eShop

PROJECT DETAILS

Arun Chougule | Project Details | 20 Aug 2021

Tables

Used PostgreSQL database. Designed tables considering the normalization rules.

Used Unique Identifier as primary key.

Tables are as follows

1) User – To store application user information
2) Item – To store Item information
3) ItemDetails – To store details information of items

Architecture

Used Client Server architecture considering the separation of concerns (SOC). Application has 3 parts

1. Model.
   For data binding on both web and API project used only one Model Class library
   a. User.
   b. ItemCategory.
   c. Item.
   d. ItemTran:- Complex object class used to perform insert operations on Item and ItemDetails tables using transaction scope
2. User interference.(UI)
   a. Asp.Net Core Web MVC app. Using HttpClient class I'm calling web APIs
   b. Wrote own WebAPIClient static class to call the APIs from UI.
   c. GET,POST, PUT methods are async to scale the avoid the latency
   d. Used JWT token for authorization and token stored in session.
   e. Used Serilog to log the errors.

3. Web API.
   a. ASP.Net Core Web API version 3.1 project, Layered architecture.
   b. Designed considering the Solid principles
   c. Used Repository pattern
   d. Created a separate configuration class for dependency injection

e. DataAccessLayer has 2 layers abstraction layer and concrete implementation.
f. BuisnessLayer designed as per the repository pattern.
g. Used OOPs concepts when required.

4. Unit Test project is added to test the methods. Used MOCK objects to test the methods.
5. To run the application. Right Click on the main solution and click on properties.
   Select multiple startup project option and start the UI and Web API projects. As shown in the below.