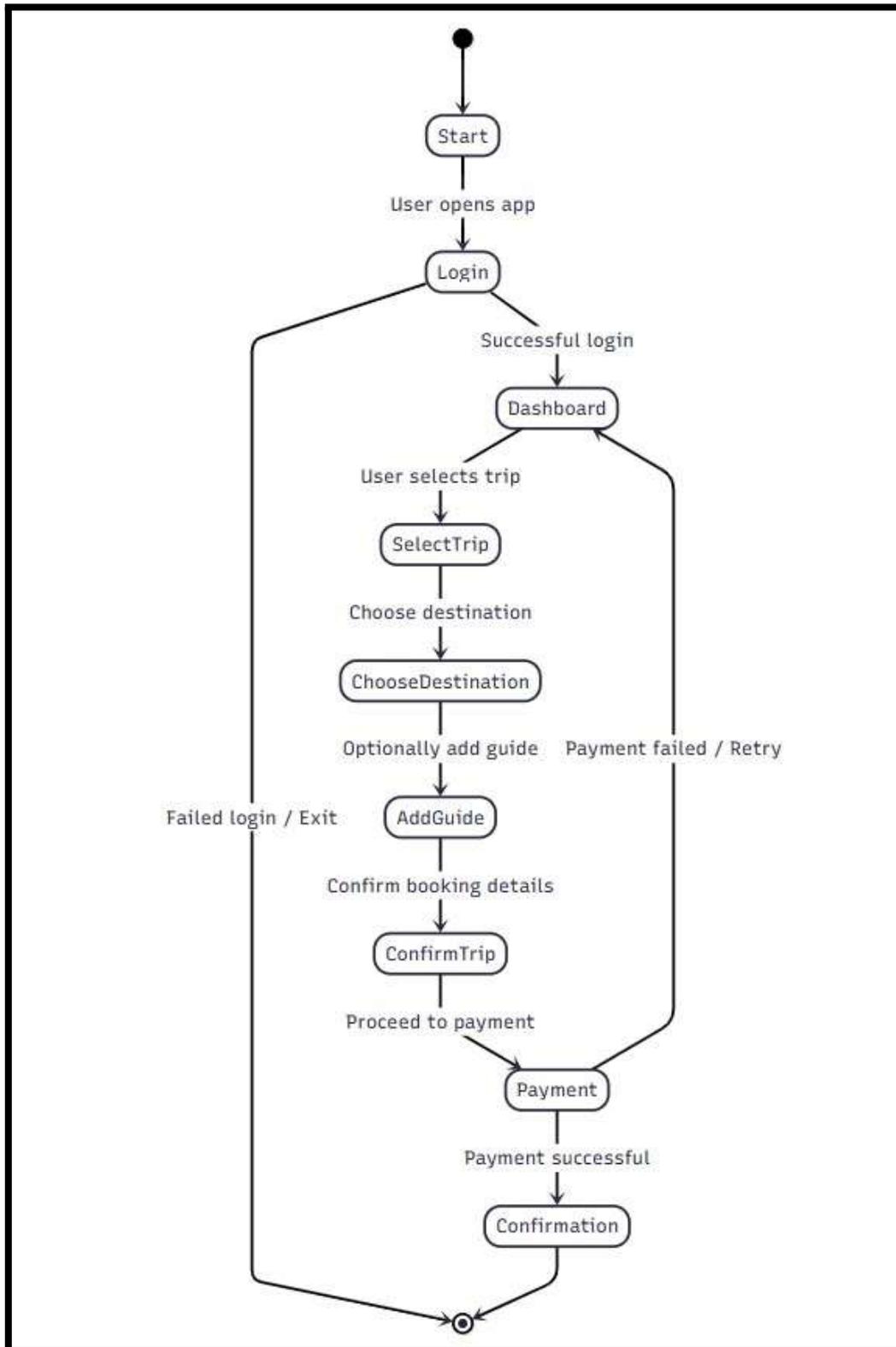
V.

VI.RESULTS AND DISCUSSION

The GetSetGo web-based travel management system was successfully implemented and tested under different user scenarios to evaluate its functionality and usability. The system allows users to register, search for available drivers, select travel routes, and book vehicles for long-distance travel through a simple web interface. During testing, the application demonstrated smooth booking management, reliable data storage, and efficient communication between travelers and drivers through the multilingual chat feature.The scheduling feature allowed users to pre-book rides by selecting the travel date and time, which improved travel planning and convenience. The backend system successfully stored user information, booking details, and travel schedules in the MongoDB database without data loss. Overall, the system provided a user-friendly interface and efficient travel booking experience.
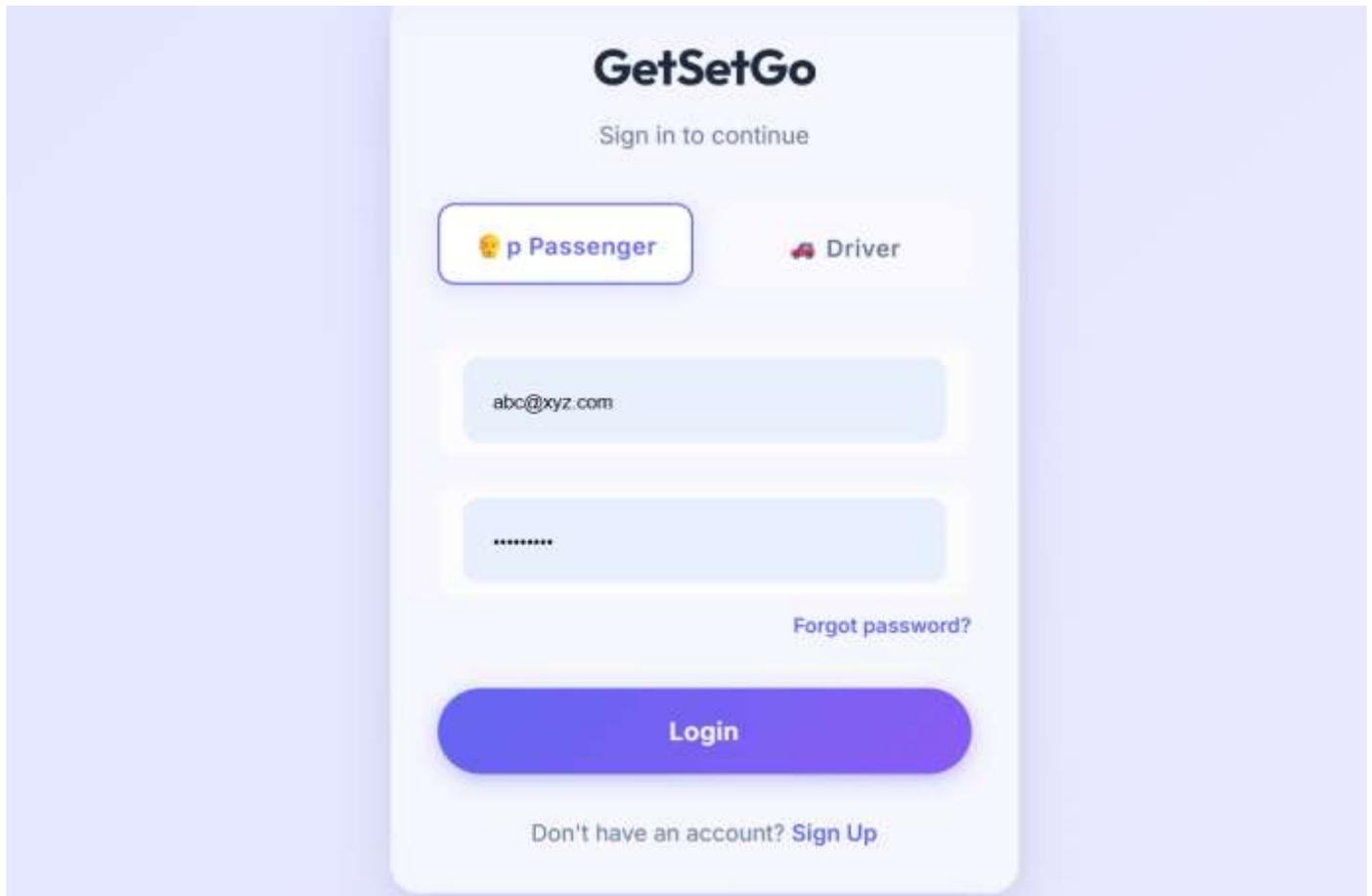
Figure 1.1

The landing page of the GetSetGo system provides a clear interface with separate login options for both drivers and users. This role-based access allows travelers to book rides and manage their travel plans, while drivers can access ride requests and manage bookings. The simple and user-friendly design ensures smooth navigation and secure access to the system.

Figure 1.2
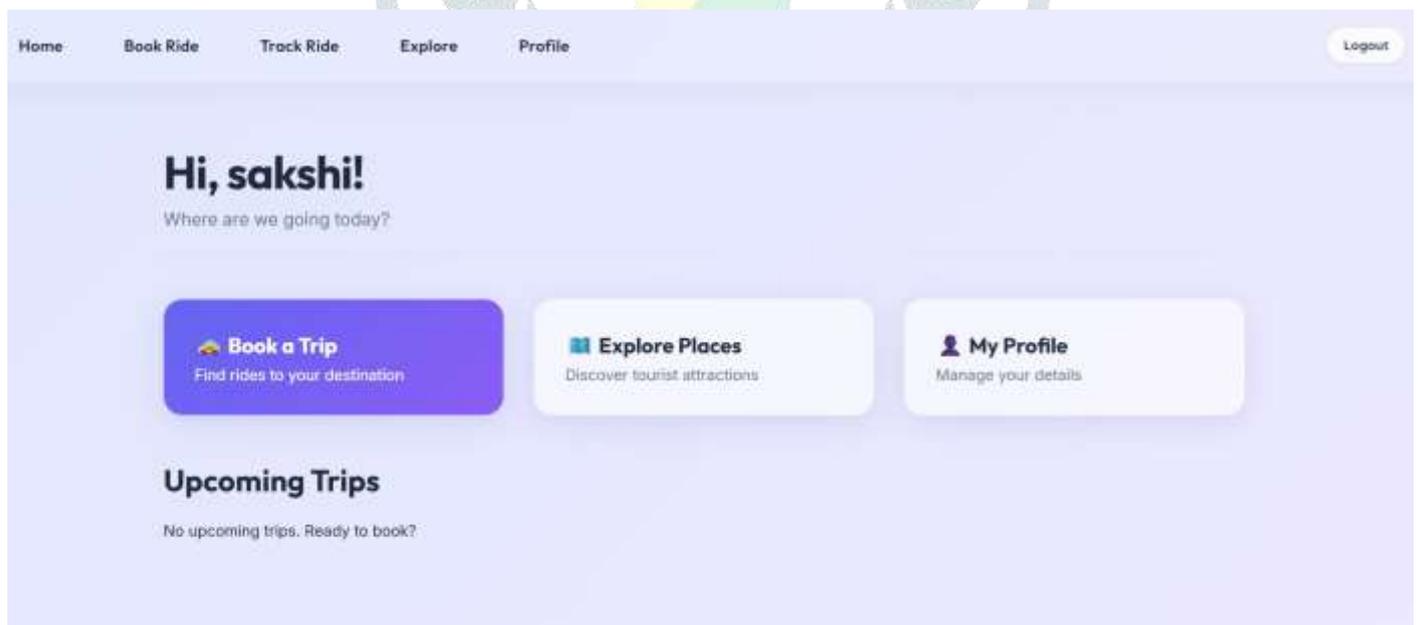
The homepage or user dashboard displays the main features of the system that allow users to manage their travel bookings. It includes options such as entering the starting location and destination, selecting the number of passengers, choosing the preferred vehicle type, and scheduling the ride by selecting the date and time. The homepage also provides navigation to other sections such as viewing booked rides and accessing travel-related information, making it easier for users to plan and manage their journeys.
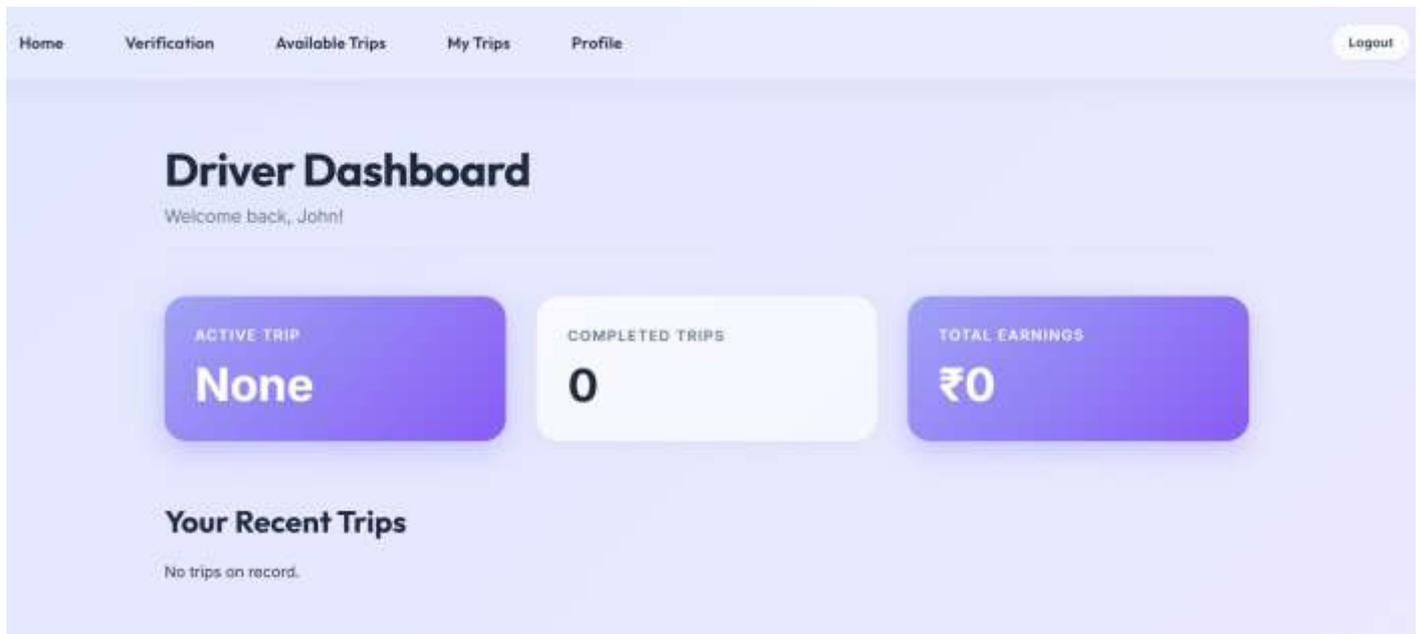
Figure 1.3

The driver dashboard provides drivers with the ability to view ride requests and manage their travel bookings. It displays information such as passenger details, travel routes, and scheduled ride timings. Drivers can also view **book ride requests** submitted by users and manage them accordingly. This interface helps drivers efficiently organize their trips and ensures smooth coordination between drivers and passengers.
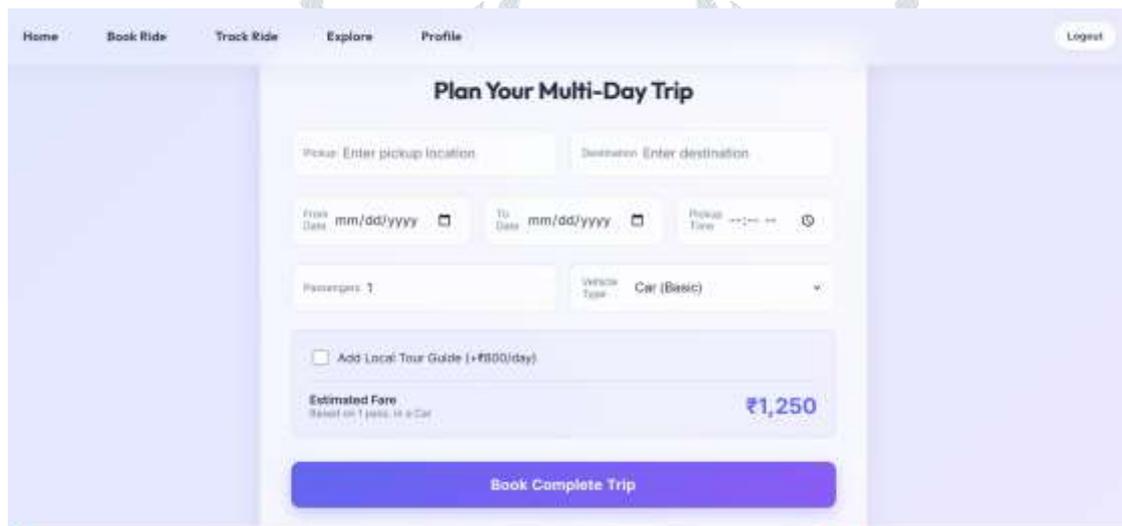

Figure 1.4

The Book Ride page allows users to enter travel details in order to request a ride. Users can provide information such as the starting location, destination, number of passengers, preferred vehicle type, and travel schedule. Once these details are submitted, the system processes the booking request and stores the ride information in the database. This page ensures that users can easily plan and confirm their travel bookings through a simple and organized interface.

VII. FUTURE SCOPE

The GetSetGo travel management system can be further enhanced by integrating additional features and advanced technologies. In the future, the system can be expanded to include a mobile application, allowing users and drivers to access travel services more conveniently through smartphones. This would improve accessibility and enhance the overall user experience.

Another important improvement would be the integration of live GPS tracking, which would allow users to monitor the real-time location of their ride during the journey. This feature would improve travel safety and transparency by enabling travelers to track their vehicle's movement and estimated arrival time.

Additionally, the system can be enhanced by introducing AI-based route recommendations that suggest the most efficient travel routes based on distance and travel conditions. Advanced analytics and reporting tools can also be added to analyze booking trends and user activity, helping improve system performance and travel planning services.

VIII. RESULT AND PERFORMANCE EVALUATION
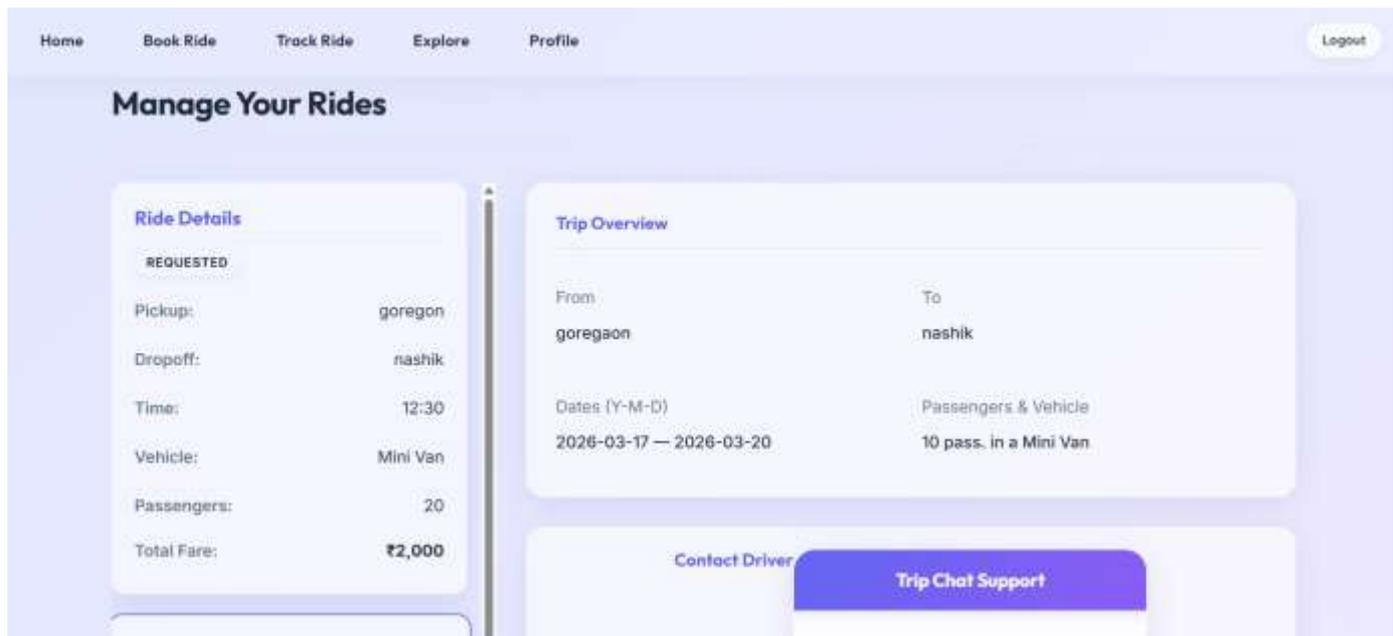
IX.   CONCLUSION

The GetSetGo web-based travel management system provides an efficient solution for simplifying long-distance travel planning and booking. The system allows users to book verified drivers and vehicles, customize travel routes, schedule rides, and communicate with drivers through a multilingual chat feature. By integrating these features into a single platform, the application improves convenience, transparency, and coordination between travelers and drivers.The system is developed using modern web technologies such as HTML, CSS, and JavaScript for the frontend, while Node.js and Express.js manage backend operations with MongoDB for database storage. The platform successfully demonstrates how technology can be used to enhance travel management services and provide a user-friendly booking experience. With future improvements such as live GPS tracking and mobile application support, the system can evolve into a more advanced and scalable travel management solution.

XI. ACKNOWLEDGMENT

X. REFERENCES

Node.js Documentation, "Node.js Official Documentation," Available: https://nodejs.org/en/docs/, Accessed January 2026.

Express.js Documentation, "Express.js – Web Framework for Node.js," Available: https://expressjs.com/, Accessed January 2026.

MongoDB Documentation, "MongoDB Database Documentation," Available: https://www.mongodb.com/docs/, Accessed January 2026.

Firebase Documentation, "Firebase Web Development Documentation," Available: https://firebase.google.com/docs/, Accessed January 2026.

MDN Web Docs, "MDN Web Docs – HTML, CSS, and JavaScript Documentation," Available: https://developer.mozilla.org/, Accessed January 2026.

W3Schools Documentation, "Web Development Tutorials and References," Available: https://www.w3schools.com/, Accessed January 2026.