McQueen Car Rentals

Done By,

S. Subhiksha – 205002101 – Batch 2

S. Supraja – 205002105 – Batch 2

Supriya Abirami A– 205002106 – Batch 2

1. Problem Statement :

Develop a database project for a car rental system. Agents can register and provide all details about the cars they wish to provide for rent with tariff and terms and conditions. Customers can register and choose the car they want to rent. Agencies can update, add, delete cars in the database. Customers can view the variety of cars available for in rent in their budget.

Functional Requirement:

We require the following entities for the project:

- Admin: To store the details of admin
- Car_desc : To store the description of cars for rent
- Customer : To store the details of customer
- Rental : To store the rental details
- Payment : To store the payment details

Non-Functional Requirement:

- Security: Only authorized customers and admins are able to use the facilities
- Appropriate error messages are displayed
- Data Inconsistency: All the appropriate scenarios have been handled

The Admin can encode 1 or more Customer information. The Customer can one or more cars from Car_desc. The Rental has only 1 detail of the Car_desc at a time or per transaction. The Admin processes one or more transactions through Payment along with Rental details and Customer Information.

Real Time application:

This is widely used by online Car Rental Companies along with other details.

Requirements:

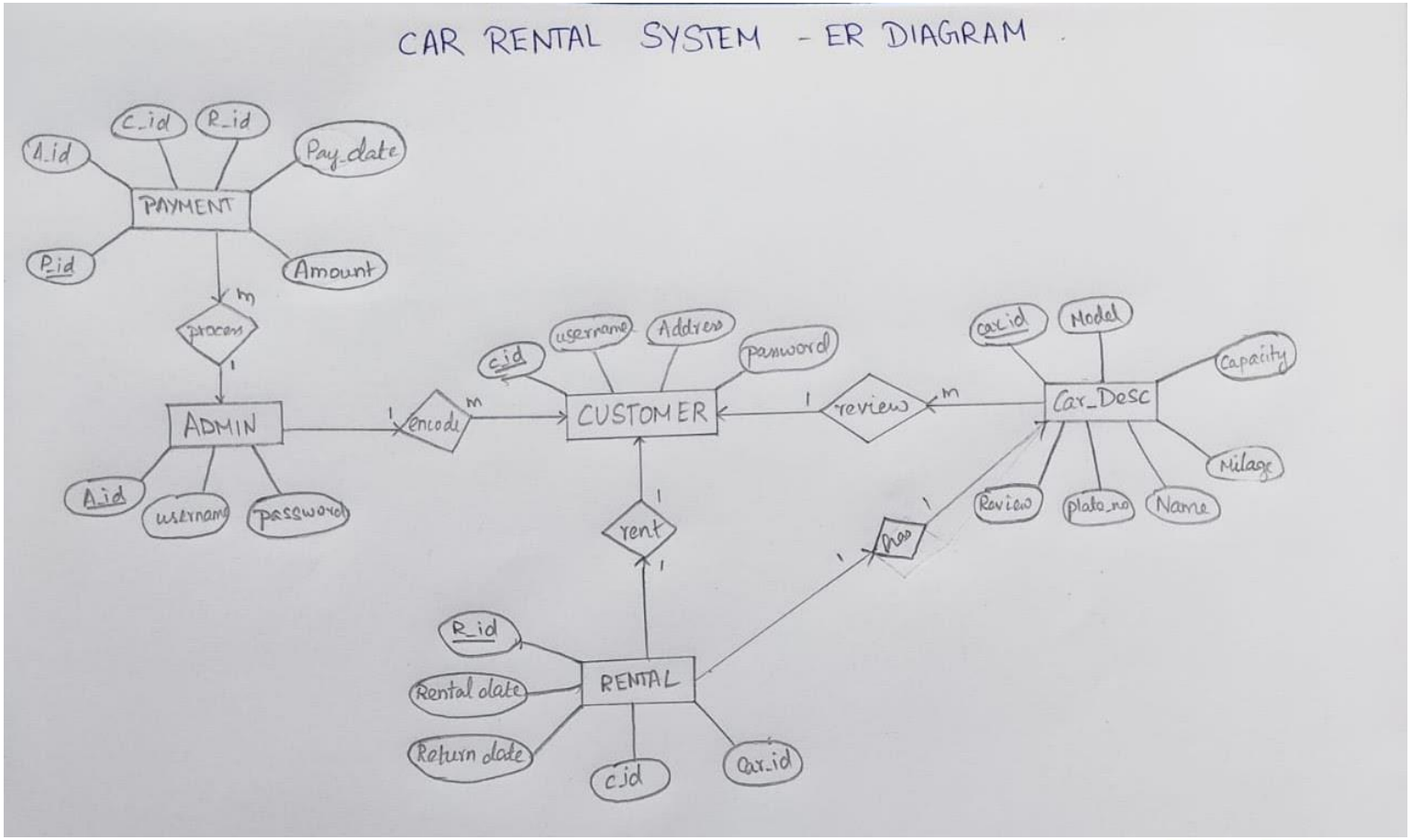
We would require the following entities with appropriate relationships:

- Admin: To store the details of admin
- Car_desc : To store the description of cars for rent
- Car Images : To store the images of cars
- Customer : To store the details of customer
- Rental : To store the rental details
- Payment : To store the payment details
- Owner : To store the details of the car owner

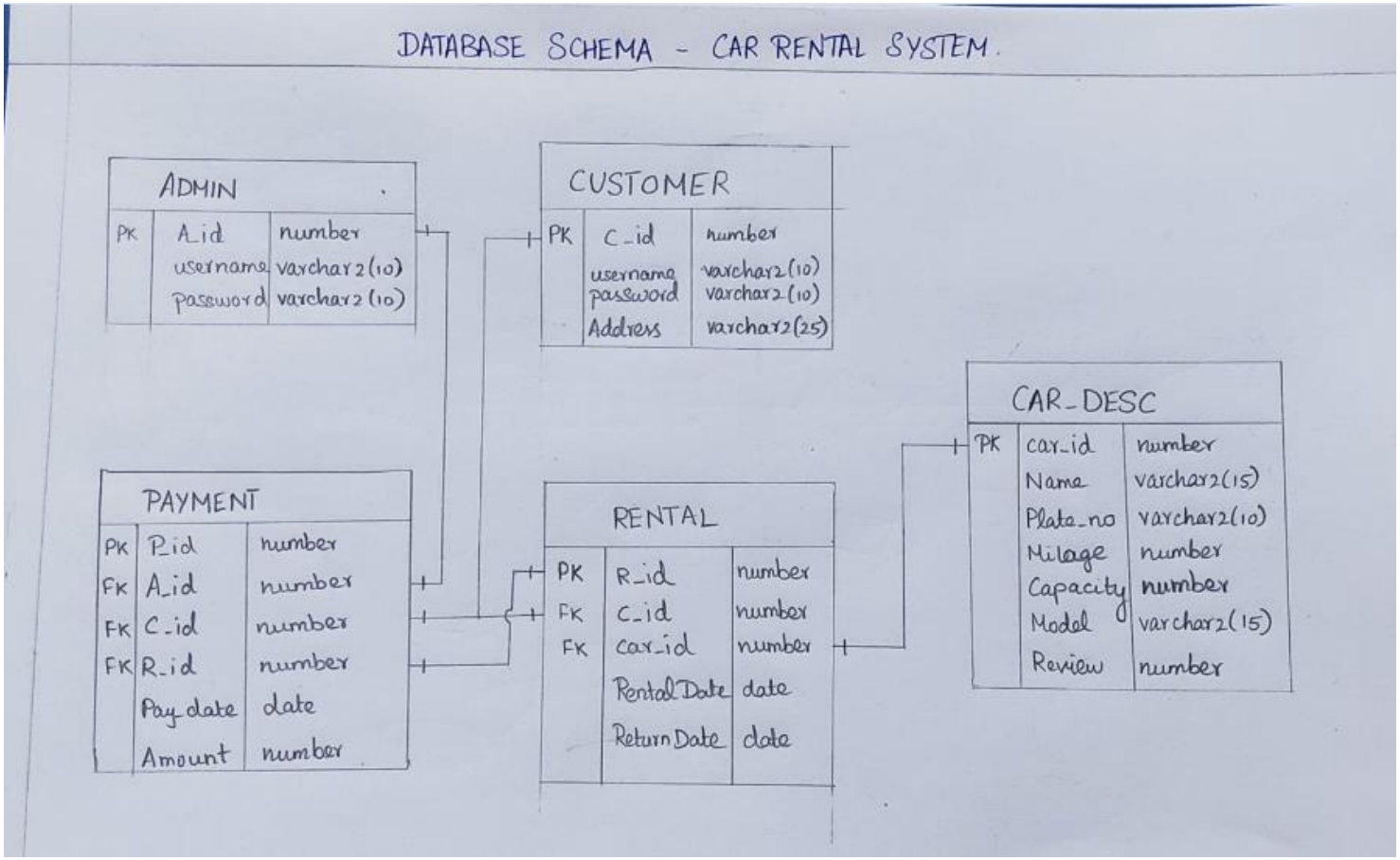
Challenges:

- To effectively deal with cancellations and updates of information regarding customer details, car details and owner details.
- To maintain consistency of booking and cancellation data.
- To ensure that the payment transactions are carried out successfully
- To avoid duplication of data and anomalies.
- To avoid risk of over-booking
- To validate the authorization properly

2. ER Diagram:


CAR RENTAL SYSTEM - ER DIAGRAM
PAYMENT
A_id
C_id
R_id
Pay_date
P_id
Amount
process
m
1
ADMIN
A_id
username
password
encodes
1
m
CUSTOMER
c_id
username
Address
password
review
1
m
Car_Desc
car_id
Model
Capacity
Mileage
Review
plate_no
Name
rent
1
1
has
1
1
RENTAL
R_id
Rental date
Return date
c_id
car_id

3. Database schema:



4. Database normalization:

- Admin:

  It has a primary key A_id and all the attributes are dependent only on it. It is atomic so the table must be atleast in 1NF, since there is no partial dependency therefore it must be atleast in 2NF. There is no transitive dependency too hence it must be in 3NF.

- Customer:

  It has a primary key C_id and all the attributes are dependent only on it. It is atomic so the table must be atleast in 1NF, since there is no partial dependency therefore it must be atleast in 2NF. There is no transitive dependency too hence it must be in 3NF.

- Car_desc:

It has a primary key Car_id and all the attributes are dependent only on it. It is atomic so the table must be atleast in 1NF, since there is no partial dependency therefore it must be atleast in 2NF. There is no transitive dependency too hence it must be in 3NF.

- Rental:

It has a primary key R_id and all the attributes are dependent on the set of R_id, Car_id, C_id. It is atomic so the table must be atleast in 1NF, since there is no partial dependency therefore it must be atleast in 2NF. There is no transitive dependency too hence it must be in 3NF.

- Payment:

It has a primary key P_id and all the attributes are dependent on the set of A_id, R_id, Car_id, C_id. It is atomic so the table must be atleast in 1NF, since there is no partial dependency therefore it must be atleast in 2NF. There is no transitive dependency too hence it must be in 3NF.

Hence the whole schema is in 3NF and it cannot be normalized further.

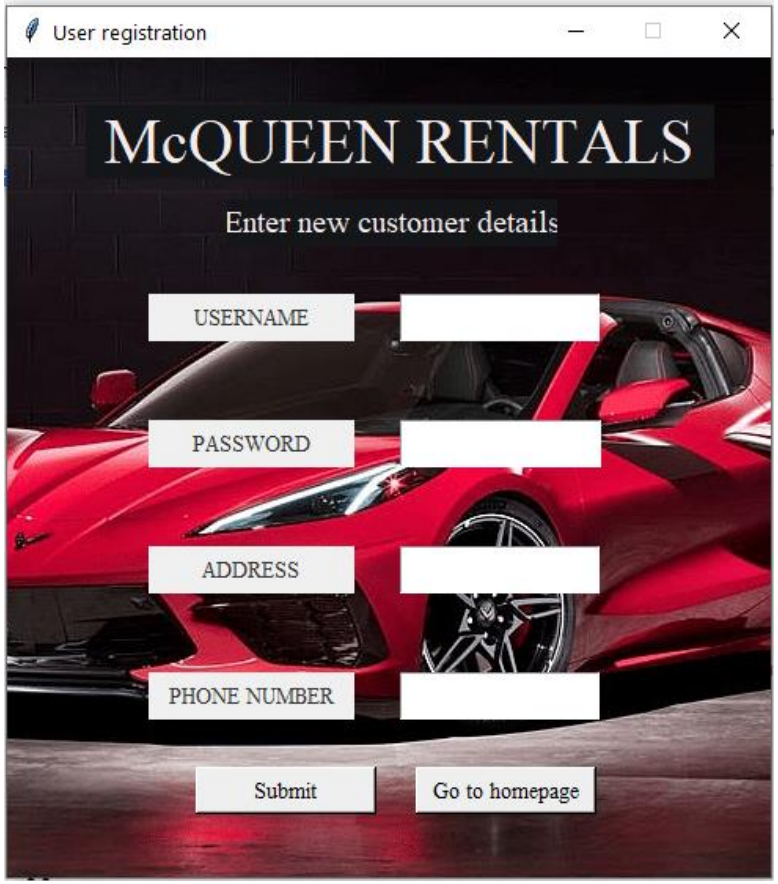
5. Coding and Implementation

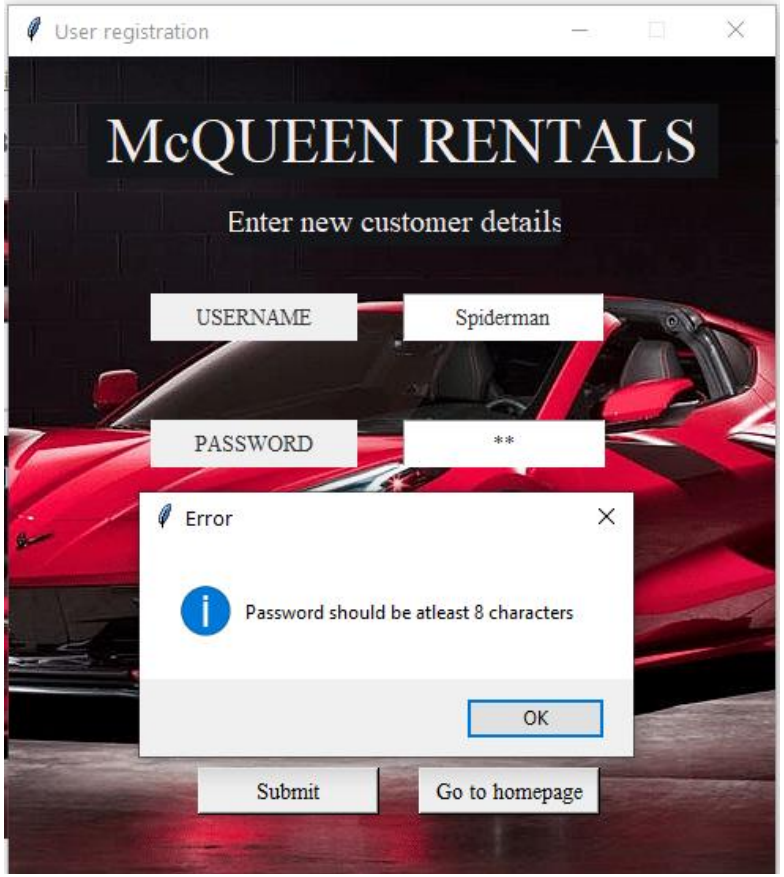
This is carried out through the following applications:

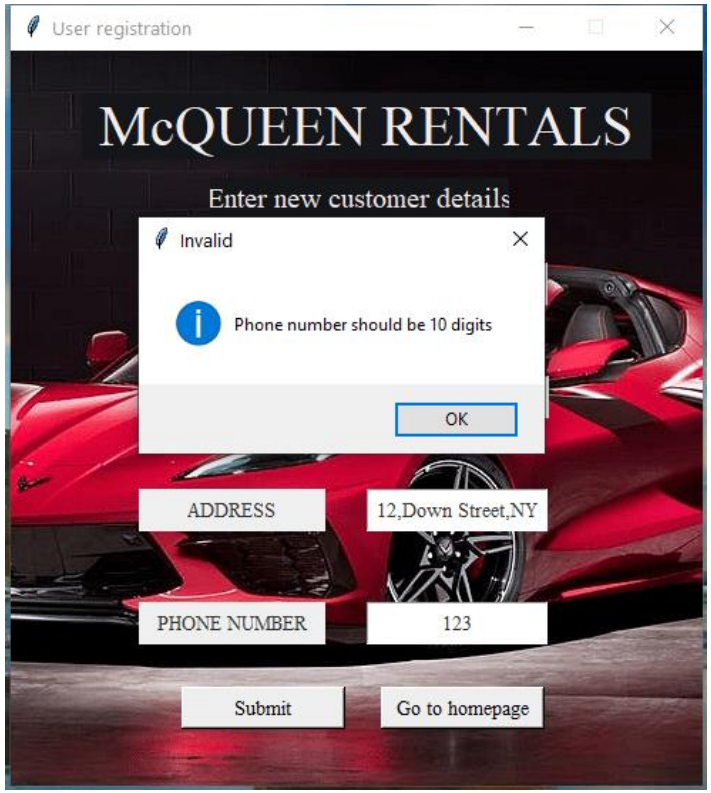
- SQL
- TKINTER

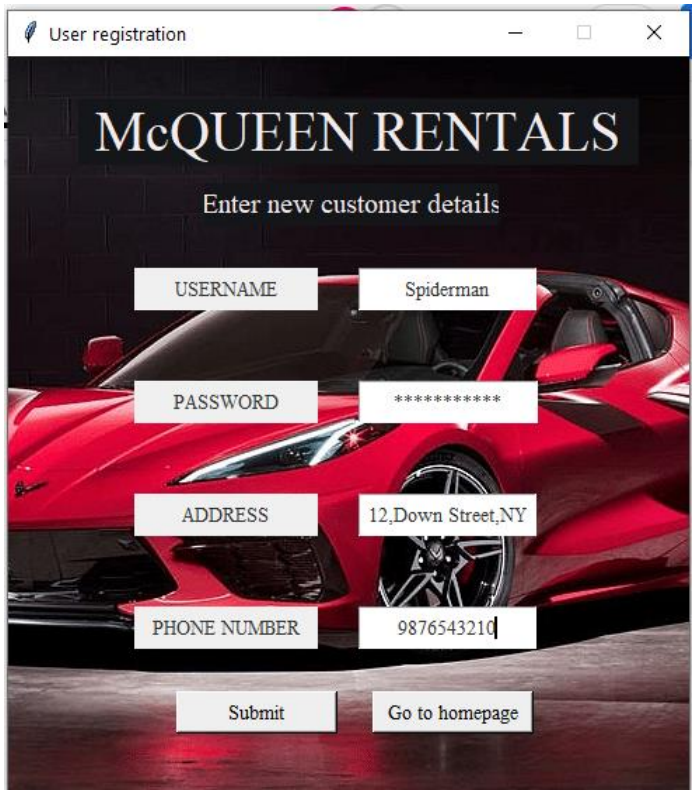
6. Results:

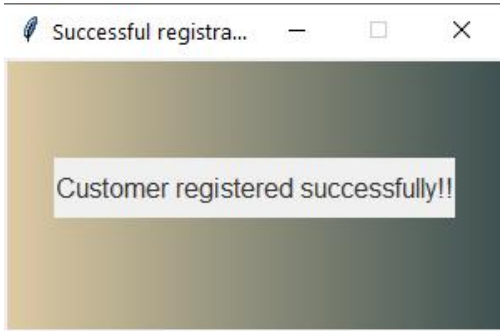

Login page
McQUEEN RENTALS
WELCOME!!
For customers
Customer login
Customer signup
Customer unsubscribe
For admins
Admin login
Admin signup
Admin unsubscribe

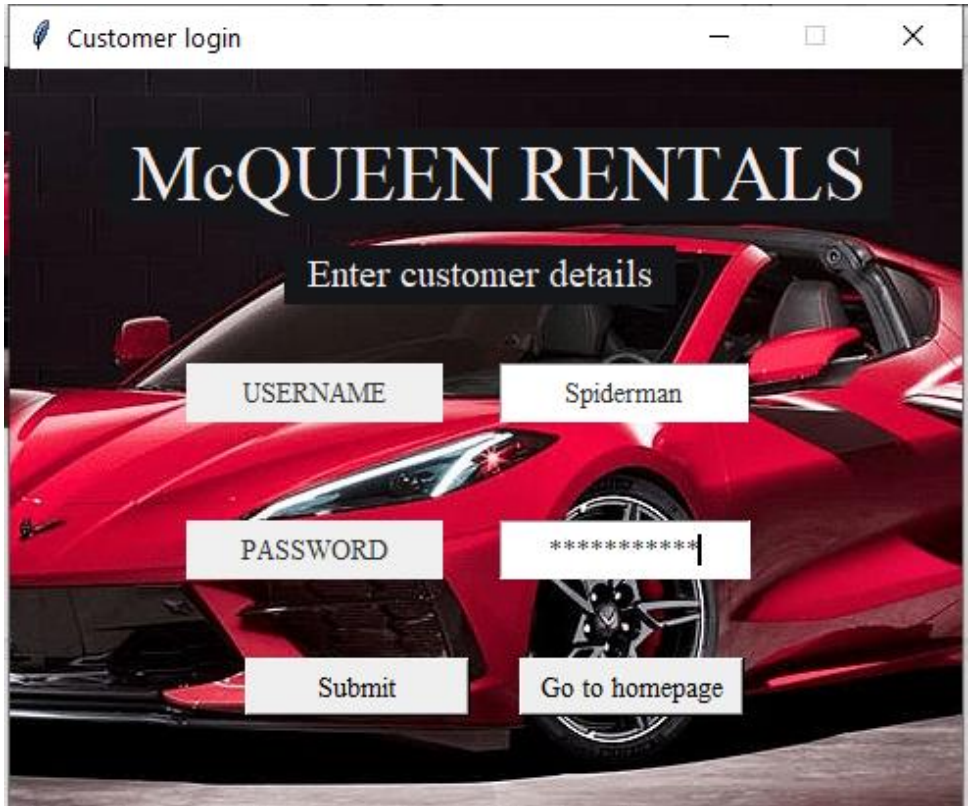

User registration
McQUEEN RENTALS
Enter new customer details
USERNAME
PASSWORD
ADDRESS
PHONE NUMBER
Submit
Go to homepage

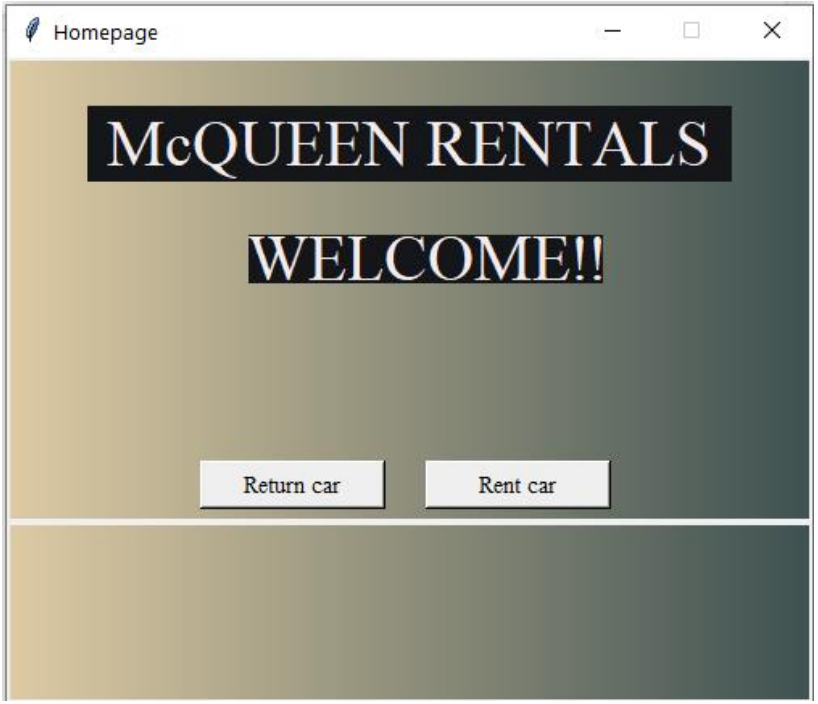

User registration
McQUEEN RENTALS
Enter new customer details
USERNAME
Spiderman
PASSWORD
**
Error
Password should be atleast 8 characters
OK
Submit
Go to homepage


User registration
McQUEEN RENTALS
Enter new customer details
Invalid
Phone number should be 10 digits
OK
ADDRESS
12,Down Street,NY
PHONE NUMBER
123
Submit
Go to homepage

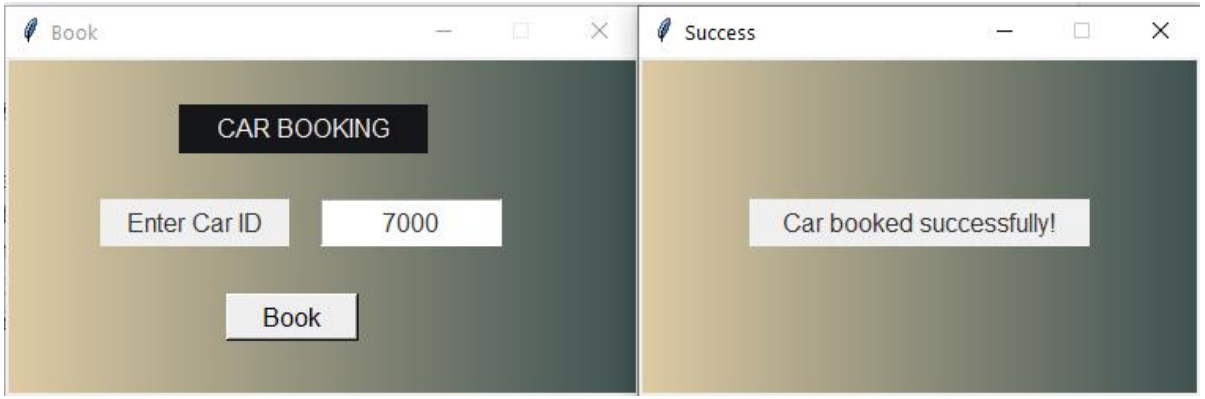

User registration
McQUEEN RENTALS
Enter new customer details
USERNAME
Spiderman
PASSWORD
****
ADDRESS
12,Down Street,NY
PHONE NUMBER
9876543210
Submit
Go to homepage

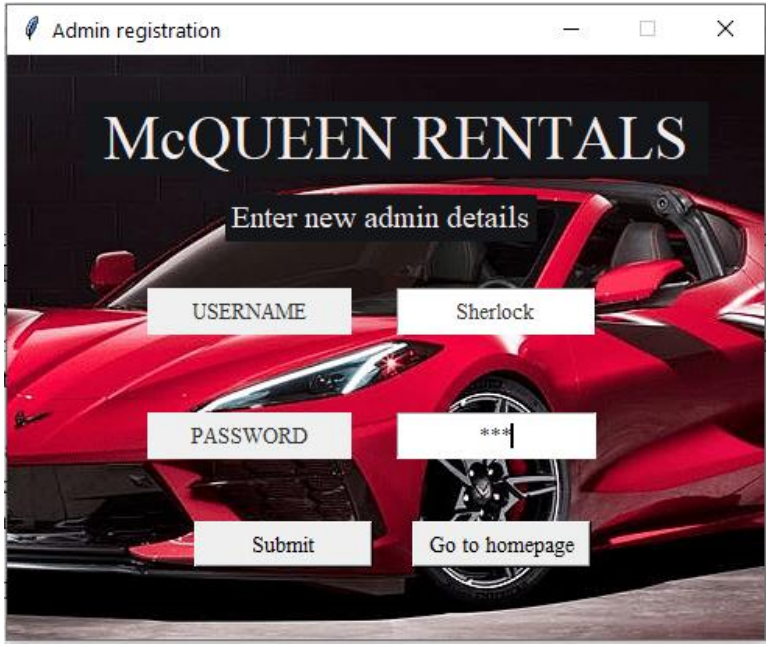

Successful registra...
Customer registered successfully!!

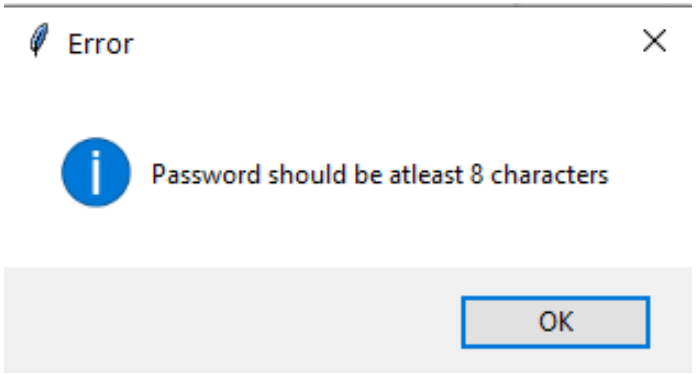

Customer login
McQUEEN RENTALS
Enter customer details
USERNAME
Spiderman
PASSWORD
****
Submit
Go to homepage

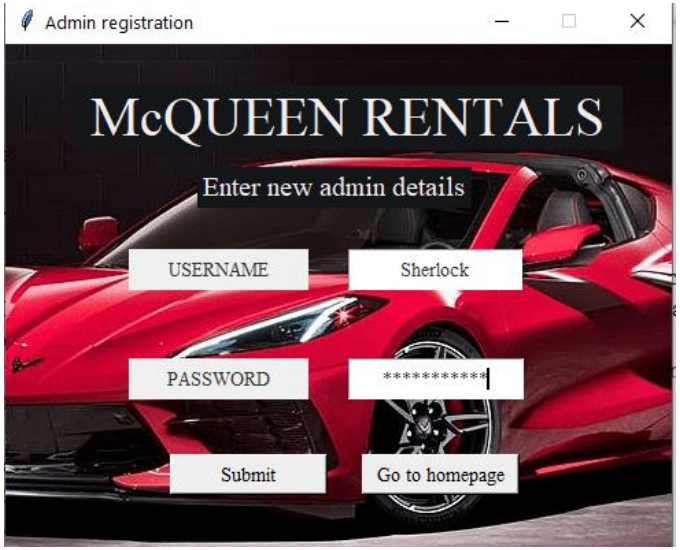

Homepage
McQUEEN RENTALS
WELCOME!!
Return car
Rent car

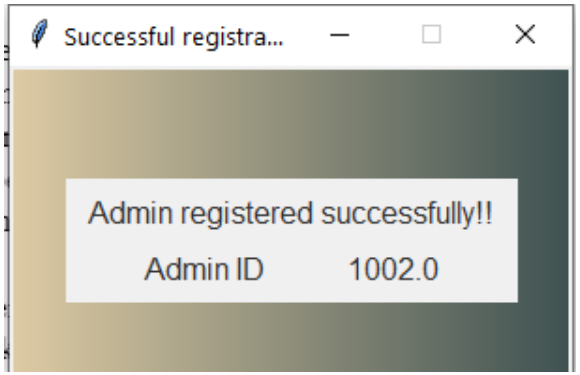

Display
Car ID
Plate No
Name
Capacity
Milage
Name
Review
7000.0
V12 AML
DB7 Volante
4.0
6.8
Aston Martin
5.0
7001.0
AP02BK1084
Baleno
5.0
21.96
Maruti Suzuki
4.5
7002.0
MH12RN1289
Duster
5.0
16.5
Renault
4.2
7003.0
IN Q 3131
Q3 1.4 TFSI
5.0
13.15
Audi
4.0
7004.0
KA08J9192
Crysta
8.0
16.5
Toyota Innova
4.4
7005.0
TN21AT4855
Fortuner
7.0
14.4
Toyota
4.5
7006.0
M TF 3880
X1 SDrzive20i
5.0
11.24
BMW
4.7
7007.0
MH43BU9429
Swift
5.0
23.2
Maruti Suzuki
4.5
7008.0
TN25B1765
Elite i20
5.0
18.6
Hyundai
3.9
7009.0
HR26TC7174
Indica eV2
5.0
25.0
TATA
3.5
Proceed to Book
Go to homepage

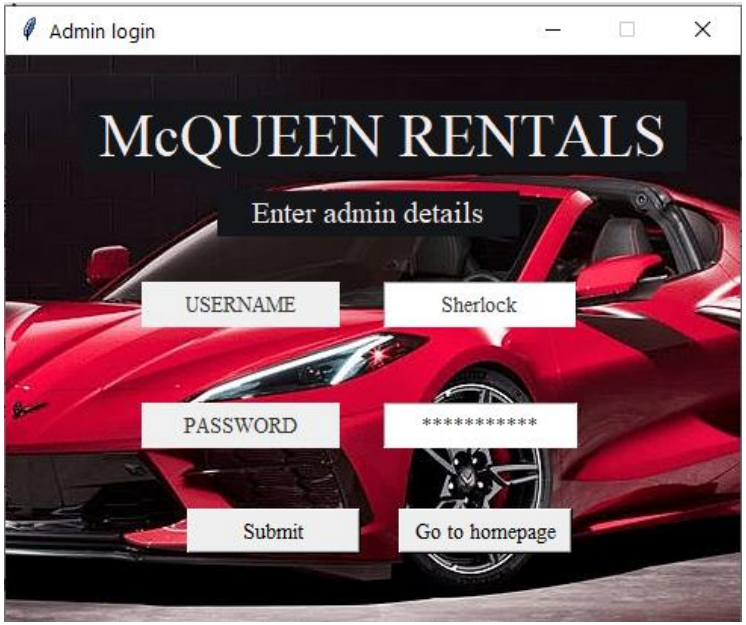

Book
CAR BOOKING
Enter Car ID
7000
Book
Success
Car booked successfully!

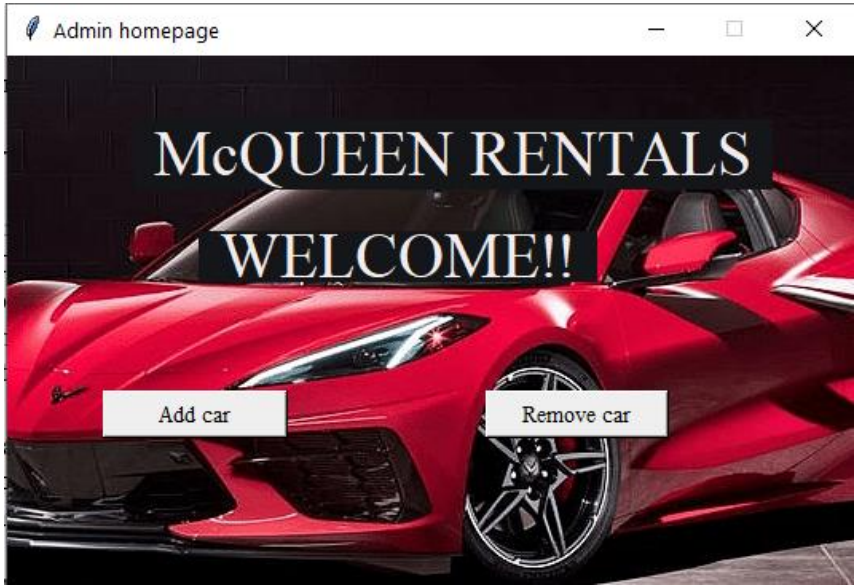

Admin registration
McQUEEN RENTALS
Enter new admin details
USERNAME
Sherlock
PASSWORD
***
Submit
Go to homepage

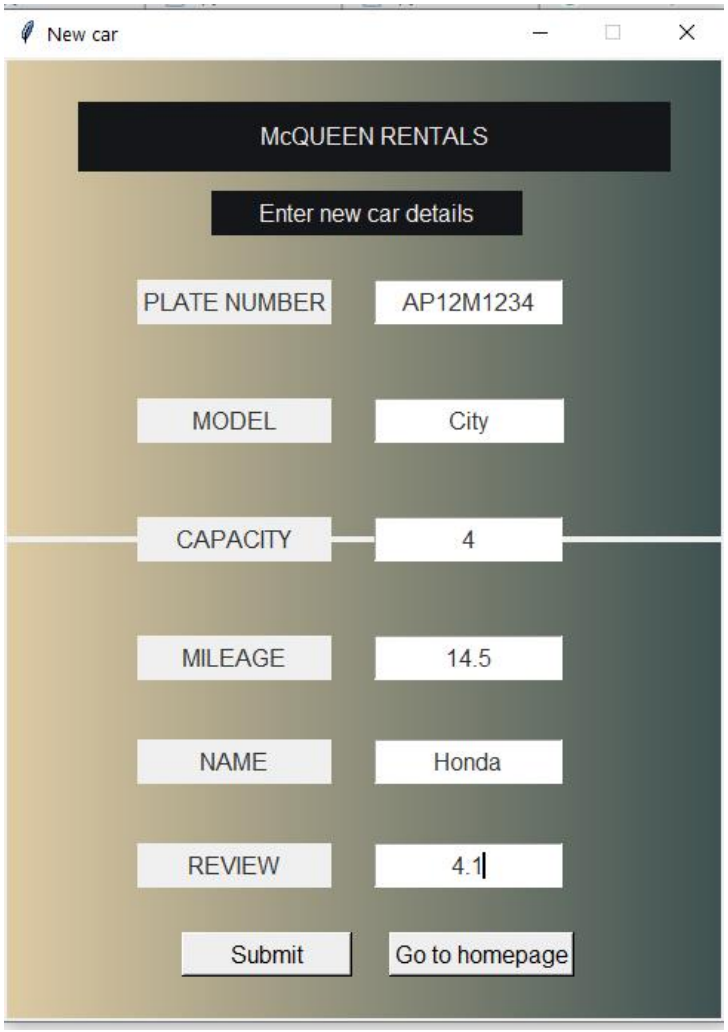

Error
Password should be atleast 8 characters
OK

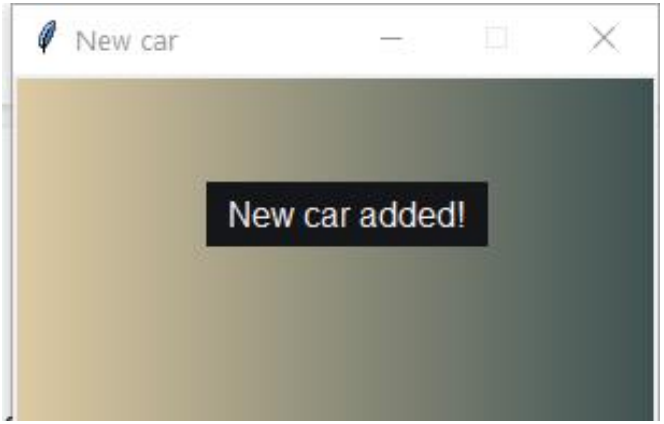

Admin registration
McQUEEN RENTALS
Enter new admin details
USERNAME
Sherlock
PASSWORD
****
Submit
Go to homepage

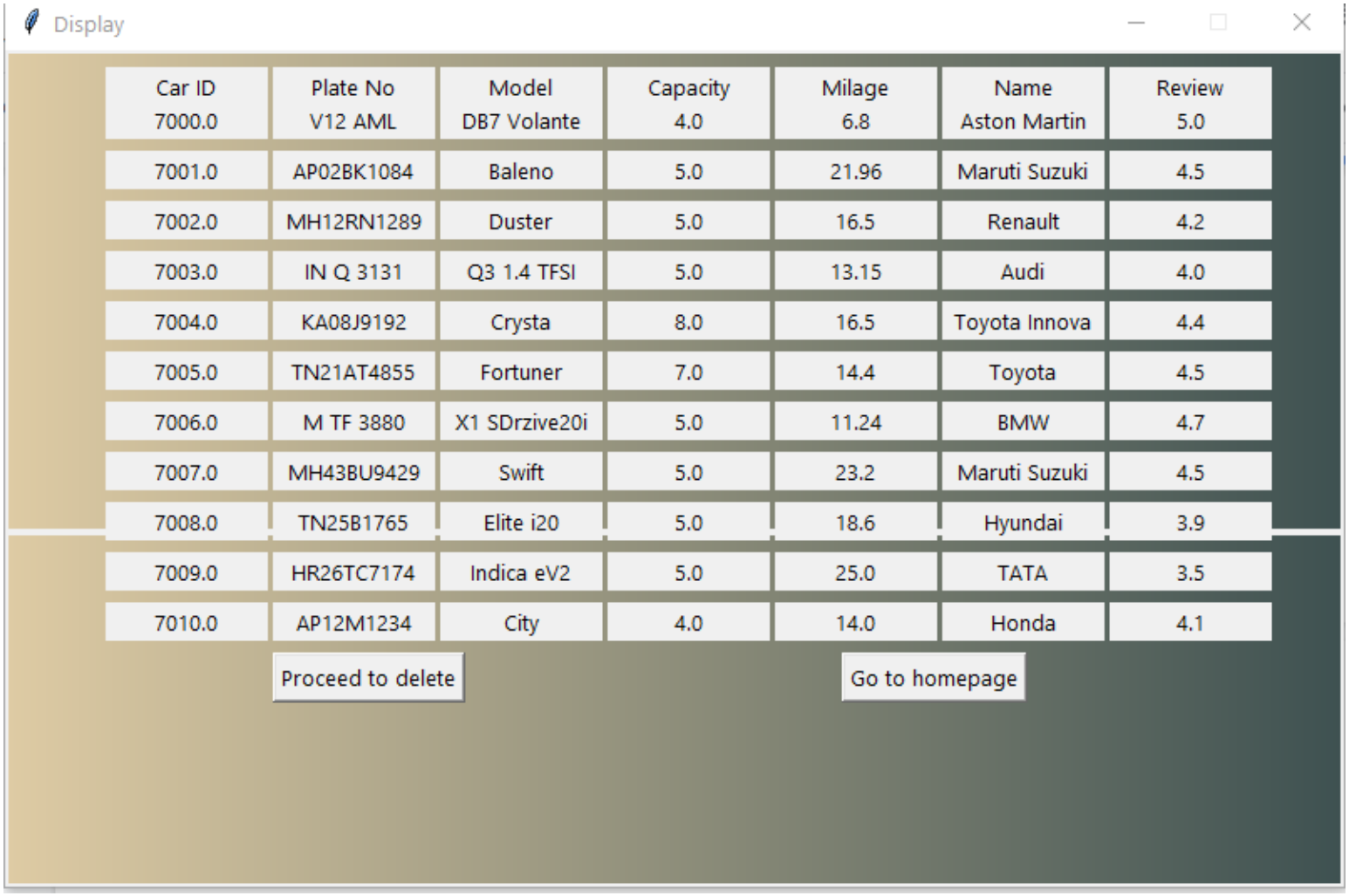

Successful registra...
Admin registered successfully!!
Admin ID 1002.0

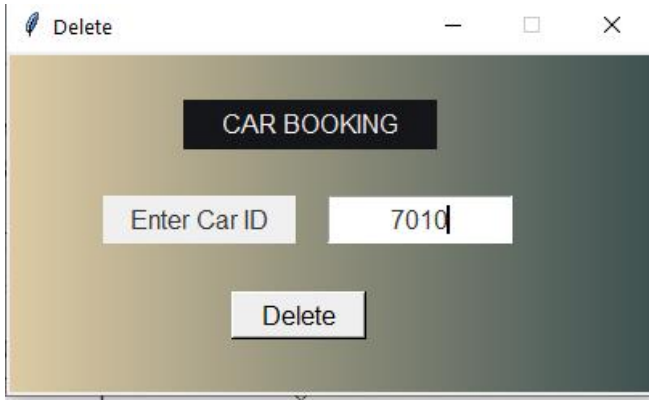

Admin login
McQUEEN RENTALS
Enter admin details
USERNAME
Sherlock
PASSWORD
****
Submit
Go to homepage

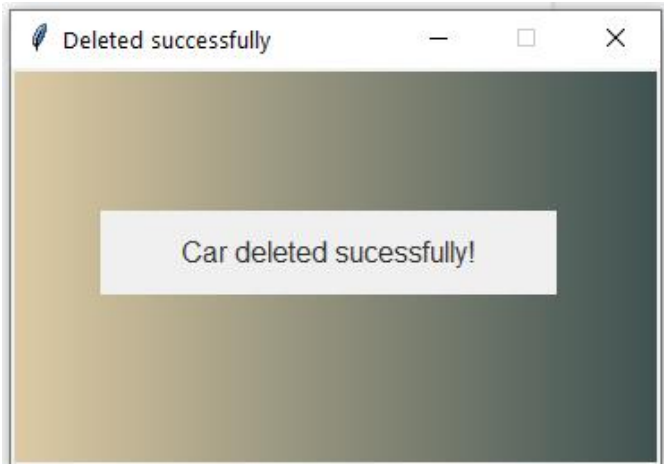

Admin homepage
McQUEEN RENTALS
WELCOME!!
Add car
Remove car

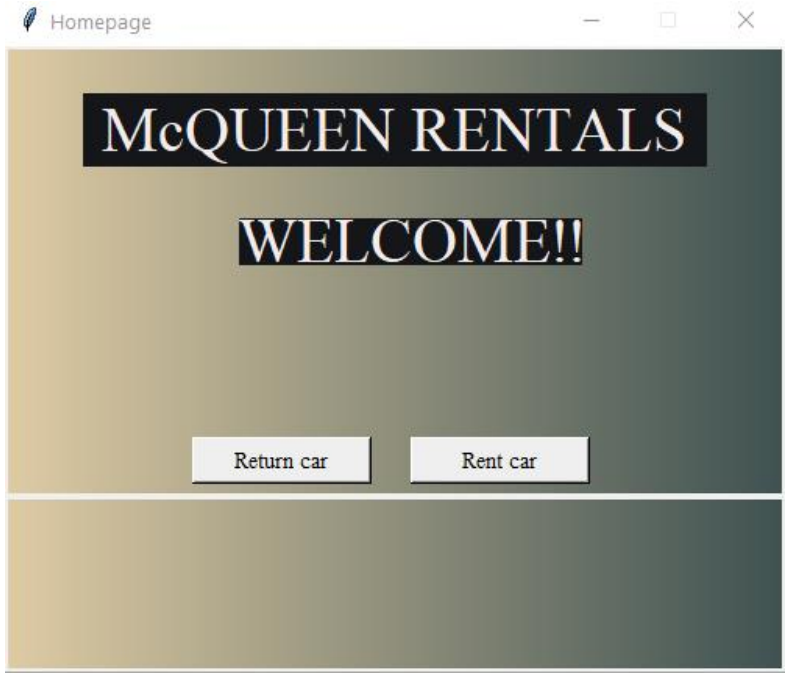

New car
McQUEEN RENTALS
Enter new car details
PLATE NUMBER
AP12M1234
MODEL
City
CAPACITY
4
MILEAGE
14.5
NAME
Honda
REVIEW
4.1
Submit
Go to homepage

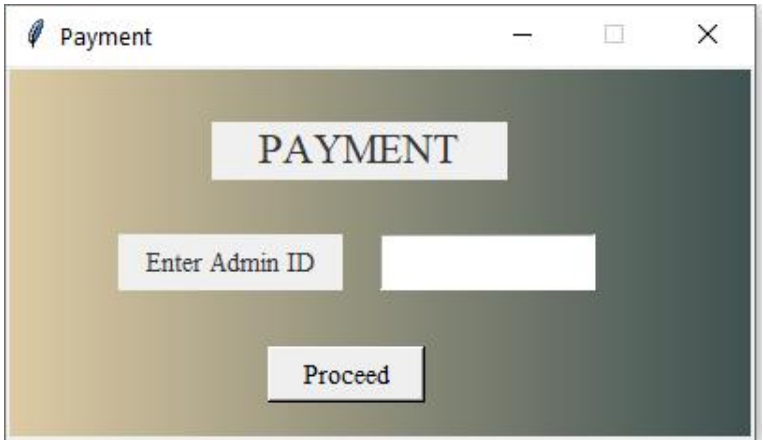

New car
New car added!

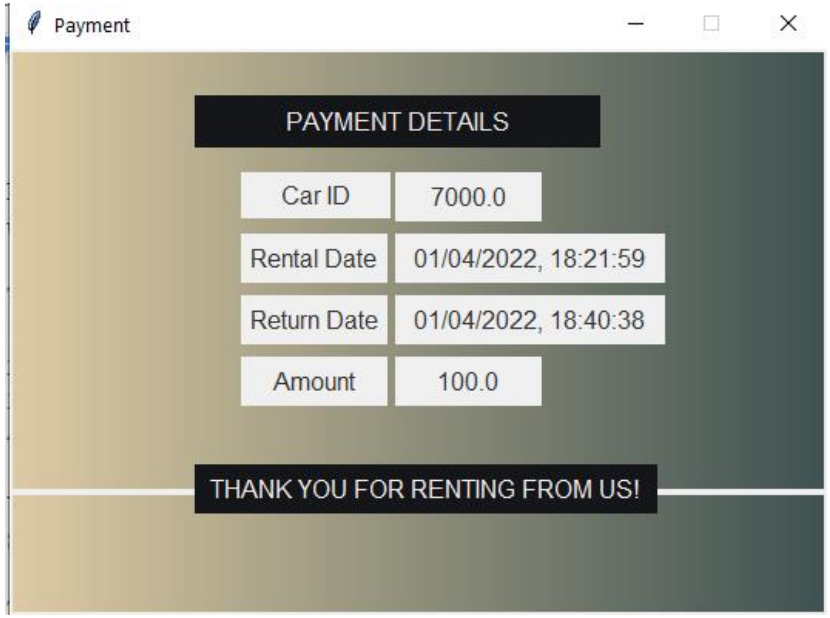

Display
Car ID
7000.0
Plate No
V12 AML
Model
DB7 Volante
Capacity
4.0
Milage
6.8
Name
Aston Martin
Review
5.0
7001.0
AP02BK1084
Baleno
5.0
21.96
Maruti Suzuki
4.5
7002.0
MH12RN1289
Duster
5.0
16.5
Renault
4.2
7003.0
IN Q 3131
Q3 1.4 TFSI
5.0
13.15
Audi
4.0
7004.0
KA08J9192
Crysta
8.0
16.5
Toyota Innova
4.4
7005.0
TN21AT4855
Fortuner
7.0
14.4
Toyota
4.5
7006.0
M TF 3880
X1 SDrzive20i
5.0
11.24
BMW
4.7
7007.0
MH43BU9429
Swift
5.0
23.2
Maruti Suzuki
4.5
7008.0
TN25B1765
Elite i20
5.0
18.6
Hyundai
3.9
7009.0
HR26TC7174
Indica eV2
5.0
25.0
TATA
3.5
7010.0
AP12M1234
City
4.0
14.0
Honda
4.1
Proceed to delete
Go to homepage

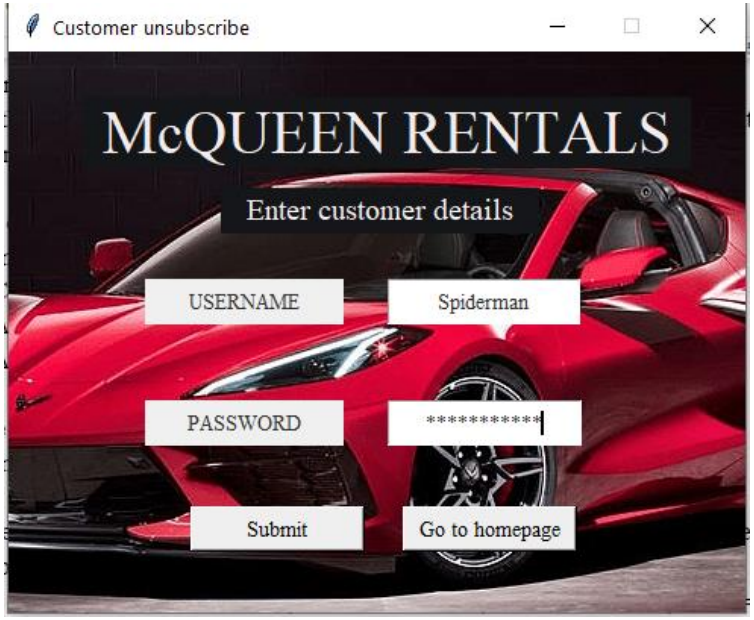

Delete
CAR BOOKING
Enter Car ID
7010
Delete

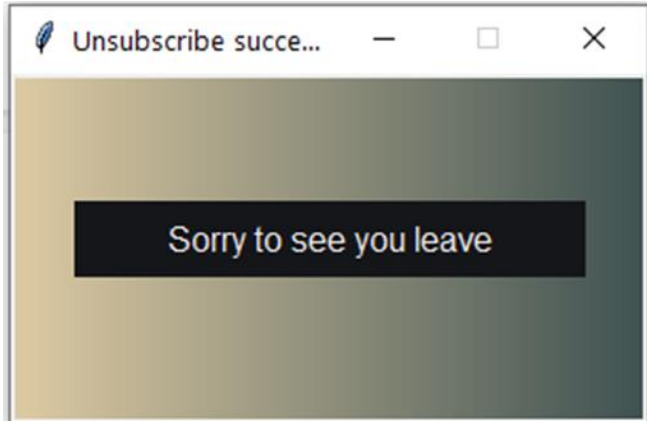

Deleted successfully
Car deleted sucessfully!

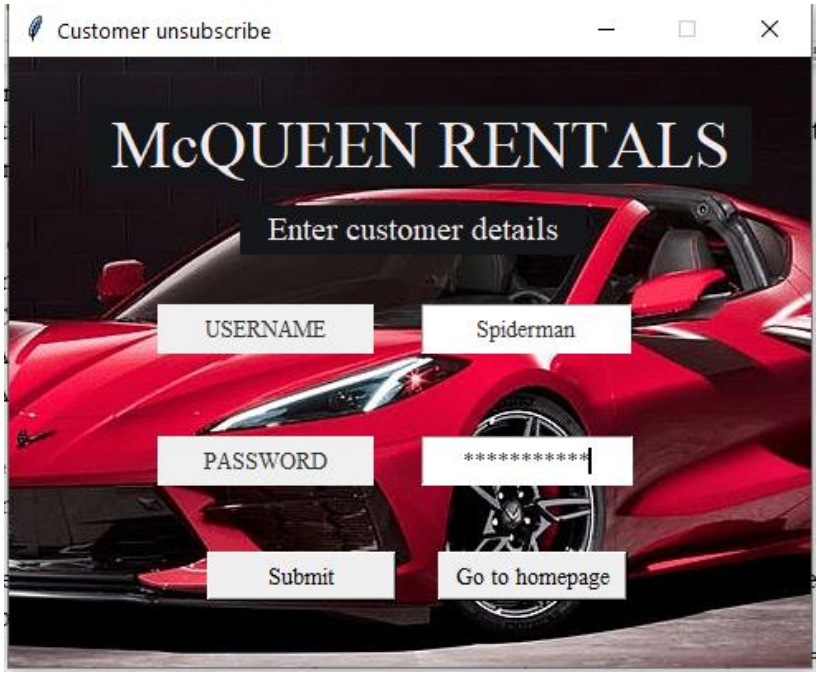

Homepage
McQUEEN RENTALS
WELCOME!!
Return car
Rent car

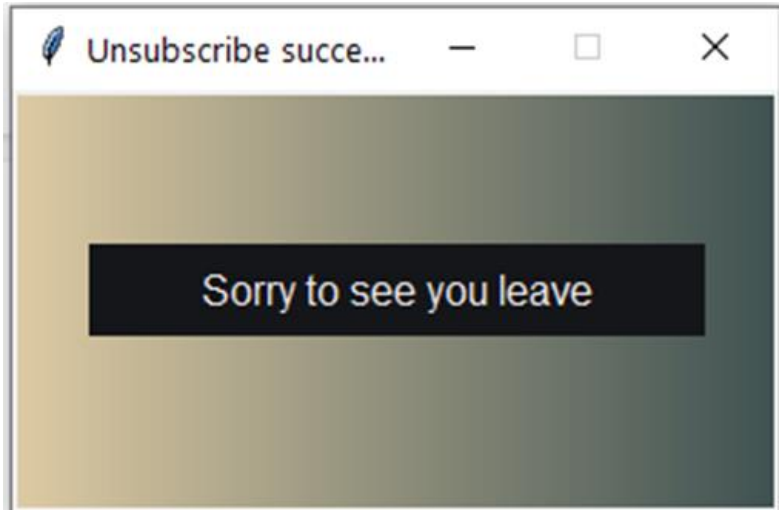

Payment
PAYMENT
Enter Admin ID
Proceed


Payment
PAYMENT DETAILS
Car ID
7000.0
Rental Date
01/04/2022, 18:21:59
Return Date
01/04/2022, 18:40:38
Amount
100.0
THANK YOU FOR RENTING FROM US!


Customer unsubscribe
McQUEEN RENTALS
Enter customer details
USERNAME
Spiderman
PASSWORD
****
Submit
Go to homepage


Unsubscribe succe...
Sorry to see you leave


Customer unsubscribe
McQUEEN RENTALS
Enter customer details
USERNAME
Spiderman
PASSWORD
****
Submit
Go to homepage


Unsubscribe succe...
Sorry to see you leave

Appendix:

Report on Python GUI tkinter:

The tkinter package (“Tk interface”) is the standard Python interface to the Tcl/Tk GUI toolkit. Both Tk and tkinter are available on most Unix platforms, including macOS, as well as on Windows systems.

Python with tkinter is the fastest and easiest way to create the GUI applications. Creating a GUI using tkinter is an easy task.

To create a tkinter app:

- 1.Importing the module – tkinter
- 2.Create the main window (container)
- 3.Add any number of widgets to the main window
- 4.Apply the event Trigger on the widgets.

We import tkinter just like any other module

Syntax: import tkinter

There are two main methods used which the user needs to remember while creating the Python application with GUI.

1. Tk(screenName=None, baseName=None, className='Tk', useTk=1):

Used to create a main window of the application

For example: m=tkinter.Tk() where m is the name of the main window object

2. mainloop(): There is a method known by the name mainloop() is used when your application is ready to run. mainloop() is an infinite loop used to run the application, wait for an event to occur and process the event as long as the window is not closed.

Syntax: m.mainloop()

There are a number of widgets which you can put in your tkinter application.

There are number of options which are used to change the format of the widget. Number of options can be passed as parameters separated by commas.

Some of the major widgets are

1. Button:

To add a button in your application, this widget is used.

The general syntax is:

```
w=Button(master, option=value)
```

master is the parameter used to represent the parent window.

2. Canvas: It is used to draw pictures and other complex layout like graphics, text and widgets.

The general syntax is:

w = Canvas(master, option=value)

master is the parameter used to represent the parent window.

3. CheckButton: To select any number of options by displaying a number of options to a user as toggle buttons. The general syntax is:

w = CheckButton(master, option=value)

master is the parameter used to represent the parent window.

4. Entry:It is used to input the single line text entry from the user.. For multi-line text input, Text widget is used.

The general syntax is:

w=Entry(master, option=value)

master is the parameter used to represent the parent window.

5. Frame: It acts as a container to hold the widgets. It is used for grouping and organizing the widgets. The general syntax is:

w = Frame(master, option=value)

master is the parameter used to represent the parent window

6. Label: It refers to the display box where you can put any text or image which can be updated any time as per the code.

The general syntax is:

w=Label(master, option=value)

master is the parameter used to represent the parent window.

7. Listbox: It offers a list to the user from which the user can accept any number of options.

The general syntax is:

w = Listbox(master, option=value)

master is the parameter used to represent the parent window.

8. MenuButton: It is a part of top-down menu which stays on the window all the time. Every menubutton has its own functionality. The general syntax is:

w = MenuButton(master, option=value)

master is the parameter used to represent the parent window.

9. Menu: It is used to create all kinds of menus used by the application.

The general syntax is:

w = Menu(master, option=value)

master is the parameter used to represent the parent window.

10. Message: It refers to the multi-line and non-editable text. It works same as that of Label.

The general syntax is:

```
w = Message(master, option=value)
```

master is the parameter used to represent the parent window.

11. Text: To edit a multi-line text and format the way it has to be displayed.

The general syntax is:

```
w  =Text(master, option=value)
```

master is the parameter used to represent the parent window.

THANKYOU