2.3.0 – Install Entity Framework

Introduction

The basic solution has been created and customized, but the web site is a static web site; has no connection to a database and has no web forms.

Configure Visual Studio Solution

The next phase is to configure the solution for an n-tier web project. This includes creating the backend projects to hold the *data objects* and the business logic. It is important to understand the interaction of the layers of the system. The generic Sequence Diagram is shown in the figure below:
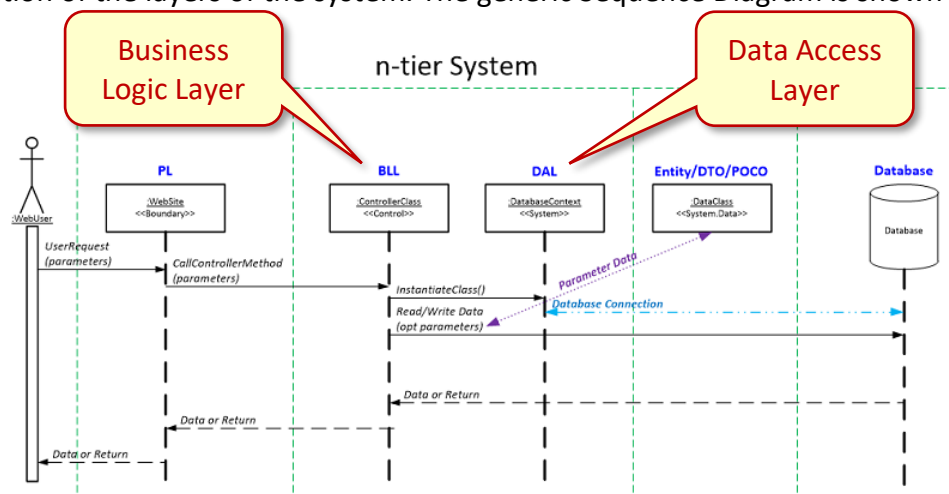

Figure 1: n-Tier System

To code this we create the Web Site project, and the System Project as shown in the figure below:
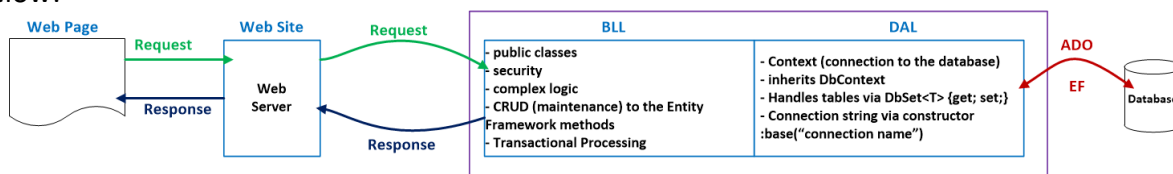

Figure 2: System Diagram

eStoreData Project

The eStoreData project was created to hold classes that will represent data from the eStore_2018 database. We need to configure the eStoreData project to be able to represent data objects (ADO = **A**ctiveX **D**ata **O**bjects). This is done by adding references to the eStoreData project.
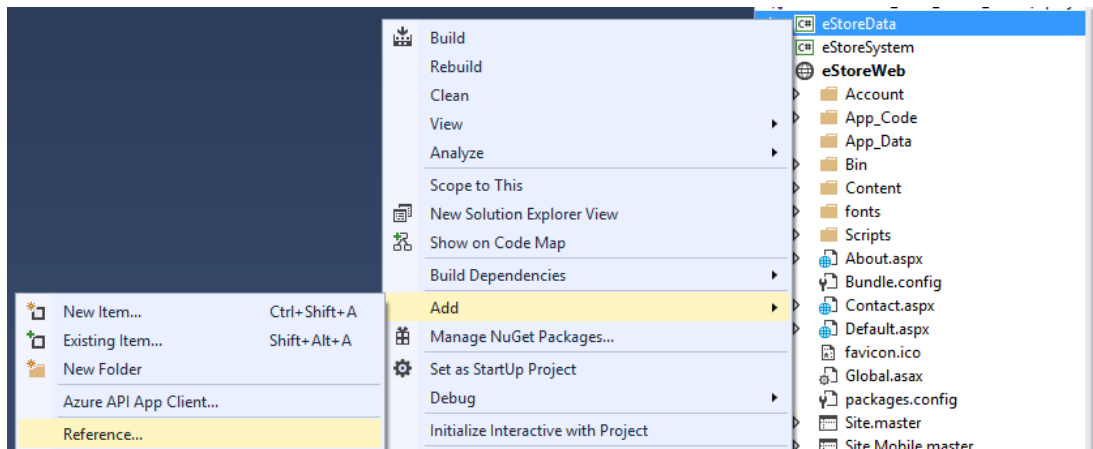
Figure 3: Add Project Reference

The references that need to be added are in the Assemblies section and are for
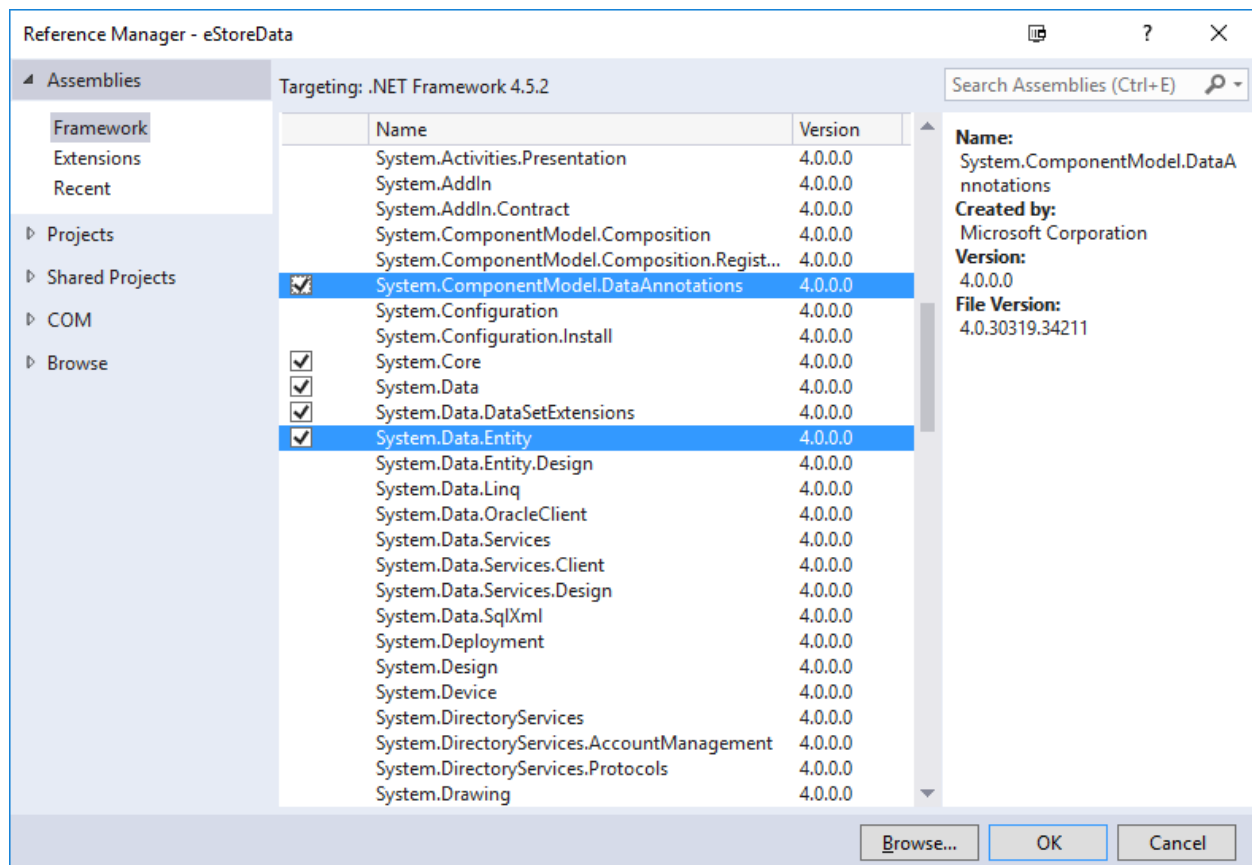**System.ComponentModel.DataAnnotations** and **System.Data.Entity**:

Figure 4: eStoreData References to Add

Once added, you can expand the **eStoreData** project in the Solution Explorer to see the
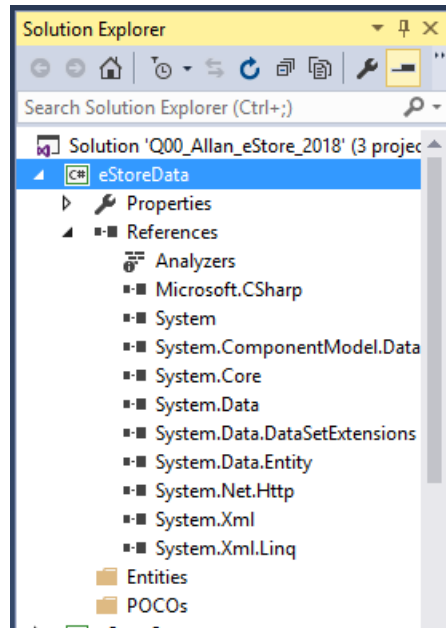following:

Figure 5: Assembly References Added to eStoreData Project

eStoreSystem Project

The eStoreSystem project was created to do two things: (1) Access the database (DAL folder), and (2) Control business processes and system messages (BLL folder). This project needs to know about the eStoreData project as it will have the definitions of all the data (**DO NOT ADD THE WEB PROJECT!**). The first step will be to add a reference to the eStoreData project, and a reference to the System.Data.Entity so it can understand the data. (Both references can be added using the Add Reference wizard):
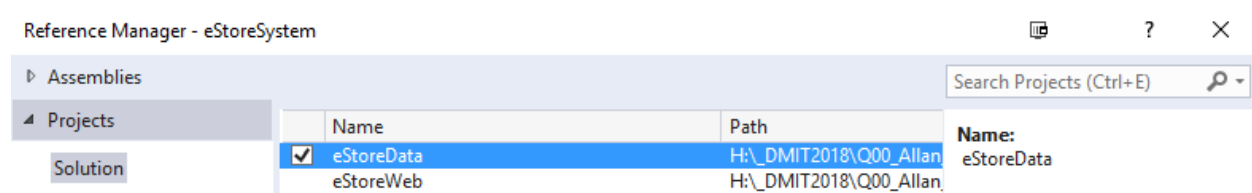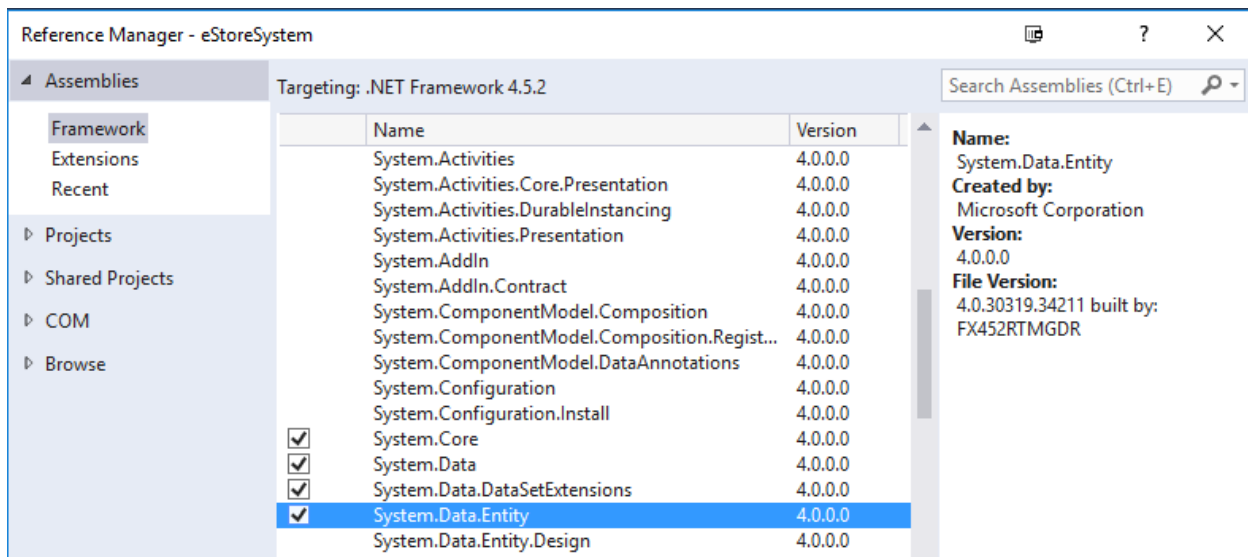

Figure 6: Add Reference to eStoreData Project

Figure 7: Adding Reference to System.Data.Entity

Once added, the **eStoreSystem** project in Solution Explorer should look like:
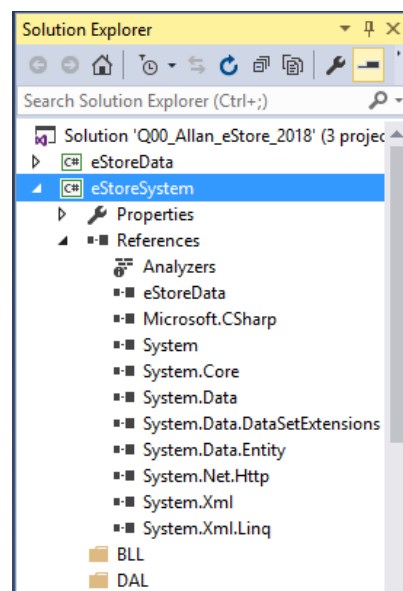

Figure 8: References Added to eStoreSystem

Entity Framework

Entity Framework (EF) is an object-relational mapper that enables .NET developers to work with relational data using domain-specific objects. It eliminates the need for most of the data access code that developers usually need to write.

In Figure 2, we saw that the DAL class was accessing the Database using EF (EF = **E**ntity **F**ramework). Adding EF is not the same as adding a reference to a project, it requires an external framework to be added to Visual Studio.

To add the Entity Framework to the eStoreSystem project, use the **NuGet Package Manager**. Right-click on the System project –> **Manage NuGet Packages…**. Select Browse and search for **Entity Framework 6.x.x** – select the newest version).
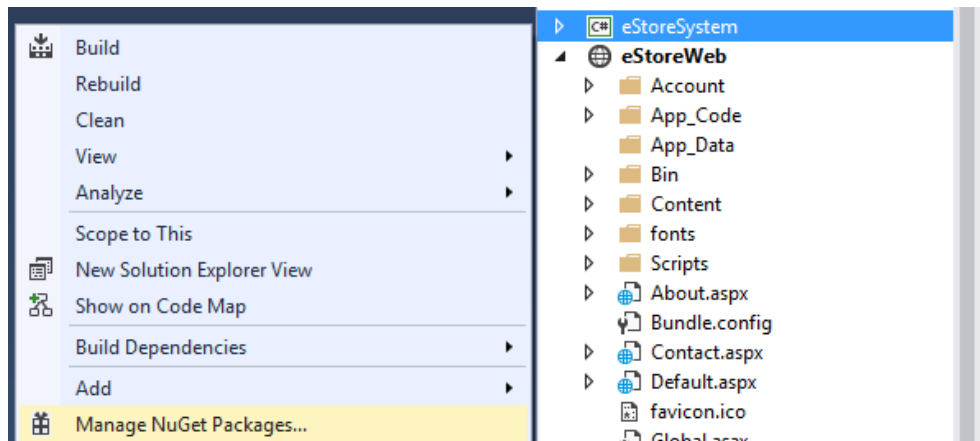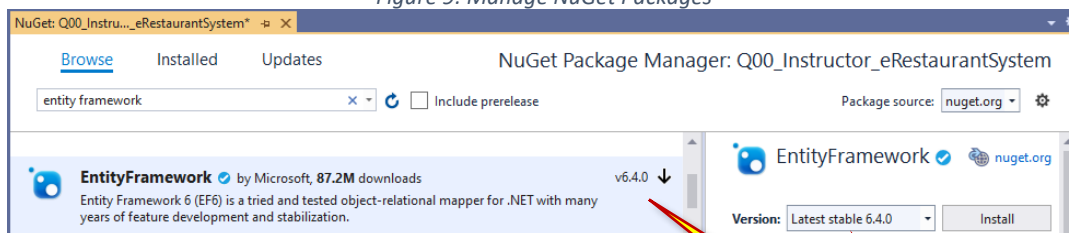

Figure 9: Manage NuGet Packages
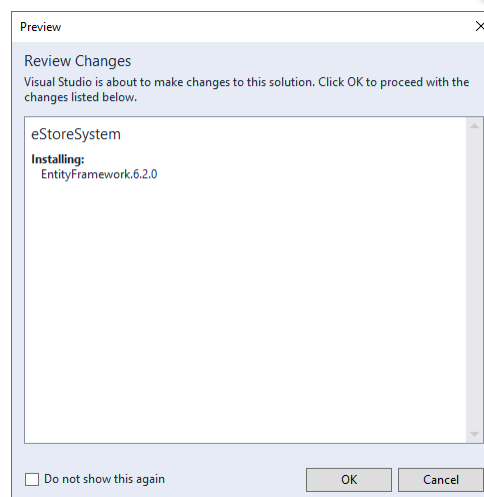

Figure 10: Select Latest Entity Framework

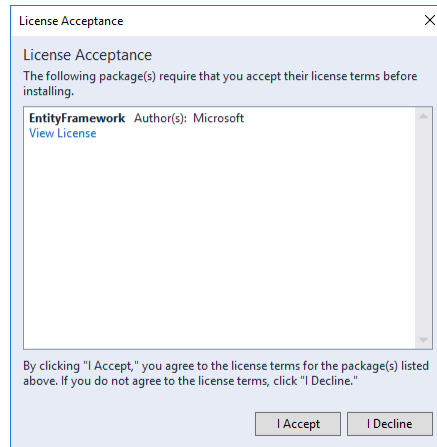
Figure 11: Review Changes

Figure 12: Accept License

Visual Studio will download and install the Entity Framework to your project. The Output window will show text like:
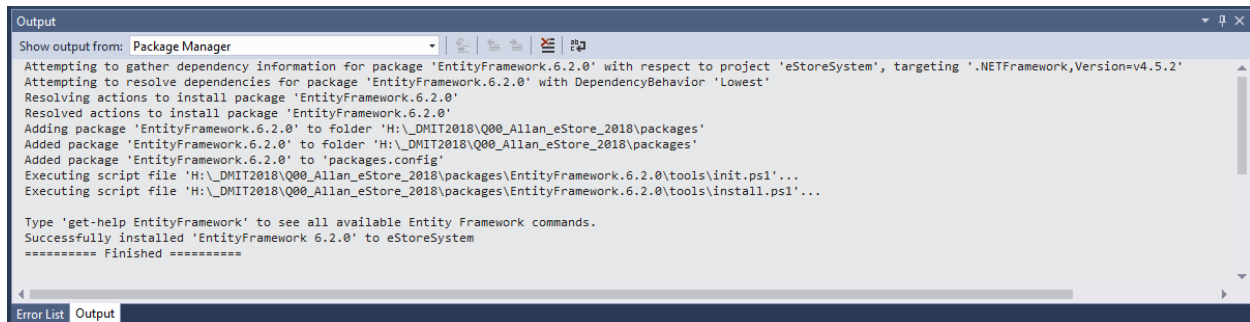

Figure 13: Entity Framework Install Output

Note: After adding the Entity Framework, the System.Data.Entity reference may disappear from you project's reference list. Even though it is not listed, it is available for any **using** statements.

Build Solution!

Before going any further with the solution setup, it is **strongly advised** to Build the solution to make sure there are no errors; warnings are okay, for now.
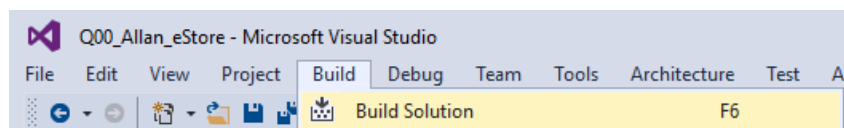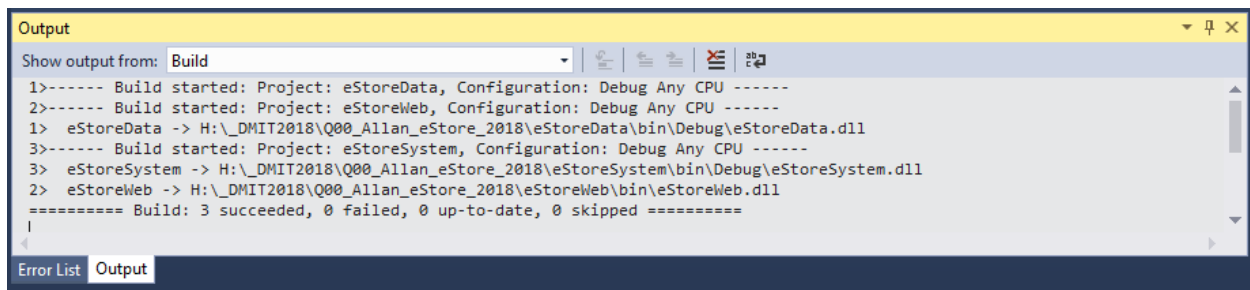

Figure 14: Build Solution

Figure 15: Successful Build Output

Exercise

Complete Exercise 2.3.1