Capstone Project - III Report

On

CAR RENTAL SYSTEM

at

U. V. Patel College of Engineering

Internal Guide :

Prof. Ketan J Sarvakar

Prepared By:

Mr. Solanki Bharat (21012011140)
Mr. Umretiya Tushar (21012011152)
Mr. Bhalodiya Maulik(21012011004)

B.Tech Semester VII (Computer Engineering)

Submitted to,

Department of Computer Engineering

U.V. Patel College of Engineering

Ganpat University, Ganpat Vidyanagar - 384 012

U.V. PATEL COLLEGE

OF ENGINEERING

CERTIFICATE

TO WHOM SO EVER IT MAY CONCERN

This is to certify that Mr. **Umretiya Tushar Pareshbhai** student of **B.Tech. Semester VII (Computer Engineering)** has completed his full semester **Capstone Project-III** work titled "**Car Rental System"** satisfactorily in partial fulfillment of the requirement of Bachelor of Technology degree of Computer Engineering of Ganpat University, Ganpat Vidyanagar, Mehsana in the year 2024-2025.

College Project Guide

Dr. Paresh Solanki,
Head, Computer Engineering

Sign

Prof. Ketan J Sarvakar

U.V. PATEL COLLEGE

OF ENGINEERING

CERTIFICATE

TO WHOM SO EVER IT MAY CONCERN

This is to certify that Mr. **Solanki Bharat Joravarsinh** student of **B.Tech. Semester VII (Computer Engineering)** has completed his full semester **Capstone Project-III** work titled "**Car Rental System"** satisfactorily in partial fulfillment of the requirement of Bachelor of Technology degree of Computer Engineering of Ganpat University, Ganpat Vidyanagar, Mehsana in the year 2024-2025.

College Project Guide

Dr. Paresh Solanki,
Head, Computer Engineering

Sign

Prof. Ketan J Sarvakar

U.V. PATEL COLLEGE

OF ENGINEERING

Ganpat University
॥ विद्यया समाजोत्कर्षः ॥

National Assessment and Accreditation Council
NAAC
A
Grade
Accredited University

CERTIFICATE

TO WHOM SO EVER IT MAY CONCERN

This is to certify that Mr. **Bhalodiya Maulik Harshadbhai** student of **B.Tech. Semester VII (Computer Engineering)** has completed his full semester **Capstone Project-III** work titled "**Car Rental System**" satisfactorily in partial fulfillment of the requirement of Bachelor of Technology degree of Computer Engineering of Ganpat University, Ganpat Vidyanagar, Mehsana in the year 2024-2025.

College Project Guide

Dr. Paresh Solanki,
Head, Computer Engineering

Sign

Prof. Ketan J Sarvakar

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to U. V. Patel College of Engineering, Ganpat University, Kherva**,** for providing me with the opportunity to develop my technical and innovative skills, which have been crucial in the successful completion of the Car Rental System project. This project has allowed me to combine the strengths of traditional software development techniques with modern technologies, ensuring performance, reliability, and security, utilizing HTML, CSS, JavaScript, Bootstrap, PHP. The support and infrastructure provided by the university were instrumental in achieving this.

I am sincerely very thankful to Prof. Ketan Sarvakar Faculty at Ganpat University. By providing very valuable time and giving guidance and opportunities to work together as a team, and to do a project in the Web Development Field. I Am very thankful to him for encouraging me to innovate my project to the highest peak.

I am very thankful to all the competitive students who are also very helpful to work together not as a competitor but as a family member very joyfully.

Thank you

ABSTRACT

The Car Rental Management System is designed to streamline and enhance the car rental process for both administrators and customers by providing a user-friendly, efficient web-based platform. This system simplifies vehicle management for administrators by offering tools to list, update, and manage the availability of rental cars. For customers, it provides an intuitive interface to search for suitable vehicles, make reservations, and communicate with the rental service. Key features include vehicle listing management, advanced search and filtering options. Accessible through major web browsers and optimized for both desktop and mobile devices, the system ensures a smooth and reliable experience for its users. Built with a strong emphasis on security, performance, and usability, the Car Rental Management System provides a robust platform that meets the diverse needs of car rental businesses and customers alike, enhancing the overall rental experience.

INDEX

LIST OF FIGURES

CHAPTER-1: INTRODUCTION

The Car Rental System is a web-based platform designed to streamline the process of renting vehicles by offering a centralized, user-friendly interface for both car rental companies and customers. The system allows rental companies to efficiently list, manage, and update their available vehicles, while customers can easily search for cars, make reservations, and communicate with the rental service. With integrated features such as vehicle search filters, booking management, and payment tracking, the system aims to simplify car rental operations and enhance the overall user experience for both parties. This document outlines the requirements necessary for developing a secure, reliable, and efficient car rental solution that meets the needs of all stakeholders involved in the vehicle rental process.

CHAPTER-2: PROJECT SCOPE

The Car Rental System project aims to develop a web-based platform that streamlines the vehicle rental process by allowing rental companies to list, manage, and update their available cars while enabling customers to search, book, and communicate effectively with rental providers. Key features will include vehicle management tools, advanced search and filtering options for customers, integrated booking and reservation systems, and secure payment processing capabilities. The system will also provide user registration, profile management, and administrative tools for overseeing operations and ensuring compliance with security and data protection regulations. The project scope will not include the development of dedicated mobile applications, instead focusing on a responsive web interface accessible via standard web browsers across various devices.

CHAPTER-3: FEASIBILITY ANALYSIS

3.1 Technical Feasibility

- **Technical Requirements:** Assess the infrastructure required for the Car Rental System, including server capacity, database management systems, network requirements, and software tools for development. Ensure that current technologies can support features like car listings, booking management, payment gateways, and user authentication.
- **Development Team:** Evaluate the availability of skilled web developers, UI/UX designers, and database administrators with expertise in technologies suitable for HTML, JavaScript, CSS, Bootstrap and PHP.
- **Technology Stack:** Select the optimal technology stack that includes scalability, security, and cross-browser compatibility. This could include front-end frameworks like HTML and CSS, back-end development tools like PHP.
- **Data Management:** Determine how vehicle and customer data will be securely collected, stored, and managed. Plan for integration with external systems (such as third-party payment providers) and ensure secure data transmission.
- **Security and Compliance:** Evaluate the necessary security protocols to safeguard user information, including encryption, secure payment processing, and user authentication. Ensure compliance with data privacy regulations like GDPR to protect sensitive customer data.
- **Scalability:** Assess whether the platform can accommodate future growth in terms of vehicle listings, number of users, bookings, and other transactions. Implementing scalable cloud hosting solutions (e.g., AWS, Azure) can help in managing growth without performance issues.

3.2 Economic Feasibility

- **Cost Analysis:** Estimate the total cost of developing and maintaining the Car Rental System, which includes software development, hosting, database management, and third-party API costs (e.g., payment gateways). Operational costs like marketing, customer support, and system upgrades also need to be considered.
- **Revenue Projections:** Forecast potential revenue streams, such as rental commissions, subscription fees for users, premium listings for rental companies, and advertising revenue. The system could also generate income from late fee charges or premium features offered to customers.
- **Return on Investment (ROI):** Calculate the expected ROI by comparing the initial investment and operational costs with the projected revenue. Estimate the break-even point based on user adoption rates, vehicle bookings, and market trends.
- **Funding Sources:** Identify possible funding sources to support the development and scaling of the Car Rental System, such as self-funding, venture capital, partnerships with vehicle rental companies, or loan financing.

3.3 Operational Feasibility

- **Market Analysis:** Conduct market research to understand the demand for online vehicle rental services. Analyse competitors, such as traditional rental agencies and peer-to-peer car-sharing platforms, and identify the unique value propositions of the Car Rental System (e.g., ease of use, competitive pricing, and vehicle variety).
- **Business Processes:** Evaluate how the system will integrate with existing car rental businesses' workflows, including vehicle availability management, booking and payment processing, and customer service. Ensure smooth integration with existing fleet management software, if applicable.
- **User Acceptance:** Gather feedback from potential users, including rental companies, independent vehicle owners, and customers, to assess their interest in using the platform. Conduct pilot testing to determine user satisfaction and adoption rates.
- **Change Management:** Develop a plan for training rental agencies and vehicle owners to use the system effectively. Offer customer support and documentation to guide users through the system's features, such as booking, payment, and managing rentals. Prepare for any operational adjustments needed during the system's rollout.

CHAPTER-4. SOFTWARE AND HARDWARE REQUIREMENT

4.1 Software Requirements

1. **Operating Systems**
   - **Server:** Windows Server.
   - **Client:** Windows.

2. **Database Management System (DBMS)**
   - MySQL For Relational Database Management System.

3. **Web Server**
   - Apache HTTP Server or Nginx

4. **Programming Languages and Frameworks**
   - **Backend:** PHP
   - **Frontend:** HTML5, CSS3, JavaScript.

4.2 Hardware Requirements

1. **Server Hardware**
   - **Processor**: Multi-core (4 cores minimum, 8 cores recommended)
   - **Memory**: 16 GB RAM minimum (32 GB recommended)
   - **Storage**:
     - SSD for fast read/write operations (500 GB minimum, scalable based on data needs)
     - Additional storage for backups and archival (1 TB HDD recommended)
   - **Network**: High-speed internet connection with at least 1 Gbps bandwidth
2. **Client Hardware**
   - **Processor**: Dual-core or higher
   - **Memory**: 4 GB RAM minimum (8 GB recommended)
   - **Storage**: 256 GB SSD or higher
   - **Display**: 1080p resolution or higher

3. **Backup and Recovery**
    - Regular automated backups to cloud storage or external drives
    - Disaster recovery plan with offsite backups

4. **Networking Hardware**
    - High-speed routers and switches
    - Firewall and VPN for secure remote access
5. **Development Workstations**
    - **Processor**: Quad-core or higher
    - **Memory**: 16 GB RAM minimum
    - **Storage**: 512 GB SSD or higher
    - **Graphics**: Dedicated GPU for graphical tasks (optional)
6. **Additional Hardware**
    - Uninterruptible Power Supply (UPS) for servers and critical hardware
    - Redundant servers for high availability and load balancing

CHAPTER-5: FUNCTIONAL & NON FUNCTIONAL REQUIREMENTS

5.1 Functional Requirements:

5.1.1 Admin Module:

Admin Sign-In

- **Input:**
  - Admin enters a valid username and password.
- **Process:**
  - The system verifies the credentials against the database.
  - Authentication is performed to ensure the admin has the appropriate access rights.
- **Output:**
  - Successful login grants access to the admin dashboard.
  - Unsuccessful login provides an error message indicating invalid credentials.

User Management

- **View Users**
  - **Input:**
    - Admin requests to view the list of all registered users.
  - **Process:**
    - The system retrieves user details from the database.
  - **Output:**
    - A list of users is displayed with their details (e.g., username, email).
- **Add User**
  - **Input:**
    - Admin provides user details such as username, email, password, and optional contact information.
  - **Process:**
    - The system validates the provided details and adds the new user to the database.
  - **Output:**
    - Confirmation of successful user addition.
- **Update User**
  - **Input:**
    - Admin modifies user information, such as contact details or password.
  - **Process:**
    - The system updates the user details in the database.
  - **Output:**
    - Confirmation of successful user update.

- **Delete User**
  - **Input:**
    - Admin selects a user to delete.
  - **Process:**
    - The system removes the selected user from the database.
  - **Output:**
    - Confirmation of successful user deletion.

Vehicle Management

- **View Vehicles**
  - **Input:**
    - Admin requests to view the list of all vehicles.
  - **Process:**
    - The system retrieves vehicle details from the database.
  - **Output:**
    - A list of vehicles is displayed with details such as make, model, year, and availability.
- **Add Vehicle**
  - **Input:**
    - Admin provides vehicle details such as make, model, year, color, rental rates, and availability.
  - **Process:**
    - The system validates and adds the vehicle information to the database.
  - **Output:**
    - Confirmation of successful vehicle addition.
- **Update Vehicle**
  - **Input:**
    - Admin modifies vehicle details such as rental rates or availability.
  - **Process:**
    - The system updates the vehicle details in the database.
  - **Output:**
    - Confirmation of successful vehicle update.
- **Delete Vehicle**
  - **Input:**
    - Admin selects a vehicle to delete.
  - **Process:**
    - The system removes the selected vehicle from the database.
  - **Output:**
    - Confirmation of successful vehicle deletion.

Booking Management

- **View Bookings**
  - **Input:**
    - Admin requests to view the list of all bookings.

- **Process:**
  - The system retrieves booking details from the database.
- **Output:**
  - A list of bookings is displayed with details such as user information, vehicle details, rental dates, and status.

- **Update Booking**
  - **Input:**
    - Admin modifies booking information such as rental dates or vehicle assignments.
  - **Process:**
    - The system updates the booking details in the database.
  - **Output:**
    - Confirmation of successful booking update.
- **Cancel Booking**
  - **Input:**
    - Admin selects a booking to cancel.
  - **Process:**
    - The system updates the booking status to canceled in the database.
  - **Output:**
    - Confirmation of successful booking cancellation.

Brand Management

- **View Brands**
  - **Input:**
    - Admin requests to view the list of all car brands.
  - **Process:**
    - The system retrieves brand details from the database.
  - **Output:**
    - A list of brands is displayed with details such as brand name and associated vehicles.
- **Add Brand**
  - **Input:**
    - Admin provides brand details including brand name and description.
  - **Process:**
    - The system validates and adds the brand information to the database.
  - **Output:**
    - Confirmation of successful brand addition.
- **Update Brand**
  - **Input:**
    - Admin modifies brand details such as name or description.
  - **Process:**
    - The system updates the brand details in the database.
  - **Output:**
    - Confirmation of successful brand update.
- **Delete Brand**
  - **Input:**

        - Admin selects a brand to delete.
    - **Process:**
        - The system removes the selected brand from the database.
    - **Output:**
        - Confirmation of successful brand deletion.

Contact Management

- **View Contacts**
    - **Input:**
        - Admin requests to view all contact requests and feedback.
    - **Process:**
        - The system retrieves contact details from the database.
    - **Output:**
        - A list of contact requests and feedback is displayed.
- **Respond to Contacts**
    - **Input:**
        - Admin provides responses or solutions to user inquiries.
    - **Process:**
        - The system sends responses to users and updates contact records.
    - **Output:**
        - Confirmation of response sent and update in contact records.

5.1.2 User Module:

User Registration

- **Input:**
    - User provides:
        - Full Name
        - Mobile Number (used as username)
        - Email Address
        - Password
        - Optional Contact Information (e.g., address, secondary phone number)
- **Process:**
    - **Validation:**
        - Check if the mobile number is unique and not already registered.
        - Validate the email address format.
        - Ensure the password meets security criteria (e.g., length, complexity).
    - **Database Operations:**
        - Hash the password for secure storage.
        - Save user details (including hashed password) in the database.
- **Output:**
    - **Success:**

- Display a confirmation message that registration was successful.
- Redirect the user to the login page or homepage.

- **Failure:**
  - Display an error message indicating issues (e.g., mobile number already in use, invalid email format).
  - Prompt the user to correct and resubmit the information.

User Login

- **Input:**
  - User provides:
    - Mobile Number (username)
    - Password
- **Process:**
  - **Validation:**
    - Verify the provided mobile number and password against stored credentials in the database.
  - **Authentication:**
    - Check if the credentials are correct.
    - Create a session or token for authenticated access.
- **Output:**
  - **Success:**
    - Redirect the user to their dashboard or profile.
    - Provide access to user-specific functionalities.
  - **Failure:**
    - Display an error message indicating invalid credentials.
    - Allow the user to retry login or initiate password recovery.

Search Car

- **Input:**
  - User provides search criteria:
    - Car Brand
    - Car Model
    - Rental Location
    - Rental Dates
    - Car Type (e.g., sedan, SUV)
- **Process:**
  - **Search Execution:**
    - Query the database for cars matching the search criteria.
    - Filter results based on availability and other parameters.
- **Output:**
  - **Success:**

        - Display a list of available cars that match the search criteria, including details like car model, brand, rental rate, and availability.
    - **Failure:**
        - Display a message indicating no cars are available that match the search criteria.
        - Offer options to modify search criteria or check for other locations/dates.

Book Car

- **Input:**
    - User provides:
        - Selected Car (from search results)
        - Rental Start Date
        - Rental End Date
        - Pickup Location
        - Drop-off Location
        - Additional Options (e.g., insurance, GPS)
- **Process:**
    - **Validation:**
        - Ensure the selected car is available for the specified dates.
        - Confirm that pickup and drop-off locations are valid.
        - Verify additional options and calculate total cost.
    - **Booking Execution:**
        - Save booking details in the database.
        - Generate a unique booking confirmation number.
        - Update car availability status.
- **Output:**
    - **Success:**
        - Display a booking confirmation message with details and a unique booking number.
        - Send an email or SMS confirmation to the user.
    - **Failure:**
        - Display an error message indicating issues with booking (e.g., car no longer available).
        - Prompt the user to modify the booking details or retry.

5.2 Non Functional Requirements:

Performance Requirements: Search results for available cars should be returned in less than 3 seconds.

The booking confirmation process, including payment validation, should complete in under 5 seconds.

Usability: The system should be accessible from different devices, including desktops, tablets, and smartphones (responsive design).

Provide clear and concise error messages and tooltips to guide users through the reservation process.

Maintainability: Modular architecture for easy updates, with thorough documentation and automated testing.

Security Requirements: Passwords must be hashed using secure algorithmsThe system should log failed login attempts and enforce account lockout after 5 consecutive failed attempts.

Scalability: The system should be able to scale horizontally by adding more servers or cloud resources to handle increased user load.The system should accommodate expansion to multiple locations.

CHAPTER-6: SYSTEM DESIGN

6.1 Use Case Diagram:

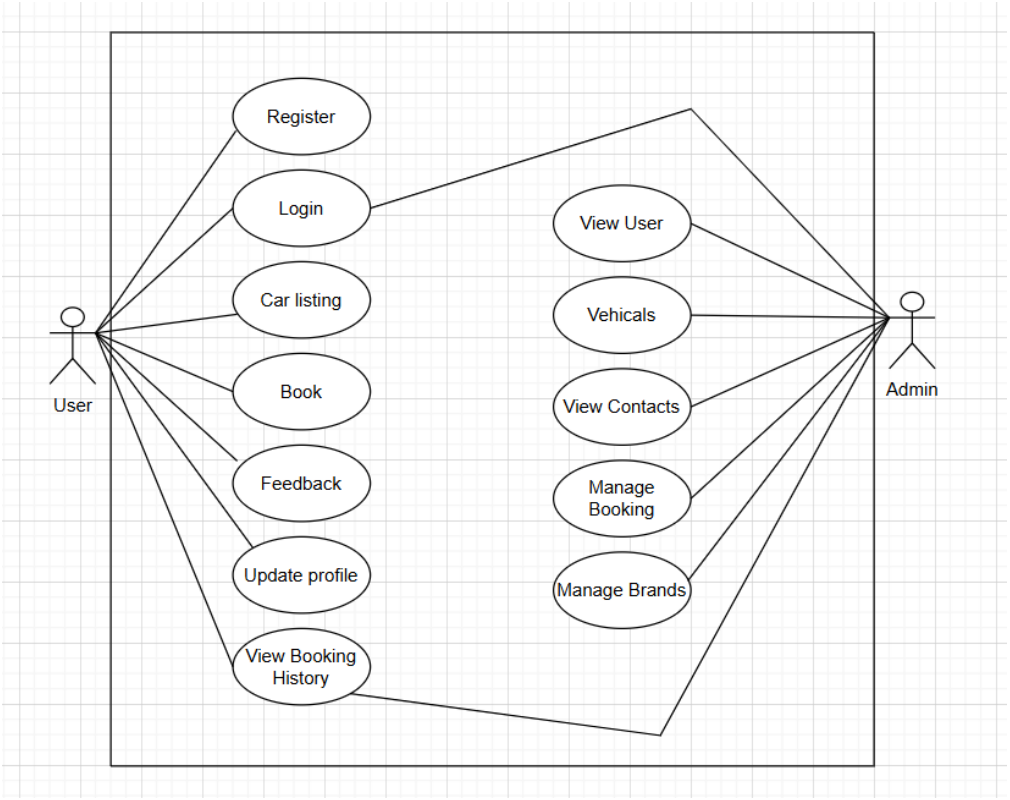

Figure 6.1: Use Case Diagram

The use case diagram for the Car Rental System shows two actors: User and Admin. Users can register, log in, view car listings, book vehicles, update profiles, view booking history, and provide feedback. Admins manage users, vehicles, contacts, bookings, and car brands, ensuring efficient system operations for both parties.

6.2 Class Diagram:

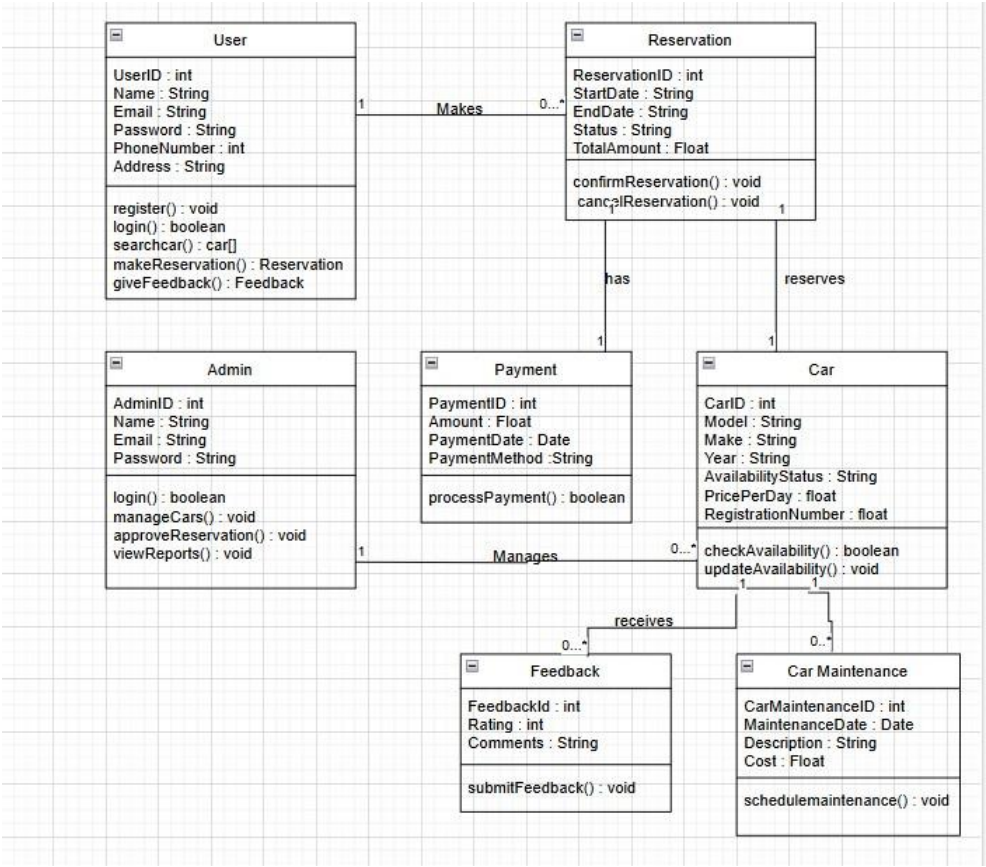

Figure 6.2: Class Diagram

The class diagram for the Car Rental System illustrates key entities: User, Admin, Car, Reservation, Payment, Feedback, and Car Maintenance. Users can register, log in, search cars, make reservations, and provide feedback. Admins manage cars, approve reservations, and view reports. Cars are linked to reservations, payments, and maintenance records.

6.3 Activity Diagram:

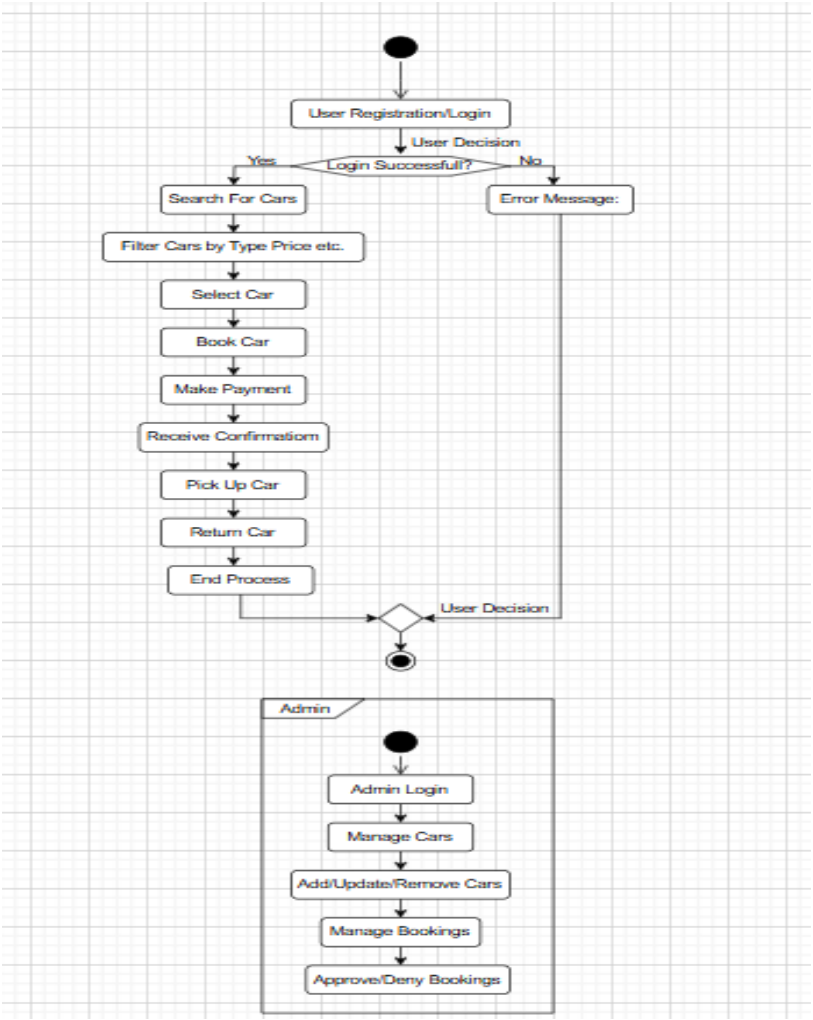

Figure 6.3: Activity Diagram

The activity diagram illustrates the process flow for both users and admins in a car rental system. Users can register, log in, search for cars, filter by criteria, book, make payments, and return cars. Admins log in, manage cars by adding, updating, or removing listings, and approve or deny bookings.

6.4 DFD Level-0 Diagram:

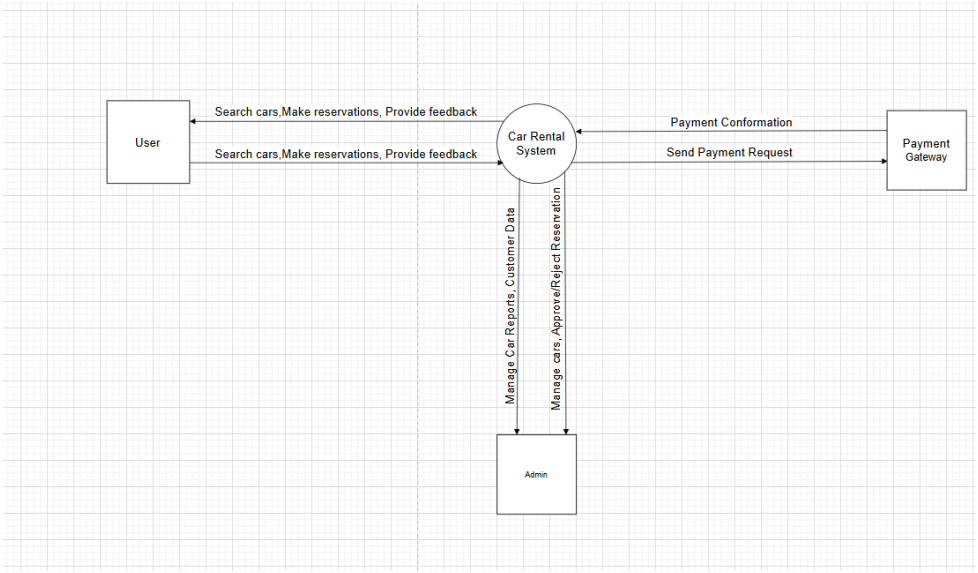

Figure 6.4: DFD Level-0 Diagram

The DFD Level 0 diagram illustrates a car rental system. The user interacts with the system to search cars, make reservations, and provide feedback. The system communicates with a payment gateway for transactions and an admin for managing cars and reservations. Admins handle car reports, customer data, and reservation approvals.

6.5 DFD Level-1 Diagram:

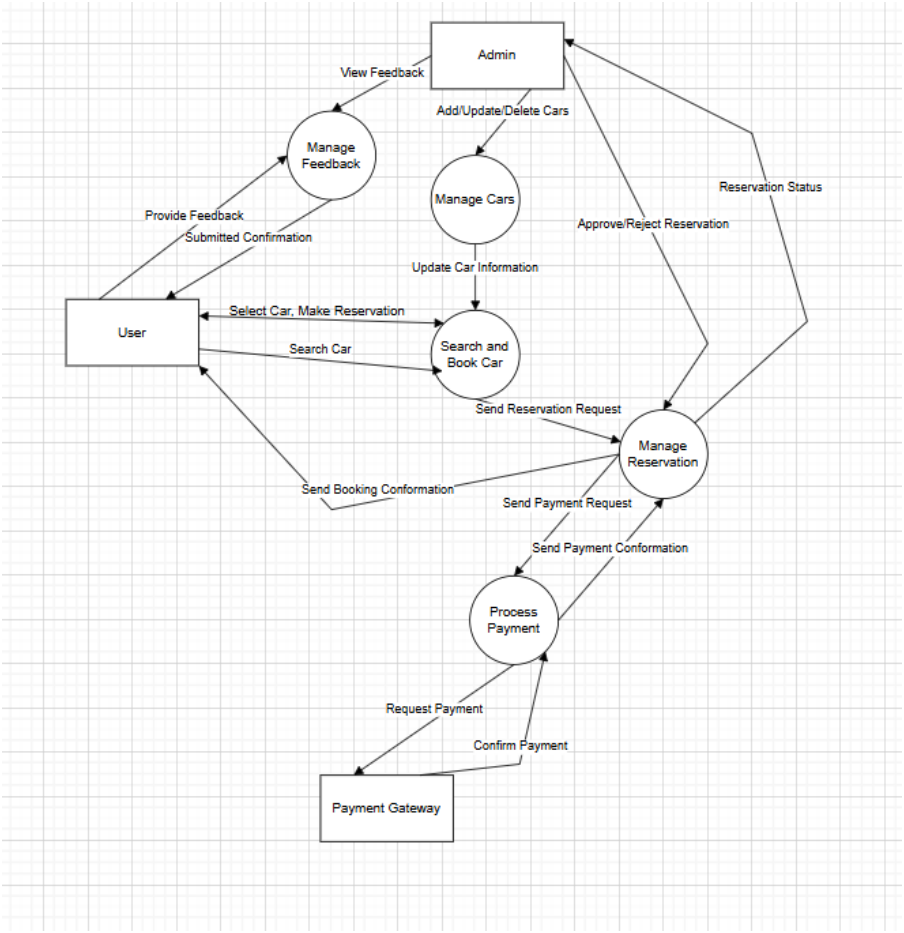

Figure 6.5 DFD Level-1 Diagram

The DFD Level-1 diagram depicts a car rental system's detailed processes. Users search for cars, make reservations, and provide feedback. Admins manage cars, reservations, and feedback. The system interacts with a payment gateway to process payments. Key processes include car management, reservation approval/rejection, feedback handling, and payment processing, ensuring smooth operations.

CHAPTER-7: PROTOTYPE

7.1 Admin Sign-In Page:

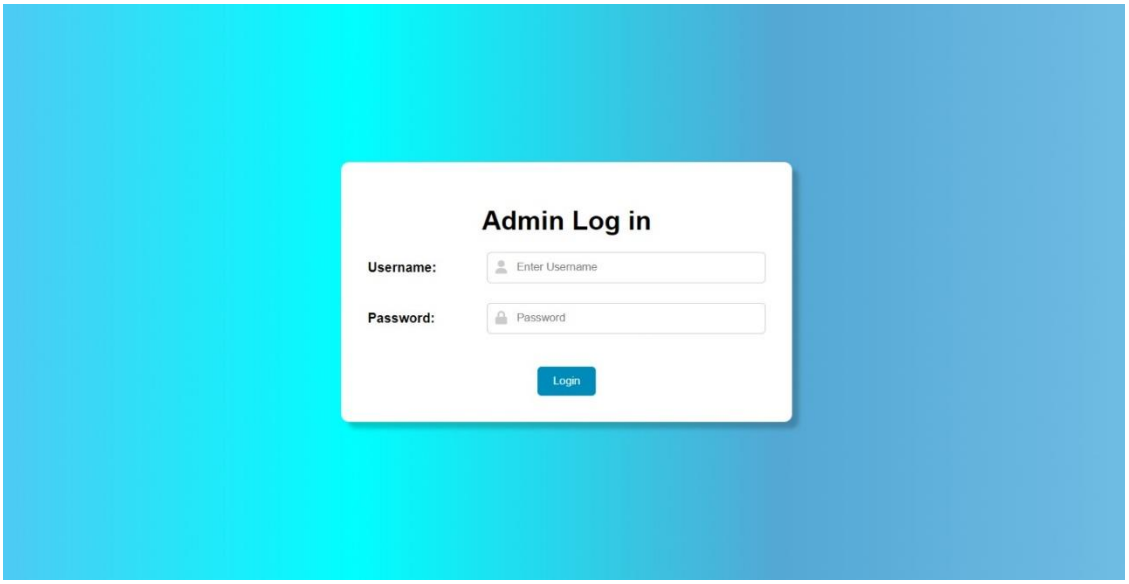

Figure 7.1: Admin Sign-In Page

It has two input fields for the admin to enter their username and password. Below the fields, there is a blue "Login" button for authentication. A link at the bottom provides navigation back to the home page.

7.2 Admin Dashboard Page:

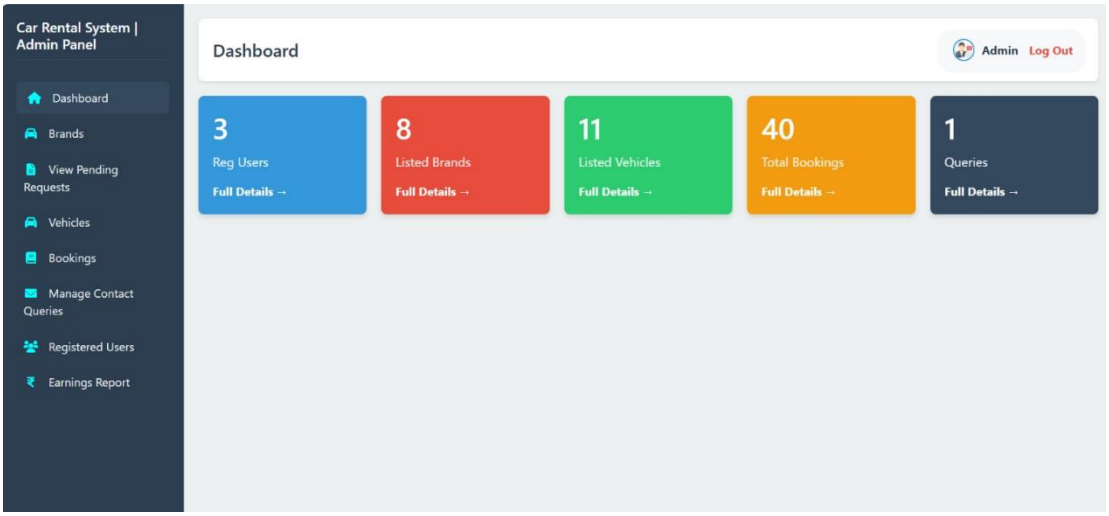

Figure 7.2: Admin Dashboard Page

The admin dashboard for a car rental system provides an overview of key metrics, including listed vehicles, bookings, brands, subscribers, testimonials, and queries. It allows for easy navigation and management of various aspects of the system, such as brands, vehicles, bookings, and user information.

7.3 Booking Page:

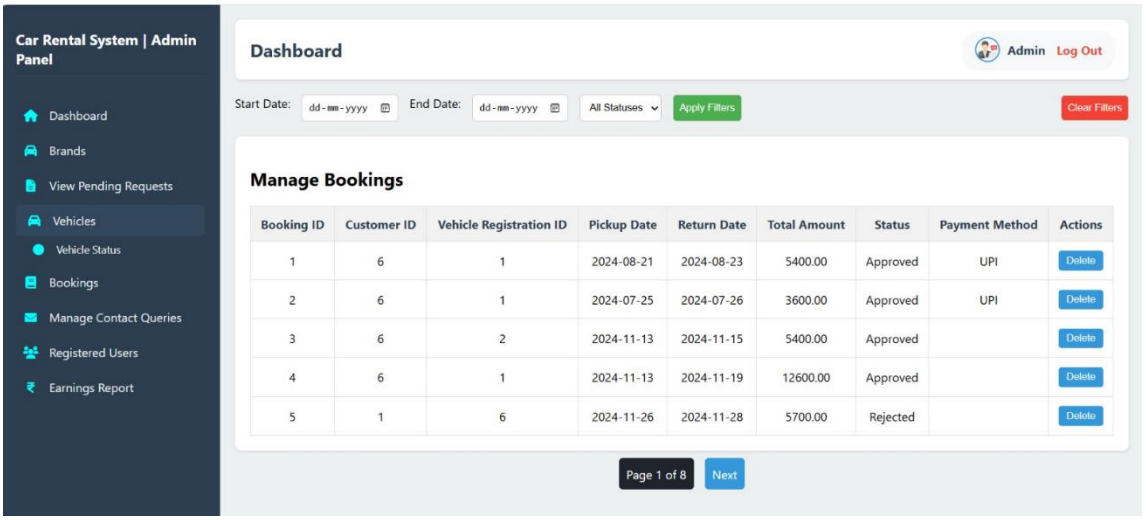

Figure 7.3: Booking Page

The admin booking page within the car rental system allows for the management of customer bookings. It displays a list of existing bookings, including customer information, vehicle details, booking dates, status, and posting dates. Administrators can view booking details and potentially take actions such as confirming or canceling bookings.

7.4 Brands Page:

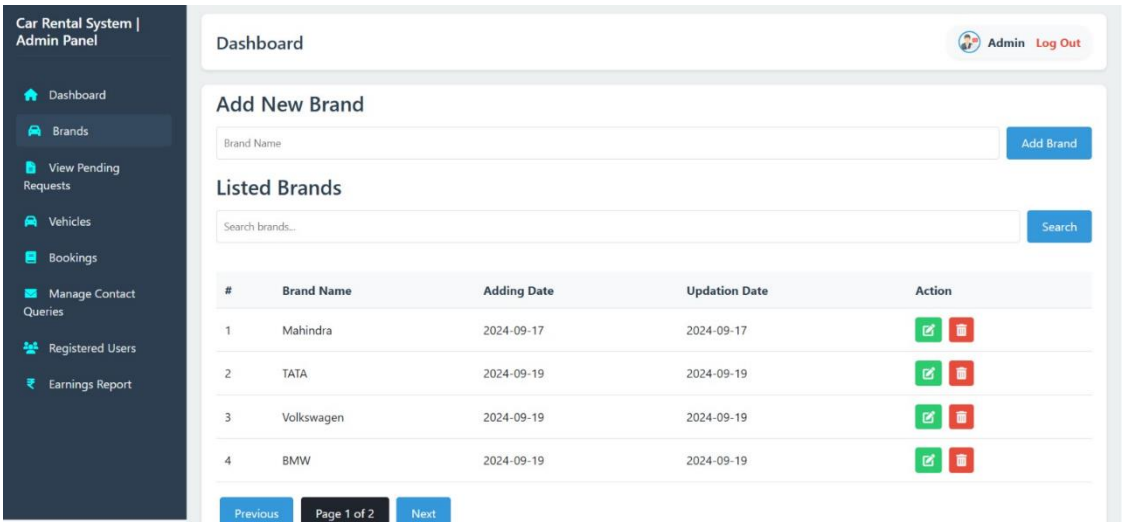

Figure 7.4: Brands Page

The admin brands page within the car rental system allows for the management of car brands listed on the platform. It displays a list of existing brands, including brand names, adding dates, and updation dates. Administrators can potentially add, edit, or remove brands from the list.

7.5 Manage Contact Page:

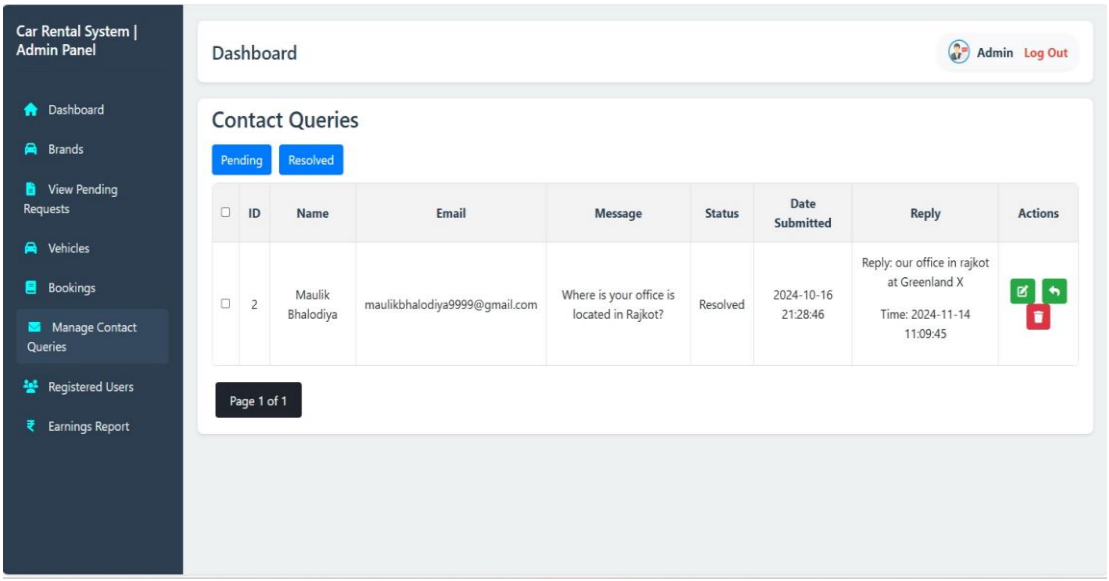

Figure 7.5: Manage Contact Page

The admin manage contact queries page within the car rental system allows for the management of customer inquiries. It displays a list of existing queries, including customer information, contact details, message content, and posting dates. Administrators can view query details and potentially respond to customer inquiries.

7.6 Vehicles Page:

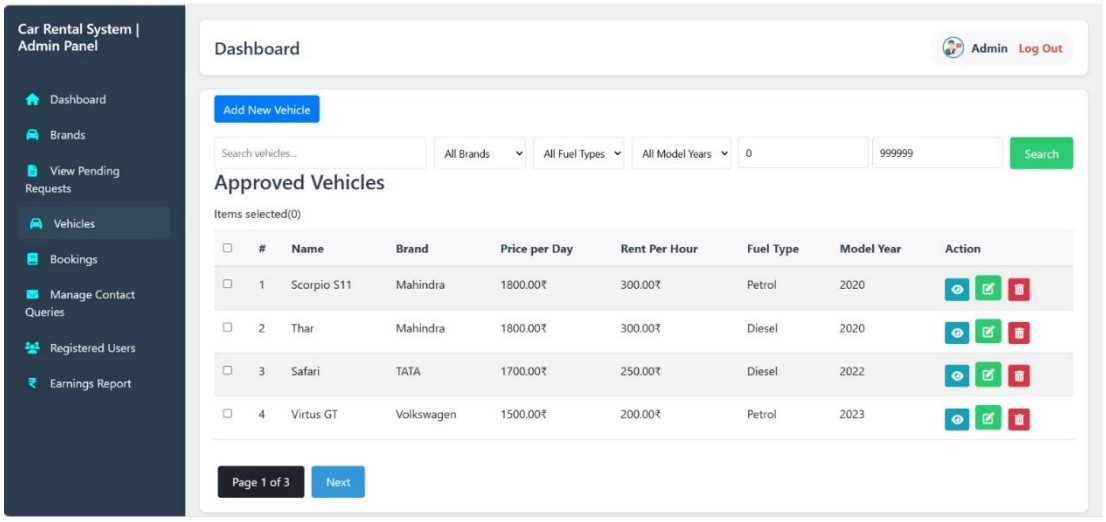

Figure 7.6: Vehicles Page

The admin vehicles page within the car rental system allows for the management of available vehicles. It displays a list of existing vehicles, including vehicle names, brands, pricing information, fuel types, model years, and actions. Administrators can potentially add, edit, or remove vehicles from the list.

7.7 Home Page:

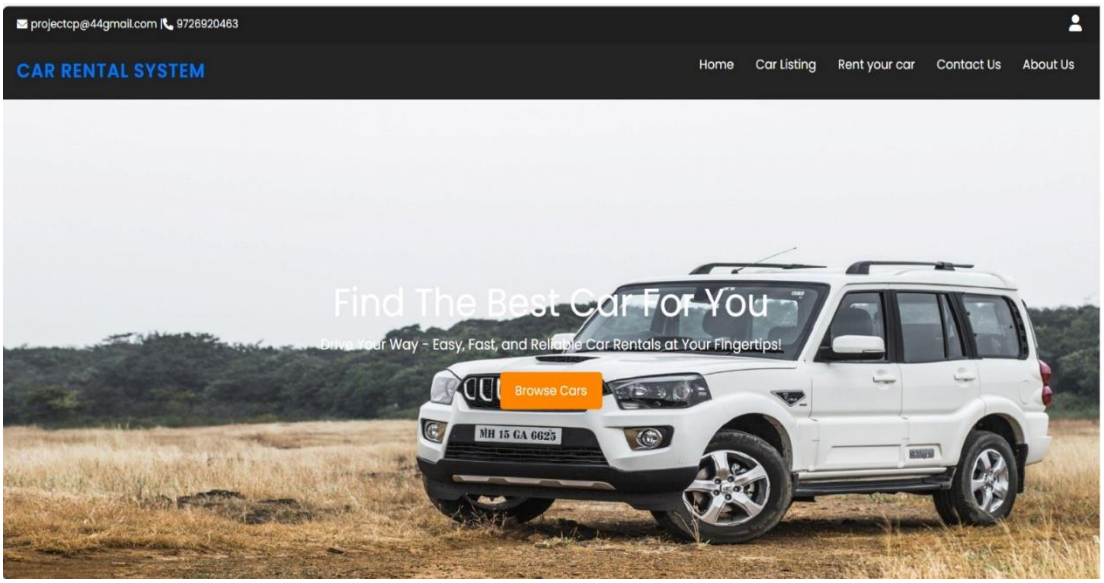

Figure 7.7: Home Page

This HTML page displays a welcoming message alongside a car image, inviting visitors to explore rental options. The navigation bar offers easy access to key sections such as car brands, car listings, FAQs, and contact information.

7.8 User Register Page:

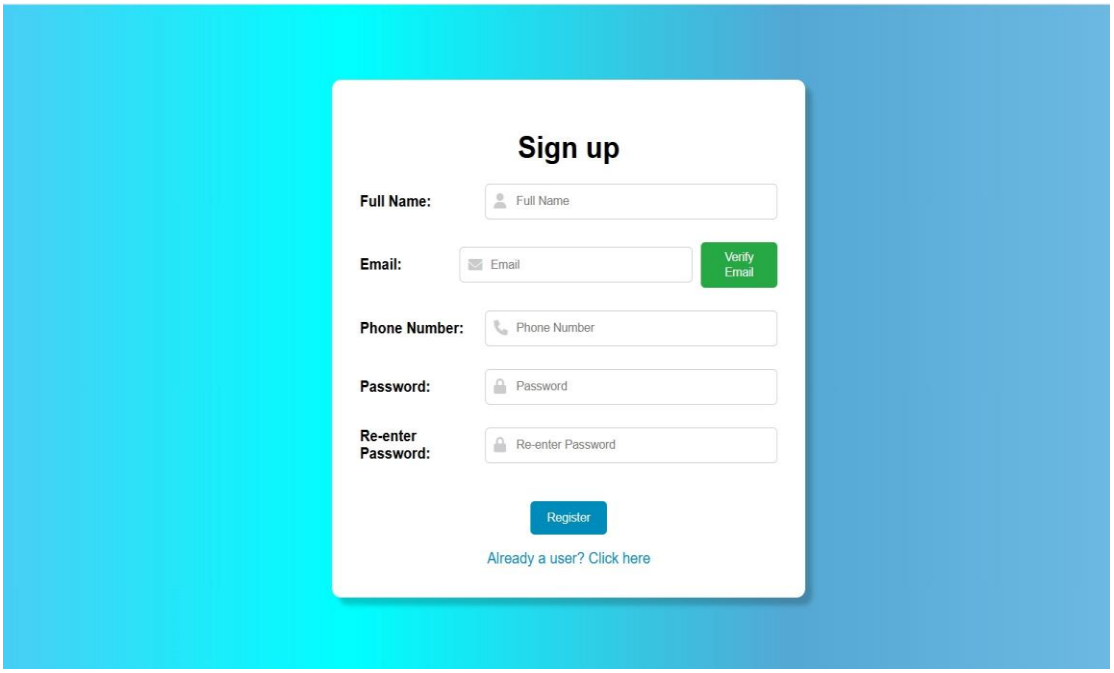

Figure 7.8: User Register Page

This page contains a user registration form that allows new users to create an account. The form includes input fields for username, password, name, contact number, and email. There is a Register button for submitting the form, and navigation links are provided for returning to login page if the user already has an account.

7.9 User Sign-In Page:

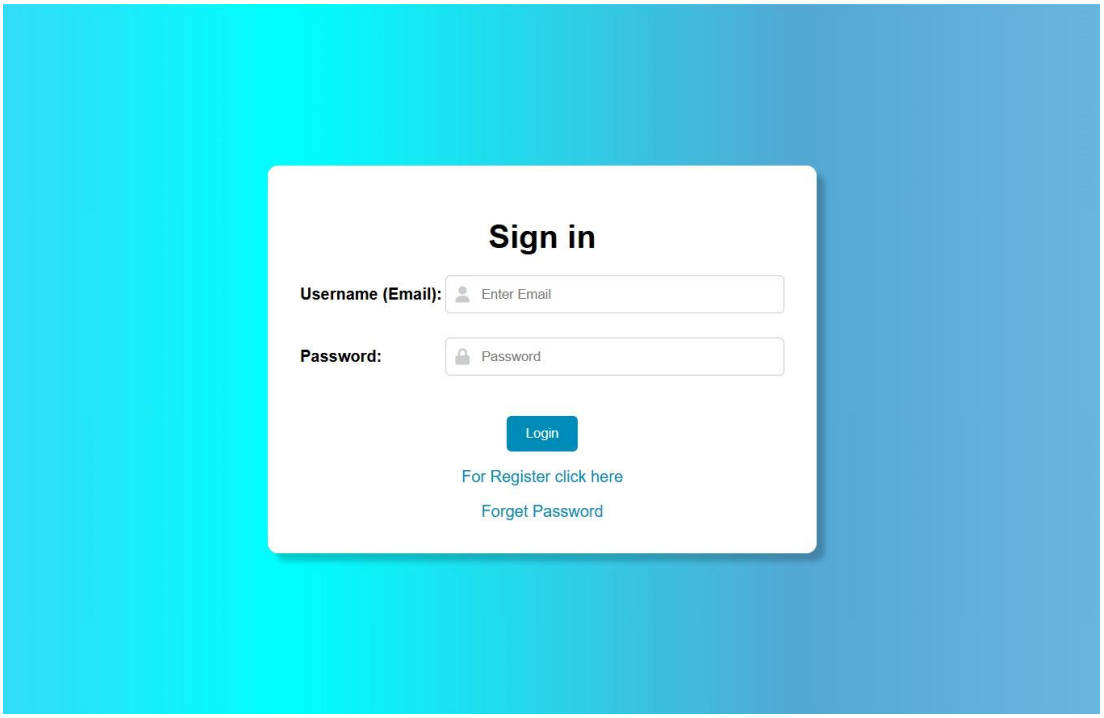

Figure 7.9: User Sign-In Page

This page presents a login interface with options, after complete login user will redirect to home page, for not registered users. Users can click to "register now" link to complete register process.

7.10 Book Car Page:

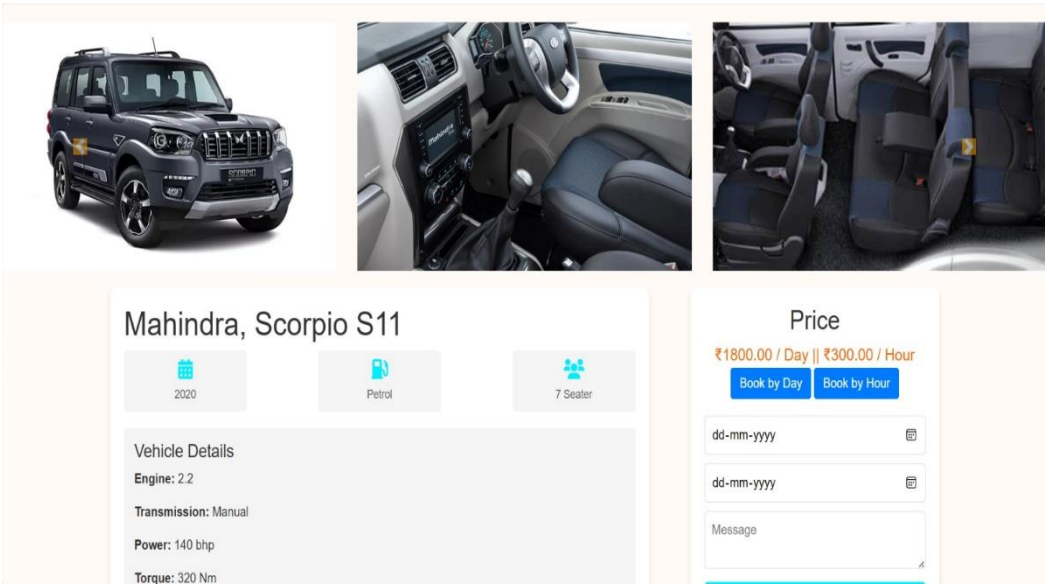

Figure 7.10: Book Car Page

The image describes a car rental booking page for a Mahindra Scorpio N. The page design is clean and structured, showcasing important details about the car and allowing the user to book the vehicle.

7.11 Car Lsting Page:

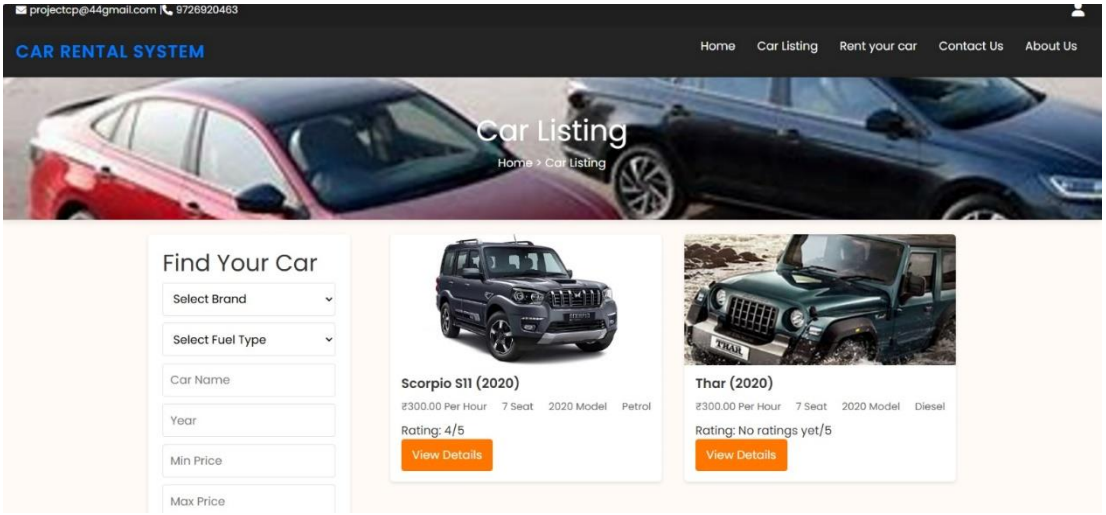

Figure 7.11: Car Lsting Page

The Car Listing page showcases available rental vehicles with images, key details like fuel type, seating capacity, and price per day and features such as engine, transmission, and mileage. Filters enable quick browsing, ensuring a seamless car selection experience for users.

7.12 Contact Us Page:

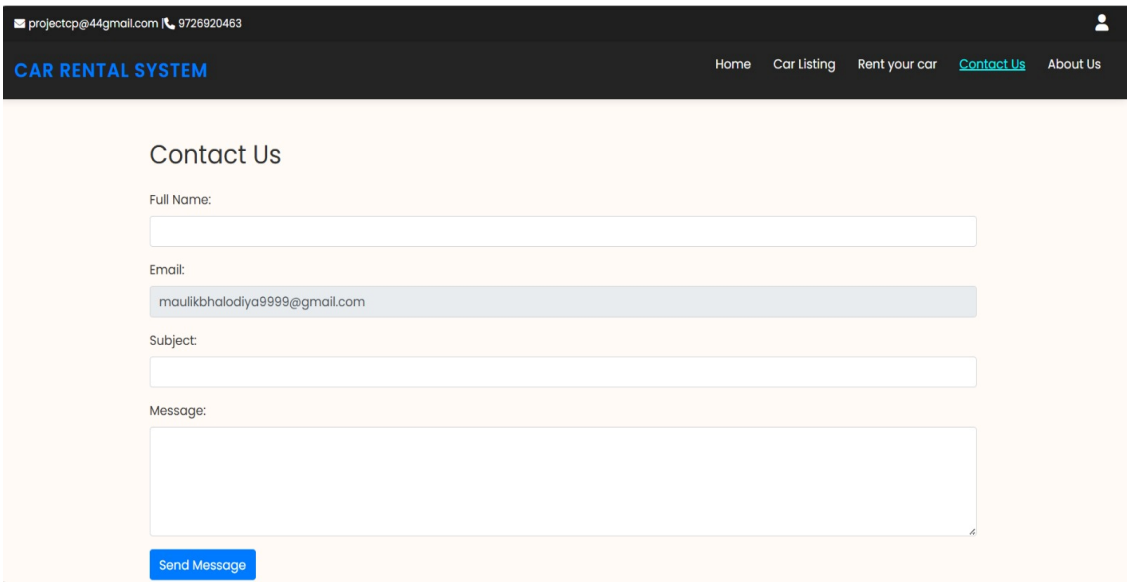

Figure 7.12: Contact Us Page

This page is "Contact Us" section with essential contact information, including address, email, and phone number. It also includes a form for submitting inquiries or feedback. The design is clean.

7.13 About Us Page:

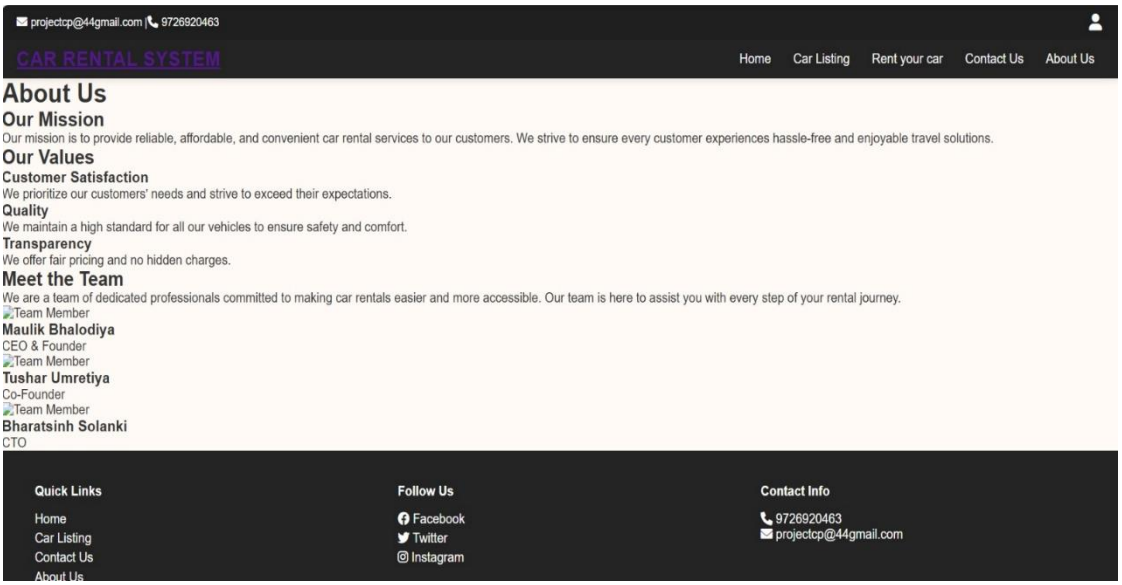

Figure 7.13: About Us Page

We simplify car rentals with our easy-to-use platform. Choose from a various models, book instantly, and enjoy transparent pricing and real-time availability. We ensure a comfort with good experience so you can focus on your journey. Drive with confidence and ease.

CHAPTER-8: ABOUT COLLEGE

U.V. Patel College of Engineering Ganpat University

Ganpat University-U. V. Patel College of Engineering (GUNI-UVPCE) is situated in Ganpat Vidyanagar campus. It was established in September 1997 with the aim of providing educational opportunities to students from It is one of the constituent colleges of Ganpat University various strata of society. It was armed with the vision of educating and training young talented students of Gujarat in the field of Engineering and Technology so that they could meet the demands of Industries in Gujarat and across the globe. The College is named after Shri Ugarchandbhai Varanasibhai Patel, a leading industrialist of Gujarat, for his generous support. It is a self- financed institute approved by All India Council for Technical Education (AICTE), New Delhi and the Commissionerate of Technical Education, Government of Gujarat. The College is spread over 25 acres of land and is a part of Ganpat Vidyanagar Campus. It has six ultra- modern buildings of architectural splendor, class rooms, tutorial rooms, seminar halls, offices, drawing hall, workshop, library, well equipped departmental laboratories, and several computer laboratories with internet connectivity through 1 Gbps Fiber link, satellite link education center with two-way audio and one-way video link. The superior infrastructure of the Institute is conducive for learning, research, and training. The Institute offers various undergraduate programs, postgraduate programs, and Ph.D. programs. Our dedicated efforts are directed towards leading our student community to the acme of technical excellence so that they can meet the requirements of the industry, the nation and the world at large. We aim to create a generation of students that possess technical expertise and are adept at utilizing the technical 'know-hows' in the service of mankind. We strive towards these Aims and Objectives:

- To offer guidance, motivation, and inspiration to the students for well-rounded development of their personality.
- To impart technical and need-based education by conducting elaborated training programs.
- To shape and mold the personality of the future generation.
- To construct fertile ground for adapting to dire challenges.