

## Assignment 01 - Summer 2023 HCM

# Building ASP.NET Web API with Desktop Application/Web Application

### 1. Introduction

A flower bouquet shop is a retail store that specializes in selling flower arrangements. These shops offer a wide variety of flower arrangements and bouquets for different occasions such as birthdays, weddings, anniversaries, and funerals. Customers can choose from a variety of flowers such as roses, lilies, carnations, chrysanthemums, and more. The bouquets can be arranged based on the customer's preferences, including color, flower type, and bouquet size. Some flower bouquet shops also offer delivery services, allowing customers to send flower arrangements to their loved ones. These shops are popular for their ability to provide quick and convenient access to flowers for any occasion. They provide a great option for those looking to make a lasting impression or show appreciation to someone special.

Imagine you're a developer of a FU Flower Bouquet Management System named **FUFlowerSystem**. To implement a part of this system your tasks include:

- Manage customer information.
- Manage flower bouquet information.
- Manage order information.

The application has a default account (admin account) whose email is “**admin@FUflowerbouquet.com**” and password is “**admin@@**” that stored in the **appsettings.json**.

The application has to support adding, viewing, modifying, and removing information - a standardized usage action verb better known as Create, Read, Update, Delete (CRUD) and Search.

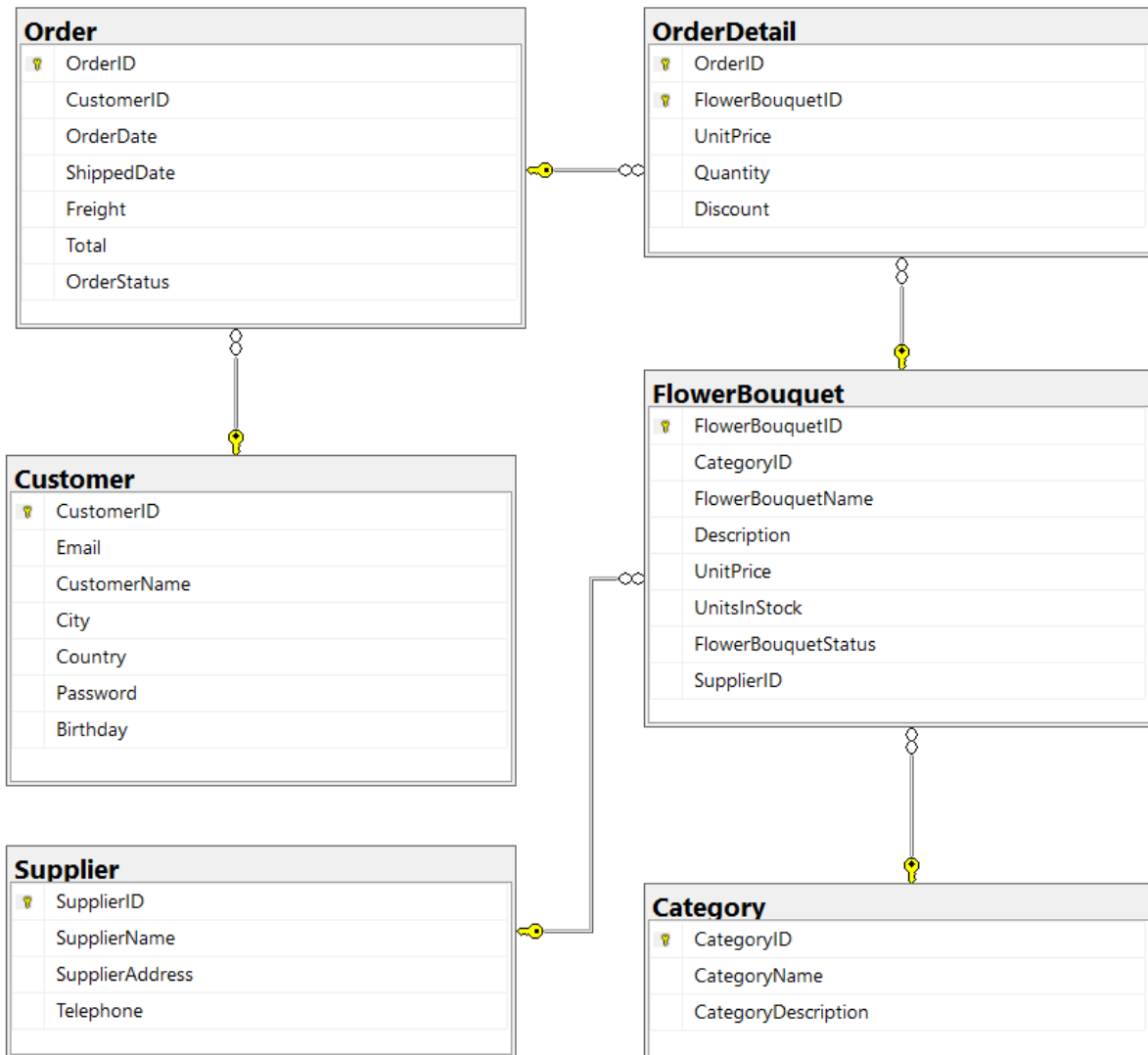
This assignment explores creating an ASP.NET Core Web API with C#, and ADO.NET or Entity Framework Core, the client application can be used as Desktop Application (Windows Forms, WPF) or Web Application (ASP.NET Core Web MVC or Razor Pages). An MS SQL Server database will be created to persist the data and it will be used for reading and managing data.

## 2. Assignment Objectives

In this assignment, you will:

- Use the Visual Studio.NET to create a Desktop/Web application and ASP.NET Core Web API project.
- Perform CRUD actions using ADO.NET or Entity Framework Core.
- Use LINQ to query and sort data.
- Apply 3-Layer architecture to develop the application.
- Apply Repository pattern and Singleton pattern in a project.
- Add CRUD and searching actions to the Desktop/Web application with ASP.NET Core Web API.
- Apply to validate data type for all fields.
- Run the project and test the actions of the Web application.

### 3. Database Design



A flower bouquet will belong to only one category and only one supplier.

A customer can order many orders. An order will have one or many flower bouquets' information.

A flower bouquet's information will belong to many orders

## 4. Main Functions

- Create Web API with Customer management, Flower Bouquet management, and Order management: Read, Create, Update and Delete actions.
- Create Client application (with Desktop/Web application) interactive with Web API to perform these functions:
  - If the user is an “**Admin**” then his/her is allowed to
    - Manage customer information.
    - Manage flower bouquet information. (The delete action will delete flower bouquet information in the case this information is not belong to any order. If the flower bouquet information is already stored in an order, just change the status.)
    - Manage order information (includes order details management)
  - Create an order report statistic by the period from StartDate to EndDate, and sort sales in descending order.
  - If user is a “**Customer**”, this customer role is allowed to:
    - Register an account.
    - Create an online order with one or many flower bouquets’ information.
    - Manage his/her the profile
    - View order history.

## 5. Notes

- You must use **Visual Studio 2019 or above (.NET5/.NET6/.NET7)**, **MSSQL Server 2012 or above** for your development tools.
- To do your BE program, you must use **ASP.NET Core Web API**. *Note that you are not allow to connect direct to database from ASP.NET Core Web API, every database connection must be used through Repository and Data Access Objects. The database connection string must get from appsettings.json file.*
- Create Solution in Visual Studio named **StudentName\_ClassCode\_A01.sln**.

- Create your MS SQL database named **FUFlowerBouquetManagement** by running code in script **FUFlowerBouquetManagement.sql**.
- Set the default user interface for your project as **Login** page/window.