International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 04 | April - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

RINEGO – FLEXIBLE RIDE BOOKING APP

Ms. P. Sasikala
Department of Computer Science and Engineering
Sri Shakthi Institute of Engineering and Technology
Coimbatore, India
sasikalacse@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@srishakthi.ac.in

Abstract - The Rinego – Flexible Ride Booking App is an advanced transportation management platform designed to simplify ride discovery, booking, and monitoring by integrating automation, real-time communication, and map-based technologies. It streamlines customer–driver interaction, vehicle registration, fare estimation, and ride lifecycle management while minimizing operational complexity. The system is developed using a robust technology stack comprising Ruby on Rails, PostgreSQL, Tailwind CSS, ActionCable, and Stripe, ensuring secure authentication, scalable processing, and efficient role-based workflows. Through automated ride handling, the platform manages booking requests, real-time chat, fare calculations, reward points and payment processing, providing transparency and enhanced user experience. With interactive interfaces for customers and drivers, RESTful architecture, live location tracking via OpenStreetMap and Leaflet.js, and accurate distance computation using the Haversine formula, the system serves as a reliable and scalable solution for modern transportation services seeking efficiency, flexibility, and user-centric mobility management.

Keywords - Ride Booking System, Ruby on Rails, PostgreSQL, Customer–Driver Interaction, Vehicle Management, Leaflet.js, OpenStreetMap, RESTful Architecture, Role-Based Access, Real-Time Communication.

I INTRODUCTION

The Rinego 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 communication, 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, Tailwind CSS, ActionCable, and Stripe to deliver a robust and user-centric ride booking experience.

The system incorporates automated ride handling mechanisms capable of managing booking requests, fare estimation, pricing negotiation, and payment processing, ensuring accuracy and fairness for both customers and drivers. Secure authentication and authorization enable controlled access for multiple user roles, while integrated communication systems support smooth interaction between users and enable future extensibility. Through structured workflows, the platform manages vehicle listings, ride statuses, reward points, and transaction records efficiently. Administrators gain access to centralized dashboards that monitor user activity, rides, payments, and system performance in real time. Customers benefit from interactive ride selection, transparent pricing, and live ride updates, while drivers can efficiently manage requests, vehicle details, and service records. With scalable architecture, secure data handling, and modular design, the Rinego 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.

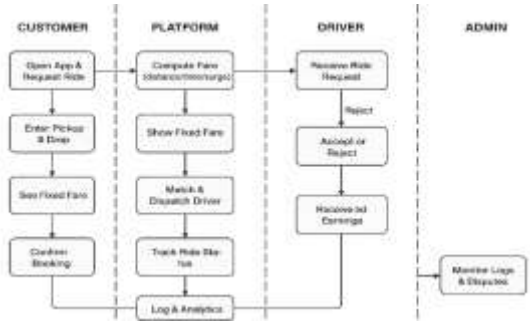

Fig 1.1: Workflow of existing syste

IV PROPOSED SYSTEM

The proposed Rinego 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 interactive map-based booking, real-time communication, and structured workflows, the system enables transparent and efficient interaction between customers and drivers. Developed using a full-stack architecture with a Ruby on Rails backend, PostgreSQL database, and integrated real-time technologies, the platform supports dynamic fare estimation, secure payment processing, and efficient ride lifecycle management. By incorporating map integration for accurate location selection, WebSocket-based communication for live updates, and administrative controls for centralized monitoring, the system enhances operational efficiency, scalability, and user experience. The platform aims to eliminate manual coordination, improve pricing transparency, and deliver a reliable, efficient, and user-friendly ride-booking solution for all users.

V METHODOLOGY

The first stage of developing the Rinego Ride Booking Web Application focuses on designing a secure and structured system capable of handling user authentication, role management, and core ride-booking operations efficiently. This begins with defining a robust data model using PostgreSQL to represent key entities such as users, customers, drivers, vehicles, and bookings. The application is implemented using the Ruby on Rails framework following the Model–View–Controller (MVC) architecture, ensuring clear separation of concerns, scalability, and maintainability. Secure authentication is established using the Devise framework, which utilizes encrypted password storage and session management to protect user credentials. Role-based access control is implemented by identifying user types and restricting access to authorized resources, ensuring that customers and drivers interact only with permitted functionalities. Throughout this stage, validation mechanisms are applied to maintain data integrity, prevent unauthorized access, and ensure system reliability.

The next phase focuses on managing the ride lifecycle through structured workflows and algorithmic processes. Customers initiate ride requests by selecting pickup and destination locations using an interactive map interface, where the system retrieves geographic coordinates and converts them into readable addresses. Fare estimation is calculated using the Haversine formula, which determines the distance between locations and applies a predefined pricing model. The system stores booking requests and assigns them to drivers based on availability and vehicle status. Drivers can accept or reject requests, and upon acceptance, the booking status is updated instantly. The workflow follows a structured state transition model, where bookings move through stages such as pending, accepted, in-progress, and completed. Each action is recorded with relevant details, ensuring traceability and smooth coordination between users.

In the final stage, the system integrates real-time communication, workflow execution, and user interaction handling to enhance overall efficiency and usability. The platform utilizes WebSocket-based communication to enable instant messaging between customers and drivers and to provide live updates during the ride. The application also incorporates GPS-based tracking, allowing users to monitor ride progress in real time through an interactive map interface. Upon ride completion, the system updates booking records and enables further actions such as confirmations and feedback. The structured workflow, combined with secure processing and real-time features, ensures a seamless and reliable ride-booking experience while supporting scalability and future system enhancements.

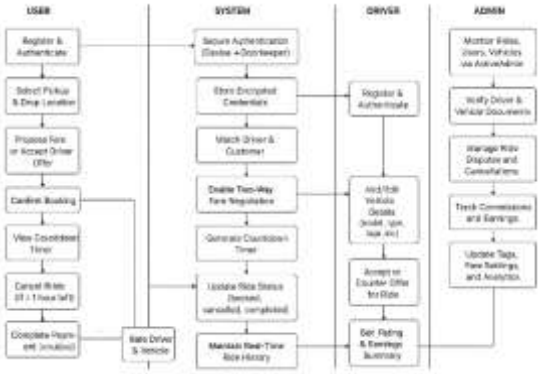

Fig 1.2: Detailed workflow diagram

VI RESULTS AND ANALYSIS

The Rinego Ride Booking Web Application has been successfully developed as a comprehensive and user-centric transportation platform capable of handling ride requests, real-time communication, fare estimation, payments, and reward management with high reliability and accuracy. The system integrates secure authentication, role-based access control, and structured workflows to deliver a transparent and efficient experience for customers, drivers, and administrators. By automating booking processes, driver assignment, communication handling, payment processing, and reward allocation, the platform significantly reduces manual effort and operational complexity compared to traditional ride-booking approaches while ensuring consistency, usability, and overall system performance.

Fig 1.3: Customer Dashboard Page

The Customer Dashboard Page allows users to navigate through different sections of the ride booking system and access key features through a centralized interface. It displays important notifications at the top of the page, keeping users updated with ride requests and status changes. The dashboard provides an overview of ride statistics such as total bookings, completed rides, and current ride status, along with quick navigation options like My Bookings, Ride History, and Profile for easy access to essential functionalities, ensuring a smooth and user-friendly experience.

Fig 1.4: Ride Booking Page

The Main Booking Interface of the ride booking system enables users to initiate a ride request by automatically detecting the "From" location based on their current position, while also allowing manual selection if required. It provides real-time information such as calculated distance, estimated travel time, and approximate fare. The interface also displays nearby available drivers on the map, enabling users to view and select suitable ride options efficiently.

Fig 1.5: My Bookings Page

The My Bookings Page represents the booking management interface of the ride booking system, where users can view both active and cancelled rides. It displays each booking in a structured format, allowing users to easily track ride status and monitor their activity. For active bookings, quick action options such as tracking the driver, chatting, and completing the ride are available. The page also shows key details including customer information, fare, and vehicle details, while cancelled bookings include cancellation time and the user who initiated the cancellation.

Fig 1.6: Live Tracking Page

The Live Tracking Page of the ride booking system allows customers to monitor the real-time location of their booked driver through an interactive map interface. It enhances communication by enabling users to chat directly with the driver during the ride. Additionally, push notifications are sent to ensure timely updates and smooth coordination between both parties.

Fig 1.7: Ride History Page

The Ride History Page represents the completed ride records of the ride booking system, allowing customers to view a detailed history of their past trips. It provides complete information for each ride along with options to manage feedback. Users can add, edit, or delete ratings for both the vehicle and the ride, enabling better experience tracking and service evaluation

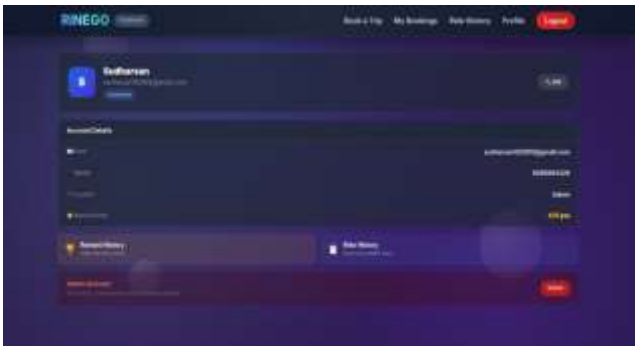

Fig 1.8:Profile page

The Customer Profile Page of the ride booking system allows users to view and update their personal details, ensuring their information remains accurate and up to date. It

also provides easy navigation to related sections such as Reward History and Ride History.

VII CONCLUSION

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

The platform demonstrated stable performance by accurately handling booking operations, fare estimation, and real-time system interactions, significantly reducing manual coordination and communication gaps. Secure data handling using Ruby on Rails and PostgreSQL, along with role-based access through authentication frameworks, 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 modern and adaptable ride booking services.

VIII FUTURE WORK

Future enhancements for the Rinego 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 based on historical ride data can be introduced to generate performance insights, identify service improvement areas, and enhance overall ride quality. Integration of secure and scalable cloud deployment can further improve system reliability, availability, and handling of large user loads. Additionally, mobile applications for Android and iOS platforms can be developed to increase accessibility and user convenience across devices. AI-based ride matching, dynamic pricing, and predictive ETA mechanisms may be incorporated to optimize driver–customer allocation and improve decision-making efficiency. Notification services using SMS and email for ride confirmations, updates, cancellations will further strengthen communication. These enhancements will improve scalability, intelligence, and real-time responsiveness, transforming the system into a more advanced and production-ready 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

[6] Huang, H. L., et al. (2023). Ridesharing Price Prediction: Exploring the Strategies of Dynamic Pricing. Highlights in Business, Economics and Management. DOI: (not provided.

[7] Banerjee, S., Johari, R., & Riquelme, C. (2015). Dynamic Pricing in Ridesharing Platforms. SIGecom Exchanges, 15(1). DOI:10.2139/ssrn.2568258

[8] MacKay, A. J., & Weinstein, S. N. (2022). Dynamic Pricing Algorithms, Consumer Harm, and Regulatory Response. Harvard Business School Working Paper. DOI:10.2139/ssrn.4052431

[9] Alwan, A. A., Ata, B., & Zhou, Y. (2023). A Queueing Model of Dynamic Pricing and Dispatch Control for Ride-Hailing Systems Incorporating Travel Times. arXiv. DOI:10.48550/arXiv.2302.02265

[10] Lai, Z., & Li, S. (2022). Spatiotemporal Pricing and Fleet Management of Autonomous Mobility-on-Demand Networks: A Decomposition and Dynamic Programming Approach. arXiv. DOI:10.48550/arXiv.2206.03298

[11] Tang, X., Zhang, F., Qin, Z., et al. (2021). Value Function is All You Need: A Unified Learning Framework for Ride Hailing Platforms. arXiv. DOI:10.48550/arXiv.2105.08791

[12] Lu, K., & Du, H. (2024). Pricing Decisions for Competitive Ride-Hailing Platforms with the Combination of Inner-Group and Inter-Group Network Externalities. Promet-Traffic&Transportation,36(1),147-163. DOI:10.7307/ptt.v36i1.320

[13] Bai, J. (2024). Analysis on the Dynamic Pricing Strategies and User Choice Behaviors of Online Car-Hailing Platforms under the Sharing Economy Model. Economics & Management Studies, 4, (Issue). DOI: (not provided)

[14] Kalra, P., Pahwa, J., Sharma, A., Malhotra, D., & Pandey, K. (2024). Dynamic Ride Pricing Model Using Machine Learning. Int. Journal of Scientific Research & Engineering Trends, 10(6). DOI:(not provided)

[15] Mejjaouli, A., & Tadj, L. (2023). Dynamic Pricing Strategies of Service Systems with Ride-Hailing Applications. E–Commerce Research. DOI:10.3390/0718-1876/20/3/224