

Интеграционное тестирование View

№ урока: 5 **Курс:** Тестирование ASP.NET Core MVC приложений

Средства обучения: Visual Studio 2019 Community Edition

Обзор, цель и назначение урока

Вспомнить, что такое интеграционные тесты. Вспомнить, что такое View. Разобраться в том, как правильно писать интеграционные тесты для View. Рассмотрение демо проекта.

Изучив материал данного занятия, учащийся сможет

- Разбираться в том, как правильно писать интеграционные тесты для View.
- Рассмотреть на практике то, как и с чего начинать писать интеграционные тесты для View.

Содержание урока

1. Напоминание про интеграционные тесты
2. View: что это и как это тестировать
3. Демо

Резюме

- **Интеграционный тест** – медленная проверка группы взаимодействующих компонентов программы.
- **View** – компоненты, отвечающие за отображение пользовательского интерфейса и взаимодействие с пользователем. В общем случае отображают данные из модели и реагируют на пользовательские действия.
- **Структура View** – Controller, ViewModel, View
- **Основное, что стоит запомнить про View** – то, что это, по сути дела, обычная HTML страница, сгенерированная с помощью языка разметки Razor на стороне сервера, и отношение к ней должно быть как к обычной HTML странице. То, как генерируется эта страница это неотъемлемая часть теста. По сути, всё интеграционное тестирование View и состоит из проверки коммуникации (что вернулось правильное View), и проверки выходного результата (проверка самого View).
- **Специфика тестирования View** такова, что конкретно на стороне ASP.NET Core MVC это лучше не делать, так как это невероятно хрупкое и неустойчивое к рефакторингу тестирование. Для тестирования HTML страниц есть более подходящие методы, которые заточены именно под их специфику.
- **Лучше оставить интеграционное тестирование View на потом**, данное тестирование не является приоритетным, в крайнем случае всегда можно проверить страницу вручную, этот вид тестирования более пригоден для ручного тестирования чем, например интеграционное тестирование базы данных.

Закрепление материала

- Что такое интеграционный тест?
- Что такое View?

- Опишите взаимосвязь между View и интеграционными тестами.
- Опишите структуру View.
- В чём недостатки интеграционного тестирования View средствами ASP.NET Core MVC?

Дополнительное задание

Задание

Реализуйте простейшее View и напишите для него интеграционные тесты.

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

Реализуйте необходимые View для проекта из предыдущих уроков. Не забудьте про необходимость использования ViewModel & ViewBag, для отработки усвоенного материала.

Задание 4

Напишите интеграционные тесты на View из предыдущего задания. Используйте любой тестовый фреймворк на языке программирования C# и любые библиотеки, помогающие написанию интеграционных тестов.

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>

Integration tests in ASP.NET Core

<https://docs.microsoft.com/en-us/aspnet/core/test/integration-tests?view=aspnetcore-3.1>

Test naming

<https://enterprisecraftsmanship.com/posts/you-naming-tests-wrong/>

Views

<https://docs.microsoft.com/en-us/aspnet/core/mvc/views/overview?view=aspnetcore-3.1>