**A PROJECT ON**

“**CAR RENTAL SYSTEM”**

SUBMITTED IN

PARTIAL FULFILLMENT OF THE REQUIREMENT

FOR THE COURSE OF DIPLOMA IN ADVANCED COMPUTING FROM CDAC



CDAC ACTS NEW DELHI , CDAC DELHI

**SUBMITTED BY:**

1) Sharad Dilip Kute (220910120045)

2) Morane Rakesh Pandharinath (220910120033)

3) Dhage Manisha Sahebrao (220910120017)

4)Vikas Pandey (220910120056)

**UNDER THE GUIDANCE OF:**

Mrs. Arpit jain

Faculty Member

Cdac Acts New Delhi

**CERTIFICATE**

This is to certify that the project work under the title ‘Car Rental System’ is done by Sharad D.Kute, Vikas , Rakesh and Manisha in partial fulfillment of the requirement for award of Diploma in Advanced Computing Course.

**Mr. Arpit jain Mr. Pankaj Mehto**

**Project Guide Course Co-Coordinator**

Date: 15.03.2023

**ACKNOWLEDGEMENT**

A project usually falls short of its expectation unless aided and guided by the right persons at the right time. We avail this opportunity to express our deep sense of gratitude towards .

We are deeply indebted and grateful to them for their guidance, encouragement and deep concern for our project. Without their critical evaluation and suggestions at every stage of the project, this project could never have reached its present form.

Last but not the least we thank the entire faculty and the staff members of Cdac Acts New Delhi for their support.

1) Sharad Dilip Kute (220910120045)

2) Morane Rakesh Pandharinath (220910120033)

3) Dhage Manisha Sahebrao (220910120017)

4)Vikas Pandey (220910120056)

**ABSTRACT**

The Car Rental System is being developed for customers so that they can book their cars from any part of the world. This application takes information from the customers through filling their details. A customer being registered in the website has the facility to book a Car which he requires. The proposed system is completely integrated online systems. It automates manual procedure in an effective and efficient way. This automated system facilitates customer and provides to fill up the details according to their requirements. It includes type of car they are trying to hire and location. The purpose of this system is to develop a web site for the people who can book their Car along with requirements from any part of the world. Car rental system provide car to User in their location on short time.

This software car Rental System has a very user friendly interface. Thus the users will feel very easy to work on it. By using this system admin can manage customer confirm and cancel booking request, customer Testimonials, customer issues. The car information can be added to the system. Or existed car information can be edited or deleted too by Administrator. There is no delay in the availability of any car information, whenever needed, car information can be Captured very quickly and easily.

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Title** | **Pg. No.** |
| 1 | Introduction |  |
| 1.1 | Objectives |  |
| 2 | Need Of Project |  |
| 3 | Requirement |  |
| 3.1 | Functional Requirement |  |
| 3.2 | Non Functional Requirement |  |
| 4 | Database Design |  |
| 5 | Coding Standard Implemented |  |
| 6 | Project Management Related Statistics |  |
| 7 | Appendix A |  |
| 8 | Appendix B | 24 |
| 9 | References | 39 |

**1. INTRODUCTION**

This project is designed so as to be used by Car Rental Company specializing in renting cars to customers. It is an online system through which customers can view available cars, register, view profile and book car. Here, User has to Login to book a car. The user can search for cars easily and book. For bookings, the user has to provide information such as Booking Dates and Text Message. All car details are provided and it also includes Car’s features and Overview.

This Car Rental System project enables renting of cars through an online system. It helps the users to search for available cars view profile and book the cars for the time period. It has a user-friendly interface which helps the user to check for cars and rent them for the period specified. They could also make payment online. The rental cars shall be categorized into economy, premium etc. Based on the type of car required by the customer, the user shall be able to make bookings. The use of internet technology has made it easy for the customers to rent a car any time. This Car Rental System makes the bookings easy. It saves time and labor. The tool shall ask the user for information such as the date and time of journey, type of car etc. Also, it will need an identification number. Using these details, the tool shall help the customer to book a car for the journey

Admin can Add/Manage car brand, manage cars, bookings, testimonial, pages and many more. It’s easy to operate and understand by users. This site makes customers easy for car rental. The design is pretty simple and the user won’t find it difficult to understand, use and navigate.

**1.1. Objective:**

* To reduce the effort of booking a car in a conventional procedure.
* To ease the search process of a customer who is in need of a car.
* To provide services to the customers in order to achieve the best customer satisfaction.
* Users who cannot drive or do not have driver license can rent cars with our drivers.
* Admin can manage the car catalog by adding or removing cars based on their availability.
* Reduced errors with rental data effectively stored in database.
* Entire network works on one single platform, seamless flow of data.
* Standardization of processes and services ensure clients receive the correct rates.

**1.2 Why to use our website?**

* We have various types of available cars such electric, hybrid, semi and auto.
* To make safe and secure journey.
* Green cars available.
* There are some special benefits of Electric car such as free car parking, car wash and low maintenance.

**1.3 Unique:**

* Insurance cars
* Safe cars as we do maintenance after every ride.
* User can try new IoT technology.
* Fuel Efficient.

**2. NEED OF PROJECT**

* A car rental is a vehicle that may be rented for a price and utilized for a specific length of time.
* Getting a rental automobile makes it easier for people to travel around when they don't have access to their own vehicle or don't own one at all.
* A person who needs transportation must call a rental car company and sign a contract.
* This method improves client retention while also making car and Employee management more straightforward.
* To ease the search process of a customer who is in need of a car.

**Specifications Of System:**

1. **Admin**: Admin is the super user of the website who can manage everything on the website except booking.
2. **Users**: Anyone can register through the registration page. After a successful registration user can log in with valid username and password.

**3. REQUIREMENTS**

**3.1 FUNCTIONAL REQUIREMENTS**

Requirement analysis is a software engineering approach that consists of a series of activities that establish the demands or conditions that must be satisfied for a new or updated product while taking into account the potential for competing requirements from different users. Functional requirements are those that are used to demonstrate the system's internal functioning nature, as well as the system's description and explanation of each subsystem. It comprises the task that the system should accomplish, the processes involved, the data that the system should contain, and the user interfaces.

These are statements of services the system should provide, how the system should react to particular inputs, and how the system should behave in particular situations. It specifies the application functionality that the developers must build into the product to enable users to accomplish their tasks. The system must allow the customer to register for booking a car. The system shall allow the customer to view detail description of particular car. The system must notify on selection of unavailable cars while booking. The system shall allow the employee to update feedback information. The system shall allow the employee to view bookings made by customers. The system shall presents information on protection products and their daily costs, and requests the customer to accept or decline regulation terms during booking.

**3.1.1 User Requirement:**

The user should be able to search for cars without signing up but while booking he/she should compulsorily need to log in. The user can sign-up if he/she doesn’t have an account and then login. For payment user can pay through UPI, Card and Cash. Also for security only user/customer can access functionality of the user/customer, even if someone tries to log in with valid credentials of different role he/she should not be able to login and an error should be shown.

Following are the functions available to the users:

* Login
* Register
* Select Location
* View Catalog
* Driver Option
* Make bookings
* Make Payment
* Confirm Payment
* Return Form
* Query
* Feedback
* **User Account**

The customer, who will hence forth be called the ‘user’, will be presented with 3 choices by the rental system, as the first step in the interaction between them. A user can choose one of these and his choice would be governed by whether he is a registered user and whether he wants to check the availability of cars. The terms ‘registered user’ are described below.

A user who has rent the cars earlier would have been given a user id and a password. This ‘personal information’ would be henceforth referred to as ‘profile’. Such a user with a profile in DB-user shall be called a ‘registered user’. A registered user will be able to check the availability of cars as well as book a car by logging into the system.

A new user, on the other hand, would either have to

1. Register himself with the system by providing personal information.
2. Log into the system.

In case of ‘a’, the new user becomes a registered user.

‘Availability of Cars’ always refers to viewing the booking schedule for given days, the price of cars and any discount offers. The system shall present the user with an option to exit from the system at any time during the following processes.

* **Registration and creation of user profile**

The system shall require a user to register, in order to book a car. It will ask the user for the following information:

Username, full name, dob, password, adhar number, license number, email, address, gender etc.

* **Making Booking of cars:**

The user should be able to search for cars based on availability and should show cars which are available or not booked on the period between the pickup and drop location. After booking for those dates the cars should be booked for only those particular days and free or others to book on other dates. User can book any number of cars.

* **Booking details:**

The user should be able to get the details of all the cars he/she has booked and for cars he is searching but not yet booked. The user should also be able to get total cost of booking before booking is done.

* **Canceling of cars:**

The user should also have option to cancel booking.

If user cancels booking before 24 hrs of pickup time immediately then they will get 80% refund.

If users cancel before 8hrs of pickup time they will get 50% refund.

If they cancel on the day of pickup then no refund is available.

* **Add/Update /remove cars and car details:**

The admin adds cars. The admin can edit car details and images. Admin can update cars availability and check their requirement as per need at any time. The car admin can also remove the cars. The car admin should also receive all booking details of his/her own cars.

* **View Booking History**

The system shall allow a user to view all information about his previous bookings. It accesses User Booking table and retrieves the details of the trip and presents them to the user in a tabular format.

**3.1.2 Admin Requirements:**

The Car admin should need to first register and then login. After that, admin can add car, driver, location selected. The admin can also manage bookings, maintain car Info, Query, feedback details. Also the admin can get all the bookings. The car admin can also delete cars, drivers, users, and locations. Also for security only car admin can access functionality of the car rental system. Even if someone tries to log in with valid credentials of different role he/she should not be able to login and an error should be shown.

Following are the functions available to the Hotel Admin:

1. Login
2. Maintain Car Info
3. Get all customers
4. Get all Booking Details
5. Get all Cars by Availability
6. Maintain Location Info
7. Maintain Driver Details
8. Return form
9. Query /Feedback

3.2 **NON FUNCTIONAL REQUIREMENTS**

It describes system elements that are concerned with how the system fulfills functional requirements.

They are as follows:

* **Security:** Only authorized corporate workers may get access to the firm's secured page on the systems, and only users with proper passwords and usernames can log in to see the users page.
* **Performance** **and Response Time**: The system should have a high-performance rate while executing user input and should be able to offer feedback or a response in a short amount of time.
* **Error handling**: Errors should be avoided as much as possible, and a suitable error message should be supplied to help the user through the recovery process. The importance of validating user input cannot be overstated. In addition, the time it takes to recover from a mistake should be between 15 and 20 seconds.
* **Availability** :

This system must be accessible at all times, 24 hours a day, seven days a week. In the event of a catastrophic system failure, the system should be backup and running within 1 to 2 business days, ensuring that the business process is not disrupted.

* **Ease of use**: Given the consumers' level of understanding, a basic yet high-quality user interface should be created to make it simple to comprehend and need minimal training.

**4. DATABASE DESIGN**

**Table 1: Account\_details**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Allow Null** | **Key/Constraints** | **length** |
| Adhar\_no | Bigint | No | 1 | 8 |
| b\_no | Varchar | Yes | 1 | 255 |
| City | Varchar | Yes | 1 | 255 |
| Pincode | Int | No | 1 | 4 |
| State | varchar | Yes | 1 | 255 |
| Street | Varchar | Yes | 1 | 255 |
| Adhar\_name | Varchar | No | 1 | 255 |
| Dob | Date | Yes | 0 | 6 |
| Email | Varchar | No | 2 | 255 |
| Gender | Bit | No | 1 | 1 |
| Licsence\_no | Varchar | Yes | 1 | 255 |
| Mob\_no | Bigint | No | 2 | 8 |
| Password | Varchar | No | 2 | 255 |
| role | Varchar | Yes | 1 | 255 |
| username | varchar | no | 2 | 255 |

**Table2: Car\_details**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Allow Null** | **Key/Constraints** | **length** |
| Car\_model | Int | No | 1 | 4 |
| Car\_average | Double | No | 0 | 8 |
| Car\_capacity | Int | No | 0 | 4 |
| Car\_company | Varchar | Yes | 0 | 255 |
| Car\_fuel | Varchar | Yes | 0 | 255 |
| Car\_gear\_type | Varchar | Yes | 0 | 255 |
| Car\_manufact\_year | date | Yes | 0 | 6 |
| Car\_rent\_perday | Double | No | 0 | 8 |
| Car\_type | Varchar | Yes | 0 | 255 |
| Insurance\_no | Varchar | yes | 0 | 255 |
| Insurance\_period | date | yes | 0 | 6 |

**Table 3: Location\_details**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Allow Null** | **Key/Constraints** | **length** |
| Pincode | Int | No | 1 | 4 |
| B\_no | Varchar | Yes | 1 | 255 |
| City | Varchar | Yes | 1 | 255 |
| State | Varchar | Yes | 1 | 255 |
| Street | varchar | yes | 1 | 255 |

**Table 4: Customer\_pickup**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Allow Null** | **Key/Constraints** | **length** |
| O\_id | Bigint | No | 1 | 4 |
| O\_date | Date | Yes | 1 | 6 |
| O\_time | Time | Yes | 1 | 6 |
| R\_date | Date | Yes | 1 | 6 |
| R\_time | Time | Yes | 1 | 6 |
| Pincode | int | yes | 1 | 4 |

**Table 5: Driver\_detail**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Allow Null** | **Key/Constraints** | **length** |
| License\_no | Varchar | No | 1 | 255 |
| B\_no | Varchar | Yes | 1 | 255 |
| Pincode | int | No | 1 | 4 |
| State | Varchar | Yes | 1 | 255 |
| Street | Varchar | Yes | 1 | 255 |
| Adhar\_name | Varchar | No | 1 | 255 |
| Adhar\_no | Bigint | No | 2 | 8 |
| Dob | Date | Yes | 0 | 6 |
| Email | Varchar | No | 1 | 255 |
| Gender | Bit | No | 1 | 1 |
| Mob\_no | Int | No | 2 | 8 |
| city | varchar | yes | 1 | 255 |

**Table 6: Order\_details**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Allow Null** | **Key/Constraints** | **length** |
| O\_id | Bigint | No | 1 | 8 |
| Car\_model | Int | Yes | 1 | 255 |
| License\_no | Varchar | Yes | 1 | 255 |
| Adhar\_no | Bigint | yes | 1 | 8 |

**Table 7: Payment\_details**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Allow Null** | **Key/Constraints** | **length** |
| O\_id | Bigint | No | 1 | 8 |
| Card\_cvv | Int | Yes | 1 | 4 |
| Card\_expiry | Date | Yes | 1 | 6 |
| Card\_name | Varchar | yes | 1 | 255 |
| Card\_no | Bigint | Yes | 1 | 8 |
| Debit\_credit | Bit | Yes | 1 | 1 |
| cash | Bit | Yes | 1 | 1 |
| razorpay | Bit | Yes | 1 | 1 |
| Total\_amount | Double | No | 1 | 8 |
| upi | Bit | Yes | 1 | 1 |
| Upi\_id | Varchar | Yes | 1 | 255 |
| Upi\_name | varchar | yes | 1 | 255 |

**Table 8: After\_return\_form**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Allow Null** | **Key/Constraints** | **length** |
| O\_id | Bigint | No | 1 | 8 |
| Comments | Varchar | Yes | 0 | 255 |
| Condition\_after\_return | varchar | yes | 1 | 255 |

**Table 9: Feedback\_form**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Allow Null** | **Key/Constraints** | **length** |
| O\_id | Bigint | No | 1 | 8 |
| Cleaning\_points | Int | No | 0 | 4 |
| Comments | Varchar | Yes | 0 | 255 |
| Service\_points | int | no | 0 | 4 |

**Table 10: Query\_details**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Allow Null** | **Key/Constraints** | **length** |
| Q\_id | Bigint | No | 1 | 8 |
| Query\_comments | Varchar | No | 1 | 255 |
| Status | Bit | No | 1 | 1 |
| O\_id | bigint | yes | 1 | 8 |

**5. CODING STANDARDS IMPLEMENTED**

**Naming and Capitalization:**

Below summarizes the naming recommendations for identifiers in Pascal casing is used mainly (i.e. capitalize first letter of each word) with camel casing (capitalize each word except for the first one) being used in certain circumstances.

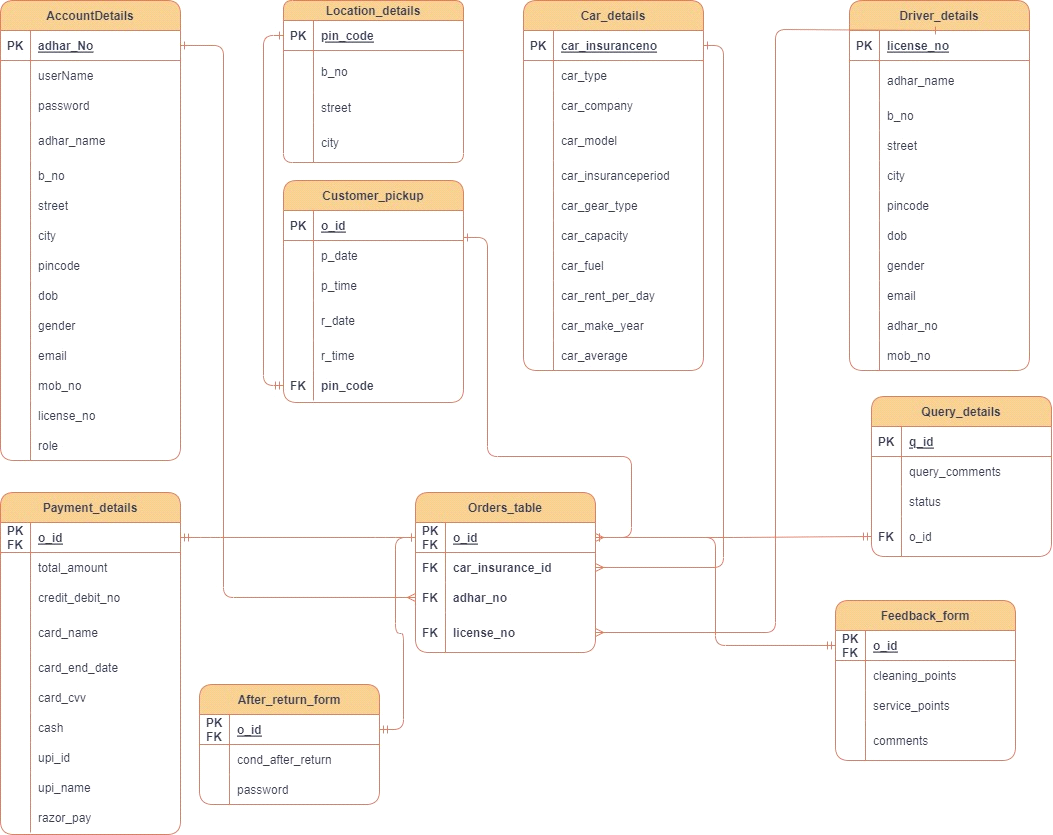
|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | **Case** | **Examples** | **Additional Notes** |
| Class | Pascal | Account, OrderDetails,  LocationDetails | Class names should be based on "objects" or "real things" and should generally be **nouns**. No ‘\_’ signs allowed. Do not use type prefixes like ‘C’ for class. |
| Method | Camel | getOrderDetails, updatePayment | Methods should use **verbs** or verb phrases. |
| Parameter | Camel | adharName,  licenseNumber | Use descriptive parameter names. Parameter names should be descriptive enough that the name of the parameter and its type can be used to determine its meaning in most scenarios. |
| Interface | Pascal with "I" prefix | JpaRepository,  AccountDetails | Do not use the ‘\_’ sign |
| Property | Pascal | Roles, bNo | Use a noun or noun phrase to name properties. |
| Associated private member variable | \_camelCase | \_adhar\_Name, \_mob\_No | Use underscore camel casing for the private member variables |
| Exception Class | Pascal with "Exception" suffix | RuntimeException, |  |

**Comments:**

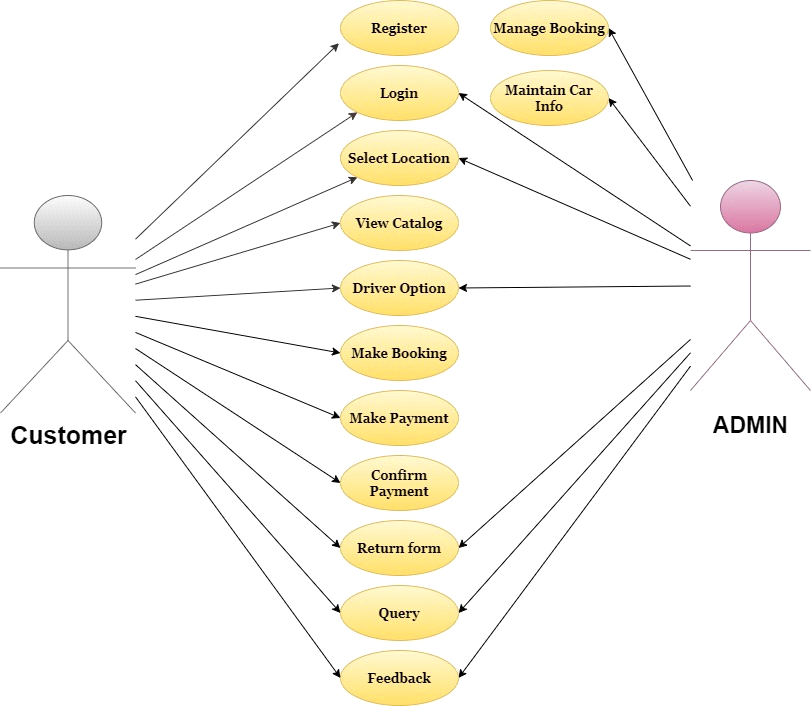
* Comment each type, each non-public type member, and each region declaration.
* Use end-line comments only on variable declaration lines.
* Separate comments from comment delimiters (apostrophe) or // with one space.
* Begin the comment text with an uppercase letter.
* End the comment with a period.
* Explain the code; do not repeat it.

**Appendix A**

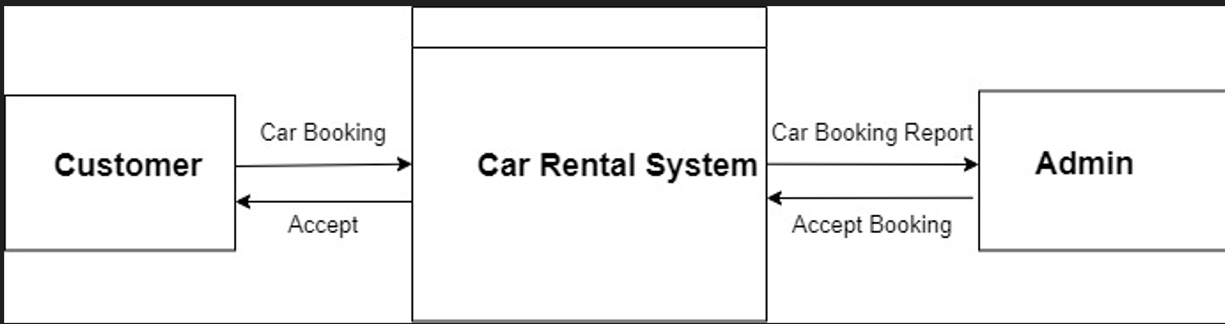
**1:Entity Relationship Diagram**



**2: Use case Diagram**



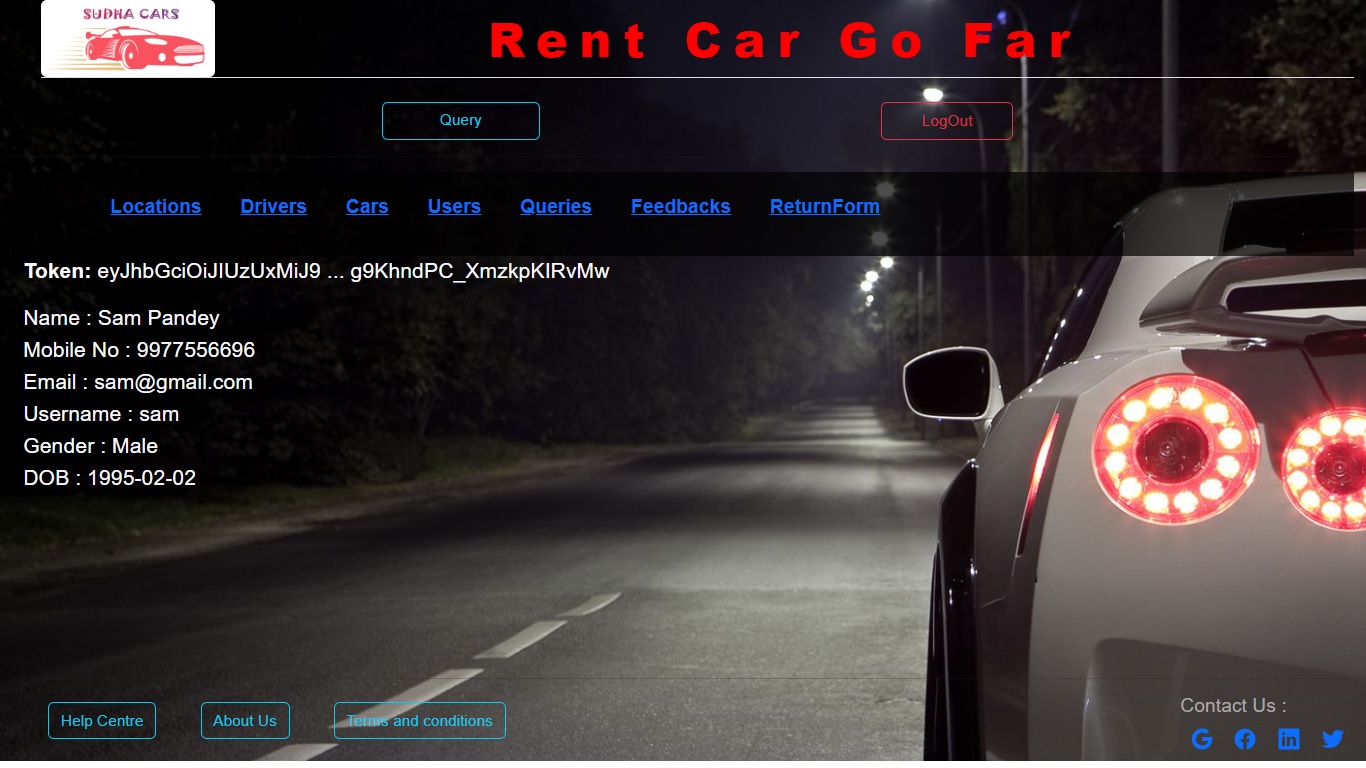
**3: Dataflow Diagram**

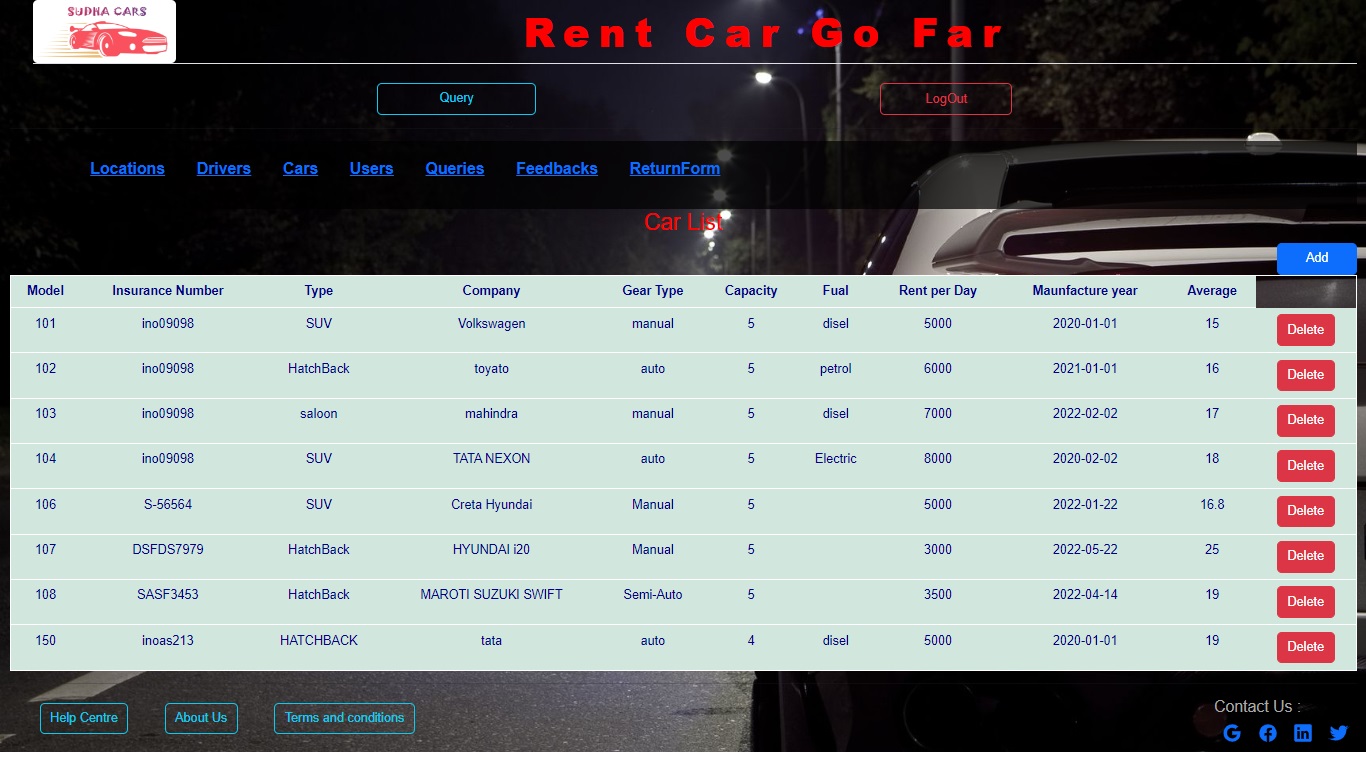


**Appendix B**

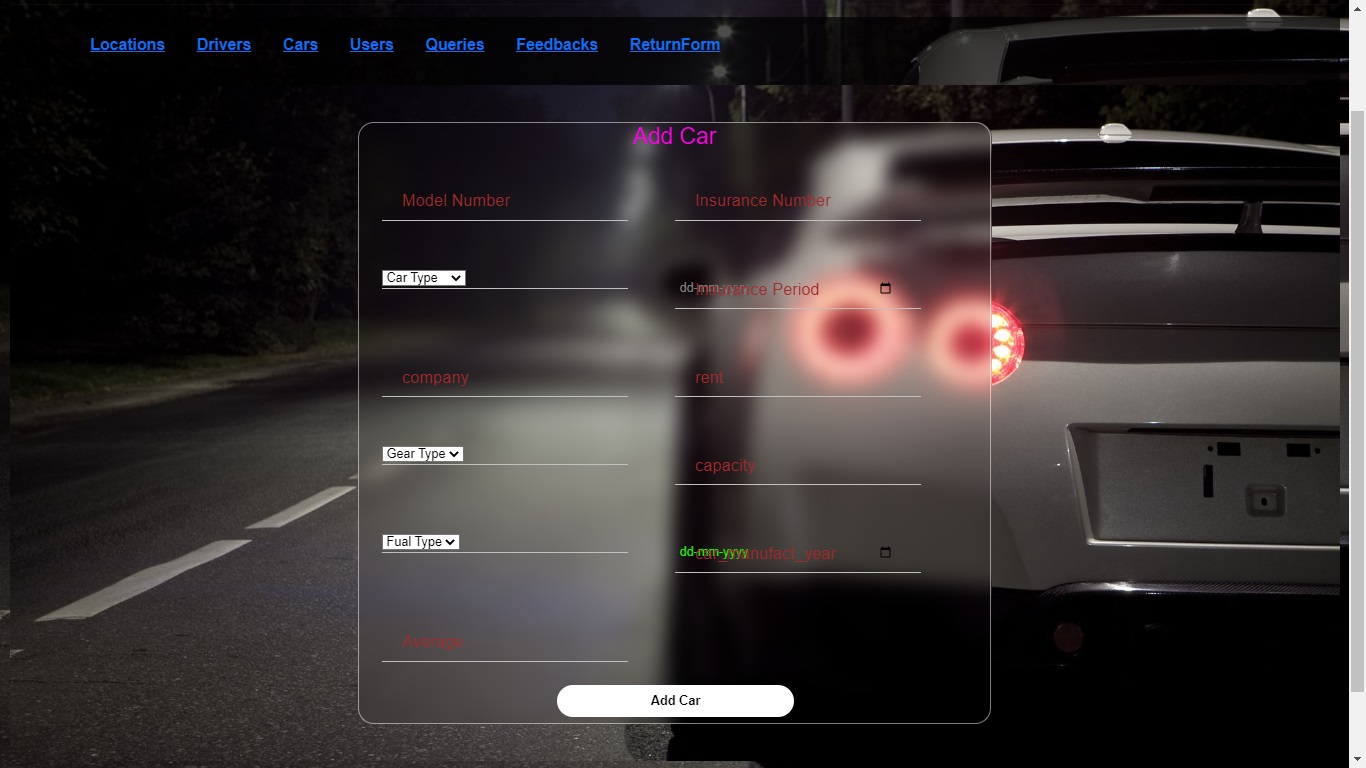
**Admin Pages:**

Profile:

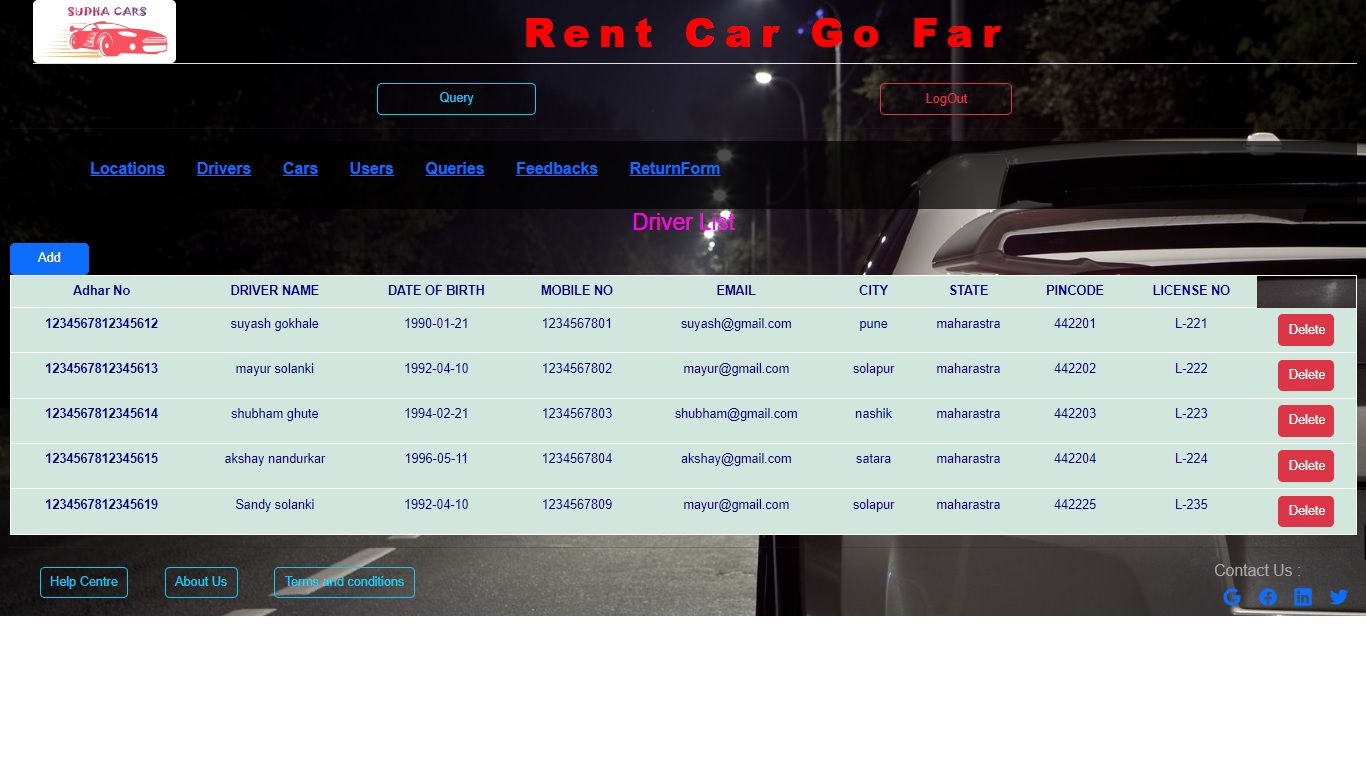
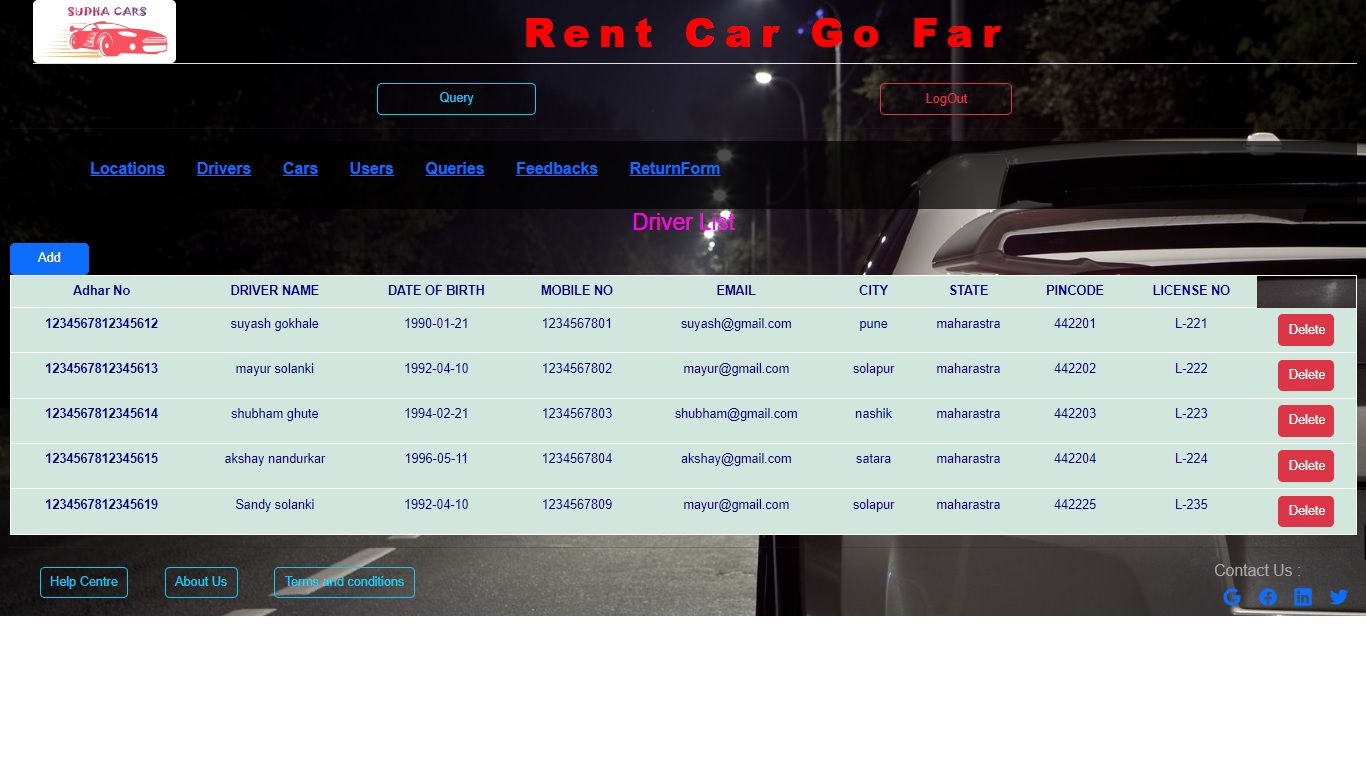


Car: 

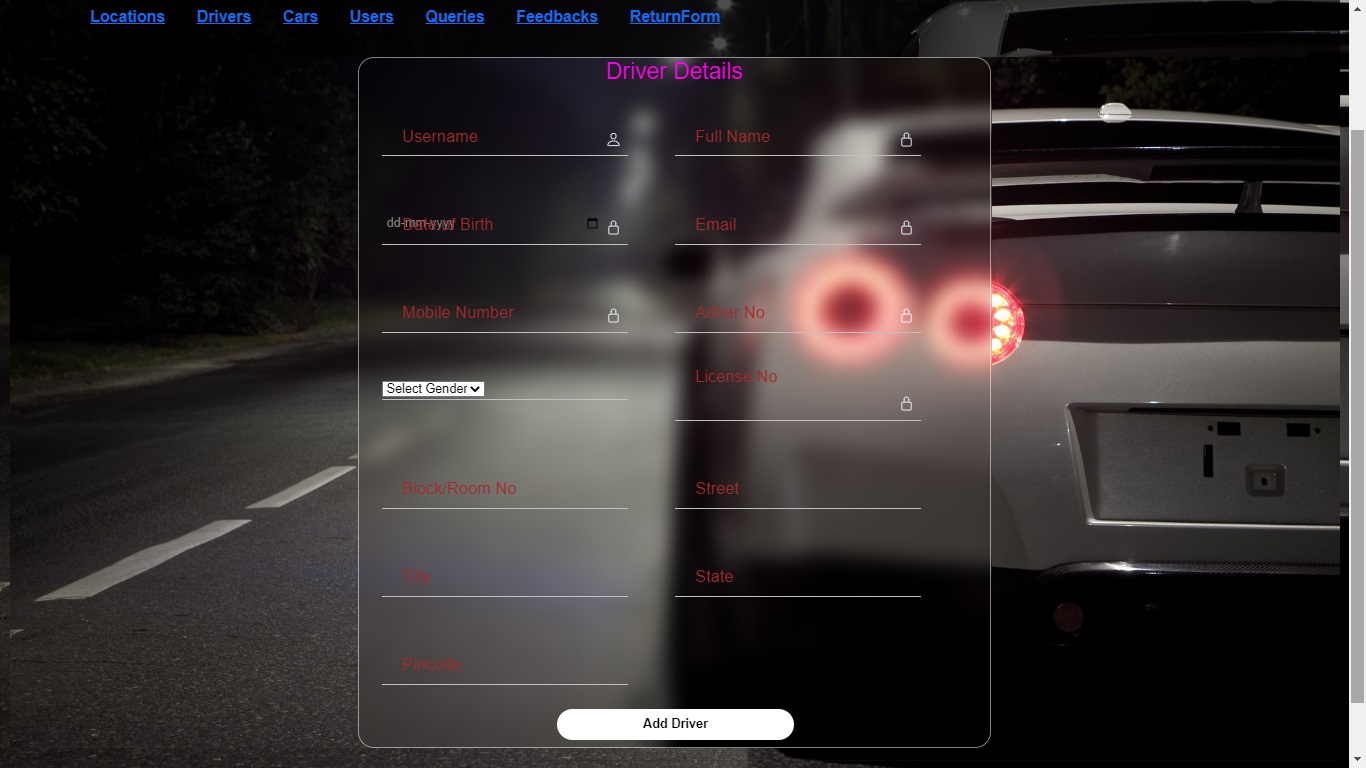
Car Add:



Driver List:



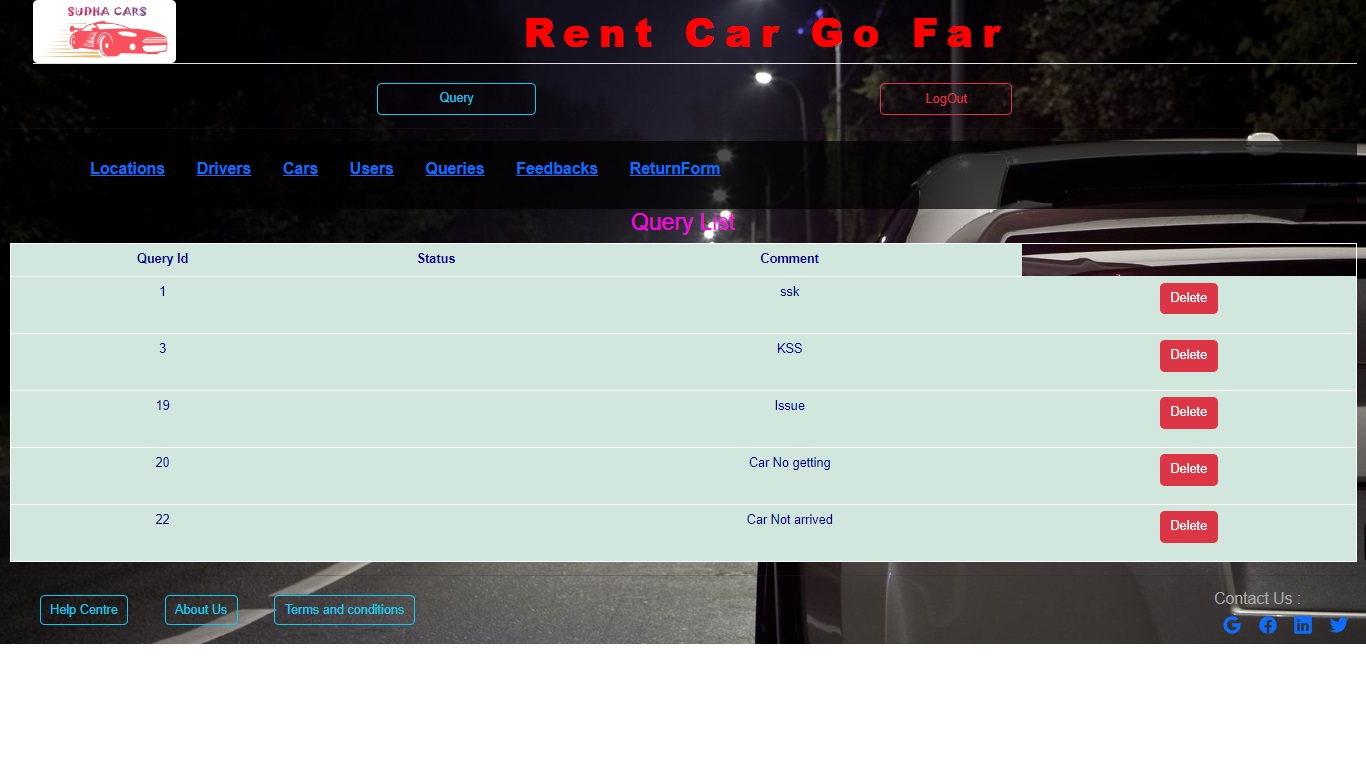
Driver Add:



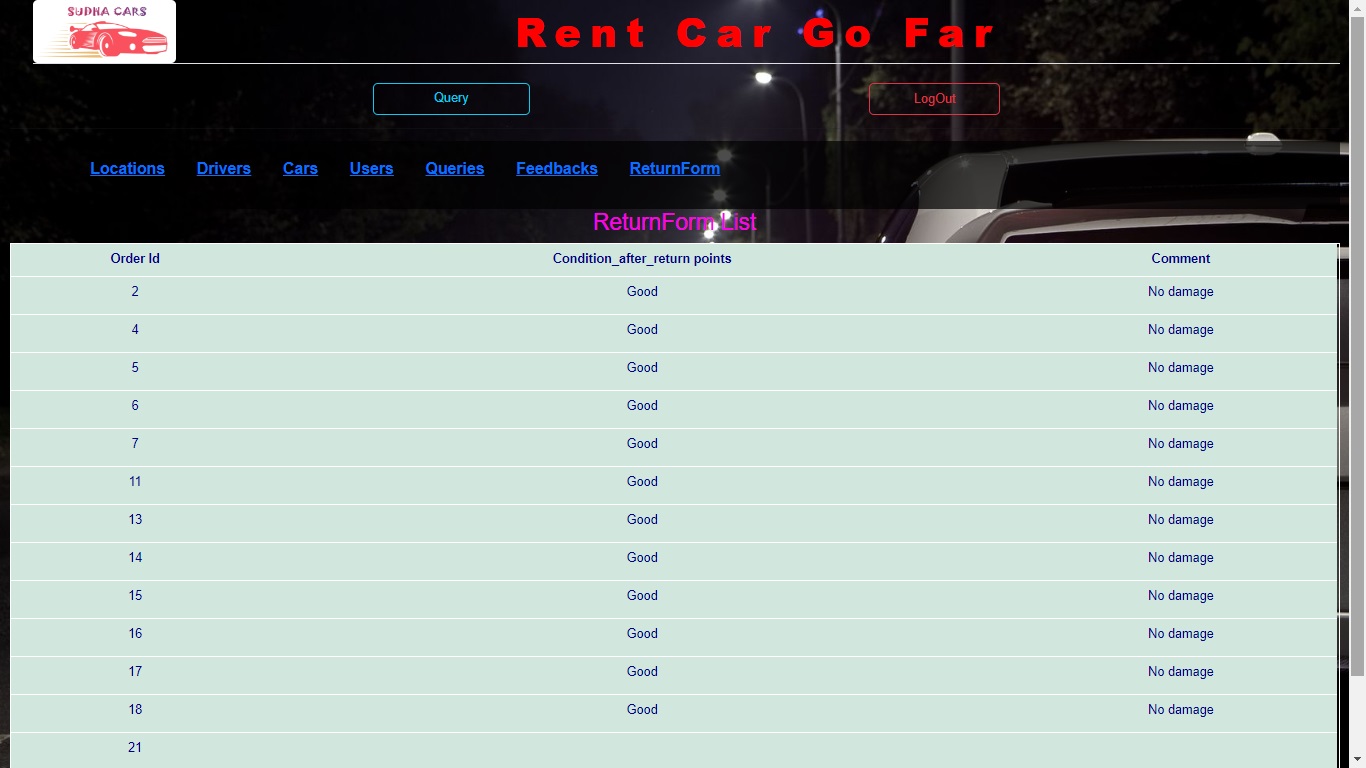
Feedback List:



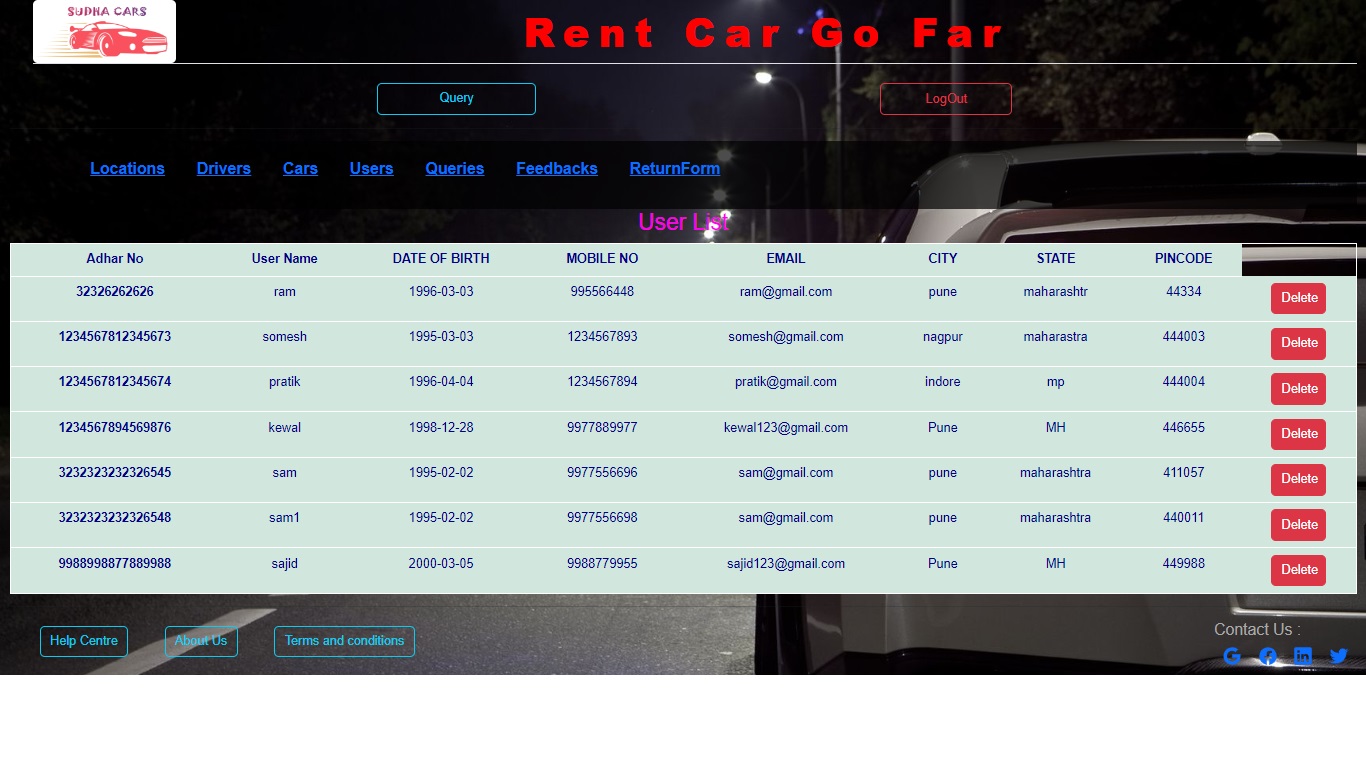
Query:



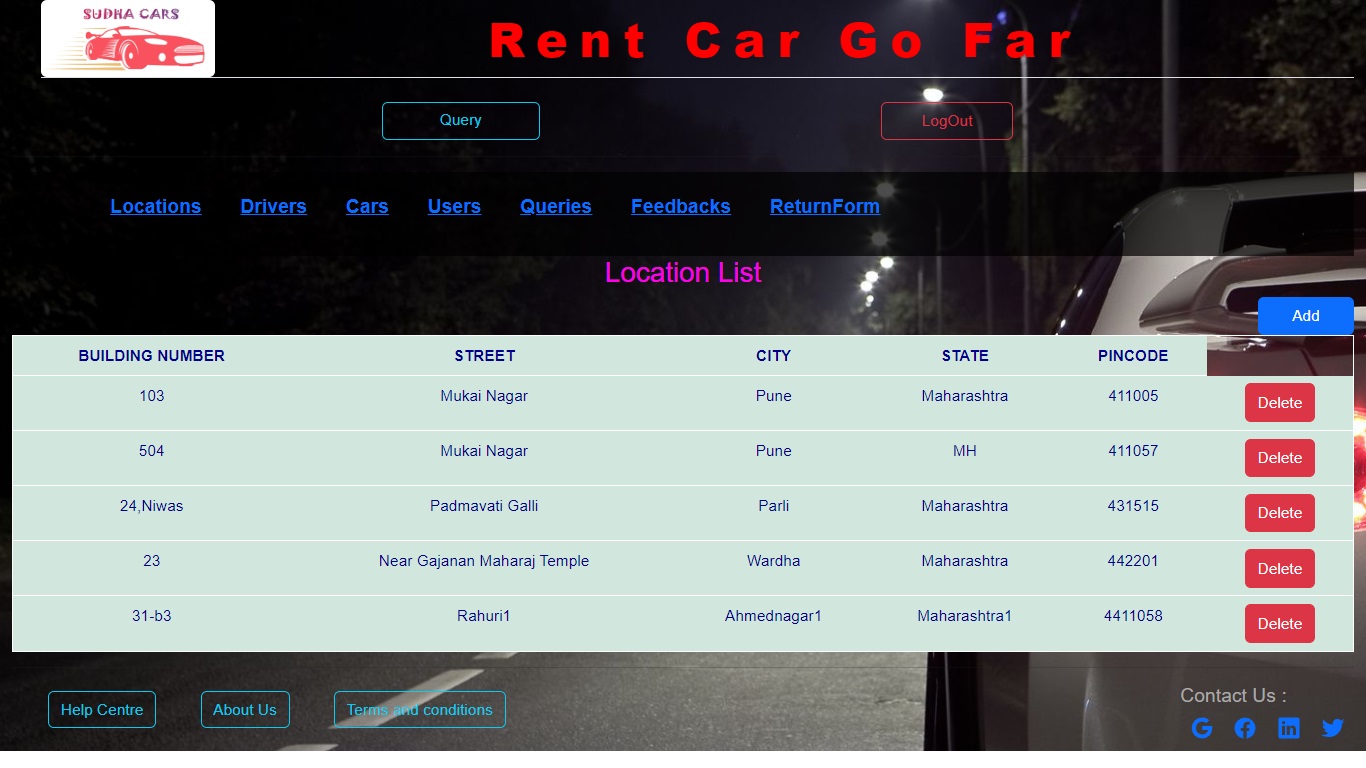
Return Form:



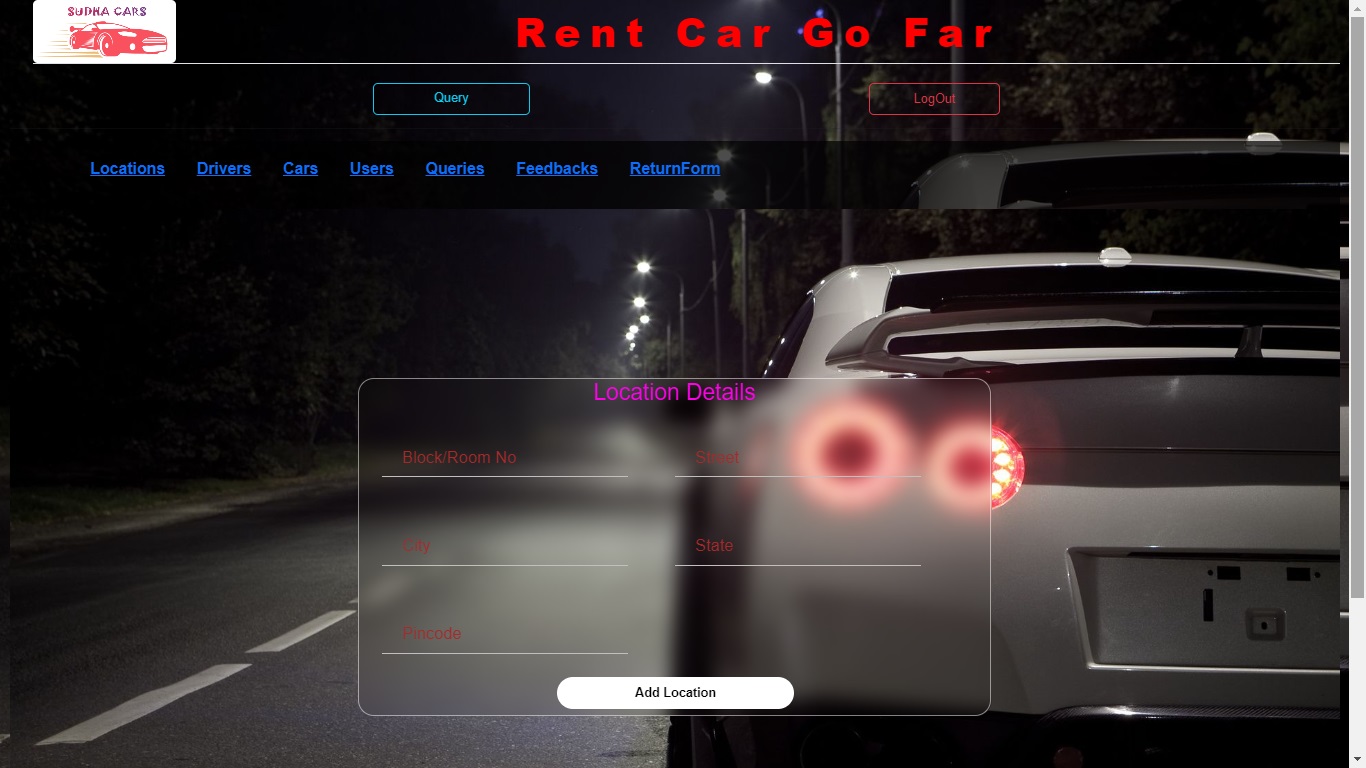
User List:



Locations:

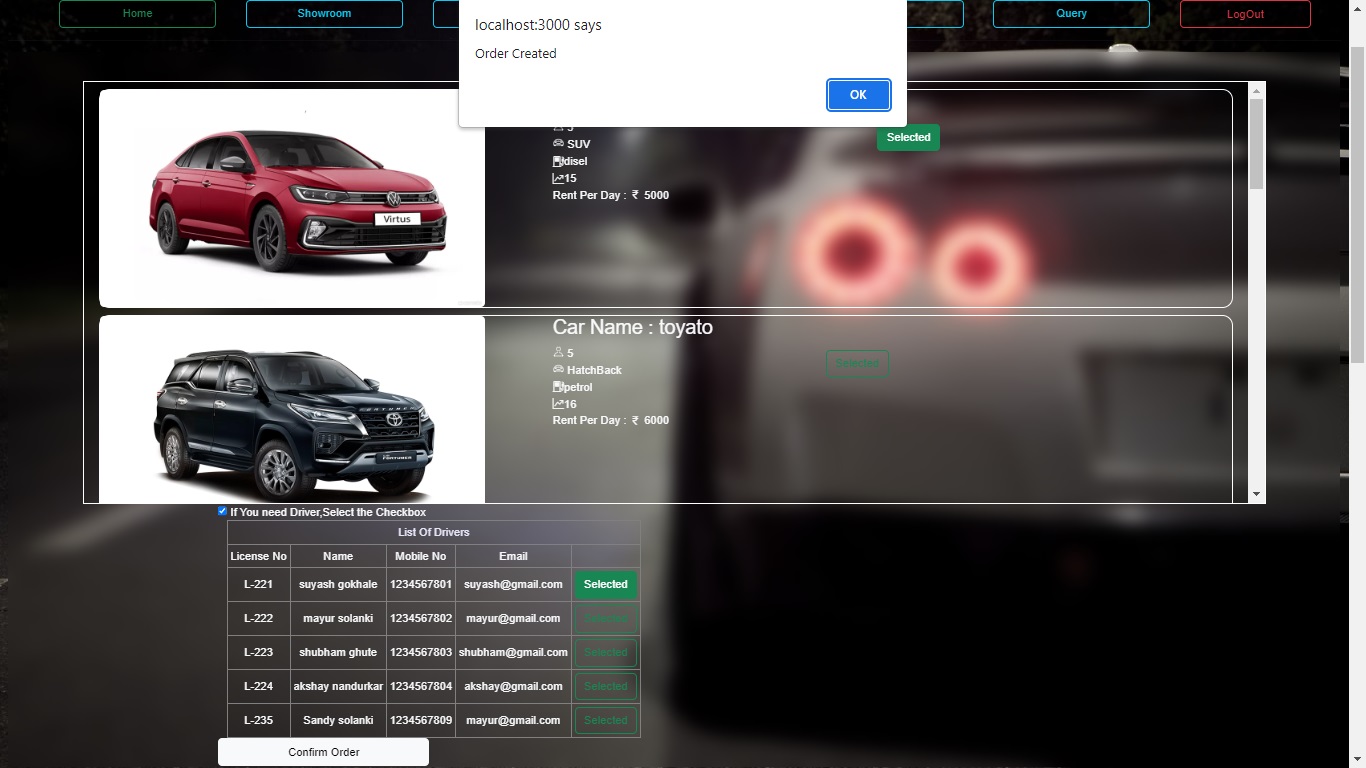


Location Add:

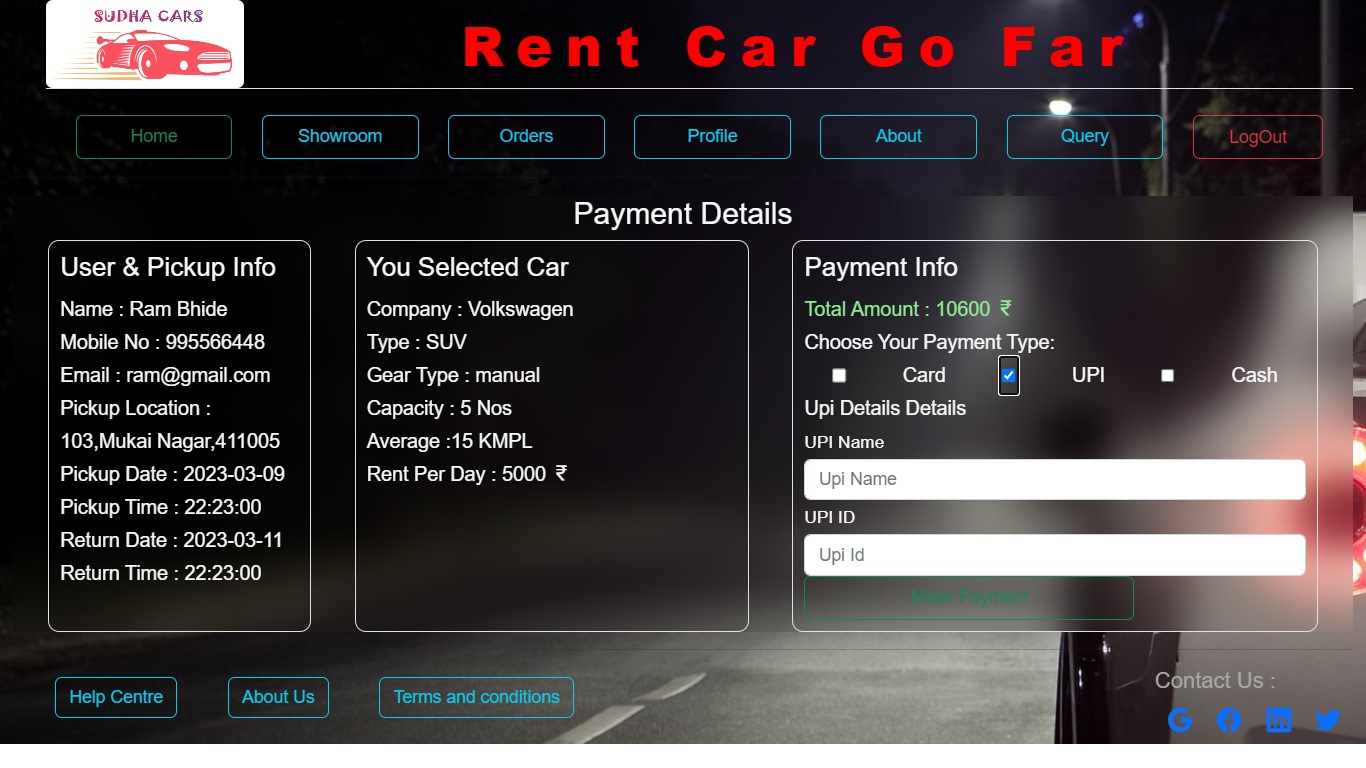


**User Pages:**

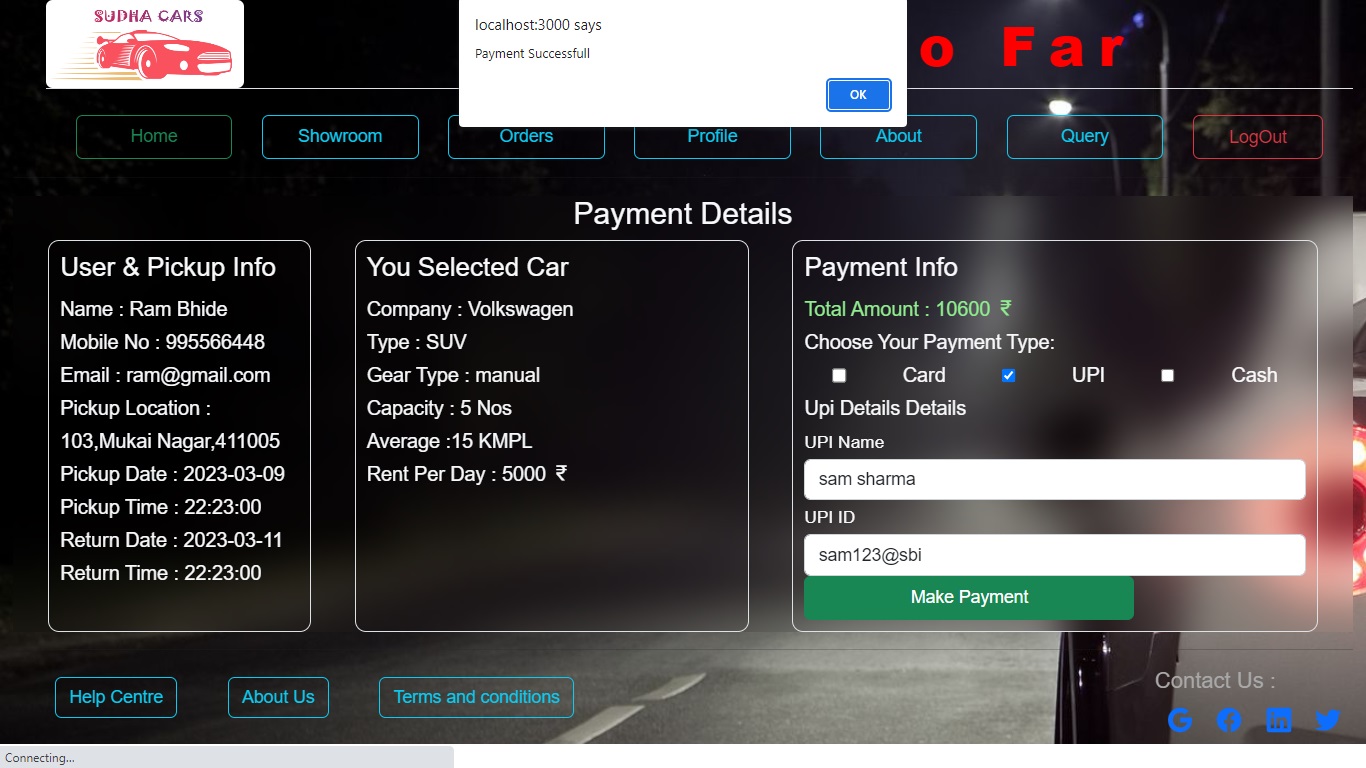
Order Created:



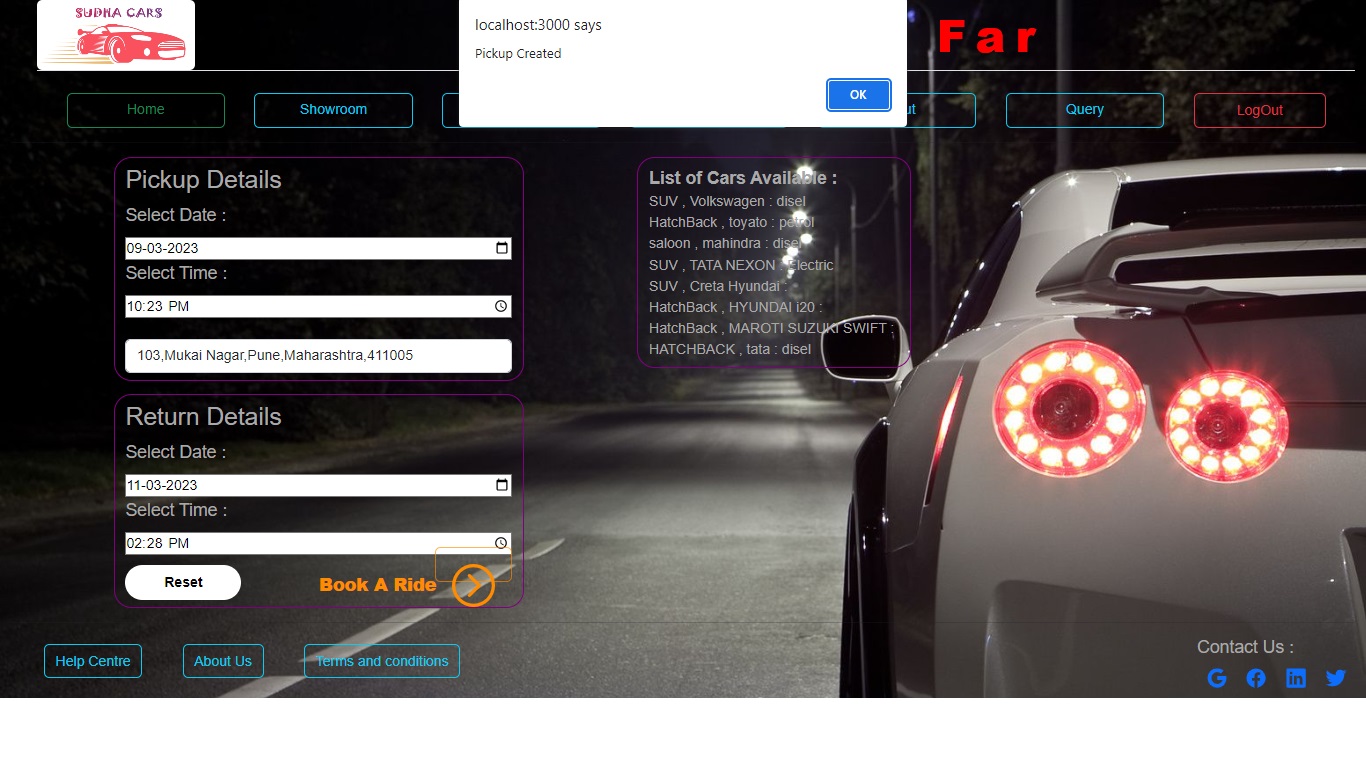
Payment Page:



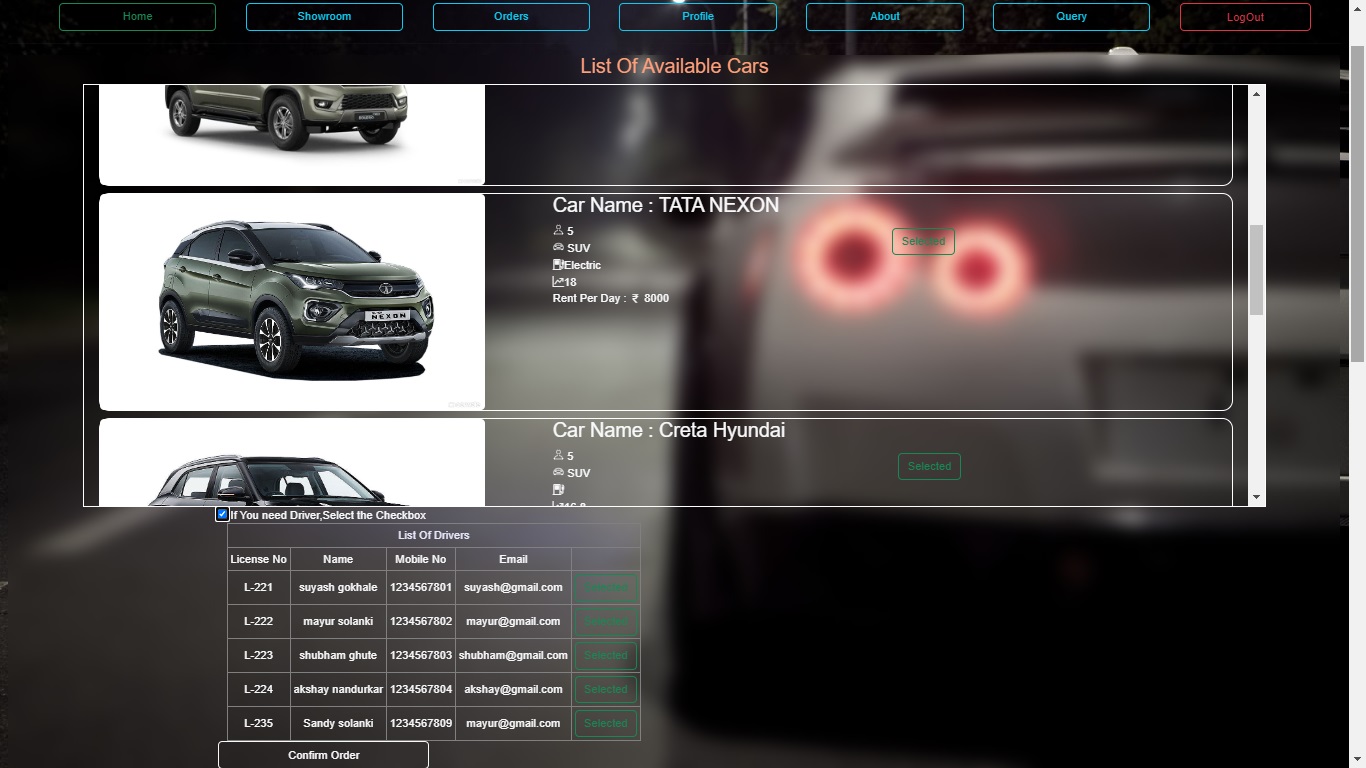
Payment Success:



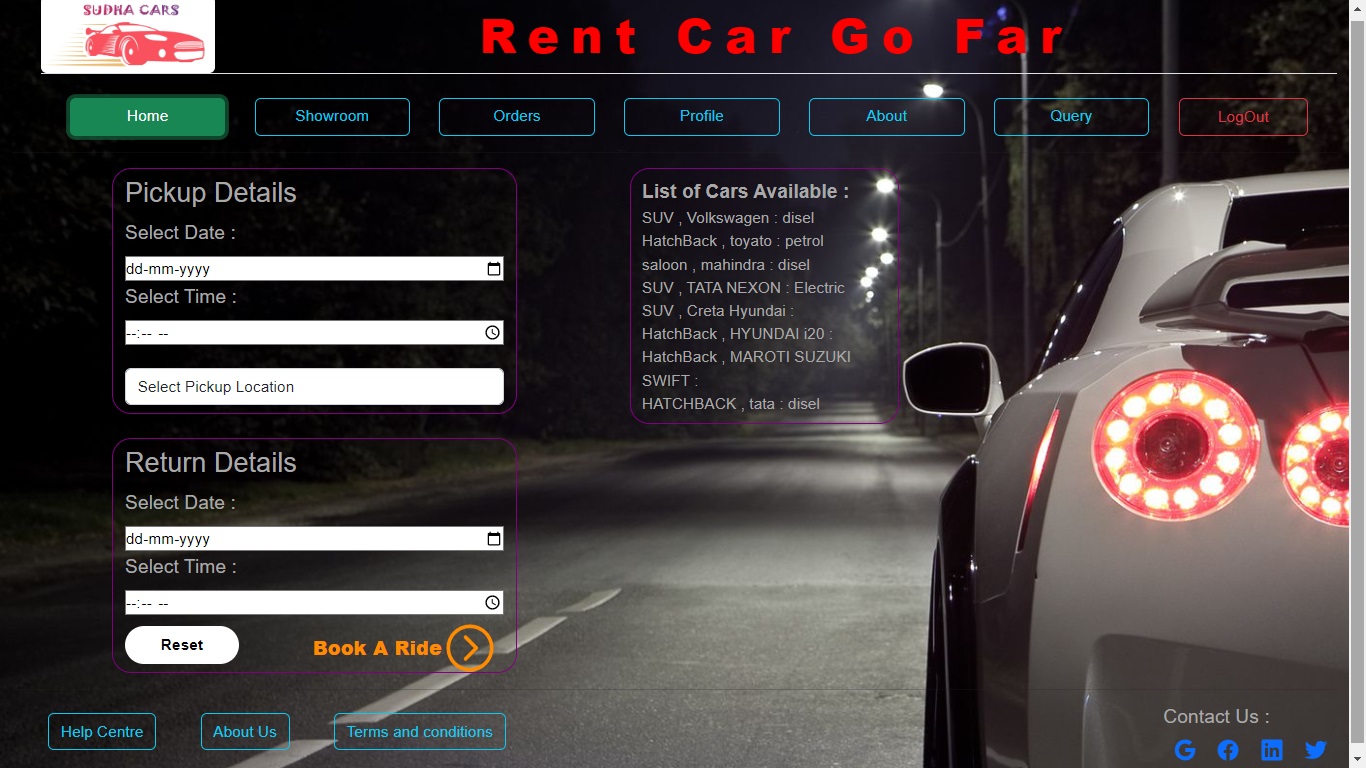
Pickup Created:



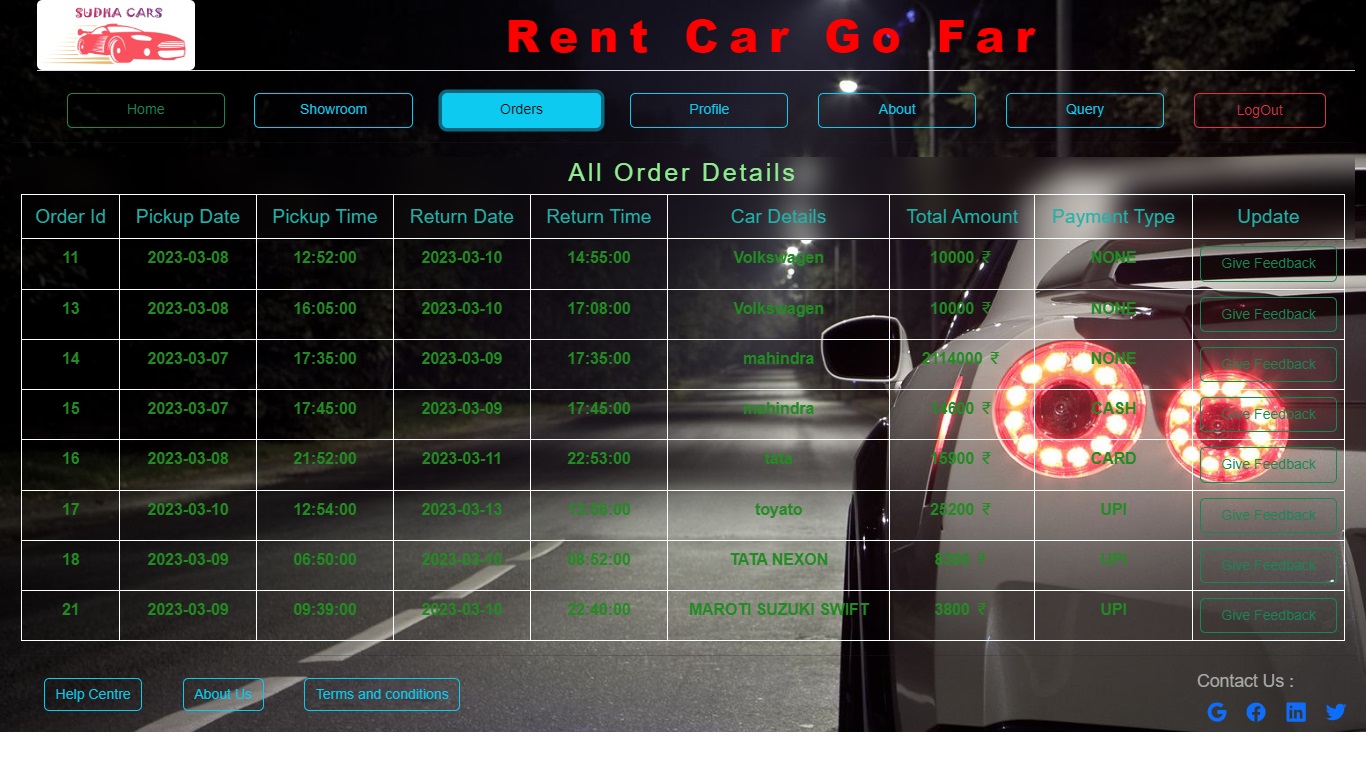
Car Page:



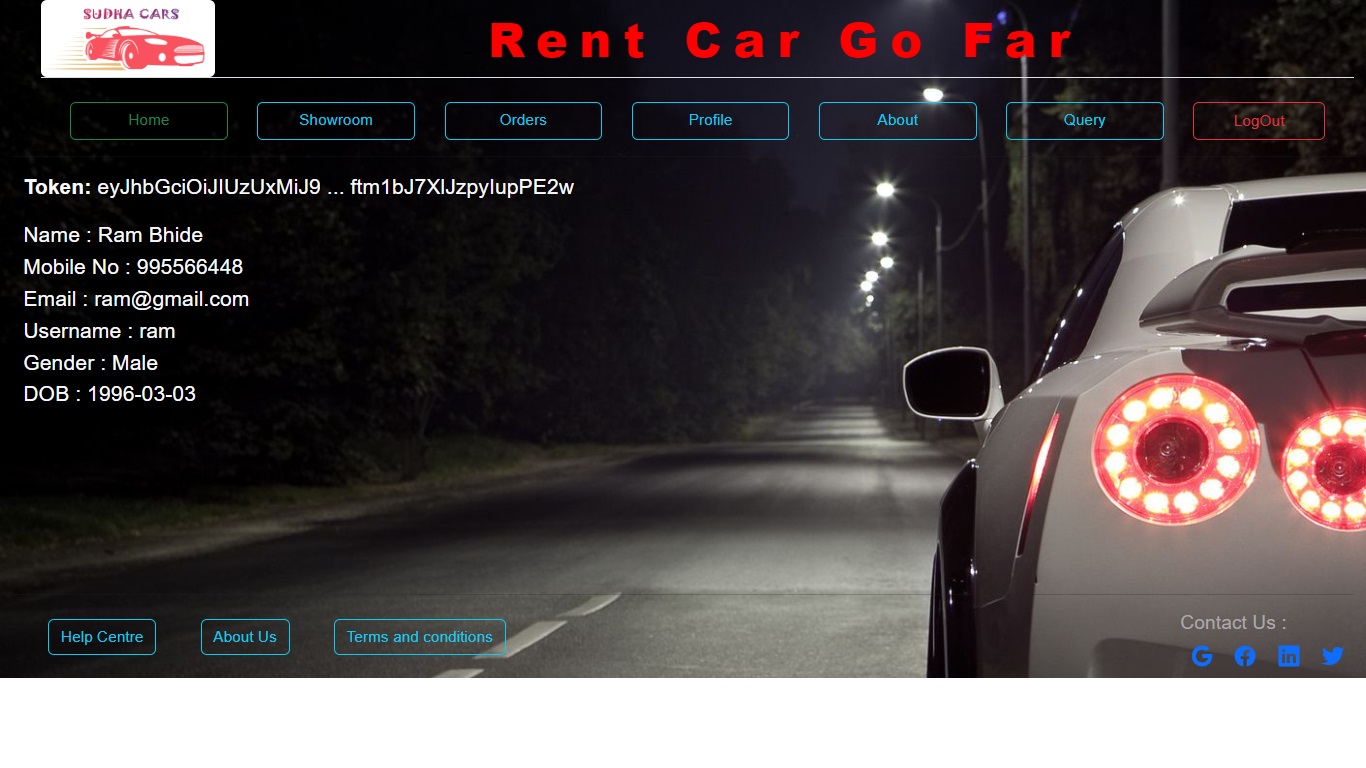
Home Page:



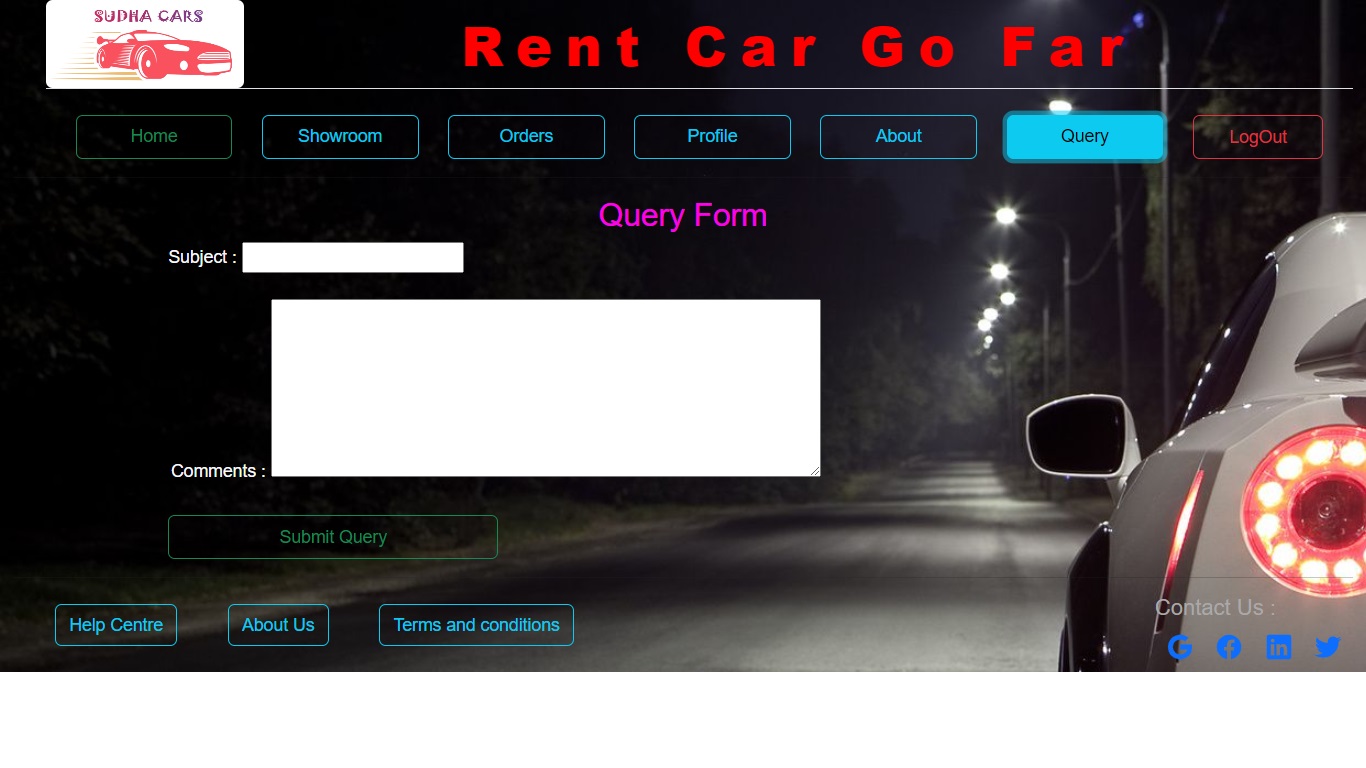
Orders:



Profile:

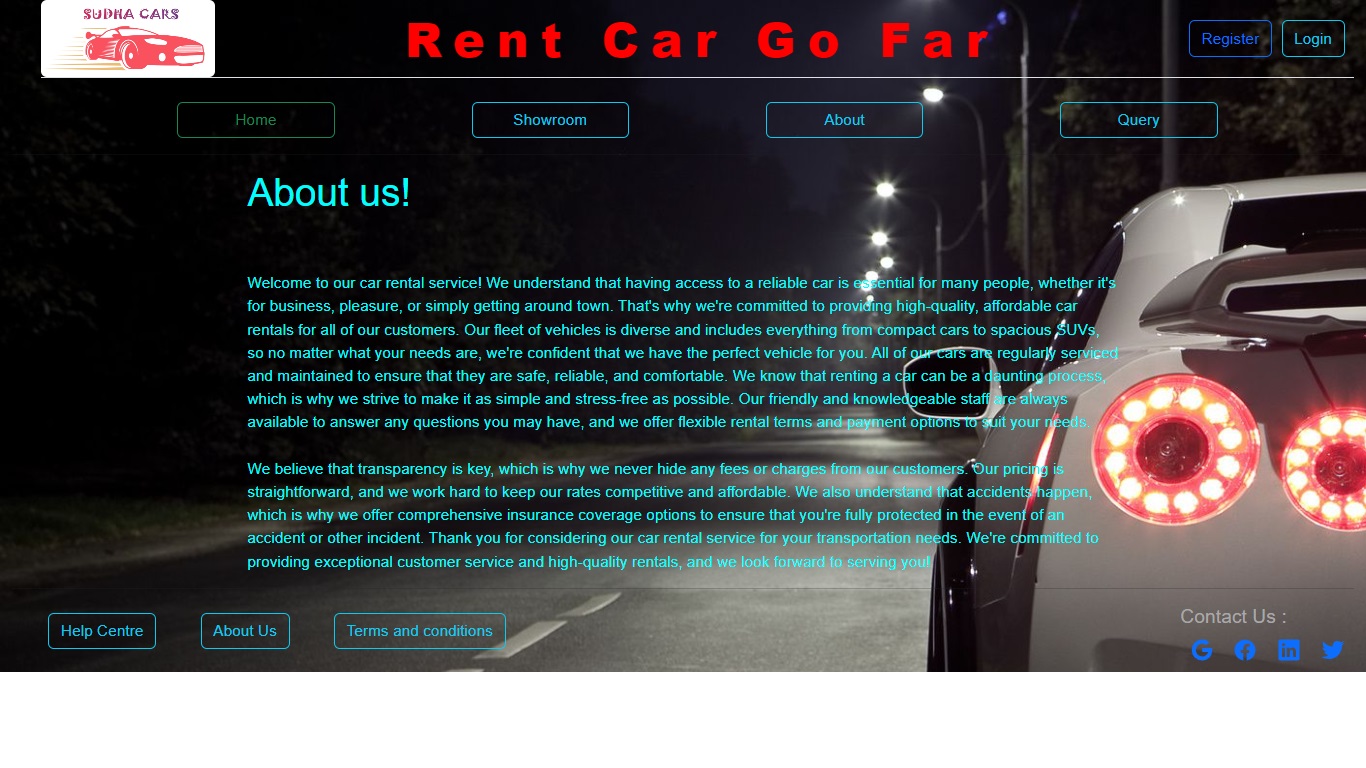


Query:

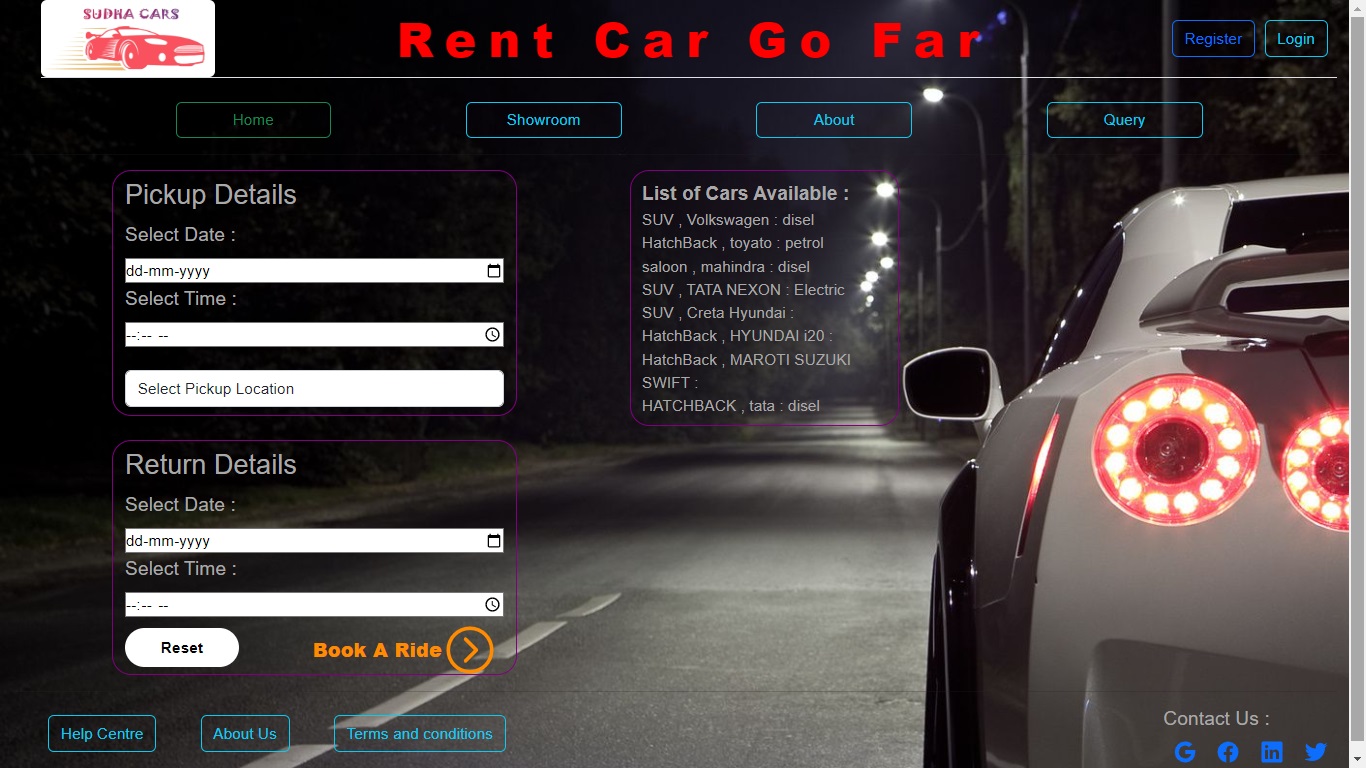


**Public Pages:**

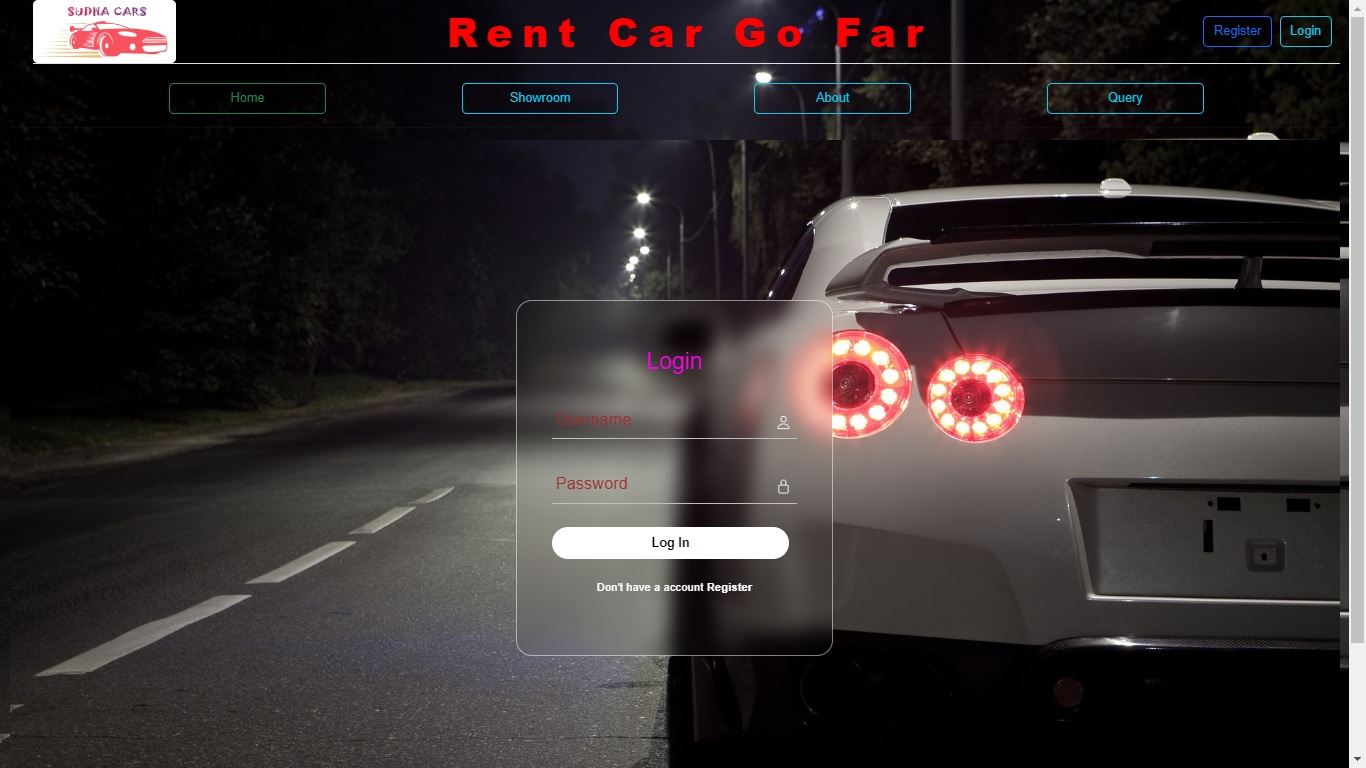
About Us:



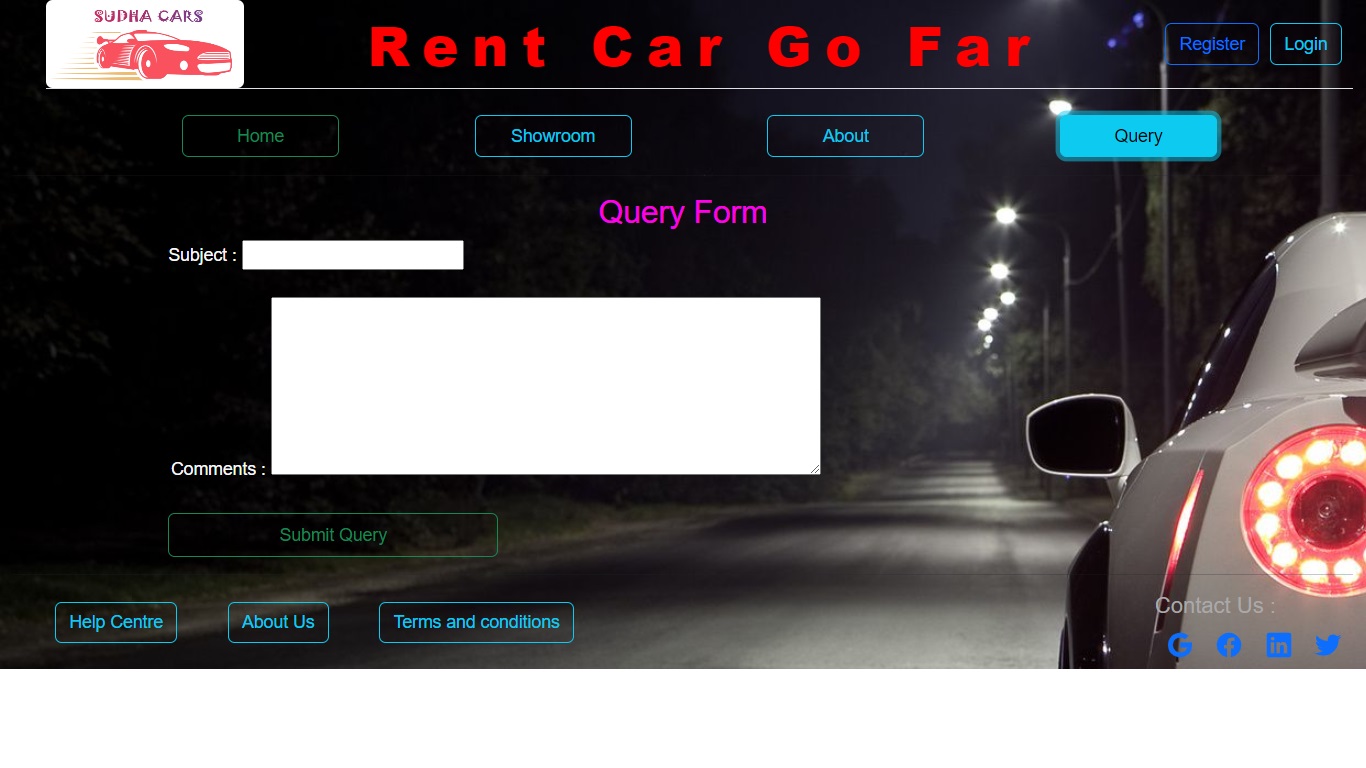
Home Page:



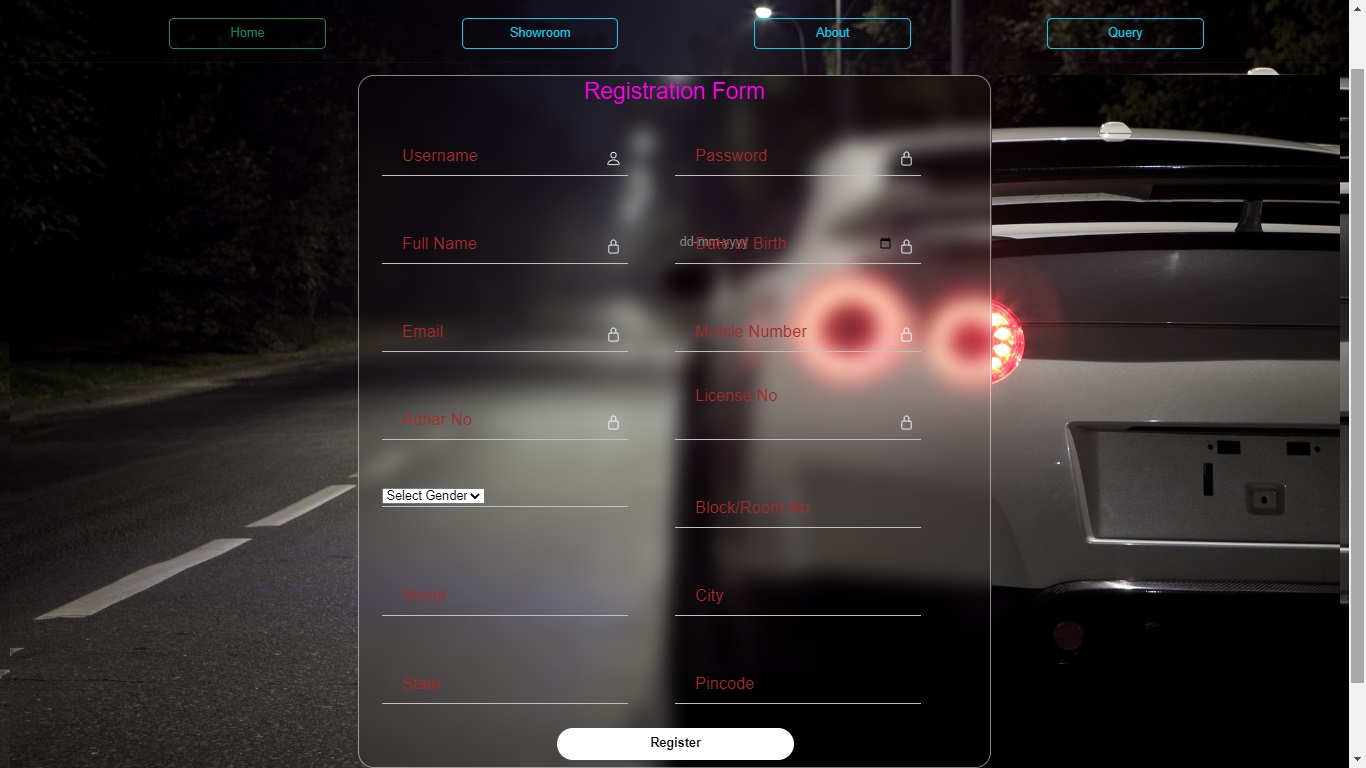
Login Page:



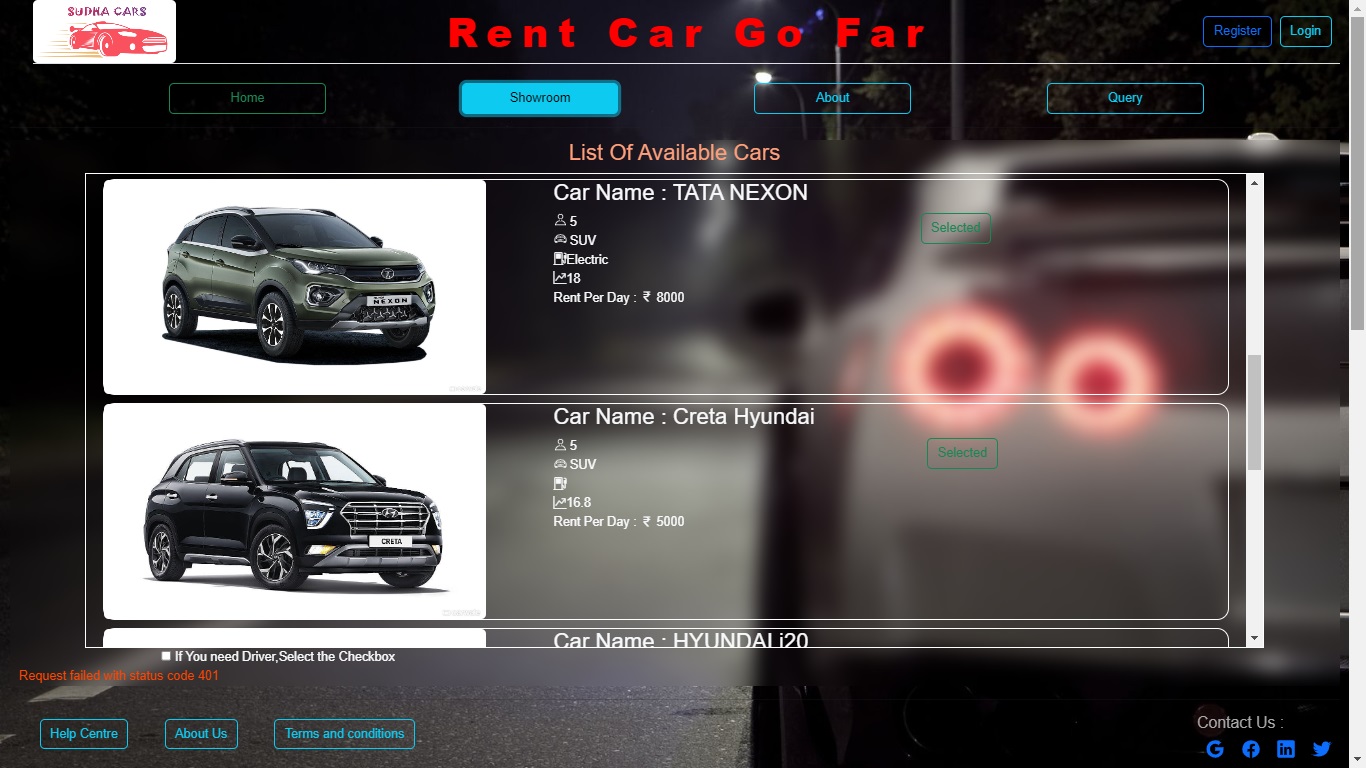
Query Form:



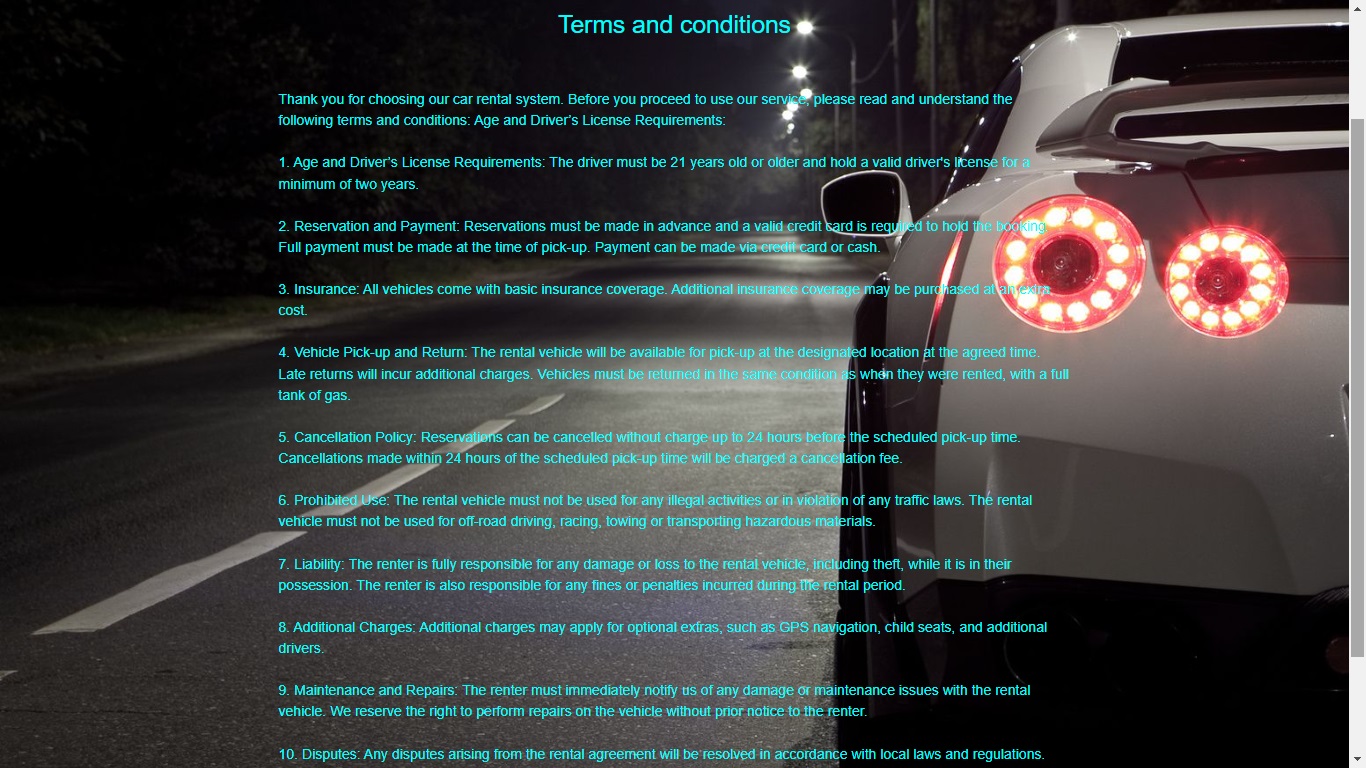
Register:



Showroom:



Terms And Conditions:



**7. REFERENCES**

https://www.blablacar.in/

<https://www.carrentingsolution.com/>

http://www.ola.cars

[http://www.w3.org](http://www.w3.org/)

[http://www.wikipedia.org](http://www.wikipedia.org/)

[http://www.stackoverflow.com](http://www.stackoverflow.com/)