PROJECT REPORT

---

aSk Ren (Car Rental Platform)

Submitted by:

Kushagra Chaudhary (12317517)

Shubh Sahu (12320996)

Alok Kumar (12319854)

Degree: B.Tech. Computer Science and Engineering

Trainer: Dr. Siddardha Dasari

University: Lovely Professional University

ACKNOWLEDGEMENT

We express our sincere gratitude to our guide Dr. Siddardha Dasari for continuous support, guidance, and encouragement throughout the development of this project. We also thank Lovely Professional University for providing the infrastructure and academic environment necessary for the successful completion of this work.

TABLE OF CONTENTS

1. Executive Summary
2. Introduction
3. Project Overview
4. System Analysis
5. System Architecture & Design
6. Database Design
7. Data Flow Diagrams (DFD)
8. Technology Stack
9. Implementation Plan
10. System Demonstration / Key Screenshots
11. Development Team Introduction
12. Advantages
13. Future Enhancements
14. Conclusion

1. Executive Summary

aSk Ren is a comprehensive full-stack digital platform designed to modernize vehicle rental processes. By leveraging the Django framework and MySQL, the project aims to replace manual record-keeping with an automated, secure, and scalable system. The primary objective is to facilitate seamless interactions between car owners and customers through a centralized administrative approval workflow. Expected outcomes include improved transparency, reduced manual errors, and a robust foundation for future commercial deployment.

---

2. Introduction

With the rapid growth of digital platforms, transportation services have evolved toward online ecosystems. The **aSk Ren** platform aims to digitize conventional rental processes by providing a secure, scalable, and user-friendly system. The application supports three primary roles: Customer, Owner, and Administrator. Customers can browse and book vehicles, owners can manage listings, and administrators monitor and approve activities.

---

3. Project Overview

3.1 Business Problem Statement

Traditional car rental services frequently depend on manual logs and physical verifications, leading to operational inefficiencies and data inconsistencies. The lack of a structured onboarding mechanism for car owners and transparent approval processes limits growth and trust.

3.2 Project Objectives

- Digitize the conventional vehicle rental ecosystem through a unified platform.
- Implement a secure, role-based access control system (Customer, Owner, Administrator).
- Enable real-time tracking of vehicle availability and booking status.

- Provide an administrative dashboard for monitoring and approving listings.

3.3 Scope (In-Scope)

In-Scope:

- Full-stack web application development using Django MVT.
- Relational database management with MySQL.
- User authentication and role-based permissions.
- Payment integration and review mechanisms.

3.4 Success Criteria

Successful execution of the user registration to payment recording workflow and stable performance under moderate concurrent loads.

---

4. System Analysis

4.1 Existing System

Traditional rental businesses rely on manual records and physical verification. This leads to inefficiencies, errors, and lack of transparency. Customer data is often isolated in localized spreadsheets, causing administrative overhead during audits.

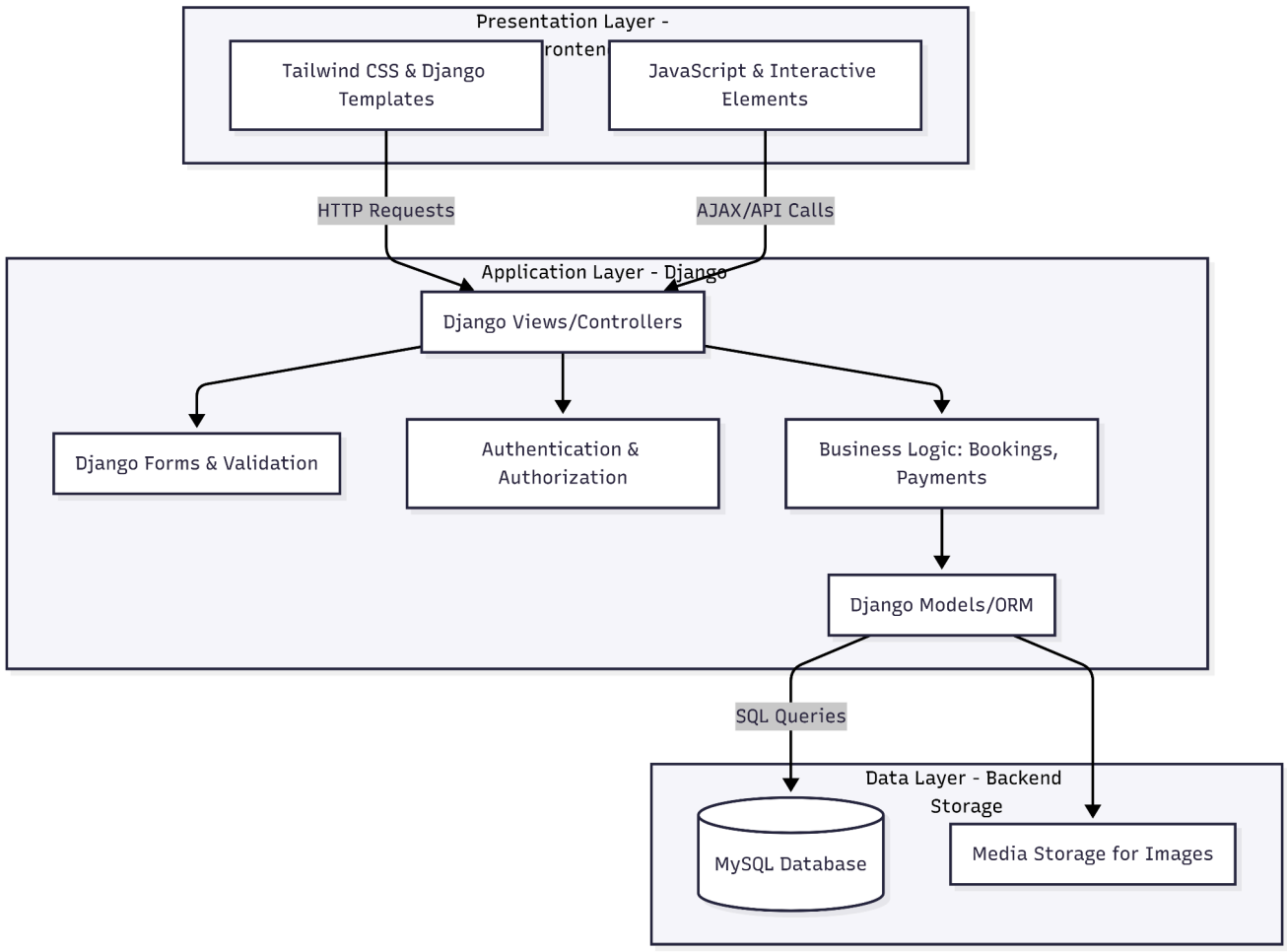
4.2 Proposed System

The proposed web-based system, **aSk Ren**, automates booking validation, maintains real-time availability, enforces role-based permissions, and stores data securely using relational database management. Features such as dashboard analytics and automated notifications significantly reduce the manual workload for managers.

---

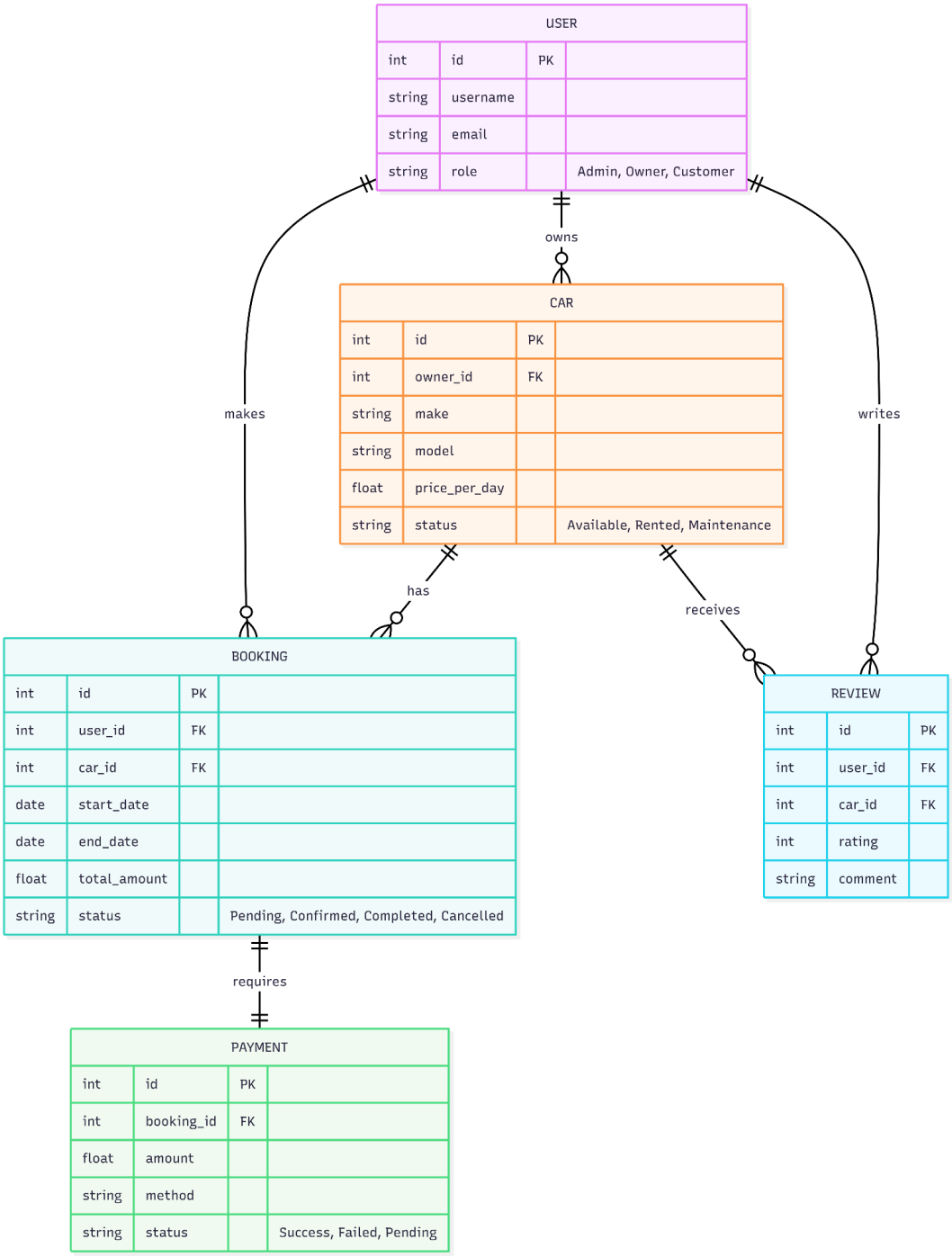
5. System Architecture & Design

The platform utilizes a modular three-tier architecture: the Presentation Layer (Frontend), the Application Layer (Backend), and the Database Layer (Storage). This ensures separation of concerns and facilitates independent scaling.



6. Database Design

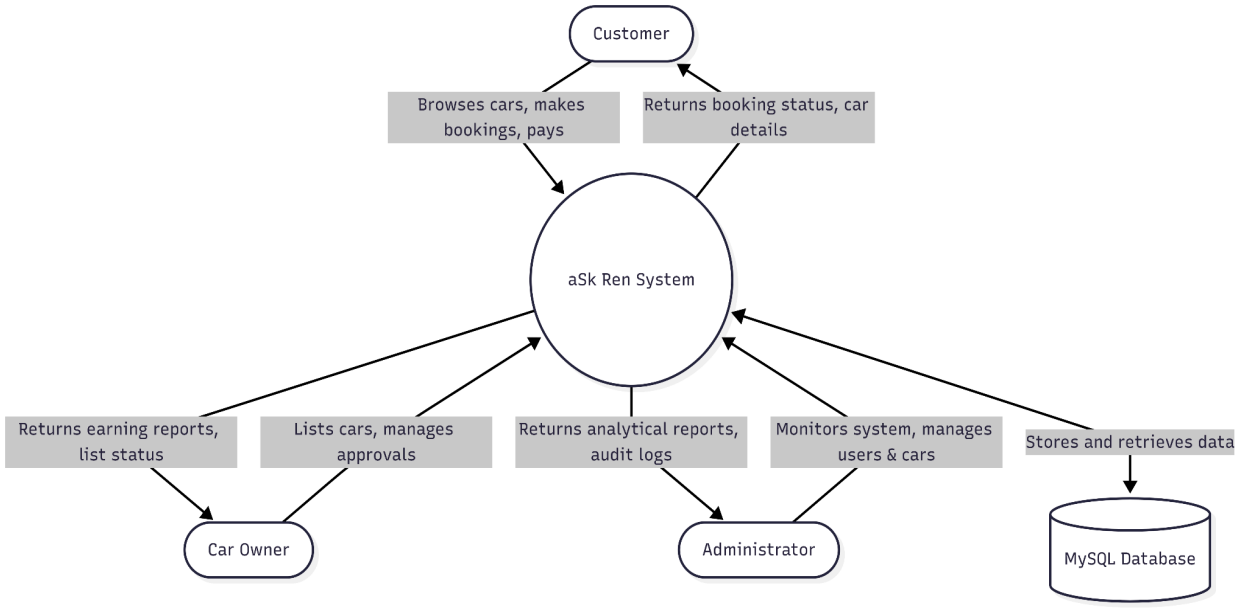
The database schema ensures referential integrity using primary and foreign keys. Relationships define interactions between users, cars, bookings, reviews, and payments.



7. Data Flow Diagrams (DFD)

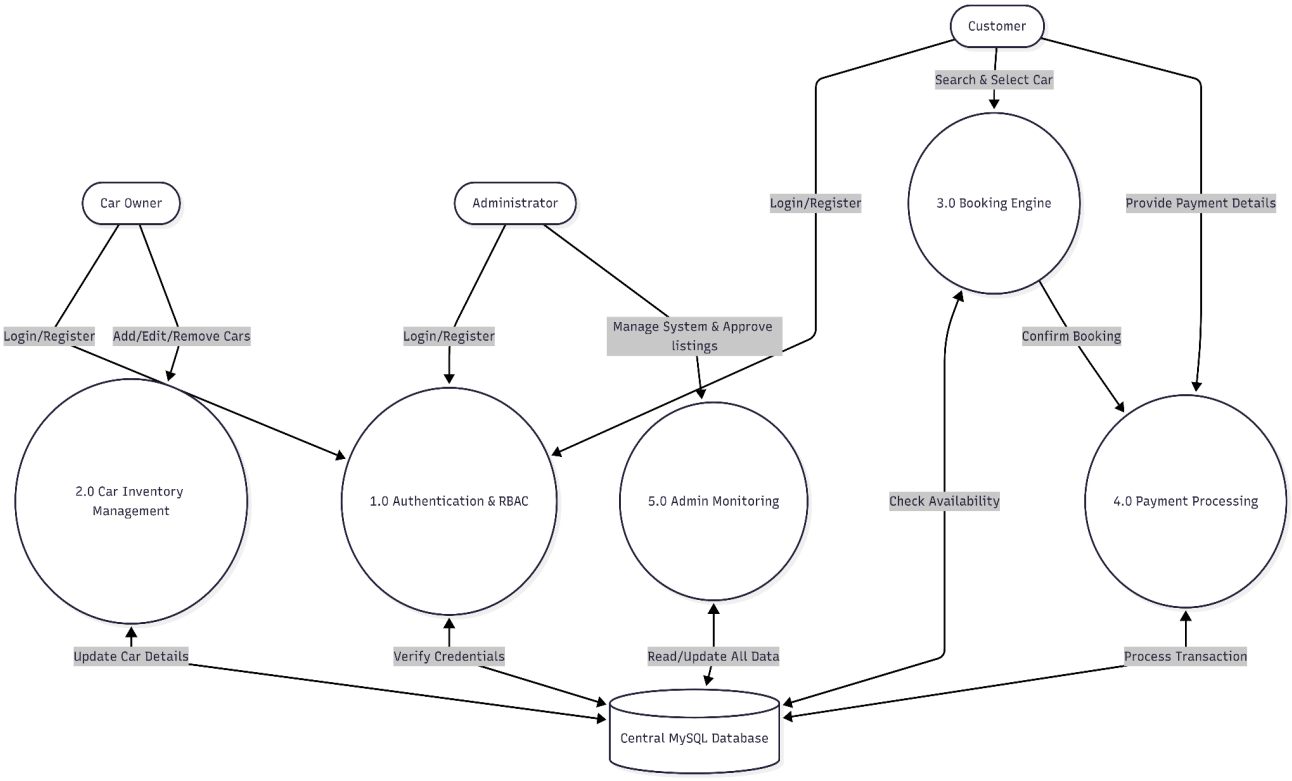
7.1 DFD Level 0 (Context Diagram)

The Context Diagram shows the interaction between the **aSk Ren** system and external entities like Users and Admins.



7.2 DFD Level 1

Level 1 DFD breaks down the system into major processes such as Authentication, Car Management, Booking, Payment Processing, and Admin Control.



---

8. Technology Stack

- **Frontend:** Tailwind CSS for responsive UI design, accompanied by HTML5 and vanilla JavaScript.
- **Backend:** Django (Python) for business logic, view orchestration, and the booking engine.
- **Database:** MySQL for robust relational storage and referential integrity.
- **Security & Compliance:** The system enforces role-based access control (RBAC) to ensure that sensitive operations like listing approval are restricted to administrators. Django's built-in authentication system provides robust protection against common web vulnerabilities (XSS, CSRF, SQL Injection).

---

9. Implementation Plan

9.1 Project Phases & Timeline

- **Phase 1: Requirements Gathering & System Analysis** Documenting project scope, analyzing the current state, and defining system architecture.
- **Phase 2: Database Modeling & Backend Development** Establishing Django apps (accounts, bookings, cars, payments), configuring MySQL schema, and implementing business logic.
- **Phase 3: Frontend Integration & Testing** Translating wireframes to Tailwind templates, integrating Django views, and conducting comprehensive integration testing.

9.2 Risk Assessment & Mitigation

- **Risk:** High concurrent traffic overwhelming the database.
- **Mitigation:** Optimized database queries, usage of Django's select_related/prefetch_related, and incorporating caching mechanisms for static/semi-static data.

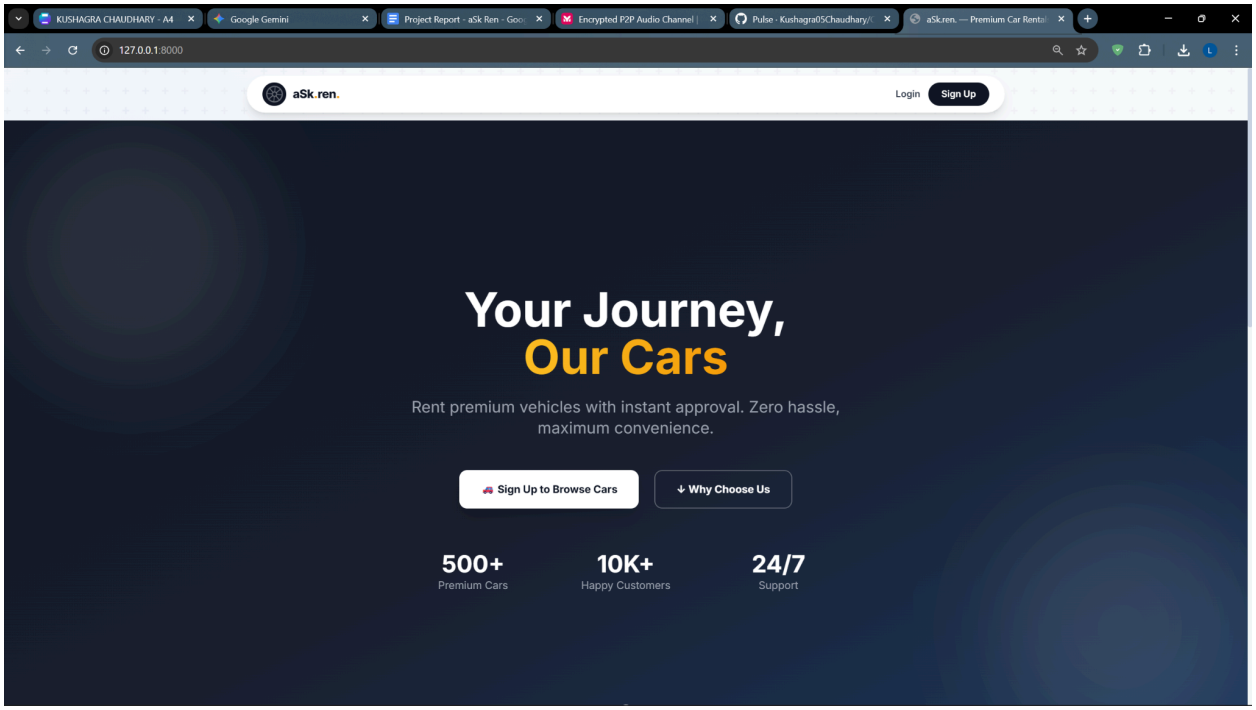
---

10. System Demonstration / Key Screenshots

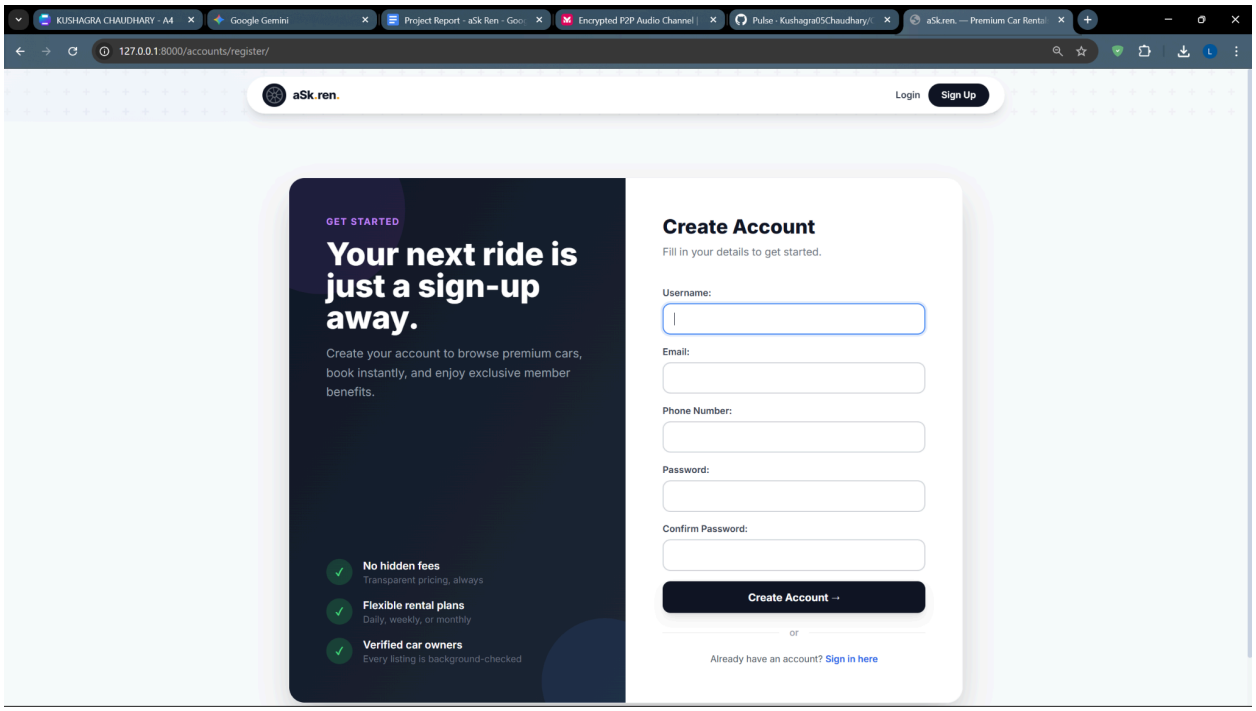
10.1 Login and Registration Page

The application includes the following key user interface screens:

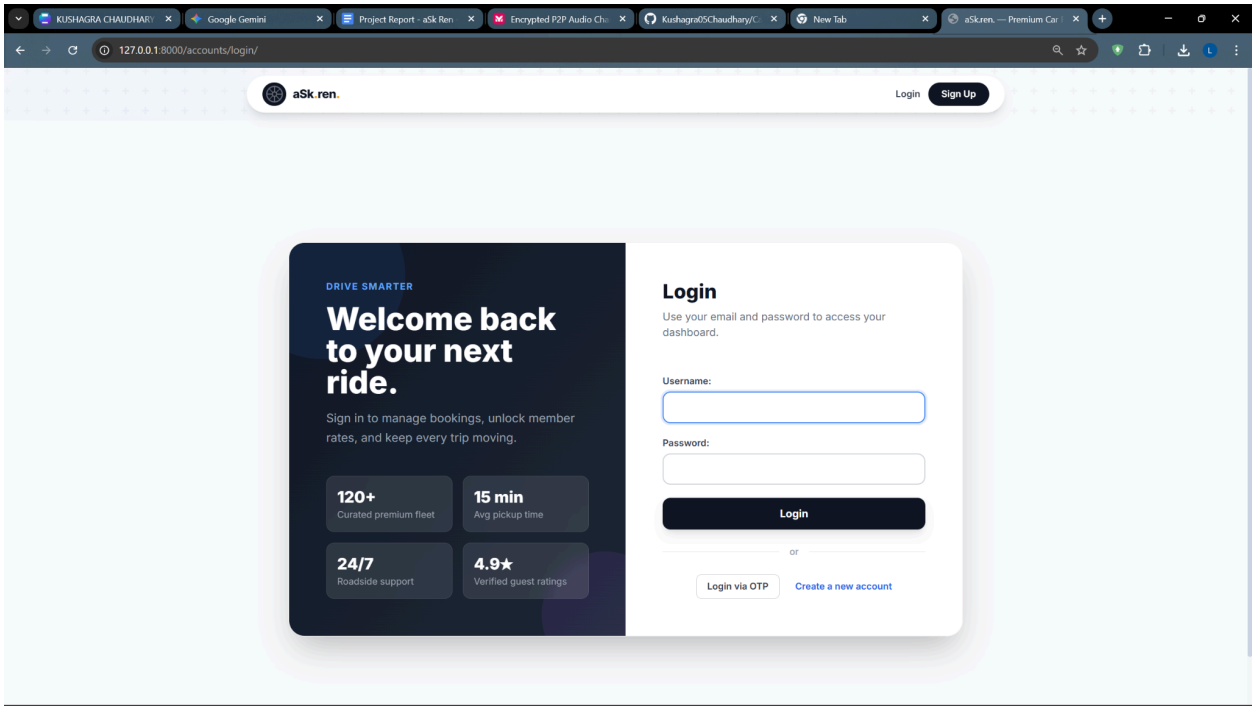
- **Landing Page:** The initial screen users encounter, typically providing an overview of the application and its core features, often with calls to action like 'Register' or 'Login'.



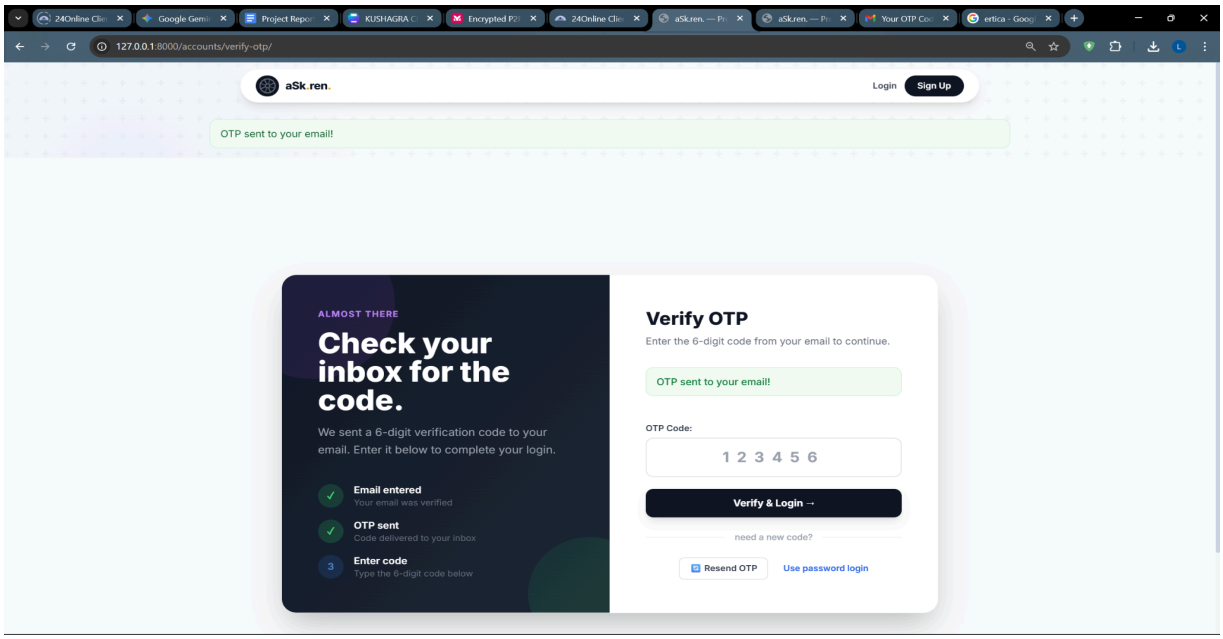
- **Registration Page:** Allows new users to create an account by providing necessary details (e.g., name, email, password, or mobile number).



- **Login Page:** Provides existing users with a mechanism to access their accounts. This page offers users the option to log in using either traditional credentials (username/password) or a One-Time Password (OTP).

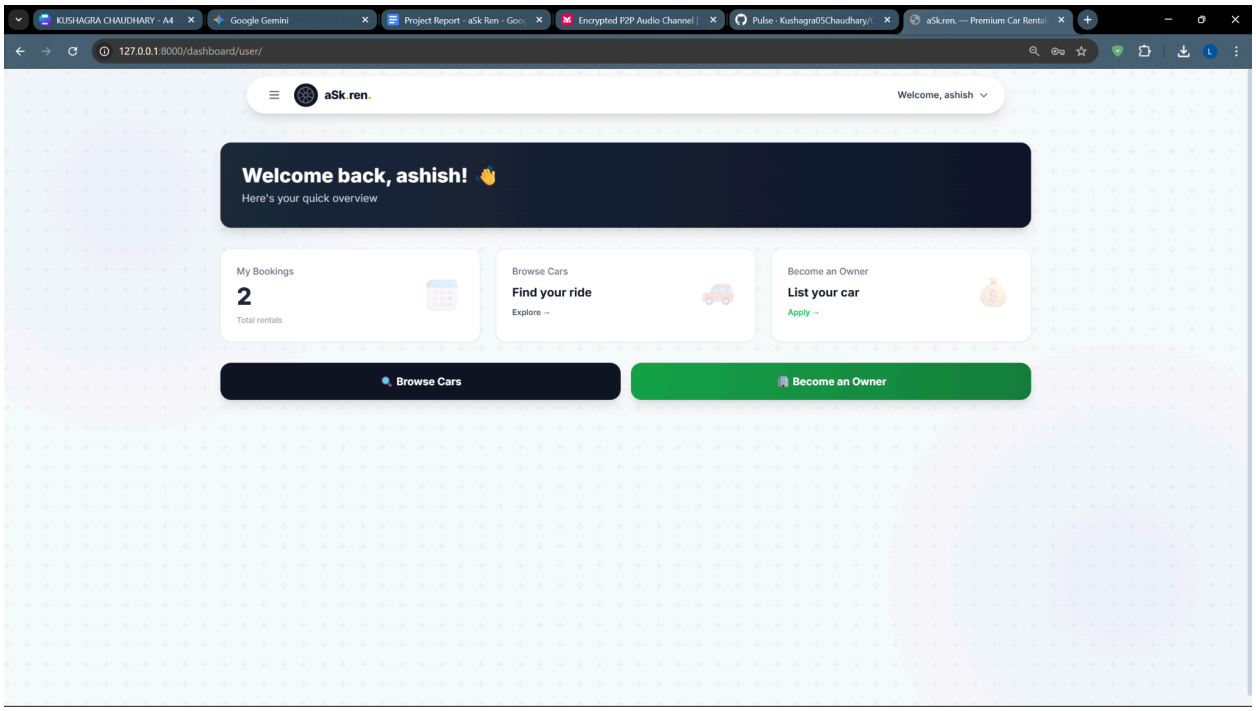


- **OTP Verification Page:** This page is a secure verification step after 'Login with OTP', prompting the user to enter the unique One-Time Password sent to their device to complete the login process.

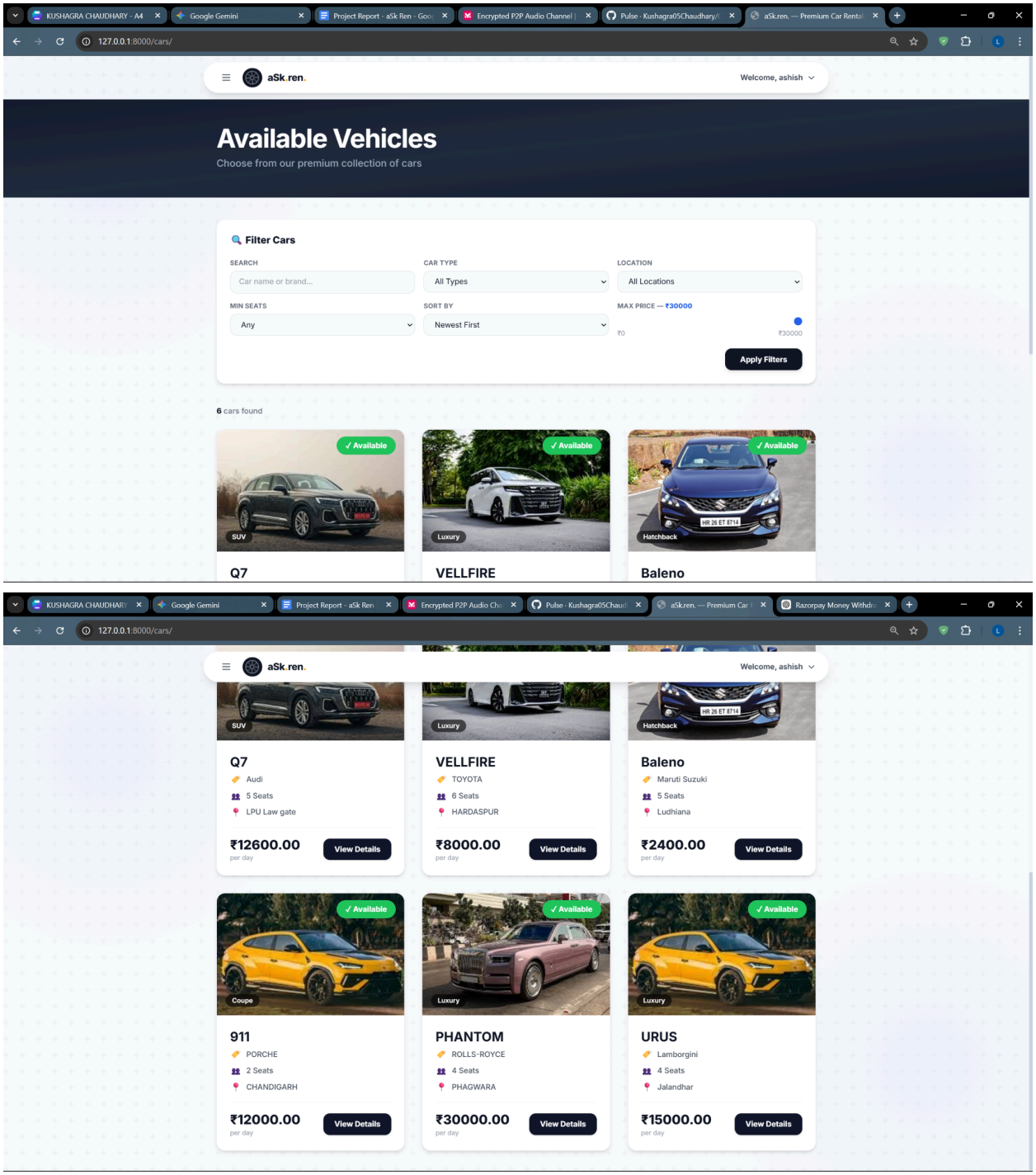


10.2 User

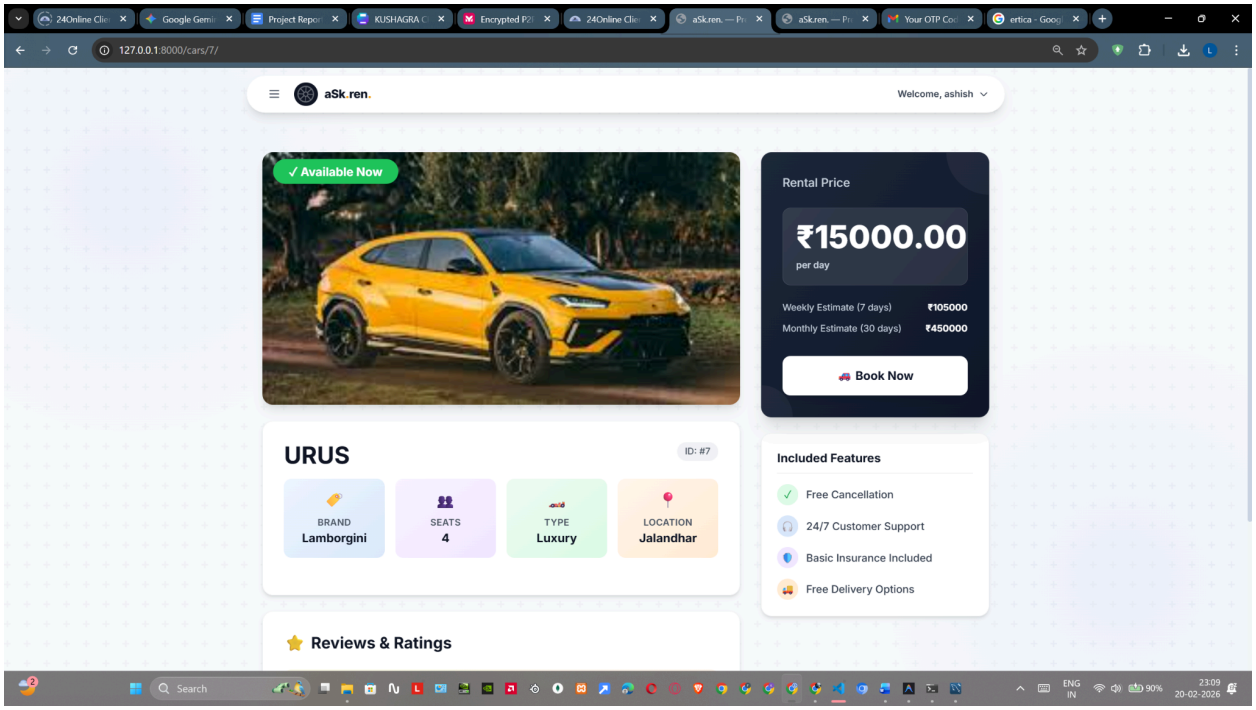
1. **User Dashboard**
   The logged-in user's central dashboard acts as a landing page, offering an immediate, personalized activity overview. Key features include: a summary of current, upcoming, or past bookings; quick links like "Browse Cars"; personalized recommendations; profile, payment, and notification management; and a display of active promotions.



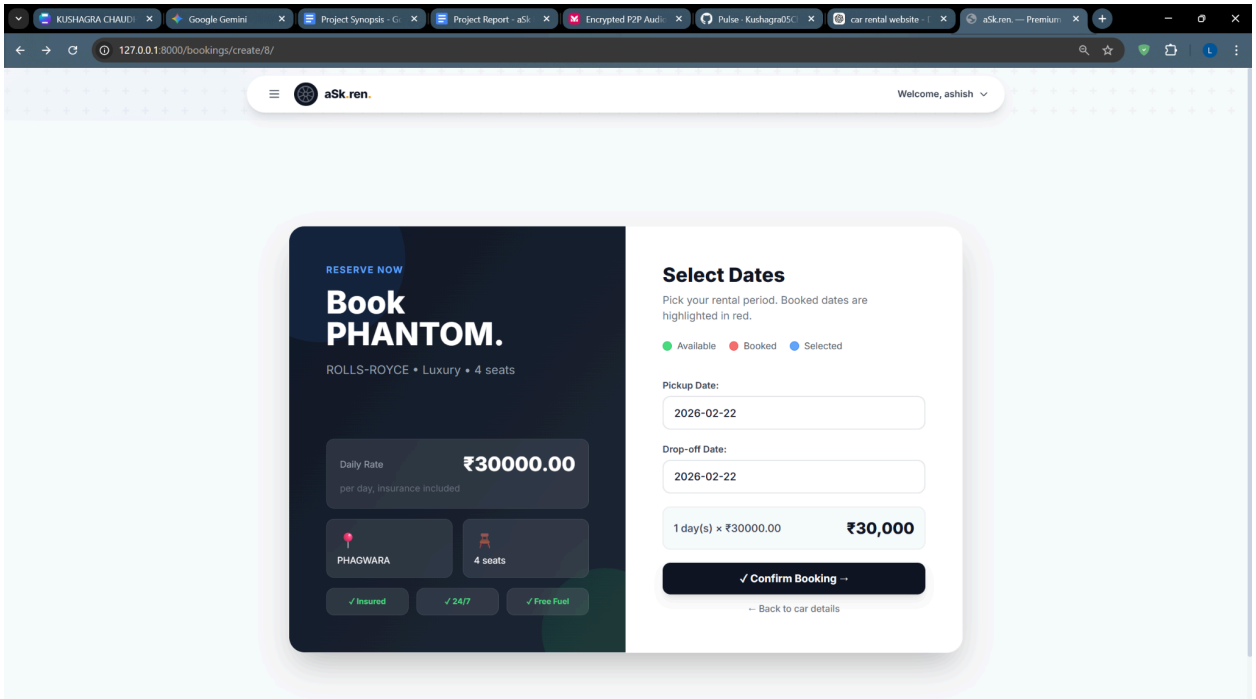
2. **Browse Cars**
   This screen displays the comprehensive catalog of rental vehicles. It features efficient search and filtering, allowing users to: view car listings (image, make, model, price); use robust filters (type, transmission, efficiency, capacity, price range); sort results (relevance, price, rating); and input dates/locations for real-time availability checks.

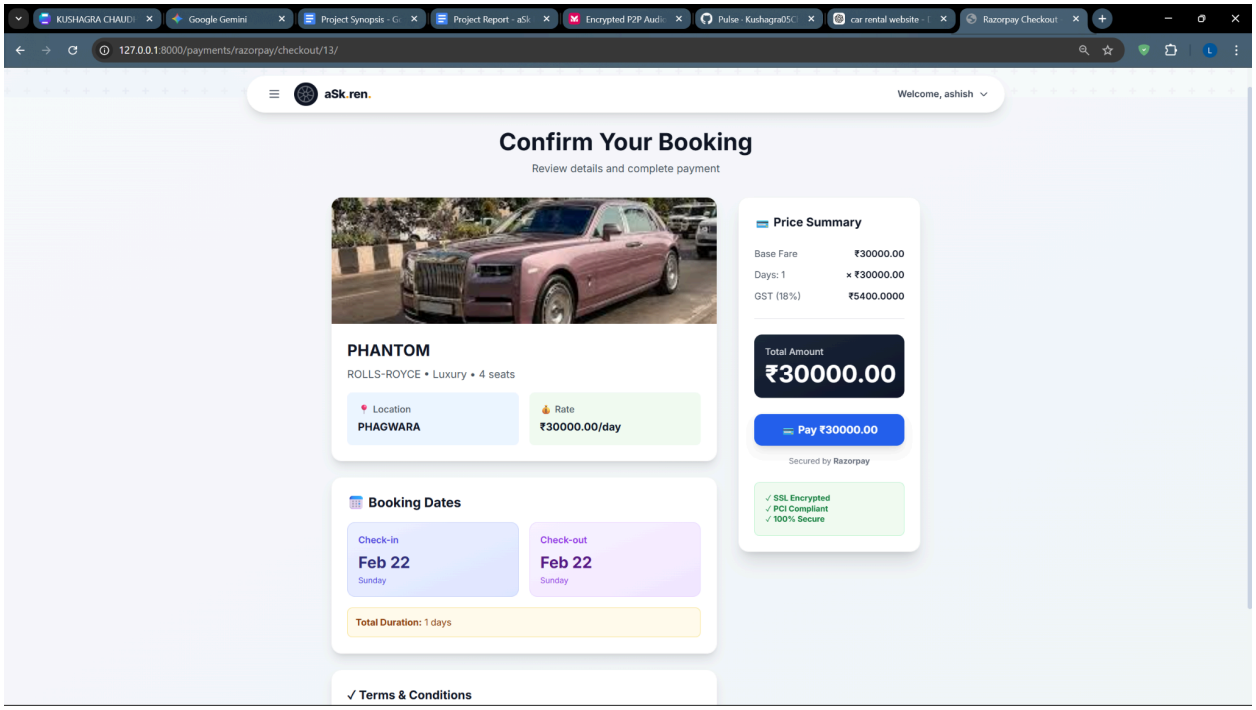


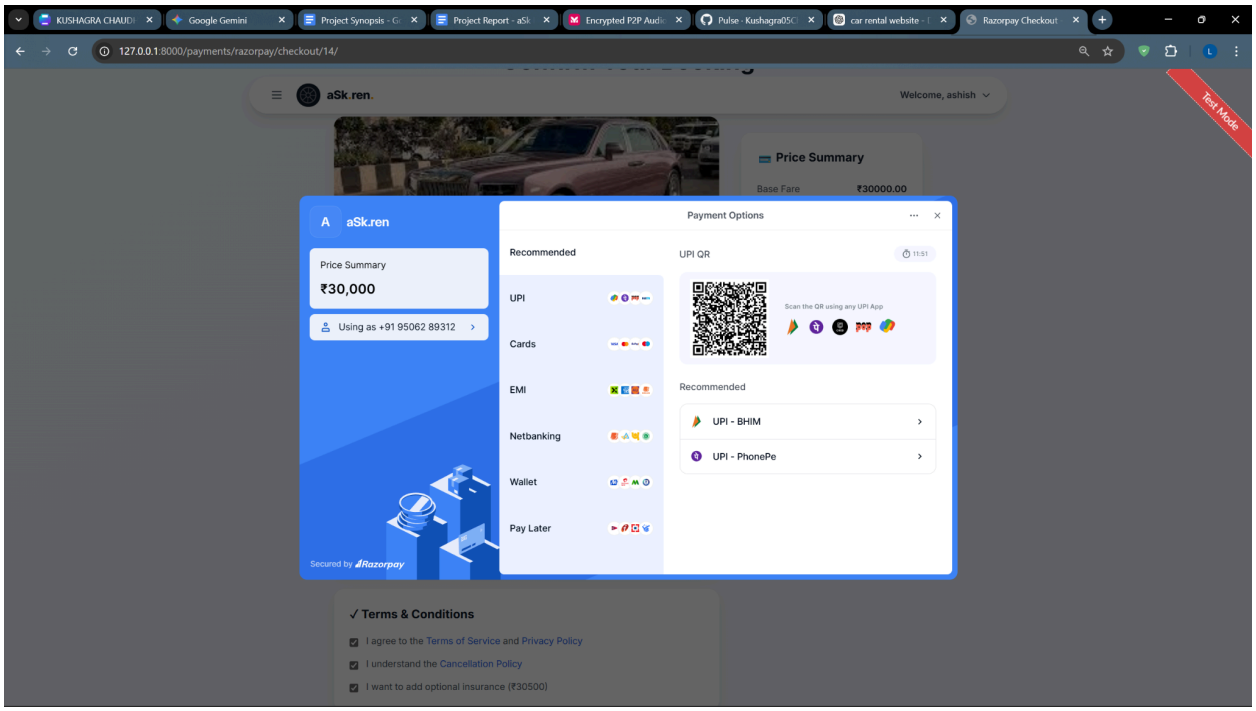
3. **View Detail**
   This final commitment page, reached after selecting a car from "Browse Cars," shows all necessary vehicle information: high-resolution photos, a features/specifications list, a detailed pricing breakdown (daily rate, taxes, insurance, fees), and rental terms. The "Book Now" button starts the booking process.



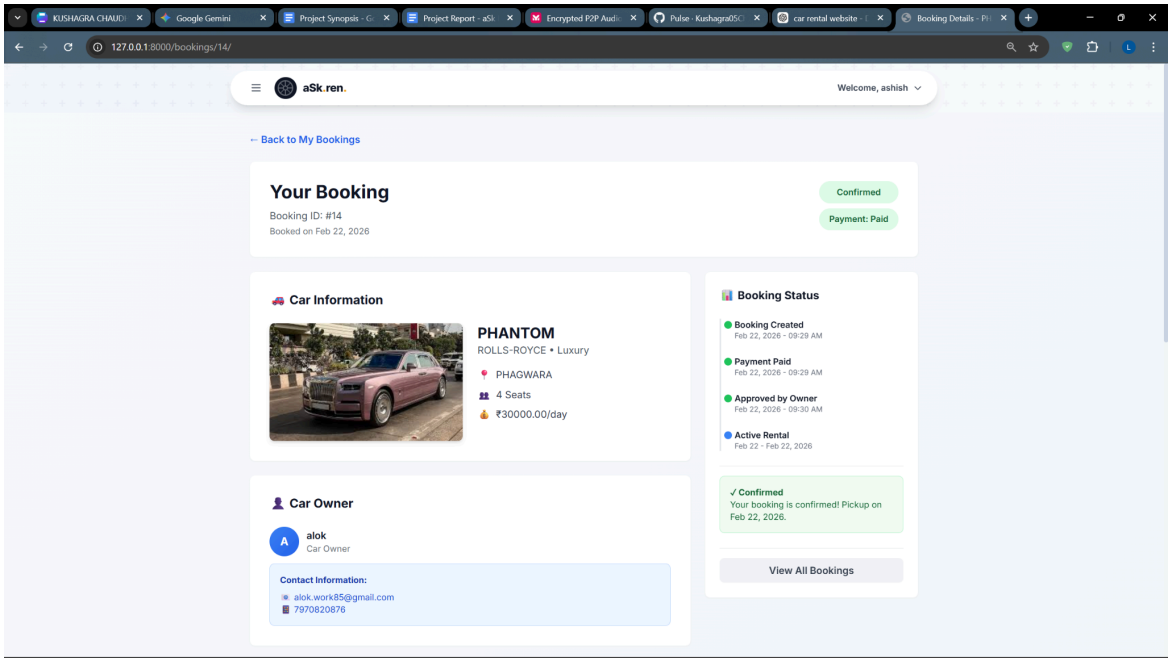
4. **Book Now**
   This screen captures rental logistics for the transactional workflow, prompting the user to: select precise pickup and drop-off dates/times and locations (including one-way options) and optionally choose add-ons (child seats, GPS, insurance). The user then proceeds to payment confirmation.

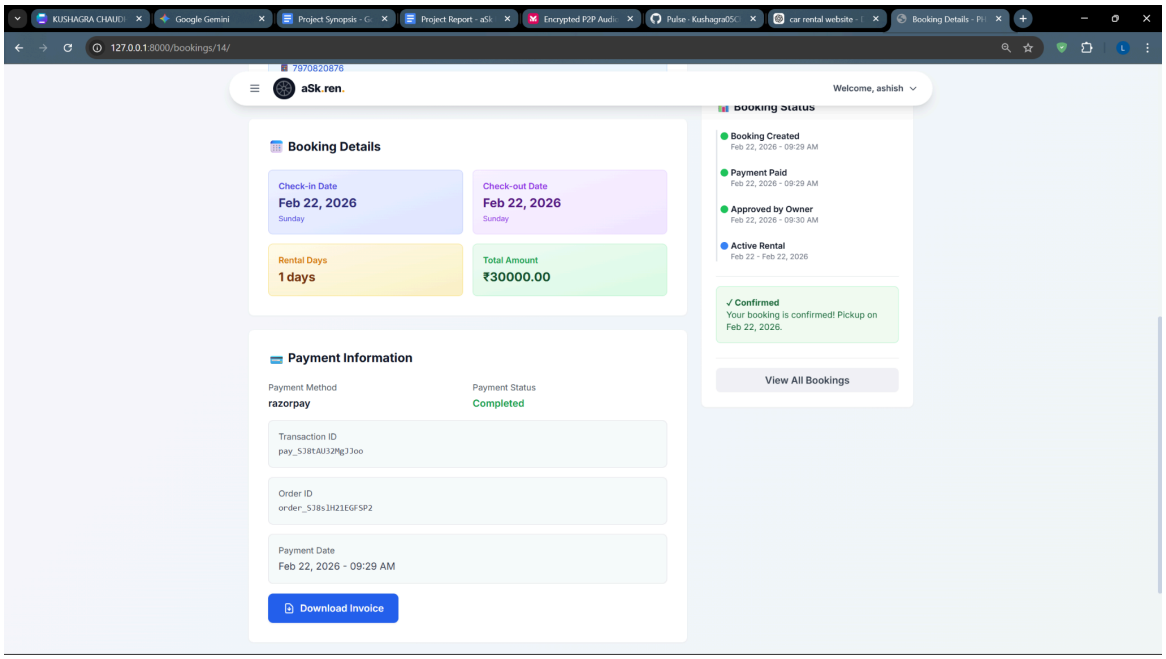


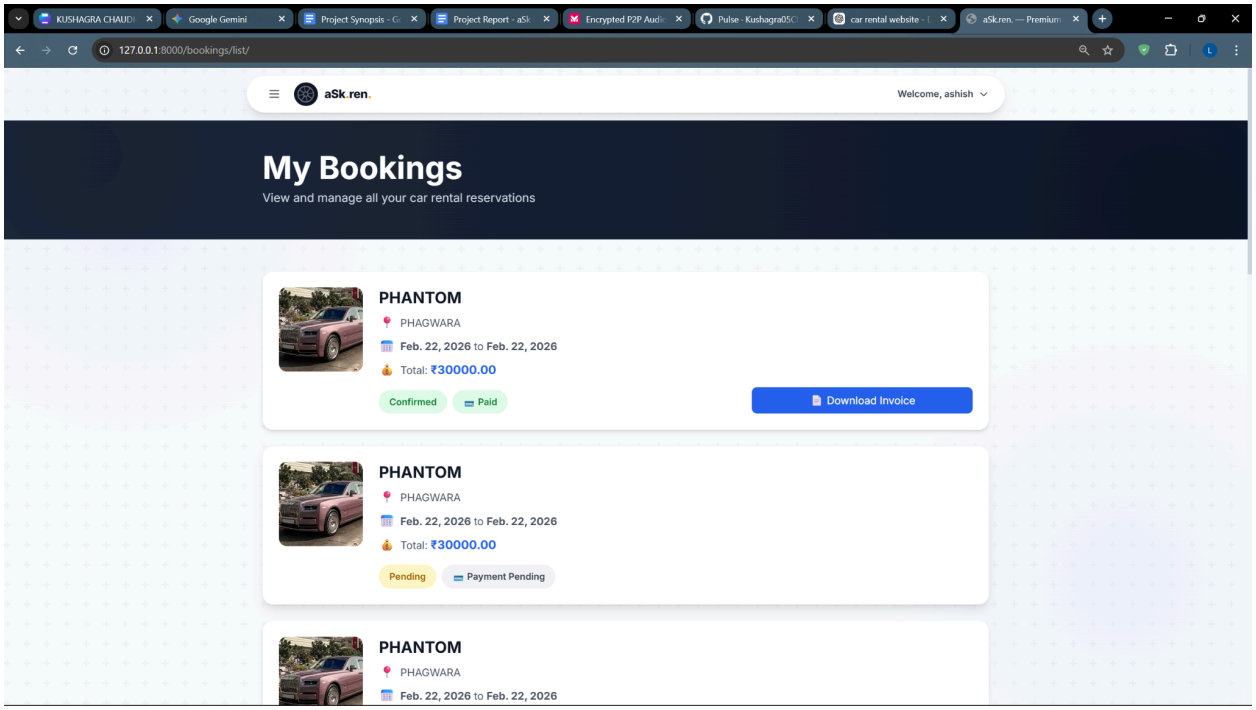

127.0.0.1:8000/payments/razorpay/checkout/13/
aSk.ren.
Welcome, ashish
Confirm Your Booking
Review details and complete payment
PHANTOM
ROLLS-ROYCE • Luxury • 4 seats
Location
PHAGWARA
Rate
₹30000.00/day
Booking Dates
Check-in
Feb 22
Sunday
Check-out
Feb 22
Sunday
Total Duration: 1 days
✓ Terms & Conditions
Price Summary
Base Fare ₹30000.00
Days: 1 × ₹30000.00
GST (18%) ₹5400.0000
Total Amount
₹30000.00
Pay ₹30000.00
Secured by Razorpay
✓ SSL Encrypted
✓ PCI Compliant
✓ 100% Secure

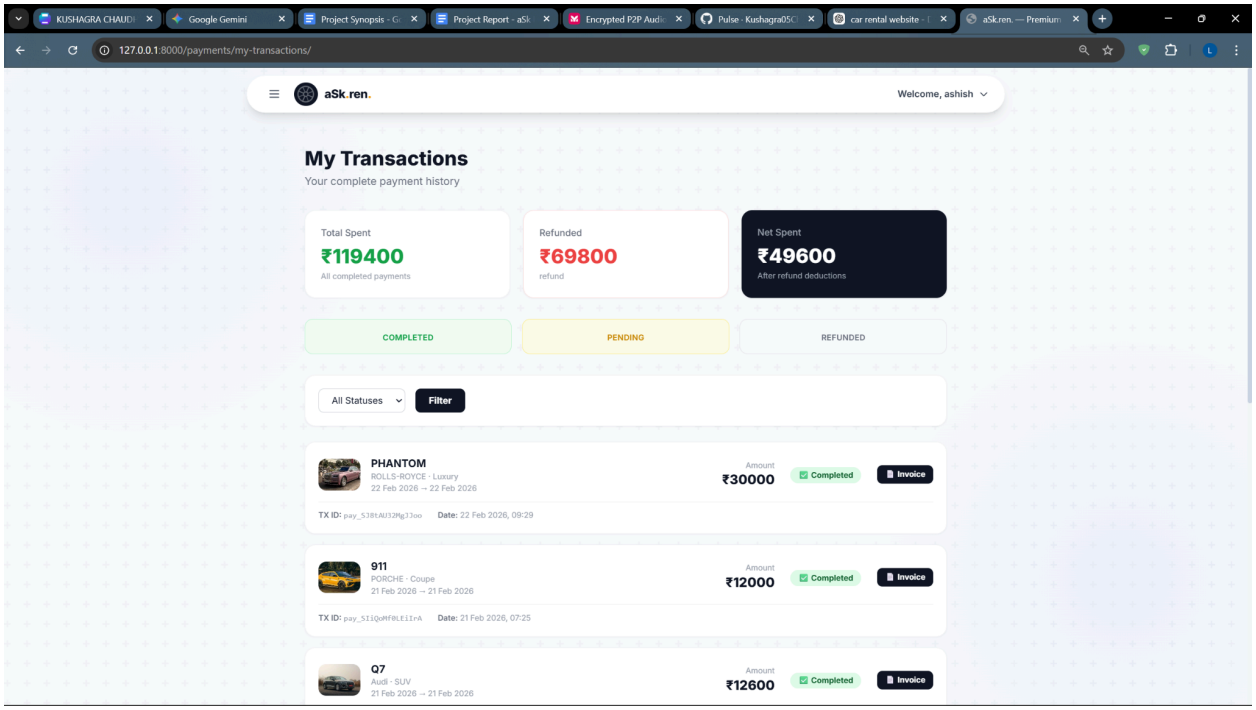

127.0.0.1:8000/payments/razorpay/checkout/14/
aSk.ren.
Welcome, ashish
Test Mode
Price Summary
Base Fare ₹30000.00
Payment Options
aSk.ren
Price Summary
₹30,000
Using as +91 95062 89312
Recommended
UPI
Cards
EMI
Netbanking
Wallet
Pay Later
UPI QR
Scan the QR using any UPI App
Recommended
UPI - BHIM
UPI - PhonePe
Secured by Razorpay
✓ Terms & Conditions
I agree to the Terms of Service and Privacy Policy
I understand the Cancellation Policy
I want to add optional insurance (₹30500)

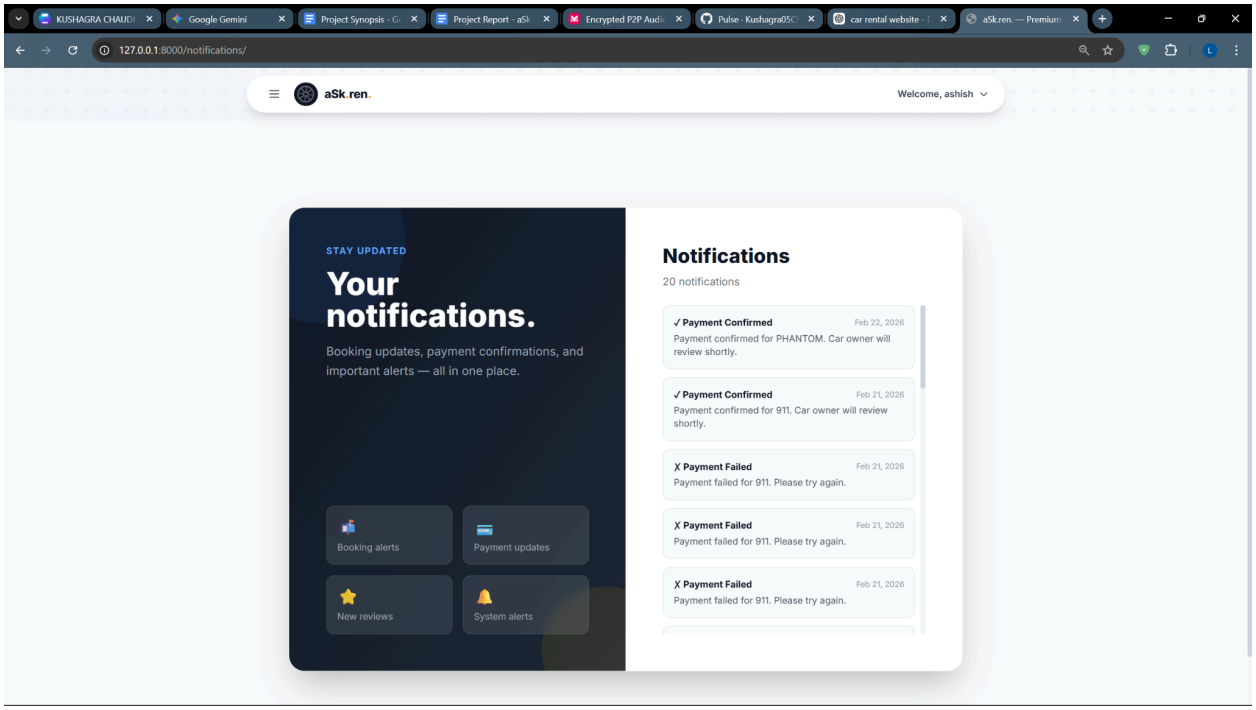
5. **Confirm Booking and Pay**
   This final stage requires the user to review the complete transaction summary (car, dates, locations, total cost including taxes/fees) and accept the rental agreement. Payment is processed securely via a third-party gateway (Razor Pay) for credit/debit cards, net banking, etc., ensuring PCI compliance. Successful payment finalizes the booking, sending a confirmation email and updating the User Dashboard.





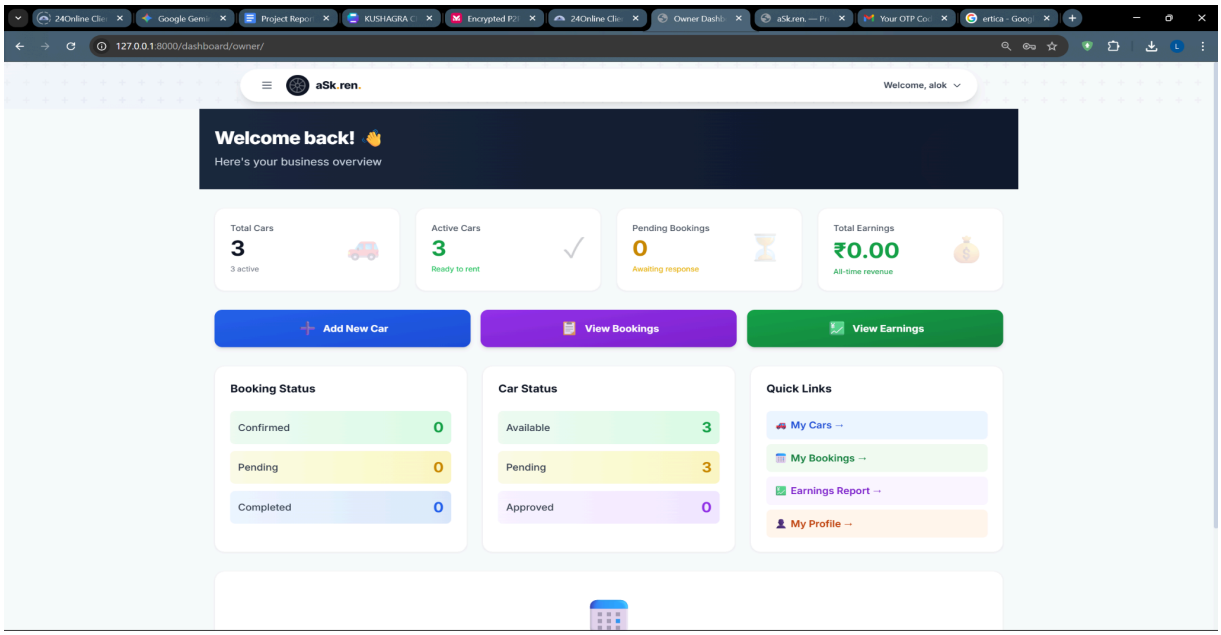

127.0.0.1:8000/bookings/list/
aSk.ren.
Welcome, ashish
My Bookings
View and manage all your car rental reservations
PHANTOM
PHAGWARA
Feb. 22, 2026 to Feb. 22, 2026
Total: ₹30000.00
Confirmed
Paid
Download Invoice
PHANTOM
PHAGWARA
Feb. 22, 2026 to Feb. 22, 2026
Total: ₹30000.00
Pending
Payment Pending
PHANTOM
PHAGWARA
Feb. 22, 2026 to Feb. 22, 2026


127.0.0.1:8000/payments/my-transactions/
aSk.ren.
Welcome, ashish
My Transactions
Your complete payment history
Total Spent
₹119400
All completed payments
Refunded
₹69800
refund
Net Spent
₹49600
After refund deductions
COMPLETED
PENDING
REFUNDED
All Statuses
Filter
PHANTOM
ROLLS-ROYCE · Luxury
22 Feb 2026 → 22 Feb 2026
Amount
₹30000
Completed
Invoice
TX ID: pay_S38tAU32Mg3oo
Date: 22 Feb 2026, 09:29
911
PORCHE · Coupe
21 Feb 2026 → 21 Feb 2026
Amount
₹12000
Completed
Invoice
TX ID: pay_SIiQoMfdLEiIrA
Date: 21 Feb 2026, 07:25
Q7
Audi · SUV
21 Feb 2026 → 21 Feb 2026
Amount
₹12600
Completed
Invoice

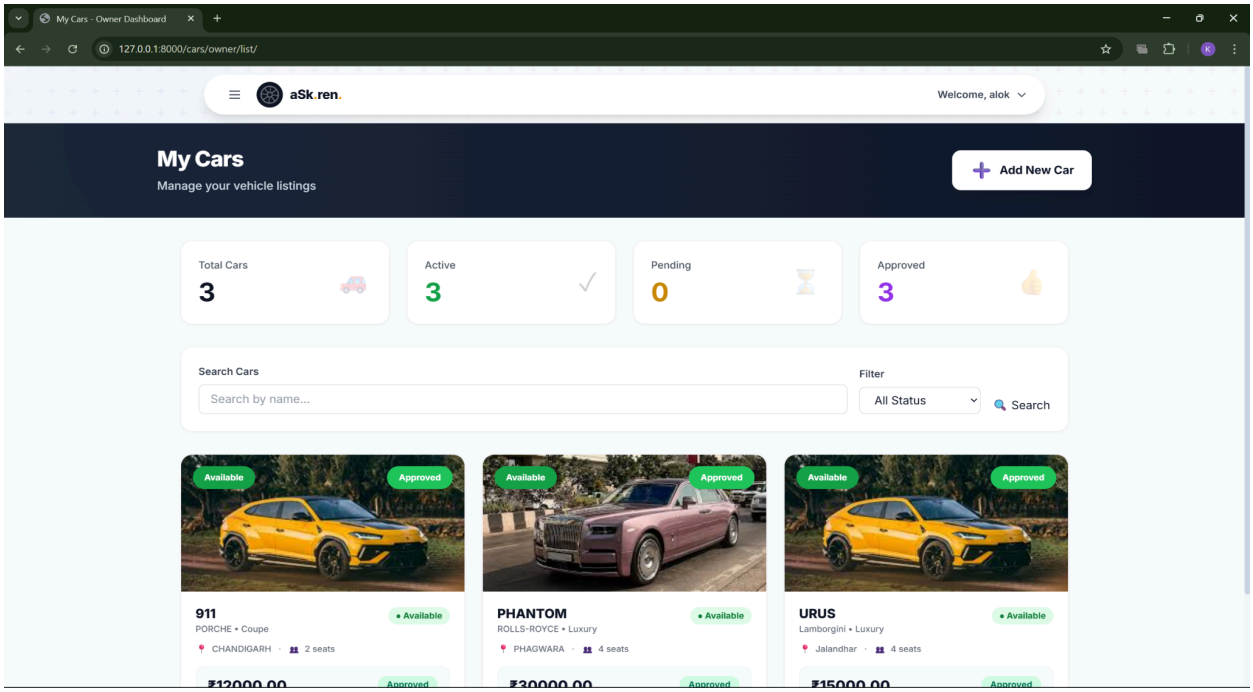


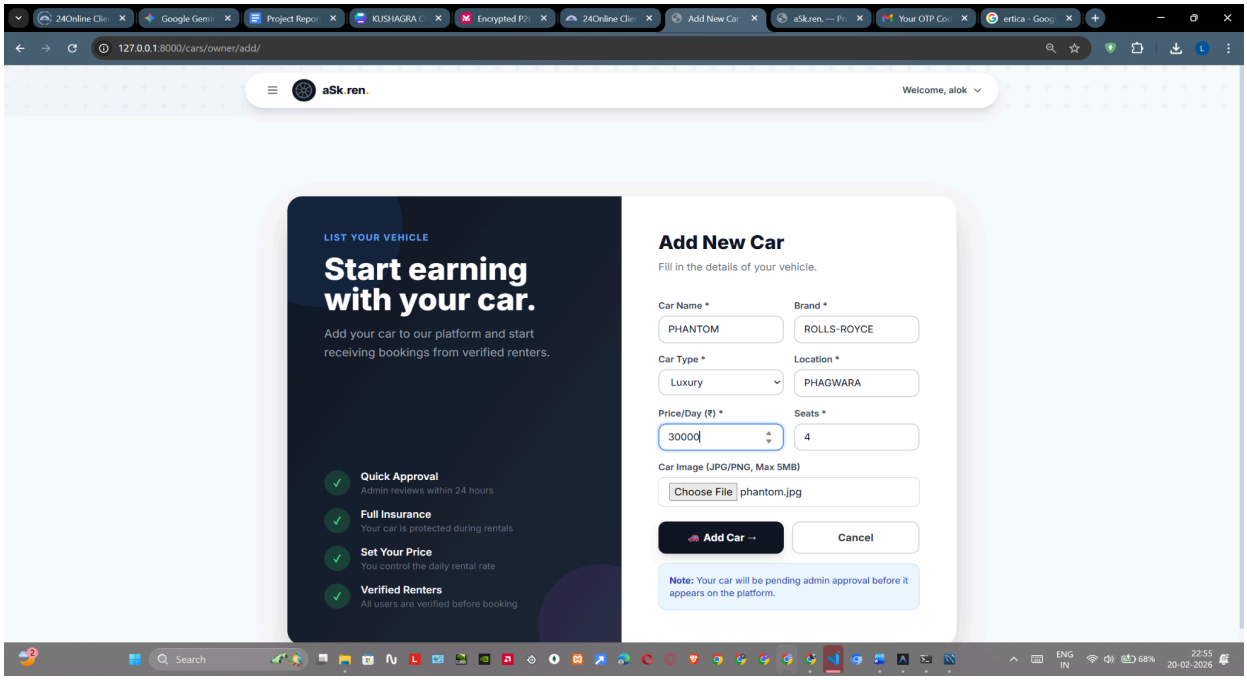
10.3 Owner

The platform's owner-side functionality offers a full suite of car-sharing business management tools. The central **Owner Dashboard** provides a high-level overview of key KPIs: total cars listed, recent bookings, upcoming reservations, earnings, and car availability, enabling quick operational assessment.

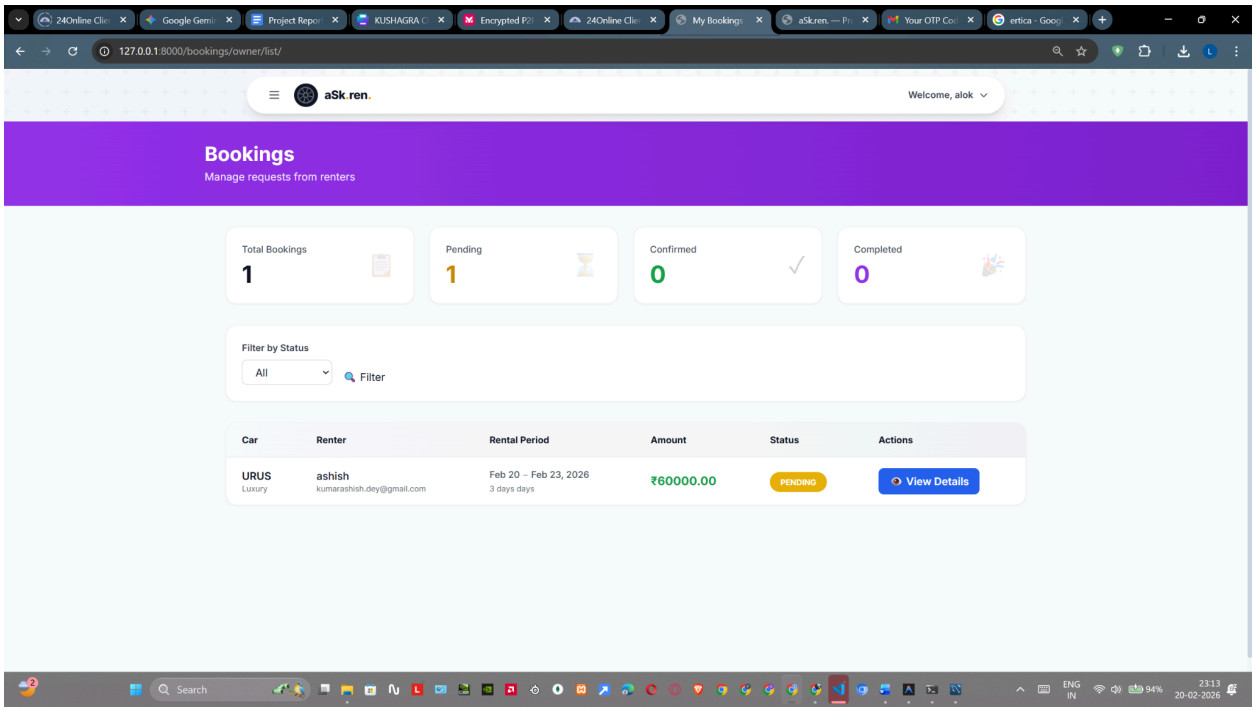


- **My Cars Management Page:**
    a. Add New Cars: List new vehicles with photos, specifications, rental rates, and pickup/drop-off locations.
    b. Manage Car Availability: Use a calendar to block out dates for unavailability and set operational hours.
    c. Edit and Update Listings: Modify car details, pricing, and photos in real-time.

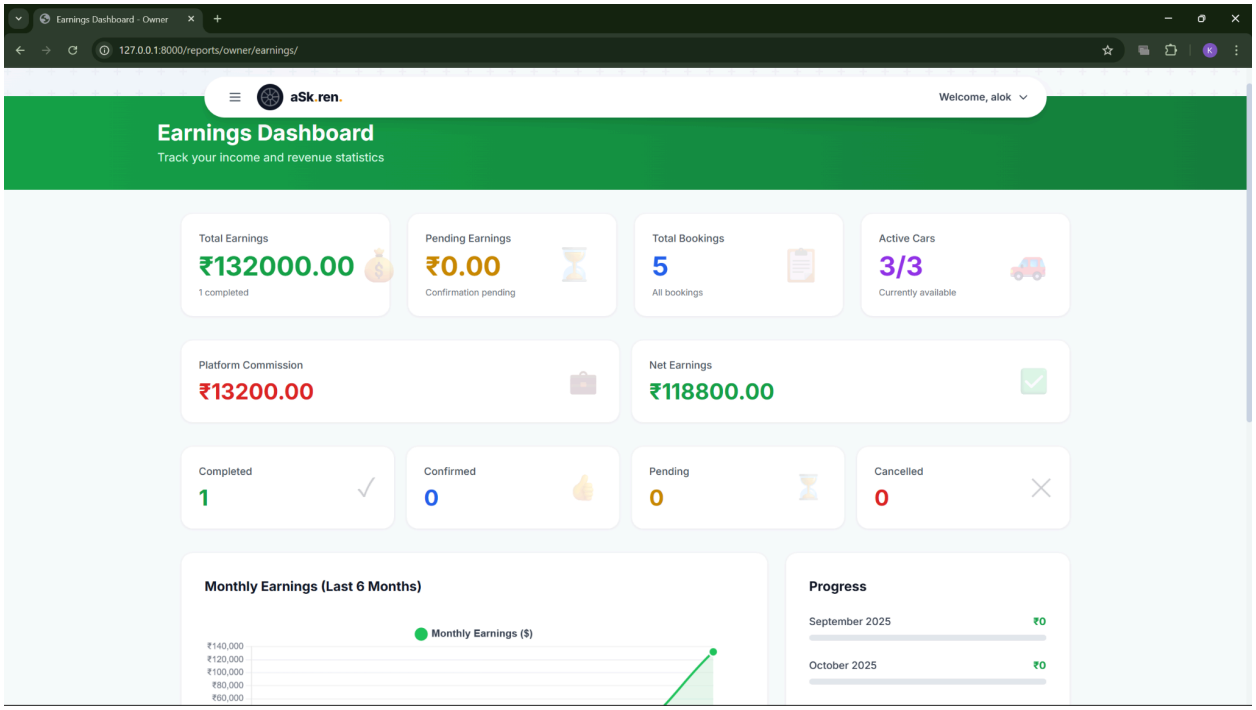




- **Booking Management Page:**
    a. Approval: Accept a request to confirm the reservation and notify the renter.
    b. Rejection: Decline a request (with an optional explanation) and inform the renter.
    c. Viewing Details: Access full booking information, including the rental period, cost, car, and renter profile/history.

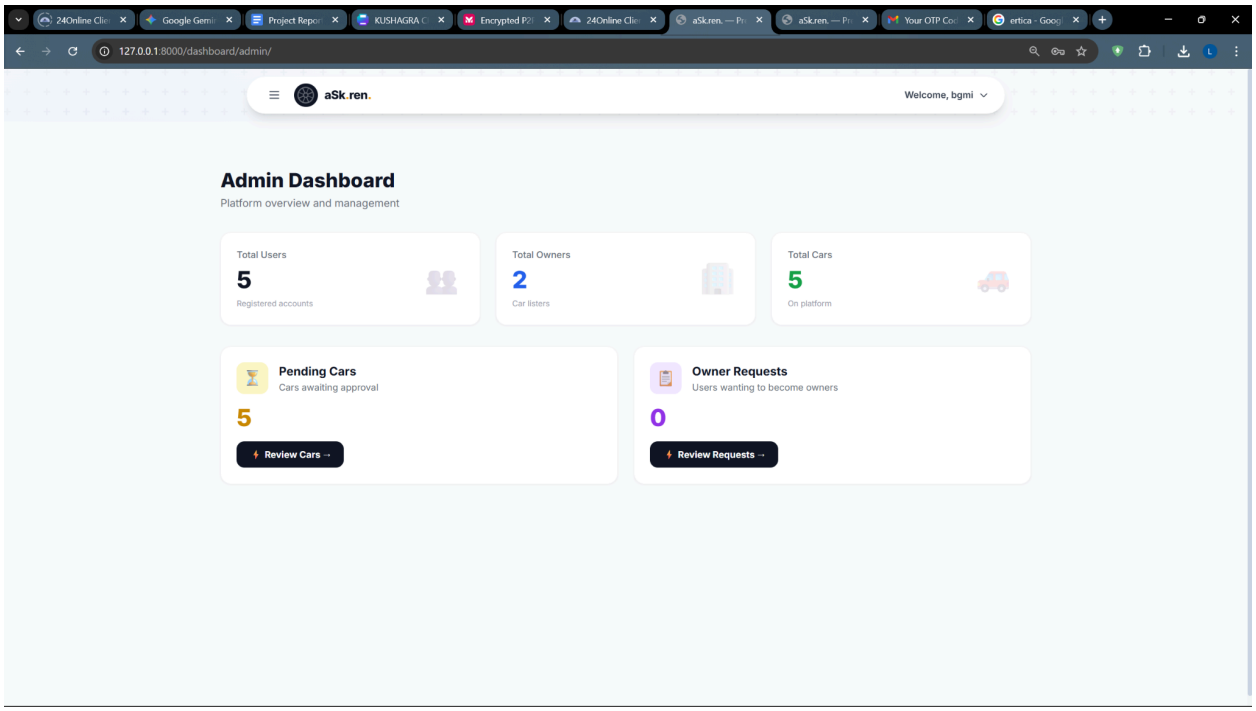


- **Earnings and Financial Reporting Page:**
    a. **Transaction History:** See completed booking records, including gross earnings, platform fees, and net payout.
    b. **Payout Tracking:** Monitor payment status (pending, processed, paid out) and review summaries.
    c. **Financial Summaries:** Access customizable (weekly, monthly, annual) graphical and tabular earnings summaries.



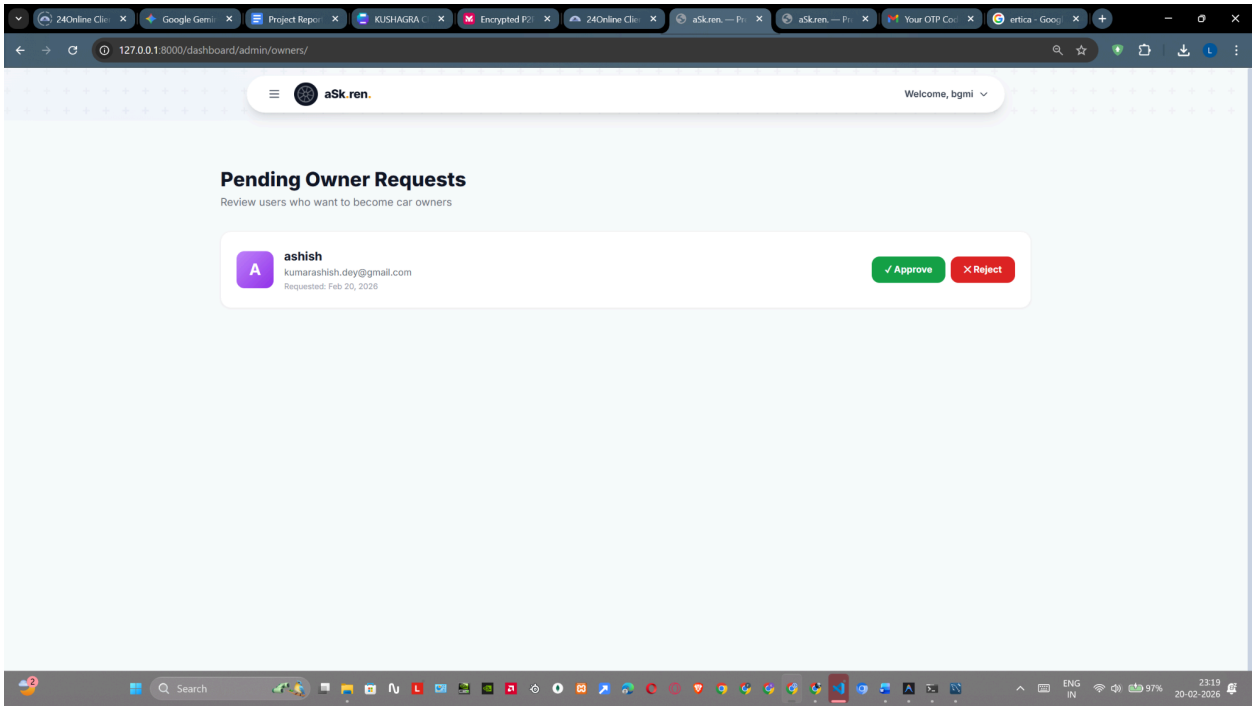
10.4 Admin

The Admin dashboard serves as the central control panel for the entire system, managing and overseeing a comprehensive set of critical functionalities essential for the platform's operation and integrity.

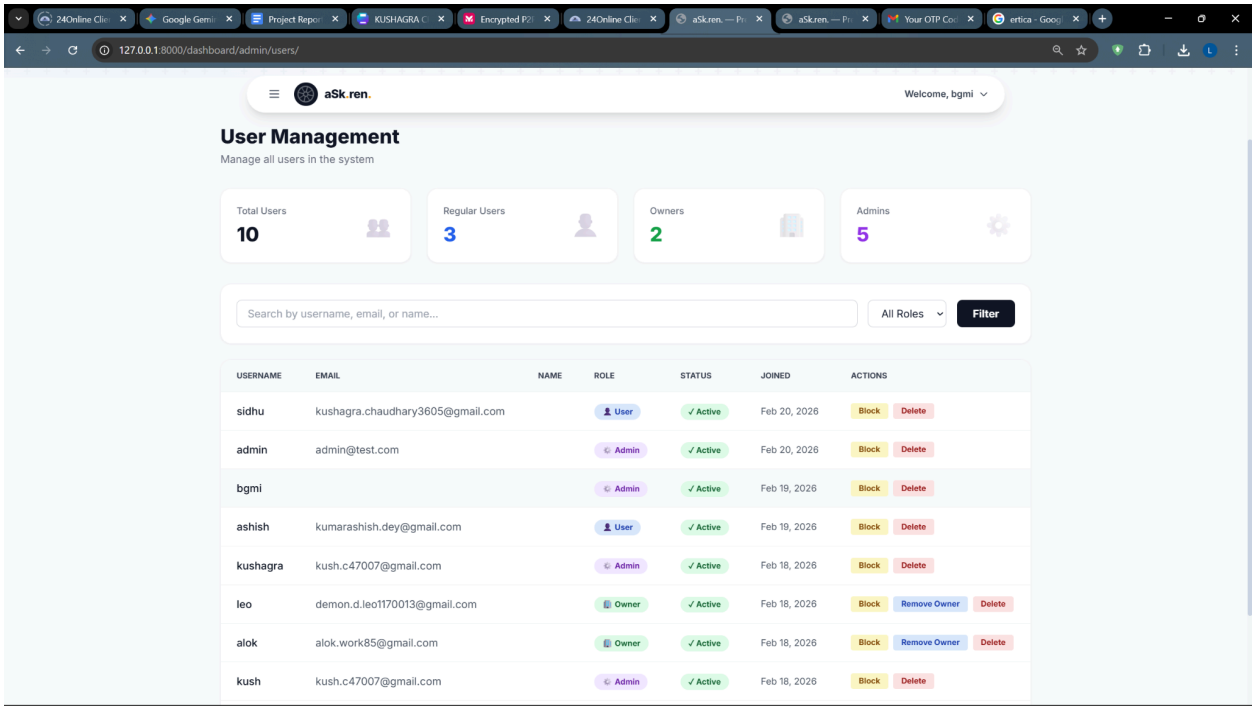


Key functionalities managed by the Admin dashboard include:

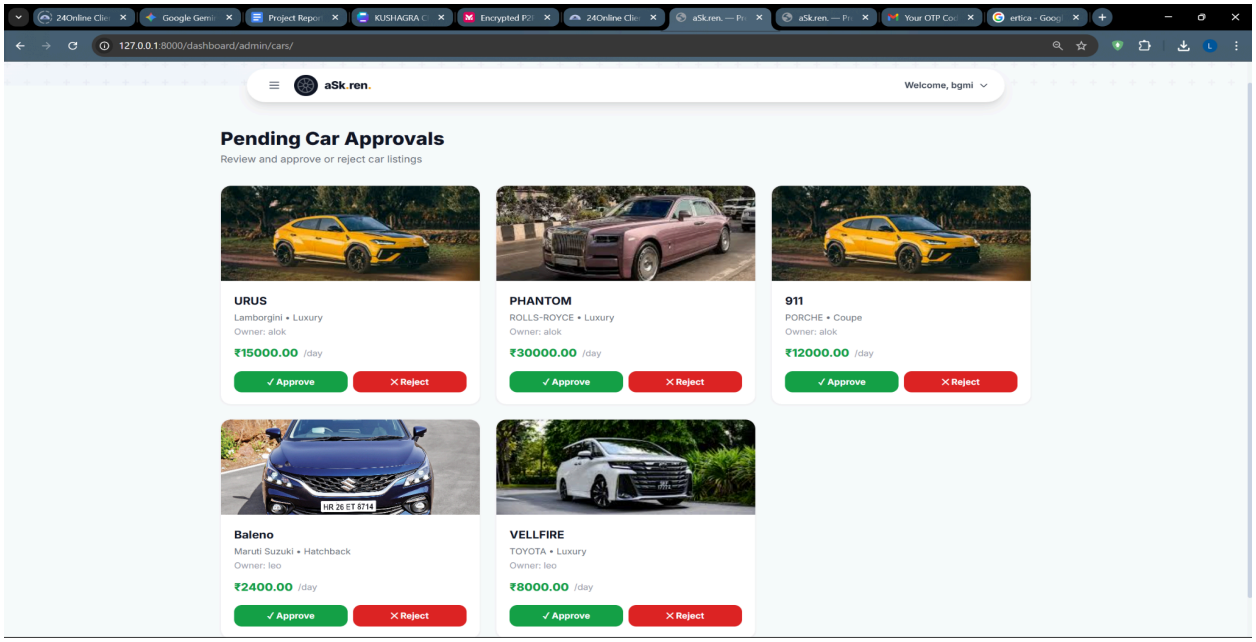
- **Owner Request Approval:** This essential module allows the admin to review and either approve or reject new car owner onboarding requests. Approval is contingent on meeting platform criteria and documentation, maintaining vehicle quality and trustworthiness.

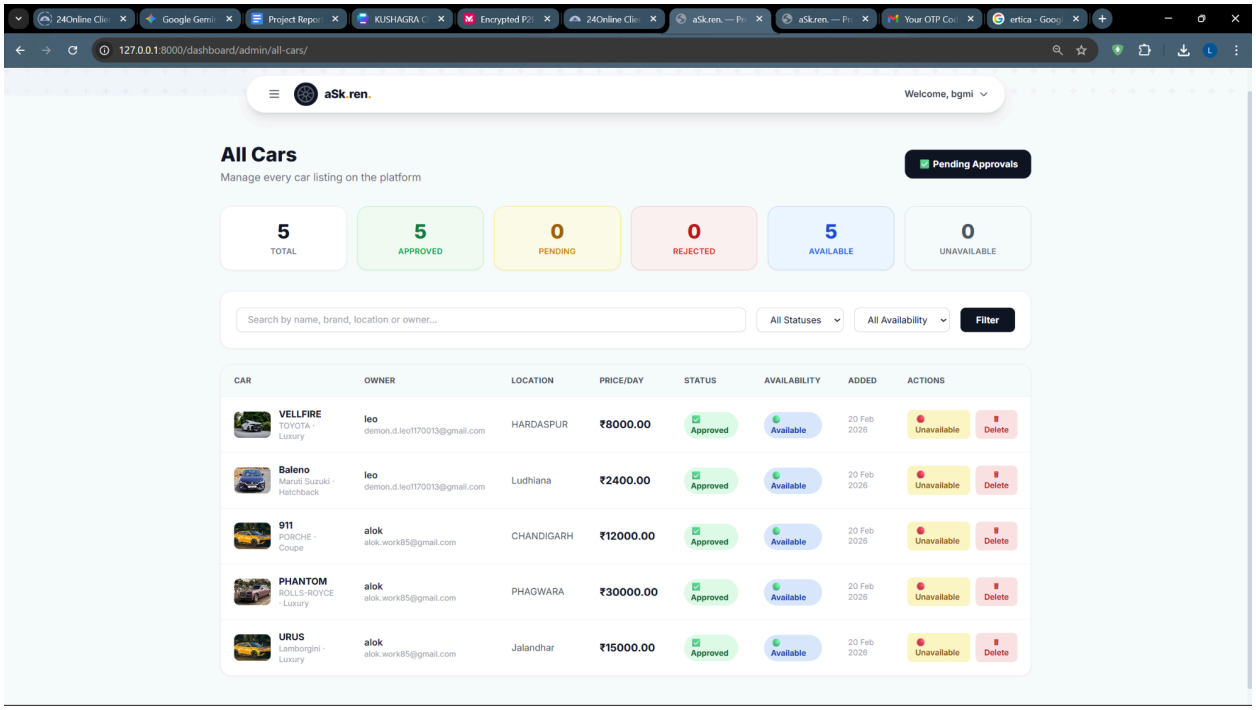


- **User Management:** The dedicated admin user management page provides full control over all registered renters and owners, including searching, viewing profiles, monitoring activity, suspending/reactivating accounts, and managing roles/permissions, ensuring a safe and regulated platform.

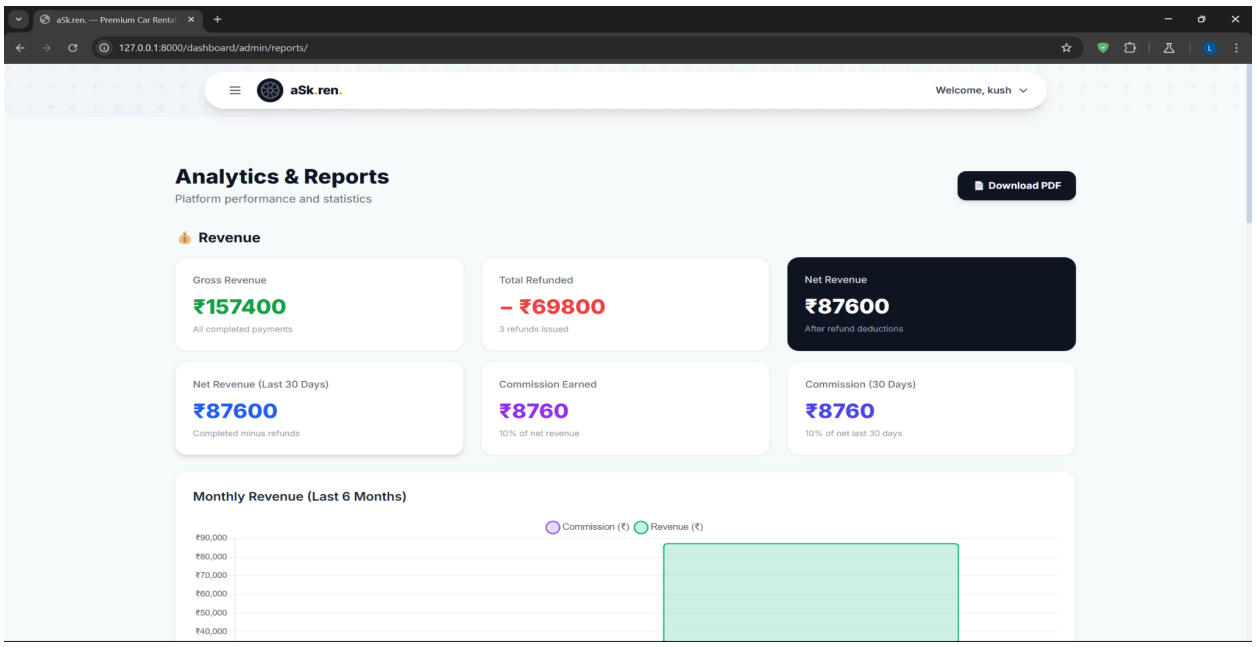


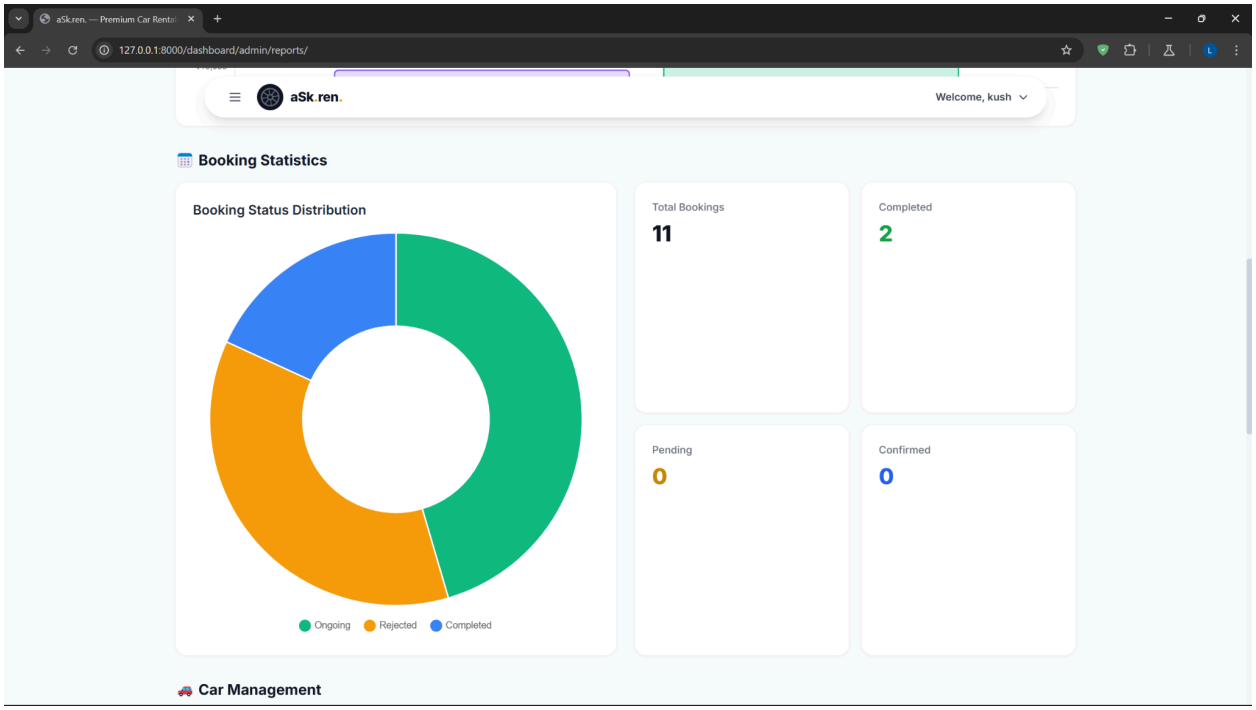
- **Car Management:** Accessible through the 'All Car' page, this section manages the complete vehicle lifecycle on the platform: initial approval (submission review, verification, compliance), availability tracking (real-time monitoring to prevent double-booking), listing maintenance (editing details, photos, pricing, temporary delisting), and permanent vehicle removal (if standards are not met or upon owner request).

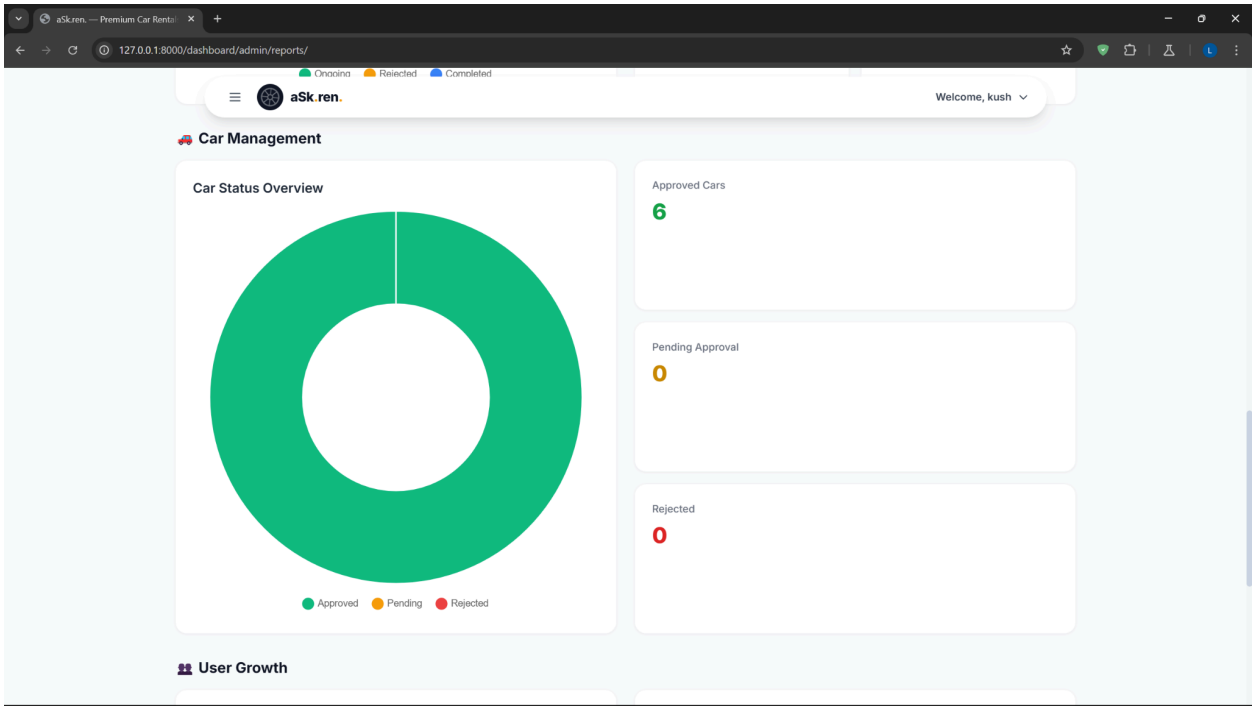


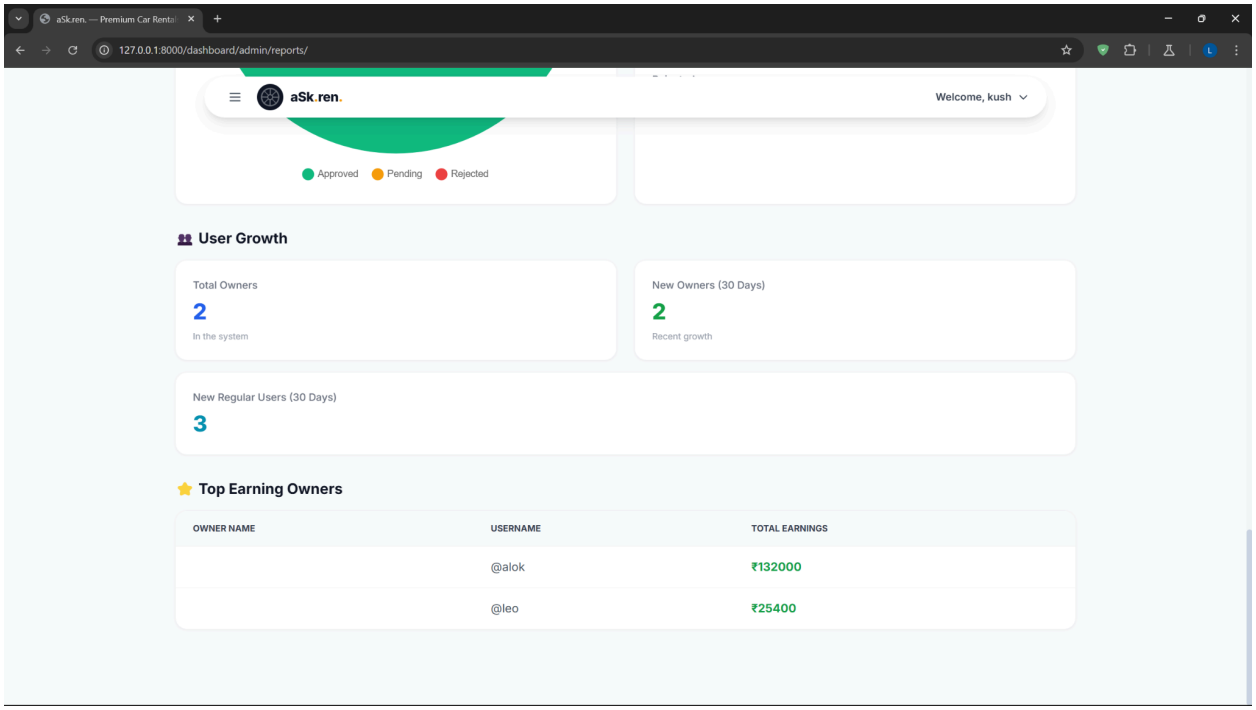


- **Analytics and Reporting:** These reports give administrators key data-driven insights across several areas:
    - **Web Page Performance:** Metrics on user engagement, traffic sources, page views, and conversion rates, aiding website optimization.
    - **Owner Performance:** Detailed reports on owner activity, including booking frequency, ratings, earnings, and compliance, enabling reward and concern resolution.

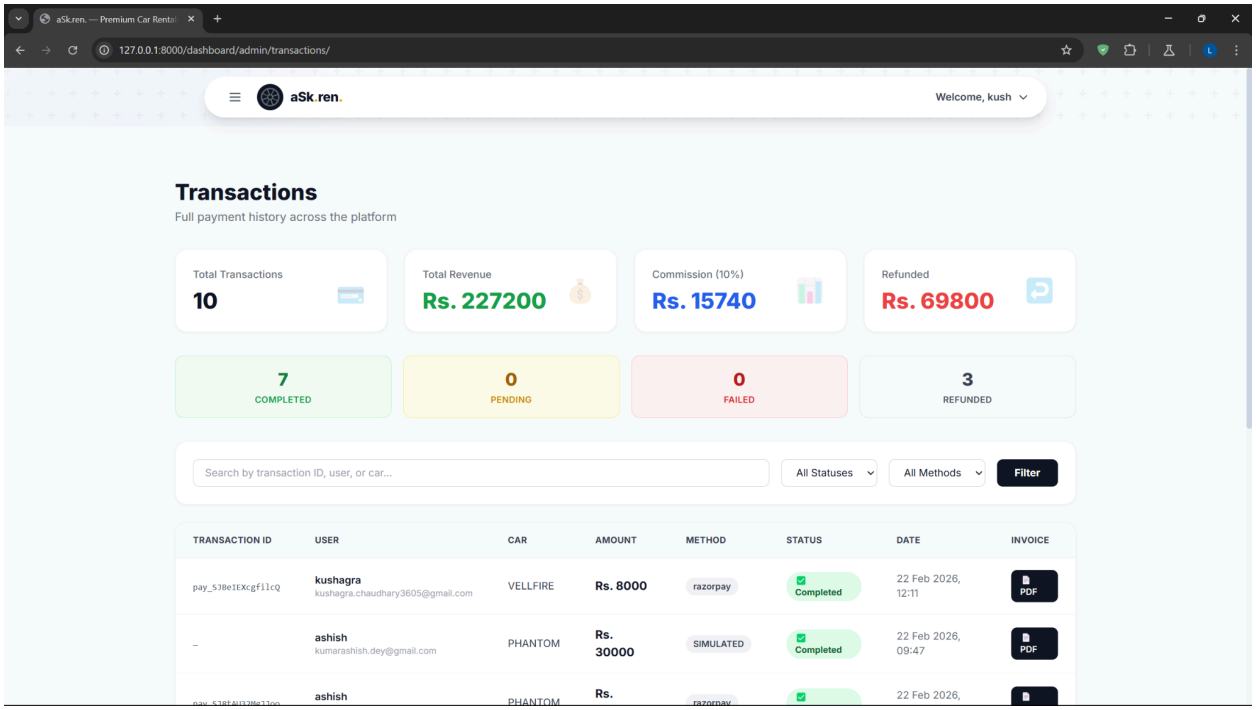



aSk.ren. — Premium Car Rental
127.0.0.1:8000/dashboard/admin/reports/
aSk.ren.
Welcome, kush
Booking Statistics
Booking Status Distribution
Ongoing
Rejected
Completed
Total Bookings
11
Completed
2
Pending
0
Confirmed
0
Car Management


aSk.ren. — Premium Car Rental
127.0.0.1:8000/dashboard/admin/reports/
aSk.ren.
Welcome, kush
Car Management
Car Status Overview
Approved
Pending
Rejected
Approved Cars
6
Pending Approval
0
Rejected
0
User Growth



- **Transaction Record:** A distinct and highly detailed transaction page logs and archives all financial data related to the platform's activities. This centralized record is essential for accounting, auditing, and dispute resolution. It captures all transaction types, including rental fees, security deposits, owner payouts, service fees, and any associated refunds or penalties, ensuring complete financial transparency and accountability.



11. Development Team Introduction

Role	Name	Student's Year	Responsibilities
Tech Lead (Full Stack Developer)	Kushagra Chaudhary	B.Tech. 3rd Year	Users, Booking & Payment features. .
Full Stack Developer	Shubh Sahu	B.Tech. 3rd Year	Accounts' dashboard and management interfaces.
Full Stack Developer	Alok Kumar	B.Tech. 3rd Year	Car management, Owner features, and reviews.

---

12. Advantages

- **Real-time Availability Tracking:** Ensures no over-bookings.
- **Secure Role-Based Access:** Enforces strict permission levels protecting personal user and payment data.
- **Scalable Modular Architecture:** Distinct Django apps make the ecosystem maintainable and flexible.

---

13. Future Enhancements

- **Mobile app development:** Bringing similar functionality to Android and iOS architectures.
- **AI-based pricing engine:** Dynamic pricing scaling proportionally to demand, weather, and time of year.
- **GPS Tracking:** Embedding tracking APIs in rental cars for precise operational oversight.
- **Advanced analytics dashboard:** Complex charts utilizing libraries like Chart.js or D3.js.

---

14. Conclusion & Next Steps

The **aSk Ren** Car Rental platform provides a secure and modular foundation for online vehicle commerce. Transitioning from legacy manual procedures to this robust Django-MySQL digital landscape yields extensive time and capital savings. Future strategic steps will prioritize deploying this application on scalable cloud infrastructure, establishing a mobile counterpart, and gradually integrating automated AI-driven logistics.
