COMMUNITY CARPOOLING SYSTEM

A Report submitted under Project-Based Learning

In Partial Fulfillment of the Course Requirements for
"Web Technologies (22IT104001)"

Submitted By

Mangireddygari Reshma	22102A040916
Kukkati Nikhitha	22102A040884
Kuniti Prabhavati	22102A040887
Mallineni Nandini	22102A040910
Majjiga Sai Kavya	22102A040907

Under the Guidance of

M.Surya

Assistant Professor

Department of CSE

MBU
MOHAN BABU
UNIVERSITY
DREAM. BELIEVE. ACHIEVE

Department of Computer Science and Engineering

School of Computing

MOHAN BABU UNIVERSITY

Sree Sainath Nagar, Tirupati – 517 102

2024-2025

MOHAN BABU UNIVERSITY

Vision

To be a globally respected institution with an innovative and entrepreneurial culture that offers transformative education to advance sustainability and societal good.

Mission

- Develop industry-focused professionals with a global perspective.
- Offer academic programs that provide transformative learning experience founded on the spirit of curiosity, innovation, and integrity.
- Create confluence of research, innovation, and ideation to bring about sustainable and socially relevant enterprises.
- Uphold high standards of professional ethics leading to harmonious relationship with environment and society.

SCHOOL OF COMPUTING

Vision

To lead the advancement of computer science research and education that has real-world impact and to push the frontiers of innovation in the field.

Mission

- Instil within our students fundamental computing knowledge, a broad set of skills, and an inquisitive attitude to create innovative solutions to serve industry and community.

- ❖ Provide an experience par excellence with our state-of-the-art research, innovation, and incubation ecosystem to realise our learners' fullest potential.
- ❖ Impart continued education and research support to working professionals in the computing domain to enhance their expertise in the cutting-edge technologies.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Vision

To become a Centre of Excellence in Computer Science and its emerging areas by imparting high quality education through teaching, training and research.

Mission

- ➢ Imparting quality education in Computer Science and Engineering and emerging areas of IT industry by disseminating knowledge through contemporary curriculum, competent faculty and effective teaching-learning methodologies.
- ➢ Nurture research, innovation and entrepreneurial skills among faculty and students to contribute to the needs of industry and society.
- ➢ Inculcate professional attitude, ethical and social responsibilities for prospective and promising engineering profession.
- ➢ Encourage students to engage in life-long learning by creating awareness of the contemporary developments in Computer Science and Engineering and its emerging areas.

B.Tech. Computer Science and Engineering

PROGRAM EDUCATIONAL OBJECTIVES

After few years of graduation, the graduates of B.Tech. CSE will be:

PEO1. Pursuing higher studies in core, specialized or allied areas of Computer Science, or Management.

PEO2. Employed in reputed Computer and I.T organizations or Government to have a globally competent professional career in Computer Science and Engineering domain or be successful Entrepreneurs.

PEO3. Able to demonstrate effective communication, engage in teamwork, exhibit leadership skills and ethical attitude, and achieve professional advancement through continuing education.

PROGRAM OUTCOMES

On successful completion of the Program, the graduates of B.Tech. CSE Program will be able to:

PO1. **Engineering Knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2. **Problem Analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3. **Design/Development of Solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4. **Conduct Investigations of Complex Problems**: Use research-based knowledge and research methods including design of experiments, analysis

and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5. **Modern Tool Usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6. **The Engineer and Society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7. **Environment and Sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9. **Individual and Team Work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11. **Project Management and Finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12. **Life-long Learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES

On successful completion of the Program, the graduates of B. Tech. (CSE) program will be able to:

PSO1. Apply knowledge of computer science engineering, Use modern tools, techniques and technologies for efficient design and development of computer-based systems for complex engineering problems.

PSO2. Design and deploy networked systems using standards and principles, evaluate security measures for complex networks, apply procedures and tools to solve networking issues.

PSO3. Develop intelligent systems by applying adaptive algorithms and methodologies for solving problems from inter-disciplinary domains.

PSO4. Apply suitable models, tools and techniques to perform data analytics for effective decision making.

Course Code	Course Title	L	T	P	S	C
22IT104001	**WEB TECHNOLOGIES**	3	-	2	4	5

COURSE OUTCOMES: *After successful completion of this course, the students will be able to:*

CO1. Demonstrate knowledge on web page design elements, dynamic content and database connection.

CO2. Analyze user requirements to develop web applications.

CO3. Design client-server applications using web technologies.

CO4. Demonstrate problem solving skills to develop enterprise web applications.

CO5. Apply HTML, CSS, JavaScript, JQuery, Bootstrap and PHP technologies for device independent web application development.

CO6. Apply web technologies to develop interactive, dynamic and scalable web applications for societal needs.

CO-PO-PSO Mapping Table:

Course Outcomes	Program Outcomes												Program Specific Outcomes			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO1	3	3	2	-	-	-	-	-	-	-	-	-	3	2	3	-
CO2	3	3	3	2	2	-	-	-	-	-	-	-	3	2	3	-
CO3	3	3	3	2	2	-	-	-	-	-	-	-	3	2	3	-
CO4	3	3	3	2	2	-	-	-	-	-	-	-	2	2	3	-
CO5	3	2	2	2	2	3	-	-	-	-	-	-	2	2	3	-
CO6				2					3	3						
Course Correlation Mapping	3	3	3	2	2				3	3			3	2	3	

Correlation Levels: 3: High; 2: Medium; 1: Low

MOHAN BABU UNIVERSITY
Sree Sainath Nagar, Tirupati 517 102

Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the Project Entitled

COMMUNITY CARPOOLING SYSTEM

Submitted By

Mangireddygari Reshma	22102A040916
Kukkati Nikhitha	22102A040884
Kuniti Prabhavati	22102A040887
Mallineni Nandini	22102A040910
Majjiga Sai Kavya	22102A040907

is the work submitted under Project-Based Learning in Partial Fulfillment of the Course Requirements for "Web Technologies (22IT104001)" during 2024-2025.

Supervisor:

Surya
Assistant Professor
Department of CSE
School of Computing
Mohan Babu University
Tirupati.

Head:

Dr. G. Sunitha
Professor & Head
Department of CSE
School of Computing
Mohan Babu University
Tirupati.

ACKNOWLEDGEMENTS

First and foremost, I extend my sincere thanks to **Dr. M. Mohan Babu**, Chancellor, for his unwavering support and vision that fosters academic excellence within the institution.

My gratitude also goes to **Mr. Manchu Vishnu, Pro-Chancellor**, for creating an environment that promotes creativity and for his encouragement and commitment to student success.

I am deeply appreciative of **Prof. Nagaraj Ramrao**, Vice Chancellor, whose leadership has created an environment conducive to learning and innovation.

I would like to thank **Dr. K. Saradhi**, Registrar, for his support in creating an environment conducive to academic success.

My sincere thanks to **Dr. B.M. Satish**, Dean of the School of Engineering and Computing, for his valuable support and guidance in all academic matters.

I am also grateful to **Dr. G. Sunitha**, Head of the Department of Computer Science and Engineering, for her valuable insights and support.

Finally, I would like to express my deepest appreciation to my project supervisor, Surya, Assistant Professor, Department of Computer Science and Engineering for continuous guidance, encouragement, and expertise throughout this project.

Thank you all for your support and encouragement.

TABLE OF CONTENTS

TOPICS	Page No.
ABSTRACT	1
INTRODUCTION	2-3
SOFTWARE REQUIREMENTS SPECIFICATION	4-7
ANALYSIS	8-11
ER DIAGRAM	12-14
TESTING	15-17
OUTPUT SCREENS	18-24
SQL CONNECTIONS AND DATABASE	25-27
ANALYSIS OF RESULT	28
ADVANTAGES AND DISADVANTAGES	28-30
FUTURE SCOPE OF ONLINE CARPOOLING SYSTEM	30-31
LIMITATION OF PROJECT	32-33
CONCLUSION	34
REFERENCES	35

LIST OF FIGURES

S.NO	FIGURE NAME	PAGE NO.
1	0-LEVEL DFD	9
2	1-LEVEL DFD	10
3	2-LEVEL DFD	11
4	COMMUNITY CARPOOLING SYSTEM ER DIAGRAMS	12
5	OUTPUT SCREENS	18-24
6	SQL DATABASE AND TABLES	25-27

ABSTRACT

This project presents the development of an innovative web-based application titled "**Community Carpool System**," crafted to streamline ride-sharing within local communities. By harnessing cutting-edge web technologies such as HTML, CSS, Bootstrap, JavaScript, and the server-side scripting power of PHP, the application offers an intuitive interface and robust database management. The primary goal of the Community Carpool System is to establish a platform that not only mitigates traffic congestion and reduces transportation costs but also fosters a sense of community by facilitating convenient ride-sharing.

Key features include user registration, profile creation, and information verification to ensure safety and trust. The platform empowers users to post available rides, browse through listings, and book or request rides from fellow community members. A sophisticated ride-matching algorithm intelligently pairs riders with drivers based on location, timing, and seat availability. Moreover, the system delivers real-time notifications and updates regarding ride status, changes, and cancellations, thus enhancing the overall user experience.

The development process adopted a comprehensive approach encompassing requirements gathering, meticulous system design, implementation, and rigorous testing. Testing validated the application's functionality and performance, proving its effectiveness in addressing the community's transportation needs.

This project not only contributes to the knowledge of web development practices but also champions sustainability and community engagement. Future enhancements may include integrating a user rating and review system and expanding the platform to host community events like carpool meet-ups.

The Community Carpool System stands as a practical and innovative solution to urban transportation challenges, encouraging collaboration and resource-sharing among community members, while promoting a more connected and eco-friendly way of commuting.

Introduction

Problem Statement

Urban transportation challenges are growing, with many individuals struggling to find affordable, convenient, and eco-friendly options. The reliance on single-passenger vehicles contributes to traffic congestion, pollution, and high fuel costs. Despite many commuters traveling along similar routes, there is a lack of platforms to connect them for shared transportation.

This results in limited sustainable commuting options, increased carbon footprint, and higher costs for individuals. A dedicated community-based carpool platform could address these issues by helping people easily find, offer, or join carpool rides, reduce expenses, and promote environmental sustainability.

The Community Carpool System aims to improve access to affordable transportation, reduce congestion, and foster a sense of community.

Objectives

The Community Carpool System aims to enhance transportation, sustainability, and social connectivity by achieving the following objectives

1. Promote Sustainable Transportation: Reduce vehicles on the road, lowering carbon emissions.

2. Reduce Traffic Congestion: Alleviate congestion, improving travel times and reducing stress.

3. Lower Transportation Costs: Share travel expenses to make commuting more affordable.

4. Improve Accessibility: Provide affordable transport options for those without personal vehicles.

5. Build Social Connections: Foster community interaction and new friendships through shared rides.

6. Enhance Safety and Trust: Ensure a secure platform with verification and feedback features.

7. Reduce Parking Demand: Decrease the need for parking space in high-density areas.

Features

User Side

Features of Community Carpooling System

User Side

❖ User Registration: New users can register for the Community Carpooling System.

❖ User Login: Users can log in to the system using their email ID and password.

❖ Profile Management: Users can view and update their profile details.

❖ Post a Ride: Users can post a ride, providing details such as pickup location, drop-off location, departure time, and available seats.

❖ Book a Ride: Users can browse available rides and book a ride that fits their needs.

❖ Ride History: Users can view their past and upcoming rides.

❖ Feedback and Rating: Users can leave feedback and rate their ride experiences.

❖ View Feedback: Users can view the feedback and ratings they have received from other users.

Admin Side

❖ User Management: Admins can manage user registrations, including approving or rejecting new registrations.

❖ Ride Management: Admins can view, edit, or remove posted rides to ensure compliance with community guidelines.

❖ Profile Viewing: Admins can view all user profiles and their ride histories.

❖ Feedback Management: Admins can monitor and address feedback and complaints.

❖ Report Generation: Admins can generate reports on system usage, ride bookings, and user activities.

❖ System Security: Admins are responsible for maintaining the security of the platform by monitoring and updating security measures.

Software Requirements Specification

The main objective of the Community Carpooling System is to provide a platform that facilitates ride-sharing within a local community. It aims to reduce traffic congestion, lower transportation costs, and foster a sense of community by enabling users to share rides conveniently. This system will streamline the process of posting, booking, and managing carpool rides, ensuring a user-friendly experience for both riders and drivers.

Software Requirements

Front End Tools

- Front End Tools
- HTML
- CSS
- JavaScript
- Bootstrap
- PHP

Back End Tools

- MySQL
- XAMPP (for local development and testing)

Frontend Design Tools

The frontend design for the Community Carpooling System utilizes a combination of HTML, CSS, JavaScript, Bootstrap, and PHP integrated development environment to create a robust and user-friendly interface.

HTML (Hypertext Markup Language)

HTML serves as the standard markup language for creating the structure and content of web pages and applications within the Community Carpooling System. It defines elements such as headings, paragraphs, lists, links, images, forms, and

other essential components necessary for presenting ride details, user profiles, and navigation.

CSS (Cascading Style Sheets)

CSS plays a crucial role in styling HTML elements and defining the layout of web pages within the Community Carpooling System. It allows developers to control the appearance of interfaces by specifying properties such as colors, fonts, margins, padding, and positioning. CSS ensures consistency in design and enhances the visual presentation of ride information, user interfaces, and interactive elements.

JavaScript

JavaScript enables interactive and dynamic behavior on web pages, enhancing user engagement and functionality within the Community Carpooling System. It facilitates features such as form validation, real-time updates, interactive elements, and dynamic content loading. JavaScript ensures a seamless and interactive user experience during ride posting, booking, feedback, and navigation.

Bootstrap

Bootstrap is a powerful front-end framework used to create responsive and mobile-first web interfaces. It provides a set of pre-styled components, grid systems, and utilities that streamline the development of consistent and visually appealing web pages. Bootstrap ensures that the carpooling platform is accessible and user-friendly across various devices and screen sizes.

PHP (Hypertext Preprocessor)

PHP serves as the server-side scripting language used for backend processing, database interactions, and dynamic content generation within the Community Carpooling System. It enables functionalities such as user authentication, ride management, feedback processing, and database management. PHP integrates seamlessly with HTML, allowing developers to embed dynamic content, execute server-side scripts, and generate personalized interfaces based on user interactions.

Back-end design tool:

The back-end of the online examination and result system will be developed and managed using MySQL, a powerful open-source relational database management system. Here's a detailed breakdown of MySQL's features and its role within the system:

MySQL (Relational Database Management System):

❖ MySQL is utilized for the permanent storage of records within the system, ensuring data integrity, reliability, and scalability.

❖ It is the most popular open-source SQL database management system, developed, distributed, and supported by Oracle Corporation.

❖ MySQL databases are relational, storing data in separate tables organized into physical files optimized for speed.

❖ The logical model includes objects such as databases, tables, views, rows, and columns, offering a flexible programming environment for managing relationships between different data fields.

❖ SQL (Structured Query Language) is used to interact with MySQL databases,
providing a common standardized language for accessing and manipulating data.

❖ MySQL software is open-source, allowing anyone to download, use, and modify the software without any cost.

❖ MySQL Server works in client/server or embedded systems, providing a multithreaded SQL server that supports various client programs, libraries, administrative tools, and application programming interfaces (APIs).

❖ MySQL Server is known for its speed, reliability, scalability, and ease of use, making it suitable for handling large databases in demanding production environments.

❖ MySQL Server is constantly under development, offering a rich set of functions and features for accessing databases over the internet securely.

Integration with Frontend:

The integration of HTML, CSS, JavaScript, Bootstrap, and PHP within the Community Carpooling System provides a comprehensive toolkit for frontend development. HTML defines the structure and content of interfaces, CSS styles the presentation and layout, JavaScript adds interactivity and dynamic behavior, Bootstrap ensures responsiveness and consistent design, while PHP handles server-side processing and dynamic content generation. This integrated approach ensures the creation of dynamic, responsive, and user-friendly interfaces that enhance the overall user experience.

Benefits

By leveraging a combination of HTML, CSS, JavaScript, Bootstrap, and PHP, the Community Carpooling System offers a robust and versatile frontend design. HTML provides the foundational structure and content, CSS enhances the visual presentation and layout, JavaScript adds interactivity and dynamic behavior, Bootstrap ensures responsiveness and visual consistency, and PHP enables server-side processing and dynamic content generation. This integrated approach ensures the development of a feature-rich, responsive, and user-friendly carpooling system that promotes efficient ride-sharing, environmental sustainability, and community engagement.

ANALYSIS

Data Flow Diagrams (DFD)

ANALYSIS

Data Flow Diagrams (DFD)

Here is a breakdown of data flow diagrams for a community carpool system project at three levels: Context (Level 0), Level 1, and Level 2.

1. Context Diagram (Level 0)

The context diagram shows the entire system as a single process and its interactions with external entities.

Entities:

User: Individuals looking to either post or book a carpool ride.

Database: Stores data related to rides, users, and booking history.

Process:

Community Carpool System: Central system that processes ride requests and bookings.

Levels in Data Flow Diagrams (DFD)

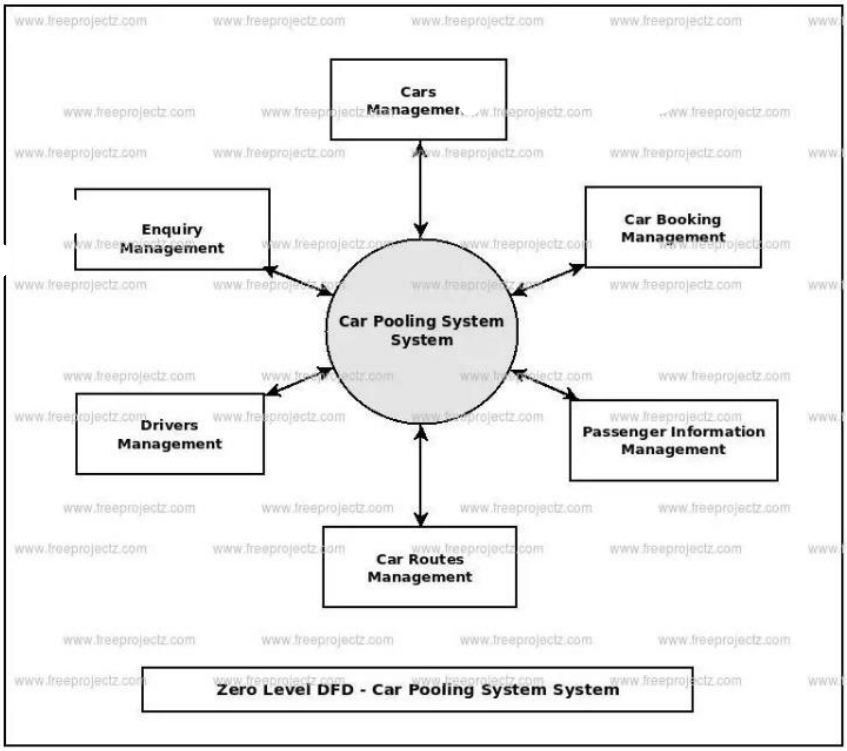
Data Flows:

User → Ride Details→ Carpool System

User ← Ride Availability Confirmation← Carpool System

User ← Booking Confirmation← Carpool System

Carpool System ↔ Ride Data, Booking Data ↔ Database



2. Level 1 DFD

This level provides a more detailed look at the processes within the Community Carpool System.

Processes:

1. Post Ride: Allows a user to post ride details, which get stored in the database.
2. Search Ride: Allows users to search for available rides based on criteria like location and timing.
3. Book Ride: Manages the booking of a selected ride and confirms booking status with the user.
4. Manage User Data: Handles user registration, login, and profile management.

Data Stores:

D1: Ride Data: Stores ride details such as date, time, and location.

D2: User Data: Stores user details (name, contact info, etc.).

D3: Booking Data: Stores booking history and current bookings.

Data Flows:

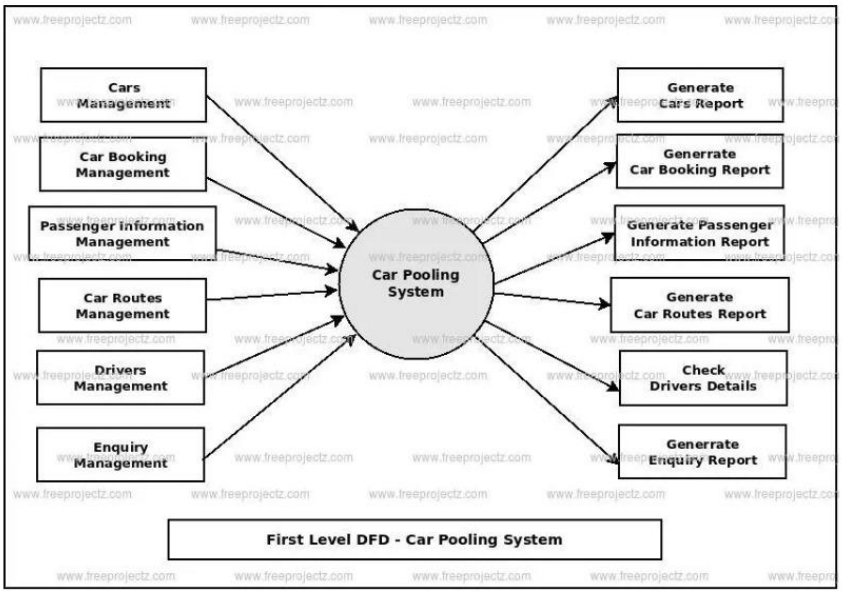
User → Ride Information → Post Ride → Ride Data

User → Search Criteria → Search Ride ↔ Ride Data → User (showing available rides)

User → Booking Request → Book Ride ↔ Ride Data, Booking Data

Book Ride → Booking Confirmation→ User

User →Profile Info→ Manage User Data → User Data



3. Level 2 DFD

Level 2 provides more granular details for each main process.

3.1 Post Ride Process

Sub-processes:

Input Ride Details: Receives and verifies user-provided ride details.

Save Ride Details: Stores verified ride details in the Ride Data store.

3.2 Search Ride Process

Sub-processes:

Input Search Criteria: Receives search criteria from the user.

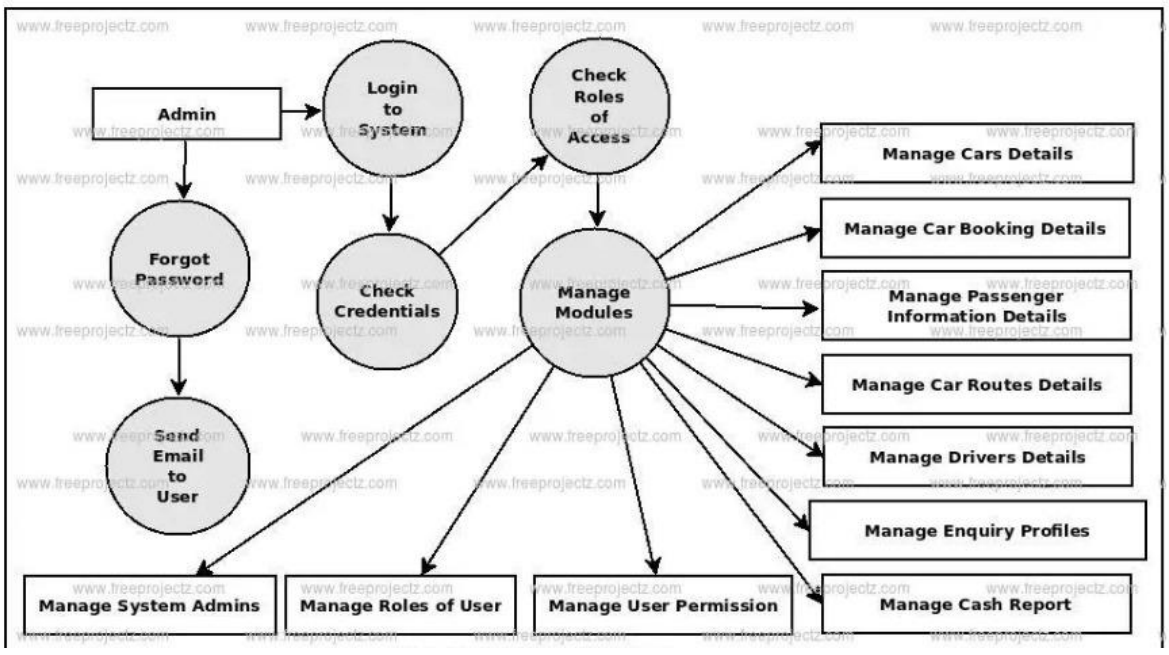
Retrieve Ride Information: Queries the Ride Data store to fetch relevant rides.
Display Results: Shows the user the list of available rides.

3.3 Book Ride Process

Sub-processes:
Check Availability: Confirms ride availability in the Ride Data store.
Save Booking: Records the booking in the Booking Data store.
Confirm Booking: Notifies the user with booking confirmation details.

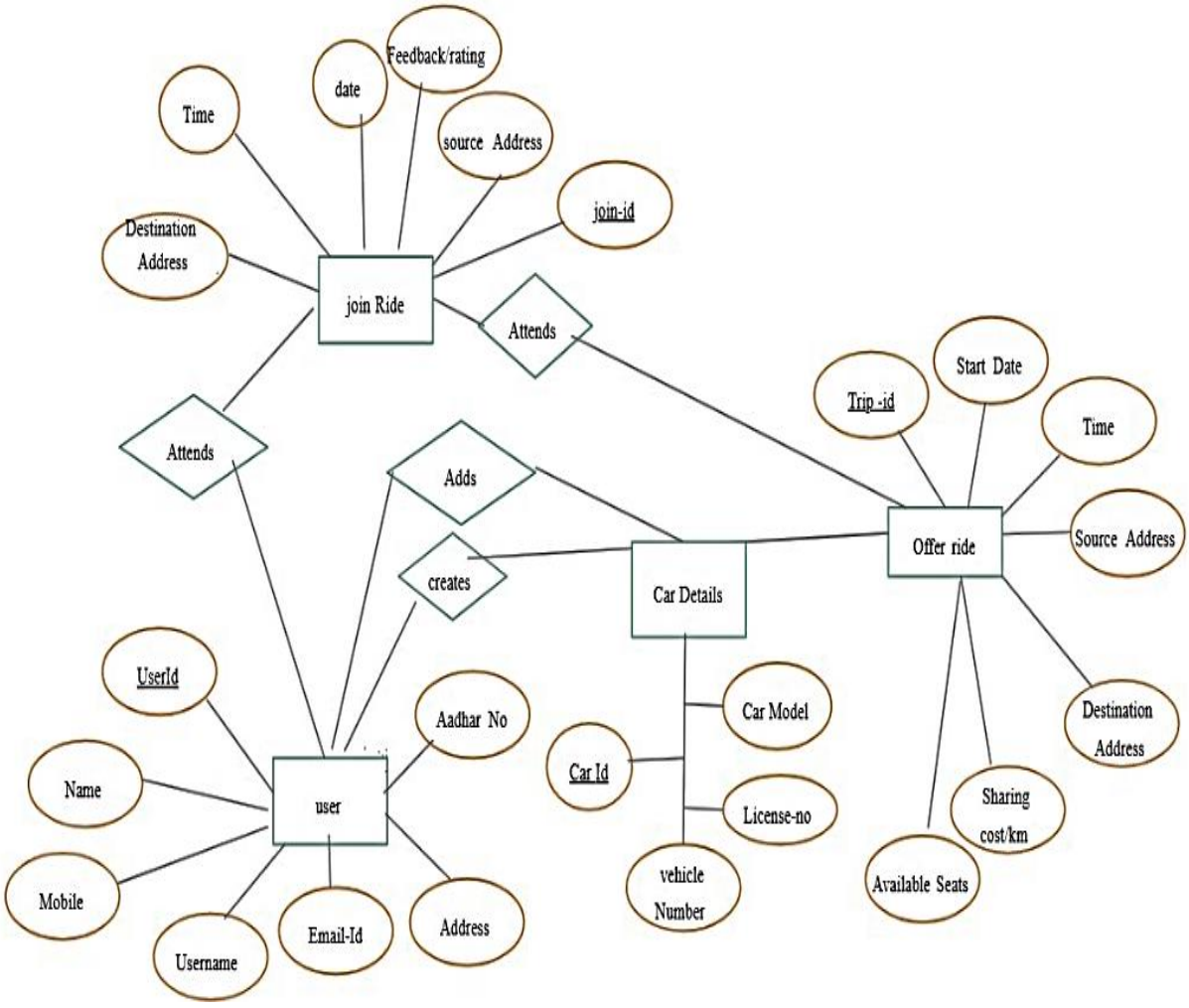
3.4 Manage User Data Process

Sub-processes:
Register User: Captures and stores new user details.
Login User: Authenticates user credentials.
Update Profile: Allows users to update their profile information in User Data.



These DFDs represent the core flow of data within your community carpool system project, highlighting how user interactions lead to data retrieval, storage, and processing within the system. If you'd like, I can create visual diagrams based on this outline.

ER Diagram



ER Diagram

Entity relationship diagram displays the relationships of entity set stored in a database. In other words, we can say that ER diagrams help you to explain the logical structure of databases. At first look, an ER diagram looks very similar to the flowchart. However, ER Diagram includes many specialized symbols, and its meanings make this model unique

An ER diagram for a community carpooling system will involve key entities such as User, Ride, Vehicle, Location, and Booking. Here’s a breakdown of the entities and their relationships:

Entities and Attributes:

User

user_id (Primary Key)
username
password

rides

ride_id (Primary Key)
origin (Foreign Key from Location)
destination (Foreign Key from Location)
date
time
available_seats

bookings

booking_id (Primary Key)
ride_id (Foreign Key from Ride)
user_id (Foreign Key from User)
seat_booked
booking_time

reviews

review_id (Primary Key)
ride_id (Foreign Key from Ride)
user_id (Foreign Key from User)
rating
review
review_time

Relationships

1. User and Rides: A user (as a driver) can offer multiple rides, but each ride has one driver.
    - Relationship: 1 (User) to M (Rides)
2. Rides and Bookings: Each ride can have multiple bookings, and each booking is associated with one ride.
    - Relationship: 1 (Rides) to M (Bookings)
3. User and Bookings: Each user can make multiple bookings, but each booking belongs to one user.
    - Relationship: 1 (User) to M (Bookings)
4. User and Reviews: Each user can write multiple reviews, but each review is associated with one user.
    - Relationship: 1 (User) to M (Reviews)
5. Rides and Reviews: Each ride can have multiple reviews, and each review is associated with one ride.
    - Relationship: 1 (Rides) to M (Reviews)

ER Diagram Representation

User (user_id, username, password)

1 -> M

Rides (ride_id, origin, destination, date, time, available_seats, driver_id)

M -> 1

Bookings (booking_id, ride_id, user_id, seat_booked, booking_time)

1 -> M

Reviews (review_id, ride_id, user_id, rating, review, review_time)

Relationships:

1. User : Rides = 1 : M
2. Rides : Bookings = 1 : M
3. User : Bookings = 1 : M
4. User : Reviews = 1 : M
5. Rides : Reviews = 1 : M

TESTING

When developing the Community Carpool System, it's crucial to ensure the application is thoroughly tested to provide a reliable, secure, and efficient user experience. Here are various types of testing that can be performed:

Functional Testing

1. Unit Testing: Test individual components (e.g., user registration, login, post ride, book ride) to ensure they work as expected.

2. Integration Testing: Verify that different modules or services (e.g., user login with ride posting, ride booking with notifications) work together correctly.

3. System Testing: Test the entire system as a whole to ensure it meets the requirements and performs its functions correctly.

4. User Acceptance Testing (UAT): Ensure that the system meets the needs and expectations of the end-users by having them test the functionality and provide feedback.

Non-Functional Testing

1. Performance Testing: Measure the system's performance under various conditions, such as high traffic or data loads, to ensure it remains responsive.

Load Testing: Test the system's behavior under expected load conditions.

Stress Testing: Determine the system's breaking point by testing it under extreme conditions.

2.Security Testing: Identify vulnerabilities and ensure that the system is protected against threats such as unauthorized access and data breaches.

3.Usability Testing: Assess the user interface and overall user experience to ensure the system is intuitive and easy to use.

4.Compatibility Testing: Ensure the system works across different devices, browsers, and operating systems.

5.Database Testing: Verify data integrity, consistency, and performance of database operations such as queries, updates, and transactions.

Specific Test Cases for Community Carpool System

User Registration and Authentication

Verify that new users can register with valid information.

Ensure users can log in with correct credentials and are blocked with incorrect credentials.

Confirm that users can change their passwords and log out successfully.

Ride Management

Test the ride posting functionality, including mandatory fields and optional details.

Verify that posted rides are visible to other users.

Ensure that users can update or delete their posted rides.

Booking Management

Check the ride booking process, ensuring users can book available rides.

Verify that booked rides are updated in both the user's and the driver's dashboards.

Confirm that users can cancel bookings if necessary.

Feedback Management

Test the feedback functionality, ensuring users can leave reviews and rate rides.

Verify that feedback is correctly stored and displayed for future reference.

Admin Management

Ensure that admins can manage users, rides, bookings, and feedback without issues.

Verify that admin actions such as user suspension or ride deletion are correctly executed and reflected in the system.

By conducting these tests, you can ensure that the Community Carpool System is robust, secure, and user-friendly, providing a positive experience for all users.

Output Screens

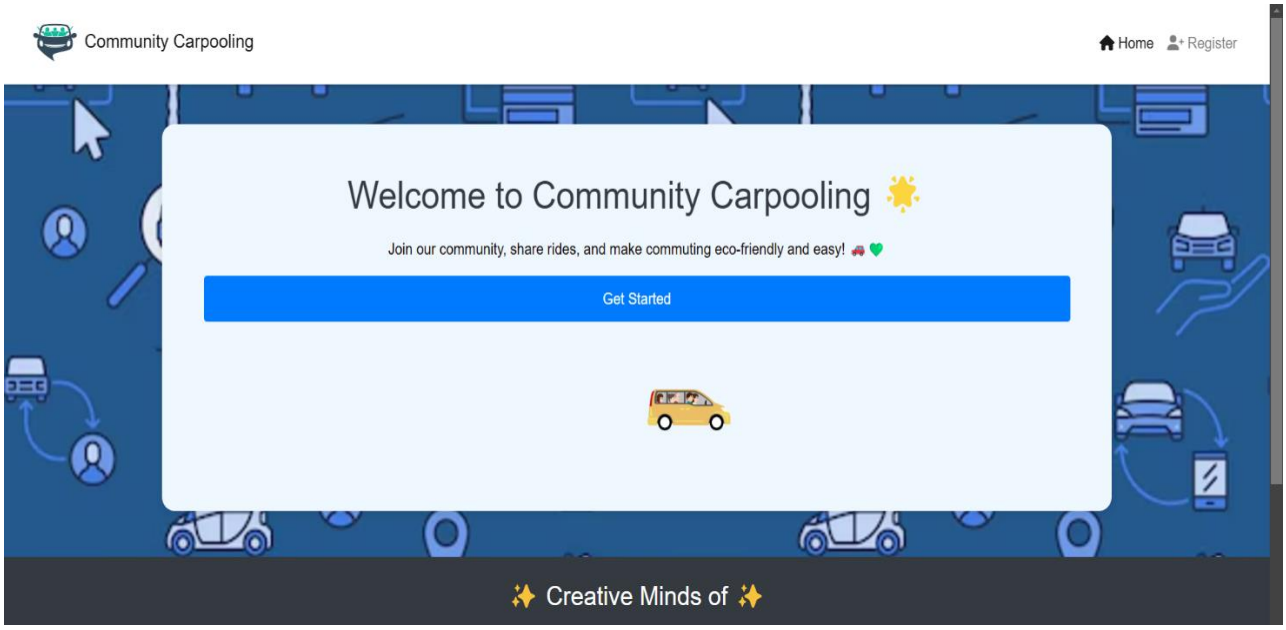


Fig : home_page

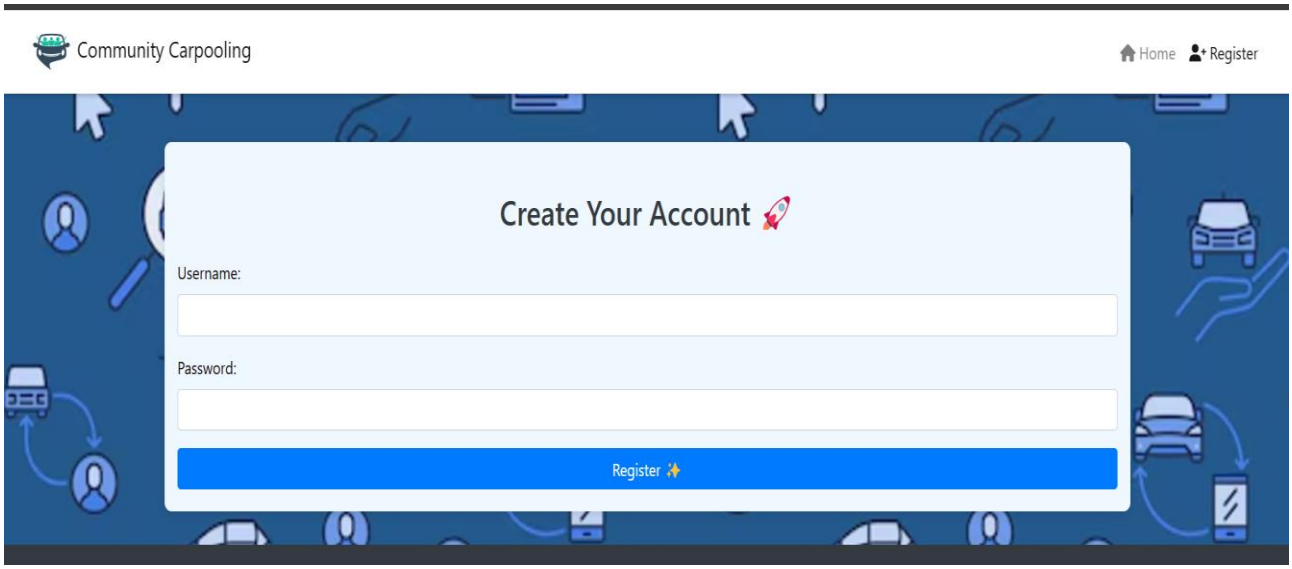


Fig : Register page

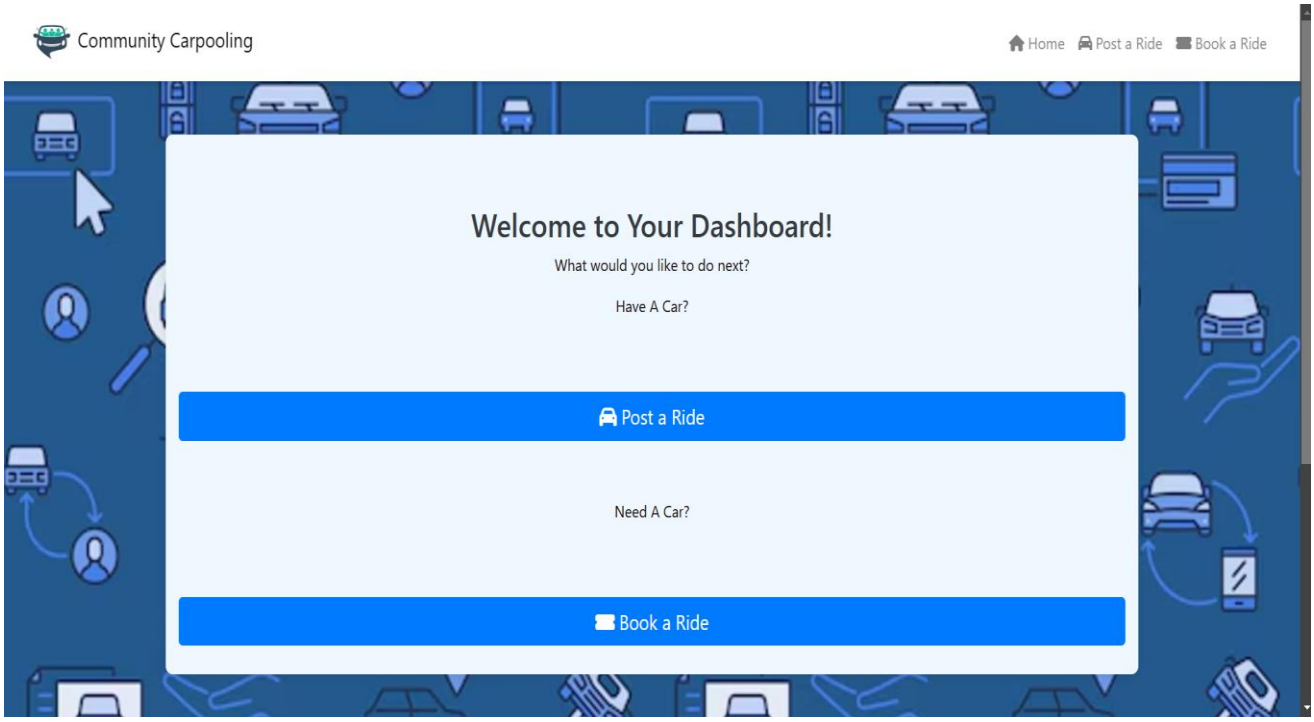


Fig : Dashboard

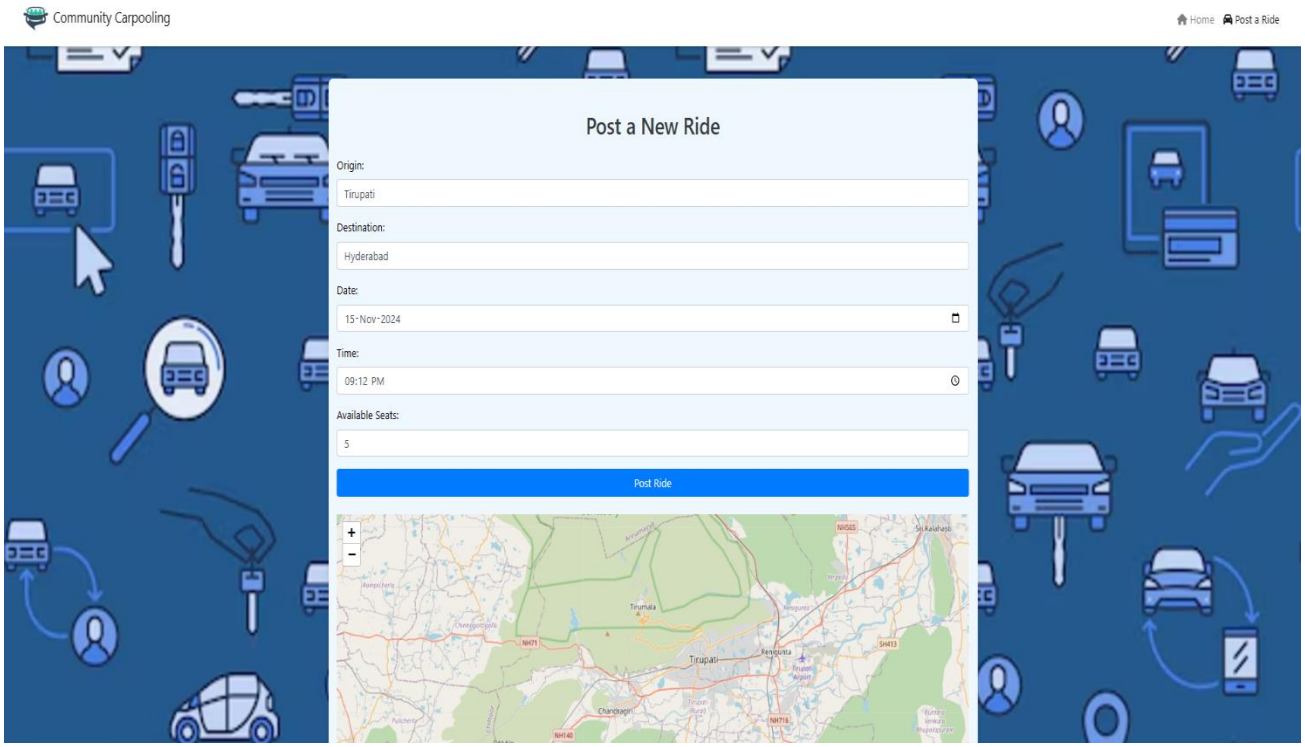


Fig :Post a New Ride page

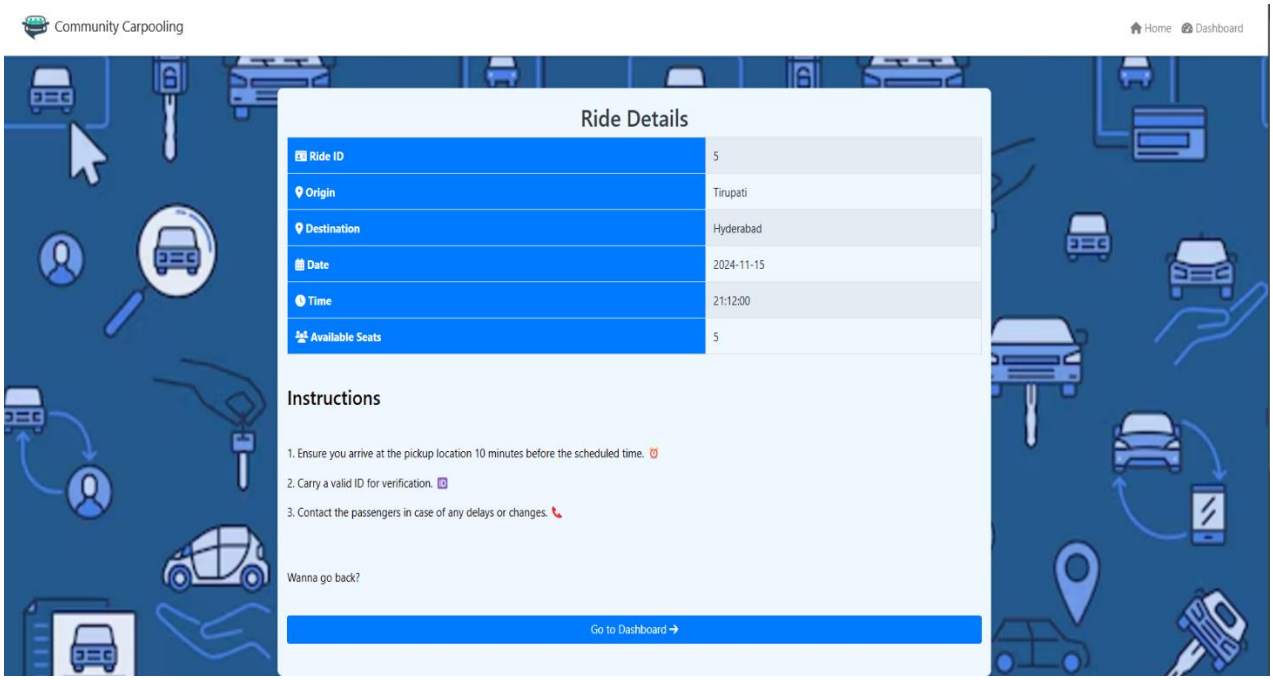


Fig : Ride Details Page

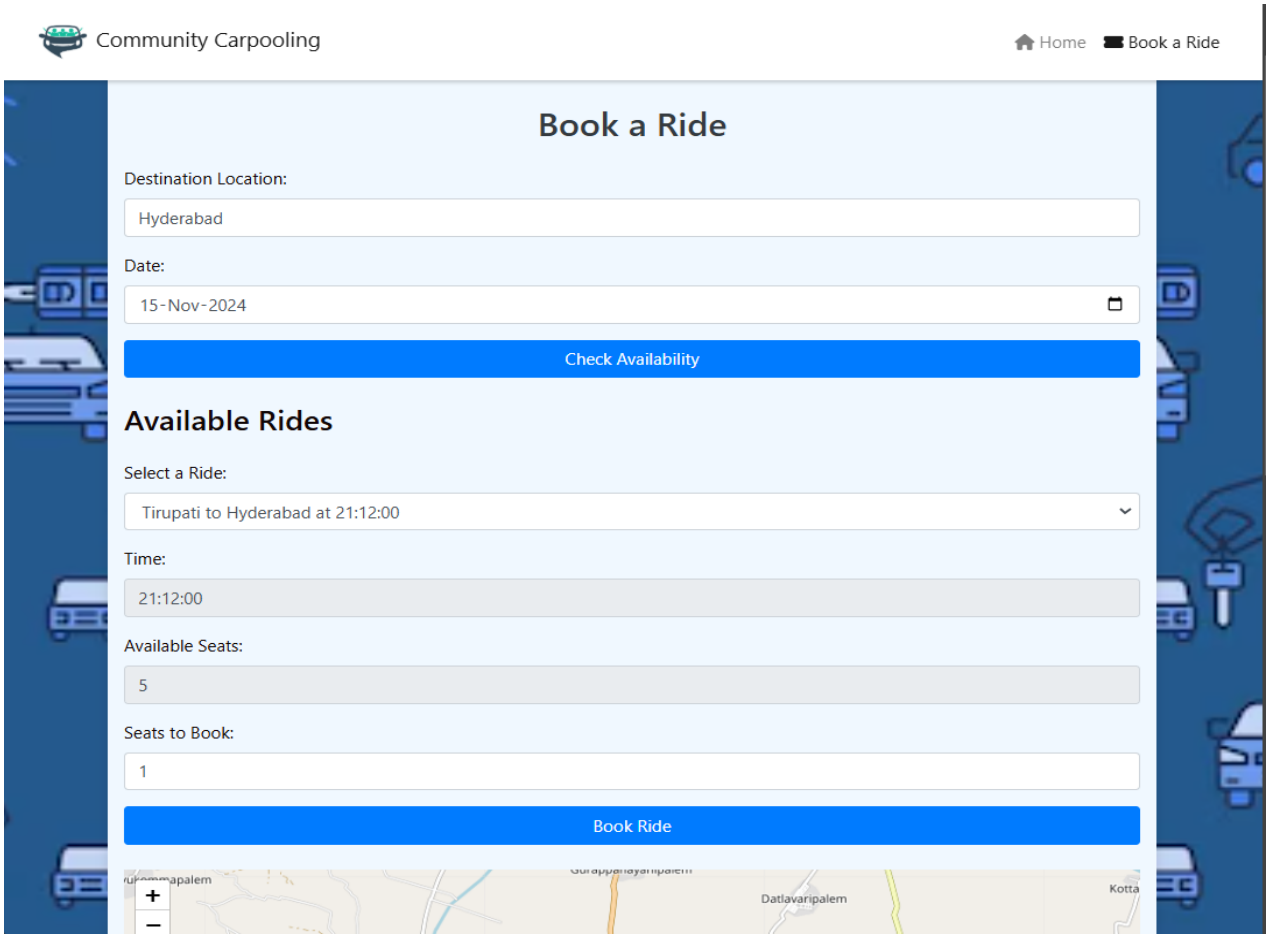


Fig : Book A Ride Page

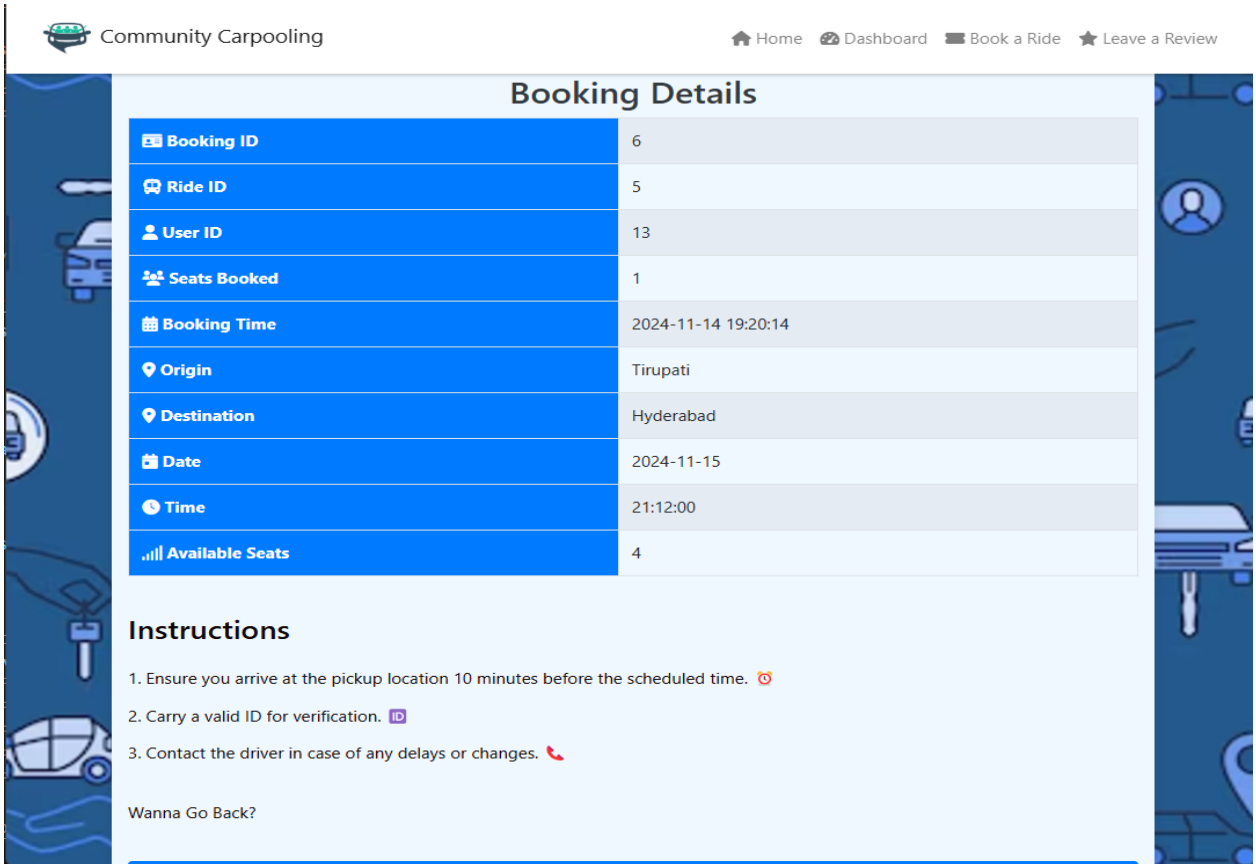


Fig : Booking Details page

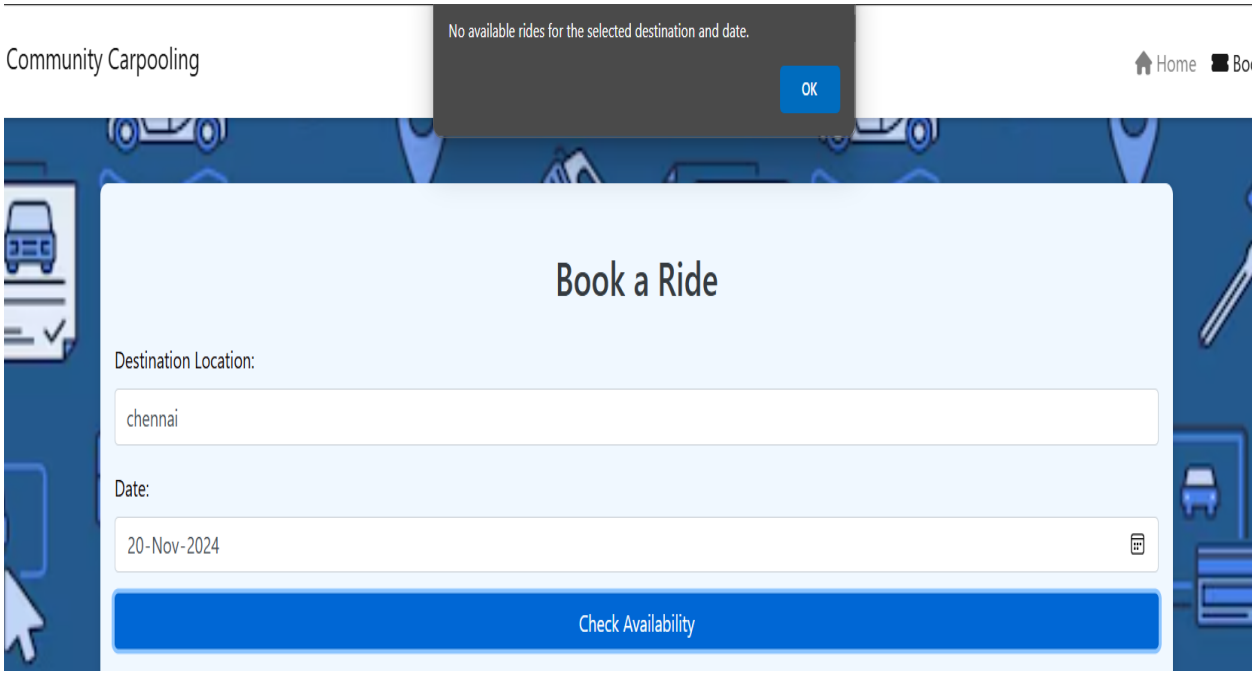


Fig : No Available rides for the destination and time

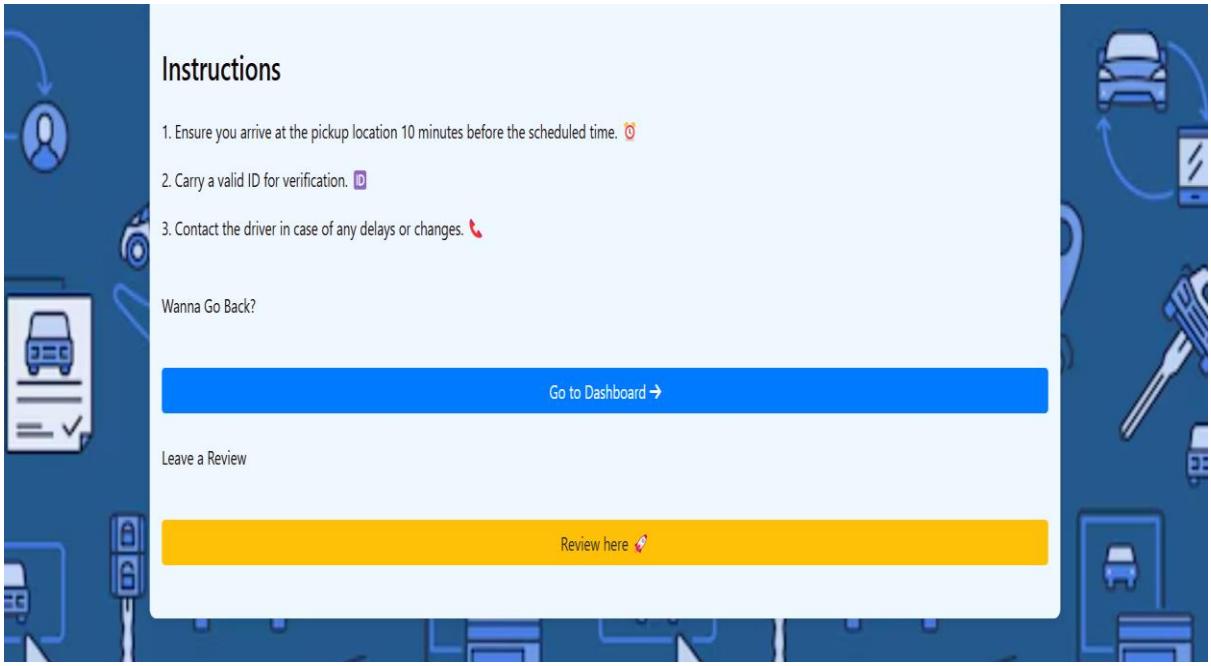


Fig: Instructions and Navigation Buttons in Booking Section

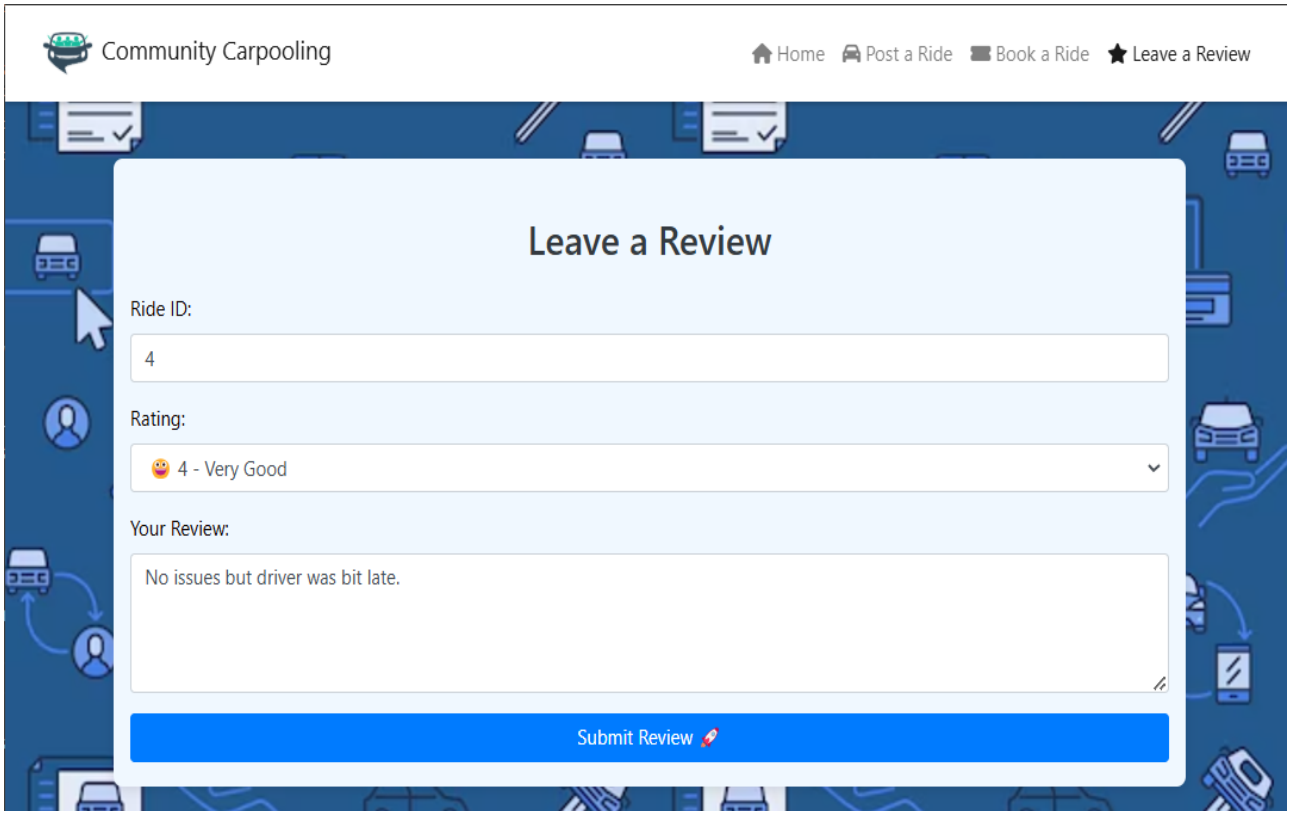


Fig : Review Page

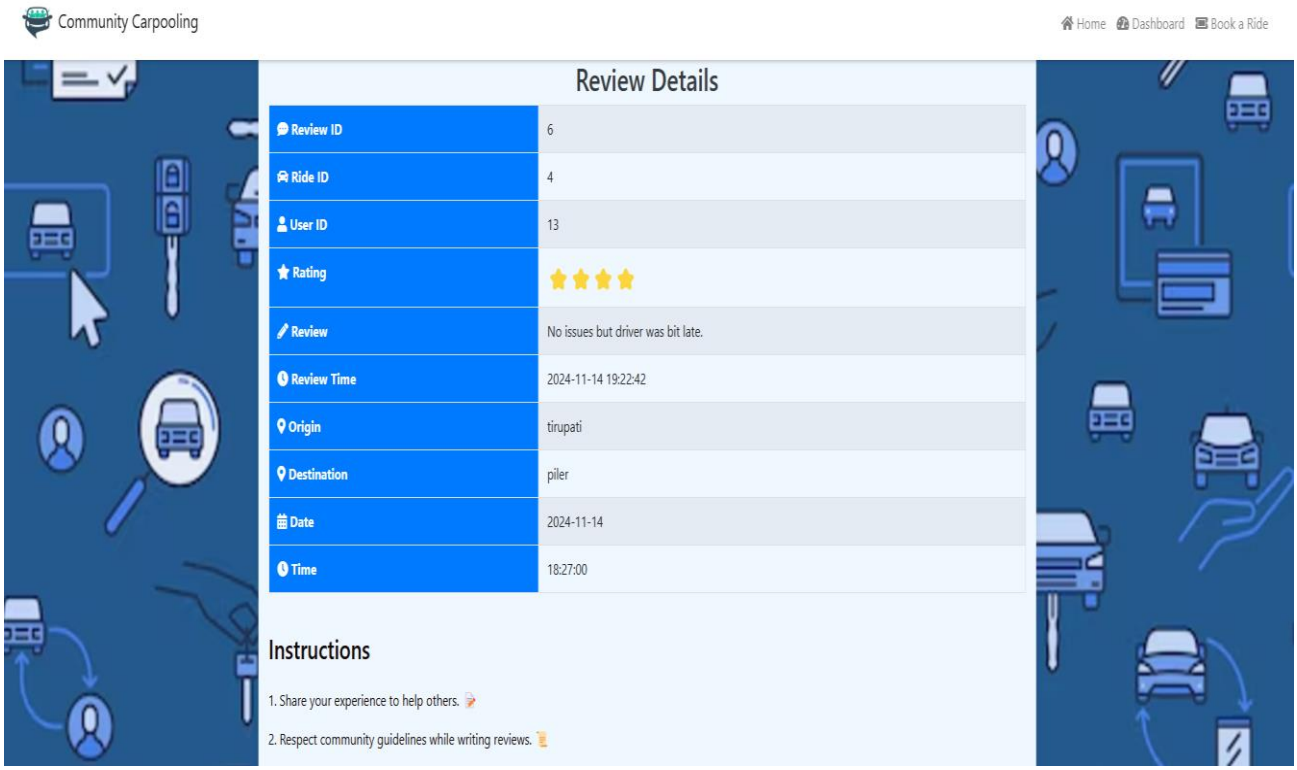


Fig : Review Details

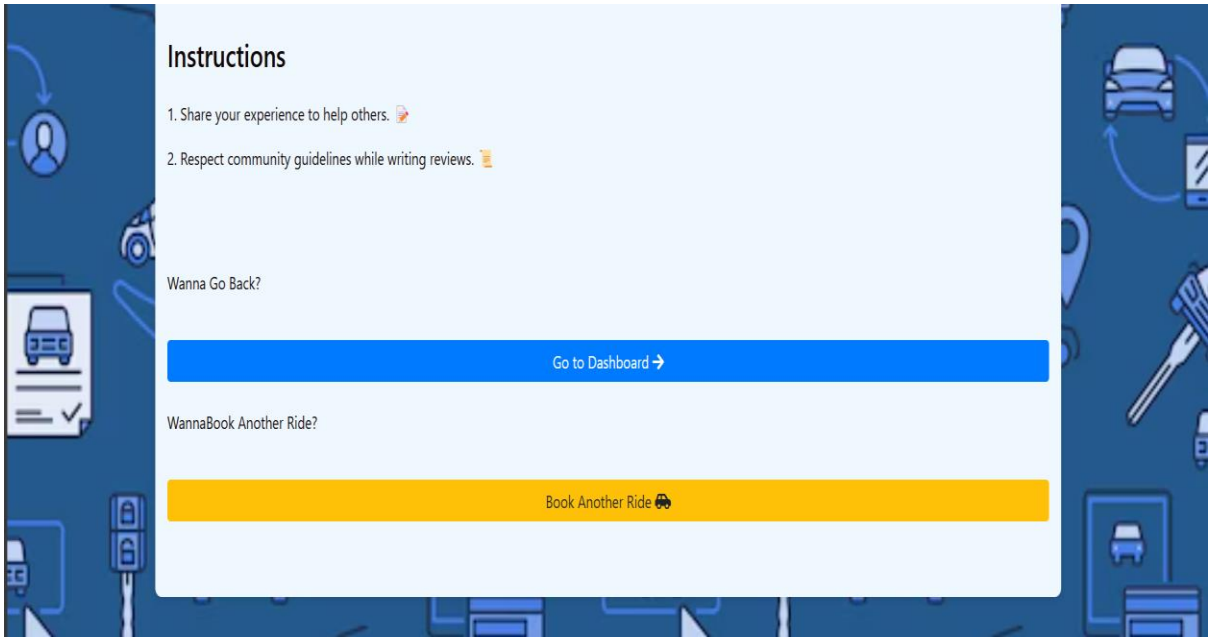


Fig: Navigation buttons after Review

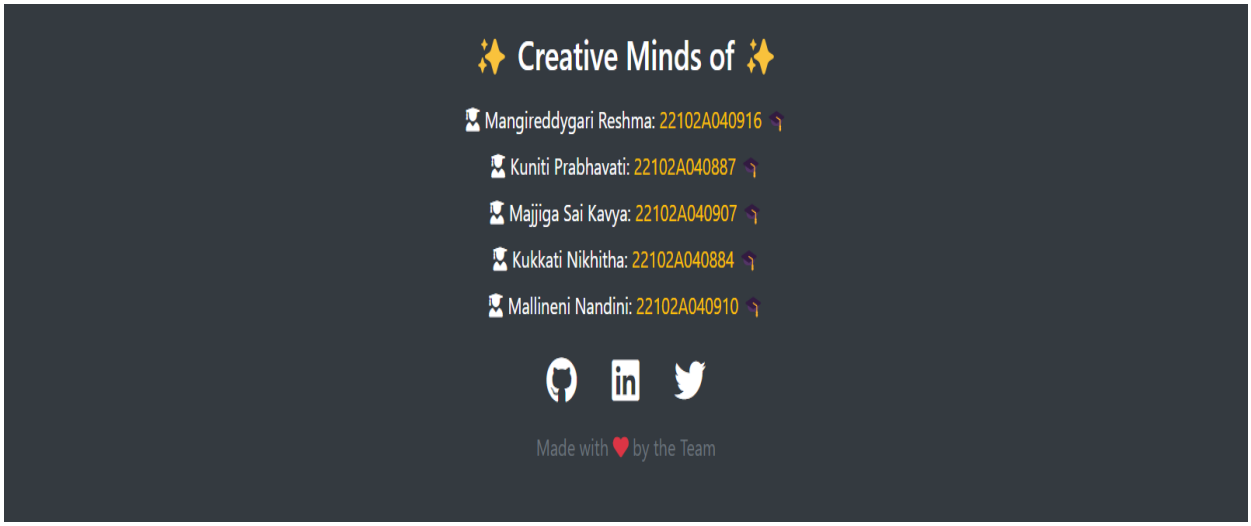


Fig : Creators Details

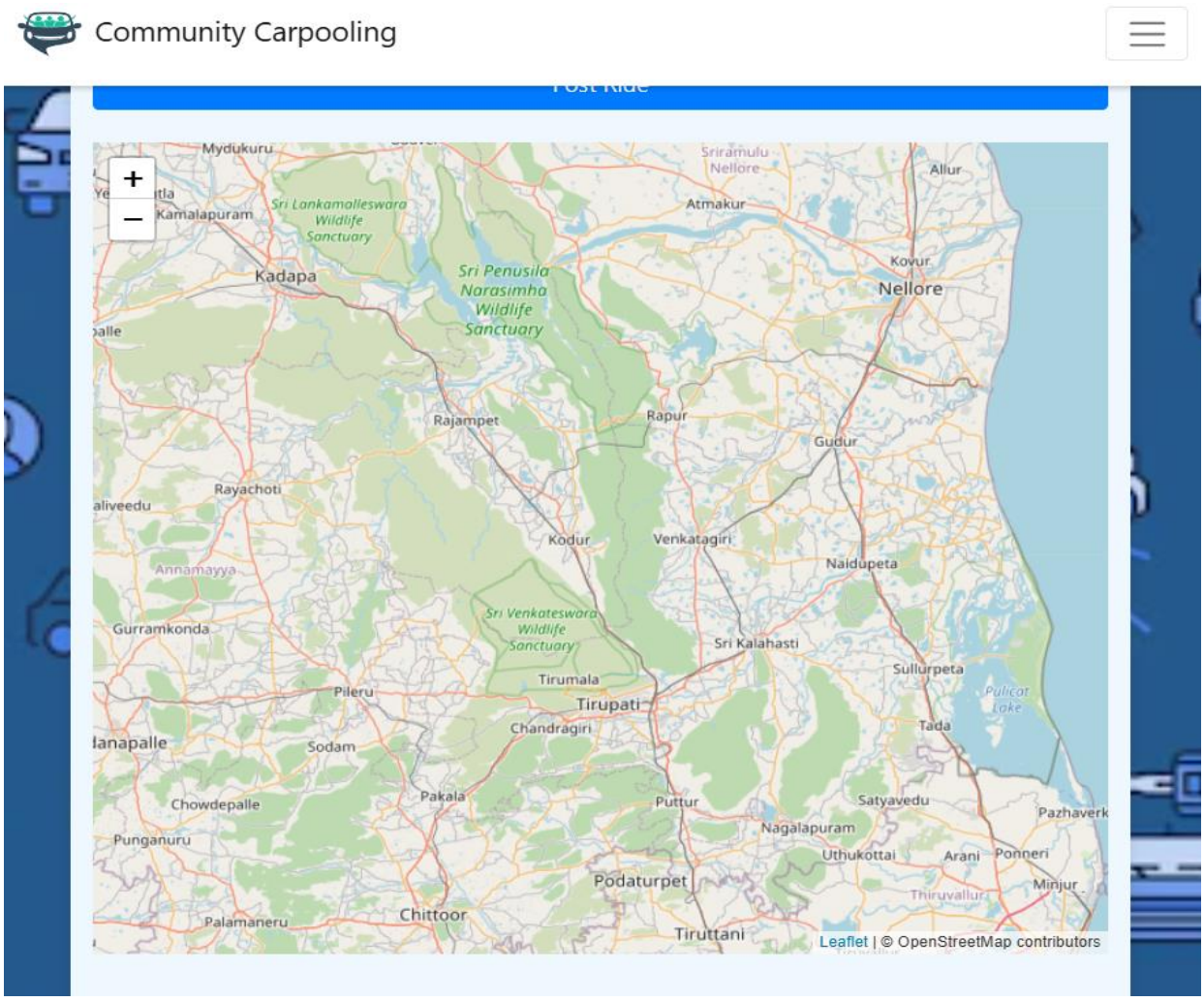
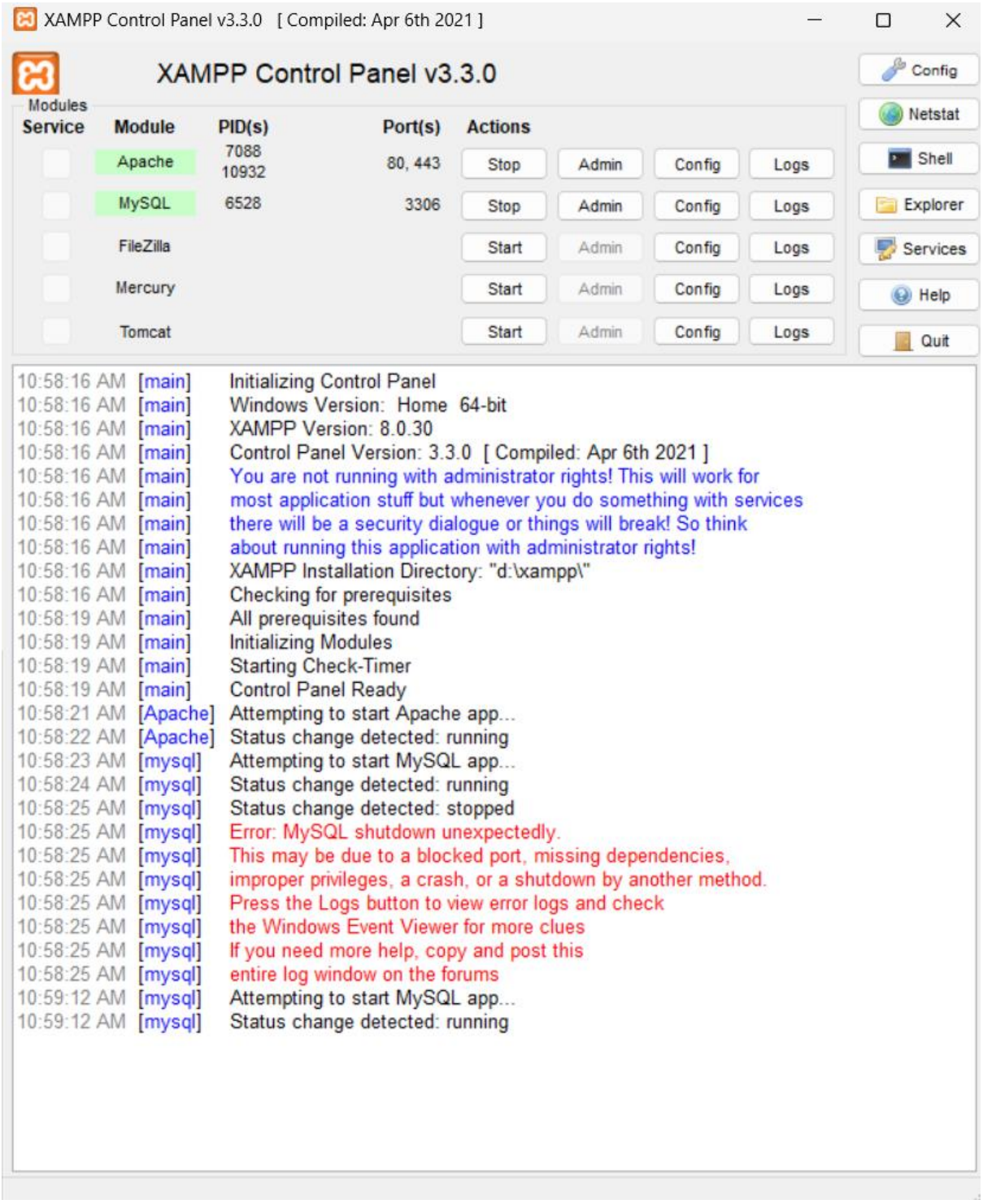


Fig: Open Street Map

SQL CONNECTION USING XAMPP:



DATA BASES

Table	Action	Rows	Type	Collation	Size	Overhead
bookings	Browse Structure Search Insert Empty Drop	6	InnoDB	utf8mb4_general_ci	48.0 KiB	-
reviews	Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_general_ci	48.0 KiB	-
rides	Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_general_ci	16.0 KiB	-
users	Browse Structure Search Insert Empty Drop	13	InnoDB	utf8mb4_general_ci	16.0 KiB	-
4 tables	Sum	29	InnoDB	utf8mb4_general_ci	128.0 KiB	0 B

Fig : Database: Carpooling

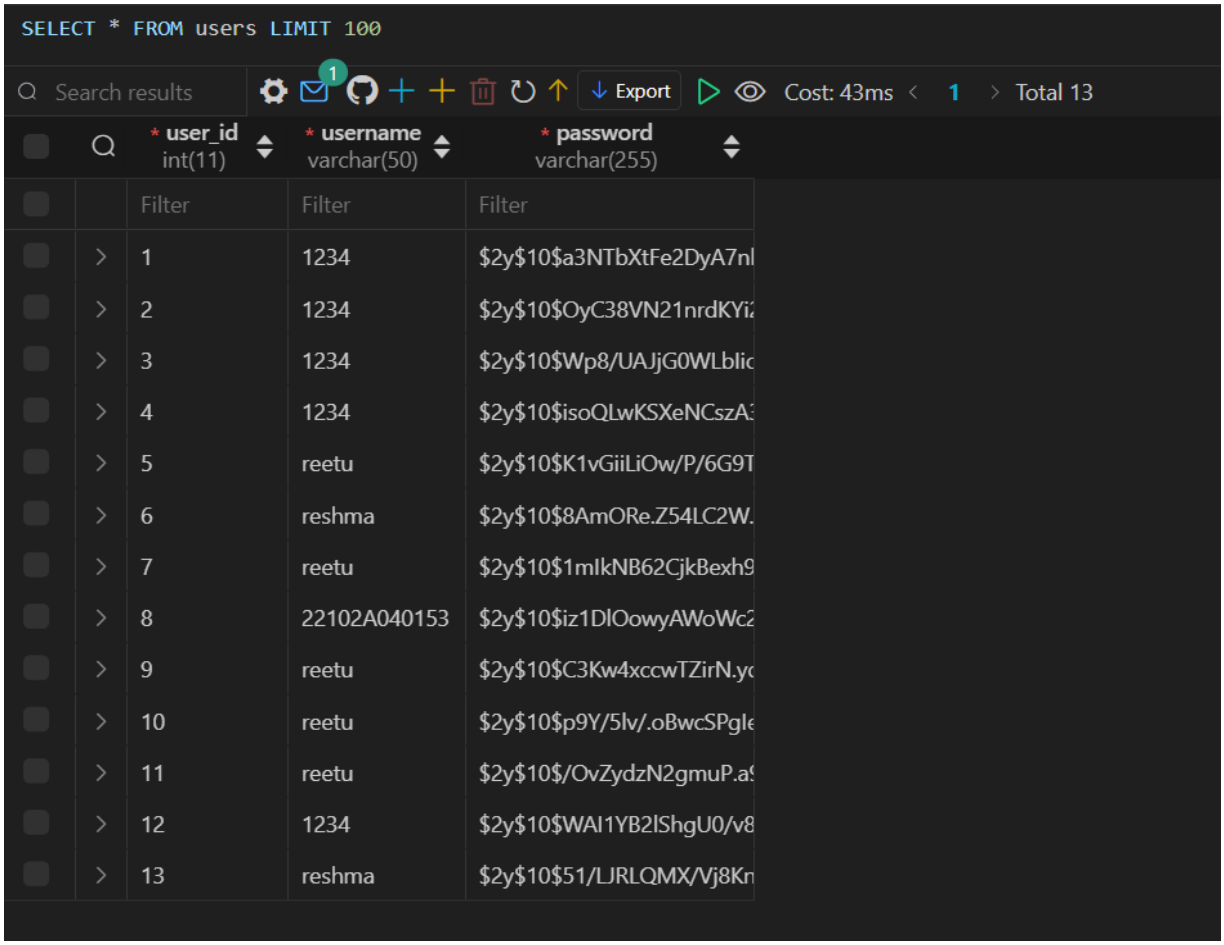
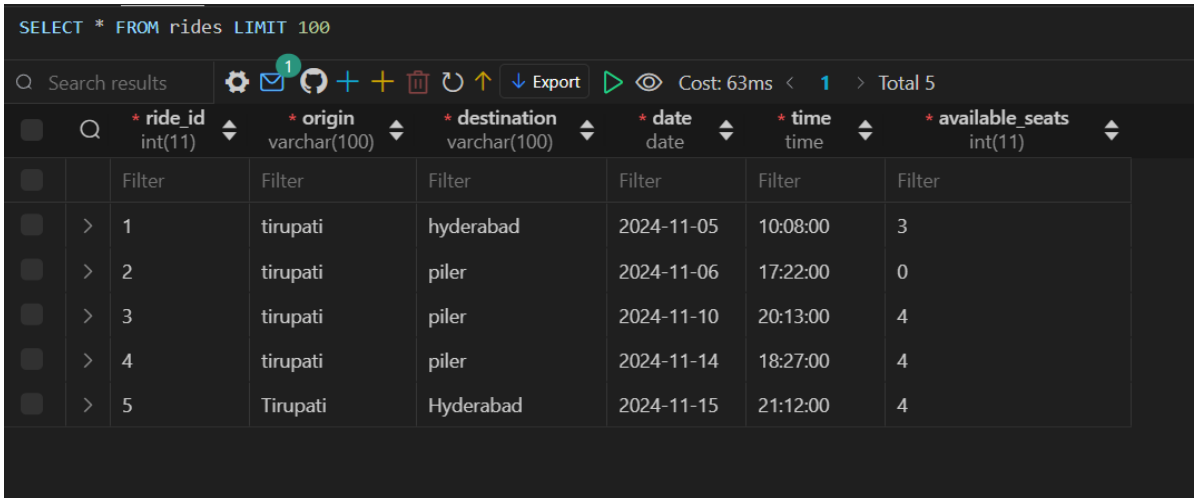


Fig : users table

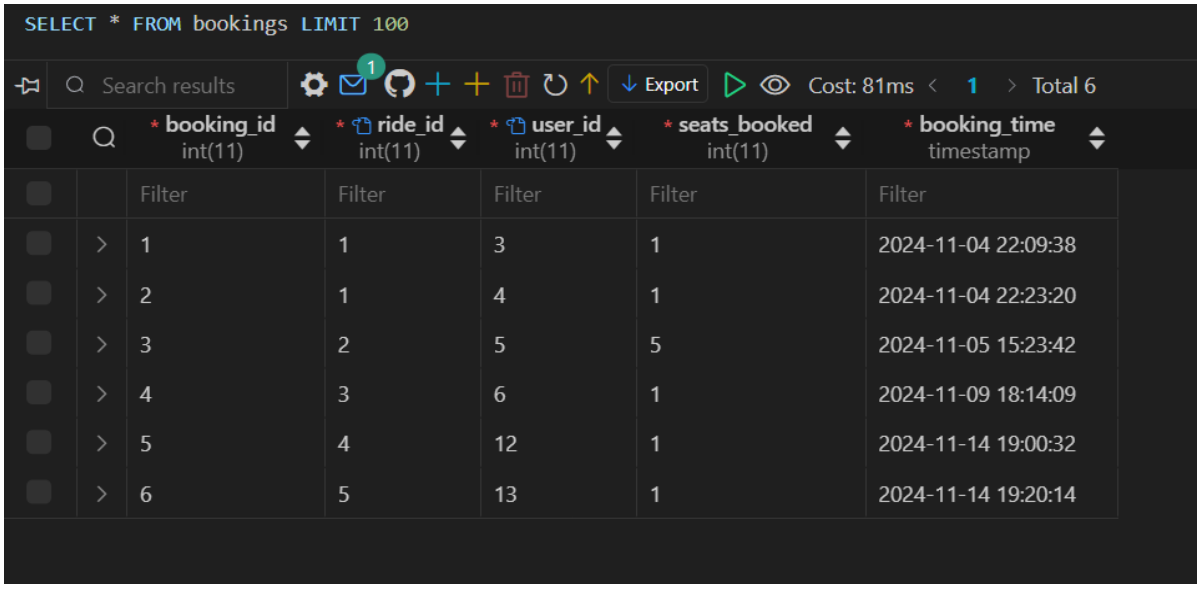


```
SELECT * FROM rides LIMIT 100
```

Search results | Export | Cost: 63ms | 1 | Total 5

* ride_id int(11)	* origin varchar(100)	* destination varchar(100)	* date date	* time time	* available_seats int(11)
Filter	Filter	Filter	Filter	Filter	Filter
1	tirupati	hyderabad	2024-11-05	10:08:00	3
2	tirupati	piler	2024-11-06	17:22:00	0
3	tirupati	piler	2024-11-10	20:13:00	4
4	tirupati	piler	2024-11-14	18:27:00	4
5	Tirupati	Hyderabad	2024-11-15	21:12:00	4

Fig : rides table

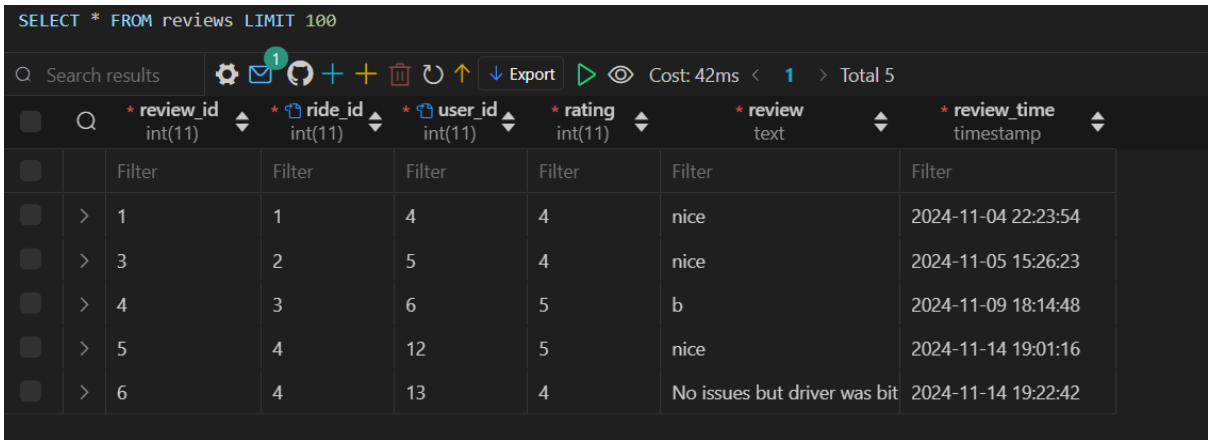


```
SELECT * FROM bookings LIMIT 100
```

Search results | Export | Cost: 81ms | 1 | Total 6

* booking_id int(11)	* ride_id int(11)	* user_id int(11)	* seats_booked int(11)	* booking_time timestamp
Filter	Filter	Filter	Filter	Filter
1	1	3	1	2024-11-04 22:09:38
2	1	4	1	2024-11-04 22:23:20
3	2	5	5	2024-11-05 15:23:42
4	3	6	1	2024-11-09 18:14:09
5	4	12	1	2024-11-14 19:00:32
6	5	13	1	2024-11-14 19:20:14

Fig : bookings table



```
SELECT * FROM reviews LIMIT 100
```

Search results | Export | Cost: 42ms | 1 | Total 5

* review_id int(11)	* ride_id int(11)	* user_id int(11)	* rating int(11)	* review text	* review_time timestamp
Filter	Filter	Filter	Filter	Filter	Filter
1	1	4	4	nice	2024-11-04 22:23:54
3	2	5	4	nice	2024-11-05 15:26:23
4	3	6	5	b	2024-11-09 18:14:48
5	4	12	5	nice	2024-11-14 19:01:16
6	4	13	4	No issues but driver was bit	2024-11-14 19:22:42

Fig : reviews table

Analysis of results:

The **Community Carpooling System** connects participants to reduce the number of vehicles on the road, leading to lower greenhouse gas emissions and decreased air pollution. For participants, it offers economic benefits by saving on fuel costs, tolls, and parking fees. Additionally, carpooling enhances community interaction and reduces commuting stress.

For administrators, the system provides data on usage and efficiency, enabling them to monitor progress, identify trends, and make informed decisions for improving urban mobility. The technological solutions, like ICT platforms, facilitate ride matching and offer real-time updates. Integration with smart city initiatives can further reduce traffic congestion, making urban transportation more efficient.

Overall, the community carpooling system supports environmental sustainability, economic savings, social interaction, and data-driven urban planning.

ADVANTAGES AND DISADVANTAGES OF COMMUNITY CARPOOL SYSTEM

Community Carpool Systems Advantages

- Economic Benefits:
    - Fuel Cost: The process saves participants money by cutting fuel costs.
    - Less Vehicle Depreciation: Because fewer individuals make the trip, it lessens the general wear on private vehicles.
- Environmental Benefits:
    - Less Carbon Emissions: Since there are fewer automobiles on the road, carbon footprints and other pollutants will also be reduced.
    - Traffic Congestion: Fewer automobiles cut down on traffic congestion
- Benefits to Individuals:

- Building Community: Carpools can help foster good friendships among people since one is able to share experiences and thoughts with others in the carpool.
- Networking Benefits: Carpooling to work or to school helps an individual network with colleagues or neighbors and, hopefully, get ahead in one's career.

Community Carpool System Disadvantages

- Coordination Problems:
  - Scheduling collisions: Scheduling harmonies can be formidable with other carpool members. You may have to adjust and change your routine.
  - Dependence on others: There's always someone else to rely on before you can go out for a trip. That can always prove quite inconvenient when the said person arrives late or cancels.
- Limited Flexibility:
  - Routes and Schedules: Since changes cannot be made at short notice and someone cannot go off on a spur of the moment where carpooling is concerned, routes and schedules are very complex to manage.
  - Geographical Boundaries: Carpooling primarily works where the necessary participants exist and routes are within close proximity to each other.
- Privacy Issues:
  - Loss of Personal Space: Carpooling rides shared provide less personal space and less privacy than when traveling alone.
  - Dangers of Safety: There is a possibility of safety concerns as well as trustworthiness of other carpool members.
- Involvement in the Initial Setup:
  - Finding Riders: It will take some time, at least in the beginning, to find

and coordinate with suitable carpool partners.

- Establishing Guidelines: That is to say, defining ground rules and expectations for the carpool requires effort as well as communication.

FUTURE SCOPE OF COMMUNITY CARPOOL SYSTEM

1. Advanced Algorithms:
   - Enhanced Ride-Matching: Implementing advanced algorithms to match riders and drivers more accurately based on their preferences, such as departure times, route similarities, and vehicle types.
   - Dynamic Routing: Using real-time data to dynamically adjust routes based on traffic conditions, road closures, and other factors to ensure timely and efficient rides.
2. Mobile Development:
   - Cross-Platform Apps: Developing native apps for both iOS and Android to ensure a seamless user experience across different devices.
   - Real-Time Notifications: Implementing push notifications to keep users informed about their ride status, including driver arrival times, route changes, and estimated arrival times.
3. Service Integration:
   - Public Transport Sync: Integrating the carpool system with public transportation schedules to provide users with comprehensive travel options.
   - Bike/Scooter Sharing: Including bike and scooter sharing options for last-mile connectivity, making the system more versatile and user-friendly.
4. Security:
   - Robust User Verification: Implementing stringent verification processes, including background checks and identity verification, to ensure the safety of all users.
   - Expanded Rating System: Enhancing the rating system to include detailed feedback on drivers and riders, promoting accountability and trust.
5. Sustainability:
   - Carbon Footprint Tracking: Developing features to track and report the carbon footprint of rides, encouraging users to make environmentally conscious choices.

- Green Partnerships: Partnering with organizations that promote sustainability, such as electric vehicle manufacturers or environmental NGOs, to support green initiatives.

6. Community Features:
   - Event Carpooling: Introducing options for users to carpool to events such as concerts, sports games, and conferences, fostering a sense of community.
   - User Groups and Forums: Creating online forums and user groups where participants can discuss carpooling experiences, share tips, and build connections.
7. Monetization:
   - Subscription Models: Offering subscription plans with benefits such as priority booking, discounted rides, and exclusive promotions.
   - Loyalty Programs: Implementing a loyalty program that rewards frequent users with points that can be redeemed for rides or other perks.
8. Analytics:
   - Usage Patterns: Analyzing user data to understand usage patterns, peak times, and popular routes, helping to optimize the system and improve user experience.
   - Community Impact Reports: Generating reports that highlight the positive impact of carpooling on the community, such as reduced traffic congestion and lower emissions.
9. Global Expansion:
   - Localization: Adapting the system for different regions by considering local languages, cultural preferences, and regulatory requirements.
   - Strategic Partnerships: Forming partnerships with local companies and governments to facilitate smooth entry into new markets and ensure compliance with local regulations.

By focusing on these areas, the system can become more user-friendly, sustainable, and impactful.

LIMITATION OF PROJECT

1. Scalability Issues:
    - Infrastructure: Managing a large number of users and rides can become complex and may require robust infrastructure to ensure smooth operation.
    - Performance: As the user base grows, the system might face performance issues like slow loading times or server crashes.
2. User Coordination:
    - Scheduling Conflicts: Users may have different schedules, making it difficult to coordinate rides.
    - Reliability: Dependence on other users can be problematic if someone cancels or is frequently late.
3. Security and Trust:
    - Safety Concerns: Ensuring the safety of users is paramount. The system must have stringent measures for user verification and background checks.
    - Trust Building: Building trust among users who are sharing rides with strangers can be challenging.
4. Limited Flexibility:
    - Fixed Routes: The system may not accommodate users with routes or timings that differ from the pre-arranged schedules.
    - Last-Minute Changes: It might be difficult to cater to last-minute ride requests or changes.
5. Geographical Limitations:
    - Coverage Area: The service may not be available in all regions, limiting its accessibility.
    - Demand Variability: Demand for carpooling services can vary significantly across different areas.
6. User Interface and Experience:
    - Ease of Use: The system must be intuitive and user-friendly, which can be difficult to achieve for users with varying levels of tech-savviness.

- Consistent Updates: Keeping the user interface updated and ensuring a smooth experience requires continuous development and maintenance.

7. Legal and Regulatory Challenges:
    - Regulations: Different regions have different regulations regarding ride-sharing, which can complicate the implementation of the system.
    - Insurance and Liability: Ensuring proper insurance coverage and managing liability issues can be complex.
8. Environmental and Economic Factors:
    - Fuel Costs: While carpooling can reduce fuel costs per person, fluctuations in fuel prices can affect overall cost savings.
    - Environmental Impact: If not managed properly, the environmental benefits could be offset by increased vehicle usage.

Conclusion

summary of findings:

The community carpooling system is a promising solution aimed at enhancing local transportation efficiency, reducing environmental impact, and fostering community interaction. By connecting people within a community who are traveling in the same direction, it offers a sustainable alternative to single-passenger commutes.

Future enhancements:

Future enhancements for the **Community Carpooling System** could focus on expanding features to increase accessibility, security, and adaptability. Enhanced ride-matching algorithms could improve the efficiency of pairing participants with compatible schedules and routes, while AI-powered route optimization could help reduce travel times and fuel consumption. Developing a mobile app version would allow participants to easily arrange rides using various devices, and integrating with public transportation systems could offer seamless multi-modal transportation options.

Multi-language support would make the system accessible to a broader, global user base, accommodating diverse linguistic needs. Implementing real-time monitoring and tracking features could enhance the safety of participants, and user verification and rating systems could build trust and reliability among users. Advanced analytics tools, such as usage trend analysis and predictive insights, could help administrators make data-driven decisions for improving the system. Finally, integrating incentive programs could encourage more people to join and stay active in the carpooling community, promoting sustainable urban mobility and fostering a strong sense of community.

REFERENCES

During the development of our community carpooling system, we have taken the reference from:

Books and Journals

These books acted as our tutors during the system development.

❖ System Analysis And Design

Kenneth E. Kendall, Julie E. Kendall

❖ An Analysis and Design of Information Systems

Grayce M. Booth

❖ Software Engineering

- Roger S. Pressman

❖ Database Management System

James A. Larson

❖ PHP: A Beginner's Guide

Riwanto Megosinarso

Online Resources

These are the following links which assisted me at each and every step in completing this project:

❖ Software Testing Tutorial

https://www.geeksforgeeks.org/

❖https://www.w3schools.com/

❖https://programmer2programmer.net

❖https://projectworld.in

❖https://1000projects.org