Fortnightly eFolio Submission – Studio 10

# Student Name and ID

Hao Xu (32767919)

# Self-Evaluation

High Distinction

# Task 10.1 (First Web API with .NET Core)

**GitHub Link**: <https://github.com/BrantleyXU/FIT5032/tree/main/week10/FIT5032_MyFirstAPI>

The following screenshot shows a successfully launched simple application integrated with web API as same as STEP 6 shown in the Week 10 tutorial specification. After executing GET request, the web browser could receive responses containing weather information.

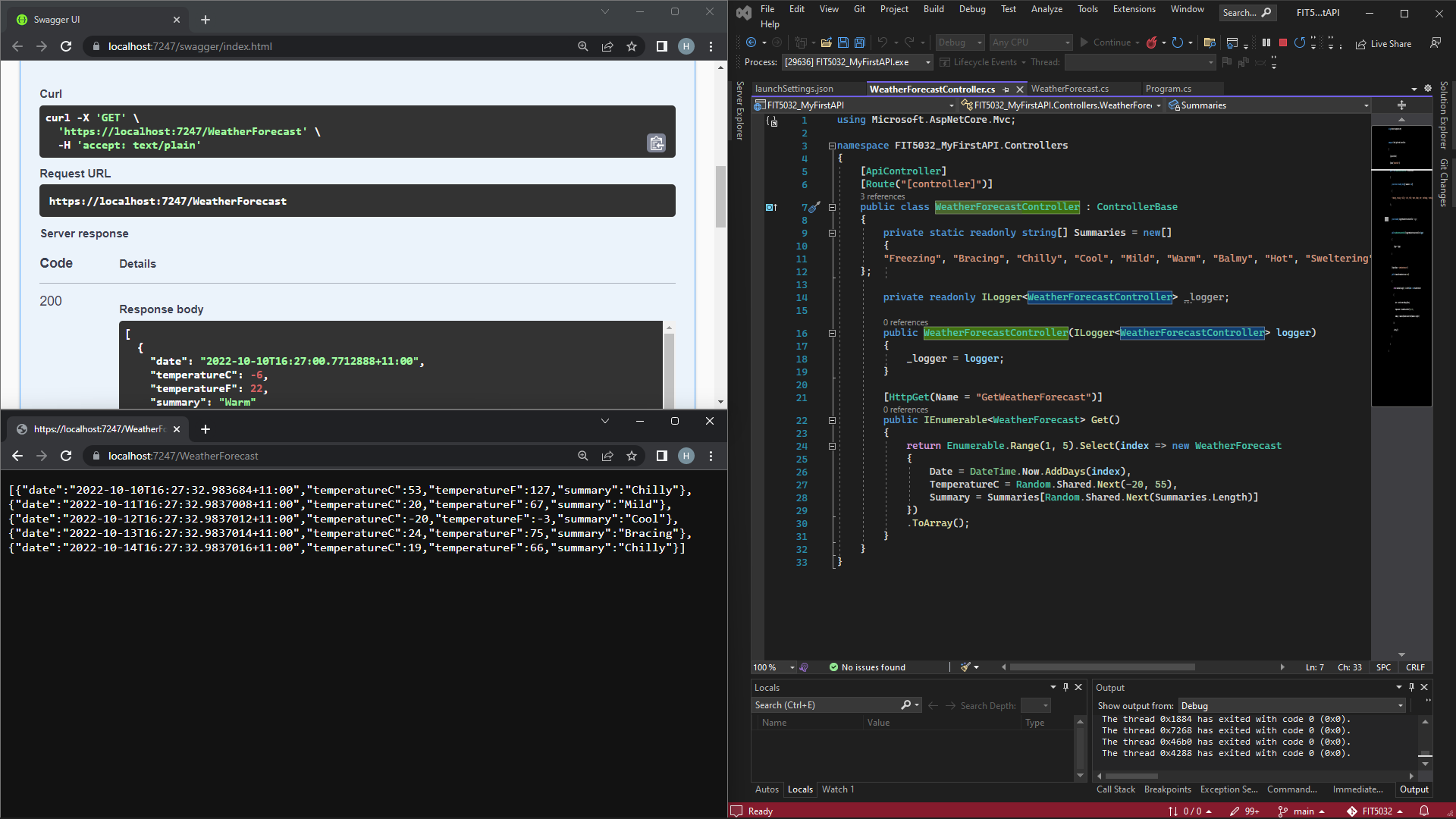


Figure 1: First Simple Web API

# Task 10.2 (Simple AngularJS Application)

**GitHub Link**: <https://github.com/BrantleyXU/FIT5032/tree/main/week10/FIT5032_SimpleAngularApp>

**CodePen Link**: <https://codepen.io/BrantleyXu/pen/wvjYdjO>

The following 2 screenshots show a successful simple AngularJS application integrated with .NET Core Web API. The steps and explanations are listed below.

1. At the browser side, I use HTML and JS code which are provided in the tutorial material. And I change the application URL to match my local host application URL.

2. At the server side (Visual Studio), I create end points to match the JS code:

2.1. I create a model called Unit.cs, with 3 attributes, Id, UnitCode and UnitName;

2.2. I create a “In-Memory” database called UnitContext.cs to store data. And I register the database in Program.cs;

2.3. I scaffold the related controller called UnitsController.cs;

2.4. I enable CORS (Cross-Origin Requests) in my application, in Program.cs file, so that my browser can talk to my server properly.

By doing so, I could successfully execute both JS and ASP.NET code, with no data yet (Figure 2). After I add a unit, the unit can be GET successfully (Figure 3), meaning that the application can work well.

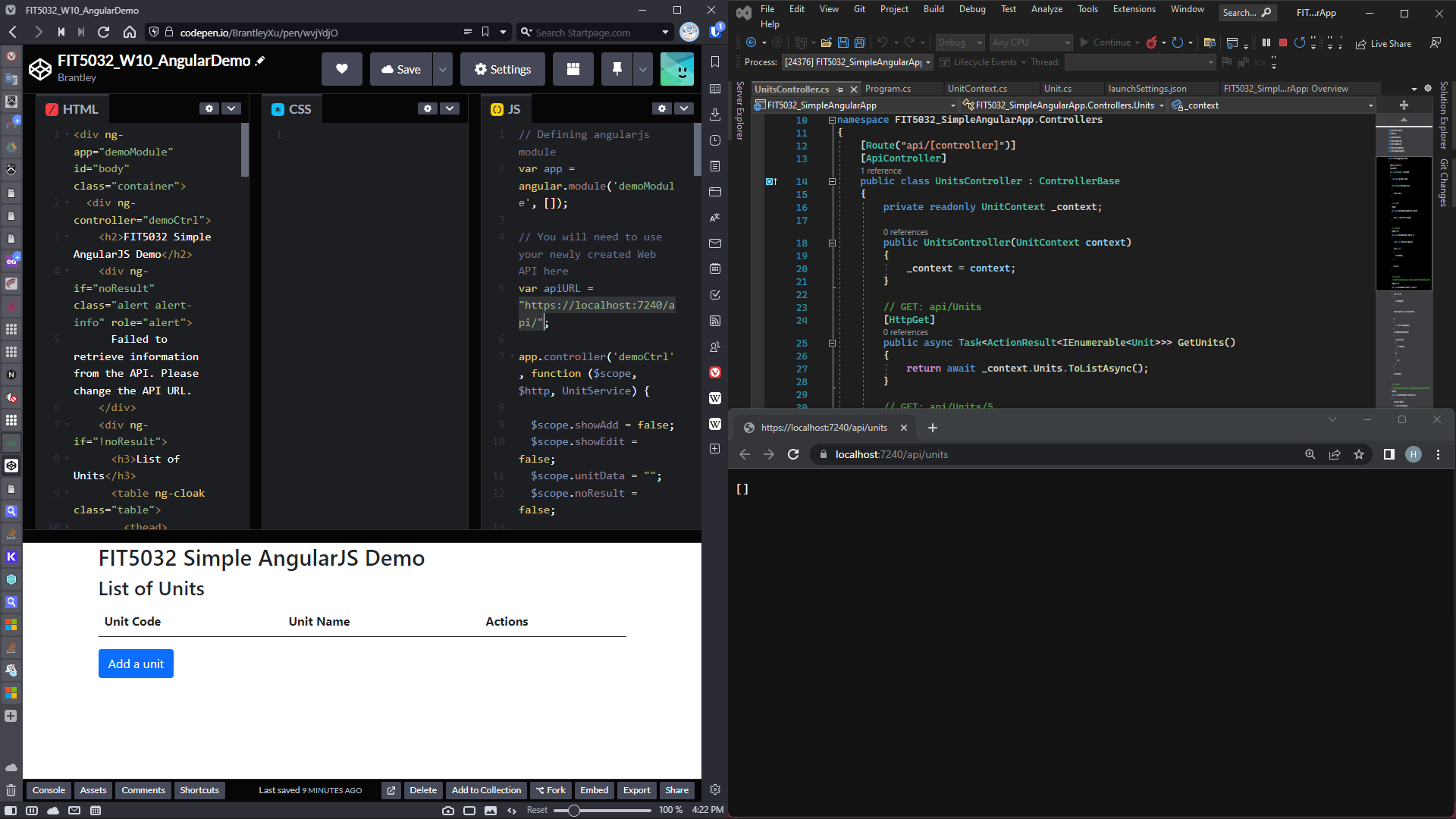


Figure 2: Initial State

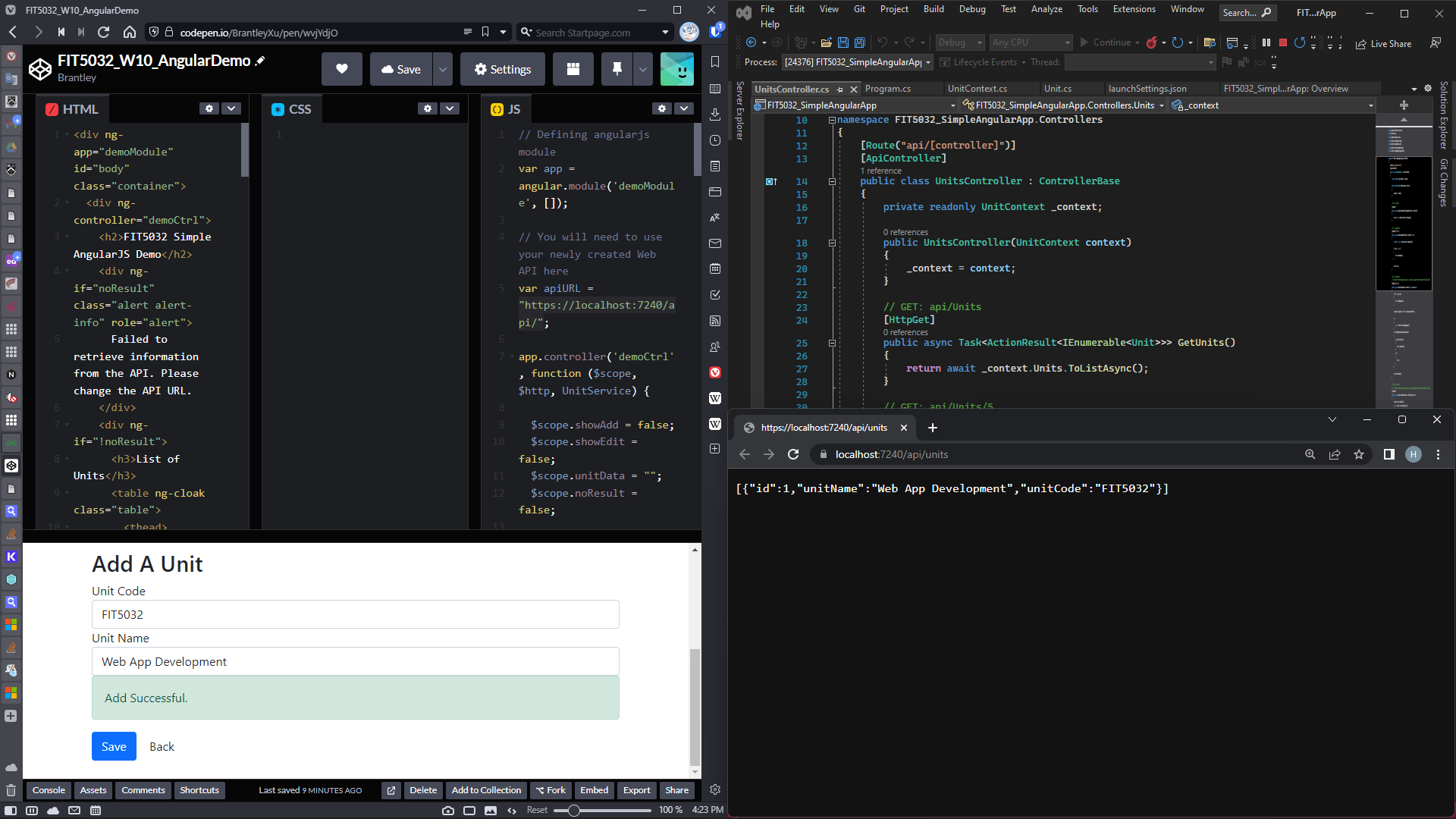


Figure 3: Add A Unit State

# References

George. (2021, July 12). CORS (3), Enable CORS In .NET Core Web API. *C# Corner*. <https://www.c-sharpcorner.com/article/enable-cors-consume-web-api-by-mvc-in-net-core-4/>

Microsoft. (2022, June 04). Enable Cross-Origin Requests (CORS) in ASP.NET Core. *Microsoft*. <https://learn.microsoft.com/en-us/aspnet/core/security/cors?view=aspnetcore-6.0>

Microsoft. (2022, October 06). Tutorial: Create a web API with ASP.NET Core. *Microsoft*. <https://learn.microsoft.com/en-us/aspnet/core/tutorials/first-web-api?view=aspnetcore-6.0&tabs=visual-studio>