

Training program:

## **Test Driven Development (TDD) using Spock or JUnit 5**

Info:

<b>Name:</b>	<b>Test Driven Development (TDD) using Spock or JUnit 5</b>
<b>Code:</b>	<b>craft-test-TDD</b>
<b>Category:</b>	Automatic testing
<b>Target audience:</b>	developers
<b>Duration:</b>	3-4 days
<b>Format:</b>	40% lecture / 60% workshop

---

It's all about the content.

# Training program

## 1. Unit testing

1.1. FIRST

1.2. test structure (given-when-then/arrange-act-assert)

1.3. naming

1.4. testable code

1.5. patterns

1.5.1. Object Mother

1.5.2. Assert Object

1.6. anti-patterns

## 2. Spock Framework

2.1. Groovy in a nutshell

2.2. structure blocks/sections

2.3. parametrized tests

2.4. testing exceptions

2.5. conditional testing

2.6. comparison: JUnit/Mockito/AssertJ vs Spock

## 3. Test Driven Development

3.1. red-green-refactor

3.2. przejrzysta struktura testu

3.3. workflow

3.4. IDE (plugins, shortcuts)

## 4. Refaktoring

4.1. code transformations

4.2. IDE support

**5. Test doubles**

5.1. Mock

5.2. Stub

5.3. Fake

5.4. Dummy

5.5. Spy

**6. TDD and integration tests with Spring Framework**

6.1. Spock/JUnit support

6.2. config

6.3. injections

6.4. contexts

6.5. cache

**7. Test Driven Development - best practices****8. Testing - advanced topics**

8.1. asynch (Awaitility, Spock)

8.2. mutation testing (PIT)

8.3. feature toggle

8.4. performance

8.5. randomness handling

8.6. microbenchmarks

8.7. RxJava

8.8. testing Event-Driven arch

8.9. mikroservices

8.10. consumer-Driven-Contracts

