A Final Project Report On

RIDE SHARING APPLICATION USING FLUTTER

Submitted in the Partial Fulfillment of the
Requirements for the Degree of Bachelor of Computer Engineering
Awarded by Pokhara University

Submitted By:

Ashish Pandey (21075560)

Saru Tiwari (21075581)

Sushil Giri (21075591)

Tirtha Singh Khadka (21075593)

Supervisor:

Er. Sushant Paudel

School of Engineering

Faculty of Science and Technology

POKHARA UNIVERSITY

April, 2025

A Final Project Report On

RIDE SHARING APPLICATION USING FLUTTER

Submitted in the Partial Fulfillment of the
Requirements for the Degree of Bachelor of Computer Engineering
Awarded by Pokhara University

Submitted By:

Ashish Pandey (21075560)

Saru Tiwari (21075581)

Sushil Giri (21075591)

Tirtha Singh Khadka (21075593)

Supervisor:

Er. Sushant Paudel

School of Engineering

Faculty of Science and Technology

POKHARA UNIVERSITY

April, 2025

COPYRIGHT

STUDENT'S DECLARATION

We hereby declare that this project work entitled **RIDE SHARING APPLICATION USING FLUTTER** is based on our original work. All concepts, data, code, and any other work from external sources have been properly cited and referenced in accordance with the guidelines provided by School of Engineering, Pokhara University

We owe all the liabilities relating to the authenticity and originality of this project work and project report.

1. Ashish Pandey (21075560) ……..
2. Saru Tiwari (21075581) ……..
3. Sushil Giri (21075591) ……..
4. Tirtha Singh Khadka (21075593) ……..

Date: 18/04/2025

SUPERVISOR'S RECOMMENDATION

This is to certify that this project report entitled **RIDE SHARING APPLICATION USING FLUTTER** prepared and submitted by below listed team of students in partial fulfilment of the requirements of the degree of Bachelor of Computer Engineering awarded by Pokhara University, has been prepared and completed under my supervision.

I hereby recommend the same for acceptance by the School of Engineering, Pokhara University.

1. Ashish Pandey (21075560)
2. Saru Tiwari (21075581)
3. Sushil Giri (21075591)
4. Tirtha Singh Khadka (21075593)

……..

Asst. Prof. Er. Sushant Paudel

School of Engineering

Pokhara University

Date: ……

School of Engineering

POKHARA UNIVERSITY

EXTERNAL EXAMINER'S RECOMMENDATION

The undersigned certified that they have evaluated this project report entitled **RIDE SHARING APPLICATION USING FLUTTER** submitted by **Ashish Pandey, Saru Tiwari, Sushil Giri, Tirtha Singh Khadka,** and their oral presentation for partial fulfillment of the degree of Bachelor of Computer Engineering and recommended to the School of Engineering, Pokhara University for acceptance of this project work/report.

…….

Er. Bikash Bhattarai
Assistant Professor
School of Engineering
Pokhara University

Date: ……

Acknowledgement

Firstly, we would like to express our deepest gratitude to our Supervisor **Er. Sushant Paudel**. His guidance, assistance and numerous discussion sessions instilled in us the confidence and enthusiasm necessary to design and build this application. We are also grateful to all the helping hands who suggested and helped us to develop this project.

Abstract

This proposal presents a ride sharing application designed to improve urban mobility, reduce environmental impact, and enhance user convenience. It offers mobility solutions by providing economic ride share along with a wide range of vehicles across bikes and cars. By using GPS and GSM technology, users can book the bike or cabs anywhere and then drivers locate them within a few meters. The application leverages cutting-edge technologies, including intelligent route-matching algorithms to ensure a seamless user experience.

Key Words: *GPS, GSM, ride sharing*

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATIONS

API	Application Programming Interface
ERD	Entity Relationship Diagram
GPS	Global Positioning System
GSM	Global System for Mobile
UI	User Interface
SDLC	Software Development Life Cycle

Chapter 1

Introduction

1.1 Background

The idea of ride-sharing has been around for many years. In the past, people used to share rides informally with friends, neighbors, or coworkers to save money and reduce fuel use. Later, carpooling became more organized, especially in offices or schools. However, these systems had many limitations, like fixed schedules and a lack of flexibility. With the development of smartphones, GPS technology, and the internet, ride-sharing has changed completely. Now, mobile applications like Uber, Ola, and Pathao have made it easy for people to book rides anytime from anywhere. These modern apps connect passengers and drivers in real time, helping people travel faster, cheaper, and more comfortably.

In the context of Nepal, cities like Kathmandu, Pokhara, and others are facing growing transportation challenges. Traffic congestion, air pollution, rising fuel prices, and limited parking spaces are becoming serious issues. While some ride-sharing platforms like Pathao and InDrive have already entered the Nepali market, there is still a need for more localized, user-friendly, and cost-effective ride-sharing solutions that address both urban and semi-urban areas. Many people in Nepal do not own private vehicles or prefer not to use them daily due to high fuel costs and road conditions. Ride-sharing apps offer a practical solution by allowing users to share rides and reduce the number of vehicles on the road. This helps to save money, reduce pollution, and improve mobility. This project is developed to meet the growing need for smart and affordable transportation in Nepal. It allows users to book rides using a mobile app, where they can choose between bike and car options. The app uses GPS to help drivers find users quickly. It also stores booking and user data using a backend system, and makes the whole process smooth with features like user-friendly UI and ride activity history. Overall, the goal of this ride-sharing app is to make transportation easier in Nepal, reduce pollution, support local drivers with earning opportunities, and help build a more connected and efficient community.

1.2 Problem Statement

A significant portion of Nepal's population relies on public transportation for daily commuting, especially in urban areas like Kathmandu. According to a study published in the Journal of Economic Structures, approximately 47.45% of people use buses and 23.59% use microbuses as their primary mode of travel, indicating that over 70% depend on public vehicles for daily transportation (Aryal and Kaneko, 2022). Additionally, more than 90% of total domestic movement of goods and passengers depends on public road transportation in Nepal (Surendra Prasad Ghimire, 2023). Despite this heavy reliance, public transportation faces serious challenges such as overcrowding, inefficiency, and irregular service availability. In many cases, especially during peak hours or in less connected city areas, public transport might not be available at all. Due to the overcrowded buses and poor roadway conditions, people often waste time and fail to reach their destinations punctually. To address these issues, this project aimed to develop a user-friendly ride-sharing application that provided an alternative mode of transport for daily commuters. The app allowed users to register, log in, and book rides by selecting pickup and drop-off locations. It also included features such as fare estimation and real-time tracking to improve the commuting experience and reduce dependence on overcrowded public transportation.

1.3 Objectives

The main objectives of this project include following:

- To provide a simple and user-friendly platform for choosing and booking rides using a mobile app.,
- To display ride-related details such as pickup and drop-off locations, estimated cost, time, and ride status,

1.4 Applications

This ride-sharing system was developed to provide a simple and accessible platform for users to book rides using bikes or cars. The application allows users to request transportation from one place to another through a mobile interface,

helping them avoid long waits for buses or taxis. The app shows the user's current location on a map, and the marker moves as the user changes position. This helps users know where they are and confirm pickup points easily. At this stage, only the user's location is shown. Real-time location sharing between drivers and users is not yet included. This application is useful for students, office workers, and anyone who needs a quick and affordable way to travel.

1.5 Project Features

The objectives of this project are the following:

- It allows users to sign up and log in using mobile numbers.
- It enabled users to book a ride by entering pickup and drop-off locations.
- It displayed the user's current location on the map and updated as the user moved.
- It allowed users to choose between bike and car options.
- It stored user and ride data securely using a backend system.
- It showed the status of the ride (pending, accepted, completed).
- It maintained a ride history for the user.
- It allowed users to cancel the ride within 5 minutes after booking.
- It allowed users to manage their profile.

1.6 Feasibility Analysis

1.6.1 Economic Feasibility

Since the project is small and app based, there is no economic barrier for the project, it is economically feasible. But in future, we will make this project more productive and more advanced. In such time, the design and development with the proper management of the servers and updates might be some costly. Until that phase, this project has cost less to build and maintain. In the case of API Key, we are using free open sources.

1.6.2 Technical Feasibility

This project is developed using the Dart programming language for the frontend, built with the Flutter framework. For the backend, we are using Node.js along with Firebase for real-time data handling and MySQL as the main database for storing user and ride information. Geo-location technologies like GPS, combined with mapping APIs such as OpenStreetMap, help with route optimization and real time ride requesting and accepting. We use Postman to create, test, and manage APIs efficiently.

1.6.3 Operational Feasibility

The ride-sharing application is easy to set up and use, making it a seamless addition to daily operations. Both riders and users find it simple to navigate. The system allows users to book rides with just a few taps and provides real-time updates on vehicle locations. It also includes features like ride-sharing, fare estimation, and ride history, making it a practical and efficient solution for users.

1.7 System Requirement

1.7.1 Software Requirement

The application is developed using the Dart programming language with the Flutter framework for the front-end. For the backend, Node.js is used, while Firebase handles real-time connections. MySQL is used for the database to store user and driver information. Postman is used to test and manage the APIs.

1.7.2 Hardware Requirement

To develop and program this ride-sharing application, a computer or laptop with at least 8GB of RAM is required. Since the project is a mobile application, the minimum hardware requirement for users to operate the app is a smartphone with modern capabilities and sufficient storage to run the app smoothly.

Chapter 2

Literature Review

2.1 Evolution

Ride-sharing apps have turned into one of the easiest ways to travel in the past two decades. In the context of Nepal, the first Nepali ride sharing application Tootle was launched in 2017 A.D. Studies highlight the role of ridesharing in mitigating traffic congestion and lowering carbon emissions through vehicle pooling (Shaheen & Cohen, 2019). The adoption of real-time geo location and route optimization technologies has been pivotal in improving user convenience and operational efficiency (Jung et al., 2020). Owning a vehicle is still out of reach of general masses in cities. The entry of ride sharing models has become a very good alternative to public transportation. These platforms introduced innovative solutions like real-time GPS location, dynamic pricing models, and user feedback systems, which significantly enhanced user experience and operational efficiency. Researchers like Masoud and Jayakrishnan (2017) have explored the environmental implications of ride sharing. They found that carpooling options offered by ride sharing apps significantly reduce carbon emissions, particularly in urban settings. This aligns with sustainability goals by encouraging shared vehicle usage over single occupancy trips. Smith (2016) observed that these apps democratize access to transportation, particularly in areas undeserved by traditional public transit.

Chapter 3

Methodology

3.1 Introduction

To achieve the specific objectives stated, a set of methods was used. System requirements were gathered using different data collection techniques, and system design was accomplished using entity-relationship diagrams and flowcharts. The system was implemented using various programming languages, and it was tested and validated to identify and correct errors and to ensure that it fulfilled its intended purpose.

3.2 Software Development Life Cycle

The development of ride sharing application followed a structured approach using the Waterfall Model in the Software Development Life Cycle (SDLC). This model was chosen for its clear, step-by-step progression through the stages of the project, ensuring the platform met user needs and educational goals effectively.

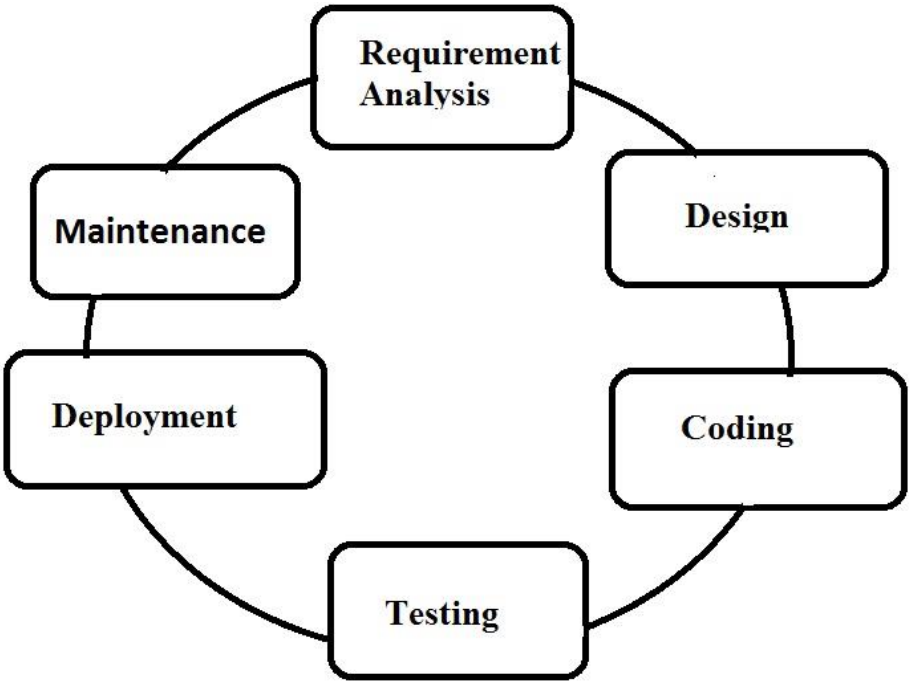


Figure 1: Software Development Life Cycle

3.3 Development Process Using Waterfall Model

The framework we have incorporated for developing this project is The Waterfall Model. This model follows a linear and sequential approach for software development, where each phase must be completed before moving into the next. The Waterfall Model was ideal for this project because of its clear and structured approach. Each phase requirements gathering, design, development, testing, and deployment was completed sequentially. Although the model limits flexibility during development, feedback was gathered after launch to refine the platform.

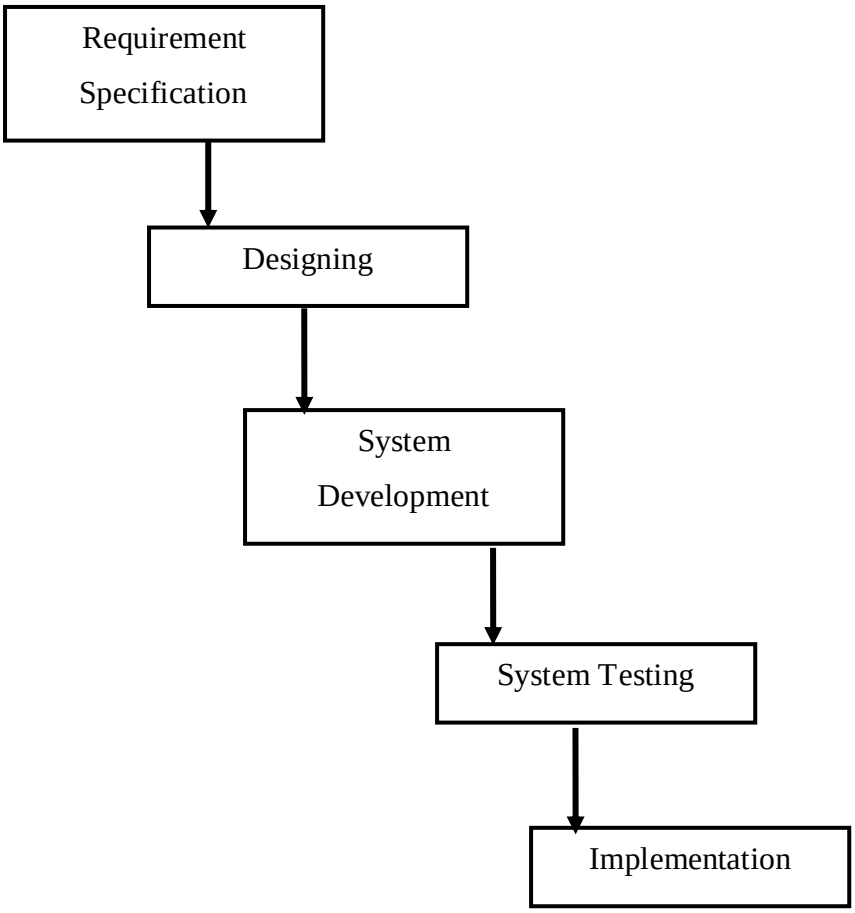


Figure 2: Waterfall Model

For this project, we have used Waterfall Methodology as our SDLC model. The Waterfall Model follows a linear and sequential approach, where each phase is completed before moving on to the next. This model is well-suited for projects with clearly defined requirements and objectives, making it an ideal choice for our ride sharing system.

3.3.1 Requirement Specification

For the development of the ride sharing system, we have specified the following requirements:

- User registration, ride booking, fare estimation.
- Earning summaries for drivers.
- Scalability to handle increasing users and rides.
- Security measures for data protection.

3.3.2 System Design Specification

For the design phase of the ride-sharing system, the focus is in creating robust and scalable architecture. The design covers the following aspects:

- A three-tier architecture comprises the presentation layer (UI), application layer (backend logic), and database layer.
- Separate modules for rider and driver interfaces.

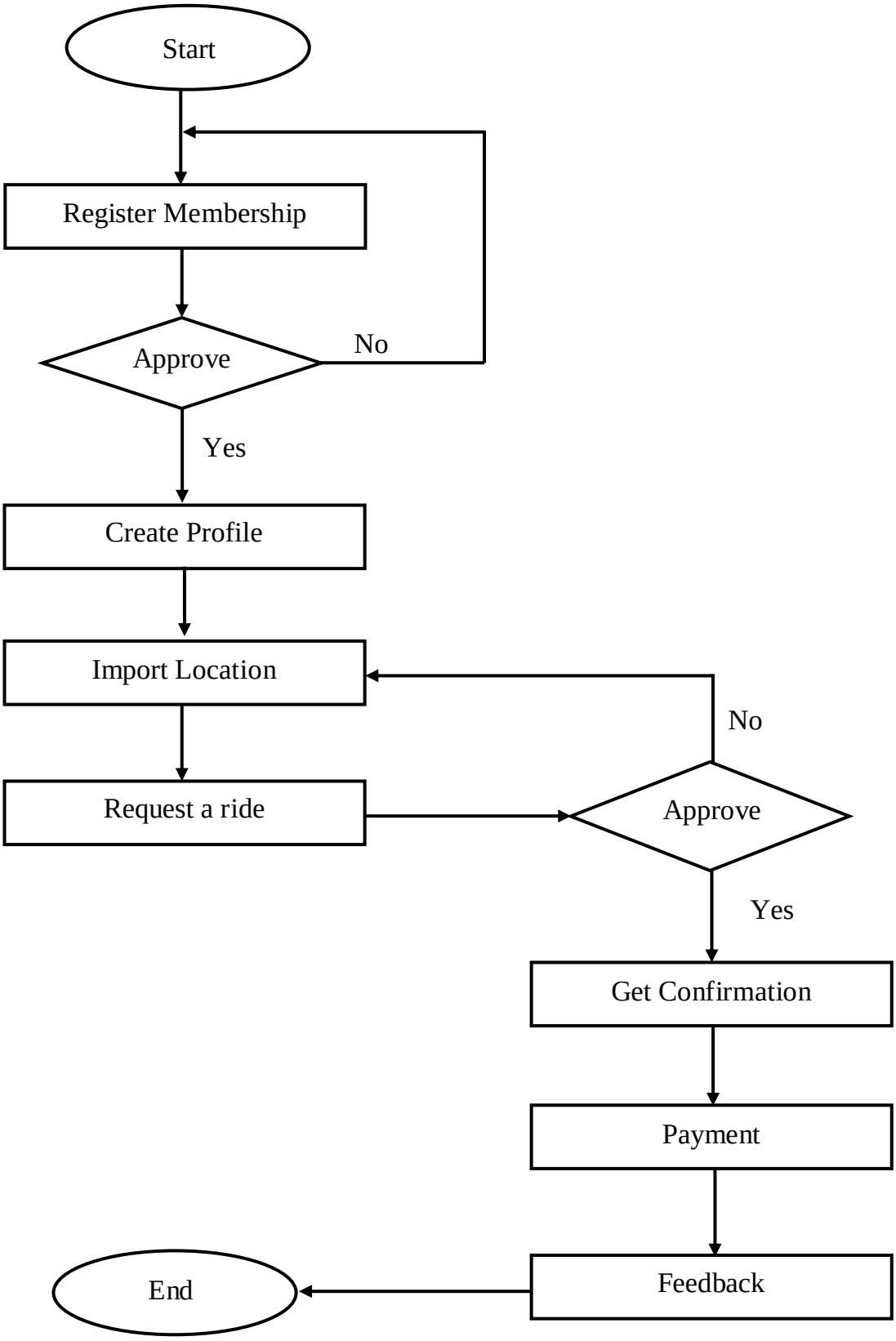


Figure 3: Flowchart

A flowchart is a diagram that visually represents a process, workflow, or system using standardized symbols and arrows. It shows the sequence of steps or decisions required to complete a task or achieve a goal. Flowcharts were used to simplify complex processes, making them easier to understand, analyze, and communicate effectively.

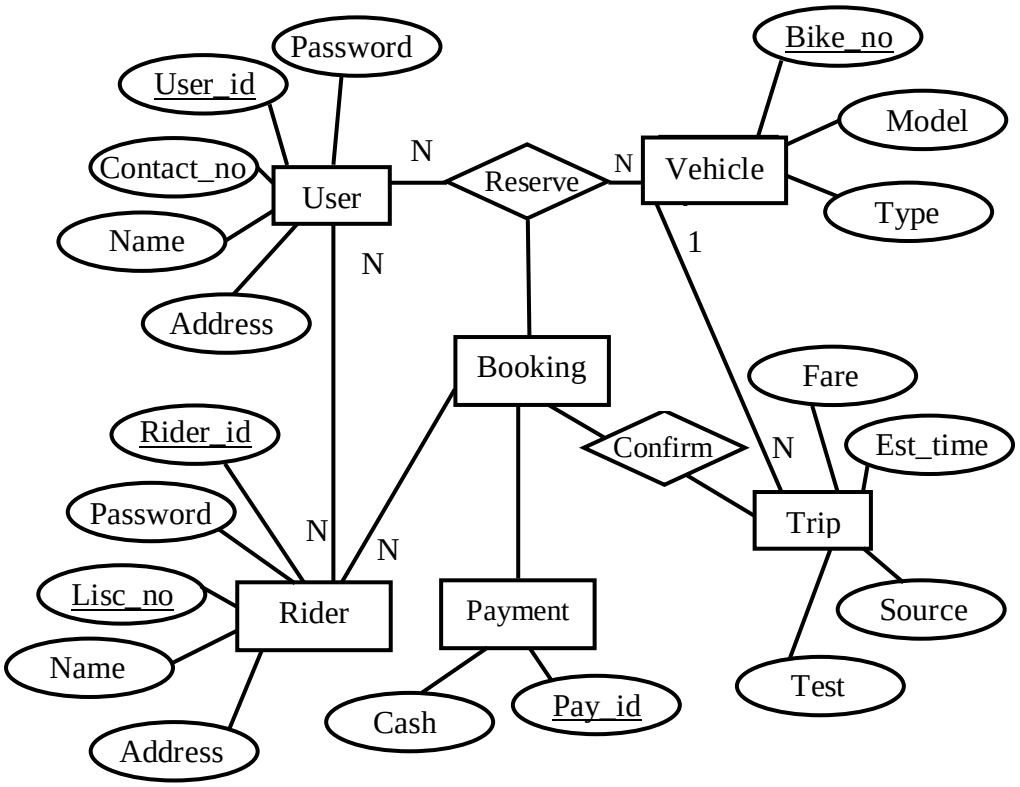


Figure 4: ER Diagram

An ER Diagram is a visual representation of the entities and their relationships within a system. It's a type of flowchart that helps understand how different entities, such as people, objects, or concepts, connect with each other.

3.3.3 System Development Specification

The system development phase for the ride-sharing system involves implementing the planned design into a functional system. This phase includes:

3.3.3.1 Frontend Development:

- Created user interfaces for rider and users using frameworks like Flutter-Dart for the mobile application.
- Integrated third-party APIs such as open street maps for location and navigation.

3.3.3.2 Backend Development:

- Developed the APIs for handling ride requests, ride matching using Node.js and tested using Postman.
- Firebase is used as backend for authentication like signup and login.

3.3.3.3 Database Development:

- Set up a relational database like MySQL for storing all kinds of data of rider and users.
- We have also included the Firebase for storing the ride details and all the transactions.

3.3.4 System Testing

The system testing phase for the ride-sharing system has focused on verifying the functionality, performance, and reliability of the system. Ensure correct functioning of key features like ride booking and driver assignment.

3.3.5 Implementation

The implementation phase involves deploying the ride-sharing system to a live environment. After final checks, the system is launched for users, and comprehensive documentation is prepared to support future maintenance and user assistance.

Chapter 4

Result

The ride-sharing application has successfully implemented key functionalities, including real-time ride booking, user registration, ride booking and fare estimation. The system also incorporates a dedicated driver interface that facilitates ride management and navigation. The front-end of the application has been developed using the Flutter framework, providing cross-platform compatibility and intuitive user experience. On the back end, APIs developed with Node.js to manage user authentication, ride matching while a MySQL database ensures storage of user profiles, ride histories and activities. And we have used Postman to test the APIs whether it is working as planned or not. By leveraging open-source APIs like OpenStreetMap, the project minimizes costs while maintaining functionality. Initial usability tests confirm that the application is user-friendly and meets operational goals, ensuring good experience for riders and drivers alike. We also have used the Firebase for the booking request to be sent to drivers from user as a real-time database.

From a social and environmental perspective, the application promotes ridesharing as a sustainable transportation alternative, reducing traffic congestion and lowering carbon emissions. It provides an accessible and affordable commuting option, particularly for urban populations with limited access to private vehicles. Preliminary testing and user feedback have been positive, with users appreciating the responsive interface and reliable performance. Functional and integration testing has validated the system's core features, such as ride booking and ride completing ensuring their smooth operation.

Chapter 5

Conclusion

5.1 Conclusion

The ride-sharing application offers a convenient, reliable, and cost-effective solution to urban transportation challenges. It ensures smooth interaction between users and drivers through a user-friendly interface and essential features such as fare estimation based on distance and time, and vehicle type selection. Additionally, the platform enhances user control and satisfaction by allowing users to cancel a booked ride within 5 minutes of confirmation. This feature improves flexibility and helps manage accidental bookings. Overall, the application contributes to smarter urban mobility while promoting efficient ride management for both users and drivers.

5.2 Limitations and Future Recommendations

5.2.1 Limitations

- **Limited User Feedback:** The absence of real-time customer feedback during the development phase restricted the ability to fine-tune the user interface and core functionalities.

- **Resource Constraints:** The project faced constraints in terms of time which limited the incorporation of advanced features and functionalities.

- **API Dependencies:** The reliance on free APIs such as OpenStreetMap posed challenges in terms of performance and advanced capabilities compared to paid services.

5.2.2 Future Recommendations

- **Performance Enhancements:** Explore paid APIs or advanced geolocation services to improve accuracy, speed, and reliability for better user experience.

- **Continuous Improvement:** Regular updates and feedback recognition should be prioritized to adapt to changing user needs and technological advancements.

References

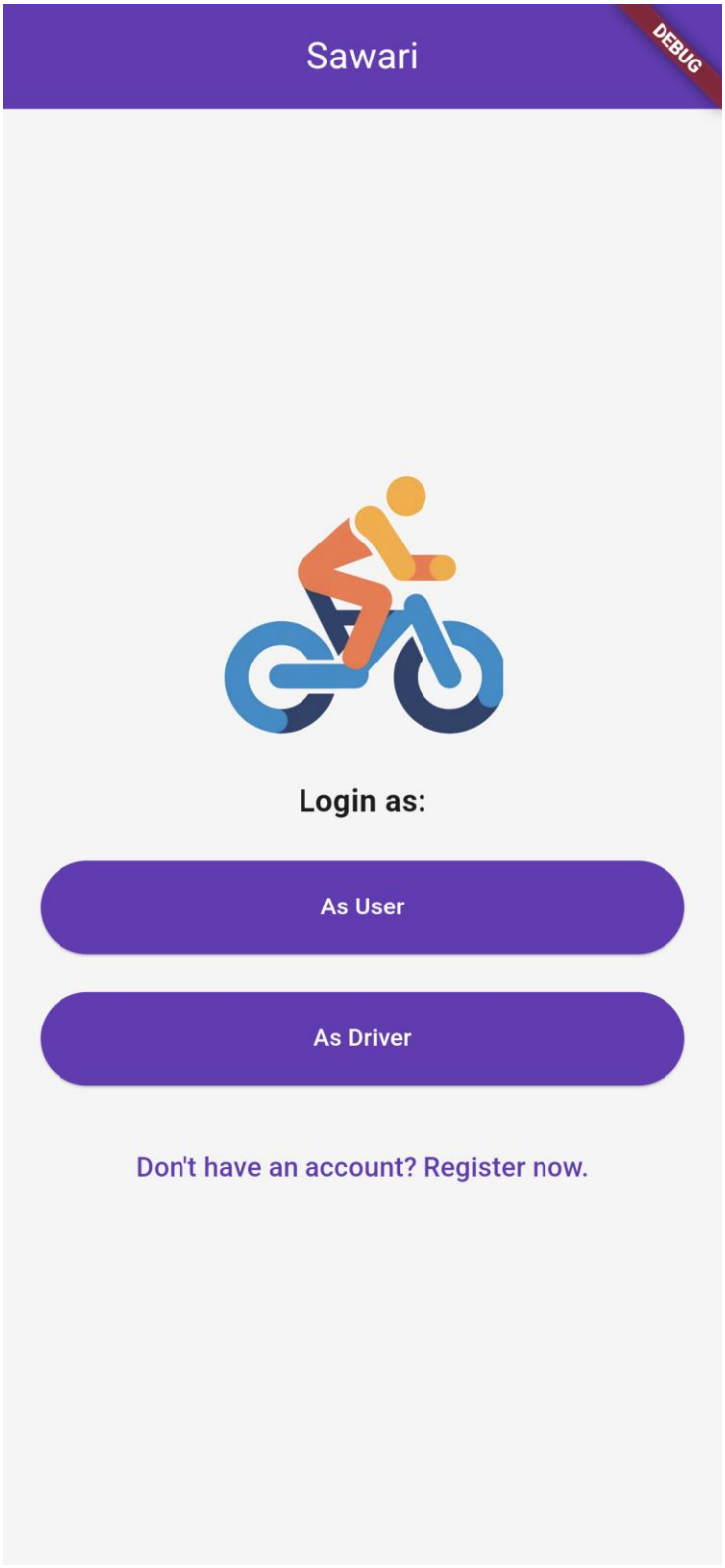
1. Masoud, N., & Jayakrishnan, R. (2017). *Ride-Sharing Applications: Review of Trends, Challenges, and Opportunities*. Transportation Research Board.
2. Cramer, J., & Krueger, A. B. (2016). *Disruptive Change in the Taxi Business: The Case of Uber*. NBER Working Paper.
3. W. Zhang, S. Guhathakurta, J. Fang, and G. Zhang, "Exploring the impact of shared autonomous vehicles on urban parking demand: An agent-based simulation approach," *Sustainable Cities and Society*, vol. 55, 2021.
4. Sefinehtesfa34. *Ride Share App: A ride-sharing application developed using Flutter*. GitHub repository.
   Available: https://github.com/sefinehtesfa34/RideShare-app.
5. *ChatGPT-4o*. Open AI.
6. S.P. Ghimire, "Public Transport in Nepal: What are real problems? What's the solution?", OnlineKhabar.

   Available:https://english.onlinekhabar.com/public-transport-in-nepal-problems.html?utm_source=chatgpt.com
7. A. Alemi, G. Circella, S. Handy, and P. Mokhtarian. "What Influences Travelers to Use Uber? Exploring the Factors Affecting the Adoption of On-Demand Ride Services in California," *Travel Behavior and Society*, vol. 13, pp. 88–104, 2018.
8. T.R. Aryal, M. Ichihashi and S. Kaneko. "How strong is demand for public transport service in Nepal? A case study of Kathmandu using a choice-based conjoint experiment," *Journal of Economic Structures*, vol. 11, 2022.

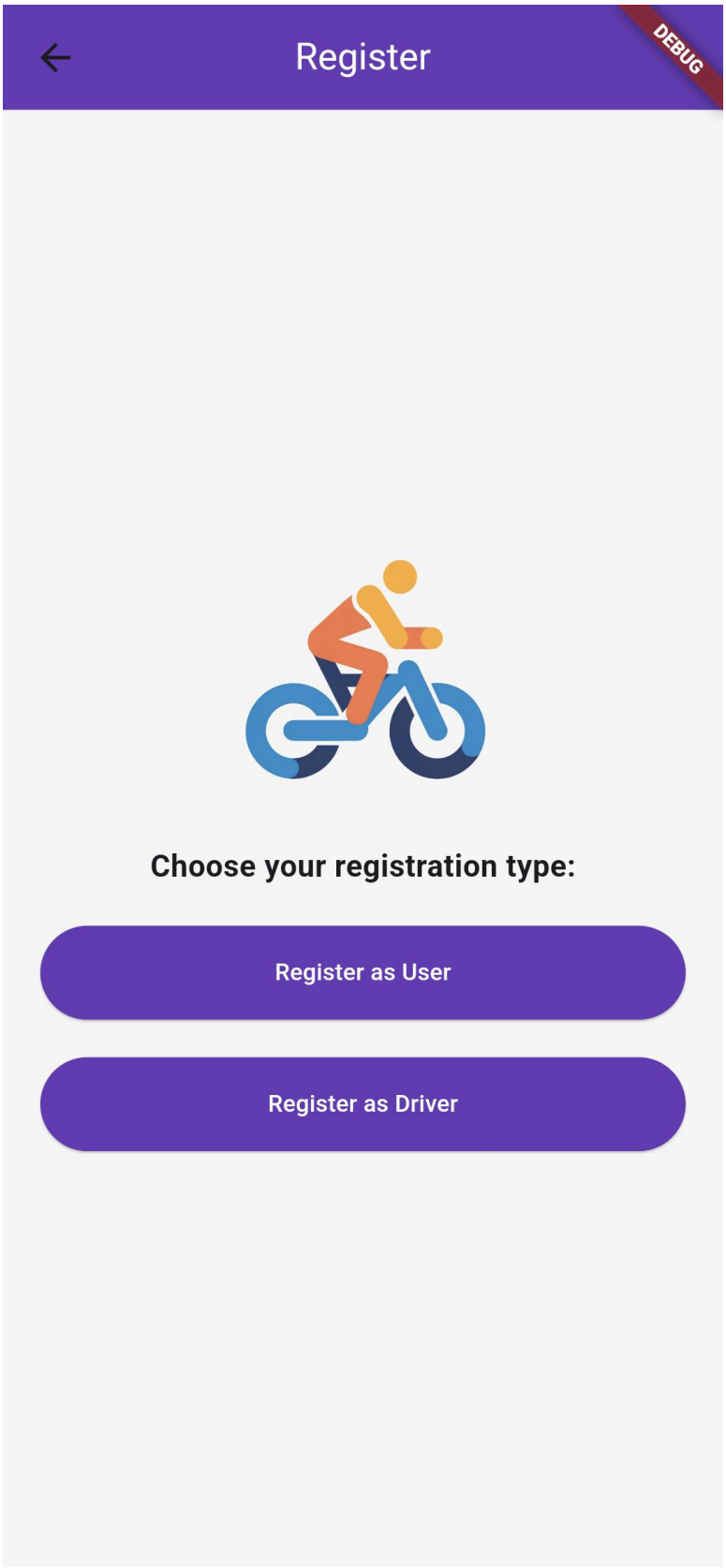
   Available:https://link.springer.com/article/10.1186/s40008-022-00287-3?fbclid=IwZXh0bgNhZW0CMTEAAR6C4N_BkPGDtlUVl5d1Av32zXRgIY0qxSqWeCkpxbGM8tPM-WpA-UrasVGw-Q_aem_-_yggU7Zp5YH91PT8MJWJw.
9. Ahmed-shorbagy. *Ride-Glide*: *A cross-platform ridesharing app built with Flutter*. GitHub repository.
   Available: https://github.com/ahmed-shorbagy/Ride-Glide.

Appendix

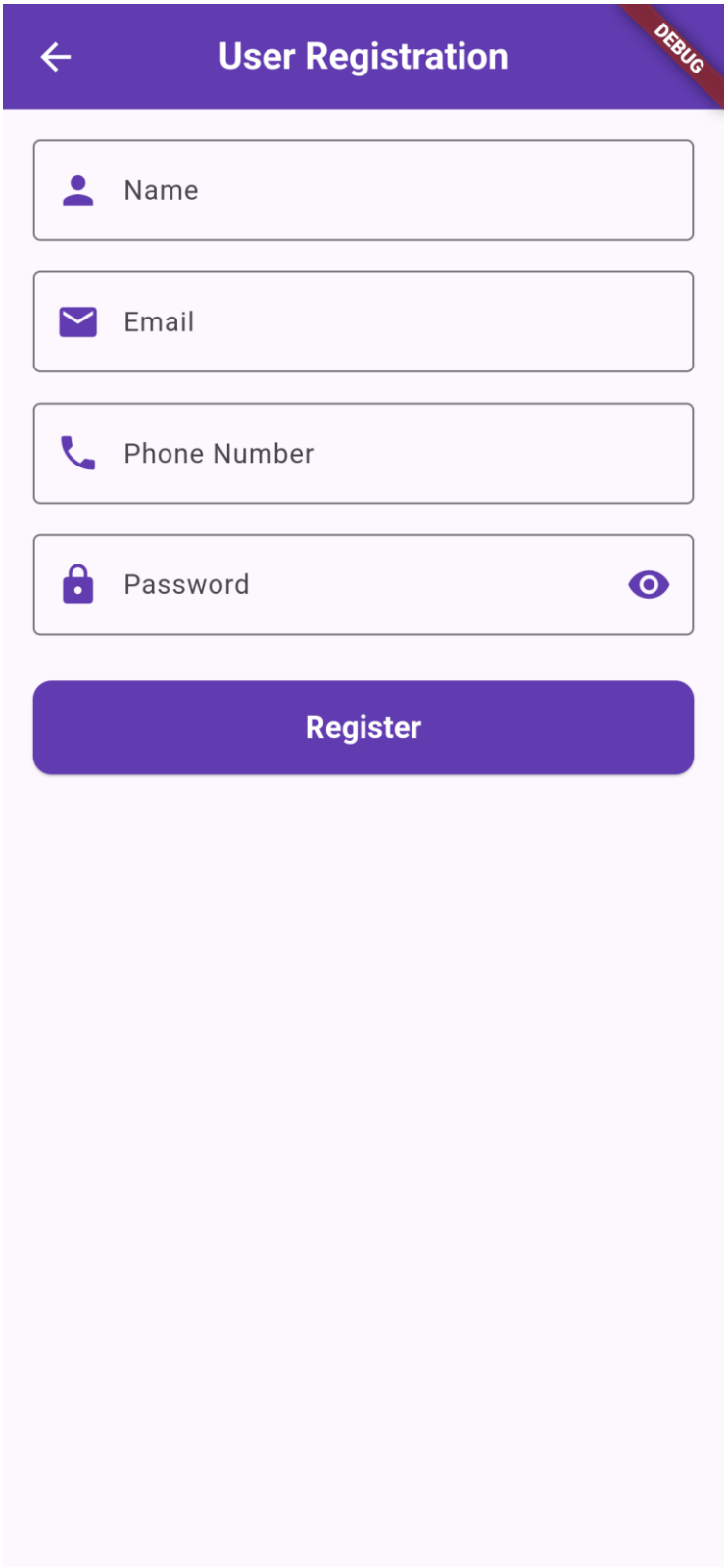
1. Create Account and Login Screen



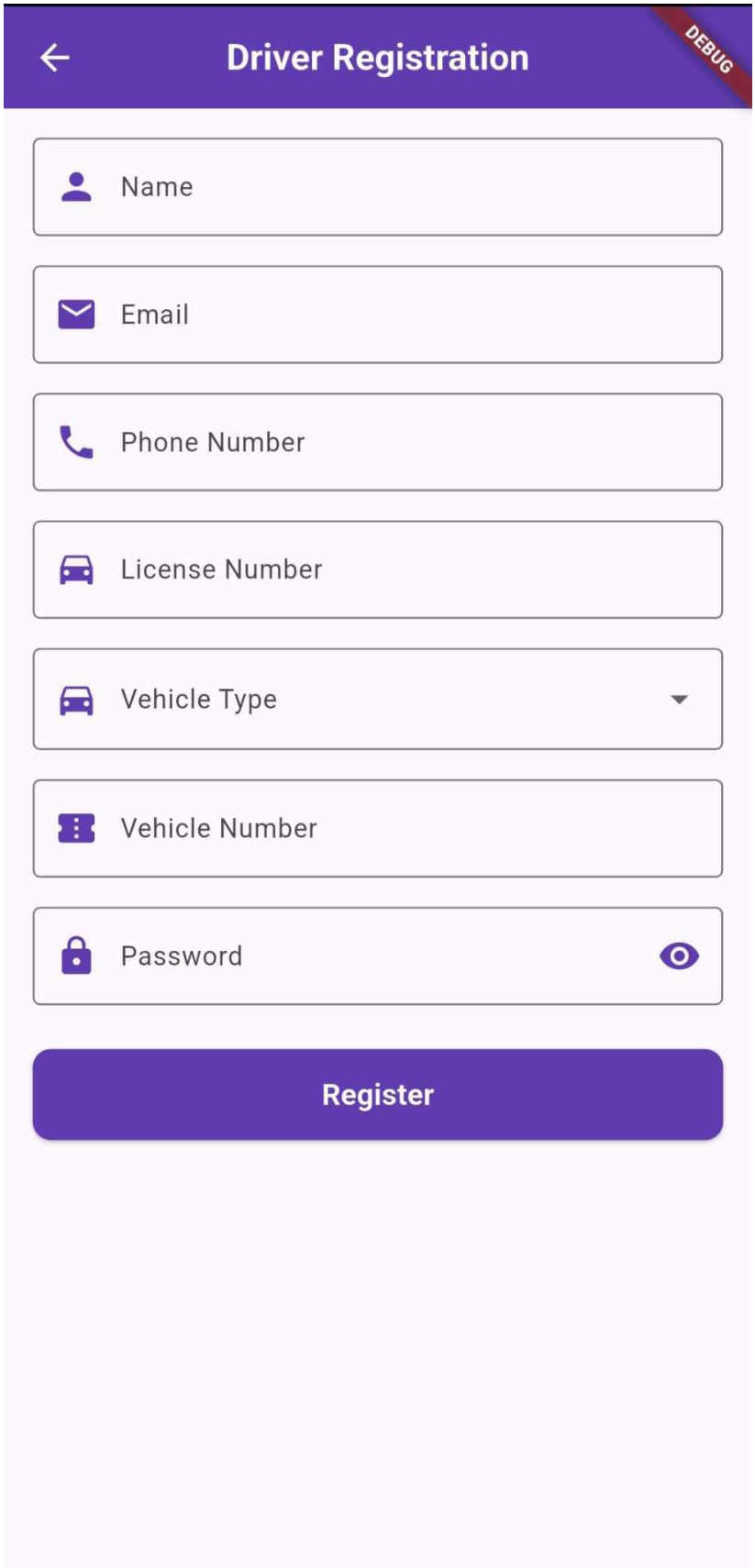
2. Registration Option Screen



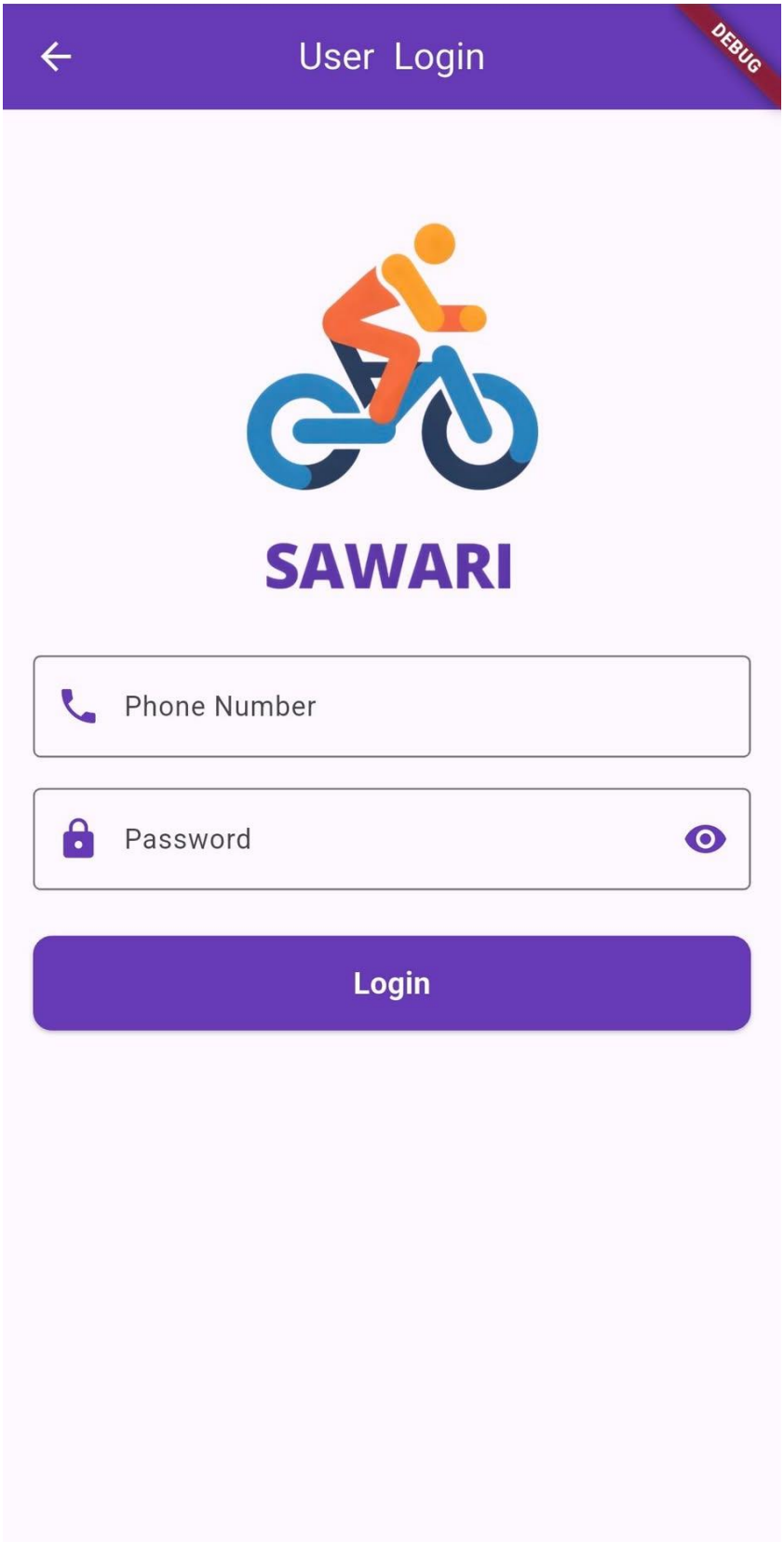
3. User Registration Screen



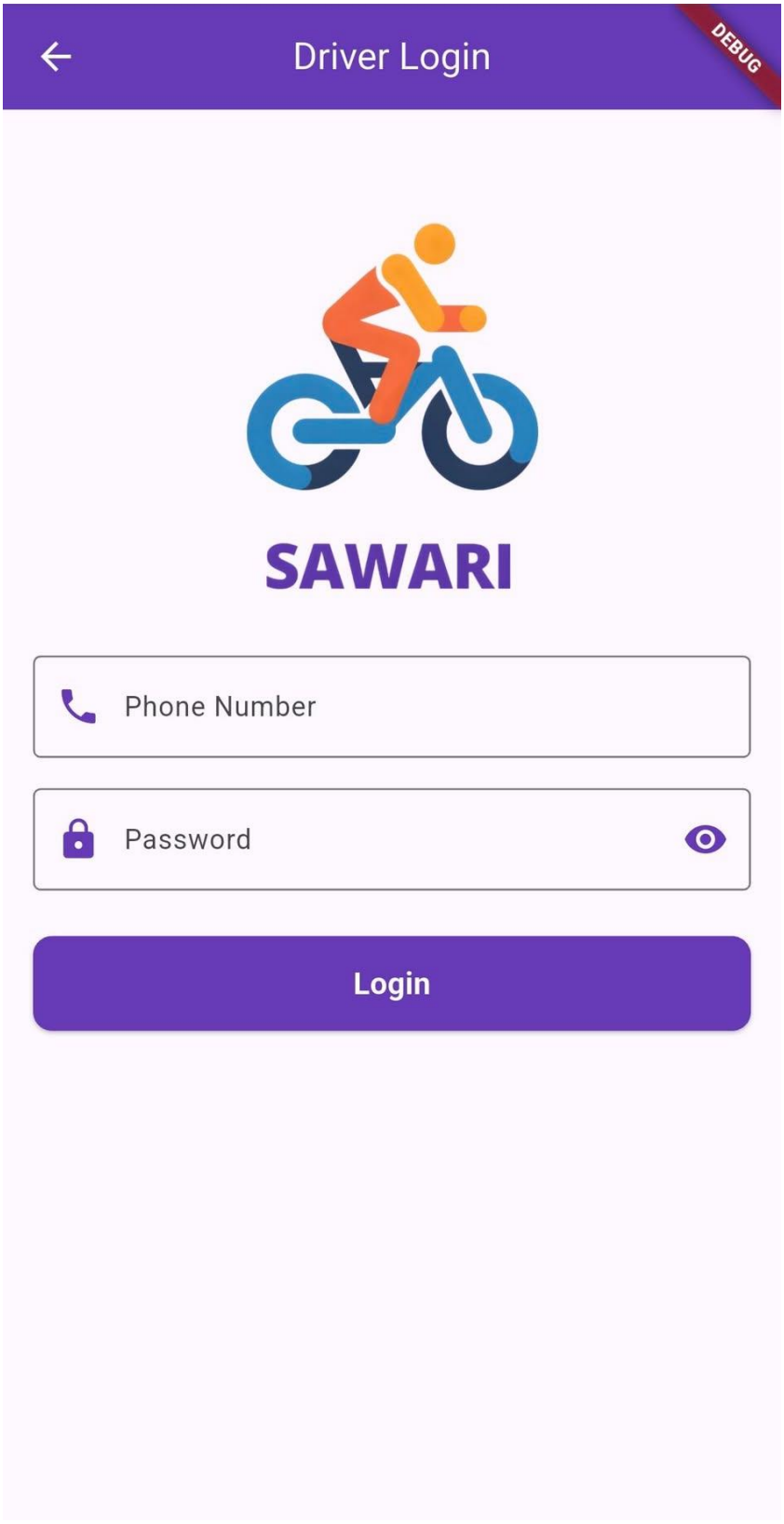
4. Driver Registration Screen



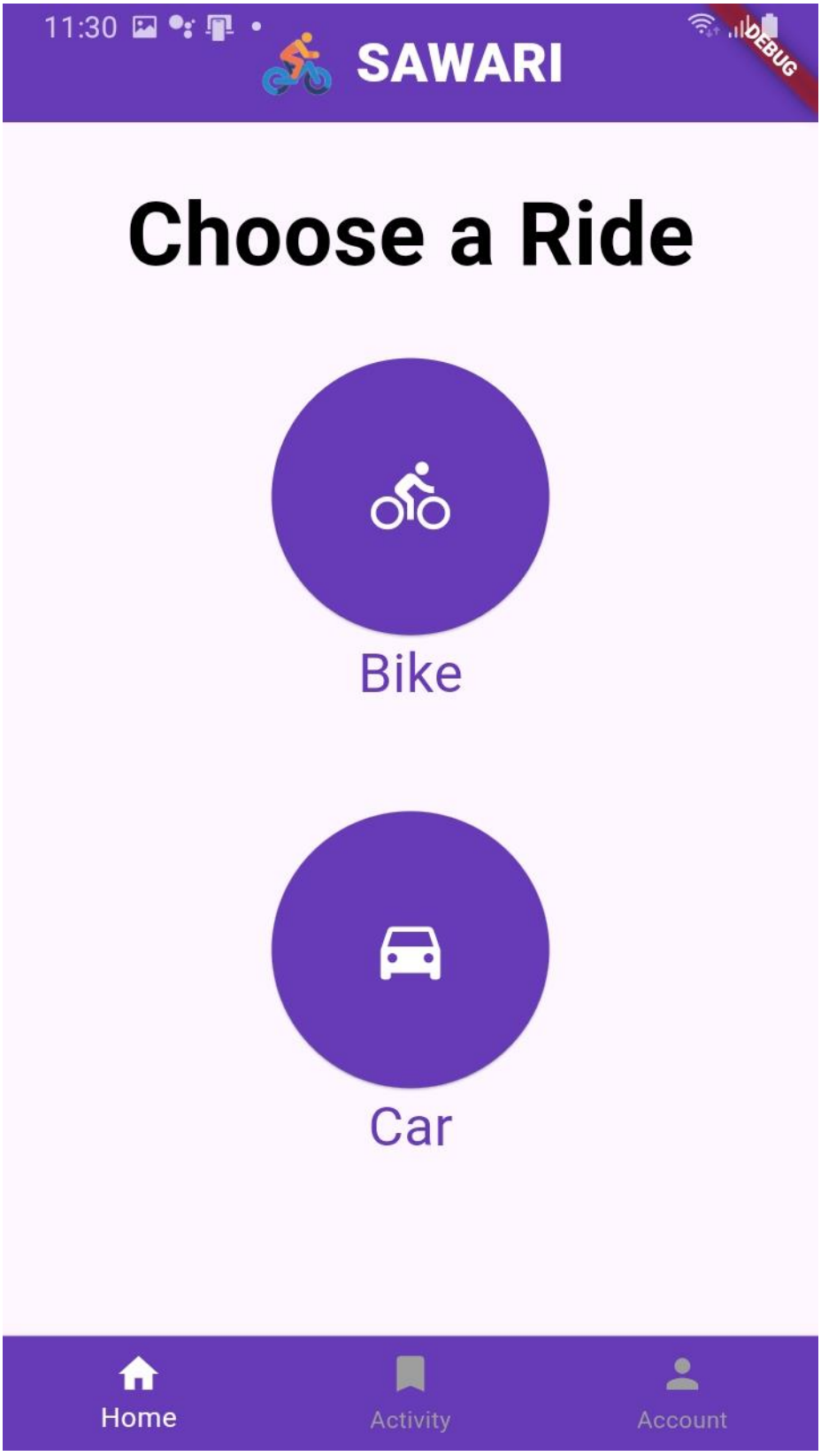
5. User Login Screen



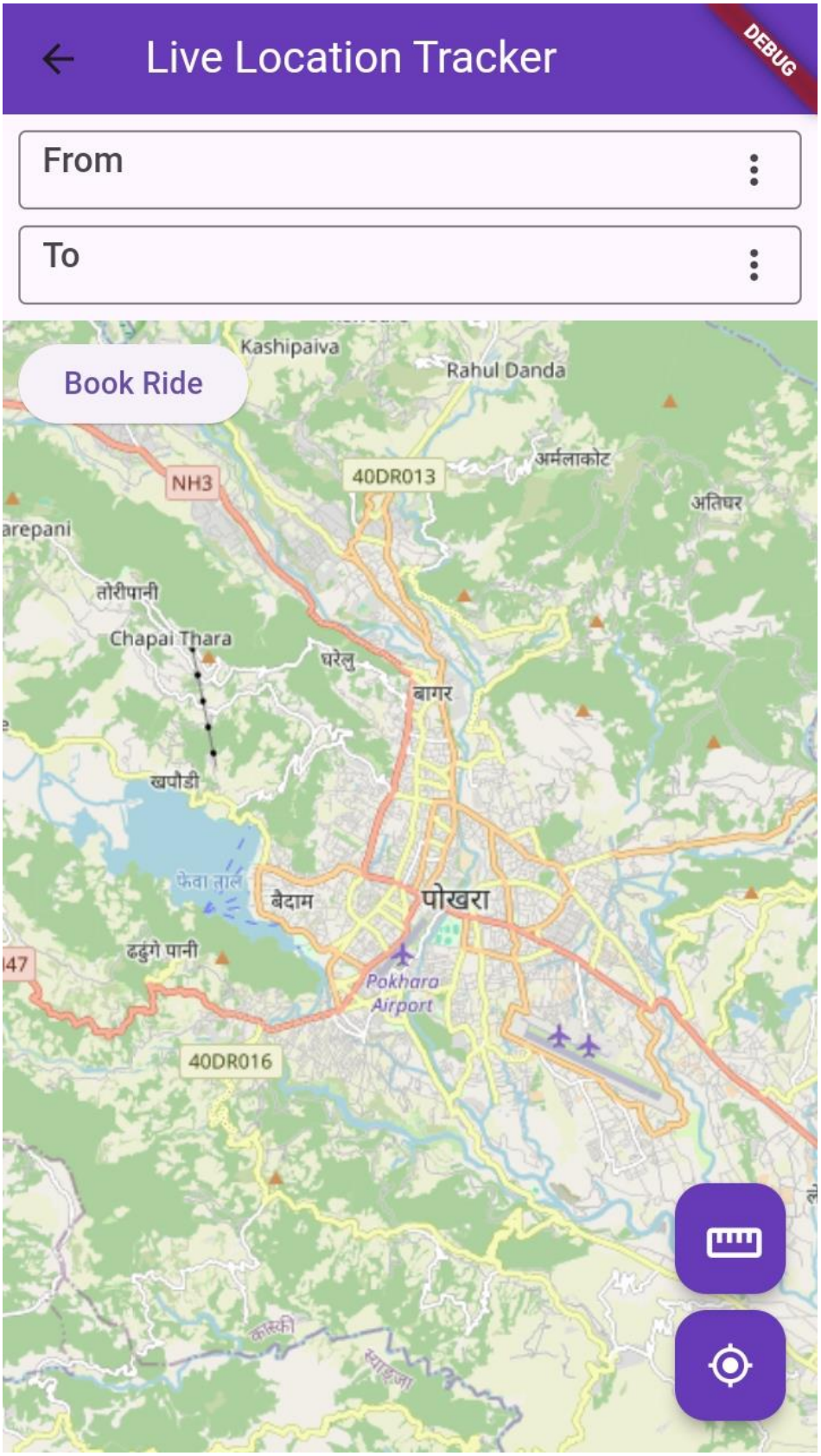
6. Driver Login Screen



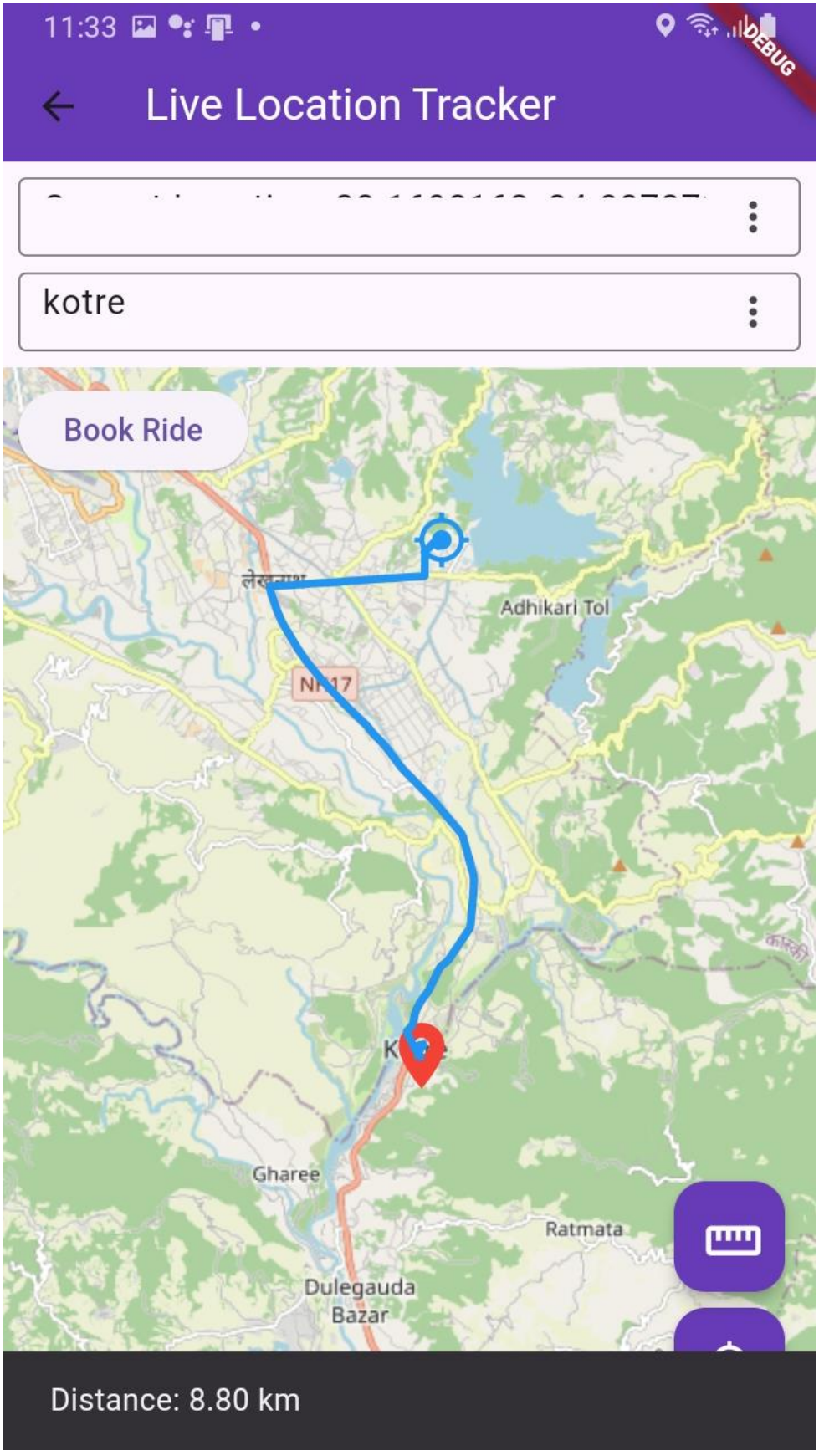
7. User Home Screen



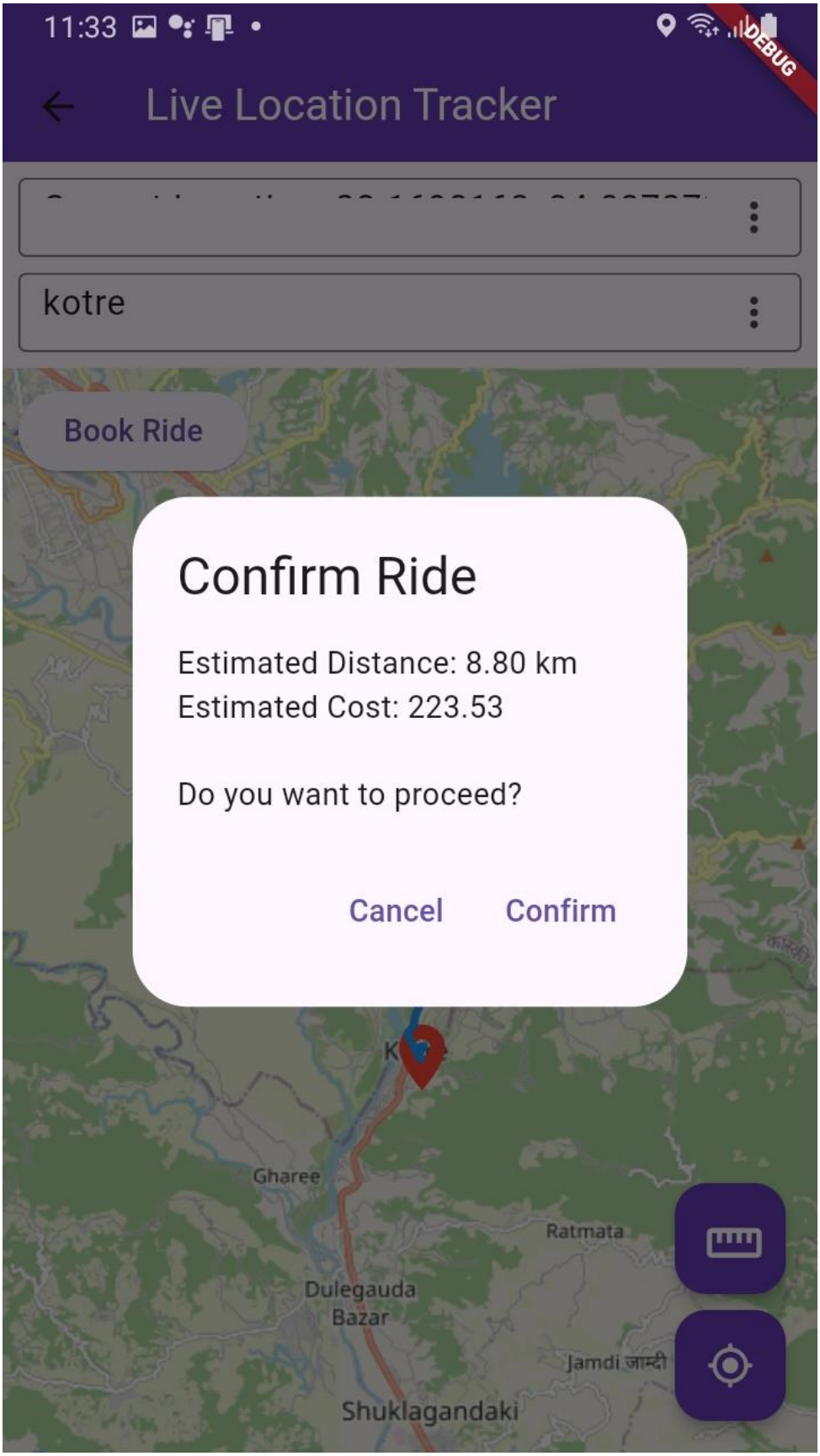
8. User Map Screen



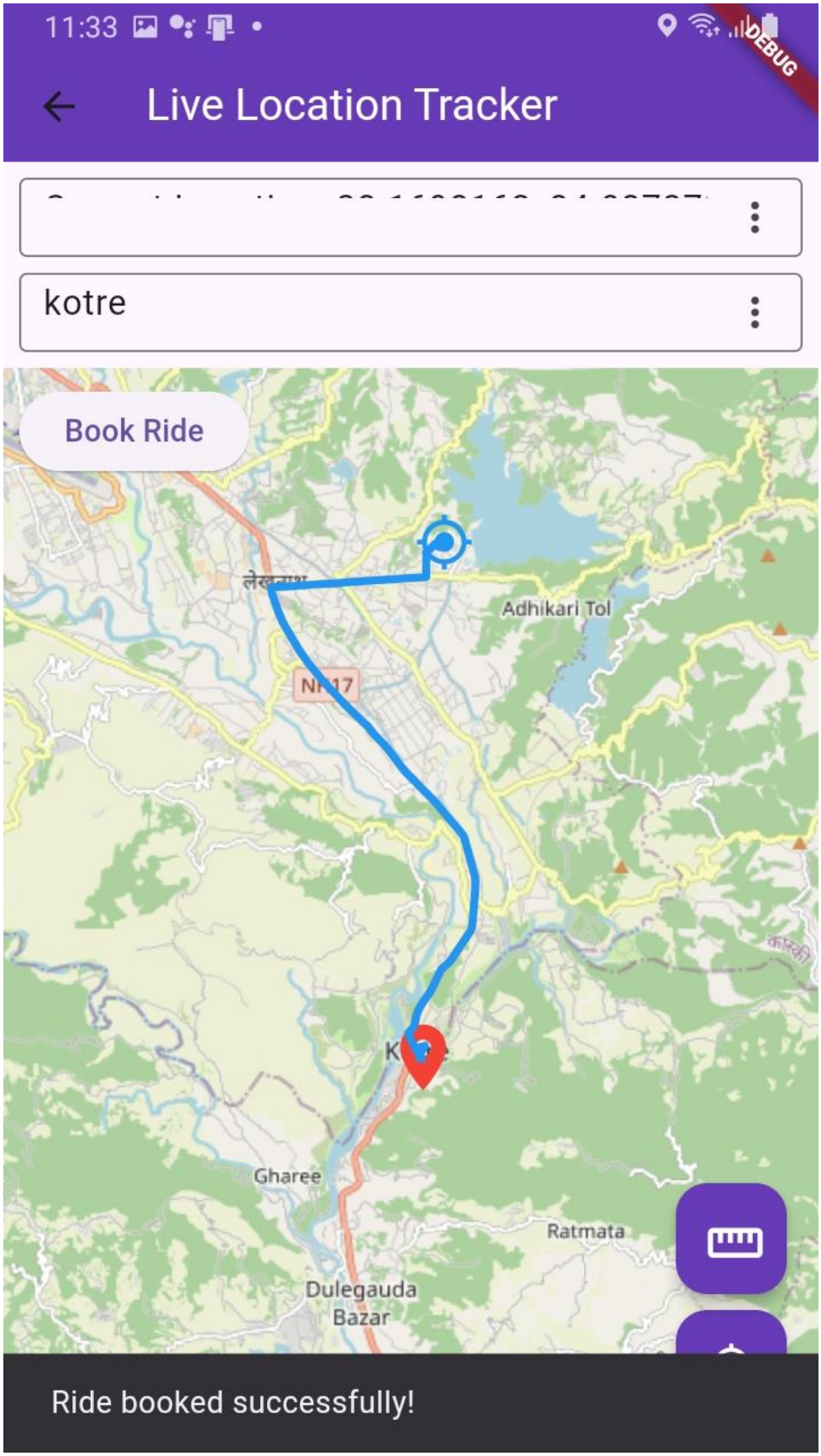
9. Path and Distance to the Destination



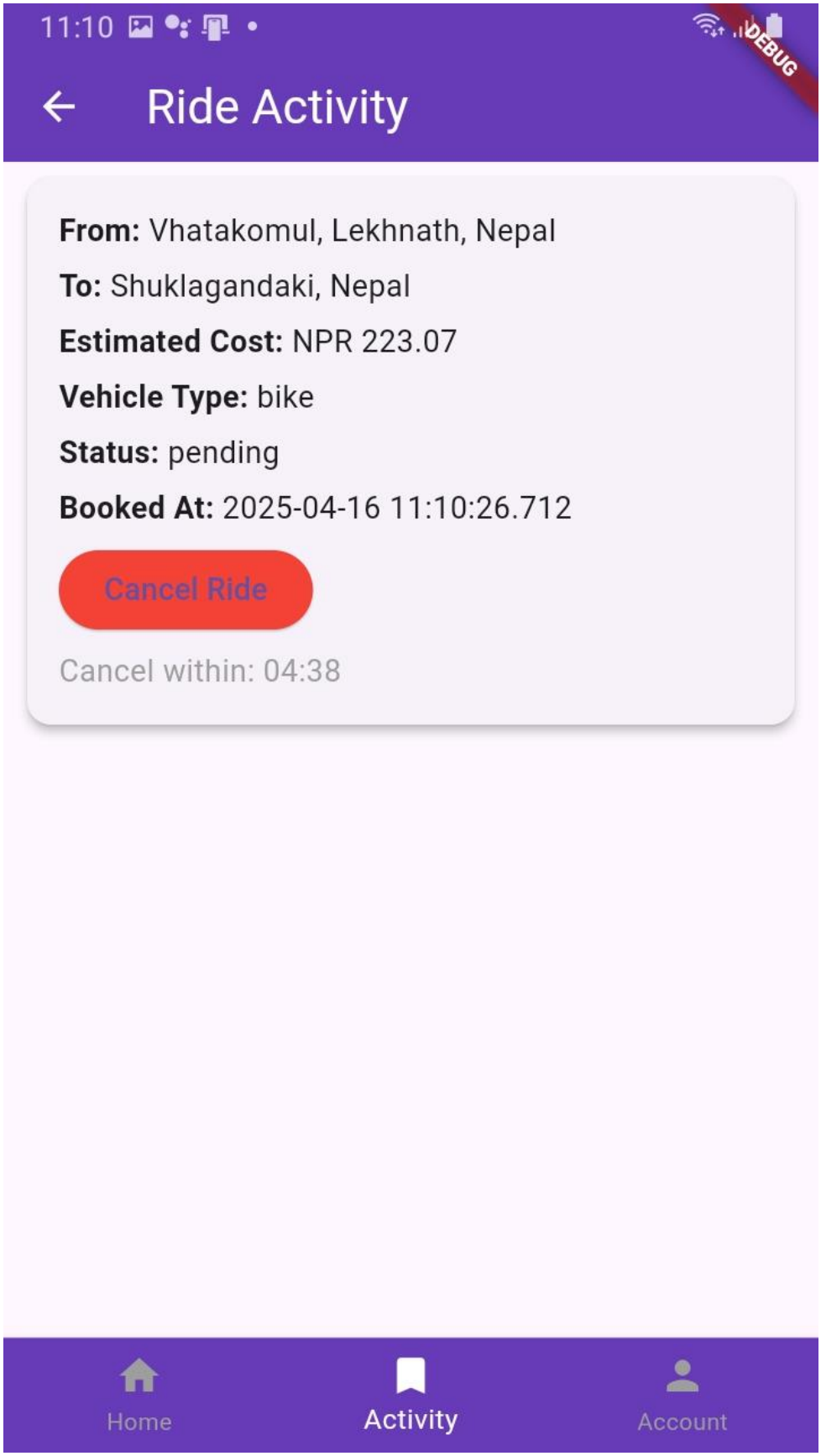
10. Ride Booking Confirmation



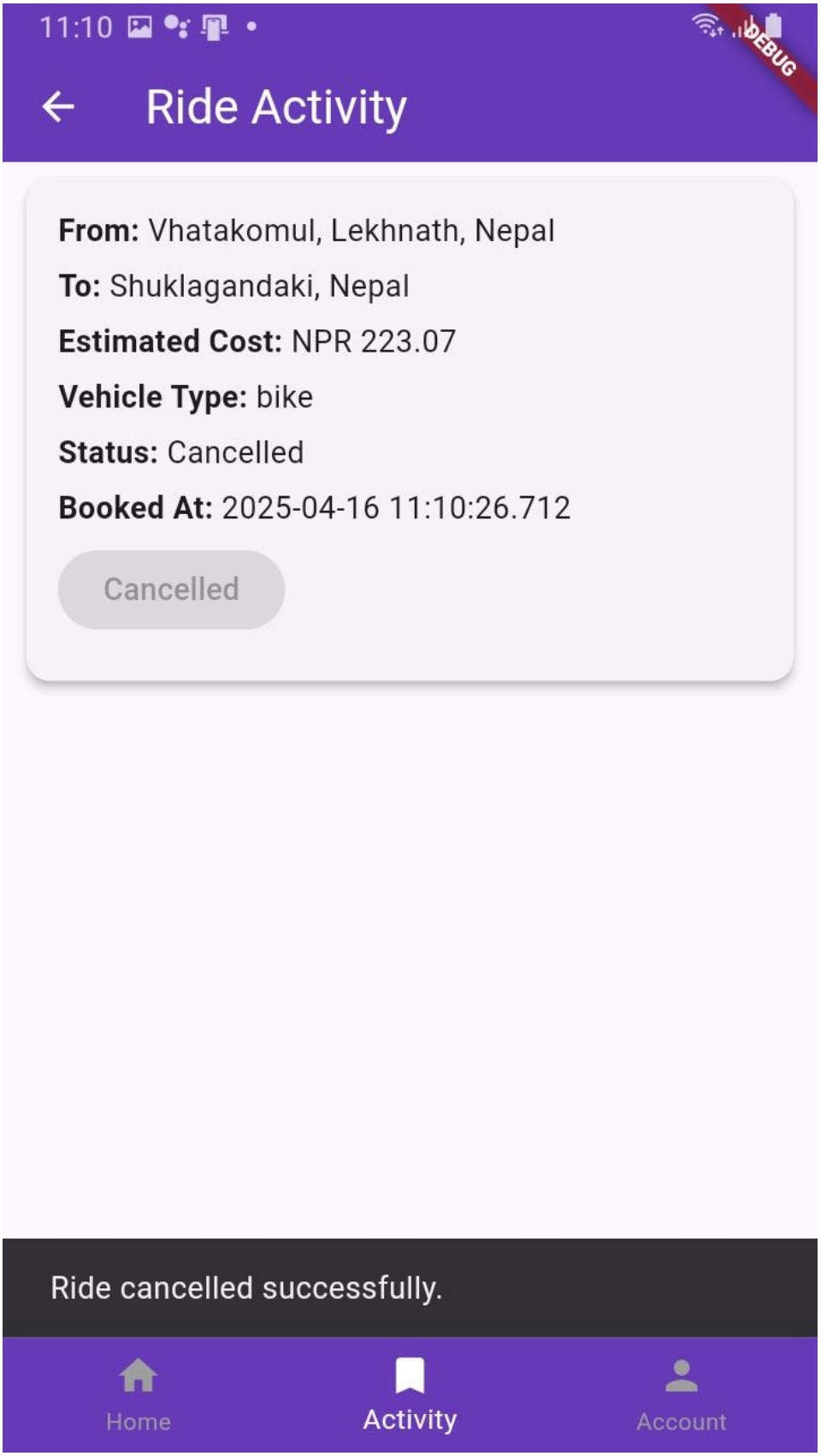
11.Ride Booked



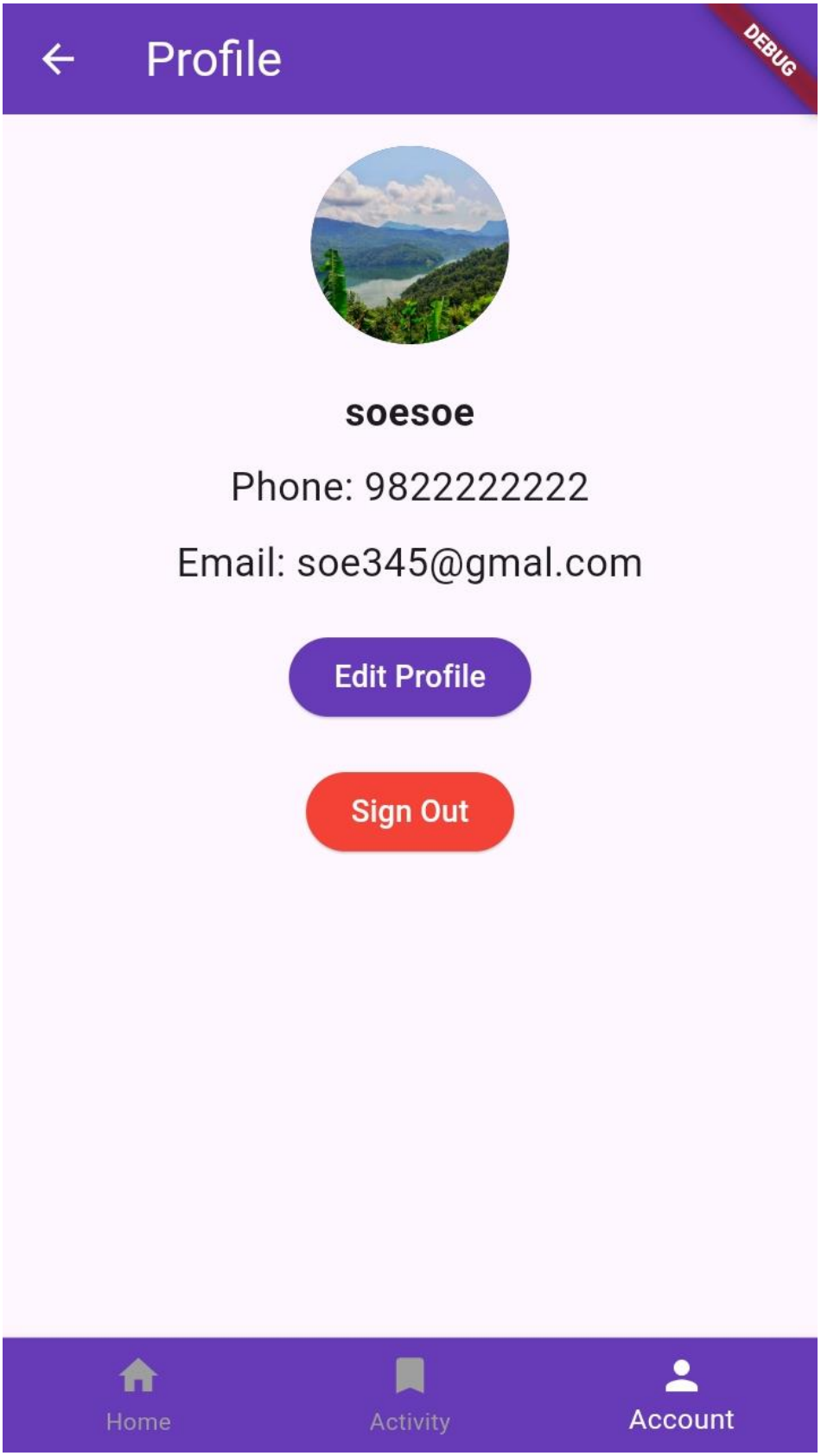
12. User's Ride Activity



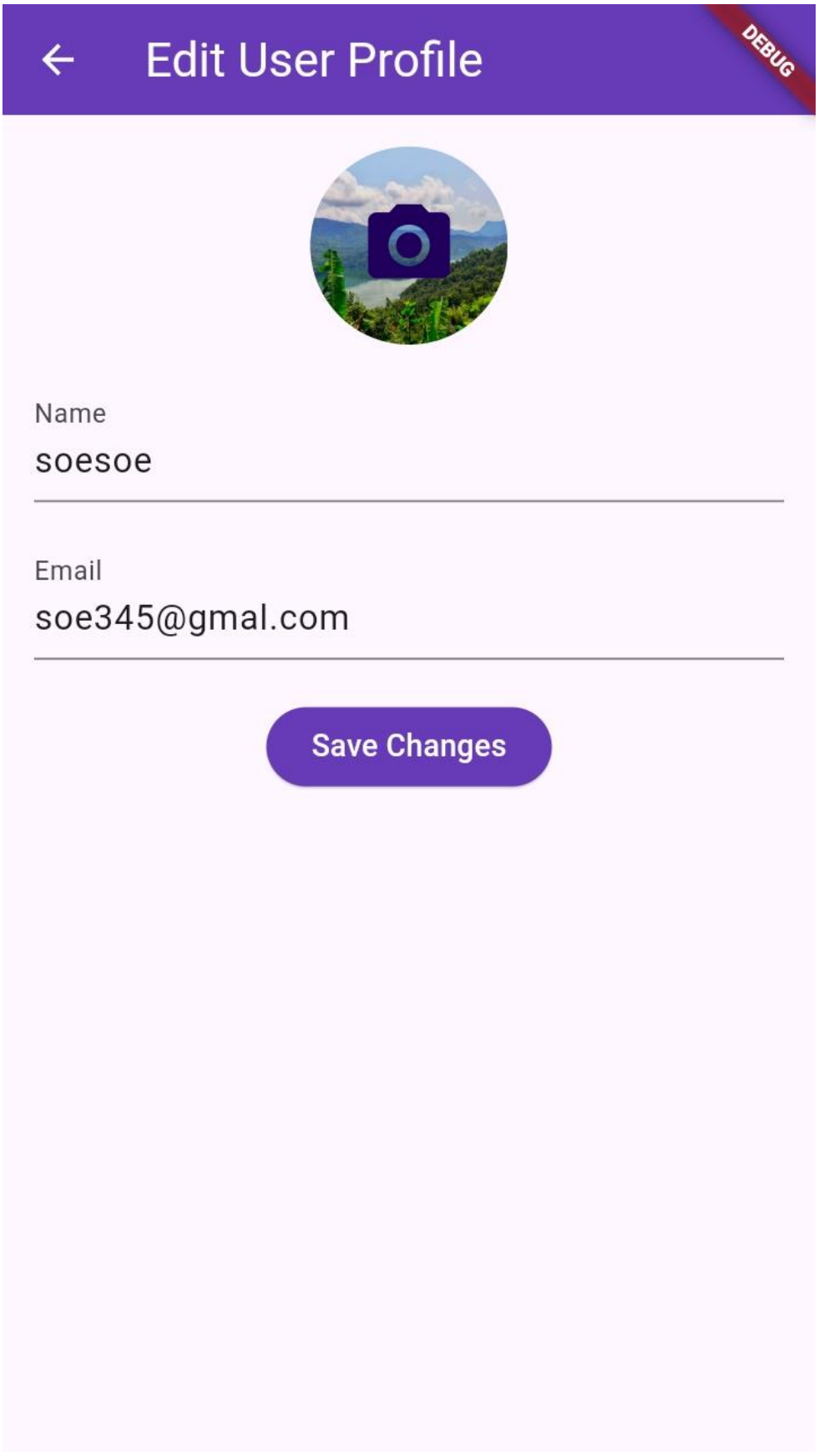
13. Cancel the ride



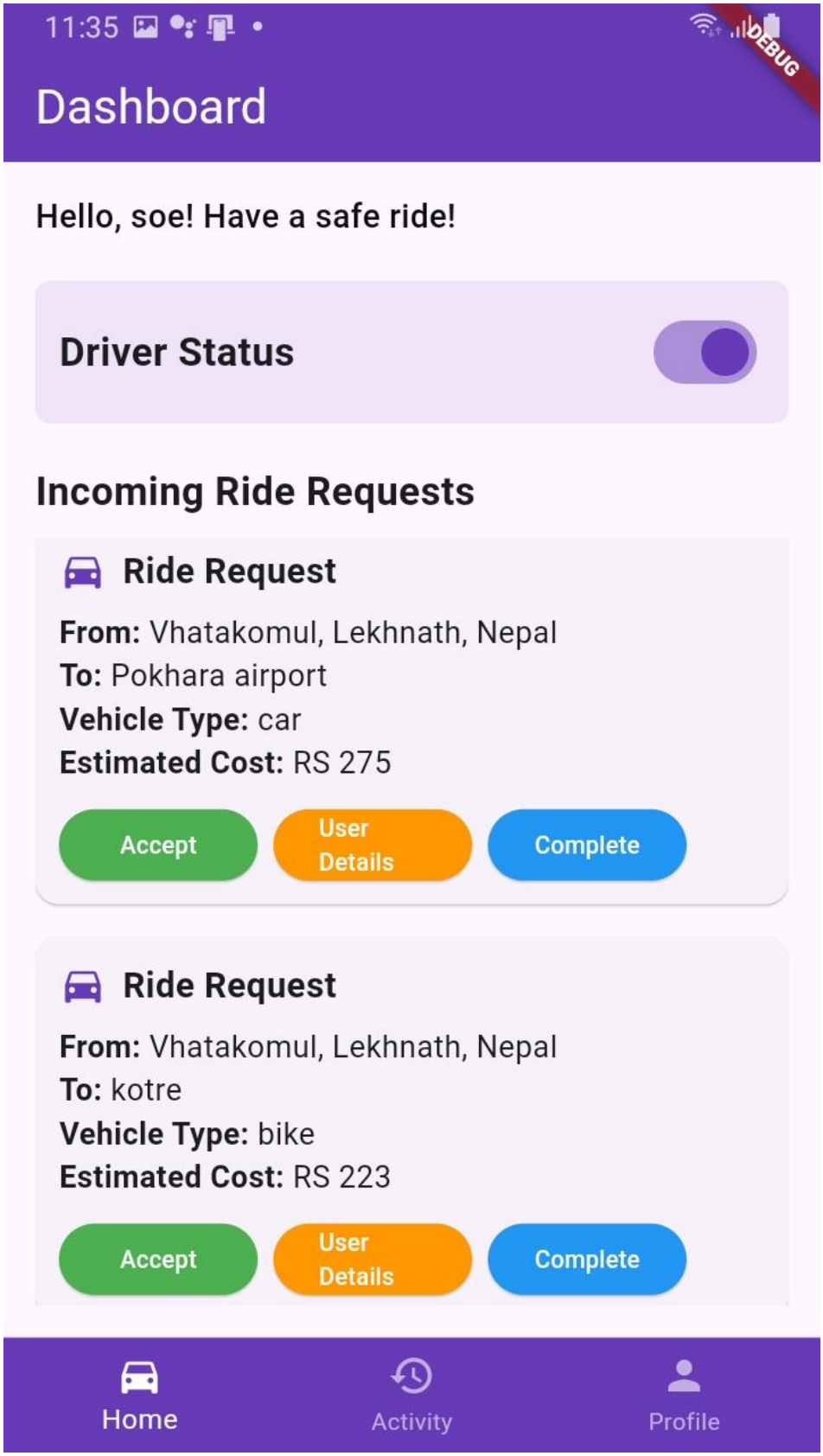
14.User Profile Screen



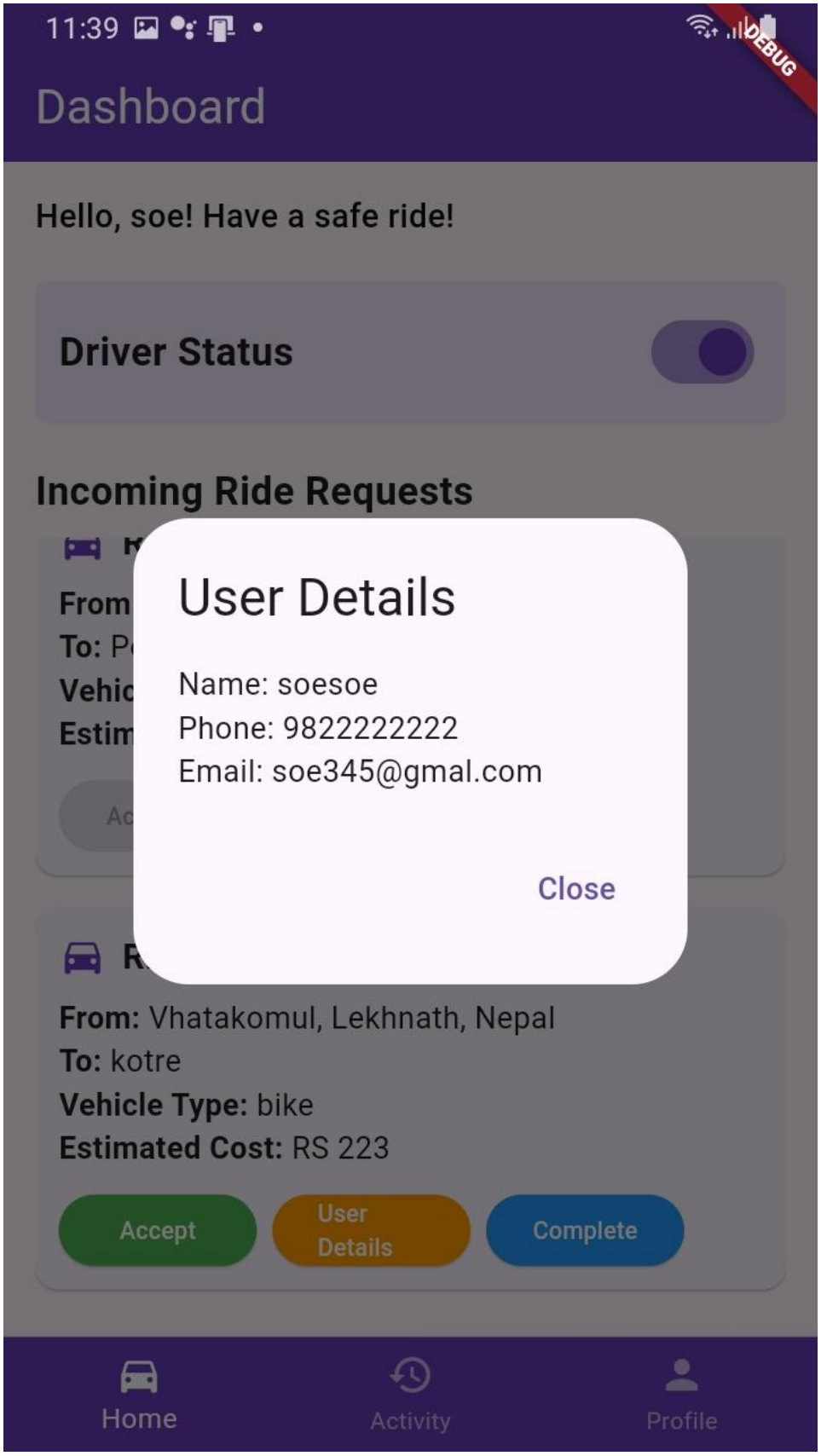
15.User Profile Edit and Update Page



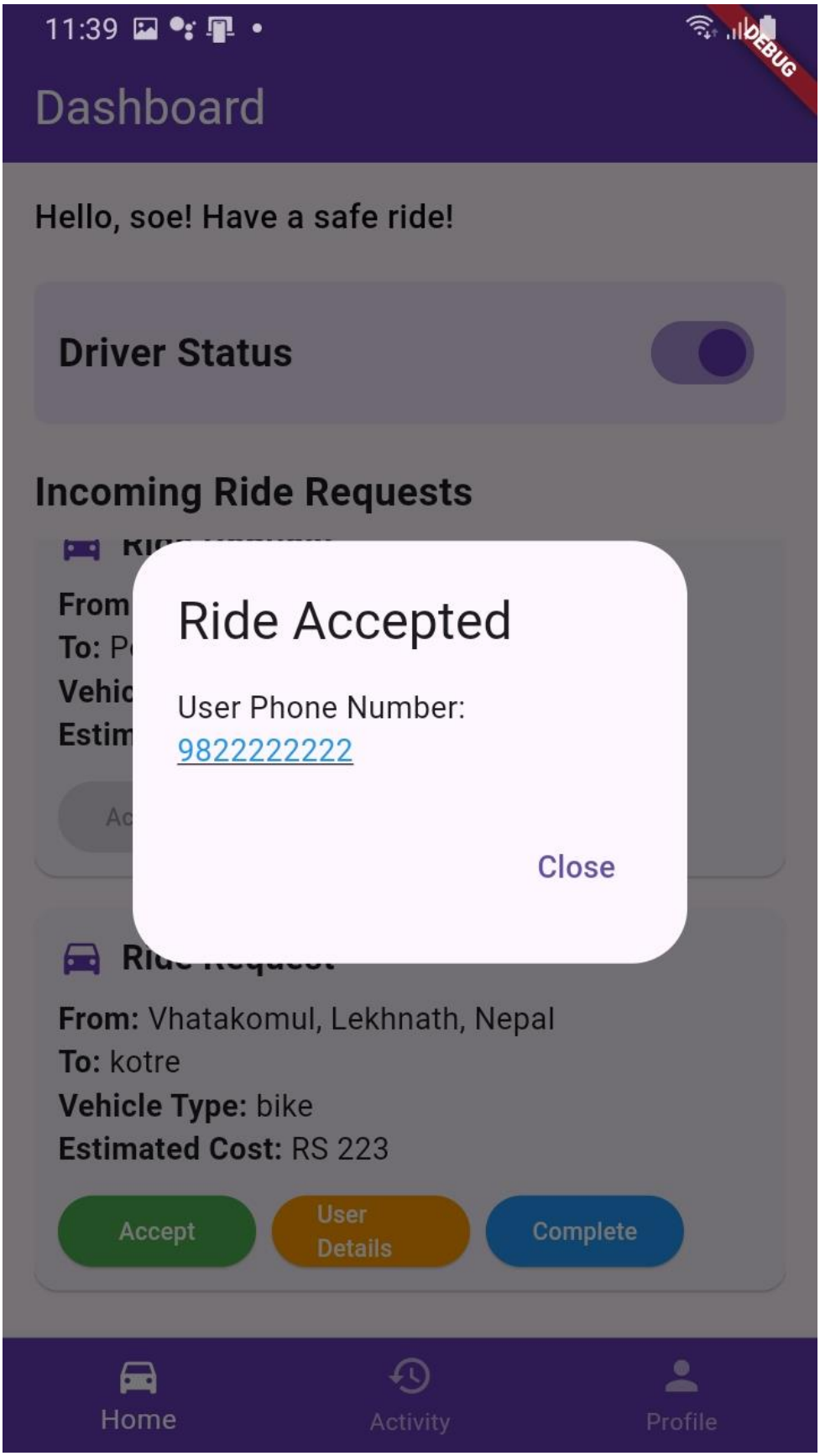
16. Driver Dashboard Screen



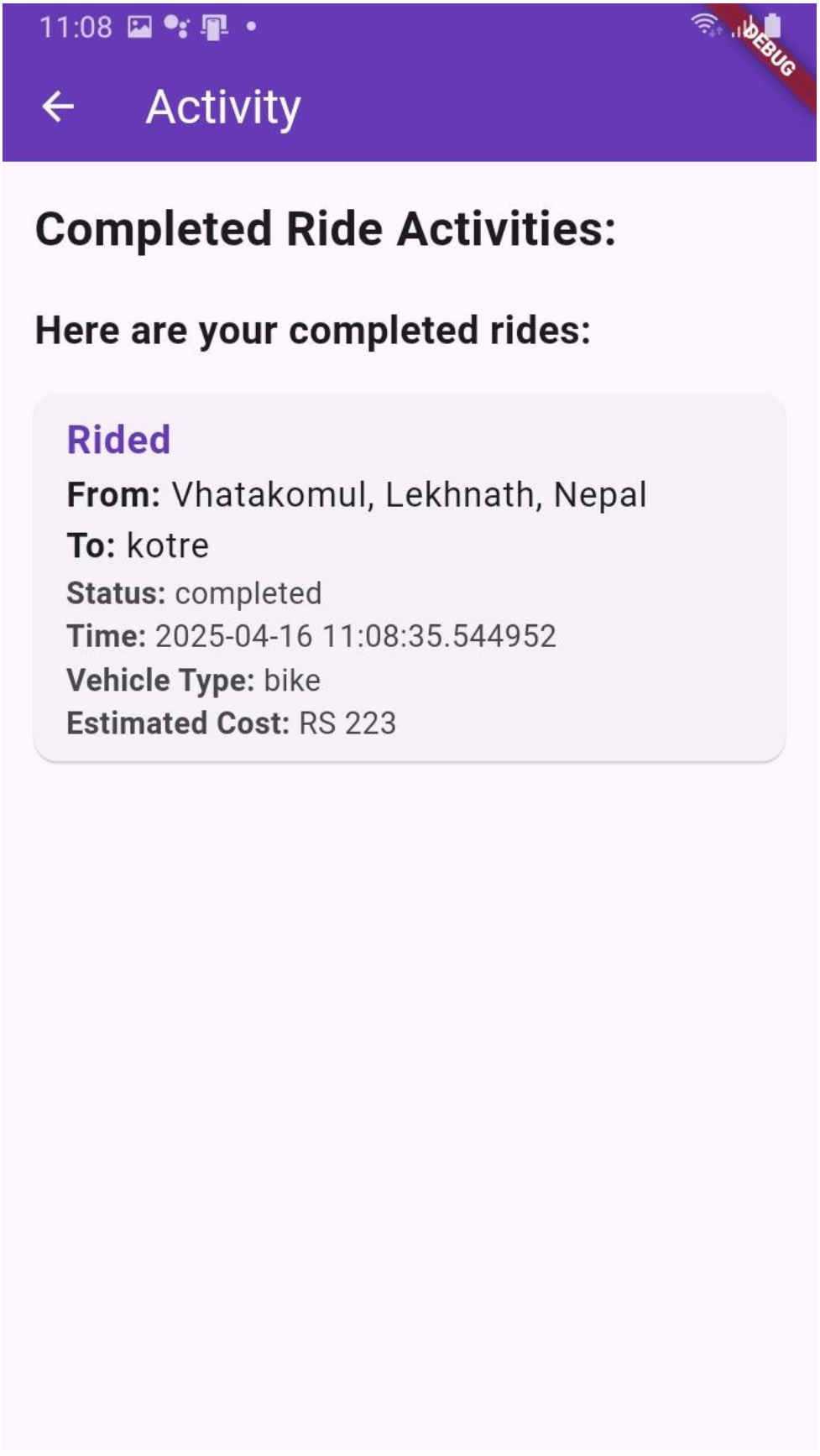
17.User Details for Driver



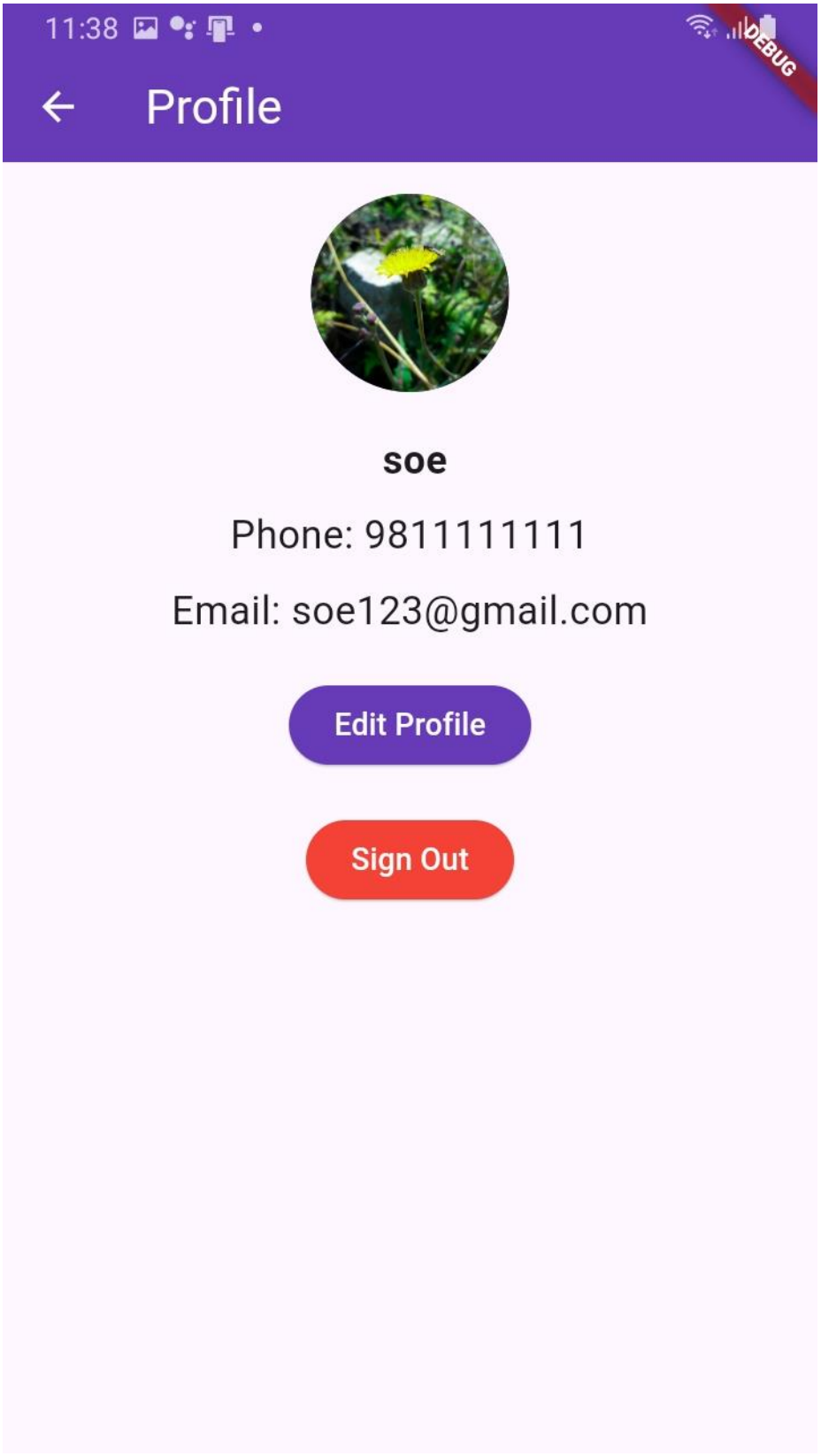
18. Accept Ride Request



19. Driver's Activity



20. Drivers Profile Screen



21. Edit Drivers Profile Screen