

UNIVERSIDAD PRIVADA DE TACNA



INGENIERIA DE SISTEMAS

TITULO:

Informe de Laboratorio 05: Utilizando Desarrollo guiado por comportamiento (BDD) para realización pruebas de software

CURSO:

Calidad y Pruebas de Software

DOCENTE:

Ing. Patrick Cuadros Quiroga

INTEGRANTES:

Yaneth Virginia Aquino Huallpa

(2017059286)

Tacna

1. Parte 1:Procedimiento

- Se utilizará el framework CoreBDD. Instalar la plantilla de proyecto dotnet a través de dotnet new -i corebdd.projecttemplate

```
C:\Users\Virginia Aquino>dotnet new -i corebdd.projecttemplate
Determinando los proyectos que se van a restaurar...
Se ha restaurado C:\Users\Virginia Aquino\templateengine\dotnetcli\v5.0.100-preview.6.20318.15\scratch\restore.csproj (en 6,77 sec).
```

Templates	Short Name	Language	Tags
Console Application	console	[C#], F#, VB	Common/Console
Class library	classlib	[C#], F#, VB	Common/Library
WPF Application	wpf	[C#], VB	Common/WPF
WPF Class library	wpflib	[C#], VB	Common/WPF
WPF Custom Control Library	wpfcustomcontrollib	[C#], VB	Common/WPF
WPF User Control Library	wpfusercontrollib	[C#], VB	Common/WPF
Windows Forms (WinForms) Application	winforms	[C#], VB	Common/WinForms
Windows Forms (WinForms) Class library	winformslib	[C#], VB	Common/WinForms
Worker Service	worker	[C#]	Common/Worker/Web
CoreBDD Test Project Template	corebdd	[C#]	CoreBDD/xUnit/BDD/TDD
Unit Test Project	mstest	[C#], F#, VB	Test/MSTest
NUnit 3 Test Project	nunit	[C#], F#, VB	Test/NUnit
NUnit 3 Test Item	nunit-test	[C#], F#, VB	Test/NUnit
xUnit Test Project	xunit	[C#], F#, VB	Test/xUnit
Razor Component	razorcomponent	[C#]	Web/ASP.NET
Razor Page	page	[C#]	Web/ASP.NET
MVC ViewImports	viewimports	[C#]	Web/ASP.NET
MVC ViewStart	viewstart	[C#]	Web/ASP.NET
Blazor Server App	blazorserver	[C#]	Web/Blazor
Blazor WebAssembly App	blazorwasm	[C#]	Web/Blazor/WebAssembly
ASP.NET Core Empty	web	[C#], F#	Web/Empty
ASP.NET Core Web App (Model-View-Controller)	mvc	[C#], F#	Web/MVC
ASP.NET Core Web App	webapp	[C#]	Web/MVC/Razor Pages
ASP.NET Core with Angular	angular	[C#]	Web/MVC/SPA
ASP.NET Core with React.js	react	[C#]	Web/MVC/SPA
ASP.NET Core with React.js and Redux	reactredux	[C#]	Web/MVC/SPA
Razor Class Library	razorclasslib	[C#]	Web/Razor/Library
ASP.NET Core Web API	webapi	[C#], F#	Web/WebAPI
ASP.NET Core gRPC Service	grpc	[C#]	Web/gRPC
dotnet gitignore file	gitignore		Config
global.json file	globaljson		Config
NuGet Config	nugetconfig		Config
Dotnet local tool manifest file	tool-manifest		Config
Web Config	webconfig		Config
Solution File	sln		Solution
Protocol Buffer File	proto		Web/gRPC

- Luego cree una nueva carpeta para su proyecto de prueba y ejecute dotnet new corebdd
- Alternativamente, puede agregar CoreBDD a un proyecto de prueba xUnit existente a través del paquete nuget dotnet add package CoreBDD

```
C:\Users\Virginia Aquino>dotnet new corebdd
The template "CoreBDD Test Project Template" was created successfully.

C:\Users\Virginia Aquino>dotnet add package CoreBDD
Determinando los proyectos que se van a restaurar...
Writing C:\Users\Virginia Aquino\AppData\Local\Temp\tmpC347.tmp
info : Agregando PackageReference para el paquete "CoreBDD" al proyecto "C:\Users\Virginia Aquino\Virginia Aquino.csproj".
info : Restaurando paquetes para C:\Users\Virginia Aquino\Virginia Aquino.csproj...
info : GET https://api.nuget.org/v3-flatcontainer/corebdd/index.json
info : GET https://api.nuget.org/v3-flatcontainer/corebdd.specgeneration/index.json
info : GET https://api.nuget.org/v3-flatcontainer/microsoft.net.test.sdk/index.json
info : OK https://api.nuget.org/v3-flatcontainer/microsoft.net.test.sdk/index.json 511 ms
info : GET https://api.nuget.org/v3-flatcontainer/microsoft.net.test.sdk/15.7.0/microsoft.net.test.sdk.15.7.0.nupkg
info : OK https://api.nuget.org/v3-flatcontainer/microsoft.net.test.sdk/15.7.0/microsoft.net.test.sdk.15.7.0.nupkg 35 ms
info : OK https://api.nuget.org/v3-flatcontainer/corebdd/index.json 1511 ms
info : OK https://api.nuget.org/v3-flatcontainer/corebdd.specgeneration/index.json 1051 ms
info : GET https://api.nuget.org/v3-flatcontainer/corebdd.specgeneration/1.0.0/corebdd.specgeneration.1.0.0.nupkg
info : GET https://api.nuget.org/v3-flatcontainer/corebdd/1.0.0/corebdd.1.0.0.nupkg
info : GET https://api.nuget.org/v3-flatcontainer/microsoft.testplatform.testhost/index.json
info : GET https://api.nuget.org/v3-flatcontainer/microsoft.codecoverage/index.json
info : OK https://api.nuget.org/v3-flatcontainer/corebdd.specgeneration/1.0.0/corebdd.specgeneration.1.0.0.nupkg 587 ms
info : GET https://api.nuget.org/v3-flatcontainer/icsharpcode.decompiler/index.json
info : OK https://api.nuget.org/v3-flatcontainer/corebdd/1.0.0/corebdd.1.0.0.nupkg 616 ms
info : OK https://api.nuget.org/v3-flatcontainer/icsharpcode.decompiler/index.json 177 ms
info : GET https://api.nuget.org/v3-flatcontainer/icsharpcode.decompiler/3.2.0.3856/icsharpcode.decompiler.3.2.0.3856.nupkg
info : GET https://api.nuget.org/v3-flatcontainer/microsoft.codeanalysis.csharp/index.json
```

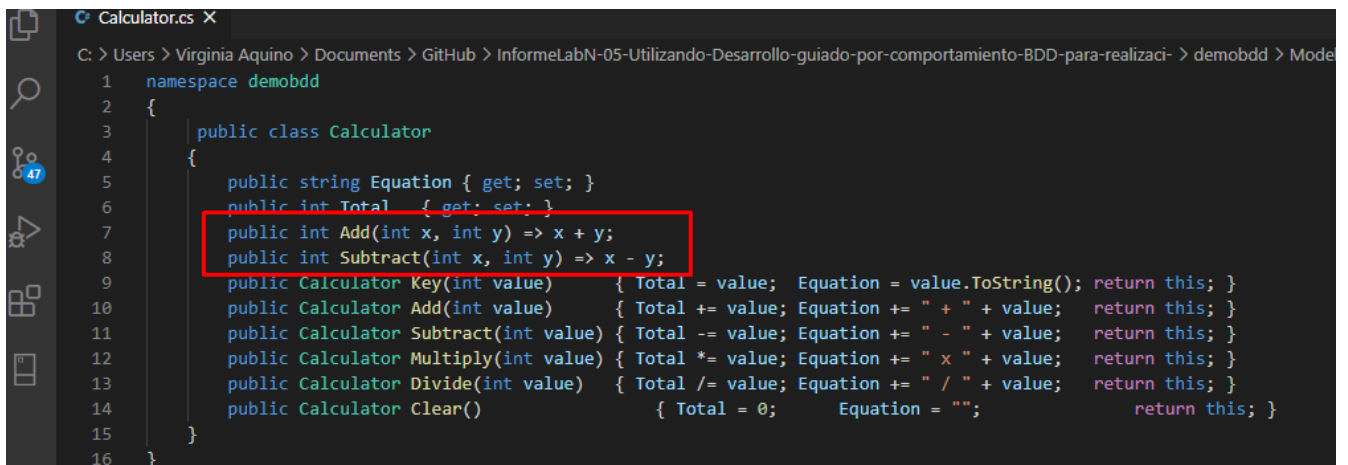
- También hay una herramienta de línea de comandos opcional para ejecutar pruebas con salida personalizada, clases de prueba de andamios (características / escenarios / pasos) y

generación de código bidireccional (pruebas de Gherkin a CoreBDD y viceversa). Hay más documentación disponible sobre la CLI en la parte inferior de esta página. `dotnet tool install -g corebdd.commandline`

```
C:\Users\Virginia Aquino>dotnet tool install -g corebdd.commandline
Puede invocar la herramienta con el comando siguiente: corebdd
La herramienta "corebdd.commandline" (versión '1.0.0') se instaló correctamente.
```

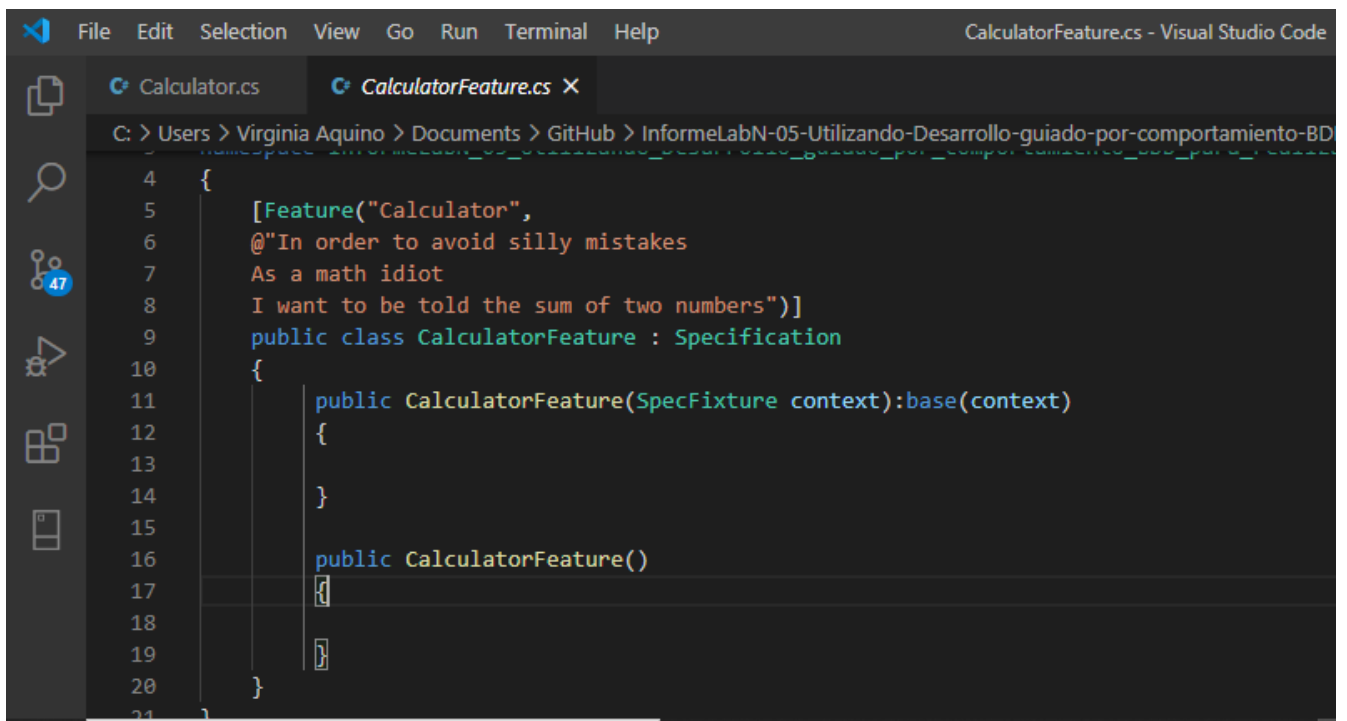
2. Escribir pruebas de CoreBDD

- Siguiendo el ejemplo habitual de la calculadora, podemos comenzar con el siguiente modelo para probar



```
Calculator.cs X
C: > Users > Virginia Aquino > Documents > GitHub > InformeLabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci- > demobdd > Mode
1 namespace demobdd
2 {
3     public class Calculator
4     {
5         public string Equation { get; set; }
6         public int Total { get; set; }
7         public int Add(int x, int y) => x + y;
8         public int Subtract(int x, int y) => x - y;
9         public Calculator Key(int value) { Total = value; Equation = value.ToString(); return this; }
10        public Calculator Add(int value) { Total += value; Equation += " + " + value; return this; }
11        public Calculator Subtract(int value) { Total -= value; Equation += " - " + value; return this; }
12        public Calculator Multiply(int value) { Total *= value; Equation += " x " + value; return this; }
13        public Calculator Divide(int value) { Total /= value; Equation += " / " + value; return this; }
14        public Calculator Clear() { Total = 0; Equation = ""; return this; }
15    }
16 }
```

- Podemos definir una Característica para cotejar un conjunto de escenarios derivando de la clase base Especificación y decorando con el atributo Característica. Tenga en cuenta que ambos constructores son necesarios para admitir los diferentes estilos de sintaxis de prueba.



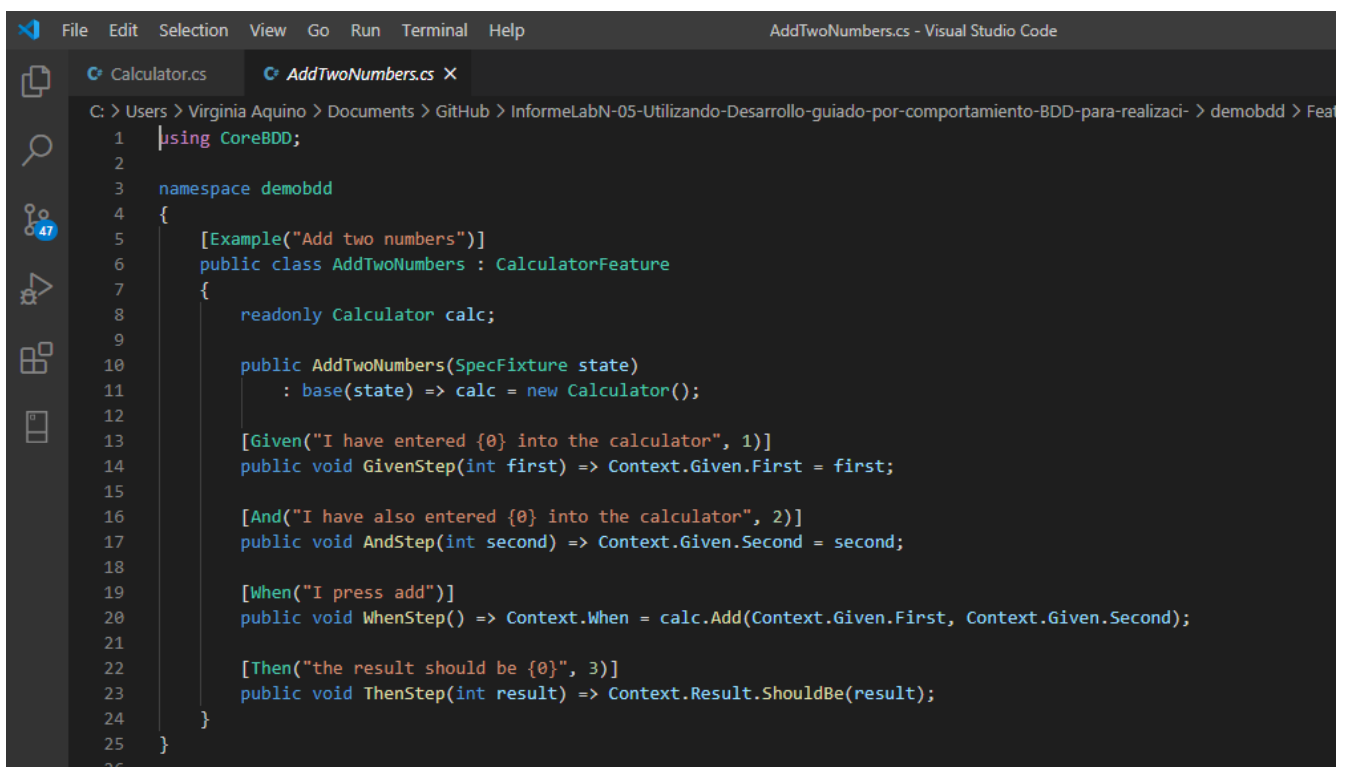
```
File Edit Selection View Go Run Terminal Help CalculatorFeature.cs - Visual Studio Code

Calculator.cs CalculatorFeature.cs X

C: > Users > Virginia Aquino > Documents > GitHub > InformeLabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BD

4 {
5     [Feature("Calculator",
6         @"In order to avoid silly mistakes
7         As a math idiot
8         I want to be told the sum of two numbers")]
9     public class CalculatorFeature : Specification
10    {
11        public CalculatorFeature(SpecFixture context):base(context)
12        {
13        }
14    }
15
16    public CalculatorFeature()
17    {
18    }
19
20 }
21 }
```

- Una vez que hemos creado nuestra Característica base, tenemos varios sabores diferentes para escribir pruebas, primero podemos generar un escenario por clase (similar a las pruebas de estilo Cucumber) con un método para cada paso de Dado / Cuándo / Entonces. Para hacer esto, simplemente herede de la nueva clase de entidad, decore con un atributo de ejemplo y proporcione los métodos Given, When, Then que se ejecutarán en orden



```
File Edit Selection View Go Run Terminal Help AddTwoNumbers.cs - Visual Studio Code

Calculator.cs AddTwoNumbers.cs X

C: > Users > Virginia Aquino > Documents > GitHub > InformeLabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci- > demobdd > Fea

1 using CoreBDD;
2
3 namespace demobdd
4 {
5     [Example("Add two numbers")]
6     public class AddTwoNumbers : CalculatorFeature
7     {
8         readonly Calculator calc;
9
10        public AddTwoNumbers(SpecFixture state)
11        : base(state) => calc = new Calculator();
12
13        [Given("I have entered {0} into the calculator", 1)]
14        public void GivenStep(int first) => Context.Given.First = first;
15
16        [And("I have also entered {0} into the calculator", 2)]
17        public void AndStep(int second) => Context.Given.Second = second;
18
19        [When("I press add")]
20        public void WhenStep() => Context.When = calc.Add(Context.Given.First, Context.Given.Second);
21
22        [Then("the result should be {0}", 3)]
23        public void ThenStep(int result) => Context.Result.ShouldBe(result);
24    }
25 }
26 }
```

- También puede definir escenarios en un solo método utilizando delegados para cada uno de los pasos y permitiendo que se definan múltiples escenarios dentro de la misma clase

```

5 {
6     public class AdvancedCalculator : CalculatorFeature
7     {
8         Calculator calculator;
9
10        [Scenario("Multiply two numbers")]
11        public void MultiplyTwoNumbers()
12        {
13
14            Given("I have a calculator", () => calculator = new Calculator());
15            When("I key in 10", () => calculator.Key(10));
16            And("I key in 5 and press multiply", () => calculator.Multiply(5));
17            Then("It sets the Total to 50", () => calculator.Total.ShouldBe(50));
18            And("It sets the equation to 10 x 5", () => calculator.Equation.ShouldBe("10 x 5"));
19        }
20
21        [Scenario("Divide two numbers")]
22        public void DivideTwoNumbers()
23        {
24            Given("I have a calculator", () => calculator = new Calculator());
25            When("I key in 10", () => calculator.Key(10));
26            And("I key in 5 and press divide", () => calculator.Divide(5));
27            Then("It sets the Total to 2", () => calculator.Total.ShouldBe(2));
28        }
29    }

```

- La sintaxis basada en métodos también admite pruebas basadas en datos, utilizando xUnit InlineData (los escenarios basados en clases aún no admiten pruebas basadas en datos).

```

28 }
29 [ScenarioOutline("Divide two numbers")]
30 [Examples(10, 2, 5)]
31 [Examples(20, 4, 5)]
32 public void DivideTwoNumbers(int number, int divideby, int result)
33 {
34     Given($"I have a calculator", () => calculator = new Calculator());
35     When($"I key in {number}", () => calculator.Key(number));
36     And($"I key in {divideby} and press divide", () => calculator.Divide(divideby));
37     Then($"It sets the Total to {result}", () => calculator.Total.ShouldBe(result));
38     //And($"It sets the equation to {number} / {divideby}", () => calculator.Equation.ShouldBe($"{number} / {divideby}"));
39 }
40 }
41 }

```

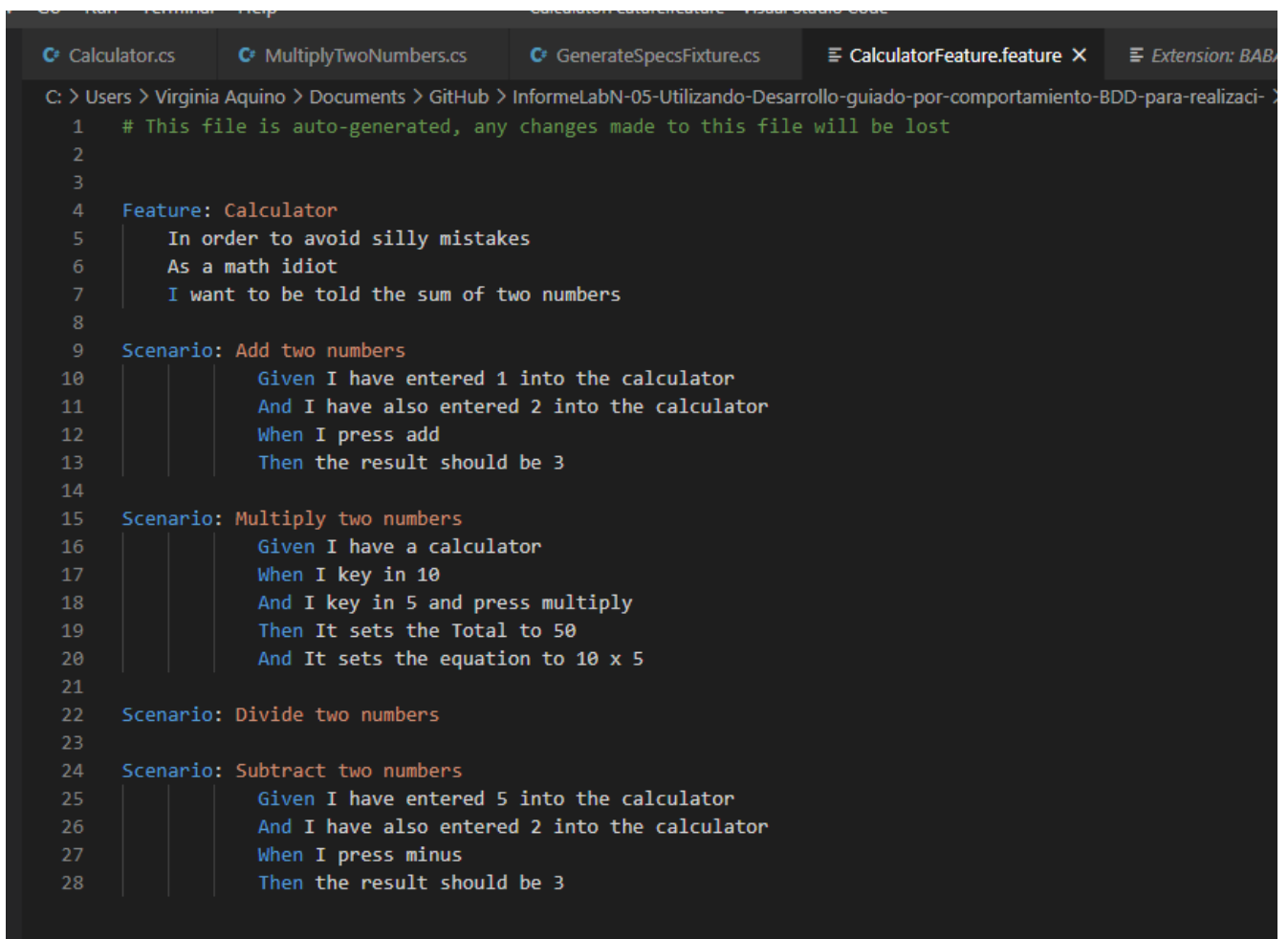
- Puede generar especificaciones de Gherkin de sus pruebas utilizando la biblioteca de extensiones CoreBDD.SpecGeneration, ya sea llamando desde una aplicación o herramienta de línea de comandos y pasando la ruta al ensamblado que contiene las pruebas, o conectando su proyecto de prueba para generar las especificaciones después de prueba de funcionamiento. Para hacer esto último, primero haga referencia a la biblioteca CoreBDD.SpecGeneration

```
PS C:\Users\Virginia Aquino\documents\github\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci- > dotnet add package CoreBDD.SpecGeneration
Determinando los proyectos que se van a restaurar...
Writing C:\Users\Virginia Aquino\AppData\Local\Temp\tmp7D3C.tmp
Info : Agregando PackageReference para el paquete "CoreBDD.SpecGeneration" al proyecto "C:\Users\Virginia Aquino\documents\github\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-.csproj".
Info : Restaurando paquetes para C:\Users\Virginia Aquino\documents\github\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-.csproj...
Info : CACHE https://api.nuget.org/v3-flatcontainer/corebdd.specgeneration/index.json
Info : El paquete "CoreBDD.SpecGeneration" es compatible con todos los marcos de trabajo especificados del proyecto "C:\Users\Virginia Aquino\documents\github\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-.csproj".
Info : Se actualizó PackageReference para la versión "1.0.0" del paquete "CoreBDD.SpecGeneration" en el archivo "C:\Users\Virginia Aquino\documents\github\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-.csproj".
Info : Ejecutando restauración...
Info : Escribiendo el archivo de recursos en el disco. Ruta de acceso: C:\Users\Virginia Aquino\documents\github\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-\obj\project.assets.json
log : Se ha restaurado C:\Users\Virginia Aquino\documents\github\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-.csproj (en 5,18 sec).
PS C:\Users\Virginia Aquino\documents\github\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci- >
```

- A continuación, cree una clase de Fixture dentro de su proyecto de prueba y llame a `GenerateSpecs.OutputFeatureSpecs` dentro del método `Dispose`, pasando el ensamblado (o ruta al ensamblado) y la carpeta de salida para las especificaciones generadas.

```
File Edit Selection View Go Run Terminal Help GenerateSpecsFixture.cs - Visual Studio Code
Calculator.cs MultiplyTwoNumbers.cs GenerateSpecsFixture.cs X
C:\Users\Virginia Aquino\Documents\GitHub\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci- > Fixtures > GenerateSpecsFixture.cs
6 namespace InformelabN_05_Utilizando_Desarrollo_guiado_por_comportamiento_BDD_para_realizaci_
7 {
8     [CollectionDefinition("CoreBDD")]
9     public class Collection : ICollectionFixture<GenerateSpecsFixture> { }
10
11     public class GenerateSpecsFixture : IDisposable
12     {
13         public void Dispose()
14         {
15             //var specsFolder = Directory.GetParent(System.Reflection.Assembly.GetExecutingAssembly().Location).Parent.Parent.Parent.FullName;
16             // GenerateSpecs.OutputFeatureSpecs(this.GetType().Assembly.Location, specsFolder + @"\Specs");
17             GenerateSpecs.OutputFeatureSpecs(this.GetType().Assembly.Location, @"..\..\..\Specs\");
18         }
19     }
20 }
21
22 }
```

- Cuando las pruebas terminan de ejecutarse, se genera un archivo `FeatureName.feature` en la carpeta Especificaciones del proyecto de prueba xUnit. Genera especificaciones de Gherkin para la característica y los escenarios relacionados. Ejemplo `CalculatorFeature.feature` :

A screenshot of the Visual Studio Code editor interface. The top bar shows several open files: Calculator.cs, MultiplyTwoNumbers.cs, GenerateSpecsFixture.cs, CalculatorFeature.feature (selected), and Extension: BAB. The main editor area displays the content of CalculatorFeature.feature. The file path is shown as C:\Users\Virginia Aquino\Documents\GitHub\InformeLabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-. The code is written in Gherkin syntax, starting with a comment that the file is auto-generated. It defines a feature 'Calculator' with the purpose of avoiding silly mistakes as a math idiot who wants to be told the sum of two numbers. Three scenarios are listed: 'Add two numbers', 'Multiply two numbers', and 'Divide two numbers'. Each scenario includes a 'Given' clause, a 'When' clause, and a 'Then' clause. The 'Add' scenario expects a result of 3. The 'Multiply' scenario expects a total of 50 and an equation of 10 x 5. The 'Divide' and 'Subtract' scenarios (partially visible) also expect a result of 3.

```
C: > Users > Virginia Aquino > Documents > GitHub > InformeLabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-
1  # This file is auto-generated, any changes made to this file will be lost
2
3
4  Feature: Calculator
5      In order to avoid silly mistakes
6      As a math idiot
7      I want to be told the sum of two numbers
8
9  Scenario: Add two numbers
10     Given I have entered 1 into the calculator
11     And I have also entered 2 into the calculator
12     When I press add
13     Then the result should be 3
14
15  Scenario: Multiply two numbers
16     Given I have a calculator
17     When I key in 10
18     And I key in 5 and press multiply
19     Then It sets the Total to 50
20     And It sets the equation to 10 x 5
21
22  Scenario: Divide two numbers
23
24  Scenario: Subtract two numbers
25     Given I have entered 5 into the calculator
26     And I have also entered 2 into the calculator
27     When I press minus
28     Then the result should be 3
```

3. Herramienta de línea de comando

- La herramienta de línea de comandos facilita la ejecución de tareas como la ejecución de pruebas con la salida de estilo Gherkin, la generación de archivos de prueba de características y escenarios predeterminados y la generación de archivos de funciones de Gherkin a partir de pruebas existentes o la generación de pruebas a partir de archivos de funciones existentes.
- Comenzando desde cero usando la plantilla dotnet y las herramientas cli: mkdir demobdd cd demobdd
- Luego cree el nuevo proyecto CoreBDD dotnet new corebdd
- Encuentra pruebas de CoreBDD en directorios actuales y secundarios y ejecuta pruebas

```

PS C:\Users\Virginia Aquino\documents\github\InformeLabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-> mkdir demobdd

Directorio: C:\Users\Virginia Aquino\documents\github\InformeLabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-

Mode                LastWriteTime         Length Name
----                -
d-----         18/07/2020    22:22         demobdd

PS C:\Users\Virginia Aquino\documents\github\InformeLabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-> cd demobdd
PS C:\Users\Virginia Aquino\documents\github\InformeLabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-> dotnet new corebdd
The template "CoreBDD Test Project Template" was created successfully.
PS C:\Users\Virginia Aquino\documents\github\InformeLabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-> corebdd test

Unhandled Exception: System.Reflection.TargetInvocationException: Exception has been thrown by the target of an invocation. ---> System.Exception: Unable to process the the project 'C:\Users\Virginia Aquino\documents\github\InformeLabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci->demobdd\demobdd.csproj'. Are you sure this is a valid .NET Core or .NET Standard project type?

Here is the full error message returned from the Microsoft Build Engine:

Microsoft (R) Build Engine versión 16.7.0-preview-20310-07+e1c9fd0c para .NET
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

MSBUILD : error MSB1008: Sólo puede especificarse un proyecto.
Modificador: Aquino\AppData\Local\Temp\tmpB4AD.tmp

Para modificar la sintaxis, escriba "MSBuild -help"

    at CoreBDD.CommandLine.Tools.Analyzer.DependencyGraphService.GenerateDependencyGraph(String projectPath) in C:\Users\Steven\Source\CoreBDD\src\CoreBDD.CommandLine\Tools\Analyzer\DependencyGraphService.cs:line 31

```

- Ejecute pruebas y luego genere archivos .feature de Gherkin en una ubicación específica
corebdd test --specs --output ./Specs

```

PS C:\Users\Virginia Aquino\documents\github\InformeLabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci->demobdd> corebdd test --specs --output ./Specs

Unhandled Exception: System.Reflection.TargetInvocationException: Exception has been thrown by the target of an invocation. ---> System.Exception: Unable to process the the project 'C:\Users\Virginia Aquino\documents\github\InformeLabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci->demobdd\demobdd.csproj'. Are you sure this is a valid .NET Core or .NET Standard project type?

Here is the full error message returned from the Microsoft Build Engine:

Microsoft (R) Build Engine versión 16.7.0-preview-20310-07+e1c9fd0c para .NET
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

MSBUILD : error MSB1008: Sólo puede especificarse un proyecto.
Modificador: Aquino\AppData\Local\Temp\tmpD018.tmp

Para modificar la sintaxis, escriba "MSBuild -help"

    at CoreBDD.CommandLine.Tools.Analyzer.DependencyGraphService.GenerateDependencyGraph(String projectPath) in C:\Users\Steven\Source\CoreBDD\src\CoreBDD.CommandLine\Tools\Analyzer\DependencyGraphService.cs:line 31
    at CoreBDD.CommandLine.Tools.Analyzer.Analyze.IsCoreBDD(String projectPath) in C:\Users\Steven\Source\CoreBDD\src\CoreBDD.CommandLine\Tools\Analyzer\Analyze.cs:line 18
    at CoreBDD.CommandLine.Tools.Analyzer.FindCoreBDDAssemblies.Find(String path) in C:\Users\Steven\Source\CoreBDD\src\CoreBDD.CommandLine\Tools\Analyzer\FindCoreBDDAssemblies.cs:line 19
    at CoreBDD.CommandLine.Tools.TestRunner.Run(Boolean generateSpecs, String path, String output) in C:\Users\Steven\Source\CoreBDD\src\CoreBDD.CommandLine\Tools\TestRunner.cs:line 21
    --- End of inner exception stack trace ---
    at System.RuntimeMethodHandle.InvokeMethod(Object target, Object[] arguments, Signature sig, Boolean constructor, Boolean wrapExceptions)
    at System.Reflection.RuntimeMethodInfo.Invoke(Object obj, BindingFlags invokeAttr, Binder binder, Object[] parameters, CultureInfo culture)

```

- Andamio de una clase de entidad CoreBDD llamada 'Iniciar sesión' en la carpeta actual
corebdd generate feature --name login --namespace demobdd
- Andamio de una clase de escenario CoreBDD llamada 'LoginToWebsite' en la función 'Iniciar sesión' corebdd generate scenario --name LoginToWebsite --feature login --namespace demobdd

```

PS C:\Users\Virginia Aquino\documents\github\InformeLabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci->demobdd> corebdd generate scenario --name LoginToWebsite --feature login --namespace demobdd
PS C:\Users\Virginia Aquino\documents\github\InformeLabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci->demobdd> corebdd test

Unhandled Exception: System.Reflection.TargetInvocationException: Exception has been thrown by the target of an invocation. ---> System.Exception: Unable to process the the project 'C:\Users\Virginia Aquino\documents\github\InformeLabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci->demobdd\demobdd.csproj'. Are you sure this is a valid .NET Core or .NET Standard project type?

Here is the full error message returned from the Microsoft Build Engine:

Microsoft (R) Build Engine versión 16.7.0-preview-20310-07+e1c9fd0c para .NET
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

MSBUILD : error MSB1008: Sólo puede especificarse un proyecto.
Modificador: Aquino\AppData\Local\Temp\tmp760.tmp

Para modificar la sintaxis, escriba "MSBuild -help"

```

- Scaffold CoreBDD Pruebas de archivos existentes '.feature' de pepinillo, especificando la ubicación de los archivos de características y la carpeta de destino para las pruebas generadas.

- Si ha seguido utilizando el ejemplo de prueba 'corebdd', elimine la carpeta 'Características' (dejando la carpeta Especificaciones con archivos .feature intactos) y luego ejecute: `corebdd generate tests -path ./Specs -output ./Features -namespace demobdd`

```

at Corebdd.CommandLine.Program.Main(String[] args) in C:\Users\Steven\source\Corebdd\src\Corebdd.CommandLine\Program.cs:line 11
PS C:\Users\Virginia Aquino\documents\github\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-\demobdd> corebdd generate scenario --name LoginToWebsite --Featu
re login --namespace demobdd
PS C:\Users\Virginia Aquino\documents\github\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-\demobdd> corebdd generate tests --path ./Specs --output ./Featur
es --namespace demobdd
PS C:\Users\Virginia Aquino\documents\github\InformelabN-05-Utilizando-Desarrollo-guiado-por-comportamiento-BDD-para-realizaci-\demobdd>

```

- Ahora debería tener los apéndices de prueba regenerados utilizando los escenarios de archivo .feature.

Features	20/07/2020 22:17	Carpeta de archivos	
Fixtures	18/07/2020 22:23	Carpeta de archivos	
Mod	Fecha de creación: 18/07/2020 22:23	Carpeta de archivos	
Spe	Tamaño: 588 bytes	Carpeta de archivos	
	Archivos: GenerateSpecsFixture.cs	Carpeta de archivos	
demobdd.csproj	18/07/2020 22:23	Visual C# Project ...	1 KB