A
Project Report
On
“Apni Riksha"

Prepared by

KAUSTAV DAS (23DCE020)
KURESH GARBADA (23DCE032)
BHAVY GONDHAVIYA (23DCE036)
HARSH GOSWAMI (23DCE037)

Under the guidance of

Mrs Bhavika Patel

Assistant Professor

A Report Submitted to

Charotar University of Science and Technology

for Partial Fulfillment of the Requirements for the

4th Semester Project – 2 (CE268)

Submitted at

Department of Computer Engineering

Devang Patel Institute of Advance Technology and Research (DEPSTAR)

Faculty of Technology & Engineering (FTE), CHARUSAT

At: Changa, Dist: Anand – 388421

April 2025

CHARUSAT
CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

CERTIFICATE

This is to certify that the report entitled "**Apni Riskha**" is a bonafied work carried out by **Kaustav Das (23DCE020), Kuresh Gardbada (23DCE032), Bhavy Godhaviya (23DCE036), Harsh Goswami (23DCE037)** under the guidance and supervision of **Prof. Mrs Bhavika Patel** for the subject **CE268 – Project 2 (CE)** of 4th Semester of Bachelor of Technology in Devang Patel Institute of Advance Technology & Research (DEPSTAR) at Faculty of Technology & Engineering – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate himself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

Mrs. Bhavika Patel
Assistant Professor
Department of Computer Engineering,
DEPSTAR, Changa, Gujarat.

Dr. Dweepna Garg
Head of Department & Assistant Professor
Department of Computer Engineering
DEPSTAR, CHARUSAT, Changa, Gujarat.

Devang Patel Institute of Advance Technology and Research

At: Changa, Ta. Petlad, Dist. Anand, PIN: 388 421. Gujarat

DECLARATION BY THE CANDIDATE

We hereby declare that the project report titled "**Apni Ricksha**" submitted by us to Charotar University of Science and Technology (CHARUSAT), Gujarat, for Partial Fulfillment of the Requirements for the Bachelor of Technology (BTech) 4th Semester Project in Computer Engineering under the guidance of Assistant Prof. Bhavika Patel. I further declare that the work carried out and documented in this project report has not been submitted anywhere else either in part or in full and it is the original work, for the award of any other degree or diploma in this institute or any other institute or university.

Kaustav Das (23DCE020)
Kuresh Garbada(23DCE032)
Bhavy Godhaviya(23DCE036)
Harsh Goswami(23DCE037)

Prof. Bhavika Patel

Assistant Professor

Department of Computer Engineering,

DEPSTAR, CHARUSAT, Changa, Gujarat.

ABSTRACT

Apni Ricksha is an online rickshaw booking platform designed to help local rickshaw drivers digitalize their services and efficiently manage their rides. Built using web technologies, the platform provides users with seamless experience for booking rickshaws, viewing fare estimates, and providing feedback. Drivers can manage their availability, accept ride requests, and update their status through a simple interface. Although real-time tracking and online payments are not currently implemented, the system is designed to be scalable for future enhancements.

ACKNOWLEDGEMENT

We, the developers of the web application "**Apni Ricksha**," with immense pleasure and commitment, would like to present the project assignment. The development of this project has provided us with a wide opportunity to think, implement, and interact with various aspects of **management skills** as well as **emerging technologies**. It has been an enriching experience that has helped us enhance our **technical** and **problem-solving abilities**.

Every successful completion of work stands on the constant **encouragement**, **goodwill**, and **support** of friends, colleagues, and people around. We would like to express our **gratitude** to the individuals who extended their valuable time, full **support**, and **cooperation** in developing this project. Without their help, this project would not have been possible.

We express our deep sense of **gratitude** towards our **Head of the CE Department**, Dr. **Dweepna Garg**, and our project guide, Assistant Prof. **Bhavika Patel**, for their unwavering support during the entire study and development phase. Their **guidance** and **encouragement** motivated us to work hard, adopt **new technologies**, and complete the project successfully.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

1.1 PROJECT DEFINATION

Apni Ricksha is a **web application** designed to simplify the process of booking **rickshaws** for **residents**, especially **students** in **Changa**. The application allows users to **book a rickshaw** by simply entering their **ride details**, such as the **pickup** and **drop locations**. Once the details are entered, users receive **fare estimates** and can submit **feedback** about the ride. The platform provides an **easy-to-use interface** for **passengers** to access **rickshaw services**, making **local travel** more **convenient** and **efficient**. **Drivers** can manage their **availability** and accept **ride requests** through a simple **dashboard**. **Apni Ricksha** aims to **bridge the gap** between **passengers** and local **rickshaw drivers**, offering a **seamless** and **accessible ride-booking solution**.

1.2 OBJECTIVE

- To provide an **easy** and **efficient** platform for **students** and **local residents** in **Changa** to **book rickshaws** online.
- To offer a **user-friendly interface** for both **passengers** and **drivers** to manage **bookings** and **availability**.
- To provide **fare estimates** before booking to ensure **transparency** and **convenience** for users.
- To simplify the **ride booking process** by allowing users to submit their **pickup** and **drop-off locations** via a simple form.
- To gather and display **user feedback** to ensure **service quality** and enhance the overall **user experience**.
- To enable **drivers** to manage their **ride requests** and **availability** with a simple dashboard.

1.3 TOOLS & TECHNOLOGIES

Frontend Technologies:

- **HTML, CSS, Bootstrap** – For structuring and styling the website.
- **JavaScript** – For adding interactivity and managing frontend logic.
- **React.js** – For building a dynamic and responsive user interface.

Backend Technologies:

- **PHP** – For server-side execution and handling requests.
- **MySQL** – For managing and storing database records.

Database:

- **MySQL** – For storing user and ride data.

Additional Tools:

- **XAMPP** – For local server hosting and database management.
- **VS Code** – For code development and debugging.
- **GitHub** – For version control and collaboration.

CHAPTER 2: DESCRIPTION

- **Apni Ricksha** is an easy-to-use **web application** designed to help **students** and **local residents** in **Changa** easily book **rickshaws** online.
- Users need to fill in basic ride details like **pickup** and **drop-off locations**, allowing them to easily book a **rickshaw**.
- The platform provides **fare estimates** before booking, ensuring transparency and **convenience** for the users.
- **Feedback** can be submitted by users to help improve the service and maintain **quality control**.
- **Drivers** can manage their **availability** and accept **ride requests** through a simple, user-friendly **dashboard**.
- The platform is designed to be **mobile-friendly**, making it accessible on both smartphones and computers.
- **Data security** is ensured as users are required to **create an account** and set a **password** to access their ride details and feedback.
- While **real-time tracking** and **online payments** are not yet implemented, the system is **scalable** for future enhancements.

CHAPTER 3: SOFTWARE & HARDWARE REQUIREMENTS

3.1 SOFTWARE REQUIREMENTS

Frontend Technologies:

- **HTML, CSS, Bootstrap** – Used for structuring and styling the web pages.
- **JavaScript** – Handles client-side logic and interactivity.
- **React.js** – For building a dynamic and responsive user interface.

Backend Technologies:

- **MongoDB** – For managing and storing database records.

Database:

- **MongoDB** – A relational database to store user, ride, and driver details.

Development Tools:

- **Visual Studio Code (VS Code)** – The primary IDE for writing and debugging code.
- **GitHub** – For version control and collaborative development.

3.2 HARDWARE REQUIREMENTS

Development Environment (Minimum Requirements):

- **Processor**: Intel Core i3 (or equivalent)
- **RAM**: 8 GB
- **Storage**: 256 GB HDD/SSD
- **Operating System**: Windows 10 / macOS / Linux
- **Internet Connection**: Required for accessing external APIs, cloud services, and testing the app.

Deployment Environment (Recommended Requirements):

- **Processor**: Intel Core i5 or higher (or equivalent)
- **RAM**: 16 GB for better performance and scalability
- **Storage**: 512 GB SSD for faster data processing
- **Operating System**: Linux (Ubuntu) / Windows Server
- **Hosting**: Cloud hosting (AWS, Firebase, or Digital Ocean)
- **Database Server**: Cloud-based MySQL or self-hosted MySQL server

CHAPTER 4: MAJOR FUNCTIONALITY

4.1 DATABASE

1. **Users Collection**
   - Stores **passenger details**, **login credentials**, and **ride history**.
   - **Fields**: user_id, name, email, password (hashed), address, phone, ride_history.

2. **Drivers Collection**
   - Maintains **driver details**, **availability**, and **ride requests**.
   - **Fields**: driver_id, name, email, phone, vehicle_info, availability_status, ride_requests.

3. **Rides Collection**
   - Manages **active and past rides** requested by passengers.
   - **Fields**: ride_id, user_id, driver_id, pickup_location, drop_location, fare, status, timestamp.

4. **Payments Collection**
   - Stores **transaction details** for payments made through the platform.
   - **Fields**: payment_id, ride_id, user_id, amount, payment_method, payment_status, timestamp.

5. **Feedback Collection**
   - Stores **user feedback** for drivers and ride quality.
   - **Fields**: feedback_id, user_id, driver_id, rating, comment, timestamp.

6. **Admin Collection**
   - Contains **admin credentials** and **control parameters** for managing the system.
   - **Fields**: admin_id, email, password (hashed), role.

4.2 HOW TO USE THE Mobile Web?

For Passengers (Booking Rides)

1. **Sign Up / Log In**
   - **New users** can sign up using their **email** and **password**.
   - **Existing users** can log in to access their account and **ride history**.

2. **Browse Available Rickshaws**
   - The **homepage** displays a list of **available drivers**.
   - Users can **select a driver** based on **location** and **availability**.

3. **Enter Ride Details**
   - Users can **input pickup** and **drop-off locations** to get a fare estimate.

4. **Book a Ride**
   - After confirming the details, users **book a rickshaw** for the ride.
   - They can also view **estimated fare** before finalizing the booking.

5. **Make Payment**
   - Payments can be made securely through **UPI**, **credit/debit cards**, or **digital wallets**.

6. **Track the Ride**
   - Once the ride is booked, passengers can **track the ride status** in real-time.
   - Updates include **Ride Confirmed**, **Driver on the Way**, **Ride in Progress**, **Ride Completed**.

7. **Rate & Review**
   - After the ride is completed, passengers can **rate the driver** and provide **feedback** on the experience.

.

CHAPTER 5: FLOW CHART

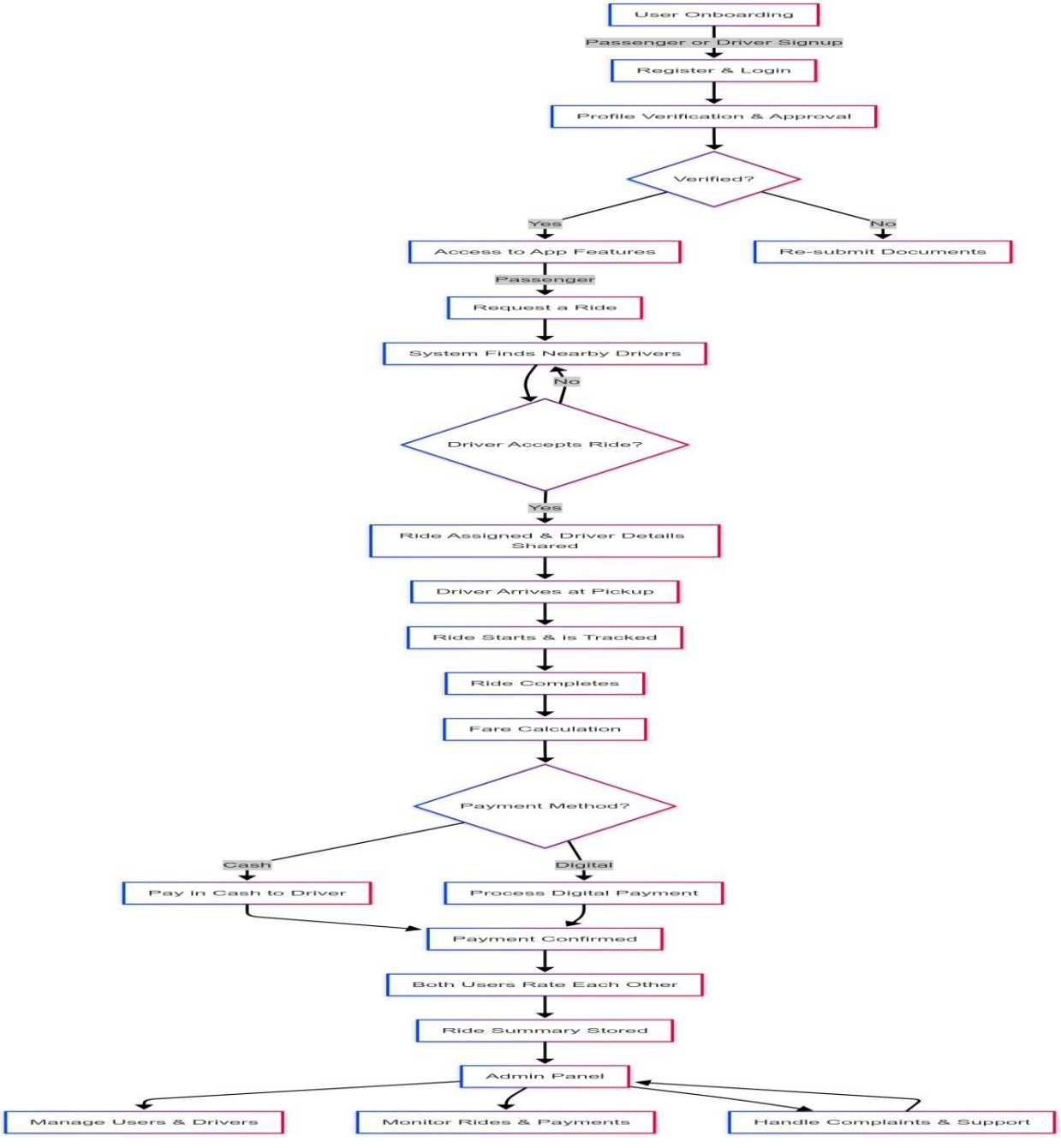

Fig 5.1 Flow Chart

CHAPTER 6: SCREENSHOTS

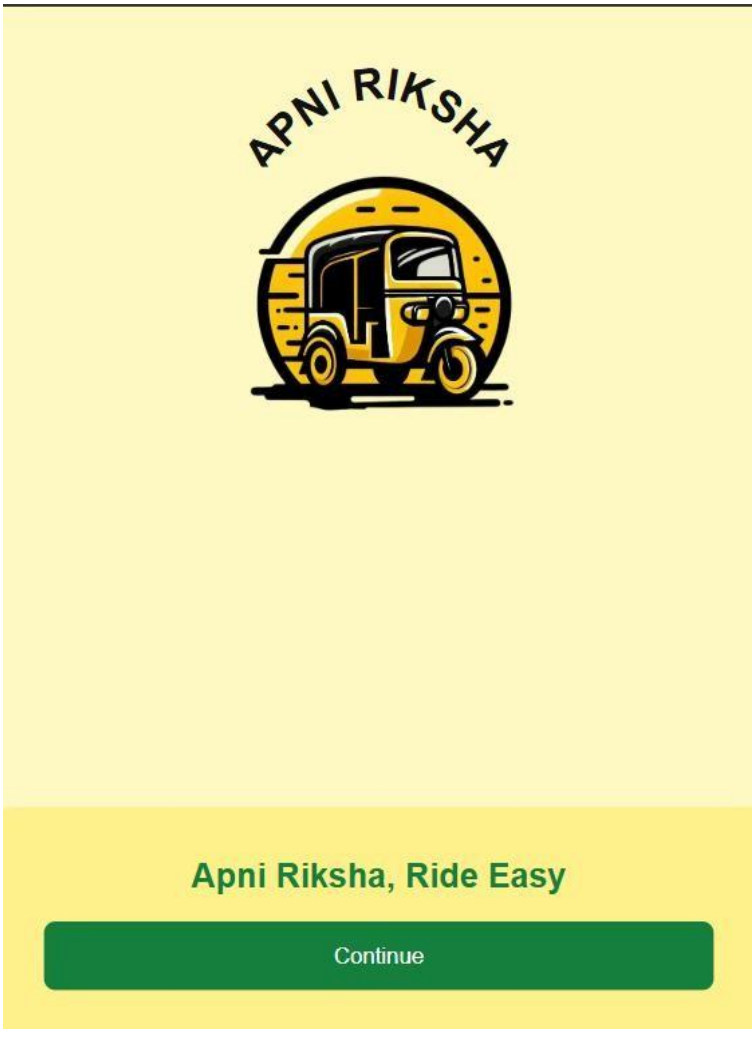

Fig 6.1 Front Page

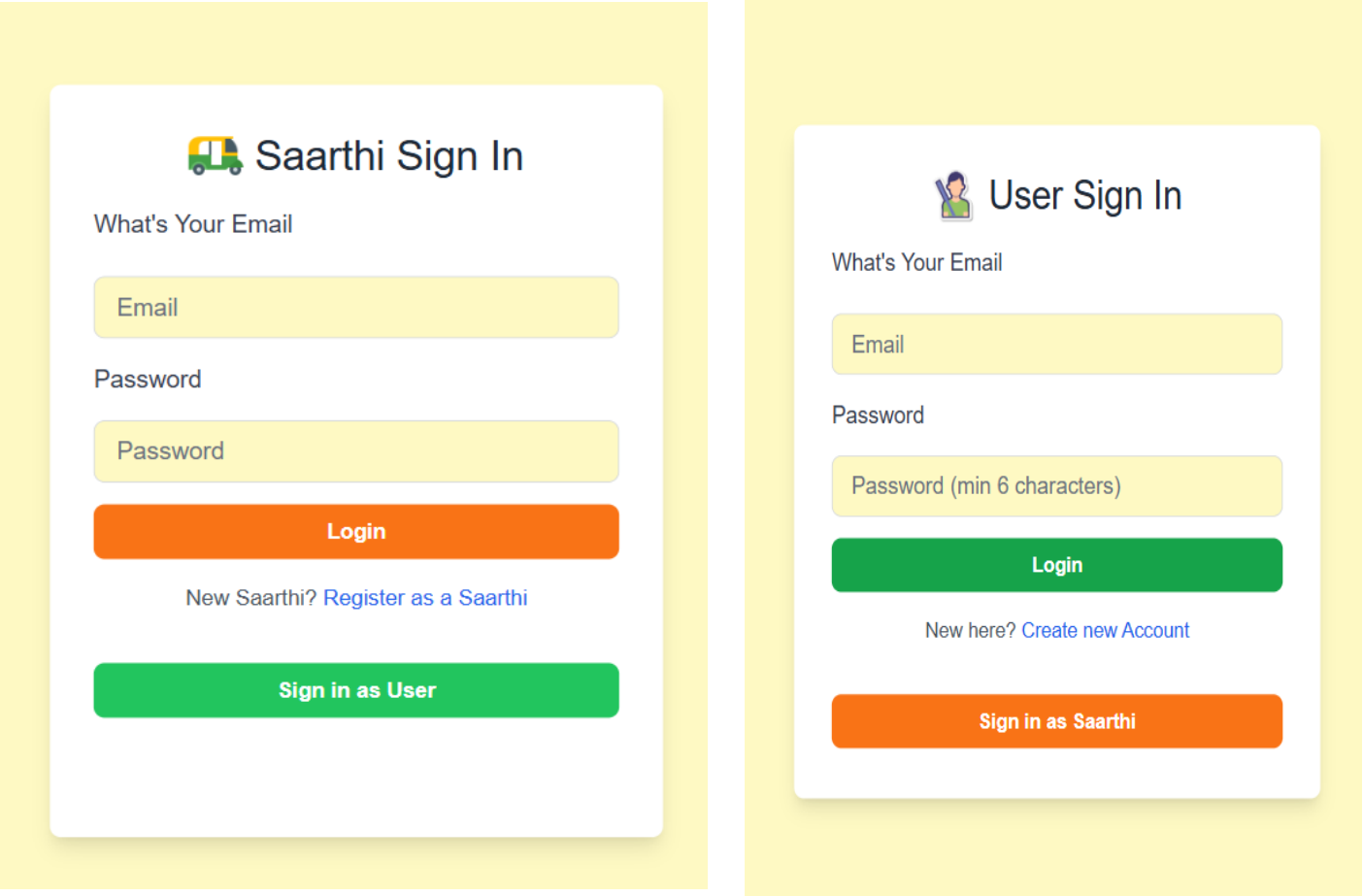

Fig 6.2 Login Page

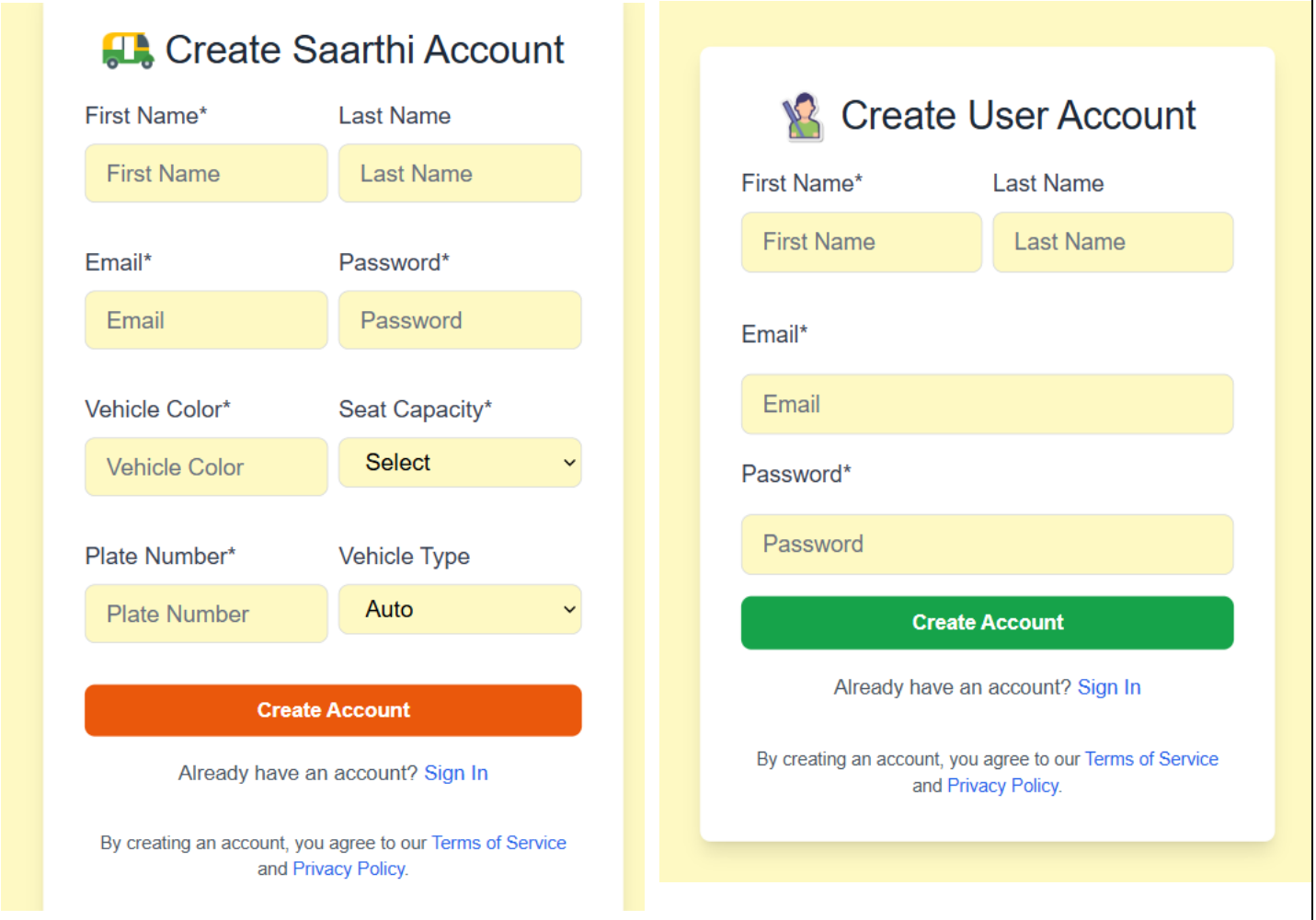

Fig 6.3 Register Page

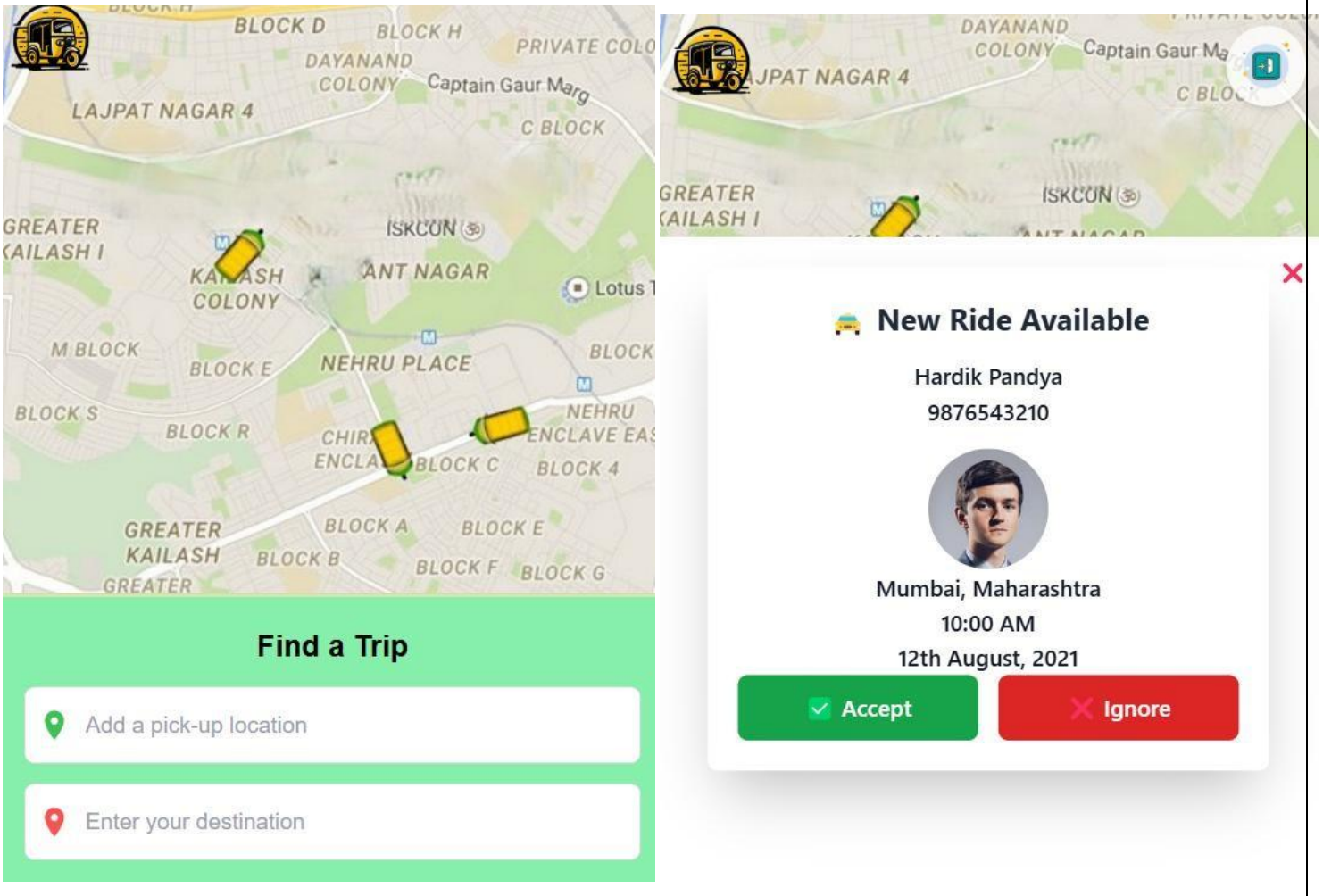

Fig 6.4 Main Page

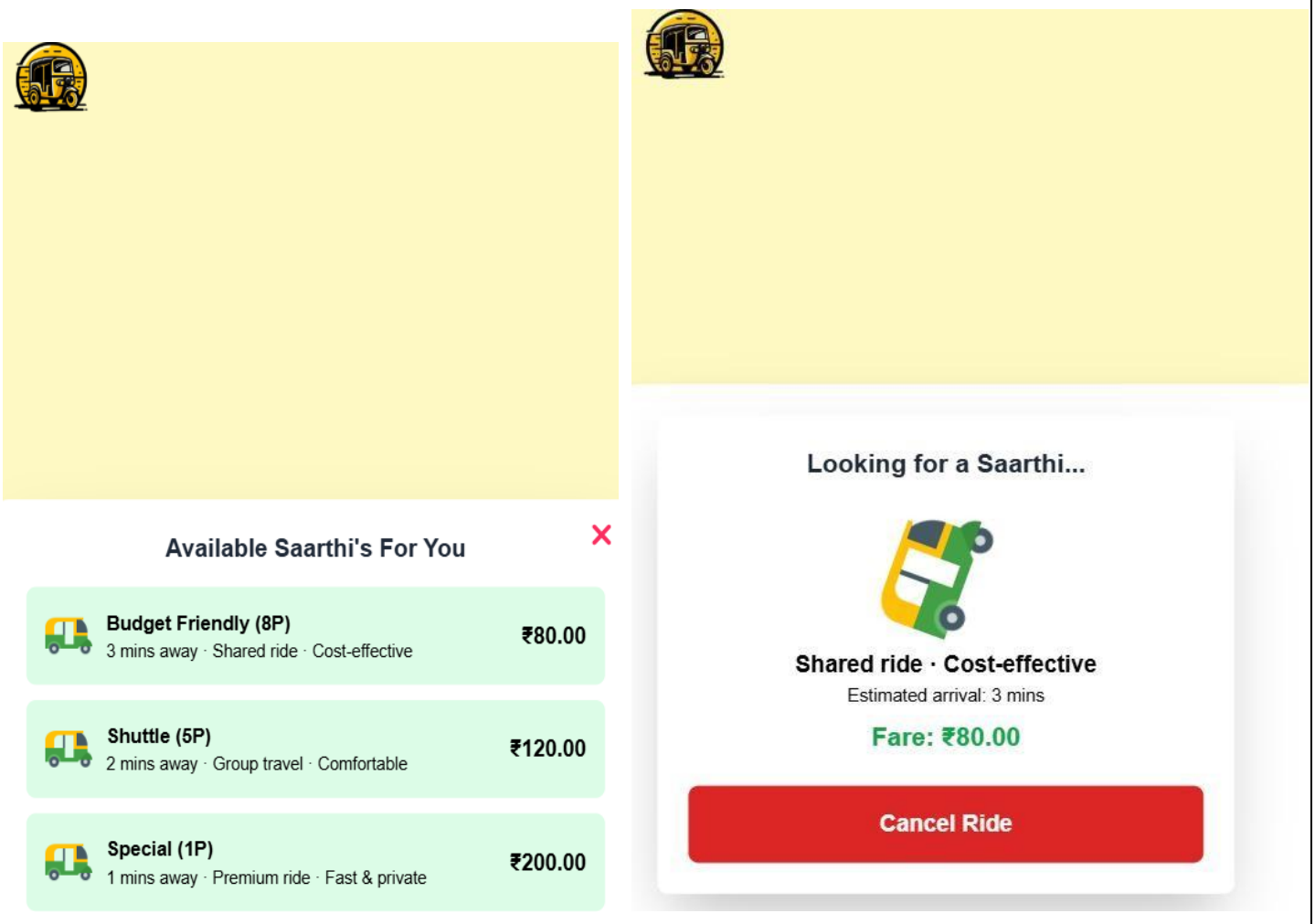

Fig 6.5 Booking Page

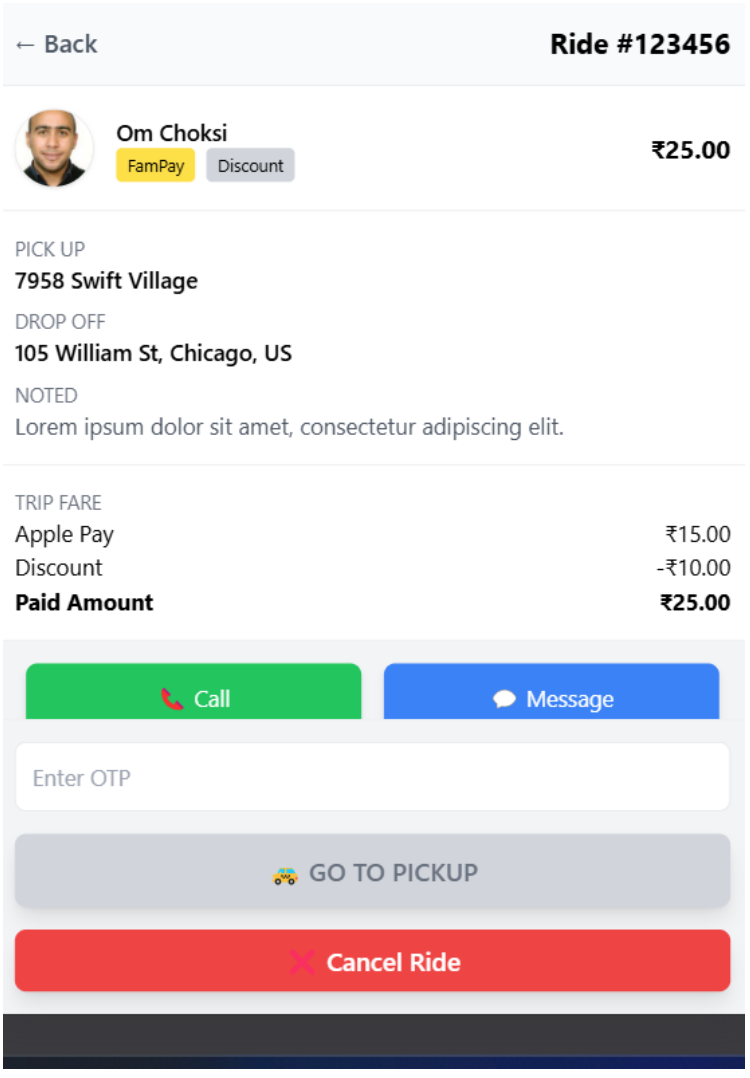

Fig 6.5 OTP Page

CHAPTER 7: LIMITATIONS

Limitations of Apni Ricksha

- **Limited Driver Availability –** Currently, the platform supports only a few local rickshaw drivers, which may restrict ride options.
- **No Real-Time GPS Tracking –** Users cannot track the rickshaw's exact location in real time, which may reduce transparency and convenience.
- **Basic UI/UX** – The interface is functional but lacks advanced design features that could enhance the user experience.
- **No Multi-Driver Management –** The system does not allow multiple drivers to operate from a shared account, limiting flexibility for fleet management.
- **No Customer Support Chat –** Users cannot chat with support directly from the platform, which may cause delays in issue resolution.

.

CHAPTER 8: OUTCOMES

- **Developing a Responsive Web Application** – Learned how to make a website compatible with different devices for a seamless user experience.
- **Implementing User Authentication** – Gained experience in creating a secure **Login and Logout** system for passengers and drivers.
- **Enhancing Data Security** – Understood how to **encrypt passwords** and protect user credentials for privacy.
- **Implementing Fare Estimation** – Learned how to calculate and display estimated fares before booking a ride.
- **Managing Ride Requests** – Gained insights into handling **ride bookings and driver availability** efficiently.

CHAPTER 9: FUTURE ENHANCEMENT

1. **Real-Time Tracking** – Implementing GPS-based tracking to allow users to monitor the exact location of their booked rickshaw in real time.

2. **AI-Based Recommendation** – Using artificial intelligence to suggest the nearest available rickshaw based on user preferences and demand patterns.

3. **Multi-Driver Support** – Enabling multiple drivers to manage bookings from a shared account for better fleet management.

4. **Loyalty and Rewards** – Introducing a rewards program where frequent users can earn discounts or free rides.

5. **Customer Support Chatbot** – Integrating an AI-based chatbot to assist users with queries and complaints instantly.

6. **Scheduled Rides** – Allowing users to pre-book rickshaws for a specific date and time to ensure availability.

7. **Mobile App Development** – Expanding Apni Ricksha into a dedicated mobile application for Android and iOS to improve accessibility and user experience.

.

CHAPTER 10: REFERENCES

[1] **HTML & CSS Documentation** – https://developer.mozilla.org/en-US/docs/Learn

[2] **JavaScript Documentation** – https://developer.mozilla.org/en-US/docs/Web/JavaScript

[3] **Bootstrap Framework** – Bootstrap Documentation

[4] **Session & Authentication in PHP** –https://www.php.net/manual/en/features.sessions.php

[5] **Google Maps API (for location-based features)** – Google Maps API Docs