**OPENCART TEST PLAN**

A blue text on a black background

AI-generated content may be incorrect.

**Prepared by: Raju A**

**Date: 17-9- 2025**

Table of Contents

[Overview 2](#_Toc209011188)

[Scope 2](#_Toc209011189)

[Inclusions 2](#_Toc209011190)

[Test Environments 3](#_Toc209011191)

[Exclusions 3](#_Toc209011192)

[Test Strategy 3](#_Toc209011193)

[Defect Reporting Procedure: 5](#_Toc209011194)

[Roles/Responsibilities 5](#_Toc209011195)

[Test Schedule 6](#_Toc209011196)

[Test Deliverables 6](#_Toc209011197)

[Pricing 7](#_Toc209011198)

[Entry and Exit Criteria 7](#_Toc209011199)

[Suspension and Resumption Criteria 8](#_Toc209011200)

[Tools 9](#_Toc209011201)

[Risks and Mitigations 9](#_Toc209011202)

[Approvals 9](#_Toc209011203)

# Overview

As part of this project, testing will be performed on the OpenCart demo web application (<https://demo.opencart.com/>) to validate key functionalities.

This Document serves as high level test planning document with details on the scope of the project, test strategy, test schedule and resource requirements, test deliverables and schedule.

# Scope

The scope of the project includes testing the following features of “<https://demo.opencart.com/>” web application.

## Inclusions

* Register
* Login and Logout
* Forgot Password
* Search
* Product Compare
* Product Display page
* Add to Cart
* Wish List
* Shopping Cart
* Currencies
* Home Page
* Checkout Page
* My Account Page
* Order History Page
* Downloads Page
* Contact Us Page
* Menu Options
* Footer Options
* Category Pages

From our understanding, we believe above functional areas need to be tested

## Test Environments

* Windows 10 - Chrome, Firefox and Edge
* Mac OS – Safari Browser
* Android Mobile OS – Chrome
* iPhone Mobile OS – Safari

## Exclusions

* All the features except that are mentioned under ‘Inclusions’
* Any third-party features or Payment gateways
* Test Automation

# Test Strategy

‘Raju’ has communicated with ‘OpenCart’ and has understood that we need to perform Functional Testing all the functionalities mentioned in the above Scope section.

As part of Functional Test, we will follow the below approach for Testing:

Step-1 – Creation of Test Scenarios and Test Cases for the Different Features in Scope.

* We will apply several Test Designing techniques while creating Test Cases
* Equivalence Class Partition
* Boundary Value Analysis
* Decision Table Testing
* State Transition Testing
* Use Case Testing
* We also use our expertise in creating Test Cases by applying the below:
* Error Guessing
* Exploratory Testing
* We prioritise the Test Cases

Step-2 – Our Testing process, when we get an Application for Testing:

* Firstly, we will perform Smoke Testing to check whether the different and important functionalities of the application are working.
* We reject the build, if the Smoke Testing fails and will wait for the stable build before performing in depth testing of the application functionalities.
* Once we receive a stable build, which passes Smoke Testing, we perform in depth testing using the Test Cases created.
* Multiple Test Resources will be testing the same Application on Multiple Supported Environment simultaneously.
* We then report the bugs in bug tracking tools and send Development Management the defect found on that day in a status end of the day email.
* As part of the Testing, we will perform the below types of Testing:
* Smoke Testing and Sanity Testing
* Regression Testing and Retesting
* Usability Testing, Functionality and UI Testing
* We repeat Test Cycles until we get the quality product.

Step-3- We will follow the below best Practices to make our Testing better:

* Context Driven Testing – We will be performing Testing as per the context of the given application.
* Shift Left Testing – We will start testing from the beginning stages of the development itself, instead of waiting for the stable build.
* Exploratory Testing – Using our expertise we will perform Exploratory Testing, apart from the normal execution of the Test Cases.
* End to End Flow Testing – We will test the end-to-end scenario which involve multiple functionalities to simulate the end user flows.

# Defect Reporting Procedure:

During the test execution

* Any deviation from expected behaviour by the application will be noted. If it can’t be reported as a defect, it would be reported as an observation/issue or posed as a question.
* Any usability issues will also be reported.
* After discovery of a defect, it will be retested to verify reproducibility of the defect. Screenshots with steps to reproduce are documented.
* Every day, at the end of the test execution, defects encountered will be sent along with the observations.

Note:

* Defects will be documented in an Excel file.
* Test scenarios and Test Cases will be documented in an excel documents.

# Roles/Responsibilities

|  |  |  |
| --- | --- | --- |
| Name | Role | Responsibilities |
| Person A | Test Manager | * Escalations |
| Person B | Test Lead | * Create the Test Plan and get the client signoffs * Interact with the application, create and execute the test cases * Report defects * Coordinate the test execution. Verify validity of the defects being reported. * Submit daily issue updates and summary defects to the client. * Attend any meeting with client. |
| Person C | Senior Test Engineer | * Interact with the application. * Create and Execute the Test Cases. * Report Defects. |
| Person D | Test Engineer | * Interact with the application. * Execute the Test Cases. * Report Defects. |

# Test Schedule

Following is the test schedule planned for the project –

|  |  |
| --- | --- |
| Task | Time Duration |
| * Creating Test Plan | 17-09-2025 to 17-09-2025 |
| * Test Case Creation |  |
| * Test Case Execution |  |
| * Summary Reports Submission |  |

# Test Deliverables

The Following are to be delivered to the client:

|  |  |  |
| --- | --- | --- |
| Deliverables | Description | Target Completion Date |
| Test Plan | Details on the scope of the project, test strategy, test schedule, resource requirements, test deliverables and schedule. | Date |
| Functional Test Cases | Test Cases created for the scope defined | Date |
| Defect Reports | Detailed description of the defects identified along with screenshots and steps to reproduce on a daily basis. | NA |
| Summary Reports | Summary Reports –  Bugs by Bug#,  Bugs by Functional Area and  Bugs by Priority | Date |

# Pricing

Not Applicable

# Entry and Exit Criteria

Below are the entry and exit criteria for every phase of Software Testing Life Cycle:

Requirement Analysis

Entry Criteria:

* Once the testing team receives the Requirement Documents or details about the project

Exit Criteria:

* List of Requirements are explored and understood by the Testing team
* Doubts are cleared

Test Planning

Entry Criteria:

* Testable Requirements derived from the given Requirements Documents or Project details
* Doubts are cleared

Exit Criteria:

* Test Plan Document (including Test Strategy) is signed off by the client

Test Designing

Entry Criteria:

* Test Plan Document is signed off by the client

Exit Criteria:

* Test Scenarios and Test Cases Documents are signed off by the client.

Test Execution:

Entry Criteria:

* Test Scenarios and Test Cases Documents are signed-off by the Client
* Application is ready for Testing

Exit Criteria:

* Test Case Reports, Defect Reports are ready

Test Closure

Entry Criteria:

* Test Case Reports, Defect Reports are ready

Exit Criteria:

* Test Summary Reports

# Suspension and Resumption Criteria

Based on the Client Decision, we will suspend and resume the Project.

We will ramp up and ramp down the resources as per Client needs.

# Tools

The following are the list of Tools we will be using in this Project:

* Jira Bug Tracking tools
* Snipping Screenshot Tool
* Word And Excel Documents

# Risks and Mitigations

The following are the list of possible risks and the ways to mitigate them:

Risk: Non-Availability of a Resource

Mitigation: Backup Resource Planning

Risk: Build URL is not working

Mitigation: Resources will work on other tasks

Risk: Less time for Testing

Mitigation: Ramp up the resources based on the Client needs dynamically

# Approvals

Team will send different types of documents for Client Approval as below.

* Test Plan
* Test Scenarios
* Test Cases
* Reports

Testing will continue to the next steps once these approvals are done.