

Automation Exercise

Automation Exercise – Web UI Automation Testing

Domain: E commerce

Project Overview

The Automation Exercise website offers multiple user functionalities such as user registration, login, product browsing, cart management, and order placement.

This project focuses on automating the end-to-end test scenarios of the website using industry-standard tools and frameworks.

The automation suite ensures that the major functionalities of the application work as expected and helps identify regression issues quickly and efficiently.

Project Objective

The key objectives of this project are:

- To automate high-priority test cases on the Automation Exercise website.
- To validate core functionalities such as:
 - User account creation, login, logout.
 - Product search, add-to-cart, and checkout workflows.
 - Subscription flows.
 - Category navigation.
- To develop a maintainable and reusable automation framework.
- To reduce manual effort and improve testing efficiency.
- To ensure quick detection of application bugs during development cycles.

Scope of the Project

- **UI automation of major user flows.**
- **Functional testing of:**
 - **Registration**
 - **Login/Logout**
 - **Product browsing**
 - **Searching**
 - **Add to cart**
 - **Checkout**
 - **Category handling**
 - **Subscription feature**

Tools & Technologies Used:

- **Programming Language:** Java
- **Automation Framework:** Selenium WebDriver
- **Test Management & Execution:** TestNG
- **Reporting:** Extent Reports
- **Data Handling:** Apache POI (for Excel integration, data-driven testing)
- **Build & Dependency Management:** Maven (POM)
- **Version Control:** GitHub
- **IDE:** (e.g., Eclipse/IntelliJ IDEA)
- **Browsers:** Chrome

Project Structure

```
AutomationExercise/
|— src/
|   |— main/
|       |— java/
|           |— drivers/      # WebDriver setup & management
|           |— pages/       # Page Object Model (POM) classes
|           |— utils/       # Utility classes
|           |— resources/
|           |   |— config.properties # Project configuration
|           |   |— log4j2.xml      # Logging configuration
|           |
|           |— test/
|               |— java/
|                   |— dataProviders/ # Data-driven testing providers
|                   |— listeners/    # TestNG Listeners (for reports, logging)
|                   |— tests/       # Actual test cases
|                   |— resources/
|                   |— testData/    # Test data files
|
|— logs/
|   |— automation.log      # Runtime execution logs
|
|— target/                # Build output
|— test-output/           # TestNG default reports
```

|— pom.xml # Maven build configuration
|— testng.xml # TestNG suite file

Test Cases

TC ID	Module	Scenario	Test Case Name	Status
TC01	User Registration	Create new user	Register User	Pass
TC02	Login	Valid login	Login with correct email & password	Pass
TC03	Login	Invalid login	Login with incorrect email & password	Pass
TC04	Login/Logout	User logout	Logout User	Pass
TC05	User Registration	Duplicate registration	Register User with existing email	Pass
TC06	Contact	Form submission	Contact Us Form	Pass
TC07	Test Cases	Page validation	Verify Test Cases Page	Pass
TC08	Products	View products	Verify All Products & Product Detail Page	Pass
TC09	Products	Search functionality	Search Product	Pass
TC10	Subscription	Newsletter subscription	Verify Subscription on Home Page	Pass
TC11	Subscription	Subscription inside cart	Verify Subscription in Cart Page	Pass
TC12	Cart	Add items	Add Products in Cart	Pass
TC13	Cart	Quantity update	Verify Product Quantity in Cart	Pass
TC14	Checkout	Register during checkout	Place Order: Register While Checkout	Pass
TC15	Checkout	Register before checkout	Place Order: Register Before Checkout	Pass
TC16	Checkout	Login before checkout	Place Order: Login Before Checkout	Pass
TC17	Cart	Remove items	Remove Products From Cart	Pass
TC18	Categories	View category products	View Category Products	Pass

Test Data

- **Login Data:** Read from *LoginData.xlsx*; each Excel row becomes one test case.

- **Signup Data:** Read from *SignUpData.xlsx*; used for multiple signup scenarios.
- **Contact Us Data:** Hardcoded sample inputs for name, email, subject, message, and file.
- **Search Products:** Hardcoded keywords like "Jeans" and "Cotton" for search testing.
- **Product ID:** Hardcoded product ID pairs for validating product-related tests.
- **Place Order:** Read from *placeOrder.xlsx*; used for verifying order placement flow.
- **Place Order with Login:** Read from *placeOrderWithLoginData.xlsx*; supports end-to-end login + order testing.

Test Reporting and Logging

- **TestNG Reports:** Provides default execution results and pass/fail status.
- **Extent Reports:** Generates detailed, interactive HTML reports with test steps and logs.
- **Logger:** Captures real-time execution logs for each test step.
- **Screenshots:** Captured on failures or critical steps for visual verification.

```

2025-11-28 19:04:24 DEBUG PackageRelationshipCollection:307 - Parsing relationship: /xl/_rels/workbook.xml.rels
2025-11-28 19:04:24 DEBUG PackageRelationshipCollection:307 - Parsing relationship: /xl/worksheets/_rels/sheet1.xml.rels
2025-11-28 19:04:24 DEBUG PackageRelationshipCollection:307 - Parsing relationship: /_rels/.rels
2025-11-28 19:04:34 INFO BaseTest:28 - ----- TESTCASE 1 : REGISTER USER -----
2025-11-28 19:04:34 INFO BaseTest:35 - Home Page is displayed successfully
2025-11-28 19:04:36 INFO BaseTest:39 - Navigated to Signup/Login page
2025-11-28 19:04:36 INFO BaseTest:44 - Signup section is visible
2025-11-28 19:04:38 INFO BaseTest:50 - Performed signup with username: users011 and email: users011@gmail.com
2025-11-28 19:04:38 INFO BaseTest:55 - Account Information section is visible
2025-11-28 19:04:44 INFO BaseTest:75 - Filled all account details and submitted the form
2025-11-28 19:04:45 INFO BaseTest:80 - Account has been created successfully
2025-11-28 19:04:49 INFO BaseTest:86 - User logged in successfully:Logged in as users011
2025-11-28 19:04:50 INFO BaseTest:90 - Clicked on Delete Account button
2025-11-28 19:04:50 INFO BaseTest:95 - Account has been deleted successfully
2025-11-28 19:04:59 INFO BaseTest:28 - ----- TESTCASE 1 : REGISTER USER -----
2025-11-28 19:04:59 INFO BaseTest:35 - Home Page is displayed successfully
2025-11-28 19:05:00 INFO BaseTest:39 - Navigated to Signup/Login page
2025-11-28 19:05:00 INFO BaseTest:44 - Signup section is visible
2025-11-28 19:05:01 INFO BaseTest:50 - Performed signup with username: users012 and email: users012@gmail.com
2025-11-28 19:05:02 INFO BaseTest:55 - Account Information section is visible
2025-11-28 19:05:06 INFO BaseTest:75 - Filled all account details and submitted the form
2025-11-28 19:05:06 INFO BaseTest:80 - Account has been created successfully
2025-11-28 19:05:09 INFO BaseTest:86 - User logged in successfully:Logged in as users012
2025-11-28 19:05:10 INFO BaseTest:90 - Clicked on Delete Account button
2025-11-28 19:05:10 INFO BaseTest:95 - Account has been deleted successfully
2025-11-28 19:05:18 INFO BaseTest:28 - ----- TESTCASE 1 : REGISTER USER -----
2025-11-28 19:05:18 INFO BaseTest:35 - Home Page is displayed successfully
2025-11-28 19:05:23 INFO BaseTest:39 - Navigated to Signup/Login page
2025-11-28 19:05:23 INFO BaseTest:44 - Signup section is visible
2025-11-28 19:05:23 INFO BaseTest:50 - Performed signup with username:  and email:
2025-11-28 19:05:34 DEBUG PackageRelationshipCollection:307 - Parsing relationship: /xl/_rels/workbook.xml.rels
2025-11-28 19:05:34 DEBUG PackageRelationshipCollection:307 - Parsing relationship: /xl/worksheets/_rels/sheet1.xml.rels
2025-11-28 19:05:34 DEBUG PackageRelationshipCollection:307 - Parsing relationship: /_rels/.rels

```

Getting Started

Gmail

Q

Nov 28, 2025 11:08:24 am

Started

Nov 28, 2025 11:08:24 am

Ended

Nov 28, 2025 11:14:14 am

Tests Passed

21

Tests Failed

3

Tests

Pass

Fail

21 tests passed
3 tests failed, 0 skipped, 0 others

Log events

Pass

Fail

21 events passed
6 events failed, 0 others

Timeline

Q

Nov 28, 2025 11:08:24 am

Tests

testUserRegistrationFlow

11:08:35 am / 00:00:19:533

Pass

testUserRegistrationFlow

11:09:02 am / 00:00:13:941

Pass

testUserRegistrationFlow

11:09:22 am / 00:00:17:331

Pass

testLogin

11:09:45 am / 00:00:09:565

Pass

testLogin

11:10:01 am / 00:00:09:662

Pass

testLoginWithInvalidData

11:10:17 am / 00:00:02:685

Pass

logoutTest

11:10:22 am / 00:00:11:079

Pass

testUserRegistrationFlow

11:28.2025 11:08:35 am 11:28.2025 11:08:55 am 00:00:19:533 #test-id=1

STATUS

TIMESTAMP

DETAILS

Pass

11:08:55 am

Test Passed

Q

Nov 28, 2025 11:08:24 am

Exception

org.openqa.selenium.NoSuchElementException

1 tests

2

org.openqa.selenium.ElementClickInterceptedException

2 tests

1 failed

2

org.openqa.selenium.NoSuchElementException

11:14:01 am

testViewCategoryProducts

Challenges & Solutions

1. **Dynamic Elements** – Used stable locators + explicit waits.
2. **Synchronization** – Added WebDriverWait, FluentWait, jQuery AJAX wait.
3. **Browser Compatibility** – Used WebDriverManager + browser factory.
4. **Test Data Management** – Externalized data using Excel/POI + DataProvider.

Conclusion

The Automation Exercise framework provides a clean and well-structured solution for UI test automation using Selenium and TestNG. With the Page Object Model, reusable utilities, TestNG suites, and Maven integration, the framework ensures easy maintenance, scalability, and efficient test execution. It serves as a solid foundation for building reliable and extendable UI automation suites.

Ultimately, this setup serves as a strong foundation for expanding automation coverage, improving test reliability, and supporting continuous delivery in modern software development environments.

Github : <https://github.com/AnaghaAni/AutomationExercise.git>