

DEPARTMENT OF COMPUTER & INFORMATION SYSTEMS ENGINEERING
BACHELORS IN COMPUTER SYSTEMS ENGINEERING

Course Code: CS-116

Course Title: Object Oriented Programming

Complex Engineering Problem

FE Batch 2024, Spring Semester 2025

Grading Rubric

TERM PROJECT

Group Members:

Student No.	Name	Roll No.
S1	Hooria Ahmed	CS-24106
S2	Fariha Nadeem Khan	CS-24107
S3	Hibab Khan	CS-24108

CRITERIA AND SCALES				Marks Obtained		
				S1	S2	S3
Criterion 1: Does the application meet the desired specifications and produce the desired outputs? (CPA-1, CPA-3)						
1	2	3	4			
The application does not meet the desired specifications and is producing incorrect outputs.	The application partially meets the desired specifications and is producing incorrect or partially correct outputs.	The application meets the desired specifications but is producing incorrect or partially correct outputs.	The application meets all the desired specifications and is producing correct outputs.			
Criterion 2: How well is the code organization?						
1	2	3	4			
The code is poorly organized and very difficult to read.	The code is readable only to someone who knows what it is supposed to be doing.	Some part of the code is well organized, while some part is difficult to follow.	The code is well organized and very easy to follow.			
Criterion 3: How friendly is the application interface? (CPA-1, CPA-3)						
1	2	3	4			
The application interface is difficult to understand and use.	The application interface is easy to understand and but not that comfortable to use.	The application interface is very easy to understand and use.	The application interface is very interesting/ innovative and easy to understand and use.			
Criterion 4: How does the student performed individually and as a team member? (CPA-2, CPA-3)						
1	2	3	4			
The student did not work on the assigned task.	The student worked on the assigned task, and accomplished goals partially.	The student worked on the assigned task, and accomplished goals satisfactorily.	The student worked on the assigned task, and accomplished goals beyond expectations.			
Criterion 5: Does the report adhere to the given format and requirements?						
1	2	3	4			
The report does not contain the required information and is formatted poorly.	The report contains the required information only partially but is formatted well.	The report contains all the required information but is formatted poorly.	The report contains all the required information and completely adheres to the given format.			
Total Marks:						

CAR RENTAL SYSTEM REPORT

1. Problem Description:

Traditional car rental methods are time-consuming and inefficient due to manual processes, lack of real-time updates, and poor customer service. This highlights the need for modern, tech-driven car rental services that offer convenience, accessibility, and efficiency.

Our project aims to solve this problem by developing a Car Rental System with a Graphical User Interface (GUI) using Python. The system is designed to automate and streamline the rental process for both customers and administrators. It enables users to:

- Browse available cars,
- View car details and images,
- Sign up or log in securely,
- Rent cars for specified durations,
- Track their rental history.

Administrators can manage the car inventory, monitor rentals, and oversee the entire system. All data—such as user accounts, available cars, and rental transactions—is stored using CSV files to maintain simplicity and compatibility, while also ensuring data persistence across sessions.

2. Distinguishing Features / Object-Oriented Concepts Used:

The project implements multiple object-oriented programming (OOP) features:

- Abstraction: BaseUserInterface is an abstract class using the ABC module.
- Encapsulation: Sensitive logic and error messages are handled via the UserError class.
- Inheritance: Specific user roles (Customer, Admin) inherit from shared base classes.
- Polymorphism: Abstract methods like view_profile() and view_car_inventory() are implemented differently across classes.

Additional highlights:

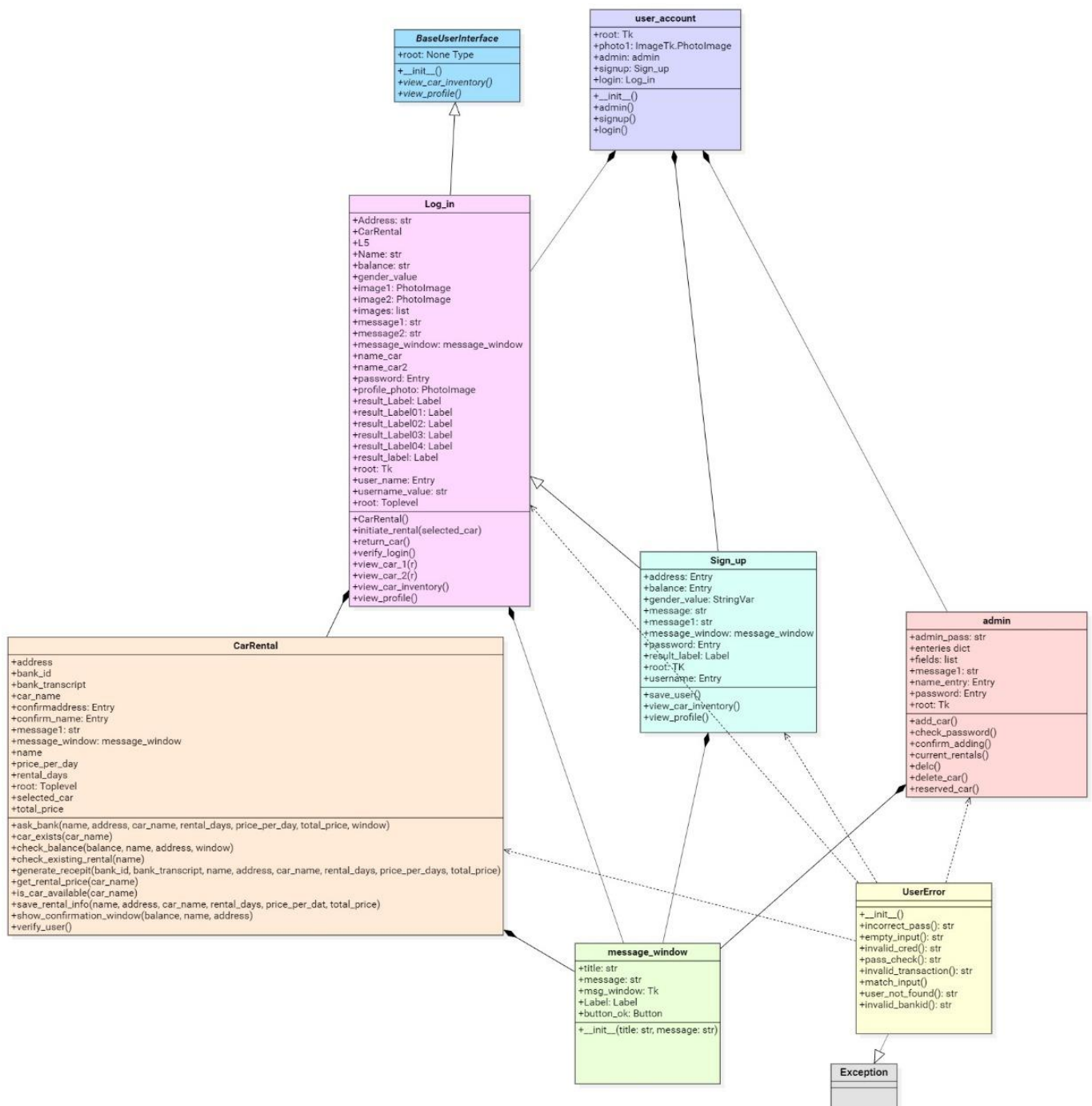
- Custom exception handling with meaningful error messages
- Modular structure separating UI logic, data handling, and business logic

3. Flow of the Project:

The application starts with a GUI interface allowing the user to either sign up or log in. Once authenticated, users interact with different features depending on their role (admin or customer). Admins can manage cars and monitor usage, while customers can browse, rent, and return cars.

Class relationships are based on OOP design. The abstract BaseUserInterface defines key methods implemented in AdminUI and CustomerUI classes.

4. Class Diagram:



5. Most Challenging Part:

One of the most challenging aspects of the project was designing a user-friendly GUI using Tkinter while maintaining backend consistency. Ensuring that the GUI interacted smoothly with file operations and handled exceptions gracefully was also complex. Additionally, designing and maintaining the class hierarchy using abstract classes required a deep understanding of Python's OOP principles.

6. New Things Learnt in Python:

During the development of this project, several new concepts and libraries were explored:

- Tkinter for GUI development
- PIL (Pillow) for handling and displaying images
- CSV file handling for persistent data storage
- Abstract base classes using the abc module
- Custom exceptions for cleaner and more modular error handling

7. Individual Contributions:

Team Member	Contribution
Student1 (Hooria Ahmed CS-106)	Contribution involves designing the Login class, which encapsulates key functionalities such as user account verification, profile display, and individual car window management. This class retrieves car information from a CSV file, streamlining data access and enhancing overall system efficiency
Student2 (Fariha Nadeem Khan CS-107)	Contribution involves developing the Admin class, which includes essential features such as adding and removing cars, as well as viewing rental history in a tabular format by extracting data from a CSV file. This functionality enhances administrative capabilities and streamlines car management processes.
Student3 (Hibab Khan CS-108)	Contribution focuses on the core car rental system functionality, managing key aspects such as car availability, rental fee calculation, and payment processing. Upon rental, the system updates the rented car records and generates a detailed bank receipt with user information. This functionality enables efficient and organized car rental transactions.

In addition to our individual contributions, we collaborated on various aspects of the project. Specifically, Student 1 and 2 worked together on exception handling, while Student 1 and 3 focused on code commenting. Student 2 and 3 jointly prepared the project report. Student 1 created the class diagram. Overall, we shared responsibilities and worked together to complete the remaining tasks, ensuring a cohesive and well-rounded final product

8. Future Expansions:

Possible future improvements and extensions to the project include:

- Integrating a real database (e.g., SQLite or PostgreSQL) for better scalability and security
- Adding online payment support for rentals
- Implementing advanced search and filter functionalities
- Creating a responsive and modern GUI with frameworks like PyQt or Kivy
- Providing login session tracking and logout features
- Introducing user analytics for the admin dashboard

9. Test Run Cases:



The screenshot shows the "CREATE A NEW ACCOUNT" form. The form has the following fields and labels:

- Enter Your Name: Muhammad Ahmed
- Enter Your Address: House No. 123, Gulshan, Karachi
- Enter Your Bank Balance: 358000
- Enter Your Password (Min-6 characters): 123ahmed
- Enter Your Gender: Male (selected), Female
- Submit button
- View Profile button
- View Car Inventory button

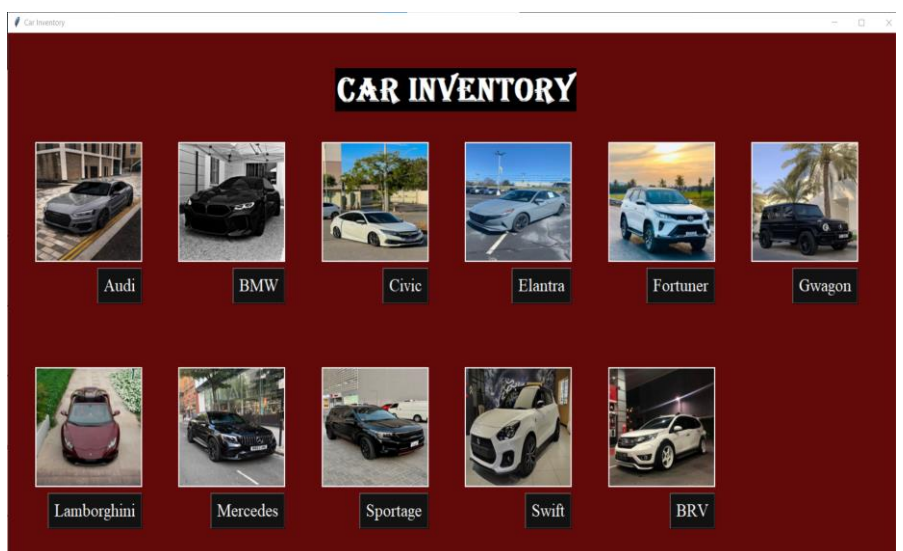
A green message at the bottom states: "Your account has been created successfully!"

The screenshot shows the "Remove Car" form. The form has the following fields and labels:

- Enter Car Name to Delete: (empty text input field)
- Delete button

The screenshot shows the user profile page. The page has a dark red background. At the top, there is a placeholder for a profile picture. Below the picture, the text "Profile Information" is displayed. The profile information includes:

- Name: Muhammad Ahmed
- Address: House No. 123, Gulshan, Karachi
- Gender: Male
- Balance: 358000.00 PKR
- No currently rented cars
- OK button



Lamborghini

LAMBORGHINI

Name: Lamborghini

Rental Price: Rs. 350000 per day

Model : Lamborghini Huracan Evo Spyder

Fuel Type: Petrol (Gasoline)

Engine Condition: 5.2-liter V10 engine delivering 640 hp

Colour: Rosso Mars

Navigation System (GPS): Equipped

Air Conditioning: Automatic dual-zone climate control



Safety Features: Overspeed warning, Anti-theft engine immobilizer

Rear-view Camera or Parking Sensors: Equipped

Mileage Limits: 100 to 150 miles

Transmission Type: automatic 7-speed dual-clutch


Seating Capacity: 2 persons

Rent This Car

Back

Profile



Profile Information

Name: Sara

Address: Lahore

Gender: Female

Balance: 345678.00 PKR

Current Rented Car: Mercedes

Car to be returned in 2 day(s)

Note: Make sure to return the rented car before renting another one

Return Car

OK

Current Rentals

CURRENT RENTALS

Name	Address	Car Name	Rental Days	Price per Day in Rs	Total Price in Rs
hibab khan	karachi	Audi	2	60000.0	120000.0
ali	gulshan-e-iqbal	Civic	1	13000.0	13000.0
sara	lahore	Swift	1	5000.0	5000.0
hooriya	liaquatabad	Fortuner	5	16000.0	80000.0
fariha	liaquatabad	Mercedes	2	50000.0	100000.0
ahmed	gulistan-e-johar	Sportage	2	16000.0	32000.0

Admin Login

ADMIN LOGIN

Enter The Password:

Admin@92018

Submit

Welcome Admin

Choose an option to proceed

Add Car

Remove Car

Cars Reserved

Current Rentals

Transaction Receipt

i

✓ Payment Confirmed

User Name: sara

Address: lahore

Car Name: Mercedes

Rental Days: 2

Price per Day: 50000.00 PKR

Total Price: 100000.00 PKR

Bank Details:

Bank ID: 3425671890

Transaction Code: 882349013

✓ Rental Approved. Enjoy your ride!

OK