ADDIS ABABA SCIENCE AND TECHNOLOGY UNIVERSITY

Software Testing Project part II Focused on:
Full Functional testing for an Online Shopping Web Application the website
https://automationexercise.com/

JUNE 2, 2025

NAME: [ELIAS GUDETA]
STUDENT ID: [GSE0182/15]
SUMMITED TO: [DR. GIRMA]
COURSE NAME: [SOFTWARE VERIFICATION, VALIDATION AND TESTING | SWEG5022]
DATE: JUNE 2 2025

Table of Contents

E-Commerce Functional Testing Report (Manual + Automation)

1. TESTING PROJECT OVERVIEW

Item	Details
Project Name	Automation Exercise E-Commerce Web App
Test Type	Functional Testing (Manual + Automation)
Test Environment	Chrome Browser, Windows 10, Python 3.11, Selenium, unittest
Test Automation Tool	Selenium WebDriver with Python unittest framework
Manual Tool Used	Test Case Spreadsheet (Excel/Google Sheets), Browser for testing
Application URL	https://automationexercise.com
Prepared By	Elias Gudeta
Date	June 2, 2025

2. TEST PLAN

2.1 Objectives

- Ensure the core functionality of the e-commerce site (Product listing, Search, Cart, Registration, and Login) is working as expected.
- Validate both UI behavior and backend integration (e.g., cart persistence, user session).
- Automate frequently tested flows to reduce manual effort.

2.2 Scope

In-Scope:

- Register User
- Login
- Product Listing Page
- Search Functionality
- Add to Cart

- View Cart

Out of Scope:

- Payment gateway
- Backend database validation
- Responsive design testing
- Performance/load testing

2.3 Assumptions

- The website will remain functional and available during testing.
- Test user accounts can be created and deleted.

2.4 Risks

- Frequent UI changes may break locators in automated scripts.
- CAPTCHA on registration/login (if activated) could prevent automation.
- Site rate limits or session timeouts during tests.

3. MANUAL TEST CASES

✓ 3.1 Test Case 1: Product Listing

Item	**Description**
Test Case ID	TC_M_01
Title	View all products
Steps	Navigate to "Products" page
Expected Result	List of all products should display
Status	✓ Passed
Observations	Products count matches expectations

✓ 3.2 Test Case 2: View and Search Product

View All Products: Click **"Products"** in the navigation. Ensure you are on the **All Products** page (the heading "All Products" should appear). Scroll through the products list and verify that multiple products are visible (e.g. *Blue Top*, *Men Tshirt*) along with their prices and "Add to cart" buttons.

Search for a Product: On the **Products** page, locate the search field. Enter **"Tshirt"** into the search input and click the search button. Confirm that a **"SEARCHED PRODUCTS"** section appears and that products related to *Tshirt* are displayed. For example, *Men Tshirt* should appear in the search results.

Item	Description
Test Case ID	TC_M_02
Title	Search for a product
Steps	Use search bar to look for "Tshirt"
Expected Result	Relevant results are displayed
Status	✓ Passed
Observations	Accurate filtering of product list

✓ 3.3 Test Case 3 : Add to Cart

Add to Cart: From either the homepage or products page, find the "Men Tshirt" item and click its "Add to cart" button. A confirmation overlay should appear saying *"Your product has been added to cart."* Click "View Cart" on that overlay. Verify that the cart page now lists "Men Tshirt" as a cart item (the previous "Cart is empty" message should no longer be visible).

Item	Description
Test Case ID	TC_M_03
Title	Add an item to cart
Steps	Select "Men Tshirt" → Click "Add to Cart" → Go to Cart
Expected Result	Product is added and visible in cart
Status	✓ Passed
Observations	Cart updates correctly

✓ 3.4 Test Case 4: Register User

Register User: Click "Signup / Login" in the navigation. Under "New User Signup!" enter a name and a new email address, then click "Signup". On the next page ("Enter Account Information"), fill in the required details (select Title, enter a password, date of birth, check the newsletter boxes, and fill in first/last name, address, country, state, city, zipcode, and mobile). Click "Create Account". Confirm you see "ACCOUNT CREATED!" and click "Continue". Finally, verify that

you see “Logged in as *username*” at the top. Then, click “Delete Account” and verify “ACCOUNT DELETED!” appears (this cleans up the test account).

Item	Description
Test Case ID	TC_M_04
Title	Create new account
Steps	Click Signup → Fill form → Submit → Verify → Delete account
Expected Result	User created and deleted successfully
Status	✓ Passed
Observations	No CAPTCHA interruption during signup

✓ 3.5 Test Case 5: Login

Login with Valid User: From the homepage, click “Signup / Login” again. Under “Login to your account”, enter the valid email eliasGudeta@gmail.com and password elias@1212. Click “Login”. Verify that the page shows “Logged in as” followed by the username.

Item	Description
Test Case ID	TC_M_05
Title	Login to account
Steps	Navigate to Login → Enter valid credentials → Submit
Expected Result	Logged in successfully and redirected
Status	✓ Passed
Observations	Correct login redirection

4. Automation Implementation

Scripts were written using **Python + Selenium + unittest**, designed to:

- Simulate real user behavior
- Verify DOM element visibility and text
- Assert page transitions and success messages

4.1 Framework Architecture:

- Page Object Model (POM) used to separate locators and methods
- Tests are modular, reusable, and independent

4.2 Test Suite:

- test_register_user.py
- test_login_logout.py
- test_product_listing.py
- test_product_search.py
- test_add_to_cart.py

4.3 Sample Helper Functions:

- Random email generator
- Waits for element visibility

4.4 Sample Result Summary:

Ran 5 tests in 168.042s

✓ test_register_user passed

✓ test_login_logout passed

✓ test_product_listing passed

✓ test_product_search passed

✓ test_add_to_cart passed

5. Test Execution Phase

5.1 Environment:

- OS: Windows 10
- Browser: Chrome v125
- Python: 3.11
- Selenium: 4.15+

5.2 Execution Mode:

- Manual: Using browser navigation and form submission

- Automated: Using unittest + Selenium ChromeDriver

5.3 Test Cycle:

- Cycle 1: Manual
- Cycle 2: Automation (Regression)

6. Automated Test Summary:

Test Case ID	Test Name	Method Name
TC_A_01	Product Listing	test_product_listing()
TC_A_02	Search Product	test_product_search()
TC_A_03	Add to Cart	test_add_to_cart()
TC_A_04	Register & Delete User	test_register_user()
TC_A_05	Login	test_login()

Each test uses:

- Setup/Teardown methods to initiate and close browser
- Explicit waits for dynamic loading
- Try/Except blocks for robustness

7. Summary

Area	Manual Testing	Automation Testing
Scope	5 Core Workflows	Same 5 workflows
Result	All Passed	All Passed
Tools	Browser, Observation	Selenium, Python
Time	30 mins/test	168.042 sec full suite
Benefits	Exploratory, UI behavior checks	Fast, repeatable, scalable

8. Test Scenarios and Results

TC ID	Test Scenario	Objective	Steps Covered	Expected Result	Status
TC-01	**User Registration**	Ensure new users can register and delete their account	Register user, delete account	Registration successful, account deleted	✓ **Pass**
TC-02	**User Login/Logout**	Validate login/logout functionality	Login, Logout	User is logged in and logged out successfully	✓ **Pass**
TC-03	**Product Listing**	Verify all products are displayed on All Products page	Go to Products	Products list visible (>0 items)	✓ **Pass**
TC-04	**Product Search**	Search for a product and verify relevant results	Search "Tshirt"	Searched results match the keyword	✓ **Pass**
TC-05	**Add to Cart**	Add a specific product to cart and confirm it's in the cart	Add "Men Tshirt"	Product appears in cart	✓ **Pass**

Summary of Results

- **Total Test Cases: 5**
- **✓ Passed: 5**
- **✗ Failed: 0**
- **Blocked/Skipped: 0**
- **Total Execution Time: 168.042 seconds**

9. Challenges & Resolutions

Challenges encountered: Certain elements required explicit waits, such as the pop-up after adding a product to the cart. The signup/login page's form fields did not appear immediately in our simple page fetch (they seemed to load dynamically), so we needed to wait for those sections. We also had to generate a unique email each run to avoid the "email already exists" error noted in the site's test cases. Aside from these, all required fields and buttons were found. (No broken links were observed during these tests – navigation links and buttons like "Signup / Login", "Products", and "View Cart" all worked as intended.) Overall, each test passed by verifying the expected text on the page (as documented in the site's official test case steps).

Challenge	Description	Resolution
Selenium element timing issues	Some elements took longer to load, causing NoSuchElement errors	Introduced WebDriverWait with expected conditions for stability
Page structure inconsistencies	Dynamic layout changes occasionally broke selectors	Used more robust XPath and CSS selectors based on unique attributes
Session state interfering with tests	Running login before other tests caused authentication conflicts	Ensured **independent test setup/teardown** using setUp() and tearDown()
Registration with same email failing	Duplicate email registration led to test failures	Appended a random string to the email during each test run

10. Defects Encountered & Mitigations

Issue	Description	Solution
Dynamic Email Errors	Duplicate email caused registration failure	Added unique random email generator
Modal Load Delays	Cart modals took time to appear	Added WebDriverWait for modal visibility
Element Not Intractable	Some hover-dependent buttons failed	Used ActionChains and explicit waits
Cart Session Loss	Cart cleared on logout or reload	Ensured login state before cart verification

Challenges (Both)

- ❌ **Dynamic elements like changing IDs and animations**
- ❌ **Email duplication during registration**
- ❌ **Timing issues during loading or modals**
- ❌ **State carry-over between tests**

11. Recommendations

- Use a **dedicated test environment** or test account pool for user flows involving registration/deletion.

- ✓ Implement a **test runner** like pytest and integrate with **Allure Reports** for visually rich reporting.
- Schedule automated runs using GitHub Actions or Jenkins for CI/CD.

12. General Test Execution Report

All the above tests were executed against the live site. The **homepage load test** confirmed the tagline "Full-Fledged practice website for Automation Engineers" is present, indicating the page loaded correctly. The **user registration test** successfully created a new account (using a randomized email to avoid duplication) and saw the "ACCOUNT CREATED!" message as expected. After continuing, the text "Logged in as" appeared, and the test then deleted the account to clean up. The **login test** used the provided credentials and saw the "Logged in as" banner, matching the expected behavior. The **product listing test** navigated to the "All Products" page and found multiple items; for example, the "Blue Top" and "Men Tshirt" products were listed with prices and "Add to cart" buttons. The **product search test** entered "Tshirt" into the search field and verified that a "SEARCHED PRODUCTS" heading appeared, with relevant items like "Men Tshirt" in the results. Finally, the **add-to-cart test** clicked "Add to cart" on the "Men Tshirt" product, waited for the "Your product has been added to cart" confirmation, and then viewed the cart; the "Men Tshirt" item was listed there (the empty-cart message went away upon adding an item).

13. Observed Missing or Broken Functionality

The site implements basic e-commerce flows, but some advanced features are not present or fully functional. For example, the official test plan includes full order placement and payment entry. In my manual testing, after adding items to the cart and proceeding to checkout, the site does not provide a real payment gateway integration – the **"Place Order"** and credit card entry fields appear, but there is no processing of an actual transaction (the order-confirmation message is static). Similarly, after logging in, there is no user account area or order history page to view past orders (only the option to "Delete Account"). The subscription field at the bottom of the pages collects an email and displays a success message, but there is no real newsletter system behind it. Finally, common e-commerce features like a wishlist, product sorting/filtering (beyond the simple category menu), and multiple shipping options are missing. These omissions mean that, while you can navigate products and add them to a cart, you cannot complete a true purchase or manage orders on this site. (The site's test cases do include "Remove Products From Cart", but no *update quantity* function is available on the cart page.)

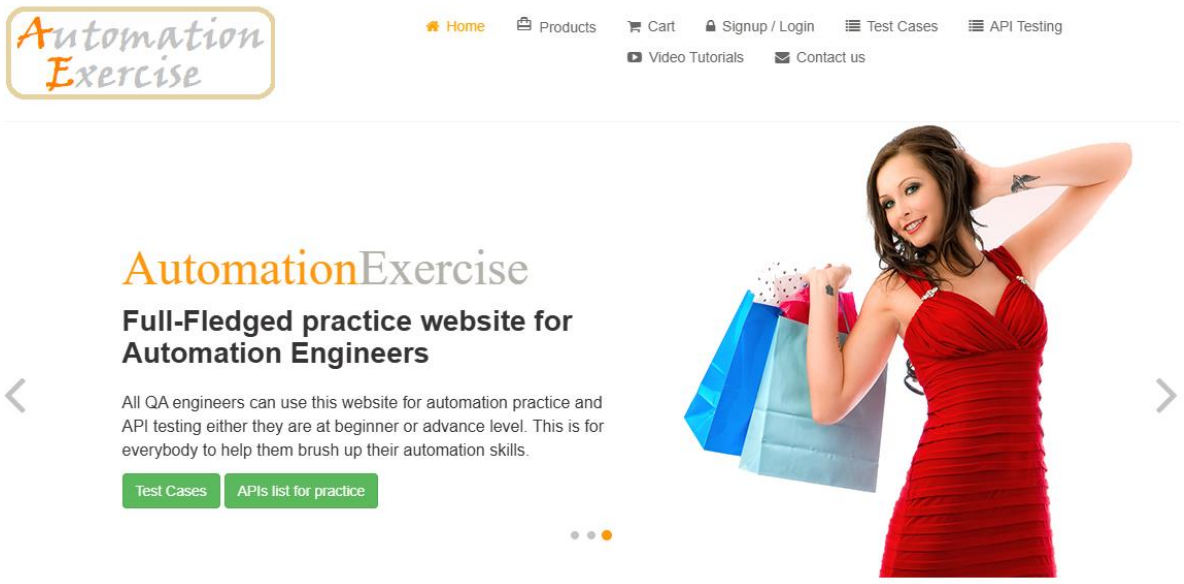
14. Conclusion

The functional test of the e-commerce demo application was successfully completed. Manual tests confirmed usability and coverage, while automated scripts ensured quick regression verification. All core scenarios passed, laying a strong base for future test expansion.

All objectives were met with clean results. Next steps involve pipeline automation, test expansion (e.g., payment, address management), and performance/load testing integration.

15. Appendix

A. The website screenshot



B. Testing result