CAMPUS CARPOOLING: OPTIMIZED RIDESHARING FOR STUDENTS USING HYBRID RIDESHARING ALGORITHM

PHASE I REPORT

Submitted by

LOGESHWARAN ELUMALAI 210701134

MOHAMED AADHIL A 210701159

In partial fulfillment for the requirement of award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

ANNA UNIVERSITY, CHENNAI

2024

RAJALAKSHMI ENGINEERING COLLEGE CHENNAI

BONAFIDE CERTIFICATE

Certified that this report titled **"CAMPUS CARPOOLING: OPTIMIZED RIDESHARING FOR STUDENTS USING HYBRID RIDESHARING ALGORITHM"** is the bonafide work of **LOGESHWARAN ELUMALAI (210701134), MOHAMED AADHIL A 210701159,** who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE
Dr. P. Kumar, M.E., Ph.D.
HEAD OF THE DEPARTMENT
Professor
Department of Computer Science and Engineering
Rajalakshmi Engineering College
Thandalam, Chennai - 602 105

SIGNATURE
Ms. B.S. Dharshini, M.E.
SUPERVISOR
Assistant Professor
Department of Computer Science and Engineering
Rajalakshmi Engineering College
Thandalam, Chennai - 602 105

Submitted to Project Viva-Voce Examination held on ____________________________

Internal Examiner **External Examiner**

TABLE OF CONTENTS

ACKNOWLEDGEMENT

Initially we thank the Almighty for being with us through every walk of our life and showering his blessings through the endeavor to put forth this report. Our sincere thanks to our Chairman **Mr. S. MEGANATHAN, B.E, F.I.E**., our Vice Chairman **Mr. ABHAY SHANKAR MEGANATHAN, B.E., M.S.**, and our respected Chairperson **Dr. (Mrs.) THANGAM MEGANATHAN, Ph.D.**, for providing us with the requisite infrastructure and sincere endeavoring in educating us in their premier institution. Our sincere thanks to **Dr. S.N. MURUGESAN, M.E., Ph.D.**, our beloved Principal for his kind support and facilities provided to complete our work in time. We express our sincere thanks to **Dr. P. KUMAR, Ph.D**., Professor and Head of the Department of Computer Science and Engineering for his guidance and encouragement throughout the project work. We convey our sincere and deepest gratitude to our internal guide, **Ms. B.S. Dharshini, M.E.** Department of Computer Science and Engineering. Rajalakshmi Engineering College for her valuable guidance throughout the course of theproject. We are very glad to thank our Project Coordinator, **Mr. Kumaragurubararan, B.E** Department of Computer Science and Engineering for his useful tips during our review to build our project.

LOGESHWARAN ELUMALAI 210701134

MOHAMED AADHIL A 210701159

ABSTRACT

A student-centric carpooling service tailored to address the unique transportation needs of college students. The platform allows users to post their starting locations, available seats, and routes, while enabling other users to select their desired pick-up points. The service employs a machine learning algorithm to optimize the matching process between riders and passengers, ensuring efficient and convenient travel arrangements. A unique feature of the system is dynamic pricing, which adjusts based on demand and supply factors, offering flexibility in payment options for rides, which can be either paid or free, depending on the rider's choice. The platform incorporates user registration and authentication to maintain security and trust within the community, with profiles for both riders and passengers that include ratings and reviews to establish a reliable user base. The machine learning component utilizes clustering and optimization algorithms to reduce total travel time and distance, thereby enhancing the overall efficiency of the service. Additional features such as an integrated payment gateway for paid rides, identity verification, emergency contact information, and ride tracking are included to ensure safety. In app messaging and notifications provide seamless communication between users. An admin panel is also provided for the management of users, rides, and payments, ensuring smooth operations and quick resolution of issues. The project aims to develop a sustainable, efficient, and cost-effective carpooling system tailored for college students, promoting environmentally friendly transportation solutions on campuses.

LIST OF FIGURES

LIST OF ABBREVIATIONS

SNO	ABBREVIATION	EXPANSION
1	HRA	Hybrid Ridesharing Algorithm
2	API	Application Programming Interface
3	PWA	Progressive Web Application
4	K -Means	K-Means Clustering Algorithm
5	GA	Genetic Algorithm
6	ML	Machine Learning
7	NLP	Natural Language Processing
8	SMSs	Shared Mobility Systems
9	VKT	Vehicle Kilometers Traveled
10	CH	Contraction Hierarchies
11	TO-PP	Time-Optimal and Privacy Preserving
12	IOT	Internet of Things
13	REST	Representational State Transfer

CHAPTER 1

1. INTRODUCTION

1.1 GENERAL

Transportation remains a significant challenge for college students, who often rely on limited and sometimes unreliable options such as public transportation, cycling, or walking. These methods can be inconvenient, especially during peak hours or in regions with inadequate public transit infrastructure. The lack of affordable, efficient, and flexible transportation options can affect students' daily lives, contributing to wasted time, increased stress, and even missed opportunities for classes, extracurricular activities, and part-time jobs. This project proposes a student centric carpooling service designed to address these transportation challenges by providing a reliable, cost-effective, and environmentally friendly solution tailored specifically to the needs of college students.

The proposed carpooling platform allows students to register as either riders or passengers, enabling them to post their starting locations, available seats, and preferred routes, while passengers can select pick-up points that best match their schedules and preferences. This user-friendly system is built around a machine learning algorithm, specifically the K-means clustering algorithm, which is employed to create optimal carpool groups by clustering users based on geographical proximity and route similarity. The algorithm helps minimize total travel time and distance by matching riders and passengers with overlapping routes, ensuring efficient and convenient travel arrangements. This optimization not only improves the overall user experience but also contributes to reduced fuel consumption and lower carbon emissions, promoting sustainable practices within the college community.

To further enhance the platform's appeal, a dynamic pricing model is implemented, which adjusts ride costs based on factors such as distance, demand, and supply. This model introduces flexibility in payment options, allowing riders to choose between offering free or paid rides. By aligning costs with real-time market dynamics, the system provides a fair and balanced approach that benefits both drivers and passengers.

Safety is a paramount concern in carpooling, and the platform is designed with robust security features, including user registration, identity verification, and emergency contact information, to create a safe and trustworthy environment for all participants. Profiles for both riders and passengers are equipped with ratings and reviews, which help build a reliable user base by promoting transparency and accountability. Building this project involves a multi-layered approach that leverages modern software development frameworks and tools. A relational database, such as PostgreSQL, will be integrated to manage user profiles, ride details, routes, and transaction histories, ensuring data consistency and integrity. The machine learning model, developed using Python libraries such as scikit learn, will be responsible for the clustering and optimization tasks to effectively match riders and passengers. The frontend of the platform will be developed using modern JavaScript frameworks like React or Angular, providing a responsive and intuitive interface for users to easily access and manage their rides. Additional features, such as real-time ride tracking using Google Maps API, in-app messaging for communication, and notifications for ride updates, are included to ensure a seamless user experience.

Moreover, the platform includes an admin panel for comprehensive oversight and management capabilities. Administrators can use this panel to monitor user activity, manage ride requests, handle disputes, and ensure compliance with community guidelines, thereby maintaining the platform’s integrity and operational efficiency. By integrating advanced machine learning techniques, secure payment processing, real time tracking, and effective user management features, this project aims to deliver a holistic carpooling solution. The goal is to create a sustainable, efficient, and community-driven transportation system that not only reduces costs and travel time for students but also fosters a sense of shared responsibility toward environmental conservation. Ultimately, this project seeks to transform how students manage their daily commutes, creating a model for other communities to adopt similar sustainable transportation solutions.

1.2 OBJECTIVE

The primary objective of the campus carpooling project is to address the transportation challenges faced by college students. By leveraging advanced algorithms and sustainable practices, the platform aims to provide an efficient, cost-effective, and user-friendly carpooling solution. Below are the key objectives of the project:

1. Efficient Ride Matching:
   Develop an optimized ride-matching system using machine learning algorithms to group users based on proximity and preferences. This ensures minimal detours and reduced travel time for riders and passengers.

2. Dynamic Pricing Mechanism:
   Implement a real-time dynamic pricing model that adjusts fares based on demand and supply. This provides flexibility for riders and drivers, ensuring fairness and affordability.

3. Safety and Trust:
   Enhance safety through user authentication, identity verification, and emergency contact integration. Build a reliable platform with ratings and reviews to foster trust among users.

4. Environmental Sustainability:
   Promote eco-friendly transportation by reducing the number of vehicles on the road. This helps lower CO_2 emissions and contributes to a greener campus environment.

5. User-Friendly Experience:
   Design a Progressive Web Application (PWA) for seamless access across devices. Include features like real-time tracking, notifications, and in-app messaging for a smooth and intuitive user experience.

1.3 EXISTING SYSTEM

Carpooling and ride-sharing platforms have gained popularity in recent years as viable solutions to transportation issues, offering economic and environmental benefits. However, most existing systems are designed to cater to a broad audience rather than focusing exclusively on niche groups like students. These platforms, such as Uber, Lyft, and BlaBlaCar, operate on generic models that prioritize maximizing user base and revenue, which can sometimes overlook the unique requirements of specific communities.

One significant feature of current ride-sharing platforms is their dynamic ride-matching algorithms. These systems connect passengers with drivers based on proximity and route similarity. While this ensures convenience for a wide range of users, the algorithms do not typically account for specific challenges like restricted budgets, fixed schedules, or safety concerns that are especially critical for students. Instead, the primary focus remains on the quickest and most profitable matches, which can lead to inefficiencies for those with unique transportation needs.

Moreover, most existing platforms use dynamic pricing strategies to balance demand and supply. While this provides financial incentives for drivers and adjusts pricing during peak hours, it often results in higher costs during high-demand periods. Students, who may have limited budgets, are particularly impacted by surge pricing, making it less accessible and affordable for them during times of critical need, such as early mornings or after classes.

Another common limitation is the lack of personalized safety measures. While platforms typically include user authentication and basic identity verification, these features are often not tailored to specific groups like students. For example, the absence of campus-specific pick-up points, in-app emergency contact systems, or restricted community access can make these platforms less suitable for a younger demographic. Furthermore, most systems do not provide mechanisms to restrict usage to verified student communities, leading to potential safety and trust concerns.

From an environmental perspective, existing platforms have made strides in reducing carbon emissions by encouraging shared rides and fewer vehicles on the road. However, their effectiveness in promoting sustainability is often diluted by their profit-driven nature. Many of these platforms do not actively incentivize or prioritize environmentally conscious travel behaviors, such as selecting electric or hybrid vehicles, or optimizing routes for minimal fuel consumption.

In terms of user experience, these platforms are typically optimized for general usability and scalability. They offer standard features like real-time ride tracking, in-app messaging, and digital payment integration. While these features enhance convenience, they lack customization for specific user groups. Students, for instance, may benefit from features like scheduled rides for fixed class timings, reduced fares for frequent users, or a system designed to minimize delays caused by rigid campus entry and exit points.

Overall, the generic nature of existing ride-sharing systems often fails to cater to the unique requirements of students. They focus on general convenience and scalability, overlooking opportunities for community-specific enhancements. The absence of targeted features, budget considerations, and robust safety measures highlights the need for a student-centric carpooling platform that is more aligned with the specific challenges and opportunities within college campuses.

1.4 PROPOSED SYSTEM

The proposed system is a student-centric carpooling platform designed to address the specific transportation challenges faced by college students. Unlike generic ride-sharing services, this system is tailored to meet the unique needs of a campus environment, providing cost-effective, safe, and eco-friendly transportation solutions. By leveraging advanced algorithms, modern technology, and user-friendly features, the platform ensures optimized ride-matching, dynamic pricing, and enhanced user trust.

A key component of the proposed system is the Hybrid Ridesharing Algorithm (HRA), which integrates K-Means clustering and Genetic Algorithms to optimize the carpooling process. The algorithm begins by grouping users based on their geographic proximity using K-Means clustering. This reduces the complexity of ride-matching by identifying potential carpool groups that share overlapping routes. Next, a Genetic Algorithm is applied to optimize the sequence of pick-up and drop-off points within each group. This two-stage approach ensures minimal travel time and distance, reducing detours and improving the overall efficiency of rides.

To further enhance the user experience, the platform incorporates a dynamic pricing model that adjusts ride costs in real-time based on demand, supply, and other factors. The pricing mechanism combines surge pricing principles with machine learning-based dynamic adjustments. This ensures fairness and affordability for students by balancing user demand with driver availability. For instance, during high-demand periods, prices increase slightly to incentivize more drivers to offer rides, while during low-demand times, prices decrease to attract passengers. Riders also have the flexibility to offer free or paid rides, depending on their preferences.

Safety and trust are paramount in the proposed system. The platform includes comprehensive user authentication processes to verify the identity of both drivers and passengers. Emergency contact integration and live ride tracking provide additional layers of security, ensuring users feel safe throughout their journey. Furthermore, the system includes a rating and review feature, allowing riders and passengers to evaluate each other based on their experiences. This fosters a sense of accountability and builds a trustworthy community.

The platform’s Progressive Web Application (PWA) design ensures seamless usability across devices, allowing students to access the service from smartphones, tablets, and desktop computers. The PWA offers features like real-time ride tracking via Google Maps API, in-app notifications for ride updates, and messaging capabilities to facilitate communication between riders and passengers. These functionalities make the system intuitive and easy to use, ensuring high adoption rates among students.

Sustainability is a core focus of the proposed system. By promoting shared rides, the platform significantly reduces the number of vehicles on the road, leading to decreased carbon emissions and fuel consumption. The optimized route planning feature further contributes to environmental conservation by minimizing unnecessary travel. This aligns with the growing need for eco-friendly transportation solutions on college campuses, encouraging students to adopt sustainable practices in their daily commutes.

The backend of the platform is powered by robust technologies to ensure scalability and efficiency. A relational database like PostgreSQL is used to manage user profiles, ride details, routes, and payment histories, maintaining data consistency and integrity. Machine learning models, developed using Python libraries such as scikit-learn, handle clustering and optimization tasks, ensuring accurate and efficient ride-matching. The integration of advanced algorithms with modern development frameworks enables the platform to deliver a seamless and reliable experience.

The proposed system addresses the limitations of existing ride-sharing platforms by focusing exclusively on the needs of students. Its innovative approach to ride-matching, dynamic pricing, and user safety ensures a superior transportation experience tailored to campus life. By combining technological advancements with sustainable practices, the platform provides an efficient, cost-effective, and environmentally conscious solution that has the potential to transform transportation within college communities.

CHAPTER 2

2. LITERATURE SURVEY

1. Anas, Gunavathi, and Kirubasri explore a machine learning-based approach for carpooling that uses Natural Language Processing (NLP) to analyze users' social media data to classify personality types. By incorporating XGBoost, their model achieves 68% accuracy in matching users with similar personalities and geographic proximity. This research emphasizes improving user satisfaction in carpooling but highlights challenges like ensuring data privacy and increasing classification accuracy. It addresses the balance between user privacy and algorithmic performance. Their findings underline the importance of leveraging personal traits for better carpooling experiences. However, broader application demands enhanced accuracy and ethical handling of user data. This study sets a foundation for personality-driven carpooling optimization.

2. Jin's study focuses on a passenger-centric taxi carpooling model that prioritizes user preferences, including detour tolerance and route similarity. It integrates a matching algorithm with dynamic pricing to enhance partner selection and optimize routes. This approach reduces waiting times and costs, catering to personalized travel experiences. However, it notes difficulties in capturing diverse user preferences and avoiding mismatches. The model demonstrates the potential of incorporating user-centric features for better operational efficiency. It encourages dynamic and personalized approaches to improve user satisfaction. Challenges include accurate preference capture and ensuring compatibility among passengers.

3. Müller et al. propose a dynamic pricing model for shared mobility systems based on idle time data, increasing vehicle utilization and profitability by 11%. Their model focuses on maximizing operational efficiency in shared transportation platforms. It emphasizes using real-time analytics to adjust pricing strategies dynamically. The study identifies the need for accurate data and advanced analytical methods to implement this model effectively. While it boosts

profitability, it also demands robust computational resources. This research highlights the role of pricing in balancing supply-demand gaps.It paves the way for optimized financial models in shared mobility ecosystems.

4. Yan et al. investigate dynamic pricing and matching strategies for ride-hailing platforms to balance demand and supply. They focus on minimizing waiting times while maintaining platform reliability. Their approach uses complex algorithms to adjust pricing and wait strategies dynamically. However, longer wait times due to certain optimizations may frustrate riders. The study stresses real-time data usage for effective decision-making in shared mobility services. While achieving operational efficiency, the balance between rider convenience and platform objectives remains a challenge. This research provides insights into adapting to fluctuating market conditions in ride-hailing.

5. Zhou and Roncoli introduce a fairness-aware framework for dynamic ridesharing, emphasizing equitable pricing and passenger matching. Their approach reduces travel time and fuel consumption by using machine learning algorithms. It highlights the importance of fairness in ridesharing, addressing biases in conventional systems. Challenges include ensuring computational efficiency and handling diverse user requirements. This framework provides a sustainable and equitable approach to ridesharing. The study demonstrates how fairness can coexist with efficiency in shared transportation systems. It calls for integrating fairness as a key parameter in future ridesharing platforms.

6. Xu et al. propose TAROT, a privacy-preserving route matching scheme for carpooling services. This system employs advanced cryptographic techniques to secure sensitive user data, such as locations and routes. TAROT reduces computational costs and communication overheads by filtering dissimilar routes early. The study showcases the importance of privacy in route optimization for carpooling. While it ensures data security, scaling the system for broader use remains challenging. TAROT is a significant step toward balancing privacy and efficiency in carpooling platforms.

7. Peng and Zhou focus on optimizing carpooling paths by incorporating real-time traffic data and passenger satisfaction metrics like waiting times. Their model enhances efficiency by dynamically adjusting routes based on real-world conditions. Challenges include seamless integration of real-time data and user acceptance of optimized routes. The study underlines the importance of adaptability in ridesharing systems. By considering passenger comfort and operational feasibility, it paves the way for improved carpooling services. This research emphasizes real-time responsiveness to user needs and environmental conditions. Its practical applications require robust data handling and user engagement strategies.

8. Ramani Bai et al. propose a blockchain-based carpooling application using smart contracts to ensure secure and transparent transactions. Their decentralized model builds trust among users by eliminating intermediaries and enhancing data security. This approach tackles issues like fraudulent activities and data manipulation. Challenges include the high cost of blockchain implementation and ensuring scalability for larger platforms. The study highlights the role of decentralized technologies in improving carpooling systems. It emphasizes trust and security as essential components of modern transportation platforms. Their findings advocate the adoption of blockchain for reliable carpooling services.

9. Hashikami et al. develop a carpooling system in Japan that integrates accident location data to identify safer routes. They classify driver skills and optimize routes to minimize accident risks. This model improves safety while considering travel distance and time. Challenges include user coordination and adapting the system across different cultural contexts. The research highlights the importance of safety in carpooling, especially in high-risk areas. It emphasizes the need for real-time data and driver evaluations. This study contributes to safer and more reliable carpooling environments.

10. Alisoltani et al. propose a space-time clustering-based method to enhance real-time ride-sharing by grouping trips using a shareability function. Their approach

reduces computation time while maintaining high-quality trip matches. It focuses on scalability and efficiency in large networks, making it suitable for urban areas with high demand.The clustering technique minimizes the complexity of matching users in real-time scenarios. However, the method requires precise tuning of clustering parameters for optimal performance. The study demonstrates the potential of combining advanced clustering with real-time operations for better shareability. It offers significant insights into improving the scalability of carpooling services.

11. Meshkani and Farooq introduce GMOMatch, a graph-based ride-matching algorithm that optimizes shared trips. The algorithm iteratively matches riders to vehicles, reducing vehicle kilometers traveled (VKT) and traffic congestion. Tested in Toronto, the method significantly improves traffic flow and travel efficiency. However, the approach demands substantial computational resources and parameter optimization. It highlights the benefits of graph-based algorithms in optimizing carpooling services. By prioritizing sustainable shared mobility, the study addresses urban congestion challenges. It underlines the need for efficient algorithms in large-scale ride-sharing networks.

12. Seng et al. review the integration of ridesharing and crowdsourcing for smart cities, focusing on technological paradigms and use cases. They emphasize the potential of crowdsourcing to enhance transportation systems, combining community participation with advanced technology. The study highlights the role of ridesharing in building efficient, citizen-driven mobility solutions. It also identifies challenges like scalability, real-time responsiveness, and user trust. By exploring diverse use cases, it provides a comprehensive view of ridesharing in smart city environments. The findings encourage the adoption of innovative, collaborative approaches in urban transportation systems.

13. Shahi et al. conduct a comparative study of pathfinding algorithms for route planning in smart vehicular networks. They identify Contraction Hierarchies (CH) as highly effective for reducing travel time and improving vehicle speed.

CH's preprocessing demands are high but result in significant runtime efficiency. The study underlines the importance of robust algorithms in dynamically changing traffic conditions. By optimizing pathfinding, it contributes to the development of responsive, real-time carpooling systems. The research advocates the adoption of CH for large-scale transportation networks. It offers critical insights into balancing preprocessing overhead and operational efficiency.

14. Mitropoulos et al. systematically review ride-sharing platforms, examining user factors, barriers, and system designs. They identify key sociodemographic and regulatory challenges that influence user adoption and satisfaction. The study stresses the importance of addressing these barriers for successful implementation of ride-sharing solutions. By analyzing platform features and user behaviors, it provides actionable insights for enhancing user experience. It highlights the role of design and regulation in fostering ride-sharing adoption. The research encourages platforms to align their features with user preferences and societal norms. It offers a roadmap for addressing adoption challenges in shared mobility.

15. Adelé and Dionisio analyze smart carpooling apps like Karos, which use machine learning to predict user mobility patterns. These apps enhance ride matches by analyzing travel habits and preferences. The study identifies challenges in user understanding of app functionalities, which can impact satisfaction. By leveraging predictive analytics, it aims to improve user engagement and ride-matching efficiency. However, the findings emphasize the need for user education to maximize app benefits. The research bridges the gap between technical capabilities and user adoption in carpooling apps.

16. Bruglieri et al. introduce PoliUniPool, a carpooling system tailored for university communities. The system matches users based on schedules and preferences to minimize travel distances. It promotes sustainability and fosters trust within university networks. Challenges include coordinating schedules and overcoming

resistance to carpooling. The study highlights the role of community-specific platforms in enhancing user adoption. By focusing on shared values and trust, it builds a strong foundation for carpooling success. The research demonstrates the benefits of localized solutions for specific user groups.

17. Hasanuddin et al. explore the use of Progressive Web Apps (PWAs) for vehicle tracking systems to combat theft. The PWA provides real-time location tracking and alerts, accessible across multiple platforms. It highlights the benefits of cross-platform compatibility and reduced development costs. The study demonstrates how PWAs enhance accessibility and engagement, even with limited connectivity. It advocates for integrating PWA technology into carpooling systems for better user experiences. The findings emphasize the practicality of PWAs in modern transportation applications. This approach supports seamless and scalable carpooling solutions.

18. Wijaya et al. discuss the advantages of PWAs in vehicle tracking, focusing on offline capabilities and user-friendly interfaces. By enabling caching and real-time tracking, PWAs enhance operational reliability. The study underscores the role of PWAs in improving accessibility and user engagement in transportation systems. It advocates for the adoption of such technologies in carpooling applications for consistent user access. Challenges include integrating advanced features while maintaining simplicity and scalability. The research highlights the transformative potential of PWAs in transportation innovation. It provides a blueprint for leveraging web technologies in shared mobility.

19. Kumar et al. discuss a smart traffic system that utilizes IoT-based technologies to enhance vehicle safety and road efficiency. The system integrates real-time data from vehicles and road infrastructure to predict and prevent accidents. It also focuses on optimizing traffic flow by reducing congestion and enhancing travel efficiency. The study explores the role of IoT in building intelligent transportation systems that adapt dynamically to changing conditions. Challenges include ensuring data accuracy, scalability, and system reliability in

diverse traffic scenarios. The research underscores the importance of smart systems in improving road safety and traffic management. It sets a benchmark for integrating IoT in advanced carpooling solutions.

20. Senthil Pandi et al. focus on improving the classification accuracy of machine learning algorithms using hyperparameter optimization techniques. Their study emphasizes the importance of parameter tuning to enhance model performance. By applying optimization strategies, they achieve significant improvements in algorithm accuracy. This approach is particularly useful in applications like carpooling, where precise user matching is crucial. The research highlights the role of hyperparameter optimization in refining machine learning-based systems. It offers insights into building more robust and efficient models for practical applications. This study serves as a foundation for integrating advanced optimization techniques in carpooling platforms.

CHAPTER 3

3. SYSTEM DESIGN

3.1 GENERAL

3.1.1 SYSTEM FLOW DIAGRAM

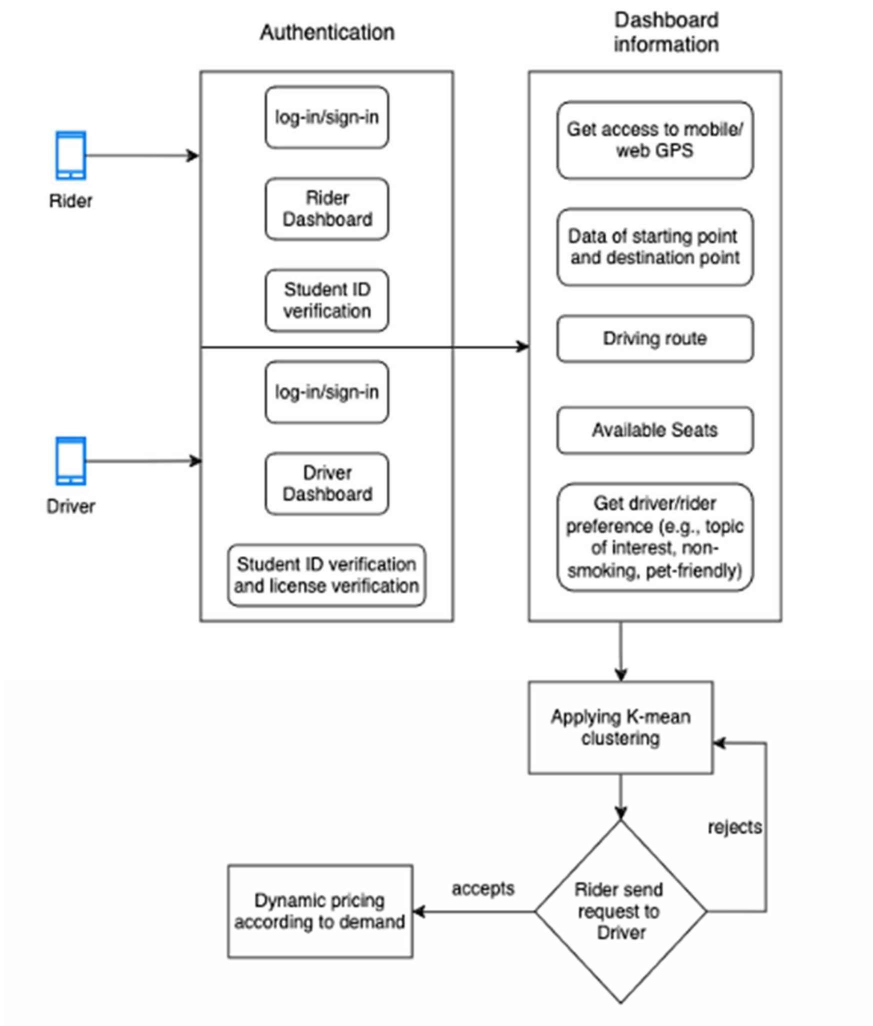

Figure 1

3.1.2 SEQUENCE DIAGRAM

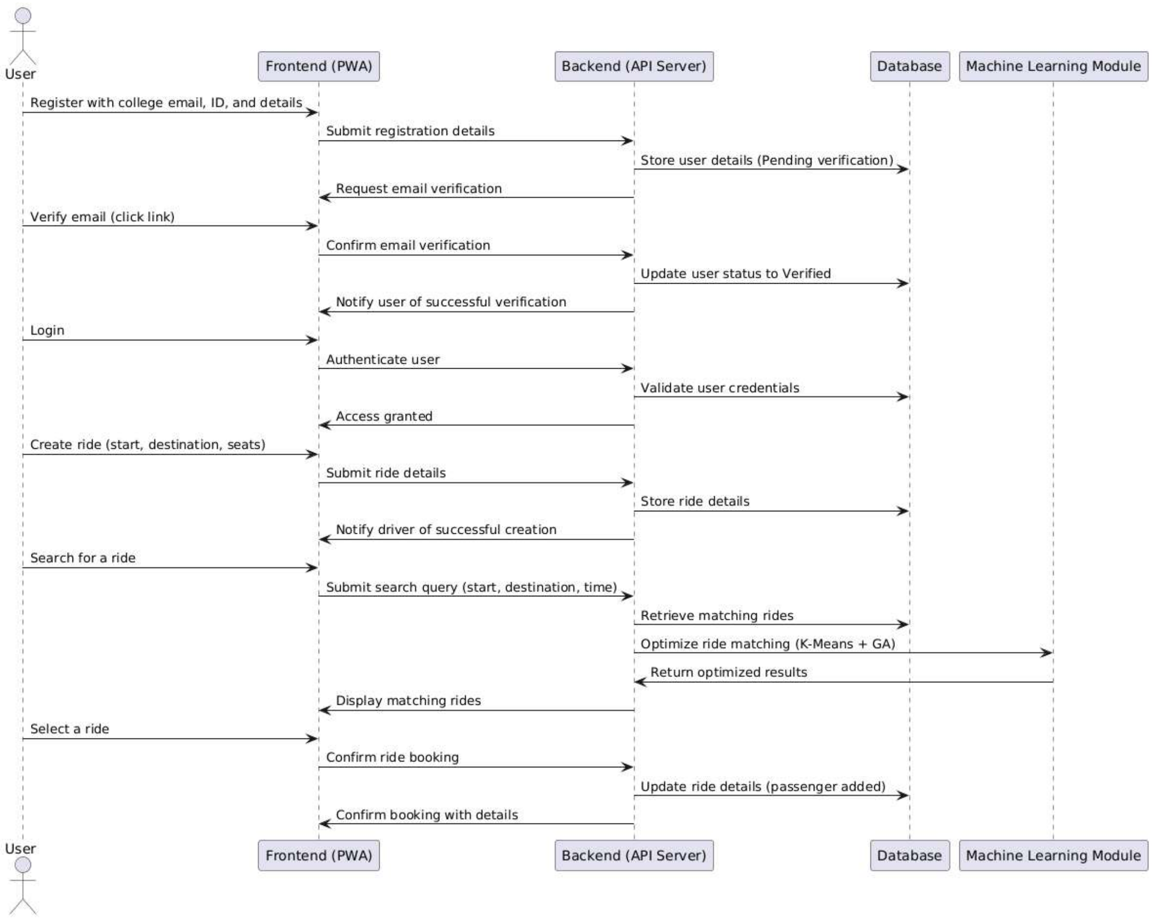

Figure 2

3.1.3 CLASS DIAGRAM

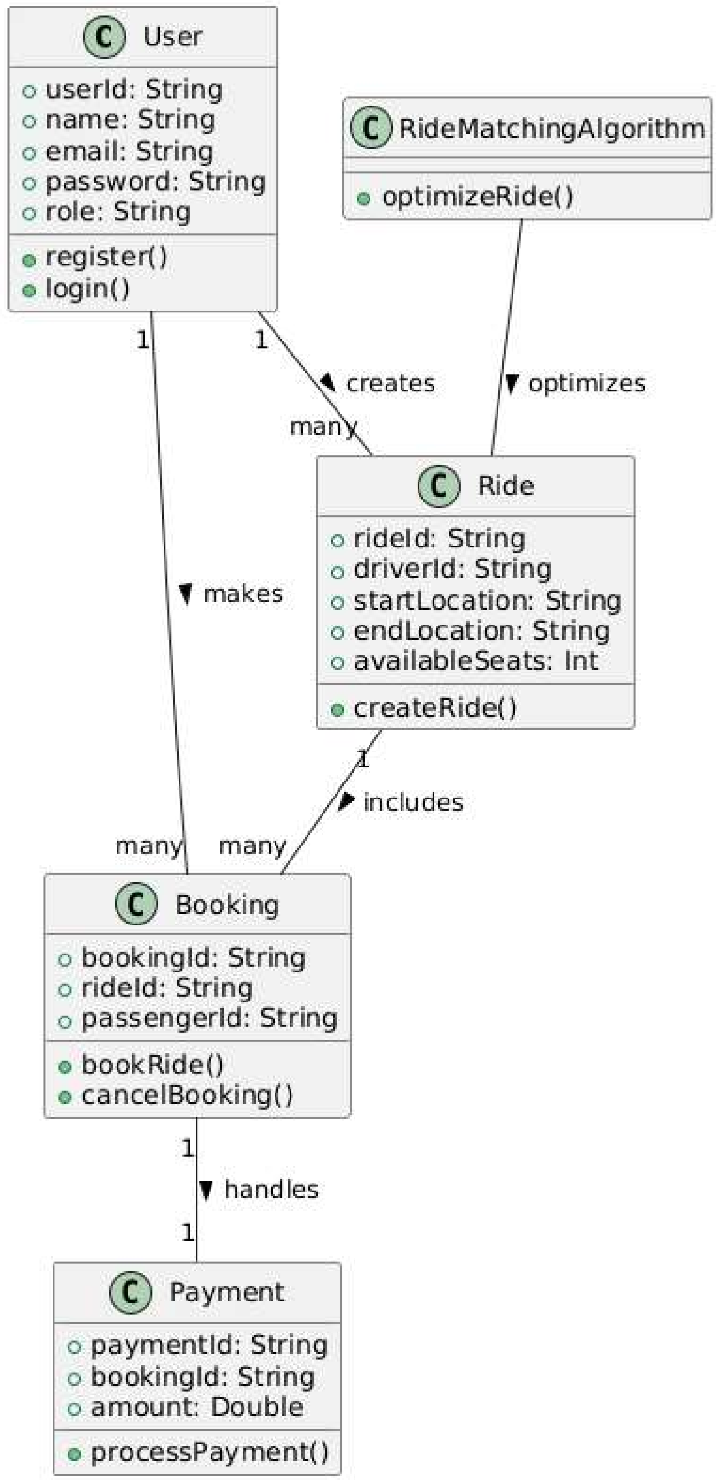

Figure 3

3.1.4 USE CASE DIAGRAM

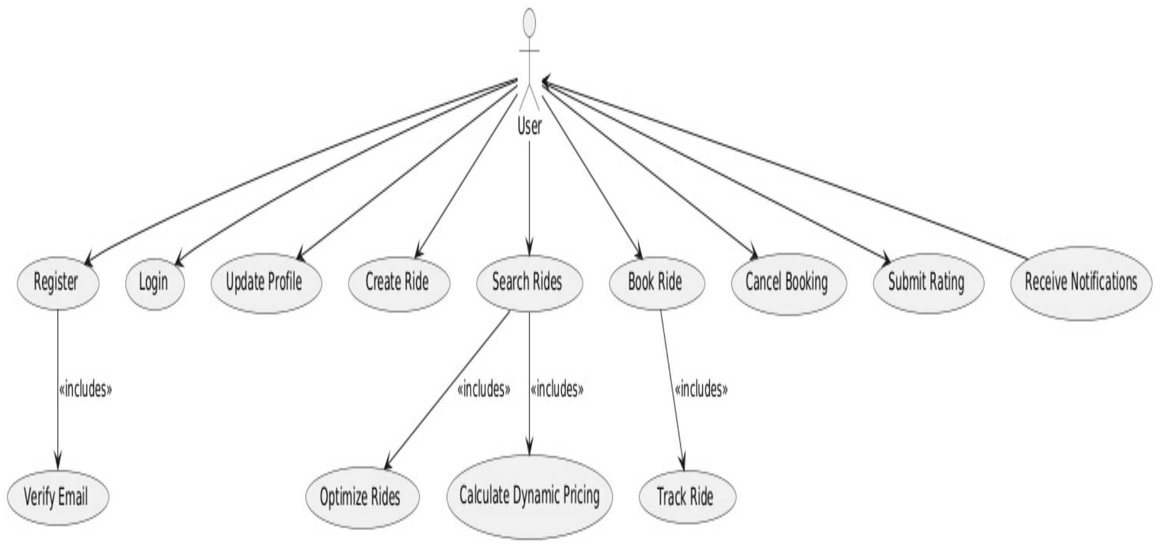

Figure 4

3.1.5 ARCHIETECTURE DIAGRAM

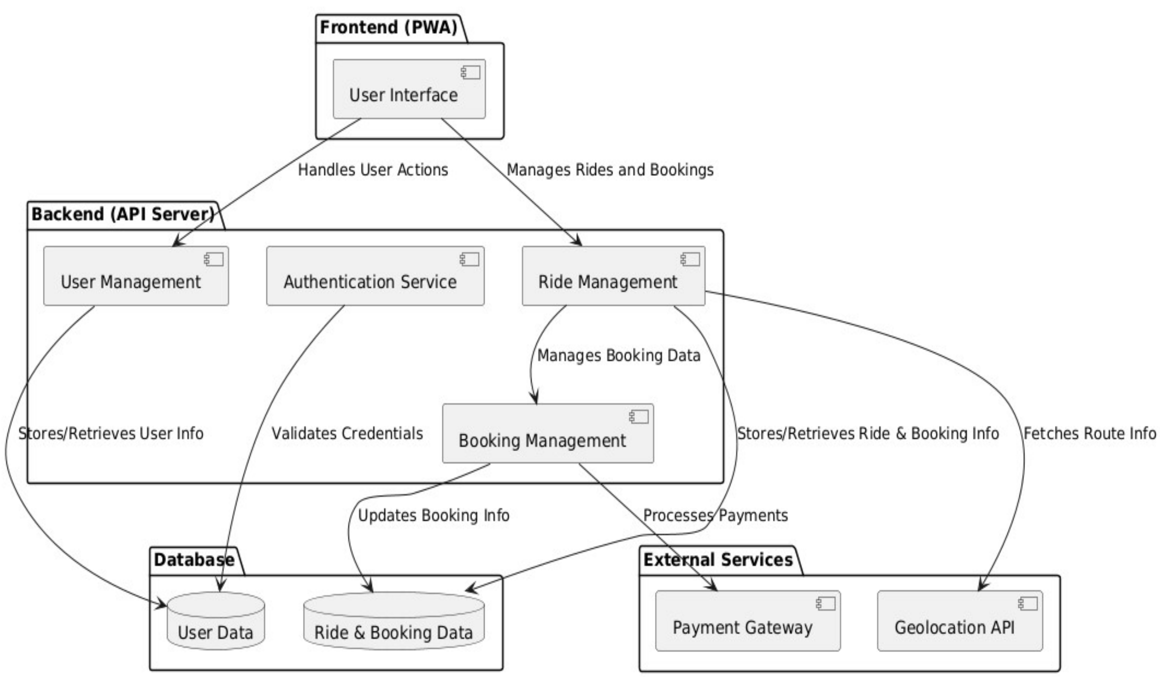

Figure 5

3.1.6 ACTIVITY DIAGRAM

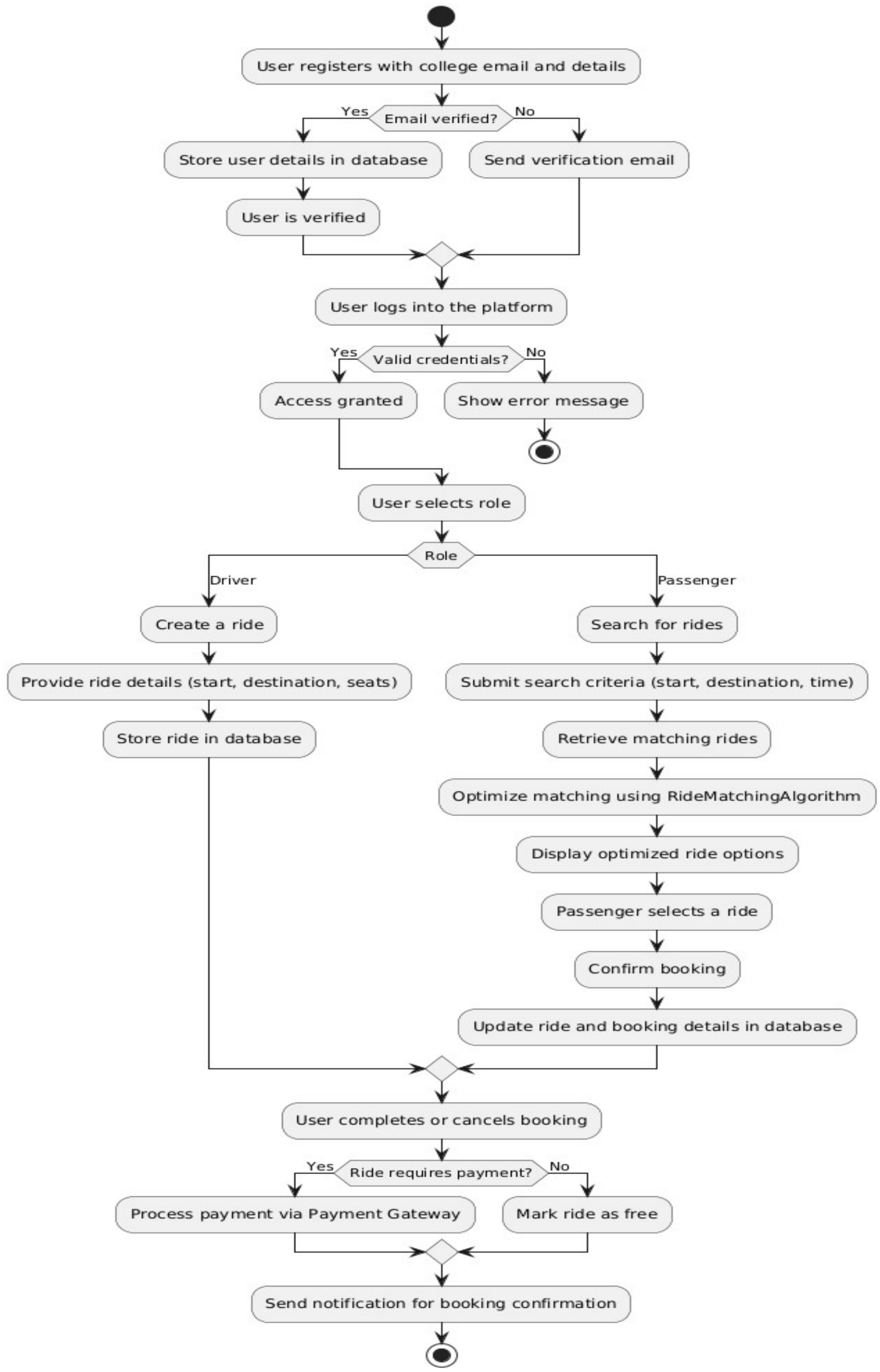

Figure 6

3.1.7 COMPONENT DIAGRAM

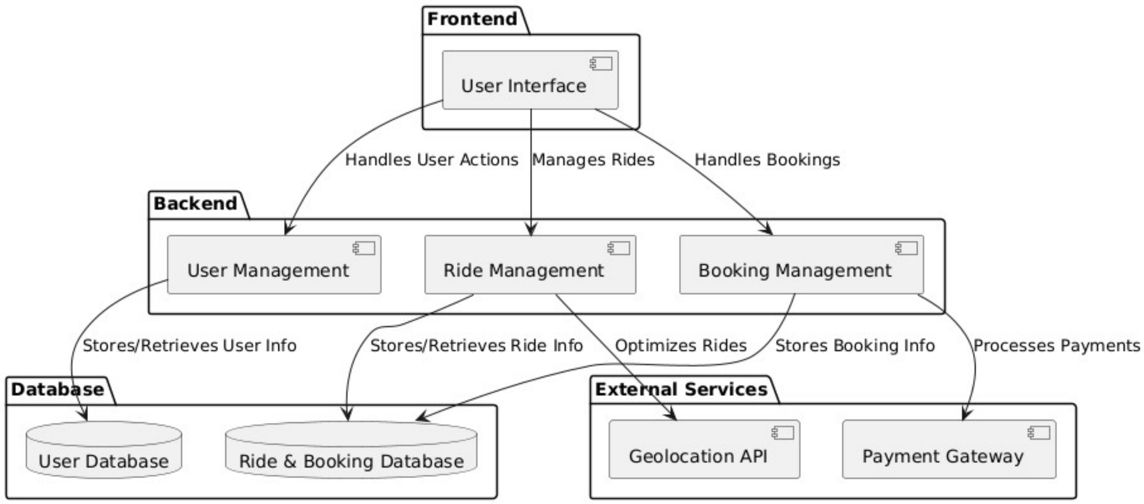

Figure 7

3.1.8 COLLABORATION DIAGRAM

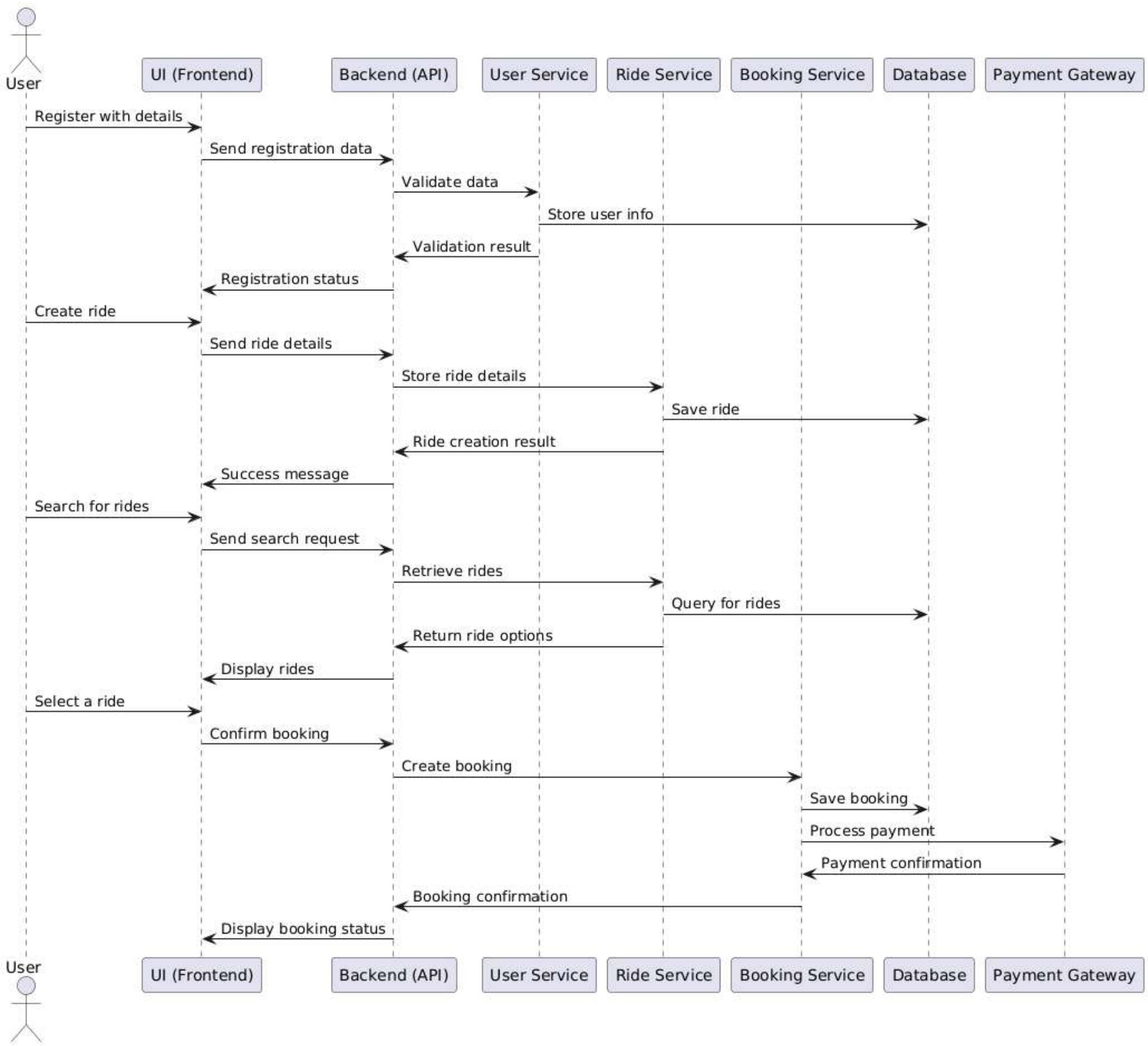

Figure 8

CHAPTER 4

PROJECT DESCRIPTION

4.1 METHODOLOGIES:

In this phase of the project, we focus on building the core features of the ride-sharing application, including the **architecture design**, **data collection**, and **preprocessing** steps required for the implementation of the **Hybrid Ridesharing Algorithm (HRA)**. The architecture follows a modular structure, utilizing the **MERN stack** (MongoDB, Express.js, React.js, Node.js) combined with **PWA (Progressive Web App)** capabilities to ensure smooth user experience across devices. The following sections provide detailed methodologies for each part of the system development, from the **architecture design** to **data collection and preprocessing** necessary for implementing the HRA.

1. Architecture Design of This App

The architecture of the ride-sharing application is built on the **MERN stack**, designed to support modularity, scalability, and cross-platform compatibility. The system follows a **client-server model**, where the frontend interacts with the backend via RESTful APIs. Key components of the architecture include:

- **Frontend**:
    - The frontend is built using **React.js**, ensuring a dynamic and responsive user interface. The application is developed as a **PWA** to support offline functionality, allowing users to interact with the app even without an active internet connection. This is particularly useful for drivers in areas with intermittent connectivity.
    - The frontend includes **driver** and **rider dashboards**, both integrated with a **map interface** powered by **Leaflet.js**, which enables route selection and matching features for the users.

- **Backend (Node.js with Express.js)**:
  - The **backend** is built using **Node.js** and **Express.js** to handle API requests and implement business logic. The backend is responsible for processing ride requests, storing ride and user information, and performing the ride matching logic.
  - The **authentication** system uses **JWT (JSON Web Tokens)** for secure user login and session management.

- **Database (MongoDB)**:
  - **MongoDB** is used to store and manage user profiles, ride information, and ride requests. The database structure is flexible, allowing the storage of unstructured data such as user details, ride locations, and dynamic data that may change over time (e.g., available seats).

- **PWA (Progressive Web App) Features**:
  - The **PWA** features ensure that the app is responsive, performs efficiently across devices, and provides a smooth experience even in low connectivity areas. This is achieved through service workers for caching and enabling offline access, along with an installable app that behaves like a native application.

This architecture ensures a seamless integration of frontend and backend services while maintaining scalability, flexibility, and offline accessibility.

Figure 9: Splash Screen

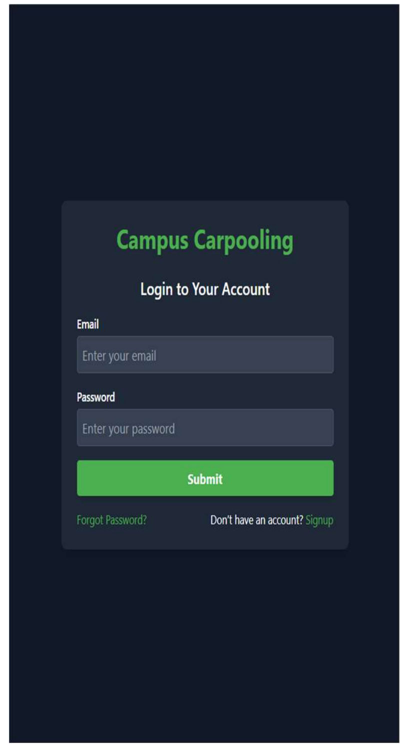

Figure 10: Login Page

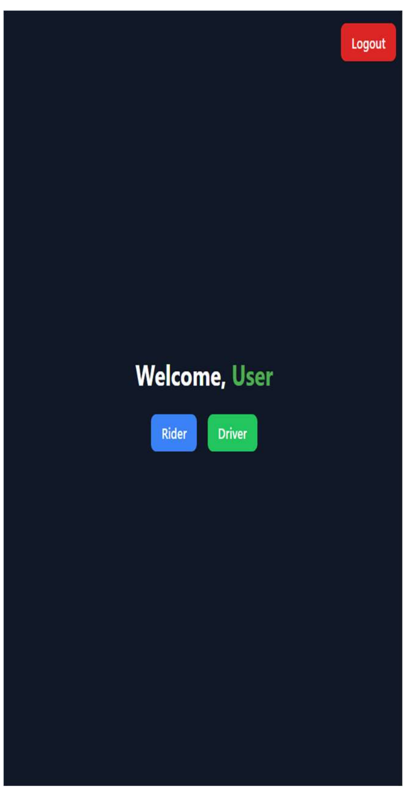

Figure 11: Home Page

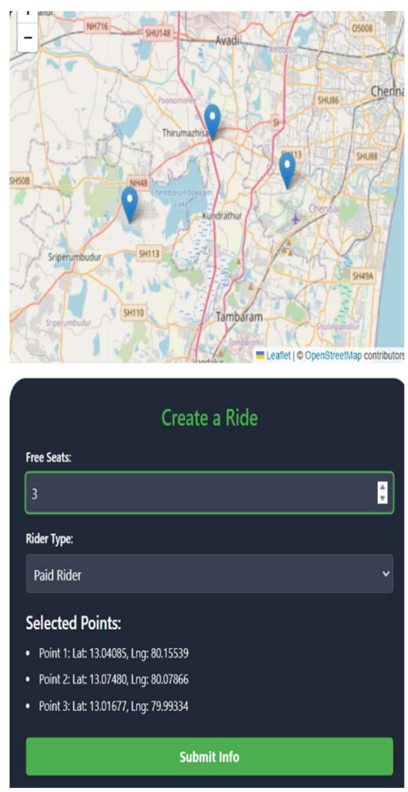

Figure 12: Driver Dashboard

2. Data Collection and Preprocessing for the Implementation of HRA

The **data collection and preprocessing** phase is a critical step for the effective implementation of the **Hybrid Ridesharing Algorithm (HRA)**, which combines **k-means clustering** and **genetic algorithm (GA)** for optimal ride matching. Here are the key steps involved in this phase:

1. **Data Collection**:
    - **Synthetic Data Generation**:
        - Since real-world data may be unavailable at this stage, **synthetic data** is generated to simulate realistic ridesharing scenarios. The synthetic data includes:
            - **Driver Data**: Each driver's starting and ending points (latitude and longitude), available seats, and route distances.
            - **Rider Data**: Each rider's pickup and drop-off locations (latitude and longitude).
            - **Distance Matrix**: The distances between each driver's route and each rider's pickup/drop-off point, which is essential for optimizing matches.
    - The data is generated using simple **randomization techniques** for location points within predefined geographic bounds (e.g., a city area) to simulate various ride and driver configurations.

2. **Data Preprocessing**:
    - **Data Validation and Cleaning**:
        - The collected data is validated to ensure consistency, such as ensuring no negative seat counts, verifying that the coordinates lie within the defined region, and removing any outliers (e.g., locations outside the city bounds).
        - The data is **normalized** to standardize all inputs (e.g., latitude/longitude values) and ensure compatibility with the k-means clustering algorithm.

- **Distance Calculation**:
  - The **distance matrix** is calculated based on the **Euclidean distance** between each rider's and driver's locations. This is essential for both the **k-means** clustering (to group riders and drivers based on proximity) and the **genetic algorithm** (to optimize the travel distances).
  - In future iterations, this can be replaced with **real-world distance calculations** (e.g., using Google Maps or OpenRouteService) to better reflect travel times and route feasibility.

3. **Feature Engineering**:
   - **Clustering Features**:
     - The **start and end points** of both drivers and riders are used to calculate proximity, forming the basis for **k-means clustering**.
   - **Additional Features**:
     - Calculated features such as **route distance** for drivers and **ride distance** for riders enhance the algorithm's ability to prioritize matches.

RAW DATA


```
{
  "drivers": [
    {
      "driver_id": "driver_1",
      "start_point": { "lat": 12.9507, "lng": 80.2276 },
      "end_point": { "lat": 12.9724, "lng": 80.2453 },
      "seats_available": 3
    },
    {
      "driver_id": "driver_2",
      "start_point": { "lat": 12.9205, "lng": 80.2150 },
      "end_point": { "lat": 13.0012, "lng": 80.2798 },
      "seats_available": 2
    }
  ],
  "riders": [
    {
      "rider_id": "rider_1",
      "pickup_point": { "lat": 12.9654, "lng": 80.2407 },
      "dropoff_point": { "lat": 12.9724, "lng": 80.2453 }  // Matches driver_1's
destination
    },
    {
      "rider_id": "rider_2",
      "pickup_point": { "lat": 12.9386, "lng": 80.2234 },
      "dropoff_point": { "lat": 13.0012, "lng": 80.2798 }  // Matches driver_2's
destination
    }
  ],
  "distance_matrix": [
    {
      "driver_id": "driver_1",
      "rider_id": "rider_1",
      "pickup_distance": 1.2,
      "dropoff_distance": 0
    },
    {
      "driver_id": "driver_2",
      "rider_id": "rider_2",
      "pickup_distance": 0.8,
      "dropoff_distance": 0
    }
  ]
}
```

PREPROCESSED DATA


```
{
 "drivers": [
   {
    "driver_id": "driver_1",
    "start_lat": 12.9507,
    "start_lng": 80.2276,
    "end_lat": 12.9724,
    "end_lng": 80.2453,
    "seats_available": 3,
    "route_distance": 3.0  // Precomputed distance between start and end points
   },
   {
    "driver_id": "driver_2",
    "start_lat": 12.9205,
    "start_lng": 80.2150,
    "end_lat": 13.0012,
    "end_lng": 80.2798,
    "seats_available": 2,
    "route_distance": 5.0
   }
 ],
 "riders": [
   {
    "rider_id": "rider_1",
    "pickup_lat": 12.9654,
    "pickup_lng": 80.2407,
    "dropoff_lat": 12.9724,
    "dropoff_lng": 80.2453,
    "ride_distance": 1.2  // Simplified as pickup distance (dropoff distance is 0)
   },
   {
    "rider_id": "rider_2",
    "pickup_lat": 12.9386,
    "pickup_lng": 80.2234,
    "dropoff_lat": 13.0012,
    "dropoff_lng": 80.2798,
    "ride_distance": 0.8
   }
 ],
```

```
  "distance_matrix": [
    {
      "driver_id": "driver_1",
      "rider_id": "rider_1",
      "pickup_distance": 1.2,
      "dropoff_distance": 0,
      "total_distance": 1.2  // Equal to pickup distance
    },
    {
      "driver_id": "driver_2",
      "rider_id": "rider_2",
      "pickup_distance": 0.8,
      "dropoff_distance": 0,
      "total_distance": 0.8  // Equal to pickup distance
    }
  ]
}
```


4.1.1 RESULT DISCUSSION

In Phase 1 of the project, the primary focus was on setting up the foundational elements for the ride-sharing application, particularly the data collection and preprocessing required to implement the Hybrid Ridesharing Algorithm (HRA). This phase involved gathering synthetic data for drivers and riders, and processing it into a structured format suitable for the HRA.

1. Results

1. Data Collection:
    - Synthetic data was generated for a small but representative sample of drivers and riders. This data included the key attributes such as:
        - Driver route (starting and ending points).
        - Rider locations (pickup and drop-off points).
        - Seat availability for drivers.
    - A distance matrix was computed based on simple Euclidean distance between pickup and drop-off points for both drivers and riders.
2. Preprocessing:
    - The preprocessing step successfully transformed the raw data into a structured format, making it easier to feed into the Hybrid Ridesharing Algorithm (HRA) in the future.

- Features like pickup distance, drop-off distance, and total distance were derived from the raw data, preparing the dataset for the next phase of the project.

3. Data Format:
   - The structured data format, including both raw data and preprocessed data, was finalized and is now ready to be used in subsequent algorithm implementations. This ensures that all necessary features (e.g., distance, seat availability) are available for ride matching once the algorithm is implemented.

2. Discussion

1. Importance of Data Preprocessing:
   - Data preprocessing is a critical step in the project because it ensures that the input data is clean, consistent, and ready for use in the HRA.
   - By transforming the raw data into a structured format with relevant features (e.g., total distance, pickup distance, ride distance), we have ensured that the HRA will be able to efficiently compute the best matches for riders and drivers.
2. Impact of Removing Ride Preferences:
   - The decision to exclude ride preferences (such as "paid" and "free" rides) in Phase 1 simplified the data collection and preprocessing steps. It allowed us to focus on core features such as proximity and seat availability, which are central to the ride matching process.
   - In future phases, ride preferences can be reintroduced to make the matching process more sophisticated, catering to different user needs (e.g., matching drivers with paid riders and ensuring availability for riders who prefer free rides).
3. Preparation for the Hybrid Ridesharing Algorithm:
   - The preprocessed data now contains all necessary features (e.g., distances, available seats) for the HRA. The next step will involve clustering the data (using k-means) and optimizing matches using the genetic algorithm.
   - This foundational work lays the groundwork for Phase 2, where the actual matching algorithm will be developed and tested.

CHAPTER 5

5.1 CONCLUSION AND WORKSPACE

Phase 1 of the project laid the foundational groundwork for the development of the ride-sharing application by focusing on the essential tasks of **data collection** and **preprocessing**. This phase did not include the actual implementation of the **Hybrid Ridesharing Algorithm (HRA)** but instead focused on preparing the data that will later be used for the algorithm's execution. By generating **synthetic data** for drivers and riders and preprocessing it into a structured format, we have ensured that the dataset contains all the necessary features to enable efficient ride matching. These features include **pickup distances**, **drop-off distances**, and **route distances**, which are essential for the matching algorithm in later stages.

The **MERN stack** was used for its flexibility and scalability, allowing us to build both the backend and frontend components of the application efficiently. The **MongoDB** database was selected for its ability to handle unstructured data, ensuring that we could easily store and retrieve dynamic ride-related information.

Looking ahead to Phase 2, the **preprocessed data** is now ready to be utilized in the implementation of the **Hybrid Ridesharing Algorithm**. The next step will involve incorporating advanced algorithms like **k-means clustering** and the **genetic algorithm** to optimize the matching process between drivers and riders. While Phase 1 focused on the **data aspect**, Phase 2 will leverage this prepared dataset to drive the system's functionality and improve ride matching efficiency. Furthermore, this phase will also involve enhancing the algorithm to account for factors such as ride preferences, real-time traffic data, and other dynamic variables.

In conclusion, Phase 1 successfully set the stage for the ride-sharing application's future development by preparing a robust, preprocessed dataset and laying the foundation for the implementation of the matching algorithm. The next phases will build upon this foundation to enhance the system's performance and provide a more user-centric and efficient ride-sharing experience.

5.2 FOR PHASE 2

In Phase 2 of the project, the focus will shift to implementing the **Hybrid Ridesharing Algorithm (HRA)**, which is the core component for optimizing ride matching between drivers and riders. The HRA will combine **k-means clustering** to group nearby riders and drivers and **genetic algorithms (GA)** to optimize the allocation of rides based on factors like seat availability, route compatibility, and travel distance. The algorithm will dynamically calculate the most efficient ride-sharing matches by considering proximity and minimizing total travel distances, ensuring an optimized, cost-effective, and convenient ride for both drivers and riders. Additionally, real-time data such as **traffic conditions** and **route deviations** will be incorporated into the algorithm to further enhance its accuracy and responsiveness.

Along with the **HRA**, Phase 2 will also focus on the implementation of **dynamic pricing simulation**. This feature will enable the app to adjust pricing in real-time based on various factors such as demand, ride distance, time of day, and traffic conditions. By using machine learning models or rule-based systems, dynamic pricing will allow the app to optimize revenue for drivers while maintaining fairness for riders. The simulation will include testing different pricing models, including **surge pricing** and **discounts for long rides**, to determine the most effective strategy for balancing supply and demand. These features will significantly improve the app’s functionality and user experience, making it more competitive and scalable in the real-world market.

In Phase 2, in addition to implementing the **Hybrid Ridesharing Algorithm (HRA)** and **dynamic pricing simulation**, the app will be enhanced with full **PWA (Progressive Web App)** features. This includes **offline capabilities** using **service workers** to cache critical data like ride details and driver availability, ensuring the app functions without an internet connection. The app will also be **installable** on users' home screens, offering a native app-like experience on both mobile and desktop. **Push notifications** will be integrated to provide real-time updates on ride statuses, driver availability, and pricing changes. These features will ensure a seamless, responsive experience across devices, increasing user engagement and accessibility.

5.3 REFERENCES

1. Anas, Mohd, C. Gunavathi, and G. Kirubasri. "Machine learning based personality classification for carpooling application." 2023 International Conference on Intelligent Systems for Communication, IoT and Security (ICISCoIS). IEEE, 2023.

2. Jin, Dian. "Research on Taxi-Carpooling Matching Schemes Considering Passenger Personalized Needs." 2024 7th International Conference on Advanced Algorithms and Control Engineering (ICAACE). IEEE, 2024.

3. Müller, Christian, et al. "Dynamic pricing for shared mobility systems based on idle time data." OR Spectrum 46.2 (2024): 411-444.

4. Yan, Chiwei, et al. "Dynamic pricing and matching in ride-hailing platforms." Naval Research Logistics (NRL) 67.8 (2020): 705-724.

5. Zhou, Ze, and Claudio Roncoli. "A fairness-aware joint pricing and matching framework for dynamic ridesharing." 2023 8th International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS). IEEE, 2023.

6. Xu, Qi, et al. "An efficient and privacy-preserving route matching scheme for carpooling services." IEEE Internet of Things Journal 9.20 (2022): 19890-19902.

7.

8. Peng, Jiarun, and Hongmei Zhou. "Matching path optimization of carpooling mode for private cars considering real-time traffic conditions and passenger satisfaction." 2023 7th International Conference on Transportation Information and Safety (ICTIS). IEEE, 2023.

9. Sudhir, Pranav, et al. "Smart Contract Based Carpooling Application for Secure and Efficient Ride Sharing." 2023 IEEE International Conference on Recent Advances in Systems Science and Engineering (RASSE). IEEE, 2023.

10. Hashikami, Hidenobu, et al. "Safe Route Carpooling to Avoid Accident Locations and Small-Scale Proof of Concept in Japan." IEEE Transactions on Systems, Man, and Cybernetics: Systems 53.7 (2023): 4239-4250.

11. Zhu, Congcong, et al. "Time-optimal and privacy preserving route planning for carpool policy." World Wide Web 25.3 (2022): 1151-1168.

12. Alisoltani, Negin, et al. "Space-time clustering-based method to optimize shareability in real-time ride-sharing." Plos one 17.1 (2022): e0262499.

13. Seng, Kah Phooi, et al. "Ridesharing and crowdsourcing for smart cities: technologies, paradigms and use cases." IEEE Access 11 (2023): 18038-18081.

14. Shahi, Gurpreet Singh, Ranbir Singh Batth, and Simon Egerton. "A comparative study on efficient path finding algorithms for route planning in smart vehicular networks." International Journal of Computer Networks and Applications 7.5 (2020): 157-166.

15. Mitropoulos, Lambros, Annie Kortsari, and Georgia Ayfantopoulou. "A systematic literature review of ride-sharing platforms, user factors and barriers." European Transport Research Review 13 (2021): 1-22.

16. Adelé, Sonia, and Corinne Dionisio. "Learning from the real practices of users of a smart carpooling app." European Transport Research Review 12 (2020): 1-14.

17. Bruglieri, Maurizio, et al. "PoliUniPool: a carpooling system for universities." Procedia-Social and Behavioral Sciences 20 (2011): 558-567.

18. Acharya, Rhea, Jessica Chen, and Helen Xiao. "Uber Stable: Formulating the Rideshare System as a Stable Matching Problem." arXiv preprint arXiv:2403.13083 (2024).

19. Hasanuddin, Muhammad Ogin, et al. "Implementation of Progressive Web App-Based Applications for Vehicle Tracking Systems." 2022 12th International Conference on System Engineering and Technology (ICSET). IEEE, 2022.

20. Kumar, P., Vinodh Kumar, S., Priya, L. (2023). Smart and Safety Traffic System for the Vehicles on the Road. In: Choudrie, J., Mahalle, P., Perumal, T., Joshi, A. (eds) IOT with Smart Systems. Smart Innovation, Systems and Technologies, vol 312. Springer, Singapore. https://doi.org/ 10.1007/978-981-19-3575-6_51.

21. S. Senthil Pandi, V. R. Chiranjeevi, K. T and Kumar P, Improvement of Classification Accuracy in Machine Learning Algorithm by Hyper Parameter Optimization(2023), 2023 International Conference on Research Methodologies in Knowledge Management, Artificial Intelligence and Telecommunication Engineering (RMKMATE), Chennai, India, 2023, pp. 1-5, doi: 10.1109/RMKMATE59243.2023.10369177