Lab 4 – WPF

Due Date: Midnight of Week11's Friday
Purpose: The purpose of this assignment is to help you become familiar with
- WPF data binding, WPF UserControl, MVVM pattern
- Entity Framework Core

Instructions: Be sure to read the following general instructions carefully:
This assignment must be completed individually. Submit your solution **through the Luminate**. You must name your submission according to the following rule: **studentID(yourlastname)_Labnumber.zip**. e.g., 300123456(**smith)_Lab#4**.zip

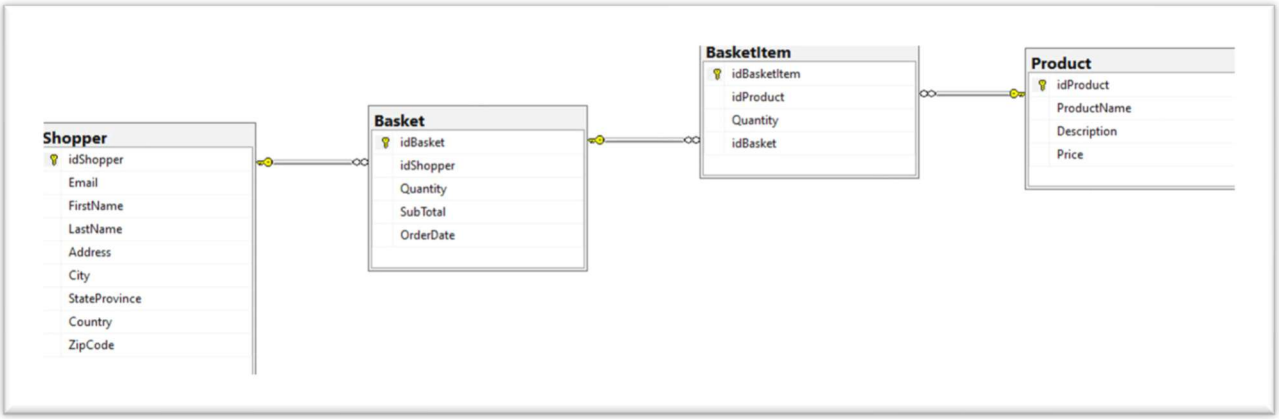
Rubric

Functionality	**Marks**
Class library project 1. Use EF Core to create data model classes (2 marks)	2
Implement a utility class to access data from database, please make sure to use asynchronous programming whenever it is suitable 1. Get data for comboBox that displays the shopper's email & IdBasket (2 marks) 2. Get data for the DataGrid (2 marks) 3. Get data for the comboBox that displays the product's information (2 marks) 4. Save data to database (2 marks)	8
MainWindow & MainWindowViewModel class 1. MainWindow.xaml (1 mark) 2. MainWindowViewMode.cs (1 mark)	2
ListOrderDetailsView.xaml & ListOrderDetailsViewModel class 1. ListOrderDetailsView.xaml (4 marks) 2. ListOrderDetailsViewModel.cs (4 mark)	8
AddNewItemView & AddNewItemViewModel 1. Finish AddNewItemView.xaml (4 marks) 2. AddNewItemViewModel.cs (4 marks)	8
Overall (readability, efficiency, maintainability)	2

Exercise 1 [30 marks]

You are asked to **implement a WPF Application using MVVM pattern**. The app is to facilitate order administrators to manage all orders.

Run ***OMS.sql*** to generate the Database, and below is the database ER diagram.



Your application needs to have following functionalities

1. After the application launches, your application's UI looks similar as below

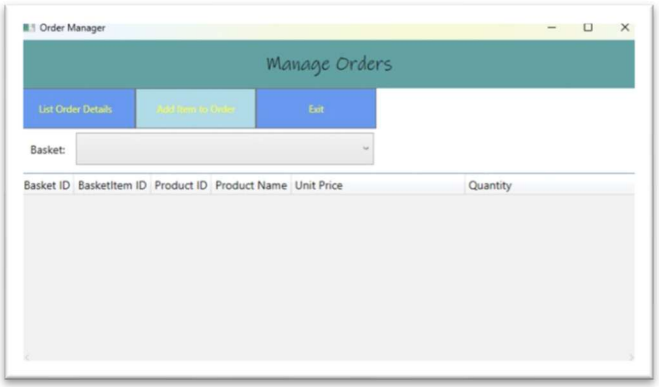


Figure 1

2. All Baskets (Shopper's email and Basket Id) need to be listed in the comboBox. After user selects one from the ComboBox, the basketItem details should be listed in the DataGrid. Please refer to Figure 2 & Figure 3

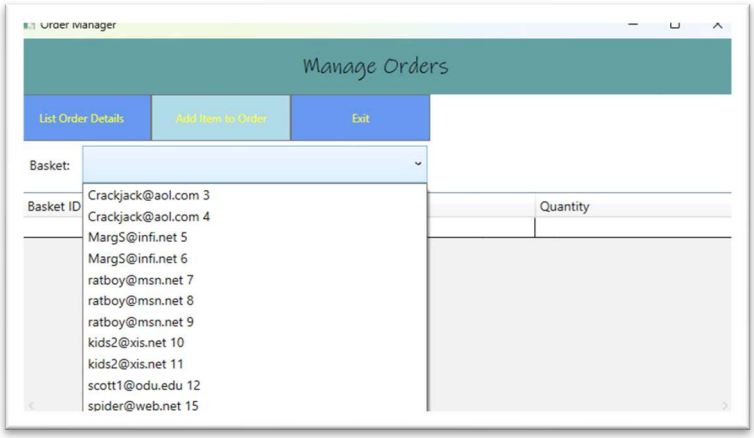


Figure 2

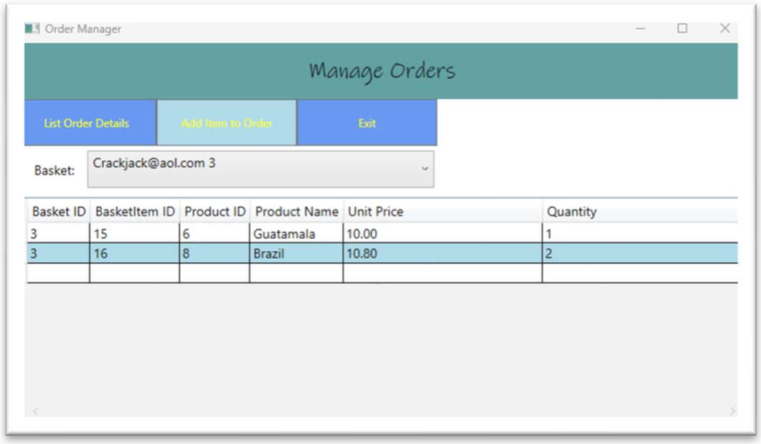


Figure 3

3. When user click on "Add Item to Order", the UI looks similar as Figure 4. Please note that all baskets are listed in the first comboBox(shown in figure 5), and all products are listed in the second comboBox (shown in Figure 6).

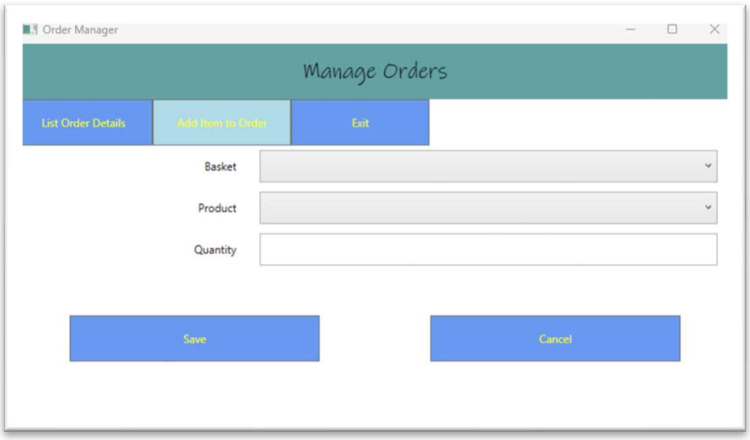


Figure 4

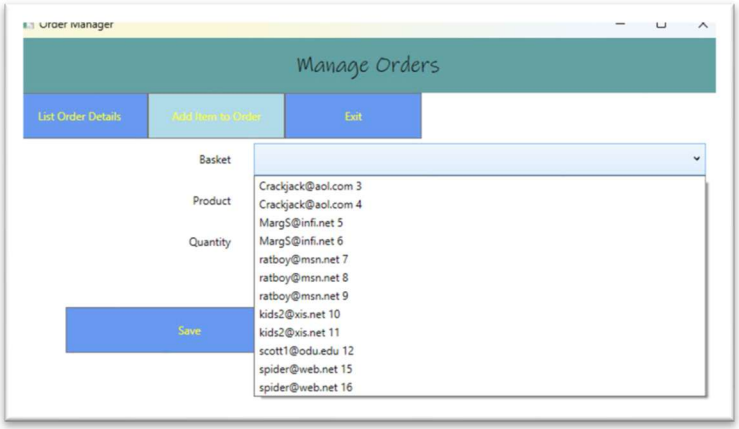


Figure 5

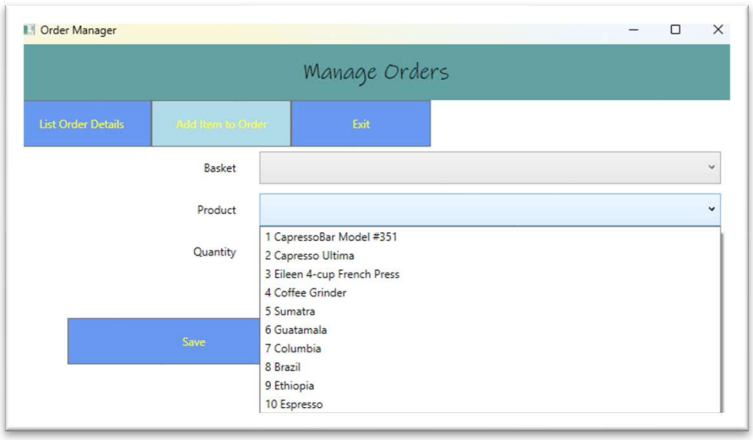


Figure 6

4. After user provides required information and clicks "Save" button, the newly added item should be added to the underlying database. Please bear in mind that the **IdBasketItem** for the new item should be current maximum ***IdBasketItem*** plus one.

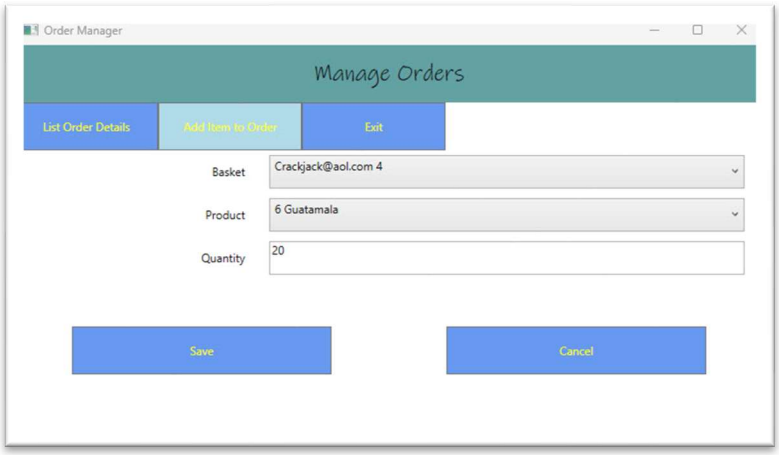


Figure 7

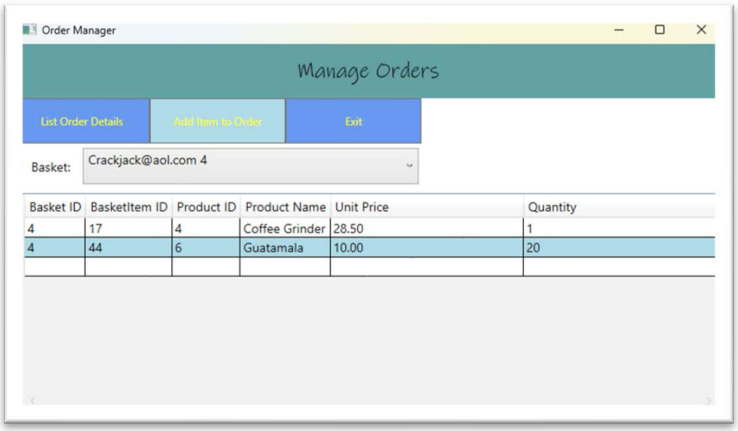


Figure 8