**Test Strategy Document**



**Project Name:** Demo nopCommerce

**Base URL:** [*https://demo.nopcommerce.com*](https://demo.nopcommerce.com)**Version:** 1.0 **Date:** 7-August-2025 **Prepared By:** Ankush

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Author | Description |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**TABLE OF CONTENTS**

1 Introduction 3

2 OBJECTIVES 3

3 Test Scope 3

3.1 In Scope 3

3.2 Out Of Scope 3

4 Test types 4

5 Test design approach 4

6 Test Environment 4

7 Testing tools 5

8 Risk mitigation 5

9 Reporting & Metrics 5

10 test data strategy 5

11 Test Schedule 6

12 Terms/Acronyms 6

**1. Introduction**

This document outlines the test strategy for validating the **nopCommerce Demo Website**. The strategy ensures that critical business workflows (customer, shopping cart, checkout, search, etc.) are thoroughly tested, with focus on functionality, usability, integration, and regression stability.

**2. Objectives**

* Validate end-to-end customer workflows (registration, login, product search, checkout).
* Ensure correct functioning of product catalog, shopping cart, and payment flow.
* Verify responsiveness and usability across major devices and browsers.
* Integrate API testing (where available, e.g., product/catalog API).
* Provide scalable automation to support continuous testing.
* Minimize defect leakage by robust regression coverage.

**3. Test Scope**

**3.1 In Scope**

* **Customer Workflows**: Registration, login, logout, profile update.
* **Product Management (Frontend)**: Browse, filter, search, sort products.
* **Shopping Cart & Wishlist**: Add/remove/update items, quantity changes.
* **Checkout Process**: Guest checkout, registered user checkout, multiple payment/shipping methods.
* **Order Tracking**: Order history, status updates.
* **Promotions & Discounts**: Coupons, gift cards, reward points.
* **UI/UX**: Responsiveness, layout consistency, accessibility checks.
* **Cross-browser Validation**: Chrome, Firefox, Edge.

**3.2 Out of Scope**

* Admin panel functionality (covered separately).
* Third-party payment gateway backend validation.
* Performance, load, and stress testing (handled by a separate strategy).
* Compatibility testing on legacy browsers.

**4. Test Types**

|  |  |
| --- | --- |
| Test Type | Purpose |
| **Smoke** | Ensure critical workflows (login, product search, checkout) work after build. |
| **Functional** | Validate all business logic and workflows (customer, cart, orders). |
| **UI/UX** | Validate layout, mobile responsiveness, accessibility. |
| **Regression** | Ensure stability after new builds/releases. |
| **API** | Validate catalog/product API responses (if enabled). |
| **Integration** | Validate cross-module consistency (cart ↔ checkout ↔ order). |

**5. Test Design Approach**

* Use **Page Object Model (POM)** for structured automation framework.
* Implement reusable functions for dropdowns, calendars, and dynamic elements.
* Data-driven testing using **TestNG Excel**.
* JavaScript executor for hidden/dynamic UI elements.
* Negative and boundary value tests for robustness.

**6. Test Environment**

|  |  |
| --- | --- |
| Component | Details |
| Browsers | Chrome |
| OS | Windows 11, macOS Ventura |
| Test Data | Guest users, registered accounts, discount coupons, sample orders |
| Tools | Selenium, TestNG, ExtentReports |
| CI/CD | Jenkins or GitHub Actions (optional) |

**7. Testing Tools**

|  |  |
| --- | --- |
| Tool | Purpose |
| Selenium + TestNG | Functional/UI automation |
| Jira | Defect tracking |
| ExtentReports | Automation reporting |
| Eclipse | IDE for scripting |

**8. Risk Mitigation**

|  |  |
| --- | --- |
| Risk | Mitigation |
| Frequent UI changes | Robust locators, fallback to XPath/JS execution |
| Test data unavailability | Dynamic test data generation, mock data usage |
| Demo site downtime | Local mock environment / retries |
| Parallel execution issues | Use ThreadLocal driver, retry logic |

**9. Reporting & Metrics**

* **Daily execution reports** (Smoke + Functional).
* **Daily defect summary** with severity analysis.
* **Final test summary** after each sprint.
* **Metrics Tracked**:
  + Test case pass % ≥ 95%
  + Regression stability ≥ 90%
  + Defect leakage ≤ 5%
  + UI coverage ≥ 90%

**10. Test Data Strategy**

* Static data: Negative cases, invalid inputs (wrong email, invalid coupons).
* Dynamic data: Unique user registrations, random product selections.
* Role-based data: Guest vs Registered user checkout.

**11. Test Schedule**

|  |  |  |  |
| --- | --- | --- | --- |
| Sprint | Area of Focus | Start Date | End Date |
| Sprint 1 | Requirement Analysis, Test Strategy, Manual Test Case Design | 05-09-2025 | 06-09-2025 |
| Sprint 2 | Automation Framework Setup | 06-09-2025 | 07-09-2025 |
| Sprint 3 | Functional & Regression Execution, Reporting & Closure | 07-09-2025 | 08-09-2025 |

# **12. Terms/Acronyms**

| TERM/ACRONYM | DEFINITION |
| --- | --- |
| API | Application Program Interface |
| QA | Quality Assurance |
| DOM | Document Object Model |
| JIRA | A tool for issue and bug tracking |
| OS | Operating System |
| CI | Continuous Integration |
| UI | User Interface |
| UX | User Experience |
| IDE | Integrated Development Environment |
| Selenium | Tool for automating web browser testing |
| Jenkins | CI/CD automation server |
| TestNG | Frameworks for Java test automation |
| POM | Page Object Model |