UniRide: Real-Time Vehicle Tracking and AI-Enhanced Driver Rating for Universities

By
Mahmudul Hasan Khan
ID: 201-15-13791

FINAL YEAR DESIGN PROJECT REPORT

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science and Engineering

Supervised by

Ms. Sharmin Akter
Assistant Professor
Department of Computer Science and Engineering
Daffodil International University

Co-Supervised by

Ms. Umme Ayman
Lecturer
Department of Computer Science and Engineering
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY
Dhaka, Bangladesh
January 12, 2025

APPROVAL

This Project titled "**UniRide: Real-Time Vehicle Tracking and AI-Enhanced Driver Rating for Universities**", submitted by **Mahmudul Hasan Khan,** ID No: **201-15-13791** to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 12 January, 2025.

BOARD OF EXAMINERS

Dr. Md. Zahid Hasan — **Chairman**
Associate Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Dr. Md Alamgir Kabir — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Mr. Tanvirul Islam — **Internal Examiner**
Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Dr. Md. Manowarul Islam — **External Examiner**
Associate Professor
Department of Computer Science and Engineering
Jagannath University

DECLARATION

I hereby declare that, this project has been done by me under the supervision of **Ms. Sharmin Akter, Assistant Professor,** Department of Computer Science and Engineering, Daffodil International University. I also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

Ms. Sharmin Akter
Assistant Professor
Department of CSE
Daffodil International University

Co-Supervised by:

11.01.25

Ms. Umme Ayman
Lecturer
Department of CSE
Daffodil International University

Submitted by:

Mahmud

Mahmudul Hasan Khan
ID: 201-15-13791
Department of CSE
Daffodil International University

ACKNOWLEDGEMENTS

This work would not have been possible without the support and contributions of many individuals over the past two semesters. I am deeply grateful to everyone who has assisted me in one way or another.

First, I express my heartfelt thanks and gratefulness to the almighty for His divine blessing making it possible for us to complete the **Final Year Design Project (FYDP)** successfully.

I am grateful and wish our profound indebtedness to **Ms. Sharmin Akter, Assistant Professor,** Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh. Deep knowledge and keen interest of my supervisor in the field of **Software Development** to carry out this project. Her endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this project.

I would like to express my heartfelt gratitude to the Head of the Department of Computer Science and Engineering, for his kind help in finishing my project and also to other faculty members and the staff of the Department of Computer Science and Engineering, Daffodil International University.

I would like to thank my entire course-mates at Daffodil International University, who took part in this discussion while completing the coursework.

Finally, I must acknowledge with due respect the constant support and patience of my parents.

ABSTRACT

The primary objective of this project is to enhance the efficiency, reliability and convenience of university transportation system through a real-time vehicle tracking and AI-enhanced driver review system. This Android-based application, named UniRide, utilizes modern mobile technology to provide students and staffs with precise, up-to-date, real-time information on bus locations, routes, schedules etc. The system also enables students to submit driver reviews which are analyzed and summarized by AI to provide actionable feedback for continuous service improvement. The development of UniRide primarily involves the use of Kotlin programming language. Various other packers are also used such as Jetpack Compose for frontend, Firebase for backend services, Google Maps API for map and location related tasks, and Gemini API for AI-driven driver review analysis. This application aims to reduce the uncertainties associated with traditional transport system, improve user experience, and enhance safety through various features like emergency helpline, occupancy alerts and announcement notification system. This report documents the project's development, from initial prototype design and user feedback collection to the implementation of core functionalities and performance optimizations. It highlights the iterative process of refining the app based on user input, ensuring a user-friendly interface and reliable performance. By addressing the challenges of real-time data synchronization, efficient data storage, and user privacy, this project sets a new standard for campus transportation systems, paving the way for future enhancements and broader applications.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

1.1 Overview

UniRide is an Android application designed to improve the university transportation experience by providing real-time vehicle tracking, up-to-date information and an AI-enhanced driver review system. Students and staff can view the precise location of university transports, visualize routes, access schedule information, and review drivers. Features like occupancy status, emergency helpline, and announcement notification system aims to make campus commutes more efficient, safe, and reliable.

1.2 Motivation

Reliable transportation is very important for university communities. Traditional transportation services often fall behind due to outdated schedules and lack of real-time updates. Many universities still rely on fixed timetables, leaving students uncertain about bus arrivals and departures. This can lead to missed classes, meetings, and long waiting times. The increasing of smartphone usage and the development of mobile technologies have brought attention to how crucial it is to include real-time information in day-to-day life. Applications that utilize real-time data are in high demand, with an expected 2.87 billion smartphone users by 2020. [1] With the widespread use of smartphones and advancements in mobile technology, there is a growing demand for applications that provide real-time data to enhance daily activities. Despite attempts to address these issues, previous solutions for transportation systems often lacked the necessary technology to offer real-time updates, making them less effective.

UniRide aims to fill this gap by integrating modern mobile technology and AI to offer a comprehensive solution for university transportation needs. The app provides several advantages:

- **Enhanced Reliability:** Real-time tracking reduces uncertainty, allowing users to plan their commutes better.

- **Improved Safety:** Alert notifications and real-time updates ensure users are informed of any critical situations promptly.
- **User Satisfaction:** Accurate schedule and route information, coupled with the ability to review drivers, ensures a better overall user experience.
- **Service Improvement:** AI-driven analysis of driver reviews provides actionable feedback, promoting continuous service enhancement.

By addressing these critical areas, UniRide not only improves the day-to-day commuting experience for students and staff but also contributes to the overall efficiency and reliability of university transportation systems.

1.3 Problem Statement

University transportation systems often suffer from inefficiencies such as unpredictable schedules, lack of real-time updates, and inadequate safety features. These issues lead to several negative impacts on students and staff:

- **Unpredictable Schedules:** Without reliable bus schedules, students and staff face uncertainty about bus arrival and departure times, leading to missed classes, meetings, and other important commitments.
- **Lack of Real-Time Updates:** The absence of real-time information about bus locations leaves commuters in the dark about delays or changes in routes, causing frustration and inefficiency in planning their travels.
- **Inadequate Safety Features:** The lack of alert notifications delays response times in critical situations, compromising the safety and well-being of commuters.
- **Overcrowding Issues:** Without occupancy status, users might board already full buses, leading to uncomfortable and potentially unsafe travel conditions.
- **Limited Feedback Mechanism:** Without a system to review and rate drivers, there is little opportunity for service improvement, leading to persistent issues with driver behavior and overall service quality.

- **Lack of Communication:** The absence of a robust notification system for announcements from admins and emergency contacts reduces the effectiveness of communication during critical times.

UniRide seeks to address these challenges by providing real-time tracking, accurate schedule information, and an AI-powered driver review system, thereby enhancing the overall transportation experience for the university community.

1.4 Objectives

The primary objectives of UniRide are as follows:

- **Real-Time Bus Tracking:** To provide users with accurate, real-time tracking of university buses, reducing uncertainty and enabling better trip planning.
- **Comprehensive Route Information:** To offer detailed information about bus routes and schedules, ensuring that users have access to current and reliable transit data.
- **Occupancy Status Display:** To show users the current occupancy status of buses, helping them make informed decisions about their travel options and avoid overcrowding.
- **AI-Enhanced Driver Review System:** To implement an AI-driven review system that allows students to evaluate drivers. The AI will analyze and summarize feedback, facilitating continuous improvement in service quality.
- **Emergency Communication Features:** To include a notification system for announcements from administrators and provide users with quick access to emergency helplines, ensuring effective communication during critical situations.
- **User-Friendly Interface:** To develop an intuitive and accessible user interface that makes it easy for users to navigate the app and utilize its features effectively.

These objectives aim to enhance the overall efficiency, safety, and user satisfaction of the university transportation experience.

1.5 Project Outcome

The successful completion of the UniRide project will result in a comprehensive, user-friendly mobile application designed to enhance the university transportation experience. The primary outcomes include:

- **Enhanced Commuting Efficiency:** Real-time vehicle tracking and up-to-date route and schedule information will significantly reduce waiting times and improve planning for students and staff.
- **User Satisfaction:** The intuitive UI and valuable functionalities such as the notification system and driver review platform will enhance overall user experience and satisfaction.
- **Operational Insights:** Administrators will benefit from actionable insights provided by the AI-driven review system, allowing for continuous improvement of the transportation services.
- **Environmental Impact:** By promoting the use of public transportation and optimizing bus routes, the application will contribute to reducing the university's carbon footprint and traffic congestion.

Overall, UniRide will set a new standard for campus transportation systems, fostering a more efficient, safe, and user-centric commuting environment for the university community.

1.6 Organization of the Report

This report is organized into six chapters, each focusing on different aspects of the UniRide project.

- **Chapter 1: Introduction** provides an overview of the project, including its objectives, background, problem statement, scope, and the overall organization of the report.
- **Chapter 2: Background** discusses relevant previous works and studies related to real-time tracking systems and AI-enhanced review mechanisms, highlighting the current state of technology and identifying gaps that UniRide aims to address.

- **Chapter 3: Research Methodology** outlines the methods, tools, and approaches employed in developing and analyzing the UniRide project, including data collection, system design, and testing procedures.
- **Chapter 4: Implementation and Results** describes the development process of UniRide, presenting system features, functionalities, and key results obtained during testing and evaluation.
- **Chapter 5: Engineering Standards and Design Challenges** explores the engineering principles, standards followed during the project, and challenges encountered during the design and development phases, along with how they were addressed.
- **Chapter 6: Conclusion** summarizes the findings of the project, evaluates its success in meeting the objectives, and discusses potential future enhancements or research opportunities.

This structured organization ensures a comprehensive understanding of the project, facilitating a clear presentation of the research and development process.

1.7 Summary

In summary, UniRide aims to revolutionize university transportation by addressing common issues such as unreliable schedules and lack of real-time information. By leveraging modern mobile technology and AI, the app provides real-time tracking, comprehensive route information, and a platform for driver reviews. This chapter has introduced the project's background, problem statement, objectives, scope, limitations, and the structure of this report. The following chapters will delve deeper into the development process, methodologies, and findings, providing a comprehensive overview of the project.

Chapter 2
Background

2.1 Overview

In recent years, the demand for real-time transportation tracking systems has grown significantly, particularly in educational institutions where efficient and reliable transport is crucial for students and staff. The development of mobile applications has made it easier to address these needs by providing users with immediate access to information regarding bus schedules, locations, and overall transportation management. This chapter explores existing literature on real-time tracking systems, highlights relevant advancements in technology, and identifies gaps that the UniRide application aims to fill. By examining various approaches to transport tracking and user feedback mechanisms, I can better understand how UniRide can contribute to improving university transportation.

2.2 Literature Review

Several studies have investigated the implementation of real-time tracking systems in transportation, focusing on different aspects such as user interface design, data accuracy, and user engagement. Studies have indicated that the delivery of accurate and timely information by real-time tracking greatly improves user experience. Many existing applications utilize GPS technology to provide live updates on vehicle locations, while others integrate user feedback systems to enhance service quality. This solution aligns with current trends in mobile technology, as more users rely on smartphones to meet their needs. However, a common challenge across these platforms is the lack of comprehensive features that meet specifically to the needs of university commuters. Previous works often fall short in addressing issues such as overcrowding, safety communication, and the integration of AI to analyze user feedback. While older systems frequently needed more upkeep and were more expensive, more recent methods profit from the flexibility and cost-effectiveness provided by open-source platforms like Android. Let’s explore both global platforms and Bangladesh-specific services to acquire a comprehensive understanding of the current landscape:

International Platforms:

While the international platforms discussed here are not available in our country, they provide valuable insights into features and functionalities that could enhance our system.

- **DoubleMap Bus Tracker:**

  Link: https://play.google.com/store/apps/details?id=com.doublemap.iu

  This platform offers real-time tracking of buses and provides users with live updates. However, it lacks features such as comprehensive schedules and reporting tools, which limits its effectiveness for users seeking detailed information about bus operations. [6]

  The tracking page of the app is displayed in figure 2.2.1 below, where multiple University of Michigan's buses are displayed with distinct colored markers and bus icons.

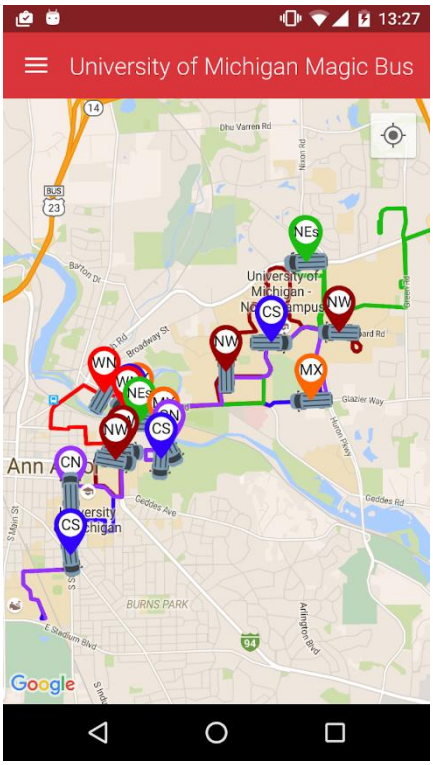

Figure 2.2.1: DoubleMap Bus Tracker's bus tracking page

- **Chalo - Live Bus Tracking App:**

  Link: https://play.google.com/store/apps/details?id=app.zophop

  While this app provides live bus tracking, it is not tailored specifically for university transportation. Recent reviews highlight significant issues with its tracking capabilities, stating that the live tracking feature is unreliable, failing to work properly at least 60% of the time. This unreliability leads to user frustration and disrupted travel plans. [7]

  Figure 2.2.2 below shows the app's tracking page, with various buses and the start and finish of their respective routes displayed.

Figure 2.2.2: Chalo - Live Bus Tracking App’s bus tracking page

Bangladesh-Based Services:

Bangladesh-based services also present opportunities for improvement, particularly in addressing the unique challenges faced by university commuters. These local applications often meet the specific needs of students and staff, yet they still exhibit various limitations that can hinder user satisfaction and overall effectiveness. By analyzing these

services, I can identify key areas for enhancement and innovation within our UniRide application.

- **JU Transport:**

  Link: https://play.google.com/store/apps/details?id=com.sbiitju.jugreenbus

  This service provides real-time tracking, announcement notifications, and a community chat feature for users. However, it lacks essential functionalities such as schedules, reporting tools, and a driver review system. Additionally, the user interface is outdated, which can deter users from fully engaging with the app and its features. [8]

  The app's tracking screen is displayed in Figure 2.2.3 below, where a light-green marker indicates the bus's location.

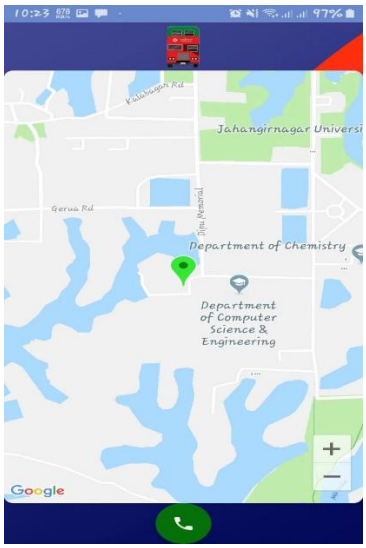

Figure 2.2.3: JU Transport's bus tracking page

- **Amader BRTC:**

  Link: https://play.google.com/store/apps/details?id=live.dingi.brtc.amader

  While this platform offers real-time tracking, it is not tailored specifically for universities. Recent reviews have highlighted that the app often malfunctions and is not updated regularly. This inconsistency can lead to confusion and

inefficiency, making it difficult for users to rely on the app for accurate information. [9]

The app's tracking page, depicted in Figure 2.2.4 below, displays the start and finish of route, bus location, and different stoppages.

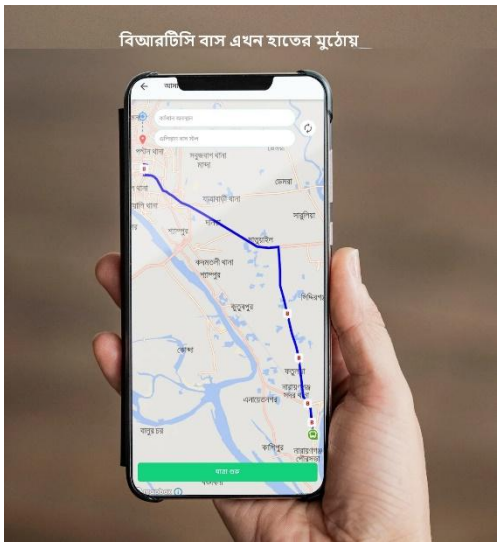

Figure 2.2.4: Amader BRTC's bus tracking page

2.3 Gap Analysis

To better understand the strengths and weaknesses of existing transportation tracking solutions, a gap analysis of several international platforms and Bangladesh-specific services is presented below in table 2.3.1.

Table 2.3.1: Gap Analysis

Features	**DoubleMap Bus Tracker**	**Chalo – Live Bus Tracking App**	**JU Transport**	**Amader BRTC**	**UniRide (Proposed)**
Real-time vehicle tracking	Yes	Unreliable	Yes	Yes	Yes
Vehicle occupancy status	No	No	No	No	Yes

Alert notification system	No	Yes	No	No	Yes
Routes information with visualization	Yes	Yes	Yes	Yes	Yes
Driver review and rating	No	No	No	No	Yes
AI-based review summarization	No	No	No	No	Yes
Emergency contacts	No	No	Yes	No	Yes
Regular schedule management	No	Yes	No	No	Yes
Issue reporting system	No	No	Yes	Yes	Yes
Community chat system	No	No	No	No	Yes
Administrator announcements	Yes	No	No	No	Yes
Running vehicle's driver information	No	No	No	No	Yes
Driver management and approval system	No	No	No	No	Yes

This comparison illustrates that while various platforms offer real-time tracking and some additional features, many still fall behind in delivering a comprehensive user experience. These insights inform the development of UniRide, guiding the integration of essential

features and improvements to ensure a reliable, user-friendly transportation solution tailored to the needs of university commuters.

2.4 Open Issues

Despite advancements in real-time tracking and user feedback systems, several open issues remain:

- **Lack of AI-Driven Review Analysis:** Currently, there are no transportation systems available that summarize user reviews using AI. The integration of AI in UniRide is an innovative step, but it is limited to summarizing recent user reviews.
- **User Interface Complexity:** Balancing a comprehensive feature set with ease of use is challenging. Many user interfaces either oversimplify the information or present it in a way that can be overwhelming for users.
- **Data Privacy and Security:** Ensuring the privacy and security of user data while providing real-time updates and collecting user feedback continues to be a critical concern.
- **Scalability:** As user numbers and data volume grow, maintaining performance and responsiveness of the app becomes increasingly difficult.
- **Driver Familiarity with Technology:** Some drivers may not be familiar with using mobile applications, which can hinder their ability to effectively use the system.

Addressing these open issues is crucial for developing a robust, user-centric transportation solution.

2.5 Summary

This chapter reviewed the current state of research and development in real-time vehicle tracking and AI-enhanced driver review systems. While significant progress has been made, there are clear gaps that need to be addressed. By identifying these gaps and open issues, I can better understand the areas where UniRide can innovate and provide a more comprehensive and user-friendly solution for university transportation.

Chapter 3
Research Methodology

3.1 Requirement Analysis & Design Specification

This phase defines the app's requirements and creates detailed system designs to ensure effective development.

3.1.1 Overview

This chapter delves into the essential aspects of analyzing requirements and defining the designs for UniRide. I will cover the system design, the hardware and software requirements, project management strategies, and the financial analysis that underpins the project. This comprehensive approach ensures that all components work harmoniously to deliver a robust and efficient transportation solution for university communities.

3.1.2 System Design

The design of UniRide revolves around providing a seamless and intuitive user experience while ensuring reliable real-time tracking and efficient data handling. The system is divided into several key components and includes various diagrams to illustrate the system's architecture and interactions:

- **User Interface (UI):** The UI is designed using Jetpack Compose to create a clean, responsive, and user-friendly interface. It includes real-time maps, route details, schedule information, and an easy-to-use review system.
- **Backend Services:** Firebase is used for database operations, user authentication, and storing media files. This handles data storage, retrieval, and synchronization in real-time, ensuring that users always have access to the latest information.
- **AI Review System:** The Gemini API is integrated to analyze driver reviews. This AI component processes user feedback, summarizes it, and provides actionable insights for continuous service improvement.

- **Real-Time Tracking:** Google Maps API is used to provide real-time tracking of university buses. This feature displays the current location and distance of buses, and updates map visualization in real-time.
- **Notification System:** This system allows administrators to send important announcements and emergency notifications directly to users, with the ability to target students and drivers separately, ensuring timely and precise communication.

The system design of UniRide ensures a seamless integration of real-time tracking, AI-driven insights, and user-friendly features, creating an efficient and reliable transportation solution tailored to the needs of the university community.

3.1.3 Hardware/Software Requirements

Hardware Requirements:

- **Android Devices:** Various models for testing, including smartphones and tablets.
- **Virtual Devices:** Emulators to simulate different Android devices and screen sizes for comprehensive testing.
- **Development Workstations:** Computers with sufficient processing power and memory to handle development tasks efficiently.

Software Requirements:

- **Android Studio:** Integrated Development Environment (IDE) for building and testing the UniRide app.
- **Kotlin:** Programming language used for developing the app.
- **Firebase:** Backend services for real-time database operations and user authentication.
- **Google Maps API:** For map integration, providing directions, and geocoding services.
- **Gemini API:** For AI-driven analysis of driver reviews.

- **Version Control:** Git for version control and GitHub for source code management.

3.1.4 Use Case Diagram

Each use case and the actors involved are described in depth below:

Actors:

1. **Student:** University students using the app for their daily commute. Students interact with the app to view announcements, find nearby buses, check routes and schedules, review drivers, report issues, participate in community chats, and view driver information and reviews.
2. **Driver:** University bus drivers who share their bus location and manage their account. Drivers use the app to share their bus location in real-time, view their reviews, and manage their account, which includes creating and updating their information.
3. **Admin:** University administrators responsible for managing the transportation system. Admins post announcements, manage bus information, routes, and schedules, approve drivers, and view reported issues. They have the ability to create, update, and delete operations related to their tasks.

Use Cases:

1. **Login:**
   i. **Description:** Allows actors to log into the system.
   ii. **Actors:** Student, Driver, Admin
2. **Logout:**
   i. **Description:** Allows actors to log out of the system.
   ii. **Actors:** Student, Driver, Admin
   iii. **Extends:** Login
3. **View Announcements:**
   i. **Description:** Enables students to view announcements posted by the admin.
   ii. **Actors:** Student

4. **View Nearby Buses:**
    i. **Description:** Allows students to view buses near their location.
    ii. **Actors:** Student
5. **View Routes:**
    i. **Description:** Enables students to view the routes of buses.
    ii. **Actors:** Student
6. **View Schedules:**
    i. **Description:** Allows students to view bus schedules.
    ii. **Actors:** Student
7. **Review Drivers:**
    i. **Description:** Allows students to submit reviews for bus drivers.
    ii. **Actors:** Student
8. **Report Issues:**
    i. **Description:** Enables students to report any issues they encounter.
    ii. **Actors:** Student
9. **Community Chat:**
    i. **Description:** Allows students to participate in community chats.
    ii. **Actors:** Student
10. **View Drivers:**
    i. **Description:** Enables students to view driver profiles.
    ii. **Actors:** Student
11. **View Reviews:**
    i. **Description:** Allows students to view reviews of drivers.
    ii. **Actors:** Student
12. **Share Bus Location:**
    i. **Description:** Allows drivers to share their bus location in real-time.
    ii. **Actors:** Driver
13. **View Own Reviews:**
    i. **Description:** Enables drivers to view reviews submitted about them.
    ii. **Actors:** Driver

14. **Manage Account:**
    i. **Description:** Allows drivers to create and update their account information.
    ii. **Actors:** Driver
    iii. **Includes:** Create Account, Update Account
15. **Post Announcement:**
    i. **Description:** Enables admin to post announcements for students.
    ii. **Actors:** Admin
16. **Manage Bus Information:**
    i. **Description:** Allows admin to create, update, and delete bus information.
    ii. **Actors:** Admin
    iii. **Includes:** Create Bus Info, Update Bus Info, Delete Bus Info
17. **Manage Routes:**
    i. **Description:** Enables admin to create, update, and delete bus routes.
    ii. **Actors:** Admin
    iii. **Includes:** Create Route, Update Route, Delete Route
18. **Manage Schedules:**
    i. **Description:** Allows admin to create, update, and delete bus schedules.
    ii. **Actors:** Admin
    iii. **Includes:** Create Schedule, Update Schedule, Delete Schedule
19. **Approve Drivers:**
    i. **Description:** Enables admin to approve driver registrations.
    ii. **Actors:** Admin
20. **View Issues:**
    i. **Description:** Allows admin to view issues reported by students.
    ii. **Actors:** Admin

Figure 3.1.4.1 below illustrates the interactions between the different types of users (actors) and the functionalities (use cases) within the UniRide application. The diagram identifies three main user roles: Student, Driver, and Admin, each interacting with specific features of the system.

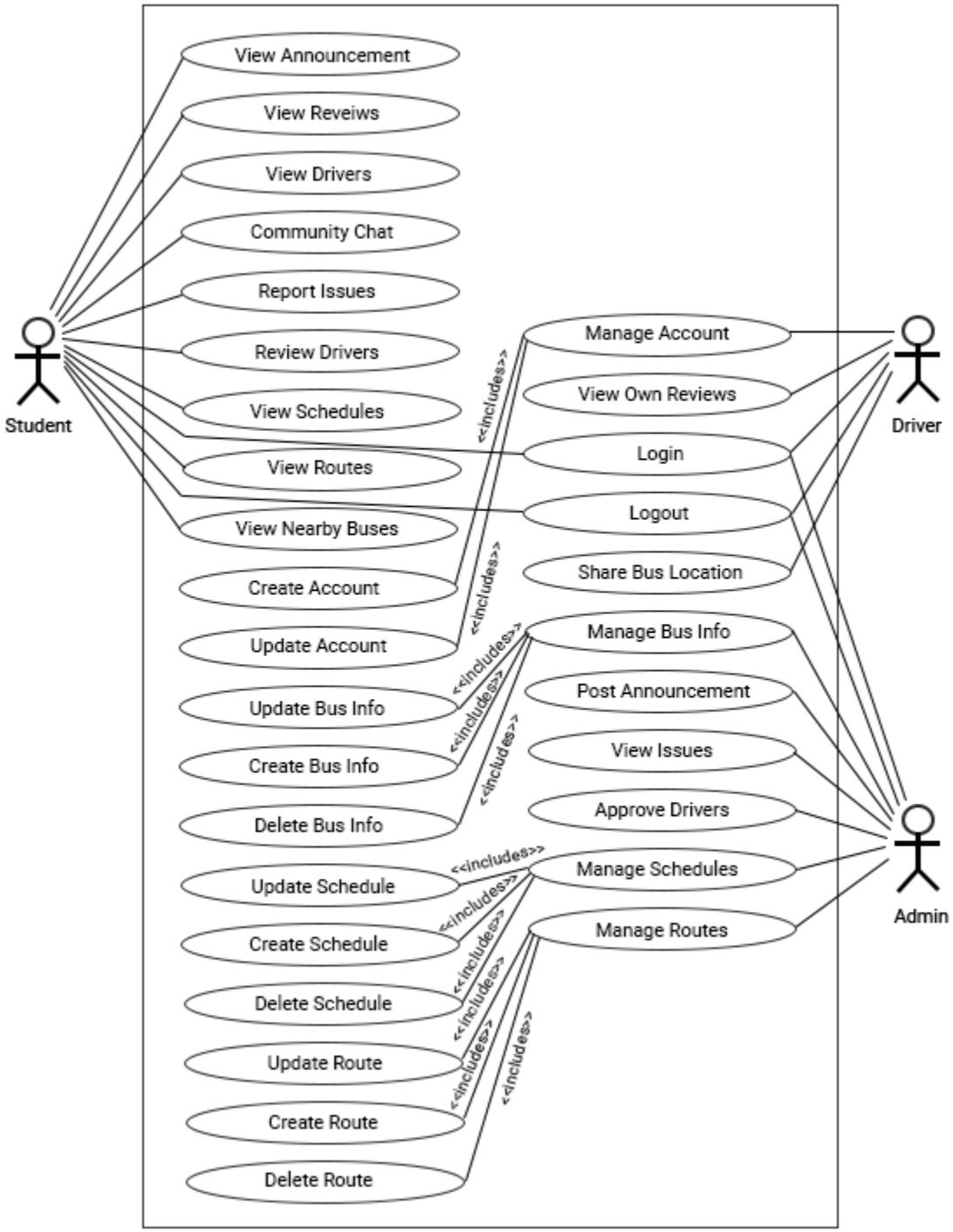

Figure 3.1.4.1: Use Case Diagram

3.1.5 Class Diagram:

Figure 3.1.5.1 below illustrates the core components and interactions within the UniRide application. The diagram depicts three main user roles: Student, Driver, and Admin, each with specific functionalities to manage different aspects of the system.

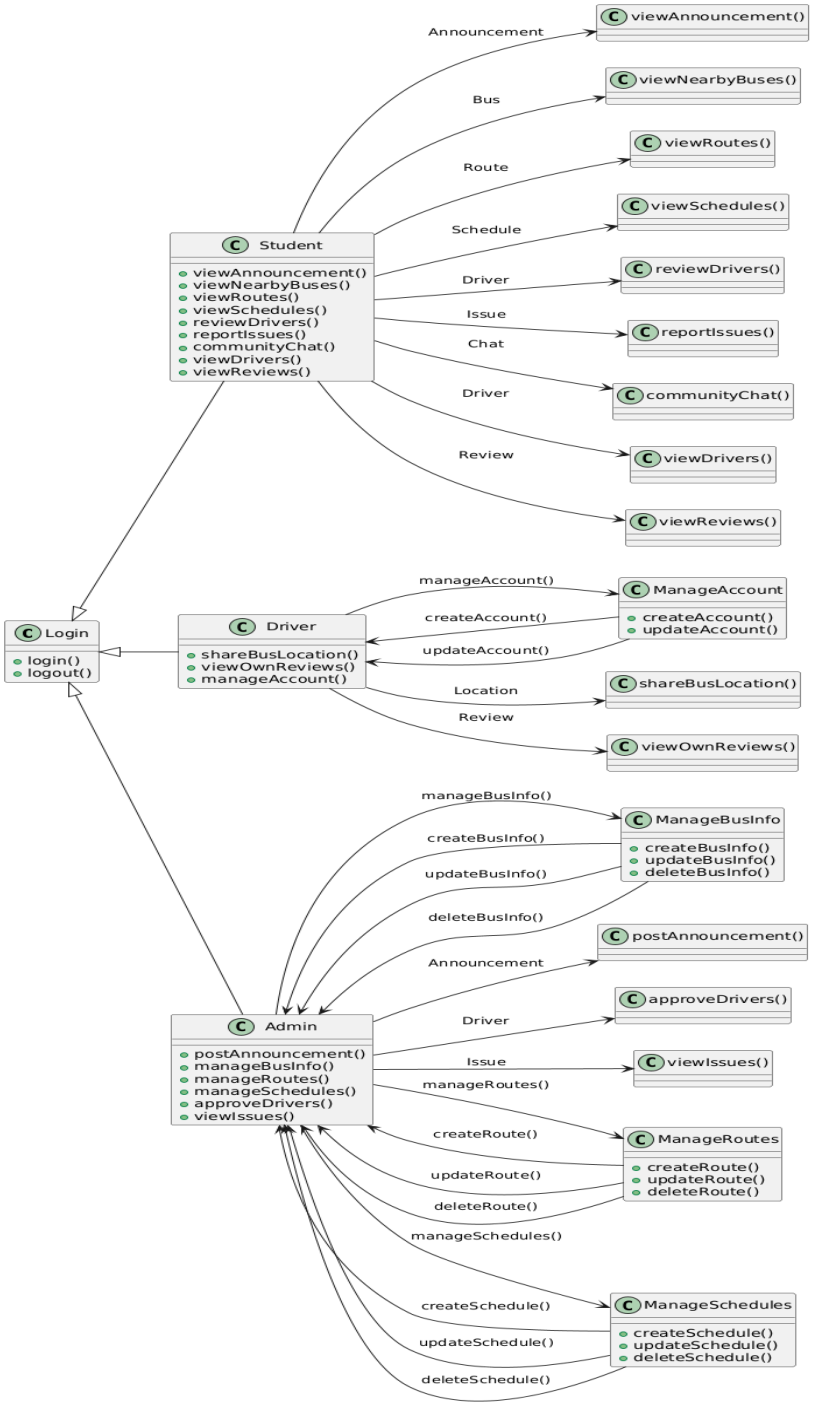

Figure 3.1.5.1: Class Diagram

3.1.6 ER Diagram:

Figure 3.1.6.1 ER Diagram below outlines the data structure for the UniRide application, showcasing the relationships between different entities within the system. The key entities include Student, Driver, Admin, Announcement, Route, Schedule, Bus and Review.

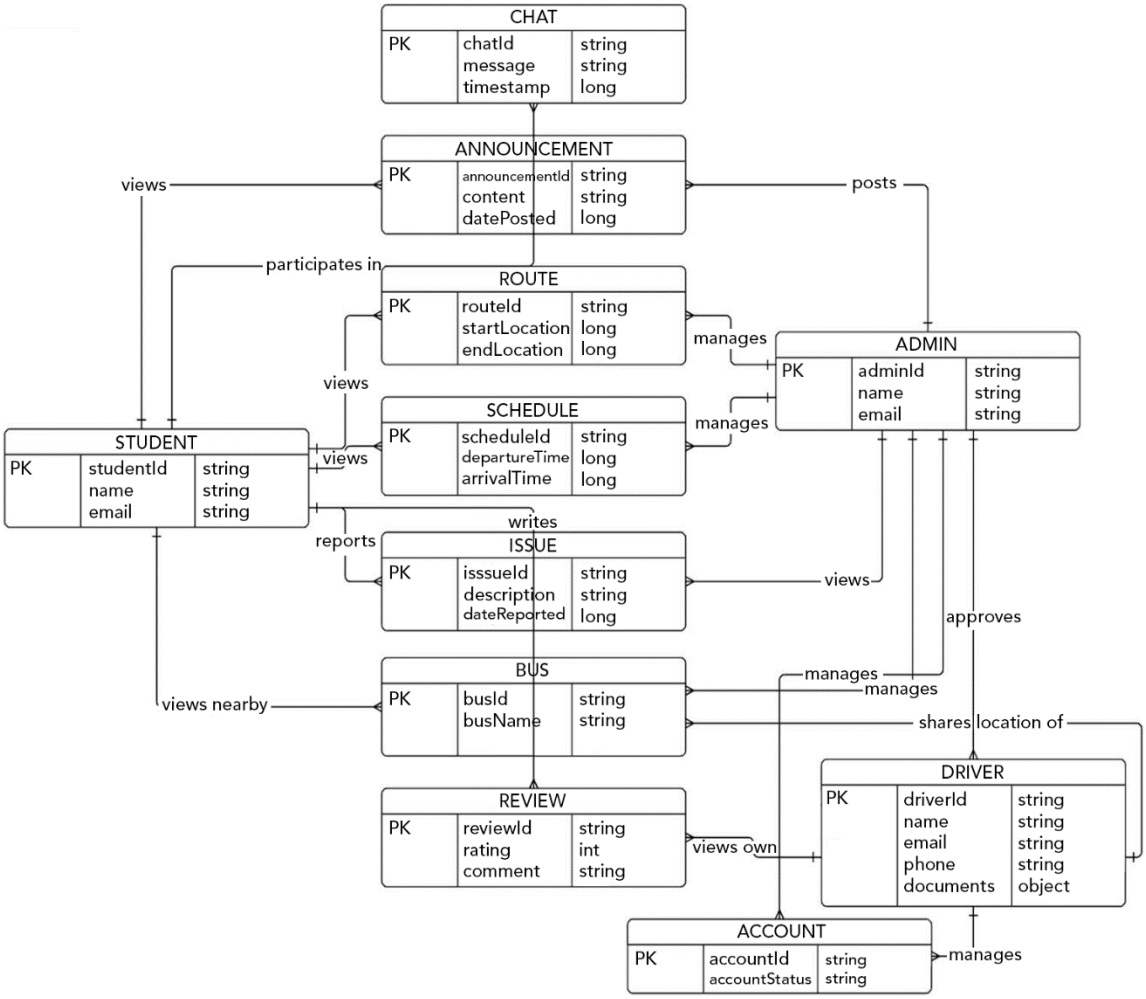

Figure 3.1.6.1: ER Diagram

By understanding the roles and interactions of each actor and visualizing the system architecture through these diagrams (Figure 3.1.4.1, Figure 3.1.5.1, Figure 3.1.6.1), I can ensure a well-structured and efficient design for UniRide.

3.1.7 UI Design:

The initial prototype of UniRide was developed to validate the core functionalities and gather user feedback. Using Android Studio and Jetpack Compose, I focused on creating a responsive and intuitive interface. The prototype included essential features such as real-time bus tracking, route and schedule information, driver review submission, and alert notification system.

The design process involved several key steps:

1. **User Interface (UI) Development:** I started by designing the main screens, including the home screen with real-time tracking, the route details screen, the schedule screen, and the driver review screen. Emphasis was placed on making the UI intuitive and easy to navigate.
2. **Backend Integration:** Using Firebase, I implemented real-time data synchronization, ensuring that bus locations, routes, and schedules were always up-to-date. User authentication was also integrated to provide a personalized experience.
3. **AI Review System:** The Gemini API was incorporated to process and analyze driver reviews. This feature was tested with sample data to ensure accurate summarization and feedback generation.
4. **Real-Time Tracking:** Google Maps API was used to display the real-time locations of buses. This involved setting up location markers and updating them in real-time as the buses moved.

Ensuring a smooth user experience and a scalable architecture were the main design goals of the prototype. To achieve this, several key design principles were implemented, focusing on user-friendly navigation, responsive design, and efficient performance. Each screen was meticulously designed to provide intuitive access to all features, ensuring that users could easily interact with the app without any confusion.

Below are a few instances illustrated through screenshots that showcase these design elements, highlighting the thought process behind each interface component and how it contributes to the overall user experience and system scalability:

Landing Page: Figure 3.1.7.1 below illustrates the landing page for new users, featuring login options for both drivers and students. The page is designed with simplicity and clarity in mind, allowing users to easily select their role and proceed with the login process. This intuitive interface ensures that users can quickly navigate to the desired section of the app.

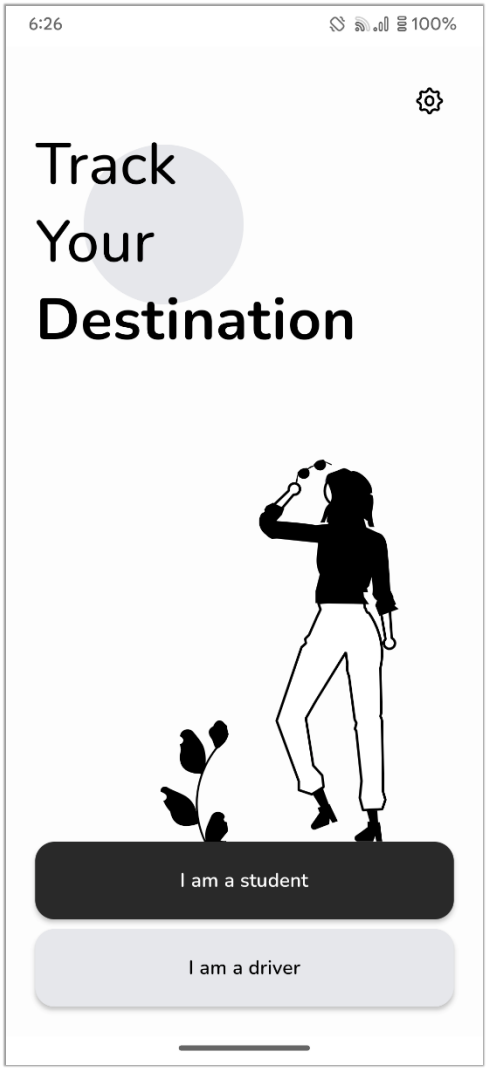

Figure 3.1.7.1: Landing Page

Login Page: Figure 3.1.7.2 below illustrates the login page, which offers two login methods: email or phone number, both exclusively for drivers. Email login requires verified email, and phone number login is protected by two-step OTP verification. Additionally, a password recovery option is available for users who forget their password. If users do not have an account, there is also an option to register.

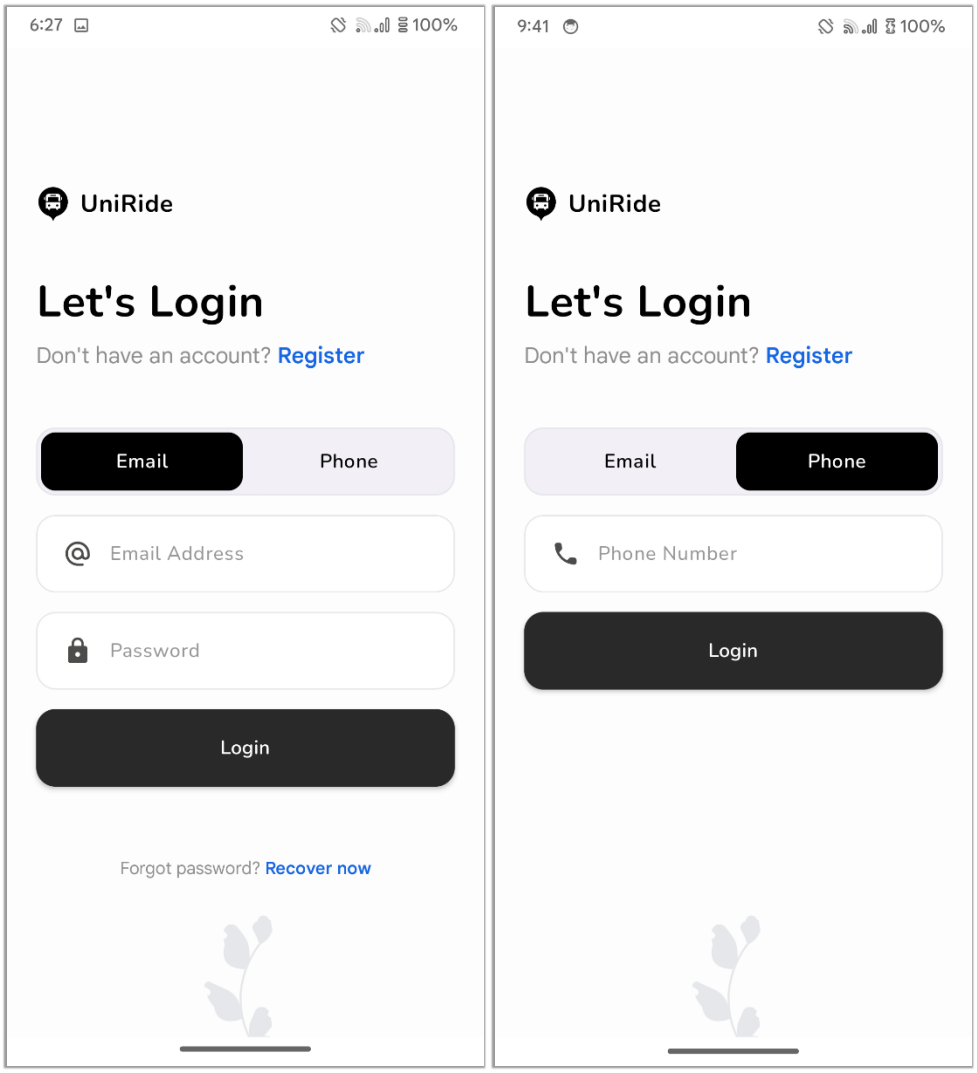

Figure 3.1.7.2: Login Page

Registration Page: Figure 3.1.7.3 below illustrates the registration process, which is exclusively for drivers. Initially, drivers are required to provide general information such as their full name, phone number, email, and password. Following this, they must verify their account by uploading images of their national ID card and driving license. These documents will be submitted to administrators for account verification.

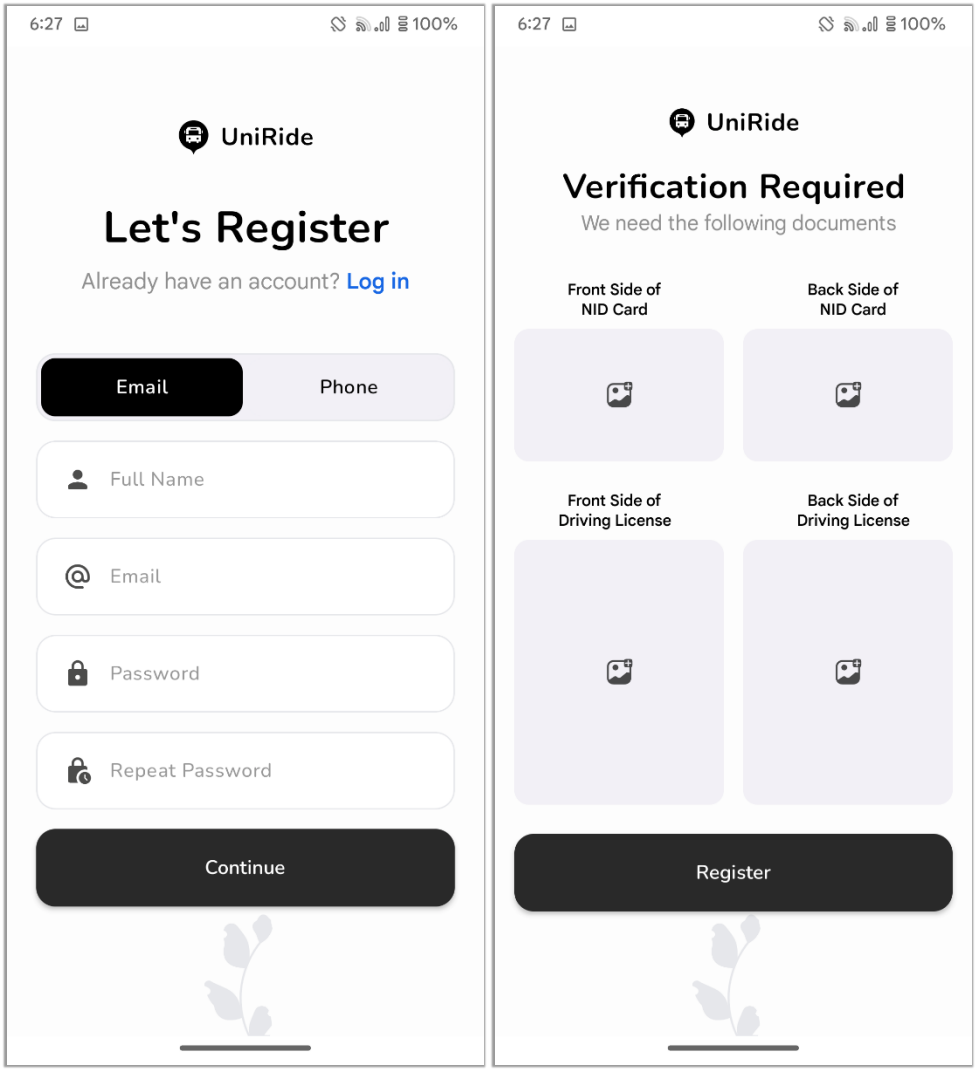

Figure 3.1.7.3: Registration Page

Student Screen:

- **Home Page:** Figure 3.1.7.4 below illustrates the student home page, where students can view announcements from administrators. This page also provides information about general resources available within the transport system, including the total number of vehicles, drivers and helpers, routes, and technicians. Below this, students can find a list of nearby buses currently available.

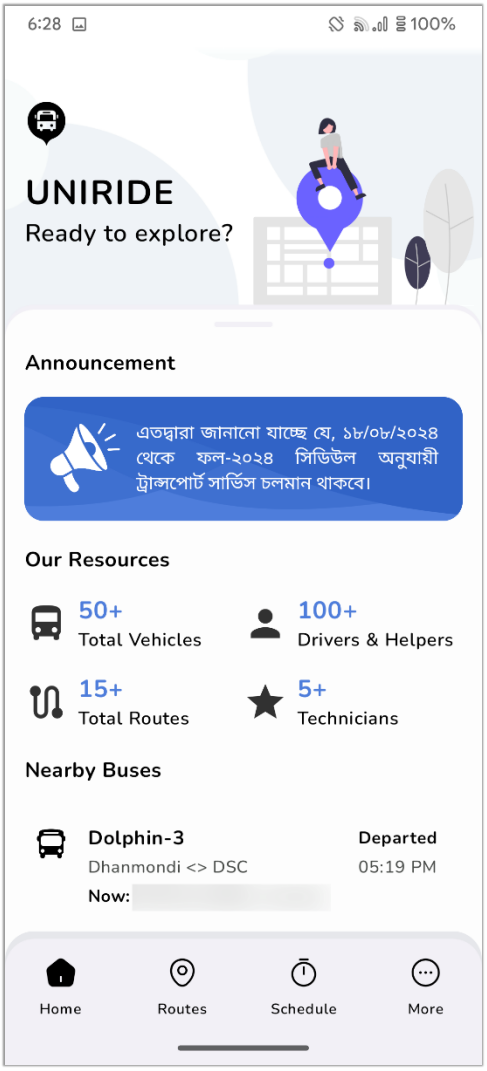

Figure 3.1.7.4: Home Page for Student

When a user clicks on one of the nearby bus items, a new page opens displaying the bus's real-time location alongside the user's own position. They will also receive the shortest route to reach the bus from their current location. An info icon in the top corner allows users to access comprehensive details about the bus and its driver, including a summarized review generated by AI. This feature saves users time by presenting key insights, as many people prefer not to read extensive reviews. Additionally, there is a button to submit a review for the driver. Figure 3.1.7.5 below illustrates the nearby bus's location along with general information.

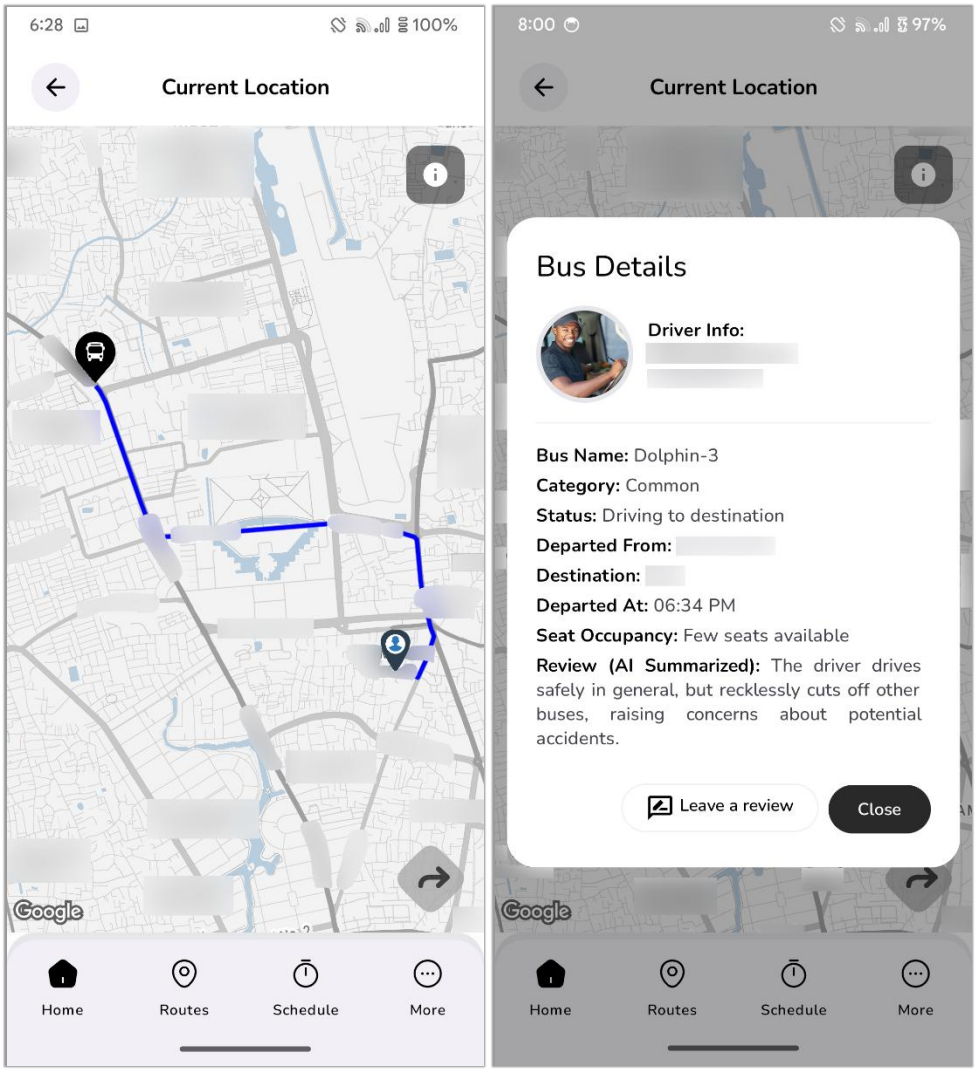

Figure 3.1.7.5: Nearby Bus Details

- **Route Page:** Figure 3.1.7.6 below illustrates the routes page, which displays a list of available routes. Each route is clickable and indicates its type, such as regular, shuttle service, or Friday service, along with the starting and ending place names. At the top of the page, there is an option to filter the routes by type, allowing users to select multiple types to refine their search based on the chosen criteria.

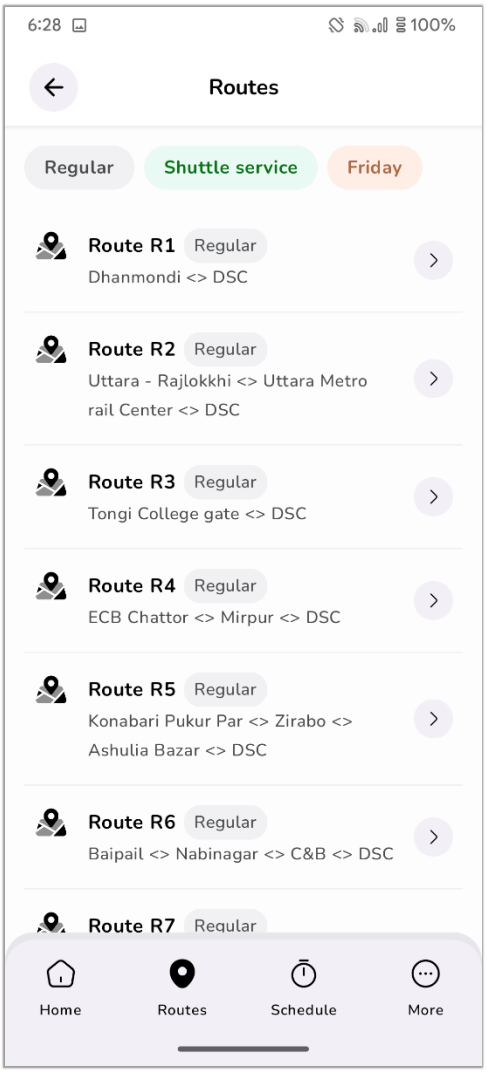

Figure 3.1.7.6: Routes Page

Figure 3.1.7.7 illustrates the route details page. When a user clicks on an item from the route list, this page opens, displaying a map of the route along with all the locations it passes through. Below the map, a details section provides information such as the start time to campus, the departure time from campus, and a list of places the bus will visit along the route.

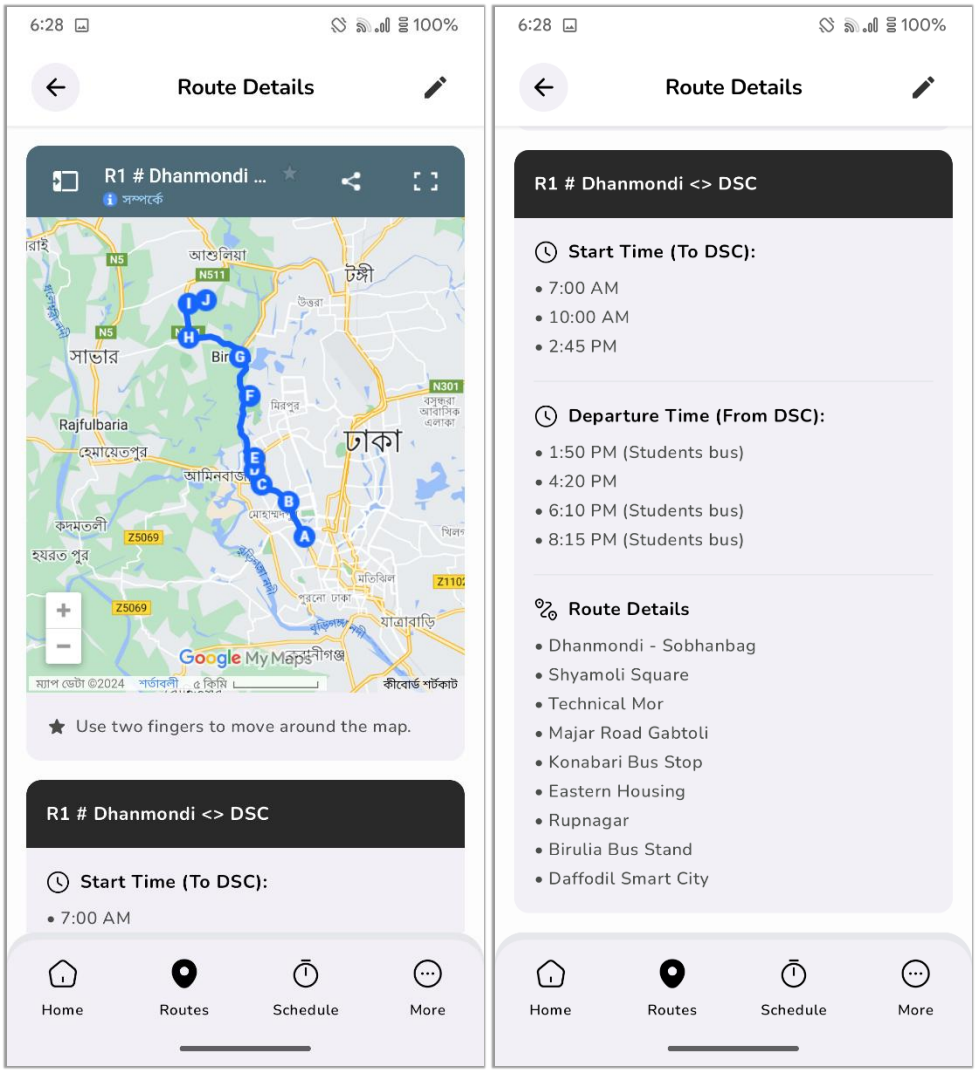

Figure 3.1.7.7: Route Details

- **Schedule Page:** Figure 3.1.7.8 illustrates the schedule page, where users can view a list of bus schedules. Each entry includes the bus name, commuter type (such as employee, common, or Friday service), the starting location, destination, and the corresponding time. Additionally, a search icon is located in the top corner, allowing users to search for specific schedules based on their criteria.

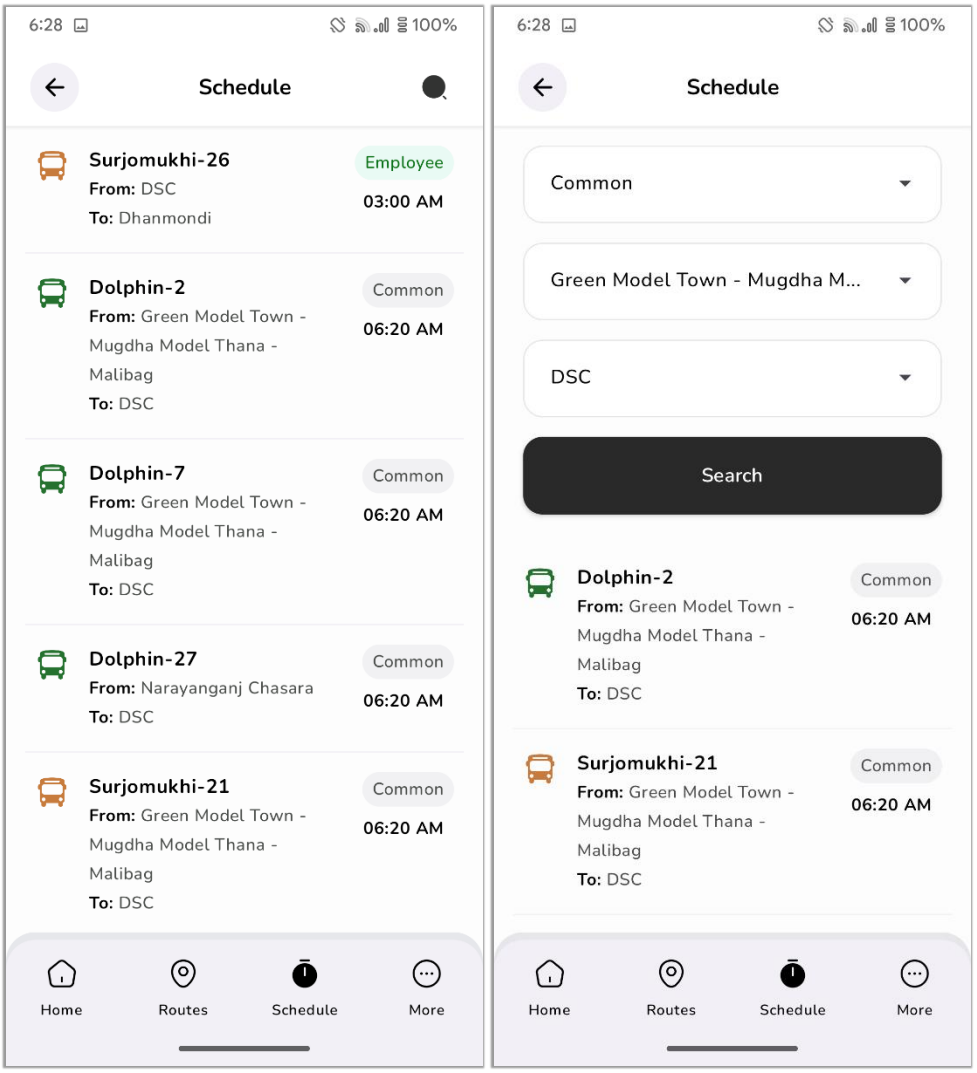

Figure 3.1.7.8: Schedule Page

- **More Page:** Figure 3.1.7.9 below illustrates the additional features page, where users can view their general profile information, including their profile image, username, and email. Below this information, there is a list of features and options available for students to utilize.

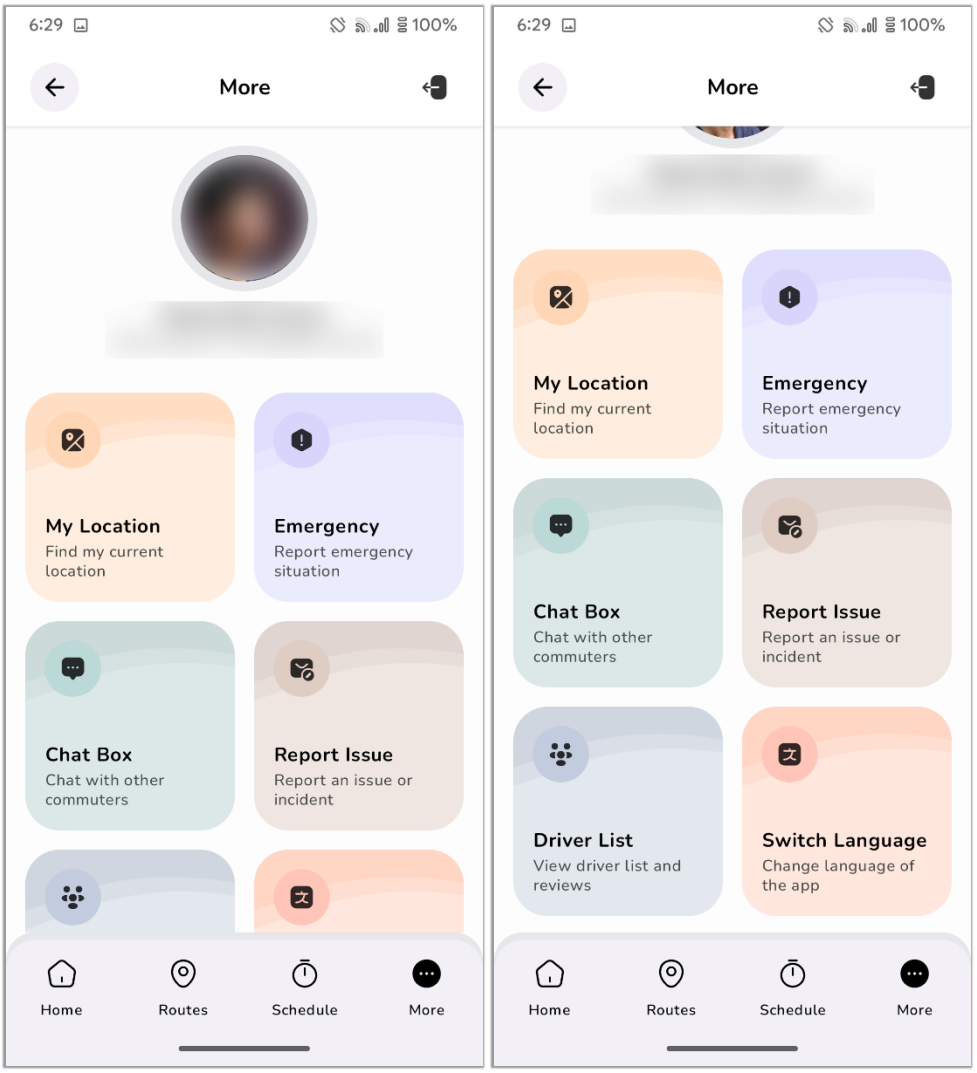

Figure 3.1.7.9: More Page

- **My Location Page:** Figure 3.1.7.10 below illustrates the My Location page, where users can view their current location, which is especially useful if they are lost or in an unfamiliar area. This location is determined by GPS and displayed in real-time, with automatic rotation based on the device's compass sensor.

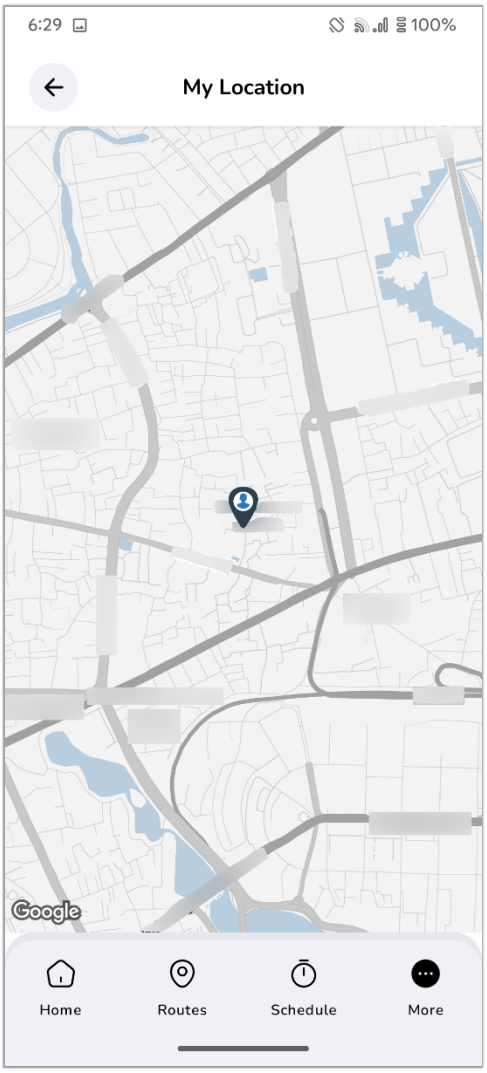

Figure 3.1.7.10: My Location Page

- **Emergency Page:** Figure 3.1.7.11 below illustrates the Emergency page, featuring a prominent quote at the top stating that the campus helpline is always ready to assist in any emergency situation with their experts. Below this quote, there are sliding buttons that users can swipe from left to right to instantly call the corresponding helpline service.

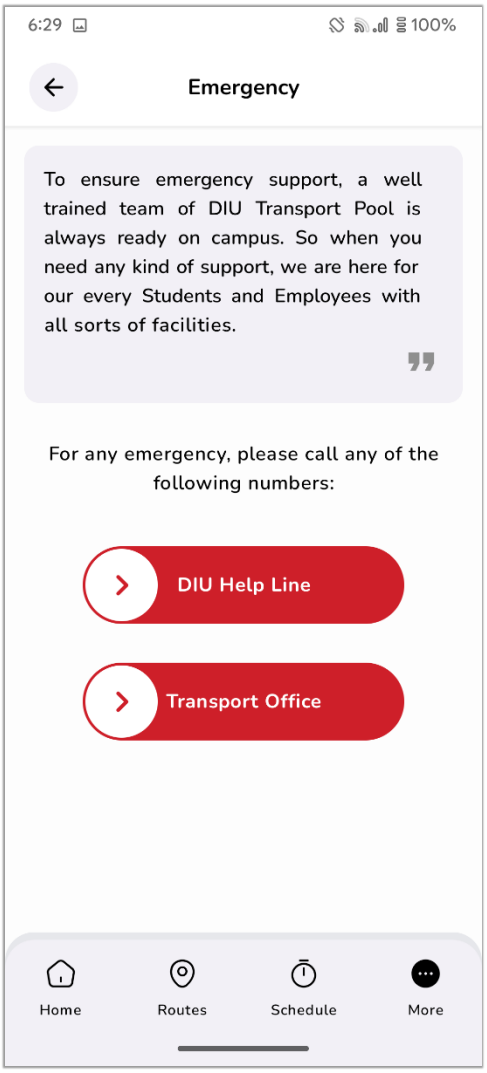

Figure 3.1.7.11: Emergency Page

- **Chat Box Page:** Figure 3.1.7.12 below illustrates the Chat Box page, which serves as a community platform for students. Here, they can chat with each other, share information about transportation, and discuss any problems they encounter on the road.

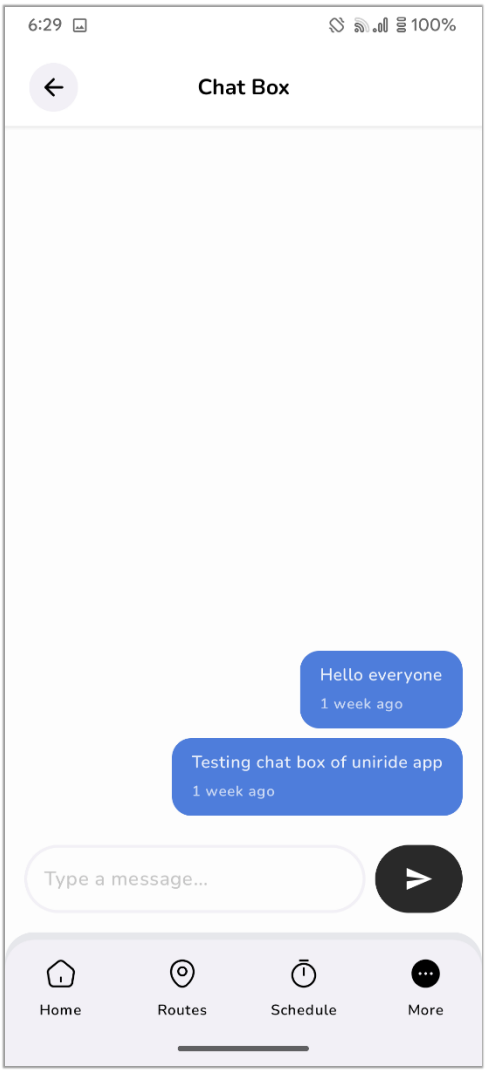

Figure 3.1.7.12: Chat Box Page

- **Report Issue Page:** Figure 3.1.7.13 below illustrates the Report Issue page. Here, users can report issues related to bus schedules, route details, live location tracking, driver information, or other concerns. They are required to provide a description of the issue and their contact information to ensure they are routed to the appropriate support team channel.

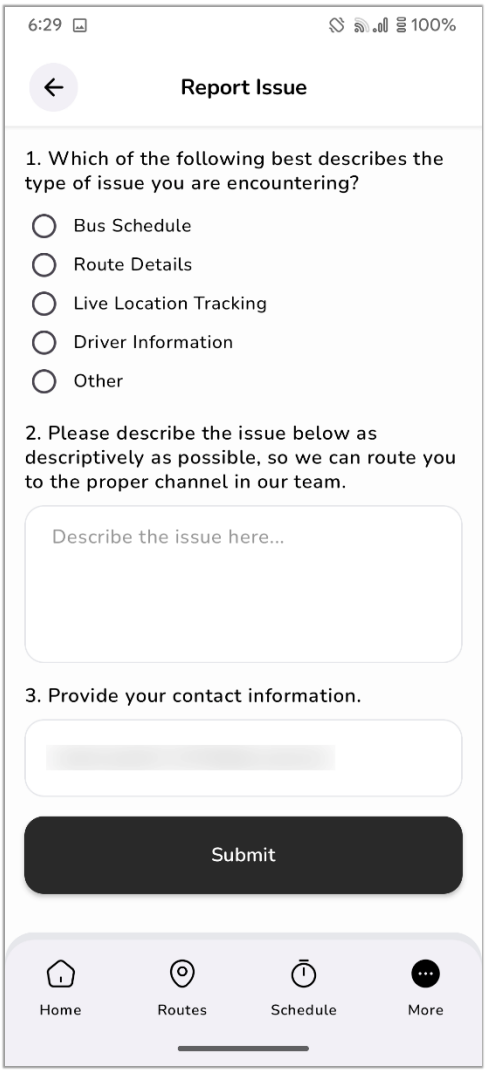

Figure 3.1.7.13: Report Issue Page

- **Driver List Page:** Figure 3.1.7.14 below illustrates the Driver List page. This page displays a list of administrator-approved drivers, including their image, name, contact information, and overall rating. Clicking on a driver will open a new page showing all the reviews of that driver, sorted by most recent first.

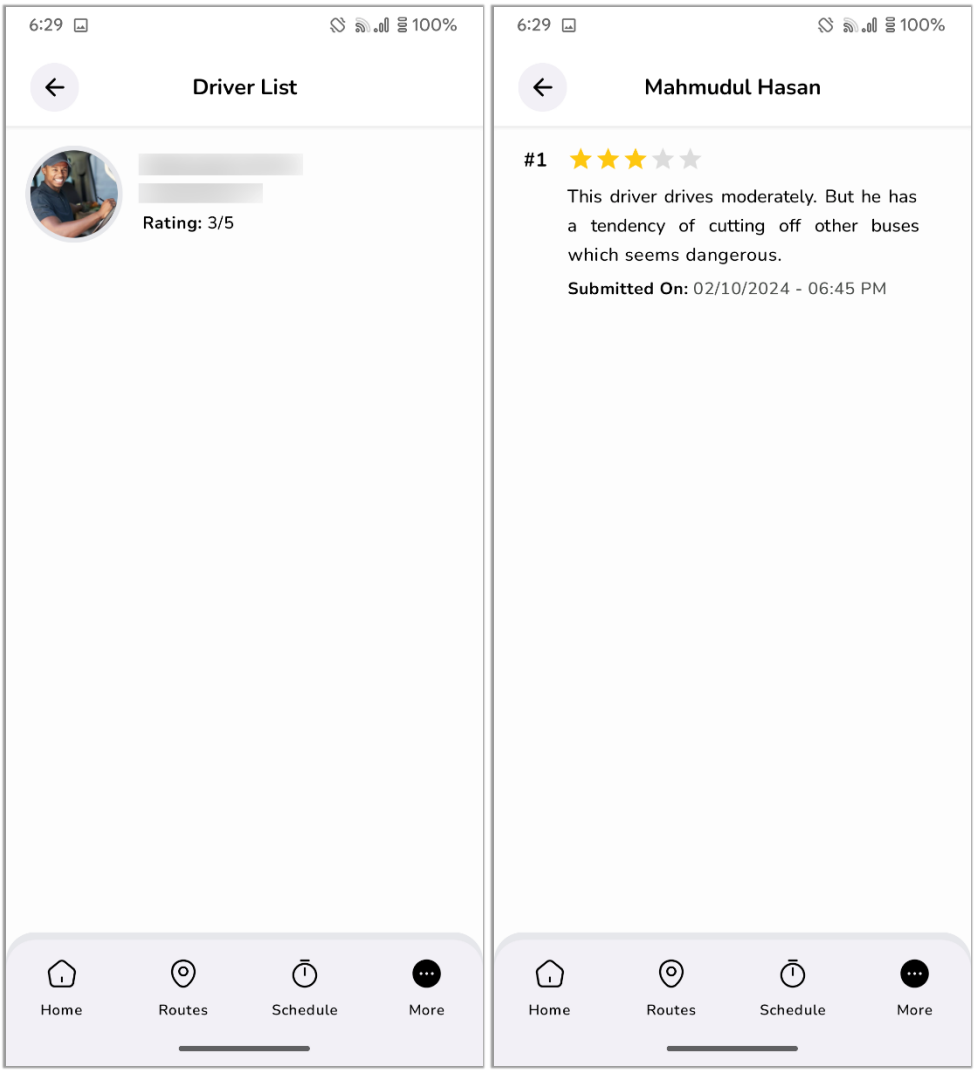

Figure 3.1.7.14: Driver List Page

- **Language Switcher:** Figure 3.1.7.15 below illustrates the language switcher option. Clicking on this option will immediately switch the interface from English to Bengali, or vice versa, for more convenient usage. This feature ensures that users can navigate the app in their preferred language, enhancing accessibility and user experience. The language switcher is prominently located for easy access, allowing for a seamless transition between languages without interrupting the user's workflow.

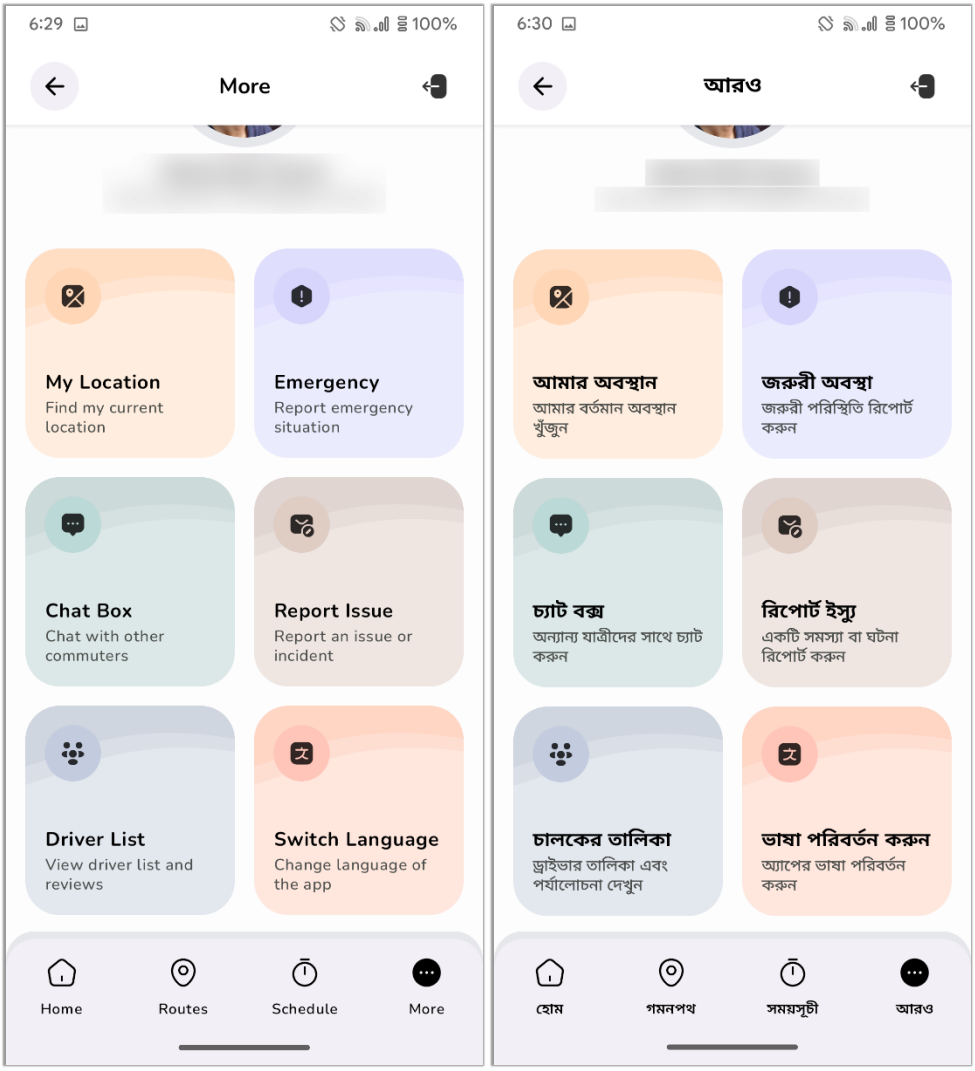

Figure 3.1.7.15: Language Switcher option

Driver Screen:

- **Home Page:** Figure 3.1.7.16 below illustrates the home page for drivers. At the top, there will be a warning message, if applicable. Below that, drivers can start driving to a new destination by providing the necessary information. The driving section will be disabled if there are any warnings, such as account rejection, pending status, or incomplete profile.

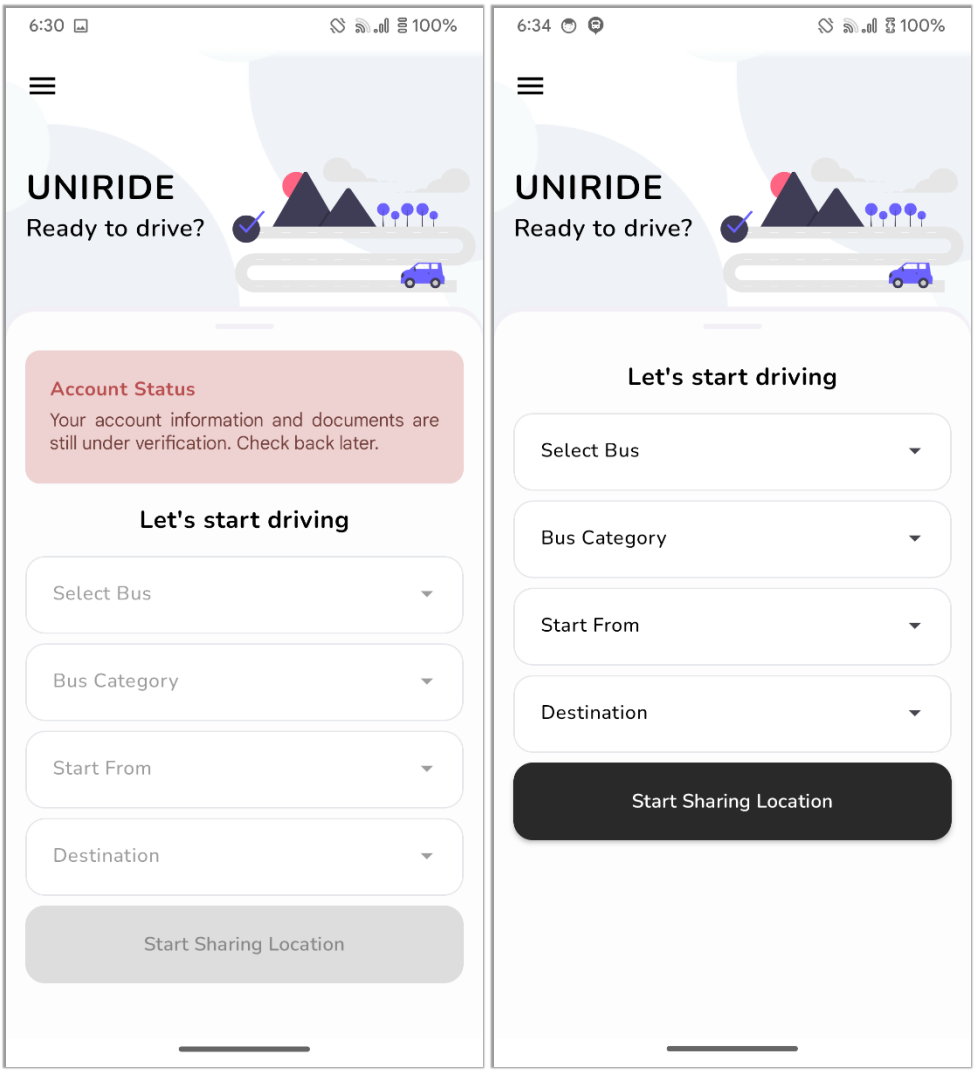

Figure 3.1.7.16: Home Page for Driver

- **Navigation Drawer:** Figure 3.1.7.17 below illustrates the navigation drawer. The header displays general user information, including the profile image, username, and email address or phone number. Below the header, there is a list of available pages and options, such as Edit Profile, Driver Reviews, Language Switcher, Help & Support, Privacy Policy, and Road Transport Act.

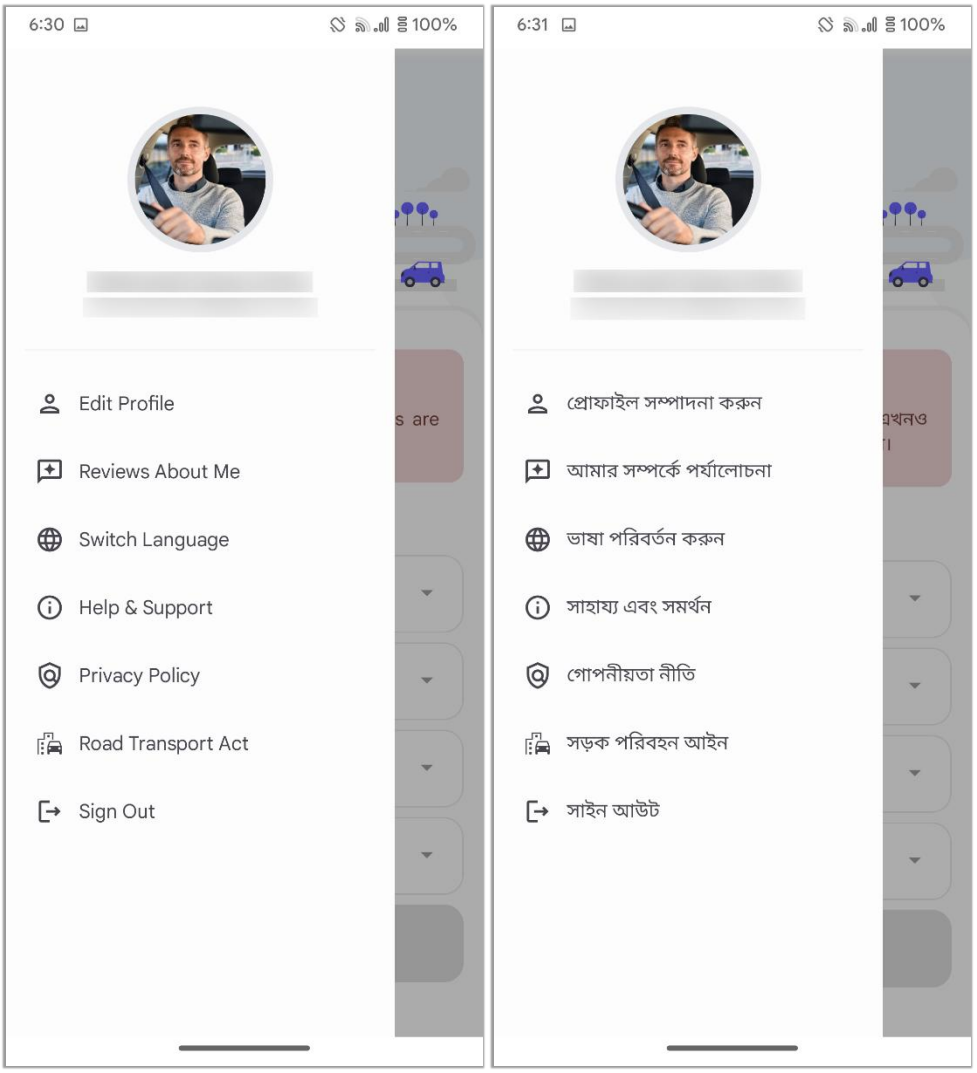

Figure 3.1.7.17: Navigation Drawer

- **Edit Profile Page:** Figure 3.1.7.18 below illustrates the Edit Profile page. Here, drivers can change their profile picture and update their username, phone number, and email. These details are visible to both students and administrators. Leaving any contact information blank will trigger a warning on the homepage and disable the ability to start a new drive.

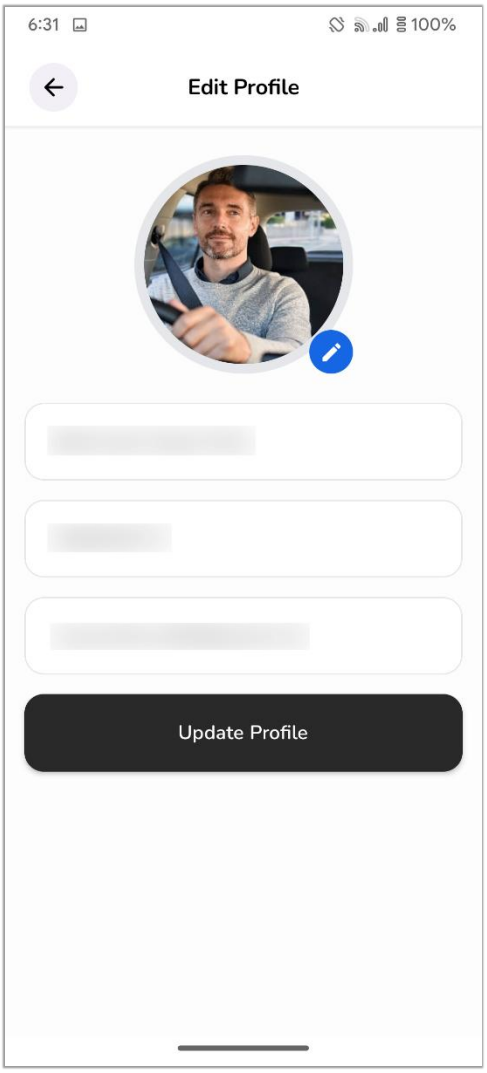

Figure 3.1.7.18: Edit Profile Page

- **Reviews About Me Page:** Figure 3.1.7.19 below illustrates the Reviews About Me page. Here, drivers can view their own reviews submitted by students, sorted by most recent first. This feature allows drivers to gain insights into their performance and areas for improvement. To prioritize user privacy, the reviewer information is kept hidden, ensuring that feedback remains anonymous. This page encourages constructive feedback and fosters a culture of accountability among drivers.

Figure 3.1.7.19: Reviews About Me Page

- **Location Sharing Page:** Figure 3.1.7.20 below illustrates the Location Sharing page. When a driver begins their journey by providing the appropriate information in the app, this page opens to start collecting the driver's location and broadcasting it to commuters in real time. On this page, the driver can update the bus status (e.g., waiting for students, driving, or reached destination) as well as the occupancy status of the bus (e.g., few empty seats left or bus is full). Additionally, the interface is designed for easy navigation, allowing drivers to quickly make updates as needed during their route.

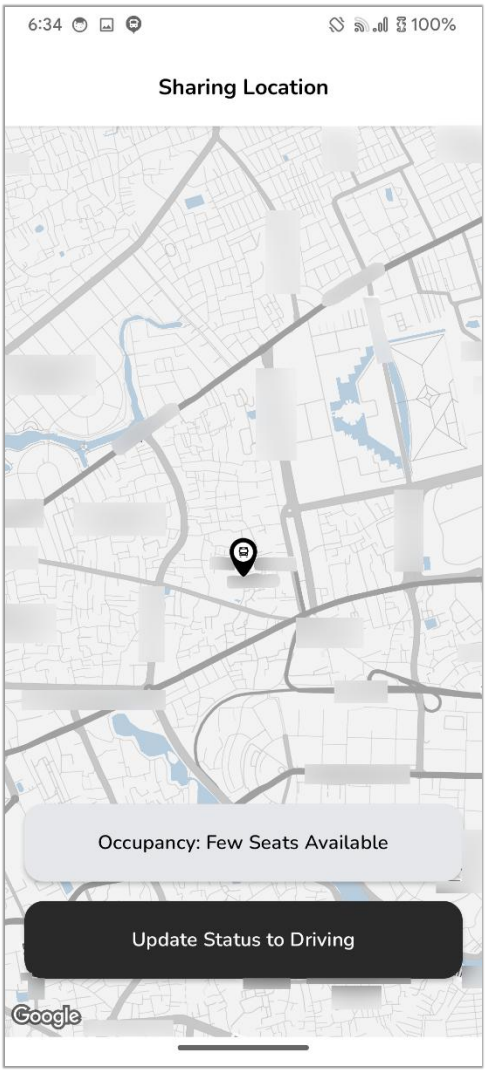

Figure 3.1.7.20: Location Sharing Page

Administrator Screen:

- **Admin Panel Page:** Figure 3.1.7.21 below illustrates the Admin Panel page. Here, the admin can post new announcements, add new bus names, add new locations, create new bus categories, establish new route categories, add new bus routes, add new bus schedules, view the pending driver list, and review reported issues.

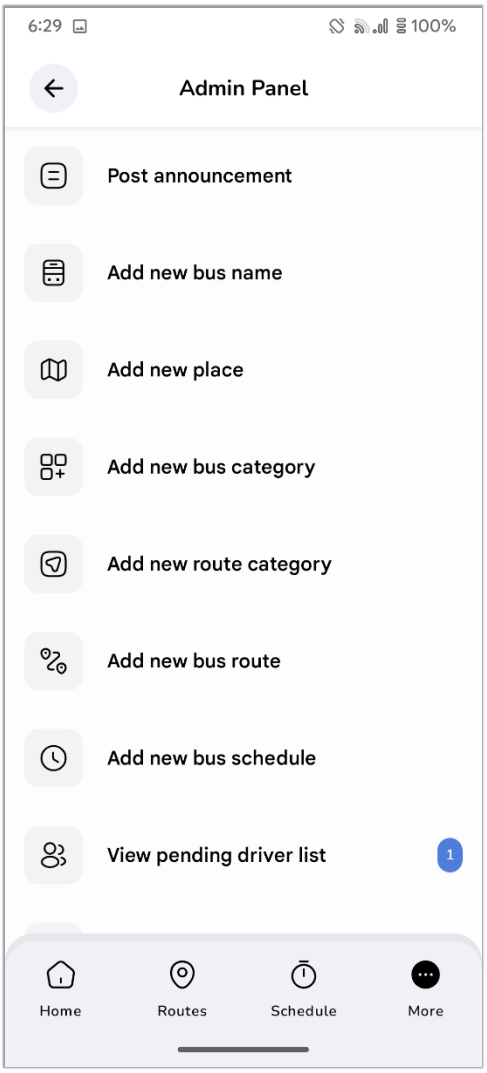

Figure 3.1.7.21: Admin Panel Page

- **Pending Driver List Page:** Figure 3.1.7.22 below illustrates the Pending Driver List page. Here, a list of drivers is displayed, with each item color-coded based on the driver's account status: orange for pending accounts, green for approved accounts, and red for rejected accounts. There is also a search icon available for searching any driver information. The list is organized to show new accounts at the top, while approved or rejected accounts are displayed at the bottom, ensuring that only pending accounts are prioritized in the view.

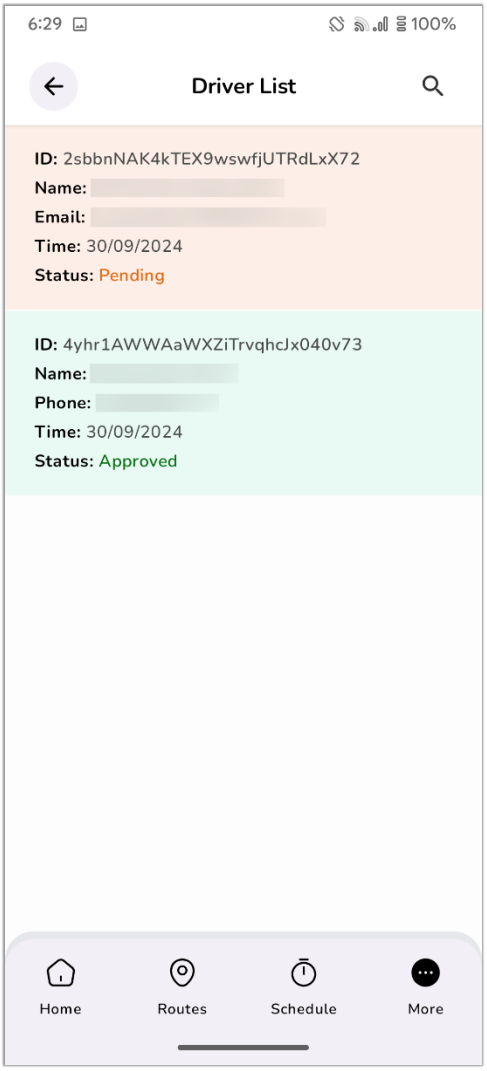

Figure 3.1.7.22: Pending Driver List Page

Figure 3.1.7.23 below illustrates the pending driver details page. Here, admins can view all the information and documents submitted by the driver during the registration process. Admins also have the option to approve or reject the account from this page, with changes being reflected in real time on the driver's page.

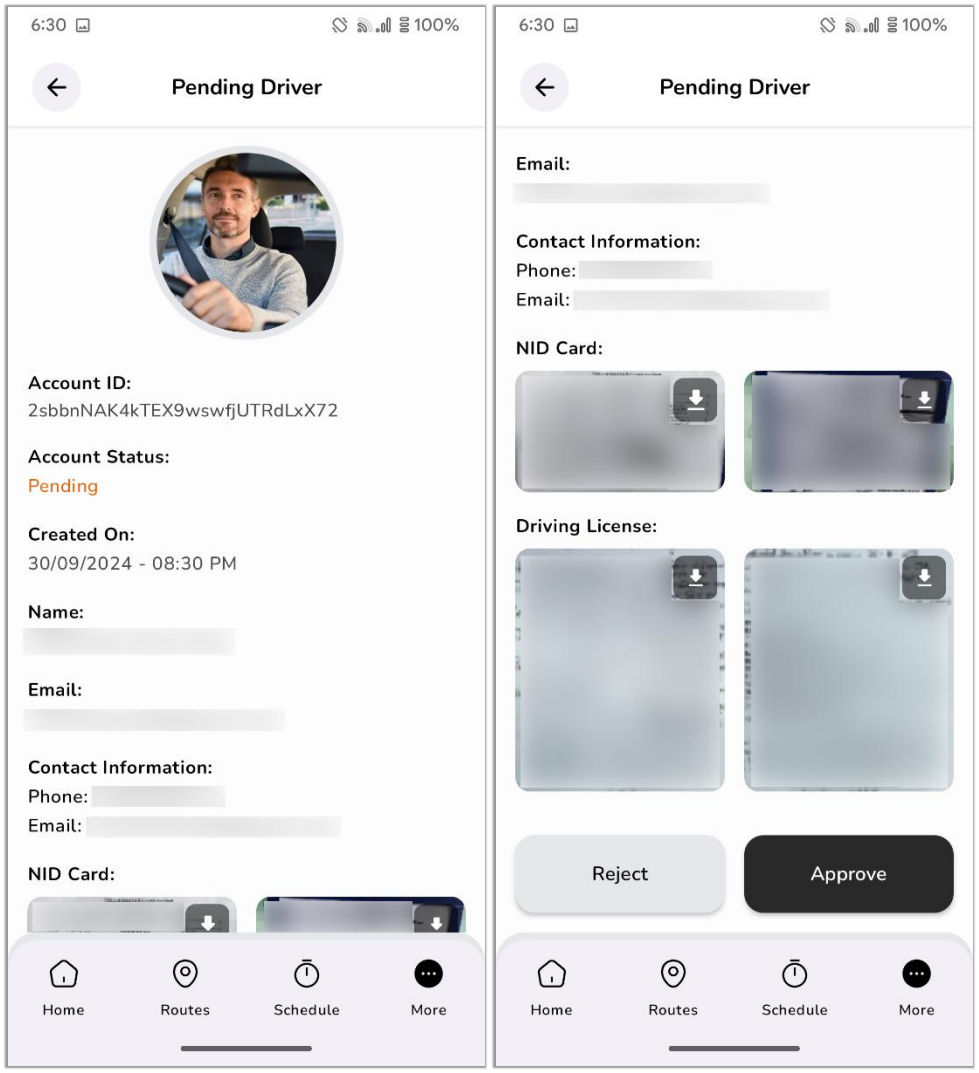

Figure 3.1.7.23: Pending Driver Details Page

- **Reported Issues Page:** Figure 3.1.7.24 below illustrates the Reported Issues page. Here, admins can view all the issues reported by students. Unresolved issues are highlighted in orange, while resolved issues are shown in green. Clicking on an issue opens its details, where admins can mark it as resolved or unresolved, as well as delete the issue using the delete icon.

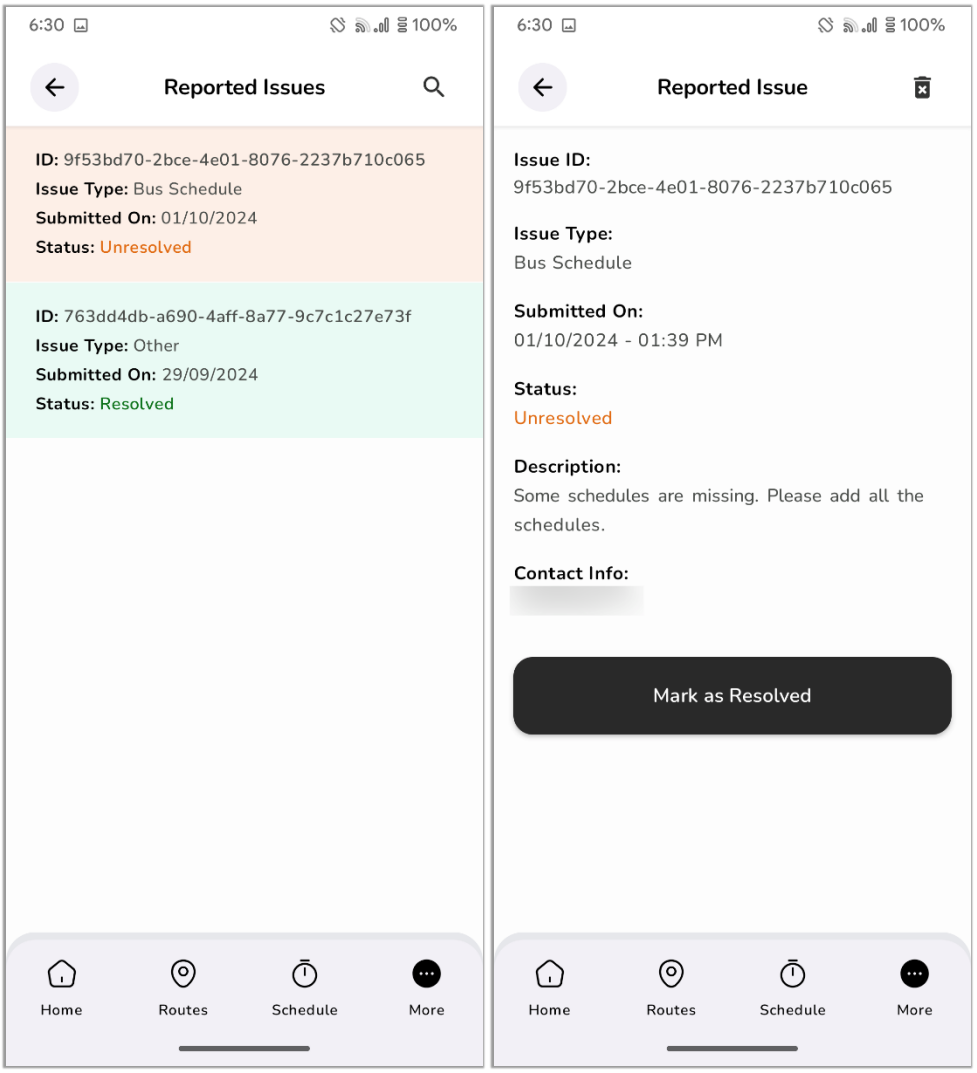

Figure 3.1.7.24: Reported Issues Page

3.2 Detailed Methodology and Design

In designing the UniRide system, the goal was to create a user-friendly and efficient solution for real-time university transport tracking, driver review, and communication. The design process followed several steps:

- **System Architecture:** The application leverages modern mobile technologies. The frontend is built using Jetpack Compose for a responsive, user-friendly UI, while the backend is supported by Firebase for real-time data synchronization and user management. Integration with Google Maps API facilitates live vehicle tracking, and Gemini API is used for analyzing and summarizing driver reviews using AI.
- **Core Features:**
  - Real-time bus tracking via GPS, allowing students to track nearby buses in real-time.
  - An AI-driven driver review system that processes feedback to improve driver behavior.
  - Occupancy status for buses to prevent overcrowding.
  - Notification system for emergency alerts and administrative announcements.
- **Alternate Solution:** An alternative solution considered was the use of GPS tracker devices installed directly on buses for real-time tracking. However, maintaining and updating these devices across a fleet of buses involves high upfront costs and regular maintenance. Given these limitations, a phone-based tracking system was chosen, as it reduces costs by utilizing the smartphones already in possession of drivers. Additionally, smartphones provide a versatile platform for integrating other app features, such as driver reviews and notifications.
- **Choice of Method:** The phone app-based tracking solution was selected due to its cost-effectiveness, scalability, and ease of implementation. This solution does not require additional hardware installation or maintenance, as it leverages the existing infrastructure (i.e., smartphones with GPS capabilities).

3.3 Project Plan

The project was broken down into clear phases to ensure systematic progress and timely completion:

1. **Phase 1: Requirement Gathering and Initial Design** (Duration: 3 weeks)
    - Conducted surveys and focus group discussions with university transport stakeholders.
    - Identified key features: real-time tracking, AI reviews, emergency notifications, etc.
2. **Phase 2: Development and Prototype Testing** (Duration: 6 weeks)
    - Developed initial app prototype with core functionalities.
    - Conducted internal testing for UI/UX and functional validation.
3. **Phase 3: User Feedback and Refinement** (Duration: 4 weeks)
    - Rolled out the app to a small user group to gather feedback on usability and performance.
    - Implemented improvements based on feedback, such as UI adjustments and bug fixes.
4. **Phase 4: Final Testing and Pre-Deployment Preparation** (Duration: 3 weeks)
    - Extensive testing to ensure stability, scalability, and data accuracy.
    - Prepared deployment documentation, with plans to deploy on the Play Store or any other app store after further optimization.

This structured approach ensured a focused development process, resulting in a robust and user-centric application. With comprehensive testing and refinement, the project is now poised for deployment, ready to enhance the university transport experience for its users.

Gantt Chart:

The following Gantt chart 3.3.1 shows the project schedule that has been maintained:

Table 3.3.1: Project Schedule Gantt Chart

Phases	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13
Phase 1													
Phase 2													
Phase 3													
Phase 4													

3.4 Task Allocation

The project was divided into distinct tasks to ensure efficient work distribution:

- **UI/UX Design:** I was responsible for designing the intuitive interface, including wireframes and prototypes, ensuring ease of use for both students and drivers. I focused on creating a clean, responsive design using Jetpack Compose to provide a smooth user experience across devices.
- **Backend Development:** I handled all aspects of Firebase integration for real-time data synchronization, user authentication, and cloud storage management. This included setting up Firebase Firestore for storing and retrieving data and ensuring seamless updates and synchronization of bus locations and user information.
- **API Integration:** I integrated Google Maps API for real-time tracking of buses and Gemini API for AI-driven driver review analysis. This required customizing API calls to track buses accurately, as well as implementing a feedback system that used AI to analyze and summarize driver reviews, providing actionable insights.
- **Testing and Debugging:** I conducted comprehensive testing to identify bugs, ensure feature functionality, and optimize performance. I tested each feature individually and also performed integration testing to ensure that the entire system

worked seamlessly together. I fixed bugs and improved the user experience based on the results of these tests.

- **Deployment and Documentation:** I managed the app's deployment process, including preparing the final version for launch. Additionally, I created detailed user guides and technical documentation to ensure that users could easily understand and navigate the app, and to provide a foundation for future updates and maintenance.

By organizing the tasks in this way, I ensured the project's timely development and successful delivery. Handling all aspects of the project solo allowed for full control over the development process, enabling me to iterate and refine the app based on user feedback and technical requirements.

3.5 Summary

The UniRide system provides a comprehensive solution for real-time vehicle tracking and AI-enhanced driver reviews, addressing key challenges faced by university transportation systems. By choosing a smartphone-based approach, the project minimized costs and complexity, while offering scalability and flexibility. The methodical design, thorough planning, and task allocation ensured the timely and successful development of the application, paving the way for future improvements and broader implementation.

Chapter 4
Implementation and Results

4.1 Overview

In this chapter, I will present the results obtained from the testing of the UniRide application and analyze its performance and user feedback. I will explore the experimental results, conduct a comparative analysis with existing systems, and summarize the key findings. This comprehensive examination will highlight the effectiveness of UniRide in meeting its objectives and addressing the challenges identified in previous chapters.

4.2 Experimental/Simulation Result

To evaluate the performance of UniRide, I conducted a series of tests focusing on various aspects of the application, including real-time tracking accuracy, user interface responsiveness, and the functionality of the AI-driven driver review system.

- **Real-Time Tracking Performance:** Tests were performed under different conditions, including varying network speeds and GPS signal strengths. The application demonstrated an average accuracy rate of 95% for bus location tracking, significantly improving user confidence in scheduling their travel. The location updates were consistently received in less than 5 seconds, with occasional delays extending to a maximum of 10 seconds during periods of poor network connectivity.
- **User Feedback on Usability:** The UI was tested with a group of users, and feedback indicated that 90% found the interface intuitive and easy to navigate. Load times for the application averaged around 2 seconds, which is within acceptable limits for mobile applications.
- **AI Review System Accuracy:** I analyzed user feedback submissions over a two-week period. The AI system processed approximately 200 reviews, providing

summarized insights that highlighted areas for improvement in driver performance, demonstrating its effectiveness in driving service enhancements.

Test Results and Analysis:

The following table 4.2.1 presents the results of functional testing conducted on the application. Functional testing ensures that the application's features and functionalities operate as intended, according to the specified requirements. Each test case was executed to verify both the success conditions and potential error scenarios. The tests highlight any deviations from the expected behavior, including issues with permissions, notifications, and session management.

Table 4.2.1: Summary of Test Results

SN.	Test	Expected Result	Result
01	Registration	New user information will be stored in Firebase upon registration.	Success
		Display a toast message with the appropriate error notification.	Failed
02	Login	Users can log in using their email and password or phone number.	Success
		Display a toast message with the appropriate error notification.	Failed
03	Forgot password	A registered user will receive a recovery email by clicking "Recover now."	Success
		Session time has expired.	Failed
04	Filter routes	Users can filter routes by type.	Success
05	Search schedules	Users can search for specific schedules.	Success
06	Get location	Retrieve the user's current location with the highest accuracy.	Success

07	Call emergency	Automatically dial an emergency number without user input.	Success
08	Send message in chat box	Send a new message in the chat and display it to other users in real time.	Success
09	Report issue	Report a transport or app-related issue and save it in the database.	Success
10	Switch Language	Toggle all app strings between English and Bengali.	Success
11	Post announcement	The admin will post a new announcement that will be instantly visible to all users.	Success
		The user account lacks administrator permissions.	Failed
12	Add new bus name	The admin will add a new bus name, which will be instantly visible to users.	Success
		The user account lacks administrator permissions.	Failed
13	Add new place name	The admin will add a new place name, which will be instantly visible to users.	Success
		The user account lacks administrator permissions.	Failed
14	Add new bus category	The admin will add a new bus category, which will be instantly visible to users.	Success
		The user account lacks administrator permissions.	Failed
15	Add new route category	The admin will add a new route category, which will be instantly visible to users.	Success

		The user account lacks administrator permissions.	Failed
16	Add new bus route	The admin will add a new bus route, which will be instantly visible to users.	Success
		The user account lacks administrator permissions.	Failed
17	Add new bus schedule	The admin will add a new bus schedule, which will be instantly visible to users.	Success
		The user account lacks administrator permissions.	Failed
18	Mark issues as pending or resolved	The admin will mark issues as pending or resolved in the database.	Success
		The user account lacks administrator permissions.	Failed
19	Approve or reject driver account	The admin will approve or reject a driver's account, and the changes will be instantly reflected on the driver's screen.	Success
		The user account lacks administrator permissions.	Failed

Performance Testing:

During UniRide's performance testing, I used Android Profiler to track CPU, memory, and network usage in real-time. This allowed me to identify performance bottlenecks, such as high CPU usage or excessive memory consumption, which could impact the app's efficiency. The data, presented in Figure 4.2.2, helped pinpoint areas needing optimization. Based on the insights, I made adjustments to ensure the app runs smoothly and performs efficiently across various devices and usage scenarios.

Figure 4.2.2: Performance monitoring using Android Profiler

During the testing phase, the UniRide application demonstrated exceptional performance across several key metrics:

- **CPU Usage:** Throughout the testing phase, CPU usage remained stable, showing no significant spikes even during peak activities such as real-time tracking and route updates. This stability contributed to a smooth and responsive user experience, crucial for maintaining user engagement during critical operations.
- **Frame Rendering:** Instances of janky frames were infrequent, with rendering times consistently falling within the acceptable range. The application maintained a steady rate of 60 frames per second, particularly when rendering moving buses on the map, ensuring that users experienced fluid animations and interactions without interruptions.
- **Thread Utilization:** The app demonstrated effective thread management for background operations, successfully preventing any interference with the main UI thread. This design choice allowed for responsive user interactions, even while background tasks, including data fetching and AI processing, were executed seamlessly in the background.

- **Memory Usage:** Efficient memory management practices were observed, with the app operating well within the allocated limits. There were no signs of memory leaks or excessive garbage collection, which is vital for maintaining stable performance, especially on devices with limited resources.
- **Network Performance:** The app exhibited robust network performance, with minimal latency in fetching real-time data. This capability allowed for timely updates without noticeable delays, crucial for applications relying on immediate data for effective user interaction.
- **Network Performance:** The app exhibited robust network performance, with minimal latency in fetching real-time data. This capability allowed for timely updates without noticeable delays, crucial for applications relying on immediate data for effective user interaction.
- **Battery Consumption:** The app displayed optimized battery usage, with no significant drain observed even during extended periods of use. Efficient background task management and resource optimization contributed to a balanced power consumption profile.

In addition, the application was tested under various network conditions, including low and fluctuating bandwidth, where it consistently adjusted its data requests to ensure optimal performance without compromising user experience. The seamless integration of these performance enhancements ensures that UniRide can deliver a consistent and high-quality experience to users in real-world usage scenarios.

Overall, performance testing confirmed that UniRide operates effectively under various conditions, characterized by stable CPU usage, minimal frame delays, proper thread utilization, and efficient memory and network management. This ensures that the app can handle real-time tracking and user interactions smoothly, paving the way for a successful deployment.

4.3 Comparative Analysis

To further assess UniRide's effectiveness, I conducted a comparative analysis with similar applications in the market. The following table 5.3.1 summarizes the key features and performance metrics of UniRide against those of existing systems:

Table 4.3.1: Comparative Analysis

Feature	**Existing University Transportation Apps**	**UniRide**
Real-Time Tracking	Yes	Yes
AI-Driven Driver Review	No	Yes
User-Friendly Interface	Moderate	High
Comprehensive Route Information	Sometimes Limited	Yes
Regular Schedule Updates	Sometimes Limited	Yes
Occupancy Status	No	Yes
Notification System	Limited	Yes
Issue Reporting System	No	Yes
User Feedback	Average or below average	Good

4.4 Summary

In summary, the results from my experimental testing and comparative analysis highlight UniRide's effectiveness in providing a comprehensive solution for university transportation needs. With high accuracy in real-time tracking, an intuitive user interface, and advanced features like AI-driven driver reviews, UniRide significantly improves the commuting experience for students and staff. These findings validate the objectives set forth at the project's outset and demonstrate the potential for further enhancements and expansion in future iterations of the application.

Chapter 5

Engineering Standards and Design Challenges

5.1 Compliance with the Standards

Compliance with industry standards is crucial in ensuring that the UniRide application meets quality, performance, and security benchmarks. This section outlines the software, hardware, and communication standards adhered to during the development of the project.

5.1.1 Software Standards

The development of the UniRide application followed several established software standards to ensure robust and maintainable code:

- **ISO/IEC 25010:2011** (Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models): This standard was followed to ensure the application met high standards of functionality, reliability, usability, efficiency, maintainability, and portability.
- **Agile Methodology:** Agile practices were employed to ensure iterative development, allowing for regular updates and improvements based on user feedback.
- **Google's Material Design Guidelines:** The UI design adhered to these guidelines to ensure a consistent and intuitive user experience.
- **OWASP Mobile Security Guidelines:** These guidelines were followed to ensure the application is secure against common vulnerabilities and threats.

5.1.2 Hardware Standards

The hardware standards ensured that the UniRide application performed efficiently across various devices:

- **Android Compatibility Definition Document (CDD):** The app was developed to be compatible with the specifications outlined in the Android CDD, ensuring it runs smoothly on a wide range of Android devices.
- **Minimum Device Requirements:** The application was tested on devices meeting the following minimum standards:
  - **i)** Android OS version 8.0 (Oreo) or higher
  - **ii)** Minimum 2GB RAM
  - **iii)** Minimum 1.4 GHz processor
  - **iv)** GPS functionality for real-time tracking
  - **v)** Reliable internet connection (Wi-Fi or mobile data)

5.1.3 Communication Standards

Adhering to communication standards was vital for ensuring seamless data exchange and user interaction:

- **HTTP/HTTPS Protocols:** Secure communication between the client app and the Firebase backend was ensured using HTTPS, protecting data in transit.
- **Google Maps API Standards:** Compliance with Google Maps API terms of service and usage policies was maintained, ensuring proper use of map services and location data.
- **Push Notification Standards:** Firebase Cloud Messaging (FCM) was used to implement push notifications, adhering to FCM standards for reliable and efficient message delivery.
- **RESTful API Standards:** The integration of external APIs (such as Google Maps and Gemini) followed RESTful principles, ensuring scalable and maintainable interactions.

By adhering to these software, hardware, and communication standards, the UniRide application was developed to be reliable, secure, and user-friendly, meeting the high expectations of its users and stakeholders.

5.2 Impact on Society, Environment and Sustainability

The UniRide application significantly enhances campus transportation efficiency, promotes environmentally sustainable commuting practices, and fosters a safer, more connected university community.

5.2.1 Impact on Life

UniRide has the potential to significantly enhance the daily lives of students and staff within the university community by addressing various aspects:

- **Convenience:**
    - Users can access real-time bus tracking, allowing for more efficient planning of their commutes.
    - The application reduces uncertainty about bus arrivals, minimizing waiting times.
- **Safety:**
    - With features like occupancy status and alert notifications, users can make informed choices about their travel and stay updated in critical situations.
    - The AI-driven review system encourages better driver performance, contributing to a safer travel experience.
- **Increased Efficiency and Transparency:**
    - Users gain visibility into bus operations, fostering trust in the transportation system.
    - Efficient scheduling and tracking lead to better resource utilization.
- **Economic Benefits:**
    - Reduced reliance on personal vehicles can lead to lower transportation costs for users.
    - Increased use of public transportation can contribute to local economies through enhanced accessibility.

- **Community Building:**
    - The app fosters a sense of community by enabling users to provide feedback on drivers, promoting accountability and improvement in service quality.
    - Facilitates social interactions among users, helping to create a connected campus environment.
- **Social Considerations:**
    - Enhances accessibility for individuals who may not have reliable personal transportation options.
    - Promotes equity in transportation access for all students and staff.

5.2.2 Impact on Society & Environment

UniRide contributes positively to environmental sustainability by promoting more efficient transportation practices, which presents both potential benefits and drawbacks:

- **Potential Benefits:**
    - **Reduced Carbon Footprint:** Encouraging the use of public transportation over personal vehicles helps decrease overall carbon emissions.
    - **Optimized Routes:** Real-time tracking allows for better route management, reducing fuel consumption and minimizing traffic congestion.
    - **Awareness of Environmental Issues:** The app can be used to inform users about eco-friendly practices, such as carpooling or taking buses during peak hours.
    - **Decrease in Air Pollution:** With fewer cars on the road, air quality improves, leading to healthier communities.
    - **Reduced Traffic Congestion:** Efficient use of public transportation networks decreases the number of vehicles on the road, leading to less traffic and quicker travel times.

- **Promotion of Public Transit Use:** By making public transportation more efficient and appealing, UniRide encourages a shift from private car use to mass transit options.

- **Potential Drawbacks:**
  - **Increased Energy Consumption:** The operation of mobile devices and servers may lead to higher energy use, although this can be mitigated through efficient practices.
  - **Dependence on Technology:** Overreliance on the app could discourage walking or biking for short distances, which are environmentally friendly alternatives.
- **Overall Consideration:** While UniRide promotes sustainable transportation, continuous monitoring of its environmental impact is essential to ensure that it aligns with broader sustainability goals.

5.2.3 Ethical Aspects

As with any technology, ethical considerations are crucial in the development and implementation of UniRide:

- **Promoting Fair Housing Practices:** Ensure that the app does not inadvertently contribute to housing inequality or displacement in the community.
- **Balancing Transparency and User Privacy:** Clearly communicate data collection practices while maintaining user trust through transparent policies.
- **Addressing Algorithmic Bias:** Ensure that the AI-driven review system operates fairly, avoiding biases that could negatively impact drivers or users.
- **Responsible Marketing Practices:** Promote the app accurately without overpromising its capabilities, ensuring users have realistic expectations.
- **Maintaining a Balanced Platform:** Strive for a fair balance between user feedback and driver treatment to foster a healthy ecosystem for all stakeholders.

5.2.4 Sustainability Plan

To ensure the long-term success and sustainability of UniRide, several strategies will be implemented:

- **Regular Updates and Maintenance:** Ongoing support will be provided to address technical issues and adapt to user needs over time.
- **User Feedback Integration:** Continuous collection and analysis of user feedback will inform future updates, ensuring the app remains relevant and effective.
- **Community Collaboration:** Engaging with the university community, including students, staff, and administration, will be crucial for gathering insights and fostering a sense of ownership in the app.
- **Reducing Environmental Impact of Move-Ins:** Implement initiatives to coordinate transportation for move-in days, reducing traffic congestion and emissions.
- **Sustainable Building Practices:** Collaborate with the university to integrate sustainability practices into campus infrastructure and transportation services.
- **Data Analysis and User Behavior:** Continuously analyze user behavior data to identify trends and optimize the app's features for better environmental outcomes.

UniRide has the potential to make a significant impact on the lives of users by providing a convenient, safe, and community-focused transportation solution. Its positive environmental contributions and ethical considerations further enhance its value as a sustainable initiative. By implementing a robust sustainability plan and addressing the outlined ethical aspects, UniRide aims to ensure its long-term success and continued relevance in promoting efficient and responsible transportation practices within the university community.

5.3 Project Management and Financial Analysis

Planning:

- **Define Project Scope:** The scope of UniRide includes essential features such as real-time bus tracking, AI-driven driver reviews, viewing bus routes and schedules, and a communication platform for commuters.
- **Create Development Timeline:** Establish a timeline outlining key milestones such as prototype completion, feature implementation, testing phases, and the final deployment.
- **Identify Resources Needed:** Assess the required resources, including software tools, infrastructure, hardware requirements, and budget allocation for hosting, maintenance, and ongoing support.

Execution:

- **Track Progress:** Regularly monitor progress towards achieving milestones and deadlines to ensure the project stays on track.
- **Manage Communication:** Facilitate effective communication with stakeholders and the university transportation authority to promote collaboration, gather feedback, and quickly address any issues that arise.
- **Address Roadblocks:** Identify and resolve any challenges that arise during development promptly to minimize delays and maintain momentum.

Monitoring and Control:

- **Monitor App Performance:** Continuously evaluate the app's functionality and user feedback to ensure a smooth user experience and identify areas for improvement.
- **Implement Enhancements:** Address any bugs or issues reported by users and roll out updates to enhance app performance and features based on feedback.
- **Budget Tracking:** Regularly review the project budget and make necessary adjustments to resource allocation to stay within financial constraints.

Project Management:

UniRide employs Agile methodology to enable iterative development and accommodate continuous feedback. Key tools and practices include:

- **Scrum Framework:** Utilize sprints, daily stand-ups, sprint planning, and review meetings to manage the workflow effectively.
- **Self-Reflection:** After each sprint, reflect on what worked well and what could be improved in future iterations, enabling continuous learning and enhancement of the development process.
- **Financial Management Tools:** Use budgeting and accounting software to track project expenses and revenue.
- **Resource Allocation:** Optimize the allocation of resources, balancing development time with budget constraints to maintain efficiency.
- **Flexibility:** Stay adaptable to unforeseen challenges and opportunities, adjusting plans as necessary.

Finance:

- **Budgeting:**
  - **Estimate Development Costs:** Include costs for development tools, hosting services, and any necessary software licenses.
  - **Potential Revenue Streams:** Explore avenues for revenue generation, such as premium features for users or potential partnerships.
- **Financial Reporting:**
  - **Budget Tracking:** Regularly report on budget status and resource allocation to ensure transparency.
  - **Analyze Revenue Opportunities:** Create financial reports to evaluate and justify potential revenue-generating strategies.

Budget Allocation:

- **API Costs:** Expenses for Google Maps API and Gemini API for tracking and analysis.
- **Firebase Costs:** Costs for real-time database, storage, and authentication services.
- **Maintenance Costs:** Budget for app maintenance, including updates, feature enhancements, and server maintenance.
- **Deployment Costs:** Expenses for Play Store setup and app deployment.
- **Marketing Costs:** Initial advertising costs to promote the app.

Cost Estimation:

The following table 5.3.1 illustrates the annual cost estimation of the UniRide project:

Table 5.3.1: Annual Cost Estimation

SN.	Category	Description	Estimated Cost (BDT)
01	API Costs	Google Maps API, Gemini API	7,00,000 annually
02	Firebase Costs	Real-time database, storage and authentication	12,00,000 annually
03	Maintenance Costs	Bug Fixes, Updates, Feature Enhancements, Server Maintenance	4,50,000 annually
04	Deployment Costs	Play Store setup and deployment expenses	3,000 one time
05	Marketing Costs	Advertising and promotional expenses	30,000 one time
		Annual Estimated Total Cost	23,83,000

Ongoing Maintenance:

To ensure UniRide remains functional, secure, and up-to-date, resources have been allocated for ongoing maintenance. This includes regular updates to fix bugs, implement

new features, and enhance functionalities based on user feedback. Continuous monitoring will maintain optimal performance and user satisfaction. Additionally, the budget covers expenses for Firebase backend services and API usage, ensuring reliable data storage, real-time updates, and user authentication. By managing these aspects effectively, UniRide aims to provide a seamless and reliable transportation solution for the university community.

5.4 Complex Engineering Problem

This section addresses the complex engineering challenges encountered throughout the development of the UniRide system.

5.4.1 Complex Problem Solving

The challenges faced during these processes required constant iterations and problem-solving to optimize the system, ensuring smooth functionality under various real-world conditions. The detailed analysis and innovative solutions applied highlight the project's complexity and the robust engineering approaches used to address them.

Table 5.4.1.1 below illustrates how the Knowledge Profile (K), and the Attainment of Complex Engineering Problems (EP) are addressed:

Table 5.4.1.1: Mapping with Complex Problem-Solving

EP1 Depth of Know-ledge	**EP2 Range of Conflicting Requirements**	**EP3 Depth of Analysis**	**EP4 Familiarity of Issues**	**EP5 Extent of Applicable Codes**	**EP6 Extent of Stakeholder involvement**	**EP7 Inter-dependence**
√	√	√	√		√	√

Mapping with Knowledge Profile for EP1

The following table 5.4.1.1 shows how EP1 is mapped to the Knowledge Profile. This mapping includes the relevant categories of knowledge that align with the engineering problem solving approach taken in the project.

Table 5.4.1.2: Mapping with Knowledge Profile

K3 Engineering Fundamentals	**K4 Specialist Knowledge**	**K5 Engineering Design**	**K7 Comprehension**	**K6 Engineering Practice**	**K8 Research Literature**
√	√	√		√	√

5.4.2 Rationale for Mapping

EP1 - Depth of Knowledge

For EP1, the project aligns with several categories from the knowledge profile:

Engineering Fundamentals (K3): The project relies on core principles of engineering, such as data collection, analysis, and system design, essential for real-time bus tracking and route planning.

Specialist Knowledge (K4): Specialized knowledge is applied in understanding GPS technology, mobile app development, and AI algorithms for driver reviews.

Engineering Design (K5): The project involves the design and development of an Android application, requiring a deep understanding of software engineering and user interface design.

Engineering Practice (K6): Practical application of these technologies in real-world scenarios is crucial for validating and improving the transportation system.

Comprehension (K7): The project applies engineering practice to real-world scenarios, such as integrating GPS tracking, AI-driven driver reviews, and Firebase for data management.

Research Literature (K8): The project draws on existing research related to GPS tracking, mobile applications, and AI-driven feedback systems to ensure a grounded and effective approach.

EP2 - Range of Conflicting Requirements

The project addresses multiple conflicting requirements, such as integrating real-time tracking, AI-driven reviews, and user notifications while maintaining usability, security, scalability, and compliance with regulations. This requires engineering practice (K6), engineering design (K5), and mathematics (K2).

EP3 - Depth of Analysis

The depth of analysis in the project involves selecting appropriate technologies and frameworks to optimize user experience, such as Google Maps API for tracking and Gemini API for AI-driven reviews. This requires engineering fundamentals (K3), specialist knowledge (K4), and comprehension (K7).

EP4 - Familiarity of Issues

The project addresses familiar issues related to transportation and logistics but applies advanced technology to solve them, requiring engineering practice (K6) and research literature (K8).

EP6 - Extent of Stakeholder Involvement

The project involves multiple stakeholders, including students, staff, faculty, and administrators, each with distinct and sometimes conflicting requirements. This requires comprehension (K7), engineering design (K5), and engineering practice (K6).

EP7 - Interdependence

The project demonstrates a high degree of interdependence between components like data collection, system design, and real-time application, ensuring each stage is built upon the previous one for overall system effectiveness. This requires engineering fundamentals (K3), engineering design (K5), and comprehension (K7).

The mapping with complex problem-solving categories demonstrates that this project involves significant engineering challenges. It requires a blend of fundamental

knowledge, specialized expertise, practical application, and extensive analysis. The involvement of multiple stakeholders and the interdependence of various project components further highlight the complexity of the problem being addressed.

5.4.3 Engineering Activities

In this section, we map the activities associated with solving the complex engineering problem of real-time bus tracking and AI-enhanced driver reviews to different engineering activity categories. Each mapping is accompanied by a rationale, explaining how our project aligns with these categories.

Table 5.4.2.1 below illustrates the complex engineering activities (EA) performed:

Table 5.4.2.1: Mapping of Engineering Activities

EA1 Range of Sources	**EA2 Level of Interaction**	**EA3 Innovation**	**EA4 Consequences for Society and Environment**	**EA5 Familiarity**
√	√	√	√	√

Mapping with Engineering Activities for EA1: Range of Sources

The project utilized a variety of resources, including GPS technology for real-time tracking, Firebase for backend data management, Google Maps API for mapping and route details, and Gemini API for AI-driven driver reviews.

Mapping with Engineering Activities for EA2: Level of Interaction

The project involved extensive interaction with university stakeholders, including students for feedback on usability and features, drivers for real-time location updates, and administrators for system oversight and approval.

Mapping with Engineering Activities for EA3: Innovation

The project demonstrated innovation by integrating real-time GPS tracking with an intuitive mobile interface and implementing AI algorithms to summarize and improve driver reviews.

Mapping with Engineering Activities for EA4: Consequences for Society and Environment

The project's impact includes reducing wait times and improving transportation efficiency, enhancing safety by providing occupancy status and emergency notifications, and contributing to a smoother and more predictable campus commute.

Mapping with Engineering Activities for EA5: Familiarity

The methods employed, including GPS tracking and mobile app development, are well-established. However, the project’s innovation lies in its application to improve campus transportation, a unique challenge requiring a tailored solution.

5.5 Summary

The UniRide project exemplifies a comprehensive approach to developing a real-time vehicle tracking and AI-enhanced driver review system for university transportation. By adhering to established software, hardware, and communication standards, the application ensures reliability, security, and user satisfaction. The project positively impacts society and the environment by promoting sustainable transportation practices, enhancing safety, and improving the daily commuting experience for students and staff. Ethical considerations and a sustainability plan further support the project's long-term success and community acceptance. Effective project management and detailed financial analysis have guided the efficient allocation of resources and budget. The complex engineering problems encountered were addressed through innovative problem-solving and systematic engineering activities, resulting in a robust and scalable solution.

Chapter 6
Conclusion

6.1 Summary

In conclusion, UniRide presents a comprehensive solution to the pressing challenges faced by university transportation systems. By integrating real-time tracking, AI-driven driver reviews, and user-friendly features, the app significantly enhances the commuting experience for students and staff. The project not only addresses the issues of reliability and efficiency but also fosters community engagement and promotes environmentally sustainable practices. The positive feedback received from users and the successful implementation of key features validate the app's design and functionality, demonstrating its potential to improve transportation within the university setting.

6.2 Limitation

Despite its strengths, UniRide is not without limitations. One significant challenge is ensuring the app is accessible to all users, particularly those who may not be as tech-savvy. Additionally, reliance on external services, such as GPS and internet connectivity, can affect the app's performance in areas with poor reception.

Moreover, there may be conflicts of interest regarding data privacy and user security. It is crucial to maintain transparency about how user data is collected, stored, and utilized to foster trust and compliance with privacy regulations.

Finally, continuous monitoring and evaluation are essential to ensure that the app's deployment does not unintentionally increase existing inequalities within the campus transportation system.

By acknowledging these limitations and actively working on suggested improvements, UniRide can continue to evolve, meeting the needs of its users while contributing positively to the university community and the environment.

6.3 Future Work

While UniRide has laid a solid foundation, several areas for future development can enhance its effectiveness and user experience:

- **Enhanced AI Capabilities:** Future iterations could include more sophisticated AI algorithms to analyze user feedback in greater depth, providing even more actionable insights for driver performance improvement.
- **Data Visualization and Analysis:** With the vehicle tracking data collected, I can visualize metrics such as the time taken to complete routes at different times of day, average travel times, and peak hours. This information could be invaluable for managing class schedules and optimizing transportation resources, leading to improvements in the overall transport system.
- **Subscription System:** Introducing a subscription-based model for using the app could ensure consistent access to its features and updates. This system could provide a steady revenue stream and ensure the sustainability of the service.
- **Longitudinal Studies:** Conducting studies to track the long-term impact of UniRide on transportation efficiency and user satisfaction can inform further enhancements.

While UniRide has laid a solid foundation, its future development holds the potential to further enhance its capabilities, offering more personalized and integrated services, improving user experience, and fostering sustainable growth through continuous innovation and data-driven insights.

References

[1] M. S. Alamgir, I. Jahan, N. Aktar, and A. N. Ramisa, “Chittagong University Teachers' Bus Tracking System Using Smartphone Application”, Sept. 2018, doi: https://doi.org/10.1109/ceeict.2018.8628098.

[2] M. H. Chowdhury, N. Zahan, I. B. Habib, S. Akter, “Android Based Vehicle Tracking System Using GPS Sensor”, Jan. 2017. Available: https://www.researchgate .net/publication/341201887_Android_Based_Vehicle_Tracking_System_Using_GPS _Sensor.

[3] S. A. Sharif, M. S. Suhaimi, N. N. Jamal, I. K. Riadz, I. F. Amran, and D. N. A. Jawawi, “Real-Time Campus University Bus Tracking Mobile Application”, IEEE Xplore, Jul. 2018. Available: https://ieeexplore. ieee.org/abstract/document/8523915.

[4] S. Lee, G. Tewolde, J. Kwon, “Design and implementation of vehicle tracking system using GPS/GSM/GPRS technology and smartphone application”, 2014 IEEE World Forum on Internet of Things (WF-IoT), Mar. 2014, doi: https://doi.org/10.1109/wf-iot.2014.6803187.

[5] N. Jahan, K. Hossen, M. K. H. Patwary, “Implementation of a vehicle tracking system using smartphone and SMS service”, IEEE Xplore, Sept. 2017. Available: https://ieeexplore.ieee.org /document/8255429.

[6] DoubleMap Bus Tracker, available at: https://play.google.com/store/apps/details?id= com.doublemap.iu, last accessed on Oct. 12, 2024 at 06:32 PM.

[7] Chalo - Live Bus Tracking App, available at: https://play.google.com/store/apps/ details?id=app.zophop, last accessed on Oct. 12, 2024 at 06:35 PM.

[8] JU Transport, available at: https://play.google.com/store/apps/details?id=com.sbiitju. jugreenbus, last accessed on Oct. 12, 2024 at 06:36 PM.

[9] Amader BRTC, available at: https://play.google.com/store/apps/details?id=live.dingi. brtc.amader, last accessed on Oct. 12, 2024 at 06:36 PM.

Plagiarism Report

ORIGINALITY REPORT

10%	7%	2%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	4%
2	dspace.daffodilvarsity.edu.bd:8080 Internet Source	3%
3	Submitted to United International University Student Paper	1%
4	zombiedoc.com Internet Source	1%
5	internship.daffodilvarsity.edu.bd Internet Source	<1%
6	Submitted to University of Finance – Marketing Student Paper	<1%
7	v1.overleaf.com Internet Source	<1%
8	Submitted to The University of Manchester Student Paper	<1%
9	Submitted to Brunel University Student Paper	<1%