Rafael Silva – rafael.silva@nackademin.se
Date:2020-10-18
Deadline: 2020-11-01
Version: 1.0

Testautomatisering – Assignment 01

This assignment *is mandatory* and is divided, so you will work on it over two weeks. You will work *individually* on this assignment.!

- Week 43:
  - You will design a set of 5-10 test cases and document them. The system under test is the Vue application (hotel application) that is used in the demo videos. Please, use the example test case in the end of this document for inspiration on how to write your own.
  - You will use cypress.io to implement the 5-10 test cases that were designed during week 43. In here you will not put much effort on clean code and such; your main goal is to be able to use cypress.io and implement your test suite.
- Week 44:
  - You will refine your cypress project from week 43, clean the code and organize it, according to Page-objects. Further you will configure your project and so be able to generate test reports out of test runs.

Rules for submission:

For this specific assignment, you will create a project with cypress, locally on your computer, and a remote repository on GitHub. When you create the new project, use the naming format *automation-namesurname* (e.g. automation-rafaelsilva) as the name of your project.

Host your project at GitHub as a public repository and submit the link to your GitHub repository at the student portal. The deadline for this assignment is Sunday the 1st November at 23.59.

Here are some things to consider:

- You will not be assessed by your skills with Git/Github, so keep it simple.
- Your project will to be ready to be run, i.e. when I correct your assignment, I do not need to make any small fix of any kind to be able to run it.
- Read carefully the instructions below and if you are uncertain of any requirement, contact me.

Part I (Requires the introductory lectures/demos about cypress and the basics in test design)

Design 5 to 10 test cases and document them. Use the example in the end of this document as an inspiration on how to document your test cases. If you prefer, use any other template of your choice. Further, implement your test suite using cypress.io (do not put much effort on cleaning the code or the structure of your project!).

Part II (Requires the lectures/demos about Page-objects and the lectures/demos about Test reports)

Your task now is to refactor your code by applying the Page-objects design pattern and, thus, make your code more structured and organized. Further, your last task in this assignment is that you configure your test project so that you can get test reports out of test runs.

When you are finished with Part I and Part II you will submit the link to your remote repository at the student portal. Please, double-check that your solution follows the instructions under *Rules for submission* in this document.

Example of a test case:

Test case: Perform login.

Designed by: Rafael Silva

Pre-conditions:

- The login credentials are stored in the system.
- The system is up available at the address: localhost:3000

Post conditions:

- None

Dependencies:

- None

Test steps:

Step	Test step	Test data	Expected result	Actual result	Pass/Fail	Notes
1	Open the browser and start the system	localhost:3000	The application is started.			
2	Assert that the page is correct		The user is in the right page			
3	Inform the username	tester01				
4	Inform the password	GteteqbQQgSr88SwNExUQ v2ydb7xuf8c				
5	Click the login button		The user gains access to the system.			
6	Assert that the user is logged in		The user is in the right page.			
7	Click the button ‘logout’.		The user exits the system is forwarded to the login page.			
8	Assert that the page is correct		The user is in the right page			