Group Project Documentation   
Basic Cross-Platform Application Programming with .NET

*Car Booking Management System*

**Prepared by Group 2**

***Nguyễn Tiến Dũng, Nguyễn Đức Sơn, Lê Quốc Uy, Vũ Long***

**Ho Chi Minh City, 2023**

**Table of Contents**

**Table of Contents ii**

**Revision History ii**

**1.** **Project Introduction 1**

1.1 Product Perspective 1

1.2 User Classes and Characteristics 1

**2.** **Database Design 1**

**3.** **System Architecture 1**

**4.** **Implementation 2**

4.1. Deployment Considerations [2](#_2s8eyo1)

4.2. Screenshots and explanations [2](#_17dp8vu)

**5.** **References 2**

**Revision History**

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
| First writing | 19 August 2023 |  | 1.0 draft 1 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Project Introduction

## Product Perspective

The objective of the Car Booking Management Project is to optimize the procedure of reserving and overseeing car rentals with efficiency. The system encompasses four distinct roles: Guest, Member, Staff, and Admin, all contributing to a fluid and user-friendly interface.

For guests, the platform provides the chance to explore accessible vehicles and their particulars without necessitating registration. Should they decide to move forward with a booking, guests have the option to register as members, thereby unlocking added privileges and perks.

The Member role affords registered users an upgraded experience. Members can seamlessly peruse the range of available cars, view their specifications, and conveniently make reservations. Moreover, members enjoy the benefit of accessing their booking history, simplifying future reservation processes. The shift from guest to member is facilitated through successful registration, granting users exclusive entry to membership advantages.

The Staff role is tailored to assist both guests and members. Staff members are entrusted with the task of managing and updating the car inventory, ensuring accuracy in vehicle particulars, availability, and pricing information. Additionally, they are on hand to address user inquiries, concerns, or alterations to bookings, contributing to elevated customer satisfaction levels.

The Admin role commands the highest level of authority and responsibility. Admins oversee the overall functionality of the system, oversee staff accounts, and possess the authorization to review and authorize membership registrations. They are also equipped with the tools required to manage system-wide settings, thereby ensuring the system's smooth and secure operation.

The Car Booking Management Project furnishes an inclusive solution that accommodates a diverse user spectrum, spanning from casual guests to devoted members, and from diligent staff to commanding administrators. Through meticulous administration and execution of each role's duties, the project's aim is to streamline the car rental process, elevate user experiences, and establish an effective and user-centric car booking environment.

## User Classes and Characteristics

The Car Booking Management System consists of four main user classes: Guest, Member, Staff, Admin.

**Guest**

* Guests are individuals seeking to rent cars. They use the system to search available vehicles, make reservations based on their start and end date.
* Guests can find the cars with the price they provided. They can view the information about the car, and able to add it into the cart

**Member**

* To become a member of the system, the guest must register an account into the system
* Members can checkout the booking, and they can see the purchase history of a booking that they had made.

**Staff**

* The system will have different staff members to manage the cars in the system
* They can view a list of cars rented by members
* They can track the availability of cars based on the bookings

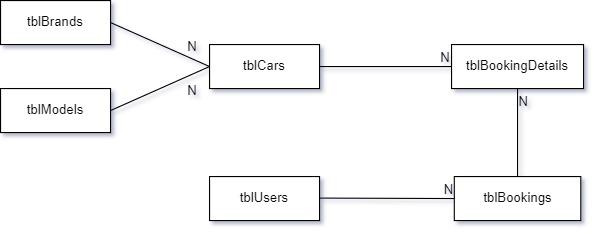
**Admin**: A system will have one admin to manage all the staff and members in the system.

In the system, there are some specific rules for members to make a car reservation

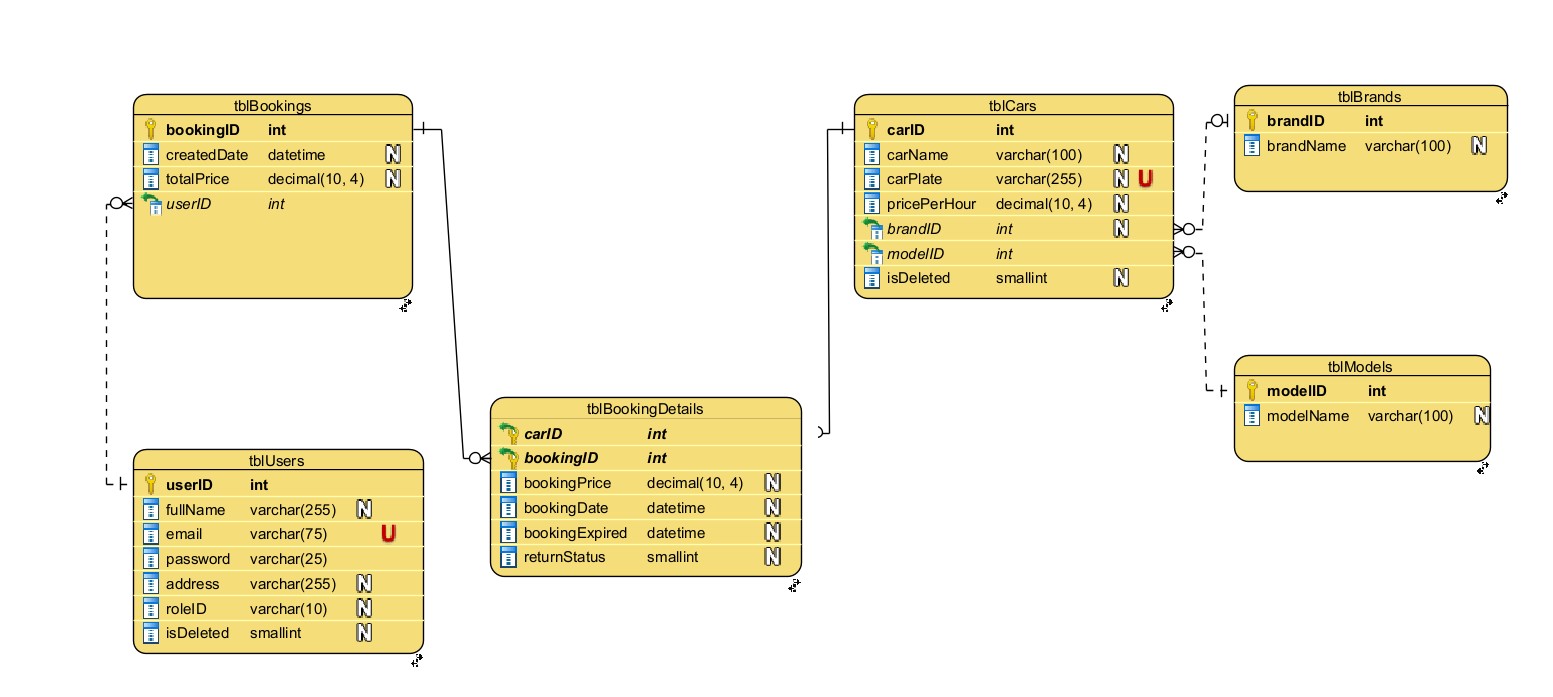
* Members can book various cars, but at the same time, they can only book a specific car.
* A booking can have many cars, but the booking and expired date will need to be different. This allows another member can book a car
* If a member has made a reservation, they can still change the booking and expiration date before checkout the booking.

# Database Design

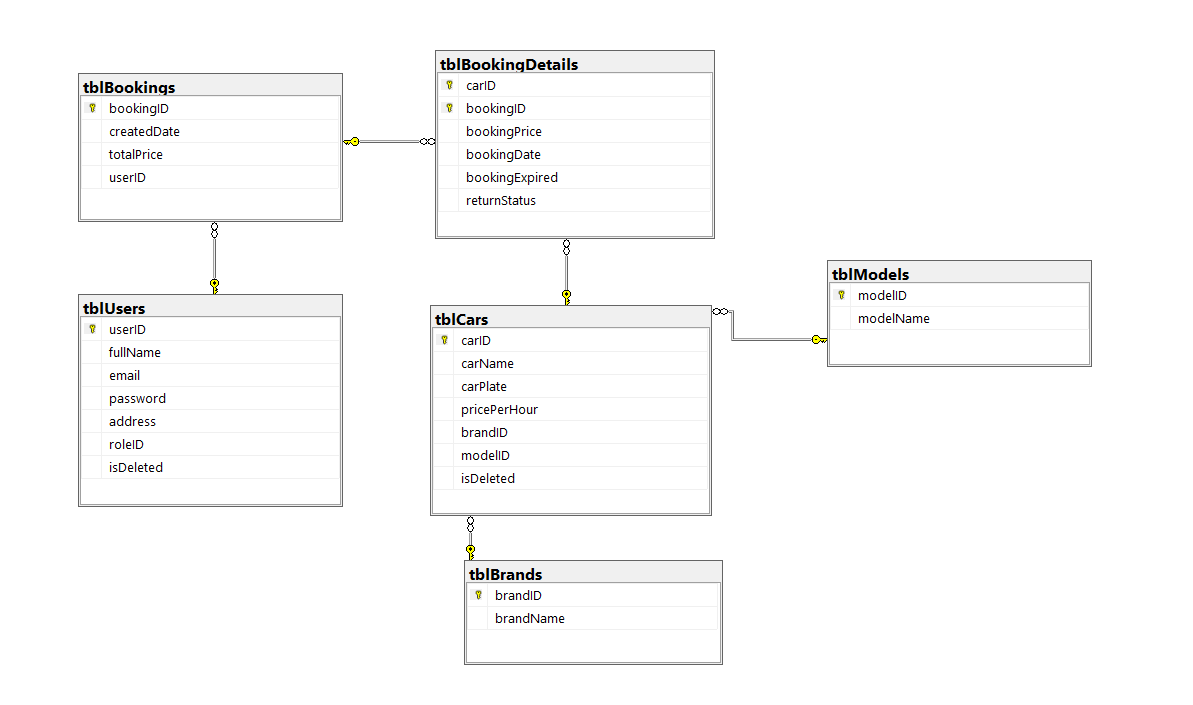
**Conceptual Diagram**

****

**Logical Diagram**

****

**Database Diagram**

****

**Notes**

* The returnStatus has three states:
  + 0: the car is not available for renting
  + 1: the car is renting
  + 2: the car is returned
* The isDeleted has two states:
  + 0: no deletion
  + 1: deletion

# System Architecture

The car booking management system follows a 3-layer architecture that ensures efficient data flow, separation of entities and business logic, and is easy to maintain. The technologies are used in this project including .NET 5, Entity Framework Core (version 5), and Microsoft SQL Server. The architecture delivers a robust and user-friendly interface to different actors, comprising guests, members, staff and admins.

* **Presentation Layer**: It is implemented through Winforms, is the user interface where users interact with the system. It will provide an intuitive and user-friendly environment for managing bookings, car reservations, and accessing various system features.
* **Business Logic Layer**: Serving as the core of the application, the business logic layer is responsible for implementing business rules, orchestrating data processing, and managing system behavior. It receives user requests from the presentation layer and determines the appropriate actions to take.
* **Data Access Layer** (Repository and Data Access Objects): It is responsible for data retrieval and storage. It is divided into 2 sub-layers: the Repository and Data Access Object layer.
  + *Repository*: This layer provides the interface for interacting with the data. It abstracts the underlying data storage and exposes methods to access and manipulate entities. It ensures a consistent and standardized way to perform CRUD (Create, Read, Update, Delete) operations on data entities.
  + *Data Access Object*: The DAO layer encapsulates the lower-level data access operations. It communicates directly with the database through technologies like Entity Framework Core 5 and MS SQL Server. It's responsible for executing database queries, managing data integrity, and ensuring efficient data retrieval and storage.

# Implementation

## Deployment Considerations (Framework, version….)

Efficiently deploying the Car Booking Management Project entails a strategic approach to transition from development to operational implementation. The process involves a series of tasks geared toward ensuring the application's stability, performance, and accessibility.

**User Access and Considerations:**

For users, including guests, members, staff, and administrators, seamless access to the Car Booking System is essential. User distribution across different time zones necessitates careful planning to provide uninterrupted service. Infrastructure adjustments may be required to optimize network accessibility and accommodate anticipated user demand. Minimizing latency is pivotal for a smooth user experience.

**Infrastructure Adaptations and Capacity Strategy:**

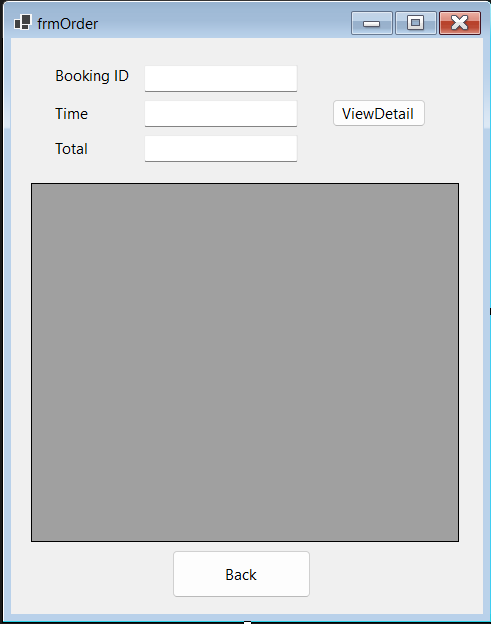
Deployment may entail necessary infrastructure modifications to meet capacity needs. Rigorous evaluation of network infrastructure ensures data transfer efficiency and reliability. Adequate data storage planning is essential to accommodate future data growth, encompassing reservations and booking history. Data migration strategies should be in place for seamless data transition. Should the application's load necessitate additional servers or cloud resources, deployment should encompass these adjustments.

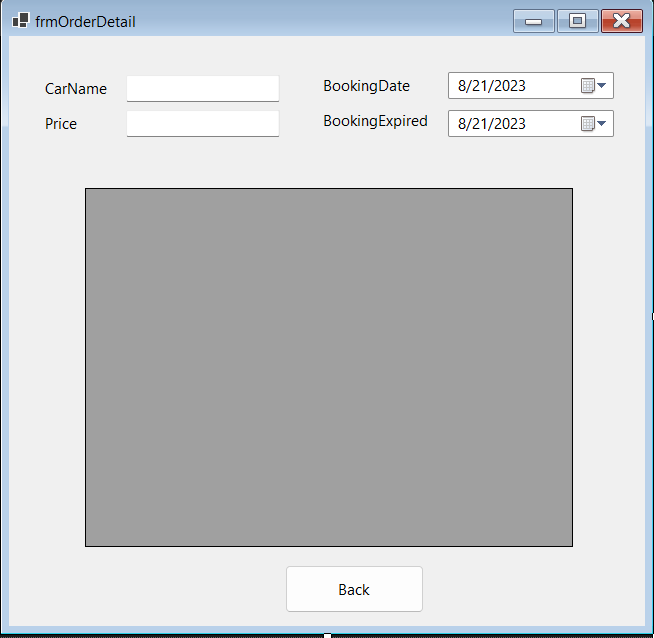
**Training and Business Process Enhancement:**

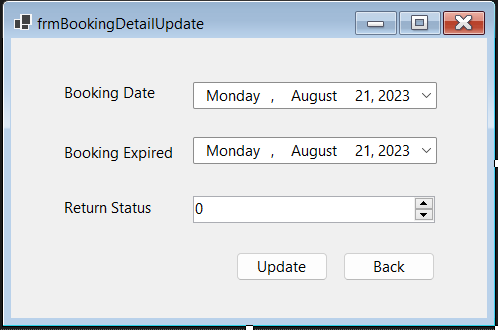
Successful deployment hinges on comprehensive training resources for both users and staff. Clear guidelines detailing Car Booking System usage, reservation processes, and member privileges are paramount. Effective communication about potential business process changes, particularly those related to car rental and management, is crucial. Staff responsibilities might require refinement to align with the new system's capabilities.

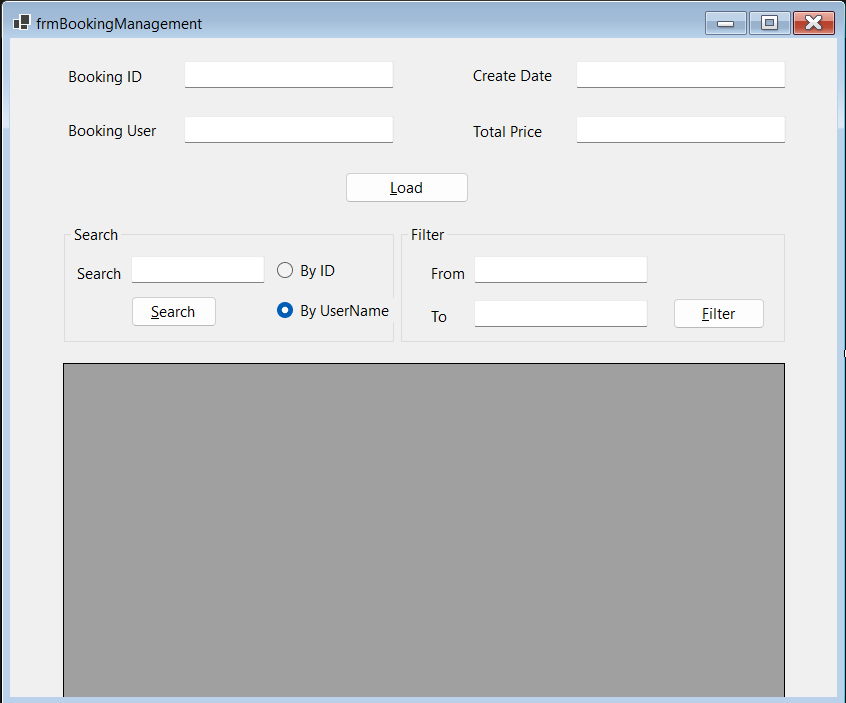
In conclusion, deploying the Car Booking Management Project necessitates meticulous planning and execution for a seamless shift to operational use. User access must cater to distribution and time zone variations. Infrastructure adaptations should optimize network performance and data storage. Thorough training materials and business process integration will facilitate user adoption and align operational workflows with the new system's features. Addressing these aspects will ensure a successful deployment that elevates car booking and management processes.

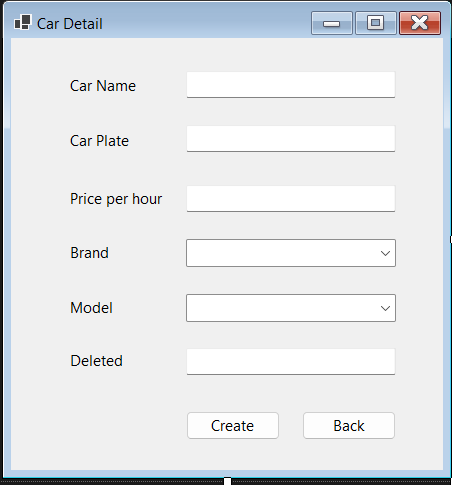
## Screenshots and explanations

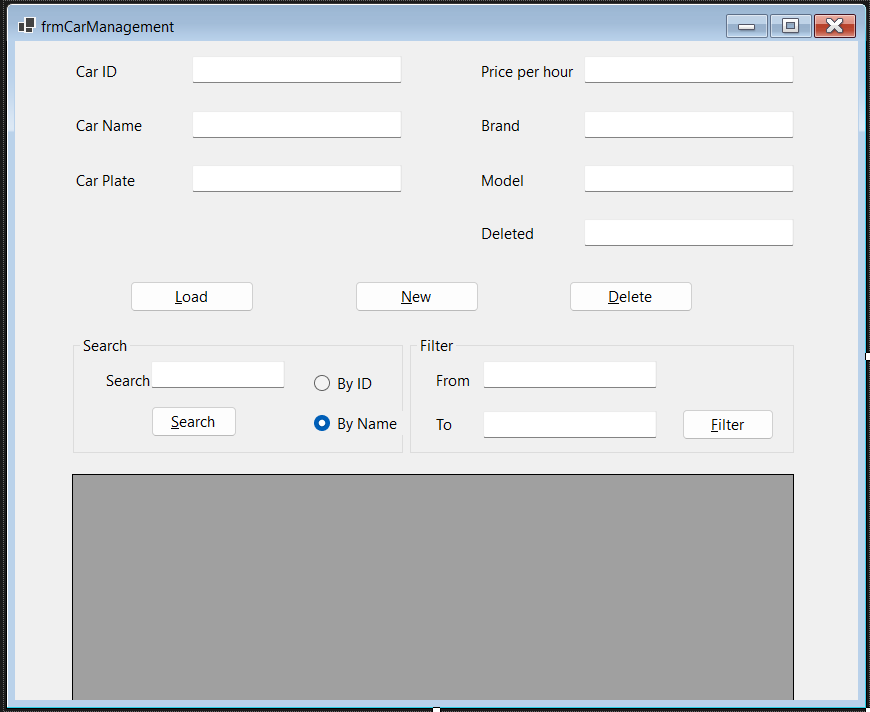


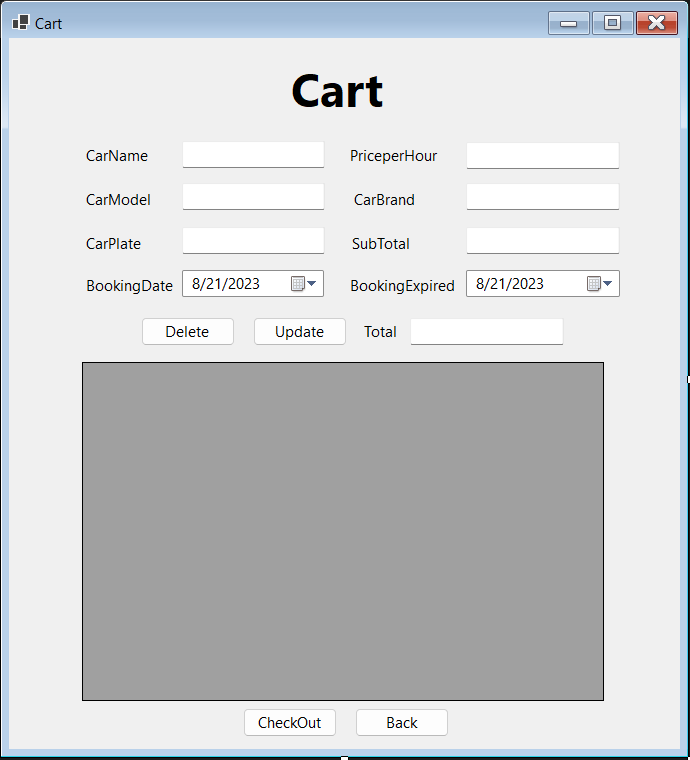


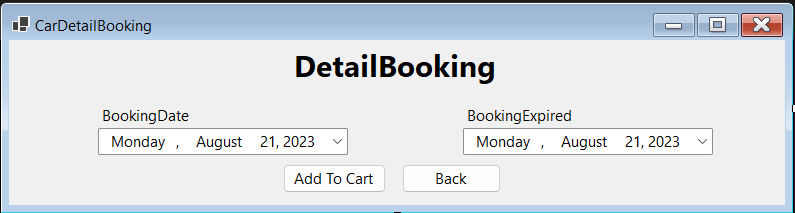


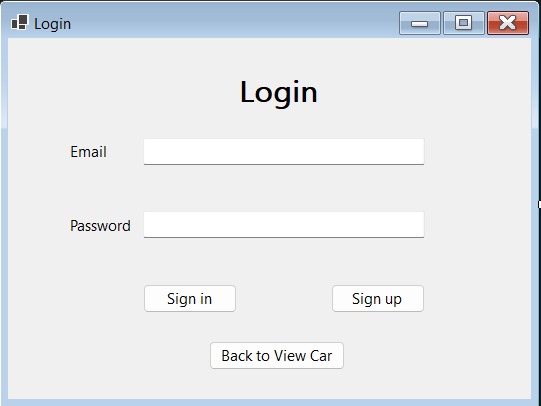


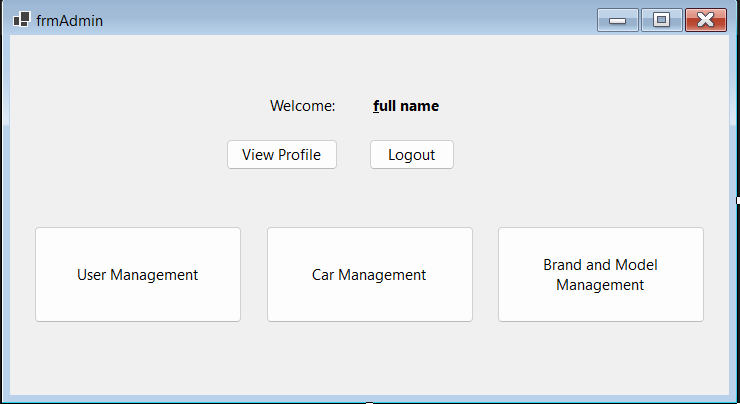


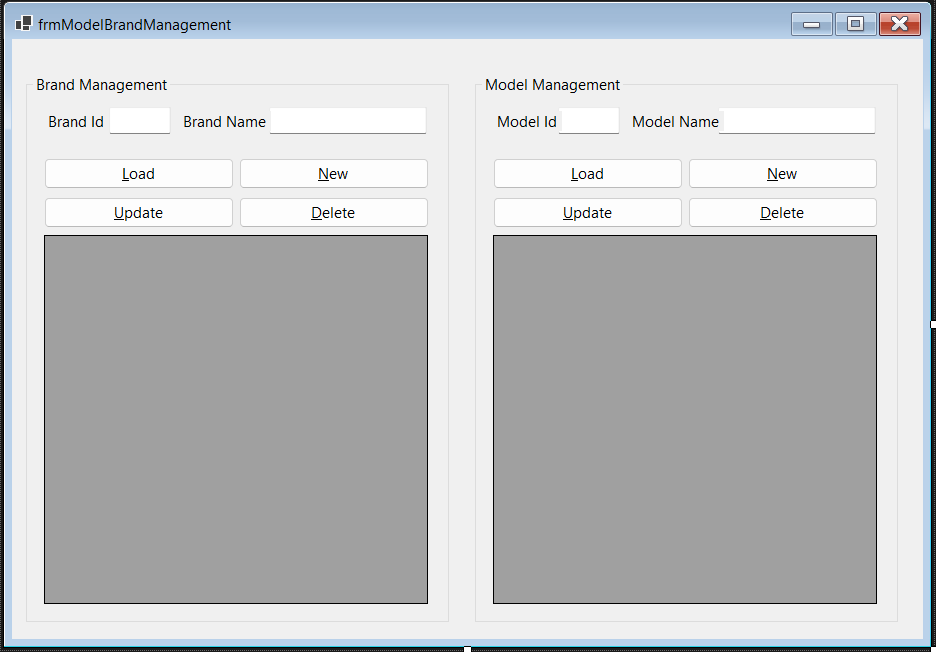


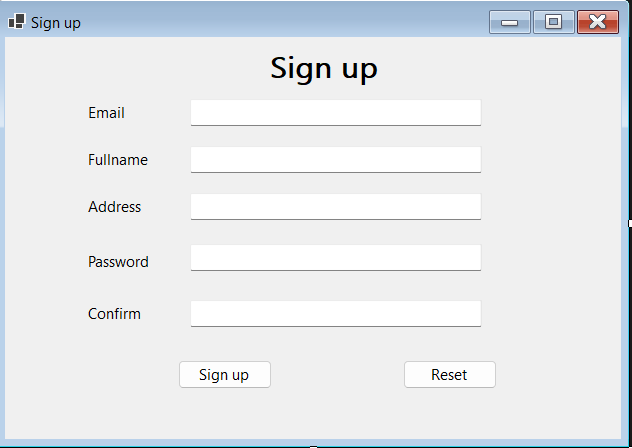


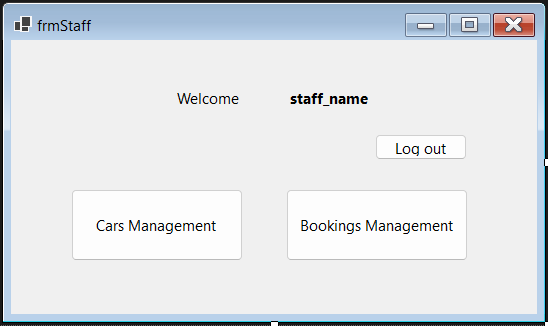


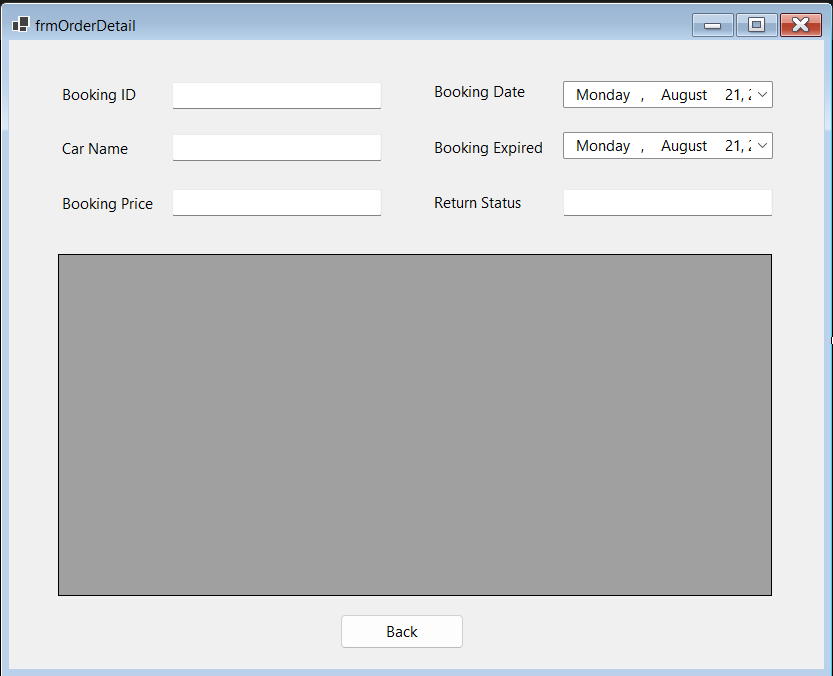


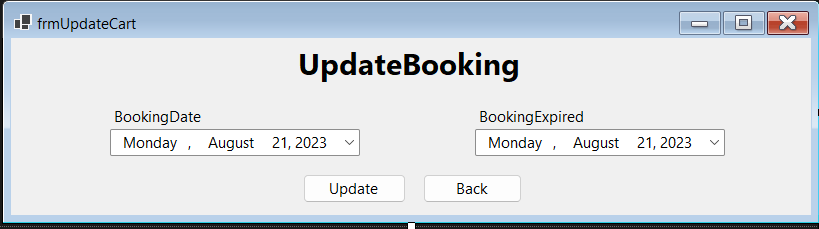


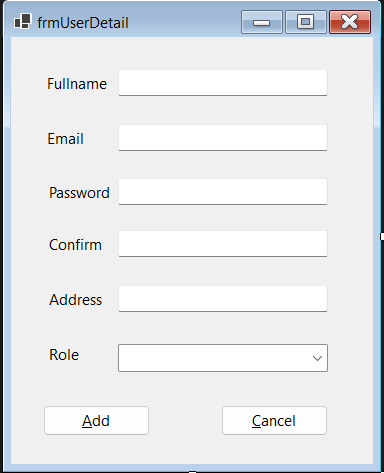


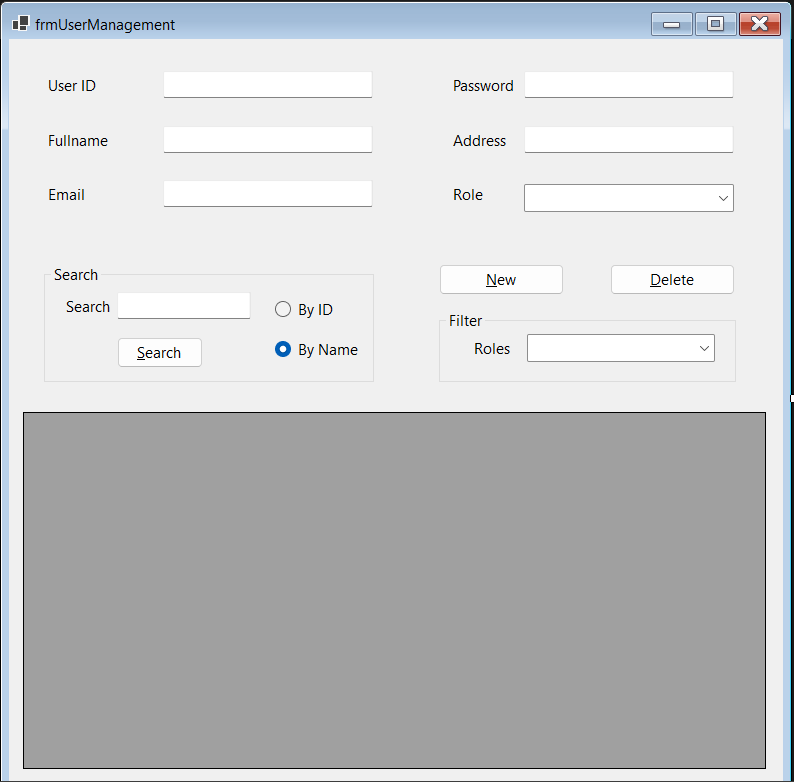


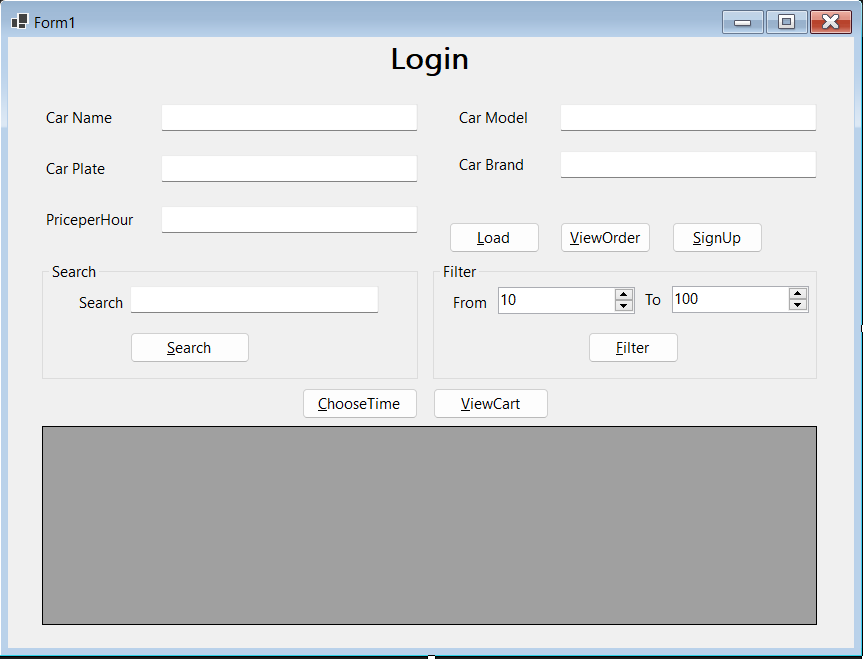












# References

NA