**Final QA Project Report**

**Project Overview:**

**Application:** nopCommerce (open-source e-commerce platform)

**Goal:** To validate the end-to-end functionality of nopCommerce across web and mobile environments using manual, automated, and CI/CD-driven testing techniques.

**Testing types covered:**

* Manual Testing
* Web Automation (Cypress and Selenium)
* Mobile Automation (Appium)
* API Testing (FakeStore API)
* CI/CD Integration via Jenkins and GitHub

**QA Strategy Summary (from Preliminary Phase):**

During the preliminary phase, foundational testing was performed to establish a QA baseline for the nopCommerce application.

* Conducted **manual testing** for core user flows — registration, login, and product purchase.
* Automated key web scenarios using **Cypress**, including login, add-to-cart, and checkout, along with accessibility validation.
* Performed **API testing** using the **FakeStore API** due to restricted access to nopCommerce APIs.
* Used **Jira** for defect tracking and **GitHub** for documentation and version control.

**Appium Testing (Mobile Browser Automation):**

Validate key nopCommerce workflows on a mobile environment to ensure responsiveness and usability.

**Setup Details:**

* **Device:** Pixel 5 Emulator
* **Android Version:** 13
* **Browser:** Chrome
* **Tool:** Appium (Node.js + WebDriver)
* **Platform:** Windows/Mac with Android Studio Emulator

**Test Scenarios Executed:** User registration, login, product navigation, and add-to-cart. The application’s responsive design worked well across the tested mobile viewport. Appium scripts successfully interacted with Chrome browser elements using XPath and accessibility IDs.

**Selenium Testing (POM Framework):**

Validate major workflows on the web platform using Selenium and the Page Object Model (POM) design pattern.

**Setup Details:**

* **Language:** Java
* **Framework:** Selenium WebDriver + TestNG
* **Design:** Page Object Model (POM)
* **Tested Application:** nopCommerce (web version)

**Test Scenarios Executed:** User registration, login, product search, add to wishlist, add-to-cart, and checkout. Tests executed successfully on Chrome. Also, Selenium POM provided better code maintenance and scalability for regression testing.

**CI/CD Integration:**

Automate the execution of Selenium tests using Jenkins pipeline triggered from GitHub commits. (POM) design pattern.

**Setup Workflow:**

* Pushed Selenium test code to **GitHub repository**.
* Configured **Jenkins job** linked to the GitHub repository.
* Jenkins pipeline automatically triggered on commits.
* Tests executed successfully and generated txt reports.

**Findings:**

* nopCommerce UI and functional workflows are stable across desktop and mobile.
* Jenkins pipeline enhances testing efficiency and reduces manual intervention.
* Responsive layout works well on mobile browsers (Pixel 5, Android 13).

**Reports and Deliverables:**

* Test Plan and Strategy Summary
* Selenium (POM) Test Scripts
* Appium Test Scripts (Mobile Browser)
* Jenkins CI/CD Configuration
* Final Presentation Video (Live Demo)