CAR RENTAL SYSTEM

A PROJECT REPORT

Submitted by

A K Goutham (220701077)

in partial fulfilment for the course

CS19542 – INTERNET PROGRAMMING

for the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

RAJALAKSHMI NAGAR

THANDALAM

CHENNAI – 602 105

TABLE OF CONTENTS

1. Introduction

1.1 Project Overview

The Car Rental System is a web-based application that allows users to rent cars from various locations. The system provides a dynamic and interactive user interface for browsing available cars, making bookings, and viewing rental history. Admins can manage car listings, view bookings, and update car availability.

1.2 Objective

The main objective of this project is to provide a complete car rental solution through a web application, with functionalities like car booking, rental history tracking, and an admin interface for managing car inventory.

1.3 Scope

The system includes user authentication, car search and booking features, and booking history. It also includes an admin dashboard for managing cars and bookings. The system is designed to be responsive, working on both desktop and mobile devices.

2. System Requirements

2.1 Hardware Requirements

- **Web Server**: Apache (via XAMPP)
- **Database Server**: MySQL (via XAMPP)
- **Operating System**: Windows, macOS, or Linux

2.2 Software Requirements

- **XAMPP**: Contains Apache server, MySQL, PHP, and PhpMyAdmin for database management.
- **PHP**: The server-side scripting language for managing sessions, database interactions, and processing user requests.
- **Bootstrap**: Front-end framework for creating a responsive layout.
- **Text Editor**: Such as VSCode or Sublime Text for coding and development.

2.3 Tools Used

- **PHP**: Handles server-side logic, user authentication, and database interaction.
- **MySQL**: Database management system to store user, car, and booking data.
- **JavaScript**: Enhances the user experience with interactive features.
- **Bootstrap**: A front-end framework for creating a responsive, modern user interface.

3. Database Design

3.1 Database Schema Overview

The Car Rental System uses a MySQL database with the following key tables:

- **Users Table**: Stores information about users (admins and regular users).
- **Cars Table**: Stores car details like availability, price, and location.
- **Bookings Table**: Stores car rental details, including user ID, car ID, and rental dates.

3.2 Relationships

- **Users to Bookings**: One-to-many relationship. A user can make multiple bookings.
- **Cars to Bookings**: One-to-many relationship. A car can be rented by multiple users over time.

3.3 Sample Queries

```
-- Fetch available cars by location
SELECT * FROM cars WHERE location = 'New York' AND availability = 'yes';

-- Fetch user's booking history
SELECT b.id AS booking_id, c.name AS car_name, b.start_date, b.end_date, b.total_price
FROM bookings b
INNER JOIN cars c ON b.car_id = c.id
WHERE b.user_id = 1;
```

4. System Architecture

4.1 Overview of System Components

The system is structured with the following components:

- **Frontend**: HTML, CSS, Bootstrap, and JavaScript for user interface development.
- **Backend**: PHP handles business logic, user authentication, and database interaction.

- **Database**: MySQL stores user data, car listings, and booking details.

4.2 Flow Diagram

The user interacts with the frontend (HTML, Bootstrap). Requests are sent to the backend (PHP), which processes the request, interacts with the MySQL database, and returns the result to the frontend.

User -> Frontend (HTML, CSS, JavaScript) -> Backend (PHP) -> Database (MySQL)

5. User Features

5.1 User Registration and Login

Users can register with basic details (name, email, password). After registration, they can log in to their account to manage their bookings and view rental history.

5.2 Searching for Cars

Users can search for cars based on location and rental dates. The available cars are displayed with their prices and a “Rent Now” button.

5.3 Car Booking

Once a user selects a car, they can choose the rental period (start and end dates). The system calculates the total price based on the selected dates and car price.

5.4 Booking History

Users can view their past rentals, which include car details, rental dates, and total price.

6. Admin Features

6.1 Admin Login

Admins can log in with special credentials to access the admin dashboard.

6.2 Managing Cars

Admins can add, edit, or delete car listings. They can also mark cars as available or unavailable based on their rental status.

6.3 Managing Bookings

Admins can view all bookings made by users, including rental details and payment information. Admins can update booking statuses (e.g., confirmed, completed).

7. Front-End Design

7.1 Layout

The layout uses Bootstrap to create a responsive design that works well on mobile and desktop devices. The homepage features a search form and a list of available cars.

7.2 Responsive Design

Bootstrap’s grid system ensures the layout adapts to various screen sizes. On larger screens, the available cars are displayed in a 3-column grid. On smaller screens, the cars are shown in a single-column layout.

8. Back-End Code

8.1 Core PHP Logic

PHP scripts handle:

- User login and session management.
- Database interactions for retrieving available cars, booking a car, and viewing booking history.
- Handling form submissions, including car search and rental dates.

8.2 Important PHP Functions

- mysqli_query(): Executes SQL queries.
- mysqli_fetch_assoc(): Fetches data as an associative array.
- password_hash(): Hashes passwords for secure storage.
- password_verify(): Verifies user passwords during login.

9. Security Considerations

9.1 User Authentication

Sessions are used to track user logins. The system ensures that only logged-in users can make bookings or view their rental history.

9.2 SQL Injection Prevention

Input data is sanitized using mysqli_real_escape_string() to prevent SQL injection vulnerabilities.

9.3 Password Hashing

User passwords are hashed using PHP's password_hash() function and verified using password_verify() during login.

10. Conclusion and Future Enhancements

10.1 Conclusion

The Car Rental System provides a dynamic and secure platform for users to rent cars. The system features user login, car search, and booking functionality. It also includes an admin panel for managing car listings and bookings.

10.2 Future Enhancements

Future improvements could include:

- Integration of a payment gateway (e.g., Stripe, PayPal) for online payments.
- A review system allowing users to leave feedback on cars.
- A mobile app version for better user accessibility.

10.3 Appendix

1. **Sample Database Schema**

```
CREATE TABLE users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100),
    email VARCHAR(100) UNIQUE,
```

```
    password VARCHAR(255),
    role ENUM('user', 'admin') DEFAULT 'user'
);

CREATE TABLE cars (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100),
    location VARCHAR(100),
    price_per_day DECIMAL(10, 2),
    image VARCHAR(255),
    availability ENUM('yes', 'no') DEFAULT 'yes'
);

CREATE TABLE bookings (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT,
    car_id INT,
    start_date DATE,
    end_date DATE,
    total_price DECIMAL(10, 2),
    FOREIGN KEY (user_id) REFERENCES users(id),
    FOREIGN KEY (car_id) REFERENCES cars(id)
);
```