Test strategy for Demo OpenCart web application

AUTHOR(S) :	Inga Niścior
STATUS:	Final
DOCUMENT DATE:	02.07.2025
NUMBER OF PAGES:	9
OWNER:	Inga Niścior

List of changes

Version	Date	Description	Author(s)
1.0	02.07.2025	Initial Draft	Inga Niścior

Table of content

1. Introduction 4
1.1 Purpose 4
1.2 Objectives 4
1.3 Assumptions 4
2. Scope 5
2.1 In scope 5
2.2 Out of scope 5
2.3 Constraints 5
3. Test Approach 6
3.1 Test Phases 6
3.2 Comprehensive Testing Strategies 6
4. Test Process 8
4.1 Defect Management 8
4.2 Test Environment 8
4.3 Test types 8
5. Risks & Mitigation 9

1. Introduction

1.1 Purpose

The purpose of this document is to define a complete test strategy for the OpenCart demo site. It outlines the testing approach, scope, techniques, tools, entry/exit criteria, and defect management. This strategy ensures functional quality, usability, and compliance of the application's front-end interface.

1.2 Objectives

- **Validate core business functionalities***:* ensure critical features such as user registration, login, product browsing, cart management, and checkout operate as intended.
- **Identify and report defects***:* detect any bugs, inconsistencies, or deviations in functionality, user interface, or system behavior.
- **Ensure usability and accessibility compliance***:* verify that the application meets established usability standards and accessibility guidelines (e.g., WCAG 2.1).
- **Provide stakeholder confidence***:* deliver comprehensive test results and metrics that demonstrate the system's stability, performance, and readiness for release.

1.3 Assumptions

- The Demo OpenCart site is publicly accessible and stable during the entire testing window
- No backend database or administrative modifications will be possible, as the site is read-only with respect to persistent data
- The demo environment reflects a simplified, production-like e-commerce workflow
- Testing will be primarily manual, with limited automation due to the demo environment constraints

2. Scope

2.1 In scope

ID	In scope
I01	user registration/login
I02	product browsing
I03	cart operations
I04	simulated checkout
I05	UI validation
I06	browser responsiveness

2.2 Out of scope

ID	Out of scope
O01	admin panel
O02	real payments
O03	security
O04	performance/load testing
O05	database testing

2.3 Constraints

ID	Constraints
C01	read-only demo environment
C02	no backend access
C03	limited automation possibilities

3. Test Approach

3.1 Test Phases

Test phase	Description
test planing	define objectives, scope, resources, schedule, and environment setup
test design	develop detailed manual test cases and scenarios based on requirements and user stories
test execution	conduct manual testing on targeted features, document outcomes, and capture evidence
defect management	log, prioritize, and track defects using a centralized bug tracking system.
test closure	analyze test results, prepare summary reports, and document lessons learned

3.2 Comprehensive Testing Strategies

Levels of testing**:**

- **System Testing (Manual GUI)**: This level focuses on validating the complete and integrated software product to ensure that it meets the specified requirements. Manual testing of the graphical user interface (GUI) is conducted to assess usability, functionality, and overall user experience.

- **Exploratory Testing**: This approach allows testers to explore the application without predefined test cases. Testers leverage their intuition and experience to identify defects and gain insights into the software's behavior.

- **Smoke Testing**: Often referred to as a "sanity check," smoke testing involves a preliminary assessment to determine whether the most critical functions of the application are working properly. This step is essential before proceeding with more in-depth testing.

Before commencing testing, it is crucial to establish the entry criteria:

- the demo environment must be accessible and stable.
- the feature scope of the application being tested should be clearly defined.
- exit criteria

To determine when testing is complete, the following exit criteria must be met:

- all test cases have been executed.
- critical defects identified during testing have been addressed and closed.
- a comprehensive test report has been completed to summarize findings and outcomes.

4. Test Process

4.1 Defect Management

- all defects will be documented in the designated bug-reports/ folder within the project repository
- tool: JIRA (simulated)
- each defect report will include:
    - **title**: concise summary of the issue.
    - **description**: detailed explanation of the problem.
    - **steps to reproduce:** clear, step-by-step instructions to replicate the defect.
    - **expected vs actual results**: description of the anticipated outcome versus what actually occurred.
    - **screenshots/attachments**: visual evidence where applicable.
    - **severity and priority**: classification based on impact and urgency.
    - **status tracking**: lifecycle states such as *Open*, *In Progress*, *Retested*, and *Closed*.

4.2 Test Environment

- Testing will be conducted on publicly accessible demo environment: https://demo.opencart.com
- Supported browsers for testing include:
    - Google Chrome (latest stable release)
    - Mozilla Firefox (latest stable release)
    - Microsoft Edge (latest stable release)
- Devices: Desktop (Windows/macOS), and smartphones for responsive testing.

*no backend or database access

4.3 Test types

Test type	Scope
functional	forms, cart, product search
usability	navigation, clarity, feedback
compatibility	Chrome, Firefox, Edge
responsiveness	devices and screen sizes
regression	checklist-based re-verification

5. Risks & Mitigation

Risk	Impact	Probability	Mitigation
read-only demo prevents data validation	High	Certain	focus on UI
no real payments possibility	Medium	Certain	simulate payments steps up to confirmation
no access to code	Medium	High	highlight in report as coverage gap
no automation possible	Medium	High	increase exploratory testing session