A

Mini Project Report

on

POOLIFY

Submitted in partial fulfillment of the requirements for the degree

Second Year Engineering – Computer Science Engineering (Data Science)

by

Nishant Chauhan	**23107010**
Sneha Edugunoori	**23107011**
Nikhil Anumalla	**23107055**
Chinmay Ghosh	**23107001**

Under the guidance of

Ms. Poonam Pangarkar

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING (DATA SCIENCE)

A.P. SHAH INSTITUTE OF TECHNOLOGY

G.B. Road, Kasarvadavali, Thane (W)-400615

UNIVERSITY OF MUMBAI

Academic year: 2024-25

CERTIFICATE

This to certify that the Mini Project report on **This System** has been submitted by **Nishant Chauhan** (23107010), **Sneha Edugunoori** (23107011), **Nikhil Anumalla** (23107055) and **Chinmay Ghosh**(23107001) who are bonafide students of A. P. Shah Institute of Technology, Thane as a partial fulfillment of the requirement for the degree in **Computer Science Engineering (Data Science)**, during the academic year **2024-2025** in the satisfactory manner as per the curriculum laid down by University of Mumbai.

Ms. Poonam Pangarkar
Guide

Ms. Anagha Aher
HOD, CSE (Data Science)

Dr. Uttam D. Kolekar
Principal

External Examiner:
1.

Internal Examiner:
1.

Place: A. P. Shah Institute of Technology, Thane
Date:

ACKNOWLEDGEMENT

This project would not have come to fruition without the invaluable help of our guide **Ms. Poonam Pangarkar**. Expressing gratitude towards our HoD, **Ms. Anagha Aher**, and the Department of Computer Science Engineering (Data Science) for providing us with the opportunity as well as the support required to pursue this project. We would also like to thank our project coordinator **Prof. Aavani N** who gave us her valuable suggestions and ideas when we were in need of them. We would also like to thank our peers for their helpful suggestions.

TABLE OF CONTENTS

Chapter 1

Introduction

POOLIFY is a comprehensive and structured ride-sharing platform designed to make daily commuting more convenient, affordable, and efficient. In today's fast-paced world, urban transportation faces numerous challenges, including increasing traffic congestion, rising fuel costs, and environmental concerns due to excessive carbon emissions. Despite the clear advantages of carpooling, such as reduced travel expenses and lower fuel consumption, the absence of a well-structured and reliable system often discourages individuals from adopting this approach. Many people travel on similar routes but end up using separate vehicles, leading to inefficiencies and unnecessary resource consumption. To address this issue, this system provides a simple yet effective solution that enables users to connect with others traveling in the same direction, either by offering a ride or finding one, thereby promoting the concept of shared mobility.

It is designed with a user-friendly interface using Python Tkinter, ensuring a seamless experience for both ride providers and passengers. The platform facilitates easy ride booking, allowing users to schedule, manage, and monitor their rides effortlessly. One of the key highlights of the system is its intuitive chat-bot, which provides instant assistance, helping users navigate the platform smoothly. Additionally, it incorporates a structured ride-matching mechanism that enhances the efficiency of the carpooling process by connecting passengers with suitable drivers based on their route preferences.

To maintain a smooth operational flow, this includes an admin panel responsible for overseeing ride provider approvals and monitoring the platform's activities. This ensures that the system remains well-regulated and functions without disruptions. The integration of a digital wallet further simplifies ride fare management, allowing users to handle payments conveniently within the platform. By combining these essential features, it offers a streamlined and structured approach to carpooling, making it a viable alternative for daily commuters looking for cost-effective and hassle-free transportation options.

This project highlights the significance of shared mobility solutions in reducing urban transport challenges while promoting an eco-friendly approach to commuting. By leveraging technology, this system aims to encourage a shift towards smarter transportation practices, reducing overall fuel consumption and minimizing the number of vehicles on the road. The platform is built with the vision of making carpooling more accessible, practical, and widely adopted, ensuring that users can benefit from a simplified and efficient ride-sharing experience. Through continuous improvements and feature enhancements, it aspires to provide an optimized solution for daily commuters, making travel not just affordable but also more sustainable in the long run.

1.1. Purpose:

Poolify addresses urban commuting inefficiencies by connecting travelers with similar routes for shared rides. With rising fuel costs, traffic congestion, and environmental concerns, the platform offers a structured, tech-driven solution to encourage carpooling. By sharing fuel expenses, users reduce commuting costs while collectively cutting carbon emissions and easing road congestion.

Beyond affordability and sustainability, it prioritizes security with user authentication, ride verification, and a rating system, ensuring a safe and reliable carpooling experience. It fosters a community-driven approach, allowing users to form regular commuting groups, transforming daily travel into a more economical, social, and eco-friendly experience.

Intended Users:

- Students & Daily Commuters – Individuals who frequently travel to schools, colleges, or workplaces and are looking for a budget-friendly, hassle-free ride-sharing option.
- Working Professionals – Employees seeking a reliable and convenient way to share rides with colleagues or other verified users to reduce fuel costs and improve travel efficiency.
- Rideshare Service Providers – Businesses or startups interested in leveraging carpooling technology to expand or integrate similar transportation services.
- Urban Planners & Transportation Authorities – Decision-makers focusing on reducing traffic congestion and enhancing public transport by incorporating carpooling solutions into city infrastructure.
- Environmental Organizations – Groups promoting sustainable travel solutions that help reduce carbon emissions and fuel consumption.

1.2. Problem Statement:

Urban transportation is becoming increasingly inefficient due to the rising number of private vehicles on the road, leading to severe traffic congestion, increased fuel consumption, and growing environmental concerns. Many individuals travel daily along similar routes, yet most vehicles operate with empty seats, resulting in unnecessary fuel wastage and excessive travel expenses. While public transportation is available, it often lacks reliability, comfort, and flexibility, making it an impractical option for many daily commuters. Ride-hailing services, on the other hand, are expensive for regular use, further increasing the financial burden on individuals.

Despite the clear benefits of carpooling, its adoption remains limited due to the absence of a structured and accessible platform that simplifies the process. Many commuters struggle to find a reliable system that allows them to share rides efficiently, making solo travel the more common yet less sustainable choice. Without an organized approach, people continue to rely on inefficient travel methods, leading to more vehicles on the road, higher transportation costs, and increased pollution.

To address these concerns, this system has been developed as a platform to demonstrate the potential of a structured ride-sharing system. It provides an interface for managing carpooling activities in a simplified manner. It incorporates essential features such as user registration, an admin panel for ride provider verification, and a wallet system for managing ride fares. Additionally, a chat-bot is integrated to assist users by providing basic guidance and responses to common queries.

By offering a well-organized system for managing carpooling activities, it showcases how software-driven solutions can contribute to reducing travel costs and improving transportation efficiency. The project aims to create a foundation for a structured ride-sharing platform that, with future enhancements, can introduce more personalized features and improve overall usability. With its focus on accessibility and affordability, this serves as an innovative step toward addressing modern urban commuting challenges while promoting smarter and more efficient travel habits.

1.3. Objectives:

This Application is developed as a structured platform aimed at simplifying ride-sharing and making daily commuting more cost-effective and organized. The system is designed to optimize transportation by promoting shared travel, thereby reducing fuel expenses, traffic congestion, and environmental impact. This service incorporates features that allow users to manage carpooling activities efficiently while maintaining an intuitive and user-friendly experience. The key objectives of the system are as follows:

- User Registration & Role-Based Features

  This platform provides a straightforward registration process where users can sign up using basic credentials. The system categorizes users into different roles, distinguishing between those who offer rides and those who seek them. Each role has specific functionalities within the platform.

- Ride Management & Booking System

  The system allows ride providers to manage carpooling activities by adding details such as travel routes, availability, and fare management. Users seeking rides can explore available listings and track ride details. The booking system ensures a streamlined process where users can manage their participation in carpooling activities.

- Profile & Account Management

  Each user has access to a dedicated profile section where they can update personal details and track their ride history. Ride providers can review their previous and upcoming listings, while other users can monitor their participation history. Additionally, a built-in wallet system facilitates ride fare management.

- Interactive Assistance & Simplified Navigation

  This application includes an interactive chat-bot to assist users by providing responses to common queries and offering guidance on platform functionalities. Additionally, a static map interface provides a clear visualization of ride details, helping users understand route information better.

- Data Management & Platform Usability

  The platform is designed to ensure efficient data handling for ride-related activities. With an intuitive interface, users can easily navigate and manage their ride-sharing activities without complications. The system also focuses on maintaining smooth coordination between users while keeping ride records structured and accessible.

1.4. Scope:

This solution is designed to enhance commuting by providing a structured carpooling platform that connects drivers and passengers traveling on the same route. The system focuses on optimizing ride-sharing, reducing commuting costs, and improving travel experiences while addressing challenges such as traffic congestion, fuel expenses.

- Ride Matching & Booking System

  This service enables drivers to create ride listings by specifying pickup and drop-off locations, date, time, and available seats. Passengers can search for rides, book available seats, and receive booking confirmations, ensuring a seamless ride-sharing experience.

- User Authentication & Profile Management

  The platform offers a role-based access system where users can register as drivers or passengers. Users can manage their profiles by updating personal details, changing passwords, and tracking their ride history, making the system personalized and user-friendly.

- Navigation & Map Integration

  Ride routes are displayed on a static map for better visualization, helping users plan their trips efficiently. Additionally, THIS SYSTEM integrates Google Maps redirection for navigation assistance, ensuring a smooth and hassle-free commute.

- Cost Sharing & Payment Management

  The system ensures transparency in ride fares, which are set by drivers, allowing passengers to make informed decisions. Users can also track expenses and manage transactions within the platform, promoting fair cost distribution and financial convenience.

- Corporate & Workplace Carpooling

  This application supports corporate carpooling, making it easier for employees to share rides, reduce transportation costs, and minimize parking issues. Organizations can adopt carpooling policies to encourage sustainable and efficient commuting within workplaces.

- Accessibility for Senior Citizens & Differently-Abled Individuals

  The platform is designed to be inclusive, providing structured and accessible ride-sharing solutions for elderly individuals and people with disabilities. By enabling users to find reliable and convenient rides, THIS SYSTEM promotes an accessible and user-friendly transportation system.

- Urban & Suburban Traffic Reduction

  By optimizing shared rides in high-demand areas, THIS SYSTEM helps reduce congestion and improve commuting efficiency. The platform contributes to lowering fuel consumption and minimizing peak-hour traffic, making urban and suburban transportation more sustainable.

- Delivery & Logistics Optimization

  It can be leveraged for small-scale delivery and logistics services, optimizing routes and sharing transport costs. Businesses can utilize the platform for cost-effective last-mile delivery solutions, enhancing efficiency in local logistics.

- Chat-bot Integration

  To enhance user experience, it includes a chat-bot that assists users with ride inquiries, support requests, and general platform navigation. This feature ensures quick responses and a more interactive user experience.

Chapter 2

Proposed system

The proposed system is designed to create a structured and user-friendly carpooling platform that enables drivers and passengers to share rides efficiently. The system provides a convenient, cost-effective, and optimized travel solution, reducing individual commuting expenses and addressing issues such as traffic congestion and fuel consumption.

Unlike traditional carpooling methods, which rely on informal arrangements or word-of-mouth communication, this application introduces a centralized and automated platform where users can create, search, and book rides seamlessly. The system is structured to enhance usability, transparency, and convenience by integrating essential features such as trip creation, ride booking, real-time seat availability updates, and map visualization.

User-friendly interface: Ensures a smooth experience for both drivers and passengers with an intuitive platform.

Role-based access: Differentiates user functionalities, allowing drivers to create trips and passengers to search and book rides.

Interactive maps and navigation: Provides visual representation of trip routes and integrates with Google Maps for real-time directions.

Cost-effective travel: Reduces travel expenses by enabling shared rides and splitting fuel costs among passengers.

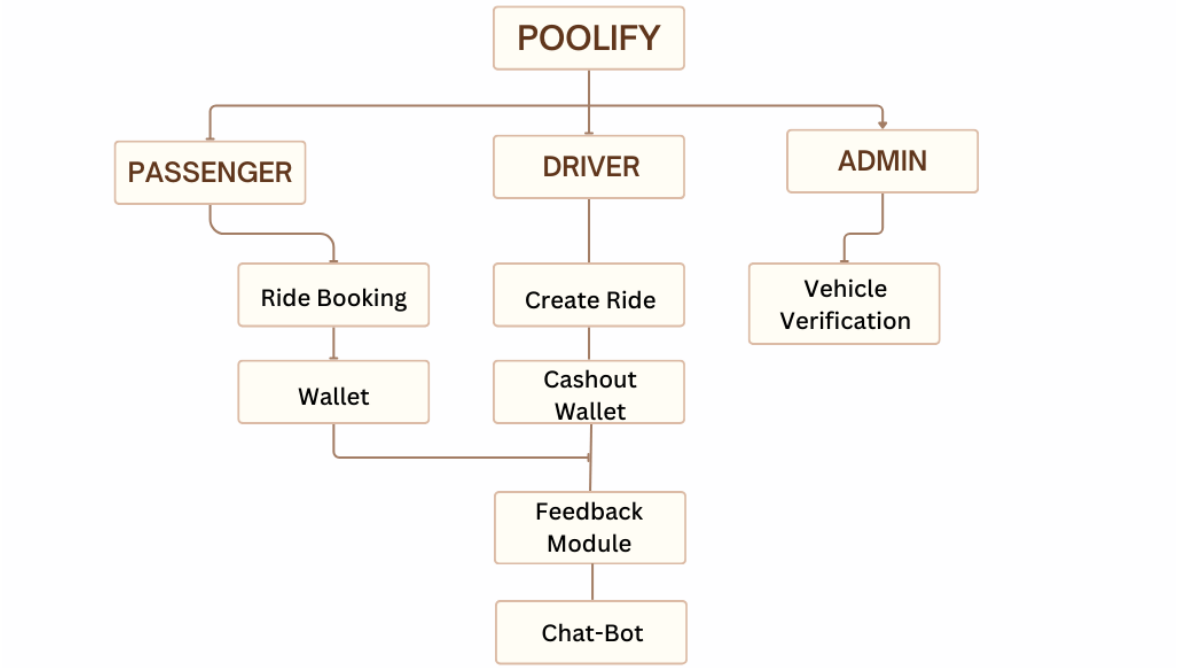

Fig 2.1 Block Diagram

As illustrated in figure 2.1, the functional architecture of the POOLIFY application is organized around three primary user roles: passenger, driver, and admin, each with distinct responsibilities and features. Passengers can book rides and manage payments through a built-in wallet system, while drivers can create rides, cash out wallet earnings, and interact with the feedback module and chat-bot for assistance. The admin is responsible for vehicle verification, ensuring that only verified vehicles operate on the platform. Shared modules like the wallet, feedback module, and chat-bot enhance user experience and streamline operations across roles. This modular structure promotes a clear separation of concerns, improves scalability, and ensures smooth functionality within the ride-sharing ecosystem.

2.1. Features and Functionality:

- Trip Creation (Driver)

  Drivers can create trips by specifying origin, destination, date, time, and available seats. Allows drivers to modify or cancel trips before a booking is confirmed. Provides a clear overview of active and past trips in the driver's profile.

- Ride Booking (Passenger)

  Passengers can search for available rides based on origin, destination, and date. Displays ride details, including driver information, available seats, and trip timing. Updates seat availability in real-time upon booking confirmation. Sends booking confirmation to the passenger with trip details.

- Interactive Maps

  Integrates Google Maps to visualize pickup and drop-off locations, trip routes, and estimated travel time. Provides an interactive interface for users to navigate their selected ride details. Redirects to Google Maps for real-time navigation assistance.

- Profile and Account Management

  Provides a dedicated profile page where users can manage personal details, update their password, and track their ride history. Allows users to check their wallet balance for fare settlements, if applicable. Ensures easy access to past and upcoming ride details for both drivers and passengers.

- User Authentication and Role-Based Access

  Implements a seamless login and signup system, including Google authentication for quick access. Differentiates access based on user roles, enabling drivers to create rides and passengers to search and book rides. Ensures role-specific features are accessible only to authorized users.

Chapter 3

Project outcomes

It aims to improve the efficiency and convenience of ride-sharing by providing a structured platform for drivers and passengers. By implementing this system, users can experience reduced travel costs, optimized routes, and a seamless booking process. The platform also enhances ride management by allowing users to track ride history and manage their trips efficiently. Ultimately, these outcomes will lead to increased ride accessibility, improved commuting experience, and reduced traffic congestion, making urban mobility more efficient and sustainable.

- Efficiency:

  The platform streamlines ride creation, booking, and payment processes, minimizing manual coordination efforts. With separate accounts for drivers and passengers, users can efficiently manage their rides and travel preferences.

- Admin Approval for Drivers:

  The admin panel allows for reviewing driver details before approval. Admins check submitted car information, including vehicle number and images, before granting access to the system. This ensures proper ride allocation.

- Flexible Pricing Model:

  Drivers can either set a fixed ride fare or enable a per-kilometer pricing system (e.g., ₹18 per km), giving them control over fare adjustments based on distance and demand.

- User-Specific Profiles:

  Passengers and drivers have distinct profile pages. Passengers can rate and review drivers based on ride experience, helping future users make informed decisions. Drivers can manage their trip history and earnings efficiently.

- Enhanced Ride Experience:

  The system integrates Google Maps for navigation and route planning, ensuring smoother travel. Real-time updates on available seats improve booking accuracy.

Chapter 4

Software Requirements

The proposed system utilizes a combination of technologies to develop an efficient and structured ride-sharing platform. The software requirements are detailed as follows:

- Frontend

  The graphical user interface (GUI) is built using Tkinter, a standard Python library that enables the creation of interactive and user-friendly interfaces. Tkinter provides built-in widgets and event-driven programming features, making it an ideal choice for developing the application's interface. The interface is enhanced with custom fonts, colors, and layout structures to ensure clarity and ease of use for both drivers and passengers.

- Backend

  The database stores:

  User Profiles: Separate entries for passengers and drivers, storing details like name, contact, and ride history.

  Ride Information: Includes trip creation details, booking status, pricing, and ride completion records.

  Admin Records: Stores driver approval requests, car details, and verification statuses.

- API & External Services

  Google Maps API used for visualizing ride routes, selecting pickup and drop-off locations, and redirecting users for navigation assistance.

- Development & Deployment Tools

  Python 3.x the core programming language for backend logic, database interactions, and API integrations.

  MySQL Connector for Python facilitates the connection between Python and MySQL, enabling smooth data transactions.

  PyCharm / VS Code the primary development environments used for coding, debugging, and managing the project efficiently. PyCharm offers advanced debugging tools for Python, while VS Code provides flexibility with multiple extensions.

Chapter 5

Project Design

The design of the ride-sharing system integrates various essential modules to ensure a smooth, reliable, and scalable user experience. The frontend architecture is responsible for handling user interactions including trip creation, login authentication, maps, and feedback. The backend logic manages processes such as role-based access, ride management, and chat-bot handling to enhance automation and user support. A centralized MySQL database supports the system by storing critical data related to users, rides, feedback, and more. This architecture ensures seamless data flow between components while maintaining performance and modularity across the application.

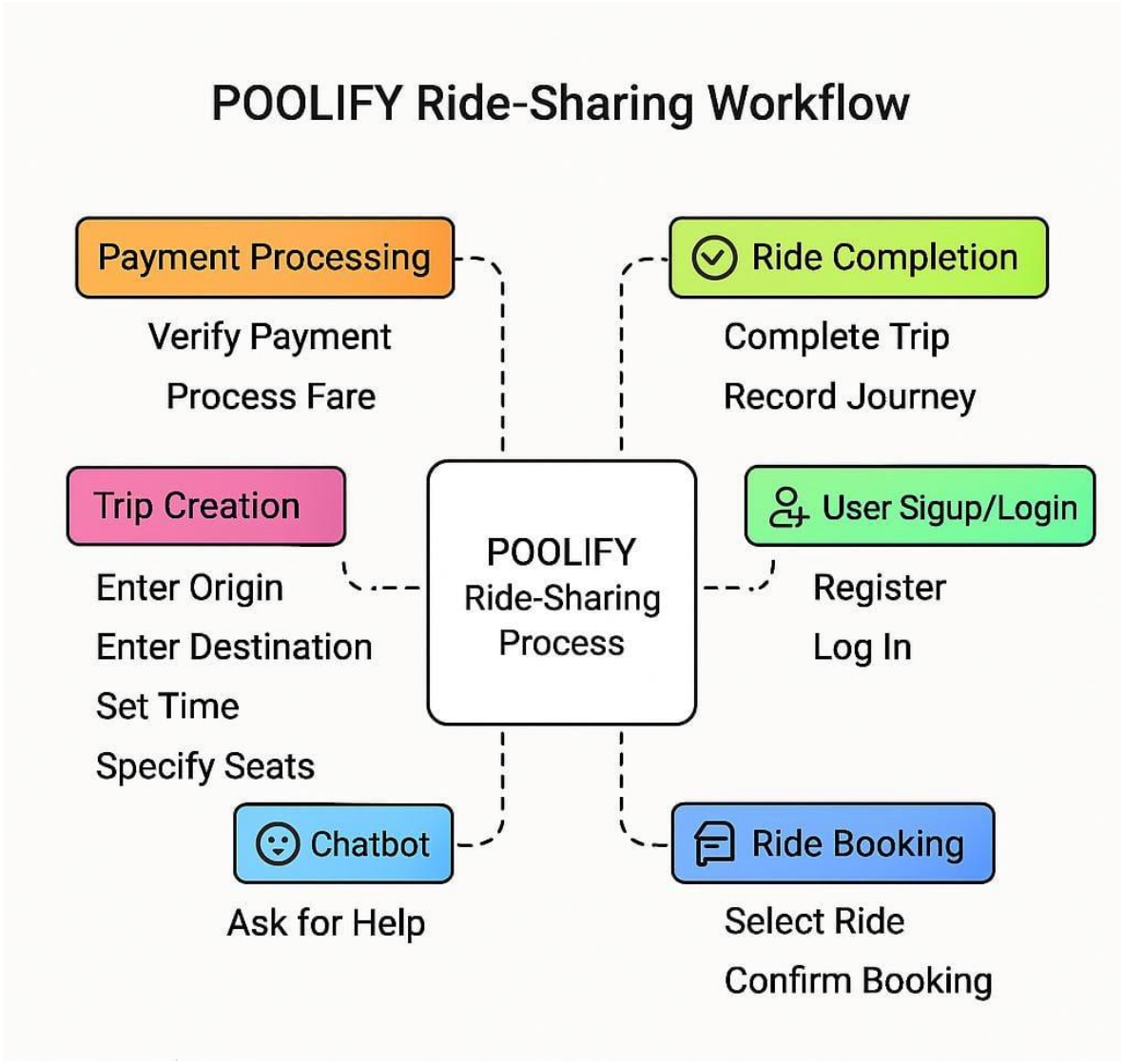

Fig 5.1 Work Flow

The above figure 2.2 illustrates the ride-sharing workflow of the POOLIFY system, showcasing a structured, process-driven approach that guides users—both passengers and drivers—through each phase of interaction within the application. This workflow ensures a streamlined experience, addressing core functional requirements from registration to trip completion, with supportive modules enhancing usability.

At the core of the diagram lies the POOLIFY Ride-Sharing Process, which integrates the following key modules:

• **User Signup/Login**: This module initiates user interaction with the system. It provides functionality for new users to register by entering their personal details and for existing users to securely log in. It ensures authentication and serves as the gateway to other features.

• **Trip Creation**: Designed primarily for drivers, this module allows them to create new ride entries by specifying origin, destination, travel time, and the number of available seats. It acts as the foundation for trip availability on the platform.

• **Ride Booking**: This module is used by passengers to search, select, and book available rides. It connects passengers with driver-created trips, updating the system in real-time to reflect seat availability and booking confirmation.

• **Payment Processing**: Once a ride is booked, the payment processing module ensures the secure handling of fare transactions. It verifies payment details, deducts the appropriate amount from the passenger's wallet or payment method, and updates the system accordingly.

• **Ride Completion**: After the journey concludes, this module records trip completion and archives relevant data, including trip summaries and timestamps. This step is essential for maintaining historical records and enabling accurate reporting.

• **Chat-bot**: Serving as an intelligent support tool, the chatbot module assists users throughout their experience. It addresses common queries, provides ride or booking assistance, and ensures real-time, automated help without the need for human intervention.

Each component in this workflow is interlinked to promote efficiency, user satisfaction, and reliable service delivery. The systematic arrangement of these modules ensures that both drivers and passengers have a clear and intuitive path to follow, while background processes such as payment and data recording operate seamlessly to support front-end operations.

Chapter 6

Project Scheduling

The Gantt chart for the POOLIFY project represents a structured timeline that outlines all major phases and tasks involved in the development of the system, mapped across a 14-week period. This schedule plays a vital role in managing the project's flow, distributing responsibilities, and monitoring task completion. The project is divided into two core phases: Project Conception and Initiation and Project Design and Implementation. Each of these phases contains multiple subtasks assigned to specific team members, with clearly defined start and end dates, durations, and a progress tracker that reflects task completion percentages.

During the initial weeks, the focus was on planning and groundwork activities. These included group formation, identifying the project topic, defining objectives, reviewing existing literature, and finalizing the scope of the mini project. This foundational phase was crucial for setting a clear direction for the project. Each member contributed to different aspects, such as topic research and interface planning, ensuring a collaborative start. Notably, all tasks under this phase reached 100% completion within the scheduled time, indicating effective coordination and timely execution.

Following the planning phase, the project transitioned into the development phase, which involved more technical and design-oriented work. Tasks included database design, module implementation, and writing reports. These were scheduled thoughtfully to minimize overlap and distribute workload evenly among team members. The use of color-coded bars on the Gantt chart visually distinguishes task timelines, making it easy to observe when each activity was active. For example, database design spanned multiple weeks, overlapping slightly with module development, showing the transition from planning to implementation.

Each task in the chart is supported by key details such as the assigned owner(s), estimated duration in days, and real-time completion status. This structure allows the team to pinpoint exactly where they stand in the development lifecycle and helps identify any tasks that may require additional attention or time. The consistent use of green (indicating 100% completion) reflects that the team maintained strong adherence to deadlines and project goals.

The Gantt chart thus serves not only as a visual schedule but as a project management tool that enhances transparency, supports accountability, and ensures that every member is aligned with the project's timeline. It helps avoid delays by showing dependencies, overlapping tasks, and remaining work—ultimately keeping the project on track and well-coordinated from initiation to final implementation.

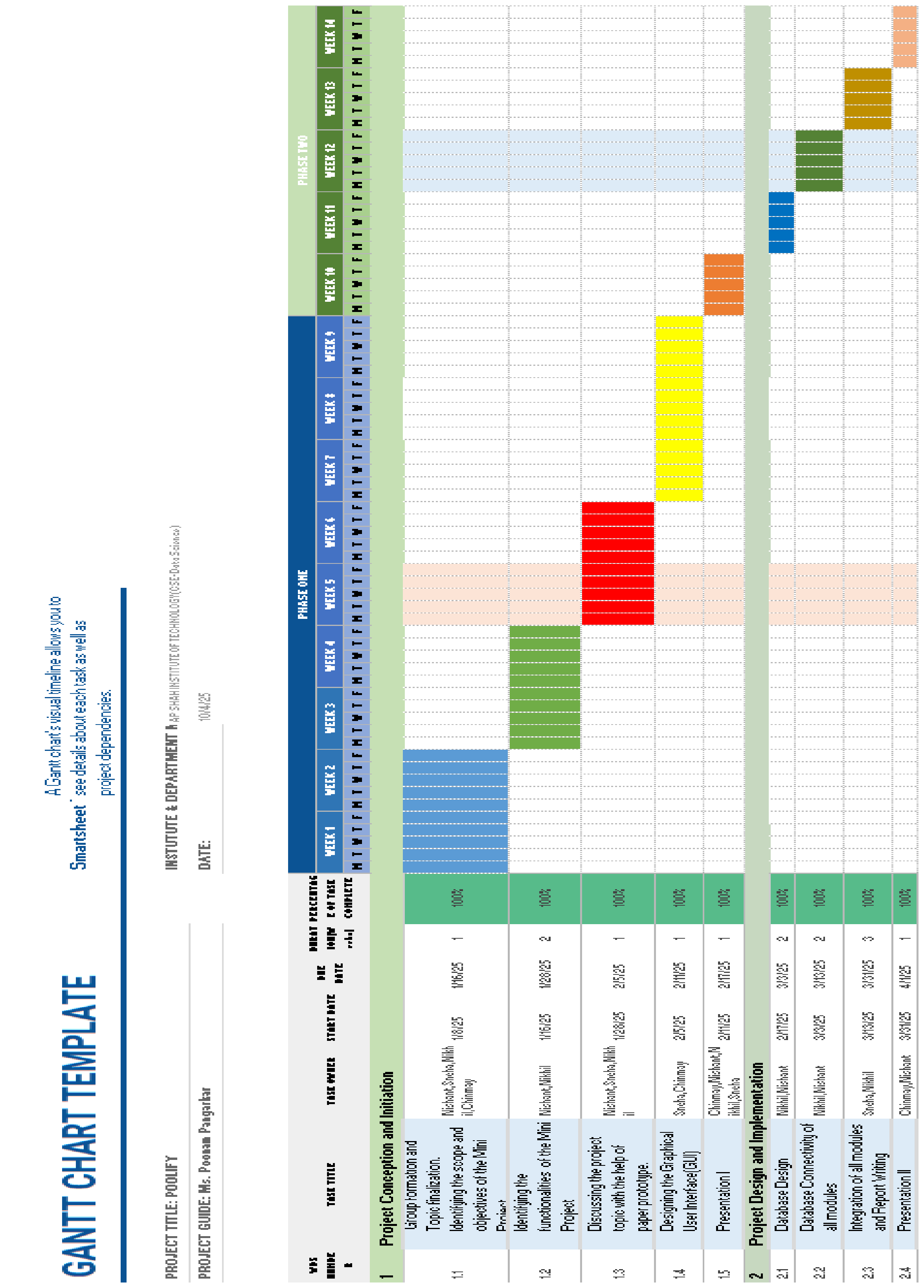

Fig 6.1: Gantt Chart

The fig 6.1 is a Gantt chart, which visually represents the timeline and progression of tasks for a project. It is organized by weeks (on the vertical axis) and divided into two main project phases: Project Conception and Initiation, Project Design and Implementation. Each phase includes various tasks, color coded for distinction, and spans over specific time periods (weeks). Task descriptions, deadlines, and responsibilities are listed, showing a clear project schedule, with progress and dependencies marked visually through bars of different colors. The chart is designed to help track the milestones and deliverables over time, facilitating project management and monitoring.

The project conception and initiation phase for the application was successfully completed by the end of the month around 01/02/25. This initial phase involved multiple foundational tasks such as group formation and topic finalization, which were completed during the first week of project commencement.

The project group was formed with four members: Nishant, Sneha, Nikhil, and Chinmay, and the selected topic was POOLIFY – A Ride-Sharing System. In the following week, the team focused on identifying the scope and objectives of the platform, outlining the key goals and defining the target users.

The identification of major functionalities was carried out by Nishant, Sneha, Nikhil, and Chinmay over a period of one week, from 16/01/25 to 28/01/25. These functionalities included ride creation, booking system, user role management, and chat-bot assistance.

The team collaboratively discussed the overall structure and logic of the project, creating a basic GUI design and paper prototype, which was finalized between 28/01/25 and 05/02/25.

The database connectivity required for the Python application was developed by Nishant, Sneha, Nikhil, and Chinmay during the period of 03/03/25 to 13/03/25, ensuring seamless integration with MySQL for real-time ride management and user interactions.

The integration of all graphical user interface (GUI) pages along with MySQL database connectivity, and the initial report writing, was carried out from 31/03/25 to 01/04/25, marking the successful completion of the implementation phase.

Lastly, the final presentation and documentation were prepared collaboratively by all four team members in the final phase of the project, highlighting their collective contributions and comprehensive understanding of the developed system.

Chapter 7

Results

The implementation system brought noticeable improvements in efficiency, user experience, and management. It simplified ride booking, trip creation, and user authentication, reducing errors and manual effort. The integration of maps enabled better navigation and ride planning for users.

Admins benefited from easy access to user data and driver applications, improving control and verification. The feedback module helped gather valuable user insights, guiding service enhancements and boosting overall satisfaction. Here are the outcomes and results observed during the development and execution of the system:

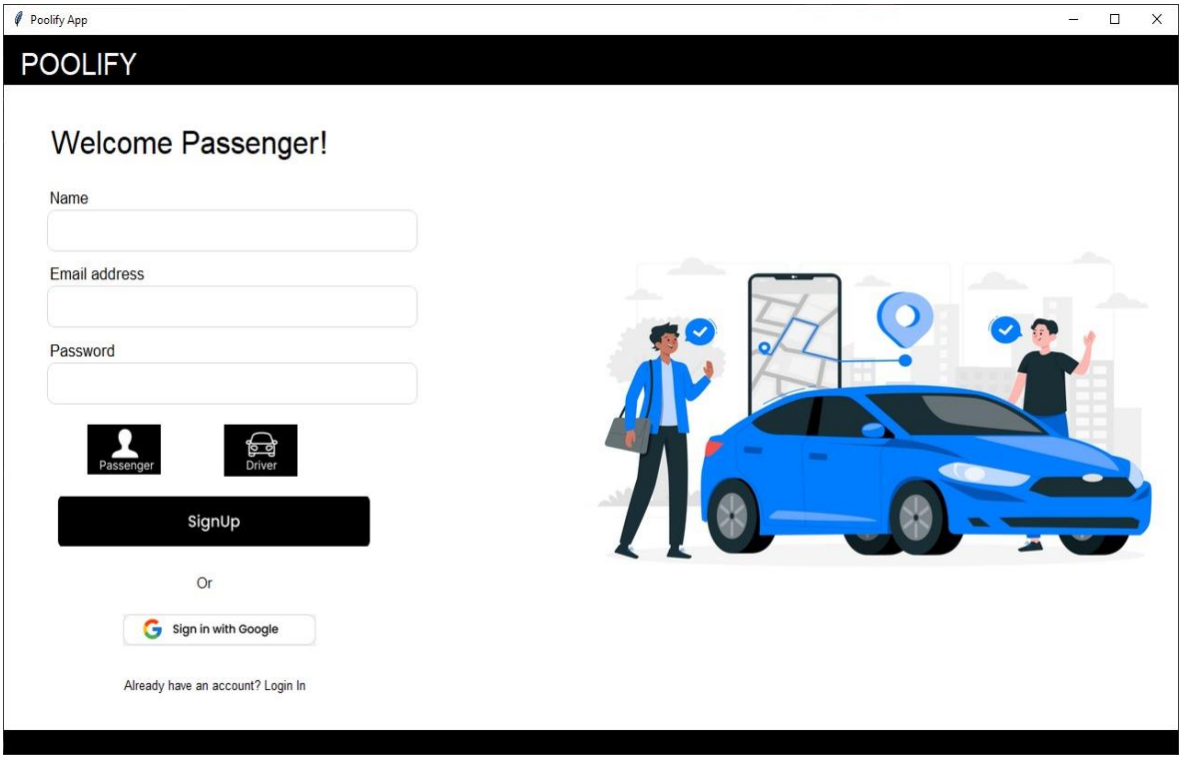

Fig 7.1: Signup Page

The fig 7.1: signup page serves as the entry point for new users to register on the platform. Users are required to enter basic details such as name, email, and password. Below the form, two distinct options are provided — Signup as Passenger and Signup as Driver — allowing users to choose their role in the system. Additionally, a "Signup with Google" option is included for faster registration using Google credentials. At the bottom, a link is provided for existing users to redirect to the login page, ensuring easy navigation.

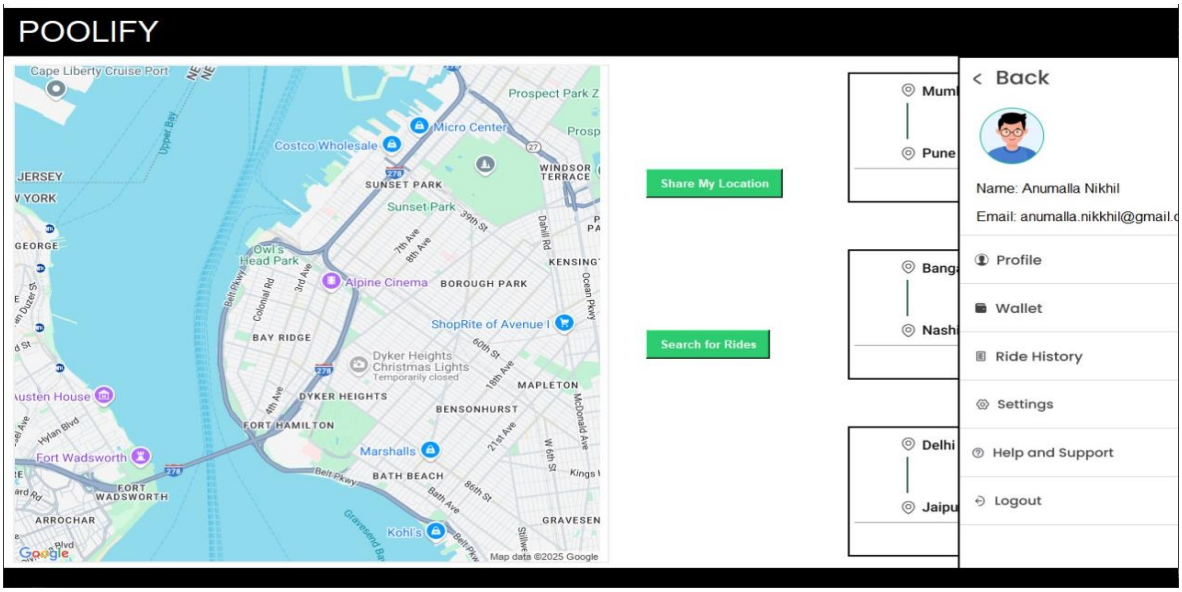

Fig 7.2: Passenger HomePage

The fig 7.2: The home page of the POOLIFY application is designed to provide users with a streamlined experience for ride searching and profile management. On the left side, an interactive Google Map is embedded to allow users to visualize nearby pickup and drop locations. Two green buttons labeled “Share My Location” and “Search for Rides” offer core functionality — enabling location sharing for better ride matching and initiating a ride search based on selected locations.

On the right side, the interface features a vertical panel displaying city options in pairs (e.g., Mumbai ↔ Pune, Bangalore ↔ Nashik, etc.) which likely serve as sample or quick-select route options. Upon clicking the user avatar or profile icon on the top right, a side panel slides in showing user details, including name and email ID. Below this, users can access different features such as Profile, Wallet, Ride History, Settings, Help and Support, and Logout, providing easy navigation to key account management sections.

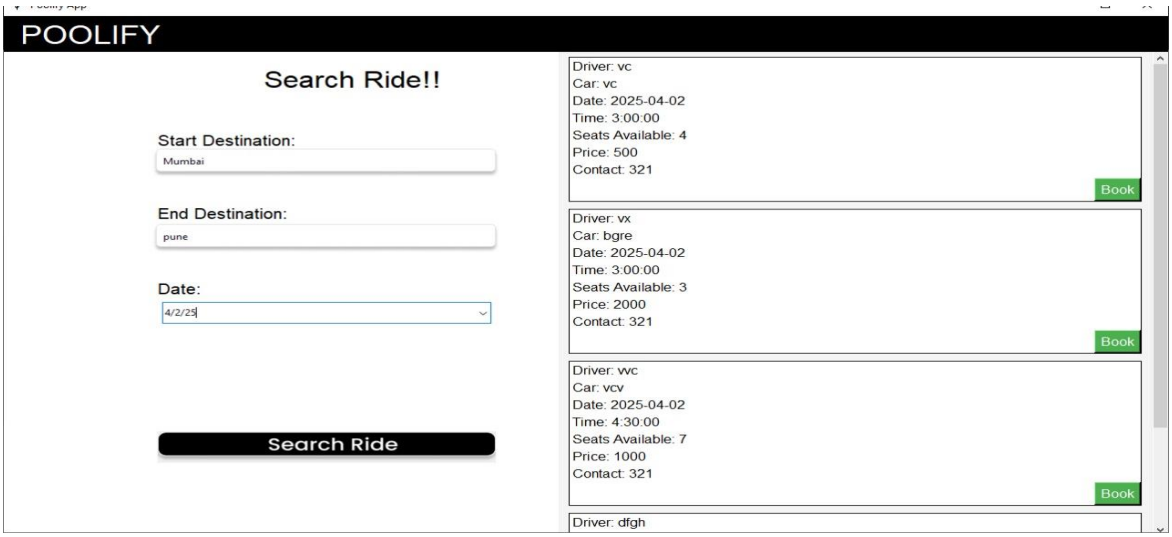

Fig 7.3: Search Ride Page

The fig 7.3: The Ride Search page in the POOLIFY application facilitates users in finding suitable rides between selected locations. On the left side, a simple form is provided where users can input their start destination, end destination, and select a date using a dropdown calendar input. A prominent Search Ride button initiates the ride search based on the entered criteria.

The right side dynamically displays a list of available ride options that match the search input. Each ride card provides detailed information including the driver's name, car, date, time, seats available, price, and contact number. A green Book button is available with each ride entry, allowing users to proceed with booking the selected ride instantly.

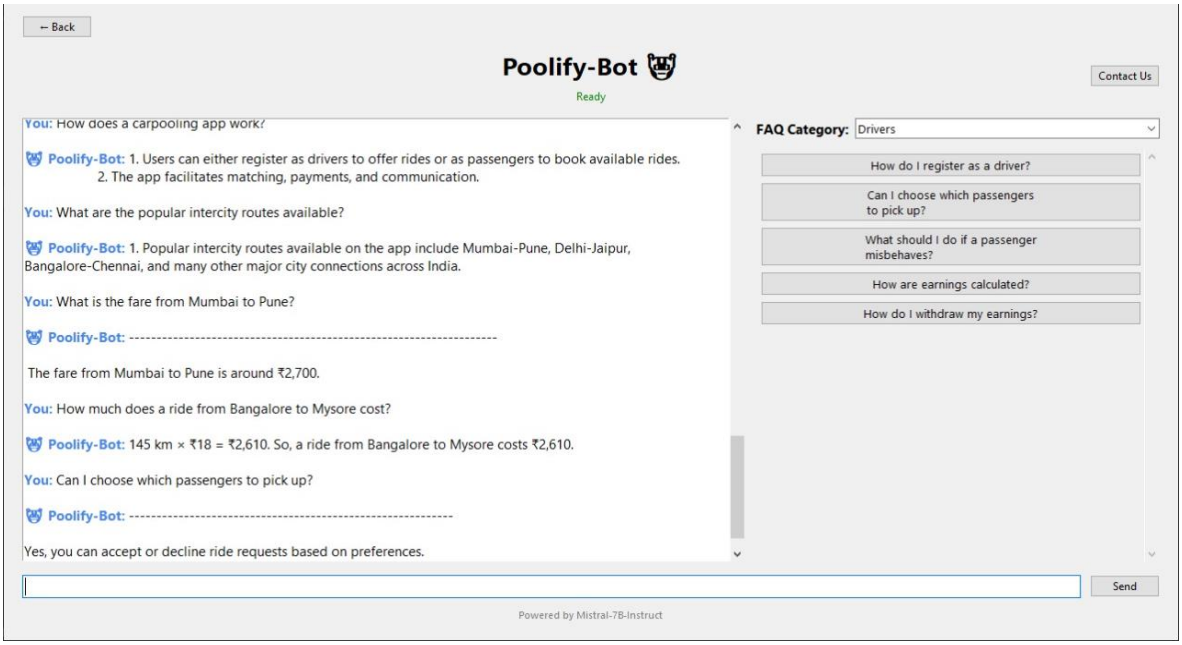

Fig 7.4: Query Chat-Bot Page

The fig 7.4:The Bot page provides an interactive chat-bot interface designed to assist users with common queries related to the platform. Users can ask questions in natural language, and the bot responds with relevant information such as how the app works, fare estimates, route details, and driver-specific concerns. A dropdown menu on the right allows users to filter FAQs by category, while predefined question buttons offer quick access to frequently asked topics. As shown in Fig 7.3, this setup enhances user experience by delivering quick, automated support directly within the app.

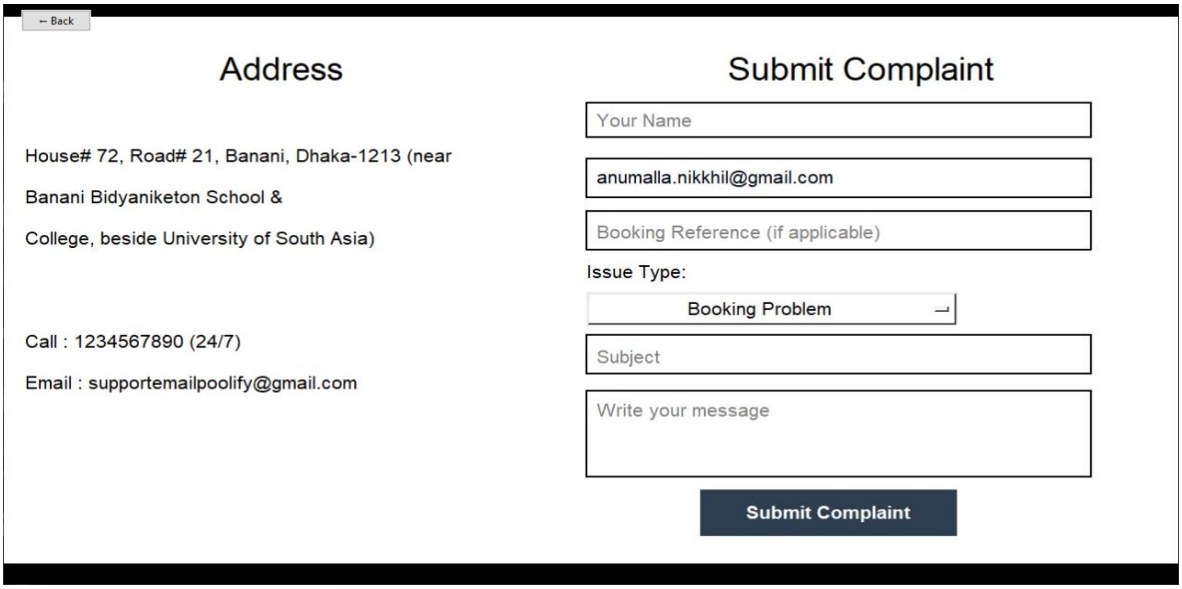

Fig 7.5: Feedback Page

The fig 7.5:The complaint submission page in the application allows users to report issues they encounter during their ride experience. The layout is divided into two sections for ease of access. The left panel provides the company's contact information, including address, phone number, and support email, available 24/7. On the right, users can fill out a structured form with fields such as name, email, booking reference (if applicable), issue type, subject, and a message box for detailed complaints. Once completed, the complaint can be submitted for prompt support. Fig 7.4 illustrates this user-friendly complaint interface designed to streamline customer assistance.

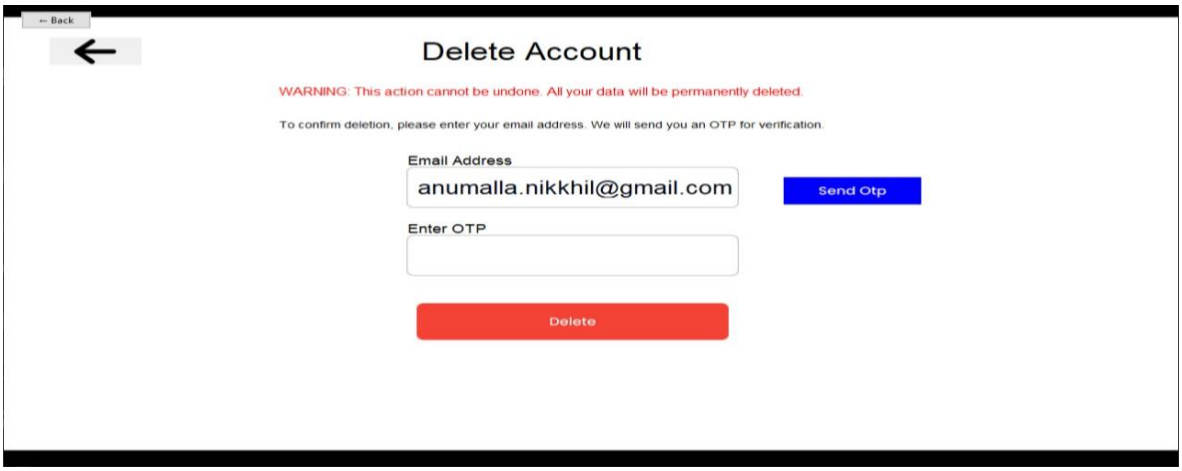

Fig 7.6: Delete Account Page

The fig 7.6:The Delete Account page in the POOLIFY application provides users with the option to permanently delete their account. At the top of the page, a clear warning message

is displayed in red, stating that the action is irreversible and all user data will be permanently deleted.

To proceed with the deletion, users must first enter their email address in the provided input field. After entering the email, they can click the "Send OTP" button, which triggers the system to send a one-time password to the entered email for verification. Below that, there is an input field to enter the received OTP.

Once the OTP is entered, users can click the red "Delete" button to complete the account deletion process. A back button on the top left corner allows users to return to the previous screen if they change their mind.

This page ensures secure and intentional account deletion with OTP verification, minimizing the risk of accidental data loss.

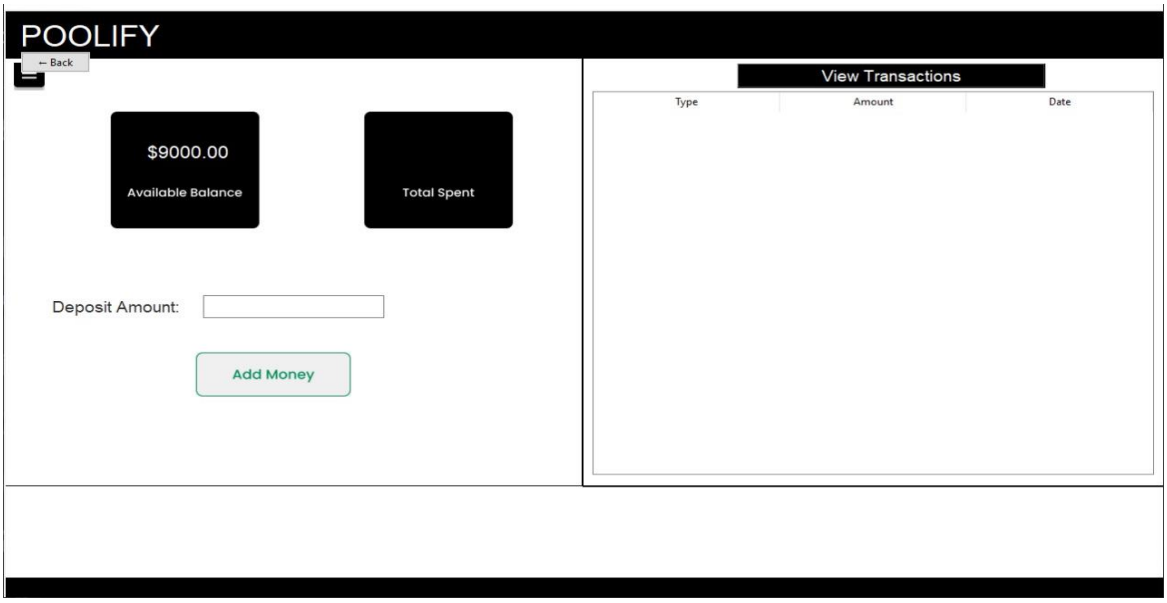

Fig 7.7: Wallet Page

The fig 7.7:The Wallet page in the POOLIFY application allows users to manage their in-app funds. On the left side of the interface, a balance card prominently displays the available balance, helping users keep track of their current wallet amount. Another card is designated for showing the total amount spent, although it appears to be empty in this screenshot.

Below the balance section, there is a field where users can enter a deposit amount, followed by an "Add Money" button, which enables users to top up their wallet.

On the right side of the screen, a transaction history panel titled "View Transactions" is available. This table is structured to show transaction type, amount, and date, though it is currently empty, possibly indicating no transactions have been made yet.

This page provides a simple and intuitive interface for monitoring balance, depositing funds, and reviewing transaction history within the POOLIFY app.

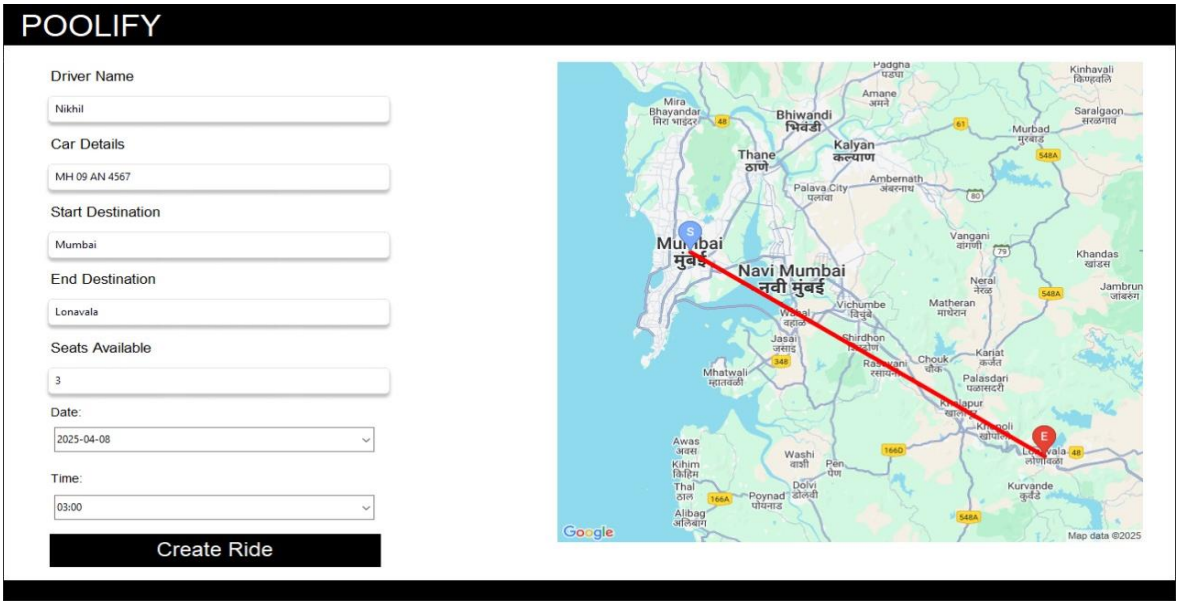

Fig 7.8: Ride Creation Page

The fig 7.8: The Trip Creation page in the POOLIFY application is designed for drivers to create new ride listings. On the left side of the interface, a form is provided where drivers can enter essential ride details including driver name, car details, start destination, end destination, number of available seats, date, and time of the ride. Once all fields are filled in, the driver can click the "Create Ride" button to submit the ride to the system.

On the right side of the screen, a Google Map is integrated to visually represent the route between the entered start and end destinations. A red line is drawn on the map to indicate the travel path from the origin to the destination, providing a visual cue for both the driver and potential passengers.

This page simplifies the process for drivers to list their available rides, while offering a clear geographic context to ensure accuracy and transparency.

Chapter 8

Conclusion

Poolify successfully establishes a structured and efficient ride-sharing platform, making it easier for both drivers and passengers to connect. By incorporating a user-friendly interface with Python Tkinter and a MySQL database, the system efficiently manages user data, ride history, and bookings. The chat-bot enhances user interaction by providing instant guidance, while the admin panel ensures smooth driver verification and overall system management.

The platform introduces flexible ride pricing and a seamless booking experience, improving accessibility and convenience. By addressing common ride-sharing challenges, this presents a well-organized approach to managing rides efficiently. Future improvements could include better UI/UX enhancements, additional personalization features, and expanded functionalities to refine the user experience further.

This project demonstrates how software-driven solutions can simplify ride coordination and enhance user engagement. With continuous enhancements can serve as a strong foundation for a more feature-rich and adaptable ride-sharing system.

References

[1] Agrawal, R., & Srikant, R. (1994). Fast algorithms for mining association rules. *Proceedings of the 20th International Conference on Very Large Data Bases (VLDB)*, 487–499.

[2] Ali, A., & Ahmad, A. (2021). Optimizing Carpooling Systems Using Real-Time Data and Machine Learning. *Journal of Transportation Technologies*, 11(2), 123–135.

[3] Shaikh, A. A., & Jamil, M. (2018). Smart Carpooling System Using Android Application. *International Journal of Computer Sciences and Engineering*, 6(5), 553–557.

[4] Tkinter GUI Programming in Python. https://docs.python.org/3/library/tkinter.html

[5] Google Maps API Guide. https://developers.google.com/maps/documentation

[6] Python MySQL Connector – Official Documentation. https://dev.mysql.com/doc/connector-python/en/