****

**Selenium Automation Of**

**NopCommerce Web Application**

**Prepared By:- Shivam Shivhare**

**Table Of Content**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No.** | **Content** | | **Page No.** |
| 1. | Introduction | | 3 |
| 2. | Objectives | | 3 |
| 3. | Project Overview | | 3 |
| 4. | Project Scope | | 4 |
|  | 4.1 | In Scope | 4 |
|  | 4.2 | Out Of Scope | 4 |
| 5. | Tools & Technologies | | 5 |
| 6. | Test Strategy | | 5 |
| 7. | Test Scenarios & Test Cases | | 5 |
|  | 7.1 | Launch Browser & Navigate to Homepage | 5 |
|  | 7.2 | User Registration (Valid/Invalid Data) | 6 |
|  | 7.3 | User Login (Valid/Invalid & My Account Verification) | 6 |
|  | 7.4 | Submodule Verification (Electronics, Apparel, Books, etc.) | 7 |
|  | 7.5 | Product Selection & Add to Cart | 8 |
|  | 7.6 | Shopping Cart Validation | 8 |
|  | 7.7 | Checkout Process Validation | 9 |
| 8. | Automation Workflow | | 9 |
| 9. | Test Results & Observations | | 10 |
| 10. | Challenges & Solutions | | 11 |
| 11. | Conclusion | | 11 |
| 12. | Future Enhancements | | 11 |
| 13. | References | | 11 |

**1. Introduction**

Automation testing is a crucial part of modern software development to ensure quality, reliability, and efficiency. Manual testing can be time-consuming and error-prone, especially for repetitive tasks.

This project focuses on automating the functional testing of the nopCommerce e-commerce platform using Selenium WebDriver and Java. nopCommerce is an open-source online shopping platform that allows users to browse products, manage accounts, place orders, and generate reports. “This project focuses on automating the functional testing of the nopCommerce e-commerce platform (<https://demo.nopcommerce.com>) using Selenium WebDriver and Java.”

The automation project aims to:

* Reduce manual effort and testing time through automation.
* Validate essential workflows like user registration, login, product selection, and adding items to the cart.
* Implement a data-driven approach using Excel for multiple test scenarios.
* Apply Page Object Model (POM) design for maintainable and scalable automation.

This report documents the entire automation process, including tools used, test strategy, test cases, results, challenges, and recommendations for future enhancements.

**2. Objective**

* Automate the functional testing of nopCommerce website modules.
* Validate key workflows: user registration, login, product selection, and cart operations.
* Ensure consistent execution and error-free performance across critical modules.
* Implement a data-driven approach for executing multiple input scenarios using Excel.
* Validate shopping cart and checkout process with billing address.

**3. Project Overview**

nopCommerce is an open-source e-commerce platform used globally for online retail. Automation testing improves efficiency, reduces errors, and ensures reliable functionality.

The project emphasizes:

* Automating repetitive tasks such as registration, login, and product management.
* Validating the application’s behaviour under multiple scenarios.
* Using Excel as a source for test inputs and expected outcomes, enabling data-driven testing.

Technologies and Tools Used:

* Selenium WebDriver
* Java
* TestNG
* Apache POI (Excel integration)
* Chrome Driver
* IDE: Eclipse

**4. Project Scope**

**4.1 In Scope:**

* Admin Login & Logout
* Dashboard access and verification of widgets, menus, and notifications
* Customer Management: Add, Edit, Delete, and Search records
* Product Management: Add, Edit, Delete, and verification of product details
* Order Management & Processing: Creation, updates, and status verification
* Reports & Analytics generation and validation
* Shopping cart validation
* Checkout process including billing address

**4.2 Out of Scope:**

* Payment gateway integration
* Performance or stress testing

**5. Tools & Technologies**

* **Automation Tool:** Selenium WebDriver
* **Programming Language:** Java
* **Test Management:** TestNG
* **Data Handling:** Apache POI for Excel integration
* **Browsers:** Google Chrome (latest version)
* **IDE:** Eclipse

**6. Test Strategy**

The test strategy ensures systematic validation of functional workflows through automation:

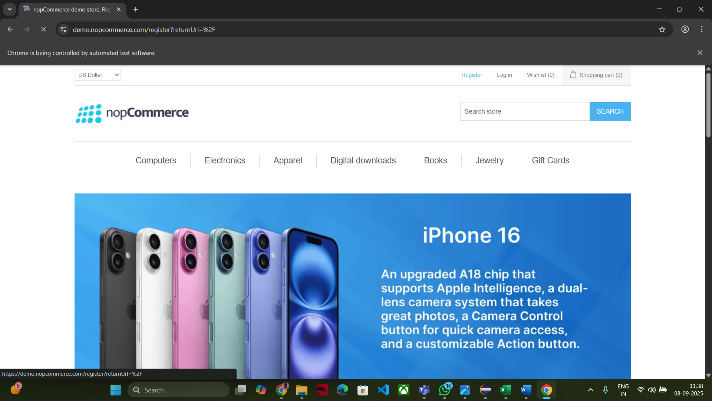
* **Approach:** Automate repetitive workflows; manual testing for exploratory or edge cases.
* **Test Levels:**
  + Smoke Testing: Validate critical modules after each build
  + Functional Testing: Verify individual feature functionality
  + Regression Testing: Re-run automated tests after updates to ensure stability
* **Test Data:** Excel sheets containing valid and invalid inputs for registration, login, product selection, cart actions, and checkout.
* **Test Environment:** Latest stable version of Chrome browser.

**7. Test Scenarios & Test Cases**

**Key Functional Scenarios:**

**7.1 Launch Browser & Navigate to Homepage**

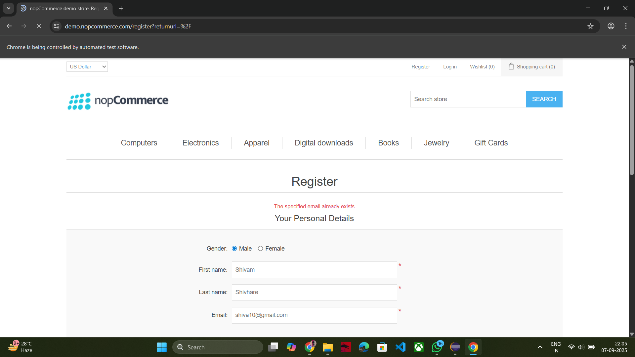
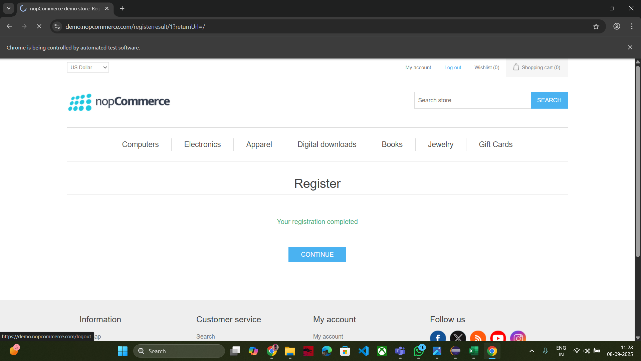
* Verify browser launch and homepage load.
* Validate homepage title.
* Verify visibility of logo, menu



**Image of Launching Browser & Homepage**

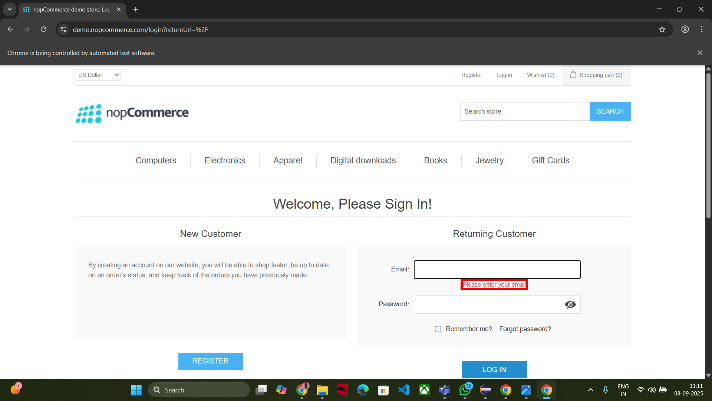
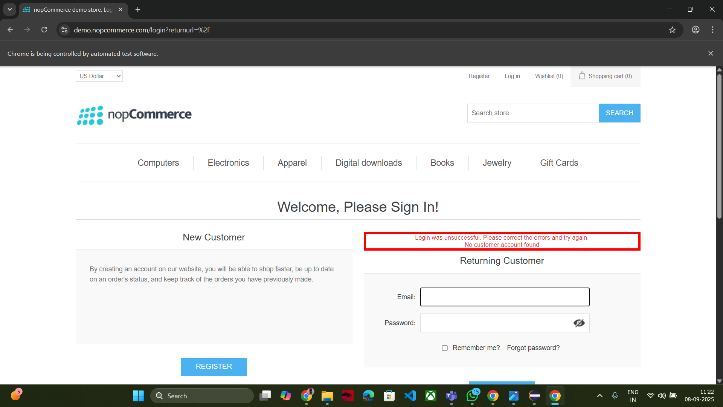
**7.2 User Registration (Valid/Invalid Data)**

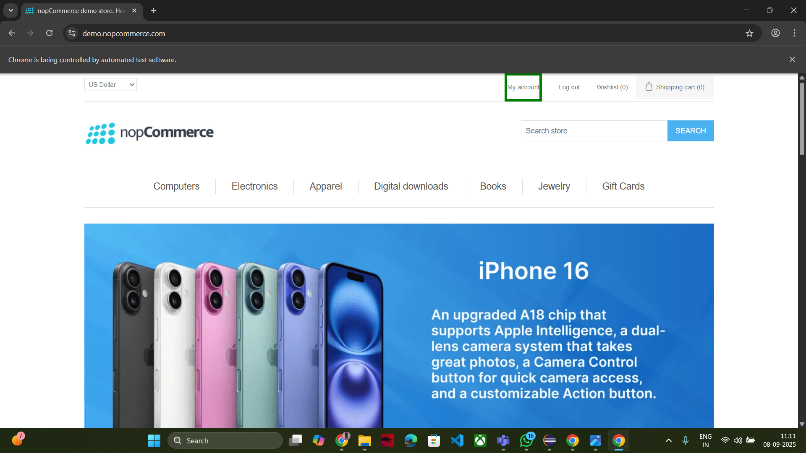
* Complete registration form with valid inputs.
* Submit and verify success message.
* Test invalid or empty inputs to validate error messages.



**Screenshot for Registration Test**

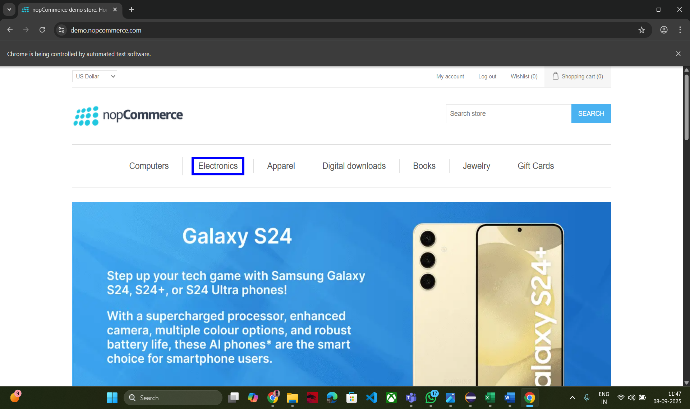
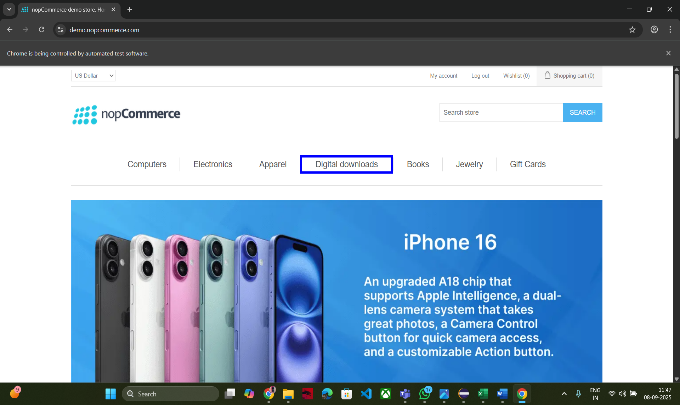
**7.3 User Login (Valid/Invalid & My Account Verification)**

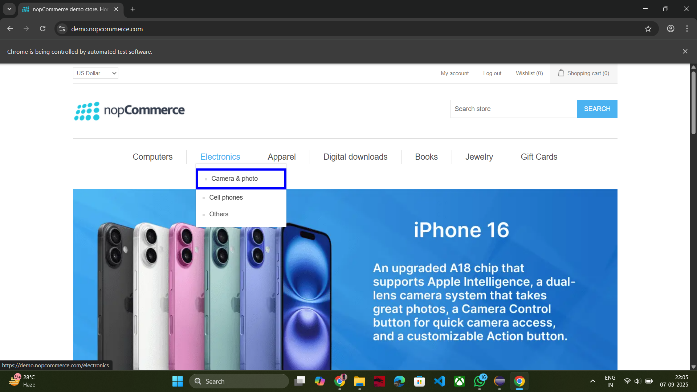
* Login with valid credentials and access dashboard.
* Attempt login with invalid or empty credentials to verify error handling.
* Confirm “My Account” is displayed after successful login.



**Images for Valid/Invalid & My Account Verification**

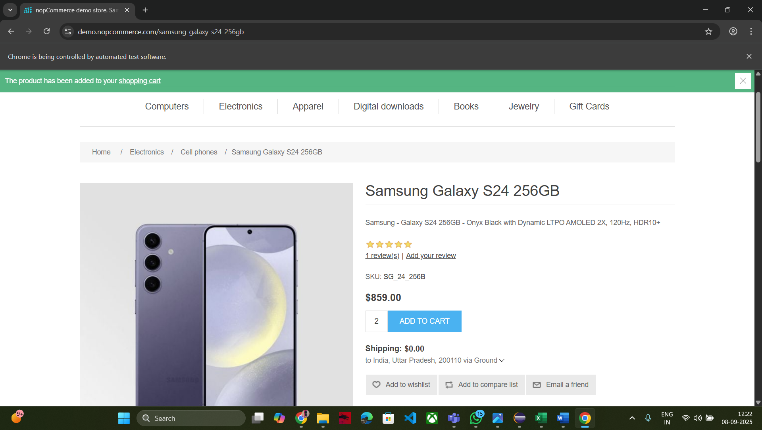
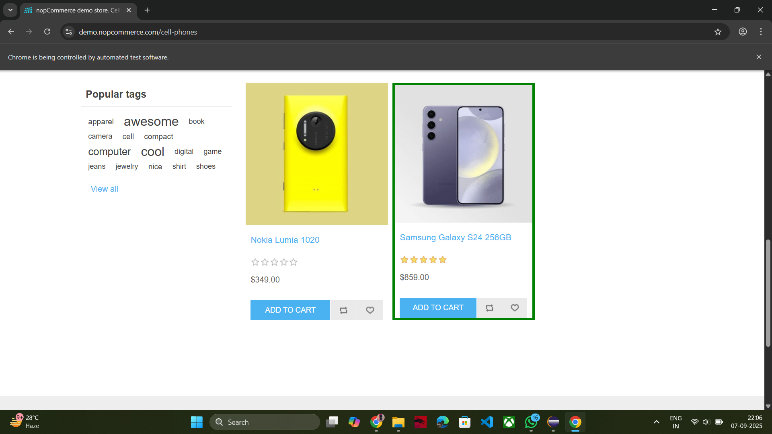
**7.4 Submodule Verification (Electronics, Apparel, Books, etc.)**

* Verify all submodules on the homepage are visible and clickable.
* Navigate to Electronics and other categories to confirm correct page loading.

****

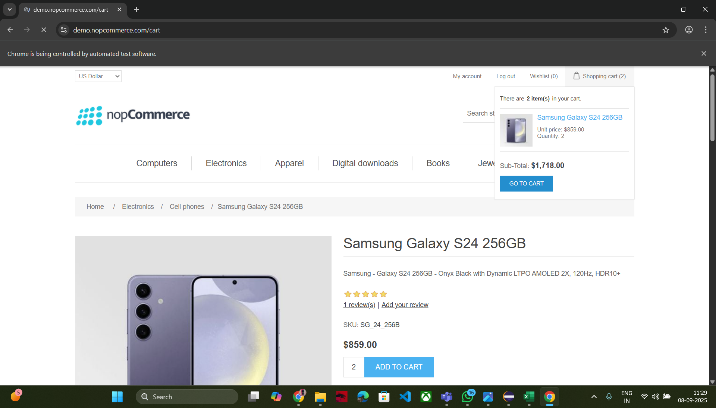
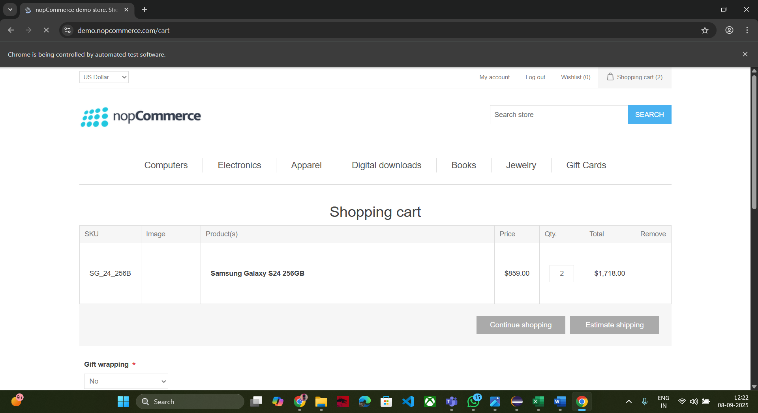
**Some Image of verifying Submodules of nopCommerce & Electronics**

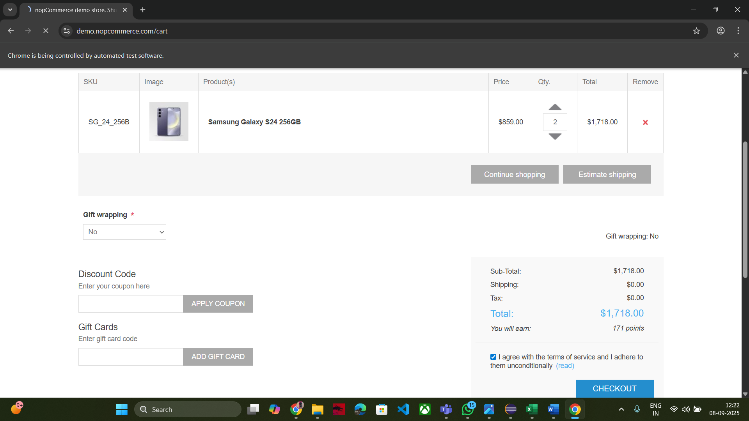
**7.5 Product Selection & Add to Cart**

* Select product (e.g., cell phone) from Electronics.
* Validate product details: image, description, price.
* Set quantity, enter shipping address, and add product to cart.
* ****Confirm cart update and success message.

**Image for Selecting Required Product & Add To Cart Successfully**

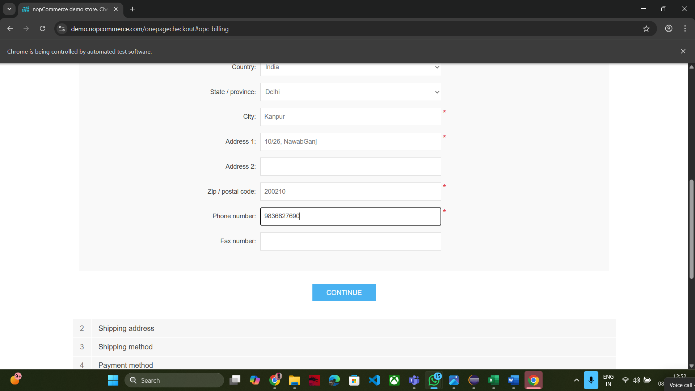
**7.6** **Shopping Cart Validation**

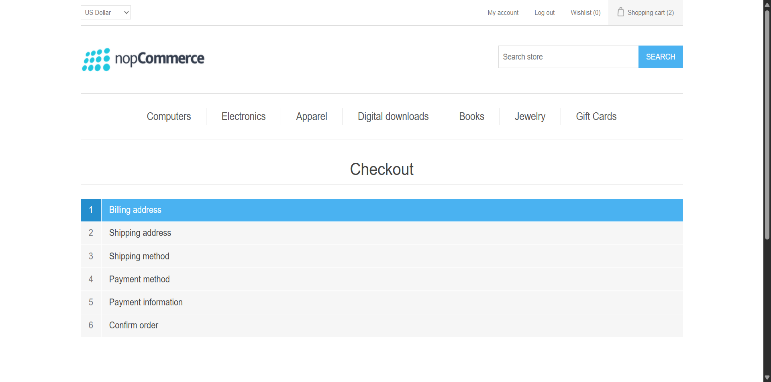
* Hover on shopping cart and validate item details.
* ****Verify product names, quantities, and images.

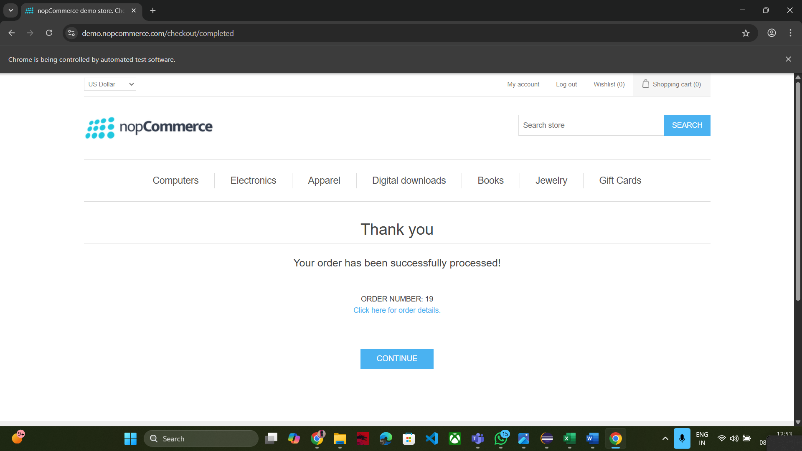
****

**Pictures Of Shopping Cart Validation**

**7.7** **Checkout Process Validation**

* Proceed to checkout with product in cart.
* Enter billing address and validate next step.





**Images Of All Checkout Processes**

**8. Automation Workflow**

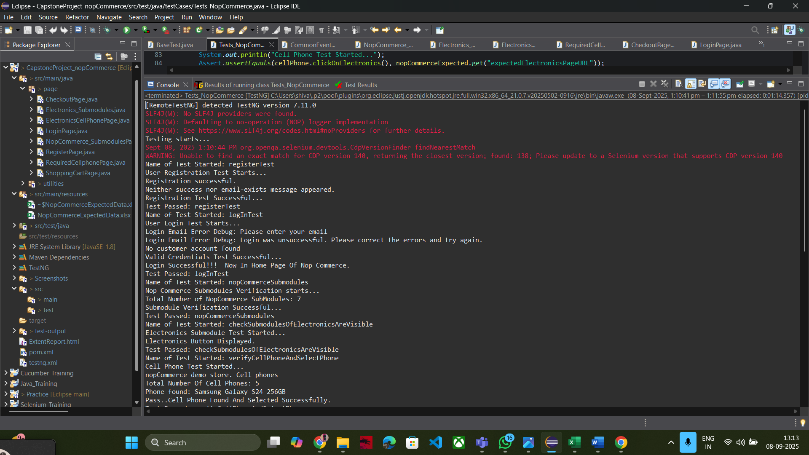
1. Launch Browser → Navigate to Homepage
2. User Registration
3. Login Verification
4. Submodule Verification
5. Navigate to Electronics
6. Click On Cell Phones
7. Select Required Product
8. Add Product to Cart
9. View Shopping Cart
10. Checkout with billing address validation

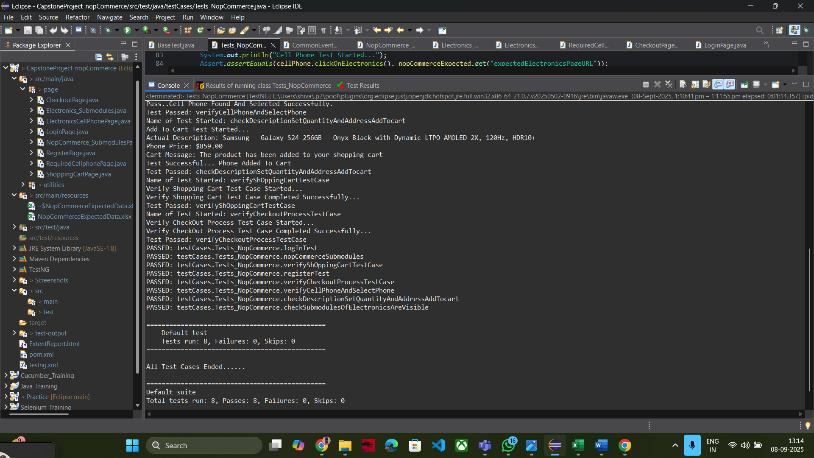
**Implementation Highlights:**

* **Page Object Model (POM):** Ensures modular and maintainable code.
* **Data-Driven Testing:** Excel sheets allow flexible execution with multiple input scenarios.

**9. Test Results & Observations**

* **Total Test Cases Executed:** 17+
* **Pass Rate:** 100% for all critical functional workflows
* **Observations:**
  + Registration and Login modules executed successfully.
  + Submodules, product selection, and Add to Cart workflows verified without issues.
  + Screenshots captured for key steps as evidence.
  + Shopping Cart and Checkout workflows executed successfully, validating end-to-end order process.

****



**Pictures Of All Logs In Console With Test Cases Results**

**10. Challenges & Solutions**

* **Dynamic Web Elements:** Handled using explicit waits.
* **Synchronization Issues:** Solved with WebDriverWait for stable execution.
* **Excel Data Integration:** Multiple sheets handled efficiently using Apache POI.

**11. Conclusion**

* Automation significantly reduced testing time and manual effort.
* All critical workflows of nopCommerce including registration, login, product selection, shopping cart, and checkout were validated successfully.
* Demonstrated effective end-to-end automation using Selenium and Java.
* Provides a strong foundation for expanding automation to additional modules and future enhancements.

**12. Future Enhancements**

* Extend automation to include payment gateway and checkout workflows.
* Implement cross-browser testing for wider coverage.
* Integrate automation suite with CI/CD tools like Jenkins for continuous testing.

**13. References**

* Selenium WebDriver Official Documentation
* TestNG Official Documentation
* nopCommerce Website for Functional Details