JAVA exercises testing

2020-10-07

SDA Software development academy
Author: Jolanta Griskeviciene
Background photo: Evening in Klaipėda, Lithuania, 2020

software development academy

Contents

Acknowledgements

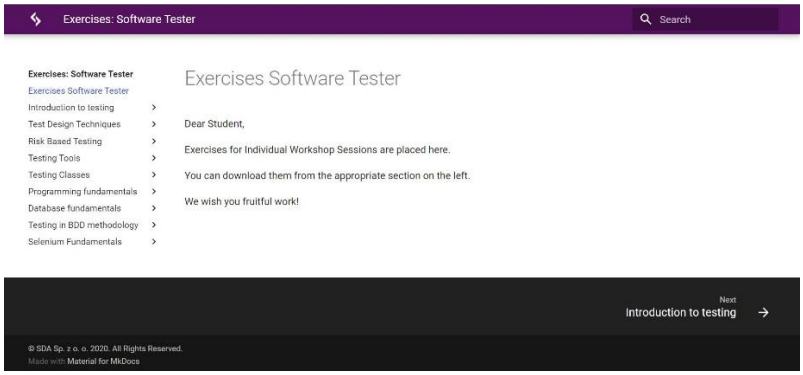
Our trainers were exceptional: friendly, positive, open, sharing knowledge with passion. I want to acknowledge our trainers:

- Sandra Budrevičienė
- Vitalij Kačanovskij
- Andrius Kaunas
- Karolis Kesminas
-

Document purpose

SDA academy students use a dedicated webpage for self-training Exercises: Software Tester (Figure 1). The purpose of this document is to describe performed testing on one of the most extensive parts of this webpage: Programming fundamentals. This part of exercises requires from users JAVA skills. JAVA IDE tool 'IntelliJ IDEA Community v.2020.1.3' was used to run JAVA scripts. TestRail was used as a tool for creation of Testing Project with Testcases, Test runs, Test Report. The bug reports were reported on GitHub public repository.

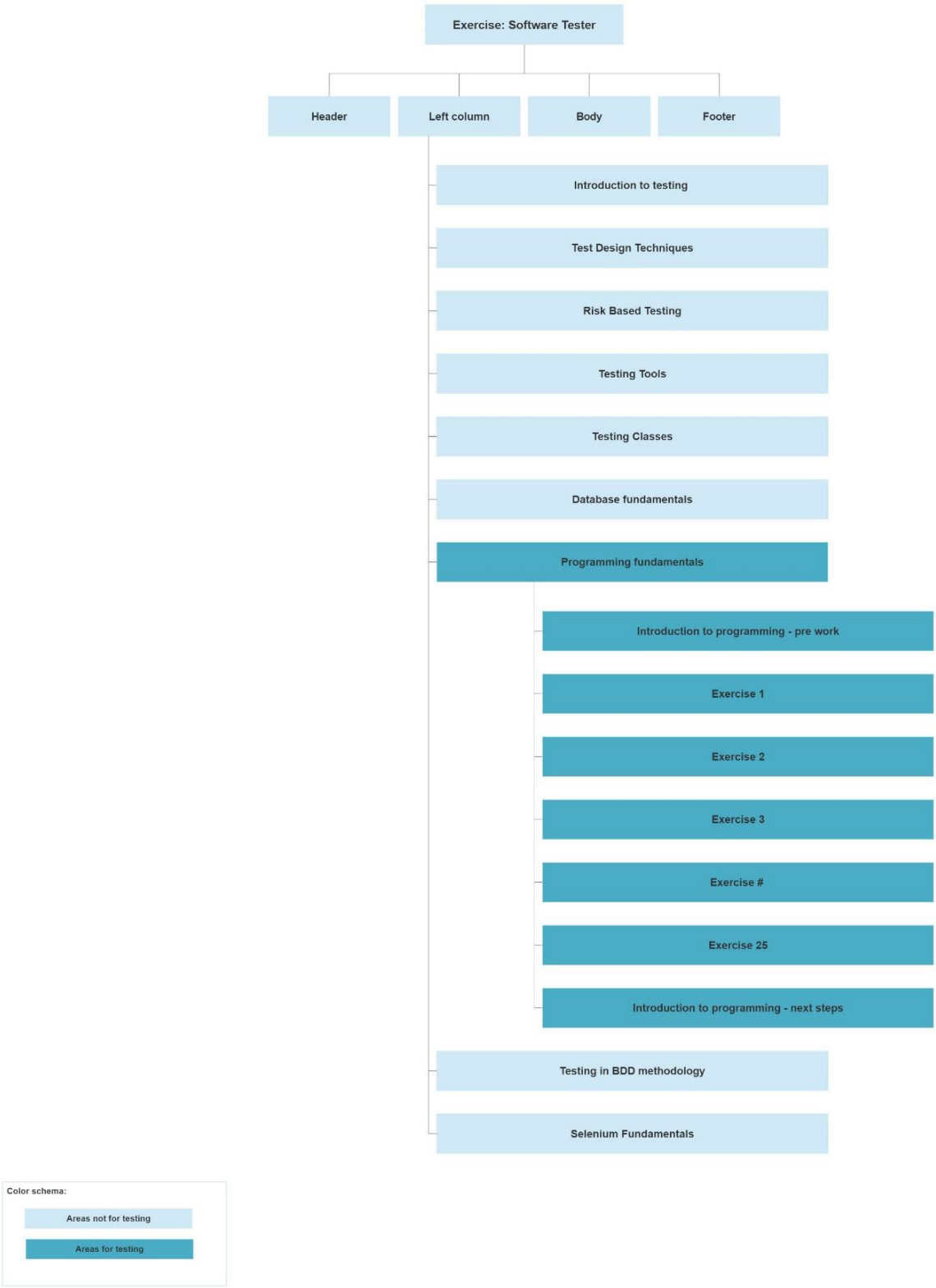
Figure 1. SDA webpage for testing.



Testing section

Webpage tree with areas for testing and not for testing are shown in Figure 2.

Figure 2. Webpage hierarchical tree and areas for testing.



Testing checklist

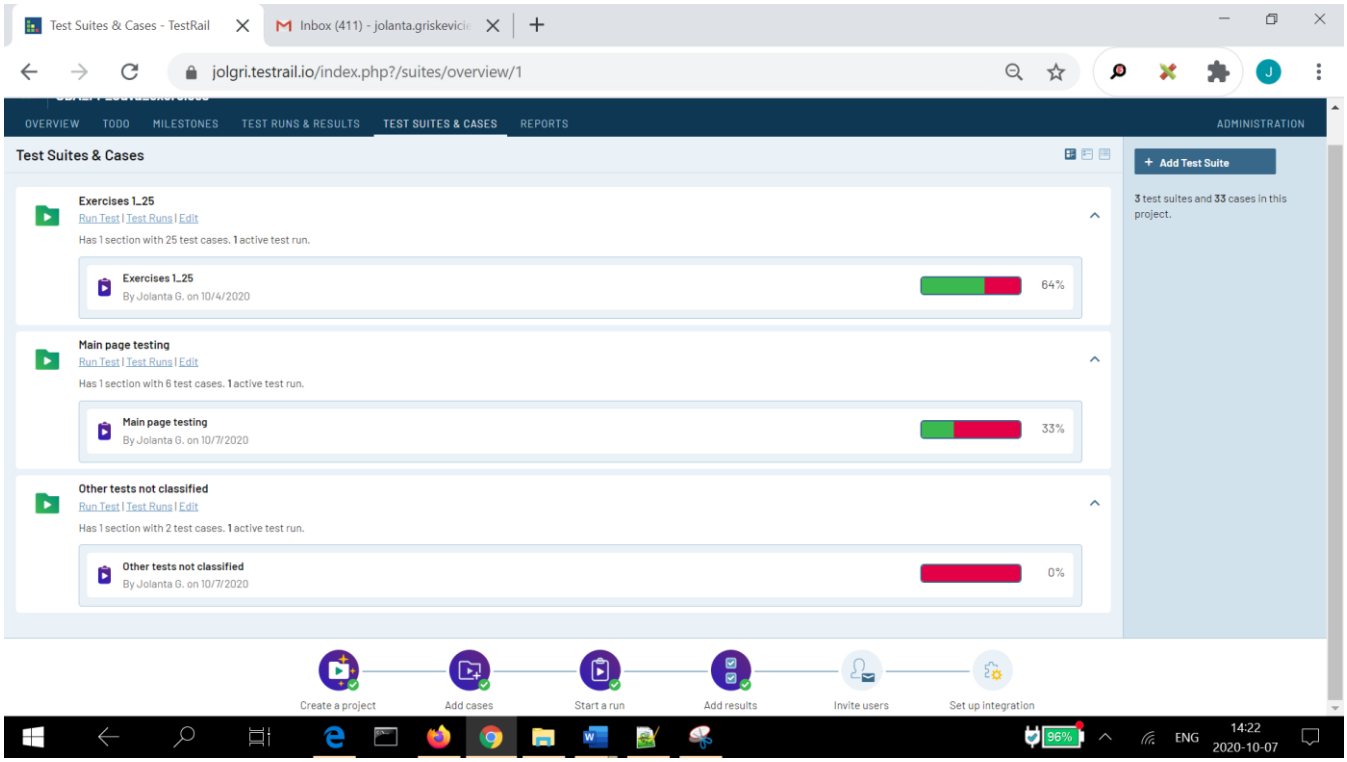
The detailed testing checklist is provided as an Annex 1. SDA_FP_2010-10-07 Checklist.xlsx. In total 51 testing points were planned.

Figure 3 The header of testing checklist (first 10 rows)

Id	Testing point	Pass/Fail/Not tested	Bug ID	Comments	URL
	Google Chrome				
1	Subpage Exercise1	Pass			
2	Subpage Exercise2	Pass			
3	Subpage Exercise3	Fail	Bug SDA_FP_T3_D1	GitHub	https://github.com/jolgri/SDA_FP_Bugs/issues
4	Subpage Exercise4	Pass			
5	Subpage Exercise5	Fail	Bug SDA_FP_T5_D2	GitHub	https://github.com/jolgri/SDA_FP_Bugs/issues
6	Subpage Exercise6	Pass			
7	Subpage Exercise7	Pass			
8	Subpage Exercise8	Fail	Bug SDA_FP_T8_D3	GitHub	https://github.com/jolgri/SDA_FP_Bugs/issues
9	Subpage Exercise9	Fail	Bug SDA_FP_T9_D4	GitHub	https://github.com/jolgri/SDA_FP_Bugs/issues
10	Subpage Exercise10	Pass			

Before test creation, I have decided to use TestRail for the management of the testing. I have created new project SDA_FP_Java_Exercises and 3 Test suits with 33 test cases:

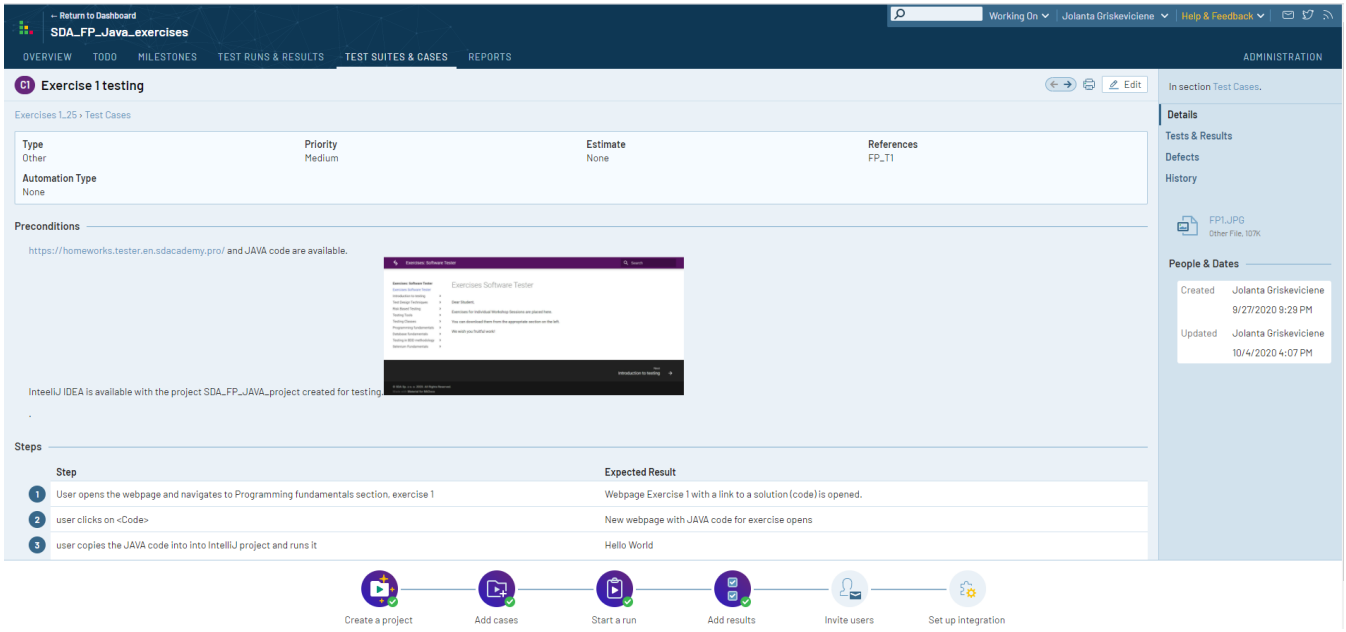
Figure 4 Project management in TestRail (testing version) with three Test suits



Detailed test cases

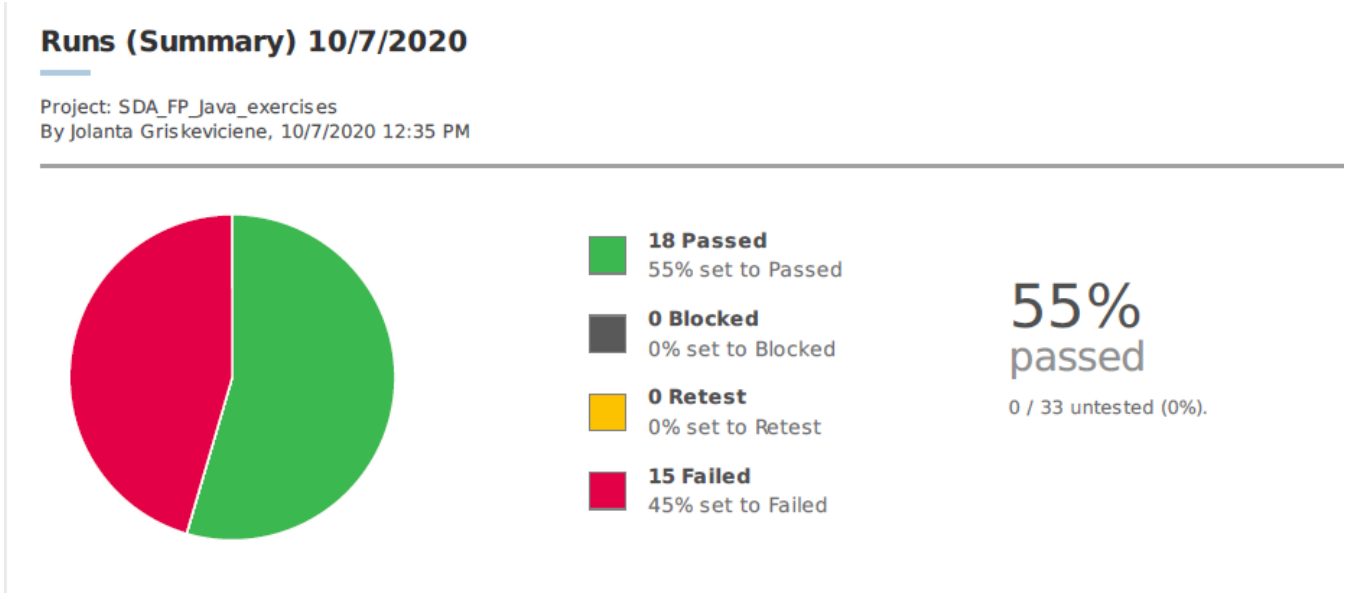
There are 33 test cases created oin TestRail environment All test cases were created and entered into TestRail test version as in Figure 5.

Figure 5. Test case example in TestRail



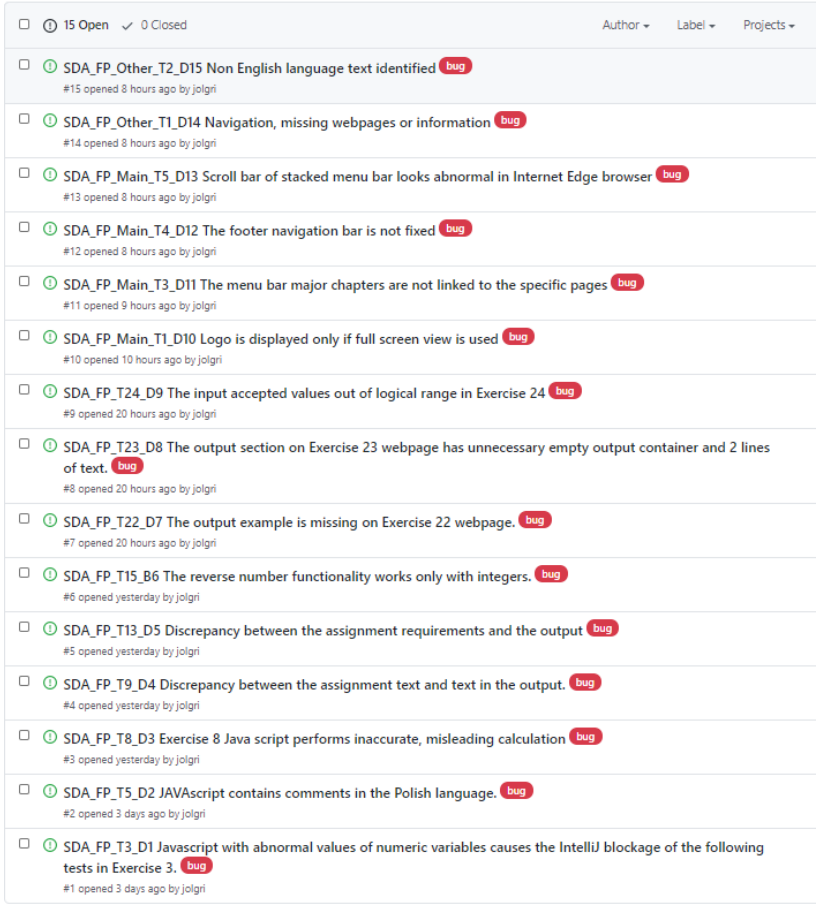
Overall 33 Test cases were run. The percentage of tests passed was 55% as shown in detailed TestRail report. Please find more details in the Annex 2 (SDA_FP_2010 10 07 TestRail report 9.pdf)

Figure 6 Test Rail Summary report figure shows 18 passed and 15 failed Test cases (55%, n=33)



Issues/bugs identified

There are 15 bugs identified and 15 detailed bug reports created in the GitHub platform: www.github.com/jolgri/SDA_FP_Bugs/issues :



Each detailed bug report contains mandatory information (bug ID, Title, Description, Steps, Expected result, Test result, Date Raised, Detected By, Status, Severity), and screenshots. Every bug report id is a kind of abbreviation of the linked information in the specific test case in TestRail environment. (Figure 8)

The bug identifier contains this information, e.g. SDA_FP_Main_T1_D14:

SDA = Software Development Academy;
FP= Final Project;
T1= Test case one in Exercises 1_25 tests suite,
Main= in Main page tests suite,
Other= in Other not classified tests suite;
D14=Defect identifier.

The detailed summary report in TestRail about performed test runs provides an overview of all 15 bugs identified and reported (Annex 3. SDA_FP_2010 10 07 TestRail-report-10.pdf).

Figure 7 TestRail Summary report. Number of defects by testing suite(n= 15)

Summary for References (Defects) 10/7/2020

Project: SDA_FP_Java_exercises
By Jolanta Griskeviciene, 10/7/2020 6:42 PM

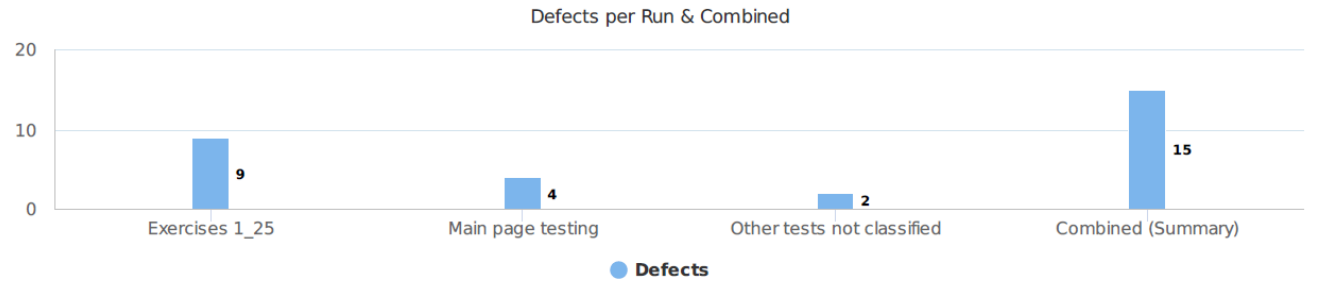


Figure 8 TestRail report. References and Defects.

References & Defects

References	ID	Title	Exercises 1_25	Main page testing	Other tests not cla...	Latest & Combined
FP_Main_1	C26	Logo hover and link to the home page		FP_Main_T1_D10		FP_Main_T1_D10
FP_Main_2	C27	Search function				
FP_Main_3	C28	Menu bar		FP_Main_T3_D11		FP_Main_T3_D11
FP_Main_4	C29	Home page Body part				
FP_Main_5	C30	Footer		FP_Main_T4_D12		FP_Main_T4_D12
FP_Main_6	C31	Scroll bars		FP_Main_T5_D13		FP_Main_T5_D13
FP_Other_T1	C32	Navigation testing			SDA_FP_Other...	SDA_FP_Other...
FP_Other_T2	C33	Language			SDA_FP_Other...	SDA_FP_Other...
FP_T1	C1	Exercise 1 testing				
FP_T2	C2	Exercise 2 testing				
FP_T3	C3	Exercise 3 testing	FP_T3_D1			FP_T3_D1
FP_T4	C4	Exercise 4 testing				
FP_T5	C5	Exercise 5 testing	FP_T5_D2			FP_T5_D2
FP_T6	C6	Exercise 6 testing				
FP_T7	C7	Exercise 7 testing				
FP_T8	C8	Exercise 8 testing	FP_T8_D3			FP_T8_D3
FP_T9	C9	Exercise 9 testing	FP_T9_D4			FP_T9_D4
FP_T10	C10	Exercise 10 testing				
FP_T11	C11	Exercise 11 testing				
FP_T12	C12	Exercise 12 testing				
FP_T13	C13	Exercise 13 testing	FP_T13_D5			FP_T13_D5
FP_T14	C14	Exercise 14 testing				
FP_T15	C15	Exercise 15 testing	FP_T15_D6			FP_T15_D6

Improvements and reflections

Self-learning is a crucial part of education. I liked the SDA Homeworks exercises very much, and due to this reason, I have decided to perform this webpage testing. The homework platform contains beneficial information and can be used by students with various level of knowledge. The trainers should provide access to the training material as early as possible at the beginning of the course.

The chapter programming fundamentals is the chapter which might be used not only by testers but by JAVA course students, too. If I could suggest, it would be a benefit for trainers and students to have one webpage SDA Students page with Training material and additional links to :
-Slides,
-Homeworks,
-ISTQB resources: materials and test examples,
-Recommended e-books,
-Software installations guides,
-SDA optional Video courses (e.g.GIT, HPPT, Agile etc.),
-SDA optional quizzes (e.g. GIT, HTTP),
-Links to various tools,
-Links to video classes for remote students,

For testing purpose, I have created the first Selenium WebDriver automated test (Annex 4. SDA_FP_2020 10 07 SeleniumWebDriver.py), which checks if we can access the specific JAVA script in Programming Fundamentals. However, this kind of automatisation is possible only with chapters, which have the same three main parts: Assignment, Output example, and solution (code). However, there is only Selenium chapter organised similarly. All training topics are structured differently. The architecture of the training material library is one of the points for discussion.

Tools used

Table 1. List of tools

ID	Name of the tool	Webpage	For what purpose was used	Comments
1	Smartdraw	https://www.smartdraw.com/	For schemas drawing	
2	IntelliJ IDEA Community v.2020.1.3	https://www.jetbrains.com/idea/	For JAVA programming	
3	TestRail	https://jolgri.testrail.io/index.php?/projects/overview/1	For Testing Project creation with Test Suites, Test cases, Test runs and Test reports	
4	GitHub	https://github.com/jolgri/SDA_FP_Bugs/issues	For Detailed Bug reports	
5	Selenium IDE	https://chrome.google.com/webstore/detail/selenium-ide/mooikfkahbdckldjjndioackbalphokd	For recording steps	
6	Visual Studio Code 2019	https://code.visualstudio.com/	To prepare and run Selenium IDE exported python file & Selenium WebDriver	
7	Google Chrome		For Browsers and responsiveness testing	
8	Internet Edge		For Browsers and responsiveness testing	
9	Firefox		For Browsers and responsiveness testing	
10	SAMSUNG Galaxy S%		For Browsers and responsiveness testing	
11	Grammarly		For Spelling check	

Annexes

- Annex 1. SDA_FP_2010-10-07 Checklist.xlsx
- Annex 2. SDA_FP_2010 10 07 TestRail report 9.pdf
- Annex 3. SDA_FP_2010 10 07 TestRail report 5.pdf
- Annex 4. SDA_FP_2010 10 07 Selenium WebDriver.py