Assignment 1

Course: IN3240/4240 Software Testing

Delivery format: Deliver in a zip-file with sensible filenames respectively to the exercises. *

Delivery Info: Required to work in groups of 2-4

Assignment deadline: **Monday 18.03.2018,** kl. **23:59**

General infomation:

This assignment is divided into four (4) parts.

The first part is related to the lecture about Test automation given by **Itera**. Here you will get familiar with the test automation tool Selenium ChromeDriver and encode your own tests. It is recommended to do **the weekly tasks** first.

The second part is about getting familiar with the online static analysis tool **SonarCloud** – used for detecting bugs, vulnerabilities and other issues in code.

The third part is about unit-testing where you shall use **Test Driven Developement** in **Cyber-Dojo** in order to solve some programming exercises.

The fourth and last part is about static testing techniques, i.e. Reviews.

Otherwise it is important to **read the assignment carefully and perform the tasks as described**.

*If you are able to form the group with all members on Devilry's group function - deliver the assignment as usual. If you are unable to form the group with all the group members (may be due to reasons of having members on different versions of the course) - let only one group member deliver the assignment, but note each members' name, username, and course version on the front page of the deliverable.

Test Automation
Test automation is about encoding your tests in order to achieve more frequent deliveries and faster testing. Manual testing require physical time and effort to ensure that the software performs as it should. Overall this method of testing can be time-consuming and repetitive. Therefore using test automation tools has been proven to be more effective, less time-consuming and sometimes a more cost effective compared to the manual approach.
Delivery: The test scripts and the test reports generated in HTML.
Go to Part 1

Static Analysis
Static analysis allows us to analyze code or requirements without actually executing, or running the code. The main advantage of this is that it allows us to evaluate the SUT (System Under Test) and detect flaws, vulnerabilities and other weaknesses in the software, that cannot be seen thorough dynamic testing. Static analyzers are also vital in detecting defects early in the software development process.
Delivery: Answers to the questions
Go to Part 2

Unit testing and Test Driven Development
Unit testing is very important to ensure the quality of the code and makes the regression testing practical possible. In Test Driven Development TDD each test is written before the related code and the testing thereby completely integrated in the development process.
Delivery: The session IDs
Go to Part 3

Static testing - Reviews
While static testing techniques will not solve all the problems, they are enormously effective. They can improve both quality and productivity by impressive factors. Among the questions that arise are: How can we evaluate or analyze a requirements document, a design document, a test plan, or a user manual? One powerful technique that can be used is **reviews**. In principle all software work products can be tested using review techniques. The goal is to help the author and to improve the quality of the document
Delivery: An essay
Go to Part 4