

## **Capstone Project**

### **Batch:**

**"Objective:** Develop a BDD Cucumber + Maven + TestNG + Java + Selenium Web Driver automation framework using the Page Object Model (POM). Implement exception handling and generate Cucumber and Allure Reports for test execution.

**Website for Testing:** <https://magento.softwaretestingboard.com/>

### **Project Requirements:**

#### **Framework Setup:**

Create a Maven-based project in Java.

Use TestNG for test execution and reporting.

Implement Selenium Web Driver for browser interactions.

Design the framework using the Page Object Model (POM) to achieve reusability. Feature File & Scenarios (At least 5 scenarios required)

#### **Scenario 1: User Registration with Invalid Data Handling**

Open the homepage.

Click the link "Create an account".

Enter invalid data (e.g., missing email or password).

Click on the button "Create an Account". Verify that an error message appears.

Enter valid details and complete registration.

Verify that the user is registered successfully.

## **Scenario 2: Login, Product Search and Add to Cart**

Open the homepage.

Click the link "SignIn".

Enter the valid required details to Login.

Search for a product (Ex., "Shirts", and select "Shirts for men" from the auto-suggested item).

Click or Select a product from the Search Results page.

Click on "Add to Cart".

Verify that the product is added to the cart.

## **Scenario 3: Checkout Process with Invalid Data Handling**

Open the homepage.

Click the link "SignIn".

Enter the valid required details to Login.

Click "Proceed to Checkout" on Cart Icon.

Enter invalid shipping details (Ex., invalid Zip/Postal Code). Verify that an error message appears.

Enter valid details and complete the checkout.

Verify that the order confirmation message is displayed.

## **Scenario 4: Login with Valid & Invalid Credentials**

Open the homepage.

Click the link "SignIn".

Enter invalid credentials and verify error message appears.

Enter valid credentials and log in successfully.

Verify that "My Account" is displayed after login.

### **Scenario 5: Adding Multiple Products to Cart & Validating Cart Summary**

Open the homepage.

Click the link "SignIn".

Enter valid credentials and log in successfully.

Search and add two different products to the cart. (Ex., Select Men -> Tops -> Jackets and Men -> Tops -> Tees)

Navigate to the cart page.

Verify that the correct products and total price are displayed.

### **Scenario 6: Logging Out and Verifying Session End**

Log in with a valid account.

Click "Logout" from "My Account".

Verify that the session ends and the user is redirected to the homepage.

### **Exception Handling:**

Handle NoSuchElementException for missing elements. Implement Try-Catch blocks in at least one test. Use custom exception messages to log errors properly.

### **Reporting:**

Generate Cucumber Reports for test execution.

Integrate Allure Reports for better visualization of test results.

### **Test Execution with Parallel Browsers:**

Configure TestNG XML to execute tests in parallel on Edge and Chrome browsers. Implement WebDriverManager for browser setup. Ensure tests run independently without dependencies.

**Data-Driven Testing:**

Use Apache POI to read test data from an Excel sheet.

Implement parameterized tests to fetch input values dynamically.

**Submission Requirements:**

GitHub Repository containing the framework code. README.md with project setup instructions. Allure Reports screenshots from test execution.