# TEST PLAN

## Opencart

Prepared by: Karthik.S

# Table of Contents

Overview ..... 2
Scope ..... 2
Inclusions ..... 2
Test Environments ..... 2
Exclusions ..... 2
Test Strategy ..... 3
Defect Reporting Procedure: ..... 4
Entry and Exit Criteria ..... 4

## Overview

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 & 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

Test environment is the setup in which testing is performed. It includes hardware, software, and network configurations required for testing. In the case of an e-commerce project, the test environment can be divided into the following:

For example, in the case of an e-commerce project, the test environment should have the following:

- Hardware: Computers, mobile devices, and any other required peripherals with the appropriate configurations.
- Software: Web browsers (Chrome, Firefox, Safari), operating systems (Windows, Mac, Linux), and any other software required for the project.
- Network: Internet connectivity, firewall settings, and network bandwidth. The network should also have the required security settings, such as SSL certificates, to ensure secure transactions during testing.
- Data: The test environment should have the necessary data required for testing, such as user data, product data, and order data. The data should be prepared and configured according to the test scenarios.

## Exclusions

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

## Test Strategy

As part of Functional Testing, 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 Environments simultaneously.
- We then report the bugs in bug tracking tool and send dev. 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:
- Regression Testing and Retesting

    - Usability Testing, Functionality & 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'd 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 a excel.
- Test scenarios and Test cases will be documented in an excel document.

## Entry and Exit Criteria

The 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 Requirements 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 (includes 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.