Test Driven Development

Hello!

I am Chandler Severson

Welcome to **Test Driven Development**

CS469 - Systems Analysis - Fall 2016

~50%

The percentage of time ⊠ and money ⊠ spent on testing software during a software project.

Why write Tests?

Human Error

Software Engineers make mistakes all the time! Whether it be from writing new code or refactoring old stuff, these mistakes can lead to nasty consequences.

$$$$$

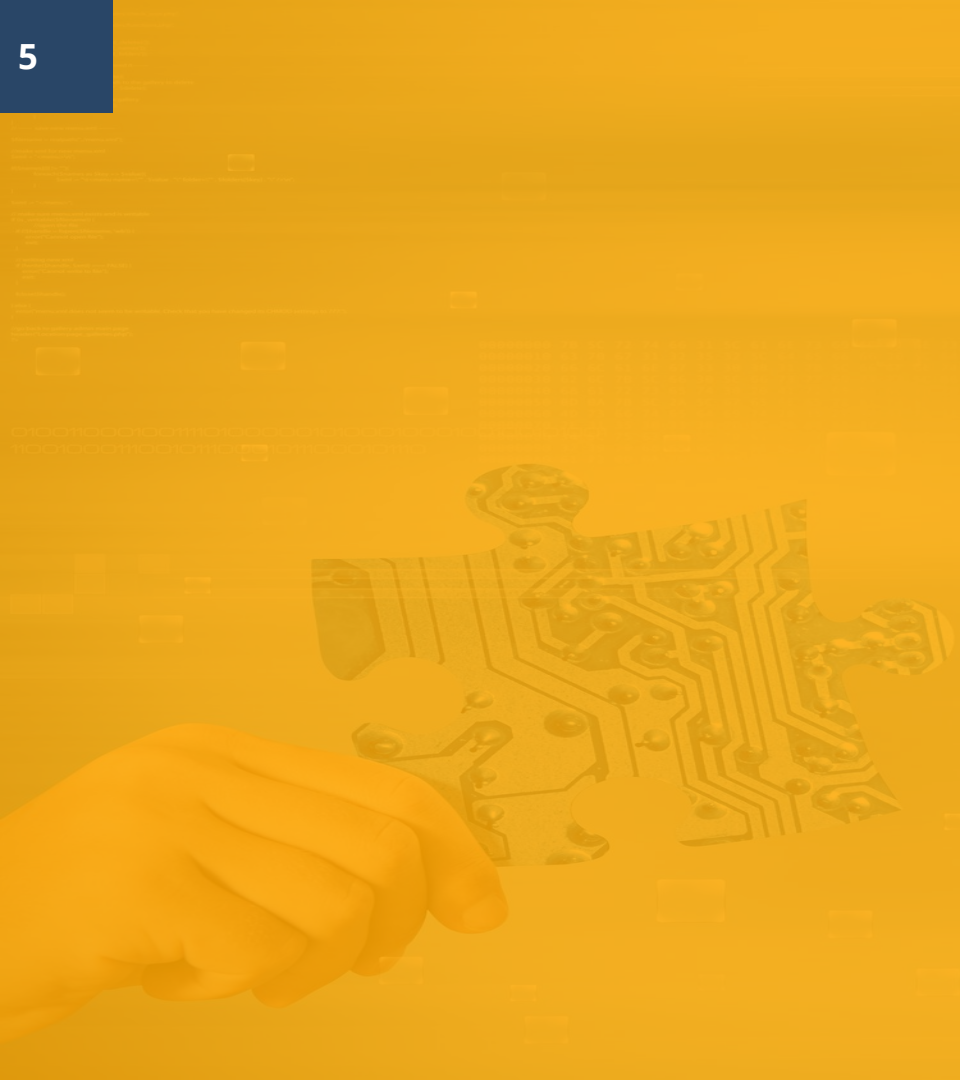
Do you like money? How about your company?
You will lose a lot of it if your software breaks in production.

Customer Love

Your customers will love the reliability of your software and will be satisfied about its great quality.

A Better Product ✔

Your software will be less likely to break, it will run faster, and developers will be able to modify it without worrying that they are breaking anything!



What is Test Driven Development?

Let’s explore the basics of TDD.

1960

The year where the first Test Driven Development Prototype was created.

Extreme Programming

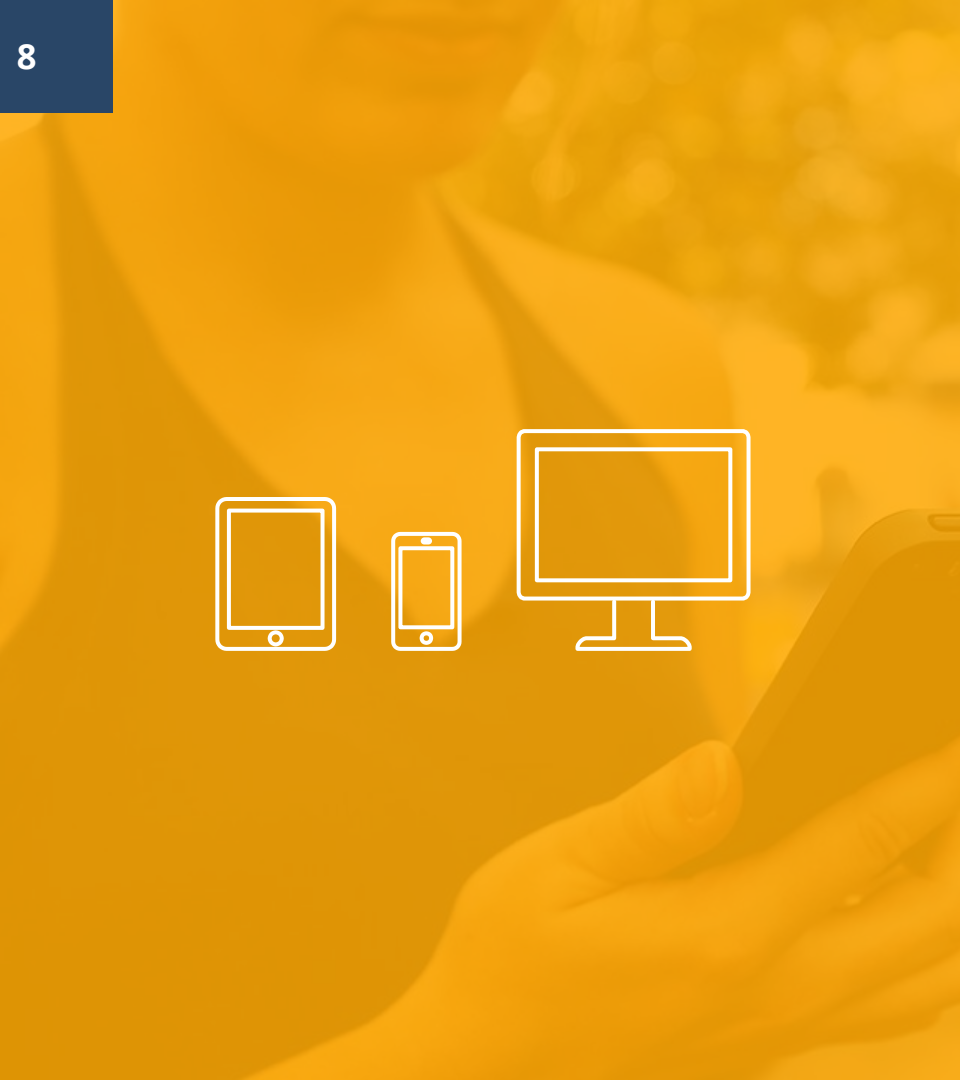
An Agile development methodology. Created the modern ‘Test-First’ development concept.

Kent Beck

The ‘Creator’ of modern Test Driven Development & Extreme Programming

Specification over Validation

- Think through requirements and/or design before writing any code.
- Write Clean code that Works.

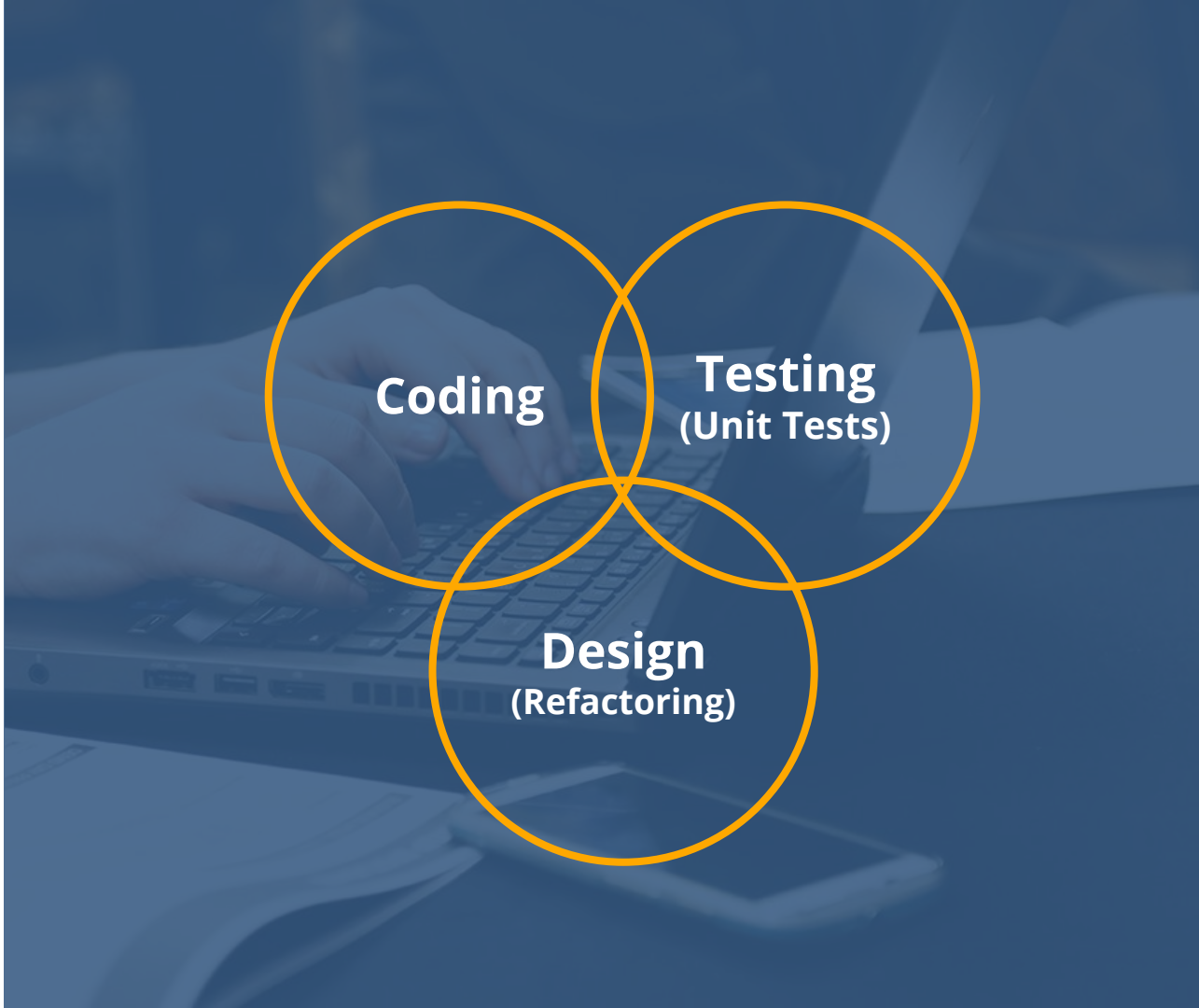


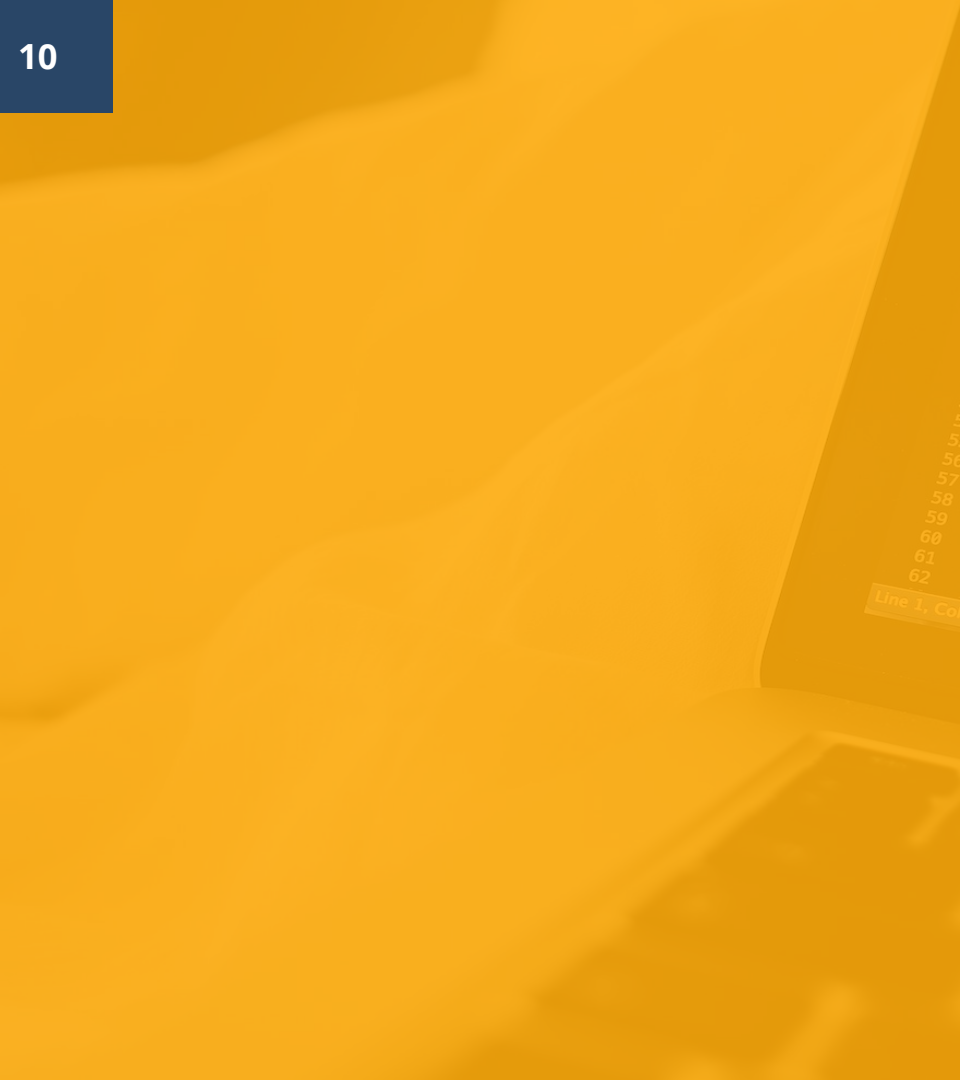
What is TDD, though?

- A **specification** technique.
- Ensures your code is thoroughly tested.
- Brings clear success metrics to development.

Test Driven Development will help increase your **confidence** that your system meets its requirements.

Activities that Interweave in TDD



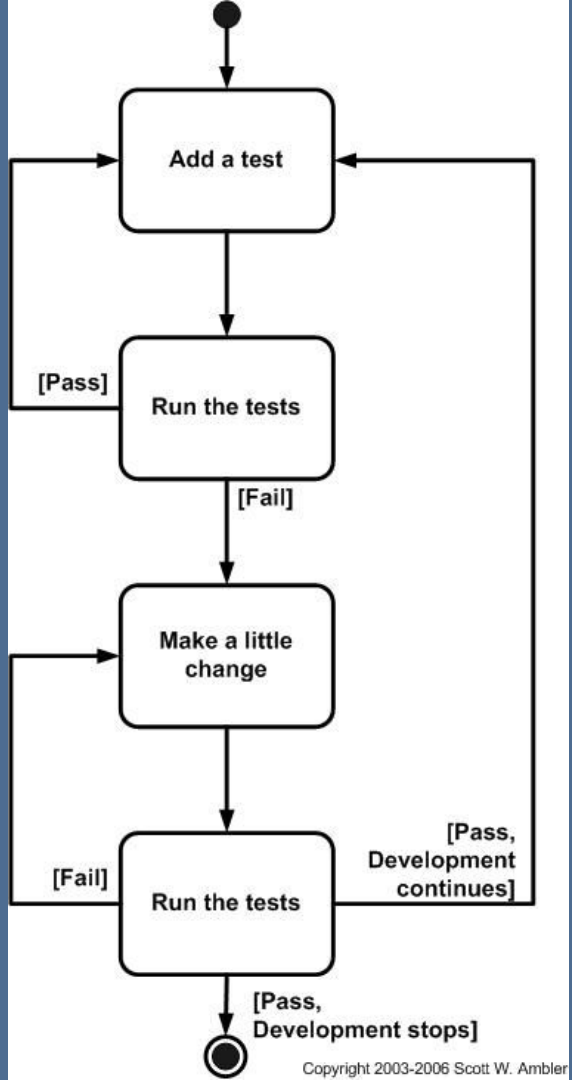


Test Driven Development

How does it actually work?

Test First Development

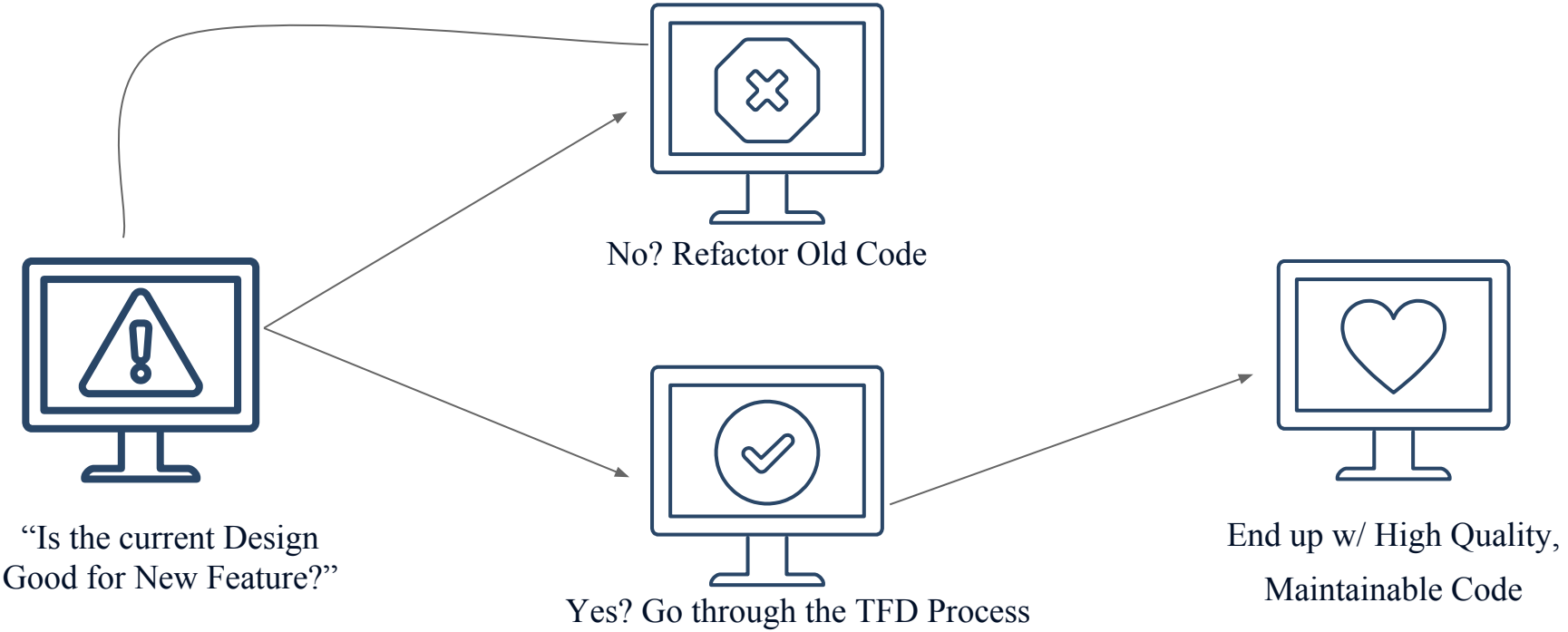
- The main building block of Test Driven Development
- Write tests before implementing functional code.



TDD = TFD + Refactoring

With this approach, you always **improve the quality** of your code and make it **easier to work with** later.

The TDD Process – Before Writing a Test



Types of Test Driven Development

Let’s explore the two types of TDD.

Two Types of Test Driven Development

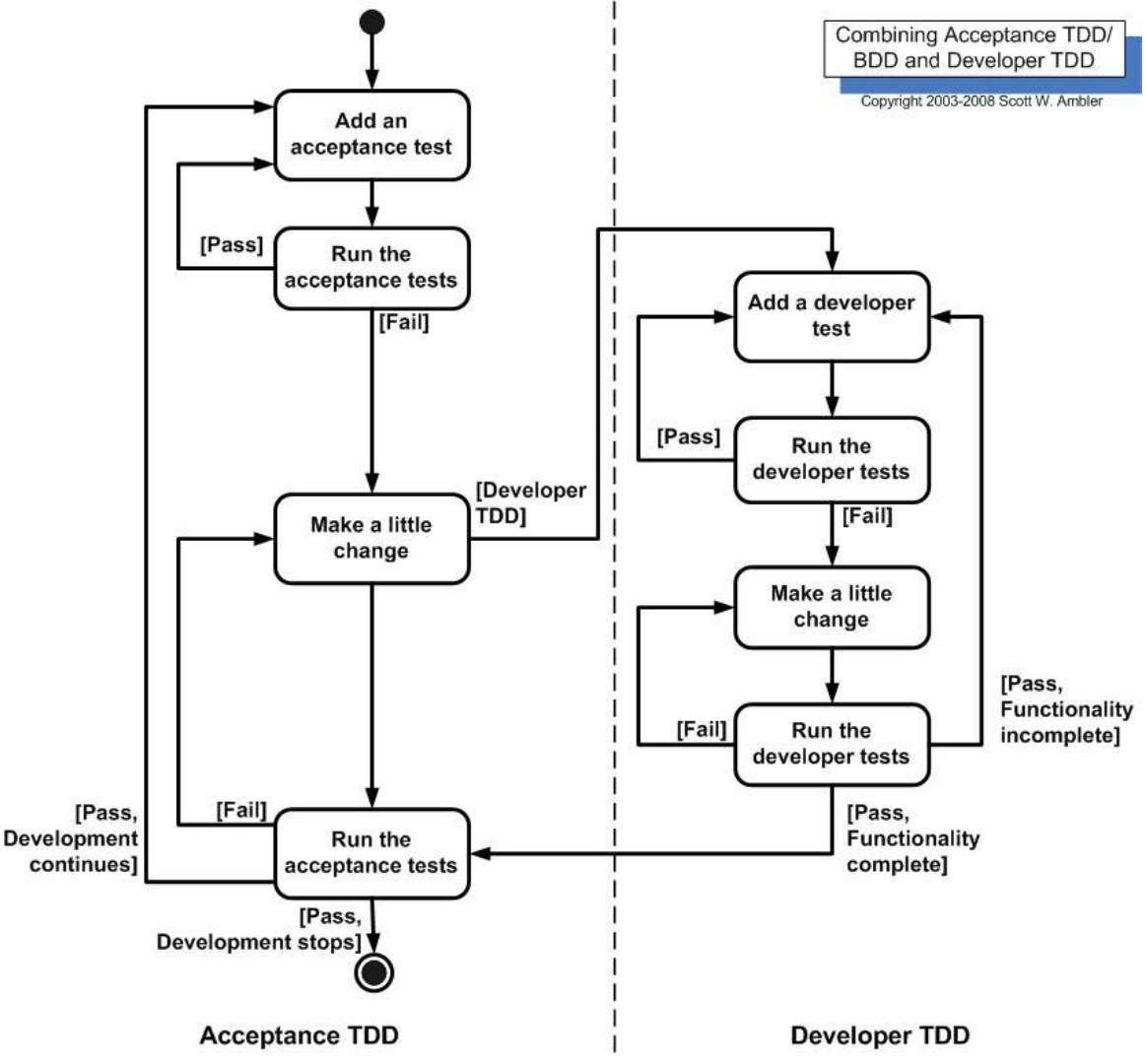
Acceptance TDD

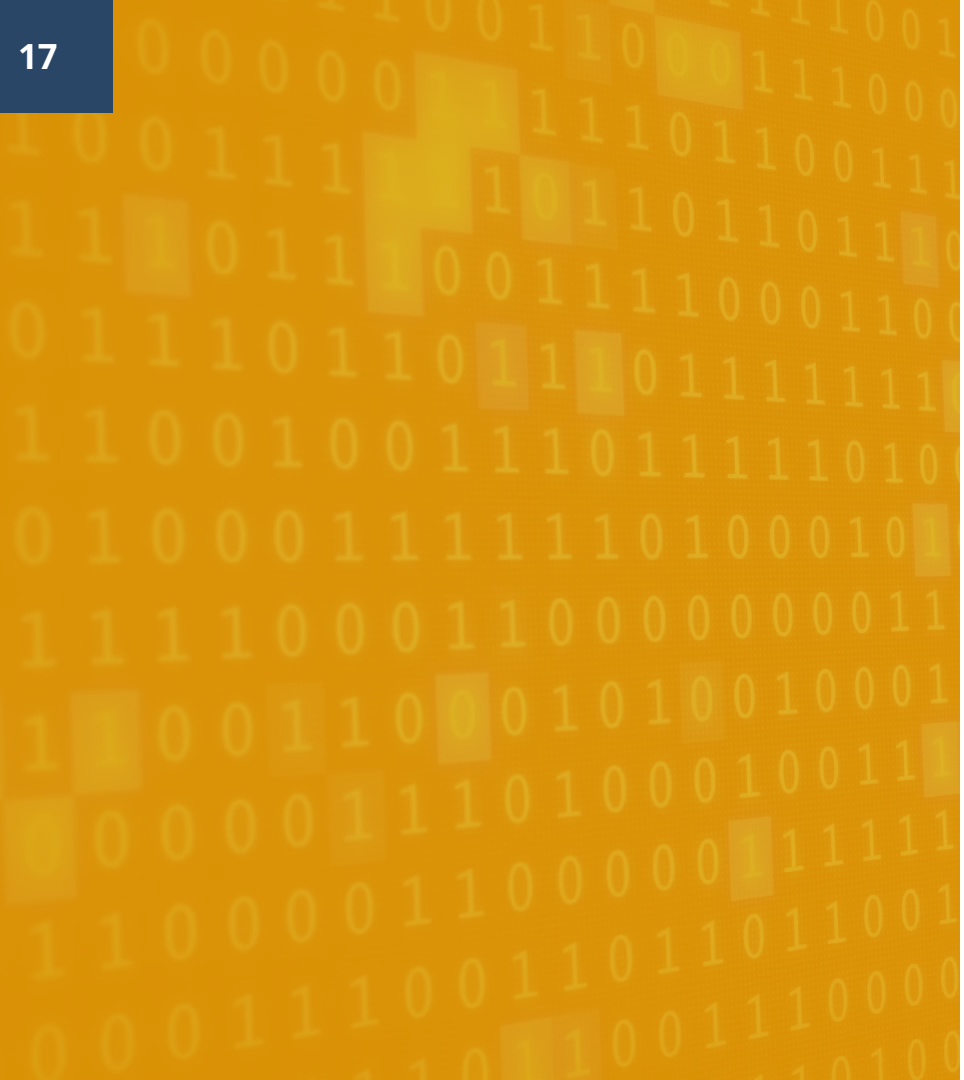
- Write tests to see if your functional code meets requirements.
- These usually test **full user stories** or use cases.

Developer TDD

- Write **small** and **specific** tests for small and specific pieces of code.
- Tests created during Developer TDD are often called “Unit Tests”

Acceptance & Developer TDD Interaction





Using Test Driven Development?

Benefits and Pitfalls of the TDD process.

Benefits of Test Driven Development

- Code Defects go ⇩
- Effort (in later stages of project) goes ⇩
- Code Quality goes ⇧

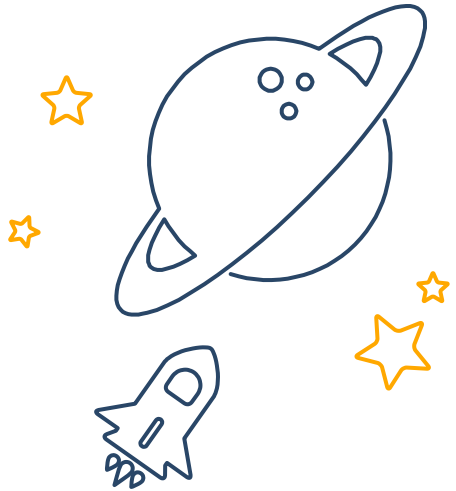
Common TDD Mistakes

Individuals:

- Infrequent Testing
- Excessively Large Tests
- Trivial Tests

Teams:

- Partial Adoption
- Poor Maintenance
- Abandonment



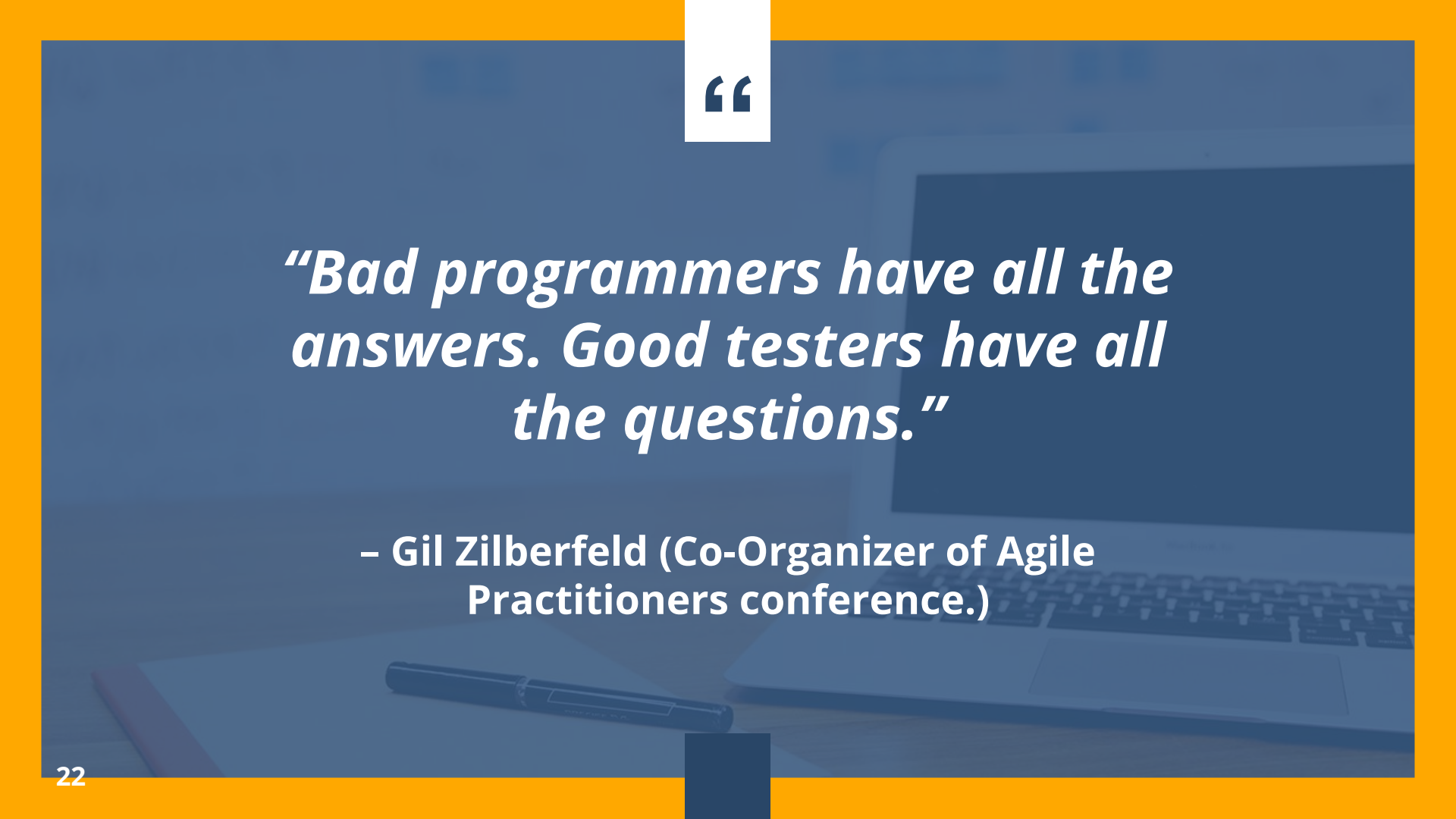
Test Driven Development

It’s magically amazing.

Thanks!

Any questions?

Chandler Severson - Test Driven Development



“Bad programmers have all the answers. Good testers have all the questions.”

– Gil Zilberfeld (Co-Organizer of Agile Practitioners conference.)

References

Ambler, S. W. (n.d.). Acceptance/Customer Tests as Requirements Artifacts: An Agile Introduction. Retrieved November 20, 2016, from http://www.agilemodeling.com/artifacts/acceptanceTests.htm

Dustin, E., Garrett, T., & Gauf, B. (2009). Implementing automated software testing: How to save time and lower costs while raising quality. Retrieved November 20, 2016, from http://www.informit.com/articles/article.aspx?p=1332758&seqNum=3

Introduction to Test Driven Development (TDD). (n.d.). Retrieved November 20, 2016, from http://agiledata.org/essays/tdd.html

TDD. (n.d.). Retrieved November 20, 2016, from https://www.agilealliance.org/glossary/tdd/

Test Driven Development. (n.d.). Retrieved November 20, 2016, from https://www.tutorialspoint.com/software_testing_dictionary/test_driven_development.htm