Test Plan

For

Opencart

|  |  |  |  |
| --- | --- | --- | --- |
| **Prepared By:** | **Yash Vardhan Singh** | **Reviewed by:** |  |
| **Version No:** | **V1.0** | **Issue Date:** |  |

## Document Revision History\*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Document Version | Submitted By | Date | Reviewed & Approved BY | Notes |
| V1.0 | Yash Vardhan Singh | 10/05/2025 |  |  |
| V1.1 |  |  |  |  |
|  |  |  |  |  |

**Table of Content**

1. ATTENDING AUDIENCE................................................................................................................... 6

2. INTRODUCTION................................................................................................................................. 7

2.1 PURPOSE........................................................................................................................................... 7

2.2 ABOUT THIS DOCUMENT...............................................................................................................7

2.3 RELATED DOCUMENT....................................................................................................................7

3. SCOPE................................................................................................................................................. 8

3.1 APPLICATION SCOPE...................................................................................................................... 8

3.2 TEST SCOPE...................................................................................................................................... 8

3.3 EXCLUSIONS TO THE TEST SCOPE.............................................................................................8

4. APPROACH.......................................................................................................................................... 9

4.1 DEVELOPMENT & TESTING METHODOLOGY.......................................................................... 9

4.2 TRADITIONAL TESTING................................................................................................................ 9

4.3 FUNCTIONAL TESTING.................................................................................................................. 9

4.4 INTEGRATION TESTING................................................................................................................ 9

4.5 LOOK AND FEEL TESTING OF WEBPAGES................................................................................ 9

4.6 BROWSER COMPATIBILITY.......................................................................................................... 9

4.7 APPLICATION TESTING ON MOBILE.......................................................................................... 9

4.8 USER ACCEPTANCE TESTING...................................................................................................... 9

5. TEST ENVIRONMENT..................................................................................................................... 13

5.1 QA / STAGING ENVIRONMENT................................................................................................... 13

5.2 UAT ENVIRONMENT..................................................................................................................... 13

6. DEFECT TRACKING MECHANISM............................................................................................... 14

7. RISK & MITIGATION....................................................................................................................... 16

8. TEST DELIVERABLES..................................................................................................................... 16

9. ASSUMPTIONS................................................................................................................................. 17

### 1. ATTENDING AUDIENCE

The following individuals and roles are the primary audience for this Test Plan. Each plays a critical part in ensuring the quality and success of the testing activities:

|  |  |  |
| --- | --- | --- |
| **Name** | **Designation / Role** | **Responsibilities / Contributions** |
| Yash Vardhan Singh | Test Engineer | Prepares Test Plan, Test Cases, and Defect Reports. Leads test activities and delivers test outputs. |
| Qusai cutlery | Software Developer | Provides 20% allocation to assist the test team during system testing and defect resolution. |
| Padmini Singh | Business Analyst | Offers 30% allocation for business clarification and supports alignment with business requirements. |
| Vraj Soni | UI/UX Developer | Supports GUI testing by validating layout, design consistency, and usability. |
| Vansh Khasia | Frontend Developer | Assists in UI-related defect resolution and integration testing of frontend components. |
| Jaykishor Kushwaha | Backend Developer | Aids in fixing backend issues and supports API validation and integration testing. |
| Pallavi Singh | Product Owner | Reviews test results, sets defect priorities, and ensures alignment with product requirements. |
| Ashish Jani | QA Lead | Oversees QA processes, ensures test quality, and coordinates communication among all stakeholders. |

# 2. INTRODUCTION 2.1 Purpose

# The purpose of this test plan is to define the scope, approach, environment and deliverables of the testing efforts for the Opencart. This document aims to provide a clear plan for the testing activities and to ensure that the software meets the specified requirements and quality standards. 2.2 About this document.

# This document describes the methods and procedures that will be used by the QA team in the testing process of the Opencart. It is meant to be used as a manual during testing works. It describes the procedure of the testing process. The test plan is intended for project managers, product developers, and QA engineers. 2.3 Related Documents

# The following documents support and supplement this test plan:

# User Stories

# Test Cases

# Defect Report

# 3. SCOPE 3.1 APPLICATION SCOPE

# OpenCart is a free and open-source e-commerce platform for creating and managing online stores. It allows businesses to build and manage their online presence, including managing products, inventory, shipping, sales, and more. OpenCart is known for its user-friendly interface, extensive customization options, and a large community of developers and users.

# 3.2 TEST SCOPE

The following modules or features are covered in this testing cycle:

* Registration
* Login
* Logout
* Forgot Password
* Search
* Add to Cart
* Wishlist
* Checkout

**3.3 EXCLUSIONS TO THE TEST SCOPE**

Any modules, sub-modules, or features not explicitly listed in Section 2.2 are excluded from the current test cycle.

Third-party applications or integrations are not included in the scope of this testing cycle and will not be tested.

**4. APPROACH**

**4.1 Development and Testing Methodology**

The project follows the **Agile development methodology**, which promotes iterative development, continuous feedback, and collaboration between cross-functional teams.

Under this methodology:

* Development is divided into short, time-boxed iterations (sprints).
* Requirements and solutions evolve through the collaboration of developers, testers, product owners, and business analysts.
* Testing is integrated throughout the development cycle, with continuous verification of features as they are developed.
* Frequent builds and deployments enable early defect detection and quicker response to changes.

**QA Activities in Agile:**

* Test planning and test case design are done incrementally with each sprint.
* Functional and integration testing is performed continuously.
* Defects are logged and tracked in real time for prompt resolution.
* Regression testing is executed in every sprint to ensure existing functionality is not broken.

This approach ensures faster delivery of high-quality software that aligns with evolving business needs.

**4.2 TRADITIONAL TESTING**

Traditional testing will be carried out to validate the core functionalities against the defined requirements.  
Test cases will be prepared using MS Excel and executed manually without the use of automation tools. All identified defects will be recorded in the Defect Report. Once a defect is fixed, the impacted areas will be retested to ensure the issue is resolved and no issues have been introduced.  
This approach includes:

* Test case creation and execution
* Defect identification and logging
* Impact analysis
* Retesting

**4.3 FUNCTIONAL TESTING**

Functional testing will ensure that all features work as per the defined requirements. This includes validation of user inputs, data processing, and expected outputs across all modules. Both positive, negative, and edge cases will be tested.

**Types of Functional Testing to be performed include:**

* Smoke Testing – to verify critical functionalities after each build.
* Sanity Testing – to validate minor fixes or changes.
* Regression Testing – to ensure recent changes do not affect existing features.
* Integration Testing – to confirm data flow between modules.
* System Testing – to check complete end-to-end flows.
* User Acceptance Testing (UAT) – to validate functionality from an end-user perspective.

**Techniques used:**

* Equivalence Partitioning and Boundary Value Analysis (BVA) for efficient input validation,
* Decision Table and State Transition Testing for logical and state-based coverage,

**Entry Criteria**   
  
● The application construction is completed.  
● The test engineers are dedicated.   
● Necessary devices, instruments, and other equipment are acquired.   
● Test environment is prepared, and the application is released to the test environment.  
  
**Exit Criteria**  
  
● All the planned tests are performed.

● There are no show-stopping errors.

● All the errors of high priority and severity are fixed.   
● The test results are evaluated, discussed, and approved  
  
**4.4 INTEGRATION TESTING**

Integration testing will focus on verifying that modules interact seamlessly and that data flows accurately between components such as:

* Order Management
* Customer Master
* Payment Processing

The Big Bang integration strategy will be followed, where all modules are integrated at once and tested as a whole.

**4.5 LOOK AND FEEL TESTING OF WEB PAGES (GUI ASPECTS)**

GUI testing will validate that:

* Layout, font, alignment, and button behavior are consistent with design guidelines
* The application is intuitive and easy to navigate
* The interface behaves consistently across screen resolutions

This testing will also verify adherence to usability standards.  
  
**4.6 BROWSER COMPATIBILITY TESTING**  
  
Browser compatibility testing ensures that the web application functions and displays consistently across different browsers and their versions. It helps identify and resolve browser-specific issues to deliver a seamless and uniform user experience.

Testing will primarily be conducted on **Google Chrome Version 135.0.7049.116 (Official Build) (64-bit)**, which is assumed to represent 100% compatibility.  
  
**4.7 APPLICATION TESTING ON MOBILE DEVICES**  
  
Key user journeys will be tested on Android mobile browsers to verify:

* Layout
* Responsiveness
* Usability

Testing on iOS devices is out of scope for this phase.

**4.8 USER ACCEPTANCE TESTING**

User Acceptance Testing (UAT) will be conducted by the client or designated end-users to validate whether the application meets business requirements.  
 The QA team will:

* Prepare UAT scenarios
* Support execution

Assist in documenting and resolving issues raised during UAT

**5.TEST ENVOIRNMENT**  
  
This section outlines the environments used during testing to ensure application stability and performance before deployment.

**5.1 QA/ STAGING ENVIORNMENT**   
  
The QA/Staging environment will be used to execute all functional, integration, GUI, and security test cases. It will closely mimic the production setup and support the following configurations:

Browsers: Google Chrome Version 135.0.7049.116 (Official Build) (64-bit)

Operating Systems: Windows 11

**5.2 UAT ENVIORNMENT**   
  
The UAT environment is a near-production setup accessed by stakeholders for validation of business workflows. It includes the final builds and will support Windows 11 and major browser (Chrome Browser) for realistic testing conditions.

**6. DEFECT TRACKING MECHANISM**

All defects identified during testing will be documented and tracked using an Excel-based Defect Report, with each defect assigned a unique identifier (Defect ID).

Defects will follow a standard Bug/Defect life cycle.  
**Defect Life Cycle Flow:**

1. **New** – Defect is logged and submitted for review.
2. **Assigned** – Assigned to a developer for investigation.
3. **Open** – Developer acknowledges and starts working on the defect.
4. **Fixed** – Developer has resolved the issue.
5. **Retest** – Tester re-executes the relevant test cases.
6. **Reopen** – If the defect persists, it's reopened and reassigned.
7. **Verified** – If the defect is resolved, it is verified by the tester.
8. **Closed** – Defect is successfully resolved and closed.
9. **Deferred** – Fix postponed for a future release.
10. **Rejected** – Not considered a valid defect (e.g., invalid scenario or duplicate).

**Communication**

Defects will be communicated to stakeholders via **email updates**, using a clear and concise bullet format that includes:

* Defect ID
* Summary
* Severity
* Status

#### Suspension and Resumption Criteria

* **Suspension Criteria:** Testing will be paused if a **blocker defect** is identified that prevents further execution. Such defects will be reported immediately.
* **Resumption Criteria:** Testing will resume once the blocker is resolved and verified through successful retesting.

#### Severity and Priority

* **Severity** (assigned by testers based on technical impact):
  + **Critical** – Complete failure or data loss.
  + **High** – Major feature failure; system still usable.
  + **Medium** – Functional deviation; not blocking.
  + **Low** – Minor issue or cosmetic defect.
* **Priority** (assigned in collaboration with stakeholders based on business needs):
  + **Critical** – Immediate fix required; blocks release.
  + **High** – Needs fix in current release.
  + **Medium** – Fix in subsequent release.
  + **Low** – May be deferred to future versions.

Defect IDs will be mapped to corresponding test cases to maintain traceability. This linkage supports defect tracking, failure analysis, and targeted regression testing.

### 7. RISK AND MITIGATION

This section identifies potential risks that may impact the testing process and outlines mitigation strategies to manage or minimize their effects.

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk Description** | **Impact** | **Likelihood** | **Mitigation Strategy** |
| Incomplete or unclear requirements | High | Medium | Conduct regular review meetings with stakeholders to clarify and update requirements. |
| Delay in build delivery from the development team | High | High | Define buffer time in the test schedule; establish clear deadlines for build releases. |
| Limited resource availability during peak testing phases | Medium | Medium | Plan resource allocation in advance; ensure backup resources are identified. |
| High number of critical defects identified late in the cycle | High | Medium | Conduct early smoke and integration testing; prioritize critical functionalities. |
| Inadequate communication among team members and stakeholders | Medium | Medium | Schedule regular stand-ups and status updates; use centralized communication tools. |
| Change in project scope or requirements mid-cycle | High | Medium | Implement a change control process; reassess test plans and priorities accordingly. |

**8. TEST DELIVERABLES**

The following test artifacts will be delivered as part of the testing process:

* Test Plan.
* Test Cases
* Test Data
* Traceability Matrix
* Defect Reports

**9. ASSUMPTIONS**

* It is assumed that a testable and stable build will be made available for QA in the staging environment.
* All user stories shared are final, reviewed by stakeholders, and reflect the required business logic.
* It is assumed that acceptance criteria will be logically inferred from user stories.
* Access to environments, test credentials, and Permission to create necessary test data will be Available.
* Only the specified features/modules in the provided user stories are in scope unless stated otherwise.
* Testing on other mentioned browsers or versions (e.g., Firefox, Edge, Safari) will not be performed unless explicitly specified or requested.
* Testing will be performed only on Android mobile browsers; iOS testing is out of scope.
* Only basic security checks will be conducted; advanced penetration testing is excluded.
* No bug tracking tools (e.g., Bugzilla, JIRA) will be used; all defects will be tracked manually using an Excel-based Defect Report and communicated via email.
* Load and stress testing are out of scope for the current testing cycle; only basic performance testing, such as measuring page load times and responsiveness under normal usage will be carried out.