

1 Основные понятия

Specflow - это “исполняемая спецификация” .NET фреймворк с открытым исходным кодом, ориентированный на следующие подходы к разработке:

Behavior Driven Development (BDD) - разработка управляемая через поведение

Acceptance test-driven development (ATDD) - разработка на основе приемочных испытаний

Specflow тесты - Это тесты написанные “максимально естественным языком” для каких либо испытаний для приемки конечного продукта. В Specflow для этого используется язык Gherkin (см. далее), пример теста с разбором [здесь](#)

Gherkin – это структурированный естественный язык (Natural Language), который используется для описания поведения системы по заданному сценарию. Может использоваться в рамках Agile-практики Behavior Driven Development (BDD)

Selenium WebDriver — это инструмент для автоматизации действий веб-браузера. В большинстве случаев используется для тестирования Web-приложений, но этим не ограничивается. В частности, он может быть использован для решения рутинных задач администрирования сайта или регулярного получения данных из различных источников (сайтов).

2 Понятия тестов Specflow

Scenario - это конкретный тест (сценарий) состоящий из шагов (Step - см. далее)

Step - это шаг конкретного теста (сценария), выполняющий определенное действие/проверку - примеры:

And Click on element with class 'zoom-in-btn' - Нажать на кнопку с классом zoom-in-btn
Then Zoom indicator value should be '300%' - Проверяет что значение индикатора zoom должно быть 300%

Тэг - это директива которая начинается с @ для того чтобы определить область (scope) для выполнения шагов(steps)/хуков (hooks) тестов

Ставится в самом начале файла Feature

Например:

@webApp - определяет тесты которые относятся к веб-приложению (будет запущен браузер)

@api - определяет тесты для тестирования API (браузер не будет запущен)

Тэги можно использовать не только для общего разделения тестов frontend или backend, а например для задания какого-то специфического общего контекста (см. Далее FeatureContext, ScenarioContext) между тестами или для всей группы тестов (для всей Feature)

3 Что могут тестировать тесты Specflow принцип работы

Specflow тесты могут тестировать не только frontend (UI), можно тестировать и бэкенд (запросы и ответы к серверу) api. Общий принцип работы такой:

- В файлах *.feature есть текст с текстами написанными на языке Gherkin
- Специальный компилятор от Specflow генерирует по данным файлам исходный код на языке C# с тестами NUnit
- Исходный код на языке C# компилируется компилятором языка C#
- на движке Specflow выполняются тесты (Scenario) написанные на языке Gherkin
- **Каждый шаг (Step) в тесте описан реализацией на C# и языке .NET**
- Все взаимодействие с браузером происходит через **Selenium Webdriver**

3.1 Разбор группы тестов (конкретного файла Feature), их шагов и их реализации

На рис.1 представлен фрагмент файла GroupDocs.Viewer.web.Document.Viewer.feature, **на линии 1** один находится тэг

Рис.1 - Файл GroupDocs.Viewer.web.Document.Viewer.feature
с группой тестов (Feature)

@webapp что указывает что тесты относятся к тестам веб-приложения и что нужно запустить веб-браузер для выполнения тестов, как это реализовано в коде будет указано чуть позднее.

Далее на линии 2 ключевое слово **Feature** начинает собственно Feature - группу тестов, через пробел идет текстовое описание "GroupDocs Viewer Web viewer check" - в данной Feature находятся тесты тестирующие интерфейс, конкретно - кнопки экрана отображения документа.

Далее на линии 4 ключевое слово **Background** начинает специальную секцию (Background секцию) **которая будет выполняться перед каждым тестом** (Scenario) в данной Feature.

На линии 5 - имеется **ШАГ (Step)**

Given Successfully rendered '6-pages.docx' document

То есть "Дан успешно прорисованный '6-pages.docx' документ" - то есть проверяется что документ был загружен и отображен на экране.

Посмотрим на реализацию данного шага (в студии для просмотра можно нажать F12):