

Database Report

Made by:

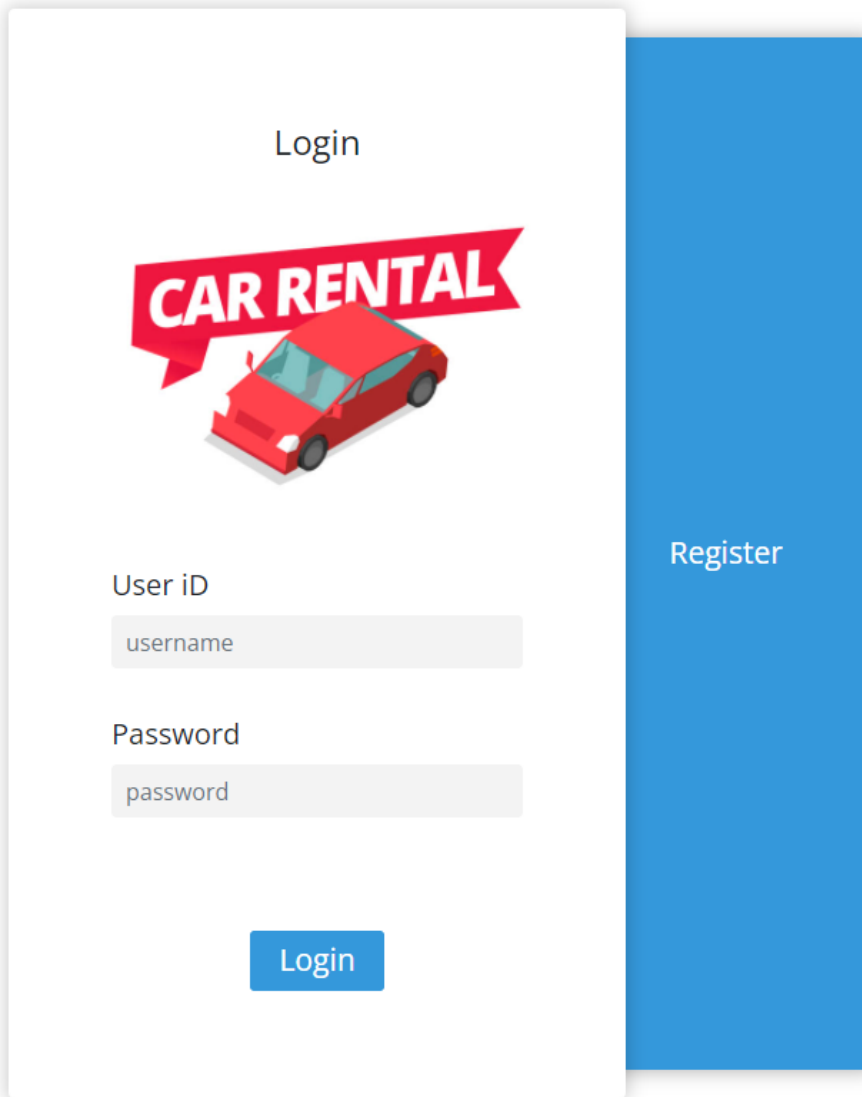
Abdallah Ashraf 6365

Islam Wael 6373

Aly Zaitoon 6368

GUI:

1) Login Page:



The image shows a login page for a car rental system. It features a white login form on the left and a blue register sidebar on the right. The login form includes a title 'Login', a 'CAR RENTAL' logo with a red car, and input fields for 'User iD' (username) and 'Password' (password). A blue 'Login' button is at the bottom of the form. The blue sidebar contains a white 'Register' button.

Login

CAR RENTAL

User iD

username

Password

password

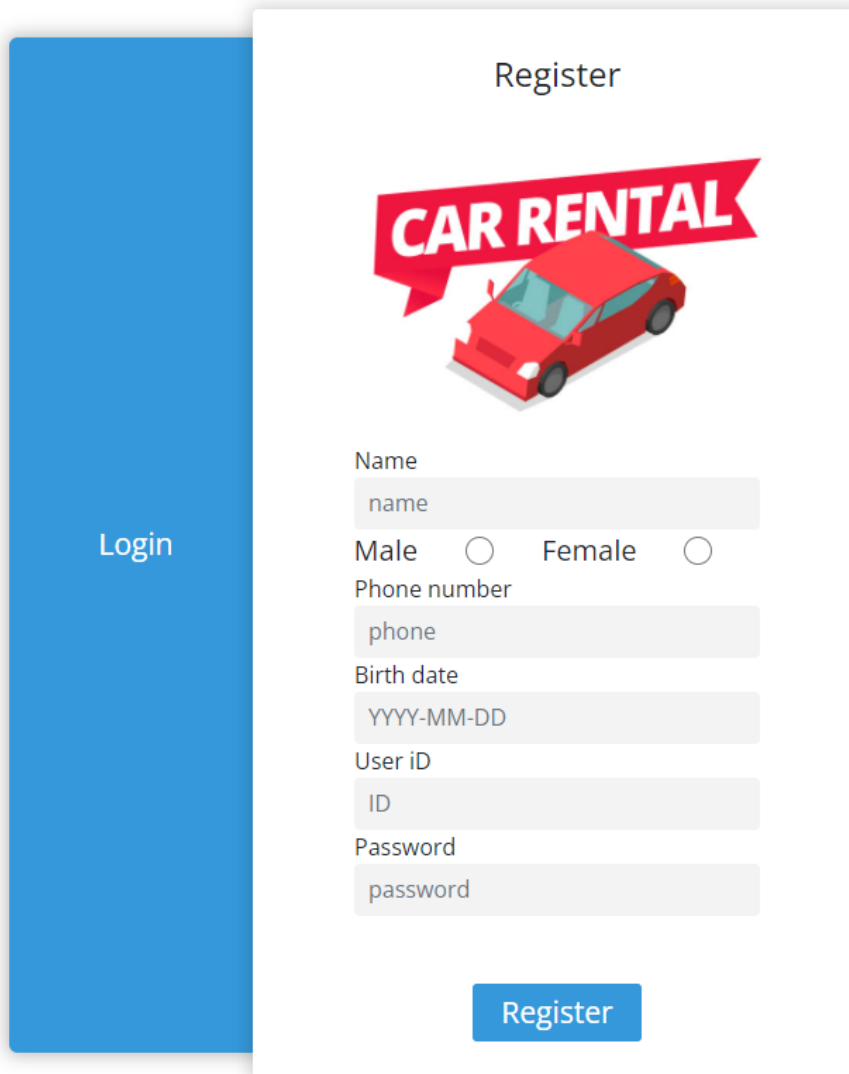
Login

Register

This page talks about a login page where the user needs to write his user ID and password in order to proceed to the next window. The user may be a customer or admin. If the user is a customer, he will be directed to a reservation for a car page. If the user is admin, he will be directed to the admin page where he can configure the data from the database and check for daily reports. After putting user ID and password and pressing login , the website connects to the database and checks if the user exists. If it fails then an alert message will display that the user is not valid. If the user

omits user ID or password , an alert message will display that there is a field required.

2) Register Page:



The image shows a user registration form titled "Register" for a "CAR RENTAL" service. On the left is a blue vertical sidebar with the word "Login" in white. The main form area is white and contains the following elements:





- Title:** Register
- Image:** A red car with a red banner above it that says "CAR RENTAL".
- Form Fields:**
 - Name:** A text input field with the placeholder "name".
 - Gender:** Two radio buttons labeled "Male" and "Female".
 - Phone number:** A text input field with the placeholder "phone".
 - Birth date:** A text input field with the placeholder "YYYY-MM-DD".
 - User iD:** A text input field with the placeholder "ID".
 - Password:** A text input field with the placeholder "password".
- Button:** A blue button labeled "Register" at the bottom.

If a user does not have an account , he will press the register button to redirect to the registration page. The user must fill in the required fields in order to create an account. When a user successfully fills the form, the website connects to the database and inserts all the values in the fields in the database. However, if the user writes a user ID which is included in the database an alert message will appear showing that user ID already exists. Omitting any fields will display an alert message. After successful

registration, the user will be directed to the login page again and fill in the user ID and password that he wrote in the register page. The registered user will always be the type of customer.

3) Customer Pages:

Reserving a car:

Sort By Price	Reservation Info	Type	Model	Price	Year	Color	Logout
	Volex Wagen Passat Black 2022-01-01					\$1500 Reserve	
	BMW x6 red 2020-08-09					\$2000 Reserve	
	Ferrari F8 Spider Yellow 2022-01-01					\$5000 Reserve	
	huandai i10 Grey 2015-09-01					\$500 Reserve	

The customer will be directed to reserving a car , the customer must select what cars he likes with its specs. The customer also can use filters to get the required car more conveniently. The customer can also search the required car by typing its name and the car will show up. After selecting the car needed, he should press the reserve button and he is directed to the pick up and return dates. If the customer wants to pay for reserving a car he will press on the payment button. If the customer did not like anything he could simply logout from his account to redirect from the login page.

Pick up and return dates:

Pick-up Date

Pick-up Date

yyyy-mm-dd

Return Date

yyyy-mm-dd

Submit

Go Back

After the customer selects a reserve for the car he liked to reserve, another page will open . The user must select the pick up date and return date. Then, the website connects to the database and inserts the customer ID and the car he reserved and the amount he needed to be paid on the car. If the customer selected the return date less than pick up date an alert message will appear showing that the date picked in return date is less than pick up date.

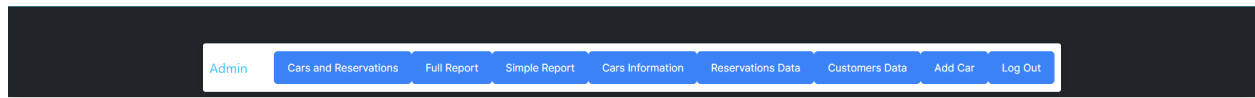
Reservations Information:

Reservation_no	User ID	Plate ID	Pickup date	Return date	Status	Payment	Pay
23	77	122345	2022-10-12	2022-10-14	reserved	3000	<div>Return and Pay</div>

Go Back

After the customer pressed the pay button , a table will appear showing all his reservations and for each row there is a pay and return button in order to pay the reservation and return the car. If he does so, then the payment will be zero and the row will be deleted from the database and the page itself.

4) Admin page:



When admin logs in he will be directed to a page where there is a navigation bar. The navigation bar contains adding new cars, viewing reservations, all cars in the market itself including out of service cars, customers data, all cars reserved by customer, a full report that consists of all everything, a simple report that consists of customer information and car model and plate id. There is a log out button of admin which is in the navigation bar that redirects admin back to the login page.

Adding a new car:

A screenshot of a web form titled 'New Car'. The form has several input fields with labels: 'Plate ID' (with placeholder 'Enter plateld'), 'Type' (with placeholder 'Enter type'), 'Model' (with placeholder 'Enter model'), 'Color' (with placeholder 'Enter color'), 'Year' (with placeholder 'Enter year YYYY-MM-DD'), 'Status' (with placeholder 'Enter status'), and 'Rate' (with placeholder 'Enter rate'). At the bottom of the form, there is a 'Choose File' button, a text indicator 'No file chosen', and a blue 'Add Car' button.

Admin can create a new car after clicking the add car button in the navigation. The page includes all the car information needed to insert this car in the database. After filling in the form, the website connects to the database and checks whether the new car is already in the list. If the plate id is the same as one of the existing cars it will display an alert message displaying that another car has the same plate id. If the admin omitted a

field then an alert message will appear that the omitted field needs to be filled. The admin can also put a picture to the car so it can be displayed on reserving a car.

Reservations Data:

Admin Cars and Reservations Full Report Simple Report Cars Information Reservations Data Customers Data Add Car Log Out						
Reservation no	User ID	Plate ID	Pickup date	Return date	Status	Payment
23	77	122345	2022-10-12	2022-10-14	reserved	3000
Filter comparator Enter Reservation no Enter Reservation no...	Filter by User ID Enter User ID...	Filter by Plate ID Enter Plate ID...	Filter comparator Enter \$Pickup date mm/dd/yyyy	Filter comparator Enter \$Return date mm/dd/yyyy	Filter by Status Enter Status...	Filter comparator Enter Payment Enter Payment...

If the admin wants to know all reservations he can press the reservations data button. The page displays a table which includes customer id and plate id with its returning date and pick up date and so on. If the admin wants to check for a specific customer id then he can filter and write the customer id in the filter section. The admin can use multiple searches to find a specific reservation. The admin can also edit, sort and filter any other data he needs.

Customers Data:

Admin Cars and Reservations Full Report Simple Report Cars Information Reservations Data Customers Data Add Car Log Out					
Customer ID	Name	Gender	Birth date	Phone number	
0	islam wael	M	1999-12-23	+201128040044	
7	ziad	M	1999-06-14	01099115820	
8	mariam	F	1999-06-10	01115834798	
77	islam wael	M	1999-10-23	+201128040044	
Filter by Customer ID Enter Customer ID...	Filter by Name Enter Name...	Filter by Gender Enter Gender...	Filter comparator Enter \$Birth date mm/dd/yyyy	Filter by Phone number Enter Phone number...	

If the admin wants to know a specific customer information then he can press the customers data button. The page displays a table which includes all customer data. By getting a specific customer, the admin can filter by writing his name on the name column to get the customer name and all his details. The admin can also change any customer information at any time.

Cars Information:

Admin Cars and Reservations Full Report Simple Report Cars Information Reservations Data Customers Data Add Car Log Out							
Plate ID	Type	Model	Color	Year	Status	Rate per day	
122345	Volex Wagen	Passat	Black	2022-01-01	active	1500	
1589	BMW	x6	red	2020-08-09	active	2000	
354	Ferrari	F8 Spider	Yellow	2022-01-01	Active	5000	
7848	huandai	i10	Grey	2015-09-01	active	500	
7896	Volvo	Xc30	Black	2021-05-29	active	1800	
Filter by Plate ID Enter Plate ID...	Filter by Type Enter Type...	Filter by Model Enter Model...	Filter by Color Enter Color...	Filter comparator Enter \$Year mm/dd/yyyy	Filter by Status Enter Status...	Filter comparator Enter Rate per day...	Enter Rate per day...

If the admin wants to know a specific car information then he can press the cars information button on the navigation bar. The page displays a table which includes all information on each car. By getting a specific car , the admin can filter by writing its model or plate ID to get all car details. The admin can also change any car information at any time.

Cars and Reservations:

Admin Cars and Reservations Full Report Simple Report Cars Information Reservations Data Customers Data Add Car Log Out													
Plate ID	Type	Model	Color	Year	Status	Rate per day	Reservation no	User ID	Pickup date	Return date	Status	Payment	
122345	Volex Wagen	Passat	Black	2022-01-01	reserved	1500	23	77	2022-10-12	2022-10-14	reserved	3000	
Filter by Plate ID Enter Pla	Filter by Type Enter Tyj	Filter by Model Enter Mh	Filter by Color Enter C	Filter comparator Enter \$Year mm/dd/yyyy	Filter by Status Enter Stat	Filter comparator Enter Rate per day Enter Rate per	Filter comparator Enter Reservation no Enter Reservati	Filter by User ID Enter U	Filter comparator Enter \$Pickup date mm/dd/yyyy	Filter comparator Enter \$Return date mm/dd/yyyy	Filter by Status Enter Stat	Filter comparator Enter Payment Enter Payment.	

If the admin wants to know all the cars reserved then he can press the cars and reservations button on the navigation bar. The page displays a table which includes all information about reservation number, plate id and so on. By getting a specific reservation, the admin can filter by writing plate ID or reservation number to get the required reservation the admin needs. The admin can also change any reservations at any time.

Full Report:

Admin Cars and Reservations Full Report Simple Report Cars Information Reservations Data Customers Data Add Car Log Out																
User ID	Name	Gender	Birth date	Phone number	Reservation no	Plate ID	Type	Model	Color	Year	Status	Rate per day	Pickup date	Return date	Status	Payment
77	islam wael	M	1999-10-23	+201128040044	23	122345	Volcx Wagen	Passat	Black	2022-01-01	active	1500	2022-10-12	2022-10-14	reserved	3000
Filter by User ID Enter En	Filter by Name Ente	Filter by Gender Enter	Filter comparator Enter \$Birth date mm/dd/yyyy	Filter by Phone number Enter Phone	Filter comparator Enter Reservation no Enter R	Filter by Plate ID Enter	Filter by Type Ente	Filter by Model Ente	Filter by Color En	Filter comparator Enter \$Year mm/dd/yyyy	Filter by Status Ente	Filter comparator Enter Rate per day Enter R	Filter comparator Enter \$Pickup date mm/dd/yyyy	Filter comparator Enter \$Return date mm/dd/yyyy	Filter by Status Enter	Filter comparator Enter Payment Enter P

If the admin wants to know every detail in reservation the he can press the full report button. The page displays a table which includes all information about cars and customers in each reservation. The admin can only view the report and it cannot be edited or can be sorted but it can be filtered.

Simple Report:

Admin														
Cars and Reservations			Full Report		Simple Report		Cars Information		Reservations Data		Customers Data		Add Car	Log Out
User ID	Name	Gender	Birth date	Phone number	Reservation no	Plate ID	Model	Pickup date	Return date	Status	Payment			
77	islam wael	M	1999-10-23	+201128040044	23	122345	Passat	2022-10-12	2022-10-14	reserved	3000			
Filter by User ID	Filter by Name	Filter by Gender	Filter comparator	Filter by Phone number	Filter comparator	Filter by Plate ID	Filter by Model	Filter comparator	Filter comparator	Filter by Status	Filter comparator			
Enter	Enter	Enter	Enter \$Birth date	Enter Phone n	Enter Reservation no	Enter	Enter	Enter \$Pickup date	Enter \$Return date	Enter S	Enter Payment			
			mm/dd/yyyy		Enter Rese			mm/dd/yyyy	mm/dd/yyyy		Enter Pay			

If the admin wants to know customer information and plate ID and model of car only he can press the simple report button. The page displays a table which includes all customer information about each customer and plate ID and model of the car. The admin can only view the report and it cannot be edited or can be sorted but it can be filtered.

Backend

We implemented the backend for our car rental system using a spring boot application and we implemented a REST API providing all the functionalities that the front need(supplying the front end with all data needed by axios ajax calls), connect to database (using jdbc) and retrieving data from database.

In Our Backend we have 3 main sets of classes

1. Entities

This set of classes contains all the database entities with its corresponding attributes, Objects of these classes act as a buffer in memory between frontend and database , we construct these objects by POST calls to be written to the database and we send them to frontend by GET calls to show info in frontend. For example we have an entity called admin which corresponds exactly to the admin table in the database.

2. Queries

This set of classes contains functions that executes all the queries we need in SQL of insert , update and delete and also it applies complex queries like joining multiple tables to give us the facility to gather more data in a single table and be able to publish reports to the admin portal.

2.1 Examples

- `select * from customer as c1 join Reservation as r1 on c1.customer_id=r1.user_id join car as c2 on r1.plate_id = c2.plate_id;`

This query gets a full report of all reservations with the customers who made it and the cars they reserved

- `"update reservation set payment=? where reservation_number=reservation_number";`

This query updates reservation record according to its reservation number (we use it to set payment to 0 when customer pays for its reservation

3. Controllers

In our implementation we had one Controller Class which represents the API and this class has instances from the Queries classes and executes them according to the received mapping (url).

3.1 Example function for GET request

```
@GetMapping(value="/getReservations")
public List<Reservation> getReservations()
{

    return reservationQuery.getReservations(jdbc);
}
```

This an example function of the API that gets a request at url

<http://localhost:8080/getCars>

Then it responds to this GET Request with all reservations.

3.2 Example function for POST request

```
@PostMapping(value = "/addCustomer")
public String addCustomer (@RequestBody Customer customer)
{
    String ret;
    ret=this.customerQuery.addCustomer(customer,jdbc);
    return ret;
}
```

This an example function of the API that gets a request at url

<http://localhost:8080/addCustomer>

Then it responds to this POST Request by creating the customer and inserting it to the database.