

NTU Ride Pilot

24-FYP-204



Session 2021-2025

BACHELOR OF SCIENCE IN COMPUTER SCIENCE

SUBMITTED BY

Abubakar Nadeem	21-NTU-CS-1291
Imran Ali Niaz	21-NTU-CS-1321
Huzaifa Ahmad	21-NTU-CS-1318

SUPERVISED BY

Supervisor: Mr. Zahid Javed

Co-Supervisor: Mr. Muhammad Naeem

Department of Computer Science

National Textile University, Faisalabad

Certification

This is to certify that this project titled “**NTU Ride Pilot**” was found to satisfy the requirement for the award of a “**Bachelor of Sciences in Software Engineering**” degree by the Department of Computer Science, National Textile University.

Supervisor

Mr. Zahid Javed

Signature

Date

Co-Supervisor

Mr. Muhammad Naeem

Signature

Date

External Examiner

Signature

Date

Internal Examiner

Dr. Hamid

Signature

Date

FYP-Convenor

Signature

Date

Head of Department

Signature

Date

Declaration

We hereby declared that this document is completely written by us, and it is totally our effort and none of anyone from outside of our group has copied it. This Report is purely written technically in accordance with our project.

Group Members:

Abubakar Nadeem

21-NTU-CS-1291

Signature

Date

E-Mail: abubakaranjum066@gmail.com

Imran Ali Niaz

21-NTU-CS-1321

Signature

Date

E-Mail: imranaliniaz786@gmail.com

Huzaifa Ahmad

21-NTU-CS-1318

Signature

Date

E-Mail: huzaifa.ahmad486@gmail.com

Acknowledgement

First of all, In the Name of Allah, the Most Beneficent, the Most Merciful. All the praises and thanks be to Allah. A lot of love for our beloved Holy Prophet MUHAMMAD (S.A.W), his guidance always helps us to get the right path. After that, the preparation of this Software Requirements Specification (SRS) document was a collaborative effort involving Abubakar, Imran, and Huzaifa. The author would like to acknowledge and express gratitude to all individuals who directly or indirectly contributed to the completion of this document. Special thanks are extended to our Supervisor Mr. Zahid Javed and our Co-Supervisor Mr. Muhammad Naeem, who provided invaluable guidance and support throughout the SRS preparation process. The author would also like to acknowledge the invaluable support and suggestions provided by Mr. Zahid Javed. This SRS document would not have been possible without the support and cooperation of all involved individuals. The author is confident that this document will serve as a comprehensive guide for the development and implementation of the NTU Ride Pilot.

Abstract

Efficient and secure transportation is vital for educational institutions, where thousands of students rely on daily commutes. However, current systems face significant challenges, including inefficiencies, security vulnerabilities, and a lack of real-time tracking and communication. Issues such as overcrowding, unauthorized access, and poor resource utilization arise when students board buses without proper verification. Additionally, the absence of real-time bus monitoring and effective communication channels leads to confusion, delays, and suboptimal transport management. The NTU Ride Pilot addresses these challenges by integrating advanced technologies like RFID-based ID verification, live GPS tracking, and automated communication tools. The system ensures only authorized students access transport services, provides real-time visibility of bus locations for students, parents, and administrators, and enables timely updates regarding delays, route changes, or emergencies. It also tracks driver performance, monitors bus occupancy, and generates insights for optimizing route planning and capacity utilization. By modernizing transport operations, the NTU Ride Pilot enhances safety, boosts operational efficiency, and delivers a reliable and user-friendly commuting experience. This solution is an ideal choice for educational institutions aiming to transform their transportation infrastructure with secure and efficient technology.

Table of Contents

Title	Page #
Certification.....	II
Declaration	III
Acknowledgement.....	IV
Abstract	V
List of Figures	XI
List of Tables	XII
List of Abbreviations	XIII
CHAPTER 1.....	1
Introduction	1
1.1 NTU Ride Pilot.....	1
1.2 Reason to Develop.....	2
1.2.1 Addressing Unique Institutional Needs.....	2
1.2.2 Enhancing Safety and Security	2
1.2.3 Bridging Communication Gaps	2
1.2.4 Optimizing Resource Management.....	2
1.2.5 Scalability and Adaptability	2
1.2.6 Sharing Live Location.....	3
1.3 Problem Statement.....	3
1.4 Purpose	3
1.5 Project Goals	3
1.6 Objectives	4
1.7 Project Scope	4
1.8 Proposed Solution.....	4
1.9 Cost Benefit Analysis	4
1.9.1 Cost Analysis.....	5
1.9.2 Operational & Maintenance Costs	5
1.9.3 Benefit Analysis	5
1.9.4 Future Recommendations for Cost Optimization	6
1.10 Project Scheduling.....	7
CHAPTER 2.....	8
Literature Review	8
2.1 Related Work	8
2.1.1 Tyler Technologies' Student Transportation Software	8

2.1.2 Edulog	8
2.1.3 Loqqat	9
2.2 RFID Technology in Transportation Systems	9
2.3 GPS-Based Bus Tracking	9
2.4 Load Management in Public Transportation	9
2.5 Predictive Arrival Systems	10
2.6 Google Maps and Map box	10
2.6.1 Google Maps API.....	10
2.6.2 Map box	10
CHAPTER 3	11
System Requirements	11
3.1 Functional Requirements.....	11
3.1.1 User Authentication and Authorization.....	11
3.1.2 Bus and Route Management	11
3.1.3 Ride Management	11
3.1.4 Bus Card Management.....	12
3.1.5 Session and Student Management.....	12
3.1.6 Driver Management	12
3.1.7 Complaint Management	12
3.1.8 Announcement Management.....	12
3.1.9 Notification and Alerts	12
3.1.10 Live Location Tracking	12
3.2 Non-Functional Requirements.....	13
3.2.1 Security	13
3.2.2 Performance	13
3.2.3 Availability	13
3.2.4 Scalability.....	13
3.2.5 Usability	13
3.2.6 Maintainability	13
3.2.7 Efficiency	13
3.3 Use Case Diagram	13
3.3.1 Use Case of General Functionalities	14
3.3.2 Use Case of Admin Panel.....	15
3.3.3 Use Case of Mobile App	16
3.3.4 Use Case of Sign In.....	17

3.3.5 Use Case of Sign Up	17
3.3.6 Use Case of Bus and Route Management.....	18
3.3.7 Use Case of Ride Management.....	18
3.3.8 Use Case of Bus Card Management	19
3.3.9 Use Case of Student & Session Management.....	19
3.3.10 Use Case of Bus Staff Management.....	20
3.3.11 Use Case of Complaint Management.....	20
3.3.12 Use Case of Announcement Management	21
3.4 Use Case Description	22
3.4.1 Description of Sign In	22
3.4.2 Description of Sign Up	23
3.4.3 Description of Bus and Route Management	24
3.4.4 Description of Ride Management	25
3.4.5 Description of Bus Card Management.....	26
3.4.6 Description of Session and Student Management.....	27
3.4.7 Description of Bus Staff Management.....	28
3.4.8 Description of Complaint Management	28
3.4.9 Description of Announcement Management.....	29
3.4.10 Description of General Functionalities	29
Chapter 4	30
Methodology	30
4.1 Agile Software Development Methodology.....	30
4.2 Selected Methodology: Agile	30
4.3 Reasons for Selecting Agile Methodology	30
4.4 Project Planning and Execution for NTURP	30
Chapter 5	32
Architecture Design.....	32
5.1 Activity Diagram	32
5.1.1 Mobile Application	32
5.1.2 Admin Panel	33
5.2 Database Design	34
5.3 Sequence Diagram.....	35
5.3.1 Admin Panel.....	35
5.3.2 Mobile App	36
CHAPTER 6.....	37

System Implementation.....	37
6.1 System Tools and Technologies.....	37
6.2 Class Diagram	37
6.2.1 Mobile Application	38
6.2.2 Admin Panel.....	39
6.3 Deployment Diagram	40
CHAPTER 7.....	41
System Testing.....	41
7.1 White Box Testing	41
7.2 Black Box Testing	41
7.3 Verification	42
7.4 Validation.....	42
7.5 Adopted Methodology	43
7.5.1 Unit Testing.....	43
7.5.2 Module Testing.....	43
7.5.3 Integration Testing	43
7.5.4 System Testing	43
7.5.5 Acceptance Testing.....	43
7.6 Test Cases	44
CHAPTER 8.....	47
Application Prototype.....	47
8.1 Mobile Application Prototype (Flutter).....	47
8.1.1 Splash & Login Screens.....	47
8.1.2 Role-Based Navigation & Welcome Screens.....	50
8.1.3 Student Ride & Route Info Screens	54
8.1.4 Driver Ride Selection & Ride Control Screens	57
8.1.5 Profile & Settings Screens	60
8.1.6 Feedback & Complaint Screen (Student and Driver)	63
8.1.7 Help & Password Recovery Screens.....	64
8.1.8 Notifications Screen (Student and Driver).....	66
8.2 Admin Panel Prototype (Web).....	67
8.2.1 Admin Login Page	67
8.2.2 Dashboard Overview.....	67
8.2.3 Sessions Management	68
8.2.3.1 Sessions Details.....	68

8.2.3.1 Add Sessions	68
8.2.4 Student Management.....	69
8.2.4.1 Student Details	69
8.2.4.2 Add Students	69
8.2.5 Driver Management	70
8.2.5.1 Driver Details	70
8.2.5.2 Add Drivers	70
8.2.6 Routes Management.....	71
8.2.6.1 Route Details	71
8.2.6.2 Add Routes	71
8.2.7 Bus Management.....	72
8.2.7.1 Bus Details	72
8.2.7.2 Add Bus.....	72
8.2.8 Announcements Management Module.....	73
8.2.8.1 Announcement Details	73
8.2.8.2 Add Announcements	73
8.2.9 Complaints Management	74
8.2.9.1 View all Complaints	74
8.2.9.2 View single Complaint.....	74
8.2.10 Rides Management.....	75
8.2.11 General Reports Module	75
References	76

List of Figures

Figure 1 Gantt Chart	7
Figure 2 Use Case of General Functionalities	14
Figure 3 Use Case of Admin Panel.....	15
Figure 4 Use Case of Mobile App	16
Figure 5 Use Case of Sign In	17
Figure 6 Use Case of Sign Up.....	17
Figure 7 Use Case of Bus and Route Management	18
Figure 8 Use Case of Ride Management	18
Figure 9 Use Case of Bus Card Management.....	19
Figure 10 Use Case of Student & Session Management	19
Figure 11 Use Case of Bus Staff Management.....	20
Figure 12 Use Case of Complaint Management	20
Figure 13 Use Case of Announcement Management	21
Figure 14 Agile Model.....	31
Figure 15 Activity diagram of mobile app.....	32
Figure 16 Activity diagram of Admin panel.....	33
Figure 17 Database Design	34
Figure 18 Sequence diagram of Admin Panel	35
Figure 19 Sequence diagram of Mobile App	36
Figure 20 Class diagram of mobile app	38
Figure 21 Class diagram of admin panel	39

List of Tables

Table 1 List of Abbreviations	XIII
Table 2 Cost Analysis	5
Table 3 Description of Sign In.....	22
Table 4 Description of Sign Up	23
Table 5 Description of Bus and Route Management	24
Table 6 Description of Ride Management	25
Table 7 Description of Bus Card Management.....	26
Table 8 Description of Session and Student Management	27
Table 9 Description of Bus Staff Management.....	28
Table 10 Description of Complaint Management.....	28
Table 11 Description of Announcement Management	29
Table 12 Description of General Functionalities	29
Table 13 Ride Start/Stop	44
Table 14 RFID Card Validation.....	44
Table 15 Complaint Submission.....	45
Table 16 Announcements	45
Table 17 Live Location Tracking	46
Table 18 Session Expiry Handling	46

List of Abbreviations

Table 1 List of Abbreviations

NTURP	NTU Ride Pilot
RFID	Radio Frequency Identification
NTURP system	Admin Panel, Driver/Student App
GPS	Global Positioning System

CHAPTER 1

Introduction

Proper transportation is a requirement in school systems of all types, where many thousands of students face daily transportation needs. But many institutions have problems such as ineffective workflow, potential and noted security threats, no real-time control and messaging systems. Children board buses without identification hence self-aggregation, trespass and misuse of transport means are common as are overcrowding. Further, there is the lack of real time tracking in buses as well as the performance of the drivers, and poor communication channels leads to confusion and poor control and co-ordination of transport operations.

To counter these challenges, we are optimistic about the implementation of the proposed NTURP system in an educational setting. This system incorporates high technologies such as RFID used for the identification process, real-time tracking using GPS, and technology-reliant alerting mechanisms for better transport logistics management. How it helps: It ensures that only approved students gain access to students' transportation services, gives real-time mapping of the bus locations to students, parents, and administrative staff and assists in timely notification on any ... delays, route alterations, or emergent situations. Further, it measures driving behaviour, controls the passenger load factor, and provides notifications for route efficiency and capacity management.

This solution does not address objectives present in the current status quo; it revolutionizes transport management turning it into a safe, optimised, and friendly environment for users. With the help of new technologies, the NTURP system increases security, optimizes the work of transport and ensures its safe functioning, which makes it a perfect choice for any school with the desire to create modern transport infrastructure.

1.1 NTU Ride Pilot

NTURP is new approach for improving transportation services and gradually making transport secure and safe for education institutions. It encompasses up-to-date solutions such as RFID for persons' identification, real-time GPS navigation, and an enhanced communication system to build an easy-to-navigate transportation environment. This system enables authorised institutions to finally determine the best routes for bus

transport, the best capacity to be provided for each transport at a given period as well as the performance of the transporters in ensuring that students, parents and other members of the institution get secure and efficient means of transport.

1.2 Reason to Develop

The development of the NTURP system is driven by several compelling factors, despite the existence of other transportation solutions:

1.2.1 Addressing Unique Institutional Needs

Schools are bound to face certain problems such as intrusion, traffic congestion and improper signalling. These specific requirements cannot be served efficiently by generic transport management systems, and thus the need to develop a transport management solution for this environment specifically.

1.2.2 Enhancing Safety and Security

This is important to avoid insecurity or that some students who are not using transport facilities benefit as others who deserve it are locked out. RFID approach encompasses accurate and almost instantaneous means of reducing misuse and increasing security for both students and drivers.

1.2.3 Bridging Communication Gaps

Failure to share information with other transport departments, with parents and students results to disorganization. This presented system entails use of instantaneous notification for news such as delay, change of route and often emergencies to promote timely and precise communication.

1.2.4 Optimizing Resource Management

Overcrowding or the underutilization of buses is a common issue. By integrating occupancy monitoring and route optimization, this system helps administrators allocate resources more efficiently, reducing costs and enhancing comfort.

1.2.5 Scalability and Adaptability

The system can suit the requirements necessary to smoothly operate with fleet in any type of educational institution regardless of the scale of transportation system present there. It is also scalable where the institutions will be able to increase their profit as it responds to the changing transport needs.

This is much more than a technology enhancement project; it is a transformational project across the transport system that takes the future of transport safety and efficiency into consideration in an educational environment.

1.2.6 Sharing Live Location

Through real-time tracking parents as well as students and school administrators can verify transportation location information. The tracking functionality cuts uncertainty while optimizing scheduling and provides boosted security thanks to real-time bus position monitoring. This initiative surpasses technology adoption to become an educational transformation that advances safety measures and managerial efficiency and operational effectiveness in the student transport system.

1.3 Problem Statement

Public and private learning institutions are among the organizations that experience high levels of challenges when it comes to transportation management. Some of the widespread problems are overcrowded or, in contrast, underfilled buses, theft, poor communication with the bus drivers, and absence of the possibility of tracking a bus's location. Such issues thus complicate movement, organization, and functioning, and pose risk to the learners, their parents, and school management. Currently available solutions do not always have the architecture and flexibility to meet these specific institutional requirements.

1.4 Purpose

The primary objective of the developed NTURP system is to become a one-stop for all the transport management needs of educational establishments. Through secure access verification, GPS tracking while the bus is in operation, and other communication features, the developers of the system intend to enhance the performance, safety, and overall communication into the student, parents, school, and bus company.

1.5 Project Goals

- Implement secure ID verification to prevent unauthorized access.
- Provide live bus location tracking for real-time visibility and improved coordination.
- Enable real-time notifications for updates such as delays, route changes, or emergencies.

- Track key metrics like speed, stop intervals, and adherence to schedules.
- Streamline bus scheduling, capacity management, and route planning.
- Ensure an intuitive interface for administrators, parents, and students.

1.6 Objectives

- Objectives of the project are as follows:
- Integrate RFID-based ID verification to ensure only authorized users board the buses.
- Provide GPS-enabled tracking for buses accessible to students, parents, and administrators.
- Enable alerts for overcrowding or underutilization to optimize bus capacity.
- Monitor driver behaviour to ensure adherence to safety and efficiency standards.
- Develop a mobile app for notifications and updates to keep all stakeholders informed.

1.7 Project Scope

Even though this system is created for educational institution it can be expanded at other domains including corporate transport, public streamlined or private bus lines. The architecture of FMGO enables flexibility and expansion depending on the transportation needs and its functionality and organization.

1.8 Proposed Solution

The system provides efficient and well-structured functional solutions for transportation management. The solution offers RFID for ID check during entry, GPS tracking for buses in real time, alert on occupancy, checking of driver performance, and mobile applications. Through this system, there will be highly improved efficiency in the running of institutions, minimized wastage of resources, and improved safety and reliability of transport for all users

1.9 Cost Benefit Analysis

As the NTU Ride Pilot functions to enhance NTU University's transportation system effectiveness combined with improved security protocols and user accessibility. This Cost-Benefit Analysis (CBA) conducts financial evaluation by aligning development expenses with operational costs against future benefits.

1.9.1 Cost Analysis

A. Hardware Costs (One-Time Costs for One Bus)

Table 2 Cost Analysis

Item	Cost (RS)	Quantity	Total Cost (RS)
RFID Device	1200	1	1200
Student Cards	40	50	2000
Connector	150	1	150
Total Hardware Cost	3350	1	3350

B. Software Costs

- Map box (Live Tracking) serves the company with free assistance under its Free Tier program.
- The free plan of Firebase Database Management operates currently at no expense.
- The added number of buses at NTU could require purchasing paid plans from Map box and Firebase.

1.9.2 Operational & Maintenance Costs

The project currently uses a free-tier structure but operational expenses alongside maintenance costs constitute the recurring expenses for the future. There are no ongoing expenses since the project development occurs under free-tier conditions. However, potential future costs include:

- The project may require costs for Firebase Paid Plan data storage and database services.
- Future web-based expansion of the system by NTU will incur both domain and hosting fees.
- The project will require expenses for both system upgrades and security patches together with future enhancement and support costs.

1.9.3 Benefit Analysis

A. Tangible Benefits

- RFID technology removes the possibility of unauthorized bus access.

- The current tracking system blocks unauthorized route changes and unauthorized stopovers.
- Automated card checker technology reduces the time needed for students to join the system.
- Customers together with administrative staff can monitor buses online and minimize waiting at bus stops.
- The digital complaint platform diminishes the time needed for resolution while guaranteeing better customer service.
- Through live monitoring admins gain better control over their bus routes which allows them to improve route scheduling according to current use levels.
- The method extends to multiple buses through basic software upgrades.

B. Intangible Benefits

- Real-time tracking combined with efficient student on boarding systems produce superior user satisfaction.
- Students along with parents achieve better peace of mind when they have verified access to the bus system.
- The implementation of digital systems provides two main benefits: it decreases physical documentation while ensuring precise information recording.
- Efficient route management through the system helps reduce emissions and fuel consumption.
- The institution stands ahead as a pioneer through its adoption of modern transportation systems.

1.9.4 Future Recommendations for Cost Optimization

- If NTU expands, evaluate paid plans for Firebase and Map box only when necessary.
- If a web-based system is implemented, consider cost-effective hosting providers.
- Reduce RFID costs by purchasing student cards in bulk.
- Use tracking data to optimize bus schedules, reducing fuel costs.

1.10 Project Scheduling

Below is the Gantt chart that has been developed for the NTURP project. This chart is intended to illustrate the project's schedule: the time when each activity was planned/started and the time when it was planned/ended. They give a broad plan of how the various tasks in the project are expected to be done, and when thus help in keeping track of the project. The time plan of the project is illustrated in the Gantt chart in Figure 1.1 below.

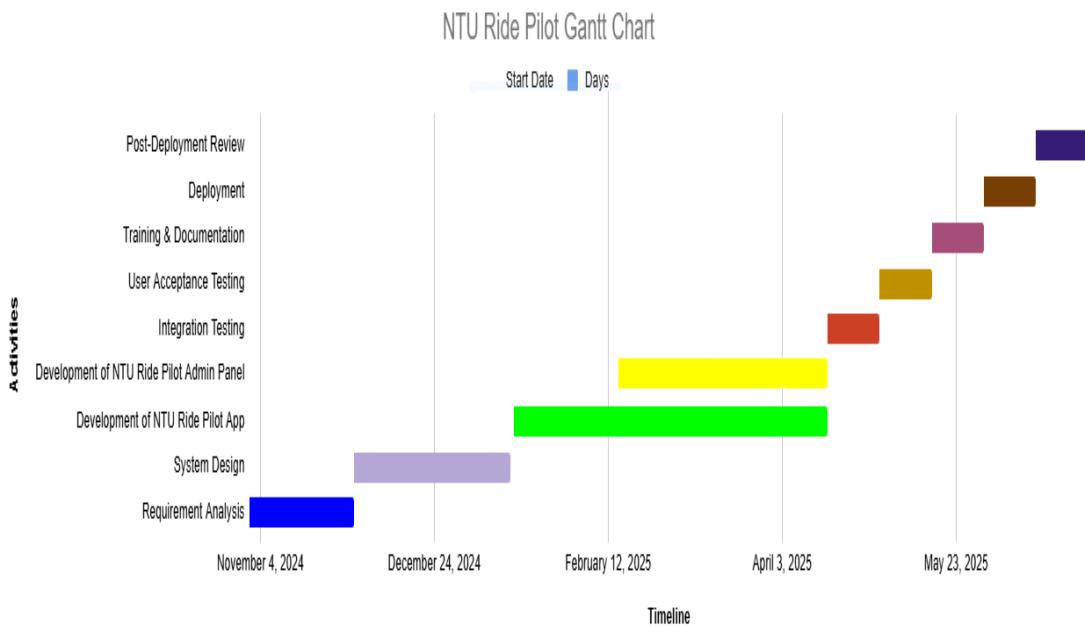


Figure 1 Gantt Chart

CHAPTER 2

Literature Review

The NTURP aims to enhance transportation services for students and parents by integrating technologies such as RFID scanning, GPS tracking, real-time data analytics, and mapping services. This chapter reviews existing literature and technologies pertinent to the project, including RFID technology in transportation systems, GPS-based bus tracking, load management in public transportation, predictive arrival systems, and the utilization of mapping APIs like Google Maps and Map box. Additionally, it examines existing student transportation management systems to identify current solutions and gaps.

2.1 Related Work

Several transportation management systems cater to student transit needs, offering features like GPS tracking, route optimization, and parent communication. Notable examples include:

2.1.1 Tyler Technologies' Student Transportation Software

Provides integrated solutions for bus routing, fleet maintenance, and parent communication, connecting various aspects of transportation management. (TylerTechnologies)

Cons:

- High implementation and licensing costs for smaller institutions.
- Steep learning curve for administrators unfamiliar with the software.
- Limited customization options for unique institutional requirements.

2.1.2 Edulog

Combines school bus routing, GPS fleet tracking, student journey ship management, and parent communication apps into a single platform, aiming to streamline transportation operations. (Edulog)

Cons:

- Complex setup and configuration for multi-campus institutions.
- Frequent updates are causing temporary compatibility issues.
- Limited integration with non-standard hardware or legacy systems.

2.1.3 Loqqat

Provides a smart real-time school bus tracker and management software, enabling route scheduling and live tracking to ensure student safety. (Loqqat)

Cons:

- Narrow focus on live tracking, lacking advanced features like fleet maintenance.
- Limited scalability for large institutions with extensive fleets.
- Higher costs for adding additional features beyond basic tracking.

2.2 RFID Technology in Transportation Systems

RFID is widely used in transportation for access control and validation of the user. The usage of an RFID tag within student cards enables easy barcode scanning and validation strengthening the parameter of user credibility. The literature review also focuses on RFID advantages concerning reliability, fast processing, and accuracy in real situations for passenger identification, which would help in determining eligibility of passenger in bus systems.

RFID technology is also used in NTURP to authenticate fee payment and obtain time-based scanning to reduce misapplication of the system. The integration of RFID with time constraints means that the delivery will be partial and will ensure compliance and consequently trust among the stakeholders.

2.3 GPS-Based Bus Tracking

With GPS, social transport has been revolutionized through tracking vehicles in the road networks. In doing so, passengers and bus administrators can track bus positions thus increasing the buses' operational visibility and service delivery. GPS tracking and the provision of estimated arrival time are found to increase the user satisfaction since the device continually and accurately indicates the location.

In NTURP, GPS tracking helps parents and students in planning their travel effectively. When partnered with predictive algorithms, GPS data eliminates the prospects of early or late estimates and keeps drivers on their toes. The same also contains important information about the best route choice and the most suitable time in the context of transportation by bus.

2.4 Load Management in Public Transportation

It is very important to strike a balance between the number of passengers and the carriage capacity in a public transport system because an excess or a shortage of

passengers causes problems. Methods like weight sensors and real-time passenger count are found efficient for measuring the bus holding capacity.

In addition to real time load analysis for safety of passengers and bus fleet, NTURP also includes efficient bus operation. Prepare for the load according to the existing pattern and data and use the pattern to estimate the demand for various products or services.

2.5 Predictive Arrival Systems

The estimates of arrival are derived from the GPS, traffic and past records making them reliable for PSA. Research shows that such systems improve user trust and satisfaction. Use of enhanced features and the integration of more advanced machine learning algorithms can of course enhance the accuracy of the predictions.

In NTURP, arrival features that predict help to cut down waiting time and thus the result is convenient to both students and parents. These features, implemented in simple presentations with user-friendly interfaces, are helpful for a user.

2.6 Google Maps and Map box

Mapping services are integral to transportation management systems, providing visualization and geolocation functionalities.

2.6.1 Google Maps API

Google Maps API is a full service solution for map integration which allows for real time traffic data, route and time estimations. The versatile Log parser and its reliability is a clear reason why developers choose to work with the company. To work in NTURP, Google Maps API is quite helpful in providing real time location of buses and their estimated time of arrival, thus making the general user interface more effective.

2.6.2 Map box

Map box is an interactive map tool which operating system can be adapted according to the preferences of the developers. It provides dynamic app theme support, working offline maps, and improved integration options. In certain niches of mapping specifically designed solutions, Map box is flexible and fast. For NTURP, Map box gives an opportunity to design compelling, user-friendly front ends for users.

CHAPTER 3

System Requirements

In this bankruptcy, all of the useful requirements of the NTURP and the overall requirement of the stockholders are documented as it's an important a part of a mission or product that allows to satisfy stakeholder's necessities. Now, we can speak system necessities, practical necessities, software program development, and present and selected methodology with the purpose of technique. These sections describe software program methodologies which are present and decided on for this assignment with the glide of machine and alertness detail depicted.

3.1 Functional Requirements

3.1.1 User Authentication and Authorization

There should be user roles supported in the system (Admin, Driver, Conductor, Student) and only allow access after a proper authentication. It must have a module for creating the user base, the role they should have, and the permission of the role to open some of the modules that a user should open.

3.1.2 Bus and Route Management

The Admin should also be able to handle bus detail and want to set territories or routes for buses. The Driver should have the facility to record/update the driving routes for individual buses. Also there exists the scenario where the Admin needs to view a list of available buses and their corresponding routes.

3.1.3 Ride Management

The Driver must be able to initiate or end a ride, with the system tracking the live location of buses during active rides. The Driver is responsible for authenticating student cards when they board the bus, and the system must record ride data, including the bus number, route, boarded students, and the driver. Both Admin and Students must have access to the bus's live location, while Admin also be able to view the complete ride history for all buses.

3.1.4 Bus Card Management

The admin must be able to assign bus cards to students, as well as revoke or enable student bus cards as needed. The system must also verify student bus cards during boarding to ensure proper access.

3.1.5 Session and Student Management

The system must allow the Admin to create and end user sessions, as well as set their expiry dates. It should automatically disable student cards when a session expires or is deleted. Additionally, the system must generate app credentials for students upon their addition to the system.

3.1.6 Driver Management

Admin must be able to add and manage Drivers and Conductors within the system. Upon registration, the system must generate app credentials for these staff members to enable secure access and management of their duties.

3.1.7 Complaint Management

Students and Drivers must have the ability to submit complaints through their apps. Admin should have a module to view, address, and resolve these complaints, and the system must maintain a record of all complaints along with their current statuses.

3.1.8 Announcement Management

Admin must be able to create and manage announcements within the system. Announcements must be delivered as notifications to Drivers and Students through their apps to ensure timely updates.

3.1.9 Notification and Alerts

The system must send notifications to Students and Drivers regarding announcements, route updates, and other relevant information. Additionally, the system must alert Admin if a bus deviates from its assigned route or leaves its designated area.

3.1.10 Live Location Tracking

The system must track and display the real-time location of buses during active rides. Both Admin and Students should be able to access this live location data via their apps, and the system must store location data for ride history and analysis purposes. This tracking feature enhances visibility and allows stakeholders to make informed

decisions. It helps minimize uncertainty regarding arrival times and delays. Additionally, stored location data can be used for performance evaluation and future route optimization.

3.2 Non-Functional Requirements

3.2.1 Security

Apply strict checks for the users' authorization and authenticity.

3.2.2 Performance

The system should be able to answer user actions on the application quickly.

3.2.3 Availability

The system should be online all the time.

3.2.4 Scalability

Also, the system must be capable providing its services to a rising number of users and data.

3.2.5 Usability

Web based system should be easy to use and navigable with an aim of reaching the users of the product.

3.2.6 Maintainability

It should be easy for the system to be maintained and upgraded from time without lots of time being consumed.

3.2.7 Efficiency

While functioning, the system should effectively manage the amount and kinds of resources used and reduce response time.

3.3 Use Case Diagram

In respect to showing graphic representations of actor communication with the components of the systems the best method therefore is to identify and draw Use Case diagrams that illustrate which actor can perform or accessing what function or component of the systems under consideration.

3.3.1 Use Case of General Functionalities

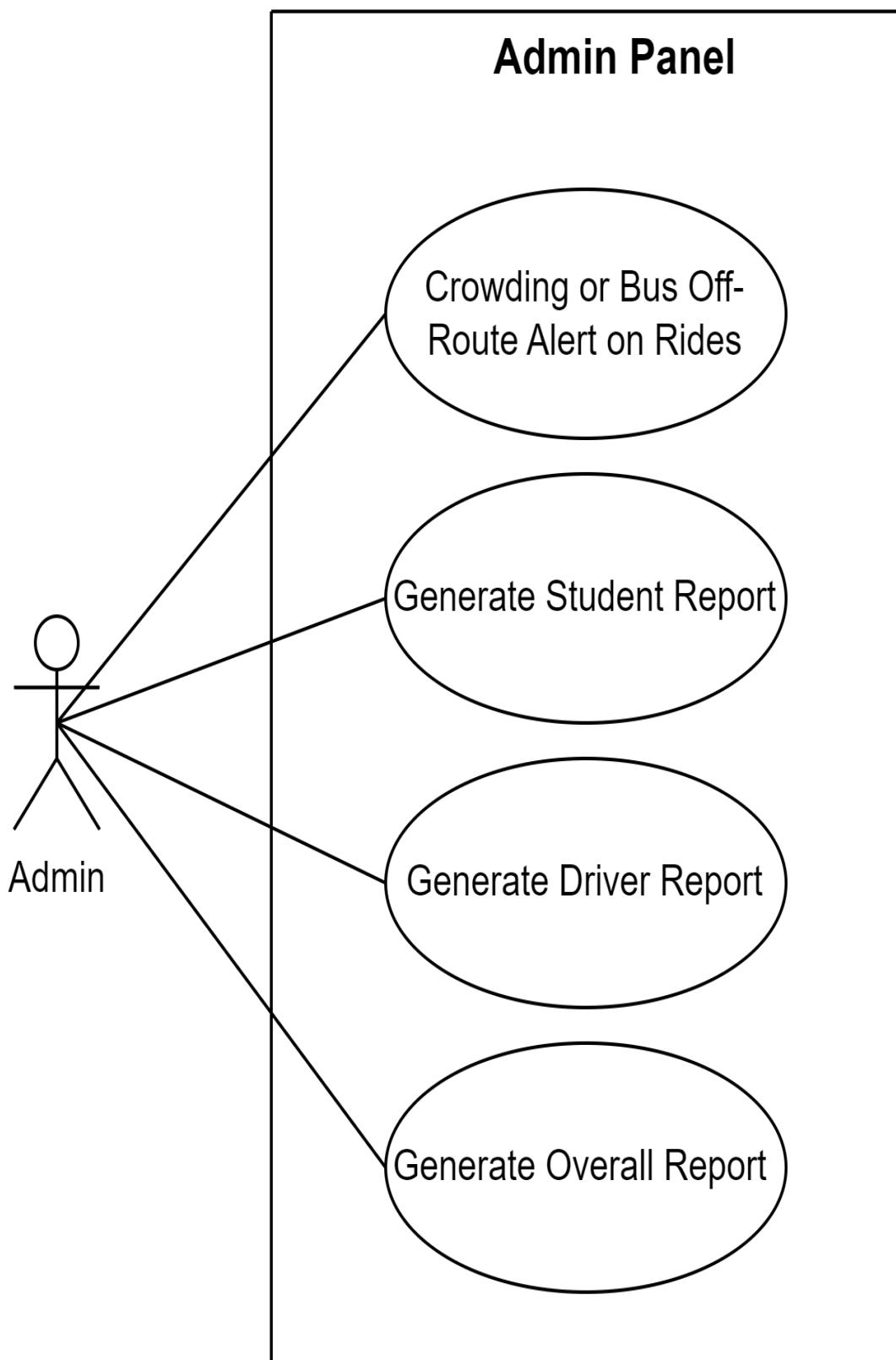


Figure 2 Use Case of General Functionalities

3.3.2 Use Case of Admin Panel

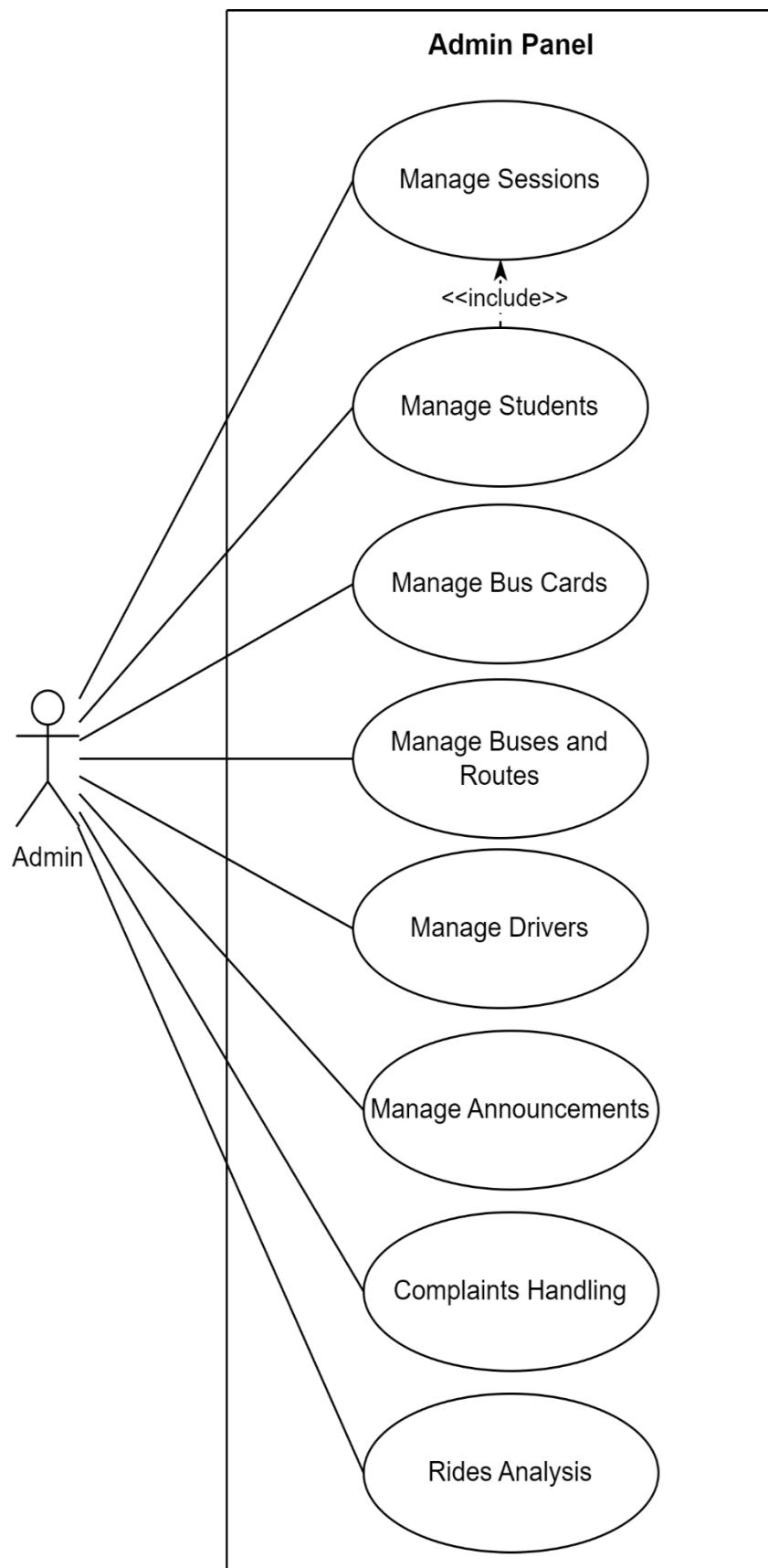


Figure 3 Use Case of Admin Panel

3.3.3 Use Case of Mobile App

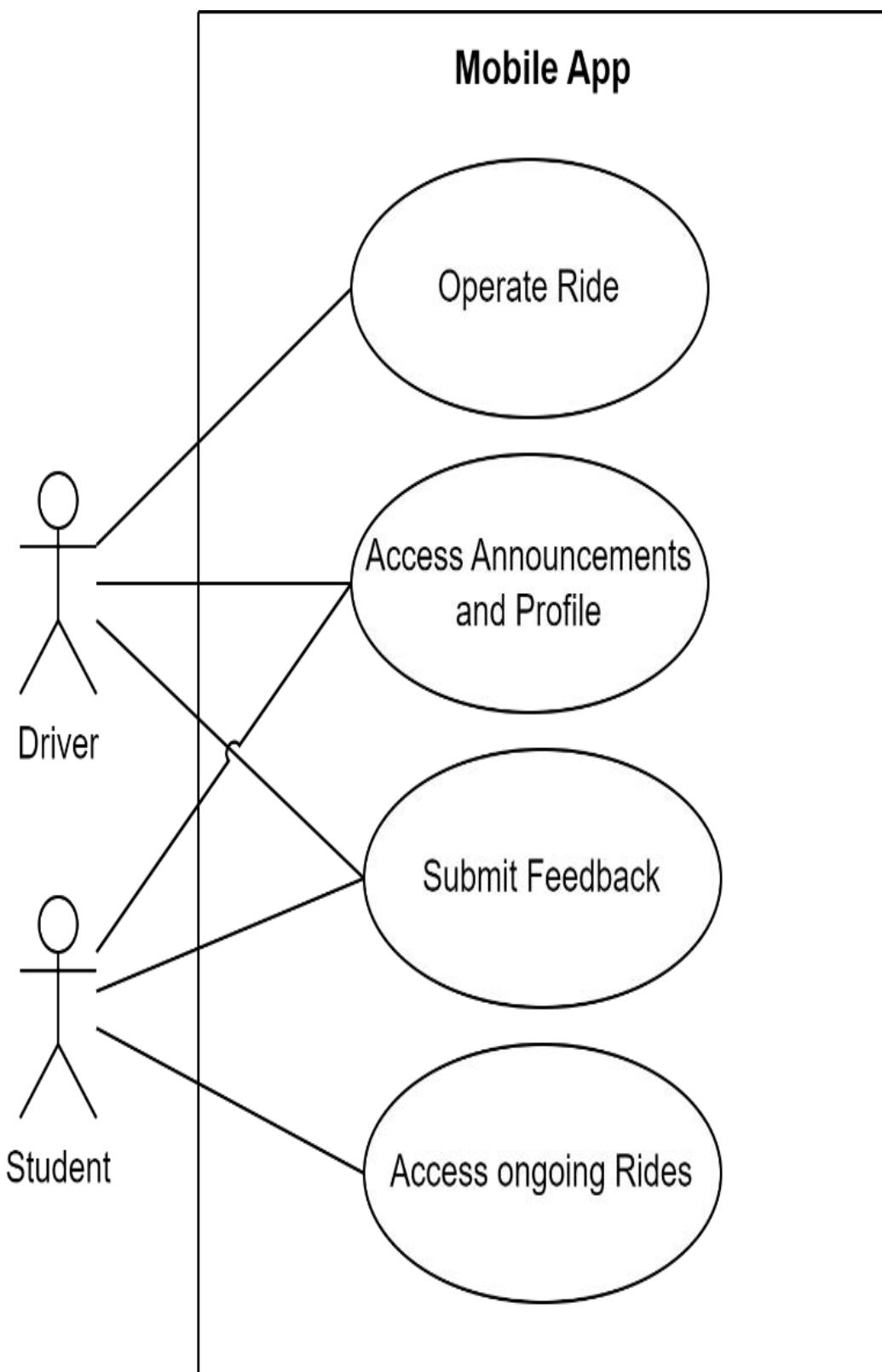


Figure 4 Use Case of Mobile App

3.3.4 Use Case of Sign In

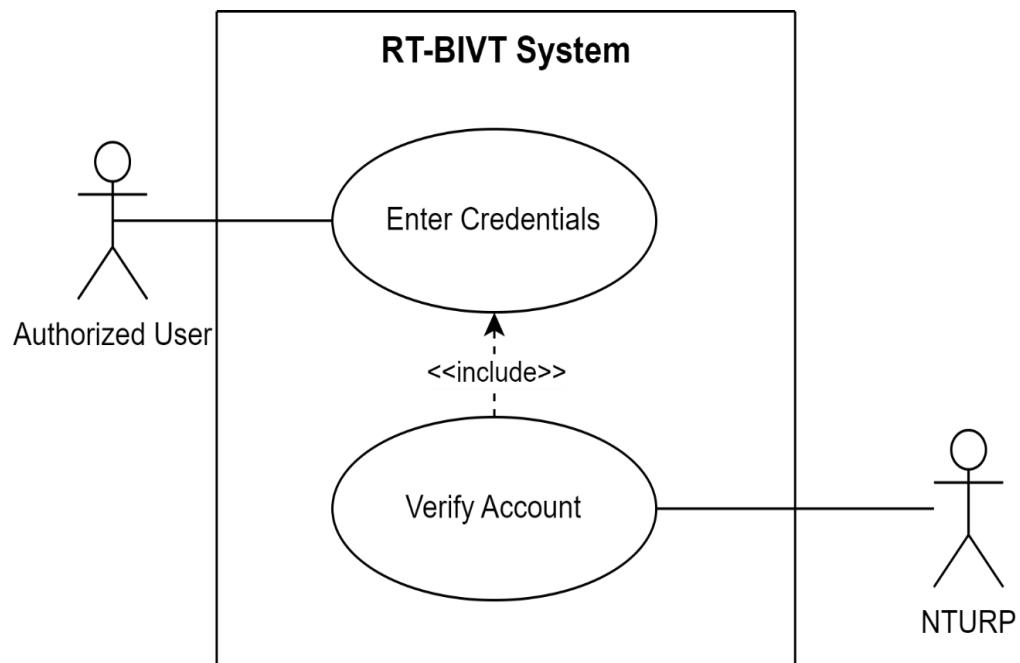


Figure 5 Use Case of Sign In

3.3.5 Use Case of Sign Up

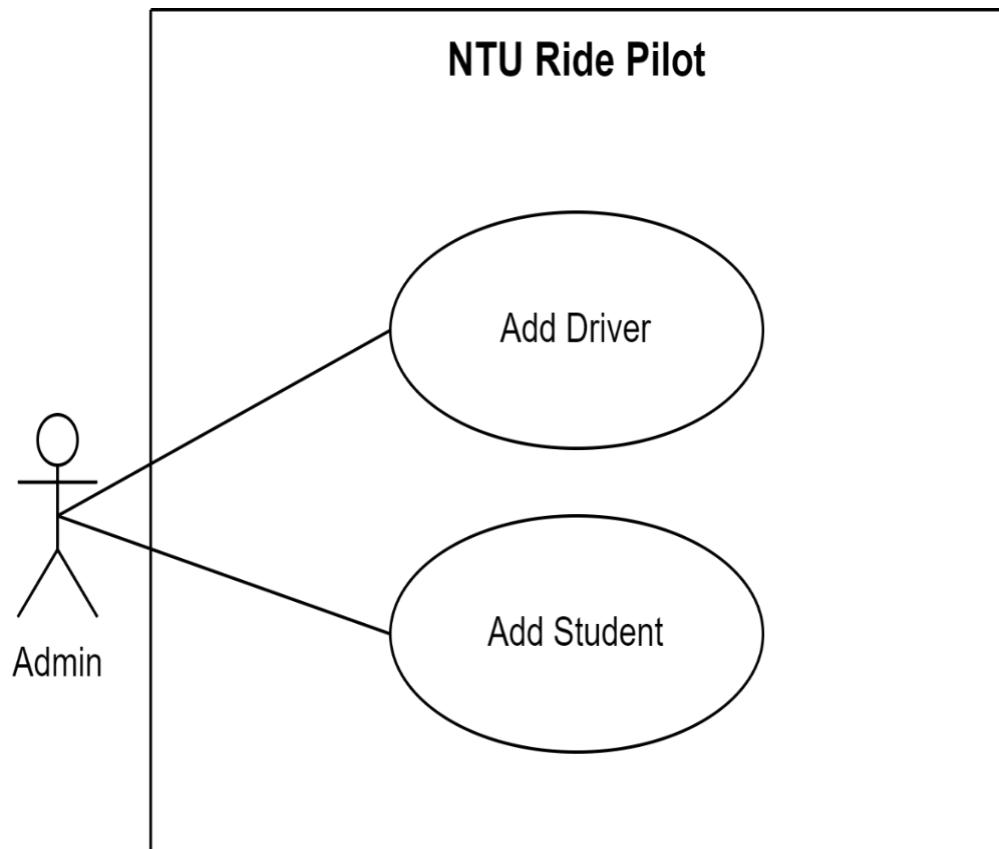


Figure 6 Use Case of Sign Up

3.3.6 Use Case of Bus and Route Management

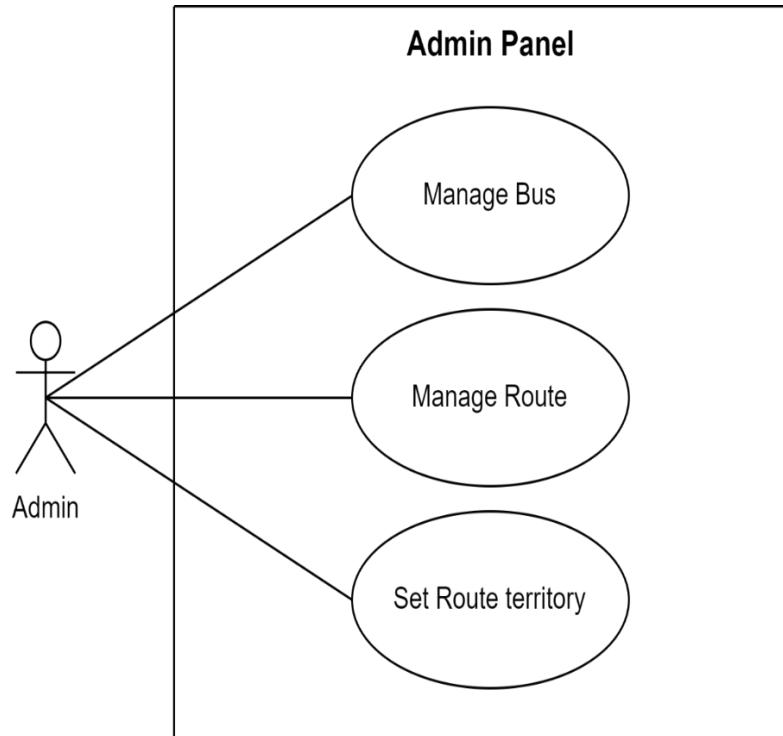


Figure 7 Use Case of Bus and Route Management

3.3.7 Use Case of Ride Management

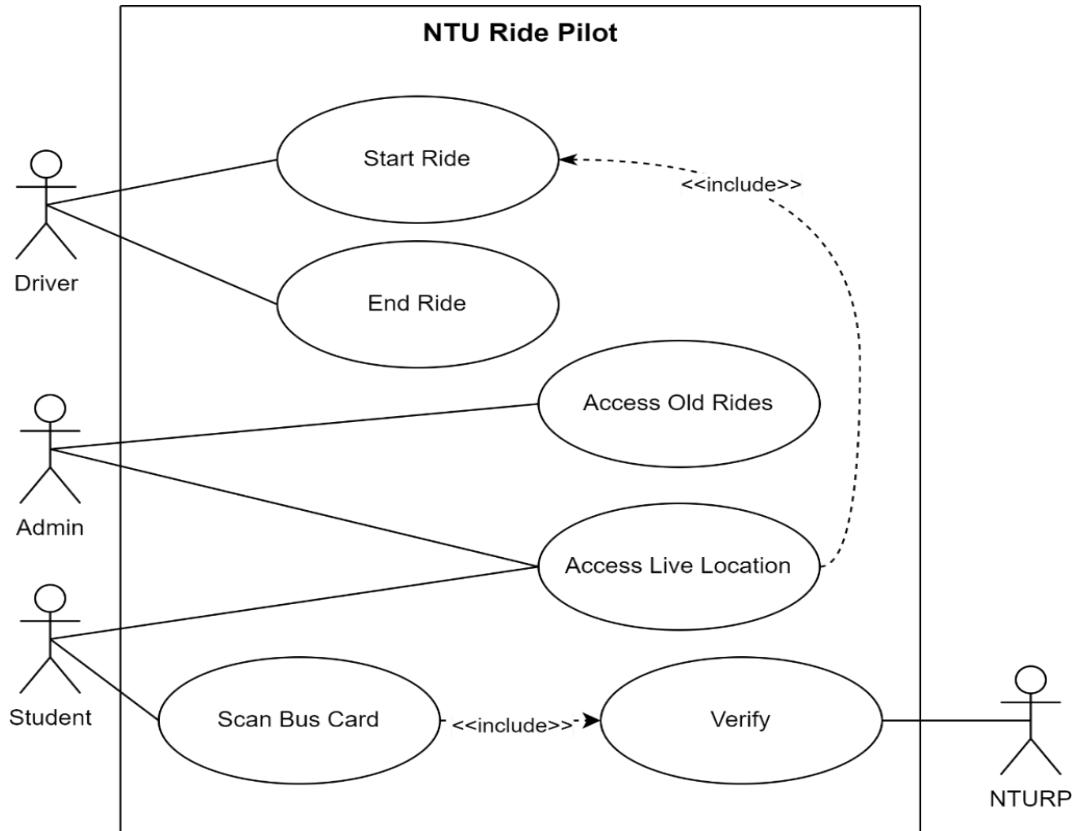


Figure 8 Use Case of Ride Management

3.3.8 Use Case of Bus Card Management

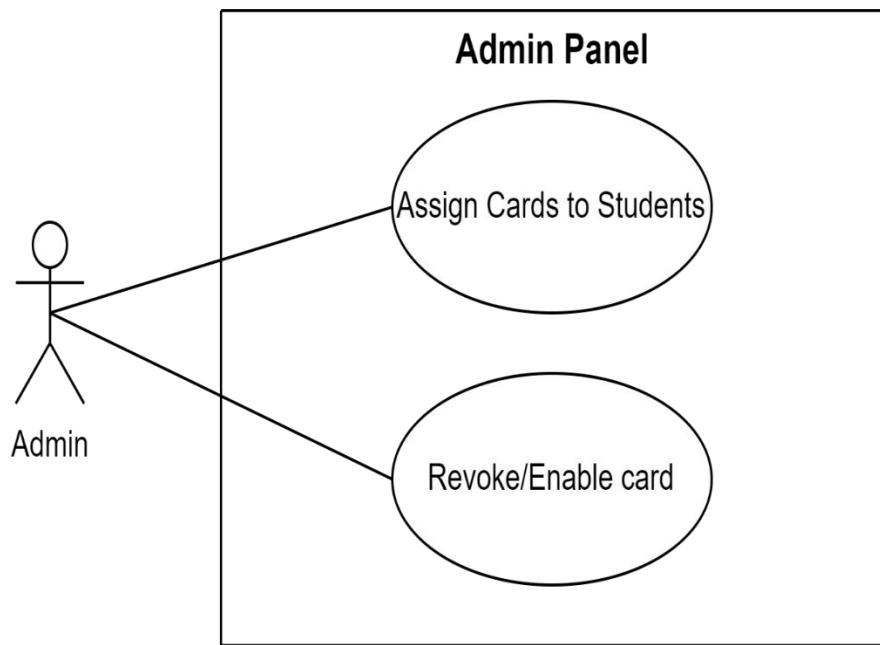


Figure 9 Use Case of Bus Card Management

3.3.9 Use Case of Student & Session Management

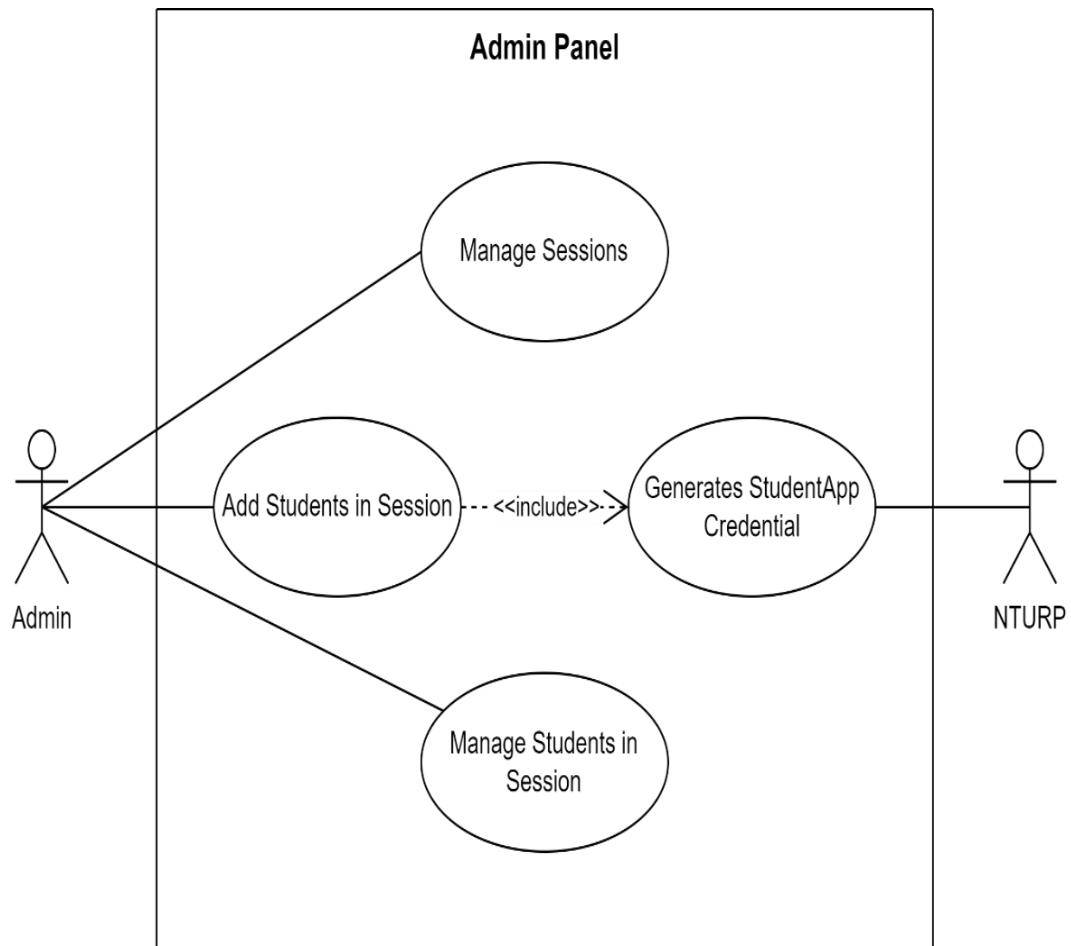


Figure 10 Use Case of Student & Session Management

3.3.10 Use Case of Bus Staff Management

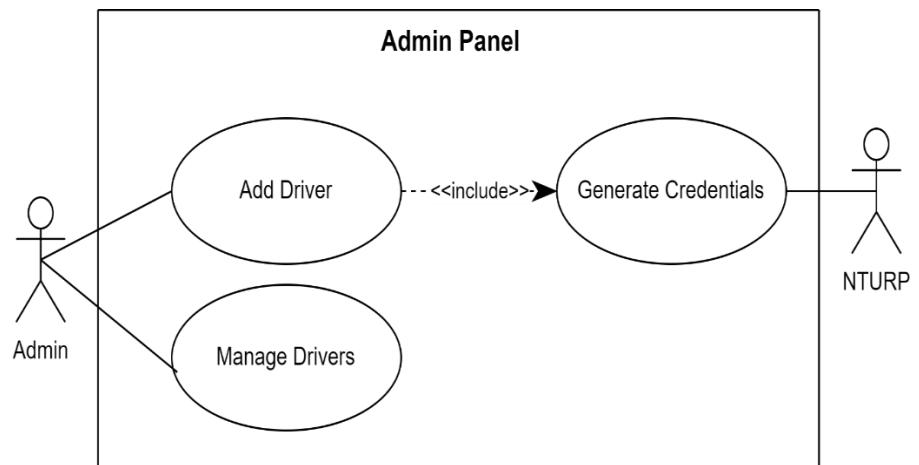


Figure 11 Use Case of Bus Staff Management

3.3.11 Use Case of Complaint Management

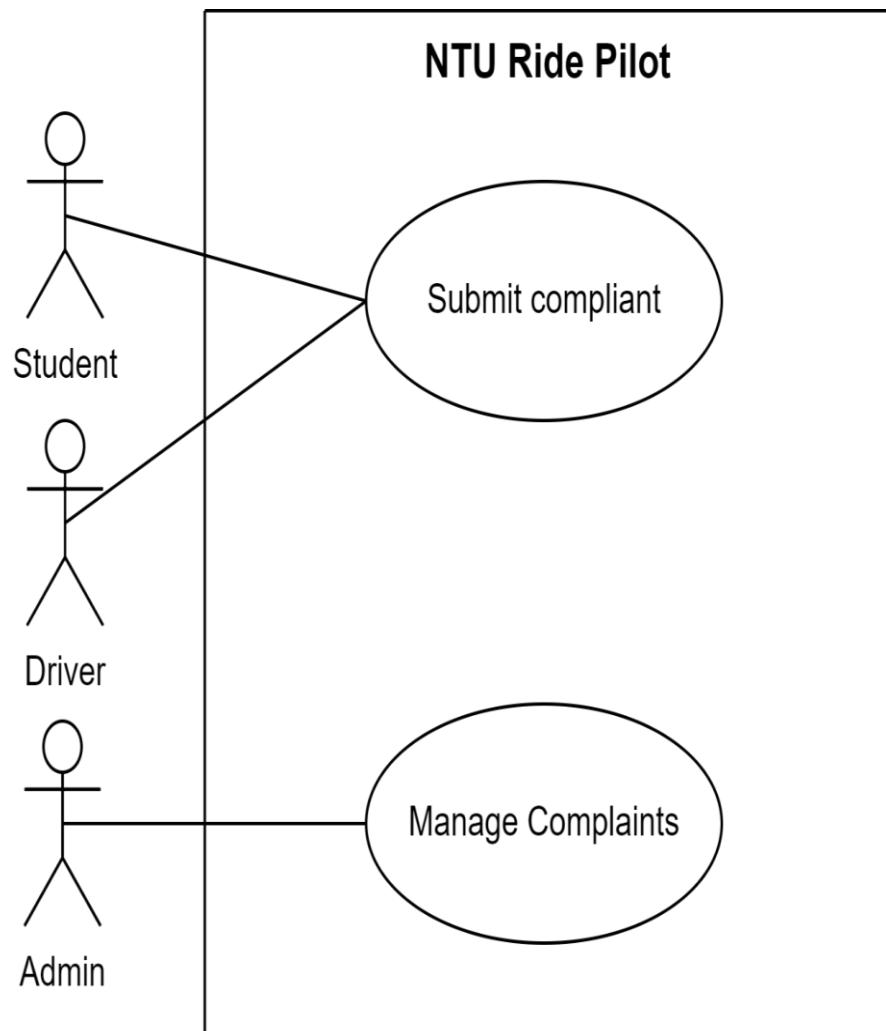


Figure 12 Use Case of Complaint Management

3.3.12 Use Case of Announcement Management

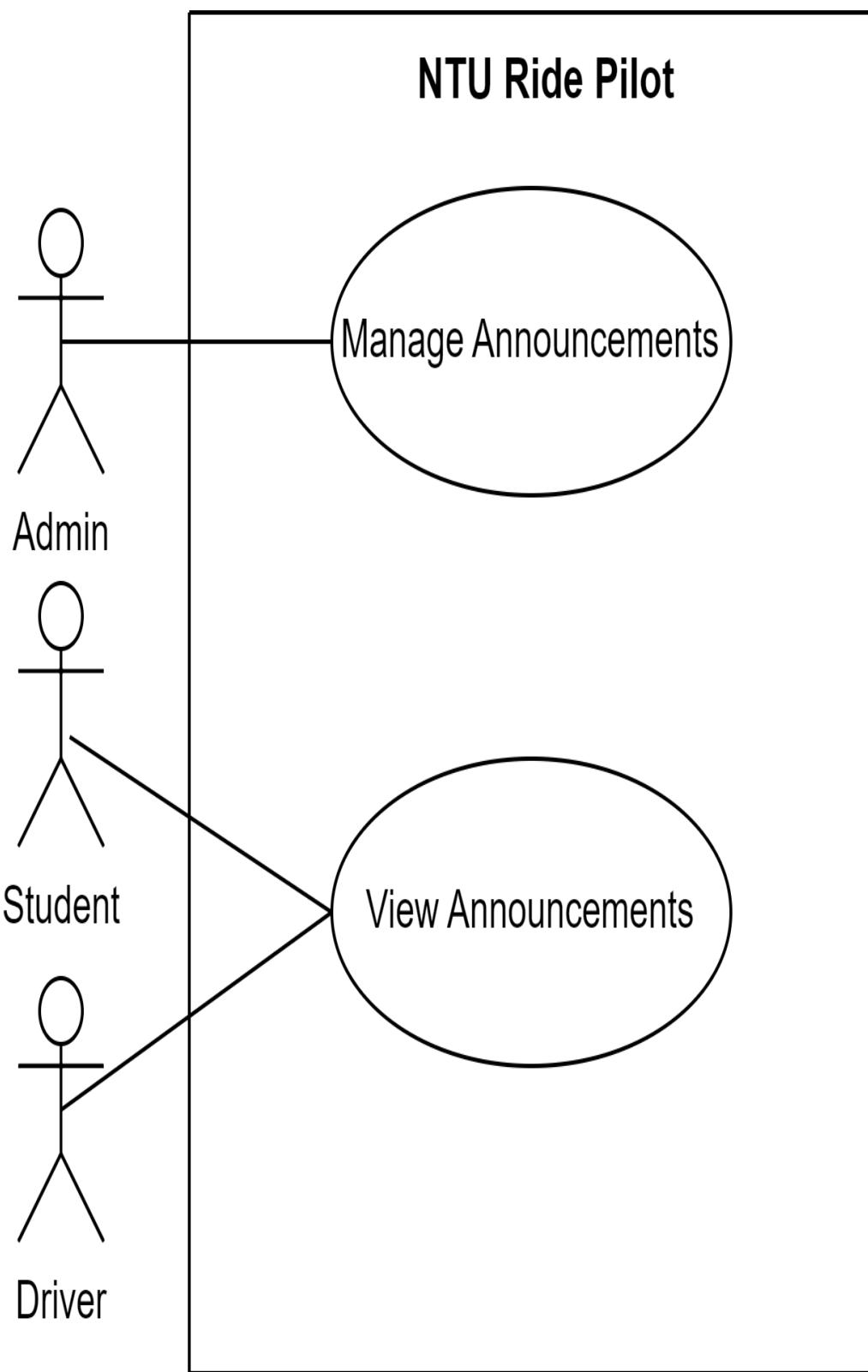


Figure 13 Use Case of Announcement Management

3.4 Use Case Description

Each use case involved in NTURP contains all necessary information including its use case id and name alongside description and pre- and post-conditions. A structured format within this description ensures both clarity and completeness during the definition of system interactions. The detailed documentation helps users understand both system behaviour and requirements from end-users.

3.4.1 Description of Sign In

Table 3 Description of Sign In

Field	Details
Use Case Id	01
Use Case Name	Sign In
Actors	Authorized User
Description	Authorized users log in to the NTURP via provided applications. Access is denied if the user does not have an account or proper credentials. This process ensures that only verified users can access the system, maintaining the security and integrity of NTURP services. The user must enter valid credentials through a secure login interface, which may include options for password recovery or multi-factor authentication in future implementations. Login attempts are logged for auditing and system monitoring.
Pre-condition	User must have an account and credentials.
Post-condition	User logs into the NTURP system or is denied access.

3.4.2 Description of Sign Up

Table 4 Description of Sign Up

Field	Details
Use Case Id	02
Use Case Name	Sign Up
Actors	Admin, Authorized User
Description	An Admin can assign specific roles and permissions to students and drivers. The admin is responsible for managing the creation of user accounts, including those for drivers and students.
Pre-condition	User must have an account, enough permissions and credentials to access system.
Post-condition	A new user is created with specific roles and permissions to operate with in NTURP.

3.4.3 Description of Bus and Route Management

Table 5 Description of Bus and Route Management

Field	Details
Use Case Id	03
Use Case Name	Bus and Route Management
Actors	Admin
Description	The admin is responsible for adding buses, defining routes, and assigning territories. This process includes registering new buses into the NTURP system by providing key details such as bus ID, capacity. The admin can define and update routes based on real-time demand, traffic conditions, or institutional requirements. Each route consists of predefined stops and schedules, ensuring optimal travel time and coverage. Additionally, the admin assigns buses to specific routes and territories, helping to maintain operational efficiency. The system allows for modifications such as reassigning buses, deactivating routes, or updating stops, ensuring flexibility and control over the transportation network.
Pre-condition	Admin must be authenticated and have enough permissions to manage buses and routes.
Post-condition	The admin effectively oversees the management of buses, routes, and territories.

3.4.4 Description of Ride Management

Table 6 Description of Ride Management

Field	Details
Use Case Id	04
Use Case Name	Ride Management
Actors	Admin, Driver, Student
Description	The driver starts and ends rides while verifying student bus cards during the trip. The system logs ride-related data and tracks the bus's live location, enabling administration, students, and parents to access ride details and monitor the bus's real-time location.
Pre-condition	The driver must be authenticated and have specified the route and the bus they are assigned to. The student must possess a valid and active bus card.
Post-condition	The driver successfully operates the Rides, while the system tracks and updates the ride details and live bus location. This allows both administrators and students/parents to access the ride information and the live location of the bus.

3.4.5 Description of Bus Card Management

Table 7 Description of Bus Card Management

Field	Details
Use Case Id	05
Use Case Name	Bus Card Management
Actors	Admin
Description	The admin is responsible for assigning and managing student bus cards. This includes generating new bus cards, linking them to registered student profiles, and ensuring each card is associated with the correct route and bus. The admin can monitor the usage history of each card to keep track of attendance, travel frequency, and route preferences. If a card is reported lost or misused, the admin can quickly revoke access to prevent unauthorized usage. Re-enabled cards can be reassigned seamlessly once the issue is resolved. The system provides clear logs of all card activities for transparency and future reference.
Pre-condition	The admin must be authenticated and have necessary permissions to manage bus cards. Students are required to be registered in the system.
Post-condition	The admin can successfully assign bus cards to students, revoke the cards to prevent further use, and re-enable them when needed.

3.4.6 Description of Session and Student Management

Table 8 Description of Session and Student Management

Field	Details
Use Case Id	06
Use Case Name	Session and Student Management
Actors	Admin, System
Description	The admin can create and terminate sessions, set session expiry, and manage students within the system. When a student is added, the system automatically generates app credentials for them. Expiring or deleting a session will deactivate all bus cards associated with that session. The admin can also update student records and manage session-wise student data for better organization. The system ensures that only active session students have access to transportation services, maintaining the integrity of student travel privileges. Admins can view session logs, track student activity, and generate reports to evaluate system usage.
Pre-condition	The admin must be authenticated and have necessary permissions to manage bus cards. Student data must be accurate and complete for the generation of credentials.
Post-condition	The admin effectively oversees both sessions and student management.

3.4.7 Description of Bus Staff Management

Table 9 Description of Bus Staff Management

Field	Details
Use Case Id	07
Use Case Name	Bus Staff Management
Actors	Admin, System
Description	The admin oversees drivers and conductors. When a new staff member is added, the system automatically generates app credentials for them.
Pre-condition	The admin must be authenticated and have necessary permissions to manage bus staff.
Post-condition	The admin effectively manages bus staff.

3.4.8 Description of Complaint Management

Table 10 Description of Complaint Management

Field	Details
Use Case Id	08
Use Case Name	Complaint Management
Actors	Student, Driver, Admin
Description	Students and drivers are allowed to submit complaints about transportation services and related concerns.
Pre-condition	All users are required to be authenticated. Administrators must also be authenticated and possess the necessary permissions to manage complaints.
Post-condition	Complaints submitted by students and drivers get reviewed and resolved by the authorities.

3.4.9 Description of Announcement Management

Table 11 Description of Announcement Management

Field	Details
Use Case Id	09
Use Case Name	Announcement Management
Actors	Admin, Driver, Student
Description	The admin shares announcements related to transportation.
Pre-condition	All users are required to be authenticated.
Post-condition	The administrator successfully posts announcements.

3.4.10 Description of General Functionalities

Table 12 Description of General Functionalities

Field	Details
Use Case Id	10
Use Case Name	General Functionalities
Actors	Users
Description	The admin can generate reports and analyse valuable information extracted from the data currently stored.
Pre-condition	The admin must be authenticated and have necessary permissions.
Post-condition	Administration makes better decisions and enhances services through the analysis of current transportation data.

Chapter 4

Methodology

4.1 Agile Software Development Methodology

Agile methodology is an iterative and incremental approach to software development that emphasizes flexibility, collaboration, and customer feedback. Unlike traditional methodologies like the Waterfall model, Agile allows for adaptive planning, evolutionary development, and continual improvement, enabling rapid and flexible responses to change. It is particularly effective in managing the complexity and unpredictability of software projects.

4.2 Selected Methodology: Agile

A software development methodology is a way to improve development work with the help of dividing the development process into distinct phases to make a system with better productivity. It also helps to structure and control the whole system. It involves different methodologies, also called the Software Development Life Cycle, that are stages for software development with a certain set of rules. Generically, we categorized the methodologies into Rapid application development and planned-driven. Waterfall, spiral is planned driven while agile is Rad based.

4.3 Reasons for Selecting Agile Methodology

- Agile allows the project to adapt to changes in requirements and technology swiftly.
- Regular feedback from users ensures that the development aligns with the user's needs and expectations.
- Agile facilitates the delivery of small, workable segments of the project, ensuring a faster time-to-market and continuous improvement.
- Regular reviews and iterations help in early identification and resolution of issues, reducing the overall risk.

4.4 Project Planning and Execution for NTURP

Agile project planning and execution involve the division of the project into sprints, with each sprint aimed at delivering a potentially shippable product increment. The key phases include:

- Gather and prioritize necessities for the NTURP undertaking, growing a product backlog.
- At the start of every dash, pick a hard and fast of capabilities from the product backlog and plan their delivery.
- Conduct day by day meetings to speak about development, demanding situations, and plan the day's paintings.
- Develop, take a look at, and combine features inside the dash.
- At the quilt of every sprint, reveal the finished paintings to stakeholders and collect comments.
- Reflect at the sprint to identify successes and regions for development.
- Plan releases based totally at the undertaking progress, stakeholder comments, and marketplace situations.

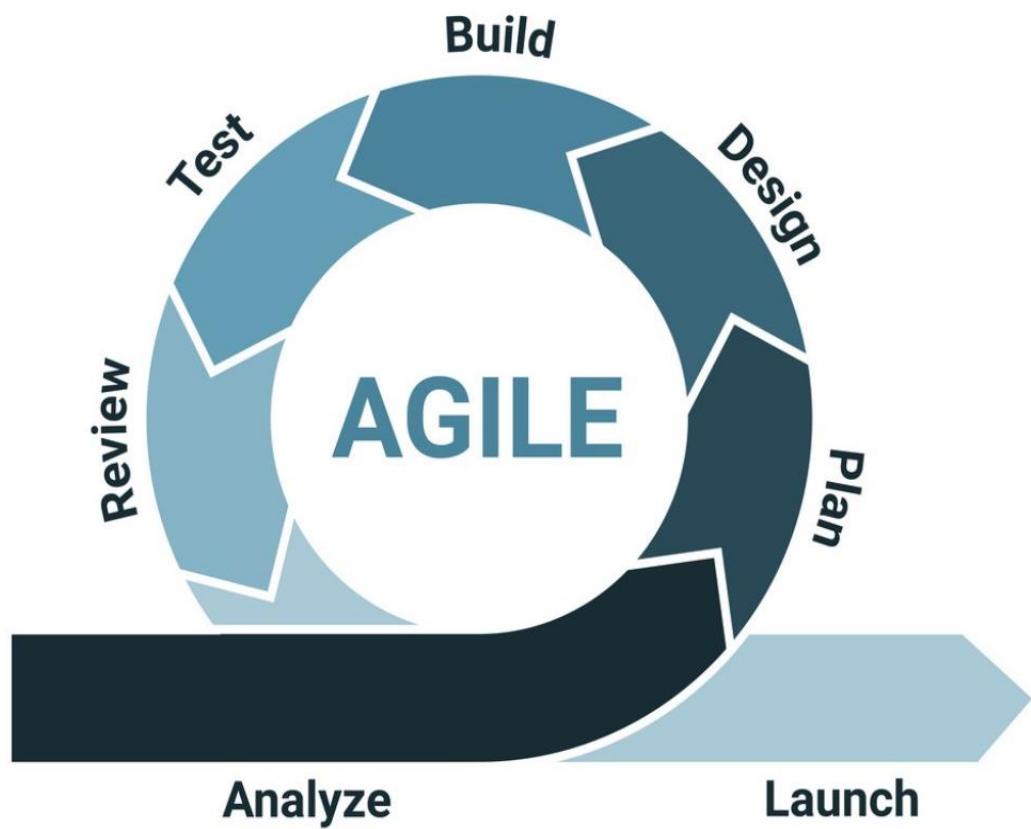


Figure 14 Agile Model

Chapter 5

Architecture Design

5.1 Activity Diagram

5.1.1 Mobile Application

This diagram illustrates the user flow for students and drivers after signing in to the mobile app. Students can view rides, profiles, announcements, and submit feedback, while drivers manage ride operations. It ensures a clear and role-specific interaction with system features based on authentication.



Figure 15 Activity diagram of mobile app

5.1.2 Admin Panel

The activity diagram for admin panel visualizes workflows for managing sessions, students, buses, drivers, and routes. After successful login, the admin can perform tasks like assigning cards, setting expiry, and handling feedback. It highlights the complete backend control flow in a structured and accessible manner.

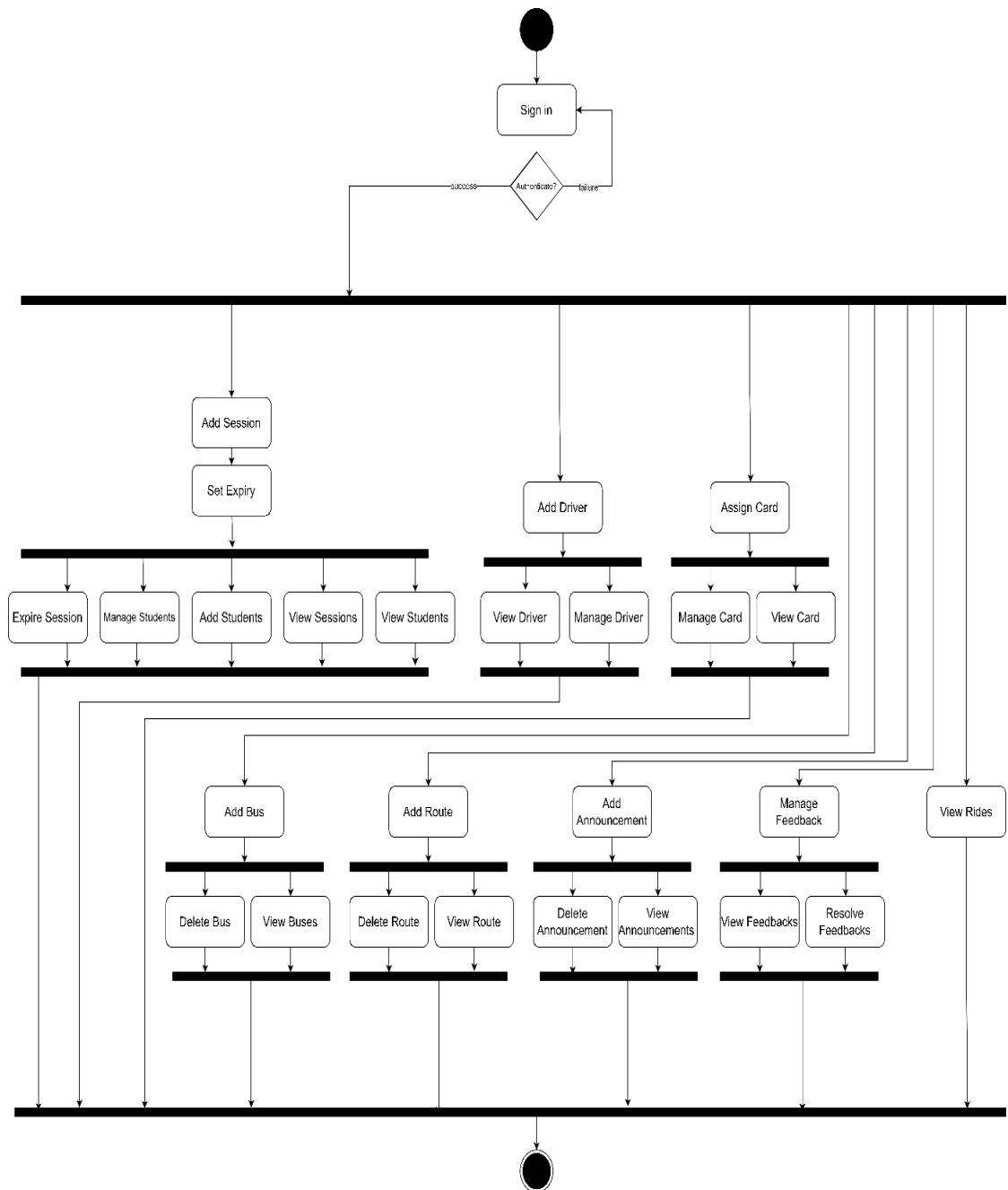


Figure 16 Activity diagram of Admin panel

5.2 Database Design

The Firestore database design for the NTU Ride Pilot system uses a collection-document model to support real-time updates and efficient data access. Top-level collections manage core system data, while the Users collection organizes role-specific access for drivers, admins, and students. This structure ensures secure, modular, and flexible data management.

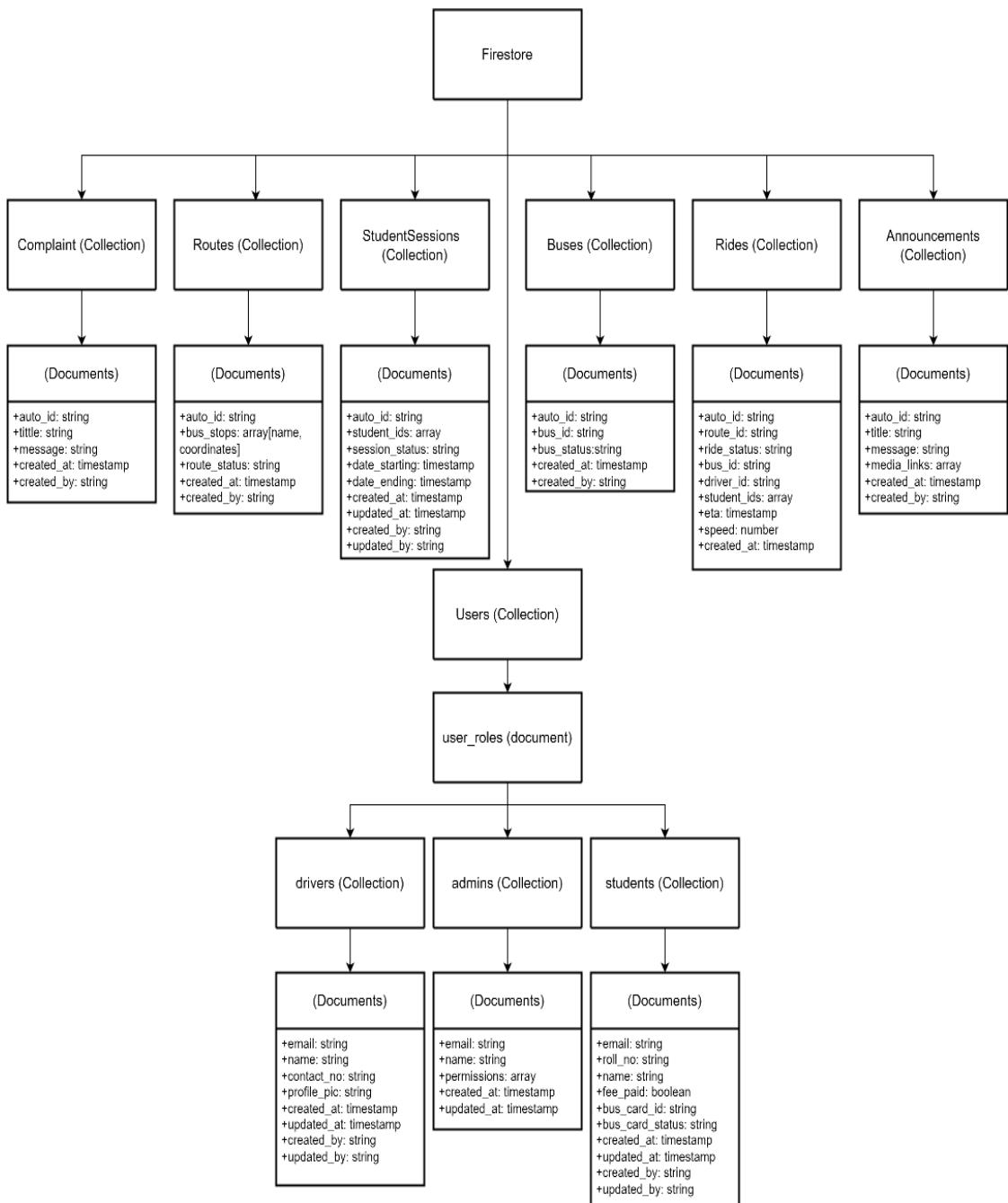


Figure 17 Database Design

5.3 Sequence Diagram

5.3.1 Admin Panel

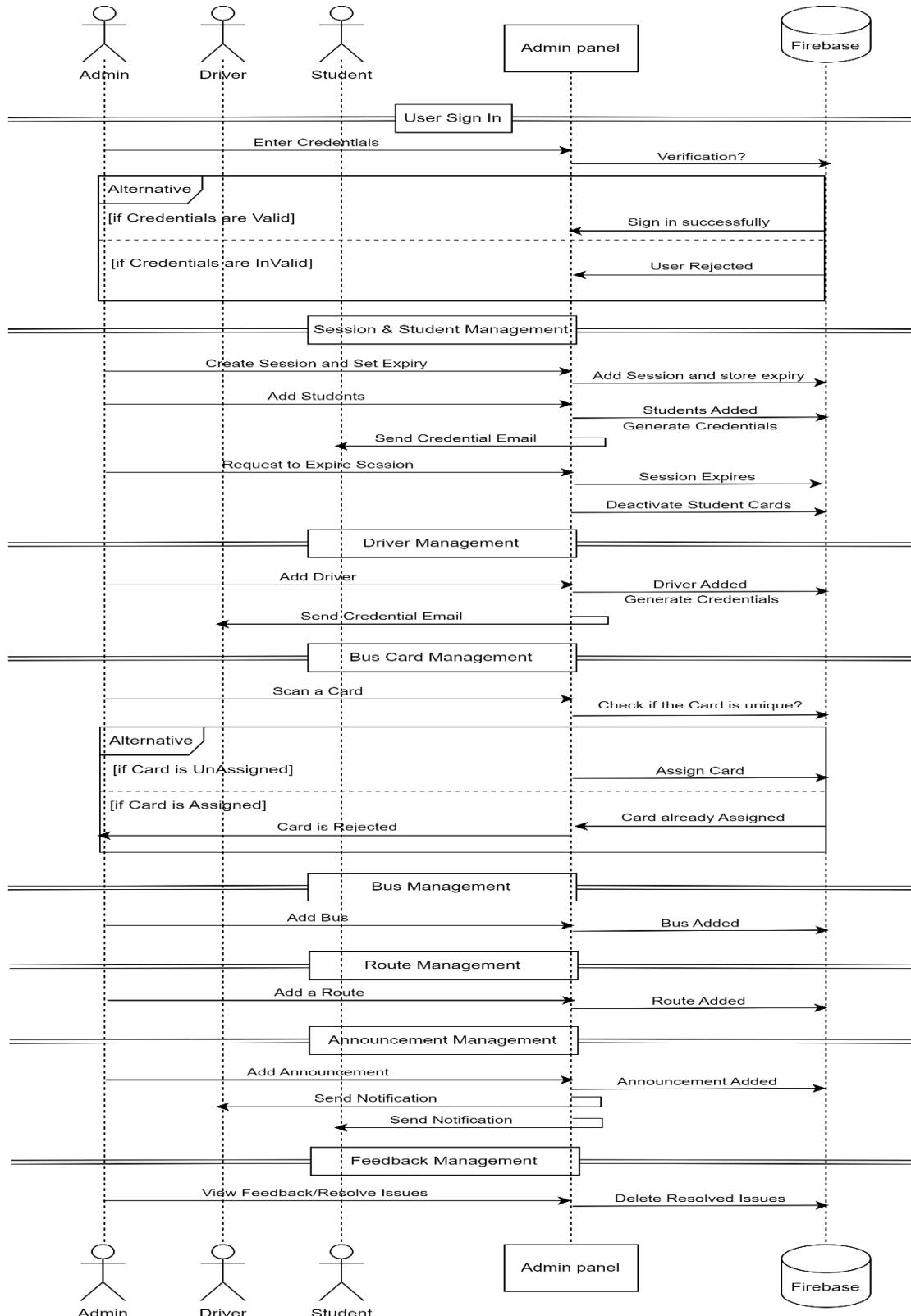


Figure 18 Sequence diagram of Admin Panel

5.3.2 Mobile App

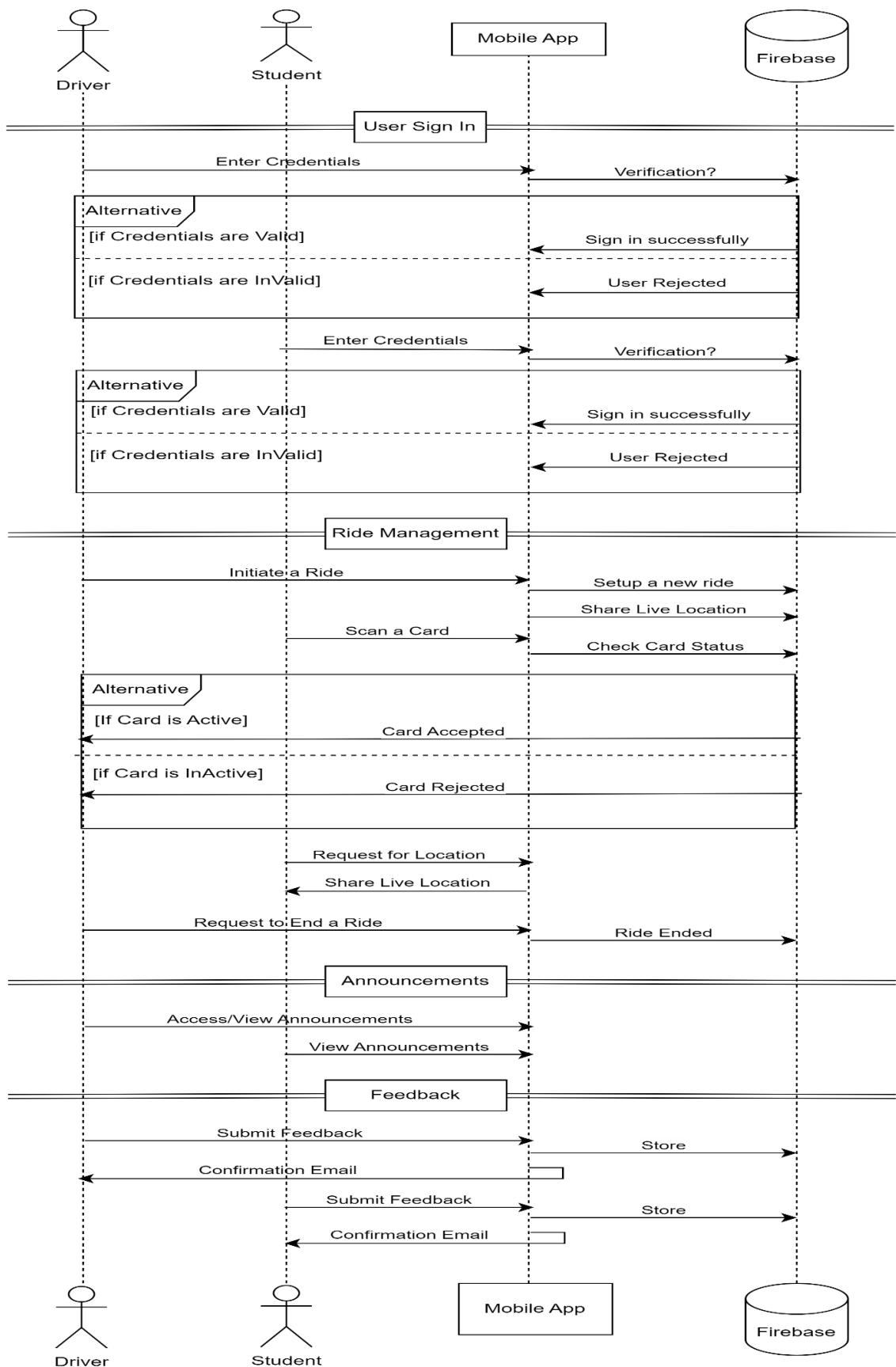


Figure 19 Sequence diagram of Mobile App

CHAPTER 6

System Implementation

After completing the design and planning phases, the system for **NTU Ride Pilot (NTURP)** was implemented using various modern tools and technologies. This chapter outlines the tools used during development, explains the class structure of the system, and provides an overview of the system's deployment architecture.

6.1 System Tools and Technologies

To ensure a responsive, scalable, and real-time system for students, drivers, and administrators, the following technologies were used:

- **VS Code:** A lightweight and powerful source-code editor used for writing and managing code for both the mobile app and admin dashboard.
- **Flutter:** Chosen for its cross-platform capabilities, Flutter enabled the development of a single mobile application for both students and drivers with native performance.
- **Firebase:** Provided backend services such as user authentication, real-time database access, cloud storage for ride/media data, and push notifications.
- **Map box:** Integrated for real-time GPS tracking, location mapping, and route visualization within both the mobile app and the admin dashboard.
- **Next.js:** Utilized to build the admin dashboard, enabling efficient server-side rendering, route management, and reactive UI for monitoring rides and managing users.
- **Cloud Fire store:** Served as the primary NoSQL database, storing structured data like student/driver details, rides, feedback, and announcements.
- **Firebase Cloud Messaging (FCM):** Used to deliver real-time alerts and notifications to users for updates such as new rides, announcements, or feedback responses.

6.2 Class Diagram

This section presents the complete class structure of the **NTU Ride Pilot (NTURP)** system, covering both the **Mobile Application** (for Students and Drivers) and the **Admin Panel** (for administrative control and system management).

The **Mobile App class diagram** outlines interactions between Student, Driver, Bus Card, Ride, Feedback, and Announcement entities, ensuring seamless communication and ride operations.

The **Admin Panel class diagram** captures administrative functionalities through classes like Admin, Bus Management, Route Management, Ride Management, Session Management, Feedback Management, and Announcement Management. These classes enable the admin to monitor users, manage rides, sessions, bus cards, and publish announcements, maintaining overall system integrity and control.

6.2.1 Mobile Application

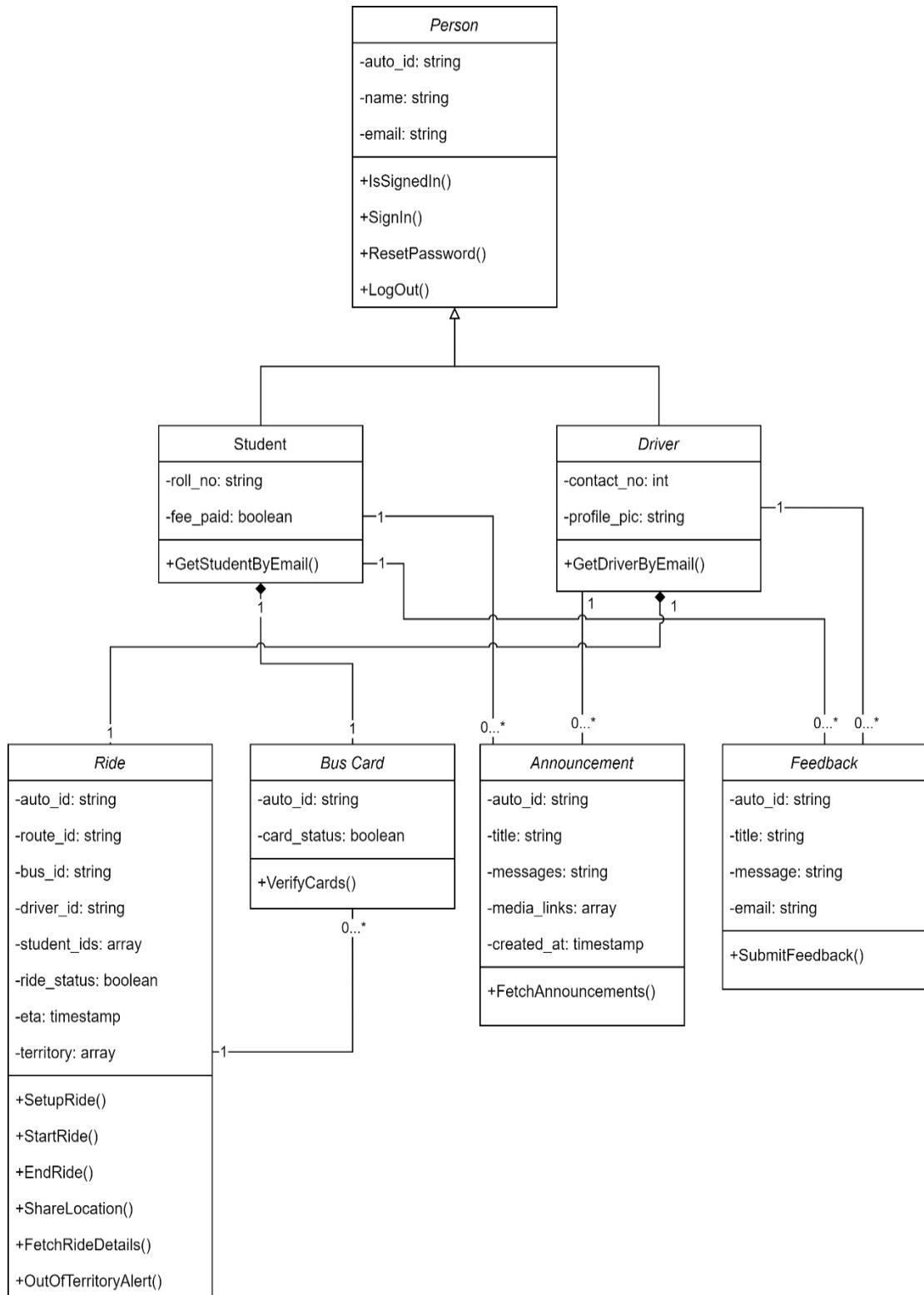


Figure 20 Class diagram of mobile app

6.2.2 Admin Panel

The class diagram for the NTU Ride Pilot system illustrates the key entities and their relationships within the system. It includes core classes such as Admin, Student, Driver, and Person, with associated modules like Bus Management, Route Management, and Ride Management. Inheritance is used where Student and Driver are specializations of the Person class. Multiplicity indicators define the cardinality of relationships between classes, ensuring data integrity and operational clarity. The diagram supports key functionalities such as session handling, announcements, feedback, and bus card tracking, providing a structured blueprint for system implementation.

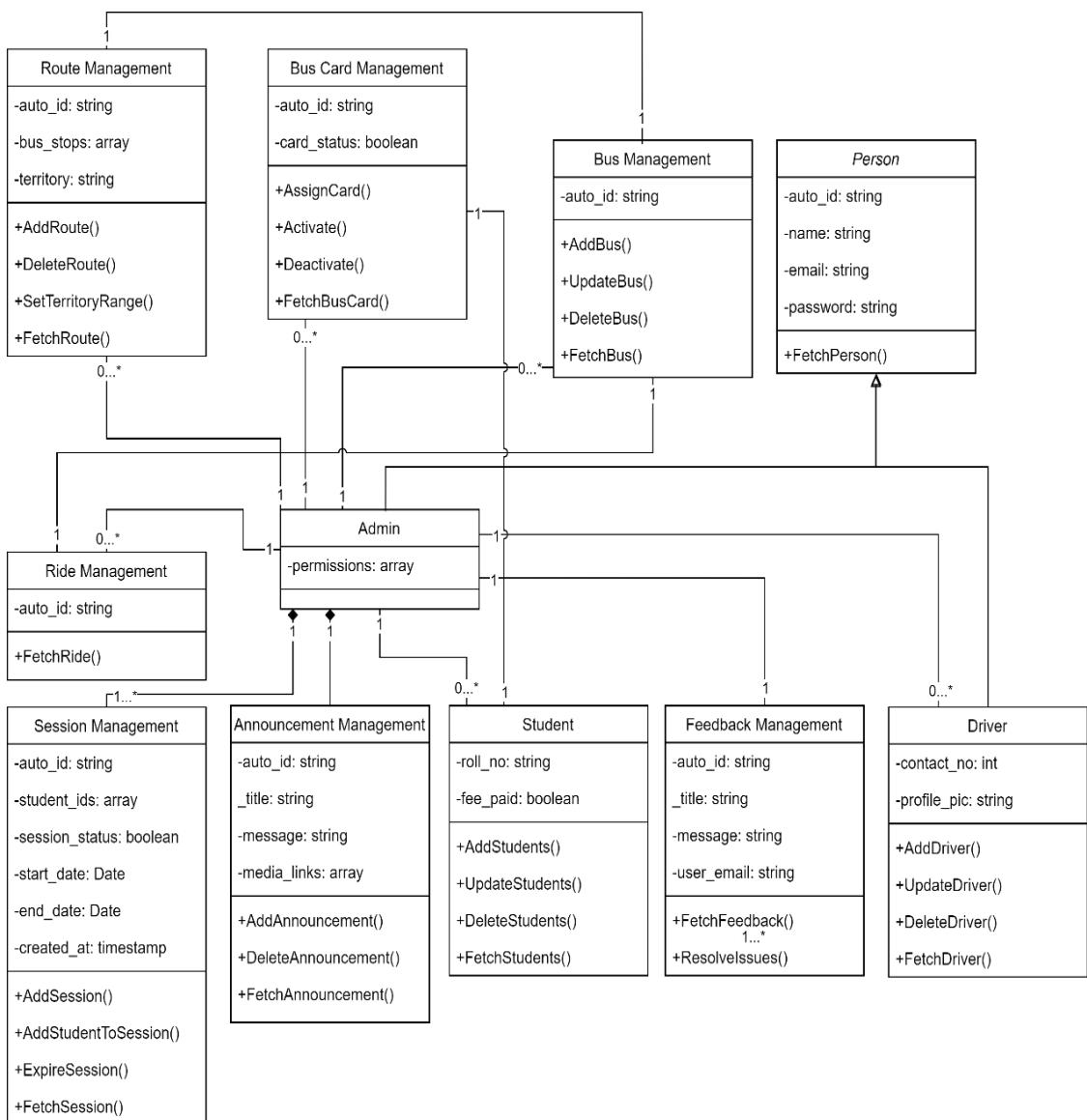


Figure 21 Class diagram of admin panel

6.3 Deployment Diagram

The deployment of the NTU Ride Pilot system follows a cloud-based architecture.

- The **Mobile Application** (for students and drivers) and the **Admin Dashboard** (built with Next.js) serve as client interfaces.
- These interfaces interact with **Firebase Services**, which include:
 - **Authentication** for secure login.
 - **Cloud Fire store** as the real-time NoSQL database.
 - **Cloud Storage** for media.
 - **Firebase Cloud Messaging** for real-time notifications.
- **Map box APIs** are integrated into both client applications for live location tracking and route mapping.

This server less architecture ensures high availability, real-time data sync, and scalability, making it ideal for a university-level transportation system.

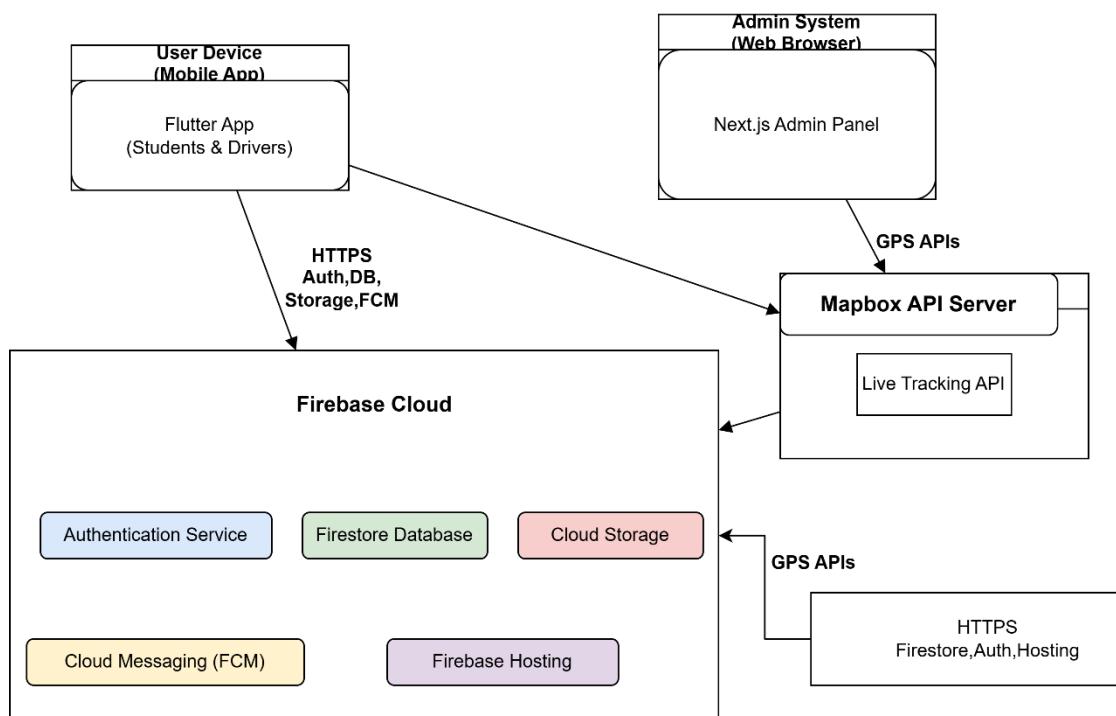


Figure 22 Deployment Diagram

CHAPTER 7

System Testing

Software testing plays a critical role in validating the system's correctness, functionality, and performance. In NTU Ride Pilot (NTURP), a combination of **White Box** and **Black Box** testing methodologies was employed to ensure both internal code integrity and end-user satisfaction. Testing was conducted across all components—Admin Panel, Student App, and Driver App—using real-time scenarios and multiple test cases.

7.1 White Box Testing

White Box Testing was used during the development phase to verify internal structures and logic. It ensured that the code worked as intended under all conditions. This included:

- **Unit Testing:** Testing individual components such as authentication, ride start/stop logic, and complaint submission.
- **Memory Leak Testing:** Ensuring Firebase listeners and subscriptions were properly disposed.
- **White Box Penetration Testing:** Checked unauthorized access routes through Admin endpoints.
- **Mutation Testing:** Altered logical conditions to ensure robustness against failures.

7.2 Black Box Testing

Black Box Testing validated the NTURP system from the user's perspective. This testing type evaluated outputs against inputs without requiring internal code knowledge.



Figure 23 Black Box Testing

Types applied:

- **Functional Testing:** Validated key modules like Login, Ride Management, Announcements, and Bus Card Handling.
- **Non-functional Testing:** Checked performance, UI responsiveness, and Firebase-based real-time data flow.
- **Regression Testing:** Ensured new updates did not break existing modules.

Other strategies used:

- **Scenario-Based Testing**
- **Equivalence Partitioning**
- **Boundary Value Testing**
- **Accessibility Testing**
- **State Transition Testing**

7.3 Verification

Verification was conducted to confirm that NTURP met all system specifications. The process included:

- **Inspection:** Manual review of feature implementations like map integration, announcement module, etc.
- **Demonstration:** Live demo of Ride flow, Card Validation, and Real-Time Tracking.
- **Testing:** Module-level and integration testing.
- **Analysis:** Logs and Firebase backend analysis to confirm correct request/response cycles.

7.4 Validation

Validation confirmed that the system met **user needs** and **project goals**. Testers, end-users, and supervisors verified:

- Real-time bus tracking accuracy via Map box.
- Functionality of push notifications via Firebase Cloud Messaging.
- Successful RFID card scans during active rides.
- Accurate ride history and complaint resolution records.

7.5 Adopted Methodology

7.5.1 Unit Testing

Each functionality (e.g., Firebase login, GPS streaming) was tested independently during development.

7.5.2 Module Testing

Modules such as Complaint Handling and Card Management were tested independently and in tandem.

7.5.3 Integration Testing

All modules were integrated and tested for interaction. For example, card validation triggering ride logging, and live location updates syncing with the Admin Panel.

7.5.4 System Testing

End-to-end testing of Student App, Driver App, and Admin Panel was performed as a whole.

7.5.5 Acceptance Testing

Stakeholders (supervisors and testers) tested the deployed system to confirm expectations were met prior to final submission.

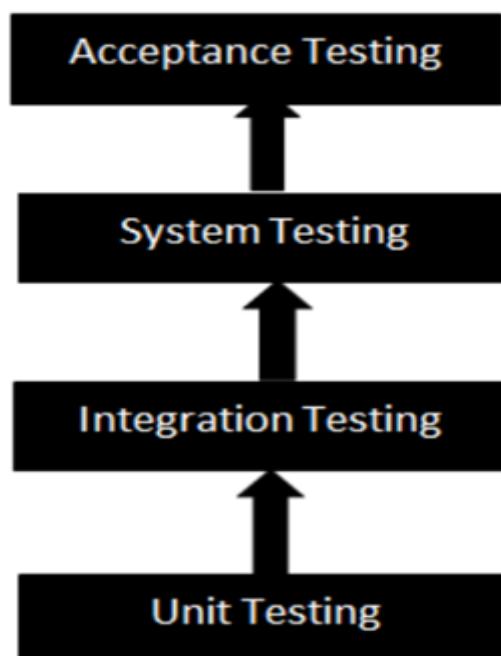


Figure 24 Adopted Methodology

7.6 Test Cases

Test Case 1: Ride Start/Stop:

Table 13 Ride Start/Stop

Component Name	Ride Management
Module Name	Driver App
Condition being tested	Start and End Ride via App
Expected Result	Ride starts/stops and logs correctly
Success Scenarios	Live location updates; ride history logs created
Failure Scenarios	Ride doesn't start; GPS not fetched
Test Result (Pass/ Fail)	Pass

Test Case 2: RFID Card Validation:

Table 14 RFID Card Validation

Component Name	Card Validator
Module Name	Driver App
Condition being tested	Scan valid/invalid card
Expected Result	Valid cards accepted, invalid rejected
Success Scenarios	Student verified and logged
Failure Scenarios	Card scan fails or grants unauthorized access
Test Result (Pass/ Fail)	Pass

Test Case 3: Complaint Submission:*Table 15 Complaint Submission*

Component Name	Complaint Form
Module Name	Student App
Condition being tested	Submit complaint
Expected Result	Complaint saved and sent to admin
Success Scenarios	Admin sees and responds
Failure Scenarios	Complaint fails to submit
Test Result (Pass/ Fail)	Pass

Test Case 4: Announcements:*Table 16 Announcements*

Component Name	Announcement Module
Module Name	Admin Panel
Condition being tested	Create and send announcements
Expected Result	Users receive push notifications
Success Scenarios	Messages delivered via FCM
Failure Scenarios	No notification received
Test Result (Pass/ Fail)	Pass

Test Case 5: Live Location Tracking:

Table 17 Live Location Tracking

Component Name	Map box Integration
Module Name	Student App / Admin Panel
Condition being tested	Display real-time bus location
Expected Result	Map updates continuously
Success Scenarios	Location pins move live
Failure Scenarios	Static or broken tracking
Test Result (Pass/ Fail)	Pass

Test Case 6: Session Expiry Handling:

Table 18 Session Expiry Handling

Component Name	Session Management
Module Name	Admin Panel
Condition being tested	Automatic deactivation of student cards upon session expiry
Expected Result	Expired sessions should revoke student card access
Success Scenarios	Cards are automatically disabled after session expiry; system logs the change
Failure Scenarios	Card remains active after session expiry; unauthorized ride access occurs
Test Result (Pass/ Fail)	Pass

CHAPTER 8

Application Prototype

The NTU Ride Pilot (NTURP) application prototype is a functional implementation of the proposed system, representing the final integration of all components into a usable product. It includes a **cross-platform mobile app** for students and drivers and a **web-based admin panel** for managing the transportation system. This chapter showcases the core modules and user interface elements implemented in both systems.

8.1 Mobile Application Prototype (Flutter)

The mobile application is developed using **Flutter** with **Firebase** as backend, featuring **role-based interfaces** for both **students** and **drivers**. It supports secure login, live tracking, ride verification, complaints, and real-time alerts. Both **light and dark modes** are supported for all modules, enhancing accessibility and UI consistency.

8.1.1 Splash & Login Screens

8.1.1.1 Splash Screen:

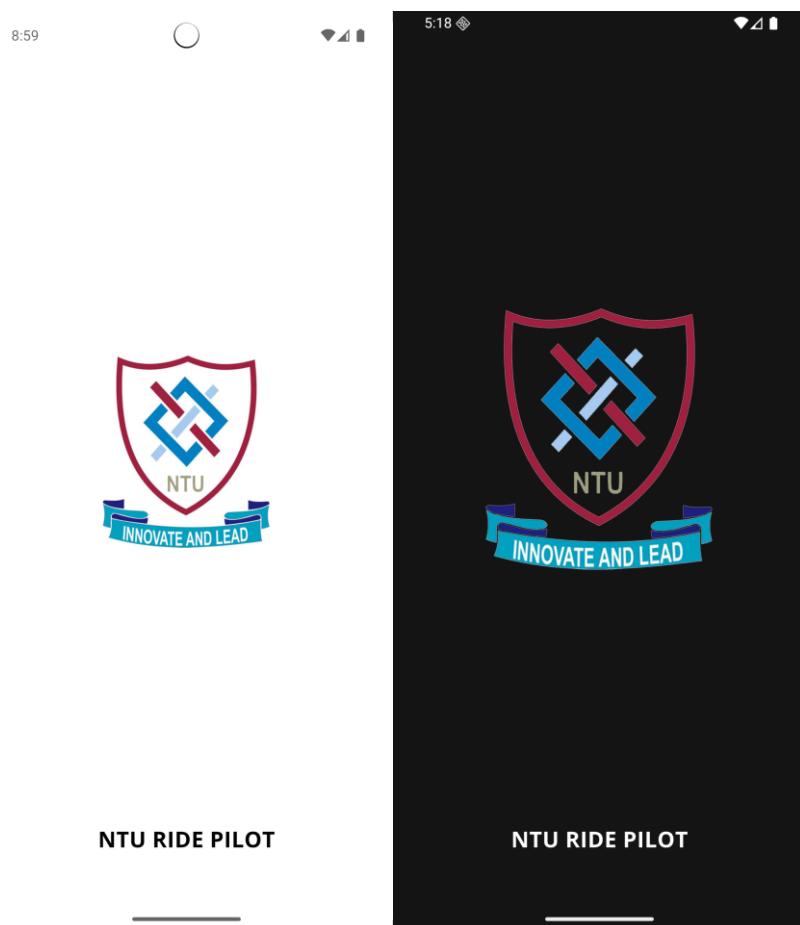


Figure 25 Splash Screen

8.1.1.2 Login Screen (Student):

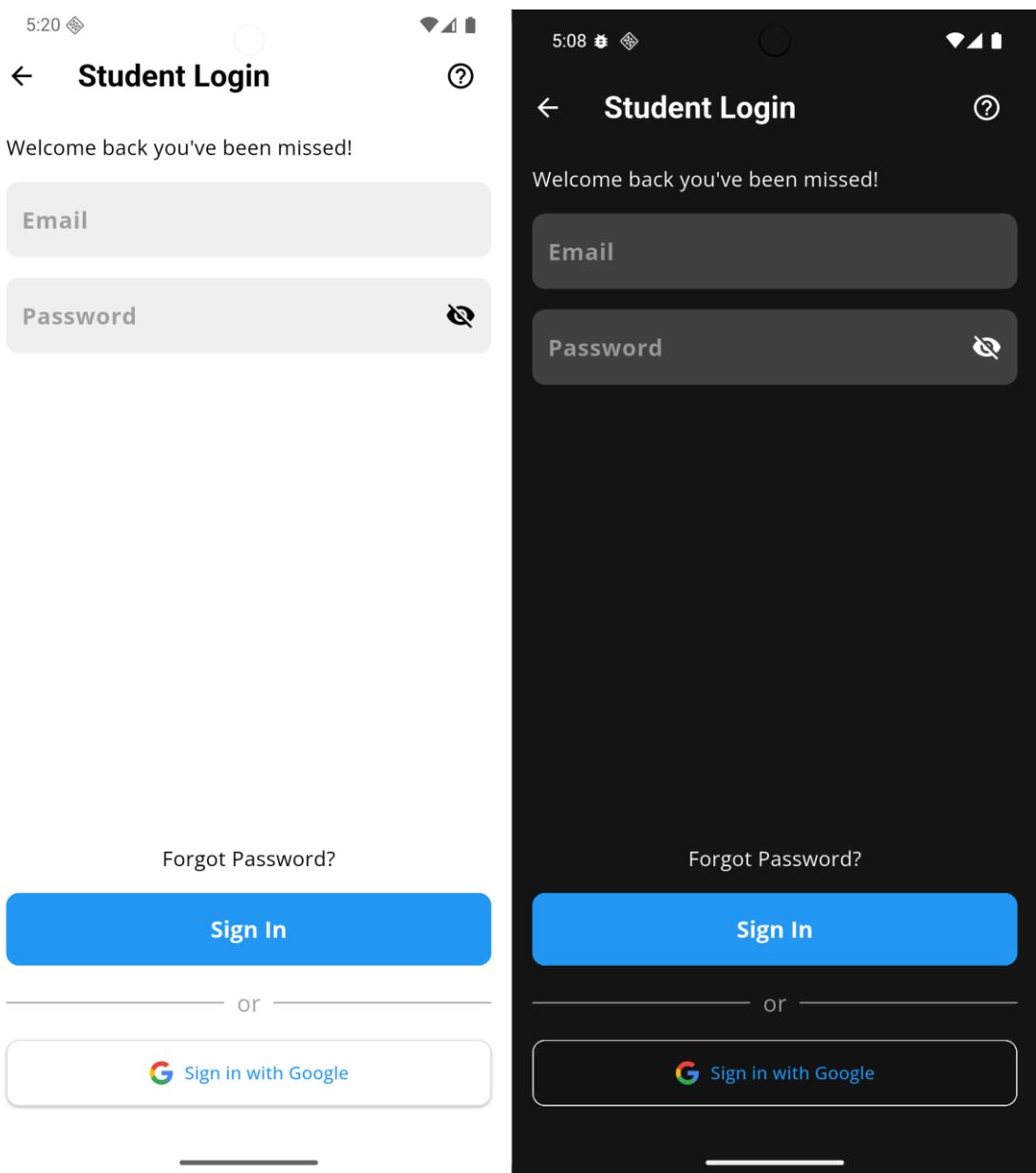


Figure 26 Student Login Screen

8.1.1.3 Login Screen (Driver):

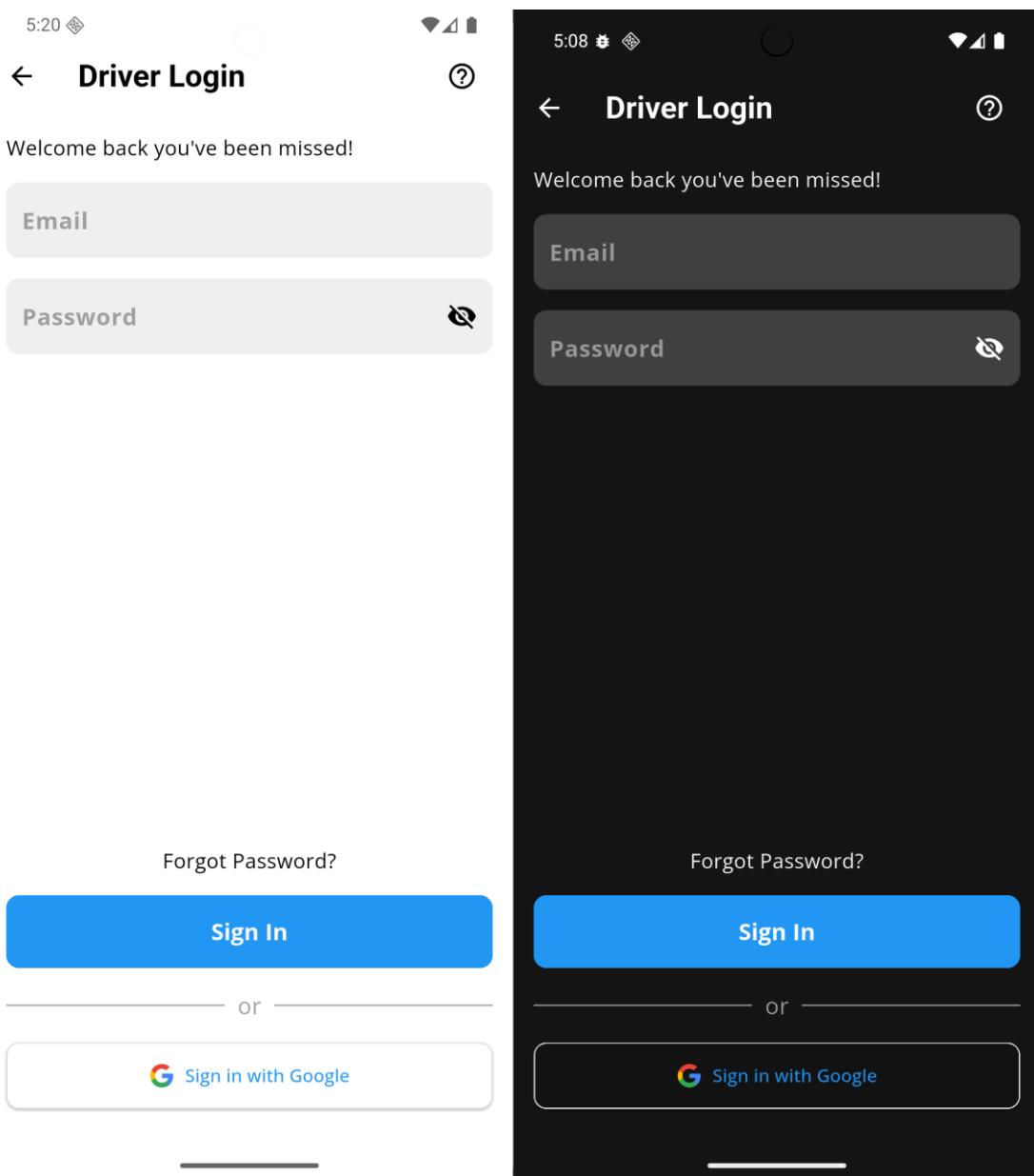


Figure 27 Driver Login Screen

8.1.2 Role-Based Navigation & Welcome Screens

8.1.2.1 Welcome Screen (Student):

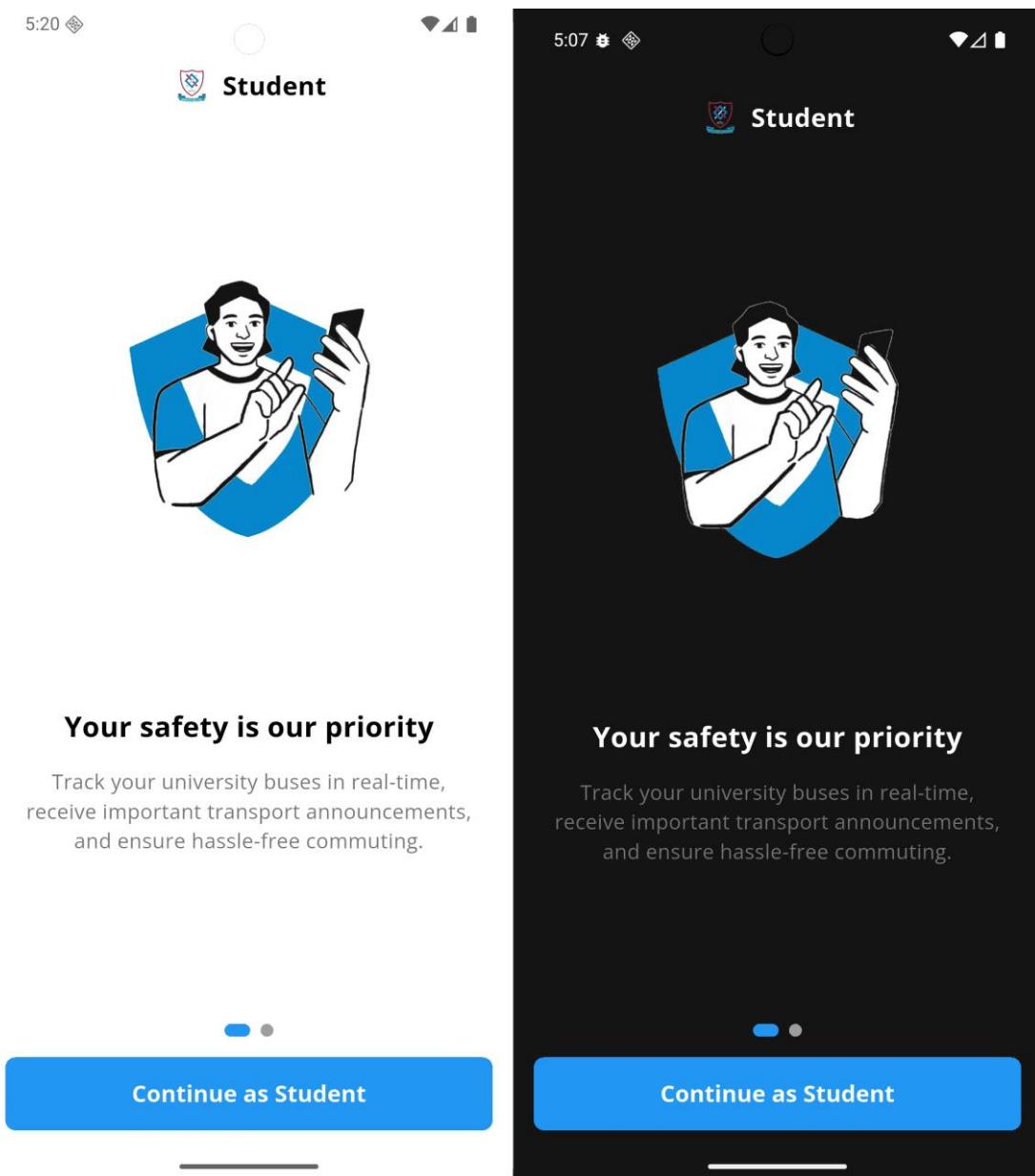


Figure 28 Student Welcome Screen

8.1.2.2 Welcome Screen (Driver):

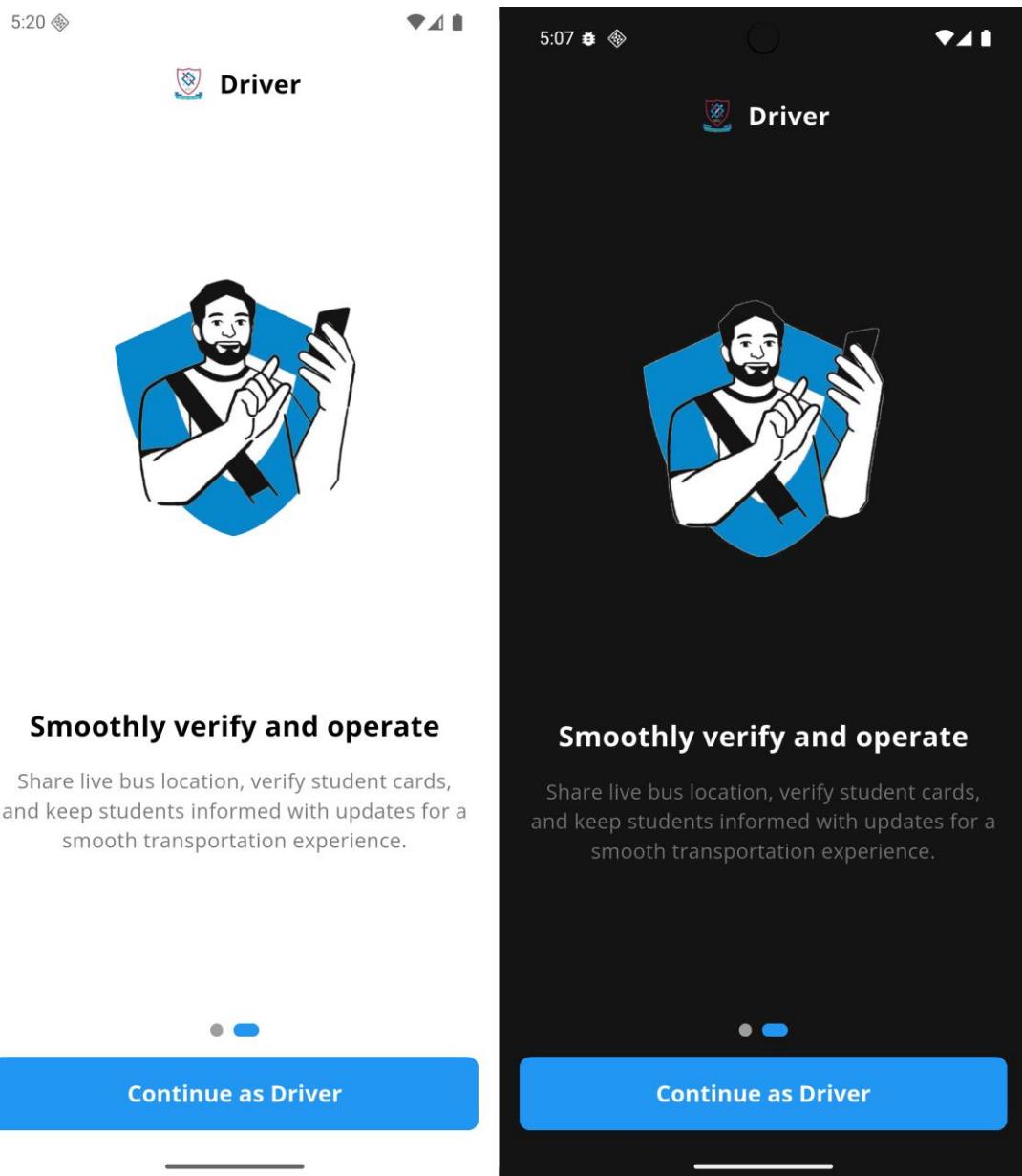


Figure 29 Driver Welcome Screen

8.1.2.3 Drawer (Student):

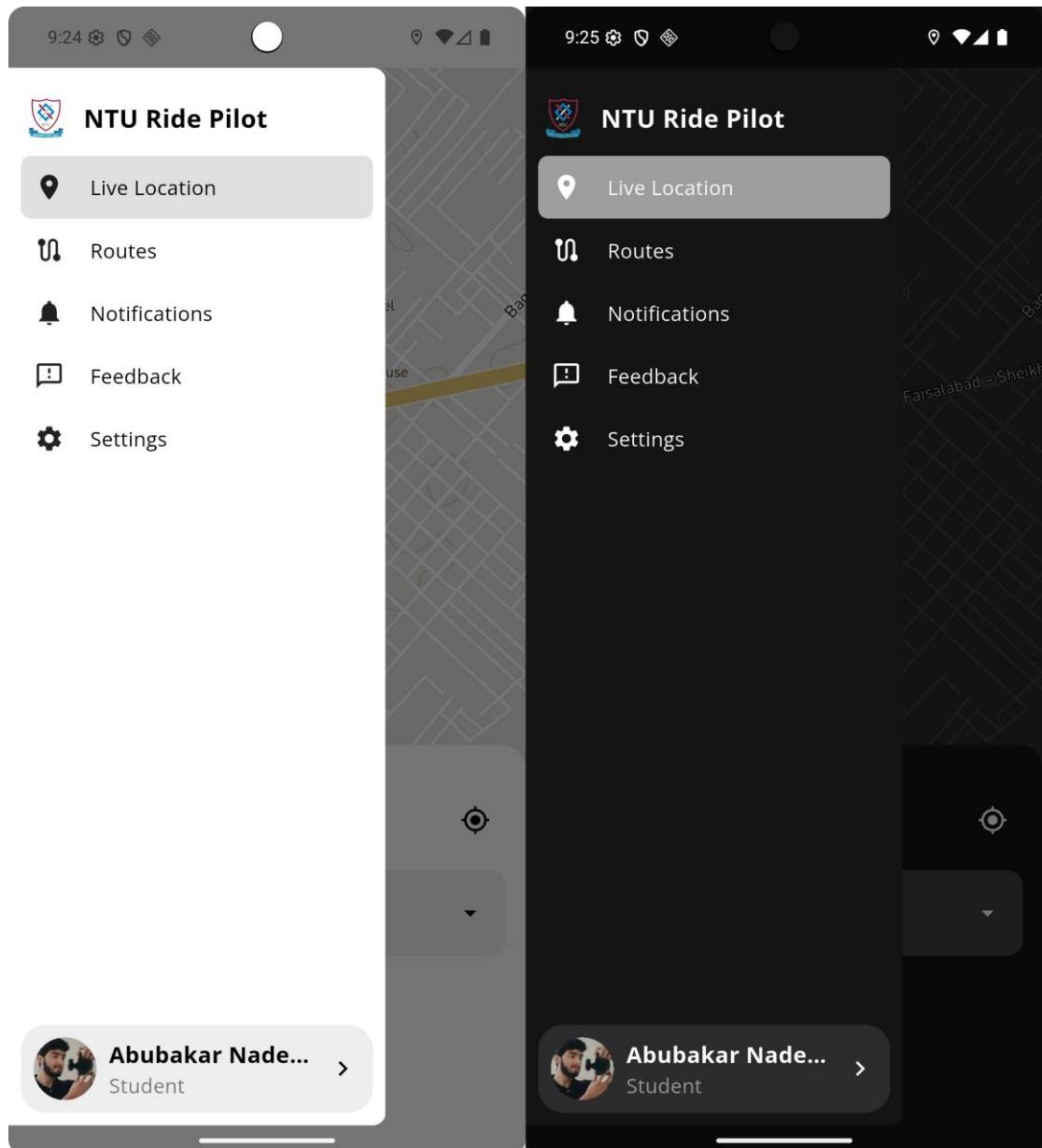


Figure 30 Student Drawer

8.1.2.4 Drawer (Driver):

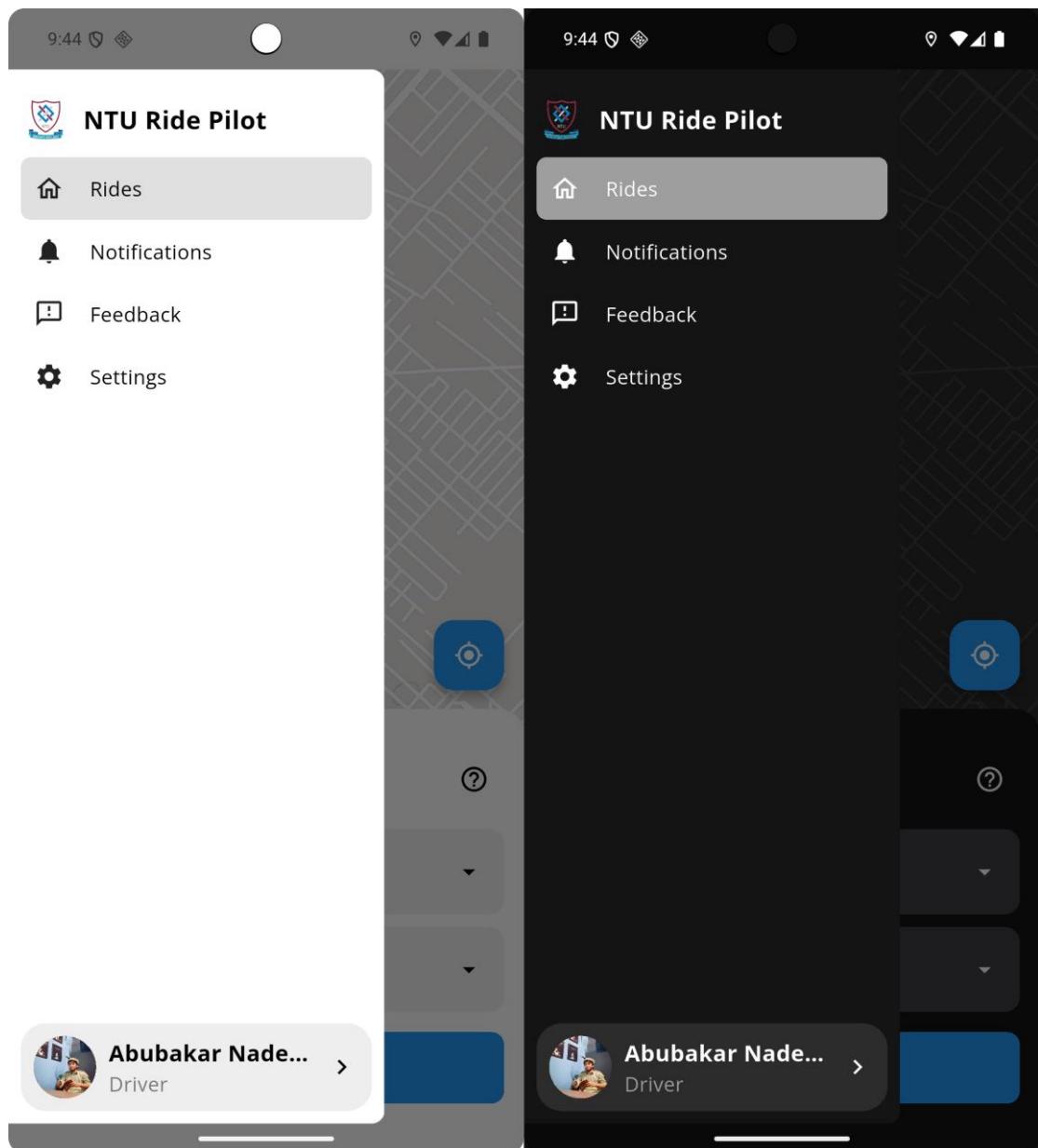


Figure 31 Student Drawer

8.1.3 Student Ride & Route Info Screens

8.1.3.1 Live Ride Tracking Screen (Student):

1.

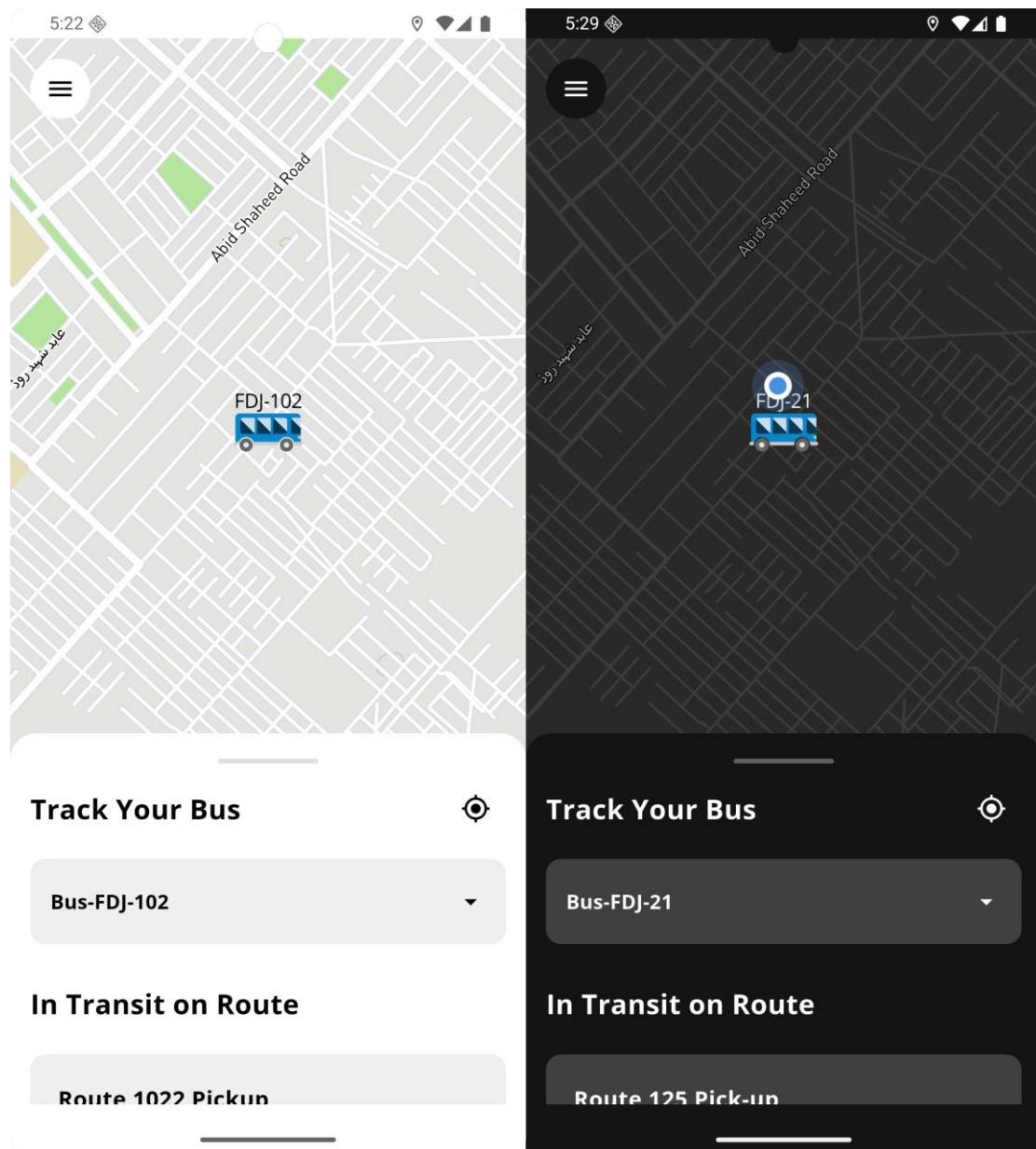


Figure 32 Live Ride Tracking Screen

2.

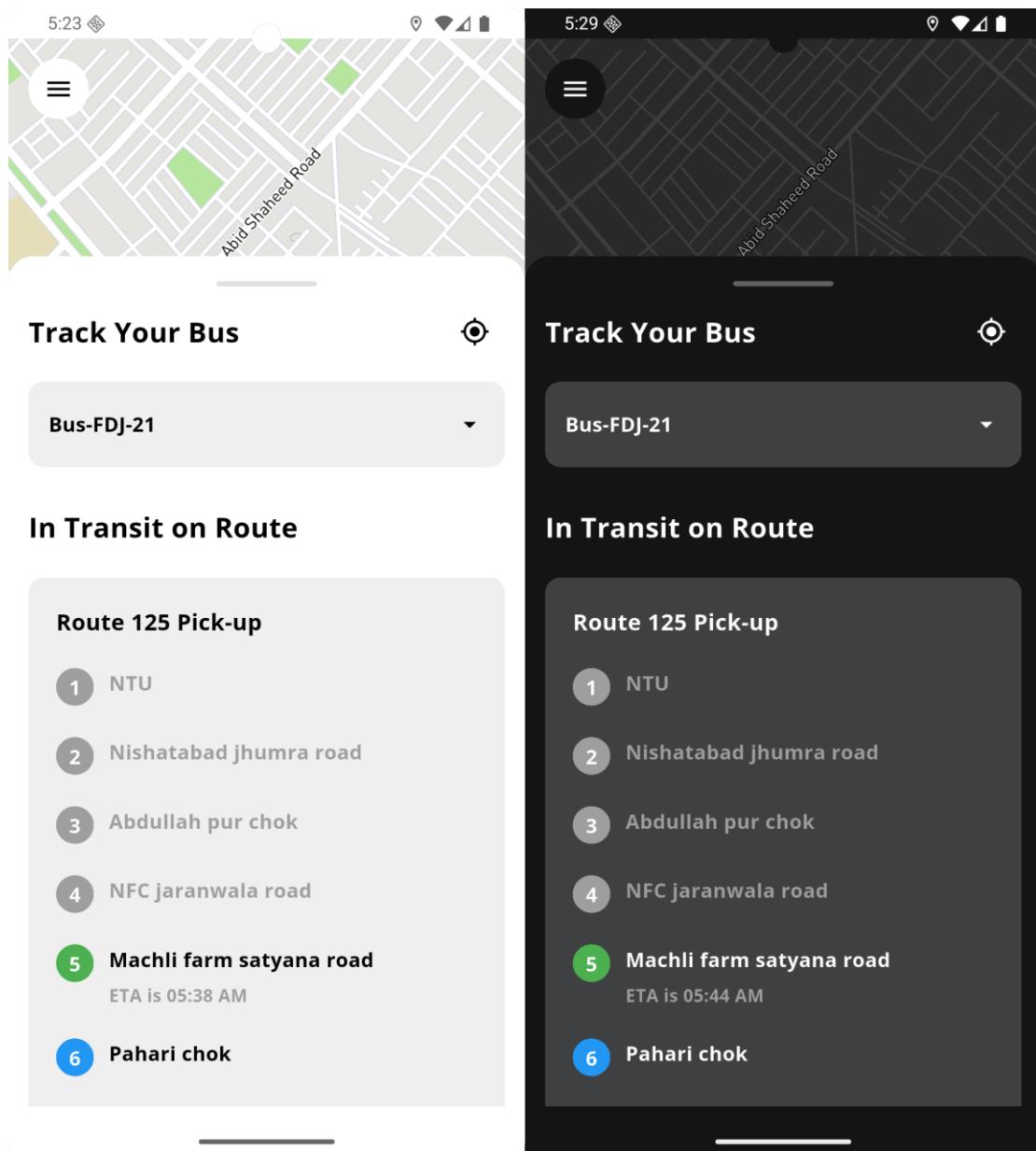


Figure 33 Live Ride Tracking Screen Detailed

8.1.3.2 View Default Routes Screen (Student):

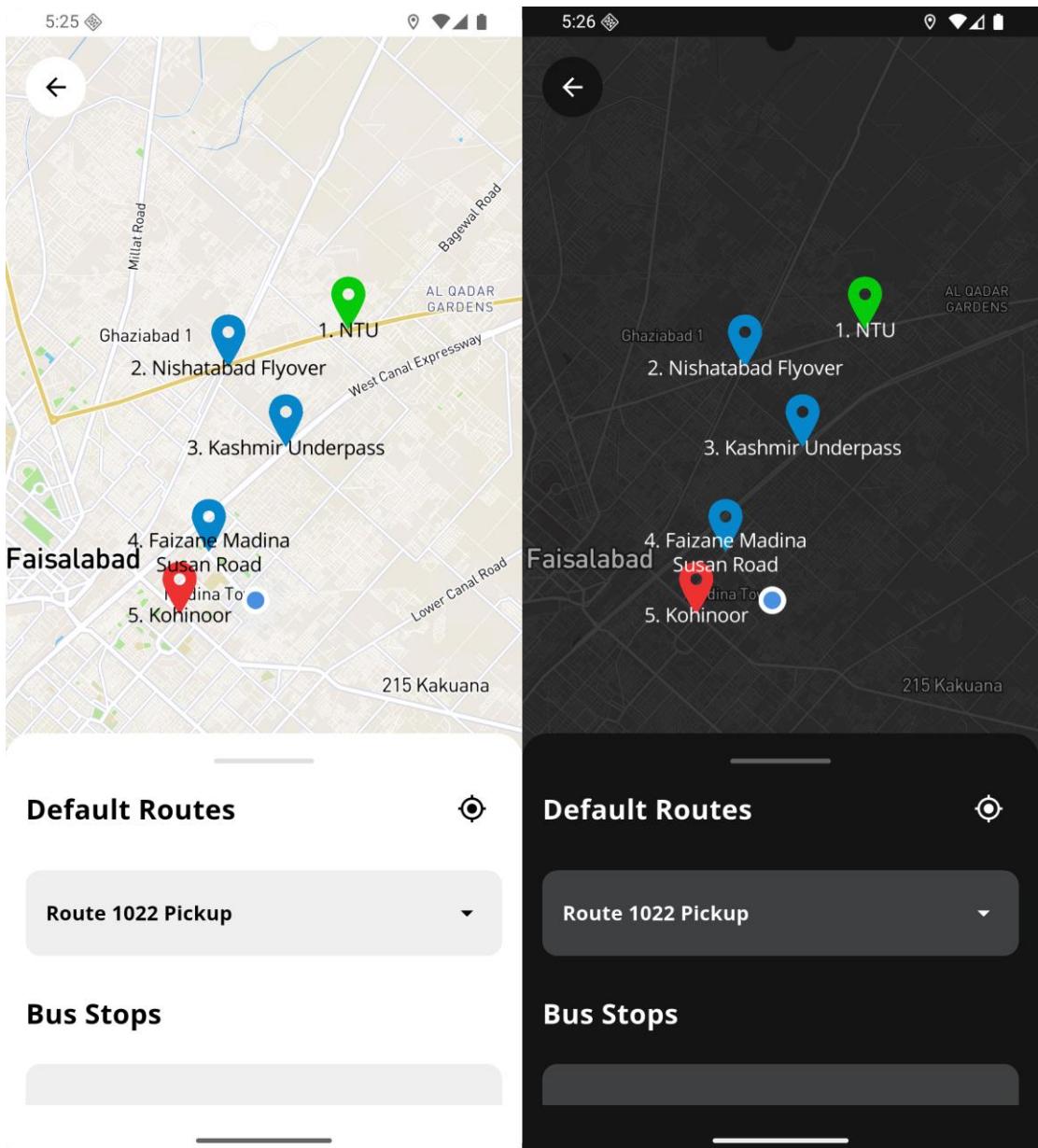


Figure 34 Default Routes Screen

8.1.4 Driver Ride Selection & Ride Control Screens

8.1.4.1 Ride Selection Screen (Driver):

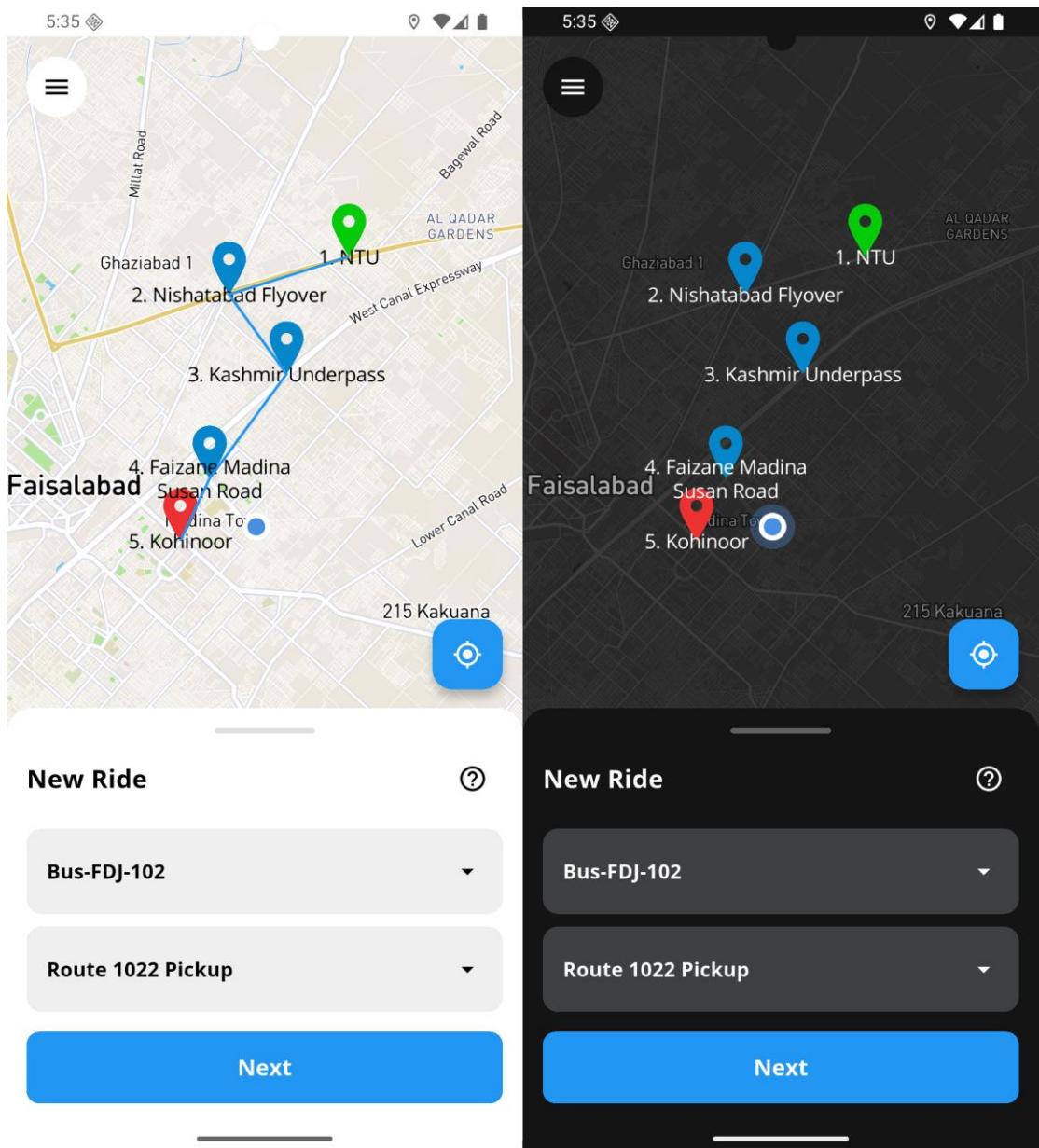


Figure 35 Ride Selection Screen

8.1.4.2 Ride Monitoring & Controls Screen (Driver):

1.

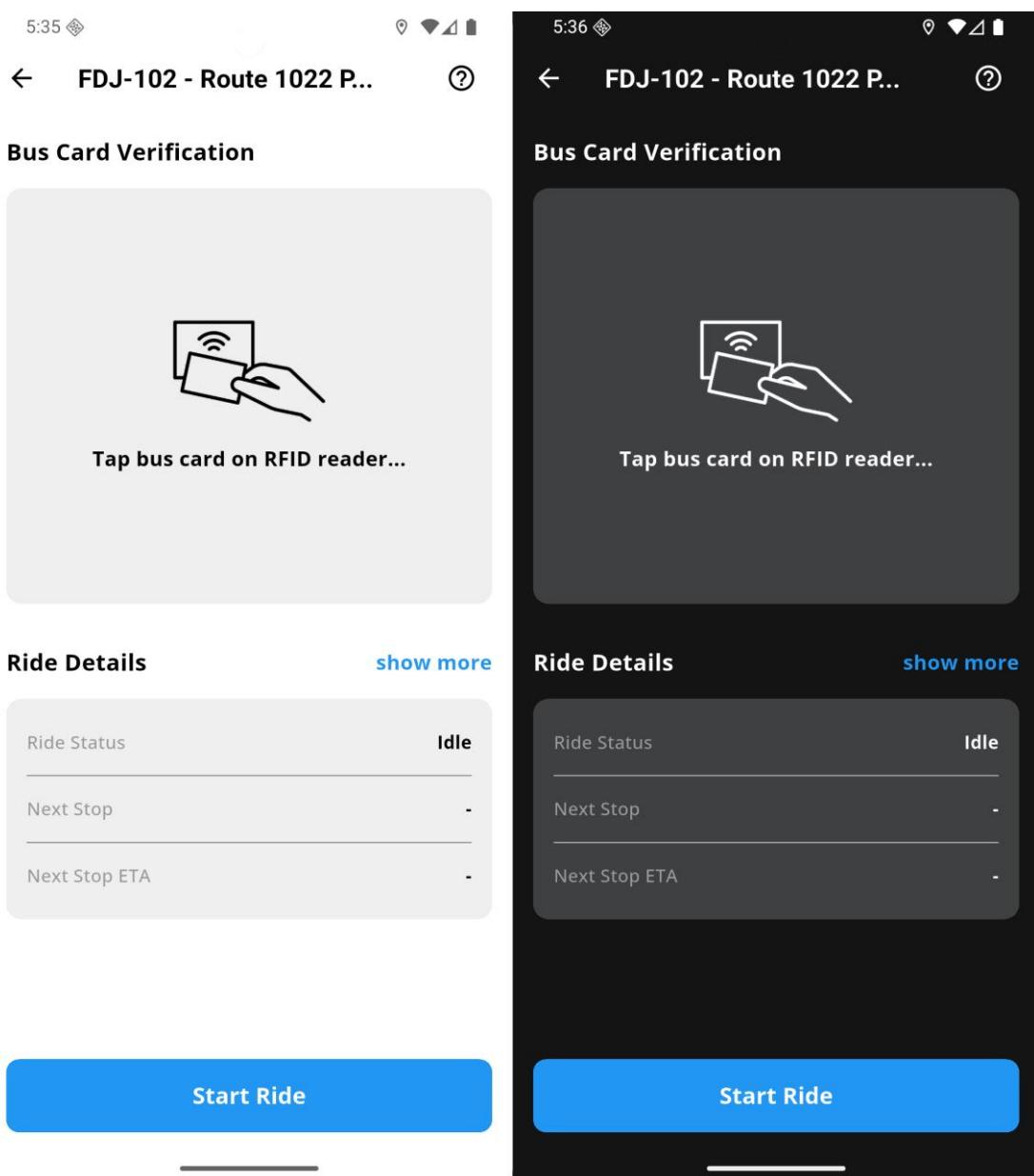


Figure 36 Ride Monitoring and Controls Screen

2.

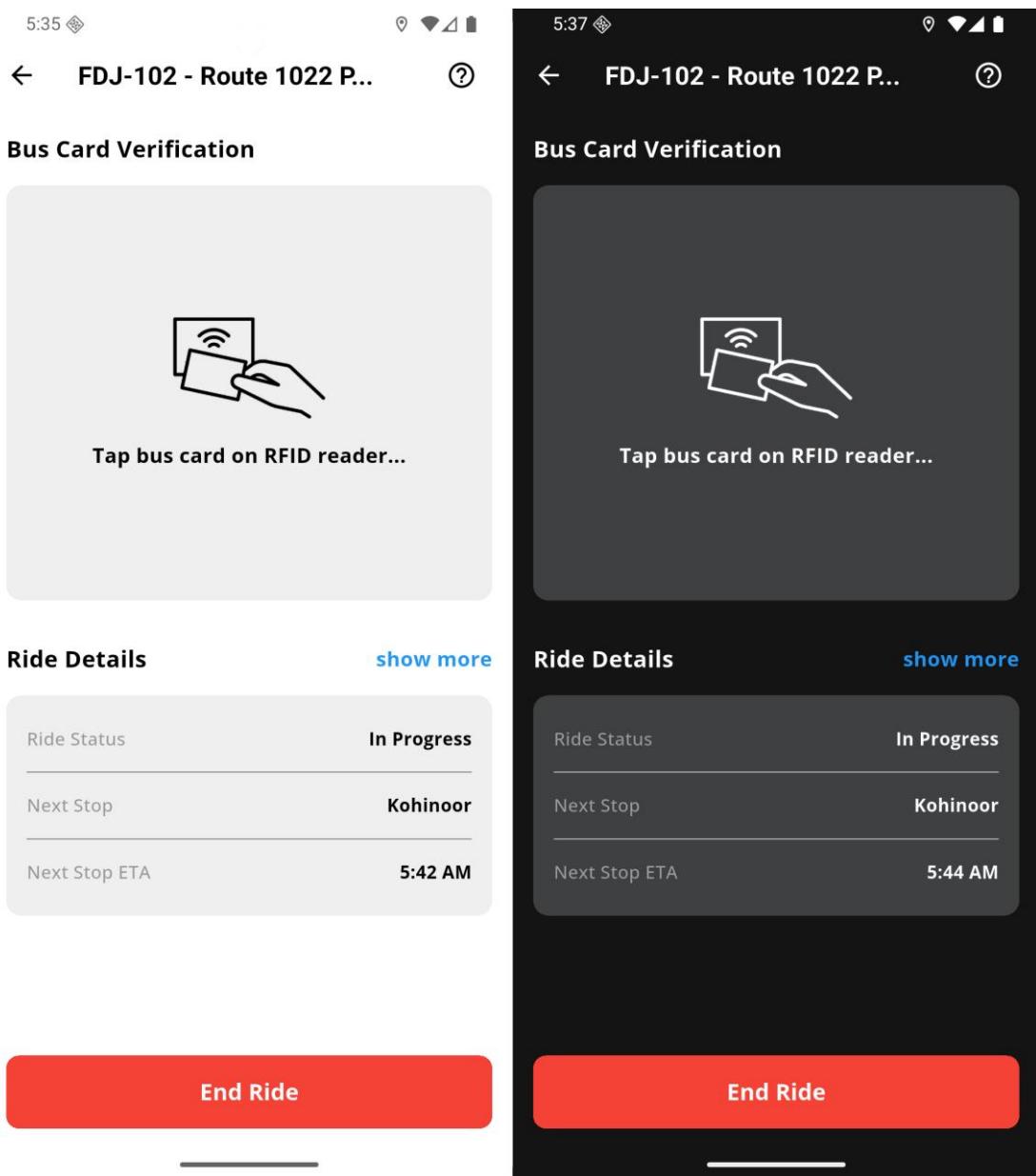


Figure 37 Ride Monitoring and Controls Screen Detailed

8.1.5 Profile & Settings Screens

8.1.5.1 Student Profile Screen:

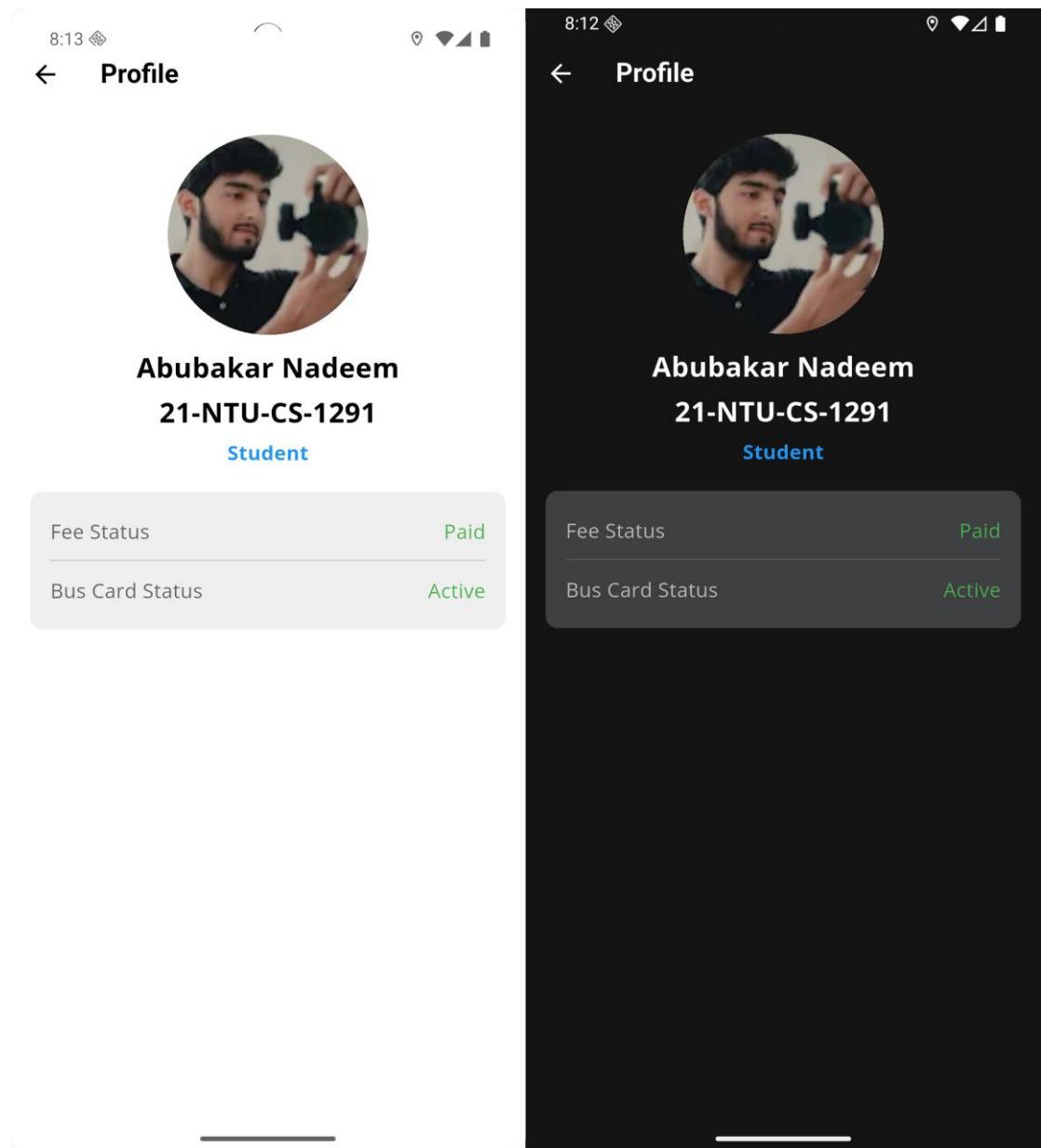


Figure 38 Student Profile Screen

8.1.5.2 Driver Profile Screen:

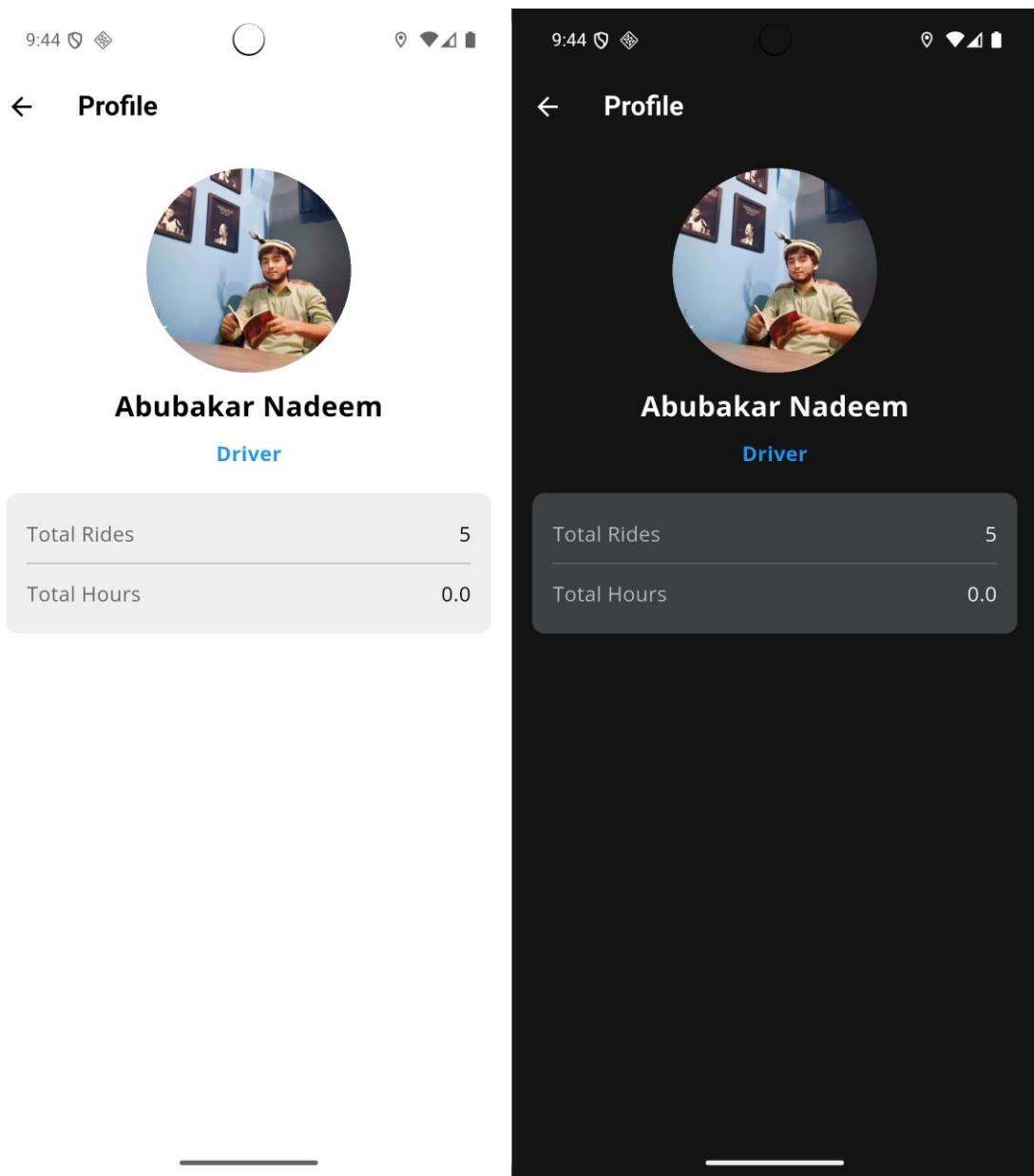


Figure 39 Driver Profile Screen

8.1.5.3 Settings Screen (Student and Driver):

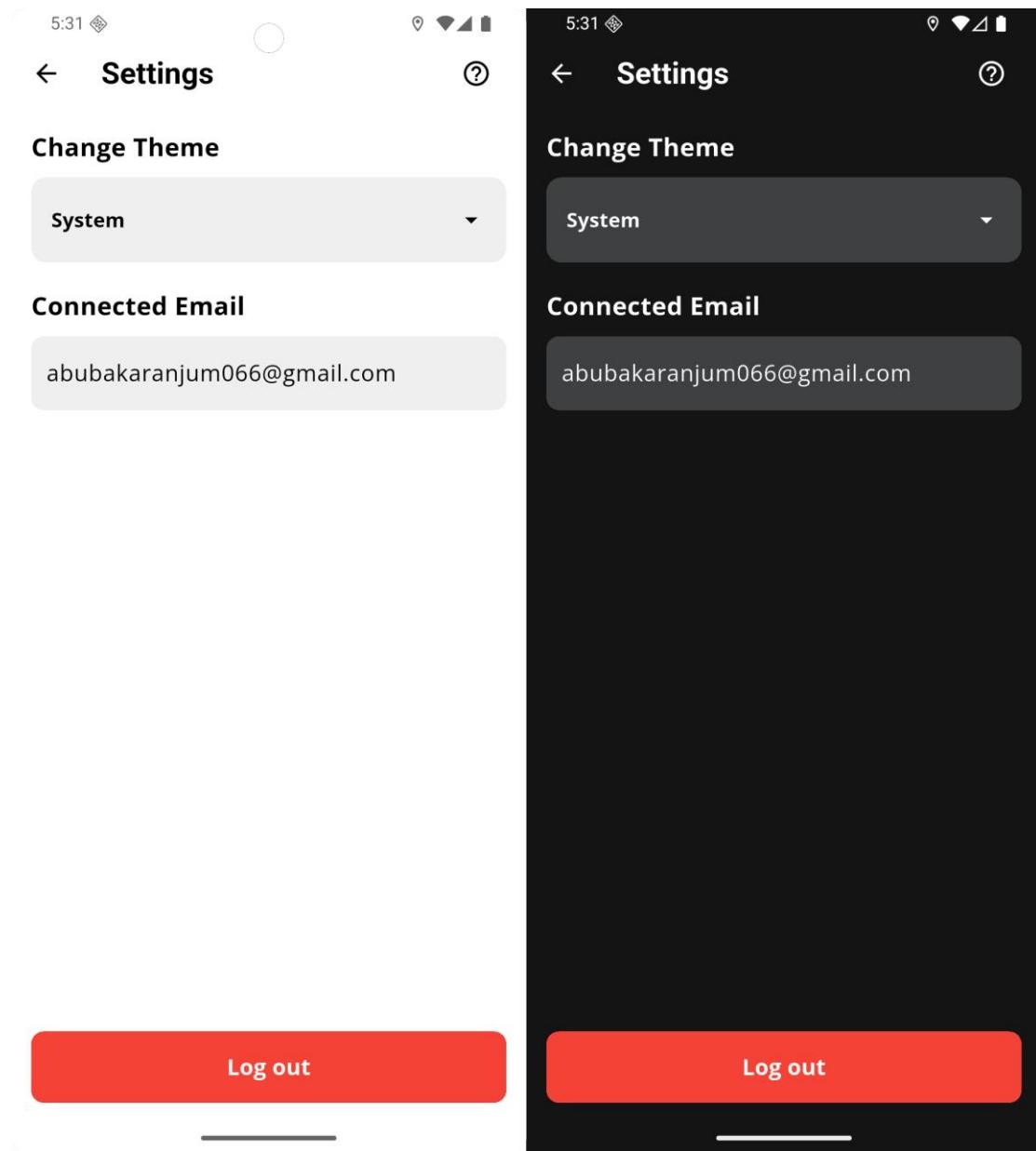


Figure 40 Settings Screen

8.1.6 Feedback & Complaint Screen (Student and Driver)

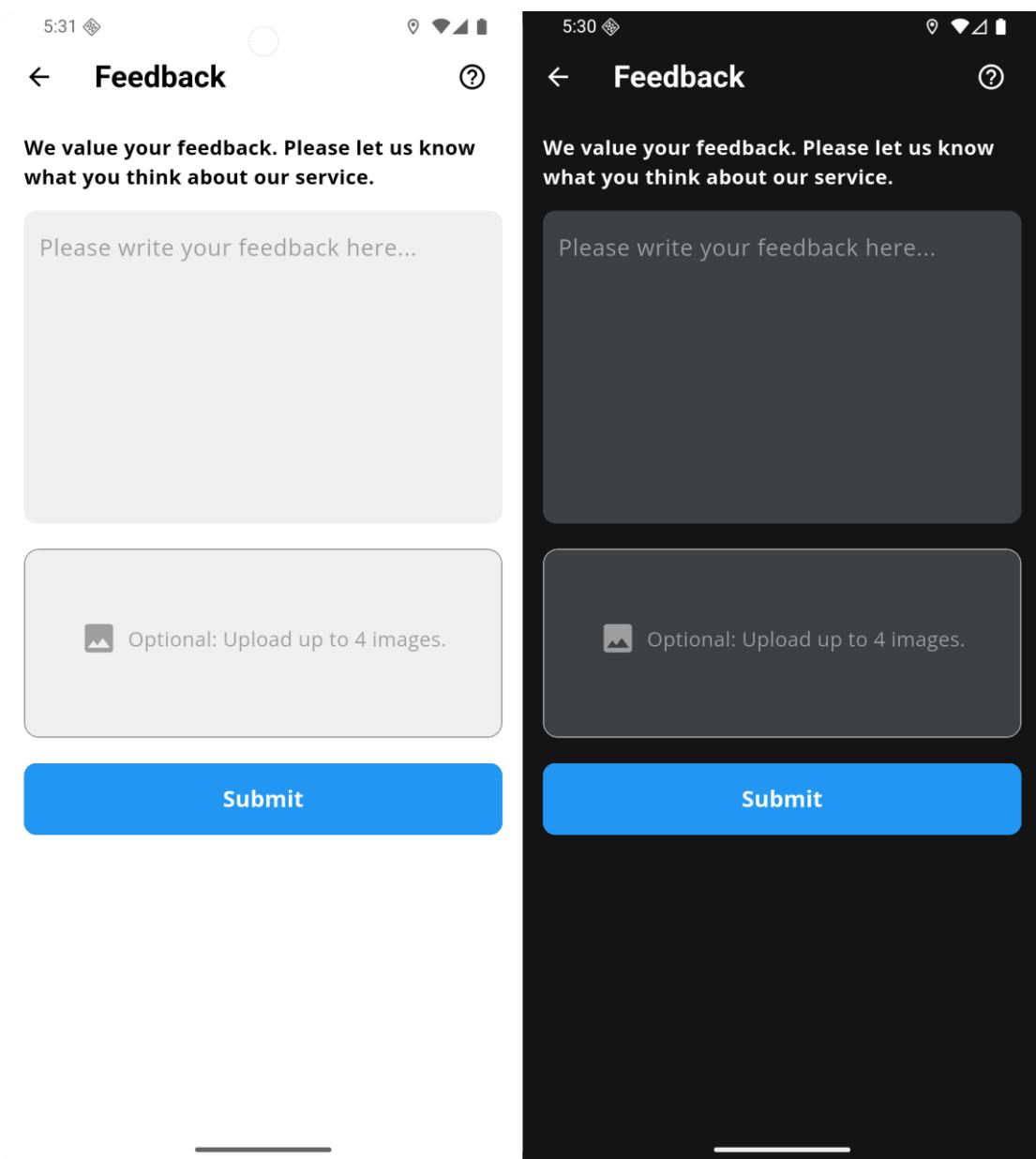


Figure 41 Feedback Screen

8.1.7 Help & Password Recovery Screens

8.1.7.1 Help Screen (Student and Driver):

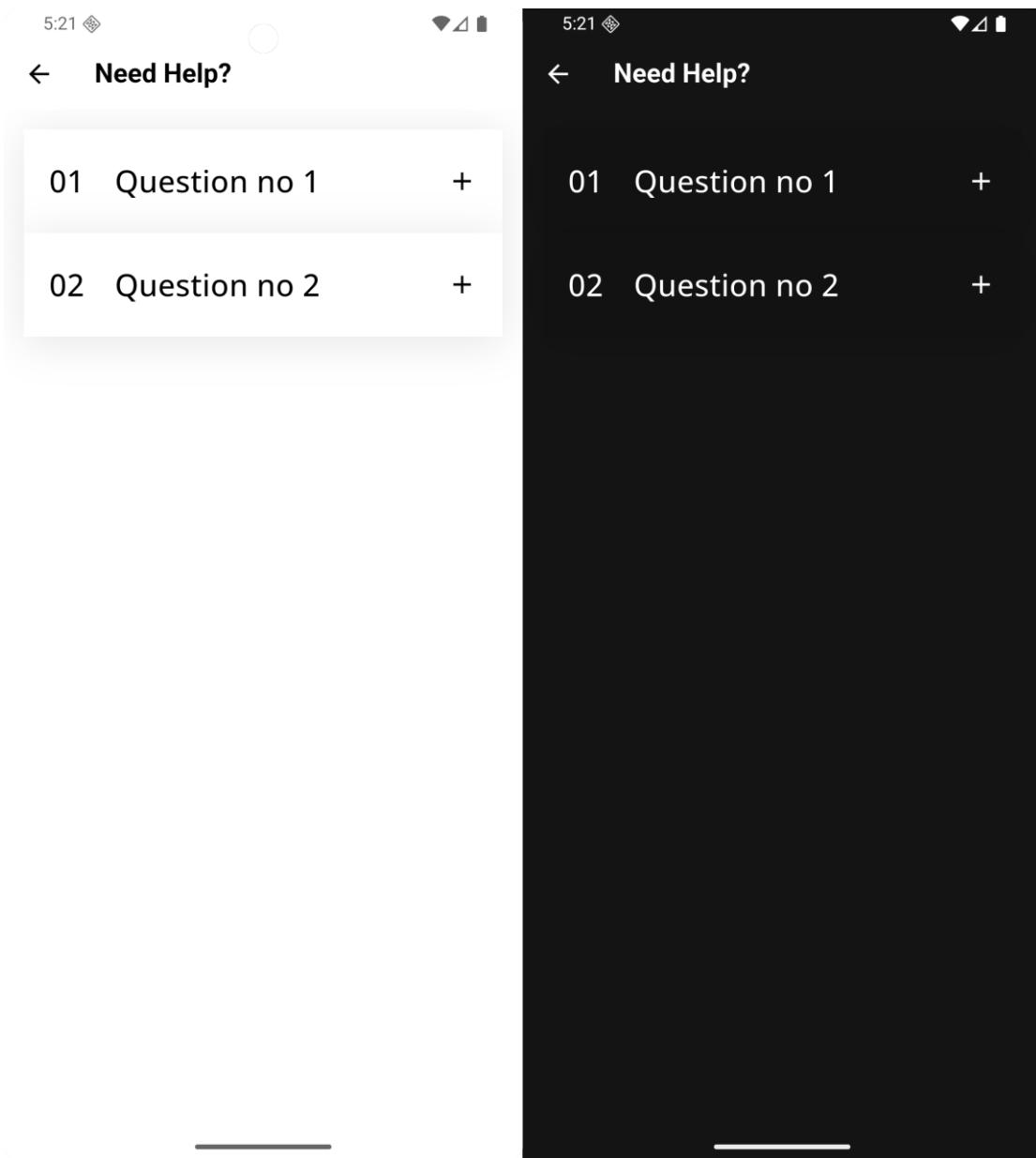


Figure 42 Help Screen

8.1.7.2 Forgot password Screen (Student and Driver):

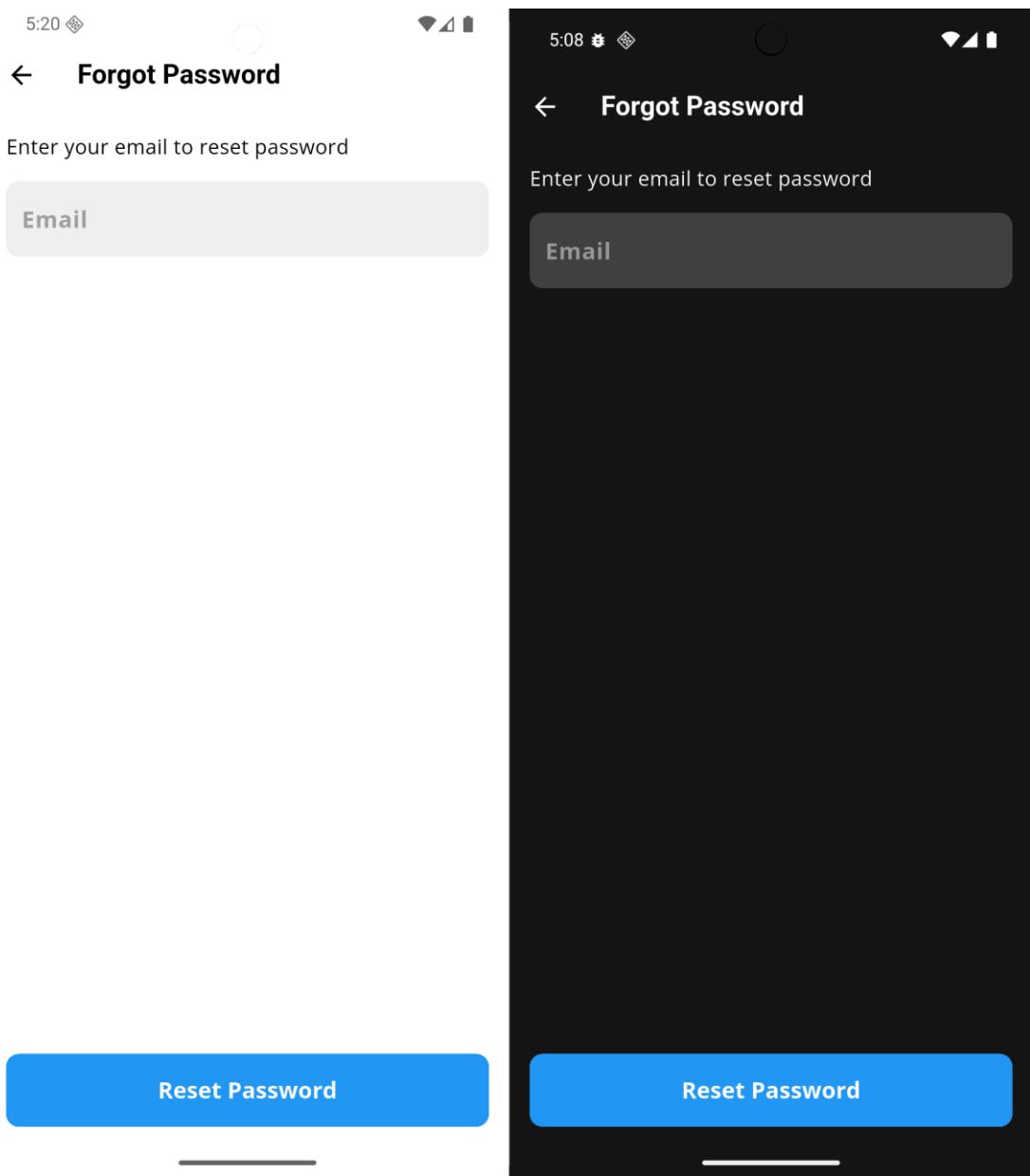


Figure 43 Forgot Password Screen

8.1.8 Notifications Screen (Student and Driver)

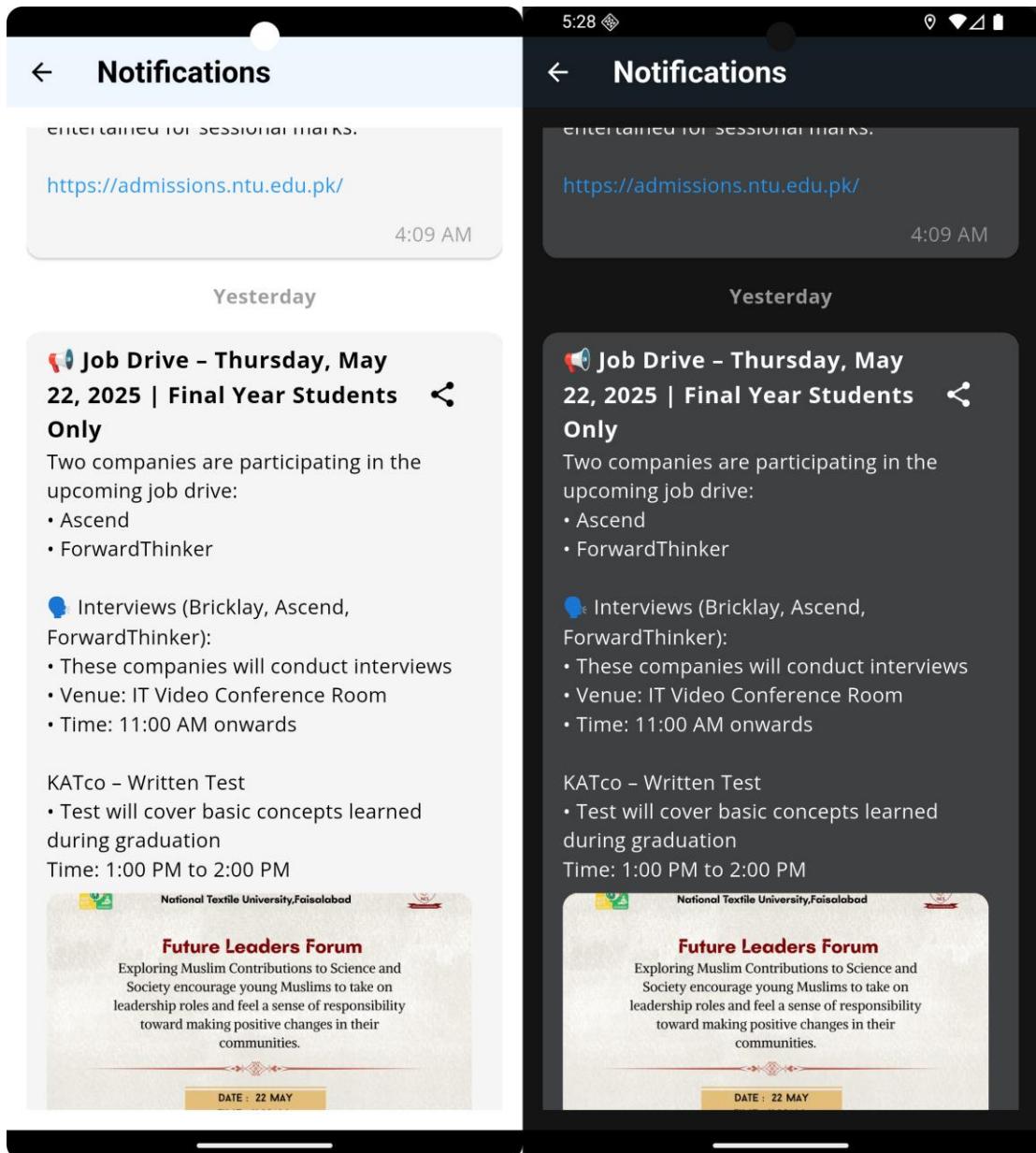


Figure 44 Notification Screen

8.2 Admin Panel Prototype (Web)

The Admin Panel is designed using Next.js with a clean, intuitive interface for monitoring and managing operations.

8.2.1 Admin Login Page

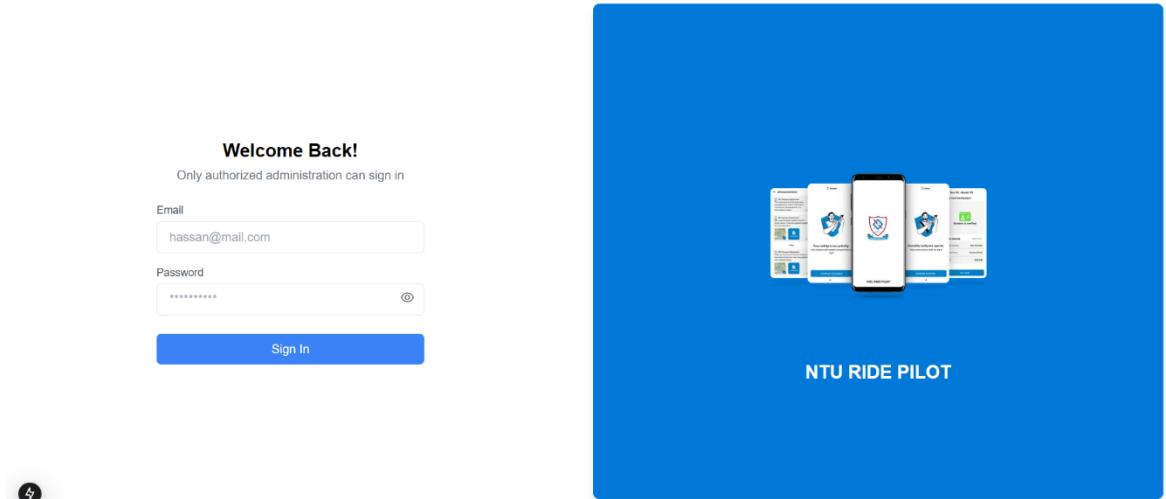


Figure 45 Admin login page

8.2.2 Dashboard Overview

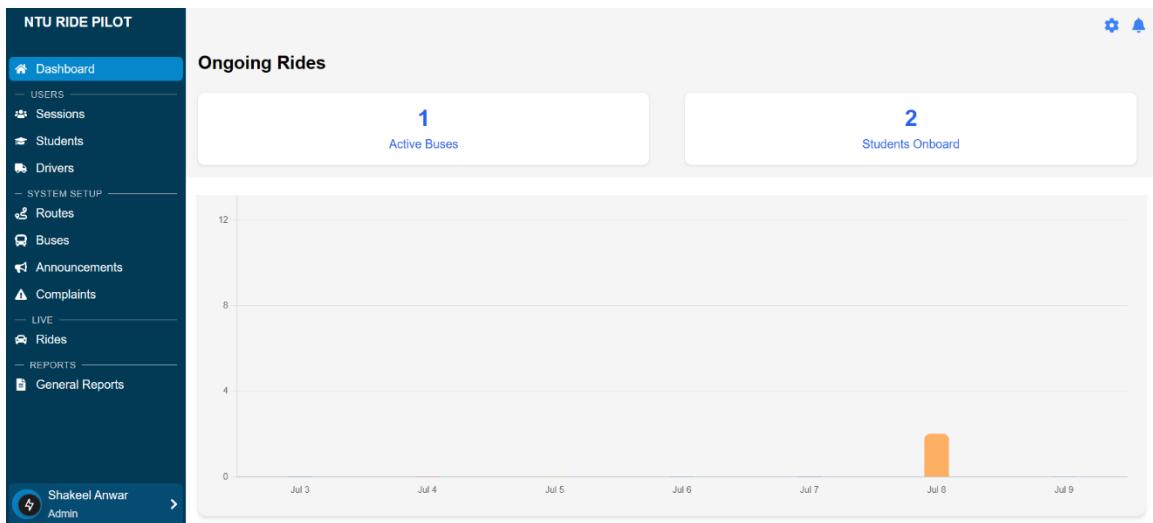


Figure 46 Dashboard Overview

8.2.3 Sessions Management

8.2.3.1 Sessions Details

The screenshot shows the NTU RIDE PILOT application interface. On the left is a dark sidebar menu with the following items:

- Dashboard
- USERS
 - Sessions (selected)
 - Students
 - Drivers
- SYSTEM SETUP
 - Routes
 - Buses
 - Announcements
 - Complaints
- LIVE
 - Rides
- REPORTS
 - General Reports

At the bottom of the sidebar is a user profile: Shakeel Anwar (Admin).

The main content area is titled "Sessions". It includes a search bar, a dropdown for "All", and a "Add Session" button. A table displays session details:

SR#	NAME	STARTING DATE	ENDING DATE	SESSION STATUS	ACTIONS
1	Fall-2025	09/07/2025	20/02/2026	Active	<button>Edit</button> <button>Deactivate</button>

Figure 47 Sessions Details

8.2.3.1 Add Sessions

The screenshot shows the "Sessions > Create Session" form. The sidebar menu is identical to Figure 47.

The form fields include:

- Name *: A text input field with placeholder "Enter session name".
- Starting Date *: A date input field with placeholder "dd/mm/yyyy".
- Ending Date *: A date input field with placeholder "dd/mm/yyyy".
- Buttons: "Clear" and "Add".

Figure 48 Add Sessions

8.2.4 Student Management

8.2.4.1 Student Details

The screenshot shows the 'Students' section of the NTU RIDE PILOT application. On the left is a dark sidebar with a navigation menu. The main area has a header 'Students' and a search bar. A dropdown menu shows 'Fall-2025'. Below is a table with columns: ID, NAME, ROLL NO, FEE, BUS CARD, and ACTIONS. The table contains six rows of student data.

ID	NAME	ROLL NO	FEE	BUS CARD	ACTIONS
1	Dania Imran	21-NTU-CS-1216	Paid	Active	<button>Edit</button> <button>Delete</button>
2	Aaniq Farooq	21-NTU-CS-1287	Paid	Active	<button>Edit</button> <button>Delete</button>
3	Abdullah Afaf	21-NTU-CS-1288	Paid	Active	<button>Edit</button> <button>Delete</button>
4	Abu Bakar	21-NTU-CS-1289	Paid	Active	<button>Edit</button> <button>Delete</button>
5	Abu Zar	21-NTU-CS-1290	Paid	Active	<button>Edit</button> <button>Delete</button>
6	Abubakar Nadeem	21-NTU-CS-1291	Paid	Active	<button>Edit</button> <button>Delete</button>

Figure 49 Student Details

8.2.4.2 Add Students

The screenshot shows the 'Add Student' form within the 'Students' section. The sidebar on the left is identical to Figure 49. The main form has fields for Roll Number, Name, Email, Fee Paid, Bus Card, and Bus Card Status. Buttons for 'Clear' and 'Add' are at the bottom right.

Roll Number *
YY -NTU- AB - 1234

Name *
Afa Bibi

Email *

Fee Paid *
Yes

Bus Card
Tap here...

Bus Card Status *
Inactive

Clear Add

Figure 50 Add Students

8.2.5 Driver Management

8.2.5.1 Driver Details

The screenshot shows the 'Drivers' section of the NTU RIDE PILOT application. On the left is a dark sidebar with a navigation menu. The 'Drivers' option is highlighted. The main area has a header 'Drivers' with a search bar and an 'Add Driver' button. Below is a table listing seven drivers:

ID	NAME	CONTACT	EMAIL	ACTIONS
1	Muhammad Dawood	+923067908383	dawood@gmail.com	<button>Edit</button> <button>Delete</button>
2	Abdullah Anjum	+923087857306	abdullahanjum075@gmail.com	<button>Edit</button> <button>Delete</button>
3	Abubakar Nadeem	+923087857306	abubakaranjum065@gmail.com	<button>Edit</button> <button>Delete</button>
4	Shamas ur Rehman	+923360700712	shamas@gmail.com	<button>Edit</button> <button>Delete</button>
5	Imran Ali Niaz	+923087857306	imranaliniaz786@gmail.com	<button>Edit</button> <button>Delete</button>
6	Huzaila	+923228862946	fsabilha511@gmail.com	<button>Edit</button> <button>Delete</button>
7	Muhammad Zahid	+923059161451	zahid@gmail.com	<button>Edit</button> <button>Delete</button>

Figure 51 Driver Details

8.2.5.2 Add Drivers

The screenshot shows the 'Driver > Add Driver' form. The left sidebar has the 'Drivers' option selected. The main form has fields for 'Name *', 'Email *', 'Contact *', and 'Profile Picture *'. There are 'Clear' and 'Add' buttons at the bottom.

Name *
Enter driver name

Email *
Enter driver email

Contact *
+92 Enter 12-digit contact number

Profile Picture *
Choose file No file chosen

Clear Add

Figure 52 Add Drivers

8.2.6 Routes Management

8.2.6.1 Route Details

The screenshot shows the 'Routes' section of the NTU RIDE PILOT application. On the left is a dark sidebar with navigation links for Dashboard, USERS, Sessions, Students, Drivers, SYSTEM SETUP, Routes (which is highlighted in blue), Buses, Announcements, Complaints, LIVE, Rides, and REPORTS. Below the sidebar is a user profile for 'Shakeel Anwar Admin'. The main area is titled 'Routes' and contains a table with four rows of route information:

SR#	NAME	STOPS	ACTIONS
1	Route 125 Pick-up	13	<button>View</button> <button>Delete</button>
2	Route 1022 Pickup	5	<button>View</button> <button>Delete</button>
3	Route 221 Drop	10	<button>View</button> <button>Delete</button>
4	Test Route	6	<button>View</button> <button>Delete</button>

Figure 53 Route Details

8.2.6.2 Add Routes

The screenshot shows the 'Add Route' page under the 'Routes' section. The sidebar and user profile are identical to Figure 53. The main area is titled 'Routes > Add Route' and contains a form with a 'Route Name *' field and a 'Save' button. Below the form is a map from Mapbox showing a yellow line representing the proposed route. The map includes labels for Ghaziabad 1, TARA GARGH KALAN, SOS Village, JUDICIAL COLONY, Airfield, East Canal Expressway, Baghpat Road, Ghora Road, Millat Road, Ghora Road, Naya Road, New Pur Road, and Jail Rd. A search bar at the bottom of the map says 'Search for a location in Punjab, Pakistan'.

Figure 54 Add Routes

8.2.7 Bus Management

8.2.7.1 Bus Details

The screenshot shows the 'Buses' section of the NTU RIDE PILOT application. On the left is a dark sidebar with a navigation menu. The 'Buses' option is highlighted. The main area has a header 'Buses' with a search bar and an 'Add Bus' button. Below is a table with four rows of bus data:

ID	REGISTRATION NO	NO OF SEATS	ACTIONS
1	1000	35	<button>Delete</button>
2	FDJ-102	25	<button>Delete</button>
3	FDJ-21	25	<button>Delete</button>
4	FDY-1808	25	<button>Delete</button>

Figure 55 Bus Details

8.2.7.2 Add Bus

The screenshot shows the 'Add Bus' form within the 'Buses' section of the NTU RIDE PILOT application. The sidebar on the left remains the same. The main area has a header 'Buses > Add Bus'. It contains two input fields: 'Registration No *' and 'Number of Seats *'. Below each field is a dropdown menu. To the right are 'Clear' and 'Add' buttons.

Figure 56 Add Bus

8.2.8 Announcements Management Module

8.2.8.1 Announcement Details

The screenshot shows the 'Announcements' section of the NTU RIDE PILOT application. On the left is a dark sidebar with various navigation links such as Dashboard, USERS, Sessions, Students, Drivers, SYSTEM SETUP, Routes, Buses, Announcements (which is highlighted), Complaints, LIVE, Rides, REPORTS, and General Reports. Below the sidebar is a user profile for 'Shakeel Anwar Admin'. The main area is titled 'Announcements' and contains a table with the following data:

ID	TITLE	MESSAGE	CREATED ON	ACTIONS
1	hi	test	07/08/2025 3:41 AM	<button>View</button> <button>Delete</button>
2	Imp notification	Alert about route changes	07/08/2025 3:41 AM	<button>View</button> <button>Delete</button>
3	Test Announcement 🎉	Just a test announcement.	07/08/2025 3:01 AM	<button>View</button> <button>Delete</button>
4	⚠️ Route Change Alert for Bus 1022! 🚍	Due to road maintenance on Elm Street, Bus 1022 wi...	07/06/2025 8:43 PM	<button>View</button> <button>Delete</button>
5	❗ Job Drive – Thursday, May 22, 2025 Final Year Students Only	Two companies are participating in the upcoming jo...	07/02/2025 4:12 AM	<button>View</button> <button>Delete</button>
6	REMINDER!	It is for the information of all the students who ...	07/01/2025 4:09 AM	<button>View</button> <button>Delete</button>
7	Fun activity! 🌟	Lorem ipsum dolor sit amet, consectetur adipiscing...	07/01/2025 1:16 AM	<button>View</button> <button>Delete</button>

Figure 57 Announcements Details

8.2.8.2 Add Announcements

The screenshot shows the 'Add Announcement' form within the NTU RIDE PILOT application. The left sidebar is identical to Figure 57. The main form has a title 'Announcement > Add Announcement'. It contains the following fields:

- Title***: A text input field labeled 'Enter Announcement Title'.
- Message***: A large text area labeled 'Enter your message here'.
- Attach Media Files (Optional)**: A file upload section with a 'Choose files' button and a message 'No files chosen'. To the right are 'Clear' and 'Add' buttons.

Figure 58 Add Announcement

8.2.9 Complaints Management

8.2.9.1 View all Complaints

SR#	NAME	EMAIL	MESSAGE	STATUS	ACTIONS
1	Abdullah Anjum	abdullahanjum075@gmail.com	driver complains	Resolved	<button>View</button> <button>Delete</button>
2	Abdullah Anjum	abdullahanjum075@gmail.com	Complaint	Resolved	<button>View</button> <button>Delete</button>

Figure 59 View all Complaints

8.2.9.2 View single Complaint

Name*
Abdullah Anjum

Email*
abdullahanjum075@gmail.com

Message*
driver complains

Attached Media Files
arringe
itsgokl4m0rlf5kgraxa.jpg

Figure 60 View single Complaint

8.2.10 Rides Management

Please Select a Ride!

Driver Contact

Ride Status Started At N/A

Next Stop ETA Next Stop N/A

Bus Capacity Student Onboard 0

Passengers Onboard Search

Figure 61 Rides Management

8.2.11 General Reports Module

SR#	ROLL NO	CREATED AT	REMARKS
1	21-NTU-CS-1900	08/07/2025, 04:05:19 08/07/2025, 04:12:03 08/07/2025, 04:13:24	1000 1000 FDJ-102
2	21-NTU-CS-1216	08/07/2025, 04:13:24	FDJ-102

Figure 62 General Reports

References

- (n.d.). Retrieved from <https://www.tylertech.com/products/student-transportation>
- (n.d.). Retrieved from <https://www.edulog.com/>
- (n.d.). Retrieved from <https://loqqat.com/>