[**I.** **Architecture & Set-up environment** 2](#_Toc98938342)

[**II.** **Database** 5](#_Toc98938343)

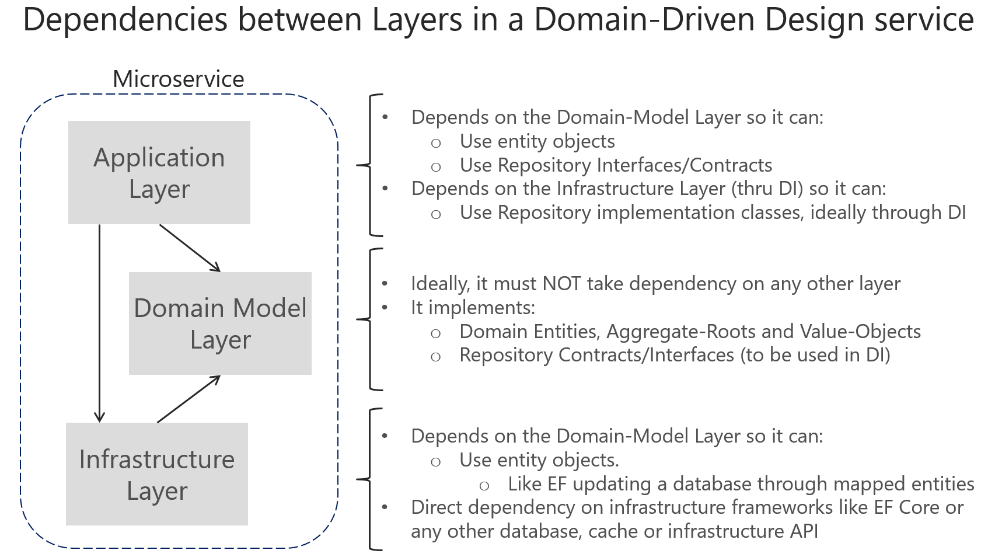
[**III.** **Results** 6](#_Toc98938344)

[**IV. Unit Test** 10](#_Toc98938345)

Assessment for **Simpson Strong-Tie Vietnam**

1. **Architecture & Set-up environment**

**Architecture**: base micro-services, DDD pattern, reference(s):



Reference [here](https://docs.microsoft.com/en-us/dotnet/architecture/microservices/microservice-ddd-cqrs-patterns/ddd-oriented-microservice)

**Tools**:

Visual Studio 2022, .net 6.0

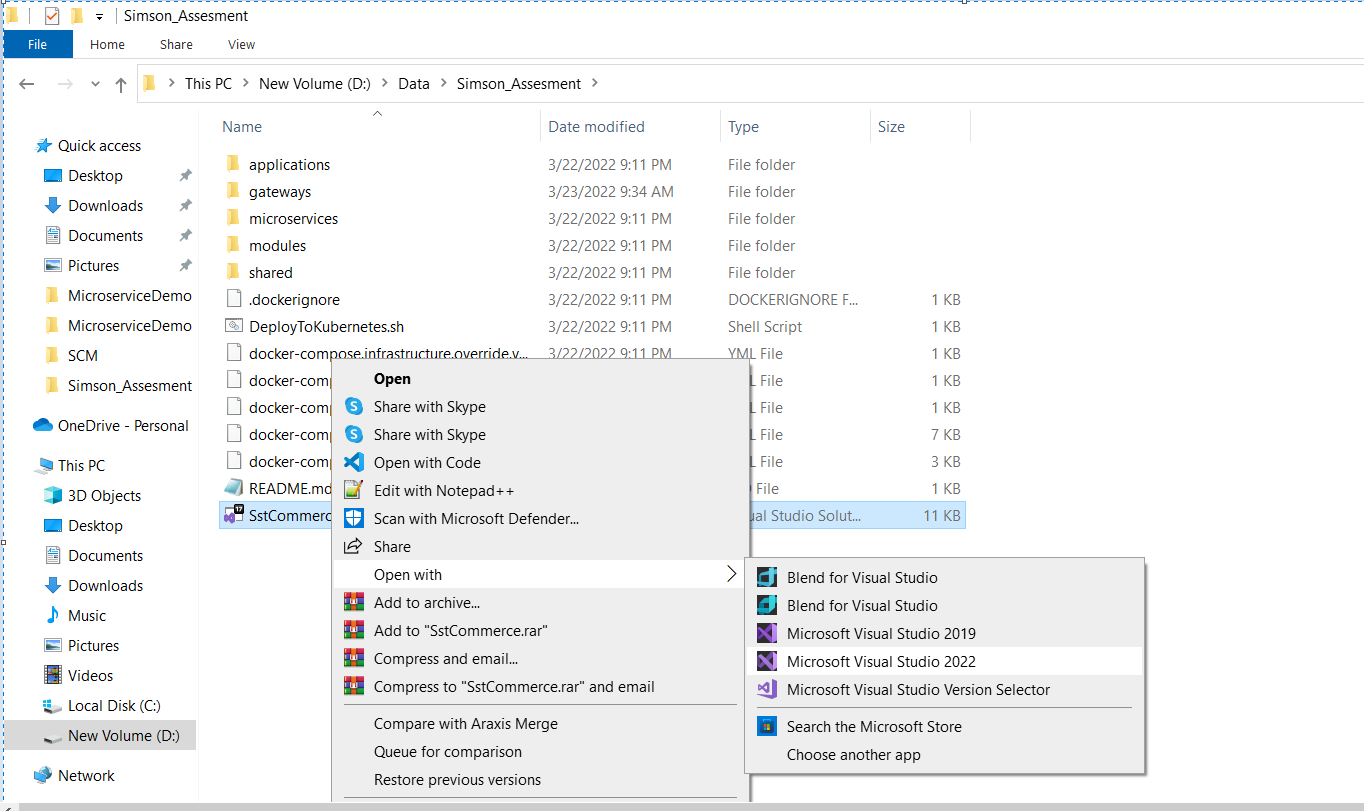
SQL Server V18.9.1

Framework: ABP framework (Installation instruction(s) [here](https://docs.abp.io/en/abp/4.4/Tutorials/Todo/Index?UI=MVC&DB=EF)), reference [here](https://docs.abp.io/en/abp/4.4/Microservice-Architecture)

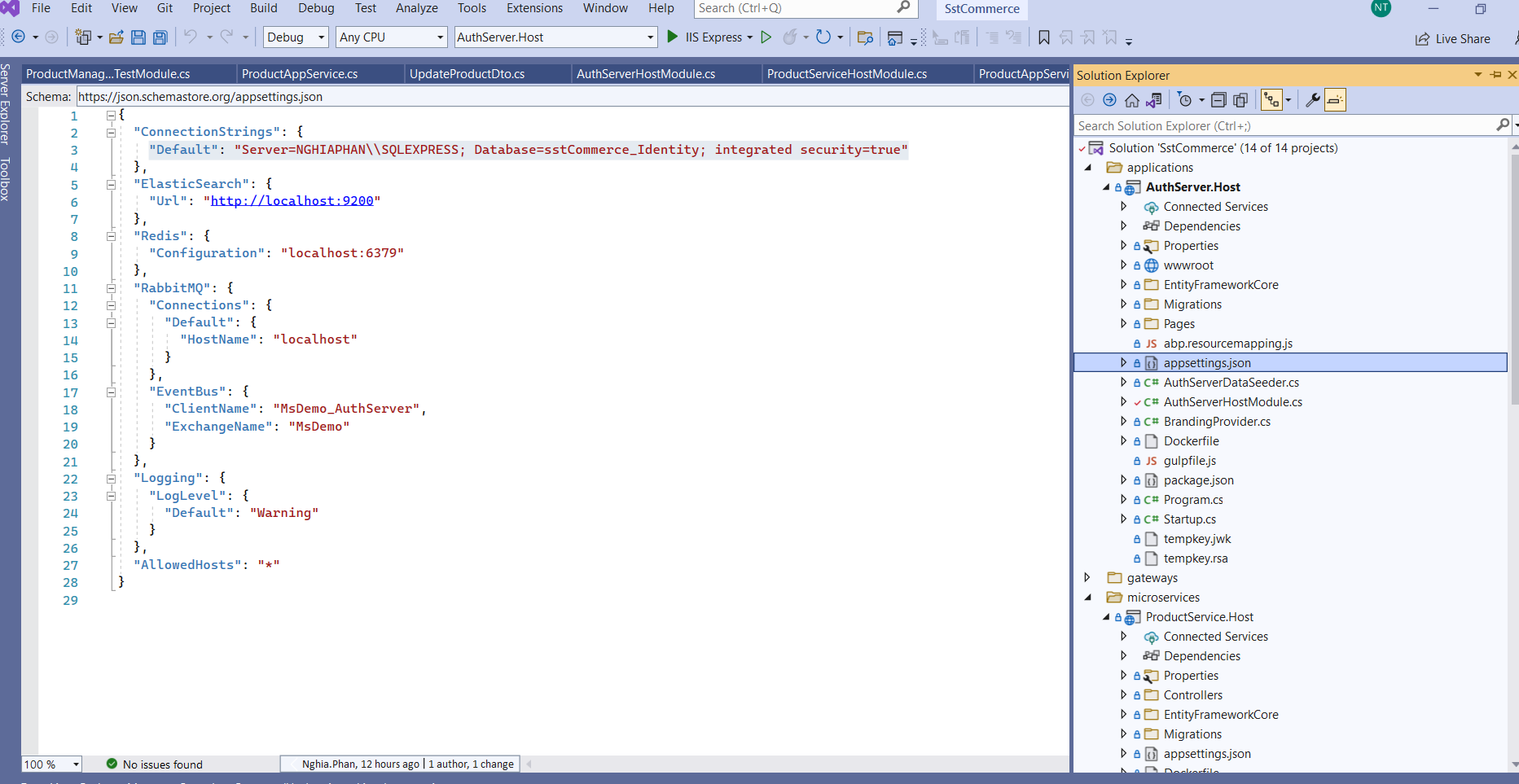
Source: <https://github.com/daicanghia/Assessment.git>

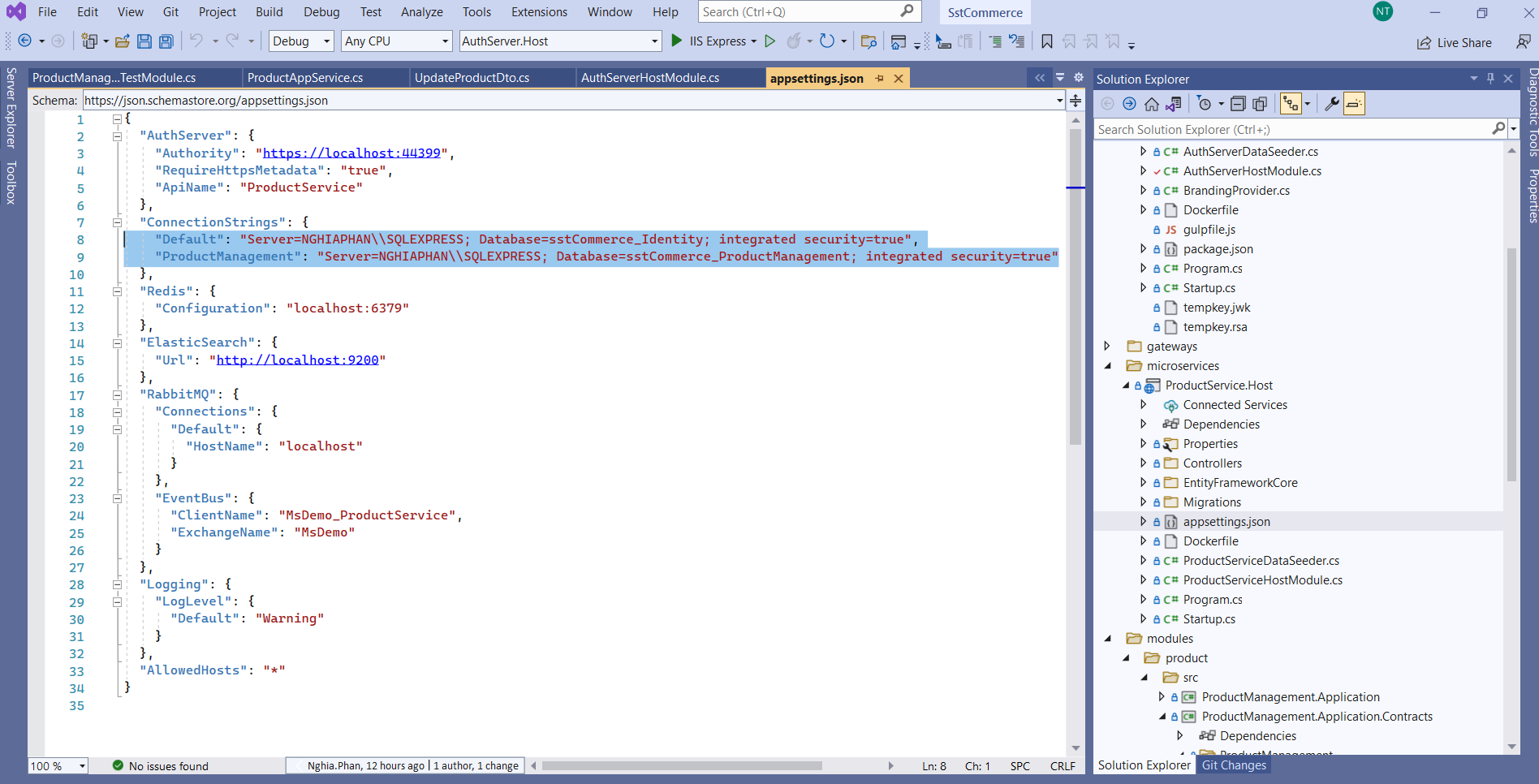
**Steps to run:**

**Step 1:** Open solution with visual studio 2022



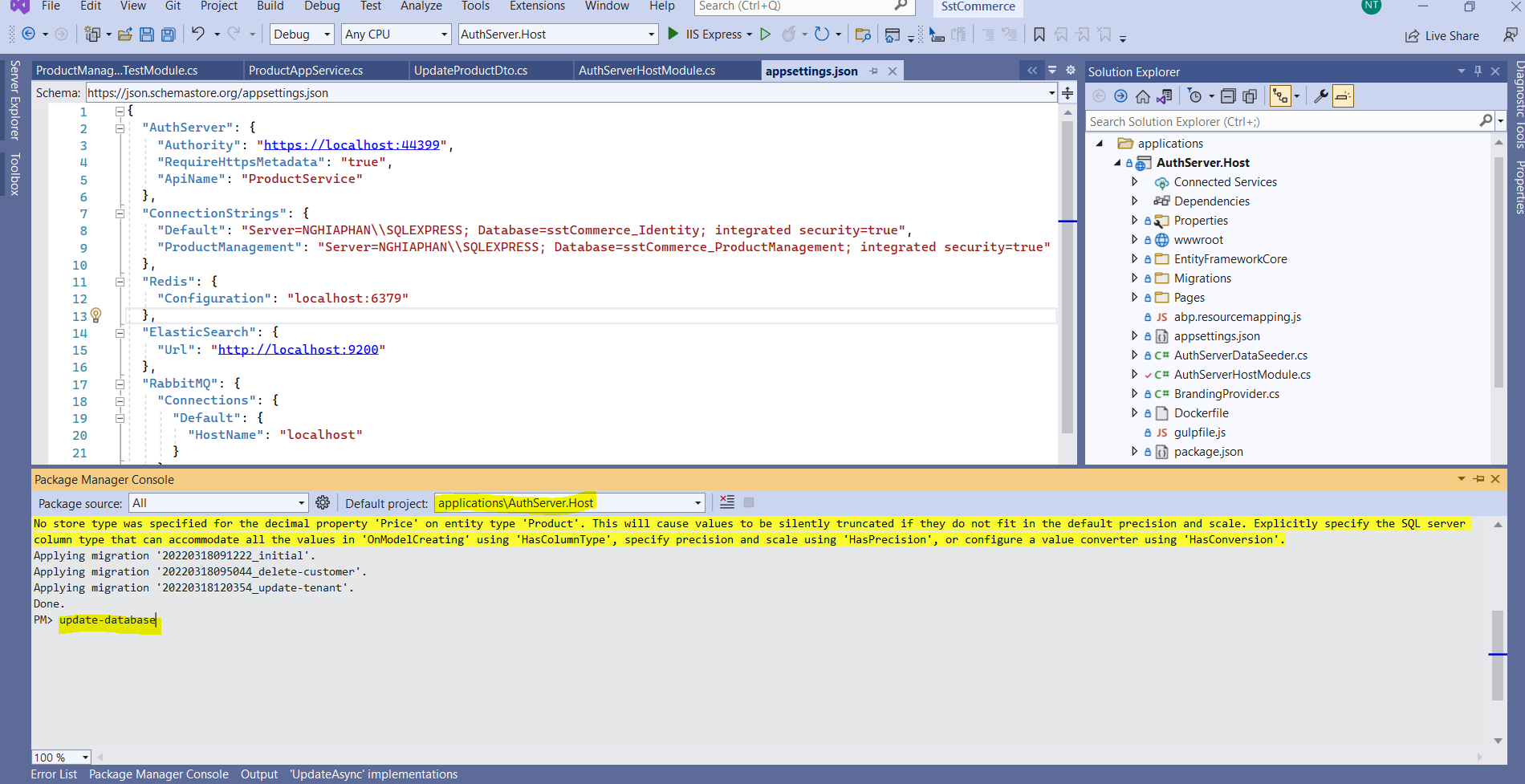
**Step 2:** Configure **AuthServer.Host** and **ProductService.Host**



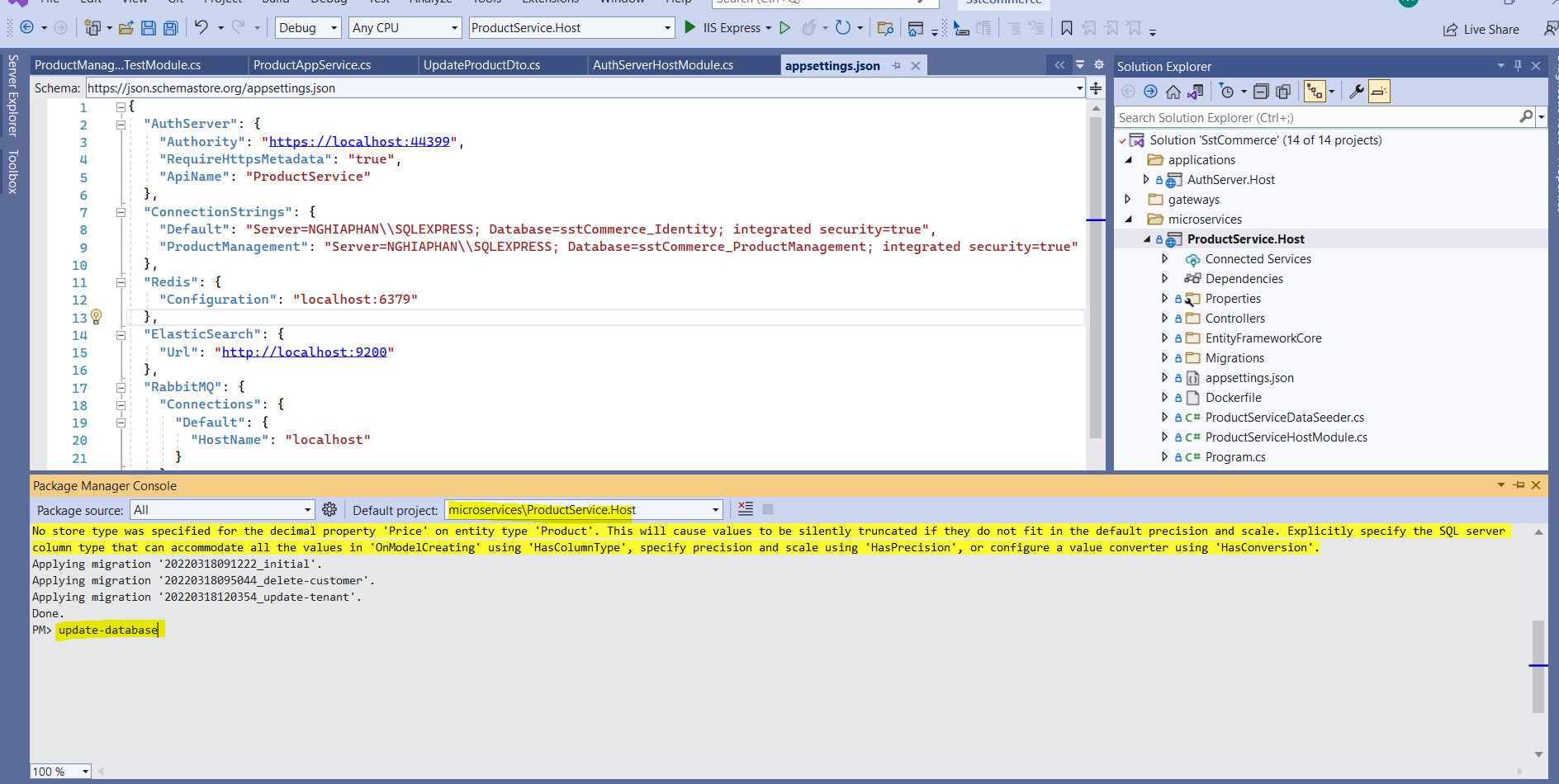


**Step 3:** Update migration **AuthServer.Host** and **ProductService.Host**

**Step 3.1**: **Set As Startup Project** for **AuthServer.Host** => Open **Package Management Console** => Type command: **update-database**

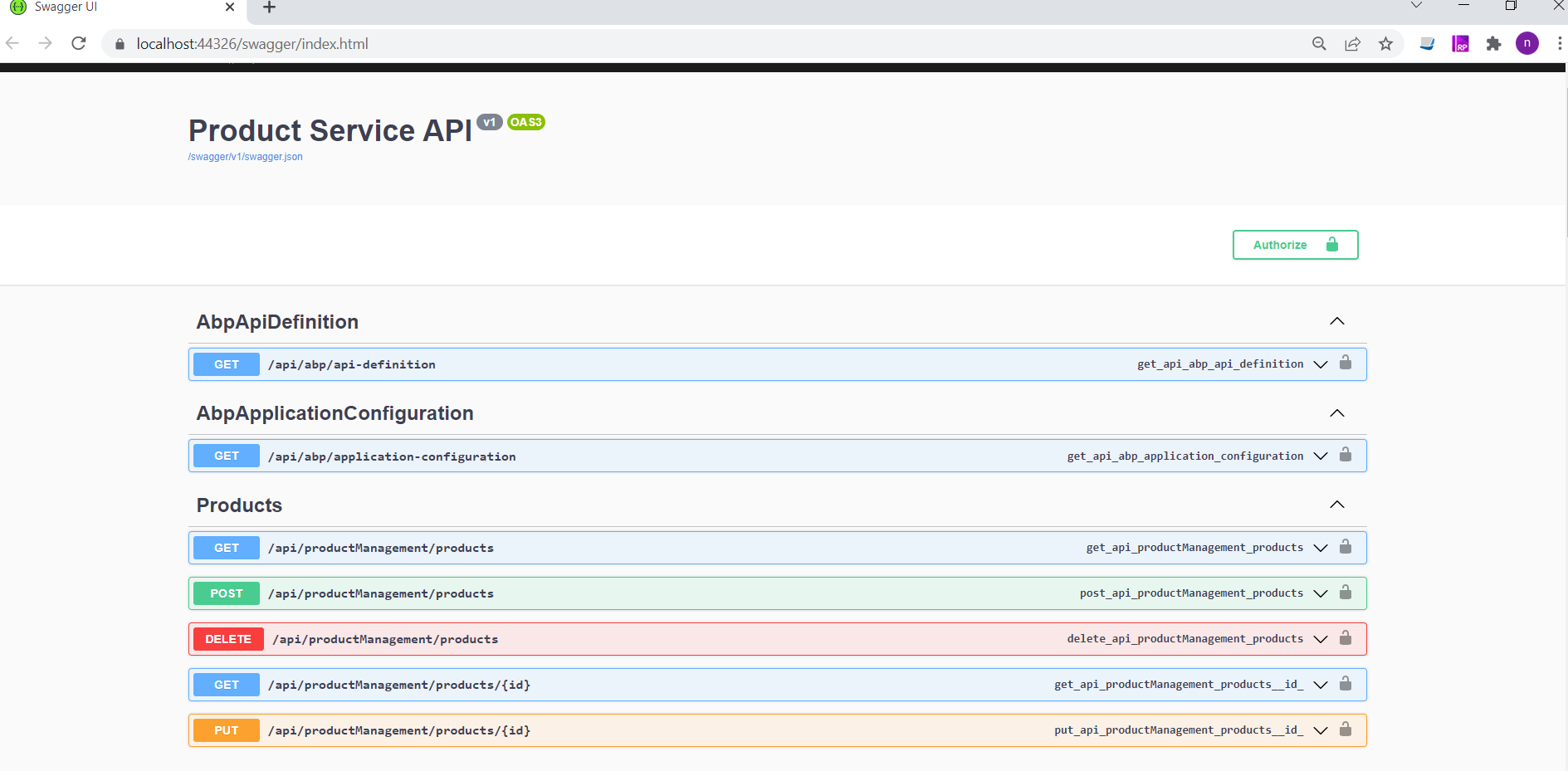


**Step 3.2**: Similar as above for **ProductService.Host**



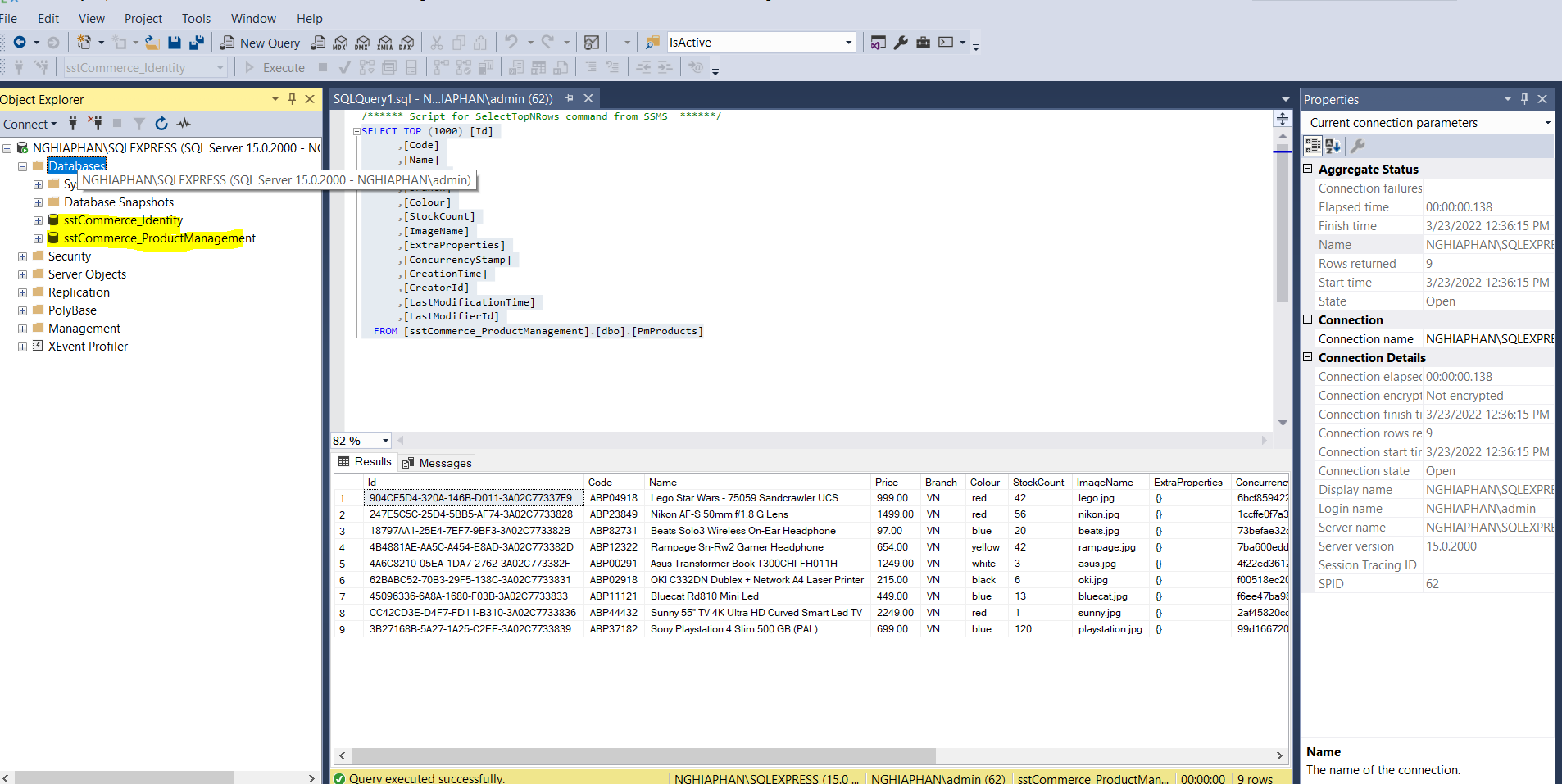
**Step 4:** Data seed **ProductService.Host**

**Set As Startup Project** for **ProductService.Host => Run (Ctrn + F5)**

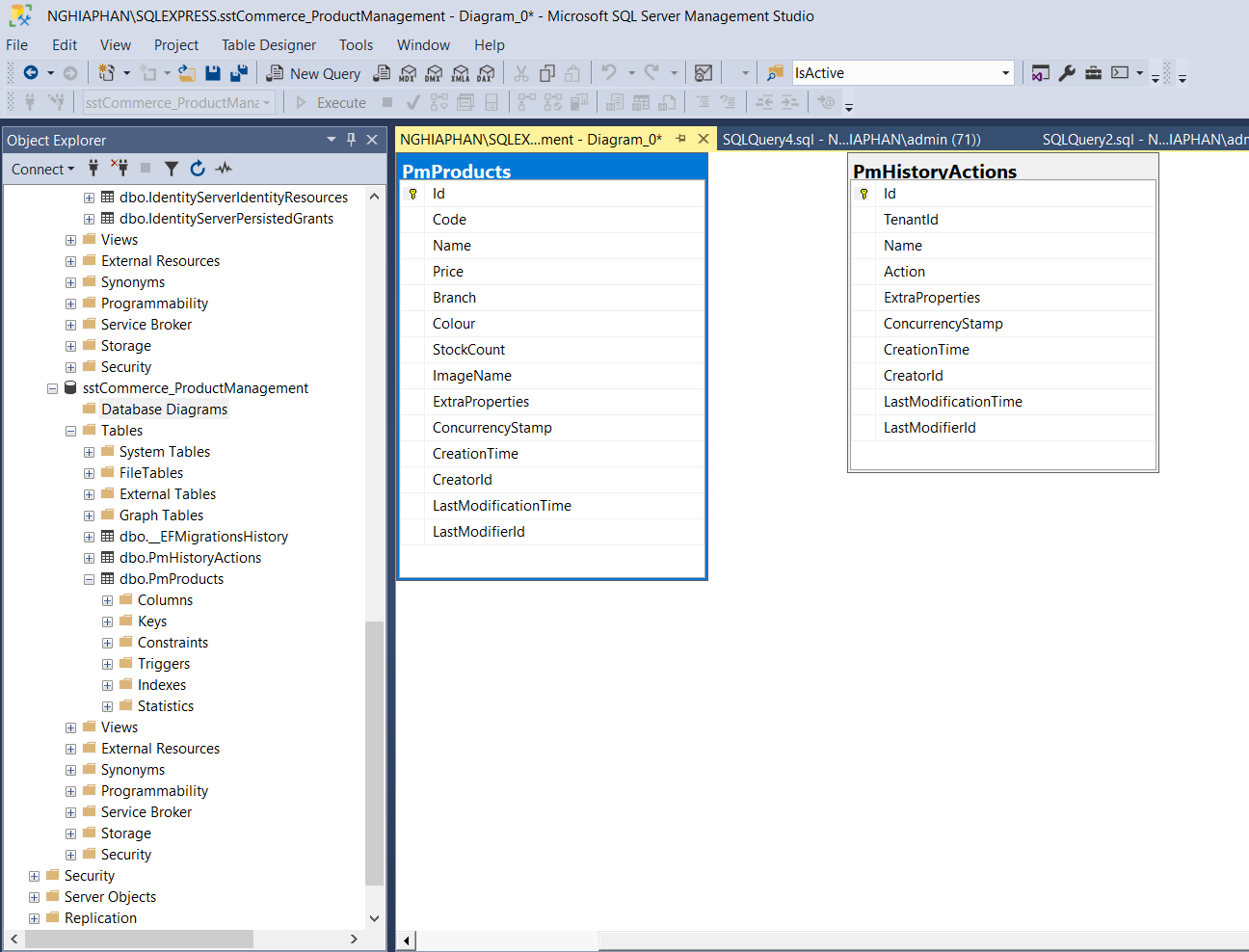


1. **Database**

Because this is micro-service architecture, User & Product store in separate database(s), this is multi-tenant architecture



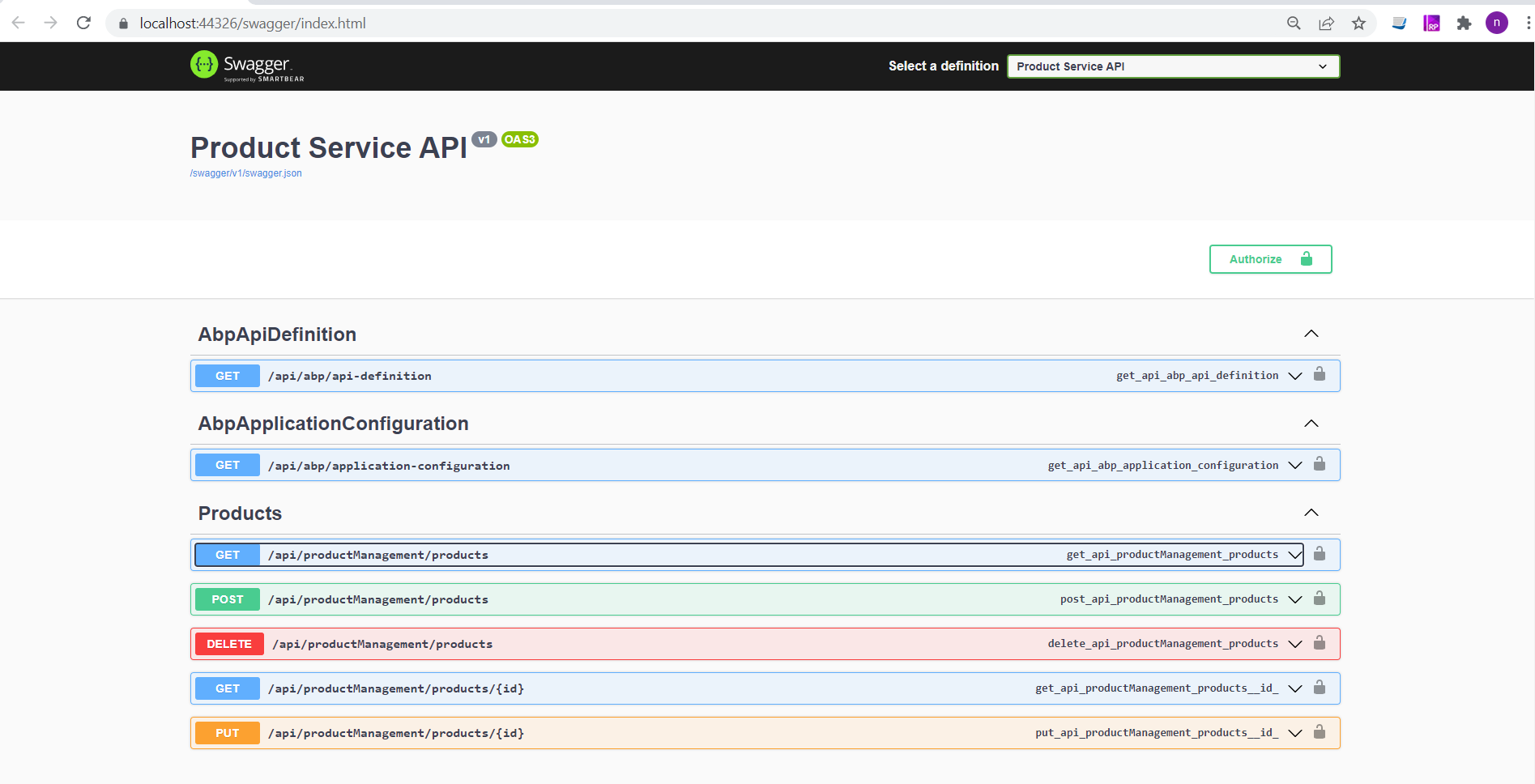
A simple diagram



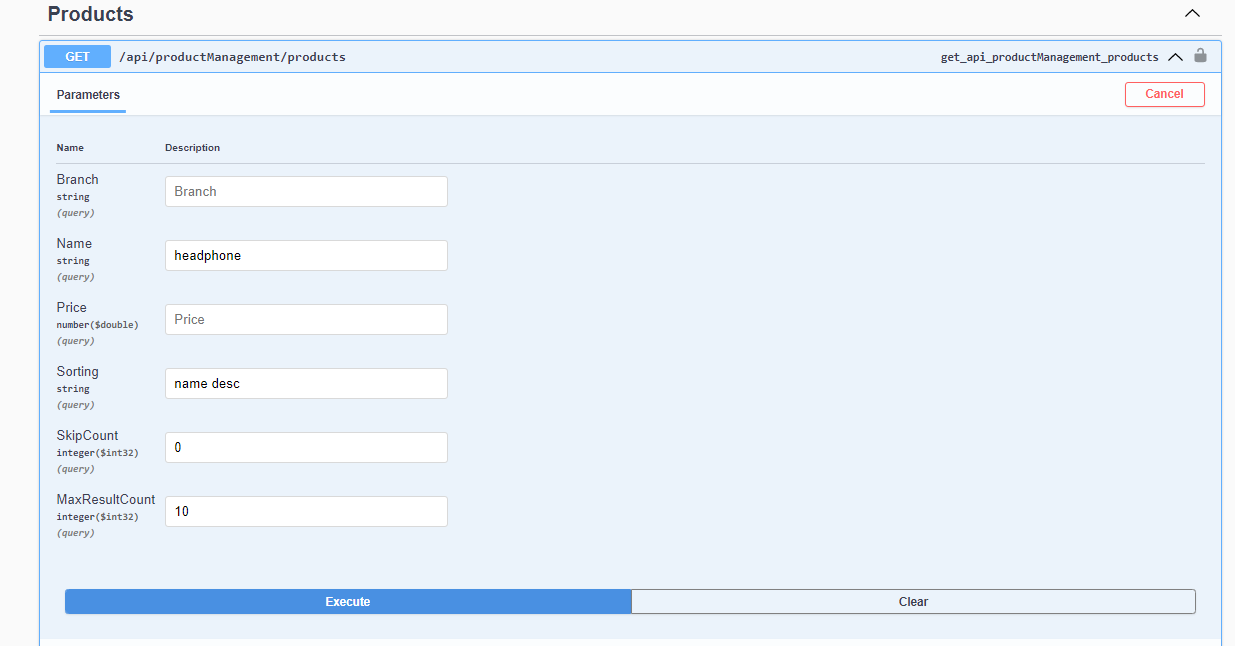
1. **Results**

This assessment is only API for product, not authorization & authorize

This is CRUP API

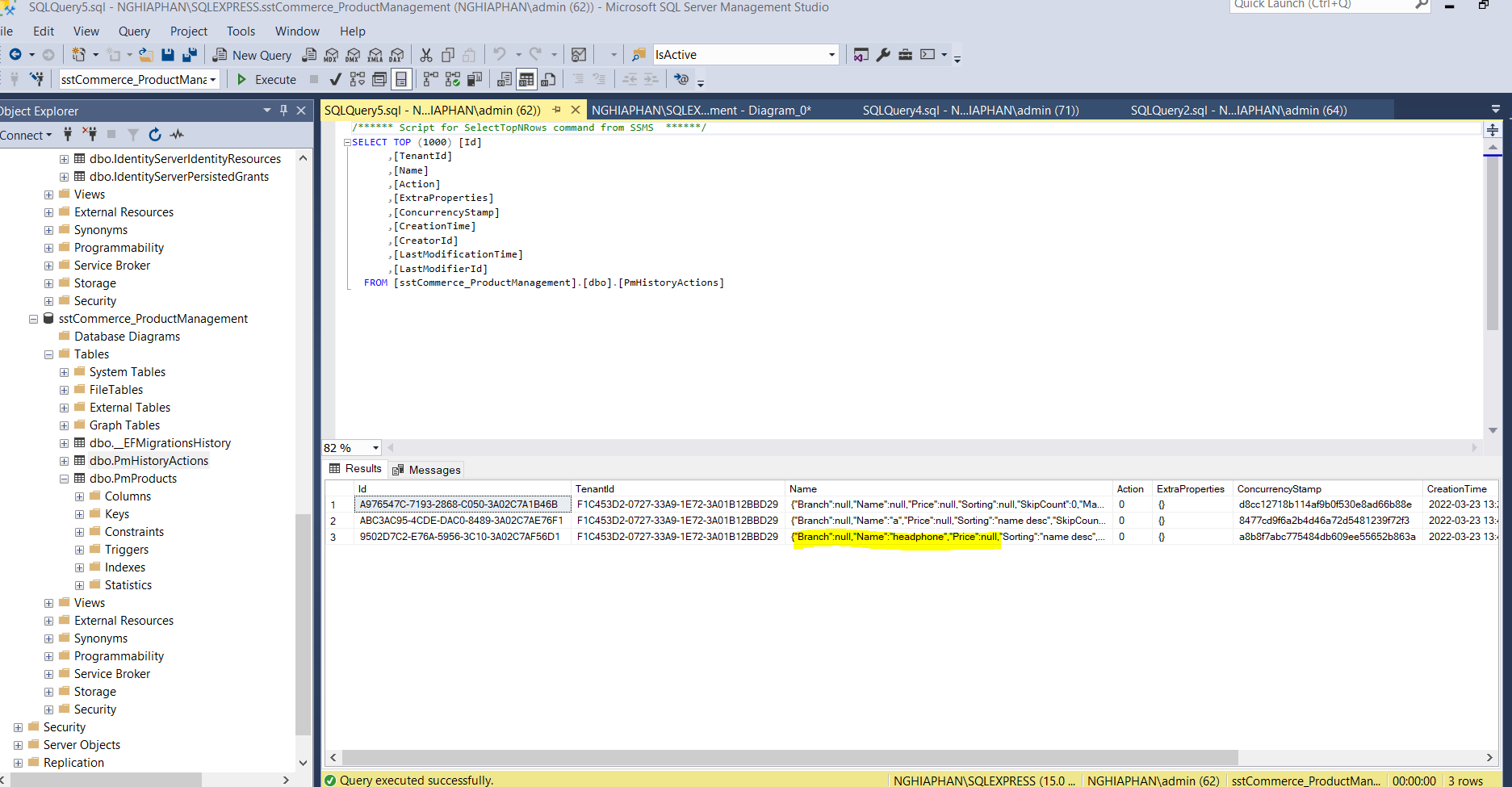


**[Get] – Products**: Get products based on parameters (Name, Branch, Price), Pagging & Sorting

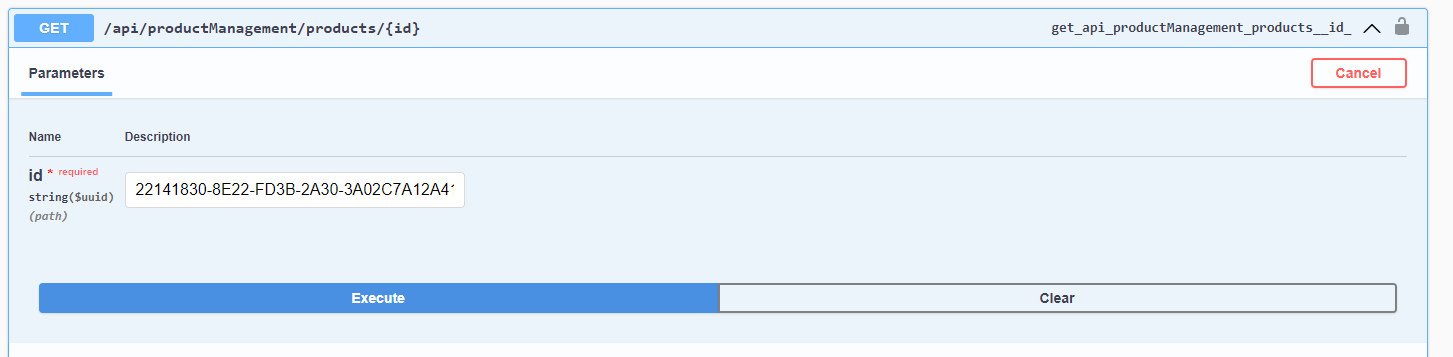




* Parameters(s) are stored in table **PmHistoryActions**



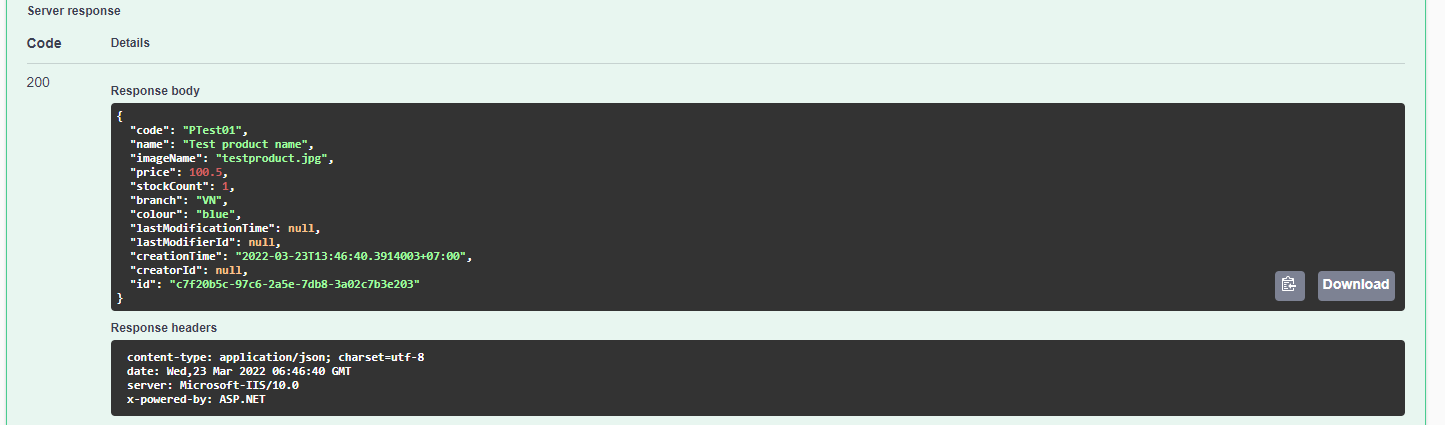
**[Get] – Product/{id}**: Get product based on id



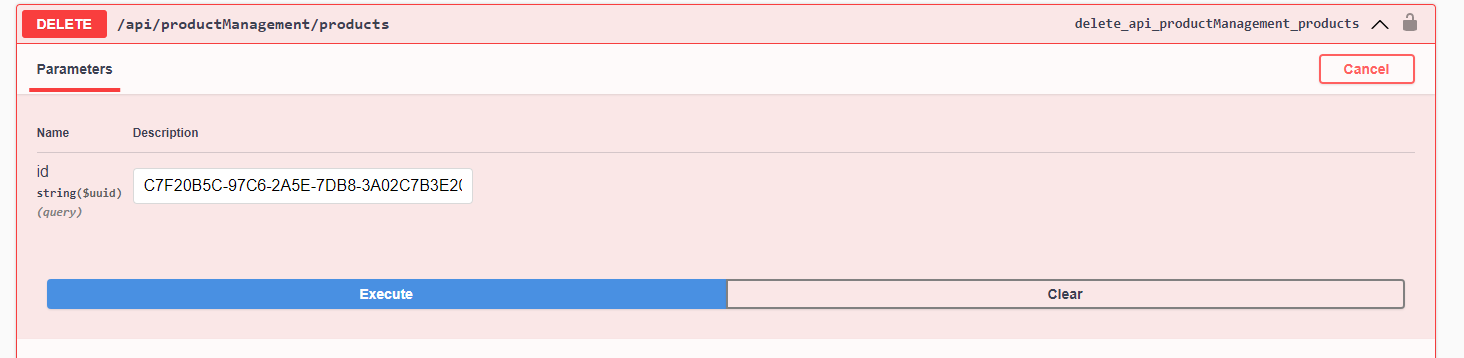


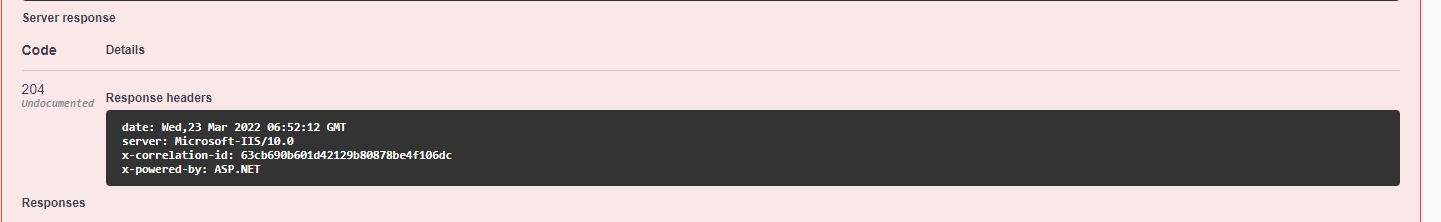
**[Post] – Products**: Create product with entry product DTO





**[Delete] – Products**: Delete a product with id

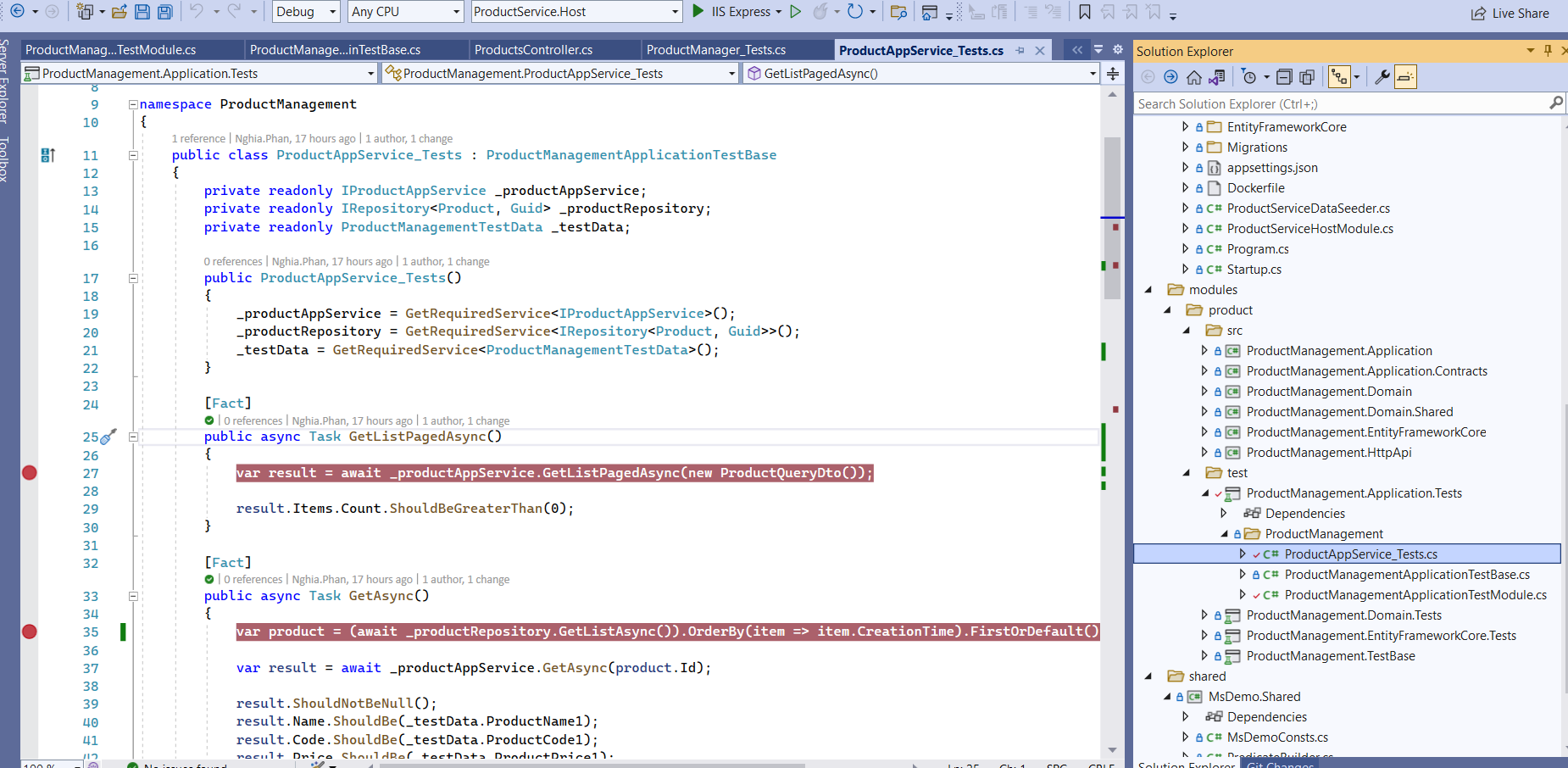


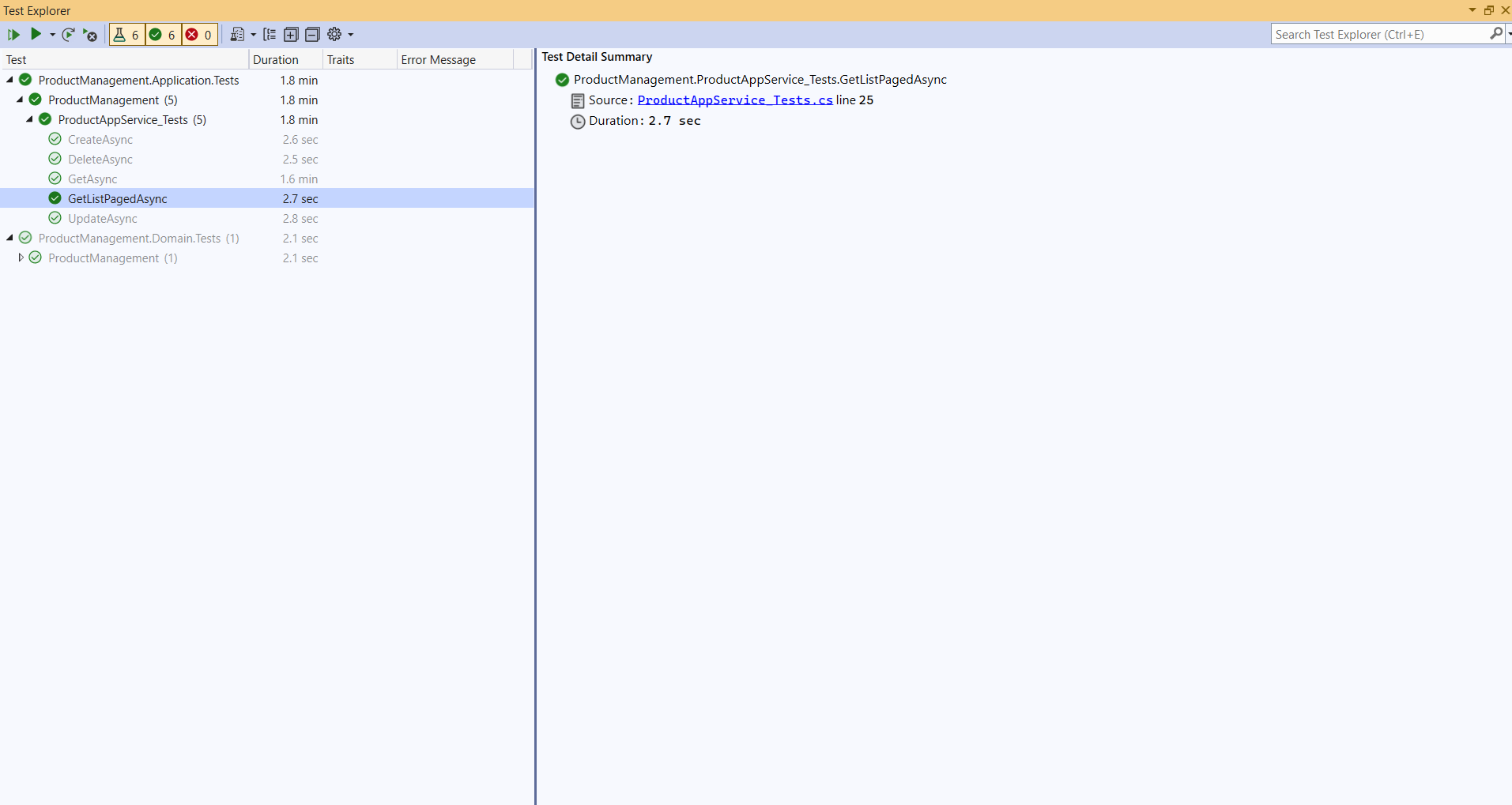


**[Put] – Products**: Update a product with id & product entry DTO

1. **Unit Test**

Using X-Unit for testing





Temporary DB in memory

