

Lab Exercise: Specify and implement one or more System Tests using SpecFlow and Gherkin

1. Install the SpecFlow for Visual Studio Extension using Tools/Extensions and Updates.. if you have not already done so. Restart VS.
2. Obtain the solution to the Microwave Oven Blackboard, and open it in Visual Studio. The specification and design is described in the document also available on Blackboard.
3. Make a new project, using the C#/Test/Unit Test Project (.NET Framework) template. When you give it a name, do NOT include a substring like ".System." as it confuses the compiler. ("SystemTest" should be OK).
4. Use NuGet package manager to install SpecFlow.NUnit, SpecFlow.NUnit.Runners and NUnit3TestAdapter to your new test project.
5. You can delete the UnitTest C# file from the project.
6. Add New Item to the project, selecting SpecFlow Feature File template.
7. For your system feature, you should describe and test setting the cooking time for the microwave oven in Gherkin. It must have the following steps:
 - 7.1.Press the Power Button 1 times
 - 7.2.Press the Time Button 1 times
 - 7.3.The Display should show 01:00
8. Use the right click menu in the .feature file to generate a test class with skeletons for your Gherkin steps.
9. See if you can build now. If there is a problem, try to remove the SpecFlowNUnitExtensions.cs file, you don't need that.
- 10.From Test/Windows open the Test Explorer. See what happens if you run the tests from the System/Acceptance test project. It shouldn't work just yet.

11. Fill in code in the skeleton class for the steps. You can let yourself be inspired by the code in the Microwave.Application/Program.cs. You must set up the System under test, using a fake for the IOutput interface. Or you can be inspired by your own code for the Microwave Integration Handin.
12. Try to build, and run the tests from the Test Explorer. Try to navigate from the Test Explorer to the specification – It can find the right place in the feature file.
13. Add one or more scenarios to the timer feature. If you have been smart, the new scenarios will reuse the step implementations you already defined, as SpecFlow will recognize parameters such as time and generate a regular expression.