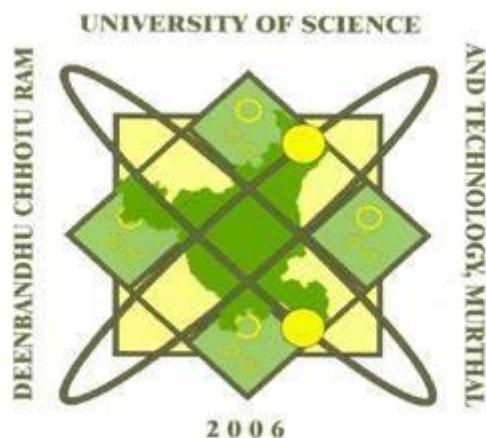


Project Report on

CarAixa

Submitted in partial Fulfilment of Requirement for the Award of the
Degree

Bachelor of Computer Application



Department of Computer Science

Hindu Institute of Management, Sonipat

Sonipat Haryana – 131001

2021 – 2024

Submitted To:

Ms. Kirti Vashishta

Hindu Institute of Management

Submitted By:

Vinay Singal

21012041056

DECLARATION

I hereby declare that all the work presented in the Project Report entitled '**CarAixa**' in the partial fulfilment of the requirement for the award of the degree of "**Bachelor of Computer Applications**" in Computer Science, '**Hindu Institute of Management', Deenbandhu Chotu Ram University of Science and Technology', Murthal' is an authentic record of our own work carried out under the guidance of **Ms. Kirti Vashishtha**.**

Date:

**Vinay Singal
(21012041056)**

CERTIFICATE

This is to certify that the Project Report on "**CarAixa**" entitled which is submitted by **Vinay Singal (21012041056)** has been successfully completed and submitted in partial fulfilment of the requirement for the award of degree **Bachelor of Computer Applications in Computer Science, Hindu Institute of Management, Deenbandhu Chotu Ram University of Science and Technology, Murthal, Sonipat** is a record of the candidate's work carried out by him under the supervision of undersigned. The matter embodied in this report is original and has not been submitted for the award of any other degree. His performance was good during the tenure.

Date:

Ms. Kirti Vashishtha

(Project Incharge)

ACKNOWLEDGEMENT

It brings me great pleasure to finally complete the Major Project in partial fulfilment of the prerequisite for the award of Bachelor of Computer Applications. I would like to thank **Ms. Kirti Vashishtha** (Project Incharge, BCA department) who was involved with me in this project to make it a success. I extend my heartfelt and profound gratitude to my respected faculty members for providing me the impulse to get this work off the ground.

Date:

Vinay Singal
(21012041056)

Table of Content

S No.	Title	Page Number
1	Introduction 1.1 About the Project 1.2 Objective of the Project	3-4 4 5
2	System Development Lifecycle (SDLC) 2.1 Introduction to SDLC 2.2 Purpose of SDLC 2.3 Various Phase of SDLC	6-13 7 7 8-14
3	System Analysis 3.1 Hardware and software Requirement 3.2 System Requirement 3.3 Reorganisation of Needs	15-18 16 16 17-18
4	System Design 4.1 Gantt Chart 4.2 ER Diagram 4.2.1 User Module 4.2.2 Admin Module 4.3 Flow Chart 4.4 DFD for Admin 4.5 DFD for User	19-29 20 21-25 21-24 25-26 27 28 29
5	Screen Design 5.1 User Module 5.2 Admin Module	30-47 31-42 43-47
6	Database Design 6.1 Database 6.2 Admin 6.3 Booking 6.4 Brands 6.5 Contact us info 6.6 Contact us query 6.7 Pages	48-54 49 50 50 51 51 51 52

	6.8 Payment 6.9 Subscribers 6.10 Testimonial 6.11 Users 6.12 Vehicles	52 52 53 53 54
7	System Testing 7.1 Testing 7.2 Types of System Testing 7.3 Test Case 7.4 System Test Cases 7.4.1 Register Module 7.4.2 Login Module 7.4.3 Admin Login Module	55-78 56 56-61 61-62 63-80 63-69 70-73 74-78
8	System Implementation	79-83
9	Result and Discussions	84-85
10	Conclusion	86-87
11	Future Scope	88-89
12	Bibliography and References	90-91

CHAPTER-1

INTRODUCTION

1.1 ABOUT THE PROJECT

CarAxia is a leading platform in the car rental industry, providing seamless access to vehicles tailored to individual needs. Much like the revolutionary impact of ecommerce on online commerce, CarAxia is dedicated to enhancing convenience and accessibility in car rentals.

In today's digital era, where efficiency is paramount, CarAxia stands out as the go-to solution for hassle-free transportation. Similar to how ecommerce simplifies online shopping, CarAxia streamlines the vehicle rental process through its user-friendly online platform.

While e-business encompasses various aspects of online operations, CarAxia focuses specifically on facilitating smooth vehicle rental transactions. Leveraging the internet's capabilities, CarAxia empowers users to effortlessly browse, select, and book rental vehicles, transcending geographical barriers and time constraints.

Embracing the core principles of electronic commerce, CarAxia utilizes digital channels to facilitate seamless service exchanges. By prioritizing online transactions, CarAxia ensures users can rent vehicles securely while facilitating the transfer of funds and data.

CarAxia is committed to elevating the rental experience beyond mere online transactions. Unlike traditional ecommerce platforms, CarAxia broadens the scope to include commercial transactions in the car rental domain, enhancing the overall rental experience for customers.

In essence, CarAxia embodies the spirit of electronic commerce by redefining how individuals access and engage with car rental services. Much like ecommerce transformed online retail, CarAxia seeks to revolutionize the car rental paradigm in the digital age.

1.2 Other objectives of this project are:

CarAxia aims to achieve the following objectives:

- Efficiently manage rental costs.
- Provide a unique and tailored customer experience.
- Build stronger business relationships.
- Enhance customer loyalty.
- Improve service efficiency.
- Target the appropriate audience effectively.
- Ensure the mobile responsiveness of the CarAxia platform.
- Increase the volume of rental bookings.

CHAPTER-2
SYSTEM DEVELOPMENT LIFE
CYCLE (SDLC)

2.1 SYSTEM DEVELOPMENT LIFECYCLE (SDLC)

The systems development life cycle (SDLC) is a structured process used in project management that outlines the stages involved in developing a system, starting from the feasibility study to maintaining the completed application. SDLC applies to both technical and non-technical systems and can involve IT technologies like hardware and software. Project and program managers usually participate in SDLC.

Every system, whether it's hardware or software, goes through a development process that's like a series of steps. SDLC provides a clear structure and framework to define these phases and steps involved in building a system. It's a method to keep everything organized and clear during development. SDLC is also known as Software Development Life Cycle, which focuses solely on the development of software.

In SDLC, documenting everything is essential, no matter which model is chosen for an application. This documentation happens alongside the development process. Some methods work better for specific projects, but ultimately, the success of a project often depends on how closely the chosen plan is followed.

2.2 Purpose of SDLC

- The main purpose of SDLC is to build a high-quality product on the basis of customer's requirements.
- It helps in successful implementation of project that satisfy user's need.
- To ensure the delivery of project on time and within budget

2.3 Various Phases of SDLC:

There are five phases of SDLC which are given below:

1. Planning
2. Requirement Analysis
3. Design
4. Implementation
5. Maintenance

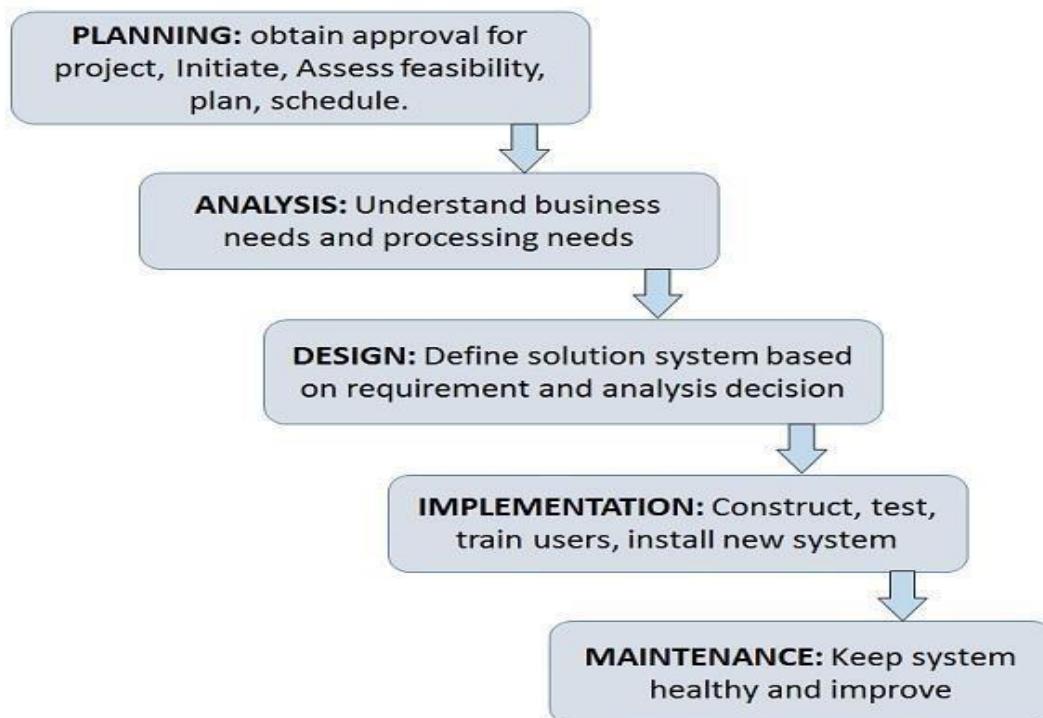


Fig. 2.3.1

2.3.1 Planning phase

At the onset of planning for the CarAxia car rental platform, it's imperative to initiate a strategic blueprinting process that encompasses the entire development cycle. Planning for a car rental website entails a comprehensive understanding of objectives, features, and technical requisites. Here's the breakdown of steps to undertake:

- Define Objectives and Goals: Clearly outline the purpose of the website, focusing on facilitating car rentals.
- Feature Identification: Compile a list of essential features such as vehicle selection, booking process, and user accounts.
- Technical Requirements: Determine the appropriate technology stack including web development tools and database systems.
- Create a Project Plan: Develop a detailed timeline with milestones for each development phase to ensure an organized approach.
- Design and UX: Draft initial sketches and wireframes to visualize the platform's layout and user experience, prioritizing simplicity.

The primary aim during this phase is to establish a robust foundation in line with CarAxia's commitment to delivering a seamless car rental experience.

2.3.2 Requirement Analysis

This phase involves gathering information about the functional and non-functional requirements of Project.

Functional Requirements:

- **User Registration/Login:**

Enable users to securely register and log in to their accounts.

- **Purchase Items:**

Implement a feature that allows users to buy items from the platform.

- **User Profiles:**

Provide users with personalized profiles where they can manage their purchases, view their cart, and submit feedback.

Non-Functional Requirements:

- **Security:**

Implement robust security measures including secure user authentication and data encryption to protect user information.

- **Compatibility:**

Ensure seamless compatibility across various web browsers such as Chrome, Firefox, and Microsoft Edge to enhance user accessibility and experience.

The Requirements phase is critical for defining the necessary functionalities and features unique to CarAxia.

2.3.3 System Design

During the System Design phase for CarAxia, the car rental website project, requirements are transformed into a technical blueprint necessary for subsequent development. This phase encompasses:

Architecture Design:

- **Frontend Design:**

Choose the technologies for the user interface (HTML, CSS, JavaScript).

Design the layout, navigation, and user interactions (possibly using frameworks like Bootstrap, fontawesome).

- **Backend Structure:**

Select a backend technology (PHP) for handling server-side operations.

Plan the database structure (MySQL) for storing user profiles, products, user orders etc.

Database Design:

- **Database Schema:**

Design the database tables/entities needed (User, product, order manager) based on the functionalities required.

Define relationships between tables (One-to-One, One-to-Many, Many-to-Many).

- **Data Storage:**

Plan how products will be stored and retrieved efficiently.

Security Measures:

- **User Authentication:**

Determine how users will register and log in securely.

2.3.4 Implementing and coding

During the Implementation and Coding phase of the CarAxia car rental website project, the primary focus is on translating the architectural and software design into actual source code. This pivotal phase encompasses the development of all website components, including but not limited to the category section, about us section, and booking system.

Frontend Development:

- HTML/CSS Structure:**

Create the basic structure of web pages using HTML.

Style the website using CSS for layout, fonts, colours, etc.

- Client-Side Interactivity:**

Implement user interactions using JavaScript.

Develop features like buy items, user authentication forms, and interactive elements.

Backend Development:

- Server-Side Logic:**

Set up the backend server (XAMPP).

Write code to handle user authentication, data retrieval, and storage.

- Database Integration:**

Implement code to interact with the database. Use MYSQL queries.

Implements query to handle CRUD (Create, Read, Update, Delete) operations for user profiles, products etc.

User Management:

- User Registration/Login:**

Implement user authentication mechanisms (signup, login, logout, forget password) with validation checks for user input.

Testing:

- Unit Testing:**

Write and conduct tests for individual components (functions, modules) to ensure they work as expected.

- Integration Testing:**

Test the interaction between frontend and backend components to ensure seamless functionality.

Deployment:

- Server Setup:**

Deploy the website on a hosting service or local server for testing purposes.

Documentation:

- Code Documentation:**

Write comments and documentation within the code to explain functionality, methods, and logic.

2.3.5 Maintenance phase

During the maintenance stage of the SDLC, the system is assessed to ensure it does not become outdated. This is also where changes are made to initial software. It involves continuous evaluation of the software in terms of its performance. After software is up and running, it often requires continuous maintenance. It includes:

Adaptive Maintenance:

- Updates and Enhancements:**

Incorporate updates to keep the website aligned with changing user needs or technological advancements.

- Feature Additions:**

Introduce new features or functionalities based on user feedback or emerging trends in the e-commerce industry.

Perfective Maintenance:

- Performance Optimization:**

Fine-tune the website's performance by optimizing code, databases, or server configurations.

- Usability Improvements:**

Enhance the user experience by refining navigation, design elements, or responsiveness.

CHAPTER 3

SYSTEM ANALYSIS

3.1 Hardware and Software requirement:

The environment for the development, operation and maintenance can be categorized into hardware requirement specification and the software requirement specification each of these requirement has been dealt with as given below:

3.2 System Requirements:

The computer system is made up of units that are put together to achieve a common goal. The requirements for the implementation of the system are as follow:

1. The Hardware
2. The Software

3.2.1 Hardware required:

Processor	Intel(R) Core (TM) i3-1005G1 CPU @ 1.20GHz 1.19GHz
Ram	4.00 GB or above
Hard Disk	50 GB or above
Input Devices	Keyboard, Mouse
Output Devices	Monitor

3.2.2 Software Required:

Operating System	Windows XP, 7, 8, 10, Linus and
Mac Browsers	Chrome, Microsoft Edge, Firefox
Front-end	HTML, CSS, JavaScript
Back-end	MySQL PHP
Server	XAMPP

3.3 RECOGNITION OF NEEDS:

In the initial phase of recognizing needs for the CarAxia car rental platform, we delve into understanding the requirements and exploring existing solutions. This phase is akin to drafting a plan for system design, involving preliminary research to gain insights into the situation.

We segment the work into different parts, similar to chapters in a book, to facilitate organization and comprehension of the system's architecture. The identified modules and sub-modules are tailored to meet the specific requirements of CarAxia.

Admin Module: Under the admin module, several sub-modules facilitate efficient management of project members, companies, and services, as well as editing the project profile. These sub-modules include:

- **Dashboard:** Provides an overview of key metrics such as registered users, listed vehicles, total bookings, subscribers, queries, and testimonials.
- **Brands:** Enables the addition and management of car brands.
- **Vehicles:** Allows for posting and managing vehicle listings.
- **Bookings:** Facilitates the management of new, confirmed, and canceled bookings.
- **Manage Testimonials:** Displays user feedback on the car rental service.
- **Contact Us Queries:** Enables admin to respond to user queries.

- **Registered Users:** Provides a list of registered users.
- **Manage Pages:** Allows management of website pages such as terms and conditions, privacy policy, etc.
- **Update Contact Info:** Enables admin to update contact information.
- **Manage Subscribers:** Manages subscriber information.

User Module:

The user module facilitates user registration and access to car rental services. It includes the following sub-modules:

- **User Registration:** Allows users to sign up and provide necessary details for registration.
- **Vehicle Catalogue:** Displays a catalogue of available vehicles for users to select based on preferences and budget.
- **Bookings:** Allows users to view and manage their bookings.

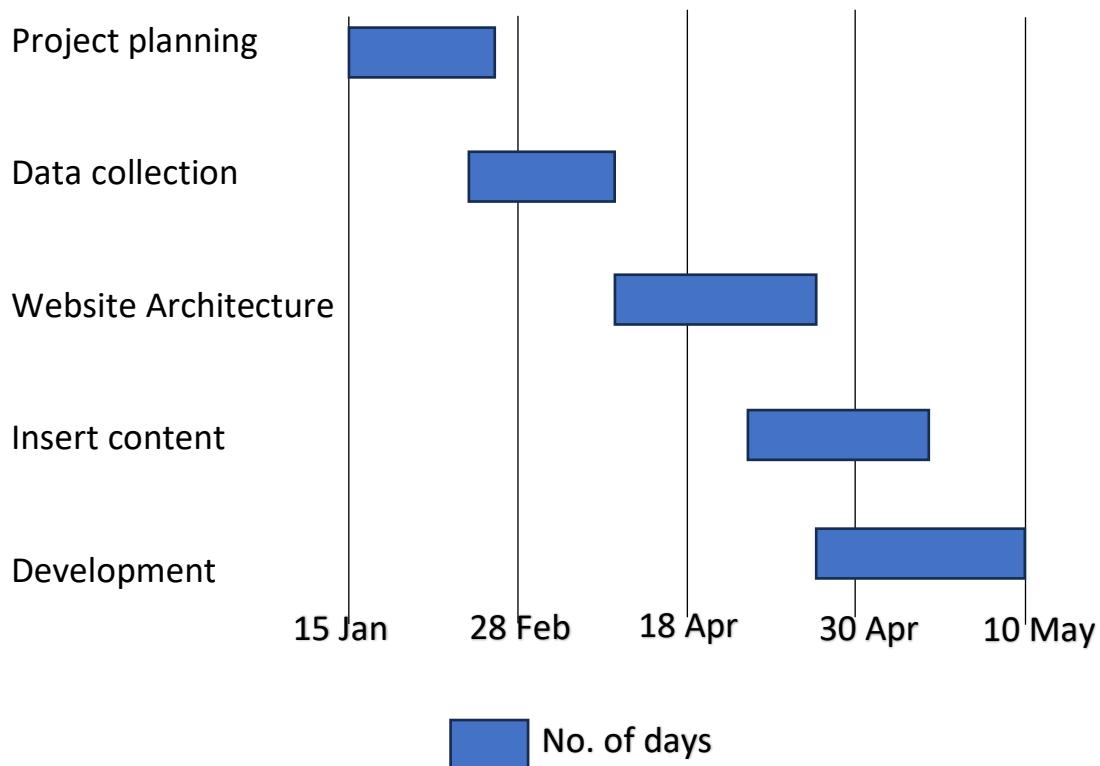
This systematic breakdown ensures that CarAxia addresses the needs of both administrators and users effectively, providing a seamless car rental experience for all stakeholders.

Chapter-4

System Design

4.1 Gannt Chart

Project Life Cycle

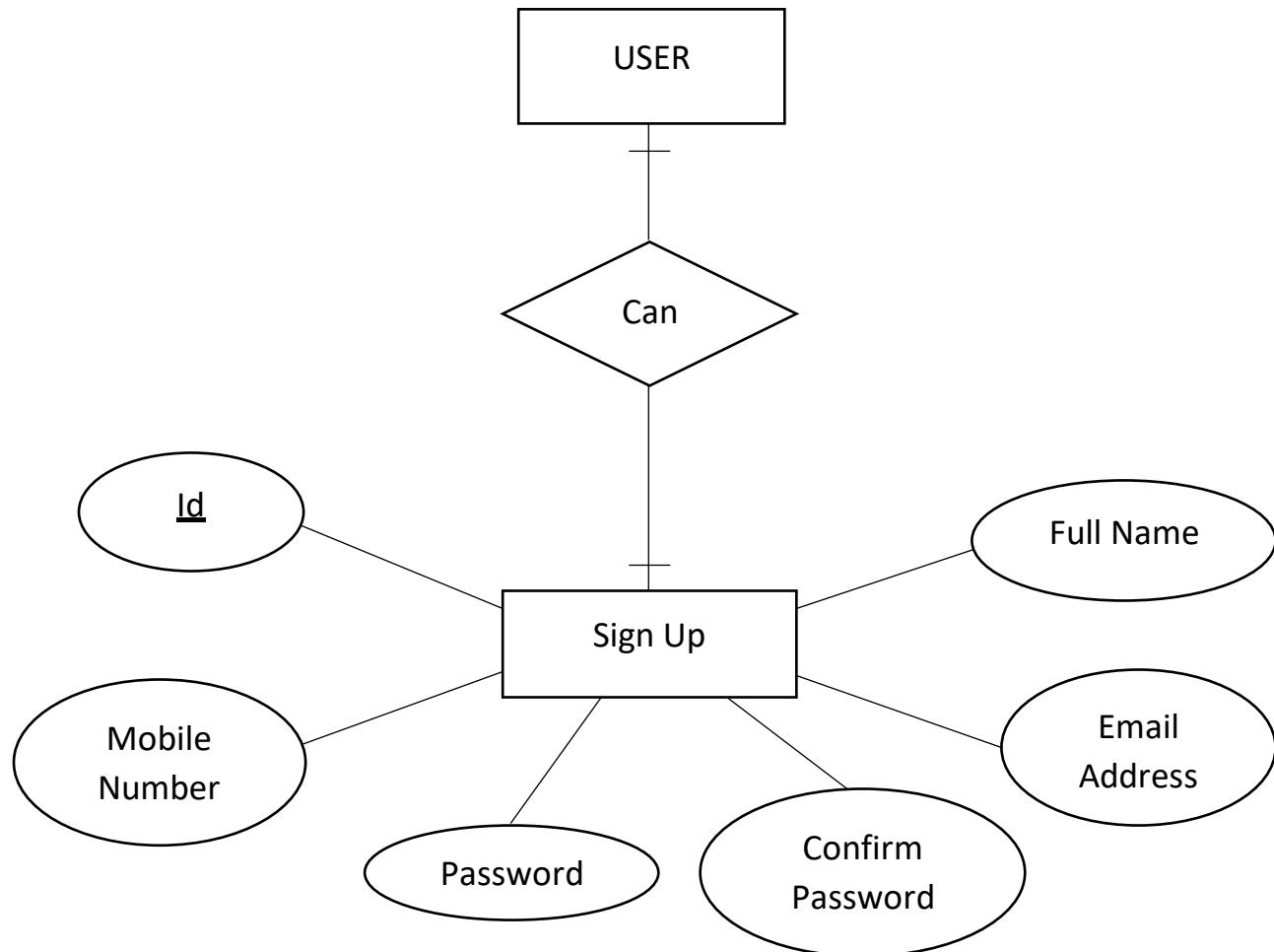


Analysis is a detailed study of various operations performed by the system and the relationship within and outside of the system that is it includes finding out in more detailed what the system problem are and what are the different new changes the user wants

4.2 ER Diagram

4.2.1 User Module

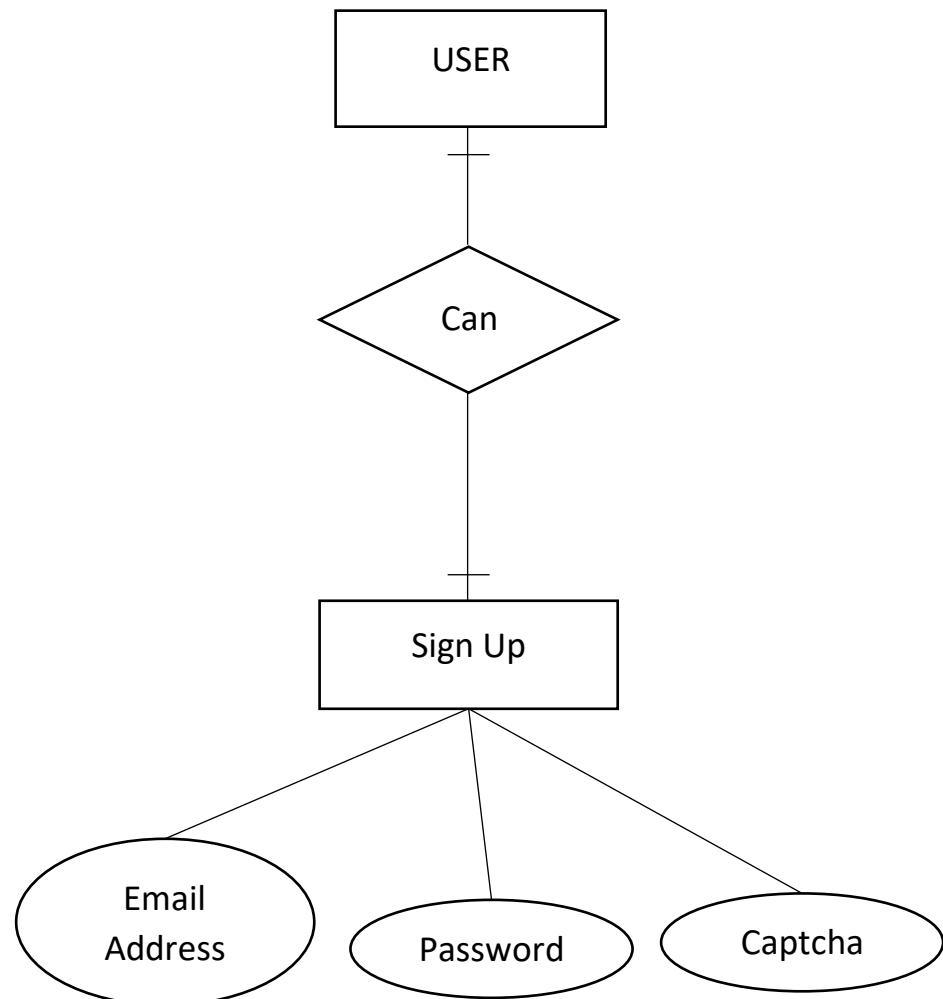
4.2.1.1 Sign Up



Explanation

- The SignUp entity contains all the attributes related to the signup process, including username, password, conform_password, address, city, and zip. It also includes the id attribute, which is the primary key for the SignUp entity.
- The one-to-one relationship between User and SignUp is established. This relationship indicates that each user has exactly one signup record, and each signup record is associated with a unique user.

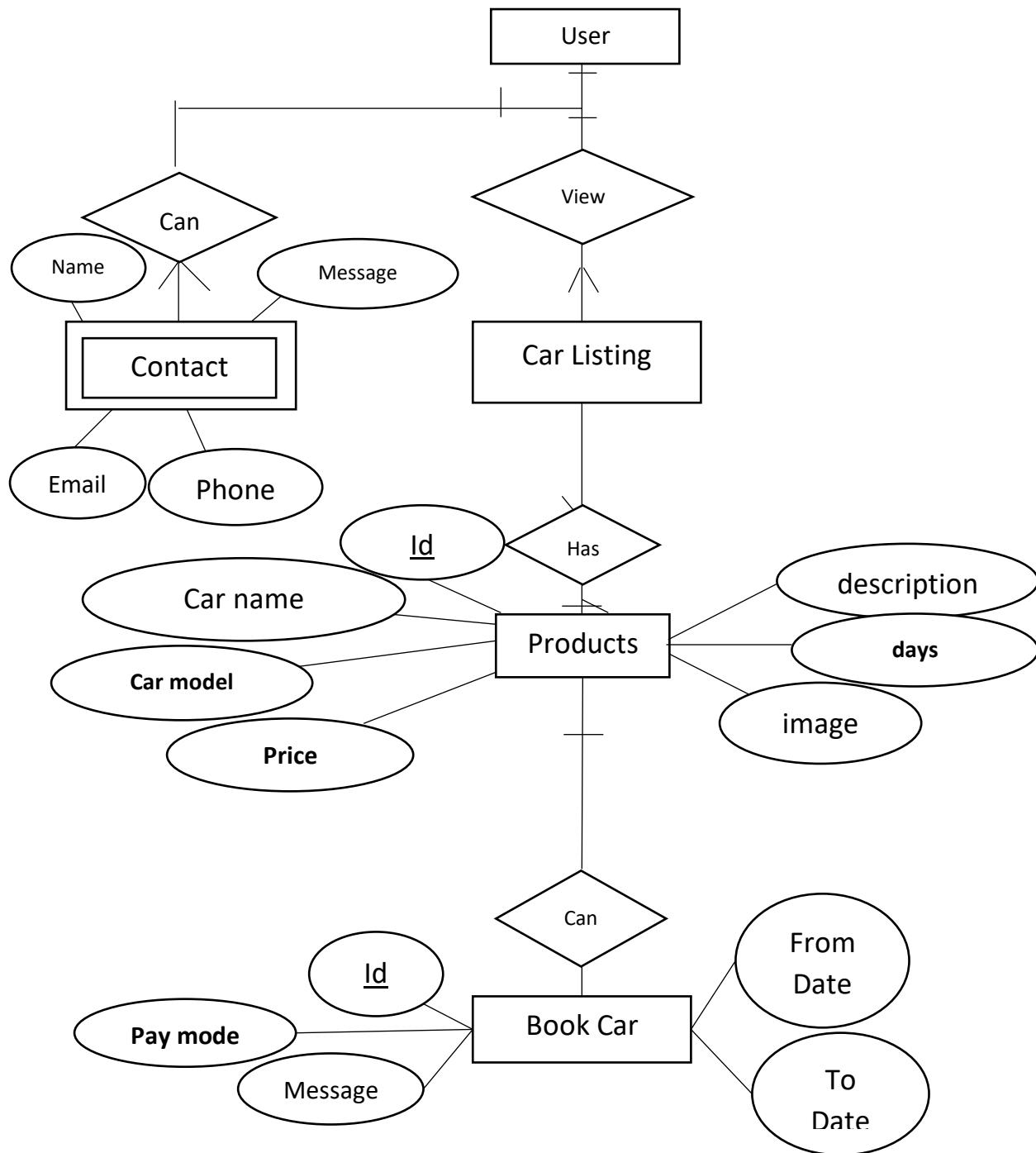
4.2.1.2. Login



Explanation

- The Login entity contains all the attributes related to the Login process, including username, password.
- The one-to-one relationship between User and Login is established. This relationship indicates that each user has exactly one Login record, at one time and each Login record is associated with a unique user.

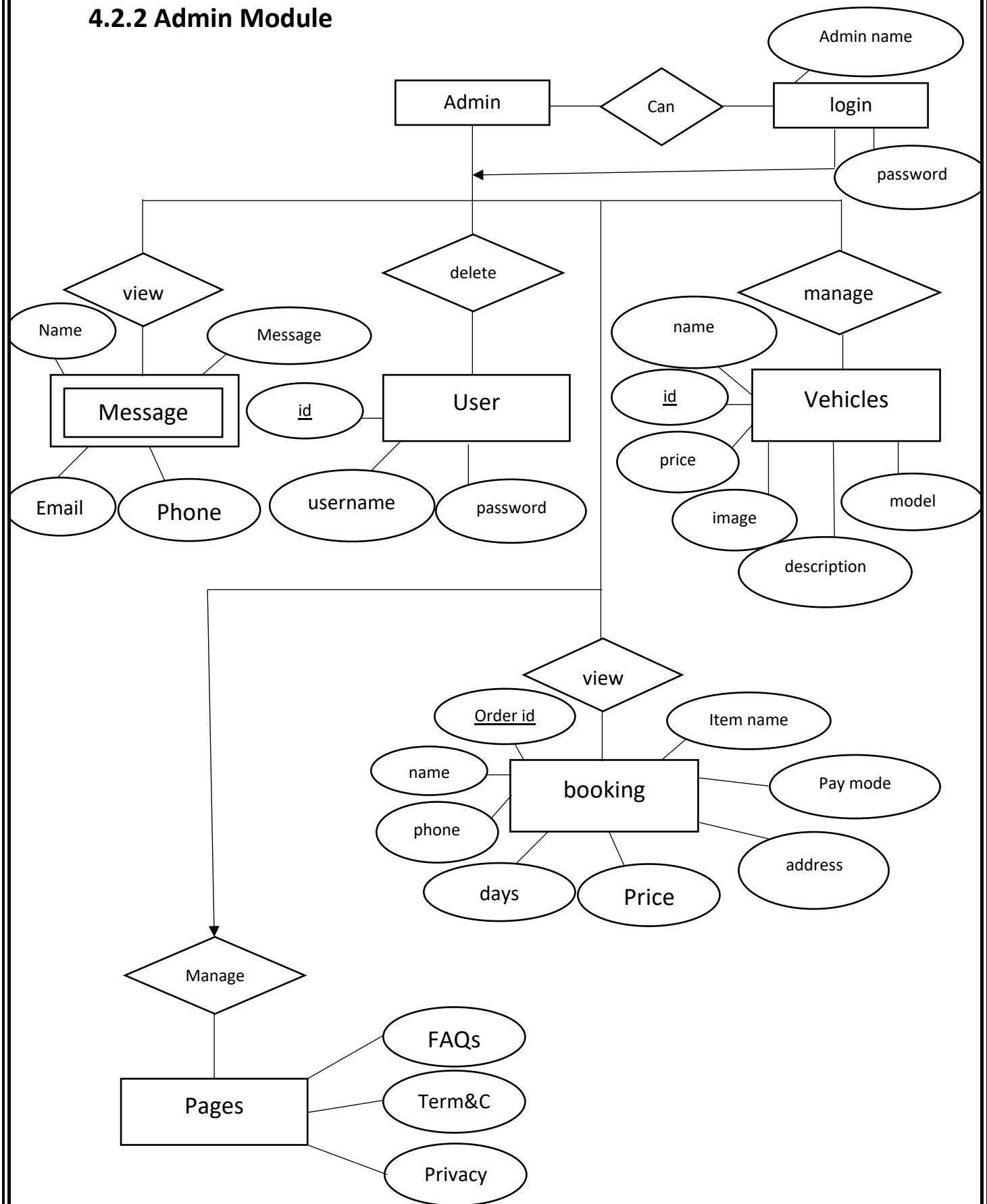
4.2.1.3 Main Module



Explanation

- The Category entity represents different categories that can contain various products
- The Car entity contains information about individual products. It includes a unique identifier (id), product details such as car_name, price, Car model, description, and features. It also has a Many-to-One relationship with the Car Listing entity, indicating that many products can belong to one Car Listing.
- The Book entity represents the information about the products a user has bought. It includes a unique identifier (id), car_name, price, along with user details such as name, from date, to date , phone_no, and pay_mode.
- The Message entity represents messages that users can send to the admin. It includes user details such as name, email, phone_no, and the message content. It also has a Many-to-One relationship with the User entity, indicating that many messages can belong to one user

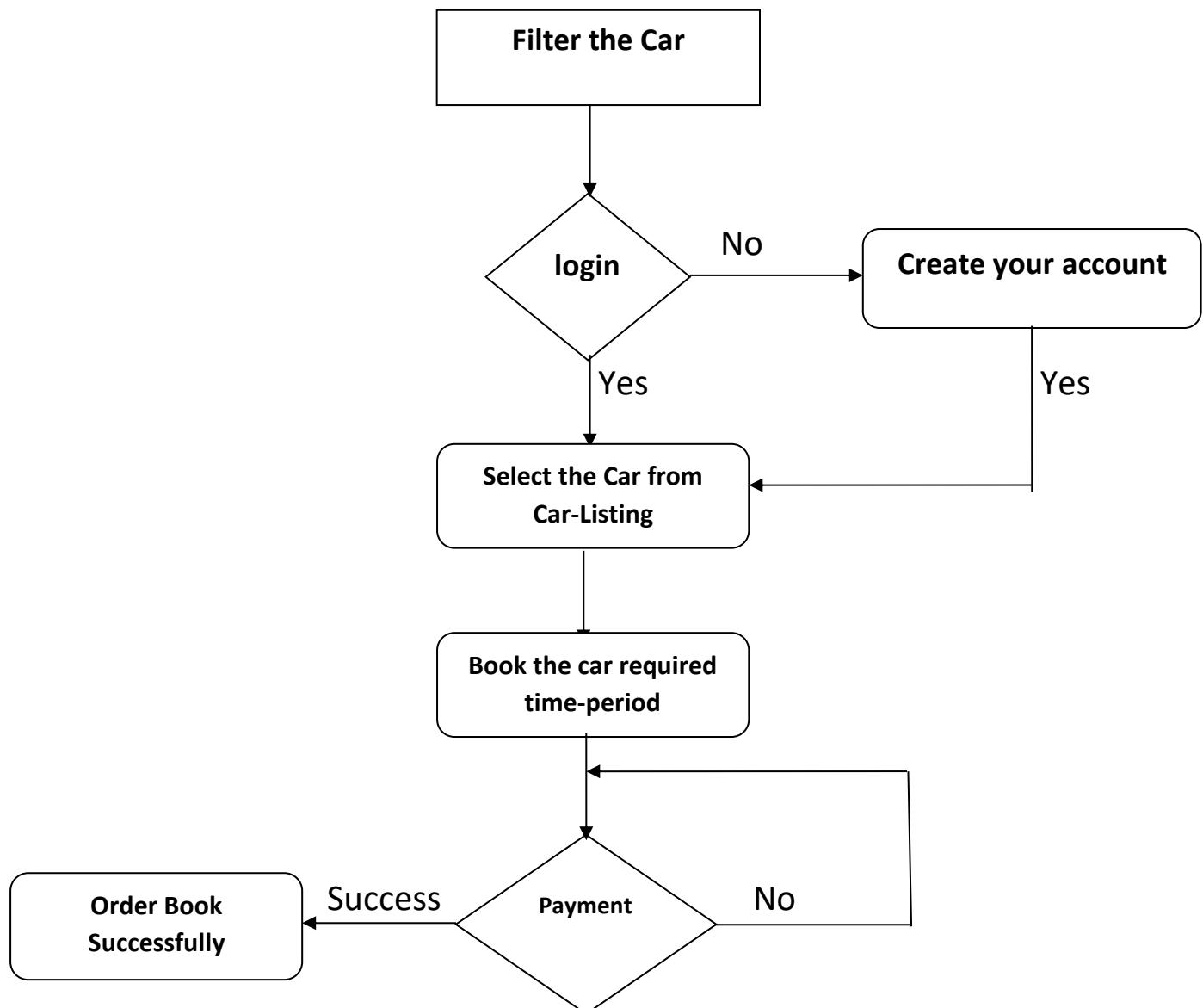
4.2.2 Admin Module



Explanation

- The Admin entity represents the administrative user. It includes a username and password for authentication purposes.
- The User entity contains information about users. It includes a unique identifier (id), username, and password. This entity allows the admin to view and potentially delete user accounts.
- The Product entity represents the products available in the system. It includes a unique identifier (id), along with details such as car_name, price, car_model, description, and image. The admin can manage products, which may include actions like adding, updating, or deleting products.
- The Message entity contains messages that users have sent to the admin. It includes a unique identifier (id), along with user details such as name, email, phone, and the message content. The admin can view messages.
- The Order entity represents the orders placed by users. It includes a unique identifier (order_id), along with details such as name, phone, address, pay_mode, item_name, price, and quantity. The admin can view order details, including the items purchased and user information.
- The Page entity manages content for various pages on the website, including FAQs, Terms and Conditions, Privacy Policy, and About Us. Each page has a unique identifier (page_name) and contains relevant information such as title, content, and metadata.

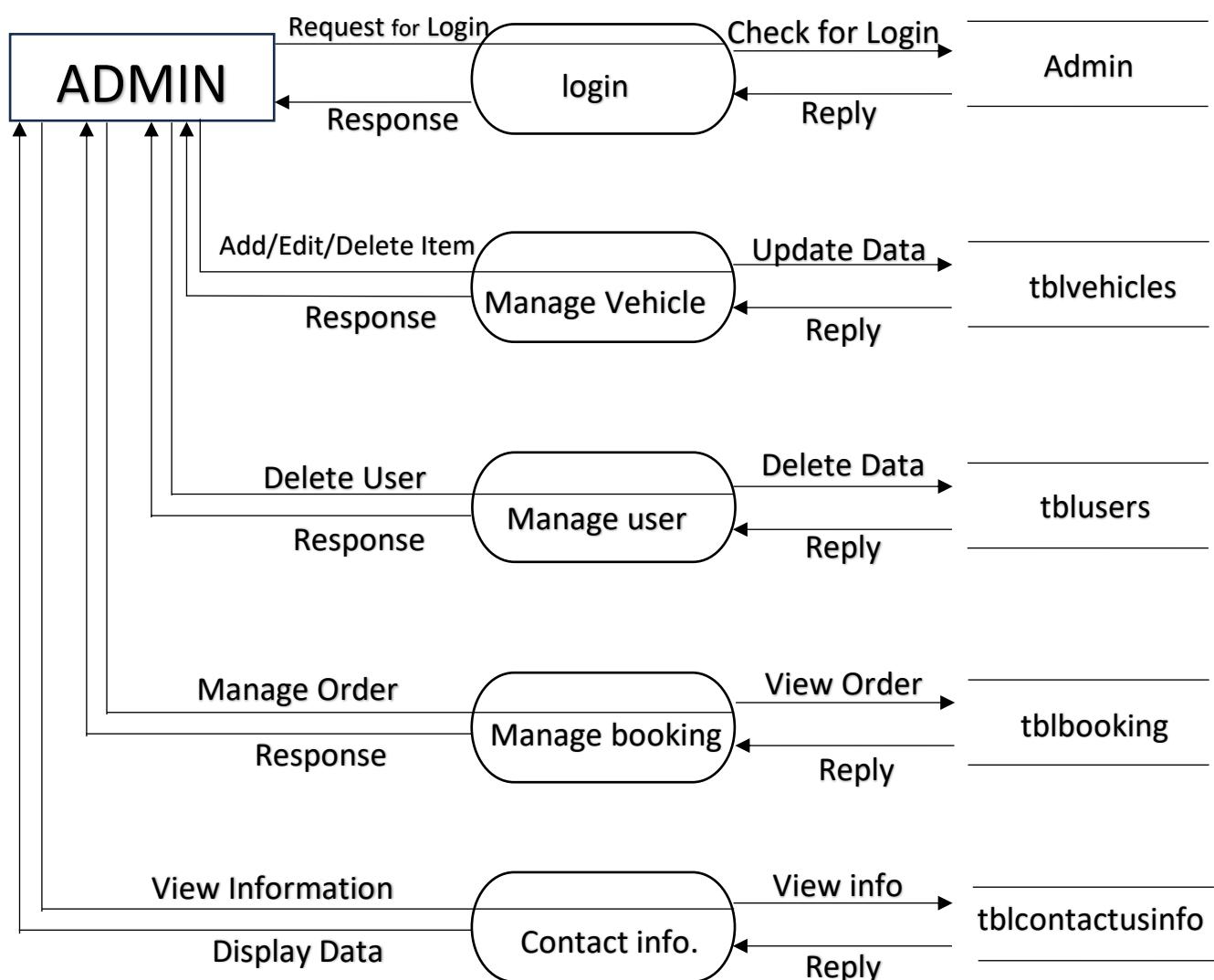
4.3 Flow Chart



- First customer chooses their desired product
- They have to click on buy now, then consumer have to login with their credential and if customer doesn't have an account then they have to first create their Account and book the car.
- The user have to enter his details and address and complete the payment process.

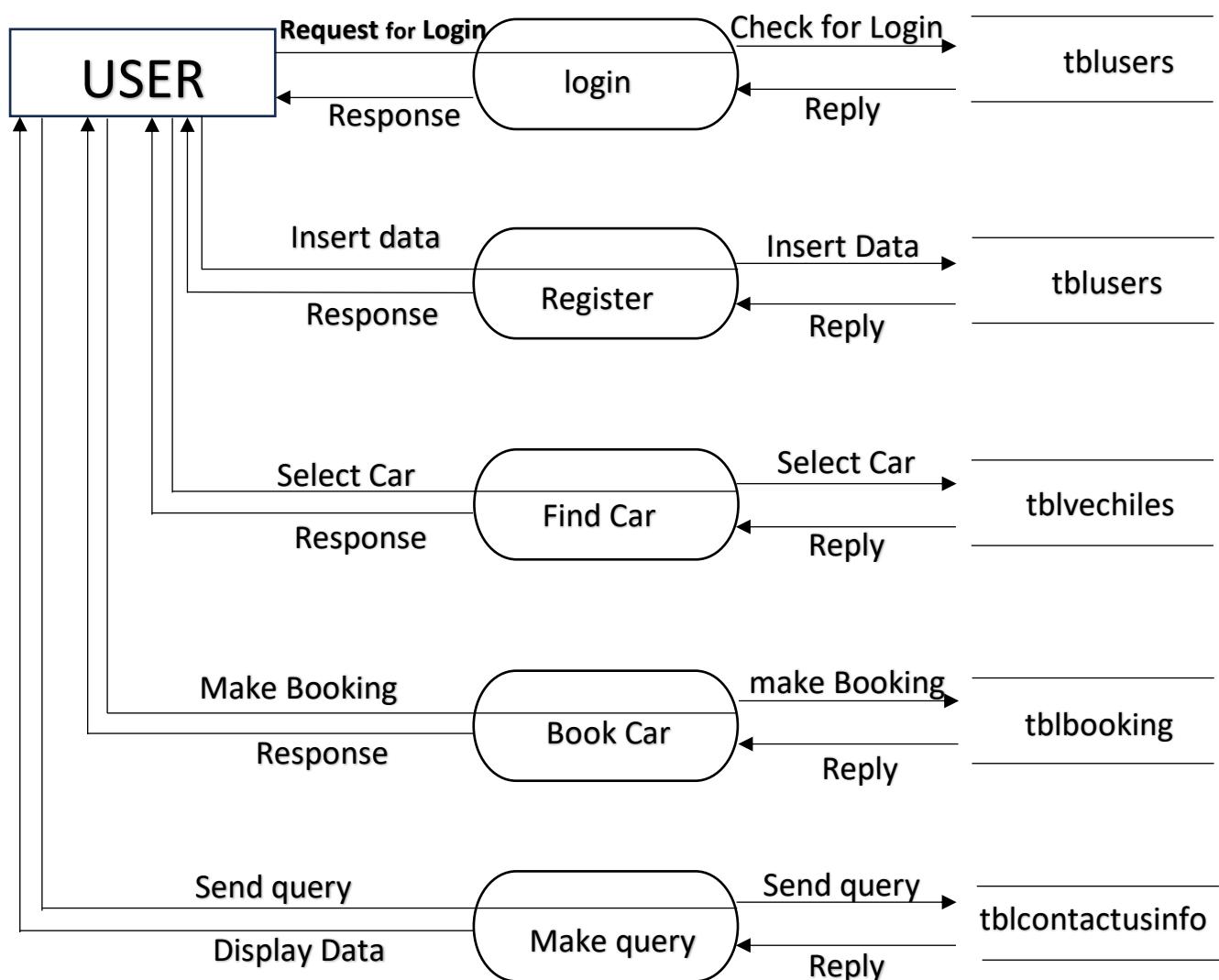
4.4 DFD OF ADMIN'S MODULE

The Admin side DFD describe the functionality of Admin, Admin is a owner of the website. Admin can first add category of item and then add items by category wise. and admin can manage order and payment detail



4.5 DFD OF USER'S MODULE

The user is all people who operate or visit our website. User is a customer of a website. User can first select product for buy, user must have to register in our system for purchase any item from our website. after register he can login to site and buy item by making online payment through any bank debit card or credit card.



Chapter 5

Screen design

5.1 User Screen Module

5.1.1 Register Form

Sign Up

Full Name

Mobile Number

Email Address

Password

Confirm Password

I Agree with Terms and Conditions

Sign Up

Already got an account? [Login Here](#)

Fig 5.1.1 Register Form

If new user buy some product from the website then it must to register first to buy some products.

5.1.2 Login Form

The image shows a mobile-style login interface. At the top left is the word "Login". At the top right is a small circular icon with an "X" inside. Below the title are two light gray input fields: one for "Email address*" and one for "Password*". Underneath these fields is a CAPTCHA box containing the code "y7SJSz". To the right of the CAPTCHA is a circular refresh icon. Below the CAPTCHA is a large red rectangular button with the word "Login" in white. At the bottom of the screen, there are two smaller text links: "Don't have an account? [Signup Here](#)" and "Forgot Password?".

Fig 5.1.2 Login Form

If old user buy something it simply login with their details, and can easily buy their products

5.1.3 Home page

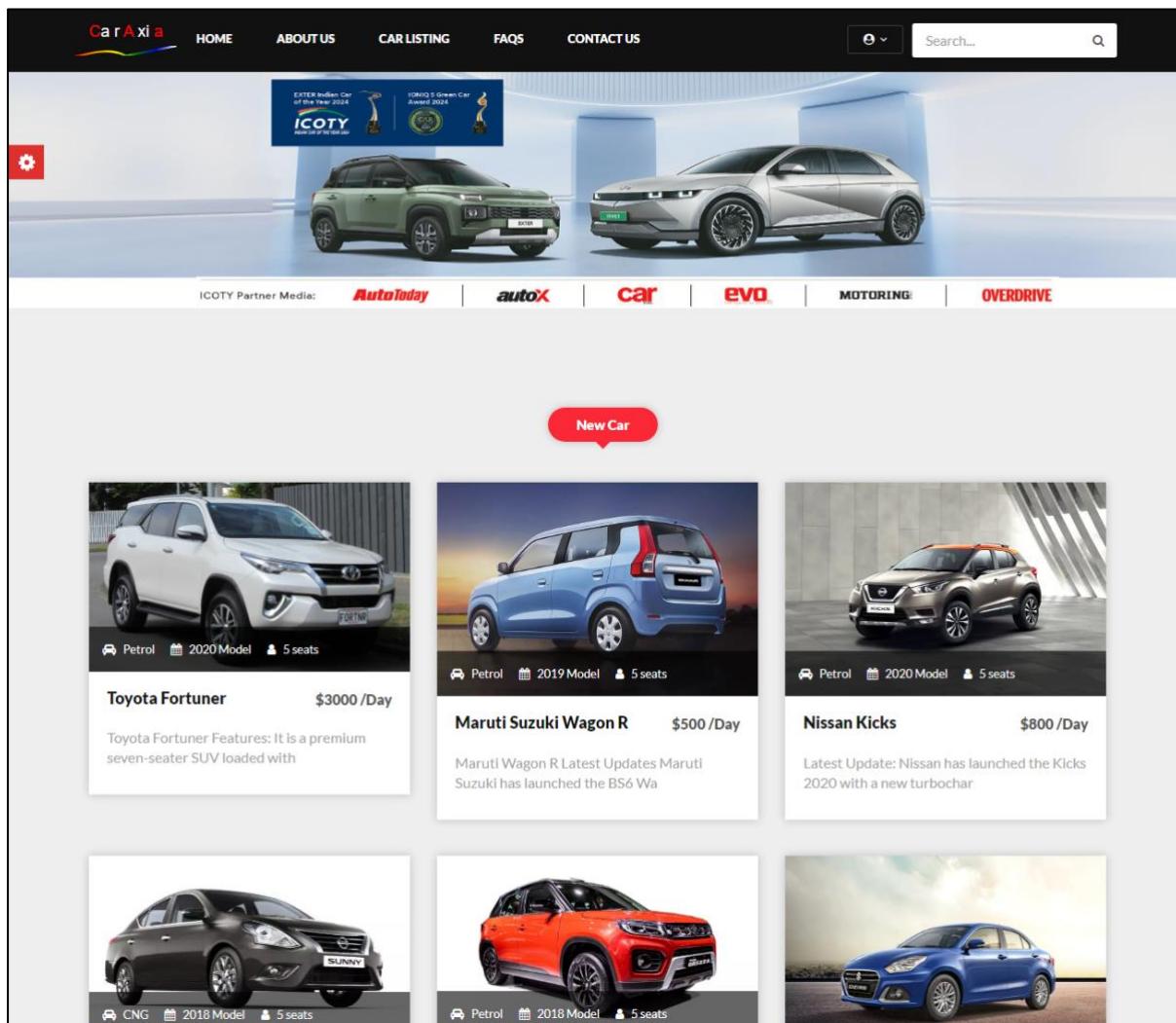


Fig 5.1.3 Home Page

Here we have shown home page of our e commerce website. The user can easily find the product

5.1.4 Home page footer

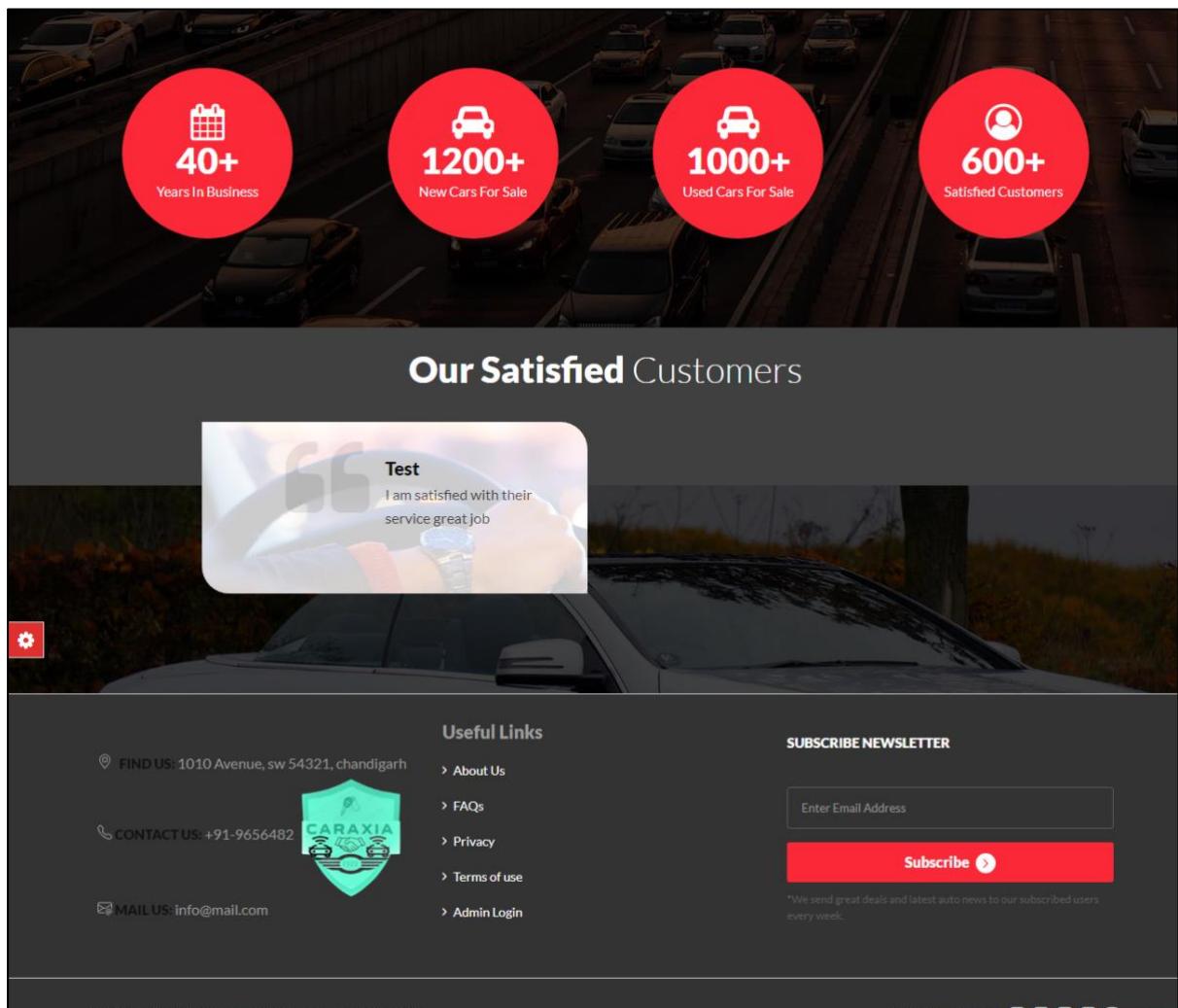


Fig 5.1.4 homepage footer

This is the footer of the website. It contains all the necessary details about the website

5.1.5 Car Listing page

The screenshot displays the 'Car Listing' section of the CarXia website. At the top, there's a navigation bar with links to HOME, ABOUT US, CAR LISTING, FAQS, and CONTACT US. A search bar is also present. The main header features the text 'Car Listing' and 'Home > Car Listing'. Below the header, there's a banner with three cars. To the left, a sidebar titled 'Find Your Car' includes dropdowns for 'Select Brand' and 'Select Fuel Type', and a red 'Search Car' button. The main content area shows a list of 9 listings:

- Toyota , Toyota Fortuner**
\$3000 Per Day
5 seats, 2020 model, Petrol
[View Details](#)
- Nissan , Nissan Sunny 2020**
\$400 Per Day
5 seats, 2018 model, CNG
[View Details](#)
- Maruti , THE NEXT-GEN ERTIGA**
\$200 Per Day
7 seats, 2022 model, Petrol
[View Details](#)
- Toyota , Land Cruiser-Diesel**
\$600 Per Day
5 seats, 2022 model, Diesel
[View Details](#)

On the left side, there's a 'Recently Listed Cars' sidebar with four items:

- Maruti , THE NEXT-GEN ERTIGA
\$200 Per Day
- Toyota , INNOVA Crysta
\$400 Per Day
- Toyota , Land Cruiser-Diesel
\$600 Per Day
- Maruti , Maruti Swift Dzire ZXI Plus
\$70 Per Day

The footer contains useful links, a newsletter sign-up form, and social media links.

Fig 5.1.5 Car Listing
This page contain the various categories of Car

5.1.6 Toyota, INNOVA Crysta page

The screenshot displays the Toyota INNOVA Crysta page on the CarAxia website. At the top, there's a navigation bar with links for HOME, ABOUT US, CAR LISTING, FAQS, and CONTACT US. A search bar is also present. Below the navigation is a large banner featuring three images: an exterior view of the car, the interior dashboard, and the interior seating area. Navigation arrows labeled 'Prev' and 'Next' are visible at the bottom of the banner.

Vehicle Details:

- RegYear: 2023
- Fuel Type: Diesel
- Seats: 7

Pricing: ₹ 400 Per Day

Booking Options:

- Share: [Facebook](#) [Twitter](#) [LinkedIn](#) [Google+](#)
- Book Now
- From Date: dd-mm-yyyy
- To Date: dd-mm-yyyy
- Message
- UPI Credit/Debit Cards
- LOGIN FOR BOOK

Similar Cars:

Car Model	Price	Seats	Fuel Type	Model Year
Toyota, Toyota Fortuner	\$ 3000	5 seats	Petrol	2020 model
Toyota, Land Cruiser-Diesel	\$ 600	5 seats	Diesel	2022 model
Toyota, INNOVA Crysta	\$ 400	7 seats	Diesel	2023 model

Fig 5.1.6 Toyota, INNOVA Crysta
This page contains various Vehicle(Car) named INNOVA Crysta

5.1.7 Toyota, Land Cruiser

The screenshot shows a car rental listing for a Toyota Land Cruiser-Diesel. At the top, there's a navigation bar with links for HOME, ABOUT US, CAR LISTING, FAQS, and CONTACT US. A search bar is also present. Below the navigation, there's a collage of images showing the exterior and interior of the vehicle. The main title is "Toyota , Land Cruiser-Diesel" with a price of ₹ 600 per day. Below the title, there are filters for Reg Year (2022), Fuel Type (Diesel), and Seats (5). To the right, there's a "Share" button with icons for Facebook, Twitter, LinkedIn, and Google+. A "Book Now" section allows users to enter their From Date (dd-mm-yyyy) and To Date (dd-mm-yyyy), send a message, and choose payment methods like UPI or Credit/Debit Cards. A red "LOGIN FOR BOOK" button is at the bottom. The "Vehicle Overview" tab is selected, displaying a detailed description of the vehicle's features and performance. Below this, there's a "Similar Cars" section featuring the Toyota Fortuner and Toyota INNOVA Crysta.

Vehicle Overview

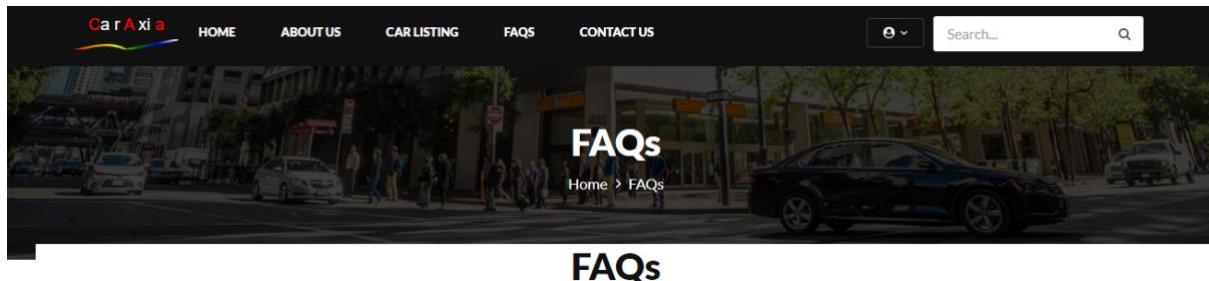
Type: V6 0.00335 m3 Turbo Diesel Engine (3.3 Litre) COLOURS THAT REFLECT CHARACTER THE COLOURS OF THE LAND CRUISER ARE NOT JUST A FEW SHADES PUT TOGETHER. IT'S CAREFULLY CURATED TO REFLECT THE INDOMITABLE CHARACTER AND UNTAMED INDIVIDUALITY OF THE LAND CRUISER. NO HOLDS BARRED The Land Cruiser is a powerhouse unlike any other. Armed with a raging 3.3 litre diesel engine that's seamlessly mated with a 10-speed automatic transmission, delivers a hefty 227kW of power and 700Nm of torque, it can conquer the unimaginable, the impossible. LUXURY THAT KNOWS NO BOUNDS The Land Cruiser is designed to be formidable and is steeped in luxury. And speaking of luxury, its interiors are simply suave and sublime with a classic ambience. The 5U-seater leather seats are plush and provide superior ergonomic support while conquering uncharted territory. Decked with 31.2 cm infotainment & 14U JBL Speakers you can dominate and conquer in full entertainment. CONTROL COMMAND CONQUER A CAPTAIN SHOULD ALWAYS BE IN CONTROL OF HIS SHIP. WITH THE LAND CRUISER, IT'S SPOT ON. IT IS ALWAYS IN CONTROL AND MAKES NO EXCUSES. WHEN WE SAY, IT NEVER FAILS YOU, NOBODY WILL EVER DOUBT IT. THAT'S HOW TRUSTED THE LAND CRUISER IS.

Similar Cars

Vehicle	Price	Model Year	Seats	Fuel Type
Toyota , Toyota Fortuner	\$ 3000	2020 model	5 seats	Petrol
Toyota , Land Cruiser-Diesel	\$ 600	2022 model	5 seats	Diesel
Toyota , INNOVA Crysta	\$ 400	2023 model	7 seats	Diesel

Fig 5.1.7 Toyota, Land Cruiser
This page contains Vehicle(Car) named Land Cruiser

5.1.8 FAQs(Frequency Asked Questions) page



1. How can I rent a car from CarAxia?

- To rent a car from CarAxia, you can either visit our website or contact our customer service team directly. Our user-friendly online platform allows you to browse through available vehicles, select your desired rental dates, and make a reservation hassle-free.

2. What types of vehicles does CarAxia offer for rent?

- CarAxia offers a wide range of vehicles to suit various needs and preferences, including compact cars, sedans, SUVs, vans, and luxury vehicles. We regularly update our fleet to provide our customers with the latest models and options.

3. What are the rental rates and payment options?

Fig 5.1.8 FAQs

This page contains the details of a FAQs(Frequency Asked Questions).

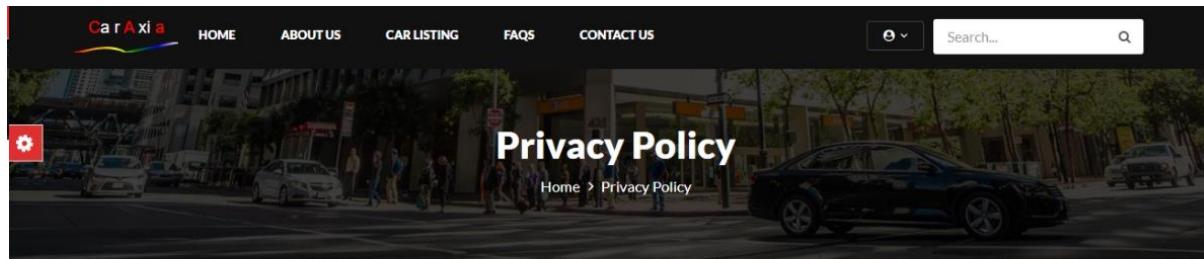
5.1.9 User Activities



Fig 5.1.9 user activities

This page contains the details of Users Activities.

5.1.10 Privacy Policy page



Privacy Policy

Last Updated: 20Jan-2024

This Privacy Policy describes how CarAxia ("we," "us," or "our") collects, uses, and shares information about you when you use our website and services.

Information We Collect

We collect information you provide directly to us when you use our website or services. This may include:

- **Personal information such as your name, email address, phone number, and postal address when you register for an account.**
- **Payment information when you make a booking or payment.**
- **Information you provide when you contact us for customer support or other inquiries.**
- **Information we collect automatically when you use our website or services, including your IP address, browser type, operating system, and device information.**

How We Use Your Information

We may use the information we collect for various purposes, including to:

- **Provide, maintain, and improve our website and services.**
- **Process and fulfill bookings and transactions.**
- **Communicate with you about your account, bookings, and updates to our services.**
- **Respond to your inquiries and provide customer support.**

Security

We take reasonable measures to protect the information we collect from unauthorized access, disclosure, alteration, or destruction.

Changes to This Privacy Policy

We may update this Privacy Policy from time to time. Any changes will be effective when we post the revised policy on our website.

Contact Us

If you have any questions or concerns about our Privacy Policy, please contact us at +91-9874563210.

Fig 5.1.10 Privacy Policy

This page contains the details of the Privacy Policy.

5.1.11 About us

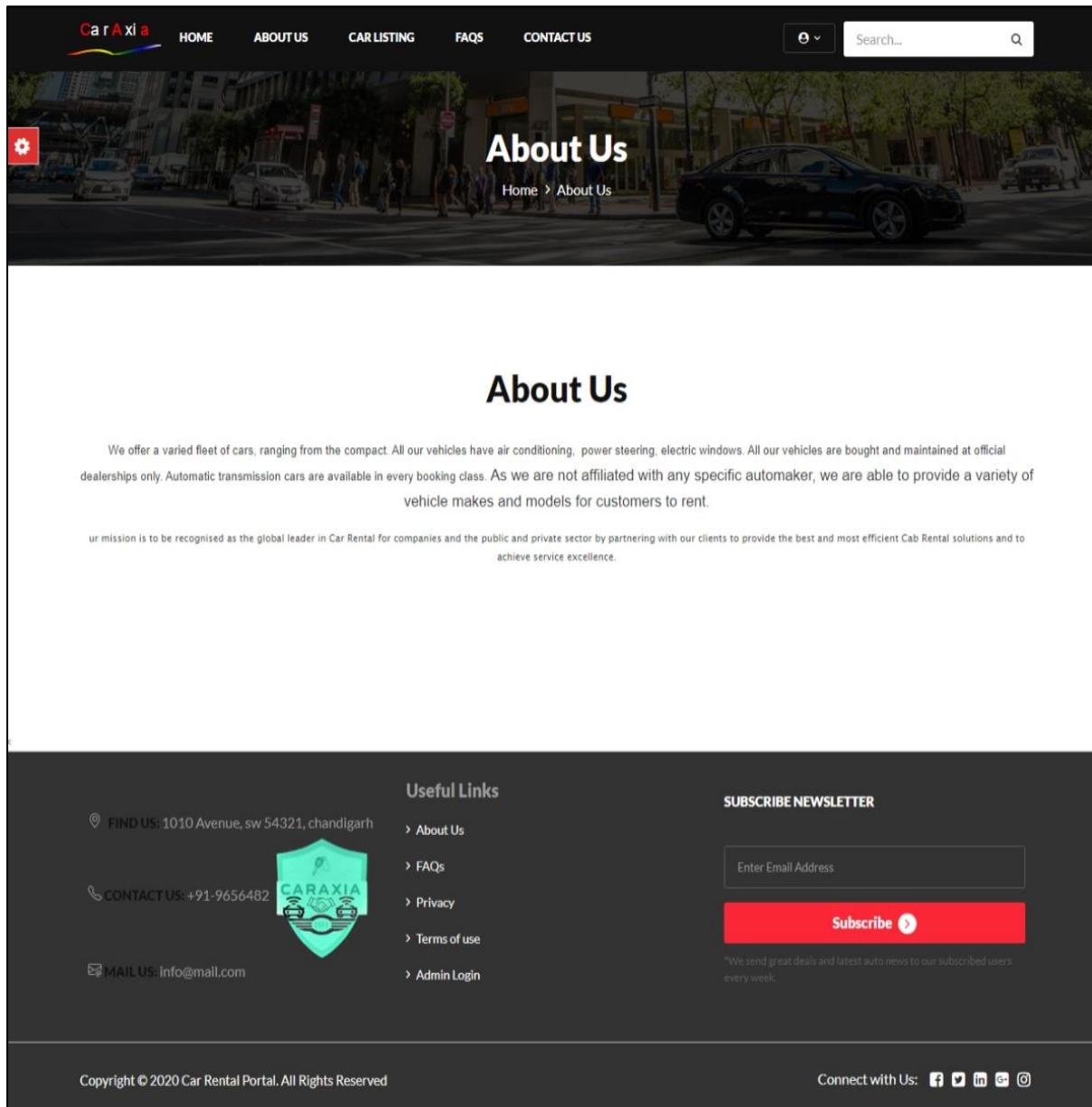


Fig 5.1.11 About us

It contains the information about the company

5.1.12 Terms and conditions

The screenshot shows the 'Terms and Conditions' page of the CarAxia website. At the top, there's a navigation bar with links for HOME, ABOUT US, CAR LISTING, FAQS, and CONTACT US. A search bar is also present. The main title 'Terms and Conditions' is centered above the content area. Below the title, a breadcrumb navigation shows 'Home > Terms and Conditions'. The main content area has a heading 'Terms and Conditions' and a welcome message. It includes a 'Table of Contents' with numbered sections from 1 to 12. The sections cover various topics such as Booking Process, Cancellation Policy, Vehicle Availability, User Responsibilities, Contact Information, Trademarks, Website Access, and User Content Submission. Each section contains detailed text and bullet points. There are also sections for Payment Terms, Privacy Policy, and a note about last updated date (20Jan-2024). At the bottom, there's a footer with 'Useful Links' (About Us, FAQs, Privacy, Terms of use, Admin Login), a newsletter sign-up form, and copyright information.

Welcome to CarAxia, your premier destination for car rental services. These Terms and Conditions govern your use of our platform and outline the terms upon which we provide our services. By accessing our website and making a booking, you agree to abide by these terms and conditions. Please carefully review the following terms:

Table of Contents:

1. Booking Process
2. Cancellation Policy
3. Vehicle Availability
4. User Responsibilities
5. Contact Information
6. Trademarks
7. Website Access
8. User Content Submission

1. Booking Process

- Users are required to provide accurate and current information during the booking process.
- By submitting a booking request, users acknowledge their understanding and acceptance of the terms outlined herein.

2. Cancellation Policy

- Users may cancel their bookings, provided they contact our administrative team before confirmation.
- Any cancellations made after confirmation may result in forfeiture of payment, as specified in our policies.

3. Vehicle Availability

- All bookings are subject to vehicle availability.
- In the event that a booked vehicle becomes unavailable, users will be promptly notified, and suitable alternatives will be offered, if feasible.

4. User Responsibilities

- Users assume full responsibility for the proper and lawful use of the rented vehicle.
- Any damages incurred during the rental period are the sole responsibility of the user, and appropriate charges may be applied.

5. Payment Terms

- Payment for bookings must be made in accordance with the specified terms and deadlines.
- Prices may vary based on factors such as vehicle type, rental duration, and additional services selected.

6. Privacy Policy

- We are committed to protecting the privacy of our users and safeguarding their personal information. Please refer to our Privacy Policy for detailed information regarding data collection, usage, and protection.

Last Updated: 20Jan-2024

For any questions or concerns regarding these terms and conditions, please contact us at your earliest convenience.

Thank you for choosing CarAxia. We appreciate your trust in us.

Useful Links

- > About Us
- > FAQs
- > Privacy
- > Terms of use
- > Admin Login

SUBSCRIBE NEWSLETTER

Enter Email Address

Subscribe

*We send great deals and latest auto news to our subscribed users every week.

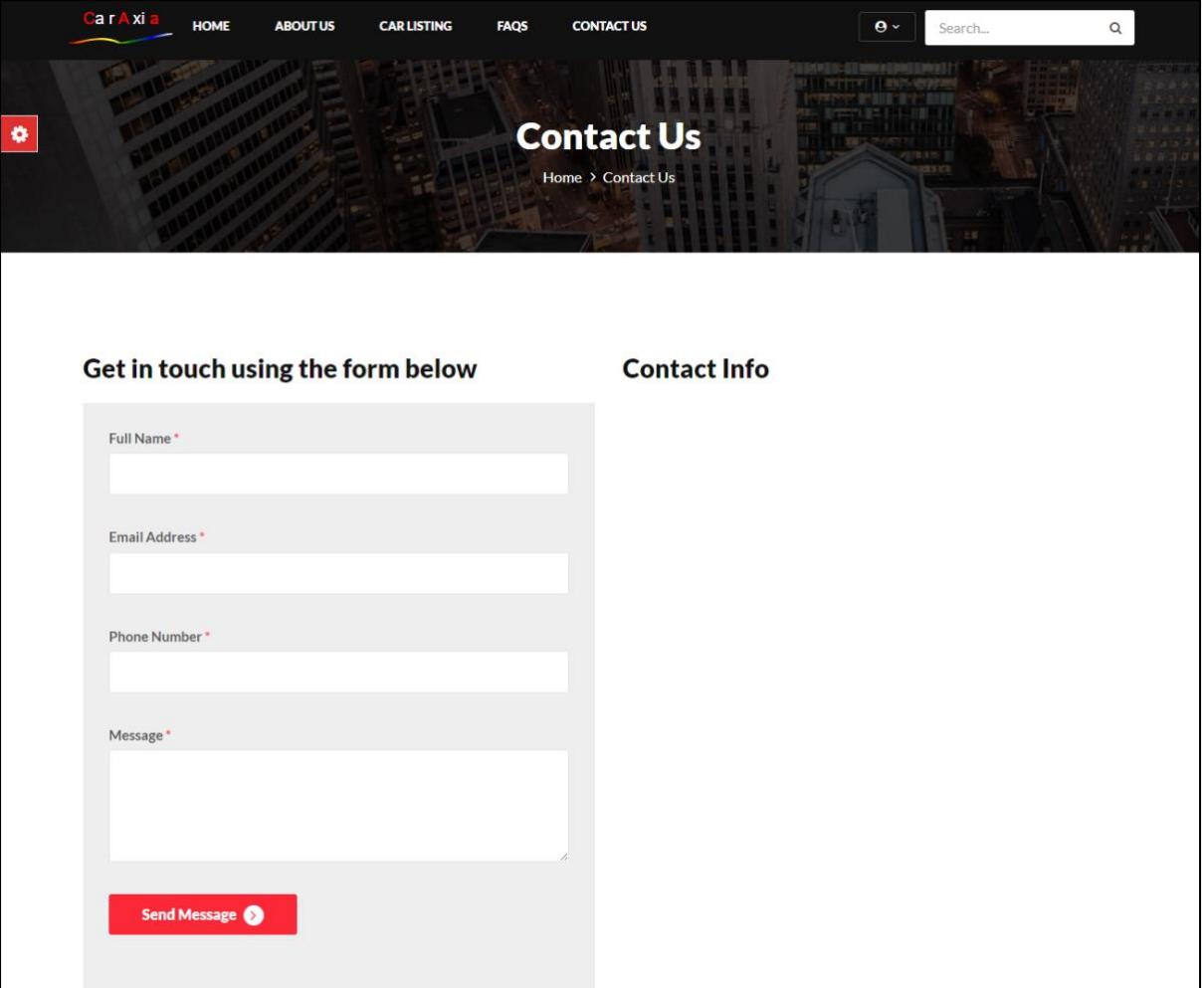
Copyright © 2020 Car Rental Portal. All Rights Reserved

Connect with Us:

Fig 5.1.12 Terms and Conditions

It contains the terms and conditions of the company

5.1.13 Contact us



The screenshot shows the 'Contact Us' page of the CarXia website. At the top, there's a navigation bar with links for HOME, ABOUT US, CAR LISTING, FAQS, and CONTACT US. A search bar is also present. The main heading 'Contact Us' is centered above a breadcrumb trail 'Home > Contact Us'. Below this, there's a section titled 'Get in touch using the form below' containing four input fields: 'Full Name *', 'Email Address *', 'Phone Number *', and 'Message *'. A red 'Send Message' button is at the bottom. To the right of the form, there's a sidebar titled 'Contact Info'.

Contact Us

Home > Contact Us

Get in touch using the form below

Full Name *

Email Address *

Phone Number *

Message *

Send Message

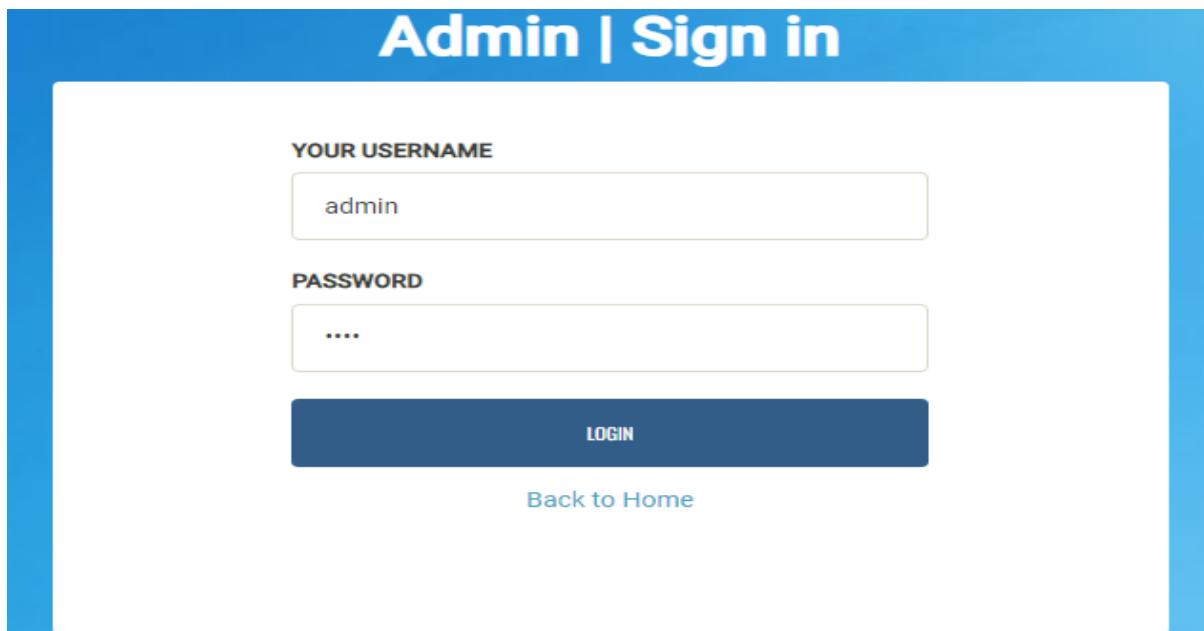
Contact Info

Fig 5.1.13 contact us

The user can send message to the company

5.2 Admin Module Design

5.2.1 Admin login



The image shows a login form titled "Admin | Sign in". It has fields for "YOUR USERNAME" (containing "admin") and "PASSWORD" (containing "****"). A "LOGIN" button is at the bottom, and a "Back to Home" link is below it.

Fig 5.2.1 Admin Login

User has to login with admin username and password to access admin dashboard

5.2.2 Admin Dashboard

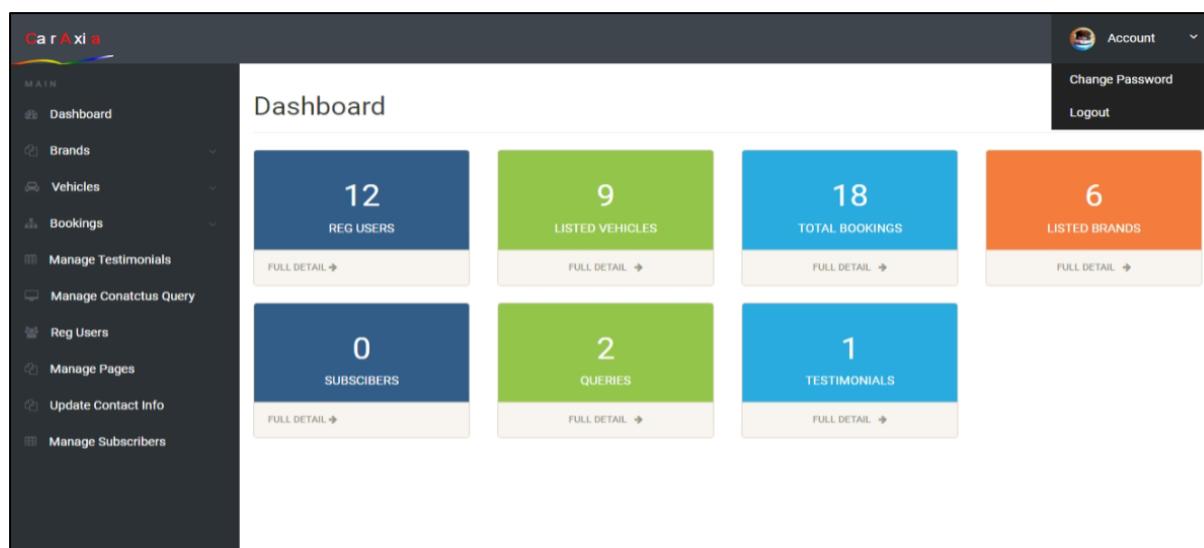


Fig 5.2.2 Admin Dashboard

This is admin dashboard where admin can perform various functions.

5.2.3 Registered Users

The screenshot shows the 'Registered Users' section of the CarAxia application. On the left, there is a dark sidebar with various menu items: Dashboard, Brands, Vehicles, Bookings, Manage Testimonials, Manage Contact Query, Reg Users, Manage Pages, Update Contact Info, and Manage Subscribers. The 'Reg Users' item is highlighted. The main content area has a title 'Registered Users' and a sub-section 'REG USERS'. It includes a search bar and a table with columns: #, Name, Email, Contact no, DOB, Address, City, Country, and Reg Date. There are five entries listed:

#	Name	Email	Contact no	DOB	Address	City	Country	Reg Date
1	Test	test@gmail.com	6465465465		L-890, Gaur City Ghaziabad	Ghaziabad	India	2020-07-07 19:30:49
2	parveen	user@gmail.com	123456					2021-05-14 18:28:35
3	VINAY SINGAL	vinaysingal123ghn@gmail.com	9467672014					2024-03-03 12:37:35
4	as	ashishpanchal731@gmail.com	9876543210					2024-03-16 07:40:58
5	PRINCE	princechahal123ghn@gmail.com	9992334303					2024-04-08 11:36:23

Showing 1 to 5 of 5 entries

Fig 5.2.3 Registered Users

Admin can view the registered users and also delete the user

5.2.4 View Brands

The screenshot shows the 'View Brands' section of the CarAxia application. On the left, there is a dark sidebar with menu items: Dashboard, Brands, Create Brand, Manage Brands, Vehicles, Bookings, Manage Testimonials, Manage Contact Query, Reg Users, Manage Pages, Update Contact Info, and Manage Subscribers. The 'Brands' item is highlighted. The main content area has a title 'Create Brand' with a 'CREATE BRAND' form containing a 'Brand Name' input field and a 'Submit' button. Below this is a section titled 'Manage Brands' with a sub-section 'LISTED BRANDS'. It includes a search bar and a table with columns: #, Brand Name, Creation Date, Upcation date, and Action. There are six entries listed:

#	Brand Name	Creation Date	Upcation date	Action
1	Maruti	2017-05-10 21:54:04	2017-06-19 12:12:23	<input checked="" type="checkbox"/> <input type="checkbox"/>
2	BMW	2017-05-18 21:54:50		<input checked="" type="checkbox"/> <input type="checkbox"/>
3	Audi	2017-05-18 21:55:03		<input checked="" type="checkbox"/> <input type="checkbox"/>
4	Nissan	2017-05-18 21:56:13		<input checked="" type="checkbox"/> <input type="checkbox"/>
5	Toyota	2017-05-18 21:59:24		<input checked="" type="checkbox"/> <input type="checkbox"/>
6	Volkswagen	2017-05-19 11:52:10	2020-07-07 19:41:09	<input checked="" type="checkbox"/> <input type="checkbox"/>

Showing 1 to 6 of 5 entries

Fig 5.2.4 View Brands

Admin can view all products and add , update and delete the different Brands

5.2.5 Booking Details

The screenshot shows the 'Booking Details' page under the 'Bookings' section of the CarAxia admin interface. The left sidebar lists various management options like Dashboard, Brands, Vehicles, Bookings, etc. The main content area is titled '#800594601 Booking Details'. It displays 'User Details' with fields for Booking No., Name, Email Id, Contact No., Address, City, and Country. Below that is the 'Booking Details' section with fields for Vehicle Name, Booking Date, From Date, To Date, Total Days, Rent Per Days, Grand Total, and Booking Status. At the bottom are 'Confirm Booking' and 'Cancel Booking' buttons, and a 'Print' link.

Fig 5.2.5 Booking Details

Admin can view the details of all the Booking placed

5.2.6 Vehicles Details

The screenshot shows the 'Post A Vehicle' page under the 'Vehicles' section of the CarAxia admin interface. The left sidebar lists various management options. The main content area is titled 'Post A Vehicle' and has a 'BASIC INFO' section with fields for Vehicle Title*, Select Brand*, Vehical Overview*, Price Per Day(in USD)*, Select Fuel Type*, Model Year*, and Seating Capacity*. Below that is an 'Upload Images' section with five image upload fields. At the bottom is an 'ACCESSORIES' section with checkboxes for Air Conditioner, Power Door Locks, AntiLock Braking System, Brake Assist, Power Steering, Driver Airbag, Passenger Airbag, Power Windows, CD Player, Central Locking, Crash Sensor, and Leather Seats. There are 'Cancel' and 'Save changes' buttons at the bottom.

Fig 5.2.6.1 Vehicles

Admin can post the new vehicles

#	Vehicle Title	Brand	Price Per day	Fuel Type	Model Year	Action
1	Toyota Fortuner	Toyota	3000	Petrol	2020	<input checked="" type="checkbox"/> <input type="button" value="X"/>
2	Maruti Suzuki Wagon R	Maruti	500	Petrol	2019	<input checked="" type="checkbox"/> <input type="button" value="X"/>
3	Nissan Kicks	Nissan	800	Petrol	2020	<input checked="" type="checkbox"/> <input type="button" value="X"/>
4	Nissan Sunny 2020	Nissan	400	CNG	2018	<input checked="" type="checkbox"/> <input type="button" value="X"/>
5	Maruti Suzuki Vitara Brezza	Maruti	600	Petrol	2018	<input checked="" type="checkbox"/> <input type="button" value="X"/>
6	Maruti Swift Dzire ZXi Plus	Maruti	1000	CNG	2020	<input checked="" type="checkbox"/> <input type="button" value="X"/>
7	Land Cruiser-Diesel	Toyota	6000	Diesel	2022	<input checked="" type="checkbox"/> <input type="button" value="X"/>
8	INNOVA Crysta	Toyota	400	Diesel	2023	<input checked="" type="checkbox"/> <input type="button" value="X"/>
9	THE NEXT-GEN ERTIGA	Maruti	200	Petrol	2022	<input checked="" type="checkbox"/> <input type="button" value="X"/>

Fig 5.2.6.2 Vehicles

It contains the information of the car and mange vehicles

5.2.7 Manage Contact Us Queries

#	Name	Email	Contact No	Message	Posting date	Action
1	Kunal	kunal@gmail.com	7977779798	I want to know brach in Chandigarh?	2020-07-07	Read
2	VINAY SINGAL	vinaysingal123ghn@gmail.com	9467672014	Hello CarAxia Team, I hope this message finds you well. I am interested in learning more about your car rental services and would like to inquire about the vehicles available for rent, rental rates, and any special offers or discounts currently available. Additionally, I have a few questions regarding the booking process, rental terms and conditions, and the availability of add-on services such as insurance coverage and GPS navigation. Could you please provide me with more details or direct me to the appropriate resources where I can find the information I need? Thank you for your assistance. I look forward to hearing from you soon.	2024-04-25 21:35:18	Pending

Fig 5.2.7 Manage Contact Us Queries

The Admin can view the users question of contact us query page

5.2.8 Manage Testimonials

#	Name	Email	Testimonials	Posting date	Action
1	Test	test@gmail.com	I am satisfied with their service great job	2020-07-07 20:00:12	Active

Fig 5.2.8 Manage Testimonials

5.2.9 Manage Pages

5.2.10 Manage Contact info & Subscribers

#	Email Id	Subscription Date	Action
1	vinaysingal123ghn@gmail.com	2024-04-23 15:10:56	X

Chapter-6

Database design

6.1 Database

The screenshot shows the phpMyAdmin interface for a database named 'carrental'. On the left, a tree view lists various databases, with 'carrental' expanded to show its tables: admin, tblbooking, tblbrands, tblcontactusinfo, tblcontactusquery, tblpages, tblsubscribers, tbtestimonial, tblusers, and tblvehicles. The main panel displays a table of 10 tables with columns for Table, Action, Rows, Type, Collation, Size, and Overhead. The table data is as follows:

Table	Action	Rows	Type	Collation	Size	Overhead
admin	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16.0 Kib	-
tblbooking	Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16.0 Kib	-
tblbrands	Browse Structure Search Insert Empty Drop	6	InnoDB	latin1_swedish_ci	16.0 Kib	-
tblcontactusinfo	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16.0 Kib	-
tblcontactusquery	Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16.0 Kib	-
tblpages	Browse Structure Search Insert Empty Drop	4	MyISAM	latin1_swedish_ci	209.0 Kib	-
tblsubscribers	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16.0 Kib	-
tbtestimonial	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16.0 Kib	-
tblusers	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	32.0 Kib	-
tblvehicles	Browse Structure Search Insert Empty Drop	9	InnoDB	latin1_swedish_ci	16.0 Kib	-
10 tables	Sum	33	InnoDB	utf8mb4_general_ci	369.0 Kib	0 B

Fig 6.1 Database structure

This is database “carrental” which contains tables which stores data of users and products:

- Admin
- tblbooking
- tblbrands
- tblcontactusinfo
- tblcontactusquery
- tblpages
- tblsubscribers
- tbtestimonial
- tblusers
- tblvehicles

6.2 Table: Admin

COLUMN	TYPE	NULL	DESCRIPTION
id (<i>Primary</i>)	int(11)	No	It is storing the admin id
UserName	varchar(100)	No	It will store the Admin logging id.
Password	varchar(100)	No	It will store the admin password.
updationDate	timestamp	No	It will store the password updating date.

Fig 6.2 Admin_login structure

It stores the data of the Admin such as Admin_name and Admin_password

6.3 Table: Booking

COLUMN	TYPE	NULL	DESCRIPTION
id (<i>Primary</i>)	int(11)	No	Booking id.
BookingNumber	bigint(12)	Yes	It will store booking number.
userEmail	varchar(100)	Yes	It will store the User email id.
VehicleId	int(11)	Yes	It will store the vehicleID.
FromDate	varchar(20)	Yes	It will store the pickup date.
ToDate	varchar(20)	Yes	It will store the return date.
pickupTiming	varchar(10)	Yes	It will store the pickup timing.
returnTiming	varchar(16)	No	It will store the return timing.
message	varchar(255)	Yes	It will store the message given by the user.
Status	int(11)	Yes	It will store the booking status.
PostingDate	timestamp	No	Posting date.
LastUpdationDate	timestamp	Yes	Updating date.

Fig 6.3 user structure

It contains the information of the users booking details.

6.4 Table: brands

COLUMN	TYPE	NULL	DESCRIPTION
id (<i>Primary</i>)	int(11)	No	It will store the car brand id.
BrandName	varchar(120)	No	It will store the car brand name
CreationDate	timestamp	Yes	It will store the brand creations date.
UpdationDate	timestamp	Yes	It will store the brand modification date.

Fig 6.4 Vehicle brands

6.5 Table: contact us info

COLUMN	TYPE	NULL	DESCRIPTION
id (<i>Primary</i>)	int(11)	No	It will store the id.
Address	tinytext	Yes	It will store the company's address.
EmailId	varchar(255)	Yes	It will store the company's email id.
ContactNo	char(11)	Yes	It will store the company's contact no.

Fig 6.5 Contact us info

It contains the information about all query of the users.

6.6 Table: contact us query

COLUMN	TYPE	NULL	DESCRIPTION
id (<i>Primary</i>)	int(11)	No	It will store the id.
name	varchar(100)	Yes	It will store the name
EmailId	varchar(120)	Yes	It will store the email id.
ContactNumber	char(11)	Yes	It will store the contact number
Message	longtext	Yes	It will store the user's query.
PostingDate	timestamp	No	Posting date
Status	int(11)	Yes	Status.

6.7 Table: Pages

COLUMN	TYPE	NULL	DESCRIPTION
id (<i>Primary</i>)	int(11)	No	It will store the id.
PageName	varchar(255)	Yes	It will store the page name.
type	varchar(255)	No	It will store the page type.
detail	longtext	No	It will store the contents of the pages.

Fig 6.7 Different pages data

6.8 Table: Payment

COLUMN	TYPE	NULL	DESCRIPTION
id (<i>Primary</i>)	int(220)	No	It will store the id.
cardno	varchar(225)	No	It will store the card number.
expdate	date	No	It will store the exp date of the card.
cvvno	int(3)	No	It will store the cvv number of the card
cname	varchar(255)	No	It will store the card holder name

Fig 6.8 user payment structure

It Contains the information of car book by some users.

6.9 Table: Subscribers

COLUMN	TYPE	NULL	DESCRIPTION
id (<i>Primary</i>)	int(11)	No	It will store the id.
SubscriberEmail	varchar(120)	Yes	It will store the subscribers email id.
PostingDate	timestamp	Yes	It will store the subscription date.

Fig 6.8 user subscribers structure

It Contains the information of subscribed by some users.

6.10 Table: Testimonial

COLUMN	TYPE	NULL	DESCRIPTION
id (<i>Primary</i>)	int(11)	No	It will store the id.
UserEmail	varchar(100)	No	It will store the user email-id.
Testimonial	mediumtext	No	It will store the testimonials massage.
PostingDate	timestamp	No	It will store the posting date.
status	int(11)	Yes	It will store the testimonials status.

Fig 6.9 user testimonial data

6.11 Table: Users

COLUMN	TYPE	NULL	DESCRIPTION
id (<i>Primary</i>)	int(11)	No	It will store the id.
FullName	varchar(120)	Yes	It will store the username.
EmailId	varchar(100)	Yes	It will store the user email-id.
Password	varchar(100)	Yes	It will store the user password.
ContactNo	char(11)	Yes	It will store the user contact number.
Dob	varchar(100)	Yes	It will store the user date of birth.
Address	varchar(255)	Yes	It will store the user address.
City	varchar(100)	Yes	It will store the user city address.
Country	varchar(100)	Yes	It will store the user country.
RegDate	timestamp	Yes	It will store the user registration date.
UpdationDate	timestamp	Yes	It will store the updating date of profile.
DL_NO	varchar(50)	No	It will store the user driver-license.
Uimage1	varchar(220)	Yes	It will store the user profile image.

Fig 6.10 It contains the information of users and their activities .

6.12 Table: Vehicles

COLUMN	TYPE	NULL	DESCRIPTION
<i>id (Primary)</i>	int(11)	No	It will store the vehicle id.
VehiclesTitle	varchar(150)	Yes	It will store the vehicle name.
VehiclesBrand	int(11)	Yes	It will store the vehicle brand.
VehiclesOverview	longtext	Yes	It will store the vehicle overview.
PricePerDay	int(11)	Yes	It will store the vehicle rent price.
FuelType	varchar(100)	Yes	It will store the vehicle fuel type.
ModelYear	int(6)	Yes	It will store the vehicle model year.
SeatingCapacity	int(11)	Yes	It will store the vehicle seating capacity.
Vimage1	varchar(120)	Yes	It will store the vehicle first image.
Vimage2	varchar(120)	Yes	It will store the vehicle second image.
Vimage3	varchar(120)	Yes	It will store the vehicle third image.
Vimage4	varchar(120)	Yes	It will store the vehicle fourth image.
Vimage5	varchar(120)	Yes	It will store the vehicle fifths image.
AirConditioner	int(11)	Yes	It will store the vehicle air-conditioner status.
PowerDoorLocks	int(11)	Yes	It will store the vehicle power door locks status.
AntiLockBrakingSystem	int(11)	Yes	It will store the vehicle antilock braking system status.
BrakeAssist	int(11)	Yes	It will store the vehicle brake-assist status.
PowerSteering	int(11)	Yes	It will store the vehicle power steering status.
DriverAirbag	int(11)	Yes	It will store the vehicle driver air bag status.
PassengerAirbag	int(11)	Yes	It will store the vehicle passenger air bag status.
PowerWindows	int(11)	Yes	It will store the vehicle power windows status.
CDPlayer	int(11)	Yes	It will store the vehicle cd player status.
CentralLocking	int(11)	Yes	It will store the vehicle central-locking status.
CrashSensor	int(11)	Yes	It will store the vehicle crash sensor status.
LeatherSeats	int(11)	Yes	It will store the vehicle leather seats status.
RegDate	timestamp	No	It will store the vehicle registration date.
UpdationDate	timestamp	Yes	It will store the vehicle modification date.

Chapter-7

Software Testing

7.1 Testing

- Software testing Software testing is the process of evaluating a software application to ensure it meets the required specifications and functions correctly.
- It includes both manual and automated tests to assess the correctness, completeness, security, and quality of the computer software.
- It is used to identify any bugs, flaws, or compatibility issues in the software before it is released to the public.

7.2 Types of Software Testing

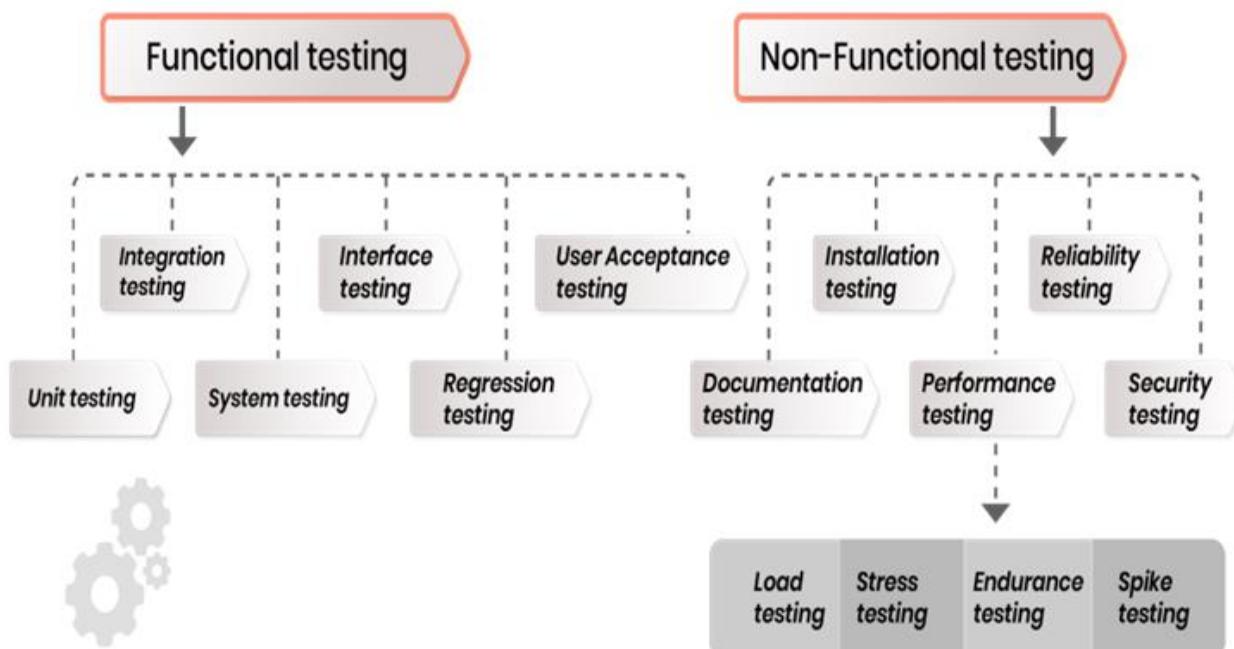


Fig 7.2.1

7.2.1 Functional Testing

- Functional testing is a type of testing in software engineering that focuses on verifying that a software application or system is functioning correctly and meets the specified requirements.
- In functional testing, the tester will create test cases that cover different aspects of the application's functionality, and then execute those test cases to ensure that the application behaves as expected. For example, a functional tester might create test cases to verify that the application can accept user input, process that input correctly, and display the expected output.
- Functional testing is typically performed early in the software development process, after the application has been built but before it is released to users. It is an important step in ensuring that the application is of high quality and meets the needs of the users.
- Functional testing can be performed manually by a tester, or it can be automated using testing tools. Automated functional testing can be particularly useful for testing applications with a large number of test cases or applications that need to be tested frequently

7.2.1.1 Unit Testing

- Unit testing is a type of software testing where individual units or components of a software are tested to determine if they are fit for use.
- Unit tests are typically written and run by software developers to ensure that code meets its design and behaves as expected.
- Unit tests can also be used to validate the logic of an application and make sure that it is functioning correctly.

7.2.1.2 Integration Testing

- Integration testing is a type of software testing that verifies the interfaces between components of a system to detect defects in the integration between components.
- Integration testing tests the combined components of a system or an application as a single entity.
- The purpose of integration testing is to verify functional, performance, and reliability of a system that is composed of multiple components.
- Integration testing is a type of software testing that aims to verify the interaction and communication between different components or systems.
- During integration testing, individual units or components are combined and tested as a group. The goal is to ensure that these units or components work together correctly and meet the requirements of the overall system. This involves testing the interfaces between the components and verifying that data is correctly passed between them.
- Integration testing can be performed at various levels, such as component integration testing, which involves testing the integration of individual components within a system, and system integration testing, which involves testing the integration of the entire system.
- Integration testing is an important part of the software development process because it helps identify any issues that may arise when different components or systems are combined and work together. It also helps ensure that the overall system meets the functional and performance requirements specified in the design.

7.2.1.3 System Testing

- System testing is a type of software testing that evaluates the end-to-end functioning of a complete system.
- It is done to ensure that the system meets the specified requirements. System testing verifies the system-level functionality and performance, and involves validation of the system's accuracy, reliability, scalability and security.
- System testing takes into account all components of the system and tests the functionality of the system as a whole.

7.2.1.4 User Acceptance Testing

- User Acceptance Testing (UAT) is a type of software testing that verifies if a software application meets the business requirements and is accepted by the end user.
- It is the final testing performed once the functional, system, integration and regression tests are completed.
- UAT is a type of black box testing, where the user provides inputs and validates the output without knowing how the system works. The goal of UAT is to ensure that the system satisfies the user needs.
- User acceptance testing (UAT) is a type of software testing that is performed by the end user or the client to verify that the software meets their business needs and requirements. It is typically the final phase of the testing process and is performed before the software is deployed in the production environment.
- During UAT, the end user or client evaluates the software by executing a set of predefined test cases and verifying that the software functions as expected.

7.2.1.5 Interface testing

- Interface testing is a type of testing in software engineering that focuses on verifying the behavior of the interfaces between different components or systems.
- An interface is a point of communication between two components or systems, and it can be either a logical interface, such as an API, or a physical interface, such as a network connection.
- In interface testing, the tester will create test cases to verify that the interfaces between different components or systems are functioning correctly and meeting the specified requirements.
- This may include testing the input and output of data through the interfaces, as well as verifying that the interfaces are properly handling errors and exceptions.

7.2.1.6 White Box testing

- White box software testing is a type of software testing in which the internal structure/design/implementation of the item being tested is known to the tester.
- This type of testing is also known as structural testing, glass box testing, clear box testing, open box testing, and code-based testing.
- The goal of white box testing is to test each statement, branch, loop, and path within the software to ensure that the code is valid and that it meets the requirements.

7.2.1.7 Black Box testing

- Black box testing is a software testing method in which the functionality of an application is tested without knowing the internal structures of the application or the code that makes it up.
- This type of testing is used to test the functionality of an application by providing input and examining the output without requiring knowledge of the internal code.
- Examples of black box testing include boundary value analysis, equivalence partitioning, and decision table testing.

7.3 Test cases

Boundary Value Analysis Test Case: This type of test case checks the correctness of software for extreme input values that are below and above the expected range.

- **Functional Test Case:** This type of test case checks the correctness of the software with respect to its intended function.
- **Negative Test Case:** This type of test case checks the correctness of software when invalid input is entered.
- **Compatibility Test Case:** This type of test case checks the correctness of software with different hardware and software platforms.
- **Performance Test Case:** This type of test case checks the correctness of the software with respect to its performance metrics, such as response time and throughput.
- **Security Test Case:** This type of test case checks the correctness of the software with respect to security issues such as authentication, authorization, and encryption.

- **End-To-End Test Case:** This type of test case checks the correctness of the software from end-to-end, including all its components and interfaces.

What are test cases in software Testing

- **Boundary Value Analysis Test Case:** This type of test case checks the correctness of software for extreme input values that are below and above the expected range.
- **Functional Test Case:** This type of test case checks the correctness of the software with respect to its intended function.
- **Negative Test Case:** This type of test case checks the correctness of software when invalid input is entered.
- **Compatibility Test Case:** This type of test case checks the correctness of software with different hardware and software platforms.
- **Performance Test Case:** This type of test case checks the correctness of the software with respect to its performance metrics, such as response time and throughput.
- **Security Test Case:** This type of test case checks the correctness of the software with respect to security issues such as authentication, authorization, and encryption.
- **End-To-End Test Case:** This type of test case checks the correctness of the software from end-to-end, including all its components and interfaces.

7.4 SYSTEM TEST CASES

7.4.1 Register module

S no	Test case	Input	Expected result	Output	Remark
1	Blank Full Name	Blank	Please fill out the field	Please fill out the field	Working
2	Blank Mobile Number	Blank	Please fill out the field	Please fill out the field	Working
3	Blank Email	Blank	Please fill out the field	Please fill out the field	Working
4	Blank Password	Blank	Please fill out the field	Please fill out the field	Working
5	Blank Confirm Password	Blank	Please fill out the field	Please fill out the field	Working
5	correct details	correct	Successful	Successful	working

Test case 1

When user wants to register with blank Full Name

Sign Up



Full Name

9467672014



Please fill out this field.

vinaysignal123ghn@gmail.com

Email available for Registration.

••••••

••••••••

I Agree with [Terms and Conditions](#)

Sign Up

Already got an account? [Login Here](#)

Fig 7.4.1.1

Test case 2

When user wants to register with blank Mobile Number

Sign Up



VINAY SINGAL

Mobile Number

vinaysignal123ghn@gmail.co



Please fill out this field.

Email available for Registration .

.....

.....

I Agree with [Terms and Conditions](#)

Sign Up

Already got an account? [Login Here](#)

Fig 7.4.1.2

Test case 3

When user wants to register with blank Email

The image shows a 'Sign Up' form. At the top right is a close button (X). The form fields include:

- Name: VINAY SINGAL
- Phone Number: 9467672014
- Email Address: (empty field)
- Password: (empty field)

A validation message 'Please fill out this field.' with an exclamation mark icon is displayed above the empty Email Address field. Below the form is a checkbox labeled 'I Agree with Terms and Conditions' and a red 'Sign Up' button. At the bottom, there is a link 'Already got an account? Login Here'.

Fig7.4.1.3

Test case 4

When user wants to register with password

The screenshot shows a 'Sign Up' form with the following fields and status:

- Name: VINAY SINGAL
- Mobile Number: 9467672014
- Email: vinaysigal123ghn@gmail.com (Status: Email available for Registration.)
- Password: A field containing several dots (.....) with an orange exclamation mark icon above it and the text "Please fill out this field." (Status: Error)
- Agreement: A checked checkbox next to the text "I Agree with Terms and Conditions".
- Sign Up button: A large red button labeled "Sign Up".

Below the form, there is a link: "Already got an account? [Login Here](#)".

Fig 7.4.1.4

Test case 5

When user wants to register with Confirm Password

Sign Up



VINAY SINGAL

9467672014

vinaysignal123ghn@gmail.com

Email available for Registration .

Confirm Password



I Agree with [Terms and Conditions](#)



Please fill out this field.

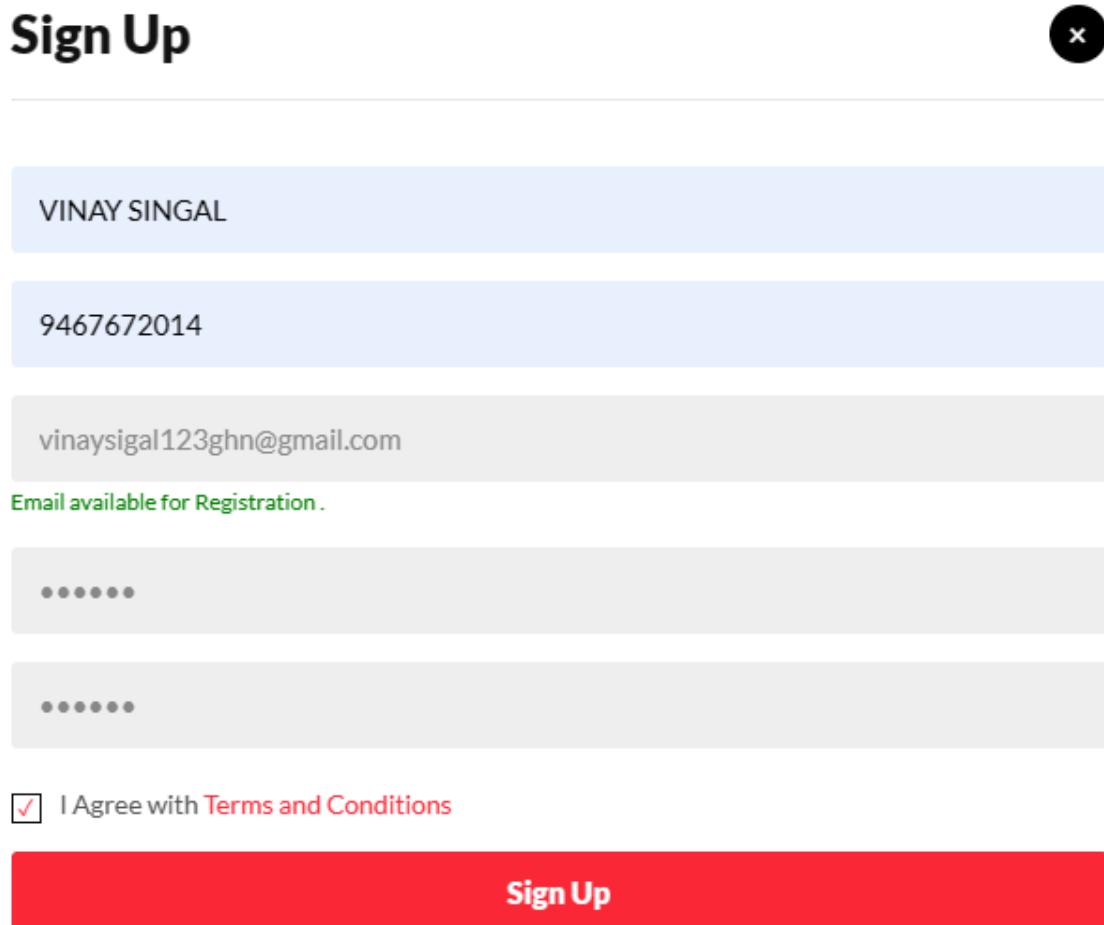
Sign Up

Already got an account? [Login Here](#)

Fig 7.4.1.5

Test case 6

When user enters all correct details, the user information is registered and redirect to login form



The screenshot shows a mobile-style sign-up form titled "Sign Up". At the top right is a black circular "X" button. The form fields are as follows:

- Name: VINAY SINGAL
- Mobile Number: 9467672014
- Email: vinaysignal123ghn@gmail.com
- A green success message below the email field reads "Email available for Registration ."
- Two masked password fields, each showing five dots.
- A checkbox labeled "I Agree with Terms and Conditions" with a checked red box.
- A large red "Sign Up" button at the bottom.

At the bottom of the screen, there is a link "Already got an account? [Login Here](#)".

Fig 7.4.1.6

7.4.2 Login Module

S No	Test Case	Input	Expected Result	Output	Remark
1	Blank all fields	Blank	Please fill out the field	Please fill out the field	Working
2	Wrong Email Address	Not Matches	Invalid username or password	Invalid username or password	Working
3	Wrong Password	Not Matches	Invalid username or password	Invalid username or password	Working
4	Wrong Captcha	Not Matches	Invalid Captcha	Invalid Captcha	Working
5	Correct Captcha, username and password	Matches	Login Successful	Login Successful	Working

Test Case 1

When user wants to login with blank field

The screenshot shows a login interface. At the top left is a 'Login' button. Below it is a 'Email address*' input field, which contains a placeholder 'Email address*'. To the right of this field is a validation message: 'Please fill out this field.' with an orange exclamation mark icon. Below the email field is a 'Password*' input field containing several dots ('.....'). In the center of the page is a CAPTCHA field with the text 'VA61H3' and a placeholder 'Enter Captcha'. To the right of the CAPTCHA is a circular refresh/circular arrow icon. At the bottom is a large red 'Login' button. Below the button are links for 'Don't have an account? [Signup Here](#)' and 'Forgot Password ?'.

Fig 7.4.2.1

Test case 2

When user wants to login with wrong Email Address

The screenshot shows a login interface. At the top left is a 'Login' button. Below it is an input field containing the email address 'vinaysingal123ghn@gmail.com'. A tooltip window titled 'localhost says' displays the message 'Invalid Details' with a small 'x' icon. Below the input field is a password field containing several dots ('.....'). In the center of the page is a CAPTCHA field with the text 'vRMAJx' and a placeholder 'Enter Captcha'. To the right of the CAPTCHA is a circular refresh/circular arrow icon. At the bottom is a large red 'Login' button. Below the button are links for 'Don't have an account? [Signup Here](#)' and 'Forgot Password ?'.

Fig 7.4.2.2

Test case 3

When user wants to login with wrong password

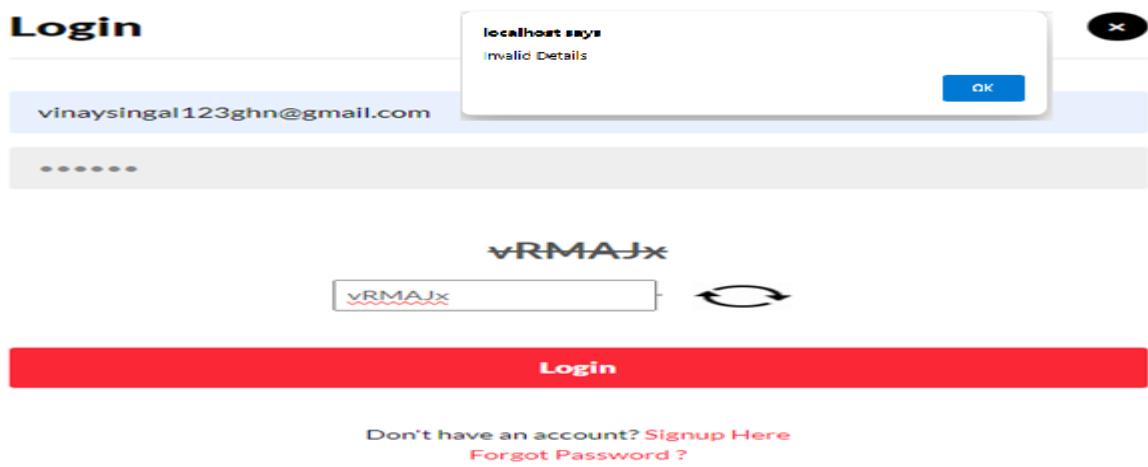


Fig 7.4.2.3

Test case 4

when user wants to login with wrong Captcha

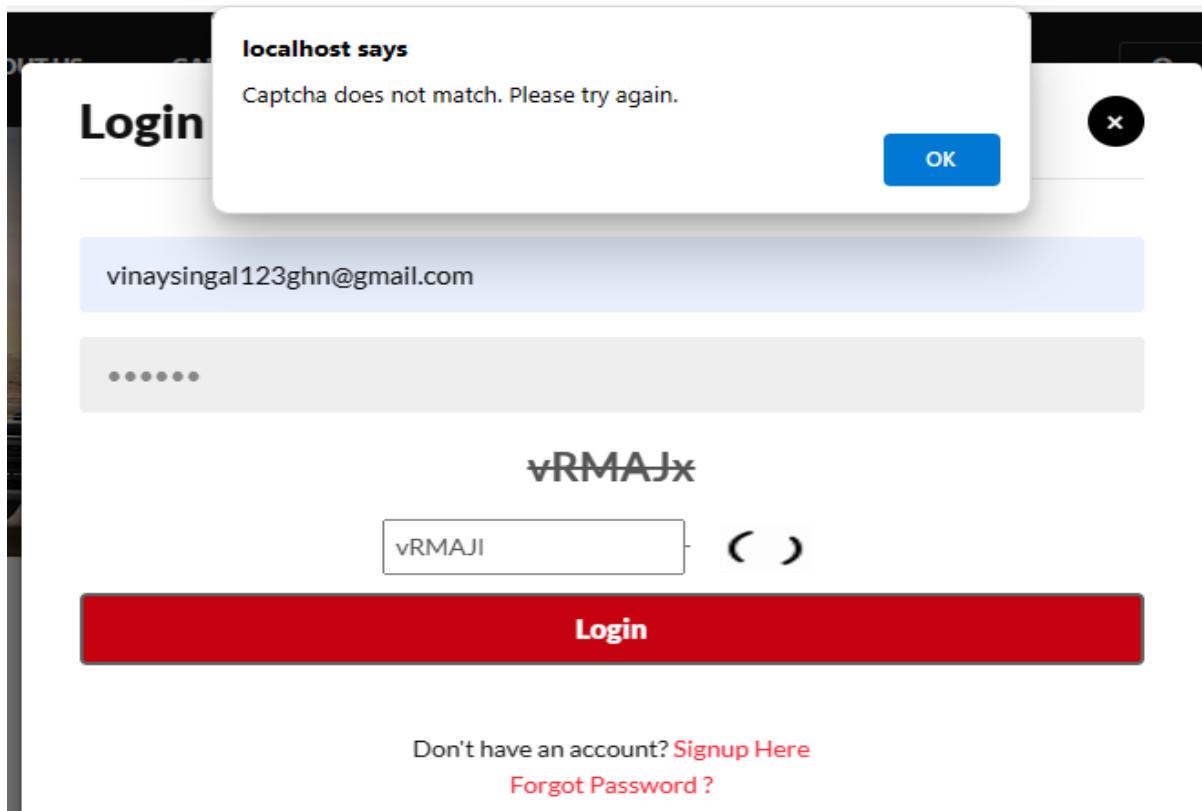


Fig 7.4.2.4

Test case 5

When user wants to login with correct username and password , the user redirect to homepage.

Login



Don't have an account? [Signup Here](#)

[Forgot Password ?](#)

Fig 7.4.2.5

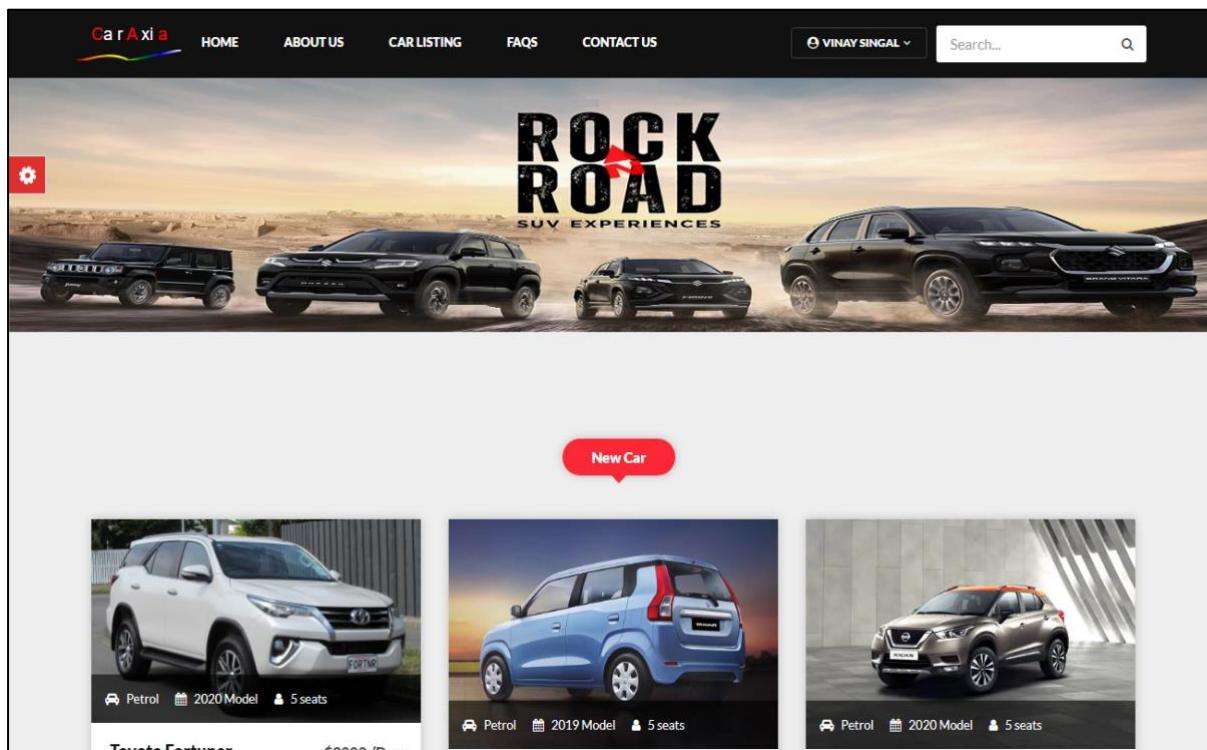


Fig 7.4.2.6

7.4.3 Admin login module

S No	Test Case	Input	Expected Result	Output	Remark
1	Blank all fields	Blank	Invalid details	Invalid details	Working
2	Wrong Username	Not Matches	Invalid details	Invalid details	Working
3	Wrong Password	Not Matches	Invalid details	Invalid details	Working
4	Correct username and password	Matches	Login Successful	Login Successful	Working

Test case 1

When user wants to login with blank username and password

The image shows a screenshot of a web-based login form titled "Admin | Sign in". The form has a blue header bar with the title. Below it is a white rectangular input area. Inside this area, there are two text input fields: one labeled "YOUR USERNAME" and another labeled "PASSWORD", both currently empty. Below these fields is a large blue "LOGIN" button. At the bottom left of the input area, there is a small link "Back to Home".

Fig 7.4.3.1

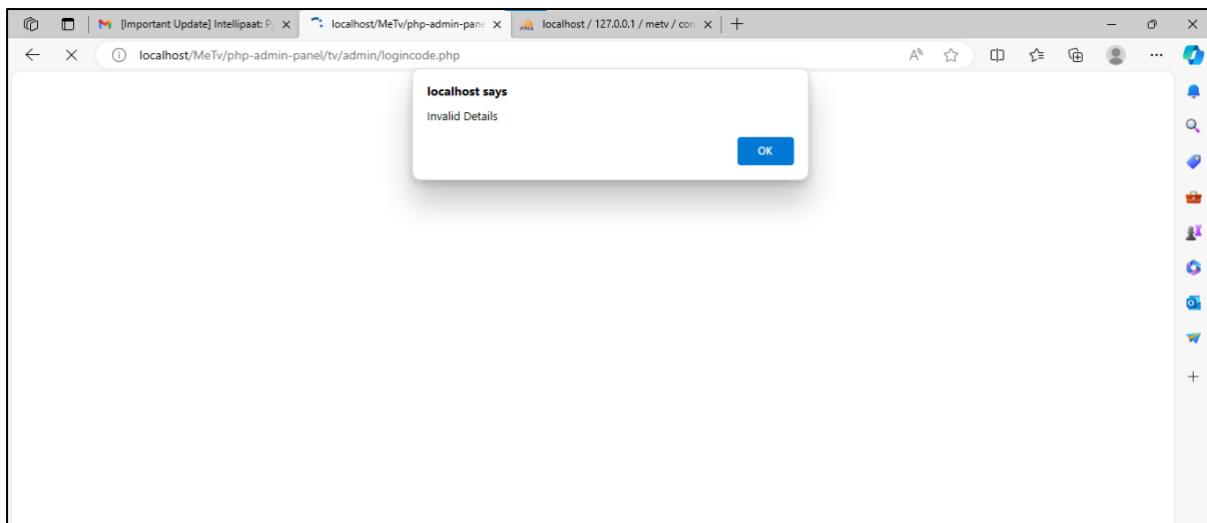


Fig 7.4.3.2

Test case 2

When user wants to login with wrong username

The image shows a login form titled 'Admin | Sign in' on a blue background. The form has two input fields: 'YOUR USERNAME' containing 'vinay' and 'PASSWORD' containing '*****'. Below the inputs is a large blue 'LOGIN' button. At the bottom left of the form is a 'Back to Home' link.

Fig 7.4.3.3

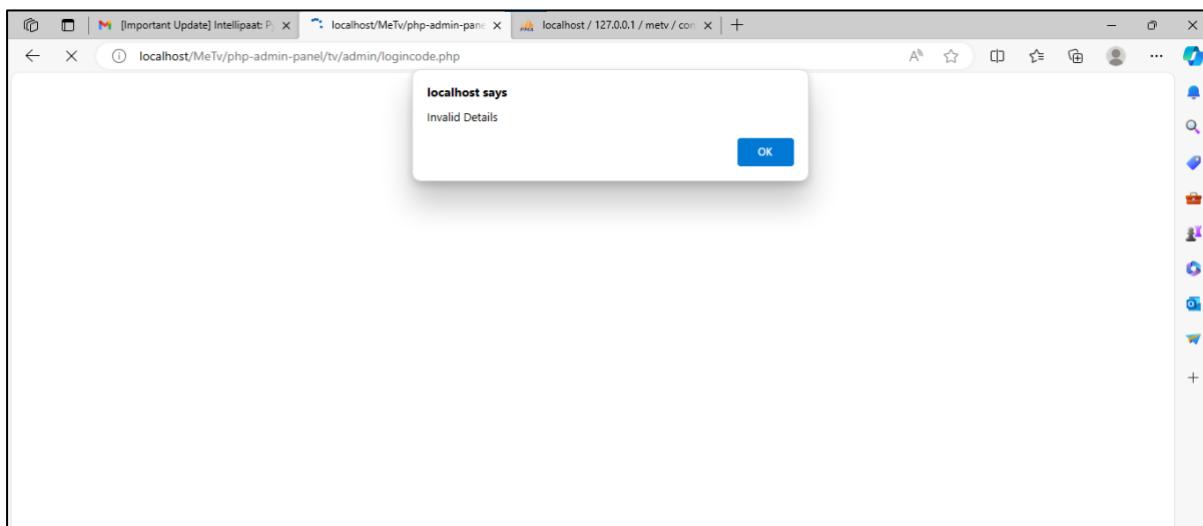


Fig 7.4.3.4

Test case 3

When user wants to login with wrong password

The screenshot shows an 'Admin | Sign in' page. The page has a blue header and footer. The main area contains a 'YOUR USERNAME' field with 'admin' typed in, and a 'PASSWORD' field with '.....' typed in. Below these fields is a large blue 'LOGIN' button. At the bottom left, there is a link 'Back to Home'.

Fig 7.4.3.5

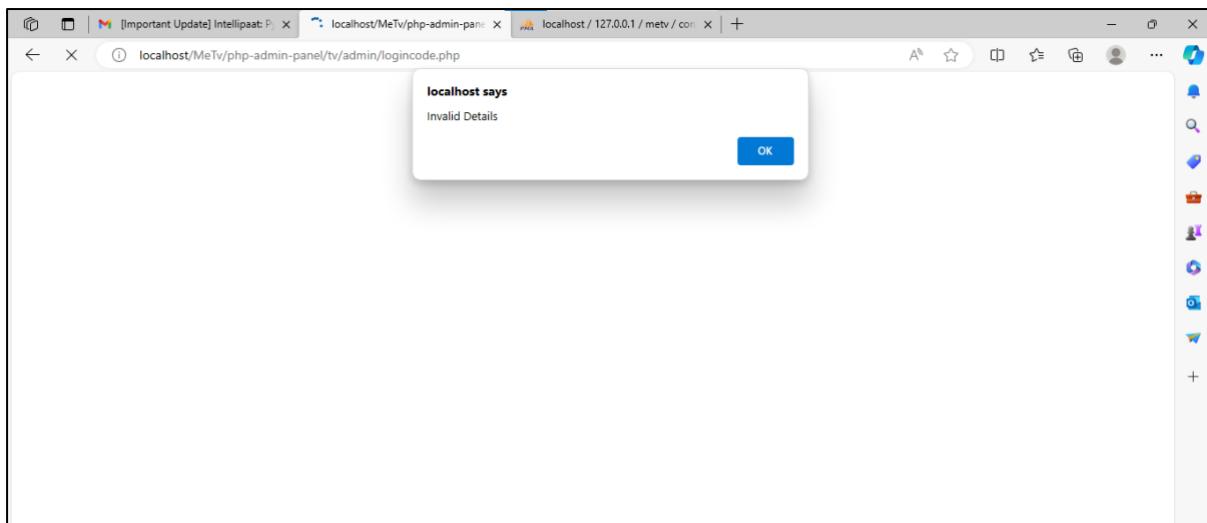


Fig 7.4.3.6

Test case 4

When user wants to login with correct username and password, the user is redirect to admin dashboard

The image shows a login form titled 'Admin | Sign in'. It has two input fields: 'YOUR USERNAME' containing 'admin' and 'PASSWORD' containing '*****'. Below the inputs is a large blue 'LOGIN' button. At the bottom left of the form is a link 'Back to Home'.

Fig 7.4.3.7

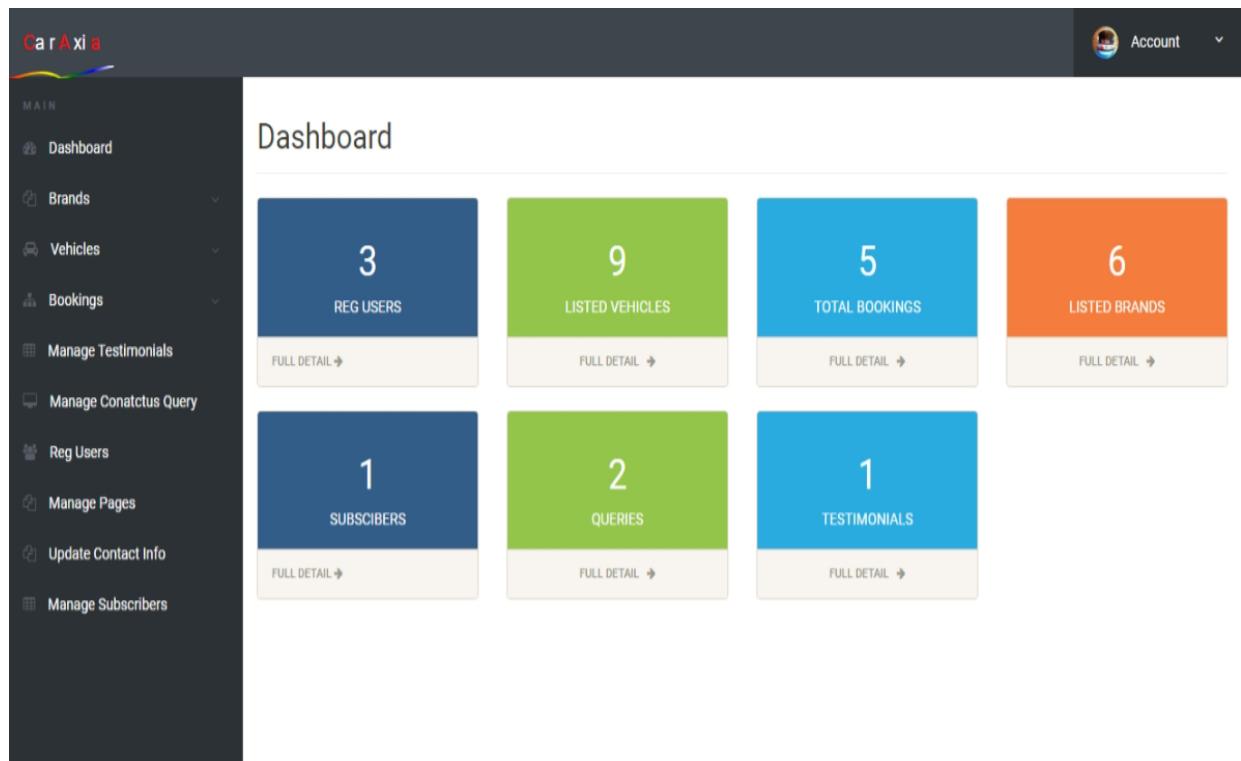


Fig 7.4.3.8

CHAPTER 8

SYSTEM IMPLEMENTATION

8.1 Install XAMPP in your system.

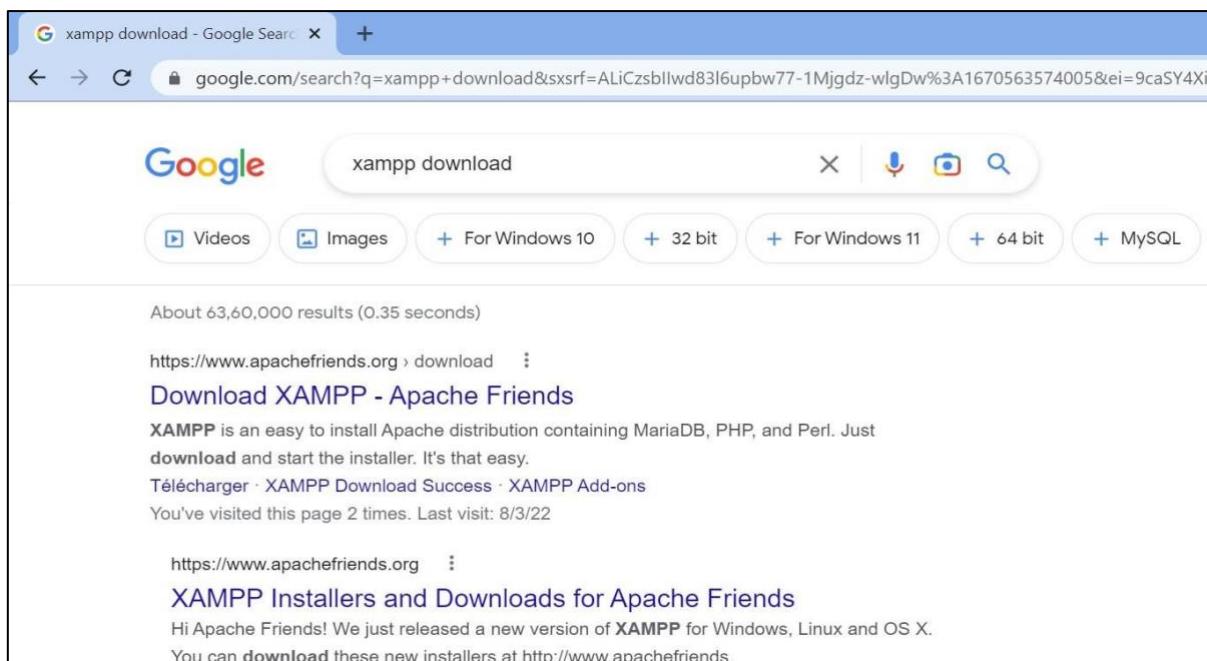


Fig 8.1 google chrome

8.2 Start XAMPP server on your device

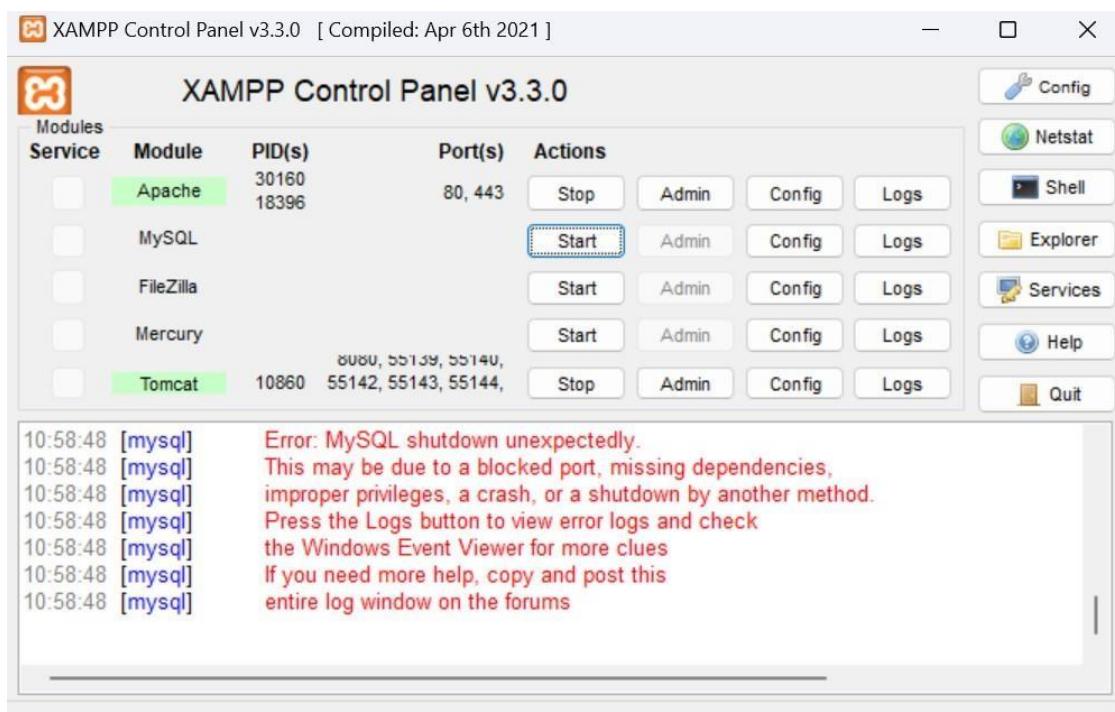


Fig 8.2 XAMPP Server

8.3 Start SQL in your XAMPP server.

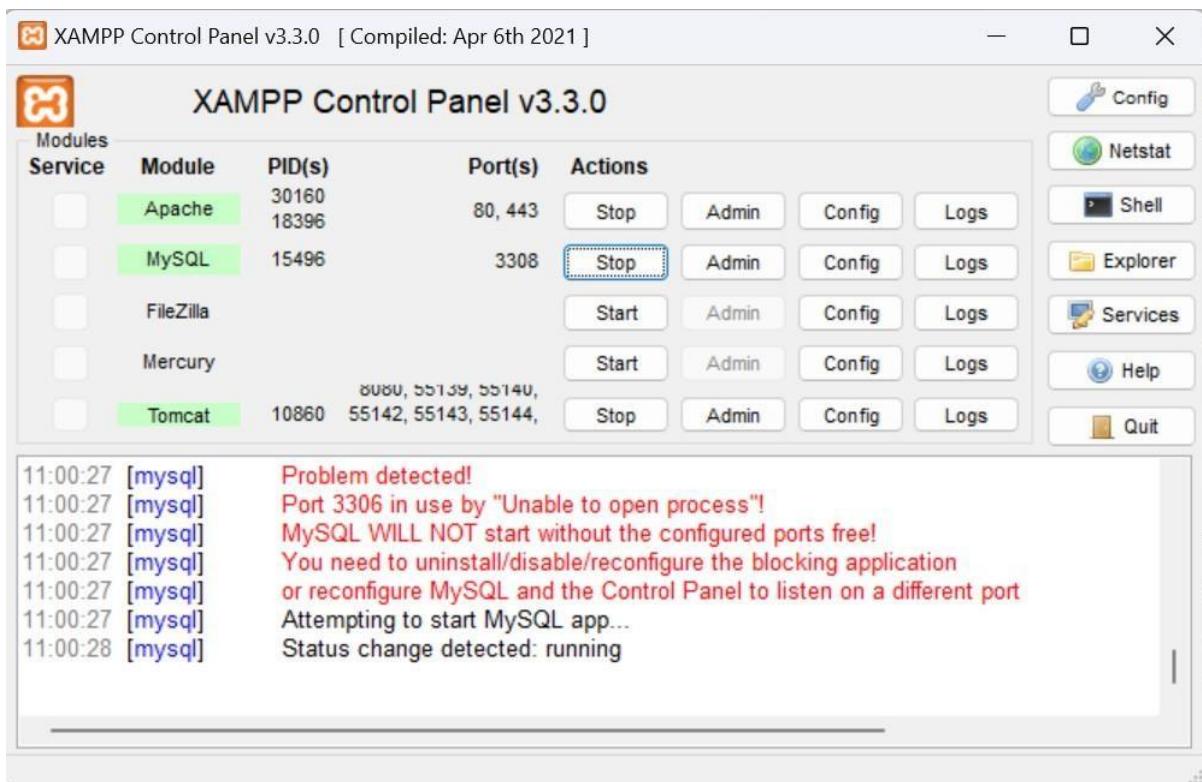


Fig 8.3 Starting SQL services

8.4 Open any browser in your system.



Fig 8.4 localhost/dashboard

8.5 Search for PhpMyAdmin in search bar.

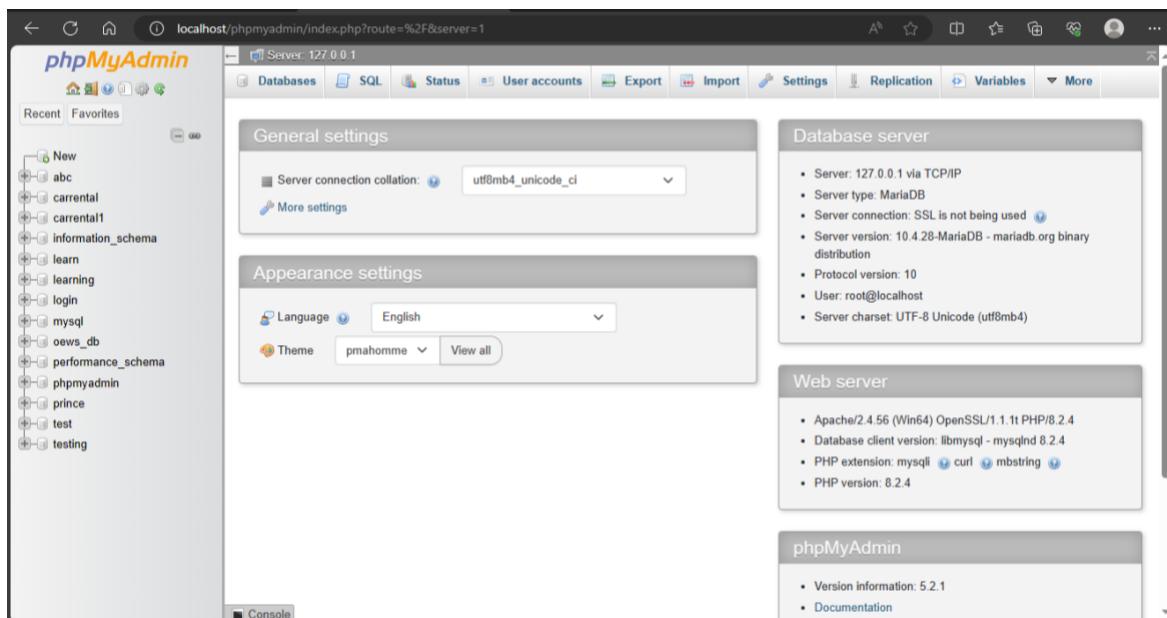


Fig 8.5 phpMyAdmin

8.6 Create a database “testing” and import the sql file in database

The screenshot shows the phpMyAdmin interface with the following details:

- Server:** 127.0.0.1
- Database:** currental
- Structure Tab:** Active
- Tables:** admin, tblbooking, tblbrands, tblcontactusinfo, tblcontactusquery, tbldpages, tblsubscribers, tbtestimonial, tbusers, tbvehicles
- Action Column:** Contains icons for Browse, Structure, Search, Insert, Empty, and Drop.
- Table Statistics:**
 - admin: 1 row, InnoDB, latin1_swedish_ci, 16.0 KiB
 - tblbooking: 5 rows, InnoDB, latin1_swedish_ci, 16.0 KiB
 - tblbrands: 6 rows, InnoDB, latin1_swedish_ci, 16.0 KiB
 - tblcontactusinfo: 1 row, InnoDB, latin1_swedish_ci, 16.0 KiB
 - tblcontactusquery: 2 rows, InnoDB, latin1_swedish_ci, 16.0 KiB
 - tbldpages: 4 rows, MyISAM, latin1_swedish_ci, 209.0 KiB
 - tblsubscribers: 1 row, InnoDB, latin1_swedish_ci, 16.0 KiB
 - tbtestimonial: 1 row, InnoDB, latin1_swedish_ci, 16.0 KiB
 - tbusers: 3 rows, InnoDB, latin1_swedish_ci, 32.0 KiB
 - tbvehicles: 9 rows, InnoDB, latin1_swedish_ci, 16.0 KiB
- Total:** 33 rows, InnoDB, utf8mb4_general_ci, 369.0 KiB, 0 B overhead.
- Console:** Bottom panel with a text input field: "Press Ctrl+Enter to execute query".
- Navigation:** Left sidebar with "Recent" and "Favorites" sections, and a tree view of databases and tables.

Fig 8.6 phpMyAdmin

Note: Place your project folder in XAMPP/htdocs folder.

CHAPTER 9

RESULT AND DISCUSSION

RESULT AND DISCUSSION

Advantages:

- Restricted Access: The information is accessible only to authorized administrators, ensuring data security and privacy.
- User Accessibility: Any admin user can access the information, providing flexibility in system usage.
- Privacy Protection: Personal information remains hidden from administrators, maintaining user confidentiality.
- Enhanced Security: Passwords are encrypted, minimizing the risk of unauthorized access or hacking.
- User-Friendly Interface: The system features an intuitive interface, facilitating ease of use for all users.
- Information Accessibility: Data is easily retrievable and can be selectively printed as needed.
- Efficient Updates: Records can be updated effortlessly, streamlining data management processes.
- Structured Database: Different database tables are utilized for storing user and product information, enhancing organization and retrieval efficiency.

Limitations:

- Technical Knowledge Required: Basic internet and browser knowledge are necessary for system operation.
- Restricted Access: Access is limited to authorized users only, potentially hindering collaboration among team members.
- Single Module Access: Users cannot access multiple modules simultaneously, limiting multitasking capabilities.
- Admin Access Requirement: Access to the dashboard and certain modules requires login as an admin, potentially causing delays in accessing essential functionalities.

CHAPTER 10

CONCLUSION

CONCLUSION

In conclusion, the development journey of our major project, CarAxia, has been an enlightening experience, allowing us to apply our academic knowledge practically. Throughout the creation of CarAxia, we've garnered valuable insights and honed skills crucial for our future roles as computer engineers.

Key lessons learned from our work on CarAxia include:

- Effective Planning: We've understood the pivotal role of meticulous planning in project success. From defining objectives to outlining milestones, a well-crafted plan has guided our project effectively.
- Structured Execution: Adhering to an organized workflow has been vital in managing tasks efficiently and ensuring project progress. By following structured processes and fostering clear communication, we've maintained project momentum.
- Collaborative Spirit: Collaboration has been central to CarAxia's success. Leveraging each team member's strengths has fostered a supportive environment that encourages creativity and problem-solving.
- Continuous Improvement: Engaging with mentors, seeking feedback, and embracing a growth mindset have been fundamental to our project journey. Learning from experiences has enabled us to adapt to challenges and refine our project iteratively.

As we wrap up our work on CarAxia, we anticipate the opportunities and obstacles that await us. The skills and experiences gained from this project will undoubtedly shape our future endeavors and contribute to our growth as computer engineers.

CHAPTER 11

FUTURE SCOPE

FUTURE SCOPE

CarAxia has been meticulously crafted with adaptability and flexibility at its core, ensuring seamless adjustments and updates to meet evolving requirements without compromising security. Robust security measures are seamlessly integrated to fortify the system against potential vulnerabilities arising from modifications. Furthermore, the system is architected to effortlessly accommodate multiple users accessing it simultaneously, ensuring scalability and optimal performance.

Moreover, CarAxia presents vast potential for future advancements, offering the flexibility to incorporate new features and enhancements to stay abreast of emerging trends. The coding structure is highly customizable, facilitating tailored solutions to meet specific user preferences and industry demands. With a well-organized and scalable codebase, CarAxia provides a robust foundation for future iterations and the integration of additional functionalities. Additionally, the reporting system can be customized to include supplementary options as required, delivering comprehensive insights and analytics tailored to user needs.

In essence, CarAxia stands poised for continuous evolution and growth, ensuring it remains a frontrunner in the realm of car rental management systems and continues to address the dynamic needs of its users and the industry at large..

CHAPTER 12
BIBLIOGRAPHY AND REFERENCES

Site for references:

- <http://www.youtube.com>
- <https://www.javatpoint.com>
- <https://www.tutorialspoint.com>