

Project: Online ToDo List Web Application

Author: Richard Saldanha

Project Environment:

1. IDE → Visual Studio 2019
2. Project Type → ASP.NET Web Application (.NET Framework)
3. ASP.NET Web Application Type → MVC
4. Framework → 4.7.2
5. Project Name → ToDoListProject
6. Two Test Users →
 1. Email : test@email.com , password: pwd123
 2. Email : user@email.com , password: Power@12345
7. Technology: ASP.NET MVC, C#, Entity Framework, Razor, HTML, CSS, Bootstrap, JavaScript, jQuery, AJAX

Project Approach:

Code First Approach:

→ I have used the code first approach for building the web application because I have focused on the domain of my application and created the Controller and Model class for the domain rather than designing of database first. The controller is created using the MVC5 Controller with views using the Entity Framework data context and the model is created using the To Do List Project Models and Data Context Class which is Application Db Context.

Project Design Pattern:

Domain Driven Design Approach:

→ I have used the domain driven design approach because it helped me to build the web application based on the project requirement with the help of properties defined in the model class. Also, DDD (Domain Driven Design Approach) emphasize on business focused software development based on requirement and it has several advantages such as Loose Coupling, Flexibility, Testability and Maintenance.

Project Security:

1. Authentication:
 - ⇒ For the authentication of this project, I have used the Individual User Accounts option in which all the user's profile information will be stored on the server.
 - ⇒ Identity model is the model for identity which I have used to handle user authentication.
 - ⇒ SSL is Enabled in my project
 - ⇒ Authorization is applied at the root level using [Authorize] means the user will have to login to the system to carry out the To Do Activity .
 - ⇒ Login and Registration is created for user.

Project Additional Information:

Project Migration:

- ⇒ I have also enabled Migrations through the NuGet Package Manager Console so that I can update as well as change the data model without having to re-create or drop the database.

Project Progress:

- ⇒ I have also created a progress bar which indicates the to do list completed.
- ⇒ On click of IsCompleted checkbox the user will see that the to do item in the to do list is marked out with refraction in the progress bar.

Project Future Work:

- ⇒ Add Dragging UI functionality for moving around the To Do activities in the To Do List.