

**INTERNATIONAL UNIVERSITY-HCMC UNIVERSITY
WEB APPLICATION DEVELOPMENT
MODULE CODE: ITCS25IU11
LECTURER: ASSOC. PROF NGUYỄN VĂN SINH**



**GROUP ASSIGNMENT:
CAR MANAGEMENT SYSTEM**

PROF : Nguyễn Văn Sinh

LAB ADVISER: Nguyễn Trung Nghĩa

GROUP MEMBERS:

Vũ Huỳnh Quốc Anh	ITITWE22003
Vũ Quốc Bảo	ITITWE22107
Phạm Thành Trung	ITITWE22170

Table of Contents

I. Home.....	3
 II. Car Module.....	3
 Login And Sign-Up.....	4
 Book Car and Booking Form.....	5
 Payment.....	7
VI. Booking Confirmation – Backend Code Explanation.....	8
 VII. Controller.....	
 9 7.1 Controller in Car Rental Backend System.....	9
 7.2 Role of the Controller.....	9
 7.3 Input Validation and Request Handling.....	9
 7.4 Coordination with Service Layer.....	9
 7.5 Response Management.....	9
 7.6 Security Enforcement.....	9
 VIII. Entities.....	10
 8.1 Car Entity.....	10
 8.2 User Entity.....	10
 8.3 Booking Entity.....	10
 8.4 Relationship Between Entities.....	11
IX. Conclusion.....	11

I. Home

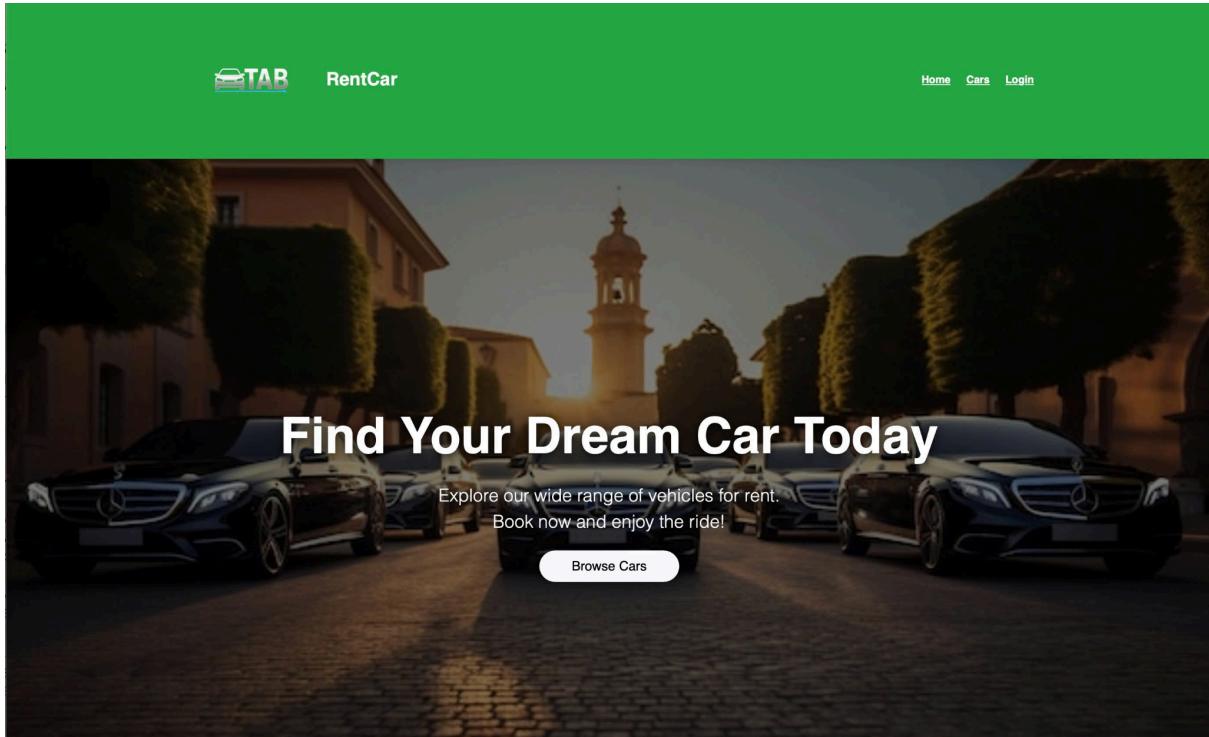
The Home module is the main entry point of the Car Rental Management System. It introduces users to the system and provides an overview of available services.

This module presents general information, promotional content, and navigation options to guide users toward other key functionalities.

Main responsibilities of the Home module include:

- Displaying system introduction and service description •

Providing navigation to Car, Login, Contact, Browse Car



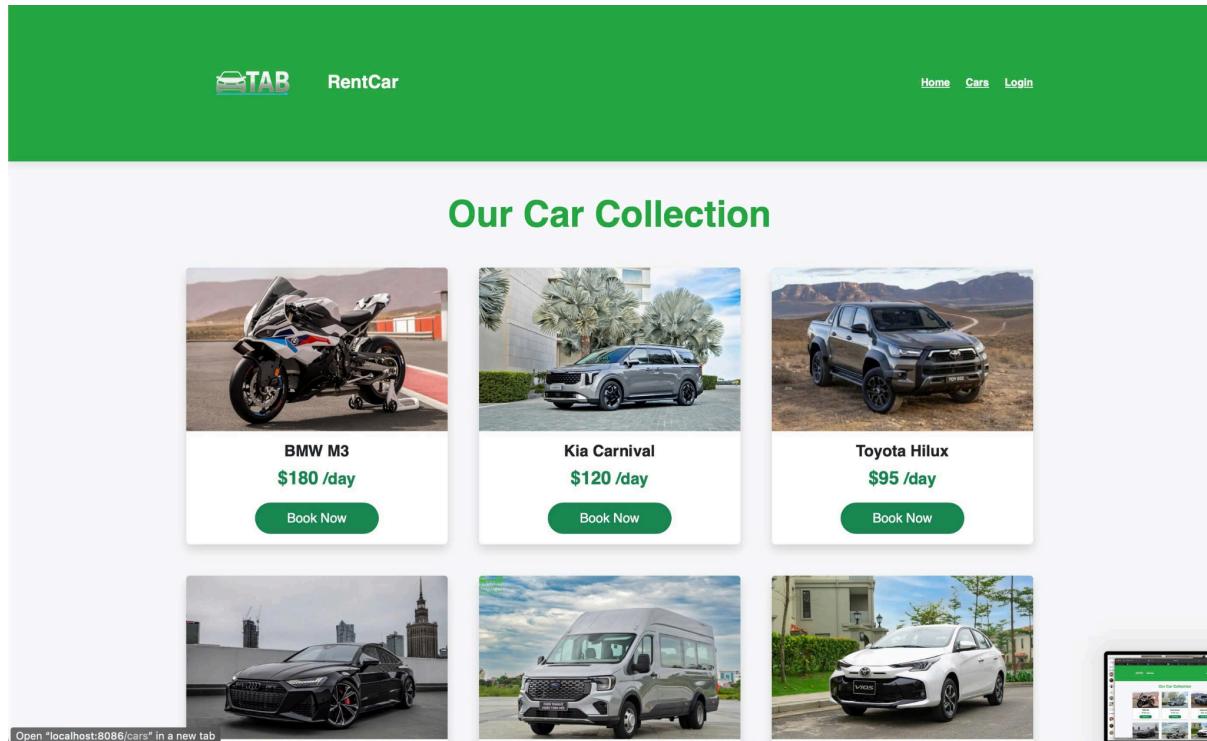
II. Car Module

The Car module manages all information related to vehicles available for rent. It allows users to browse, view, and compare cars before making a booking decision.

Key functionalities include:

- Displaying a list of available cars
- Showing detailed car information such as model, price, and availability

- Supporting filtering and sorting options



III. Login And Sign-Up

The Login module allows registered users to access the system securely.

Users must provide valid credentials to authenticate their identity.

Main features:

- Secure user authentication
- Error handling for invalid credentials
- Access control to protected system features

This module plays a critical role in protecting user data and system resources. The

Sign-Up module enables new users to create an account in the system. It collects

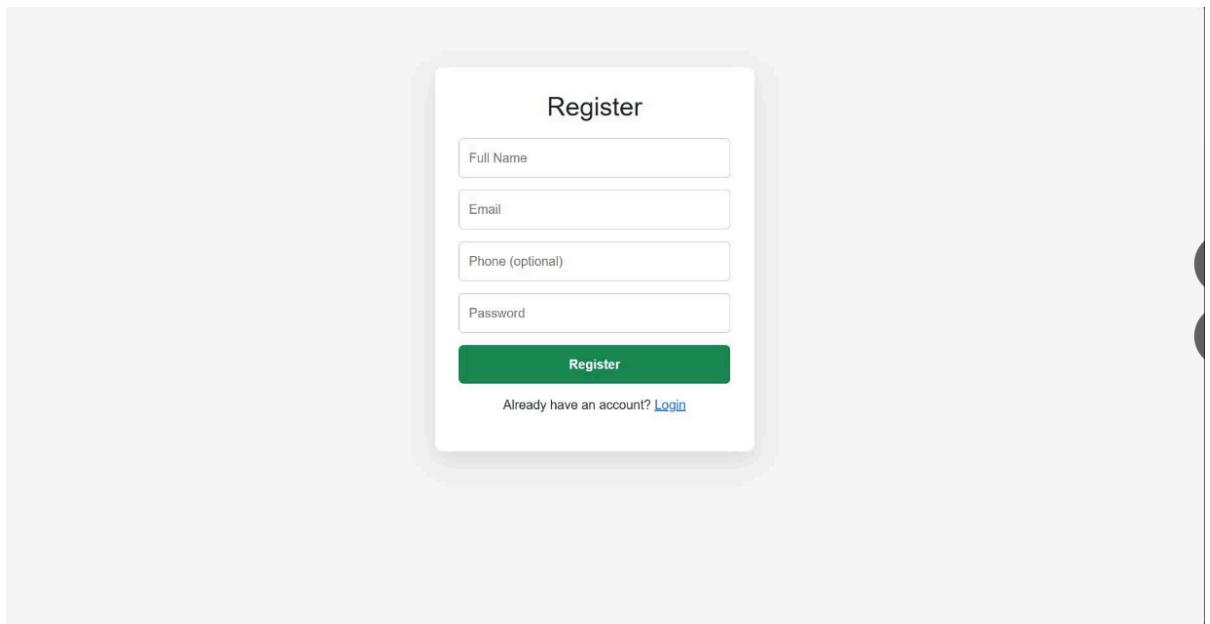
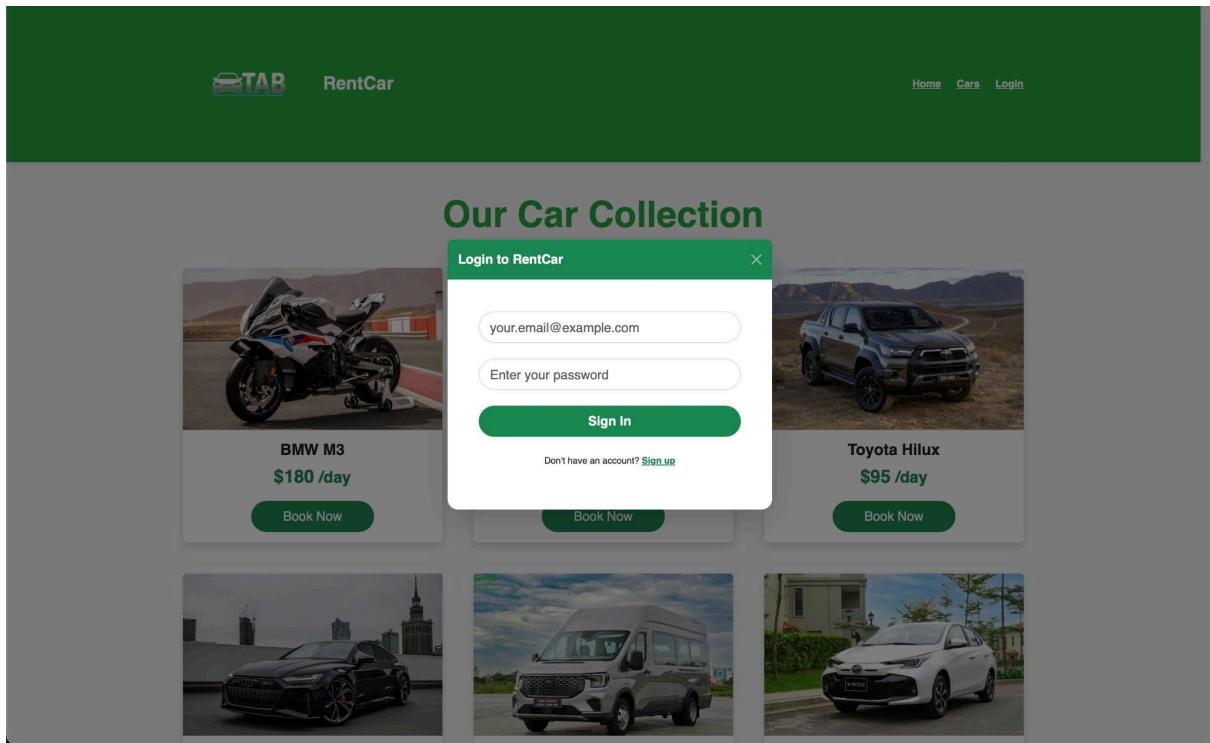
personal information and stores it securely in the database.

Main functions include:

- User registration form

- Input validation and duplicate account prevention
- Confirmation of successful account creation

This module supports system growth by allowing new users to join easily.



IV. Book Car and Booking Form

The Book Car module allows users to reserve a selected vehicle. It connects directly with the Car and Payment modules.

This module includes:

- Booking form for selecting rental dates
- Automatic cost calculation based on rental duration
- Availability verification before confirmation

Accurate data collection in this module helps prevent booking conflicts.

After the user submits the booking form, the form data is sent to the backend booking controller.

This data includes the selected car, rental dates, pickup location, and user information.

The backend first performs input validation to ensure that all required fields are provided and that the rental period is valid.

After validation, the system checks the availability of the selected car to avoid overlapping bookings.

If the car is available, a booking record is created with a **pending** status. At this stage, the booking is not confirmed yet and awaits payment processing.

V. Payment

The Payment module handles all financial transactions within the system. It ensures that payments are processed securely and accurately.

Key responsibilities include:

- Displaying total rental cost
- Processing online payments
- Confirming successful transactions

This module enhances user convenience and transaction reliability.

Once the booking form is successfully processed, the system moves to payment handling.

The backend receives the selected payment method and payment details from the user.

The system calculates the total rental cost based on the rental duration and the price of the selected car.

After that, the backend verifies the payment information and processes the transaction.

If the payment is successful, the booking status is updated from **pending** to **confirmed**.

The confirmed booking is then stored in the database and a confirmation response is returned to the user.

 Payment Method

Booking Summary			
Car Toyota Hilux	Price/Day \$95.0	Pick-up Date 2025-12-25	Drop-off Date 2025-12-26
Customer Thành Trung Phạm	Phone +84394674592	Pick-up Location Ba	Drop-off Location Ria
Email ttrung1326@gmail.com		Total Days 1	Total Price \$95.0

Additional Notes

 Cash Payment

You will pay in cash when you pick up the vehicle at our location. Please bring the exact amount or we can provide change.

Important Information

- Payment must be made in cash upon vehicle pick-up
- Please bring a valid driver's license
- A security deposit may be required

Confirm Booking

VI. Booking Confirmation – Backend Code Explanation

After the user submits the booking form, the request is sent to the backend booking controller.

The controller first extracts booking information such as car ID, rental dates, and user ID from the request.

Next, the system calls the car service to check whether the selected car is available during the requested period.

This is done by querying the booking data and validating that there is no overlapping reservation.

If the car is available, the backend creates a new booking object and calculates the total rental cost based on the rental duration and car price.

The booking data is then stored in the database with a status set to “**confirmed**”.

Finally, the backend returns a booking confirmation response containing the booking ID, car details, rental period, and total price.

This response is used to display the booking confirmation to the user or trigger further actions such as payment processing.

VII. Controller

7.1 Controller in Car Rental Backend System

The controller acts as the main entry point for all client requests in the Car Rental system.

It is responsible for receiving requests from the booking form, validating input data, and coordinating interactions between different backend components.

7.2 Role of the Controller

When a user submits a booking or selects a payment method, the request is first handled by the controller.

The controller extracts necessary information such as user ID, car ID, rental dates, and payment details from the request.

7.3 Input Validation and Request Handling

Before passing data to the business logic, the controller performs basic validation checks.

This includes verifying required fields, checking data formats, and ensuring that the request comes from an authenticated user.

Invalid requests are rejected immediately to protect backend resources.

7.4 Coordination with Service Layer

The controller does not implement business logic directly.

Instead, it delegates tasks to service components such as booking service, car service, or payment service.

This separation of concerns improves maintainability and code organization.

7.5 Response Management

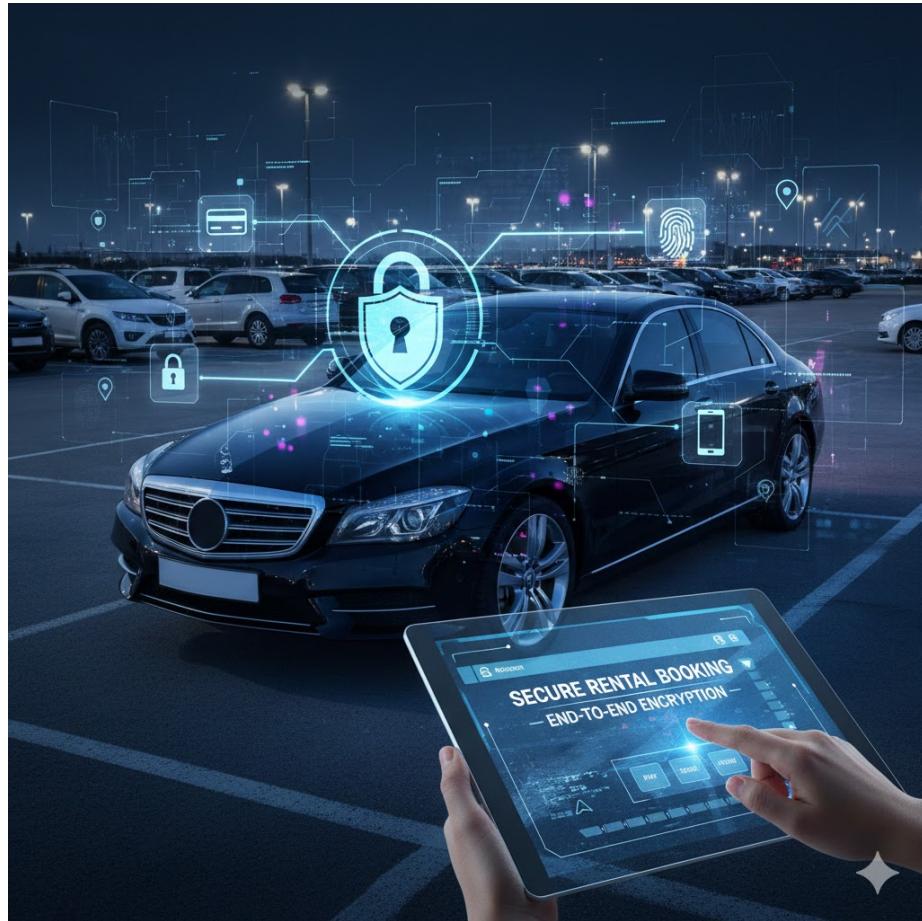
After receiving results from the service layer, the controller formats and returns a response.

For example, it may return booking confirmation details, payment status, or error messages.

7.6 Security Enforcement

The controller enforces security rules by ensuring that only authorized users can access booking and payment endpoints.

It also prevents unauthorized access to sensitive operations such as booking modification or cancellation.



VIII. Entities

8.1 Car Entity

The Car entity represents vehicles available for rental.

It contains attributes such as car ID, brand, model, price per day, and availability status.

This entity is used to determine which cars can be booked and to calculate rental costs.

8.2 User Entity

The User entity stores information about registered users.

It includes user ID, name, contact details, and authentication-related data. The backend uses this entity to identify users and control access to booking and payment functions.

8.3 Booking Entity

The Booking entity records rental transactions between users and cars. It contains booking ID, user ID, car ID, rental period, booking status, and total cost. This entity plays a central role in confirming reservations and preventing double bookings.

8.4 Relationship Between Entities

Entities are interconnected within the backend system. A user can create multiple bookings, each booking is linked to one car, and each booking is associated with a payment record. These relationships ensure data consistency and reliable backend operations.

IX. Conclusion

In conclusion, this project successfully demonstrates the design and implementation of a backend-oriented Car Rental system. The system effectively manages core functionalities such as car booking, payment processing, and data handling through a well-structured backend architecture.

By applying clear separation of concerns between controllers, services, and entities, the system ensures maintainability, scalability, and secure data processing. Backend validation, authentication, and availability checks play a crucial role in preventing invalid bookings and protecting user information.

Overall, the project highlights the importance of backend logic in building reliable rental systems and reflects a solid understanding of backend design principles relevant to real-world applications and academic study.