



TDD – TEST DRIVEN DEVELOPMENT

E-PORTFOLIO

BY

KAI SCHWARK

AGENDA

Why testing?

How do we test?

Now the other way round

Practice-Task

Conclusion

WHY TESTING?

Changes

- Be confident
- Be safe

Proof of functionality

- For you
- For your client

Less maintenance-time

- Less new bugs
- Less time debugging

HOW DO WE TEST?



Formulate a
wague idea of
functionality



Write code



Test by hand



Correct found
errors



Maybe write a
test

HOW DO WE TEST? - UNITTESTS

- Biggest cut of all the tests
- Precise
- Short
- Only one usecase of a functionality of a function/class
- No all-knowing miracle tests!

HOW DO WE TEST? – ONLY UNITTESTS?

Unit test vs. Integration test



HOW DO WE TEST? - INTEGRATIONTESTS

- Testing that seperately developed modules work together properly
- Test that a system of multiple modules work as expected

HOW DO WE TEST? – USEFUL CONVENTIONS (FOR ME)



One test-class contains test for only one class that needs to be tested



The test-class should have the same name as the class to be tested with the Addition "Test" at the end (ex. MyClass -> MyClassTest)



Self-documenting names for tests.
name of the tested function + testcase + expected outcome.
(ex. "setSpeed_speedCanNotBeNegative_speedShouldBeUnchanged()").

HOW DO WE TEST? - REMINDER



Write Code



Fix Errors



Write automated
tests (eventually)

NOW THE OTHER WAY ROUND - TDD



Think what the Code
should do



Write detailed tests
for every use-case



Write Code against
the tests

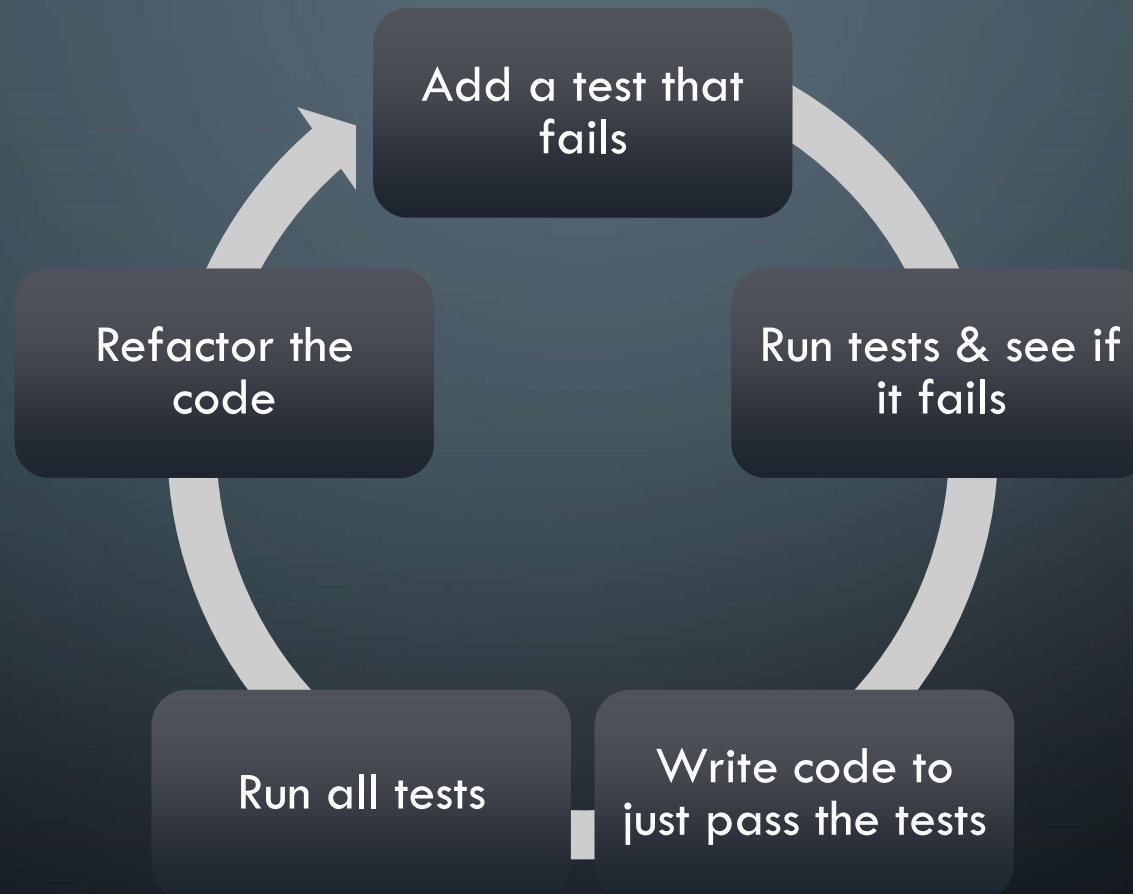
TDD - BENEFITS

- Thinking before coding -> Improved productivity
- Shorter implementation
- Less unneeded code and redundancy
- Easier debugging
- 100% test-coverage
- Modularized, flexible & extensible code
- Easier documentation though self documenting tests

TDD – LIMITATIONS

- Functional testing (ex. GUI design)
- False sense of security (see unit- vs integration-test GIF)
- Badly written tests cause maintenance overhead & blind spots

PRACTICE-TASK



CONCLUSION

We've seen the benefits and limitations of TDD and now have an idea on how to start adopting and using the TDD approach at least partially in our projects to further increase the quality of our code.



THANK YOU
FOR YOUR
TIME!



SOURCES

- https://cdn-images-1.medium.com/max/1600/1*4-T6VVnULaszi9ydHQ7Sfw.gif
- https://www.youtube.com/watch?v=QCif_r8eK4
- <https://brainhub.eu/blog/test-driven-development-tdd/> (13.05.2019)
- <https://www.it-agile.de/wissen/agiles-engineering/testgetriebene-entwicklung-tdd/> (10.05.2019)
- <https://martinfowler.com/bliki/IntegrationTest.html> (14.05.2019)