

TDD or not TDD



# Светлана Кривенко

- 4,5 года как Scrum Master
- 11 лет как Java Full Stack / Team Lead (Outsource)
- со-организатор TDD практик  
онлайн (Группа Technical Excellence)  
оффлайн (Клуб Московских Программистов)



# TDD больше 20 лет



Кент Бек - пере-открыватель TDD

- 1999: Экстремальное программирование
- 2000: Разработка через тестирование

2001: Agile Манифест

КЕНТ БЕК

## ЭКСТРЕМАЛЬНОЕ ПРОГРАММИРОВАНИЕ

«ЧИСТЫЙ КОД, КОТОРЫЙ РАБОТАЕТ, — ВОТ ЦЕЛЬ,  
К КОТОРОЙ СТОИТ СТРЕМИТЬСЯ»

## РАЗРАБОТКА ЧЕРЕЗ ТЕСТИРОВАНИЕ

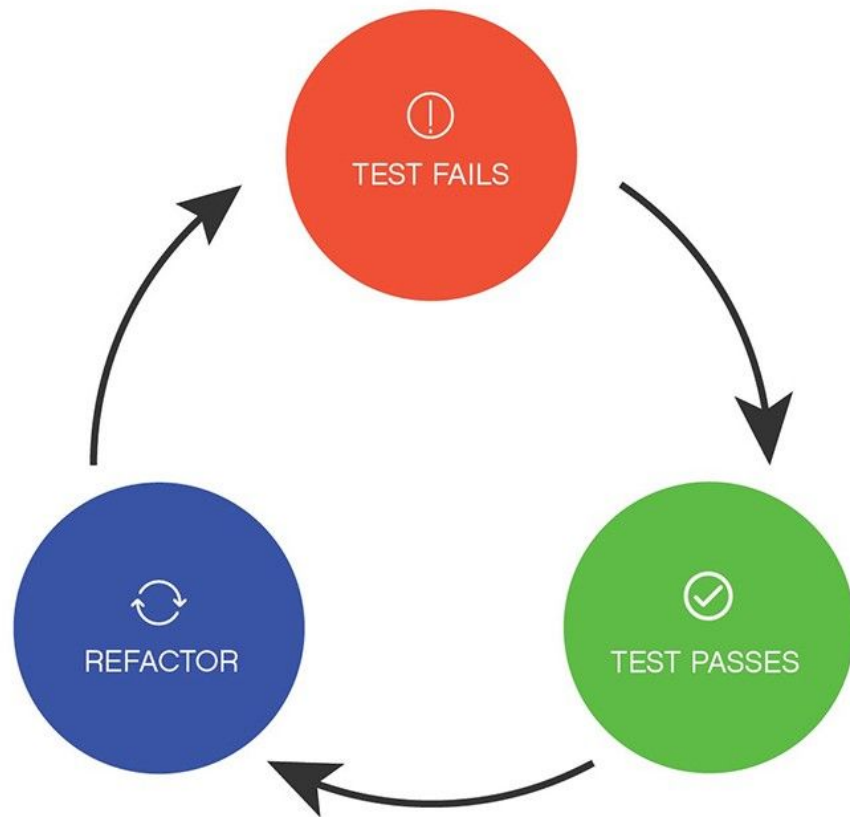
# TDD

  
Addison-Wesley

 ПИТЕР®

# TDD на минималках

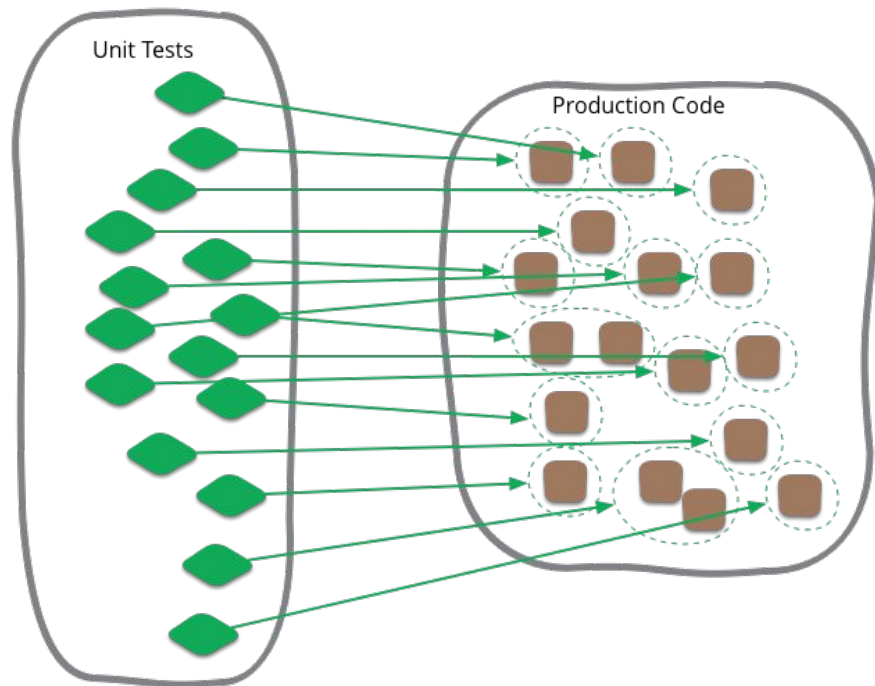
1. **Red** - падающий тест
2. **Green** - минимальный работающий код
3. **Refactor** - чистый код, который работает
4. Repeat - следующий шаг



# Unit Test на минималках

Тест - для проверки функциональности  
(не покрытия кода или поиска ошибок)

- Облегчает изменения и рефакторинг
- Упрощает интеграцию
- Документирует
- Помогает проектированию





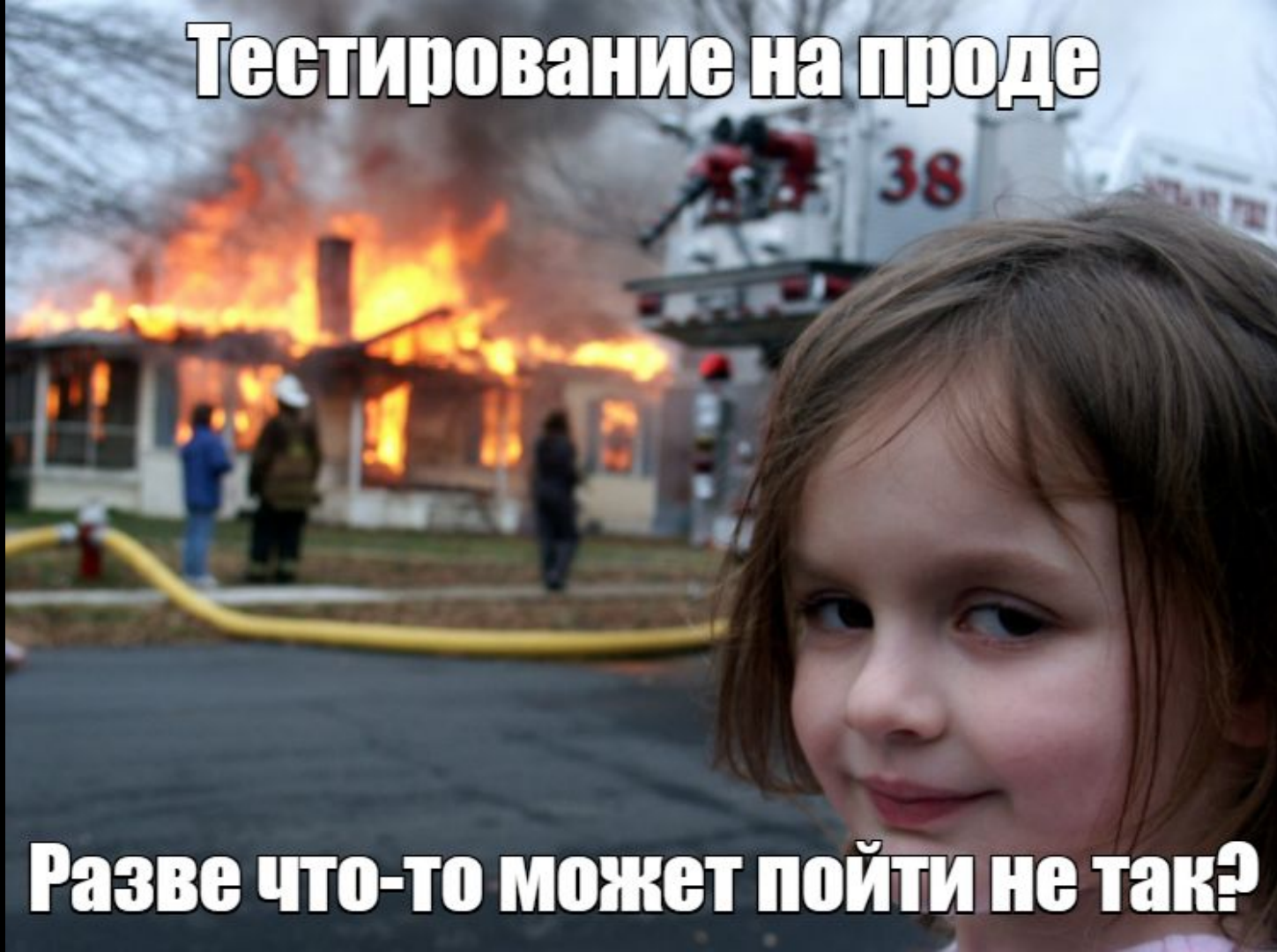
# Зачем нужен TDD?

Помогает

- получать быстро обратную связь
- разбивать на части
- не добавлять легаси
- уверенно рефакторить и релизить
- меньше времени тратить на дебаг и баг фиксы



**Тестирование на проде**



**Разве что-то может пойти не так?**

**Релиз в пятницу вечером не  
проблема**

**Если суббота рабочая**

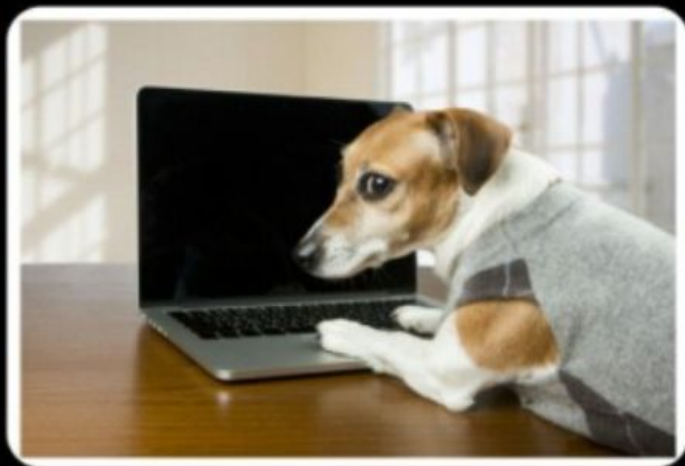
Opening  
Mon  
Tue-Thur  
Fri-Sat  
Sunday



# Два состояния каждого программиста



**Я гениален**



**Без понятия,  
что я делаю**

A wide-angle shot of a large, empty theater. The stage is at the far end, flanked by dark curtains. The ceiling is high and filled with numerous stage lights, some of which are illuminated. The audience seating consists of rows of light-colored seats, with some sections of darker seats visible on the sides. The word 'Демо' is overlaid in large, white, bold Cyrillic letters in the center of the image.

# Демо

<https://drive.google.com/file/d/18ET1Us-SicId65a-4UOLeuwyPnMG4p0/view?usp=sharing>



**Не верю**

## Аргументы против

- писать тесты долго, на это нет времени
- баг фиксом и дебагом никого не испугать
- команда / тим лид это не поддержит
- в реальности это сложно и непонятно
- непонятно, зачем это мне, это работа других
- TDD работает не везде
- есть то, что нельзя протестировать
- не все проблемы можно решить тестами

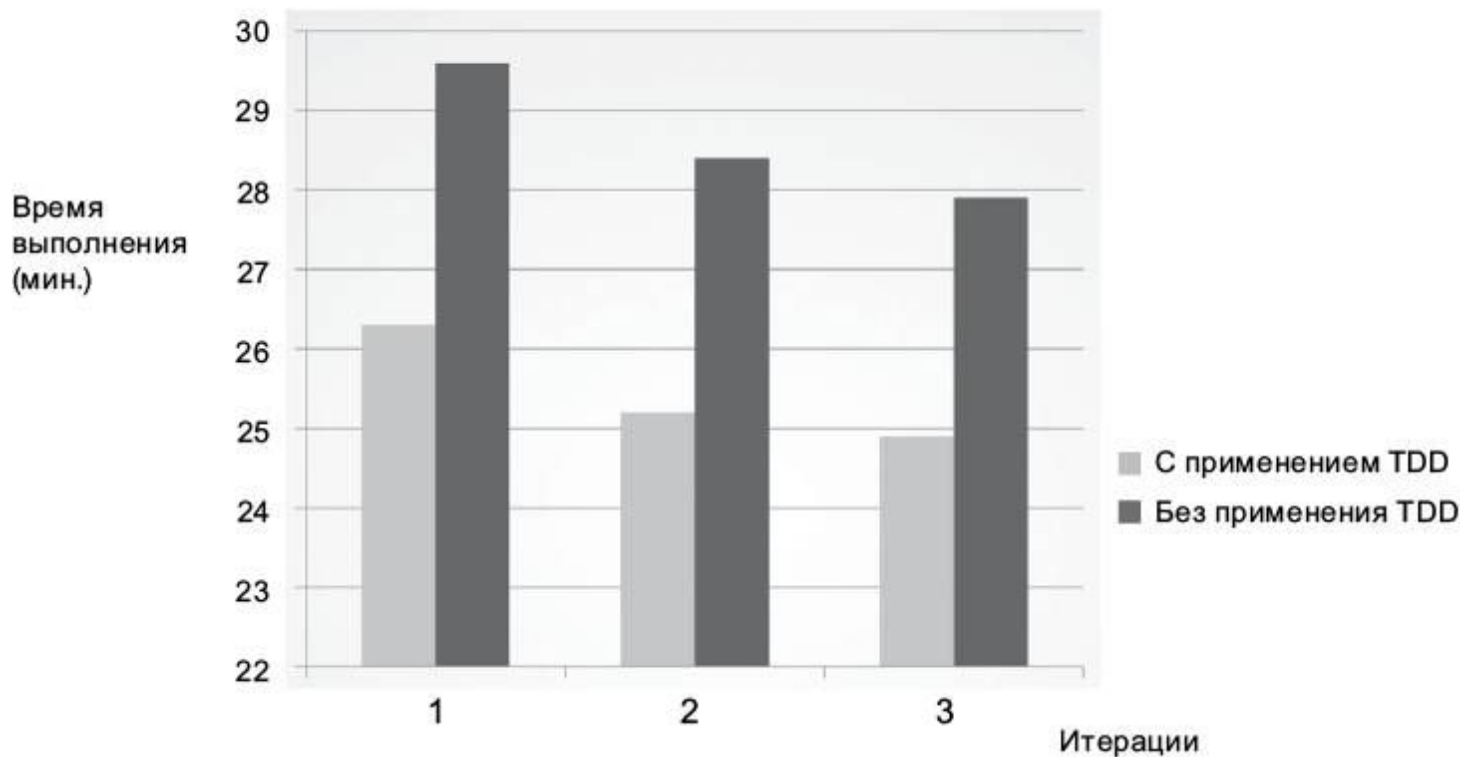
## Аргументы против

→ писать тесты долго, на это нет времени

**Не верю**

# TDD - это быстро

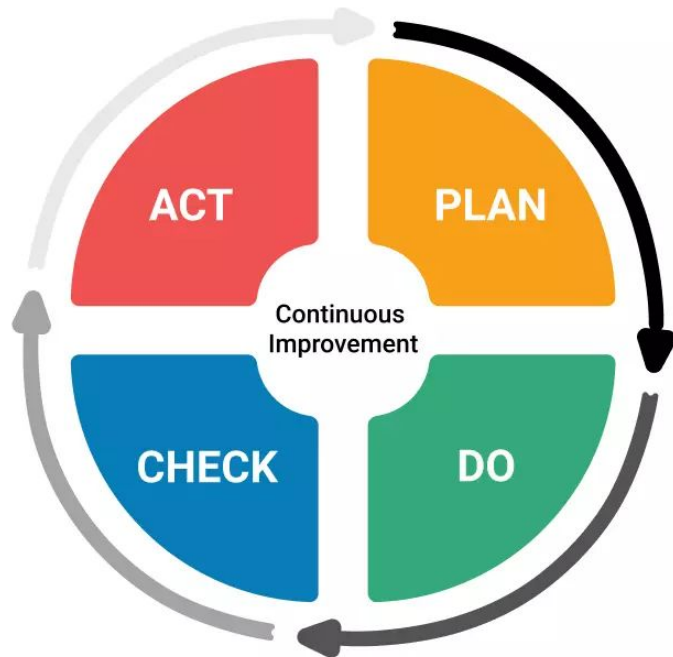
Упражнение по преобразованию в римские числа





# TDD и бизнес

- TDD ускоряет команду с точки зрения Time To Market (время от идеи до поставки)
- TDD повышает техническую гибкость, а именно она ограничивает организационную гибкость





**Не верю**

## Аргументы против

→ баг фиксом и дебагом никого не испугать

**Иногда лучше остаться спать дома в пн,**

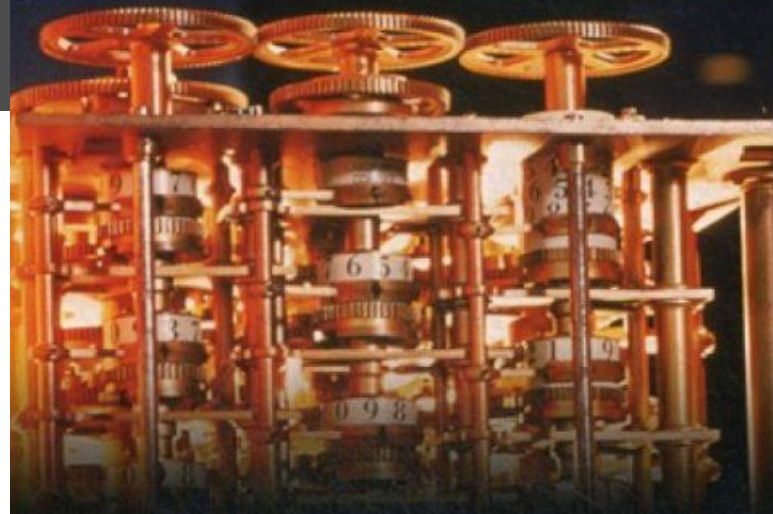
**чем провести всю неделю отлаживая  
написанный в пн код**

# TDD и Легаси

Легаси - это?

- Ценный код, который сложно понять и страшно менять
- Майкл Физерс: “Для меня легаси - это просто код без тестов”

СЕРИЯ КНИГ РОБЕРТА К. МАРТИНА



## ЭФФЕКТИВНАЯ РАБОТА С УНАСЛЕДОВАННЫМ КОДОМ

МАЙКЛ К. ФИЗЕРС

## Аргументы против

→ команда / тим лид это не поддержит

**Не верю**

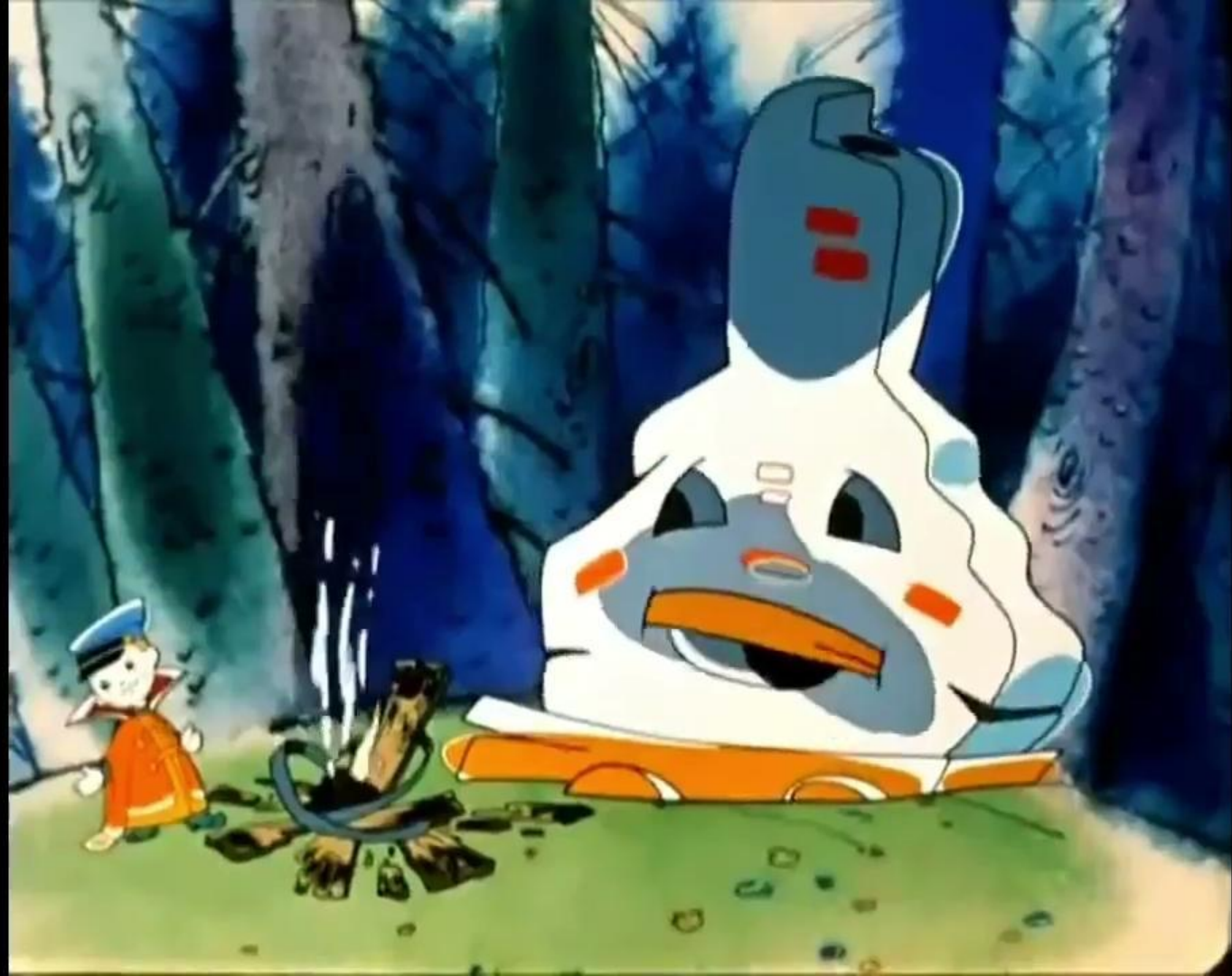


# TDD - навык отдельного разработчика



“Я не великий программист,  
я просто хороший программист с замечательными привычками”

**Кент Бек**





**Не верю**

## Аргументы против

- в реальности это сложно и непонятно
- непонятно, зачем это мне, это работа других

# TDD - другой стиль

- Нужно время на обучение
- Сначала будет медленнее
- Есть риск нахвататься вредных привычек
- Есть риск сдать слишком рано,  
не поняв преимуществ



**ожидания**



**реальность**



**Нельзя просто так взять**

**и не написать тест**





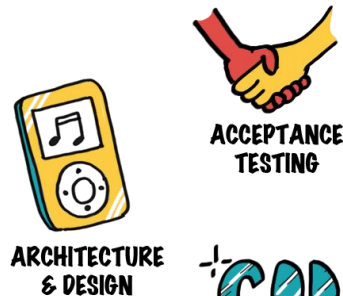
**Не верю**

## Аргументы против

- есть то, что нельзя протестировать
- тесты не решат всех проблем

# Техническое совершенство

- TDD поощряет слабую связанность, простоту, гибкое конфигурирование
- TDD накажет, если не уделять внимание дизайну
- TDD недостаточно: паттерны, SOLID принципы, принципы дизайна, GRASP, object calisthenics и т.д.





**Не верю**

## Аргументы против

- TDD работает не везде
- есть то, что нельзя протестировать

Да, TDD не серебряная пуля или  
волшебная таблетка

**Тесты не будут падать**

**Если вы не будете их писать**



A wide-angle shot of a large, empty theater. The stage is at the far end, flanked by dark curtains. The audience seating consists of many rows of light-colored seats, with some rows of darker seats at the very front. The ceiling is high and dark, with numerous bright spotlights hanging from it. The word 'Демо' is written in large, white, bold Cyrillic letters across the center of the image.

# Демо

<https://drive.google.com/file/d/1dT7MYioSMhEG1Zn0rrMIE6aSlx4PP6S1/view?usp=sharing>



# Что такое хороший тест?

- Понятные имена e.g. Given (Дано) When (Когда) Then (Тогда)
- Arrange (задать данные) / Act (выполнить действие) / Assert (сравнить результат)
- Одна проверка / одна концепция на тест
- FIRST
  - ◆ Fast - Быстрота
  - ◆ Independent - Независимость
  - ◆ Repeatable - Повторяемость
  - ◆ Self-Validating - Очевидность
  - ◆ Timely - Своевременность



# Что можно делать дальше?

- Практиковать на упражнениях
- Практиковать в реальных задачах
- TDD бывает недостаточно, добавлять другие практики для вашего контекста
- Учить hot keys в среде разработки, потому что с TDD рефакторинг станет постоянным


# Agile Technical Practices Distilled

A learning journey in technical practices and principles of software design



Pedro M. Santos, Marco Consolaro  
and Alessandro Di Gioia

**Packt**  
www.packt.com

A close-up of Bruce Lee in a martial arts pose, wearing his iconic yellow jumpsuit. He is looking slightly to the side with a determined expression, and his hands are raised in a defensive or offensive stance. The background is dark and out of focus.

**Я не боюсь того, кто изучает 10000 ударов**

**Я боюсь того, кто изучает TDD**