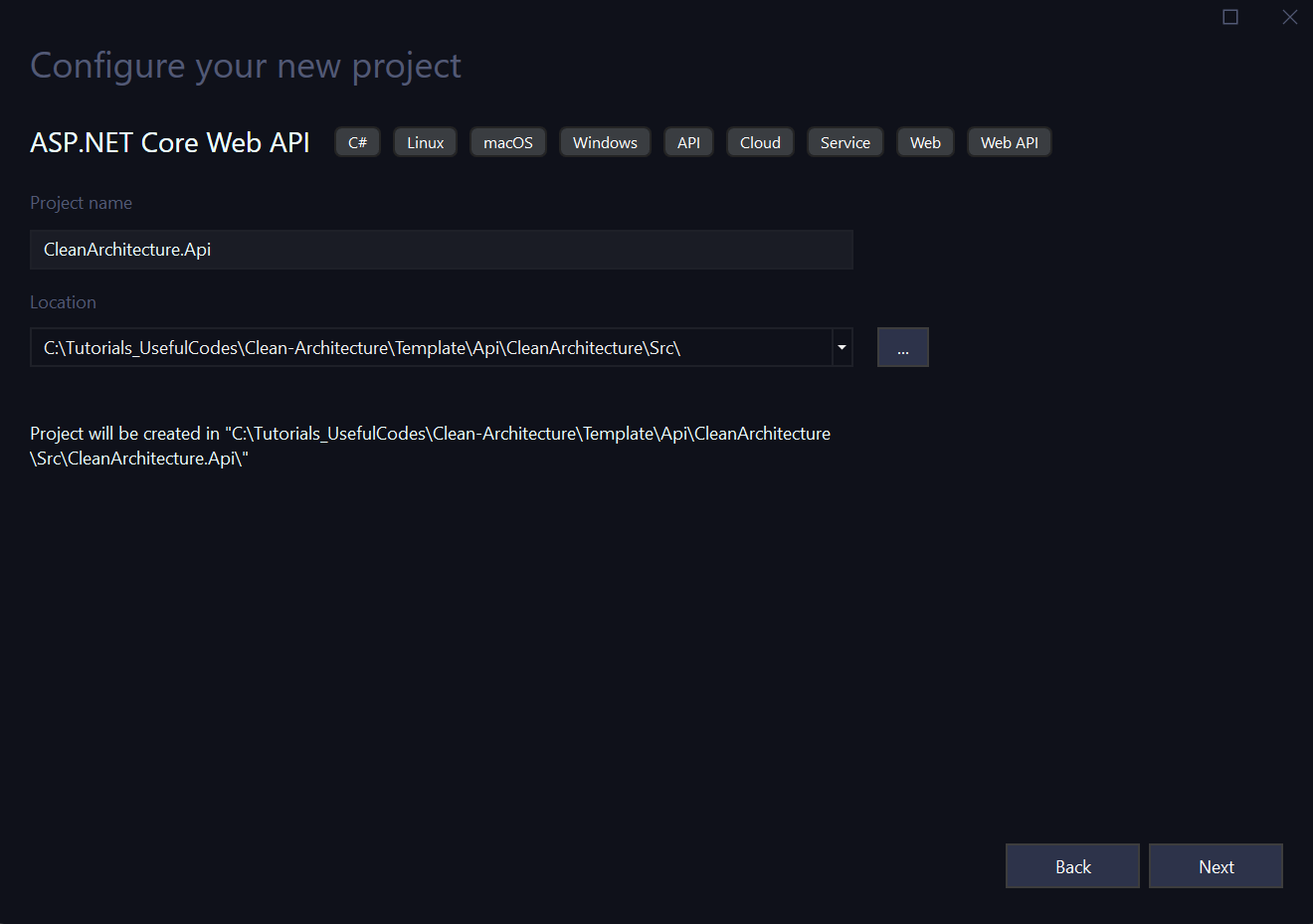
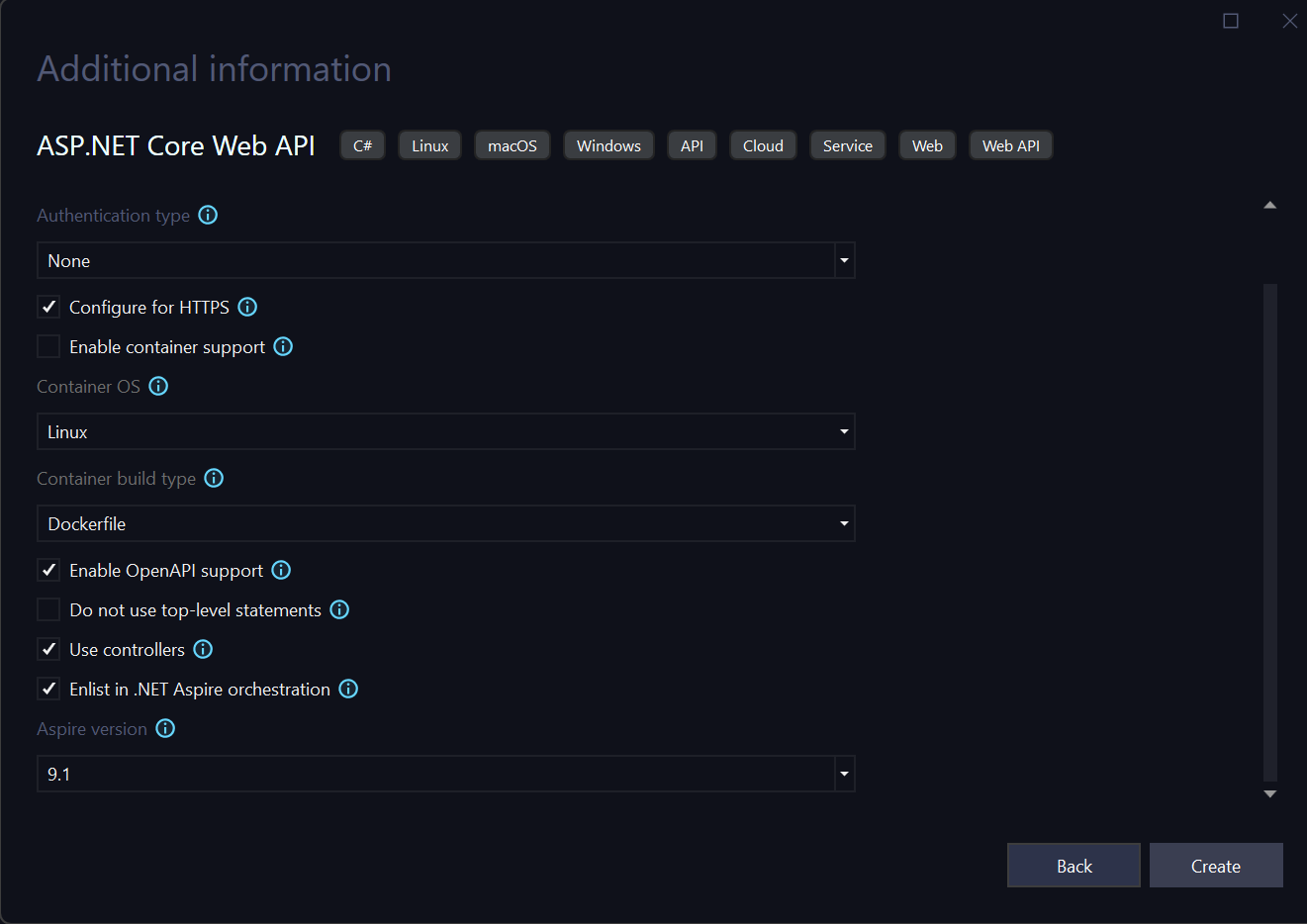
Clean Architecture Project Template -- Api

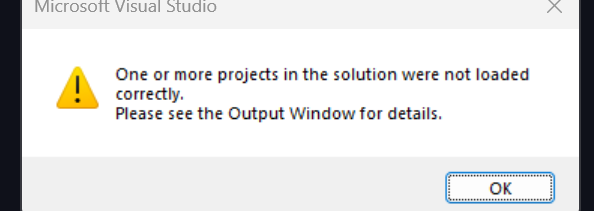
Right Click on Src Folder and Create the Asp.Net Core WebAPI Project





Make sure to check Enlist in .NET Aspire orchestration. This creates the Aspire Project in the wrong folder, click to remove and move the project under Src Folder.

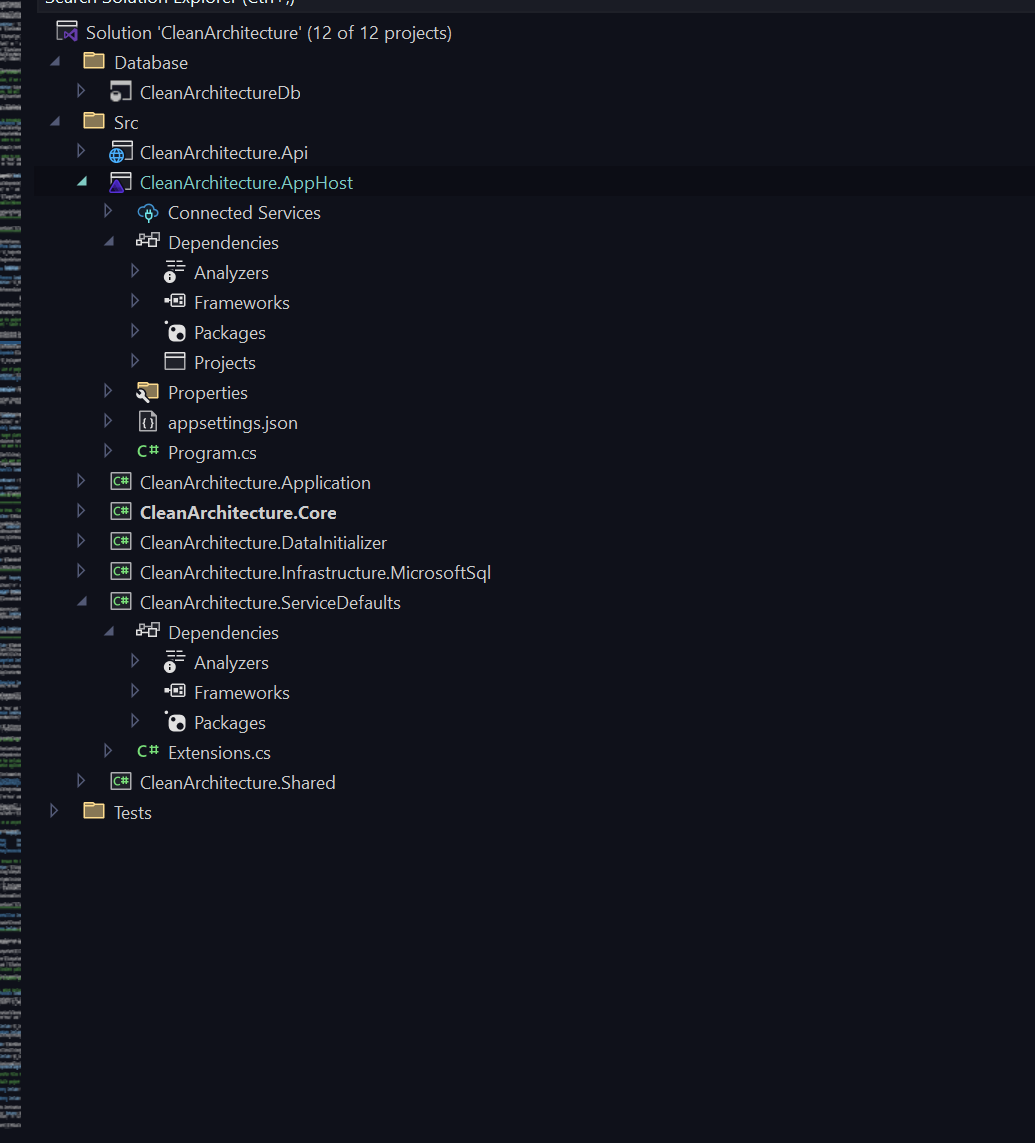
Note: Close Solution and move both Shared and Aspire Host project inside the Src Folder. When you reopen the project you will see the following



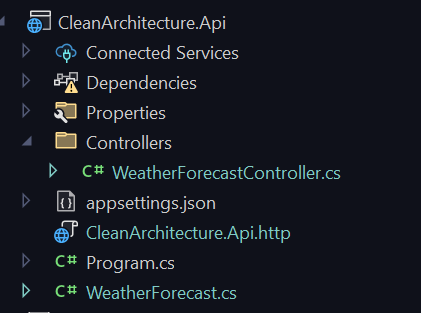
Click Ok to Continue. Remove both projects Service Defaults and Aspire Host and Re Add them back.

Fix References in both Service Default and Aspire Host.

Create the Project, your Project Structure looks like the following



Remove the following Highlighted files from Api Project



Add Reference to the Following if not exists:

1. Service Defaults
2. Core
3. Application

Add the following to the core project from Core folder

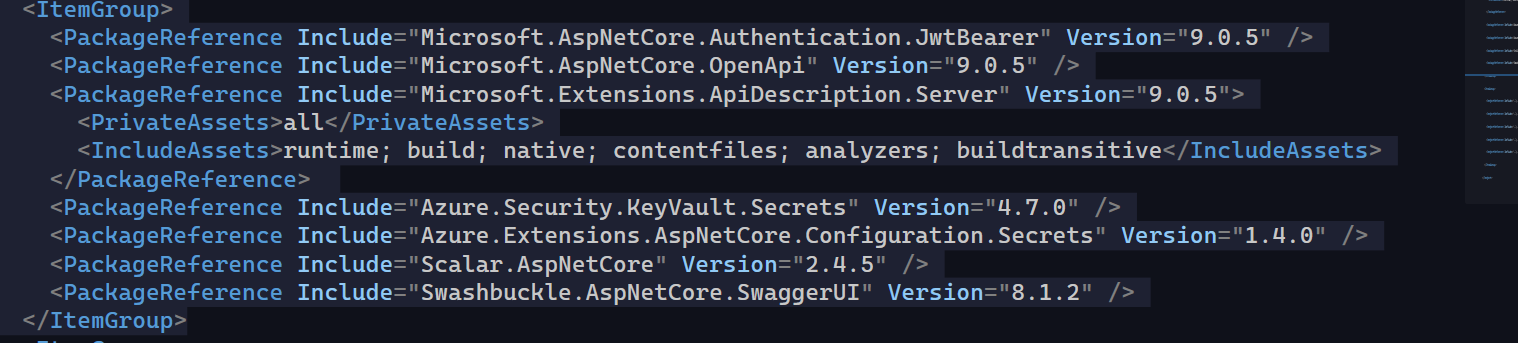
1. IDispatcher

Add the following to the Application from Application Folder

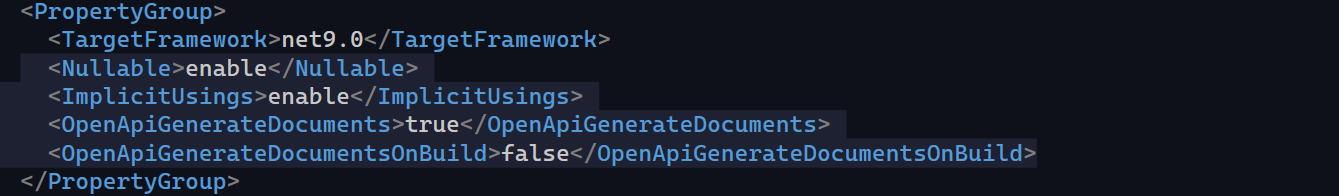
1. Dispatcher 🡪 Features Folder

Add Dependencies in the Application 

Install the following NUGET Packages



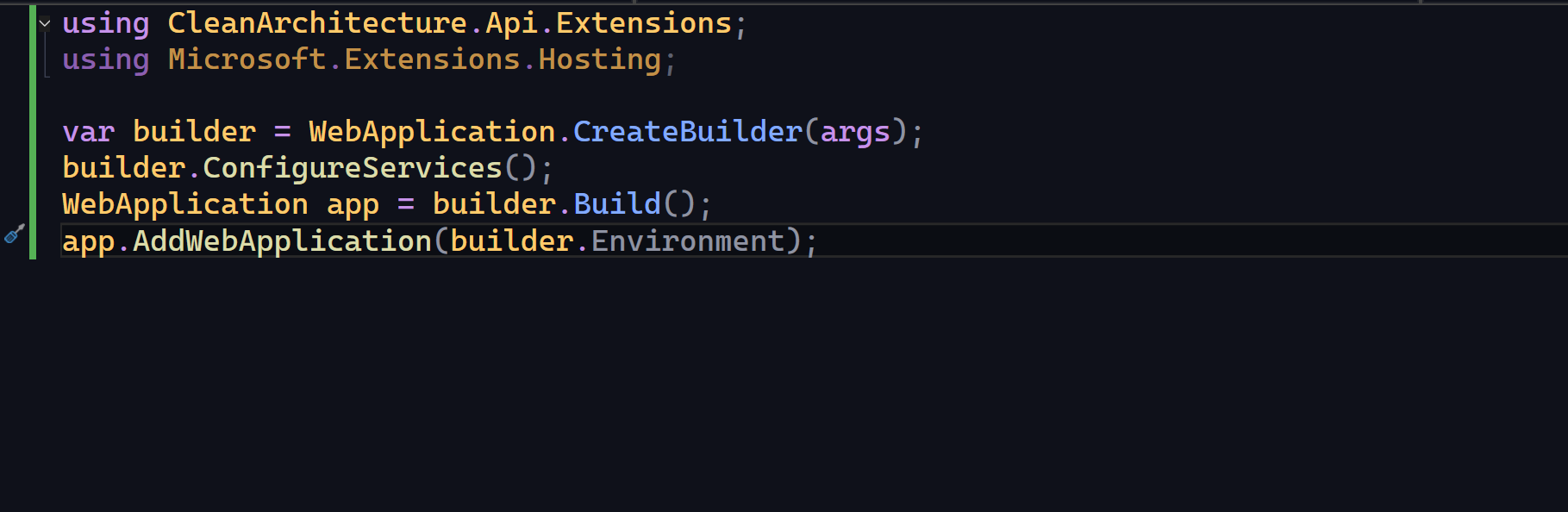
Also Ensure the following:



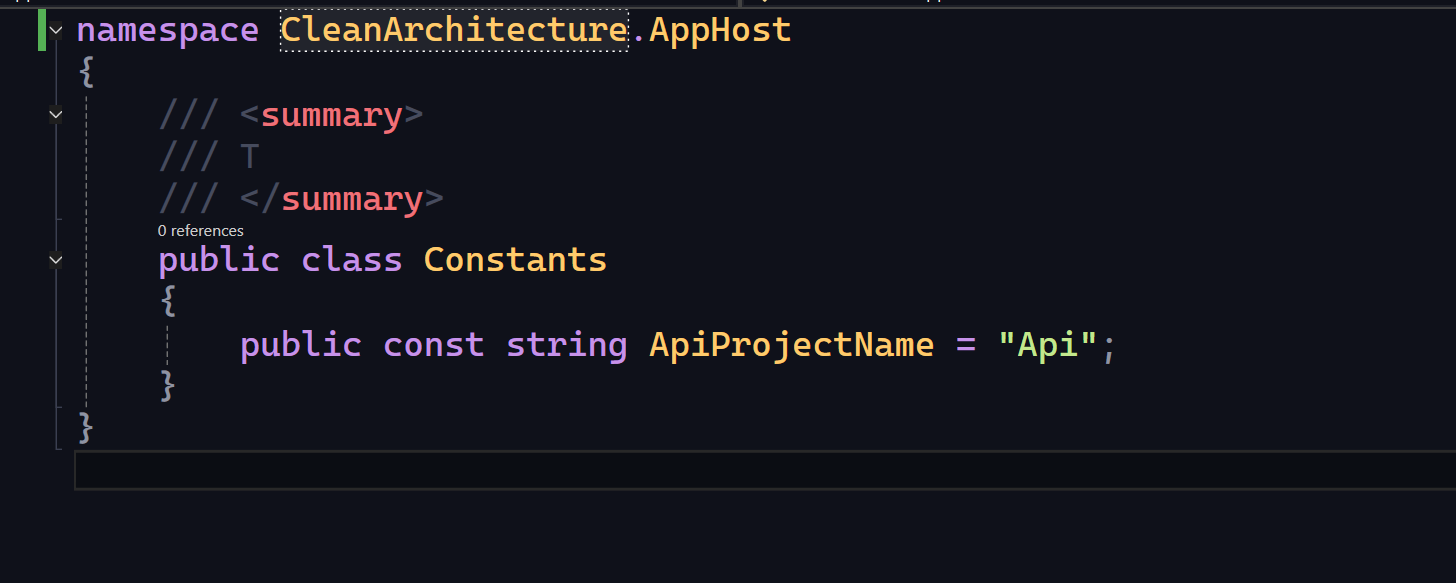
Create a Folder Extensions, and add the following

1. ApplicationBuilderExtensions.cs
2. WebApplicationBuilderExtensions.cs

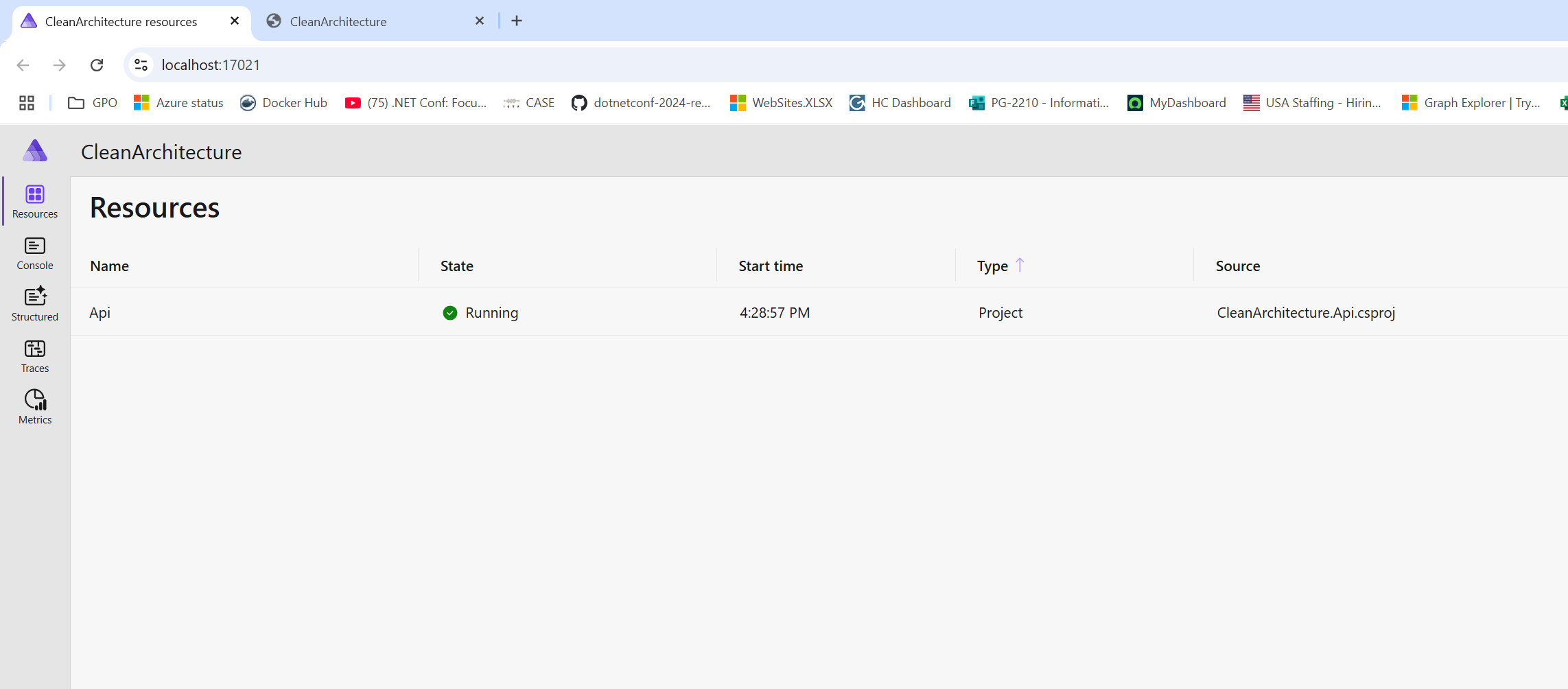
Open Program.cs File and replace it with the following:



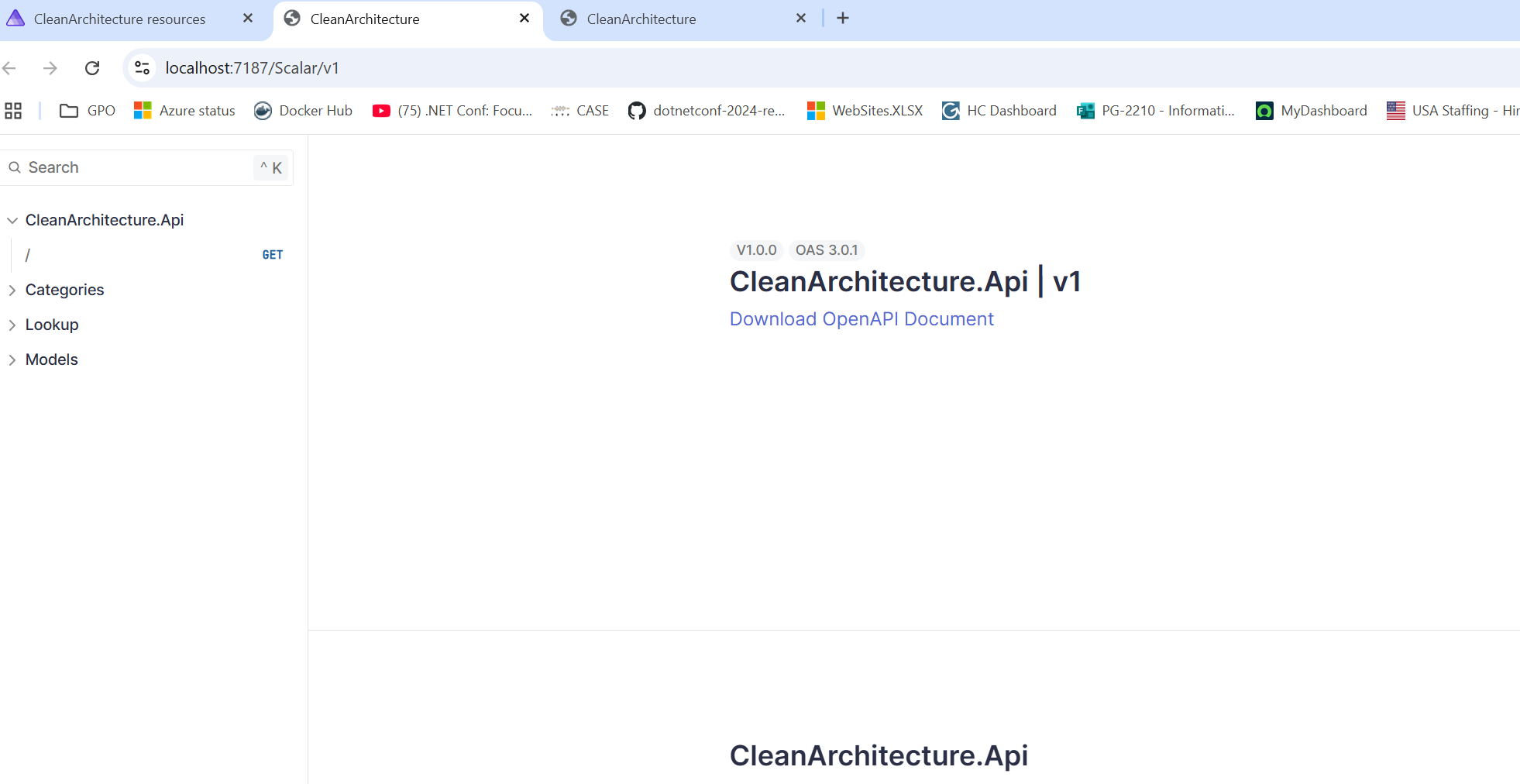
Open AppHost Project and add a the following file:

1. Add Nuget package: Docker.DotNet
2. Constants.cs 
3. Add Extensions.cs file 
4. Progrom.cs

Run the AppHost Project,



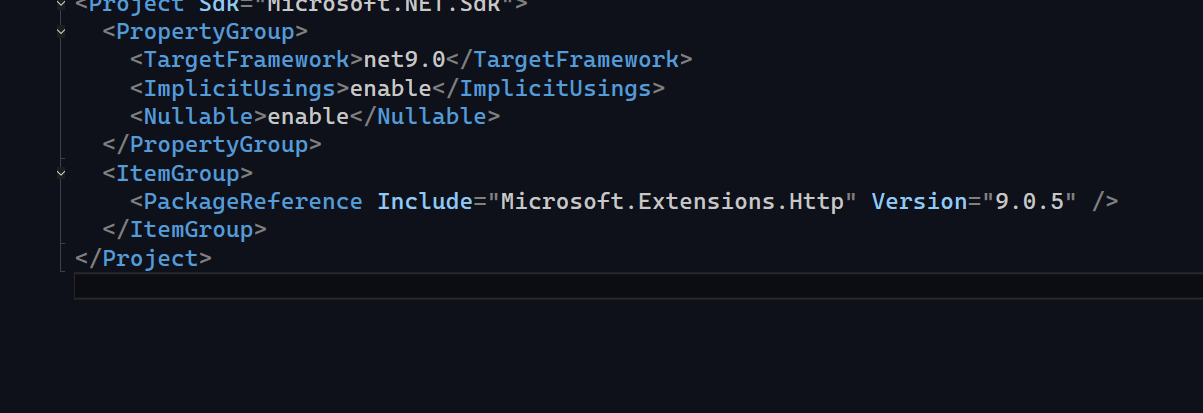
Click on the Link to View the Scalar UI.



Add Api Shared Project

Right Click and Create CleanArchitecture.ApiShared library project in the Src folder

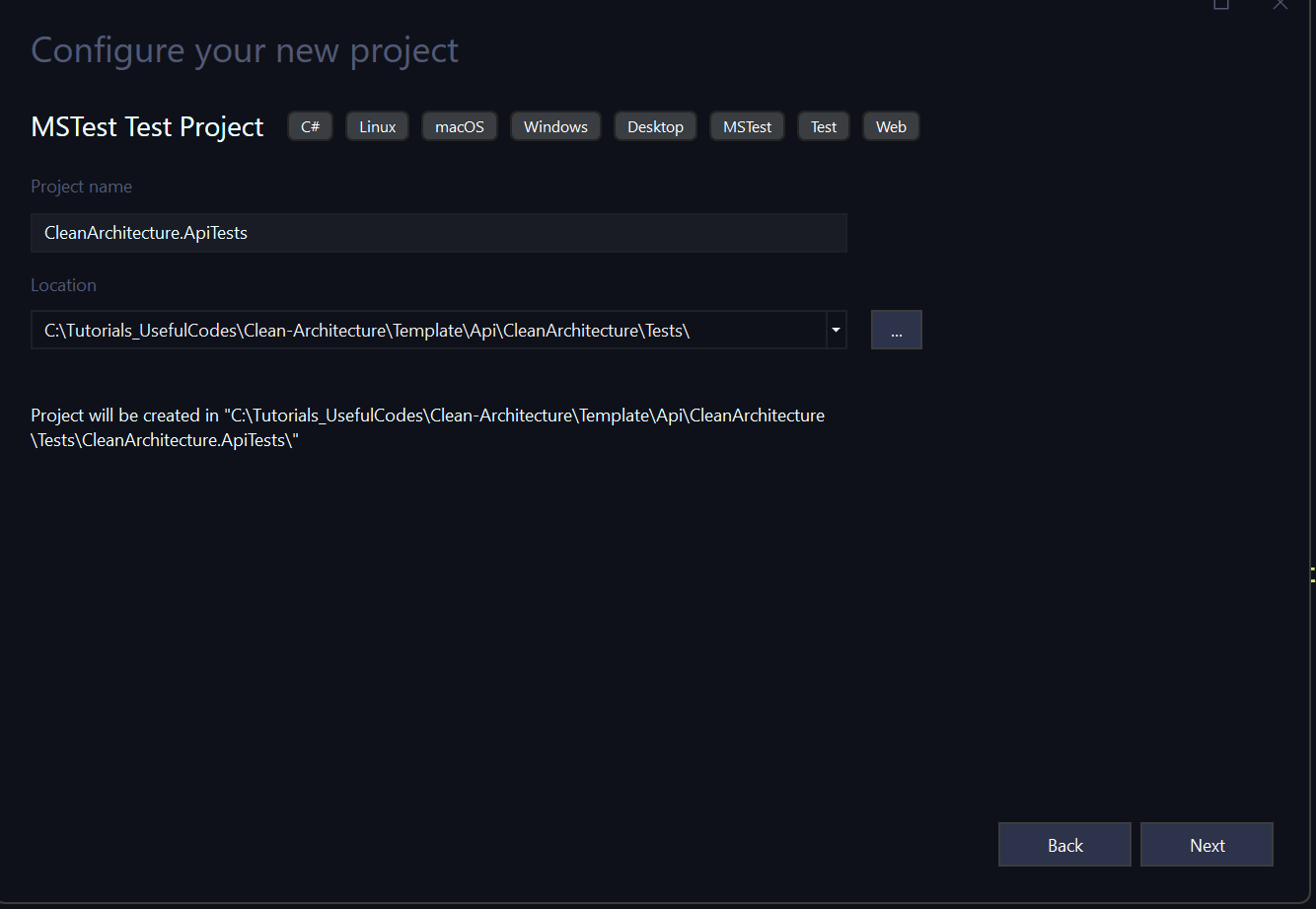
Project file look like the following:

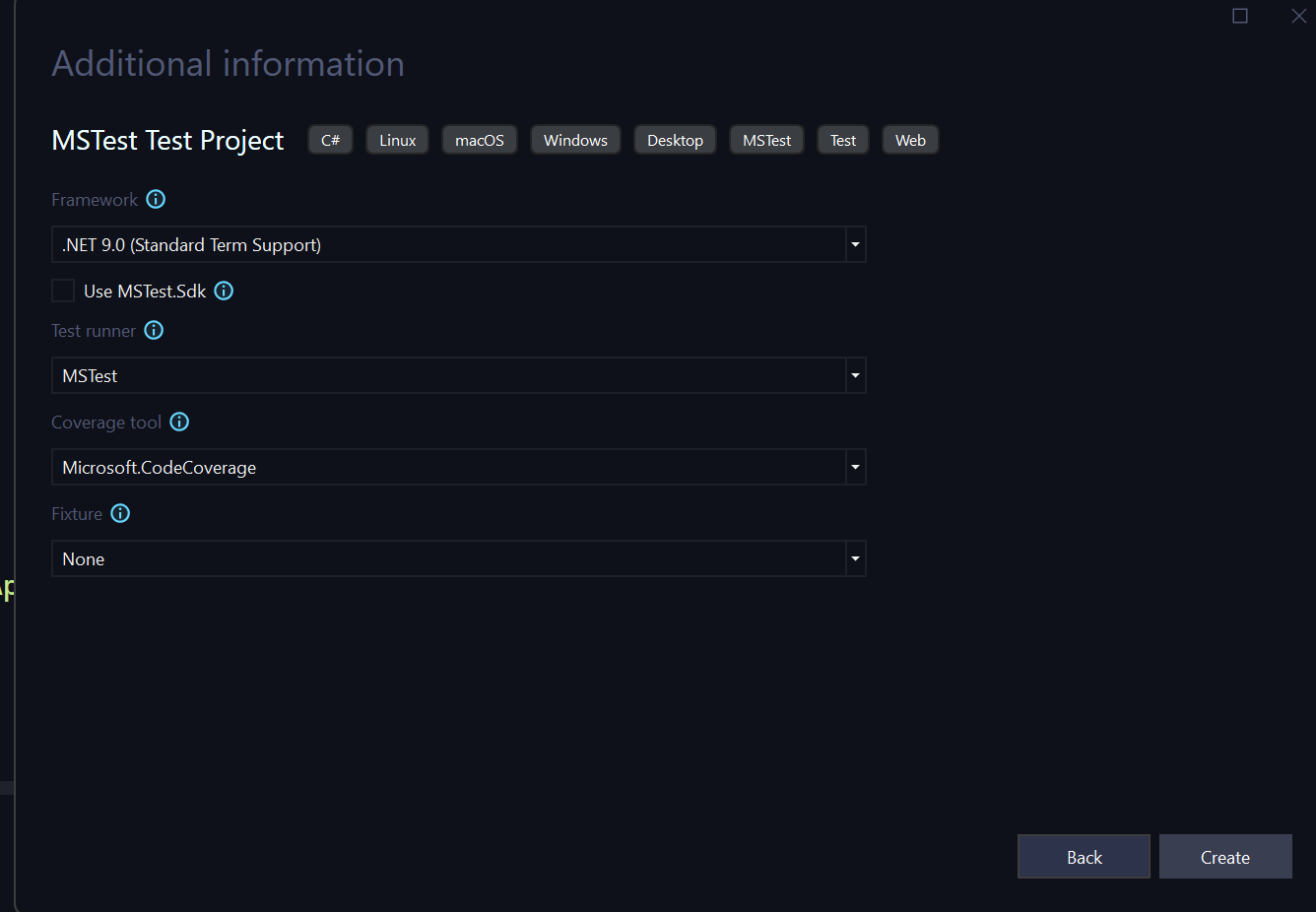


Copy files from ApiShared Folder to the root of the project. Build the Project

Add Api Test Project

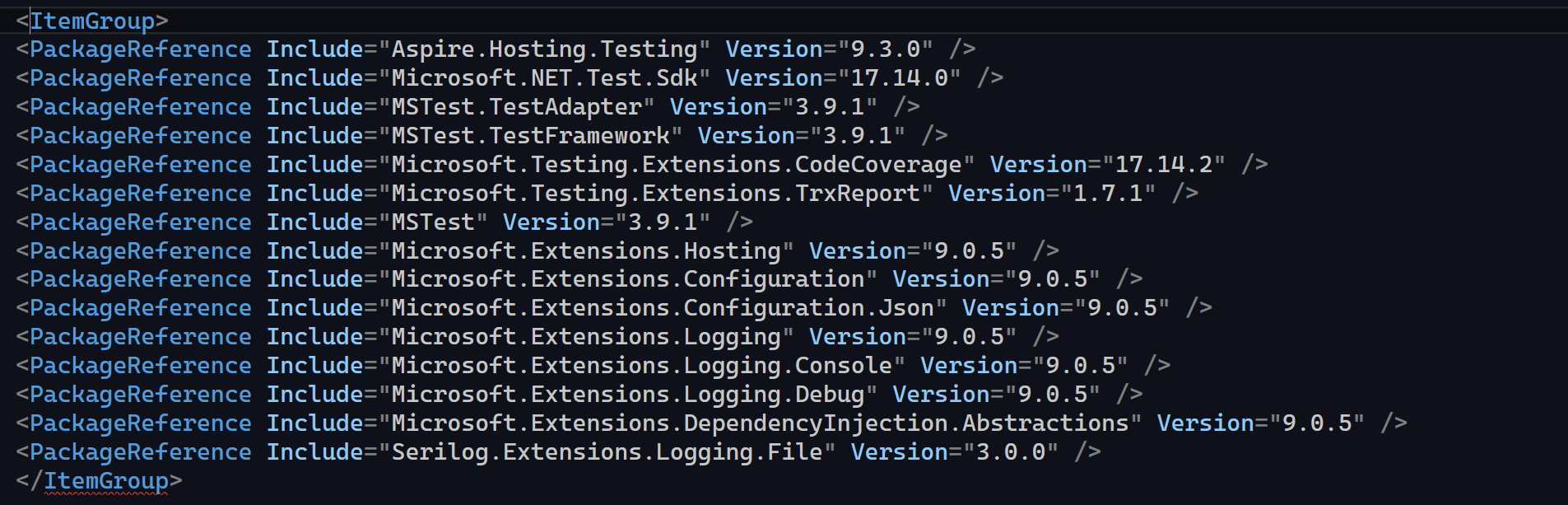
Right click and add MsTest Project under Test folder





Click to Create the Project.

Ensure the following nuget packages:



Add Reference to ApiShared, AppHost Project

Copy Files for the ApiTest folder to the root of the project

Copy File if needed from AspireHost, remove connectionString from api, and RunDbInitializer variable

Run the project, you should see container running for your sql server.

Note:

Do not forget to change GetAll method to use Query String, and remove http from the base address of launchprofile in api project