2024387 .

Car Rental and Buying Management System

Quick Submit

Quick Submit

Ghulam Ishaq Khan Institute of Engineering Sciences & Technology, Topi

Document Details

Submission ID

trn:oid:::1:3239894220

Submission Date

May 5, 2025, 2:58 PM GMT+5

Download Date

May 5, 2025, 3:56 PM GMT+5

File Name

2024387,_Cs112_project_report.docx

File Size

1.6 MB

15 Pages

775 Words

4,484 Characters

0% detected as AI

The percentage indicates the combined amount of likely AI-generated text as well as likely AI-generated text that was also likely AI-paraphrased.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Detection Groups

0 AI-generated only 0%
Likely AI-generated text from a large-language model.

0 AI-generated text that was AI-paraphrased 0%
Likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

Disclaimer
Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (it may misidentify writing that is likely AI generated as AI generated and AI paraphrased or likely AI generated and AI paraphrased writing as only AI generated) so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.

CS-112 Project : Car Rental and Buying Management System.

By

M.Hamdoon: 2024387, Basil Joseph: 2024138 ,

Yahya Khalid Khan : 2024664 ,M.Zeeshan Naseem:2024495

1.INTRODUCTION

The projects purpose is for the automation of Car renting and purchasing processes for both customers and administrators alike. This C++ project provides:

1.Proper protected authentication and verification for both customers and admins of the system

2.comprehensive management of different vehicles and their subtypes depending on the customers demands

3. Transaction process for either rent or purchasing of the chosen cars

4. Proper in depth receipts for any transaction made

This system not only solves the hardwired problem of manually keeping records of any transactions but also provides a digital solution to track transactions made by users .

2.Summary Of Functional Requirements:

Functional requirements are a set of basic instructions or functionalities that a system follows to be able to smoothly perform the tasks given by the users

1.User Authentication:

⇢Registration and Login for customers

⇢Admin Login based on a predetermined login info providing special privileges to manage the system

2.Car Managing:

⇢Add and Remove any Cars and their subtypes from inventory (**admin only**)

⇢Search Vehicles by inputting their make and model and then also displaying the available vehicles (**customer specific**)

3.Transactions:

⇢Process Renting of cars with their return dates

⇢Process purchasing of vehicles with ability to add warranty.

4.Administrative Functions:

⇢Adding and removing of cars from inventory

⇢Managing car and their subtypes inventory

5.Other Features:

⇢Menus are displayed and repeated for each action and part of the system

⇢upon exiting the program the user is thanked!

3.Tools and Technologies Used:

⇢C++ is used as a programming language

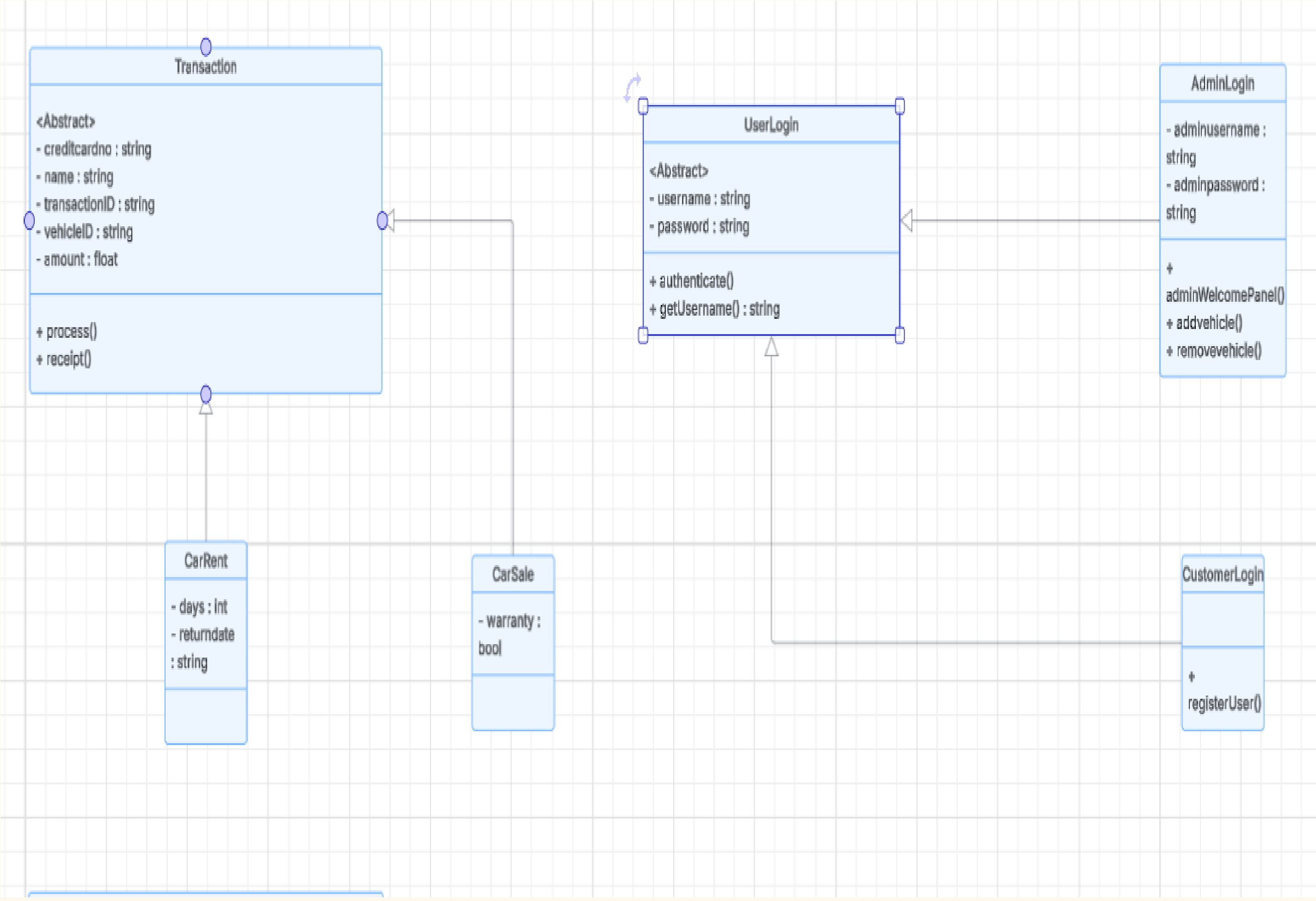
⇢Some of the libraries used are :**<iostream>,<ifstream>,<ctime>,<iwindows.h>,<string>**

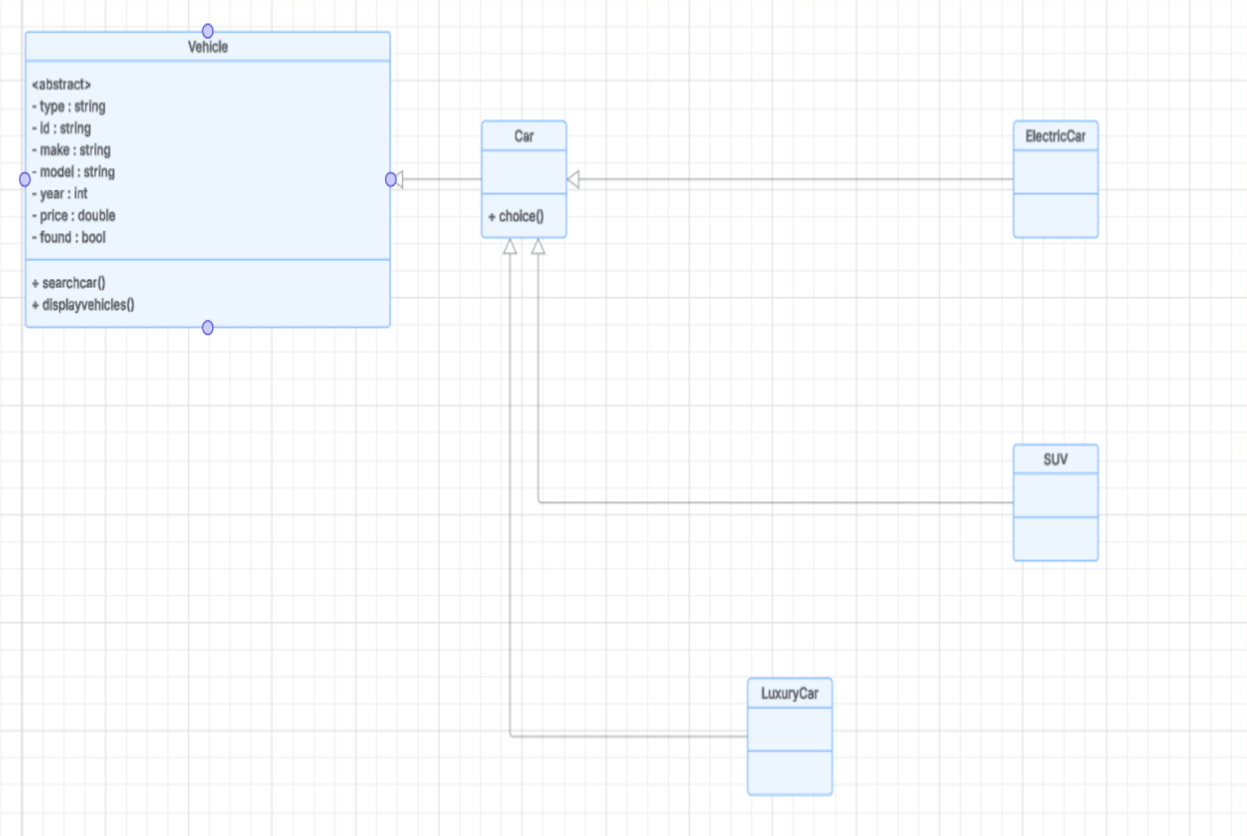
⇢Interface used is Command line interface

⇢VsCode and Devc++ were used as the IDEs

⇢Windows was used for the OS.

4.System design:


Transaction
<Abstract>
- creditcardno : string
- name : string
- transactionID : string
- vehicleID : string
- amount : float
+ process()
+ receipt()
UserLogin
<Abstract>
- username : string
- password : string
+ authenticate()
+ getUsername() : string
AdminLogin
- adminusername : string
- adminpassword : string
+ adminWelcomePanel()
+ addvehicle()
+ removevehicle()
CarRent
- days : int
- returndate : string
CarSale
- warranty : bool
CustomerLogin
+ registerUser()



5.Concepts Used And Their Implementation:

1.Encapsulation:

It is used throughout the program to hide details regarding the said processes

⇢private/protected members are used throughout all classes

⇢For e.g in the transaction class private data like credit card number is private whereas interface processes like process() and receipt() are made public

⇢user login class privates the username and password only being able to access them using getter methods

2.Inheritance:

It is used throughout the program to create hierarchy between classes

⇢**Vehicle abstract base class** is inherited by car which is further inherited by specialized types like **suv,electric car,luxury car.**

⇢Transaction base class is inherited by carrent and carsale classes aswell

⇢**User login** abstract baseclass is inherited by **adminlogin and customerlogin**

3.Polymorphism:

It is implemented in several ways:

⇢Transaction base class showcases the use of pure virtual functions like **process() and receipt()** which are after that overridden in derived classes like **carrent and carsale** with their individual implementations

⇢**vehicle and userlogin** are abstract base classes that contain pure virtual functions like **searchcar( ), display vehicles( ), authenticate( )** forcing derived classes to implement these methods so that they dont get abstract themselves

4.Abstraction:

It is implemented in several ways:

⇢Abstract base classes like v**ehicle, userlogin and transaction** define interfaces without any implementation

⇢hiding complex operations like process() in transaction class

6.CHALLENGES:

⇢**File permission issues:**Soemtimes it would say permission denied when trying to remove files or access files in the program. So we tried to implement proper file handling with error checking

⇢**Irregular looping problems:** sometimes when printing the receipt the program would give improper loops that go on till infinite resolution fixed it by adding breaks in the loops where required

⇢**Files not displaying:** sometimes the cars would not display properly due to invalid string formatting in files so we fixed it with proper formatting

7.TESTING:

⇢**Unit Testing:**

Verified correct login/logout functionality and tested all admin operations to add and remove cars. Also tested all transaction and receipt processes

⇢**Integration Testing:**

Verified all the work done from login to transaction part of the program

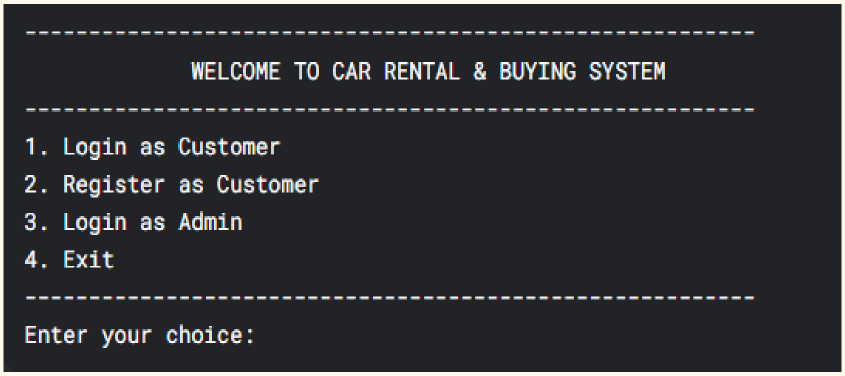
⇢**Edge cases:**

Using numbers as input instead of string when required causes infinite loops.

Empty files causes nothing to display aswell.

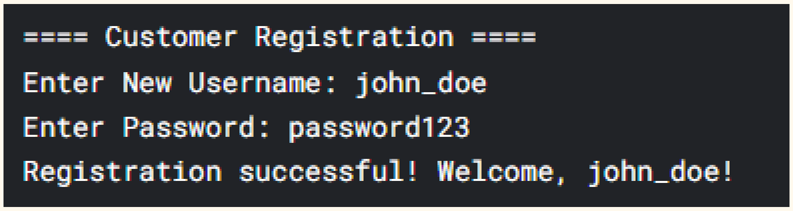
8.OUTPUTS:

1.Main Menu:



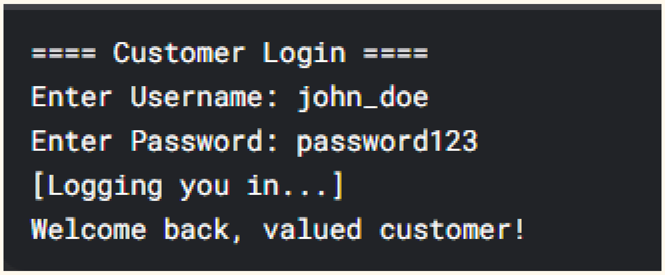
```
---------------------------------------------------------
             WELCOME TO CAR RENTAL & BUYING SYSTEM
---------------------------------------------------------
1. Login as Customer
2. Register as Customer
3. Login as Admin
4. Exit
---------------------------------------------------------
Enter your choice:
```

2.Customer Reg:



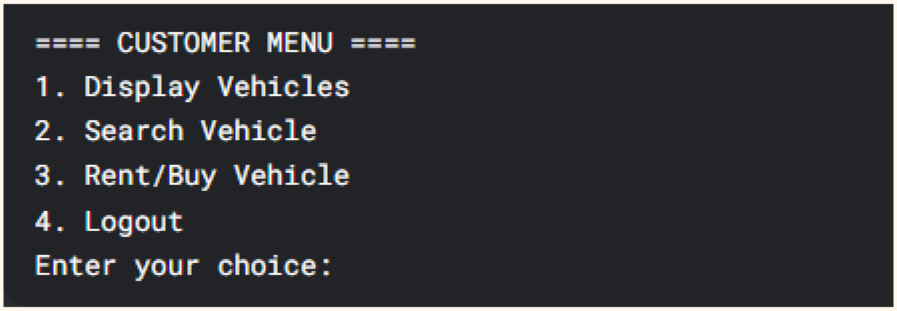
```
==== Customer Registration ====
Enter New Username: john_doe
Enter Password: password123
Registration successful! Welcome, john_doe!
```

3.Customer Login:



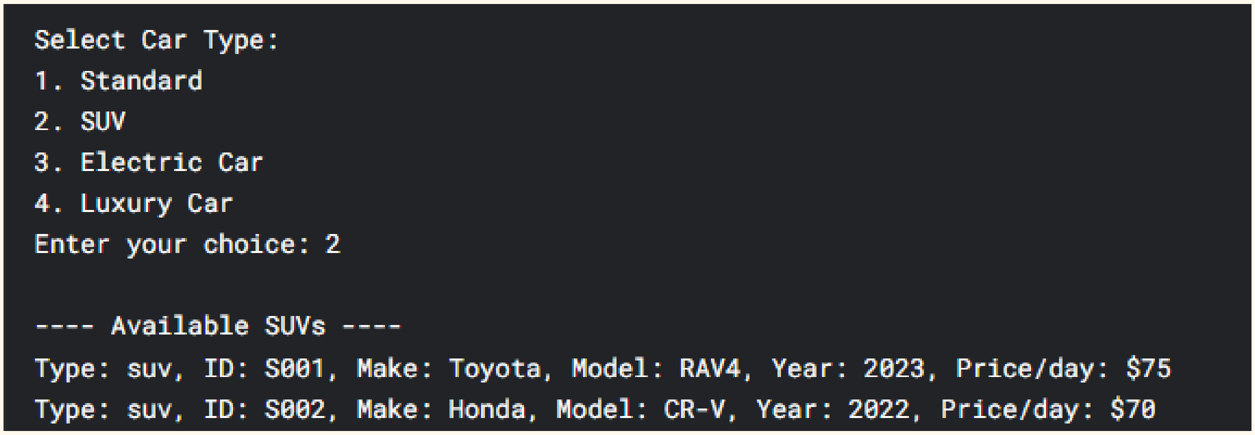
```
==== Customer Login ====
Enter Username: john_doe
Enter Password: password123
[Logging you in...]
Welcome back, valued customer!
```

4.Customer Menu:



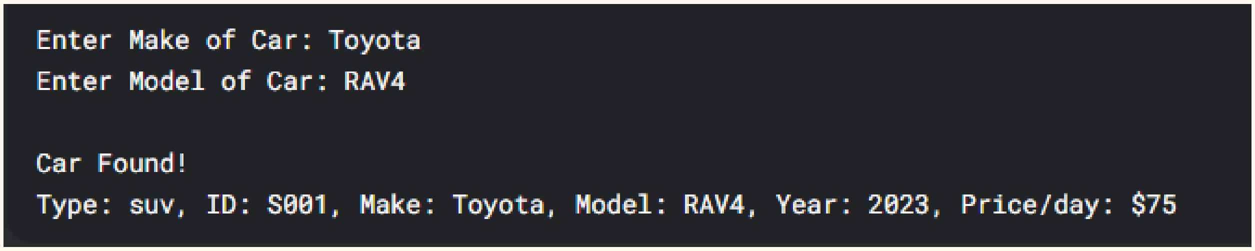
```
==== CUSTOMER MENU ====
1. Display Vehicles
2. Search Vehicle
3. Rent/Buy Vehicle
4. Logout
Enter your choice:
```

5.Display Vehicles:



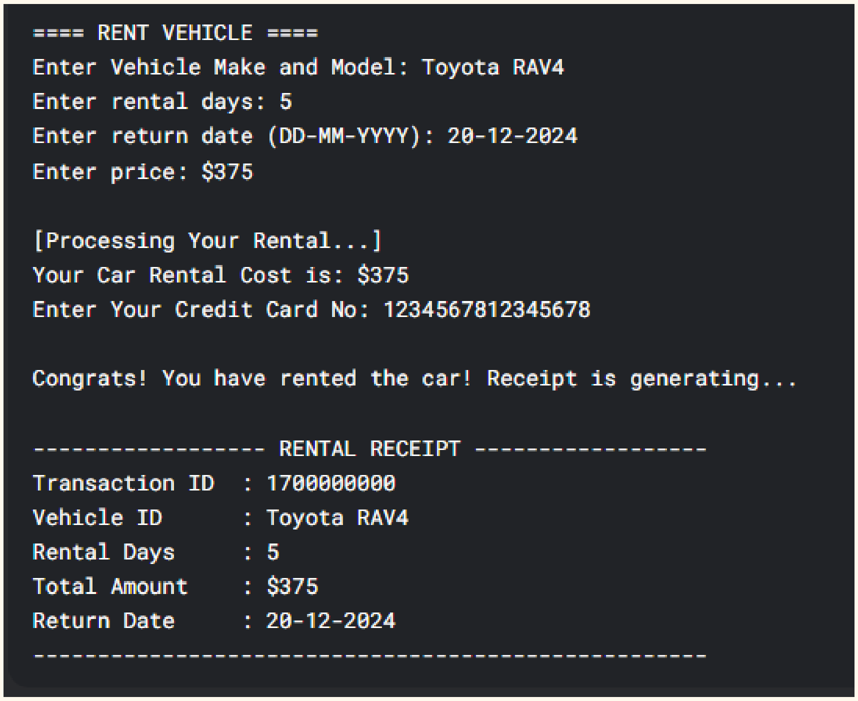
```
Select Car Type:
1. Standard
2. SUV
3. Electric Car
4. Luxury Car
Enter your choice: 2

---- Available SUVs ----
Type: suv, ID: S001, Make: Toyota, Model: RAV4, Year: 2023, Price/day: $75
Type: suv, ID: S002, Make: Honda, Model: CR-V, Year: 2022, Price/day: $70
```

6.Search Vehicles:



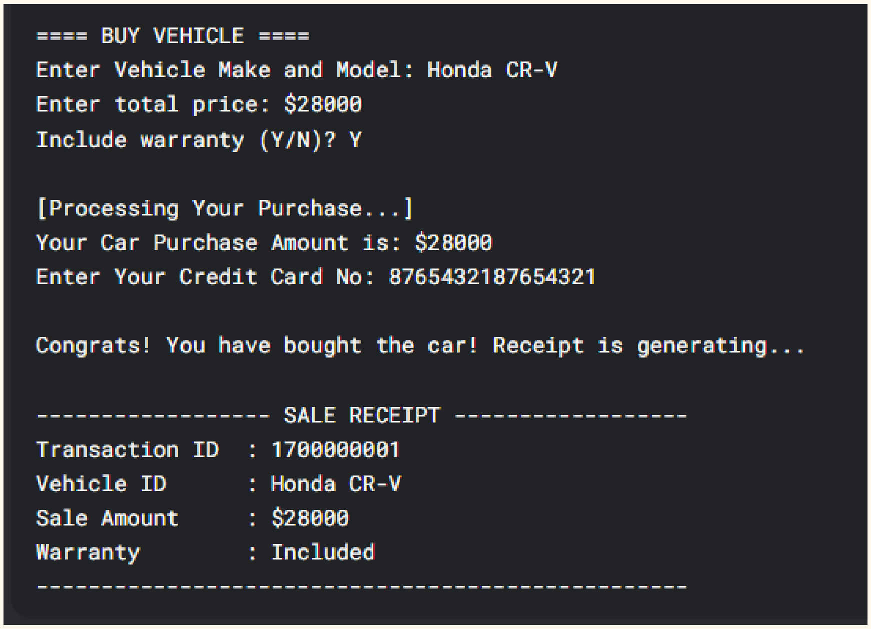
```
Enter Make of Car: Toyota
Enter Model of Car: RAV4

Car Found!
Type: suv, ID: S001, Make: Toyota, Model: RAV4, Year: 2023, Price/day: $75
```

7.Rent Vehicles:



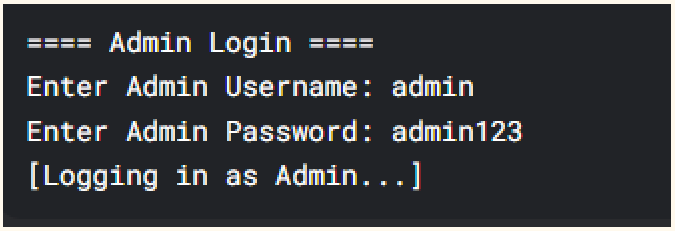
```
==== RENT VEHICLE ====
Enter Vehicle Make and Model: Toyota RAV4
Enter rental days: 5
Enter return date (DD-MM-YYYY): 20-12-2024
Enter price: $375

[Processing Your Rental...]
Your Car Rental Cost is: $375
Enter Your Credit Card No: 1234567812345678

Congrats! You have rented the car! Receipt is generating...

------------------ RENTAL RECEIPT ------------------
Transaction ID   : 1700000000
Vehicle ID       : Toyota RAV4
Rental Days      : 5
Total Amount     : $375
Return Date      : 20-12-2024
----------------------------------------------------
```

8.Buy Vehicle:



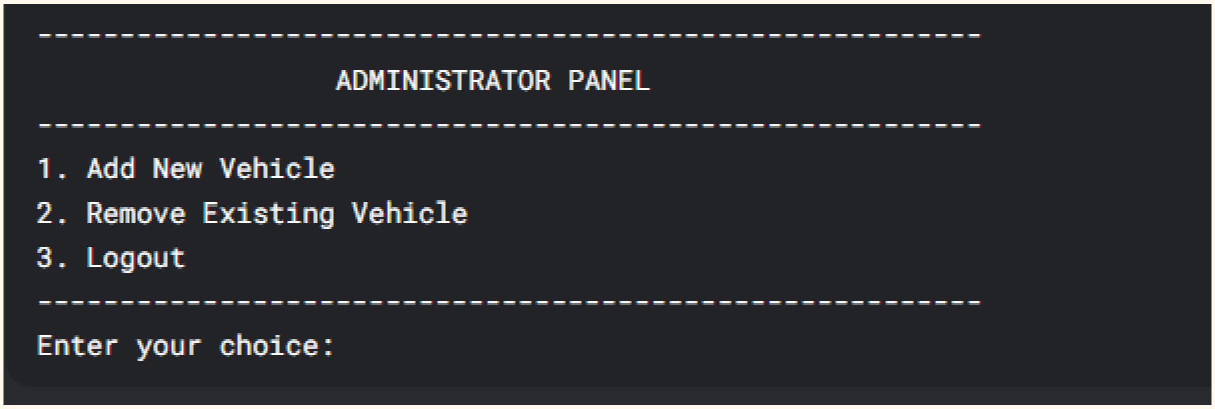
```
==== BUY VEHICLE ====
Enter Vehicle Make and Model: Honda CR-V
Enter total price: $28000
Include warranty (Y/N)? Y

[Processing Your Purchase...]
Your Car Purchase Amount is: $28000
Enter Your Credit Card No: 8765432187654321

Congrats! You have bought the car! Receipt is generating...

------------------ SALE RECEIPT ------------------
Transaction ID   : 1700000001
Vehicle ID       : Honda CR-V
Sale Amount      : $28000
Warranty         : Included
--------------------------------------------------
```

9.Admin LOGIN:



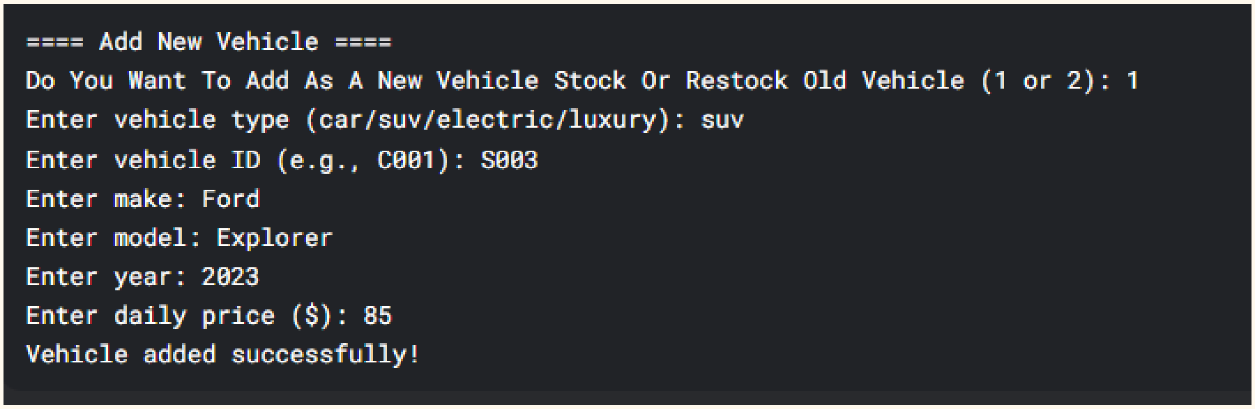
```
==== Admin Login ====
Enter Admin Username: admin
Enter Admin Password: admin123
[Logging in as Admin...]
```

10.Admin Panel:



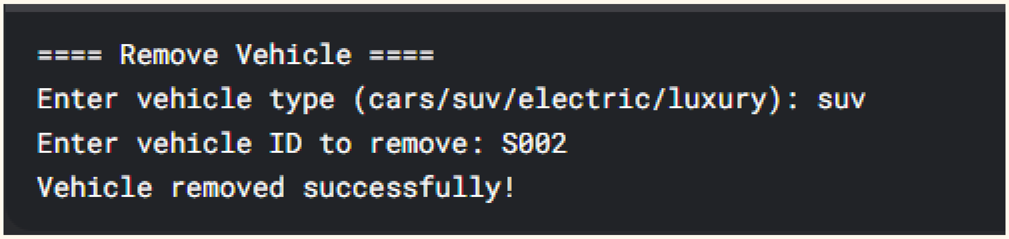
```
----------------------------------------------------------
                  ADMINISTRATOR PANEL
----------------------------------------------------------
1. Add New Vehicle
2. Remove Existing Vehicle
3. Logout
----------------------------------------------------------
Enter your choice:
```

11.Add Vehicle:



```
==== Add New Vehicle ====
Do You Want To Add As A New Vehicle Stock Or Restock Old Vehicle (1 or 2): 1
Enter vehicle type (car/suv/electric/luxury): suv
Enter vehicle ID (e.g., C001): S003
Enter make: Ford
Enter model: Explorer
Enter year: 2023
Enter daily price ($): 85
Vehicle added successfully!
```

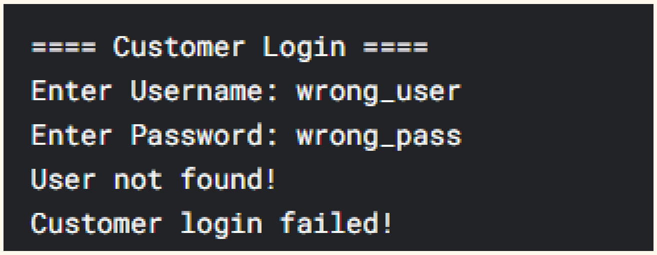
12.Remove Vehicle:



```
==== Remove Vehicle ====
Enter vehicle type (cars/suv/electric/luxury): suv
Enter vehicle ID to remove: S002
Vehicle removed successfully!
```

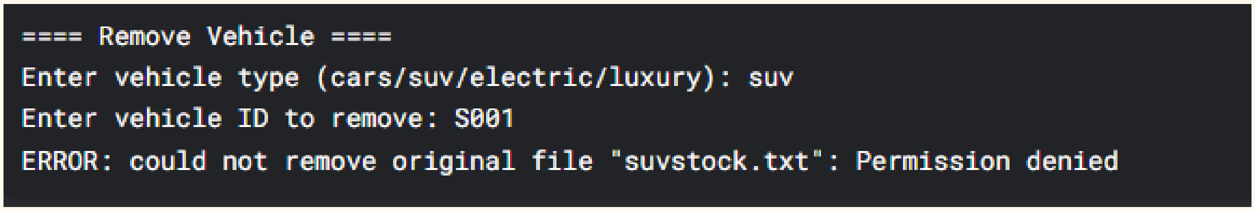
13.ERROR CASES:

A.Invalid Login:



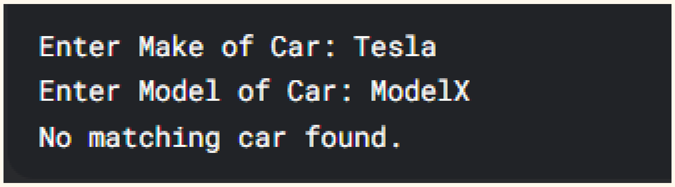
```
==== Customer Login ====
Enter Username: wrong_user
Enter Password: wrong_pass
User not found!
Customer login failed!
```

B.File Permission Error:



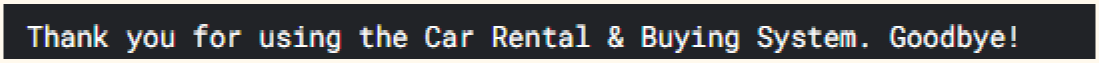
```
==== Remove Vehicle ====
Enter vehicle type (cars/suv/electric/luxury): suv
Enter vehicle ID to remove: S001
ERROR: could not remove original file "suvstock.txt": Permission denied
```

C.Vehicle Not Found:



```
Enter Make of Car: Tesla
Enter Model of Car: ModelX
No matching car found.
```

14.Program Exit:



```
Thank you for using the Car Rental & Buying System. Goodbye!
```

9.Areas to Improve Upon:

1.Better reports for the admin:

⇢Generate daily and weekly reports for the admin to analyze and export them to an excel sheet

2.Filter Options:

⇢Filter vehicles based on their price ranges and search by more criterias aswell

3.Better Payment:

⇢Support for multiple payment method and confirmation email for payment

4.Gui Implementation:

⇢switch from CLI to GUI