

# **Module: GUI Design and Database Connection**

# **Module Code: COMP4604**

# **Course Code: TU819/4**

# **Course Group: Group A/B**

# **Student Names: Jason Leonard & Brian Twene**

# **Student IDs: C19500766, C19344543**

# **Submission Date: 04/12/2022**

# **Topic: Vehicle Dealership/Rental System**

[**Module: GUI Design and Database Connection** 1](#_Toc121233923)

[**Module Code: COMP4604** 1](#_Toc121233924)

[**Course Code: TU819/4** 1](#_Toc121233925)

[**Course Group: Group A/B** 1](#_Toc121233926)

[**Student Names: Jason Leonard & Brian Twene** 1](#_Toc121233927)

[**Student IDs: C19500766, C19344543** 1](#_Toc121233928)

[**Submission Date: 04/12/2022** 1](#_Toc121233929)

[**Topic: Vehicle Dealership/Rental System** 1](#_Toc121233930)

[**Introduction** 3](#_Toc121233931)

[**Register Page (Customer)** 3](#_Toc121233932)

[**Login Page (Customer)** 4](#_Toc121233933)

[**Reset Password Page (Customer)** 5](#_Toc121233934)

[**Main Menu Page (Customer)** 6](#_Toc121233935)

[**Home Page (Customer)** 7](#_Toc121233936)

[**Rental / Purchases Page (Customer)** 8](#_Toc121233937)

[**Purchasing a Vehicle (Customer)** 9](#_Toc121233938)

[**Current Rentals (Customer)** 10](#_Toc121233939)

[**Login Page (Admin)** 11](#_Toc121233940)

[**Main Menu (Admin)** 12](#_Toc121233941)

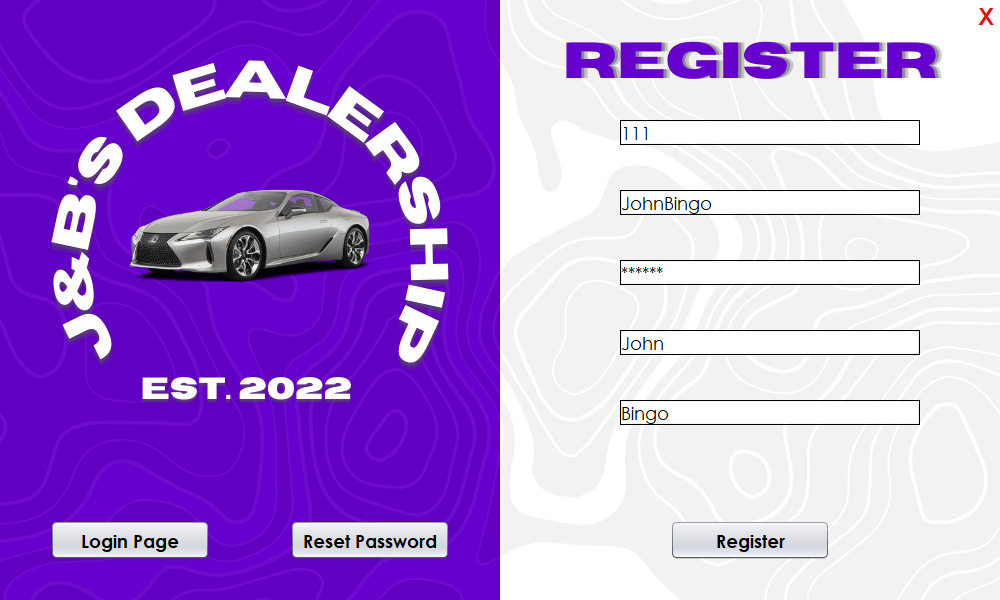
[**Manage Customers Page (Admin)** 12](#_Toc121233942)

[**Manage Vehicles Page (Admin)** 14](#_Toc121233943)

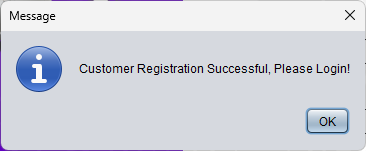
# **Introduction**

This test record document is based on a software solution for a vehicle dealership which allows a user to both purchase vehicles, rent vehicles and return vehicles when they’re finished using them. Additionally, the program also allows an admin to alter a user’s and vehicles table which is located within a Microsoft Access database, and allows the admin to add, edit, and remove information from their respective tables. Our application also features stock and inventory control and allows management to print a list of users and a list of our stock and inventory. Lastly, an admin is also able to use this program to check overdue accounts for people who have not returned their rented vehicle.

## **Register Page (Customer)**



If correct it’ll output the following:

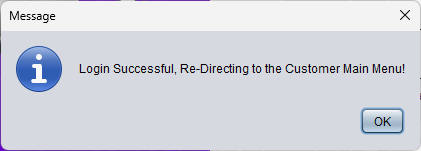


## **Login Page (Customer)**

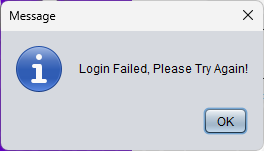
Graphical user interface, application

Description automatically generated

If correct it’ll output the following:



If incorrect it’ll output the following:

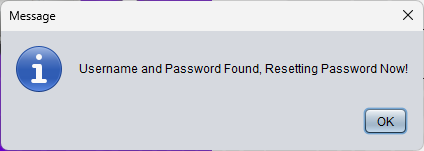


## **Reset Password Page (Customer)**

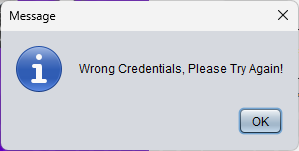
Graphical user interface, application

Description automatically generated

If correct it’ll output the following:



If correct it’ll output the following:



## **Main Menu Page (Customer)**



## **Home Page (Customer)**

Graphical user interface, application

Description automatically generated

## **Rental / Purchases Page (Customer)**

Graphical user interface, application

Description automatically generated

Here we’ll click “Buy” on the “Luxury” vehicle.

## **Purchasing a Vehicle (Customer)**

Graphical user interface, application, Teams

Description automatically generated

The vehicle once we pressed “Buy Now” has been added to the “Purchase” table.

A screenshot of a computer

Description automatically generated

After we’ve clicked on the “Luxury” vehicle is shows the above, and when we click “Buy Now” it’ll add the car into the “Purchase” table.

## **Current Rentals (Customer)**

Graphical user interface

Description automatically generated

We choose our location and then click “Return”.

Graphical user interface, application, Word

Description automatically generated

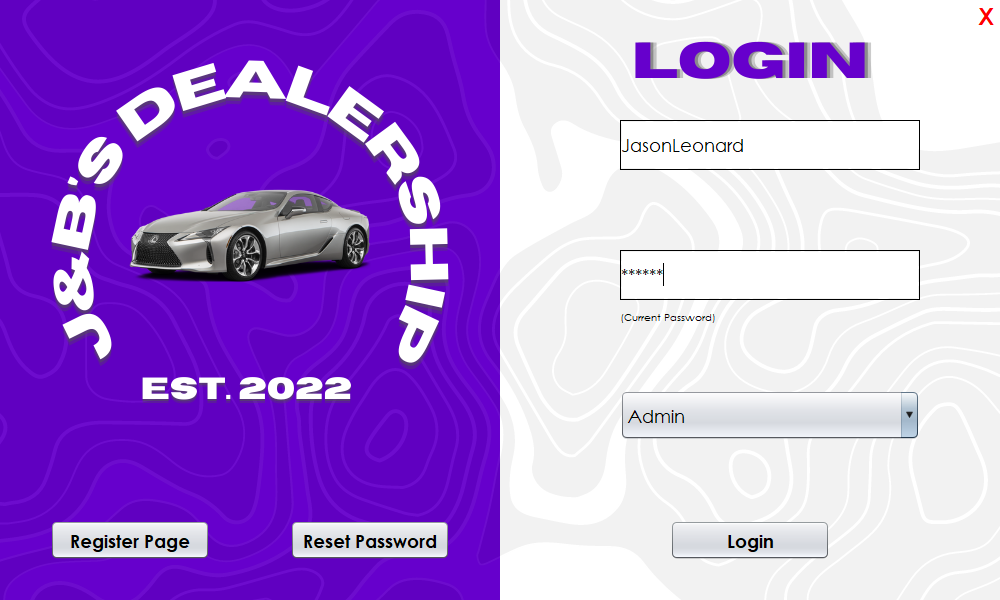
Afterwards we then have to confirm our option and we get this dialog above, stating “OK”.

Graphical user interface, application, table

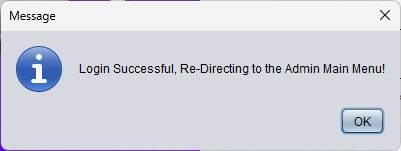
Description automatically generated

Once we press “OK”, the vehicle gets added into our “Rental” history. The same occurs for when we purchase a vehicle, it then gets added into our “Purchase” history.

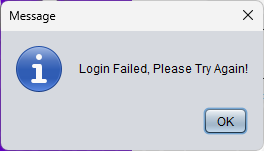
## **Login Page (Admin)**



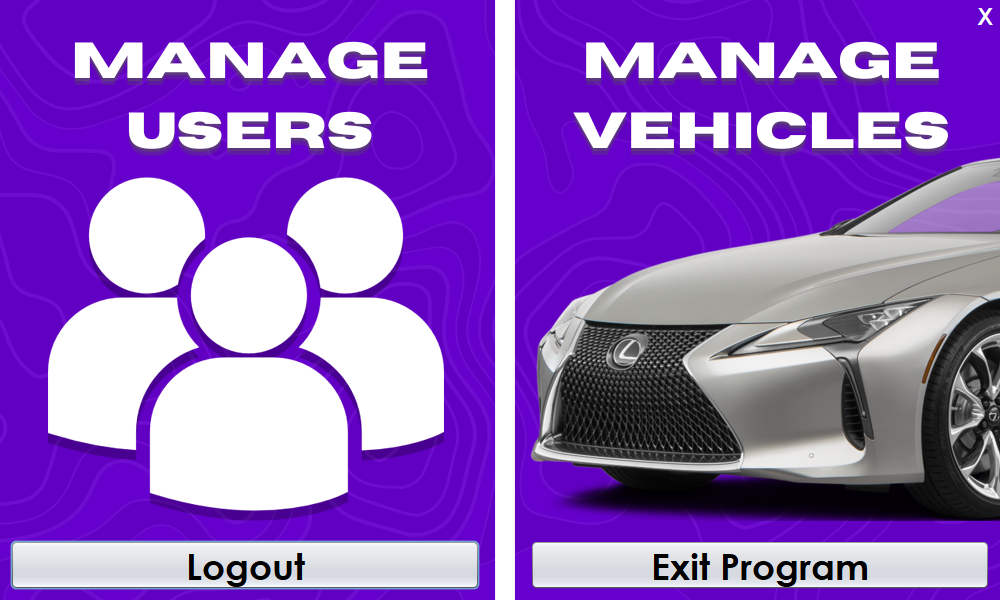
If correct it’ll output the following:



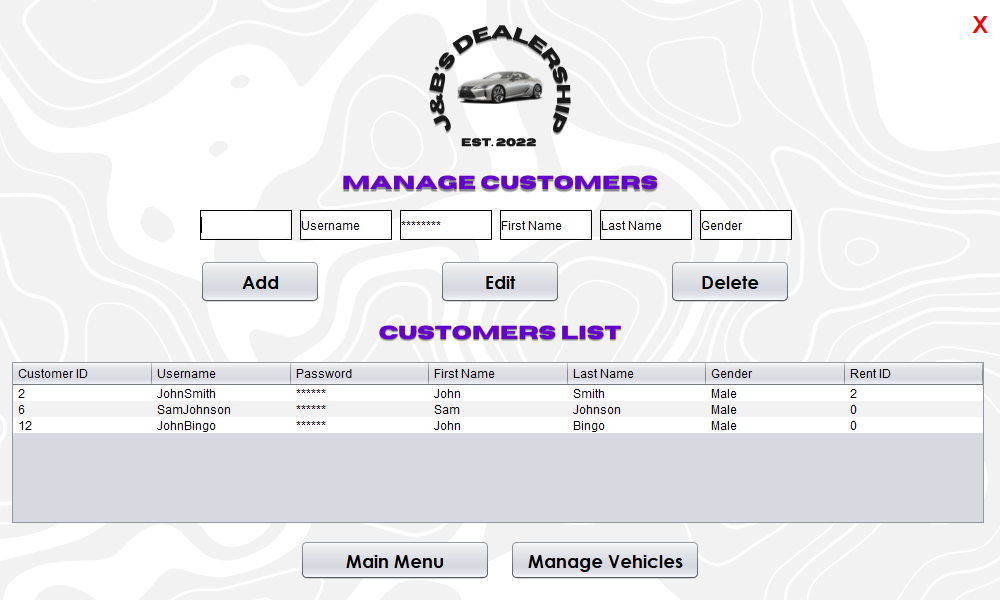
If incorrect it’ll output the following:



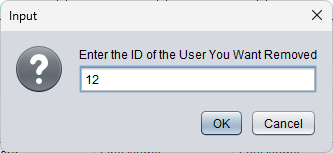
## **Main Menu (Admin)**



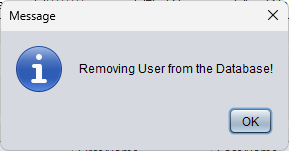
## **Manage Customers Page (Admin)**



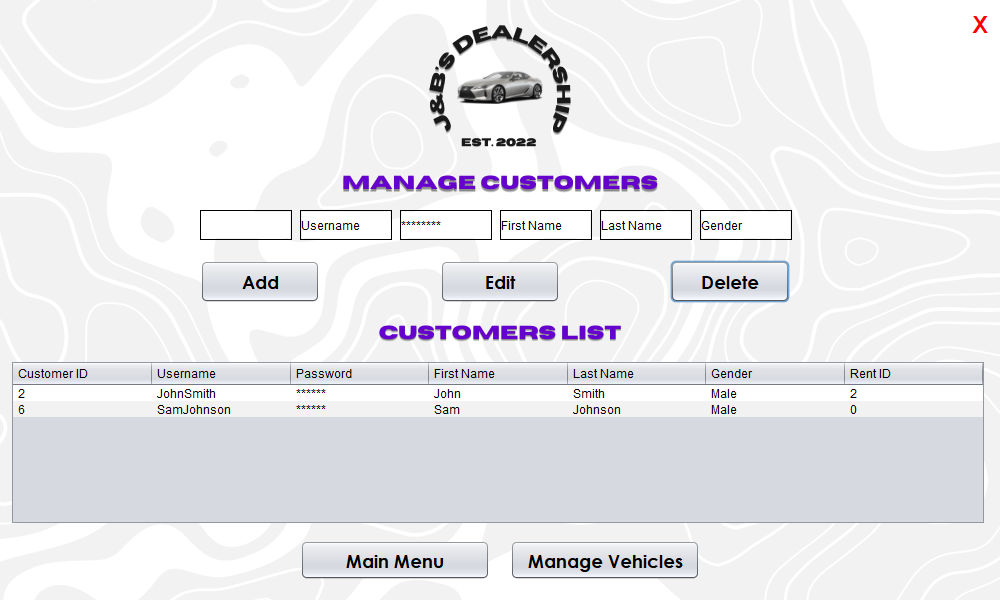
We then input the ID of a user we want to remove.



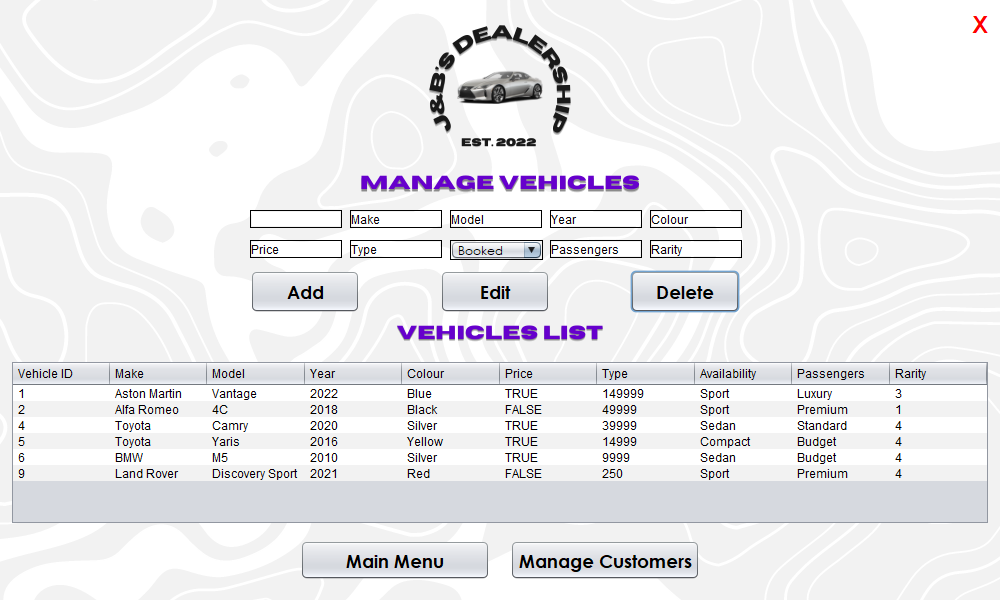
If ID has been found, it’ll output the following:



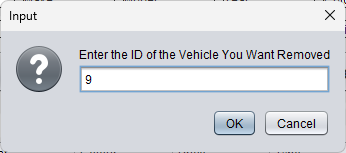
It’ll then display the user has been removed in our table below:



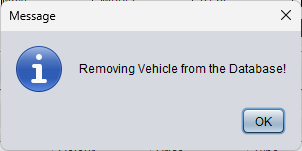
## **Manage Vehicles Page (Admin)**



We then input the ID of a user we want to remove.



If ID has been found, it’ll output the following:



It’ll then display the vehicle has been removed in our table below:

