DATABASE PROJECT REPORT

NAME: MUHAMMAD AFNAN SOOMRO

ROLL NO: K21-4846

RENT A CAR

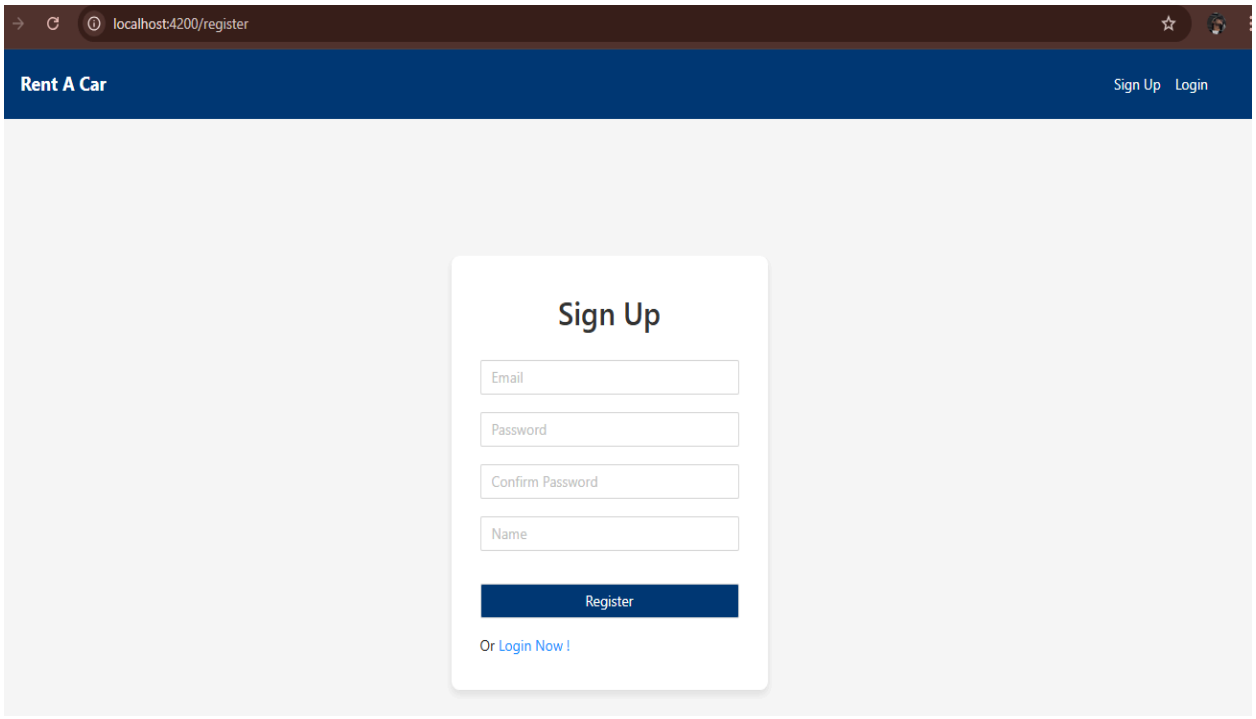
OVERVIEW

The **"Rent A Car"** project is a database-driven application designed for car rental services. It provides functionalities for two distinct user roles: **Admin** and **Customer**. The application is structured to facilitate smooth interaction between the service provider (admin) and users (customers) with features such as car management, booking, and user authentication.

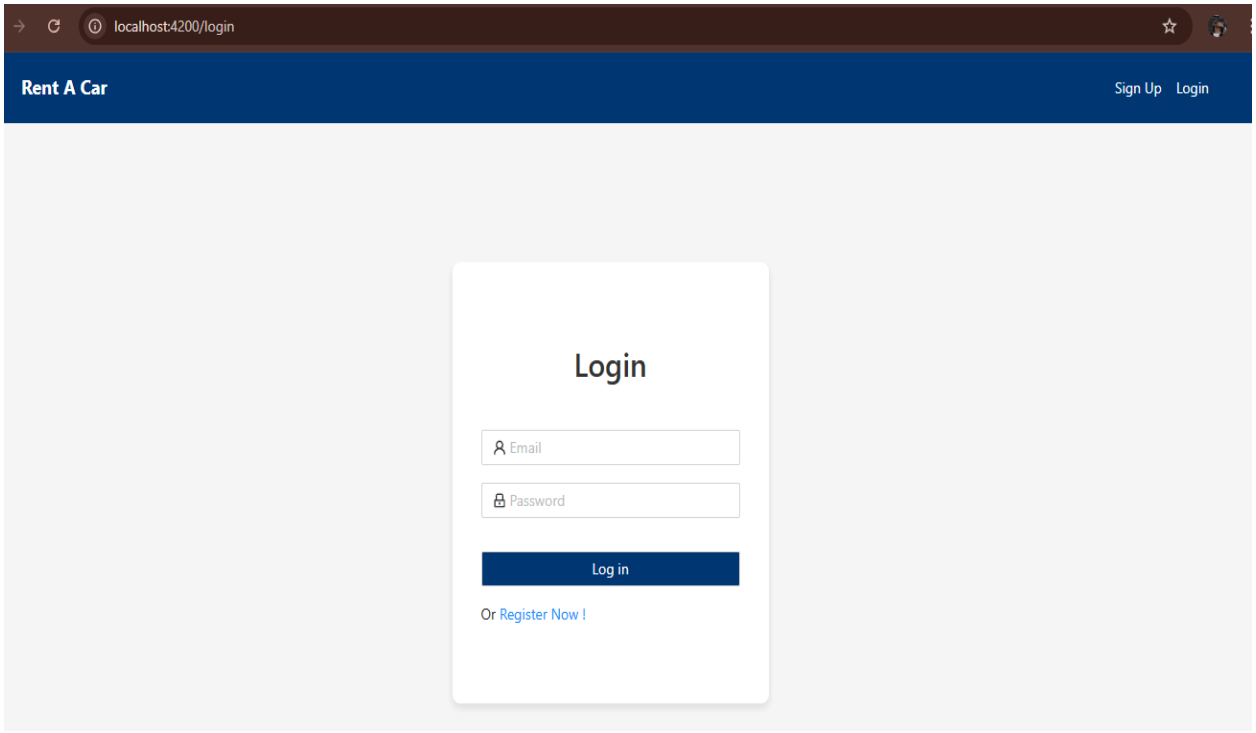
PROJECT FEATURES

AUTHENTICATION

1. **Signup Functionality**:
   Customers and admins can register their accounts with proper input validation for data integrity.



2. **Login Functionality**:
   Secure login with validation ensures that only authorized users can access the system.



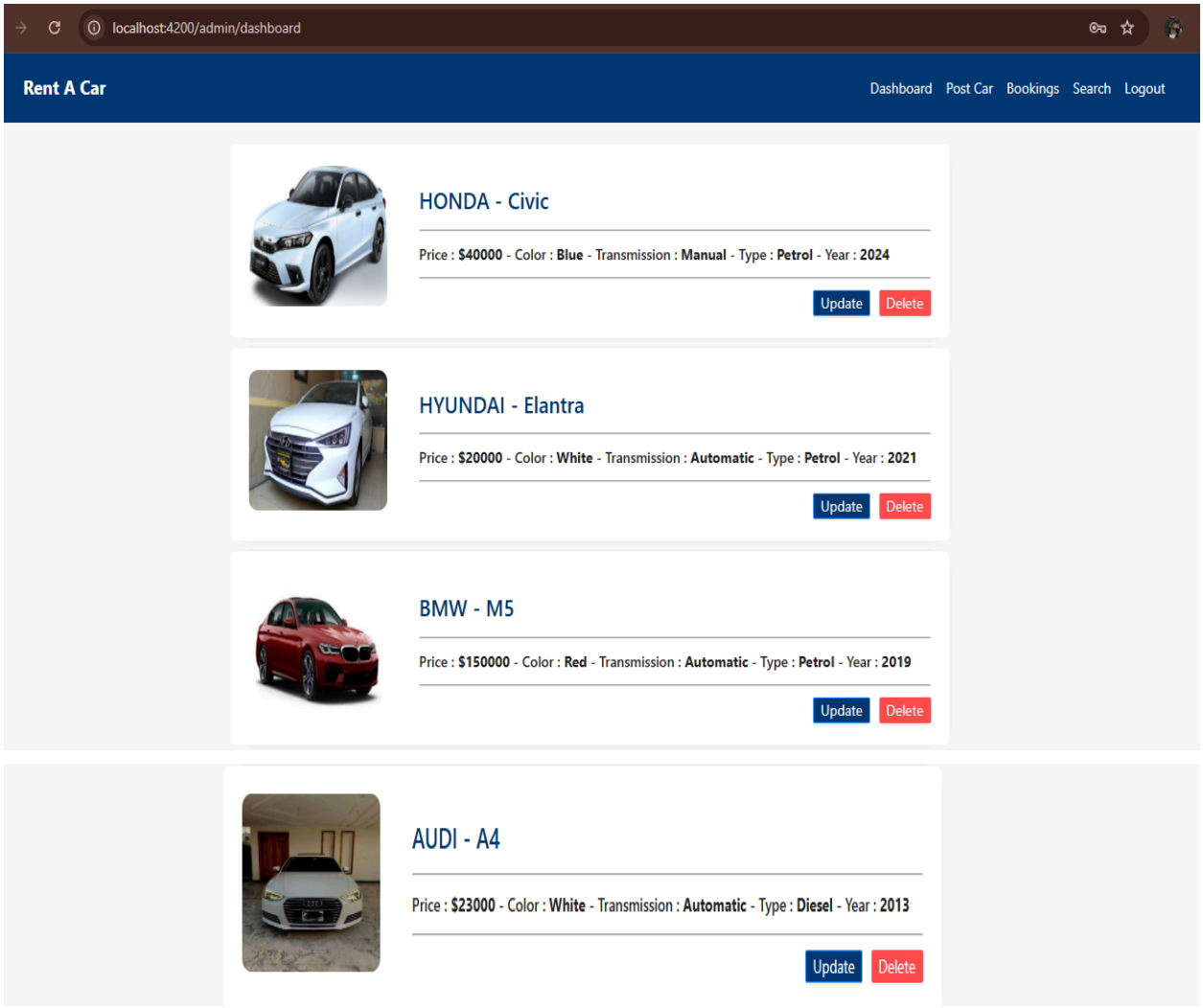
3. **Password Encryption:**
   Passwords are stored in encrypted form.

id	email	name	password
1	admin@gmail.com	Admin	$2a$10$e4whAHcbVTwfuZ4xPExiZeRd2HzpB6j1wr8Y8AslQZd22slEiPi4e
2	afnan@gmail.com	Afnan	$2a$10$a0wa9NG1arp0fyeOG4xDReZGwMYsAbMVyxY.7ZnDMgSRAMfab3uNK
3	ali@gmail.com	Ali	$2a$10$8gaeeu4fHusRDBP5Ew7JVOpS433QDI0Gbk0ZCde5lfKn06woN48l6
4	basit@gmail.com	Basit	$2a$10$FaeJCmAUnXa1FZ1/zSpLPOvPr5OAMC1xkYfIKsijGn/r6VggVigN6
5	ahmed@gmail.com	Ahmed	$2a$10$ynoXPxDJ5p.1/OQME3ik2esQzujCqrxNig7BUfJgKi9WYZwJHqUp2

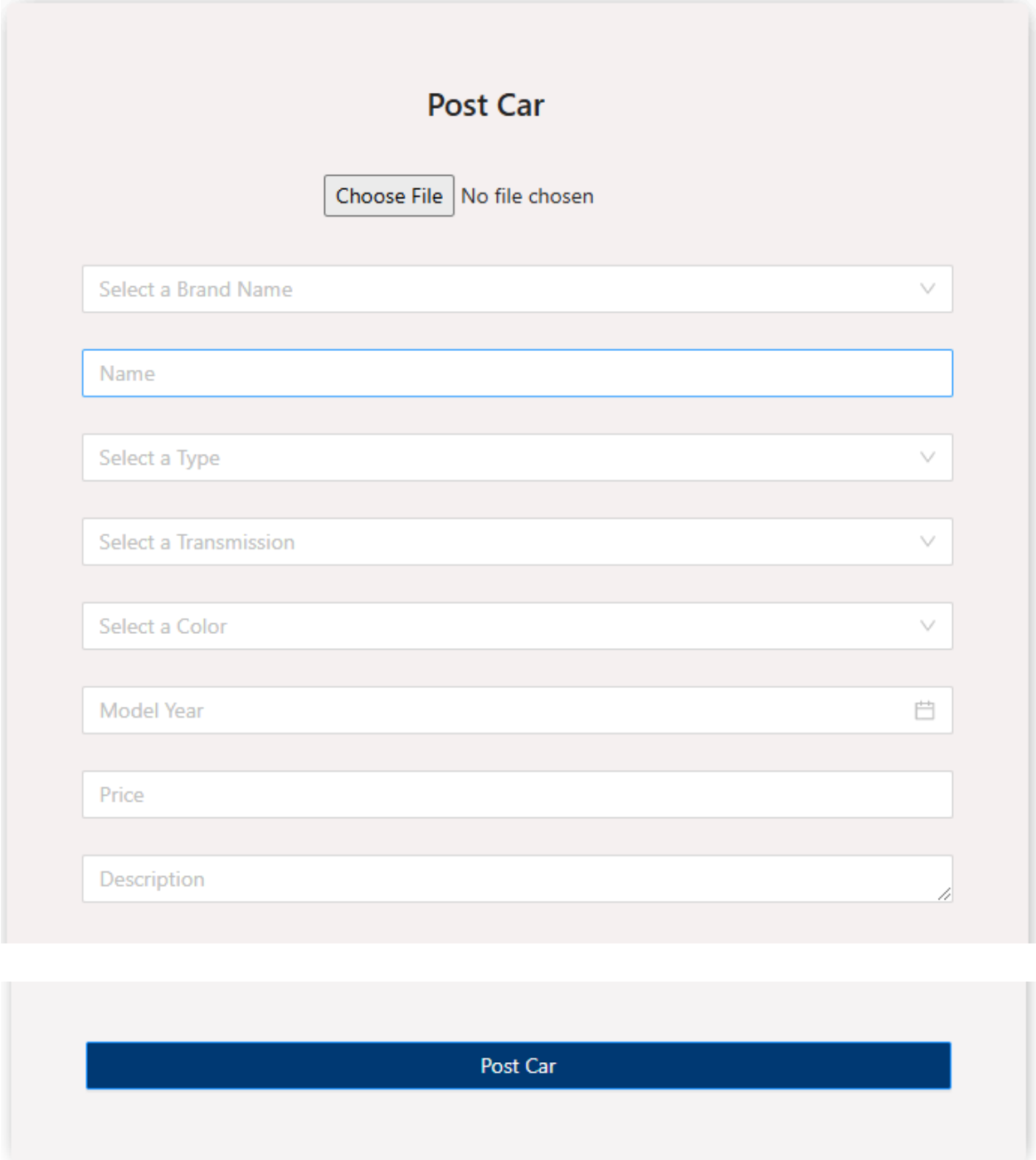
ADMIN PANEL

The admin panel is a dashboard designed to manage the car rental service efficiently. It includes the following features:

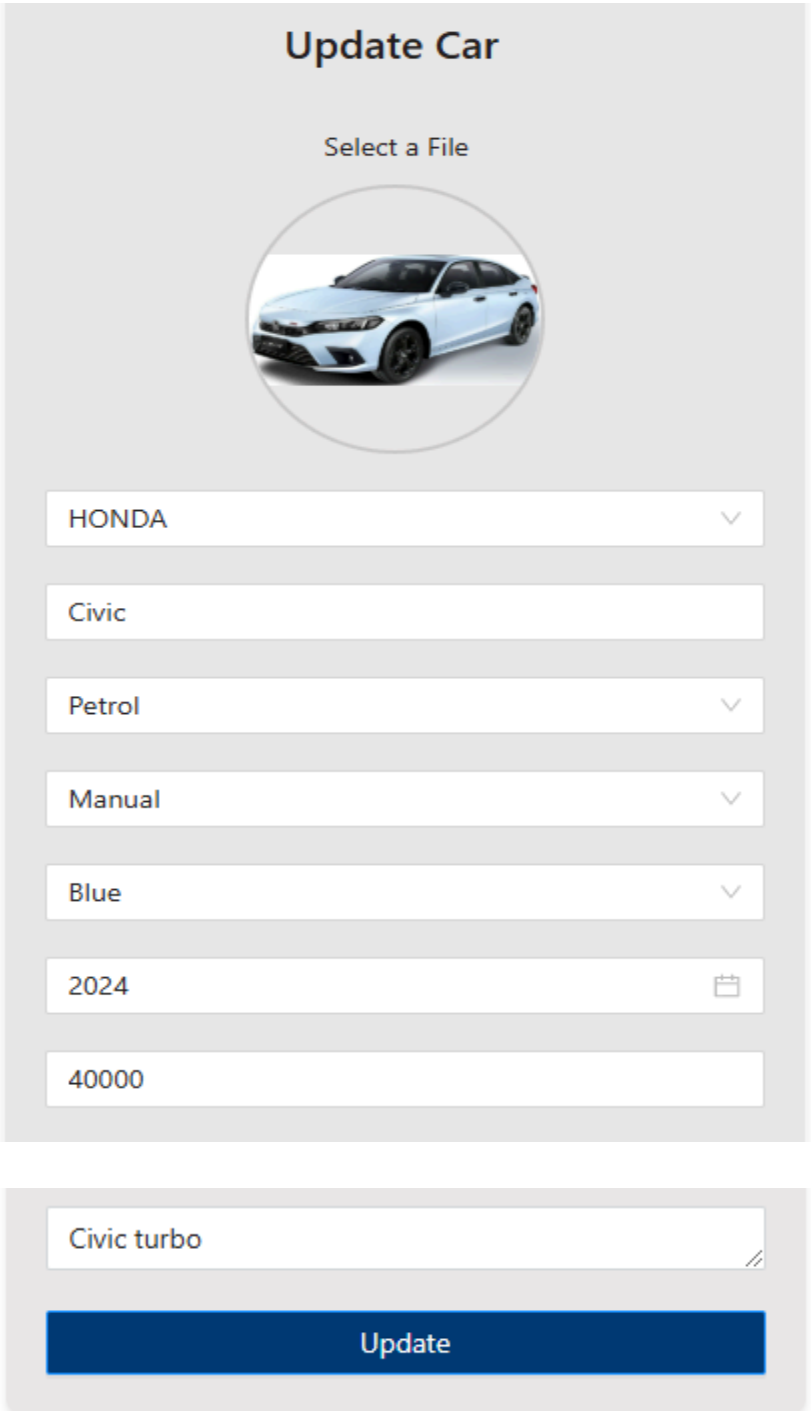
1. **View All Available Cars**:
   The admin can see a list of all cars currently available for booking.



2. **Post New Cars**:
   Admins can add details for new cars to the system, including specifications, rental cost, and availability.

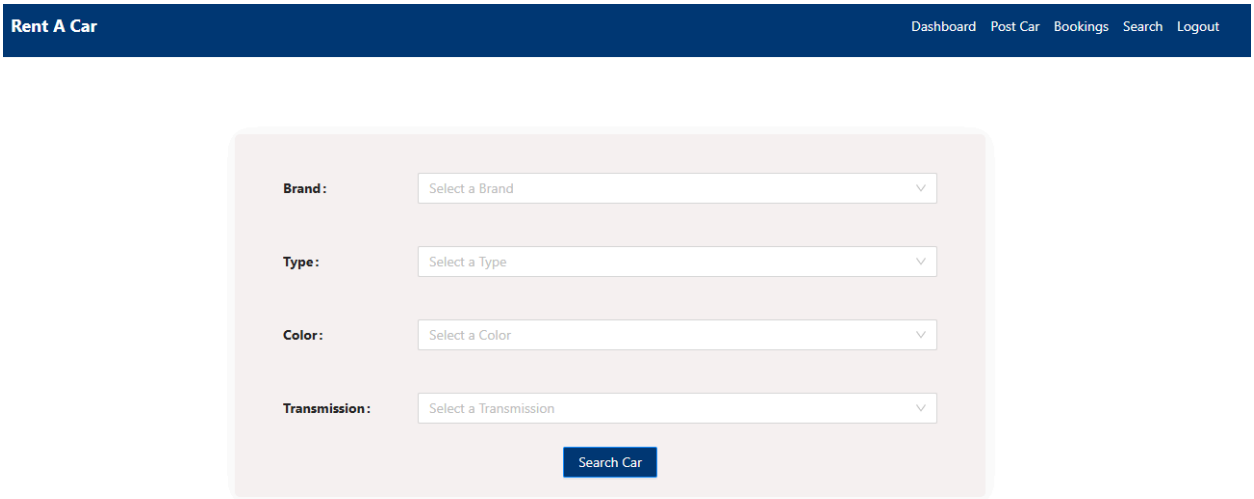


3. **Update Existing Cars**:
   Information about existing cars, such as price or availability, can be modified.

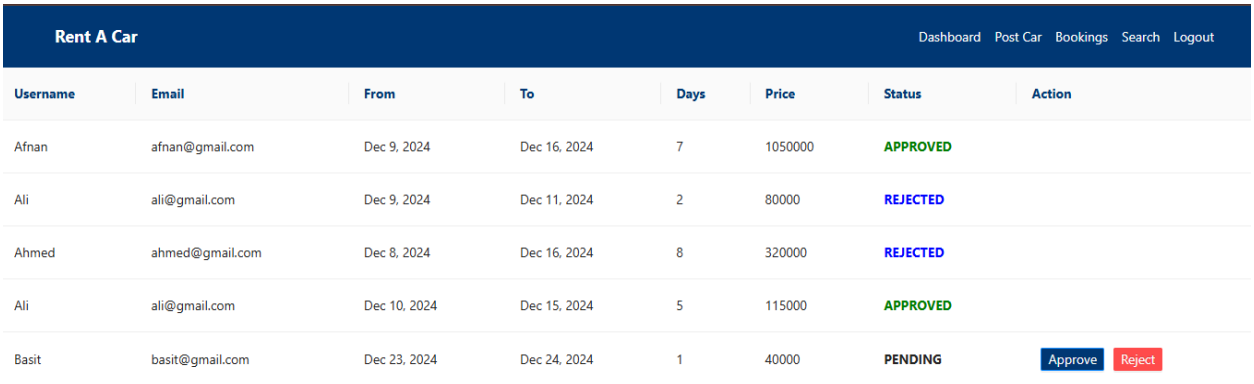


4. **Delete Cars**:
   Admins can remove cars that are no longer available for rental.

5. **Search Cars**:
   A search functionality allows admins to find cars quickly using filters like car type, brand, or rental price.



6. **View Car Bookings**:
   Admins can view all bookings, including pending, approved, and rejected requests.



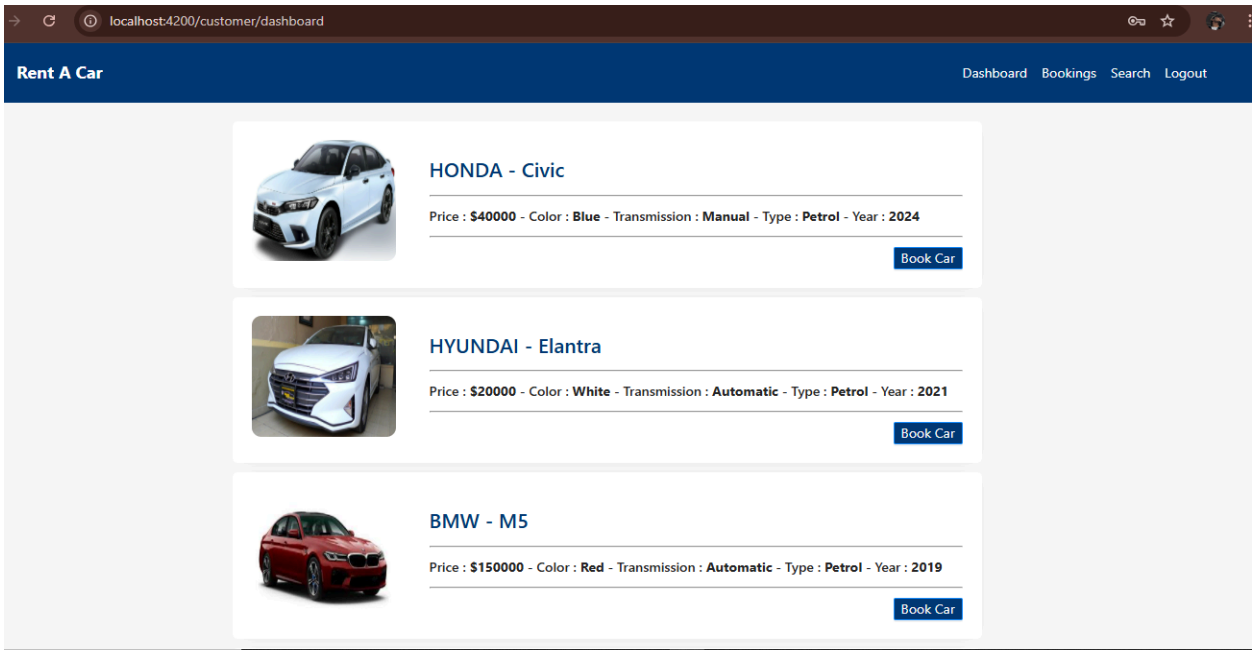
Username	Email	From	To	Days	Price	Status	Action
Afnan	afnan@gmail.com	Dec 9, 2024	Dec 16, 2024	7	1050000	APPROVED	
Ali	ali@gmail.com	Dec 9, 2024	Dec 11, 2024	2	80000	REJECTED	
Ahmed	ahmed@gmail.com	Dec 8, 2024	Dec 16, 2024	8	320000	REJECTED	
Ali	ali@gmail.com	Dec 10, 2024	Dec 15, 2024	5	115000	APPROVED	
Basit	basit@gmail.com	Dec 23, 2024	Dec 24, 2024	1	40000	PENDING	Approve Reject

7. **Manage Booking Requests**:
   Admins can approve or reject booking requests based on car availability.

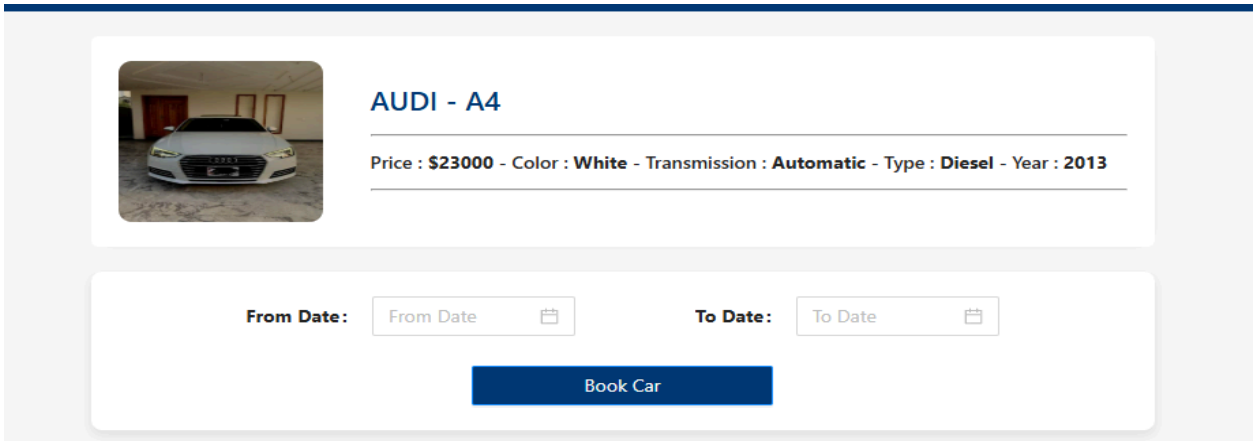
CUSTOMER PANEL

The customer panel is a user-friendly dashboard for customers to access the car rental services. It offers the following features:

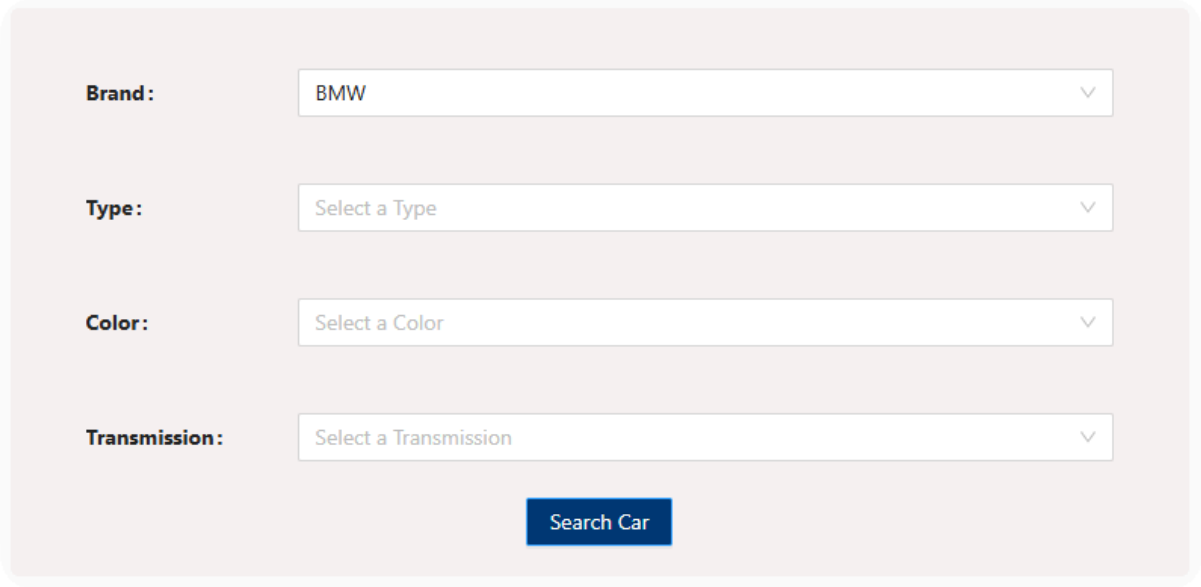
1. **View All Available Cars**:
   Customers can browse through the list of cars available for rent.



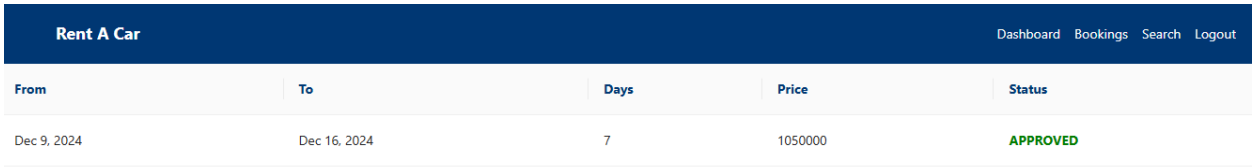
2. **Book a Car**:
   Customers can place booking requests by selecting a car and specifying the rental duration.



3. **Search Cars**:
   A search bar helps customers locate specific cars based on their preferences.



4. **View Past Bookings**:
   Customers can see a history of their previous bookings for reference or record-keeping.



ENTITY RELATIONSHIP DIAGRAM (ERD)

ENTITY	ATTRIBUTES	RELATIONSHIP
Users	user_id (primary key), email, name, password, user_role	One-to-Many with Bookings (user_id in Bookings)
Cars	car_id (primary key), brand, color, description, image, model_year, name, price, transmission, type	One-to-Many with Bookings (car_id in Bookings)
Bookings	booking_id (primary key), amount, booking_status, days, from_date, to_date, car_id (foreign key), user_id (foreign key)	Many-to-One with Users Many-to-One with Cars

NORMALIZATION

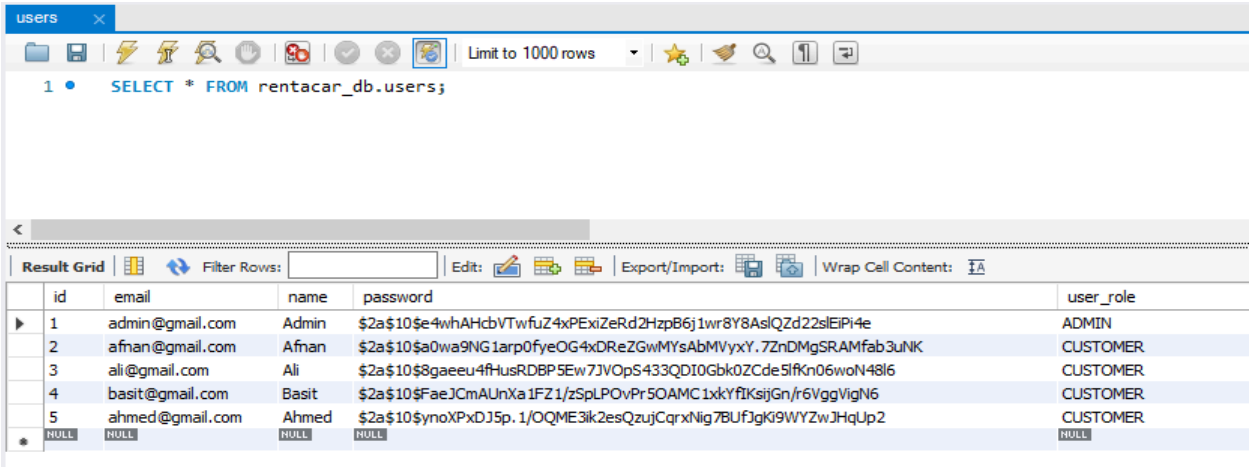
FIRST NORMAL FORM (1NF)

Each column should contain atomic value

USER ID	USER NAME	CAR ID	BRAND	BOOKING ID	STATUS	START DATE	END DATE

SECOND NORMAL FORM (2NF)

Remove partial dependency. Create separate tables for Users, Cars, and Bookings.

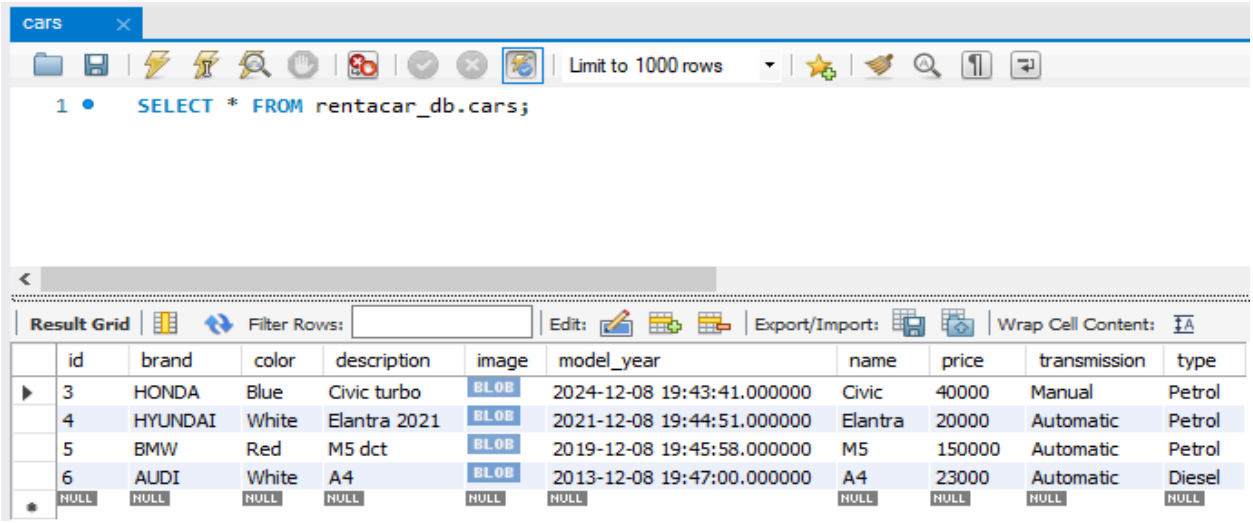


users

```
SELECT * FROM rentacar_db.users;
```

Result Grid

id	email	name	password	user_role
1	admin@gmail.com	Admin	$2a$10$e4whAHcbVTwfuZ4xPExiZeRd2HzpB6j1wr8Y8AslQZd22slEiPi4e	ADMIN
2	afnan@gmail.com	Afnan	$2a$10$a0wa9NG1arp0fyeOG4xDReZGwMYsAbMVyxY.7ZnDMgSRAMfab3uNK	CUSTOMER
3	ali@gmail.com	Ali	$2a$10$8gaeeu4fHusRDBP5Ew7JVOpS433QDI0Gbk0ZCde5lfKn06woN48l6	CUSTOMER
4	basit@gmail.com	Basit	$2a$10$FaeJCmAUnXa1FZ1/zSpLPOvPr5OAMC1xkYfIKsijGn/r6VggVigN6	CUSTOMER
5	ahmed@gmail.com	Ahmed	$2a$10$ynoXPxDJ5p.1/OQME3ik2esQzujCqrxNig7BUfJgKi9WYZwJHqUp2	CUSTOMER

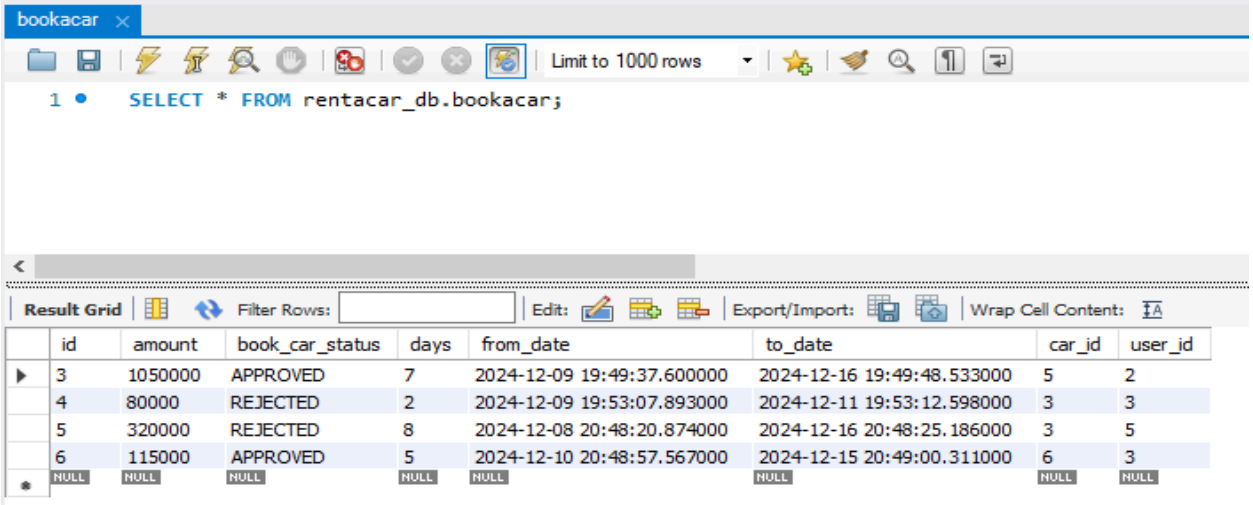


cars

```
SELECT * FROM rentacar_db.cars;
```

Result Grid

id	brand	color	description	image	model_year	name	price	transmission	type
3	HONDA	Blue	Civic turbo	BLOB	2024-12-08 19:43:41.000000	Civic	40000	Manual	Petrol
4	HYUNDAI	White	Elantra 2021	BLOB	2021-12-08 19:44:51.000000	Elantra	20000	Automatic	Petrol
5	BMW	Red	M5 dct	BLOB	2019-12-08 19:45:58.000000	M5	150000	Automatic	Petrol
6	AUDI	White	A4	BLOB	2013-12-08 19:47:00.000000	A4	23000	Automatic	Diesel



bookacar

```
SELECT * FROM rentacar_db.bookacar;
```

Result Grid

id	amount	book_car_status	days	from_date	to_date	car_id	user_id
3	1050000	APPROVED	7	2024-12-09 19:49:37.600000	2024-12-16 19:49:48.533000	5	2
4	80000	REJECTED	2	2024-12-09 19:53:07.893000	2024-12-11 19:53:12.598000	3	3
5	320000	REJECTED	8	2024-12-08 20:48:20.874000	2024-12-16 20:48:25.186000	3	5
6	115000	APPROVED	5	2024-12-10 20:48:57.567000	2024-12-15 20:49:00.311000	6	3

THIRD NORMAL FORM (3NF)

Remove transitive dependency. Ensure non-key attributes depend only on the primary key.

Users: No change required (attributes directly dependent on user_id).

Cars: No change required (attributes directly dependent on car_id).

Car Bookings: No change required (attributes directly dependent on booking_id).

TECHNOLOGY AND TOOLS

1. **Backend**:
   The backend is developed using **Spring Boot Maven** in **Java**, providing a robust and scalable API for all functionalities.

2. **Frontend**:
   The frontend uses **Angular** with **Ng-Zorro** components for a responsive and modern user interface. Development is performed in **Visual Studio Code**.

3. **Database**:
   MySQL is used as the database management system, designed using **MySQL Workbench** to ensure structured data storage and efficient retrieval.

CONCLUSION

The **"Rent A Car"** project is a comprehensive beginner-level application that demonstrates the effective integration of frontend, backend, and database technologies to manage a car rental system. The project successfully implements essential features such as role-based user access, car management, and booking functionalities, showcasing a complete lifecycle of data handling and user interaction.

Through the development process, key skills were learned and applied, including:

- **Backend Development** with Spring Boot for creating RESTful APIs.
- **Frontend Development** with Angular and Ng-Zorro for a responsive and user-friendly interface.
- **Database Design** with MySQL for structured data storage and retrieval, ensuring normalization for data integrity and efficiency.

The project emphasizes:

1. **Role-Based Functionality**: Separate modules for admins and customers ensure a clear distinction of access and actions.
2. **Data Integrity and Validation**: Both frontend and backend validations prevent unauthorized or invalid data entry.
3. **CRUD Operations**: Comprehensive implementation of Create, Read, Update, and Delete functionalities.
4. **User Authentication**: Secure login and signup mechanisms to protect user data and ensure privacy.

This project is a practical learning experience for beginners, covering user authentication, role-based functionality, and CRUD operations. Future extensions could include features like payment integration, car reviews, or location-based car searches.