



National University
of computer and emerging sciences

Software Testing

Project Report

Automation Framework Using Selenium and C#

Group Members:

- | | |
|-----------------|----------|
| 1. Sarfaraz ali | 19k-1281 |
| 2. Shazim Abbas | 19k-1351 |

Introduction

In this project, we created an end-to-end test automation framework using Selenium and C# of an e-commerce website. We created several test cases for all functionalities. The framework was designed to be data-driven, keyword-driven, and based on the Page Object Model (POM) design pattern. The objective was to enhance the efficiency and maintainability of test automation efforts while ensuring robustness and reusability.

Website

[Swag Labs \(saucedemo.com\)](https://saucedemo.com)

Pages

1. Base Page
2. Login
3. Logout
4. Items Cart
5. Items Filter
6. Checkout
7. Test Cases

Test Cases

We created 22 test cases to validate different functionalities of the website and validate the expected results. The test cases are data-driven using DataRow.

1. **TestCase_01** [Category: Login] [Positive]

Login With Correct Username Correct Password

This test case contains 2 test cases for 2 different data rows and it validates the Login functionality by logging in with valid credentials.

2. **TestCase_02** [Category: Login] [Negative]

Login With Empty Username Empty Password

This test case validates the Login functionality by logging in with Invalid credentials.

3. TestCase_03 [Category: Login] [Negative]

Login With Correct Username Empty Password

This test case validates the Login functionality by logging in with Invalid credentials.

4. TestCase_04 [Category: Login] [Negative]

Login With Empty Username Correct Password

This test case validates the Login functionality by logging in with Invalid credentials.

5. TestCase_05 [Category: Login] [Negative]

Login With Correct Username Wrong Password

This test case validates the Login functionality by logging in with Invalid credentials.

6. TestCase_06 [Category: Login] [Negative]

Login With Wrong Username Correct Password

This test case validates the Login functionality by logging in with Invalid credentials.

7. TestCase_07 [Category: Login] [Negative]

Login With Wrong Username Wrong Password

This test case validates the Login functionality by logging in with Invalid credentials.

8. TestCase_08 [Category: AddToCart] [Positive]

Add Item to Cart from Products Page

This test case validates the Add To Cart functionality by adding a product in cart from products page.

9. TestCase_09 [Category: AddToCart] [Positive]

Add Item to Cart from Product Description Page

This test case validates the Add To Cart functionality by adding a product in cart from product description page.

10. TestCase_10 [Category: RemoveFromCart] [Positive]

Remove Item from Cart from Products Page

This test case validates the Remove From Cart functionality by removing a product from cart from shopping cart page.

11. TestCase_11 [Category: RemoveFromCart] [Positive]

Remove Item from Cart from Product Description Page

This test case validates the Remove From Cart functionality by removing a product from cart from product description page.

12. TestCase_12 [Category: Checkout] [Positive]

Item Checkout With Valid Details

This test case validates the Checkout functionality by checking out from shopping cart page with valid details.

13. TestCase_13 [Category: Checkout] [Negative]

Item Checkout With InValid Details

This test case contains 4 test cases for 4 different data rows and it validates the Checkout functionality by checking out from Shopping Cart page with InValid details.

14. TestCase_14 [Category: Logout] [Positive]

Logout from Products Page

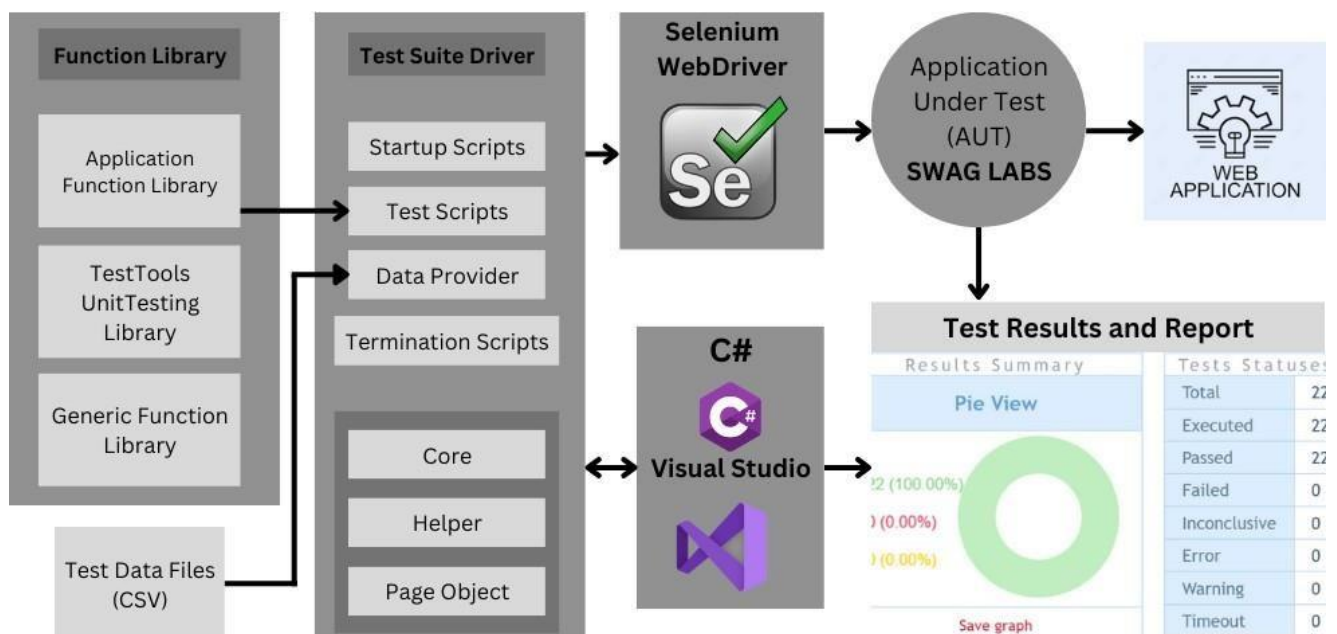
This test case contains 4 test cases for 4 different data rows and it validates the Logout functionality by logging out the from the Menu Options.

15.TestCase_15 [Category: FilterItems] [Positive]

Filter Products

This test case validates the Filter functionality by filtering the items on the Products Page.

Architecture Diagram



Tools / Technologies / Platform

- Selenium WebDriver
- C#
- **IDE:** Visual Studio
- **Browser:** Chrome (ChromeDriver)

Conclusion

We automated the testing of an e-commerce website and created several test cases to validate different functionalities of the website and validated the expected results. The framework's data-driven, keyword-driven, and POM-based approach enhanced the efficiency, maintainability, and reusability of test automation efforts. Automated testing helps to identify bugs and errors early in the development process, resulting in more stable and reliable applications. This framework can serve as a foundation for future test automation tasks, promoting consistency, reliability, and reusability.