# TEST PLAN Atlanthbh QA Engineer Test

E-commerce test

|  |  |  |
| --- | --- | --- |
|  | Name | Date |
| Created by: | Emsad Ahmetašević | 13.02.21 |
| Reviewed by: |  |  |

Target: <http://automationpractice.com/index.php>

Assigned: 12.02.21

Deadline: 19.02.21 10:00

Task:

1. Write test cases for this application (concentrate on real user flows)
2. From all the test cases, identify what you think represents a Smoke Test
3. Identify positive and negative test cases
4. With a programming/scripting language and testing framework of your choice, automate the smoke test from #2
5. Create Github repository and push your assignment solution with the documentation on how to run the automated tests there (focus on writing comprehensive Readme file)
6. Report bugs if you find any, with detailed steps to reproduce

​

Task execution:

1. Design and create test cases.
2. Create test scripts in Testcafe
3. Run test scripts.
4. Identify bugs and create a description of them.
5. Create a report of execution.
6. Create github repository
7. Create a detailed readme file with a description of how to run scripts
8. Upload all files to the repository
9. Send finished test task

Tools:

Testing framework: [TestCafe](https://devexpress.github.io/testcafe/)

Reporter plugin: [testcafe-reporter-html](https://www.npmjs.com/package/testcafe-reporter-html)

Code editor: [vscode](https://code.visualstudio.com/)

Repository: [github](https://github.com/)

How to install and run:

1. Install testcafe (A node.js tool to automate end-to-end web testing)

|  |
| --- |
| npm install testcafe |

2. Install testcafe html reporter (A testcafe plugin for creating html report files)

|  |
| --- |
| npm install testcafe-reporter-html |

3. Main run

|  |
| --- |
| testcafe chrome:headless \*\*/smokeTC.test.js -r html:reports/smokeTC-report.html |

|  |
| --- |
| testcafe chrome:headless \*\*/positiveTC.test.js -r html:reports/positiveTC-report.html |

|  |
| --- |
| testcafe chrome:headless \*\*/negativeTC.test.js -r html:reports/negativeTC-report.html |

Documentation:

Repository: [Atlantbh - QA Engineer - Test](https://github.com/emsad87/atlantbh-qaengineer-test)

Files:

Folder ["test-cases"](https://github.com/emsad87/atlantbh-qaengineer-test/blob/main/test-cases) contains .xlsx file with test case list and bug report list

Folder ["test-scripts"](https://github.com/emsad87/atlantbh-qaengineer-test/blob/main/test-scripts) contains all automated test case files

* [smokeTC.test.js](https://github.com/emsad87/atlantbh-qaengineer-test/blob/main/test-scripts/smokeTC.test.js) (Executes smoke test cases)
* [positiveTC.test.js](https://github.com/emsad87/atlantbh-qaengineer-test/blob/main/test-scripts/positiveTC.test.js) (Executes positive test cases)
* [negativeTC.test.js](https://github.com/emsad87/atlantbh-qaengineer-test/blob/main/test-scripts/negativeTC.test.js) (Executes negative test cases)
* [data.js](https://github.com/emsad87/atlantbh-qaengineer-test/blob/main/test-scripts/data.js) (contains all data needed to execute test cases)
* [pageElements.js](https://github.com/emsad87/atlantbh-qaengineer-test/blob/main/test-scripts/pageElements.js) (contains all page elements selectors needed to execute test cases)
* [cat.jpg](https://github.com/emsad87/atlantbh-qaengineer-test/blob/main/test-scripts/cat.jpg) (Image file needed in a test case) I like cats xD

Folder ["reports"](https://github.com/emsad87/atlantbh-qaengineer-test/blob/main/reports) contains html report files automaticaly created

Description:

These are the test task result files for Atlantbh - QA Engineer - Test

Created automated Smoke test cases, positive and negative test cases in separated files.

I did not automate all test cases, left some for manual testing

Also did not cover all possible test cases, probably left some of them, there are a lot of test case possibilities

Could also create some helper files to simplify code and avoid repeating - but there was not a lot of code repeating so no need for now and could also create a custom reporter plugin

Commented important part of code and also did console logs to show what the code is executing - to keep track of progress during code run in headless mode

If having trouble runing files see solution in FIX part of this reademe file

Created by:

[Emsad Ahmetašević](https://emsad87.github.io/)