Ride Booking Web App

Mrs. Deepthi Nair
Department of Computer Science and Engineering
Sri Shakthi Institute of Engineering and Technology
Coimbatore, India
@siet.ac.in

Rokith S
Department of Computer Science and Engineering
Sri Shakthi Institute of Engineering and Technology
Coimbatore, India
rokiths22cse@srishakthi.ac.in

Santhoshkumar A
Department of Computer Science and Engineering
Sri Shakthi Institute of Engineering and Technology
Coimbatore, India
santhoshkumara22cse@srishakthi.ac.in

Sudharsan V
Department of Computer Science and Engineering
Sri Shakthi Institute of Engineering and Technology
Coimbatore, India
sudharsanv22cse@siet.ac.in

Surya R
Department of Computer Science and Engineering
Sri Shakthi Institute of Engineering and Technology
Coimbatore, India
suryar22cse@srishakthi.ac.in

Navin Prasanth S
Department of Computer Science and Engineering
Sri Shakthi Institute of Engineering and Technology
Coimbatore, India
navinprasanths22cse@srishakthi.ac.in

Abstract — **The Ride Booking Web App** is an advanced transportation management platform designed to simplify ride discovery, booking, and monitoring by integrating automation, real-time tracking, and modern web technologies. It streamlines customer–driver interaction, vehicle management, pricing negotiation, and ride lifecycle handling while reducing coordination overhead. The system is built using a robust technology stack consisting of Ruby on Rails, PostgreSQL, Devise, Doorkeeper, and Active Admin, ensuring secure authentication, scalable processing, and efficient role-based workflows. Through automated ride handling, the platform manages requests, payments, reward points, and status updates, offering transparent operations and reliable service experiences. With interactive dashboards for customers, drivers, and administrators, RESTful API support, and real-time ride insights, this system serves as a transformative solution for modern transportation services seeking efficiency, scalability, and user-centric mobility management

Keywords - Ride Booking System, Ruby on Rails, PostgreSQL, Customer–Driver Interaction, Vehicle Management, RESTful API, Role-Based Access, Transportation Services.

I INTRODUCTION

The Ride Booking Web Application is an advanced transportation management solution designed to modernize and improve the efficiency of ride booking and coordination processes. By integrating automation, role-based workflows, and real-time updates, the system addresses key challenges in traditional ride management environments, where manual coordination is often inefficient, error-prone, and lacks transparency. As digital transportation services continue to grow, there is an increasing demand for scalable and secure platforms that provide seamless interaction, real-time ride tracking, and reliable service management. This application leverages a modern technology stack consisting of Ruby on Rails, PostgreSQL, Devise, Doorkeeper, and Active Admin to deliver a robust and user-centric ride booking experience

The system incorporates automated ride handling mechanisms capable of managing booking requests, pricing negotiations, and payment processing, ensuring accuracy and fairness for both customers and drivers. Secure authentication and authorization enable controlled access for multiple user roles, while RESTful APIs support smooth integration with future web or mobile front-end technologies. Through structured workflows, the platform manages vehicle listings, ride statuses, reward points, and transaction records efficiently. Administrators gain access to intelligent dashboards that monitor user activity, rides, payments, and performance analytics in real time. Customers benefit from quick vehicle discovery, transparent pricing, and live ride updates, while drivers can efficiently manage requests, pricing, and service records. With scalable architecture, secure data handling, and modular design, the Ride Booking Web Application serves as a reliable and transformative solution for modern transportation services.

II LITERATURE REVIEW

[1] J. Biswas, A. Thakur, and P. R. Reddy, *"Smart Cab Management System Using Web Technologies*," International Journal of Computer Applications, 2022.
This study presents a comprehensive web-based cab management platform that automates key workflows such as user registration, ride requests, driver allocation, and payment reconciliation. The authors emphasize the importance of a structured relational database model to manage entities including customers, drivers, vehicles, and ride transactions. Server-side logic is used to match riders with drivers and maintain transactional records for auditing and reporting. The study also highlights usability factors such as simple booking interfaces and transparent ride status updates to improve user adoption. The findings directly support our system's use of PostgreSQL and Ruby on Rails for structured data modelling, transactional integrity, and efficient ride workflow management.

[2] K. Sharma and N. Gupta, "*Web-Based Taxi Booking System with Real-Time Tracking*," IEEE International Conference on Smart Computing and Informatics, 2023.

This paper explores the integration of real-time tracking in web-based taxi booking systems to improve transparency, safety, and user trust. The authors demonstrate how continuous ride status updates enable customers and administrators to monitor trip progress and validate completed rides. Technical considerations such as data privacy, polling frequency, and performance optimization are discussed in detail. The research supports the relevance of real-time ride updates in transportation platforms and informs future enhancements of our system, particularly in implementing secure and efficient ride tracking features.

[3] P. Agrawal and R. Singh, "*Enhancing Transportation Systems Using RESTful Web Services*," International Journal of Advanced Computer Science and Applications (IJACSA), 2023.
This study highlights the role of RESTful web services in building scalable and modular transportation systems. The authors explain how APIs enable multiple client applications to access core functionalities such as vehicle search, ride booking, and status tracking without exposing backend logic. Best practices including endpoint versioning, stateless design, authentication, and rate limiting are discussed. The findings strongly support our decision to implement RESTful APIs using Ruby on Rails with Doorkeeper-based authorization for secure and extensible system integration.

[4] S. Patel and M. Deshmukh, "*Role-Based Access and Security in Online Ride Booking Systems*," International Conference on Data Science and Security (ICDSS), 2024.
This paper examines authentication and authorization mechanisms used in ride booking platforms, with a focus on role-based access control and secure session handling. The authors analyze common security threats and recommend established frameworks such as Devise, OAuth2, and JWT to mitigate risks. Emphasis is placed on restricting administrative privileges and safeguarding sensitive operations like fare adjustments and payment reversals. This research directly justifies our use of Devise for authentication and Doorkeeper for authorization, ensuring secure, role-aware access control in our system.

[5] L. Zhang, T. Zhao, and Y. Chen, "*Admin Dashboards for Effective Fleet Management in Ride-Sharing Platforms*," Journal of Software Engineering and Applications, 2024.
This study focuses on the design and effectiveness of administrative dashboards in ride-sharing platforms. The authors identify key features such as user management, vehicle monitoring, transaction reconciliation, dispute handling, and performance analytics. The research demonstrates that well-designed dashboards reduce operational workload and enable data-driven decision-making. These insights directly support our integration of ActiveAdmin, which provides administrators with centralized control and real-time visibility into system operations.

[6] M. K. Das and R. Pandey, "*Payment Gateway Integration in Web-Based Transportation Systems*," IEEE Transactions on Industrial Informatics, 2023.
This paper discusses secure payment integration strategies for transportation platforms, focusing on tokenization, compliance standards, and transaction reconciliation. The authors analyze failure scenarios such as partial payments and chargebacks and recommend idempotent payment flows to maintain financial consistency. The findings guide our payment module design, transaction logging, and administrative reconciliation features, ensuring secure and reliable payment processing within the ride booking system.

III PROBLEM STATEMENT

Traditional ride booking systems face significant limitations in delivering flexibility, transparency, and fairness within rapidly expanding transportation environments. Existing platforms rely heavily on centralized pricing models and automated decision-making, which restrict user control and reduce meaningful interaction between customers and drivers. Fixed fare generation requires users to accept system-defined prices, often leading to dissatisfaction among drivers due to reduced earnings and among customers due to lack of pricing options. Current ride management practices lack customizable pricing, transparent fare breakdowns, and adaptable workflows, making it difficult for stakeholders to understand cost structures or influence ride conditions. The absence of negotiation-based booking further limits fairness and accessibility for users with varied budget and service expectations.

Additionally, the lack of modular architecture and open integration prevents deeper system extensibility and operational insight. Monolithic designs and restricted API access contribute to inefficiencies in scaling, maintenance, and integration with modern web or mobile technologies. Fragmented ride data, limited administrative control, and opaque commission handling make it challenging to monitor platform performance or optimize operations effectively. These limitations highlight the need for a modern, flexible ride booking system that supports price negotiation, ensures transparency, enables role-based management, and provides scalable integration capabilities. A user-centric and API-driven solution is essential to enhance trust, improve efficiency, and deliver a balanced experience for both customers and drivers.

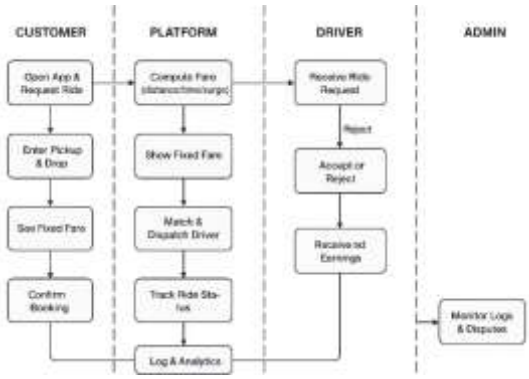

Fig 1.1: Workflow of existing system

IV PROPOSED SYSTEM

The proposed Ride Booking Web Application is designed to modernize transportation management by providing a secure, flexible, and user-centric platform for booking and managing rides. Leveraging intelligent price negotiation mechanisms, role-based workflows, and real-time ride management, the system enables fair and transparent interaction between customers and drivers. Developed using a full-stack architecture with a Ruby on Rails backend, PostgreSQL database, and integrated authentication and authorization mechanisms, the platform supports dynamic fare quoting, secure payments, and efficient ride lifecycle handling. By incorporating Active Admin for centralized administration, Devise for authentication, and Doorkeeper for API authorization, the system enhances operational control, scalability, and future integration capabilities. The platform aims to reduce brokerage dependency, improve pricing fairness, and deliver a reliable, efficient, and transparent ride-booking experience for all stakeholders.

V METHODOLOGY

The first stage of developing the Ride Booking Web Application focuses on designing a secure and structured backend capable of handling ride workflows, user interactions, and real-time validations efficiently. This begins with defining a robust data model using PostgreSQL to represent core entities such as users, drivers, vehicles, rides, payments, and reward points. The application is implemented using the Ruby on Rails framework following the Model–View–Controller (MVC) architecture, ensuring clear separation of concerns, maintainability, and scalability. Secure authentication is established using the Devise gem, enabling role-based access for customers, drivers, and administrators. Authorization for API-level interactions is handled through Doorkeeper OAuth, ensuring controlled and secure access for future mobile or frontend integrations. Throughout this stage, strong backend validations are implemented to prevent data inconsistencies, scheduling conflicts, and unauthorized actions, thereby maintaining system reliability and transparency.

The next phase focuses on managing the complete ride lifecycle through structured workflows and business logic. Customers initiate ride requests by selecting vehicles, proposing fares, and specifying ride details such as pickup location, drop location, and start time. The backend enforces multiple validation rules, including time constraints, overlapping ride prevention, and request conflict resolution. Drivers can respond to ride requests by accepting, rejecting, or negotiating the proposed fare, enabling a flexible and fair pricing mechanism. Once both parties agree on a fare, the system confirms the booking and activates countdown timers and ride status tracking. All actions including negotiations, confirmations, rejections, and cancellations are logged with timestamps and user identifiers, ensuring traceability and accountability throughout the booking process.

In the final stage, the system integrates payment handling, reward management, and administrative monitoring to support operational efficiency and long-term scalability. After ride completion, payment confirmation updates the ride status and triggers reward point allocation and rating submissions.

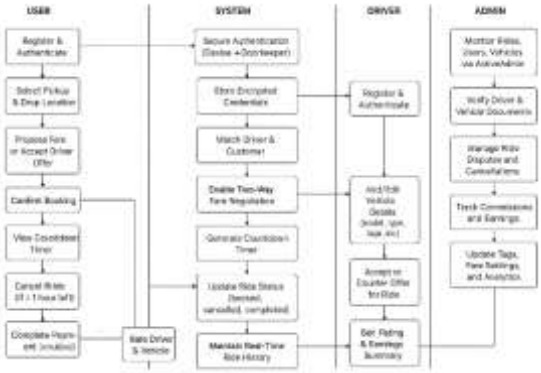

Fig 1.2: Detailed workflow diagram

VI RESULTS AND ANALYSIS

The Ride Booking Web Application has been successfully developed as a comprehensive and user-centric transportation management platform capable of handling ride requests, fare negotiation, payments, and ride history tracking with high reliability and accuracy. The system integrates structured backend validations, role-based workflows, and real-time processing to deliver a transparent and efficient booking experience for customers, drivers, and administrators. By automating ride scheduling, negotiation handling, cancellation logging, and reward management, the platform significantly reduces manual intervention and operational ambiguity when compared to traditional centralized ride-booking systems.

Fig 1.3: User Dashboard Page

The User Dashboard Page, user can navigate and see various pages as well as the new notifications at the top of the page.

Fig 1.4: Browse Vehicles Page

The Browse Vehicles Page, powered with advanced filtering with vehicle model, type, ratings, tags and driver's name.

Fig 1.5: New Booking Page

The New Booking Page, where customer can request any new booking for the selected vehicle by filling up the start location, end location, start time and initial price.

Fig 1.6: Ride Request Page

The Ride Request Page of driver, any new ride request from any customers appears here, the driver can propose a new price or even reject the ride request.

Fig 1.7: Ride History Page

The Ride History Page illustrates the completed rides with ratings, the customer can add/edit/delete their ratings for the vehicle as well as ride.

Fig 1.8: Reward History page

The Reward History Page of customer, display the reward points which are earned by the customer by involving in various activities.

VII CONCLUSION

The Ride Booking Web Application represents a meaningful advancement in transportation technology by combining secure authentication, dynamic pricing negotiation, and structured ride management to deliver a reliable and user-centric booking platform. The system simplifies ride booking, management, and tracking while ensuring transparency and trust between customers and drivers through well-defined digital workflows.

The platform demonstrated stable performance by accurately handling ride scheduling, fare negotiation, and real-time backend validations, significantly reducing manual coordination and pricing conflicts. Secure data handling using Ruby on Rails and PostgreSQL, along with role-based access via Devise, ensured consistency and safety across operations. Overall, the system provides a scalable, efficient, and transparent solution that enhances user experience and establishes a strong foundation for flexible and future-ready ride booking services.

VIII FUTURE WORK

Future enhancements for the Ride Booking Web Application aim to evolve the platform into a more intelligent, scalable, and feature-rich transportation ecosystem. The system can be extended with real-time GPS-based ride tracking and live map integration to improve route visibility, safety, and user confidence during trips. Advanced analytics for driver and customer ratings can be introduced to generate performance insights, identify service improvement areas, and enhance overall ride quality. Integration of secure online payment gateways will enable seamless digital transactions and reduce manual payment handling. Additionally, mobile applications for Android and iOS platforms can be developed to increase accessibility and user convenience. AI-based ride matching and fare prediction mechanisms may be incorporated to optimize driver–customer pairing and dynamic pricing decisions. Notification services using SMS or email for ride confirmations, cancellations, and reminders will further enhance communication. These enhancements will strengthen scalability, real-time responsiveness, and user trust, transforming the system into a comprehensive and intelligent ride-booking solution.

IX REFERENCE

[1] S. Banerjee, R. Johari, and C. Riquelme, "Dynamic Pricing in Ridesharing Platforms," SIGecom Exchanges, vol. 15, no. 1, 2015. DOI:10.2139/ssrn.2568258

[2] C. Yan, H. Zhu, N. Korolko, and D. Woodard, "Dynamic Pricing and Matching in Ride-Hailing Platforms," Naval Research Logistics, 2020. DOI:10.2139/ssrn.3258234

[3] L. Sun and P. Erftemeijer, "Optimal Pricing for Ride-Sourcing Platforms," Transportation Research Part B, 2019. DOI:10.1016/j.trb.2019.03.008

[4] S. Heydari and E. A. Noughabi, "Dynamic Pricing in Ride-Hailing Intelligent Transportation Systems Using Deep Reinforcement Learning," SSRN, 2024. DOI:10.2139/ssrn.4746254

[5] P. Kalra, J. Pahwa, A. Sharma, D. Malhotra, and K. Pandey, "Dynamic Ride Pricing Model Using Machine Learning," International Journal of Scientific Research & Engineering Trends, vol. 10, no. 6, 2024