

Test-driven Development: The Big Picture

WHAT IS TEST-DRIVEN DEVELOPMENT?



Course Outline



Software development challenges

What is test-driven development?

Different ways of testing applications

Test-driven development in action

Strategies/techniques for testing code

Test-driven development gotchas

Prerequisites

No TDD experience required



*Experience writing code
(C#, Java, some JavaScript)*

Target Audience

Software Developers



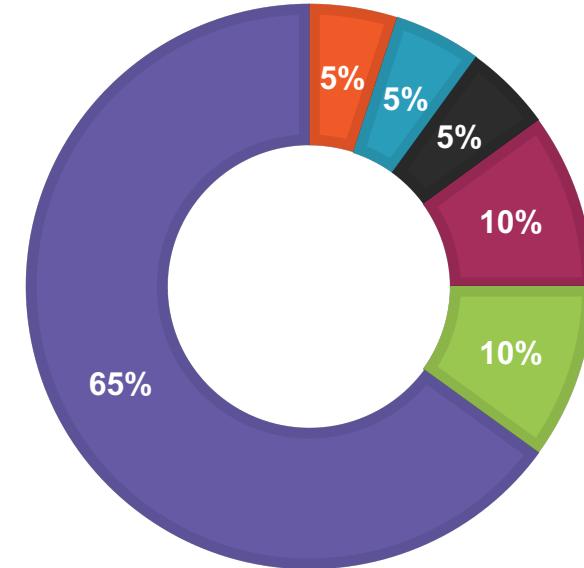
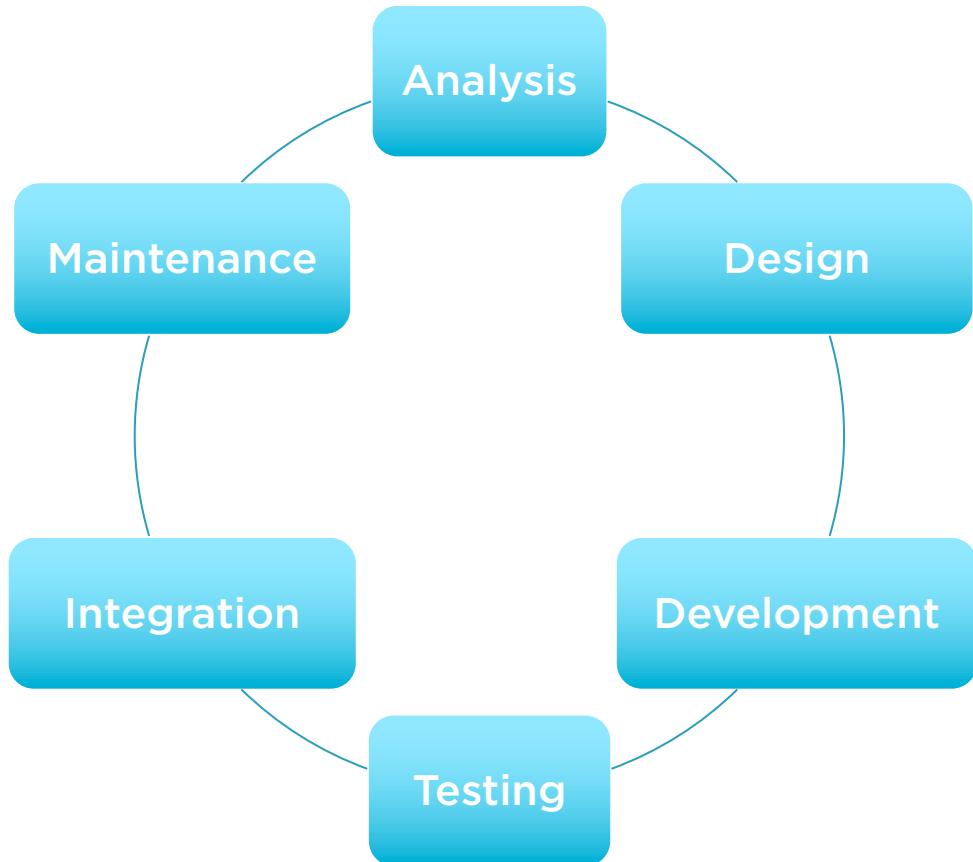
Software Testers



Software Managers



Costs of Developing Software



- Analysis
- Design
- Development
- Testing
- Integration
- Maintenance

Maintenance accounts for **65%** of all software development costs!

Maintenance Challenges



Code Entropy

- *Brittle over time*
- *Increased rigidity*



Isolated Ownership

- *Coder-owned silos*
- *Lack of team empowerment*

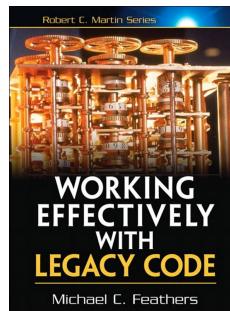


Infrequent Validation

- *Regression-prone*
- *High risk of code changes*

Legacy Code.

- 1) “source code *inherited from someone else*”
- 2) “source code *inherited from an older version of the software*”
- 3) “code *without tests*”



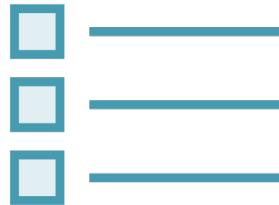
What Is Test-Driven Development?

Test.

“A procedure intended to establish the quality, performance, or reliability of something, especially before it is taken into widespread use.”



**Satisfies
requirements**



**Responds correctly
to all input**



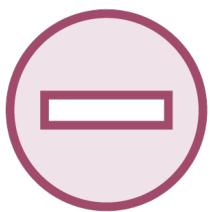
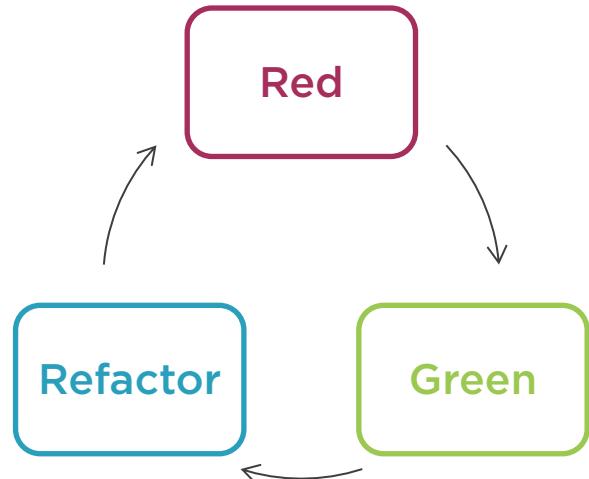
**Acceptable
performance**

Test-Driven Development.

“A **software development process** that relies on the repetition of a **very short development cycle**: requirements are turned into very specific test cases, then the software is improved to pass the new tests, only.” - Wikipedia

“**Red. Green. Refactor.**”

Red. Green. Refactor.



Red

Write test that fails



Green

Make test pass



Refactor

Refactor/cleanup code

Is Test-Driven Development a “New Thing”?

Test-Driven Development History



"Simple Smalltalk Testing"
SUnit (1994)
Kent Beck

1960s

1989

eXtreme Programming
Kent Beck

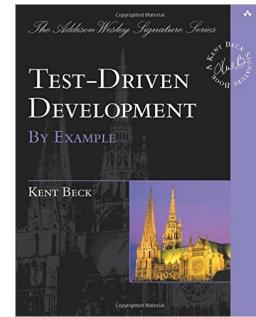
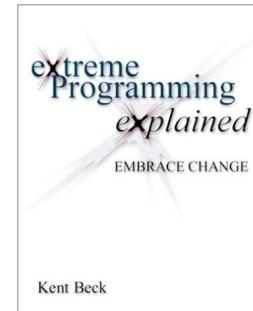
1999

TDD: By Example
Kent Beck

2002

JUnit

*Kent Beck
Erich Gamma*



Why Practice Test-Driven Development?

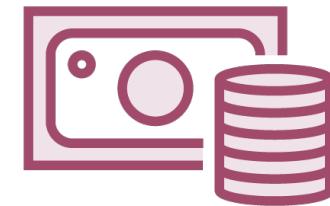
Business Benefits



**Requirements
Verification**



**Regression
Catching**



**Lower
Maintenance Costs**

Developer Benefits



Design-First



Momentum



Confidence

Code Kata.

Kata – “A system of individual training exercises for practitioners of karate and other martial arts.”



Practice



Learning

Keeping Focused on the Customer

What Was Spec'd



CC: DanielCD
<http://bit.ly/2jyS1sD>

What Was Delivered



CC: Public Domain

What Customer Wanted



CC: Rudolf Stricker
<http://bit.ly/2jec9AK>

Summary



Software development challenges

What is Test-Driven Development?

Different ways of testing applications

Test-Driven Development in action

Strategies/techniques for testing code

Test-Driven Development gotchas

Jason Olson

SENIOR SOFTWARE
ENGINEER

@jolson88

www.codingcoda.com