CODE FIRST DATABASE MIGRATIONS

Requirements:

ASP.NET EF CORE 2.2

Razor Pages

CRUD WEB Application

When coding the PMO web application, frequent changes were made to the database model. When changes had to made to the database, errors would occur routinely. Development in a team environment can produce multiple migrations. This makes it difficult to resync the database causing a lack of cohesion of between the entities and their relationships. To keep the database in sync with the application, it was necessary to adopt the ‘**Code First”** methodology when creating the CRUD functionality for the PMO.

The Code first approach was the more efficient way to develop the database in an environment where changes was necessary. The database needs to keep sync with the data models while in development. There will be changes made to the database throughout the life span of the PMO. The application needs to be able to make changes such as adding a new column without having to create a new database. To update the Schema, we used migrations instead dropping and recreating the database. ([Dystra, Smith and Anderson, 2019) Using data migrations introduce he possibility of losing data when you add or update migrations improperly.

MODELS, Razor Pages, Scaffolding

Models of the .cs classes are used to create entities. Essentially the DB Schema.

Razor Pages are ‘Scaffolded” using the Models as Context please refer to the Microsoft.net Tutorials as a refence on how to create Razor Pages in the EF Core ASP.Net. <https://docs.microsoft.com/en-us/aspnet/core/razor-pages/?view=aspnetcore-3.1&tabs=visual-studio>

CRUD Functionality.

The GUI the Administrator profile can be a guide for creating the GUI for the User and Viewer GUI’s. If dummy records are blank, re seed the database and refer to the dot.net tutorials on database migrations for ‘Code First” database. <https://docs.microsoft.com/en-us/aspnet/core/tutorials/razor-pages/sql?view=aspnetcore-3.1&tabs=visual-studio>

The CRUD GUI can Create, Read, Update and Delete entries to the database tables.