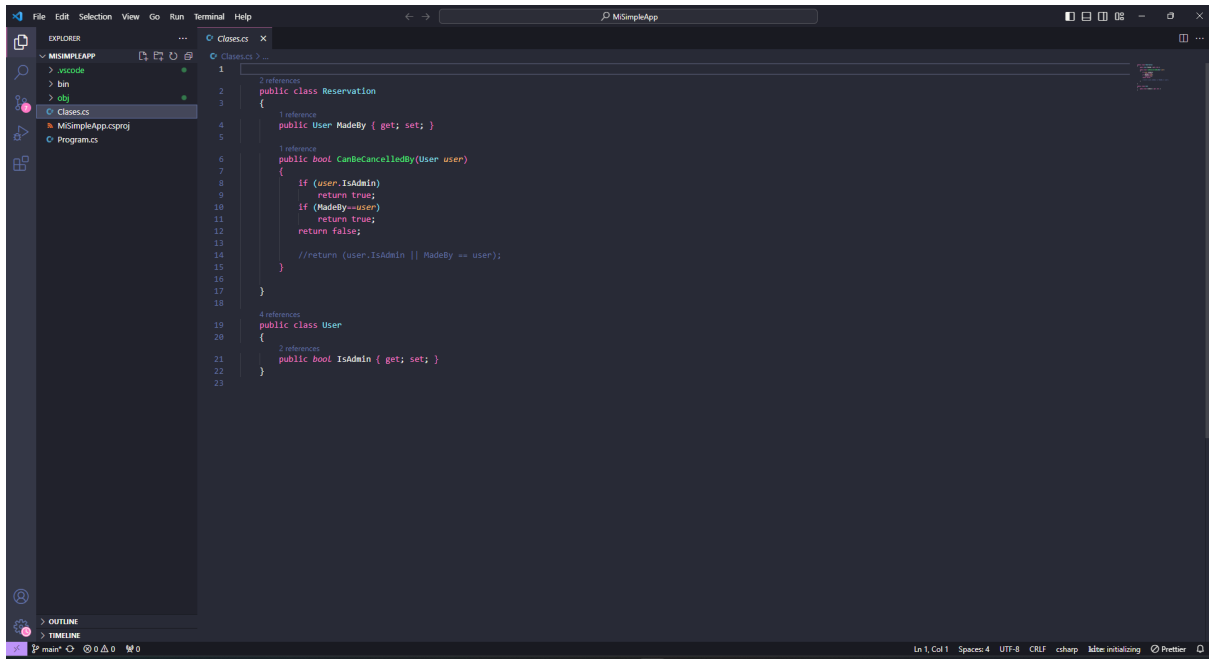


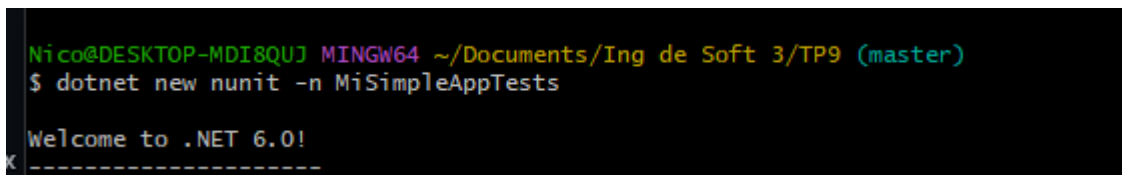
Ingeniera de Software 3

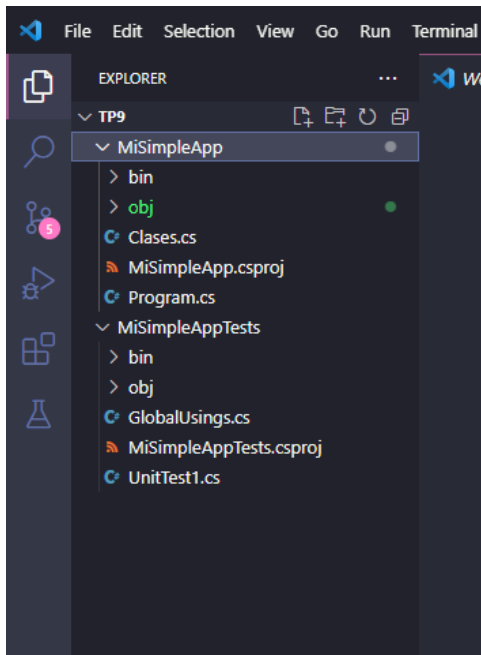
TP 9: Pruebas unitarias

1- Clonamos el codigo al directorio y lo abrimos con VS Code



2- Instalamos NUnit e instalamos los paquetes

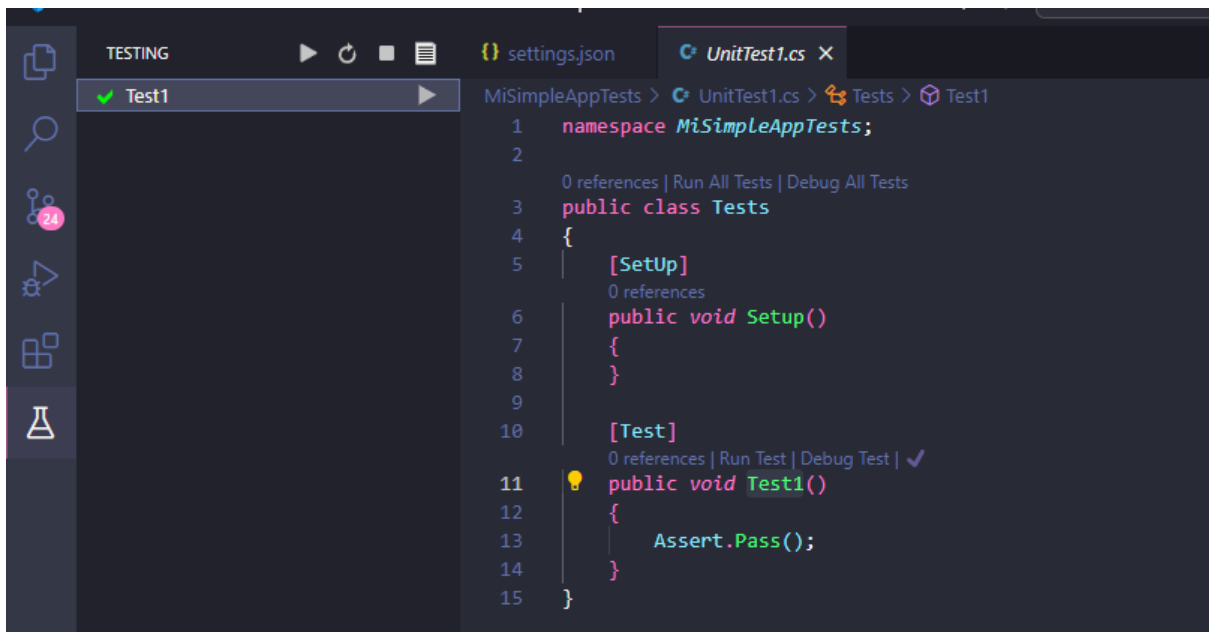




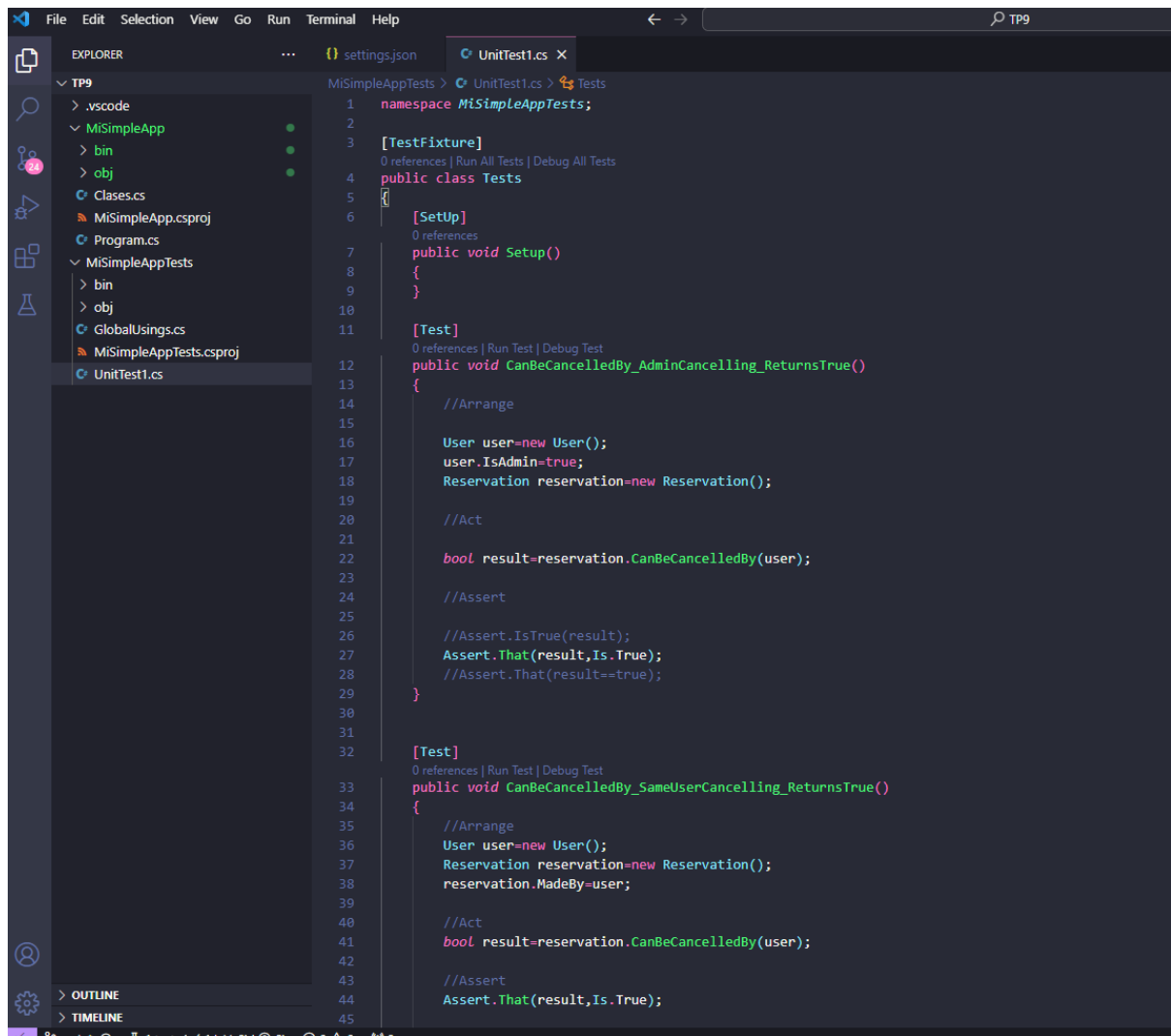
3- Modificamos el .json



4- Ahora podemos correr el test

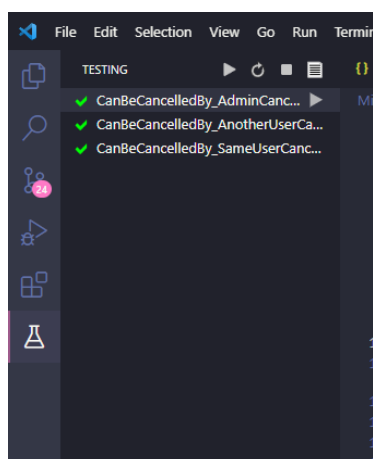


5- Ahora modificamos el archivo UnitTest1.cs con el código en la página



```
1 namespace MiSimpleAppTests;
2
3 [TestFixture]
4 public class Tests
5 {
6     [SetUp]
7     public void Setup()
8     {
9     }
10
11     [Test]
12     public void CanBeCancelledBy_AdminCancelling_ReturnsTrue()
13     {
14         //Arrange
15
16         User user=new User();
17         user.IsAdmin=true;
18         Reservation reservation=new Reservation();
19
20         //Act
21
22         bool result=reservation.CanBeCancelledBy(user);
23
24         //Assert
25
26         //Assert.IsTrue(result);
27         Assert.That(result,Is.True);
28         //Assert.That(result==true);
29     }
30
31     [Test]
32     public void CanBeCancelledBy_SameUserCancelling_ReturnsTrue()
33     {
34         //Arrange
35
36         User user=new User();
37         Reservation reservation=new Reservation();
38         reservation.MadeBy=user;
39
40         //Act
41
42         bool result=reservation.CanBeCancelledBy(user);
43
44         //Assert
45         Assert.That(result,Is.True);
```

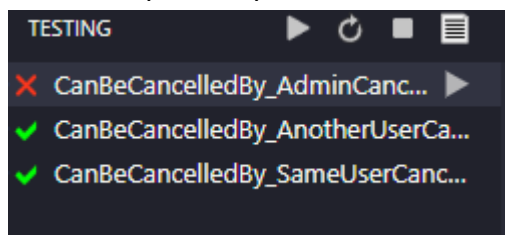
6- Ejecutamos los test:



7- Modificamos el archivo Clase.cs y cambiamos de true a false

```
1  
2 8 references  
3 public class Reservation  
4 {  
5     3 references  
6     public User MadeBy { get; set; }  
7  
8     4 references  
9     public bool CanBeCancelledBy(User user)  
10 {  
11     if (user.IsAdmin)  
12         return false;  
13     if (MadeBy==user)  
14         return true;  
15     return false;  
16  
17     //return (user.IsAdmin || MadeBy == user);  
18 }  
19  
20 12 references  
21 public class User  
22 {  
23     3 references  
24     public bool IsAdmin { get; set; }  
25 }  
26  
27 }
```

Y ahora la primera prueba da error



8- Modificamos nueva mente el archivo comentando y descomentando algunas partes

The screenshot shows the Visual Studio IDE with the following tabs: settings.json, UnitTest1.cs, and Clases.cs 1, M X. The file explorer on the left shows the path: MiSimpleApp > Clases.cs > Reservation > CanBeCancelledBy. The code editor displays the following C# code:

```
1
2     8 references
3     public class Reservation
4     {
5         3 references
6         public User MadeBy { get; set; }
7
8         4 references
9         public bool CanBeCancelledBy(User user)
10        {
11            // if (user.IsAdmin)
12            //     return true;
13            // if (MadeBy==user)
14            //     return true;
15            // return false;
16
17            return (user.IsAdmin || MadeBy == user);
18        }
19    }
20
21    12 references
22    public class User
23    {
24        3 references
25        public bool IsAdmin { get; set; }
26    }
```

9- Ahora corremos el test desde la linea de comando con “dotnet test”

The screenshot shows a terminal window with the following output:

```
$ dotnet test
Determining projects to restore...
  Restored C:\Users\Nico\Documents\Ing de Soft 3\TP9\MiSimpleApp\MiSimpleApp.csproj (in 132 ms).
  Restored C:\Users\Nico\Documents\Ing de Soft 3\TP9\MiSimpleAppTests\MiSimpleAppTests.csproj (in 262 ms).
C:\Users\Nico\Documents\Ing de Soft 3\TP9\MiSimpleApp\Clases.cs(4,21): warning CS8618: Non-nullable property 'MadeBy' must contain a non-null value when exiting constructor. Consider declaring the property as nullable. [C:\Users\Nico\Documents\Ing de Soft 3\TP9\MiSimpleApp\MiSimpleApp.csproj]
MiSimpleApp -> C:\Users\Nico\Documents\Ing de Soft 3\TP9\MiSimpleApp\bin\Debug\net7.0\MiSimpleApp.dll
MiSimpleAppTests -> C:\Users\Nico\Documents\Ing de Soft 3\TP9\MiSimpleAppTests\bin\Debug\net7.0\MiSimpleAppTests.dll
Test run for C:\Users\Nico\Documents\Ing de Soft 3\TP9\MiSimpleAppTests\bin\Debug\net7.0\MiSimpleAppTests.dll (.NETCoreApp,Version=v7.0)
Microsoft (R) Test Execution Command Line Tool Version 17.7.1 (x64)
Copyright (c) Microsoft Corporation. All rights reserved.

Starting test execution, please wait...
A total of 1 test files matched the specified pattern.

Passed! - Failed:    0, Passed:    3, Skipped:    0, Total:    3, Duration: 33 ms - MiSimpleAppTests.dll (net7.0)
```

Parte 6:

6.1- Clonamos MySimpleWebApi y creamos el proyecto de pruebas:

```
Nico@DESKTOP-MDI8QUJ MINGW64 ~/Documents/Ing de Soft 3 (master)
$ mkdir TP9_1

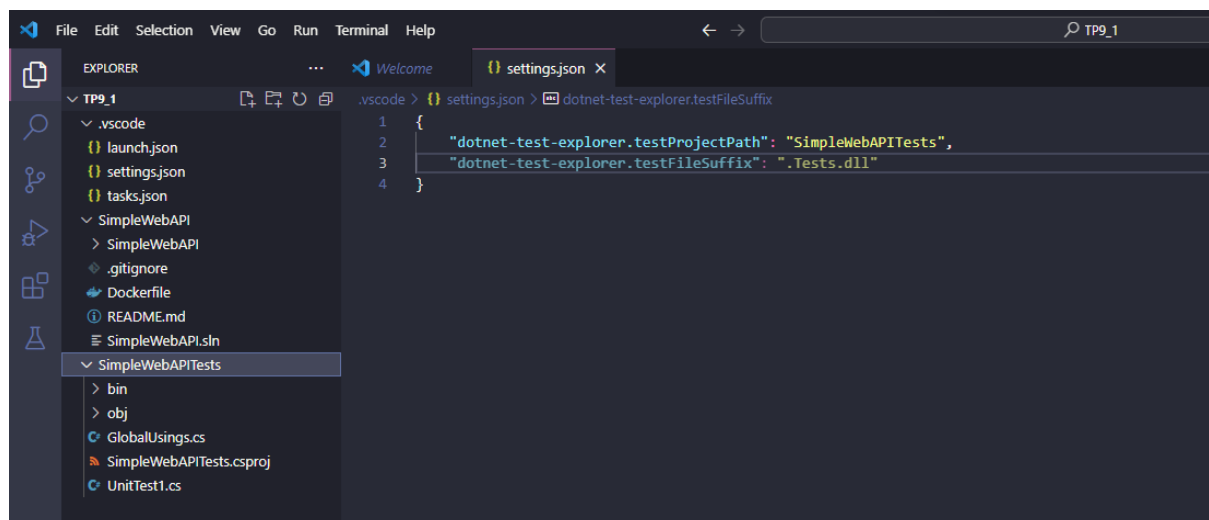
Nico@DESKTOP-MDI8QUJ MINGW64 ~/Documents/Ing de Soft 3 (master)
$ git clone https://github.com/ingsoft3ucc/SimpleWebAPI.git
fatal: destination path 'SimpleWebAPI' already exists and is not an empty directory.

Nico@DESKTOP-MDI8QUJ MINGW64 ~/Documents/Ing de Soft 3 (master)
$ dotnet new nunit -n SimpleWebAPITests
The template "NUnit 3 Test Project" was created successfully.

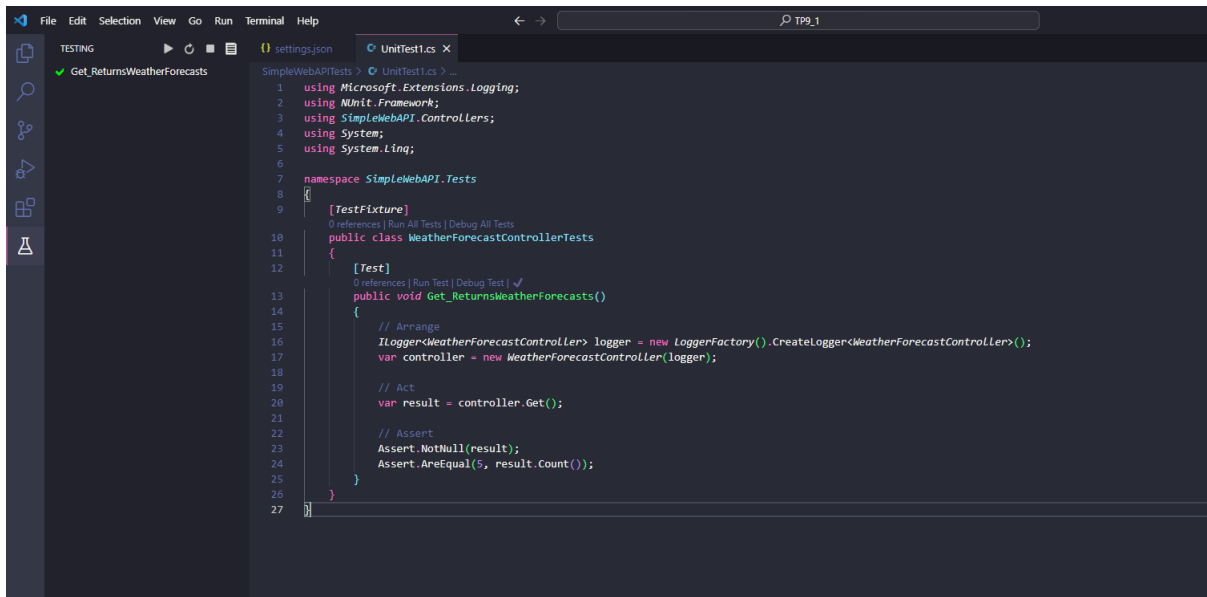
Processing post-creation actions...
Restoring C:\Users\Nico\Documents\Ing de Soft 3\SimpleWebAPITests\SimpleWebAPITests.csproj
j:
  Determining projects to restore...
  Restored C:\Users\Nico\Documents\Ing de Soft 3\SimpleWebAPITests\SimpleWebAPITests.csproj (in 317 ms).
Restore succeeded.

Nico@DESKTOP-MDI8QUJ MINGW64 ~/Documents/Ing de Soft 3 (master)
$ cd SimpleWebAPITests/

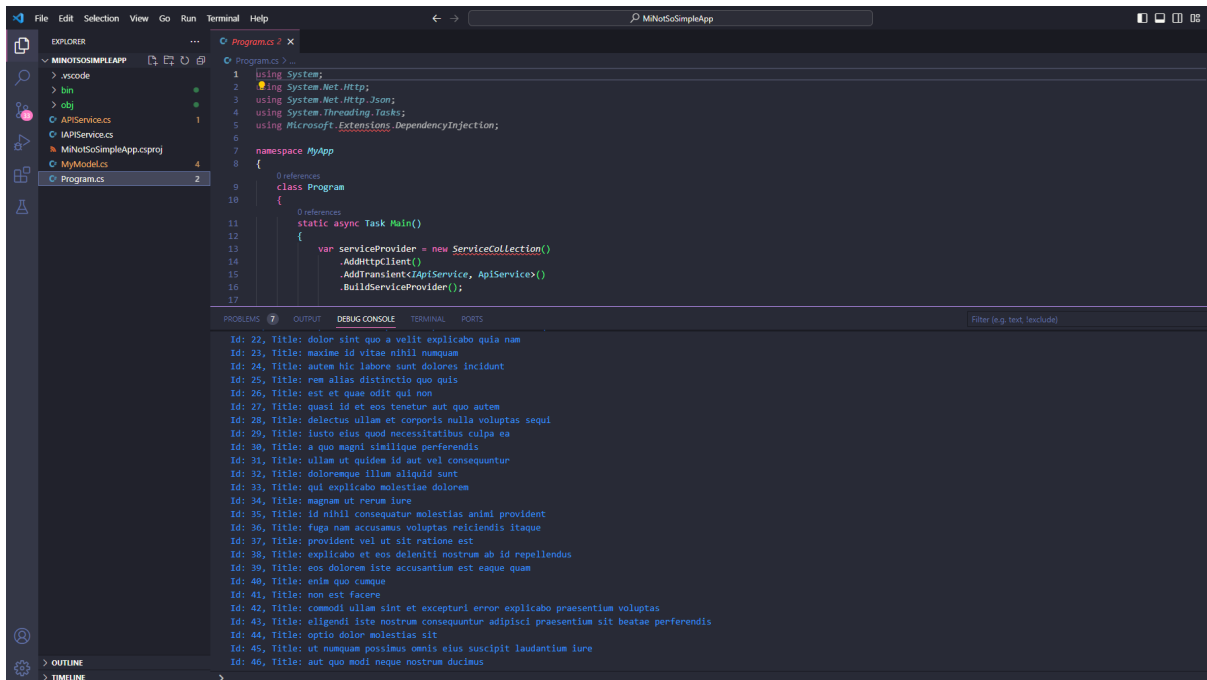
Nico@DESKTOP-MDI8QUJ MINGW64 ~/Documents/Ing de Soft 3/SimpleWebAPITests (master)
$ dotnet add reference ../SimpleWebAPI/SimpleWebAPI/SimpleWebAPI.csproj
Reference '..\SimpleWebAPI\SimpleWebAPI\SimpleWebAPI.csproj' added to the project.
```



6.2- Modificamos el código y corremos el test



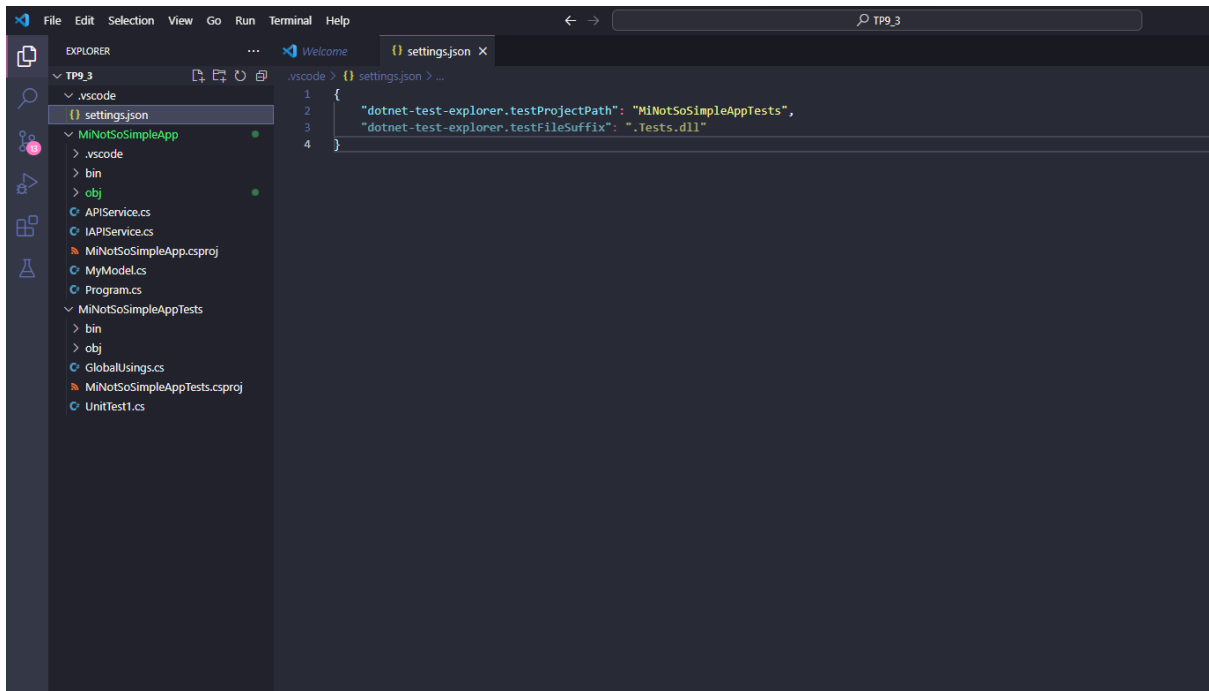
8.1- Clonamos la repo y ejecutamos la API



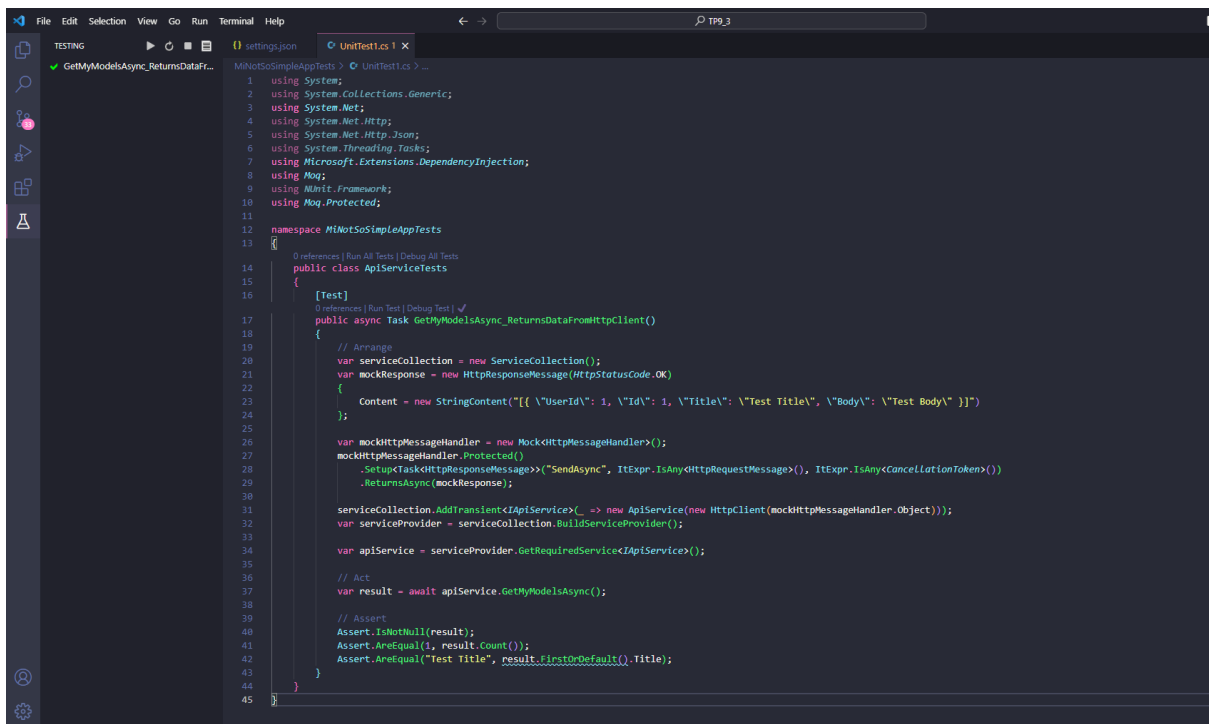
Primero creo un servicio que implementa el servicio IAPI Service y despues llamamos a una fuente de datos externa. Vamos a llenar la variable MyModels con los datos de los Posts y los va a ir escribiendo en consola.

El servicio externo de la API es: <https://jsonplaceholder.typicode.com/posts/>

Creamos el proyecto NUnit y configuramos.



8.4- Modificamos el codigo Test1 y corremos el Test:



Y por linea de comando:


```

Nico@DESKTOP-MDI8QUJ MINGW64 ~/Documents/Ing de Soft 3/TP9_3/MiNotSoSimpleAppTests (master)
$ dotnet test
Determining projects to restore...
  Restored C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleAppTests\MiNotSoSimpleAppTests.csproj (in 411 ms).
  Restored C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleApp\MiNotSoSimpleApp.csproj (in 411 ms).
C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleApp\MyModel.cs(8,19): warning CS8618: Non-nullable property 'Title' must contain a non-null value when exiting constructor. Consider declaring the property as nullable. [C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleApp\MiNotSoSimpleApp.csproj]
C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleApp\MyModel.cs(9,19): warning CS8618: Non-nullable property 'Body' must contain a non-null value when exiting constructor. Consider declaring the property as nullable. [C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleApp\MiNotSoSimpleApp.csproj]
C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleApp\APIService.cs(18,16): warning CS8603: Possible null reference return. [C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleApp\MiNotSoSimpleApp.csproj]
MiNotSoSimpleApp -> C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleApp\bin\Debug\net7.0\MiNotSoSimpleApp.dll
C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleAppTests\UnitTest1.cs(42,43): warning CS8602: Dereference of a possibly null reference. [C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleAppTests\MiNotSoSimpleAppTests.csproj]
C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleAppTests\UnitTest1.cs(41,13): warning NUnit2005: Consider using the constraint model, Assert.That(actual, Is.EqualTo(expected)), instead of the classic model, Assert.AreEqual(expected, actual) (https://github.com/nunit/nunit.analyzers/tree/master/documentation/NUnit2005.md) [C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleAppTests\MiNotSoSimpleAppTests.csproj]
C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleAppTests\UnitTest1.cs(42,13): warning NUnit2005: Consider using the constraint model, Assert.That(actual, Is.EqualTo(expected)), instead of the classic model, Assert.AreEqual(expected, actual) (https://github.com/nunit/nunit.analyzers/tree/master/documentation/NUnit2005.md) [C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleAppTests\MiNotSoSimpleAppTests.csproj]
MiNotSoSimpleAppTests -> C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleAppTests\bin\Debug\net7.0\MiNotSoSimpleAppTests.dll
Test run for C:\Users\Nico\Documents\Ing de Soft 3\TP9_3\MiNotSoSimpleAppTests\bin\Debug\net7.0\MiNotSoSimpleAppTests.dll (.NETCoreApp,Version=v7.0)
Microsoft (R) Test Execution Command Line Tool Version 17.7.1 (x64)
Copyright (c) Microsoft Corporation. All rights reserved.

Starting test execution, please wait...
A total of 1 test files matched the specified pattern.

Passed! - Failed:    0, Passed:    1, Skipped:    0, Total:    1, Duration: 184 ms - MiNotSoSimpleAppTests.dll (net7.0)

```

8-5: Ahora lo hago fallar modificando el valor de Assert.AreEqual = 1 a 2

```

10 using Xunit;
11
12 namespace MiNotSoSimpleAppTests
13 {
14     [RunAllTests | Debug All Tests]
15     public class ApiServiceTests
16     {
17         [Test]
18         public async Task GetMyModelsAsync_ReturnsDataFromHttpClient()
19         {
20             // Arrange
21             var serviceCollection = new ServiceCollection();
22             var mockResponse = new HttpResponseMessage(HttpStatusCode.OK)
23             {
24                 Content = new StringContent("[{ \"UserId\": 1, \"Id\": 1, \"Title\": \"Test Title\", \"Body\": \"Test Body\" }]")
25             };
26
27             var mockHttpMessageHandler = new MockHttpMessageHandler();
28             mockHttpMessageHandler.Protected()
29                 .Setup

```

