

TDD or not TDD?

# Светлана Кривенко

- ★ ~ 4,5 года как Scrum Master (Raiffeisen, Tutu.ru)
- ★ ~ 11 лет как Java Full Stack / Team Lead в прошлом (Outsource)
- ★ со-организатор TDD практик  
онлайн (Группа Technical Excellence)  
оффлайн (Клуб Московских Программистов)



# TDD - подход, которому больше 20 лет

Кент Бек - пере-открыватель TDD

- 1999: Экстремальное программирование
- 2000: Разработка через тестирование

2001: Agile Манифест

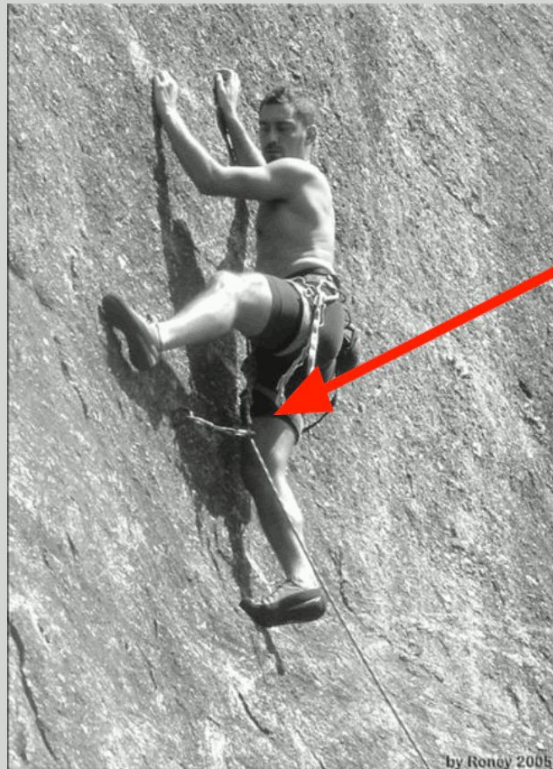


# TDD на минималках

1. **Red** - падающий тест
2. **Green** - минимальный работающий код
3. Refactor - чистый код

Зачем нужен?

- ★ ускоряет разработку функционала
- ★ помогает со страхом при разработке
- ★ помогает не добавлять легаси
- ★ помогает меньше дебажить



Unit Test

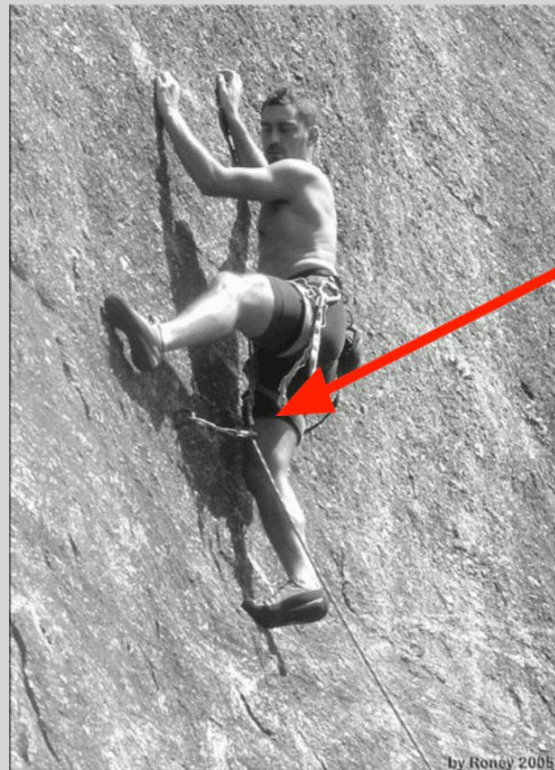
by Roney 2005

# Unit Test на минималках

Тест - для проверки функциональности,  
а не покрытия кода или поиска ошибок

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

Секрет тестов - немного позже



Unit Test

by Roney 2005

# TDD на минималках



**VERSUS**

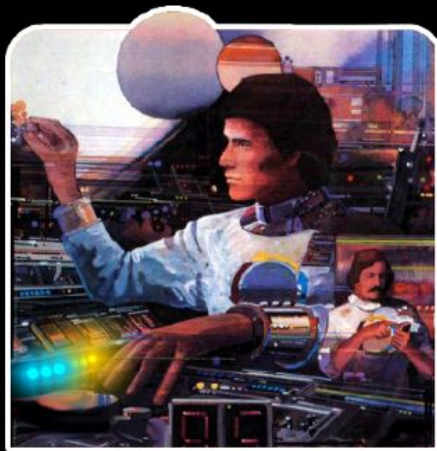
How it feels like to run your code after writing hundreds of lines without testing



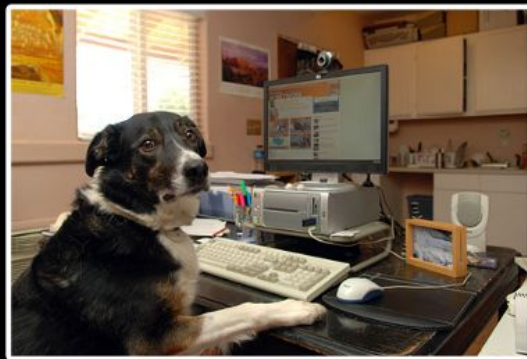


# TDD на минималках

## THE TWO STATES OF EVERY PROGRAMMER



**I AM A GOD.**



**I HAVE NO IDEA  
WHAT I'M DOING.**

# TDD Kata

<https://drive.google.com/file/d/18ET1Us-Siclid65a-4UOLeuwvPnMG4p0/view?usp=sharing>



## TDD - навык отдельного разработчика



“I'm not a great programmer; I'm just a good programmer with great habits”

**KENT BECK**

# Tech Excellence Video

<https://drive.google.com/file/d/183CGleA1dYwHPhFekdZels-pUlihZSKN/view?usp=sharing>

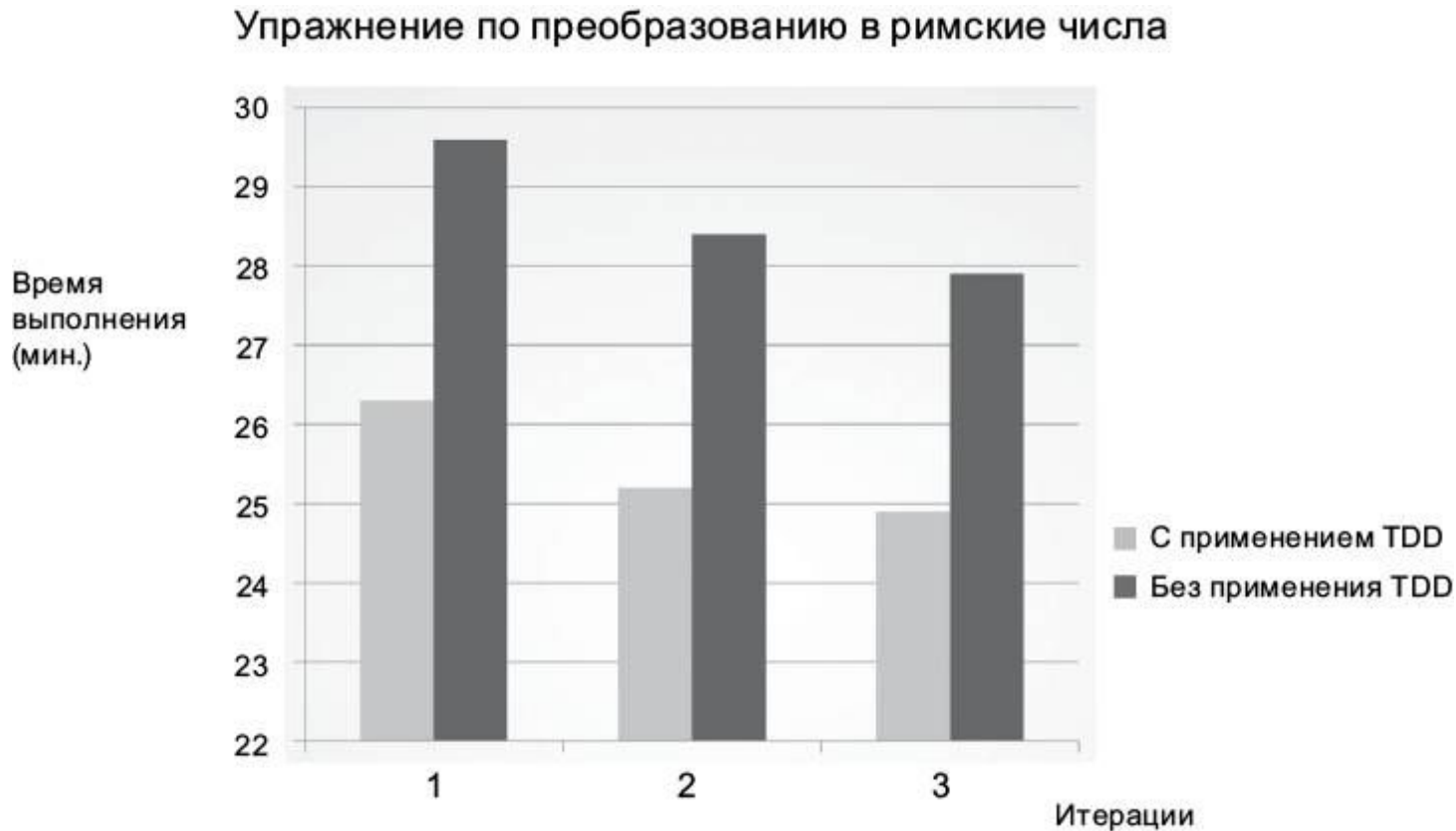
# TDD - как сноуборд или сёрфинг

Выработка нового стиля программирования

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



# TDD - на самом деле быстро

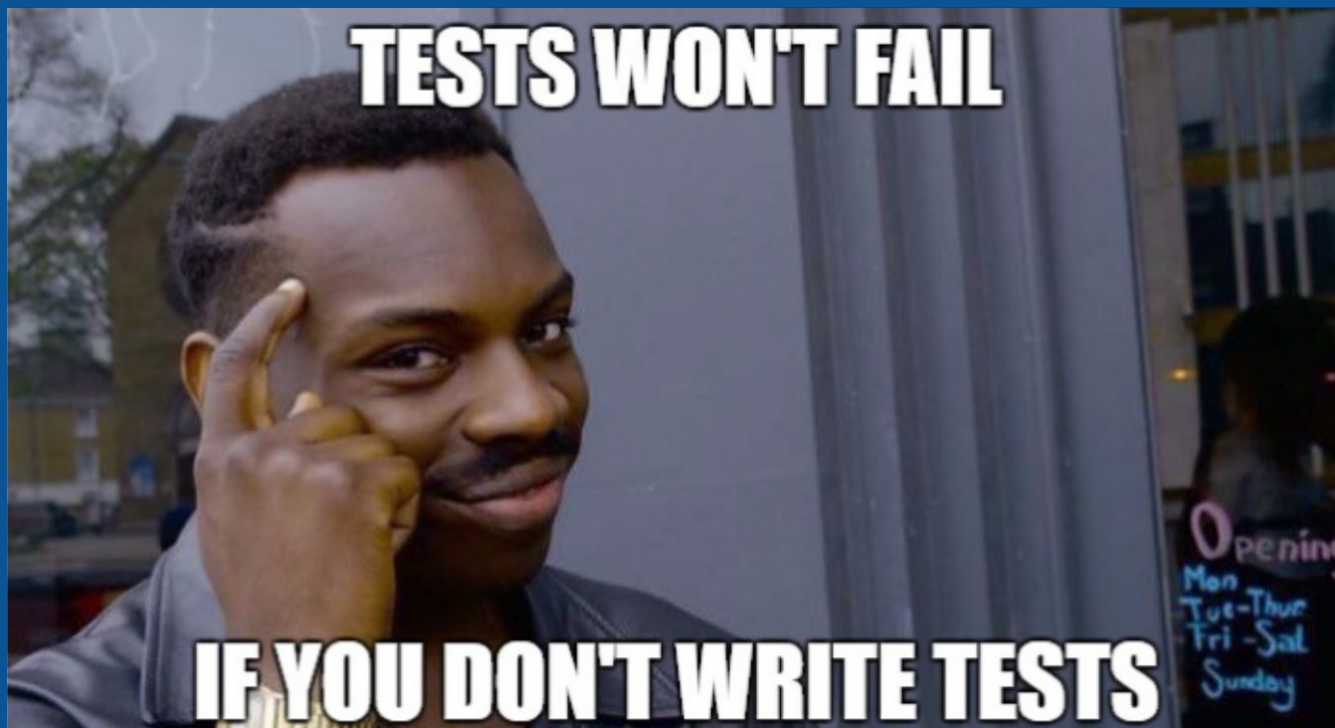


# TDD - не серебряная пуля

TDD подходит не всегда



TDD - не серебряная пуля



# TDD Kata

<https://drive.google.com/file/d/1dT7MYioSMhEG1Zn0rrMIE6aSIx4PP6S1/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



# TDD и Архитектура

TDD поощряет слабую связанность, простоту, гибкое конфигурирование

SOLID - принципы:

- ★ единственной ответственности
- ★ открытости / закрытости
- ★ подстановки Б. Лисков
- ★ разделения интерфейсов
- ★ инверсии зависимостей

Роберт Мартин (XP, Agile)



# TDD и Архитектура

Секрет - не в написании теста,  
а в написании кода,  
пригодного для тестирования



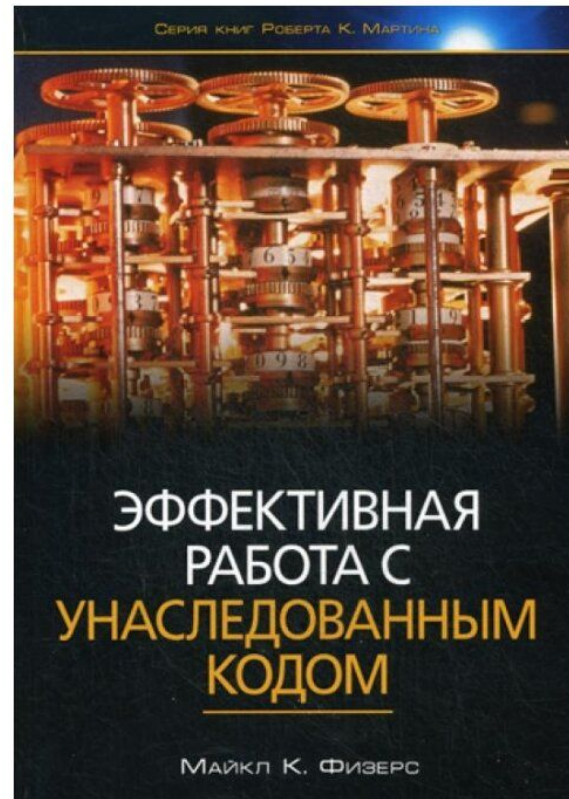
# TDD и Легаси

Легаси - это?

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

P.S. Помните страховочный крюк?

TDD помогает не упасть ниже страховки



# TDD и Бизнес

- ★ TDD ускоряет команду с точки зрения TTM (время от идеи до поставки)
- ★ TDD повышает техническую гибкость, а именно она ограничивает орг. гибкость



# TDD Prepared Kata Video

# Практика TDD

ОНЛАЙН - Technical Excellence



<https://technical-excellence.ru/>

<https://t.me/technicalexcellenceru>



ОФФЛАЙН - Мск программисты



<https://prog.msk.ru/>

<https://t.me/progmsk>





# Практика TDD

Я не боюсь того, кто изучает  
10000 различных ударов.  
Я боюсь того, кто изучает один  
удар 10000 раз.

