National University of Computer & Emerging Sciences
Karachi Campus

OOP PROJECT PROPOSAL

PROJECT TITLE:

Car Rental Management System with User Authentication and Payment Integration

GROUP MEMBERS:

1.Mahnoor Hussain (Student ID: 23K-0535)

2. Urooj Baloch (Student ID: 23K-0071)

---

1. INTRODUCTION:

Background:

This project aims to create a Car Rental Management System, designed using C++ programming. The system will facilitate car rentals by allowing users to register, log in, select cars, input rental duration, and process payments (both cash and credit card). It will involve core programming concepts such as object-oriented design, input validation, and user authentication.

Problem Statement:

Car rental services often face challenges in providing a seamless, secure, and efficient user experience, particularly in terms of managing bookings, user authentication, and payment processing. This project will address these challenges by providing a straightforward car rental system with basic authentication and payment functionality.

Objectives:

- Develop a C++ application for car rental that includes user authentication and car selection.
- Integrate payment methods (cash and credit card).
- Ensure basic validation of inputs, such as email format and login credentials.
- Implement receipt generation and rental duration calculations.
- Demonstrate the use of object-oriented programming (OOP) to model car, user, and payment data.

---

2. SCOPE OF THE PROJECT

Inclusions:

- User Management: Registration and login functionality for users with email and password validation.

- Car Selection: Display a list of available cars (economy and luxury) and allow users to select a car based on their preferences.

- Payment Options: Users can choose between cash or credit card payment methods. Credit card details will be validated in a simple way.

- Receipt Generation: Generate a rental receipt that includes the car details, total price, and transaction confirmation.

- Basic Validation: Email format validation, password checking, and ensuring that rental hours are entered correctly.

Exclusions:

- Web Interface: This will not be a web-based application. The project will focus on a console-based solution.

- Complex Payment Integration: Real-time payment gateway integration is beyond the scope. We will simulate credit card payments without actual transaction processing.

- Database Management: The system will not store data permanently in a database. All data will be temporary during the execution of the program.

---

3. PROJECT DESCRIPTION:

Overview:

The Car Rental Management System is a console-based application developed in C++. It will allow users to register, log in securely, select from available cars (economy or luxury), specify the rental duration,

make a payment, and receive a receipt. The project will use objectoriented programming principles to create classes for users, payments, and vehicles.

Technical Requirements:

- Programming Language: C++
- Libraries: Standard C++ libraries such as `iostream`, `string`, `iomanip`, `ctime`, and `cstdlib` for input/output handling, user validation, and random receipt generation.
- Development Environment: Any IDE or compiler that supports C++ (e.g., Code::Blocks, Microsoft Visual Studio, or GCC).
- Platform: The system will be console-based and compatible with desktop operating systems.

Project Phases:

1. Research & Planning: Research car rental system requirements, define user needs, and determine project structure.

2.Design: Create the system architecture with OOP principles, designing classes for users, cars, and payments.

3.Implementation: Write the C++ code for each module (user management, car rental selection, payment handling).

4. Testing & Debugging: Test each feature, including user login, car selection, payment validation, and receipt generation.

5. Documentation: Compile a final report detailing the design, implementation, and system functionality.

---

4. METHODOLOGY

Approach:

We will work in a collaborative, iterative approach. Each team member will be responsible for specific modules of the project, with regular check-ins to ensure progress. We will begin by designing the classes and their relationships, then move on to implementation, testing, and refining the system.

Team Responsibilities:

- Member 1: Responsible for designing and implementing the `user` class (registration, login), `paymentdetails` class (payment handling), and managing the overall program flow.

- Member 2:Responsible for designing and implementing the `car` class (car selection, details display, receipt generation) and handling testing, debugging, and integrating user interactions with the car selection and payment methods.

---

5. EXPECTED OUTCOMES:

Deliverables:

- Working Application: A console application that allows users to rent cars, authenticate, select cars, input rental hours, make payments, and generate receipts.

- Final Report: A detailed report covering the project design, implementation, challenges, and conclusions.

- User Manual: A guide to help users interact with the system, detailing how to register, rent a car, and complete the payment process.

Relevance:

This project demonstrates core ICT concepts such as user authentication, basic payment processing, input validation, and objectoriented design. It will provide hands-on experience in building an interactive software system, which is a valuable skill in the field of software development.

---

6. RESOURCES NEEDED:

Software:

- C++ Compiler/IDE: Code::Blocks, Microsoft Visual Studio, or any C++ development environment. Other Resources:

- Online Tutorials: Tutorials for learning advanced C++ concepts like object-oriented programming and file handling (if needed).

- Instructor Support: Assistance from the instructor for troubleshooting and guidance during the implementation phase.
