

Функциональное тестирование: BDD

№ урока: 8 **Курс:** Тестирование ASP.NET Core MVC приложений

Средства обучения: Visual Studio 2019 Community Edition

Обзор, цель и назначение урока

Вспомнить, что такое функциональный тест. Разобраться в том, что такое BDD подход. Разобраться со структурой функциональных тестов в подходе BDD, рассмотреть признаки хорошего функционального теста и типичные ошибки в подходе BDD. Рассмотреть фреймворк SpecFlow для функционального тестов в стиле BDD. Рассмотрение демо проекта. Резюме знаний, полученных за курс.

Изучив материал данного занятия, учащийся сможет

- Разбираться в том, как правильно писать функциональные тесты в подходе BDD.
- Разбираться в том, какие признаки хорошего функционального теста в подходе BDD.
- Разобраться в том, как избежать типичных ошибок при реализации функционального теста в подходе BDD.
- Рассмотреть на практике, как и с чего начинать писать функциональные тесты в подходе BDD.
- Получить представление о фреймворке SpecFlow.

Содержание урока

1. Напоминание про функциональное тестирование
2. Что такое BDD
3. Язык Gherkin
4. Фреймворк SpecFlow
5. Демо
6. С чего начать тестирование ASP.NET Core MVC приложений
7. Схема тестирования ASP.NET Core MVC приложений
8. Выводы

Резюме

- **Функциональное тестирование** – тестирование реализации функциональных требований программы.
- **BDD (Behavior Driven development)** – тестирование реализации функциональных требований программы, на основе описания её поведения с помощью специального языка предметной области.
- **Преимущества BDD** – устойчивость к рефакторингу и понятность для нетехнических специалистов.
- **Язык Gherkin** – специальный логический язык для описания предметной области.
- **Фреймворк SpecFlow** – специальный тестовый фреймворк предназначенный для написания функциональных тестов в BDD стиле с помощью языка Gherkin.
- **Ключевое преимущество функциональных тестов в стиле BDD заключается в объединении триады:** разработчики, тестировщики, бизнес, с помощью выражения функциональных требования программы на специальном языке, который максимально близок к предметной области и понятен людям, которые не являются техническими

специалистами. Именно за счёт формирования такого всеобщего языка данный подход предпочтительнее чем наивный.

- **Ключевым недостатком функциональных тестов в стиле BDD является то, что** это дополнительная когнитивная нагрузка, это повышенный уровень ответственности и повышенное количество коммуникаций, а чем больше коммуникаций – тем больше замедляются процессы. В итоге здесь идёт выбор между скоростью и качеством.
- **Для новых проектов** идеально будет сначала выписать все функциональные требования в виде BDD тестов, а дальше просто их реализовать.
- **Для старых проектов** идеально будет сначала зафиксировать текущую бизнес-логику в виде unit-тестов для дальнейшего её рефакторинга.
- **Самые важные тесты** – это тесты на вашу бизнес-логику, то, что коренным образом отличает ваше приложение от остальных. При этом, это могут быть как unit так и функциональные тесты. Дальше идут интеграционные тесты на проверку корректности взаимодействия компонент. Было бы неплохо создавать по тесту на каждый заведенный баг в вашем приложении.
- **Запомните самое важное** – вы не сможете проверить абсолютно всё! Проверяйте только самое важное в начале! Изначально качественная архитектура вашего проекта ведёт к меньшему количеству багов, к меньшей потребности в тестах и к более высокому качеству самих тестов!
- **Пишите тесты осознанно, они ваш друг, а не обязанность!**

Закрепление материала

- Что такое функциональный тест?
- Что такое BDD?
- Каковы преимущества BDD?
- Назовите недостатки BDD?
- Опишите структуру языка Gherkin.
- Опишите варианты начала тестирования ASP.NET Core MVC приложений.

Дополнительное задание

Задание

Напишите функциональные тесты с применением фреймворка SpecFlow на некоторый функционал в ASP.NET Core MVC программы, без реализации этого функционала.

Самостоятельная деятельность учащегося

Задание 1

Выучите основные понятия, рассмотренные на уроке.

Задание 2

Дополнительно найдите сопутствующую информацию по теме. Составьте сводную таблицу по основным понятиям, найдите схожие и отличающиеся моменты в описаниях, постарайтесь сформулировать своё понимание данной темы.

Задание 3

Напишите функциональные тесты на фреймворке SpecFlow на бизнес-логику вашего проекта из предыдущих уроков. Используйте любые библиотеки, помогающие написанию функциональных тестов.

Задание 4

Попробуйте сначала написать функциональные тесты на фреймворке SpecFlow, а затем написать приложение, которое удовлетворяет эти тесты.

Задание 5

Попробуйте сначала написать unit-тесты на бизнес-логику, а затем бизнес-логику и соответственно приложение, которое удовлетворяет эти тесты.

Рекомендуемые ресурсы

Test ASP.NET Core MVC apps

<https://docs.microsoft.com/en-us/dotnet/architecture/modern-web-apps-azure/test-asp-net-core-mvc-apps>

Functional testing

<http://softwaretestingfundamentals.com/functional-testing>

User story vs requirements

<https://blog.testlodge.com/user-story-vs-requirements>

Gherkin

<https://cucumber.io/docs/gherkin/>

SpecFlow Getting Started

<https://specflow.org/getting-started/>