PROJECT REPORT

Submitted by

Cevin Christo	**RA2211042010002**
Neel Delivala	**RA2211042010004**
Raghavendra Pratap Bohra	**RA2211042010035**
Rajitha Ananthakrishnan	**RA2211042010038**

Under the Guidance of

Dr G ELANGOVAN

Assistant Professor, Department of Data Science and Business Systems

In partial satisfaction of the requirements for the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND BUSINESS SYSTEMS

SCHOOL OF COMPUTING

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR - 603203

MAY 2023

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
Deemed to be University u/s 3 of UGC Act, 1956

SRM INSTITUTION OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR-603203

BONAFIDE CERTIFICATE

Certified that this Project Report titled **"Car rental system"** is the bonafide work done by **Cevin Christo (RA2211042010002), Neel Delivala (RA2211042010004), Raghavendra Pratap Bohra (RA2211042010035) and Rajitha Ananthakrishnan (RA2211042010039)** who completed the project under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other work.

SIGNATURE

Dr. G Elangovan

OODP – Course Faculty
Assistant Professor
Department of Data Science and Business Systems
SRMIST

SIGNATURE

Dr. M Lakshmi

Head of the Department
Department of Data Science and Business Systems
SRMIST

TABLE OF CONTENTS

Problem Statement:

Design and implement a car rental system that efficiently manages the rental process, provides a user-friendly interface for customers, and enables effective management of car inventory for the rental agency.

The car rental system should address the following challenges:

1. User Management: The system should support user registration, authentication, and authorisation, allowing both customers and staff members to have different levels of access and privileges.

2. Vehicle Inventory Management: The system should provide functionalities for adding new vehicles to the inventory, updating their availability status, and removing them when they are no longer in service or need maintenance.

3. Reservation and Booking Management: Customers should be able to search for available cars based on their preferred location, date, and other specifications. The system should handle reservations, including checking availability, calculating rental costs, and generating invoices. It should also support modifications and cancellations of reservations.

4. Rental Duration and Pricing: The system should calculate rental durations accurately, considering factors such as pick-up and return dates and times. It should also calculate the total rental cost based on factors like the type of vehicle, rental duration, additional services (e.g., insurance, GPS), and any discounts or promotions applied.

5. Reporting and Analytics: The system should provide management with comprehensive reports and analytics to monitor business performance, including data on reservations, revenue, popular car models, and customer feedback. These insights can help in making informed decisions and optimising operations.

6. Customer Support: The system should provide customer support features, such as a help desk or ticketing system, to address customer inquiries, complaints, and requests for assistance effectively.

7. Integration and Scalability: The system should be designed in a modular and extensible manner to allow for future integration with other systems or third-party services (e.g., GPS tracking, online travel agencies). It should also be scalable to accommodate increasing numbers of vehicles, customers, and transactions.

8. Security and Privacy: The system should prioritize the security of customer and transaction data, implementing appropriate encryption and access control measures. It should also comply with relevant data protection regulations to ensure customer privacy.

By addressing these challenges, the car rental system will streamline the rental process, enhance customer satisfaction, and enable efficient management of the rental agency's operations.

Modules of project :

In an Object-Oriented Design and Programming (OODP) approach, the car rental system can be divided into several modules to encapsulate different functionalities and promote modularity. Here are some possible modules for the car rental system:

1. User Management Module:

- User Registration: Handles user registration and account creation.
- Authentication: Manages user authentication and login.
- Authorisation: Controls user access privileges and permissions.

2. Vehicle Management Module:

- Vehicle Inventory: Manages the collection of available vehicles, including adding, updating, and removing vehicles.
- Vehicle Availability: Tracks the availability status of each vehicle.
- Vehicle Maintenance: Handles maintenance and service scheduling for vehicles.

3. Reservation Module:

- Reservation Management: Facilitates the process of making and managing reservations, including checking availability, calculating costs, and generating invoices.
- Reservation Modification/Cancellation: Allows customers to modify or cancel existing reservations.

4. Rental Module:

- Rental Process: Manages the process of renting a vehicle, including checking availability, calculating rental duration, applying pricing rules, and generating rental agreements.

5. Integration Module:

- Third-Party Integrations: Integrates with external services or APIs, such as GPS tracking, online payment platforms, or online travel agencies.

6. Security Module:

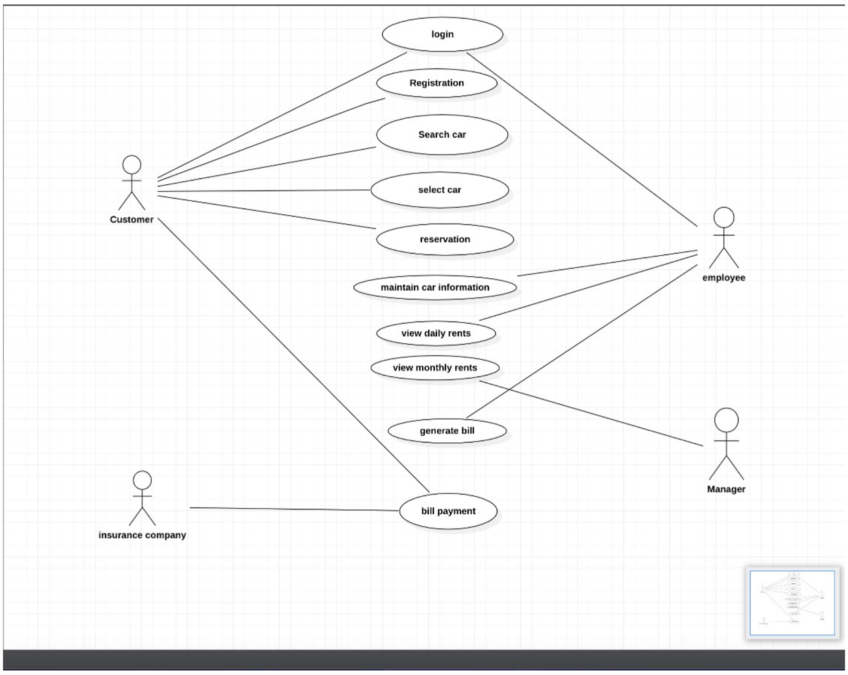
- Data Encryption: Implements encryption techniques to secure sensitive data.
- Access Control: Manages user access to system features and data.
- Privacy Compliance: Ensures compliance with data protection regulations.

These modules represent a high-level breakdown of the car rental system, focusing on the major functionalities and responsibilities. However, the actual design may vary based on specific project requirements and design decisions.

UML Diagrams:

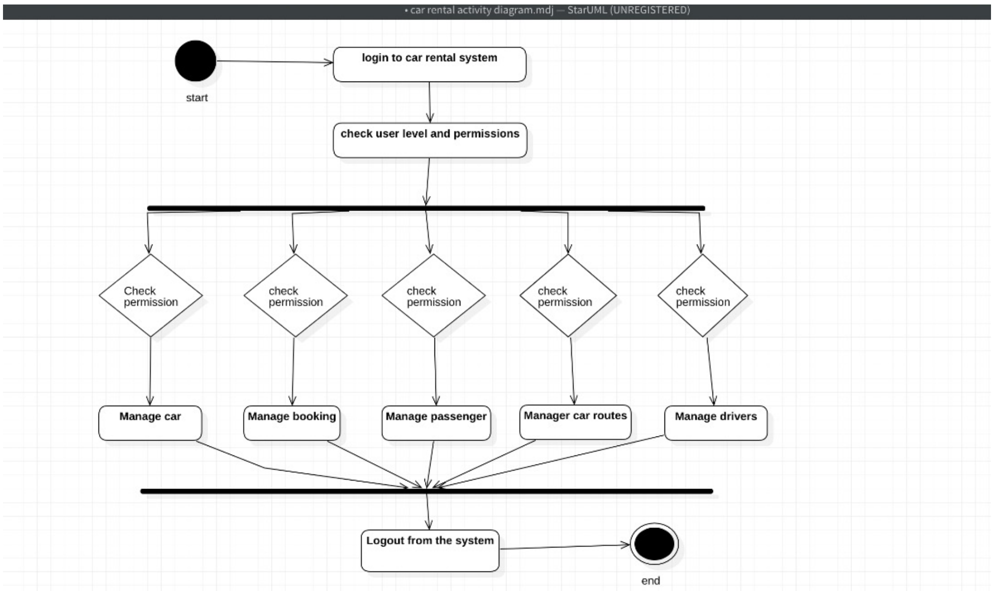
1. Use Case Diagram:

The designed car rental system use case diagram has two main illustrations. These illustrations describe the system's general processes and specific processes using include and extend.The general use case diagram for car rental system shows the general functions that the system could do. It is based on the transactions needed in rental management.



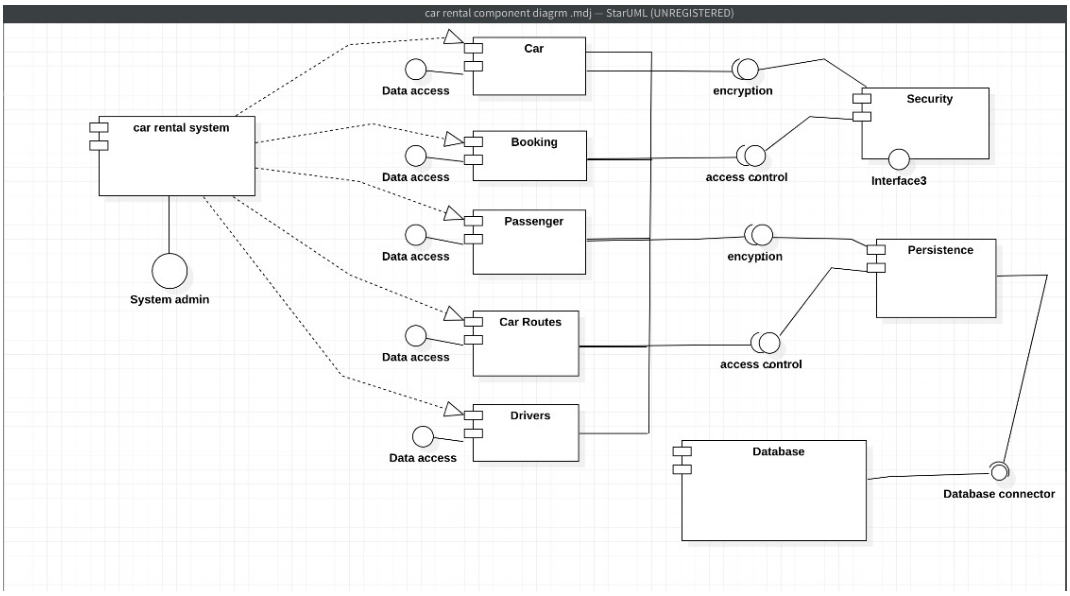
2. Activity diagram:

The UML **activity diagram for car rental system** is a diagram that presents the flow of system activities. It is one of the methods used to document the system behavior in terms of activities and development.



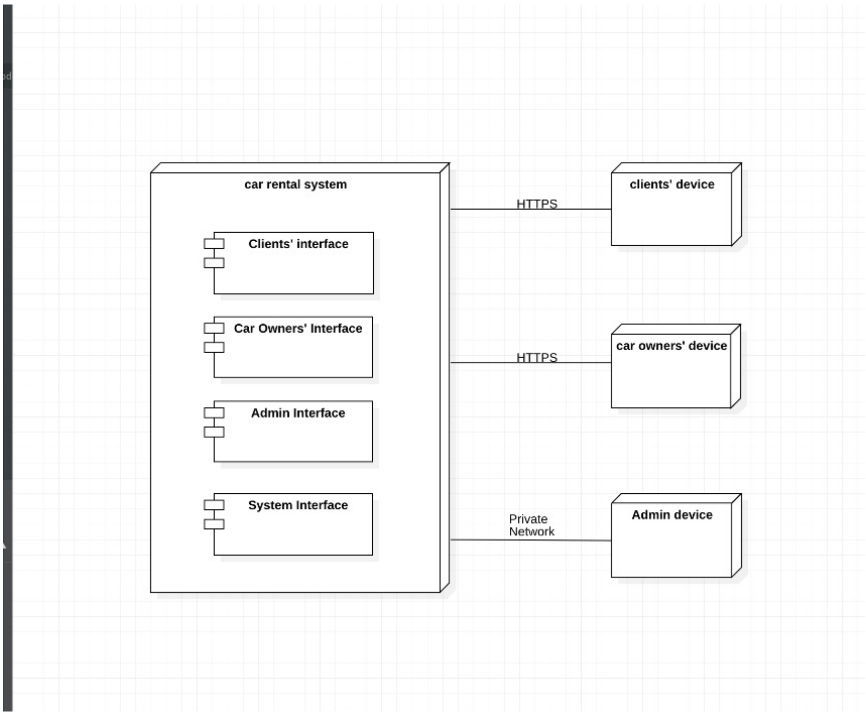
3.Component diagram:

The component diagram for car rental system is used to show how the parts work together to make the car system operate correctly. A component diagram shows how the software's parts are organised and how they depend on each other. This diagram gives a high-level look at the parts of a system.



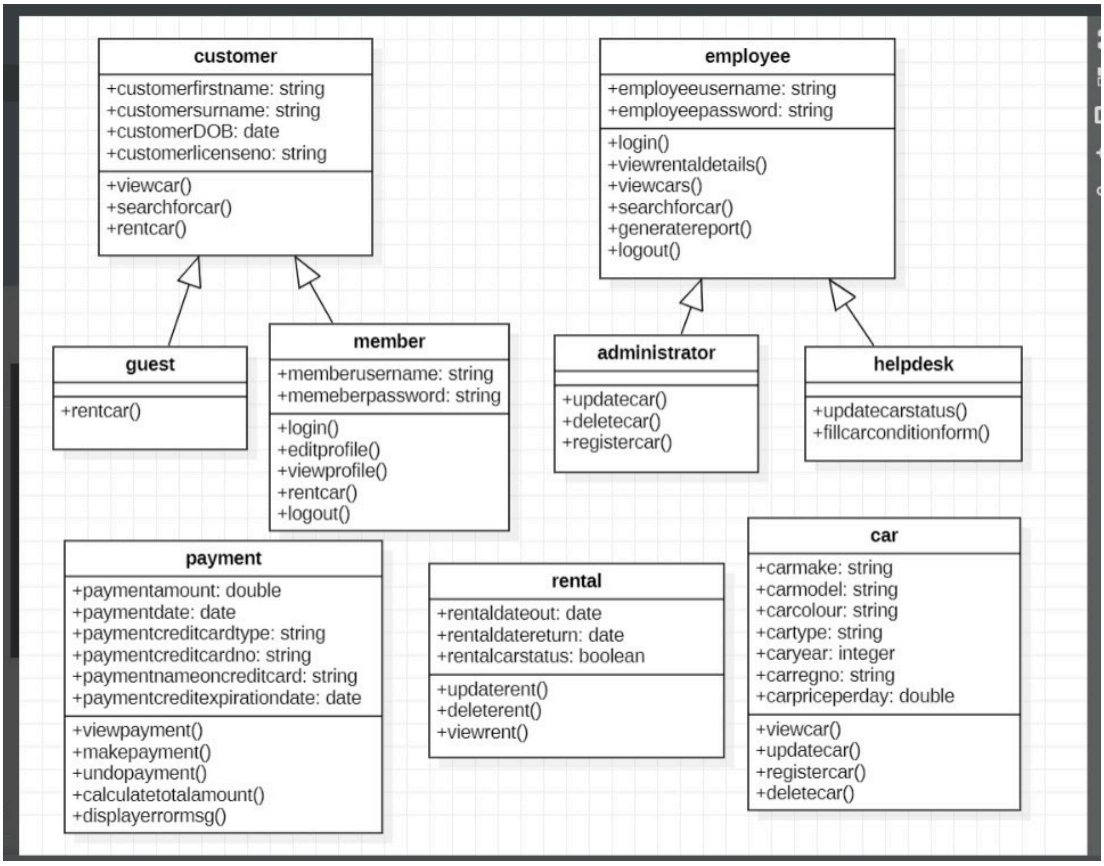
4. Deployment diagram:

The deployment diagram for the car rental system shows the system's architectural structure through nodes and their association. This UML deployment diagram presents the relationships between nodes (software and hardware), as well as the physical distribution of car rental processes.

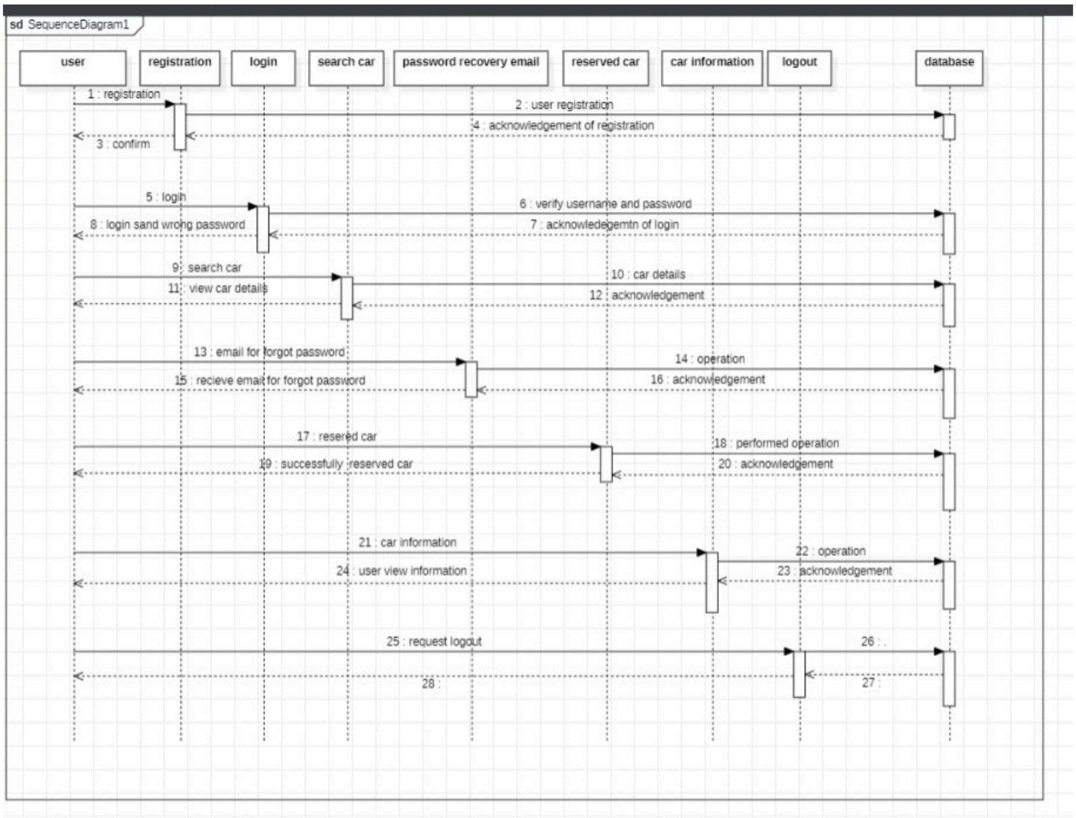


5. Class diagram:

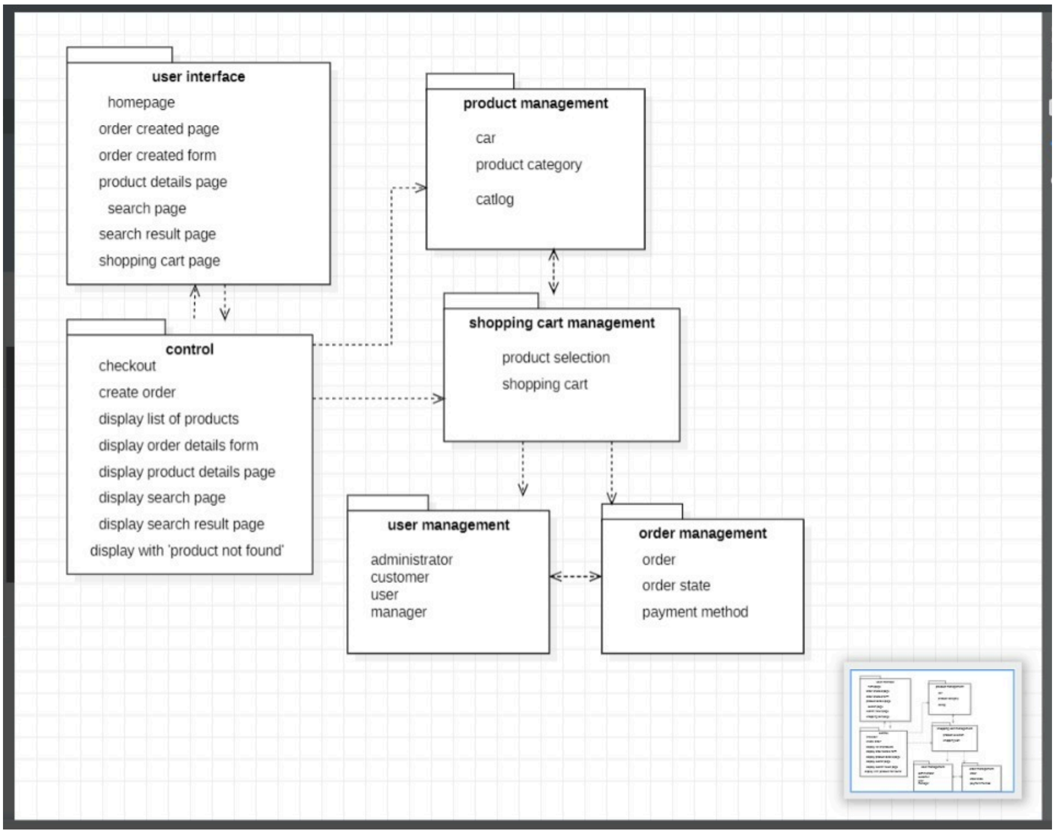
Class diagram is a designed structure that shows the systems' classes and their relationships. This UML Class Diagram is made to guide programmers along with the Online Car Rental system development.It saves both time and money.Customers can get immediate assistance from the rental system.Customers can book their rentals quickly and easily using the rental system.It saves paper by eliminating the need for contracts, invoicing, and other paperwork.



6. Sequence diagram:



7. Package diagram-



Code-

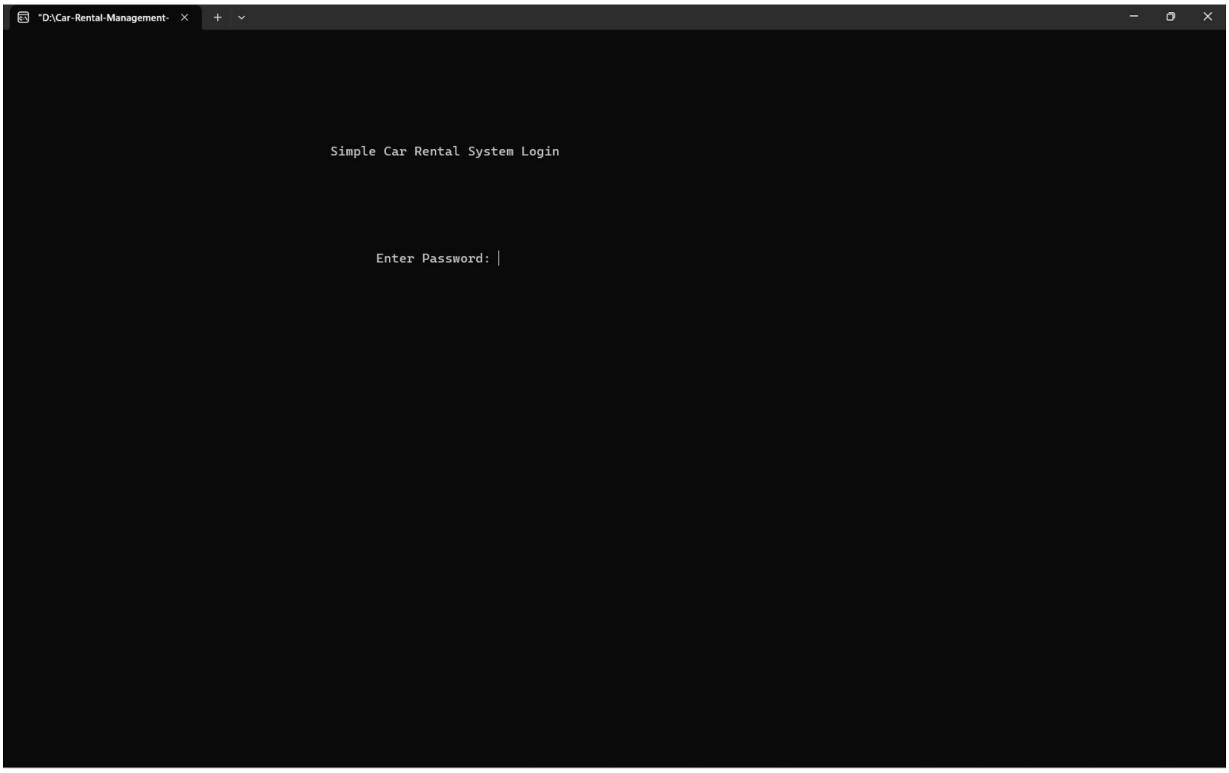
```
#include <iostream>
#include <conio.h>
#include <stdlib.h>
using namespace std

 struct Cars

                        6        "Hyundai Verna" "Toyota Innova Crysta" "Maruti Brezza" "Mahindra
XUV300" "Honda City" "BMW 320i"

    string model [6]= {"2023","2.4GX 7 STR","LXI","XUV 300 W4","City SV","2020 320i"};

    string seats [6]={"5","7","5","5","5","5"};

    string mileage [6]={"20.6 Km/l ","12 Km/l","19.8 Km/l","20 Km/l","18.4 Km/l","13.2 Km/l"};

    int price [6] ={250,341,216,208,235,667} ;




   };

   struct Lessee

   string Name [6];
   int payment_acc [ 6] ;
   };

 Cars car ;
 Lessee lessee ;

   void Minu
 {


   int num=1;
 for(int i=0 ;i<6;++i)
```

```
{
            cout<<"\t\t\t";
   cout<<"Enter " <<num<<"\t- To Select  "<<car.mark[i]<<endl;
   num++ ;
}

}

void Details (int theChoice)
 {
            system("CLS");
                         "\n\n\n\t\t\t-----------------------------\n"
                "\t\t\tYou Have Selected - "                                1
                "\t\t\t-----------------------------\n\n\n"
    cout<<"\t\t\tModel : "<<car.model[theChoice-1]<<endl;
                "\t\t\tSeating Capacity : "                              1
    cout<<"\t\t\tMileage : "<<car.mileage[theChoice-1]<<endl;
    cout<<"\t\t\tPrice : "<<car.price[theChoice-1]<<"Rs/hr"<<endl;


 }

 void user_input (int theChoice)
  {
            system("CLS");
       int j=theChoice-1 ;
                  "\t\t\t----------------------------------------\n"
                  "\t\t\tPlease Provide Your Personal Details  : \n"
                  "\t\t\t----------------------------------------\n\n"
                  "\n\tNOTE: PROVIDE FIRST NAME ONLY, DONOT ENTER SPACE WHILE PROVIDING
NAME,\n\n\n\n\n"
                  "\t\t\tEnter Your Name : "
      cin>>lessee.Name[j];
       int hrs, pay_amt;
                   "\t\t\tNumber of hours car is to be rented:"
       cin>>hrs;
       pay_amt=hrs*car.price[j];
                   "\t\t\tAmount to be paid is:"                      "Rs.\n"

  }


int main
```

```
{
            int login();
    login();


         string decide ="yes" ;
                 "\t\t\t----------------------------------------------\n"
                 "\t\t\t\tSIMPLE CAR RENTAL SYSTEM \n"
                 "\t\t\tWelcome to Our Company ,Choose from the menu : "
                 "\t\t\t----------------------------------------------\n"
     while(decide!="exit")
     {
         Minu();
                 "\n\n\n\t\t\tYour Choice: "
     int theChoice ;
     cin>>theChoice ;
     Details(theChoice);
                 "\n\n\n\t\t\tAre You Sure, you want to rent this Car? (yes /no /exit ) : "
     cin>>decide ;
     if(decide=="yes") {
        user_input(theChoice);
                    "\n\n\t\t\tDo you want to continue ?(yes/no) "
        cin>>decide;
        if (decide=="no") break ;
system("CLS");
     }

   else
      if(decide=="no")
      {
            system("CLS");
         continue ;
      }
     else if  (decide=="exit")
     {
            system("CLS");

         break ;
     }

   }
     }
    return 0
}
int login
   string pass ="";
   char ch;
              "\n\n\n\n\n\n\n\t\t\t\t\t  Simple Car Rental System Login"
               "\n\n\n\n\n\n\n\t\t\t\t\t\tEnter Password: "
   ch = _getch();
   while(ch != 13){
     pass.push_back(ch);
      cout << '*';
      ch = _getch();
   }
   if(pass == "CSBS"){
                 "\n\n\n\n\t\t\t\t\t  Access Granted! Welcome To Our System \n\n"
      system("PAUSE");
      system("CLS");

   }else{
                 "\n\n\n\n\t\t\t\t\tAccess Aborted...Please Try Again!!\n"
      system("PAUSE");
      system("CLS");
      login();
   }
}
```

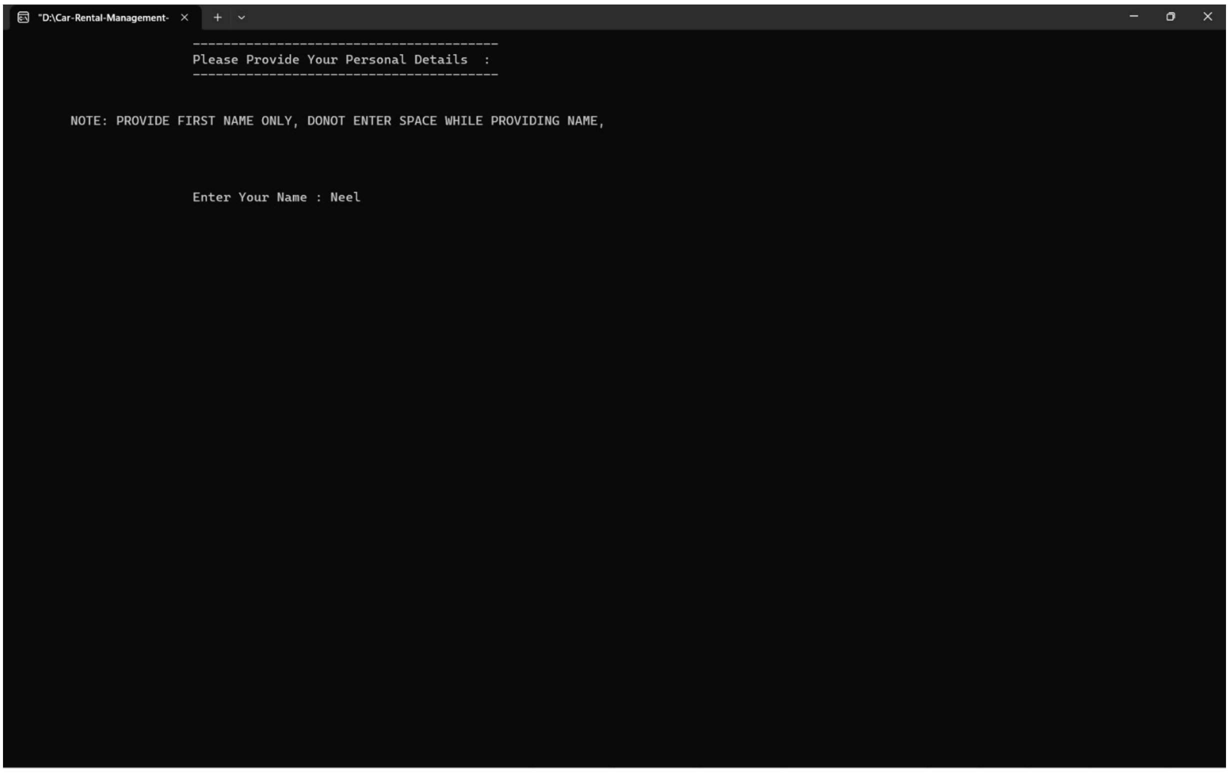
`

OUTPUT:

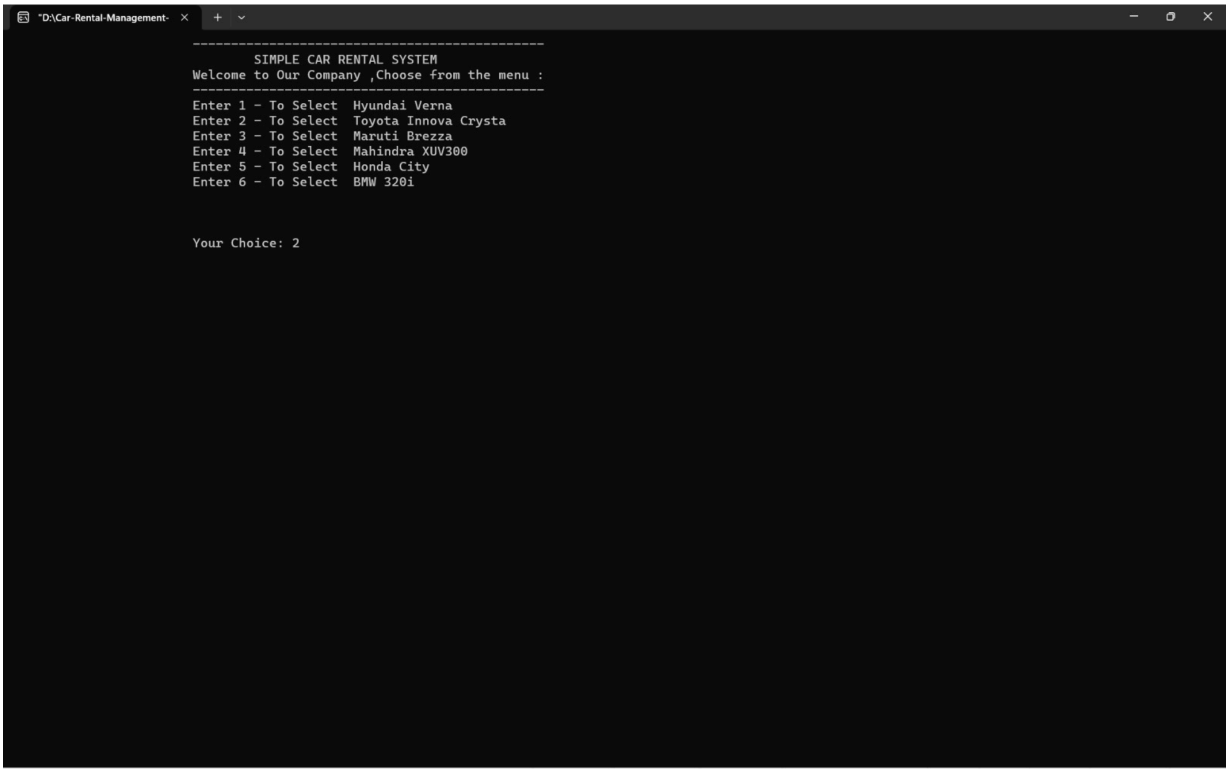


```
                          Simple Car Rental System Login



                              Enter Password: 
```

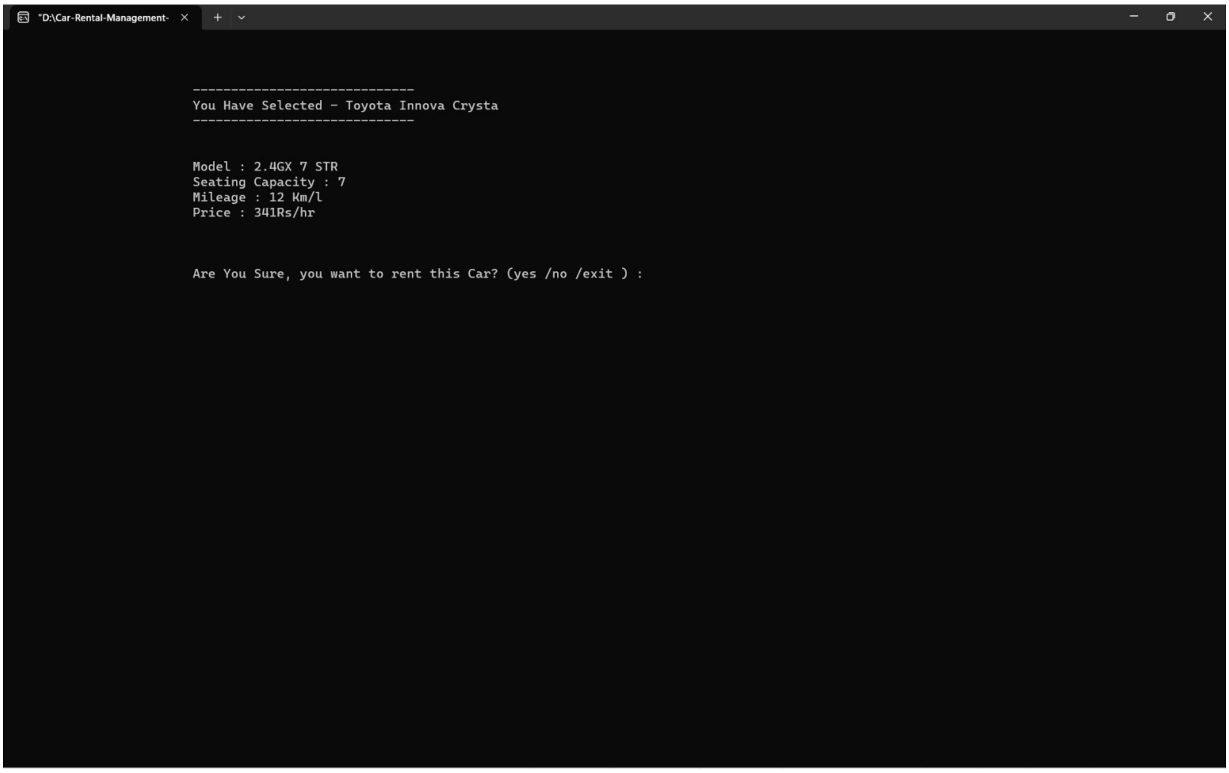
PASS- CSBS



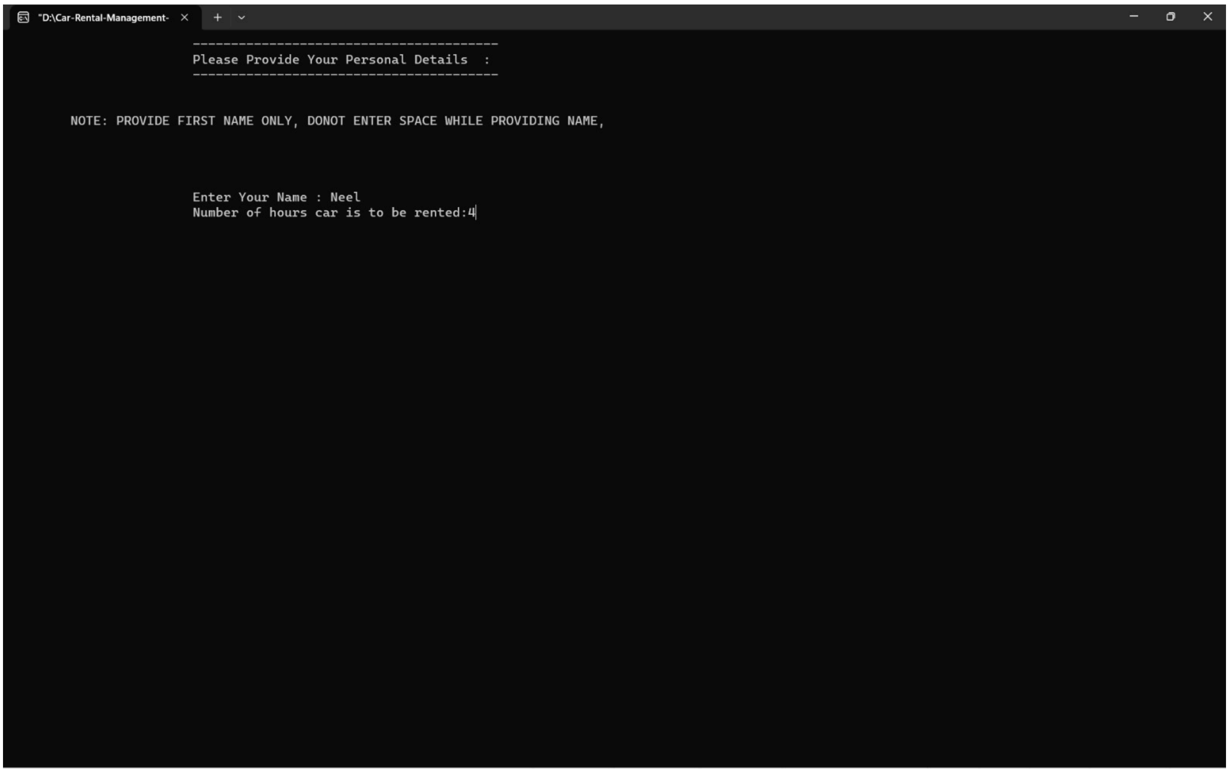
```
               ----------------------------------------
               Please Provide Your Personal Details  :
               ----------------------------------------


     NOTE: PROVIDE FIRST NAME ONLY, DONOT ENTER SPACE WHILE PROVIDING NAME,



               Enter Your Name : Neel
```



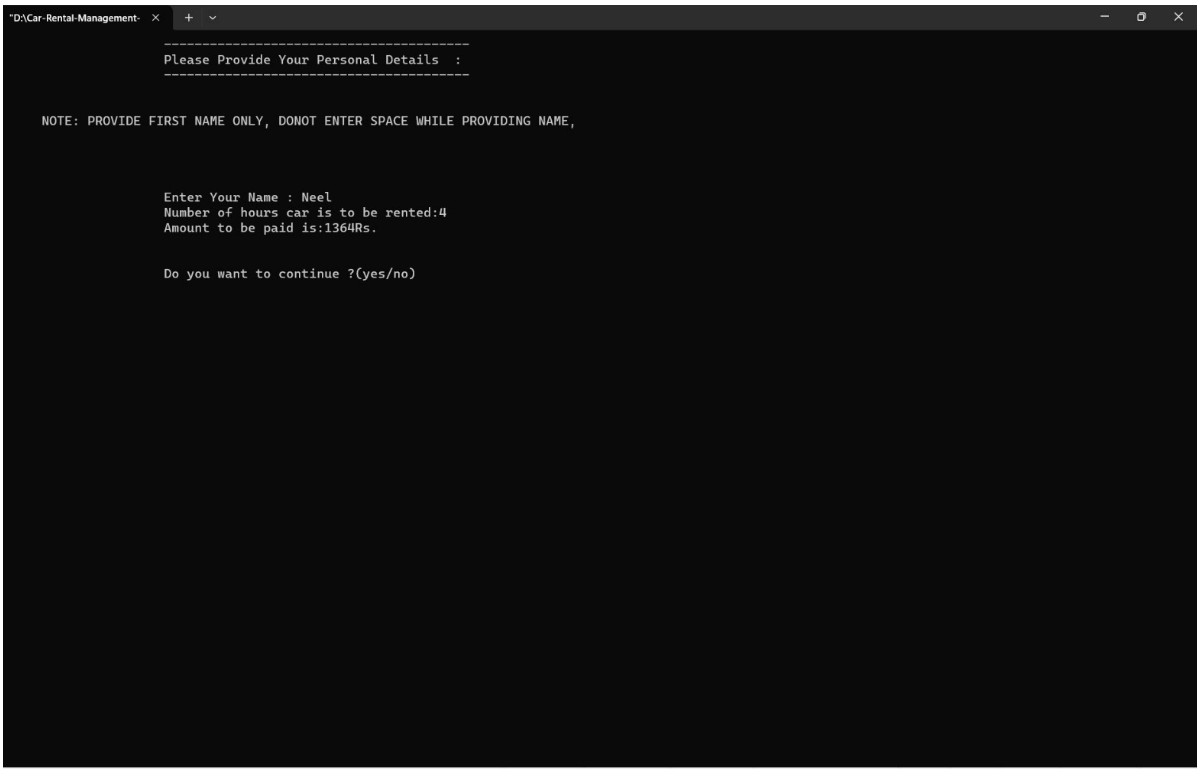
```
----------------------------------------------
        SIMPLE CAR RENTAL SYSTEM
Welcome to Our Company ,Choose from the menu :
----------------------------------------------
Enter 1 - To Select  Hyundai Verna
Enter 2 - To Select  Toyota Innova Crysta
Enter 3 - To Select  Maruti Brezza
Enter 4 - To Select  Mahindra XUV300
Enter 5 - To Select  Honda City
Enter 6 - To Select  BMW 320i



Your Choice: 2
```



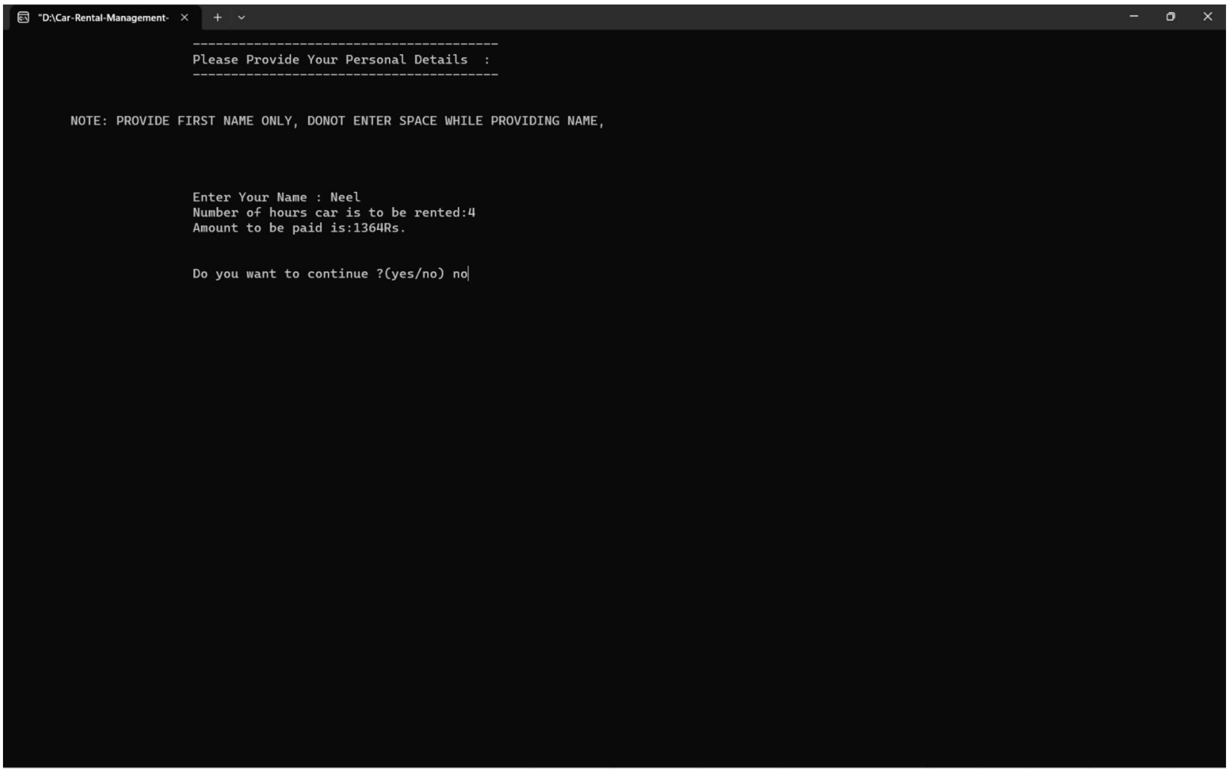
```
-----------------------------
You Have Selected - Toyota Innova Crysta
-----------------------------


Model : 2.4GX 7 STR
Seating Capacity : 7
Mileage : 12 Km/l
Price : 341Rs/hr



Are You Sure, you want to rent this Car? (yes /no /exit ) :
```



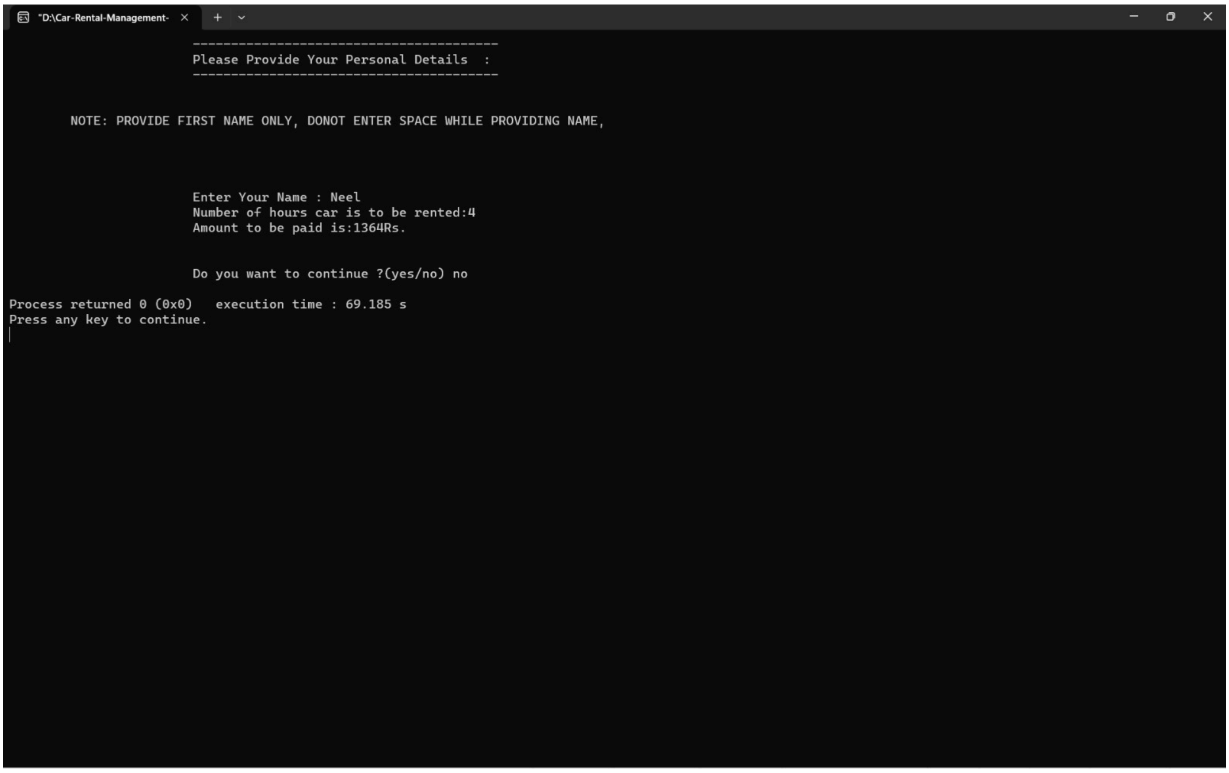
```
                   ----------------------------------------
                   Please Provide Your Personal Details  :
                   ----------------------------------------


       NOTE: PROVIDE FIRST NAME ONLY, DONOT ENTER SPACE WHILE PROVIDING NAME,



                   Enter Your Name : Neel
                   Number of hours car is to be rented:4
```



```
                   ----------------------------------------
                   Please Provide Your Personal Details  :
                   ----------------------------------------


       NOTE: PROVIDE FIRST NAME ONLY, DONOT ENTER SPACE WHILE PROVIDING NAME,



                   Enter Your Name : Neel
                   Number of hours car is to be rented:4
                   Amount to be paid is:1364Rs.


                   Do you want to continue ?(yes/no)
```



```
                        ----------------------------------------
                        Please Provide Your Personal Details  :
                        ----------------------------------------


        NOTE: PROVIDE FIRST NAME ONLY, DONOT ENTER SPACE WHILE PROVIDING NAME,



                        Enter Your Name : Neel
                        Number of hours car is to be rented:4
                        Amount to be paid is:1364Rs.


                        Do you want to continue ?(yes/no) no
```



```
                        ----------------------------------------
                        Please Provide Your Personal Details  :
                        ----------------------------------------


        NOTE: PROVIDE FIRST NAME ONLY, DONOT ENTER SPACE WHILE PROVIDING NAME,



                        Enter Your Name : Neel
                        Number of hours car is to be rented:4
                        Amount to be paid is:1364Rs.


                        Do you want to continue ?(yes/no) no

Process returned 0 (0x0)   execution time : 69.185 s
Press any key to continue.
```

Conclusion and Result for Car Rental System Project in OODP -

The Car Rental System project developed using Object-Oriented Design and Programming (OODP) has been successfully completed, resulting in a robust and efficient system for managing car rentals. The project aimed to provide a user-friendly interface for customers to browse and book cars, while also offering administrative features for managing the fleet, reservations, and customer data.

Key Features and Functionality:

1. User Registration and Authentication: The system allows users to create accounts, log in, and authenticate themselves securely. This ensures that only authorized users can access the system.

2. Car Inventory Management: The system provides an interface for the administrators to manage the car inventory. They can add new cars, update existing car information (e.g., model, year, availability, rental price), and remove cars from the system.

3. Reservation Management: Customers can search for available cars based on their preferences (e.g., location, date, car type) and make reservations. The system handles the reservation process, including checking availability, calculating rental fees, and generating invoices. Administrators can also view and manage reservations, such as approving or canceling them.

4. Billing : The system generates accurate invoices for each reservation, considering factors like rental duration, additional services, and applicable taxes.

5. User-Friendly Interface: The system offers an intuitive and user-friendly interface for both customers and administrators. Customers can easily browse and book cars, while administrators can efficiently manage the system's various aspects.

7. Scalability and Extensibility: The design of the system follows OODP principles, allowing for easy scalability and extensibility. New features and functionalities can be added in the future without significantly impacting the existing codebase.

Overall, the Car Rental System project successfully achieved its objectives and delivered a fully functional and reliable system for managing car rentals. It provides a seamless experience for customers to rent cars and offers a comprehensive set of tools for administrators to efficiently handle the rental operations. The system's robust design and functionality make it adaptable to the evolving needs of the car rental industry.

References-

1. geeksforgeeks.com

2. itssourcecode.com

3. The C++ Programming Language (4th Edition) By Bjarne Stroustrupetkar.

4. Let us C By Yashavant P Kanetkar.

5. Practical C++ Programming (2nd Edition) By Steve Oualline