® RV Educational Institutions ®
RV College of Engineering ®

Autonomous Institution
Affiliated to Visvesvaraya
Technological University,
Belagavi

Approved by AICTE,
New Delhi

Software Quality Testing and Automation

MSE261T

Experiential Learning Report

On

“Automation testing of e-commerce website using Selenium and Python”

Submitted by

Vandana Pranavi A M 1RV23SE18

Gurukiran P S 1RV23SSE06

Submitted to

Rashmi R

Assistant Professor

Department of Information Science and Engineering

M. Tech in Software Engineering

2024-25

`

TABLE OF CONTENTS

Sl.no	Chapters	Page no.
1	Introduction	1
2	Testing Approach	2
3	Test Suites and execution	3-4
4	Results	5-6
5	Error Analysis and Developer Role	7
6	Conclusions and recommendations	8

`

Chapter 1

Introduction

1.1 Overview

Automated testing is a crucial component of modern software development. This document provides an in-depth analysis of the automated testing process applied to the Magento E-commerce website using Selenium WebDriver with Python. The report includes a structured test approach, test case execution results, and recommendations for improvement.

1.2 Objectives

- Validate core functionalities such as login, product search, selection, and checkout.
- Identify and document failed test cases for further analysis.
- Define the developer's role in diagnosing and fixing detected issues.
- Enhance the overall test automation framework for efficiency and reliability.

`

Chapter 2

Testing Approach

2.1 Type of Testing

Automation Testing – This approach ensures consistent and repeatable test execution while reducing human effort. Automation allows for rapid validation of application functionalities across different test cycles.

2.2 Testing Tool Used

Selenium WebDriver with Python – Selenium is a widely used tool for automating web applications. Python provides robust scripting capabilities that allow interaction with web elements programmatically, making the automation process efficient and scalable.

2.3 Testing Methodology Adopted

The testing methodology follows **Functional Testing** principles, which focus on verifying the core functionalities of the web application. The test script validates the following:

- **User Authentication:** Ensuring users can log in with valid credentials.
- **Product Search and Selection:** Testing search functionality and product filtering.
- **Cart Operations:** Verifying that selected items can be added to the cart.
- **Checkout Process:** Ensuring users can initiate the checkout process.

A structured test case approach is adopted, using explicit waits to handle dynamic elements and exception handling to capture errors effectively.

`

Chapter 3

Test Suites and Execution

3.1 Test Suite 1: Login and Product Search (Including Failed Cases)

Test Case ID	**Test Scenario**	**Expected Result**	**Actual Result**	**Status**
TC_01	Navigate to Magento website	Website loads successfully	Website loads successfully	Passed
TC_02	Click on Sign In link	Sign-in page appears	Sign-in page appears	Passed
TC_03	Enter valid credentials and login	User is logged in	User is logged in	Passed
TC_04	Search for a Jacket	Search results display relevant products	Search results displayed	Passed
TC_05	Select a product	Product page opens	Product page opens	Passed
TC_06	Select size 'M'	Size 'M' should be selected	Size 'M' selected	Passed
TC_07	Select color 'Purple'	Purple color should be selected	Blue color option was not found	**Failed**
TC_08	Add product to cart	Product should be added successfully	Product not added due to missing selection	**Failed**

`

3.2 Test Suite 2: Cart and Checkout (Passed Cases)

Test Case ID	Test Scenario	Expected Result	Actual Result	Status
TC_09	Click on Cart icon	Cart dropdown should appear	Cart dropdown appears	Passed
TC_10	Proceed to Checkout	Checkout page should load	Checkout page loads successfully	Passed

3.3 Observations from Test Execution

- **Successful Tests:** Most core functionalities, including login, search, and checkout, worked as expected.
- **Failed Tests:** Issues were found in selecting a product color and adding items to the cart.
- **Potential Causes of Failures:** Dynamic element loading, incorrect element selection, or missing UI components.

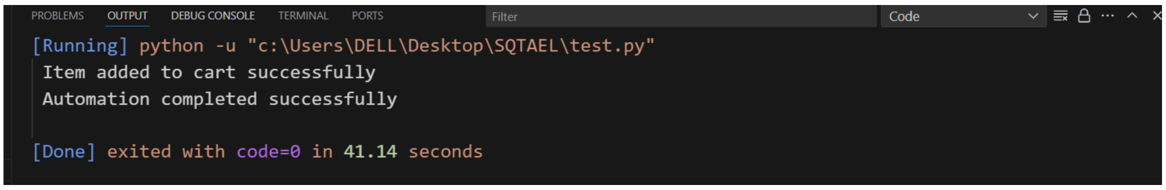
`

Chapter 4

Results

4.1 Screenshots of results

4.1.1 Output of the executed code



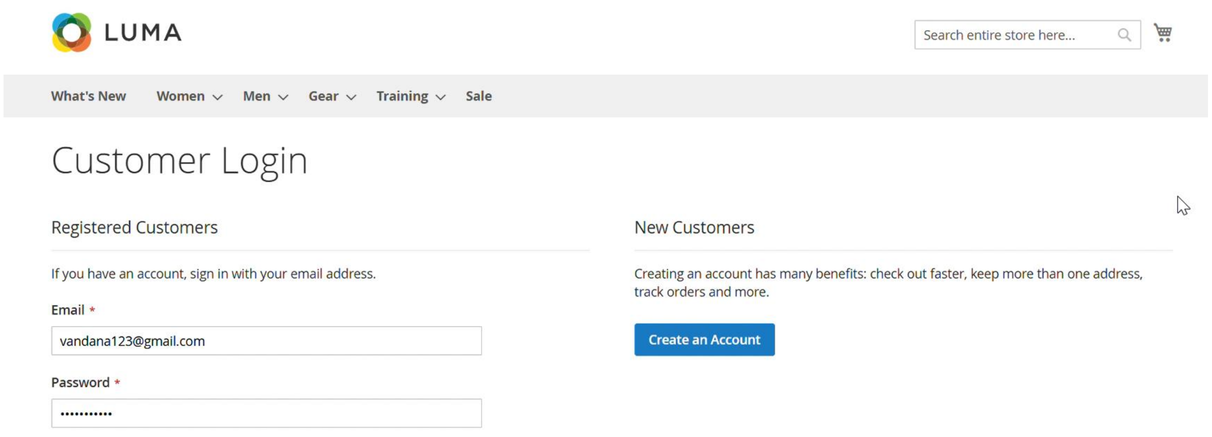
```
[Running] python -u "c:\Users\DELL\Desktop\SQTAEL\test.py"
 Item added to cart successfully
 Automation completed successfully

[Done] exited with code=0 in 41.14 seconds
```

Fig : 4.1.1 Outputs of the executed code

20250322-1016-30.99
02872.mp4

4.1.2 Automation testing of logging in with default credentials



Customer Login

Registered Customers

If you have an account, sign in with your email address.

Email *

vandana123@gmail.com

Password *

•••••••••••

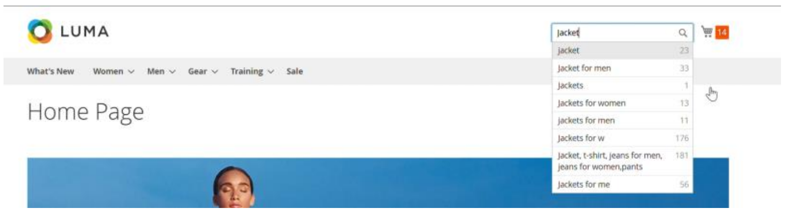
New Customers

Creating an account has many benefits: check out faster, keep more than one address, track orders and more.

Fig : 4.1.2 Automation testing of logging in with default credentials

`

4.1.3 The Jacket item is being searched in the webpage



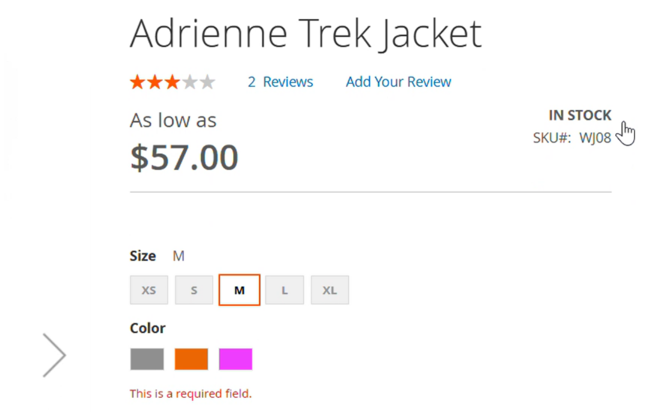
Home Page

Jacket

jacket	23
Jacket for men	33
Jackets	1
Jackets for women	13
jackets for men	11
Jackets for w	176
Jacket, t-shirt, jeans for men, jeans for women,pants	181
Jackets for me	56

Fig : 4.1.3 The Jacket item is being searched in the webpage

4.1.4 Item is being added to the cart



Adrienne Trek Jacket

★★★☆☆ 2 Reviews Add Your Review

As low as

$57.00

IN STOCK

SKU#: WJ08

Size M

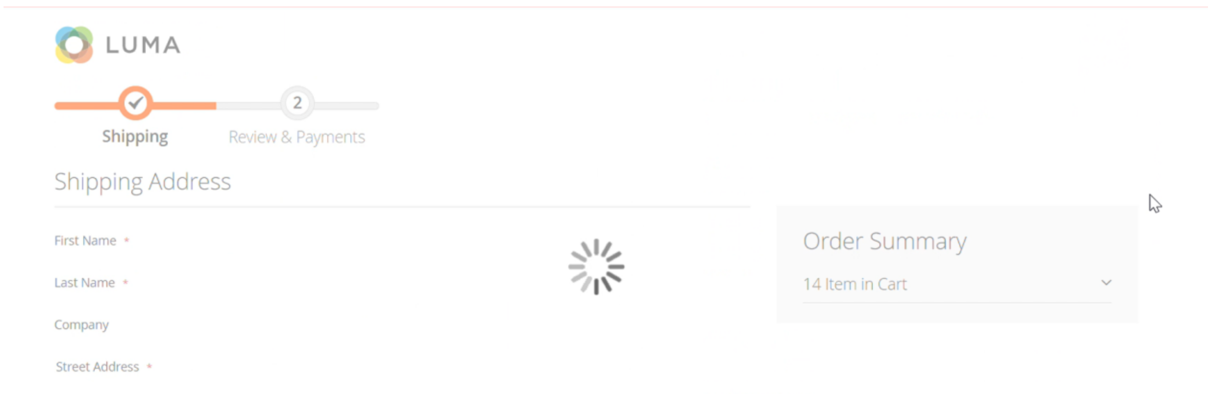
XS S M L XL

Color

This is a required field.

Fig : 4.1.4 Item is being added to the cart

4.1.4 Item placed in the cart for checkout



Shipping Address

First Name *

Last Name *

Company

Street Address *

Order Summary

14 Item in Cart

Fig : 4.1.5 Item placed in the cart for checkout

`

Chapter 5

Error Analysis and Developer Role

5.1 Identified Issues

- **Failure in selecting the 'Purple' color option** – The script could not locate the required element.
- **Cart addition failure** – The script could not proceed due to the missing color selection.
- **Potential UI Changes:** If the UI elements have changed, updating the selectors is necessary.

5.2 Developer Role in Correcting the Error

To fix the identified issues, the developer should:

1. **Debug the script** – Verify if the correct element locators are used.
2. **Ensure element visibility** – Implement explicit waits to ensure elements are loaded before interaction.
3. **Update element selectors** – If the UI structure has changed, update XPath or CSS selectors accordingly.
4. **Implement better error handling** – Use try-except blocks to capture and log errors dynamically.
5. **Perform retesting** – Validate that fixes resolve the issue without breaking other functionalities.

`

Chapter 6

Conclusion and Recommendations

6.1 Conclusion

The automation script successfully validates the critical functionalities of the Magento website. Failures identified in the test execution indicate areas that need improvement in the automation framework. Addressing these failures will improve test stability and reliability.

6.2 Recommendations

- **Enhance Element Identification:** Use more robust locators such as data attributes instead of class names that might change.
- **Increase Test Coverage:** Expand test scenarios to include negative testing (e.g., incorrect credentials, unavailable products, etc.).
- **Optimize Script Execution:** Use parallel execution to speed up test runs.
- **Automate Error Logging:** Implement automated error reporting to identify issues quickly.

By implementing these recommendations, a more reliable and scalable automated testing framework can be established.

`