**Assignment 02**

**Building a Shopping Website with ASP.NET Core Razor Pages**

# Introduction

Imagine you are an employee of a product retailer named **PizzaStore**. Your manager has asked you to develop a web application for member management, product management, and order management.

The application has to support adding, viewing, modifying, and removing products—standardized usage action verbs better known as Create, Read, Update, Delete (CRUD) and Search. This assignment explores creating an application using Razor Pages .NET Core Web Application, C#, and Entity Framework Core. The MS SQL Server database will be created to persist the data and it will be used for reading and managing data.

# Assignment Objectives

In this assignment, you will:

* Use the Visual Studio.NET to create ASP.NET Core Web Application project.
* Perform CRUD actions using Entity Framework Core.
* Use LINQ to query and sort data.
* Apply to validate data type for all fields.
* Run the project and test the ASP.NET Core Web Application actions.

# Database Design

# 

# Main Functions

* Create database using Entity Framework Core.
* Member management, Product management, and Order management: Read, Create, Update and Delete actions. Creating and Updating actions must be performed by popup dialog
* Search Product by ID , ProductName (by keyword of ProductName), UnitPrice.
* Create a report statistics sales by the period from StartDate to EndDate, and sort sales in descending order
* Member registration by AccountID (unique) and Password. If the user is “**Staff**” (Type=1) then allows to perform all actions, otherwise, the normal (Type=2) user is allowed to view/create/update the profile and view their orders history.

# Guidelines

# Activity 01: Build a solution [01 mark]

**Step 01**. Create Project/Solution named SignalRAssignment

**Step 02**. Choose the Project template for creating an ASP.NET Core application with example Razor Pages content: ASP.NET Core Web App.

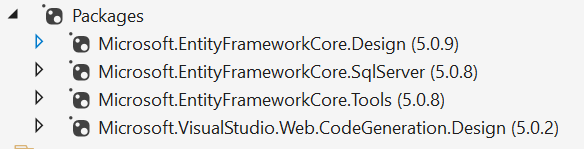
# Activity 02: Create database with Entity Framework [03 marks]

**Step 01**. Create Model

With AppUser, please note that the application will have two types: Member (type = 0) and Staff (type = 1).

**Step 02**. Create database context with Entity Framework ApplicationDBContext

**Step 03**. Add-Migration and then Update Database



# Activity 03: CRUD product information using Razor Pages [05 marks]

**Step 01**. View list of Pizza products and view the detail pizza product information (all users have permission to view products).

**Step 02**. Create registration form; check the required validation for all fields. The registration will have member permission (type = 0).

**Step 03**. Only staff member can create a new pizza product.

**Step 04**. Only staff member can update an existing pizza product.

**Step 05**. Only staff member can delete an existing pizza product.

# Activity 05: Run the project and test all actions [01 mark]

# 

# 

# 