

Thinkbridge Task

ShopBridge Product Api

ShopBridge

Introduction

Shop Bridge is an e-commerce application. This document revolves around the backend webapi for products feature. This products API has capabilities such as Create, Read, Update, Delete for the products/inventory.

This API serves as RESTful Api for the management of products.

Technologies Used

This section shows technologies used to make the webapi

Framework:

- Asp.Net Core 6 (Minimal Api)

Runtime:

- .Net 6

Database

- SqlServer LocalDB

ORM

- EntityFramework Core 6

How to run a project?

Here are the steps listed to run a project.

Prerequisites

- Visual Studio 2022
- .Net 6

Steps

1. Clone the repo
2. Open the solution in vs2022
3. Make sure dependencies has been restored
4. Open Package Manager Console (Tools -> Package Manager -> Package Manager Console)
Run these two commands in PMC
 - a. Add-Migration InitialCreate
 - b. Update-Database
5. Set ShopBridgeBackendMinimalApi as your startup project (if it is not set as startup project)
6. Click on Run



7. Browser will open automatically and shows Swagger UI Page

Test the Project

This project is implemented Swagger UI so you can use graphical client for sending request to the webapi.

You can access swagger at <https://localhost:{port}/swagger>

Extras:

In this project I've tried to use all the latest techs and tried to implement good architecture here are the following things to consider and to which I am proud of

1. Implemented Everything in Minimal Api that launched with Asp.Net core 6 in November.
2. Implemented Dependency Injection
3. Created Services separately with Interface implementation and registered it with dependency injection so it can be consumed later by controller with constructor injection.
4. Used EntityFramework core with MS SqlServer LocalDB (this can be replaced by other SQL server just by changing Connection string in appsettings.json)
5. Made some minor changes so browser opens with swagger page as a default page

Time Spent

Development/Debugging/Testing – 90 mins

1. Data store design – 10 mins
2. API and service logic – 45 mins
3. Debugging – 35 mins

Documentation – 60 mins