



E-Portfolio - Denny Flämig – 06.06.2018

# Agenda

- Why do we test units?
- Why do we need Mockito?
- What are mocks?
- How do we use mocks?
- Real examples

# What is a unit?

- Small part of the application
- Ideally only one elementary functionality
  - Does only one defined thing
  - Defined input has a defined output / outcome

# Why do we test units?

- Automated testing of functionality
  - Defined usecases
  - Reliable execution
- Fast testing of functionality
  - Run tests often during development to find problems early
- Isolated testing of functionality
  - Directly know what is broken if a test fails

# Problems

- Automated testing
  - Only possible if the given JUnit environment allows the execution of the tested methods
  - If methods use external dependencies the environment has to be provided and configured -> additional effort and error-prone
- Fast testing
  - External dependencies are often much slower than local in-memory operations
- Isolated testing of functionality
  - There is no isolation given if any external dependencies are used

# Solution

- Avoid to use the real environment
  - Use something under your own control to simulate the environment
- >Mock the environment with Mockito

# What are mocks?

- Mocks are instances of classes and interfaces without the real behaviour of the mocked class/interface
- Mocks can define behaviour for methods
- Mocks can verify calls for methods

## Reasons for mocking

### **Isolate the tested unit**

- Mock away unwanted behaviour
- Mock away expensive environment
- Take shortcuts
- Fake behaviour
- Avoid implementation of dummy classes



# What can Mockito do for us?

- Create mocks of classes and interfaces
  - `Mockito.mock(Class...)`
- Create partial mocks of object instances (they are mocks and have the same abilities but call real methods)
  - `Mockito.spy(Object...)`
- Create behaviour on mocks
  - `Mockito.when(mock.method()).thenReturn(value)`
  - `Mockito.doReturn(value).when(mock).method()`
- Verify interactions on mocks
  - `Mockito.verify(mock).method(expectedValue)`

# General rules

- Do not mock types you don't own
- Don't mock value objects
- Don't mock everything

Live example

- <https://github.com/dennyfl/MockitoPresentation>