



**Microsoft** Partner  
Silver Learning

ASP.NET Core MVC

# Тестирование ASP.NET Core MVC приложений

Интеграционное тестирование компонентов ASP.NET Core MVC

# Тестирование ASP.NET Core MVC приложений

## Introduction



Науринский Юрий  
.NET Developer

 yurii.naurynskyi

 yurii.naurynskyi

.NET Core

ASP.NET Core

ASP.NET Core MVC

Тема

## Интеграционное тестирование компонентов ASP.NET Core MVC

# Тестирование ASP.NET Core MVC приложений

## Интеграционное тестирование компонентов ASP.NET Core MVC

1. Что такое интеграционный тест
2. Признаки хорошего интеграционного теста
3. Структура интеграционного теста
4. Что может проверять интеграционный тест
5. Типичные ошибки в интеграционных тестах
6. Компоненты фреймворка: что это и как это тестировать
7. Демо

# Тестирование ASP.NET Core MVC приложений

## Что такое интеграционный тест

**Интеграционный тест** – медленная проверка группы взаимодействующих компонентов программы.

Хороший интеграционный тест – не использует лишних внешних зависимостей, и не дублирует unit-тесты!

Плохой unit-тест – требует слишком много внешних ресурсов, работает за нестабильное время, не приносит дополнительных гарантий!

# Тестирование ASP.NET Core MVC приложений

## Термины

### Медленный

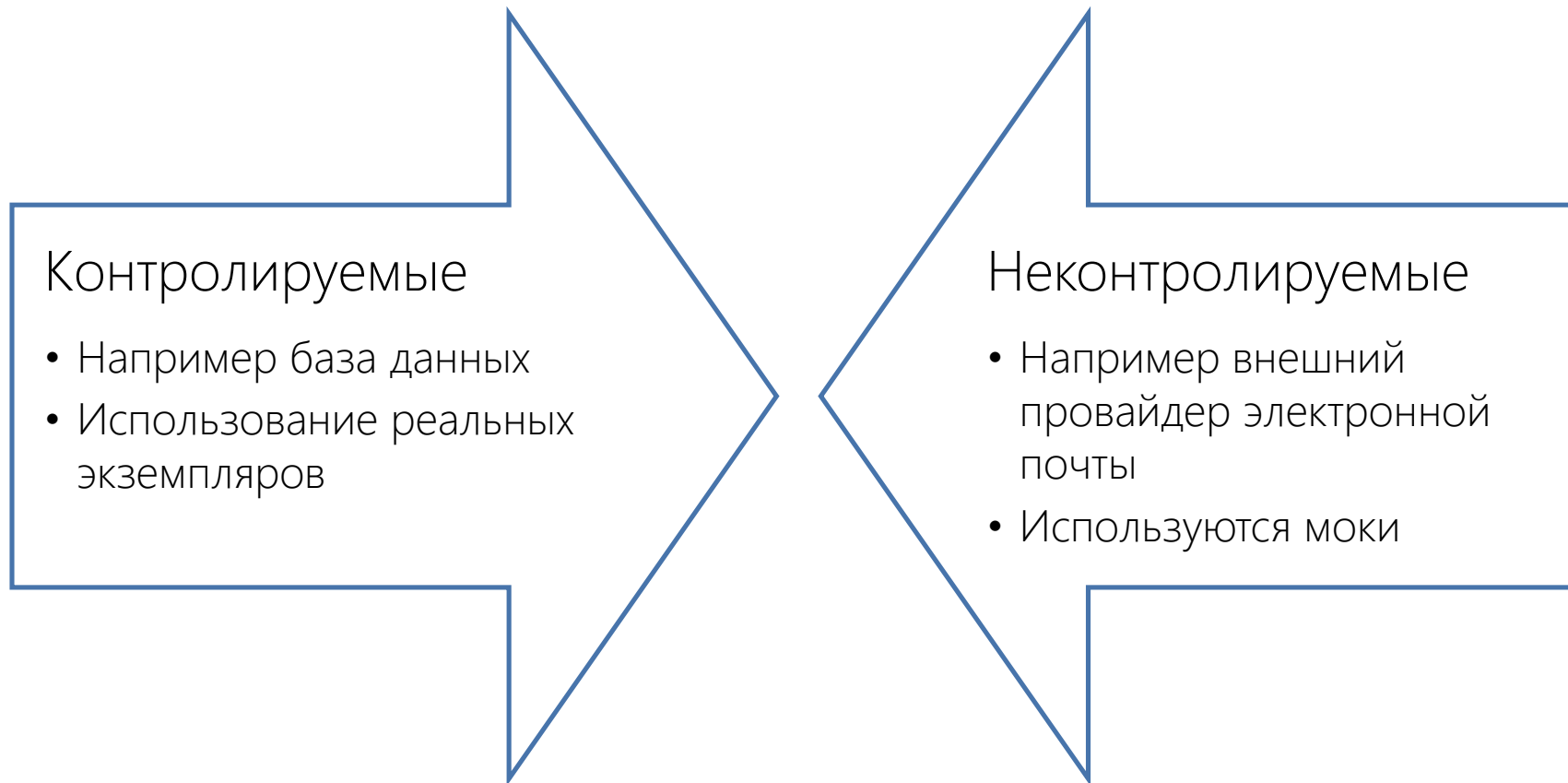
- Используются внешние ресурсы, не относящиеся к программе
- Параллельное выполнение требует необходимой подготовки тестов
- Сложная подготовка тестового окружения
- Тесты всё ещё изолированы друг от друга!

### Группа взаимодействующих компонентов программы

- Только необходимые для теста компоненты
- Желательное минимальное количество компонентов - 2

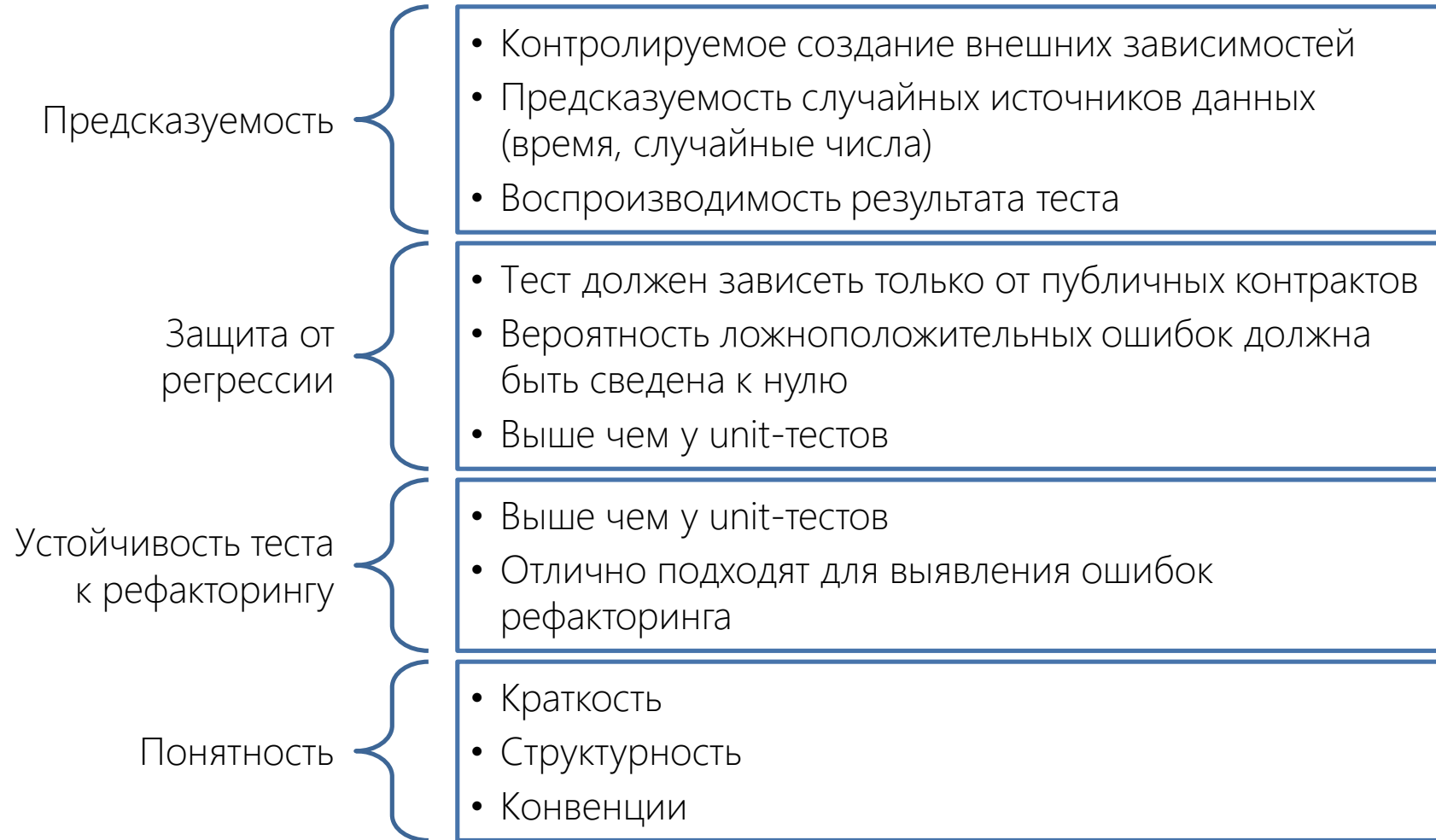
# Тестирование ASP.NET Core MVC приложений

## Внешние зависимости



# Тестирование ASP.NET Core MVC приложений

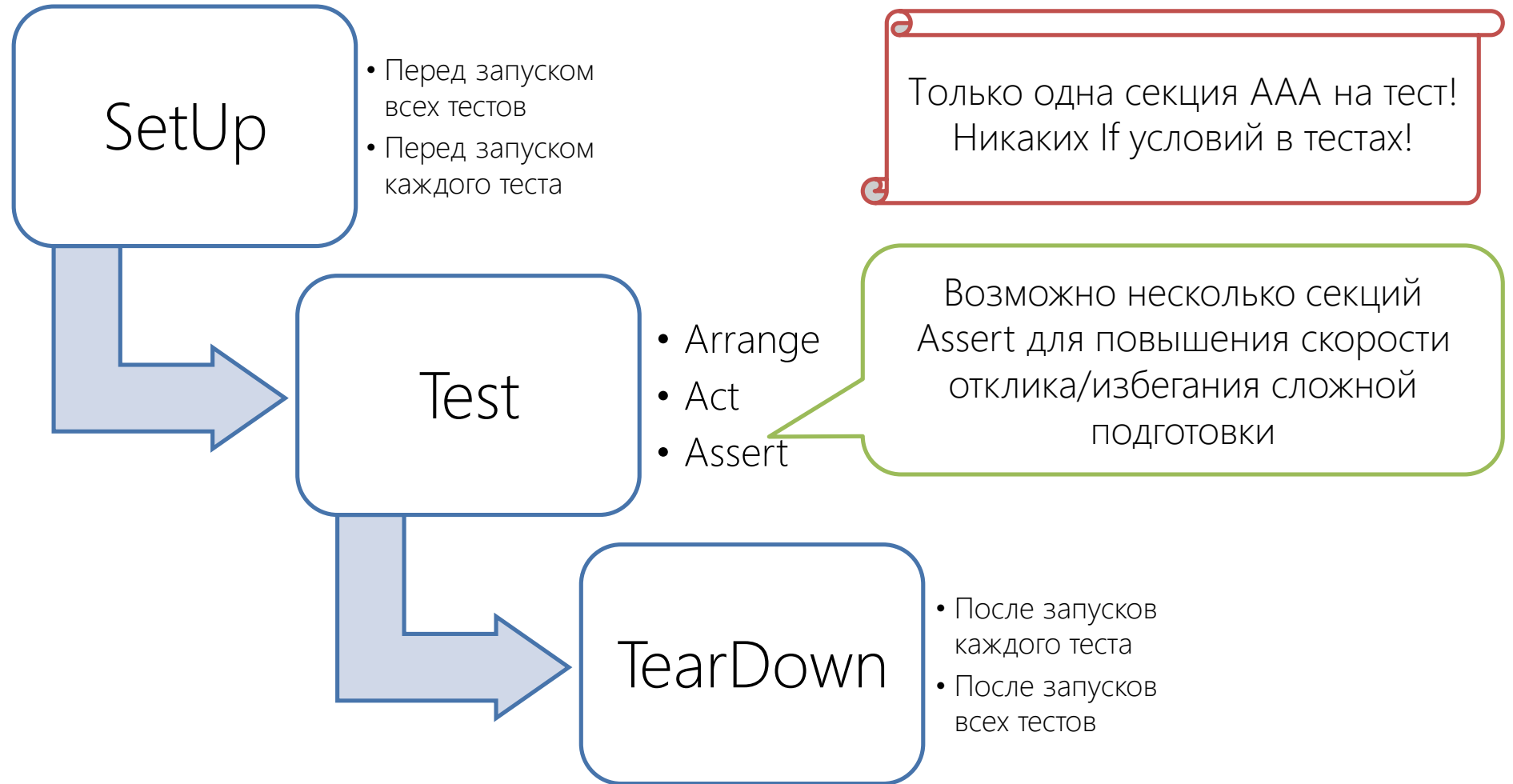
## Признаки хорошего интеграционного теста





# Тестирование ASP.NET Core MVC приложений

## Структура интеграционного теста



# Тестирование ASP.NET Core MVC приложений

## Что может проверять интеграционный тест

### Выходной результат

- Возможно как часть проверки контролируемой внешней зависимости

### Состояние

- В случае контролируемой внешней зависимости
- В случае неконтролируемой зависимости не имеет смысла, так как используются моки

### Коммуникацию

- В случае неконтролируемой внешней зависимости (так как используются моки)
- В случае контролируемой зависимости, сам факт изменения состояния является подтверждением коммуникации

# Тестирование ASP.NET Core MVC приложений

## Типичные ошибки в интеграционных тестах

Слишком много  
уровней  
абстракции

Круговые  
зависимости

Недетерминированное  
время выполнения теста

Тест-пустышка

Завязывание на детали  
реализации

Дублирование unit-  
тестов

# Тестирование ASP.NET Core MVC приложений

## Компоненты фреймворка: Routing

**Routing** – отвечает за сопоставление HTTP запросов и передает сопоставленные запросы на соответствующий контроллер.

Это контролируемый аспект работы программы, деталь реализации, поэтому здесь используется реальный экземпляр приложения.

# Тестирование ASP.NET Core MVC приложений

## Компоненты фреймворка: Validation

**Validation** – логика проверки на валидность HTTP запросов, переданных на контроллер.

Это контролируемый аспект работы программы, деталь реализации, поэтому здесь используется реальный экземпляр приложения.

# Тестирование ASP.NET Core MVC приложений

## Компоненты фреймворка: Configuration

**Configuration** – совокупность настроек программы, используемых в дальнейшей работе.

Это контролируемый аспект работы программы, деталь реализации, поэтому здесь используется реальный экземпляр приложения.

Но возможно использование внешних, неконтролируемых источников конфигурации, тогда используются моки.

# Тестирование ASP.NET Core MVC приложений

## Компоненты фреймворка: Security

**Security** – аутентификация и авторизация HTTP запросов к контроллеру.

Это контролируемый аспект работы программы, деталь реализации, поэтому здесь используется реальный экземпляр приложения.

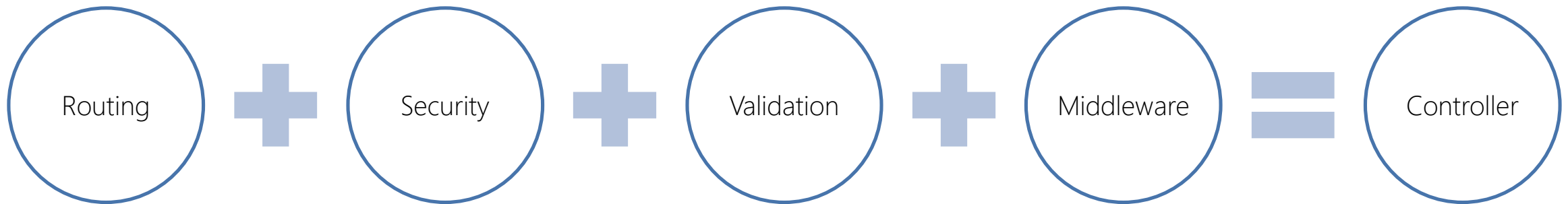
Но возможно использование внешних, неконтролируемых инструментов безопасности, тогда используются моки.

Ошибки первого рода, когда авторизованные пользователи классифицируются как нарушители, нежелательны.

Ошибки второго рода, когда нарушители классифицируются как авторизованные пользователи, не допустимы!

# Тестирование ASP.NET Core MVC приложений

## Компоненты фреймворка: Controller





## Демо

# Проверка знаний

TestProvider.com



Проверьте как Вы усвоили данный материал на [TestProvider.com](http://TestProvider.com)

TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и для общей оценки знаний IT специалиста.

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.

# Тестирование ASP.NET Core MVC приложений

Спасибо за внимание! До новых встреч!



Науринский Юрий  
.NET Developer

.NET Core  
ASP.NET Core  
ASP.NET Core MVC

# Информационный видеосервис для разработчиков программного обеспечения

