Getting Started with Unit Testing

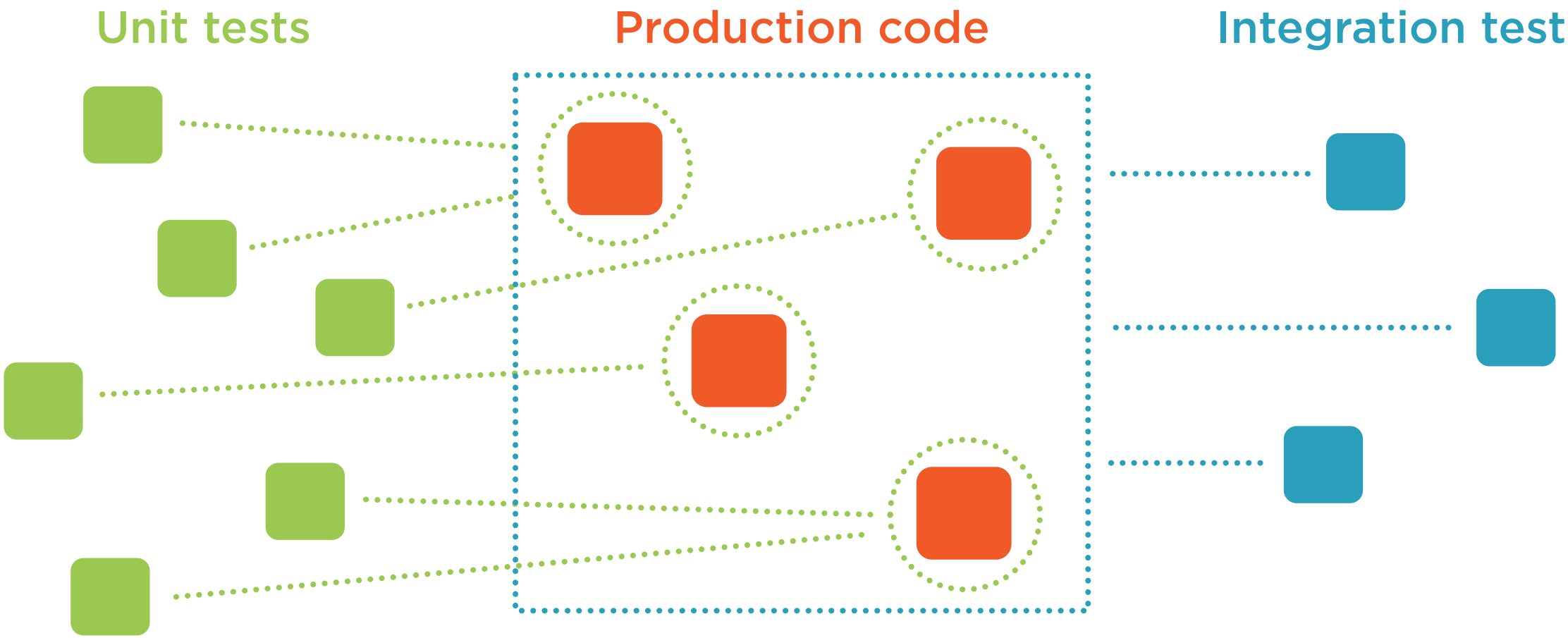
Thomas Claudius Huber
MICROSOFT MVP (WINDOWS DEVELOPMENT)

@thomasclaudiush www.thomasclaudiushuber.com

Module Outline

- Unit Testing Basics
- Write and run Unit Tests in Visual Studio with MSTest and xUnit
- Test Driven Development (TDD) Basics
- Unit Testing in FriendStorage

Unit Testing Basics



Unit Test

A small automated test, coded by a programmer, that verifies whether or not a small piece of production code - a unit - works as expected in isolation.

Tim Ottinger & Jeff Langr

Effective Unit Tests Are F.I.R.S.T.

Fast

to run them frequently

Independent

of each other

Repeatable

in any environment

Self-validating

and so either passing or failing

Timely

written before the production code

Popular Unit Testing Frameworks for .NET

MSTest

Comes with
Visual Studio

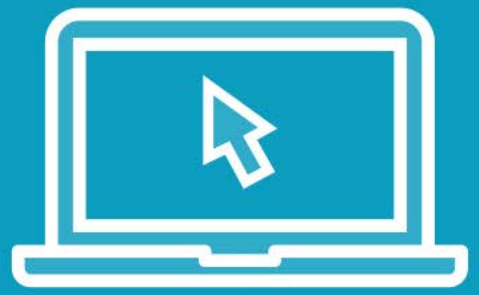
NUnit

A complete rewrite
of JUnit for .NET
http://nunit.org

xUnit

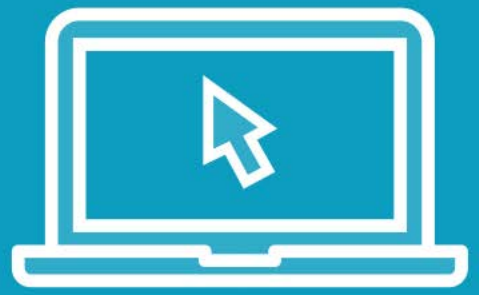
Written by the original
inventor of NUnit v2
http://xunit.github.io

Demo



Write and run a Unit Test with MSTest

Demo



Write and run a Unit Test with xUnit

Use parameters for the test method

Configure xUnit with a .json-file

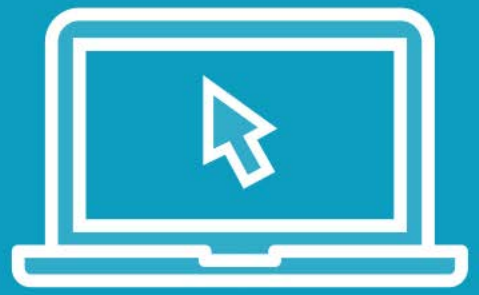
Good Structure of a Unit Test

```
[Fact]
public void AddShouldReturnSumOfValues()
{
    // Arrange
    var calc = new Calculator();

    // Act
    int result = calc.Add(2, 3);

    // Assert
    Assert.Equal(5, result);
}
```

Demo

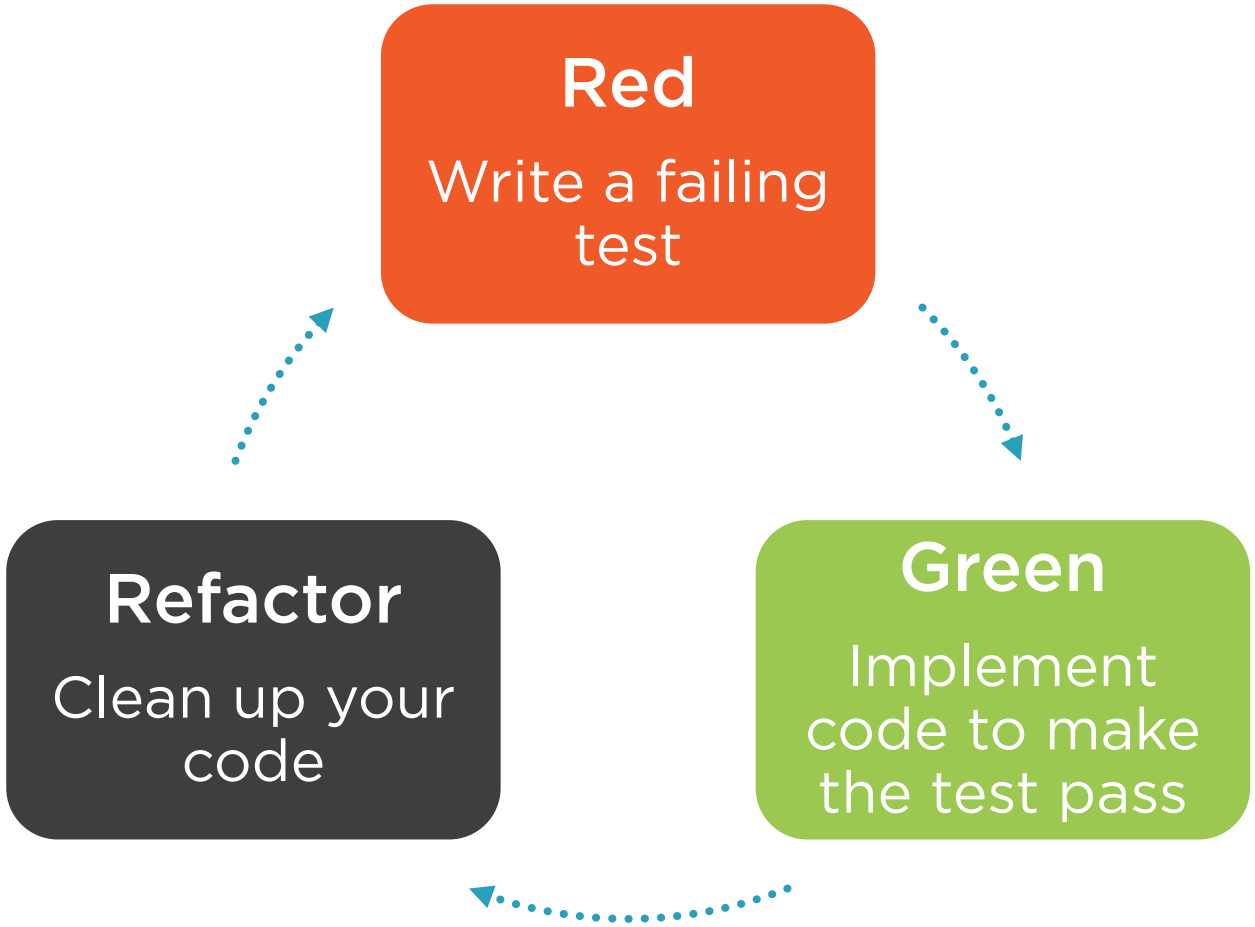


Add a test-project to FriendStorage

Use xUnit as a testing framework

Create a test-class for the NavigationViewModel

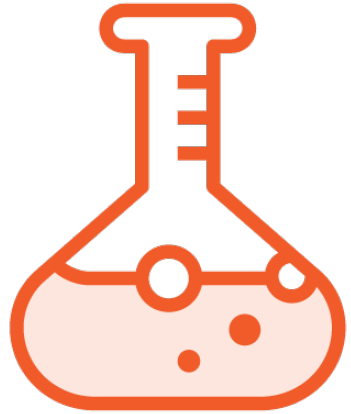
Test Driven Development (TDD) Basics



Using Spikes in TDD

You think about a possible solution to build the feature

You're not sure if your idea will work

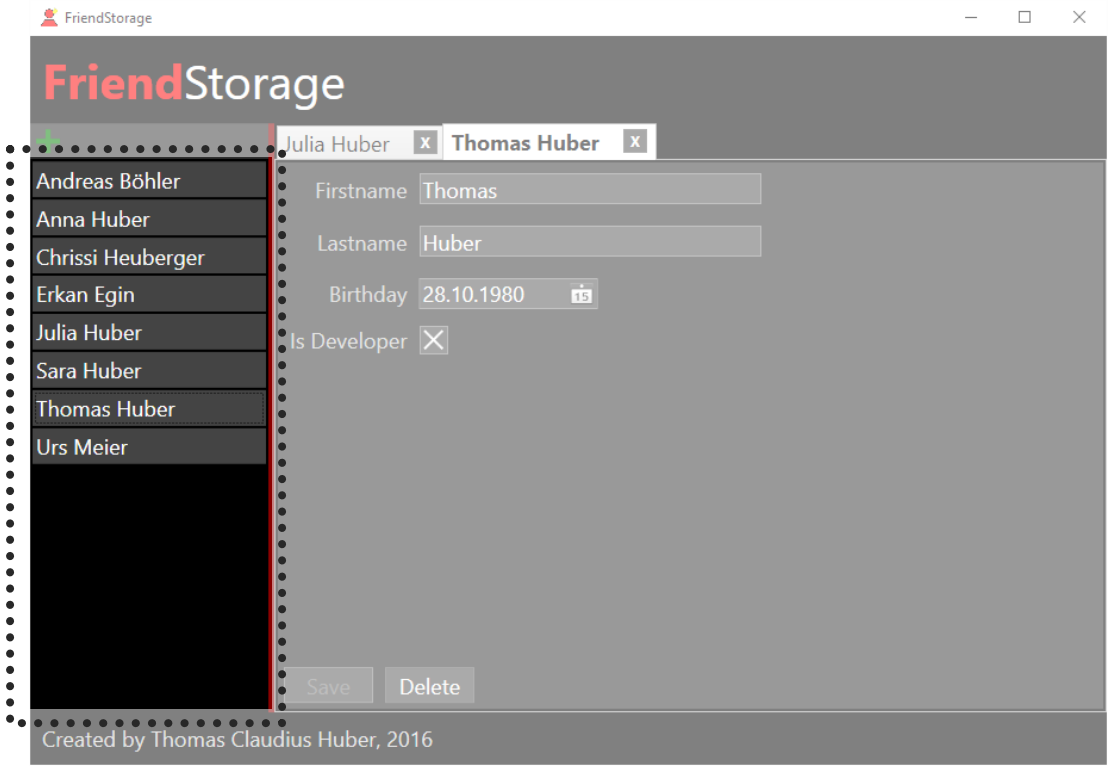


Let's hack the idea down without tests to see if it will work

A spike is an experiment without tests to ensure that your idea will work

Spiking the Navigation of FriendStorage

Spike the loading of the navigation items



Summary

Clean Unit Tests are F.I.R.S.T.

TDD is using Unit Tests to drive the design of your code

Spikes are experiments in TDD