Test Report

Project Name: **eMAG.ro**

Document Date: 13.11.2020

Table of Contents

Testing Scope 3

1.1. Test Environment – [GitHub](https://github.com/luciancalinescu95/Final-Project-eMAG.ro) 3

1.2. Test Specification 4

1.3. Test Cases 4

1.4. Automated Testing 5

1.5. API Testing 6

Test Results 7

2.0. Executed Test Cases - Functional 7

2.1. Executed Test Cases - Non-Functional 7-8

2.2. Automated Testing 8

2.3. API Testing 8

Bugs and Improvements 9

1.5. New defects 6

1.6. Validated bugs 7

1.7. Old bugs still open 7

Conclusions 8

# Testing Scope

The testing scope of this project was to test the main functionalities of eMAG.ro from a normal user and tester perspective and find any possible features which were not working correctly and possible improvements for this website.

**Why eMAG.ro?**

It is one the biggest e-commerce chain in Romania with a complex website that features many important functionalities for a shopping website to be tested.

**Testing approach?**

Without knowing or having access to the source code the best approach was black-box testing. We are aware of the main functionalities of the website and what are they supposed to do and we can build the test cases around that.

**What was actually tested?**

Key functionalities for a shopping website like login-logout, creating an account, add to bag, checkout, shopping cart operations, wishlisting products, filtering, search bar.

## Test Environment – [GitHub](https://github.com/luciancalinescu95/Final-Project-eMAG.ro)

Tests were run in the production because we don’t have access to the stage environment. In the github repository you can find the manual Test Cases Plan and Execution Report in TestLink, the UI Tests that were automated with Cypress using JavaScript, the Postman collection of tests for the API Tests and the performance scores of the website that resulted from running Lighthouse.

| Test Environment | Software Details | Comments |
| --- | --- | --- |
| Production | OS:Windows 10 Pro (10.0.19041)  Mobile: Android 9  Devices: Desktop PC, Mobile | Android:eMAG official app; Samsung Internet Browser |
| Browsers:Mozilla Firefox 82.0.3 (64-bit); Google Chrome 86.0.4240.198 (64 bits); Microsoft Edge 86.0.622.56 (64-bit) | DevTools app for Chrome |

1.2. Test Specification

Mind Maps for the website and it’s main functionalities that were tested

* [Website-mindmap](./Mind%20Maps/eMAG.ro%20-%20Site%20Mind%20Map.xmind)
* [Testing-mindmap](./Mind%20Maps/eMAG.ro%20-%20Testing%20Mind%20Map.xmind)

Test Cases Plan → [link](./Test%20Cases%20Plan/Test_Cases_plan_2020-11-10%20.doc)

Test Cases Execution Report → [link](./Test%20Cases%20Plan/Test%20Plan%20Execution%20Report%20-%20eMAG.ro%20-2020-11-13.pdf)

1.3 Test Cases:

|  |  |  |
| --- | --- | --- |
| **Test Case ID(TM-QA08-number)** | **Functionality** | **Comments** |
| 448,449,450,451,452,525 | Login |  |
| 453,454,517,518,519,523,524 | Register Account |  |
| 455,456,457,501 | Favourite Products |  |
| 459,460,461,474,475,520 | Add To Bag |  |
| 462,463,464,465,466,469,470,471,472,  473,541 | Shopping Cart |  |
| 504,503,502,467,468,538,539,542,543 | Account Page(Operations) |  |
| 483,484,485,486,487,488,489,490,491 | Search Bar |  |
| 494,495,496,497,498,499,500,521 | Filter Products |  |
| 533,534,535,536 | - | User Interface |
| 507,508,528,530,531,540 | - | End-to-end |
| 477,476,478,479,480,481,537,505,506,  492,522,526,527,532,529 | - | Non-Functional Tests |

1.4 Automated Testing

For simplifying test execution, increasing the speed of executing manual tests and saving time, the manual tests are automated using the Cypress environment with JavaScript. Note that not all tests can be automated at this point (eg. Login tests require captcha or products changing price or name constantly etc.) and are subject to change. At this point the following tests are automated:

|  |  |  |
| --- | --- | --- |
| **Test Case ID(TM-QA08-number)** | **Functionality** | **Comments** |
| 459,460,461,474,475,520 | Add To Bag |  |
| 462,463,464,469,470,472 | Shopping Cart |  |
| 483,485,486,487,488,489,490,491, | Search Bar |  |
| 494,495,496,497,498,499,521 | Filter Products |  |
| 455,456,457 | Favourite Products |  |

1.5 API Testing

To better understand and check the functionalities, testing techniques, and the execution of test cases, API Testing can be approached. API (Application Programming Interface) enables communications and data exchange between tow separate software systems. Instead of using user inputs and outputs, we can use software to send calls to API, get an output and check the system’s response. API Testing was done using the Postman tool to make requests and check the website’s response on various functionalities.

API test collection → [link](./API%20Test%20Collection/eMAG.ro.postman_collection.json)

API test run results → [link](./API%20Test%20Collection/eMAG.ro.postman_test_run.json)

# Test Results

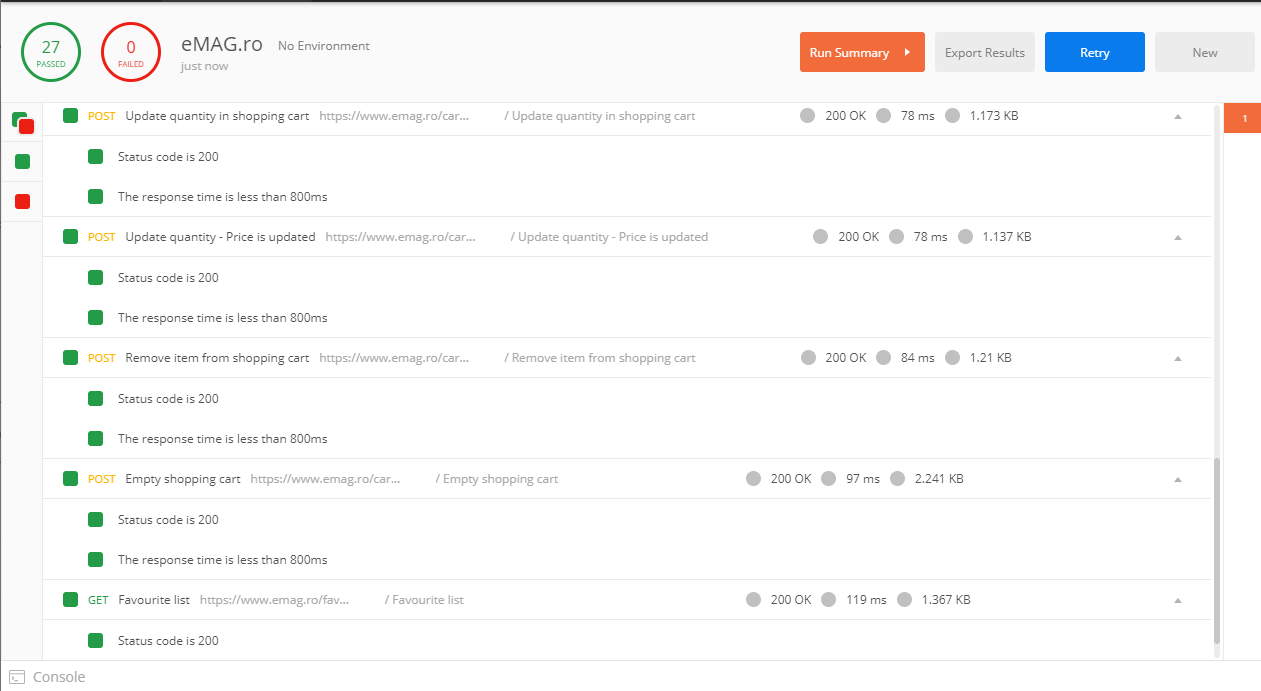
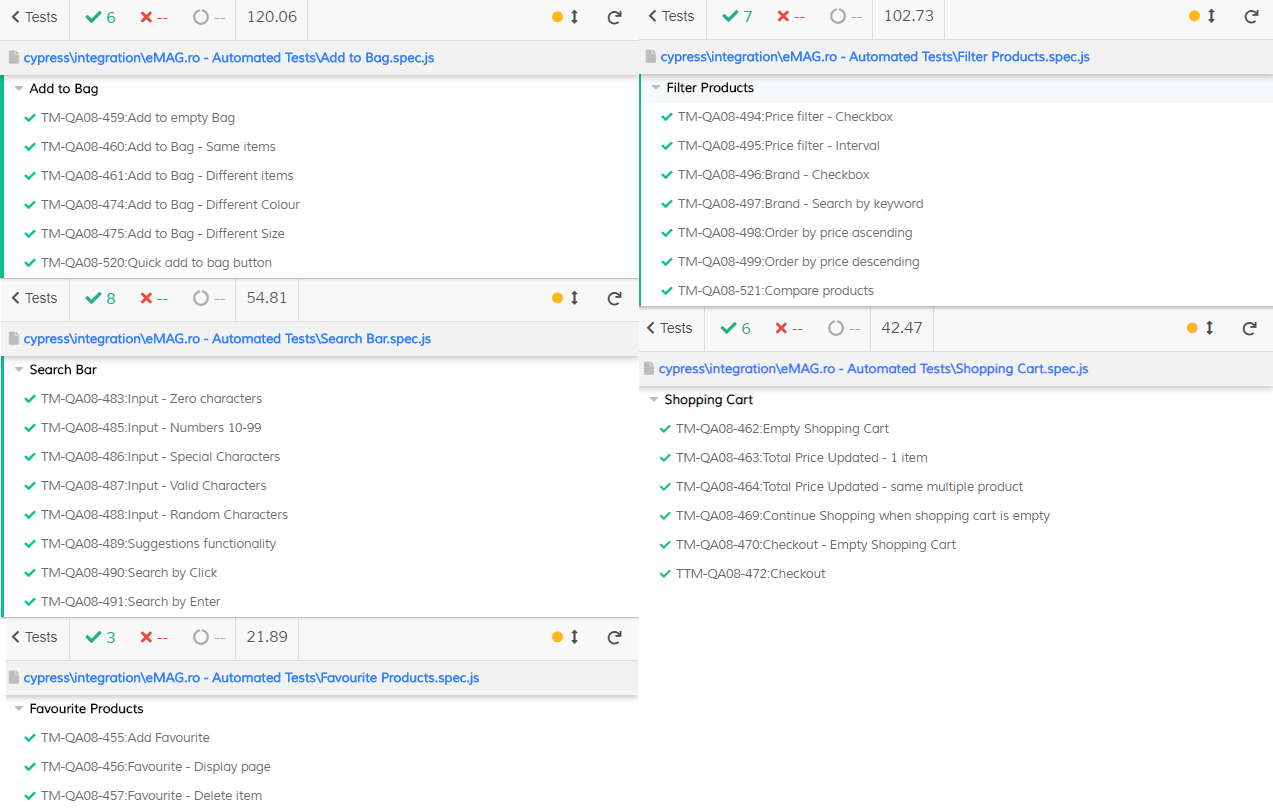
# 2.0. Executed Test Cases – Functional

| Functionality | Total | Pass | Failed | Blocked | Not Run | Defects |
| --- | --- | --- | --- | --- | --- | --- |
| Login | 6 | 6 | 0 | 0 | 0 |  |
| Register Account | 7 | 7 | 0 | 0 | 0 |  |
| Favourite Products | 4 | 4 | 0 | 0 | 0 |  |
| Add To Bag | 6 | 6 | 0 | 0 | 0 |  |
| Shopping Cart | 11 | 11 | 0 | 0 | 0 |  |
| Account Page Operations | 9 | 9 | 0 | 0 | 0 |  |
| Search Bar | 9 | 9 | 0 | 0 | 0 |  |
| Filter Products | 7 | 6 | 1 | 0 | 0 | 1 |
| User Interface | 4 | 4 | 0 | 0 | 0 |  |
| End-to-end | 6 | 4 | 2 | 0 | 0 | 2 |
| **Overall Results** | **70** | **67** | **3** | **0** | **0** | **3** |

2.1. Executed Test Cases – Non-Functional

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Description | Total | Pass | Failed | Blocked | Not Run | Defects |
| Performance | 2 | 1 | 1 | 0 | 0 | 1 |
| Compability | 7 | 7 | 0 | 0 | 0 |  |
| Security | 5 | 4 | 1 | 0 | 0 | 1 |
| API | 1 | 1 | 0 | 0 | 0 |  |
| **Overall Results** | **15** | **13** | **2** | **0** | **0** |  |

2.2 Automated Testing

2.3 API Testing

# Bugs and Improvements

3.0 Bugs and Defects

| Defect ID | Defect summary | Severity |
| --- | --- | --- |
| 101 | Colour filter not working correctly | moderate |
| 102 | Guests can’t checkout | moderate |
| 103 | Guests can’t create wishlist | moderate |
| 104 | Performance scores are not up to metrics | moderate |
| 105 | No email validation link sent | major |

## 

# Conclusions

Smoke Test Cases, for the main functionalities in a percent of 100% were executed, 90.5% passed and 9.5% failed. The failed test cases are related to the Cart functionality.

A total number of 20 new bugs were identified, half of them were identified as major severity, the other half as normal severity. No bugs were validated.

In addition to Smoke and Positive testing, testing types such as: Functional Testing, Exploratory, Compatibility, Negative, UI, Usability, were covered.

Exploratory testing was performed for important functionalities. A large number of bugs were identified during exploratory testing, 9 of them are major, most of them related to the Shopping Cart, but also for other functionalities.

Most of the bugs were identified by Negative Testing.

Compatibility testing was also performed, for different browsers and devices. The bugs reproduce on all browsers. In addition to the bugs identified on Windows OS, that also reproduce on Android, 1 new major bug related to the Shopping Cart was identified on the mobile device.

Considering the number of high severity bugs identified, on all important functionalities, the fixing of at least the major severity bugs is recommended, although the application is already in production.