**SOFTWARE VERIFICATION, VALIDATION AND TESTING**

TESTING DOCUMENTATION

PROJECT NAME

Booking.com

Prepared by:

**Ajla Šišić**

Proposed to:

**Samed Jukić, Assist. Prof. Dr.**

**Aldin Kovačević, Teaching Assistant**

Date of submission: 16.01.2022.

**Contents**

[1. Introduction 3](#_Toc93267935)

[1.1. About the Project 3](#_Toc93267936)

[1.2. Project Functionalities and Screenshots 3](#_Toc93267937)

[2. Test Plan 3](#_Toc93267938)

[2.1. Scope 3](#_Toc93267939)

[2.2. Testing Environment and Tools 3](#_Toc93267940)

[3. Test Execution 4](#_Toc93267941)

[3.1. Test Scenario Name 4](#_Toc93267942)

[9. Conclusion 12](#_Toc93267943)

[9.1. Testing Summary 12](#_Toc93267944)

[9.2. Final Thoughts 12](#_Toc93267945)

# 1. Introduction

## 1.1. About the Project

For my project, I have chosen to work on the most popular website for booking hotels, motels and apartments worldwide, booking.com, which can be found on the following link: <https://booking.com/>

On this website, users can find various features like booking a place to stay in their desired location and at their desired date. You can also search all available flights to your destination, as well as reserve transportation to your location. It is also handy for finding nearby attractions and organizing a private vehicle for the time of your stay.

It can also be helpful for finding guests for your hotel or apartment, since users have an option to place their property on the site and manage the availability and price there. Guests can communicate with the owners as well. All and all, a very useful site used globally and daily for traveling purposes.

## 1.2. Project Functionalities and Screenshots

The plan was to test most of the features and functionalities that the website provides, those being the following:

* **Booking your stay** – I entered a hotel based on my choice, date and number of guests, after which you should choose a room and enter the details for the reservation to be completed.
* **Finding attractions** – Firstly find attractions at a particular location and checked out the first one that the search filter offered us.

# 2. Test Plan

## 2.1. Scope

I decided to test all front-page (landing page) functionalities that the page generates, including sub-functionalities that you come to. These sub-functionalities included filtering searches, input of data, selecting different options and checking how they perform. I didn’t plan to fully complete a reservation since that would cause disturbance to the property owners. Also I didn’t want any interaction with them so I skipped the chat feature as well.

Generally, I focused on the most used function of the website.

## 2.2. Testing Environment and Tools

For this project, I used Eclipse IDE for Java developers in which I used Java as a programming language for creating test cases. Other than that, I used ChromeDriver which is an open-source tool for automated testing of webapps across many browsers. It provides capabilities for navigating to web pages, user input, JavaScript execution, and more. I have also used Selenium, a collection of language specific bindings to drive a browser used to create robust, browser-based regression automation suites and tests, scale and distribute scripts.

# 3. Test Execution

## 3.1. Test Scenario Name

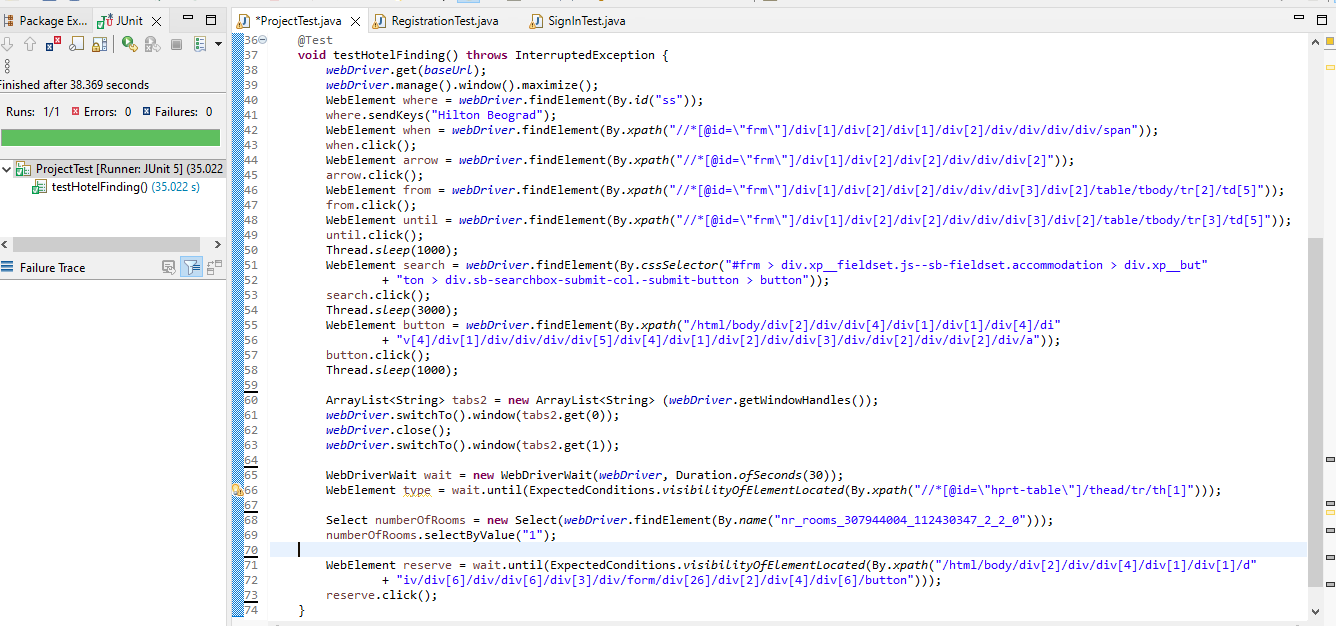
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Registration | | | | |
| **Description:**  Trying to create an account | | | | |
| **Pre-condition(s):** | | | | |
| **Test Steps:**  1. Go to the landing page  2. Click register  3. Enter email  4. Click button to continue  5. Wait until the page loads field for entering password  6. Enter password  7. Click create | **Test Data:**  - [ajla.sisic00@gmail.com](mailto:ajla.sisic00@gmail.com) as email and Testing123 as password; These fields should be changed in order for the test to work. | **Expected Result:**  To create a new user | **Actual Result:**  The user has been successfully created and available for usage. | **Status:**  PASS |
| **Notes:** If too many attempts are made, there will be a human verification popped up which can fail the test | | | | |



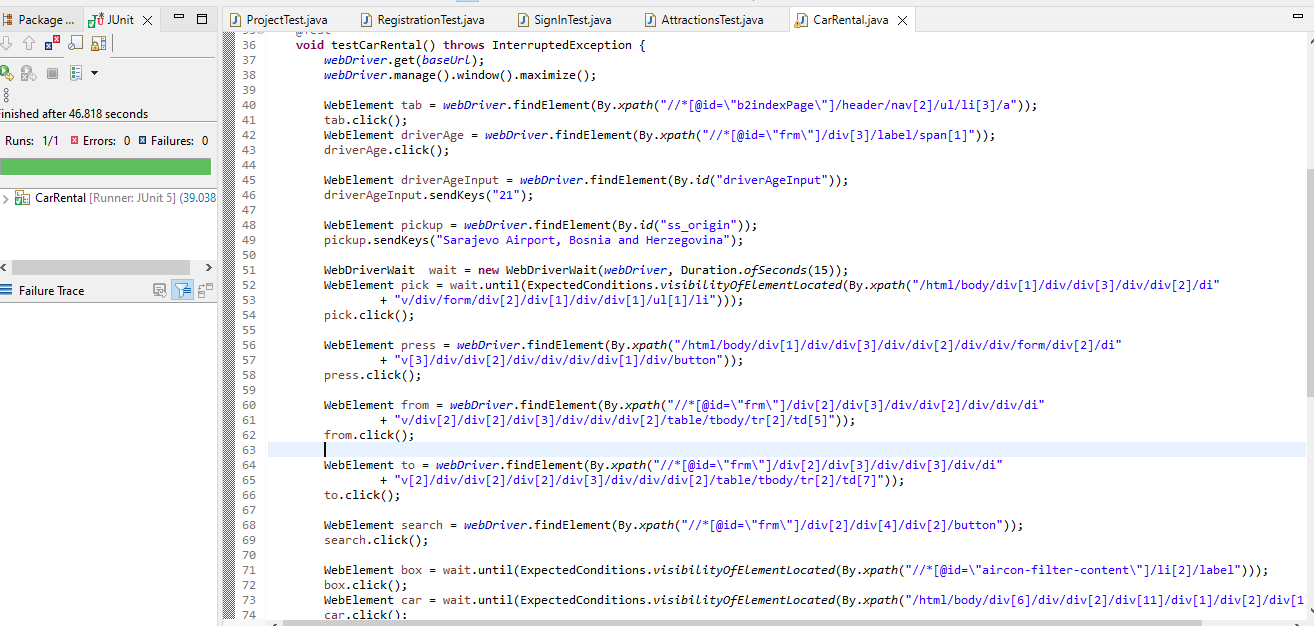
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Log in | | | | |
| **Description:**  Logging into the website with a registered account | | | | |
| **Pre-condition(s):** Have a registered account to use as test data | | | | |
| **Test Steps:**  1. Go to the landing page  2. Click sign in button  3. Enter email  4. Click continue button  5. Wait until password field is shown  6. Enter password  7. Click to log in | **Test Data:**  The data used as email and password are already registered users (previously registered) | **Expected Result:**  To successfully log in to the website | **Actual Result:**  Logged in. | **Status:**  PASS |
| **Notes:** | | | | |

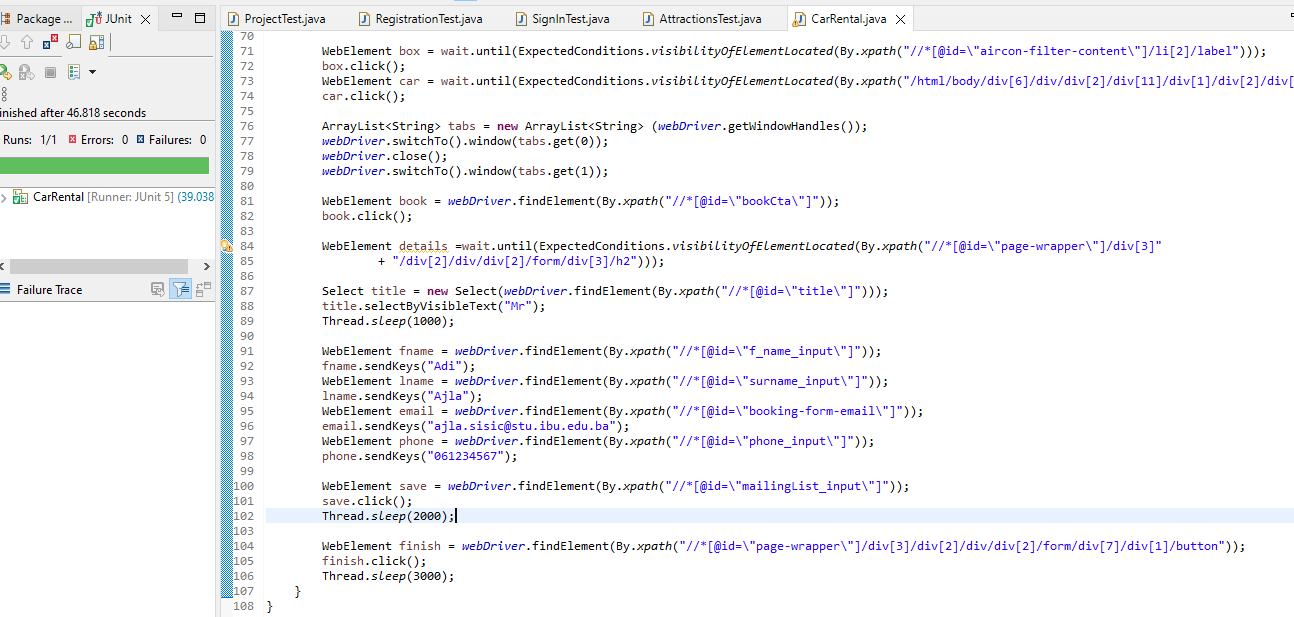


|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Find specific hotel | | | | |
| **Description:**  Trying to find Hilton Hotel in Belgrade | | | | |
| **Pre-condition(s):** | | | | |
| **Test Steps:**  1. Go to landing page  2. Enter hotel  3. Click on dates  4. Click on top-right arrow to scroll months  5. Click on two dates  6. Click search  7. Click first hotel  8. Handle two tabs  9. Land on specified hotel | **Test Data:** | **Expected Result:**  Click on the hotel specified in the search bar at the landing page | **Actual Result:**  As expected | **Status:**  PASS |
| **Notes:** | | | | |

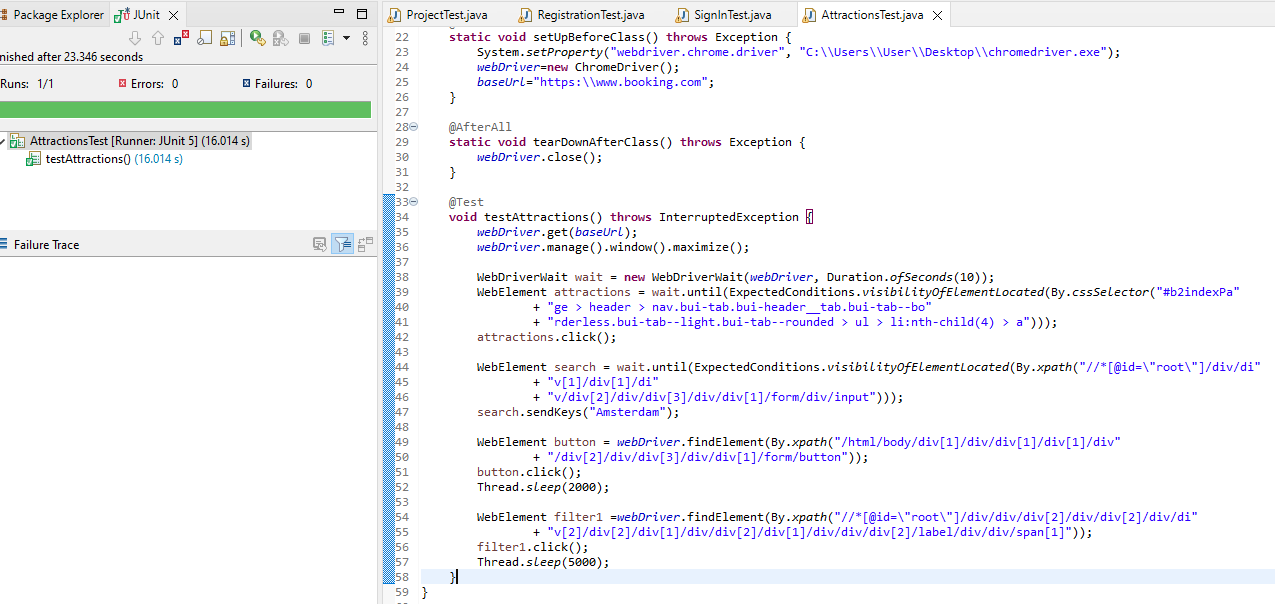


|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Car reservation | | | | |
| **Description:**  Reserving a car using the websites feature for searching available car rentals | | | | |
| **Pre-condition(s):** The page shouldn’t be translated because of the search parameter (test data) | | | | |
| **Test Steps:**  1. Go to landing page  2. Click car rentals tab  3. Unselect driver age  4. Enter 21 as age  5. Enter pickup location  6. Wait until a pick pops  7. Click first pick  8. Select date  9. Search  10. Wait until vehicles load  11. Click first vehicle  12. Input data  13. Click reserve | **Test Data:**  - Sarajevo International airport | **Expected Result:**  To come to the final page before reserving a vehicle | **Actual Result:**  Success | **Status:**  PASS |
| **Notes:** | | | | |

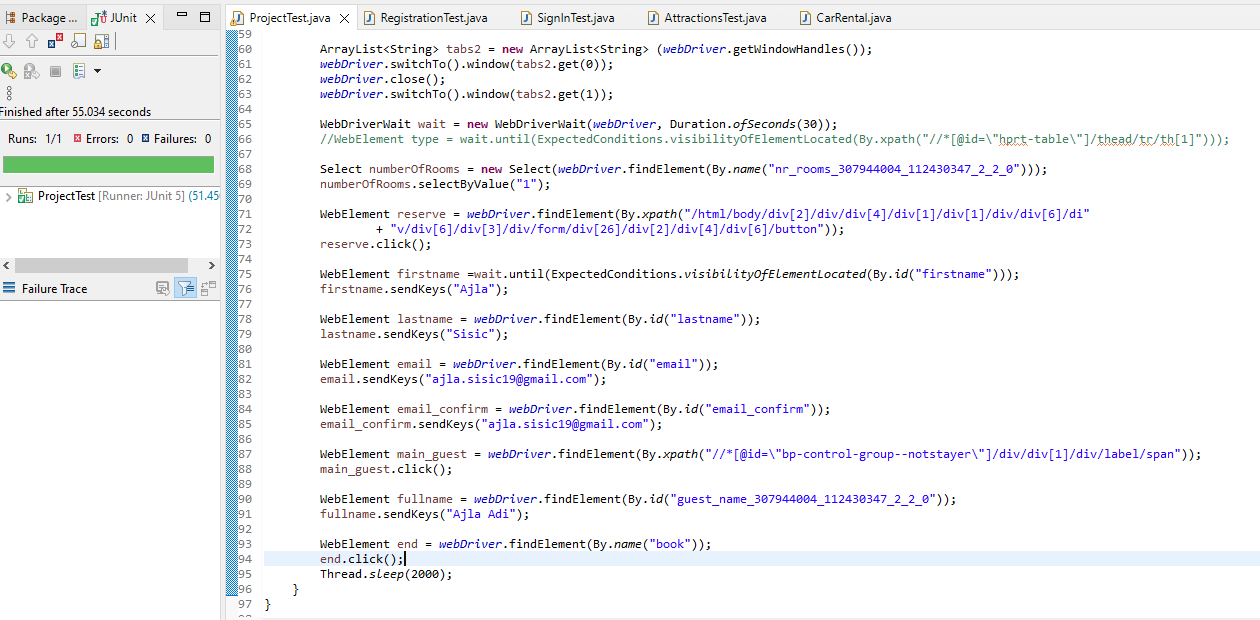




|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Searching attractions | | | | |
| **Description:**  Finding attractions for a specific location | | | | |
| **Pre-condition(s):** | | | | |
| **Test Steps:**  1.Go to landing page  2. Wait until attractions tab load  3. Click attractions  4. Wait until search pops  5. Search for location  6. Click search  7. Wait 2s  8. Toggle filter | **Test Data:**  Amsterdam | **Expected Result:**  To find local attractions for Amsterdam | **Actual Result:**  As expected | **Status:**  PASS |
| **Notes:** | | | | |



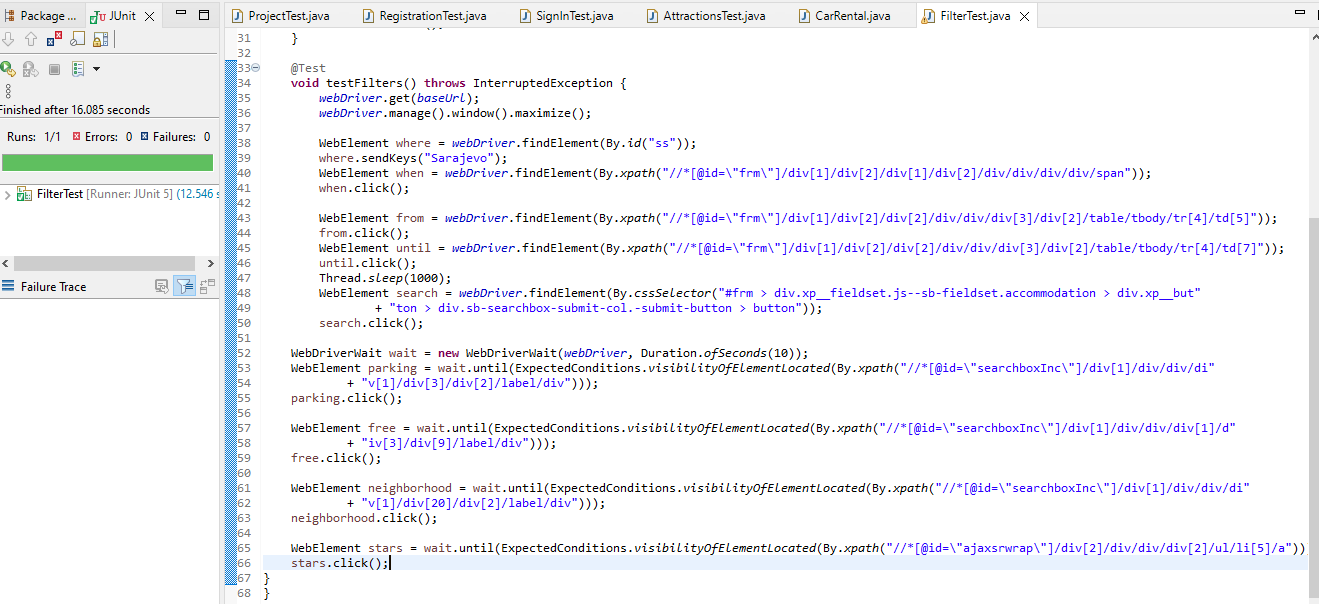
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Booking a room | | | | |
| **Description:**  Reserving a specific hotel we searched for | | | | |
| **Notes:** The reservation cannot be performed till the end since the captcha cannot be solved as a form of prevention against automated attacks. | | | | |
| **Test Steps:**  1. Go to landing page  2. … 9. \*same as test name: find specific hotel\*  10. Select room  11. Click reserve  12. Enter name  13. Enter email  14. Select main guest  15. Enter full name  16. Click reserve | **Test Data:**  Hilton Hotel Belgrade | **Expected Result:**  To come up to credit card details page just before finishing reservation of hotel | **Actual Result:**  As expected | **Status:**  PASS |



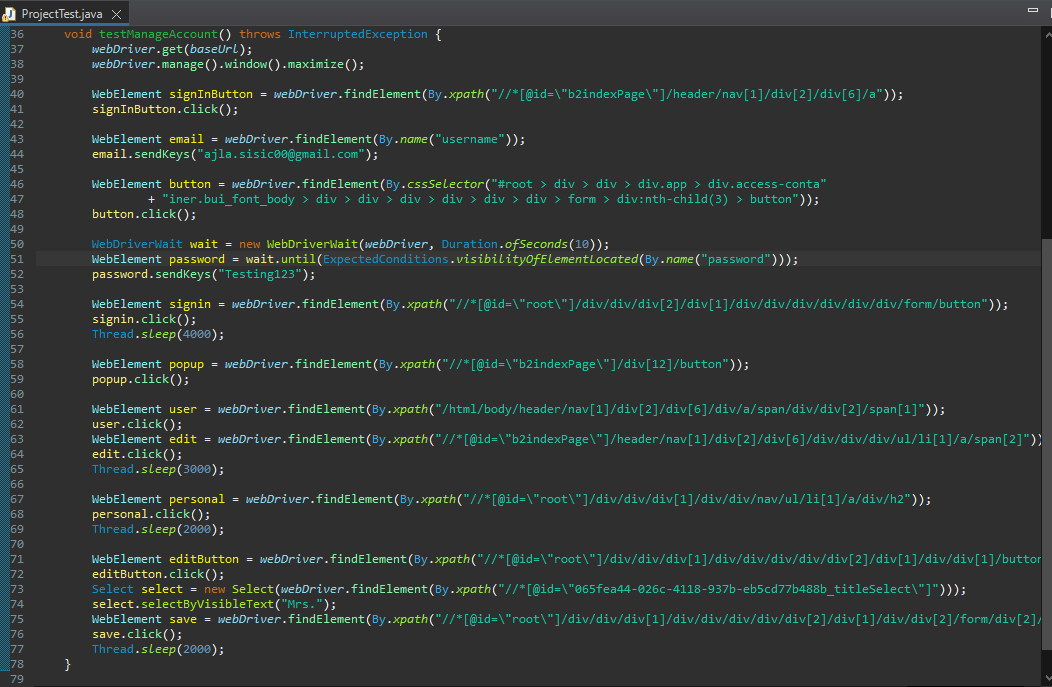
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Museums | | | | |
| **Description:**  Testing the museum search option | | | | |
| **Pre-condition(s):** | | | | |
| **Test Steps:**  1. Go to landing page  2. … 8. \*same as test name: searching attractions\*  9. Click museums  10. Choose date  11. Search museums  12. Wait until number of tickets show  13. Click next | **Test Data:**  Amsterdam | **Expected Result:**  To come to a certain point of searching museums after entering tickets | **Actual Result:**  As expected | **Status:**  PASS |
|  | | | | |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Filters | | | | |
| **Description:**  Trying out different filters in search options | | | | |
| **Pre-condition(s):** | | | | |
| **Test Steps:**  1. Go to landing page  2. Search city  3. Search date  4. Click search  5. Wait until parking filter is loaded  6. Click parking filter  7. Click free cancelation filter  8. Enter neighborhood filter  9. Enter rating filter | **Test Data:**  Sarajevo, Bascarsija | **Expected Result:**  See if filters affect the search parameter | **Actual Result:**  Filters affect the search parameter | **Status:**  PASS |
| **Notes:** | | | | |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Edit user account | | | | |
| **Description:**  Attempting to change user settings | | | | |
| **Pre-condition(s):** Have a registered account | | | | |
| **Test Steps:**  1. Go to landing page  2. Click sign in  3. Enter credentials  4. Sign in  5. Close pop-up  6. Click user account  7. Click edit  8. Choose personal settings  9. Click edit title  10. Select Mrs.  11. Click save | **Test Data:** | **Expected Result:**  To change title from NULL to Mrs. | **Actual Result:**  As expected | **Status:**  PASS |
| **Notes:** This will only work if a pop-up shows up and not too many attempts have been made previously due to human verification | | | | |



# 9. Conclusion

## 9.1. Testing Summary

|  |  |  |  |
| --- | --- | --- | --- |
| **Testing Tool** | **Total Tests** | **Passed Tests** | **Failed Tests** |
| Java programming language, Selenium, Chrome driver - all finished in Eclipse IDE | 10 | 10 | 0 |

## 9.2. Final Thoughts

As this is a worldwide-popular website, I expected that it has protection from automatic usage of the website, where I had many issues while attempting to test it. Since the website changes every minute due to its high usage and many requests that change the website, all web elements of these functionalities that can be found are being changed every minute, making it very hard to pinpoint the element we are trying to test, resulting in test failure.

Unfortunately, it isn’t possible to manipulate such elements however I accepted all tests that passed at least once, which all did. Once the element changes, the test fails as it cannot find the element and continue the test.

Although it is a website that is protected from automated tools such as Selenium, it was interesting to see how well it is created. You cannot perform any of the reservations until the end, even if you leave time to solve the captcha manually.