

**Web Technology Final Project**

Topic: Rental car Management System

Names: UMURERWA Alice

ID:24074

1. **Purpose:**

The purpose of the Rental Car Management System is to provide a web-based platform for managing the operations of a car rental business. The system aims to streamline the rental process, improve customer experience, and enhance the efficiency of car inventory management.

1. **Requirements:**

* User Management:

1. Registration and login functionality for customers and administrators.
2. User roles and permissions to differentiate between customers and administrators.
3. Allow customer to change their profile

* Car Inventory Management:

1. Ability to add, update, and delete car details (e.g., make, model, year, availability, rental rates).
2. Search and filtering options for customers to find available cars based on criteria (e.g.name and manufacture).

* Reservation Management:

1. Allow customers to make reservations for specific cars and dates.

* Admin Dashboard:

1. An intuitive and secure administration panel for managing cars, reservations, and users.
2. Notify the client whether the reservation approve or rejected

* User-Friendly Interfaces:

1. Responsive and intuitive user interfaces for customers and administrators.
2. Easy navigation and clear instructions for performing tasks.
3. **Expected Outcomes:**
4. Efficient Car Inventory Management: The system should enable the rental company to easily add, update, and delete car details, including make, manufacture, year, availability, and rental rates. It should provide search and filtering options for customers to find available cars based ,car type.
5. Seamless Reservation Process: The system should allow customers to make reservations for specific cars and dates.
6. **Specific Constraints:**

Technology: The system should be developed using the Spring Boot framework, ensuring compatibility with Java-based technologies and databases.

Security: The system should prioritize the security of user data, including personal information. It should implement secure authentication and authorization mechanisms.

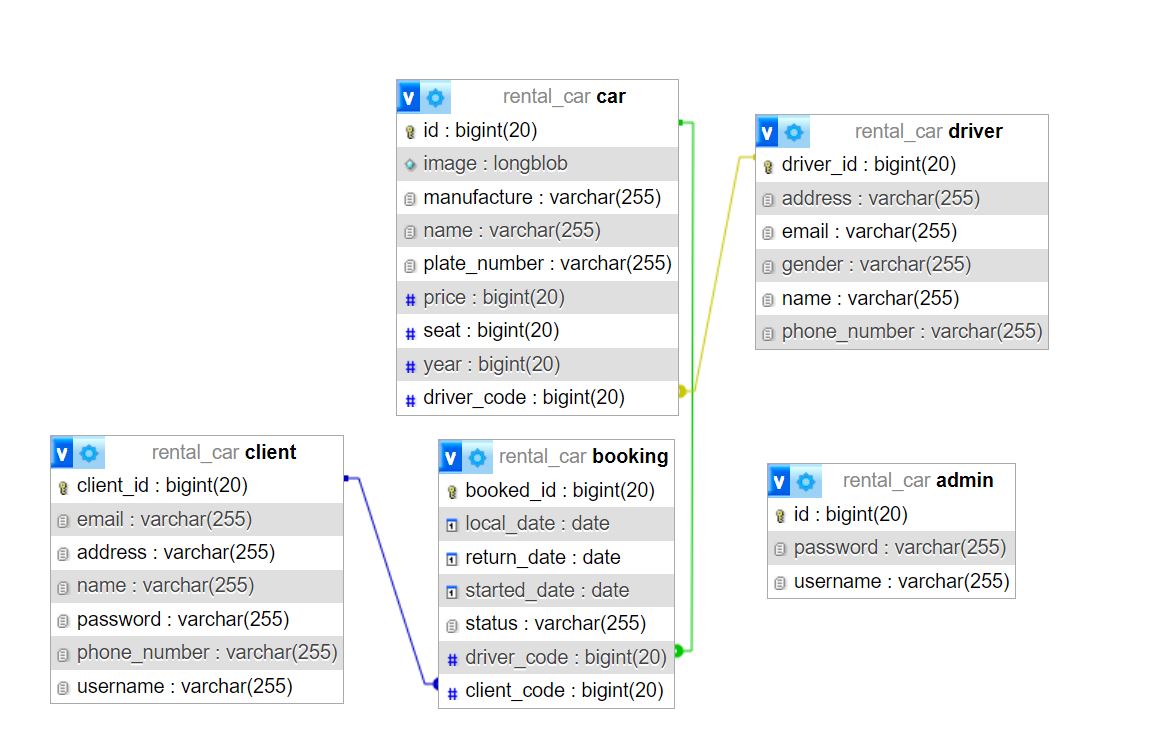
Scalability: The system should be designed to handle a growing number of cars, customers, and reservations. It should be scalable to accommodate future expansion and increased system usage.

User-Friendly Interfaces: The system should provide intuitive and user-friendly interfaces for customers and administrators. The interfaces should be responsive, easy to navigate, and provide clear instructions for performing tasks.

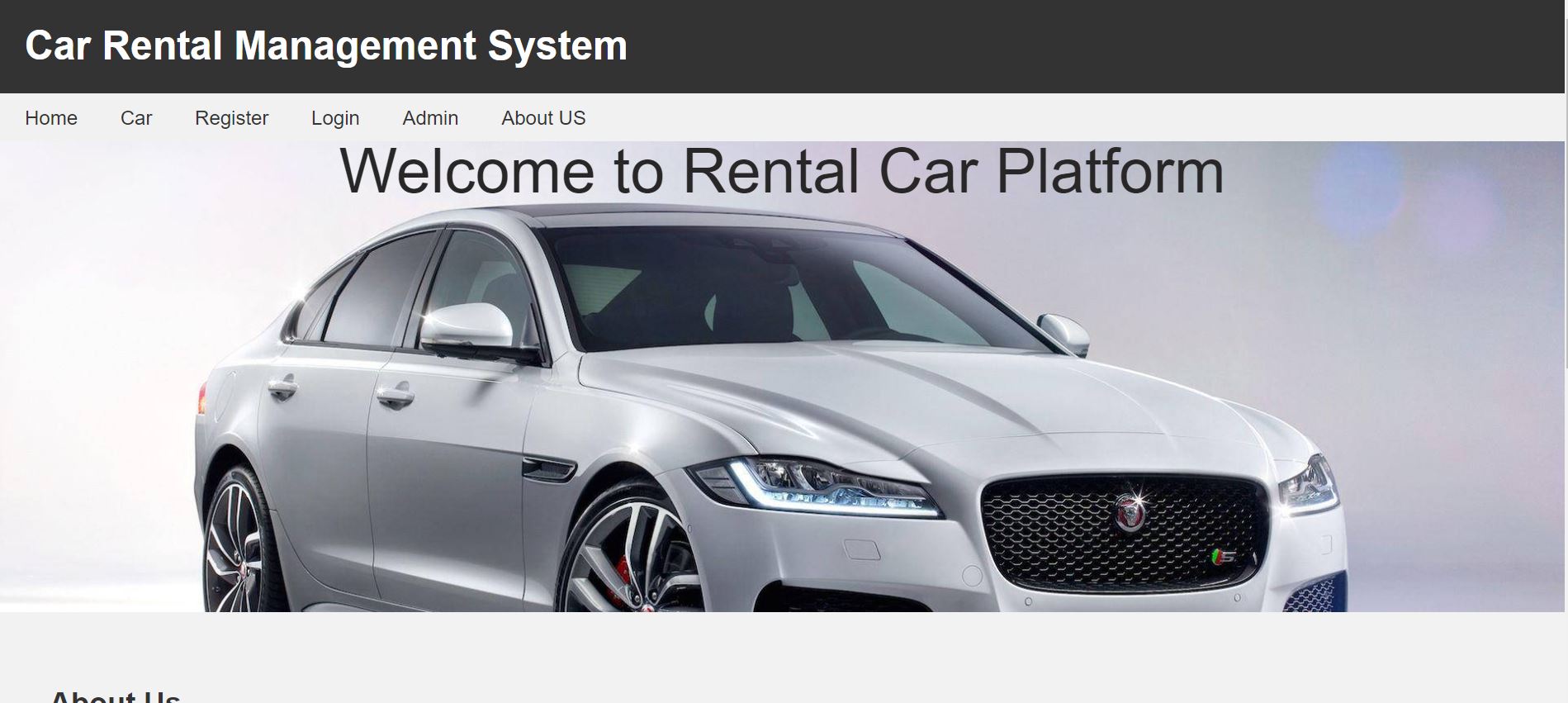
Documentation: The project should include comprehensive documentation, including installation instructions, user manuals. This documentation will assist users and developers in understanding and utilizing the system effectively.

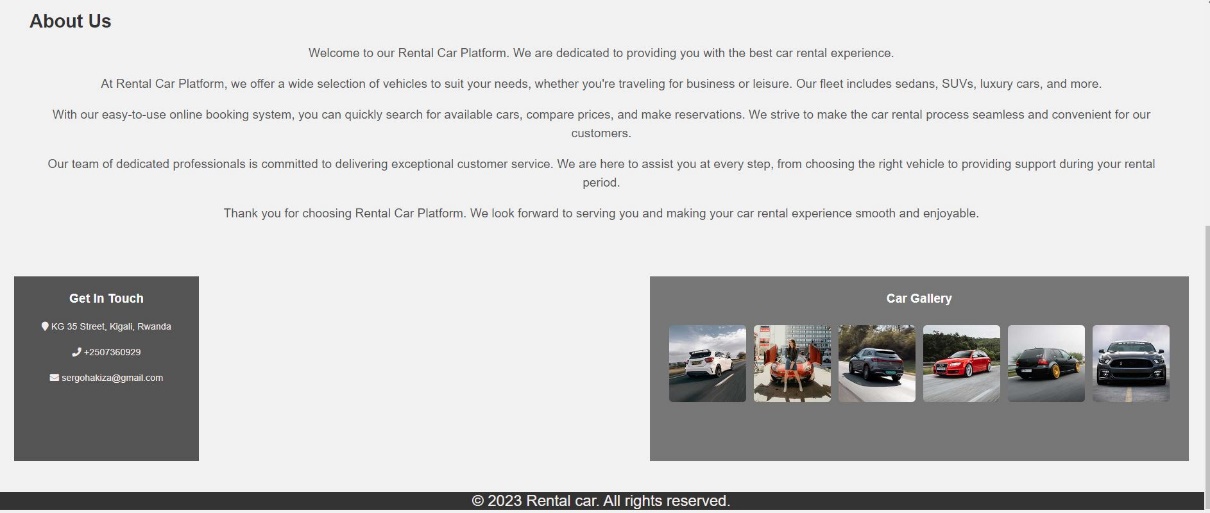
1. **Limitations:**
2. System Performance: The system's performance may be influenced by factors such as server capabilities, network infrastructure, and database performance.
3. Customization and Extensibility: While the system aims to meet common requirements for a rental car management system, customization and extensibility may be limited. Adding new features or functionalities not initially considered in the requirements may require additional development efforts.
4. The System is limited to the payment functionality user my required to pay manually

**5.Database Schema:**

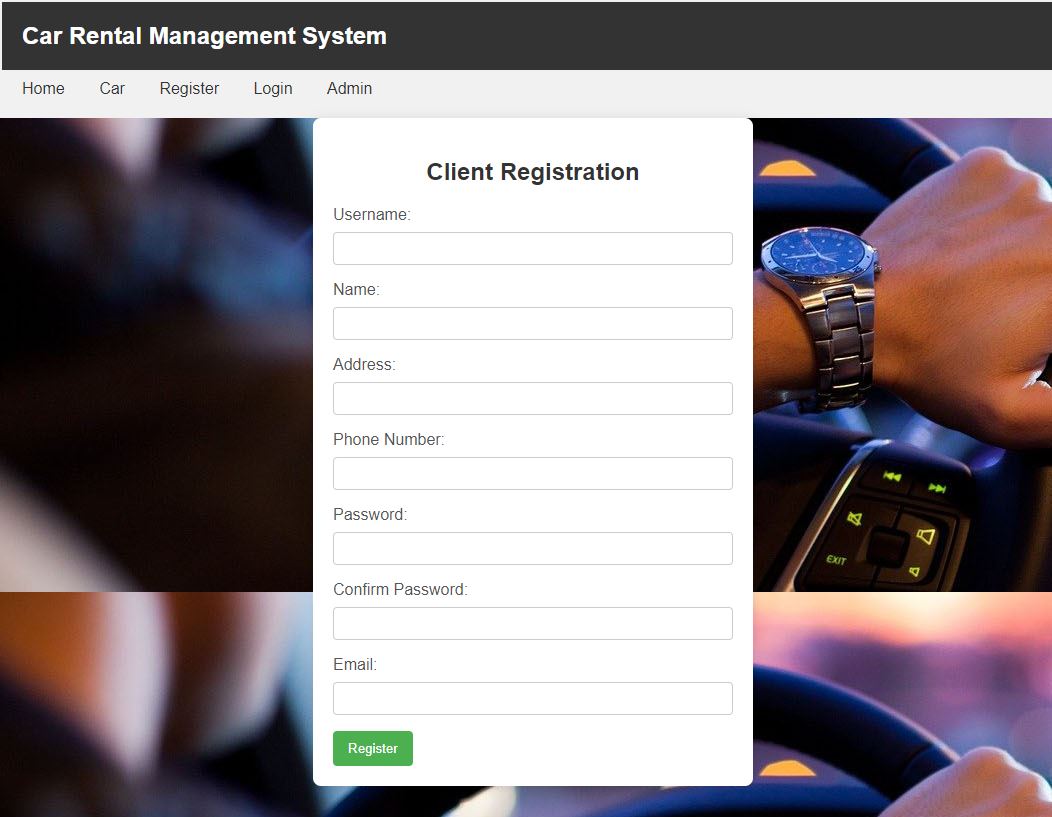
****

1. **User Documentation:**

**I.** First once user load the application he view Welcome 

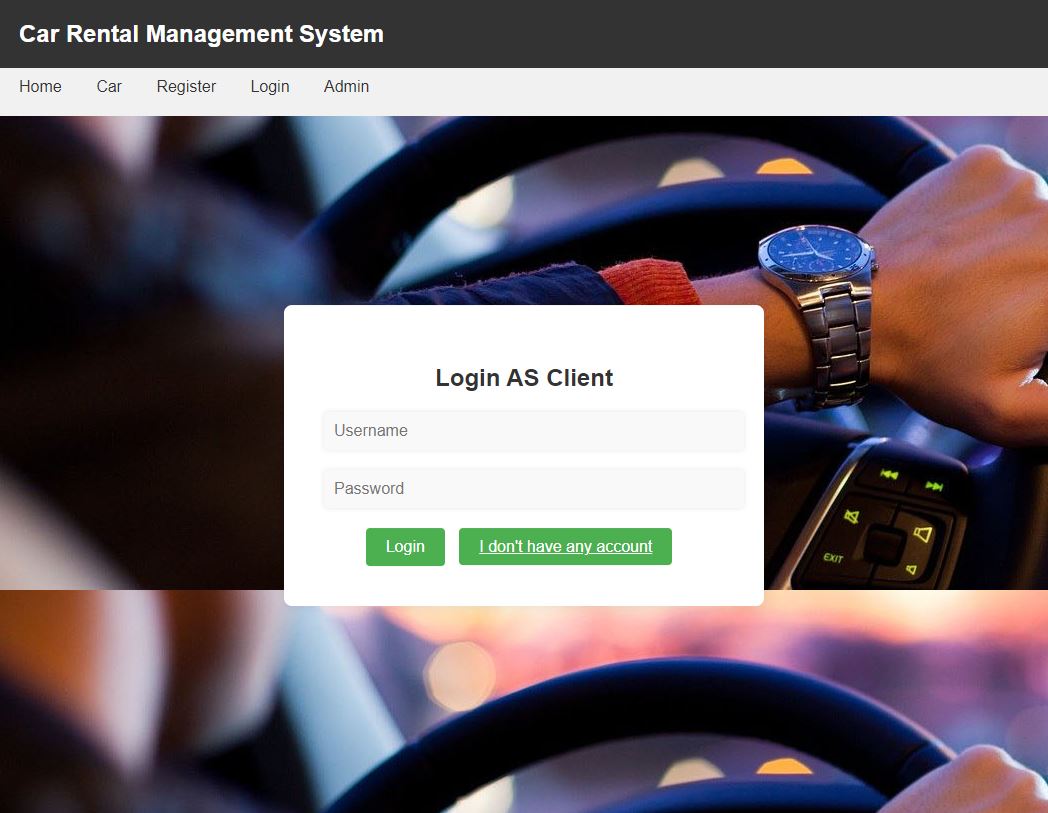


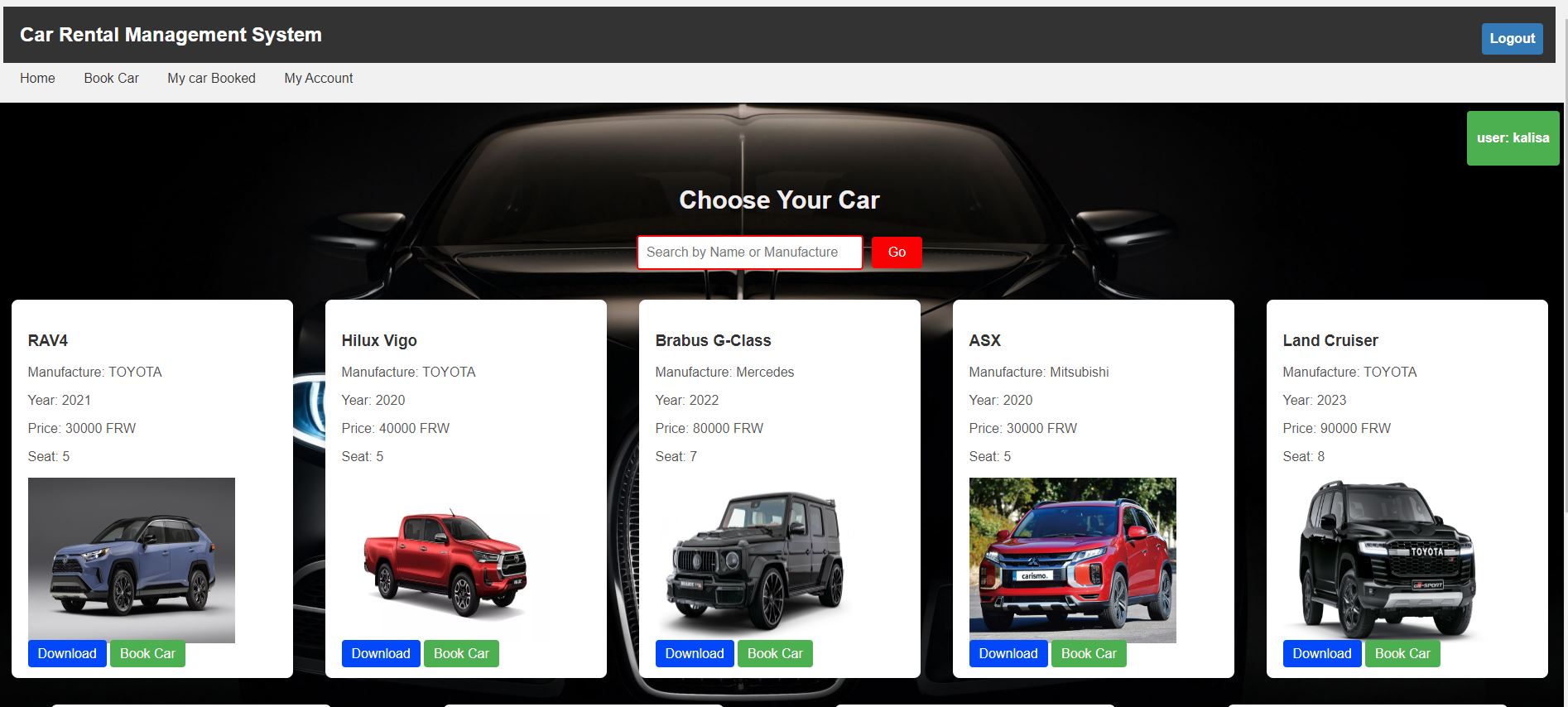
II. For the new user to Enter in the system need to make Registration after the Login as user



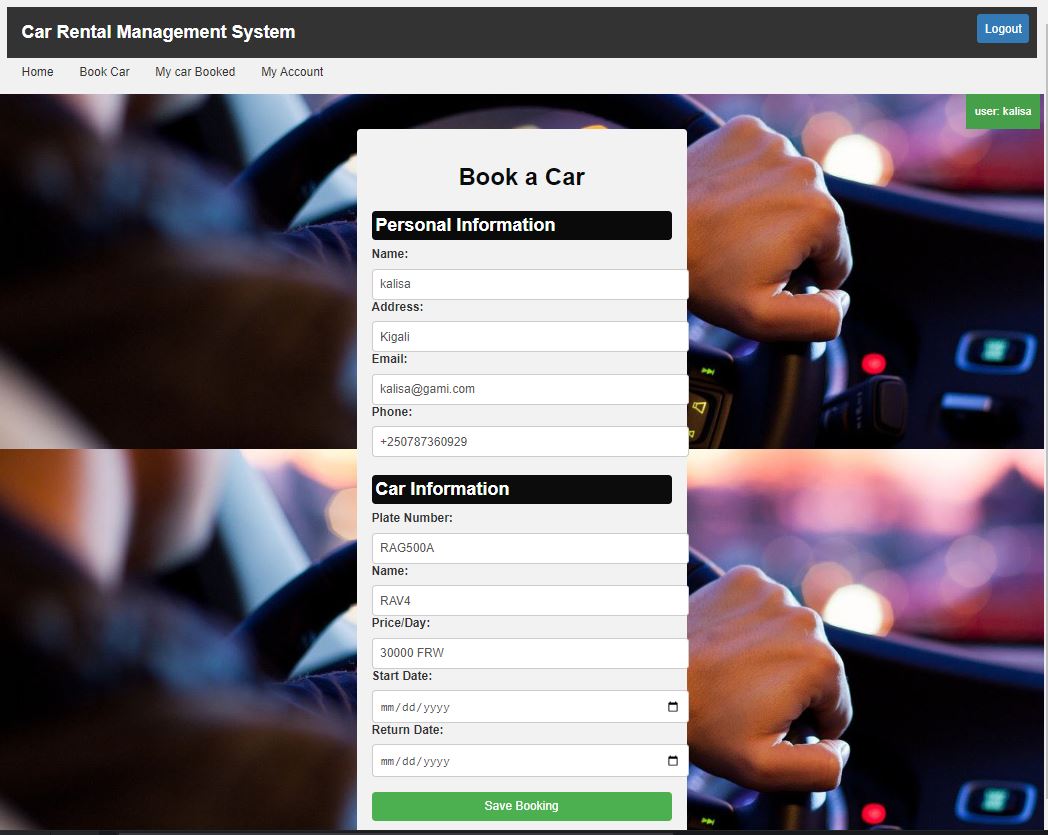
On Login page, user should Enter the username and password entered during registration otherwise the system will not allow him to continue. For this tutorial let use .

Username: kalisa, password :1234

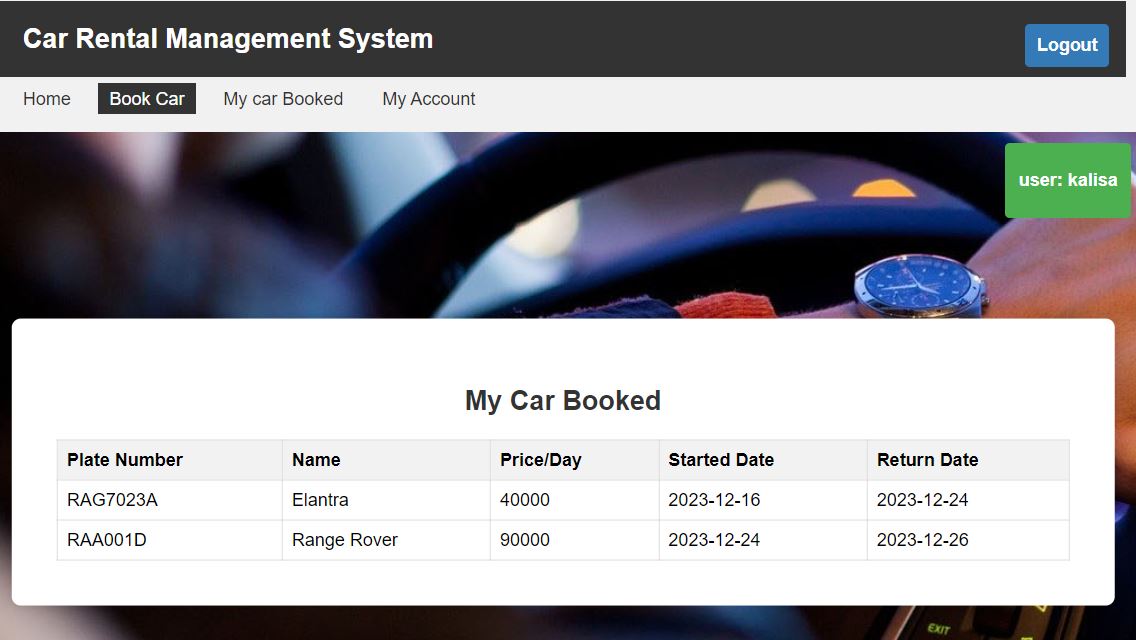


When login successfully user will direct to his or her account where he can perform make thing like booking, download the image of cars, view booked car and account details, search  


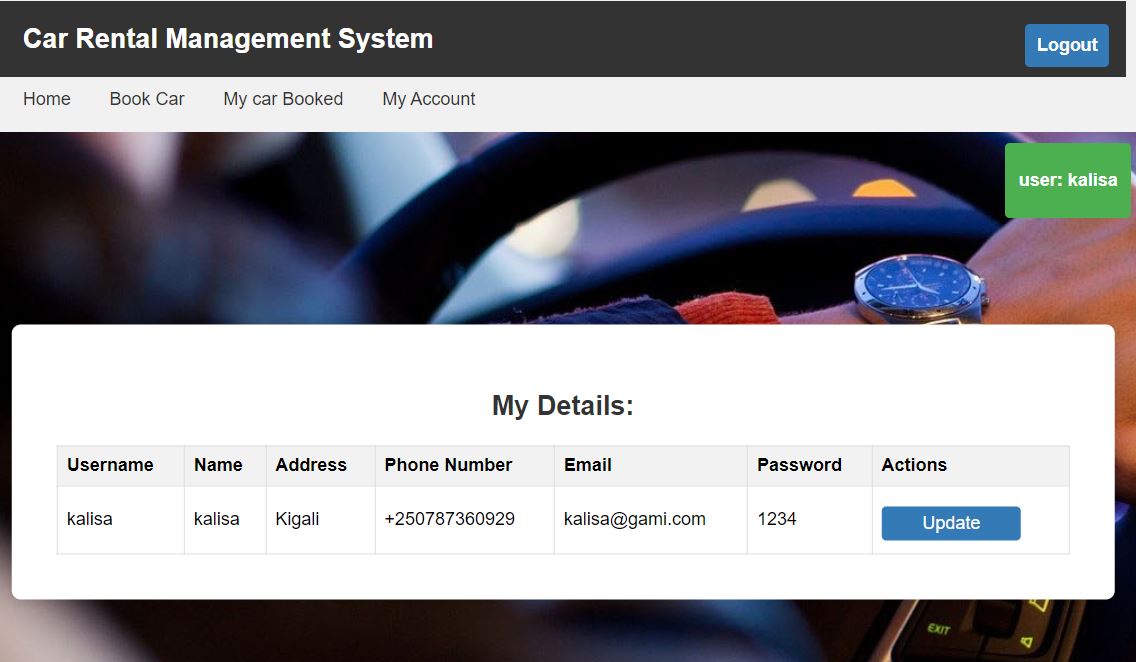
When user want to book car, click on book car button then the fill the form by choosing the started date and return date. After clicking saving booking, he waits the admin to approve the request



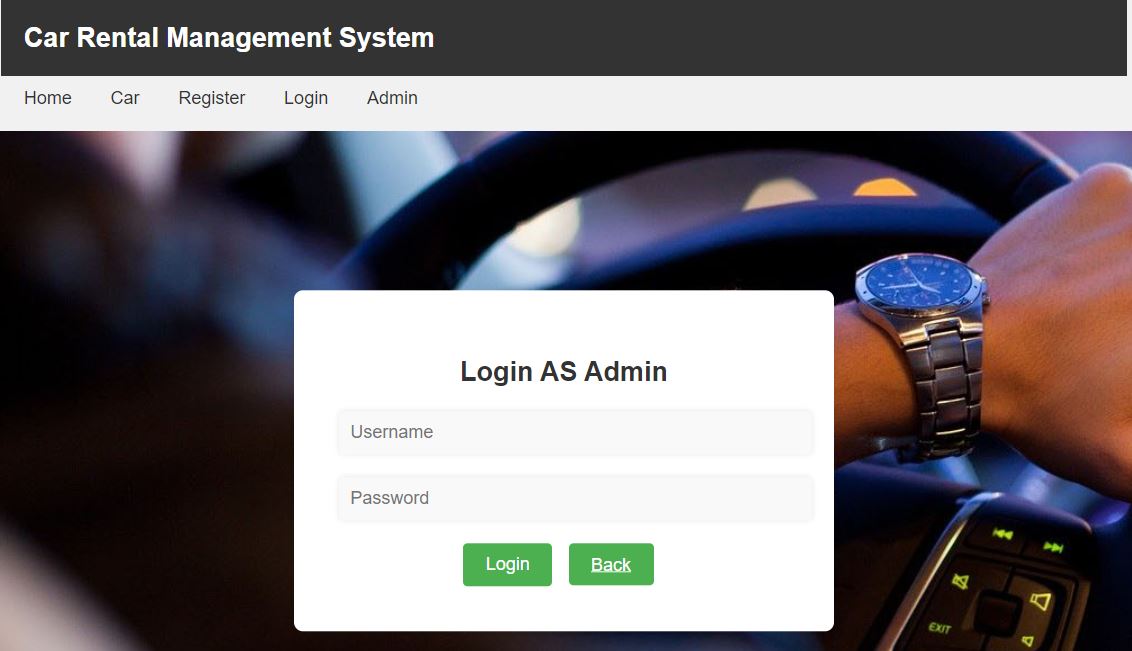
Here the client can booked request that have been approved by Admin

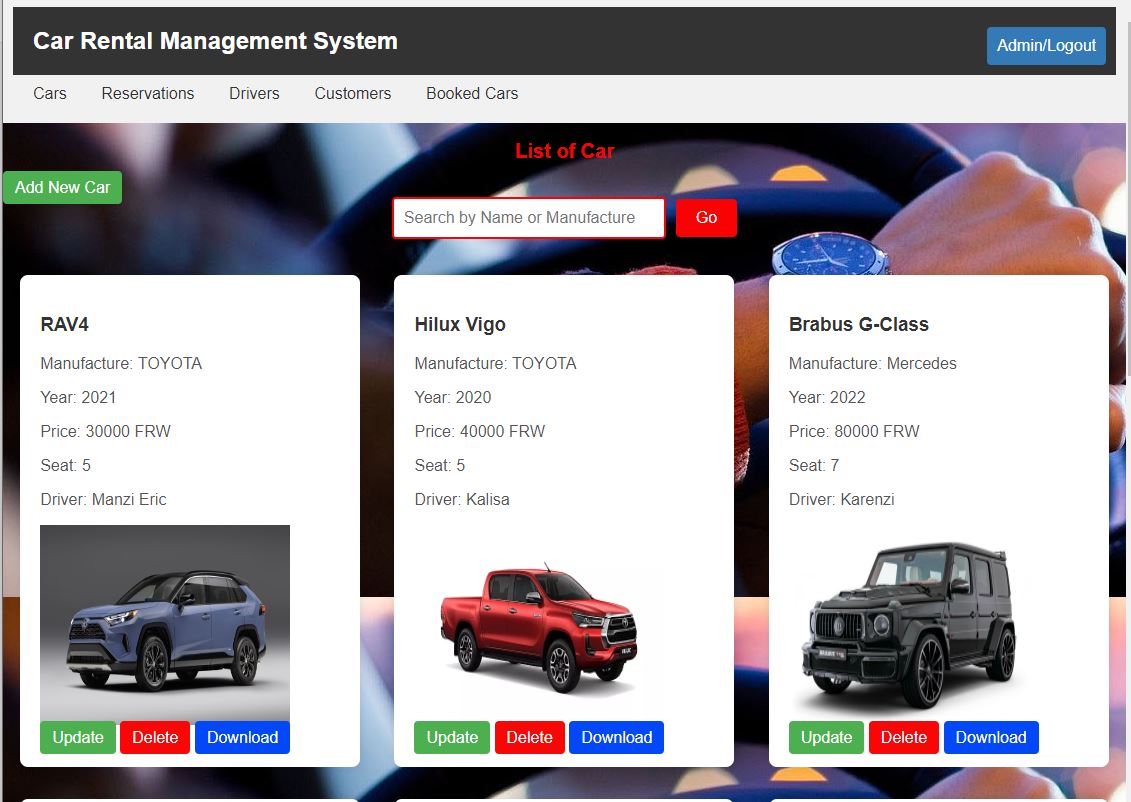


Here the user can view his or her information and can update it if needed. The user can log at any time if he/her press Logout button.

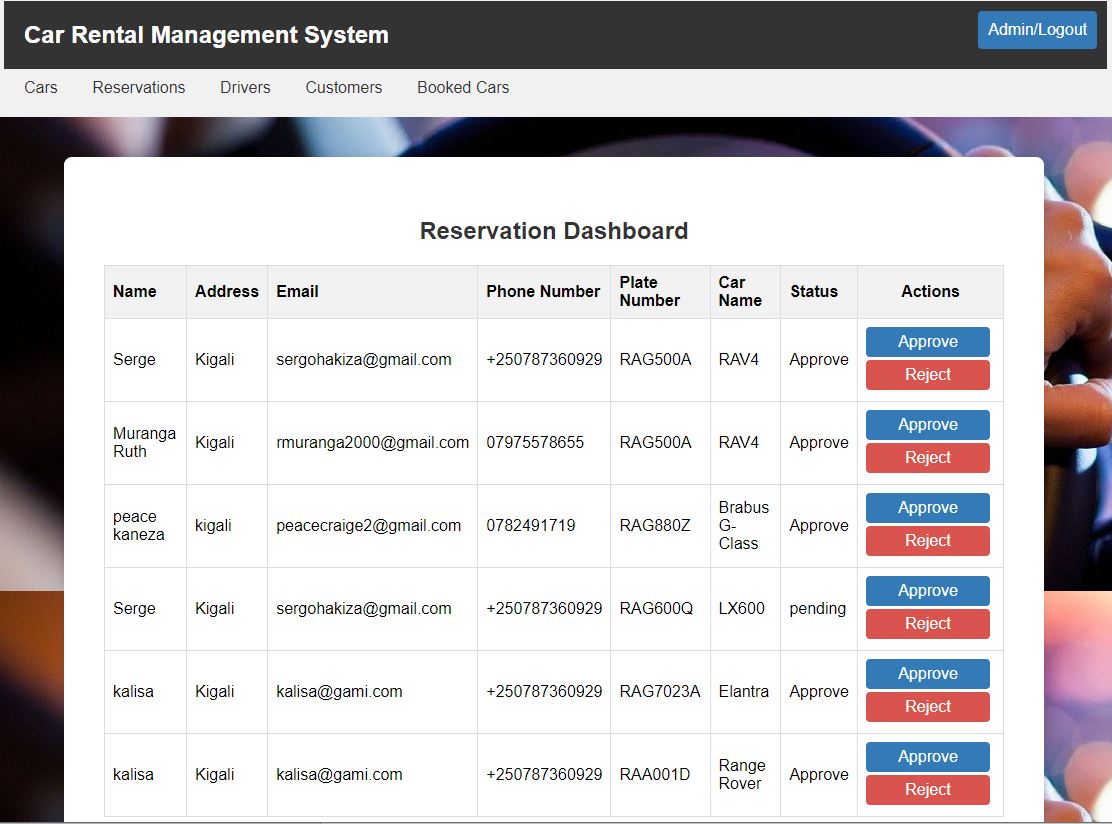


For Admin Account the Administrator must login as admin by providing the username and password

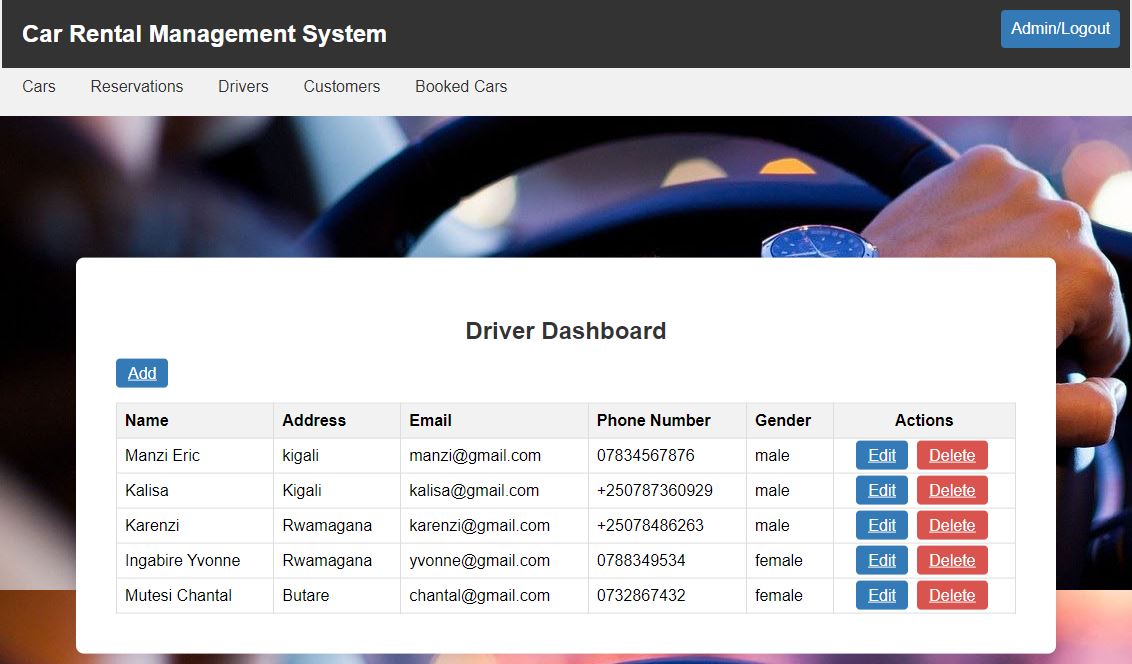


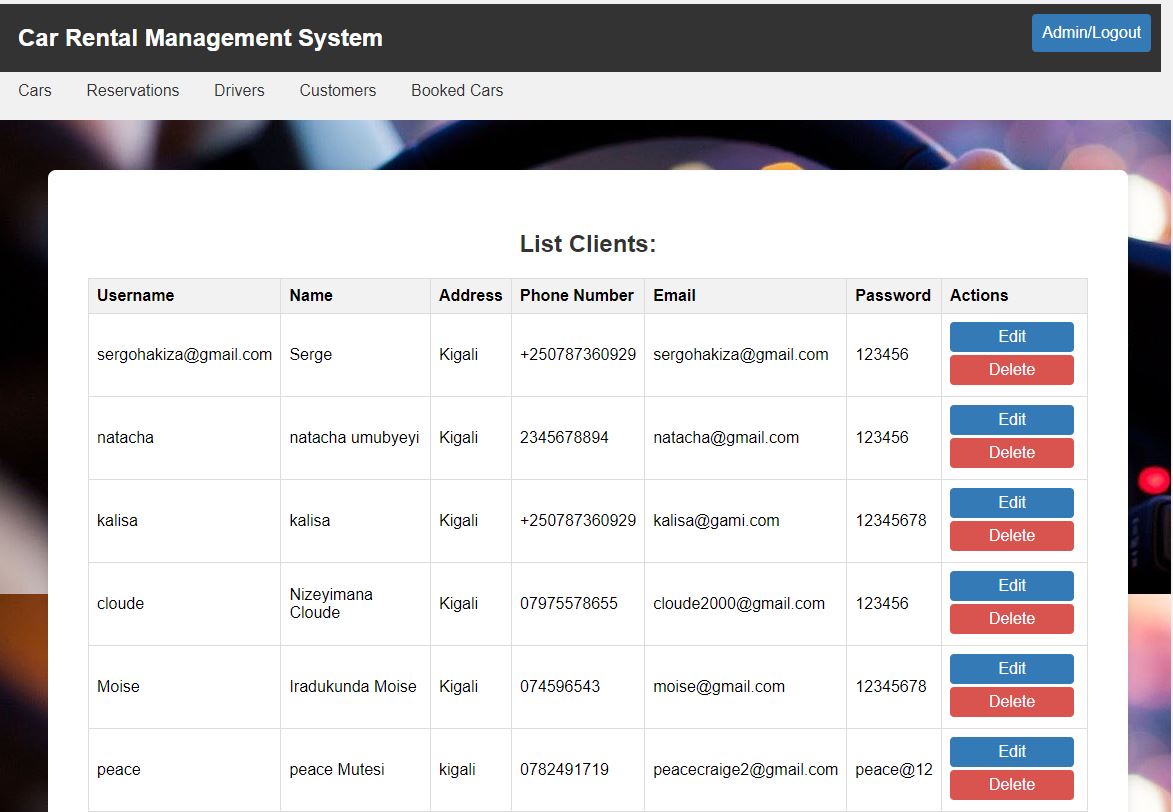


Admin can approve or reject the booked request and the system will Automatically sending the email to the customer.

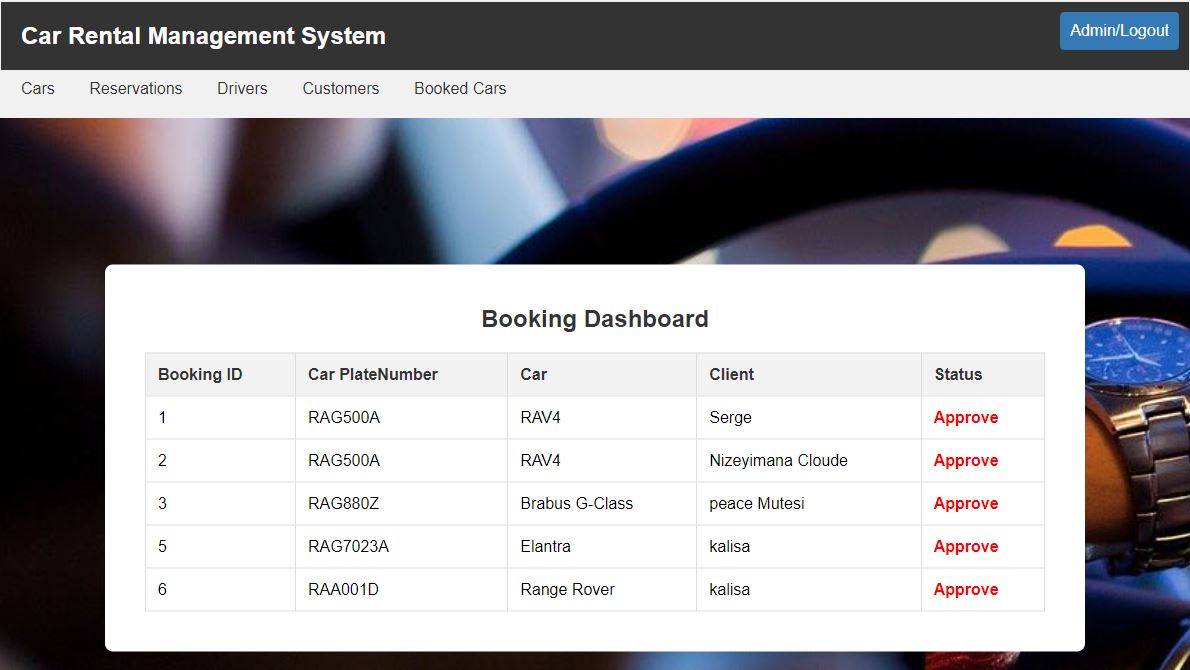


Admin can also add, update and delete the driver information

  
Admin can also add, update and delete the client information



Admin can also view the booked car with the request approved information



1. **Technical Documentation: Rental Car Management System**

**Architecture Overview:**

The Rental Car Management System is developed using the Spring Boot framework, which follows a layered architecture pattern. The system consists of the following layers:

a**. Presentation Layer:** This layer handles user interaction and displays the user interfaces. It includes controllers responsible for handling HTTP requests and rendering appropriate responses.

b. **Service Layer:** The service layer contains the business logic of the application. It encapsulates the core functionalities and acts as an intermediary between the presentation and data access layers.

c. **Data Access Layer:** This layer interacts with the database and handles data persistence. It includes repositories or DAOs (Data Access Objects) responsible for querying and manipulating data.

d. **Database**: The system uses a relational database management system (RDBMS) such as MySQL for data storage.

**Implementation Details:**

The Rental Car Management System is implemented using Java programming language and the Spring Boot framework. Some key implementation details include:

* 1. **Spring Boot:** The system leverages the Spring Boot framework for rapid application development and to simplify configuration and deployment.
  2. **Spring MVC:** Spring MVC (Model-View-Controller) is used to handle the presentation layer and manage the flow of requests and responses.
  3. **Spring Data JPA:** Spring Data JPA is used to simplify the implementation of the data access layer. It provides abstractions and automatic CRUD (Create, Read, Update, Delete) operations for interacting with the database.
  4. **Hibernate:** Hibernate is used as an Object-Relational Mapping (ORM) tool to facilitate the mapping between Java objects and the underlying database tables.
  5. Security: Spring Security is utilized for implementing authentication and authorization mechanisms, ensuring secure access to the system's resources.
  6. Dependency Management: Maven or Gradle is used for managing project dependencies and building the application.

**Additional Technical Details:**

Here are some additional technical details relevant to the Rental Car Management System:

* 1. Server-Side Technologies: The system runs on a Java Servlet container such as Apache Tomcat.
  2. Front-End Technologies: The user interfaces are built using HTML, CSS, thymeleaf and JavaScript frameworks.

e. Deployment: The system can be deployed on various platforms for source code is on github and hosted on Heroku.