

Creating Unit Tests to Support Business Changes Without Fear



Gill Cleeren

ARCHITECT

@gillcleeren www.snowball.be



Overview



Why we need unit tests

Writing unit tests





A question from Bethany

- I know that my store will evolve quite a lot over time. Won't that cause issues when we need to make a lot of changes to the code?



Why We Need Unit Tests



A unit test is an automated piece of code that invokes a unit of work in the system and then checks a single assumption about the behavior of that unit of work.

From The Art of Unit Testing, Roy Oshero



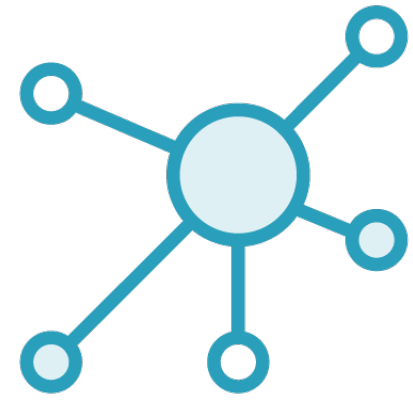
Unit Tests



Block of code



Public API



Isolated and
independent

Unit Tests

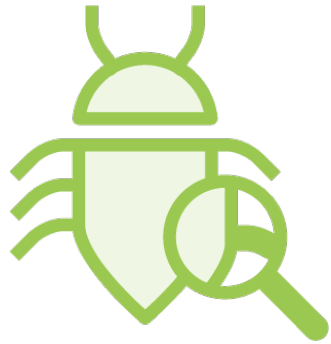
Consistent

Automated

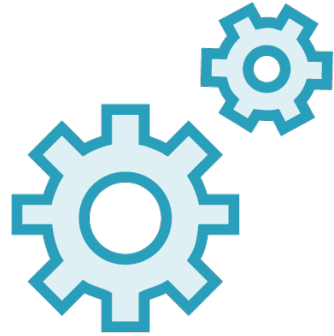
Fast



Why Do We Need Unit Tests?



Find bugs



Change without
fear of breaking
something



Improve quality



Documentation
of the code



Arrange

Act

Assert



A Simple Unit Test

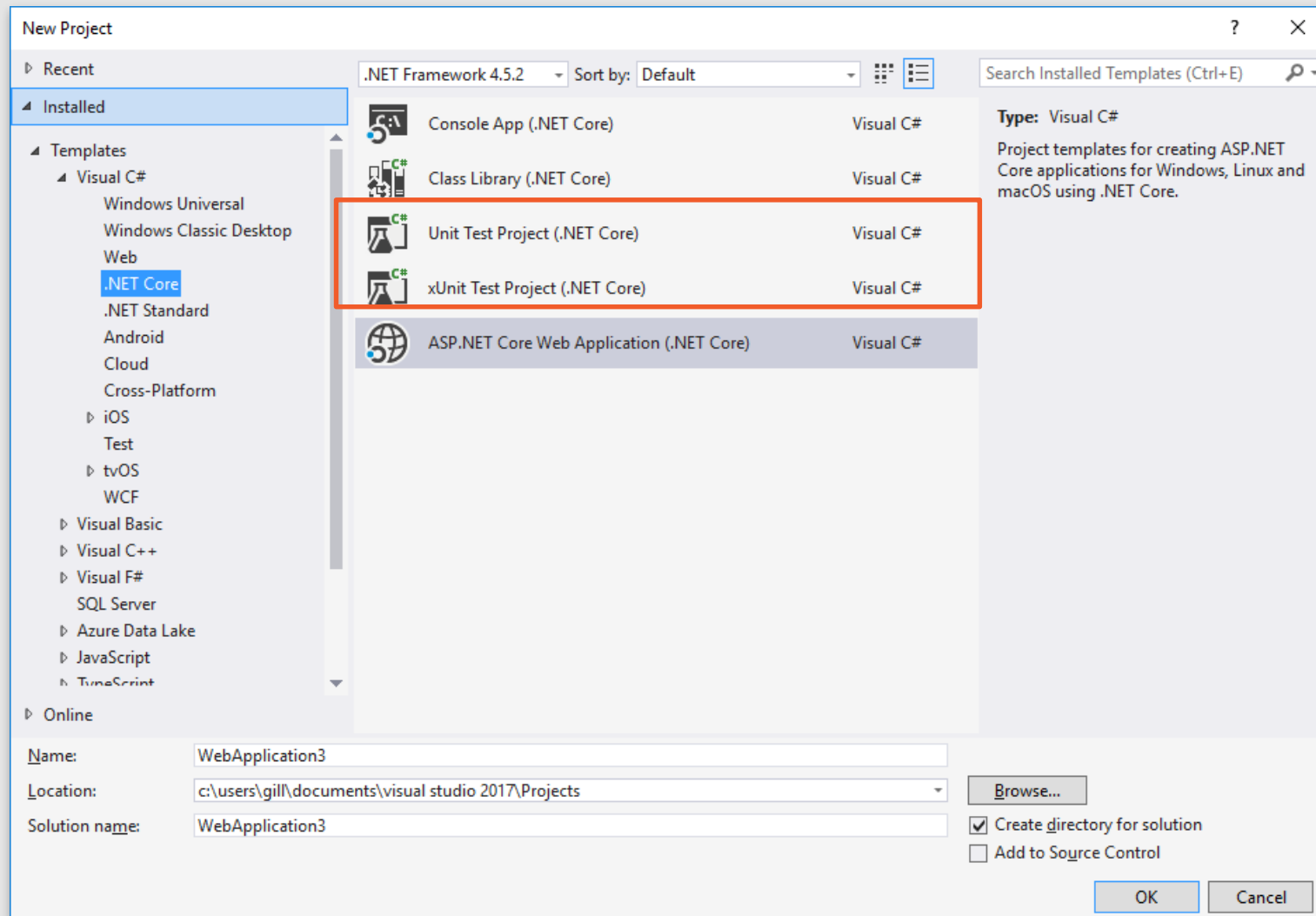
```
public void CanUpdatePiePrice()  
{  
    // Arrange  
    var pie =  
        new Pie { Name = "Sample pie", Price = 12.95M };  
  
    // Act  
    pie.Price = 20M;  
  
    //Assert  
    Assert.Equal(20M, pie.Price);  
}
```



Writing Unit Tests



Unit Testing ASP.NET Core



Demo



Creating unit tests for the controller
using xUnit

Adding mocks using Moq

Parameterizing unit tests



Summary



Unit tests create a safety net around our code

Unit tests can be added on multiple levels in our code





Up next:
Diagnosing application issues

