

# Testing

Stefan KLIKOVITS



**UNIVERSITÉ  
DE GENÈVE**

# Today

- Testing with JUnit
- Test Driven Development (TDD)
- Mocking

# **What you should know (after)**

- Why do we test?
- What levels of testing exist?
- What is a double/mock/stub/fake/spy/... ?
- Coverage (what is it and what does it say?)

# Why do we test?

- Testing  $\neq$  Validation & Verification
- Increase confidence in our code
- Save time (= money)
- Can we test everything?

# Levels of testing?

- Unit



- Integration

- System

- Acceptance

# JUnit 4

- `@Test`
- `@Before`, `@After`
- `@BeforeClass`, `@AfterClass`
- `@Test(timeout = 500)`
- `@Test(expected = MyException.class)`

# Maven Quickstart

```
~ $ mvn archetype:generate  
-Dfilter=maven-archetype-quickstart
```

- OR: Eclipse...
- PS: switch to JUnit 4.12

# Exercise

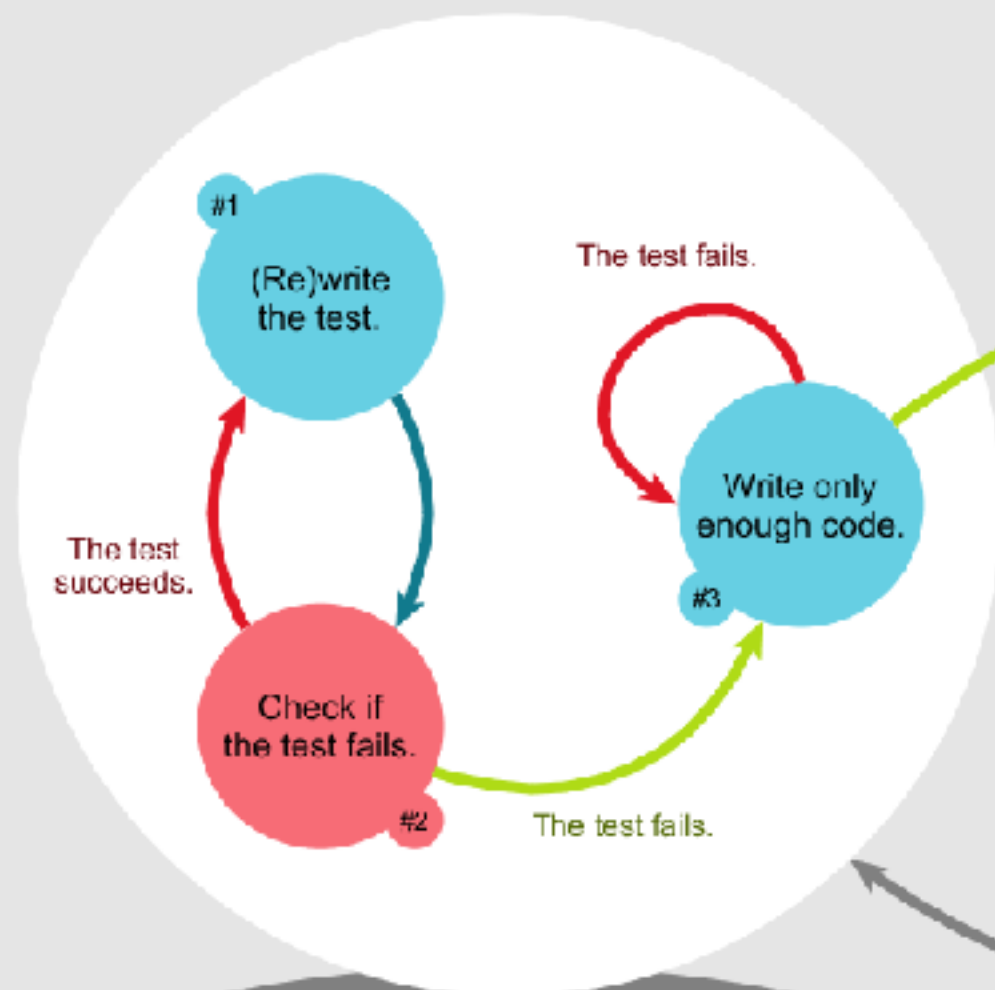
- Person & PersonValidator
- Validate Name
- Validate Username



# TDD

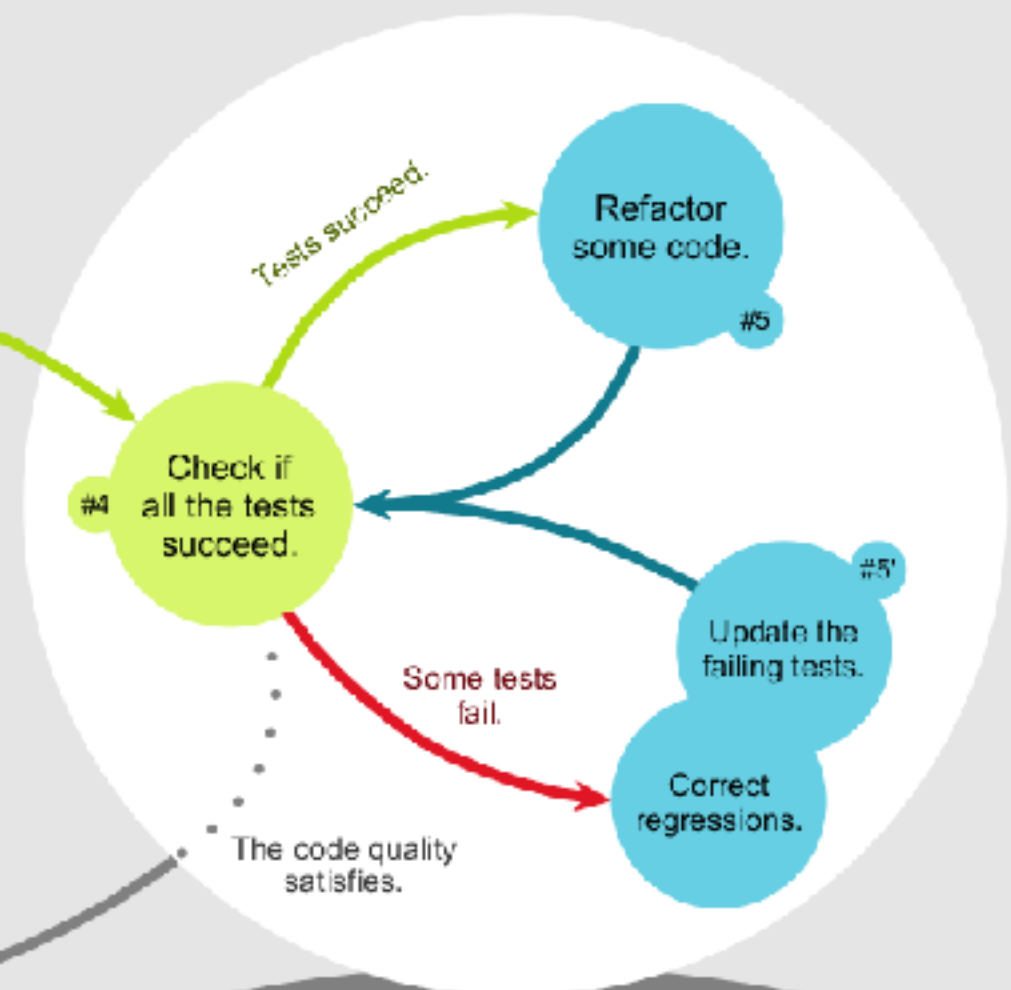
- specify the requirements (tests), then implement
- YAGNI
- when not to TDD?

## TEST-FIRST DEVELOPMENT



focus  
Completion of the contract  
as defined by the test

## REFACTORING



focus  
Alignment of the design  
with known needs

Iterate

# Exercise - e-mail

local-part@domain

## Local part:

- ASCII char set
- A-Z a-z 0-9
- !#\$%&'\*+,-./=?^\_`{|}~
- Dot (.), not first, not last, no two dots

## Domain:

- dot-separated labels
- A-Z a-z
- 63 char max.
- hyphen (not first or last char)

# Test doubles ...

- Nobody is an island !
- Dummy, Fake, Stub, Spies, Mocks
- Test interaction with other classes
- Test un-available dependencies
- example with Mockito

# Mockito

## mockito-core v2.15.0

```
import static org.mockito.Mockito.*;
```

```
MyClass mock = mock(MyClass.class)
```

```
when(mock.func()).thenReturn(value)
```

```
verify(mock).func()
```

# Coverage

- How much testing is “enough”?
- a measure of which parts of the SC are executed
- Function / Branch / Statement / Condition
- MC/DC
- EclEmma / JaCoCo

# Example

```
myFunction(int a, int b, int c, int d) {  
    if (a == 0) {  
        print("first condition")  
    }  
  
    if ( (b == 2 || c < 0) && d > 10) {  
        print("second condition")  
    }  
}
```

# **What you should know (after)**

- Why do we test?
- What levels of testing exist?
- What is a double/mock/stub/fake/spy/... ?
- Coverage (what is it and what does it say?)



# Exercises, Reading, etc.

! Use Testing in your project (from the start)

+ Learn JUnit + Mocking (now!)

+ What's a mock? What's a stub? What's the difference?

<https://martinfowler.com/articles/mocksArentStubs.html>

+ Coverage metrics & what they mean

~ How to Misuse Code Coverage - Brian Marick

<http://www.exampler.com/testing-com/writings/coverage.pdf>

~ TestCoverage

<https://martinfowler.com/bliki/TestCoverage.html>

/ Gerard Meszaros - XUnit Test Patterns