



TEST PLAN

Product Name: OpenCart (Frontend)



Prepared by: FARHEEN





Table of Contents

Overview	2
Scope	2
Inclusions.....	2
Test Environments	2
Exclusions	2
Test Strategy.....	3
Defect Reporting Procedure:	4
Roles/Responsibilities.....	4
Test Schedule	5
Test Deliverables.....	5
Pricing	5
Entry and Exit Criteria	5
Suspension and Resumption Criteria.....	6
Tools.....	7
Risks and Mitigations	7
Approvals.....	7



Overview

Within the context of the project, has been tasked by OpenCart with the testing of several functionalities within the web application located at "https://demo.opencart.com/".

This document functions as an overarching test planning guide, encompassing information about project scope, testing strategy, resource needs, testing timeline, as well as the deliverables and their respective schedules.

Scope

Within the project's framework, the focus lies on testing specific functionalities within the 'https://demo.opencart.com/' web application, encompassing the following features.

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

Based on our comprehension, we are of the view that the functional aspects mentioned above require testing.

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

'Farheen' has engaged in correspondence with 'OpenCart' and has comprehended the requirement to carry out Functional Testing for all the functionalities delineated in the aforementioned Scope section.

In the context of Functional Testing, we intend to adhere to the following approach for conducting tests:

Step#1 – Creation of Test Scenarios and Test Cases for the different features inscope.

- During the creation of Test Cases, we will employ various Test Design techniques
 - Equivalence Class Partition
 - Boundary Value Analysis
 - Decision Table Testing
 - State Transition Testing
 - Use Case Testing
- In addition, we leverage our proficiency to craft Test Cases, utilizing the following methods:
 - Error Guessing
 - Exploratory Testing
- We priorities the Test Cases

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

- Initially, we will conduct Smoke Testing to verify the operational status of essential application functionalities.
- Initially, we will conduct Smoke Testing to verify the operational status of essential application functionalities.
- Once we secure a stable build that meets the criteria of the Smoke Testing, we delve into comprehensive testing by implementing the created Test Cases.
- Once we secure a stable build that meets the criteria of the Smoke Testing, we delve into comprehensive testing by implementing the created Test Cases.
- Subsequently, we log the identified bugs into the bug tracking tool and, at the close of the day, dispatch a status email to the development management containing the defects detected during that day.

- As a part of our Testing activities, we will execute the following Testing types:
 - Smoke Testing and Sanity Testing
 - Regression Testing and Retesting
 - Usability Testing, Functionality & UI Testing
- We iterate through Test Cycles until we attain the desired level of product quality.

Step#3 – We will follow the below best practices to make our Testing better:

- **Context-Driven Testing:** We will conduct Testing in accordance with the unique context of the provided application.
- **Shift Left Testing** – We will commence the testing process right from the initial stages of development, foregoing the need to wait for a stable build.
- **Exploratory Testing** – Utilizing our expertise, we will engage in Exploratory Testing alongside the conventional execution of Test cases.
- **End to End Flow Testing** – We will conduct testing of end-to-end scenarios that encompass various functionalities, aiming to replicate the flows experienced by end users

Defect Reporting Procedure

During the test execution –

- We will record any deviations from the expected behavior demonstrated by the application. When it doesn't meet the criteria for classification as a defect, we will log it as an observation or issue, or present it in the form of a question.
- Any problems related to usability will also be documented.
- Once a defect is identified, it will undergo retesting to ensure its reproducibility. We document the defect with accompanying screenshots and detailed steps for reproduction.
- At the conclusion of each day's test execution, we will send a report containing the encountered defects as well as any observations.

Note:

- We will record defects in an Excel spreadsheet.
- Test scenarios and Test cases will be recorded within an Excel document.



Roles/Responsibilities

Following is the role or responsibilities are distributed for the project-

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 defect reports 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	Start Date to End Date
▪ Test Case Creation	Start Date to End Date
▪ Test Case Execution	Start Date to End Date
▪ Summary Reports Submission	Date

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

NA

Entry and Exit Criteria

The following criteria define the conditions for initiation and conclusion in each stage of the Software Testing Life Cycle:

Requirement Analysis

Entry Criteria:

- Once the testing team obtains the Requirements Documents or information pertaining to the project

Exit Criteria:

- The Testing team delves into the list of Requirements, comprehending their intricacies and implications and queries are addressed and resolved

Test Planning

Entry Criteria:

- The Testable Requirements are formulated based on the provided Requirements Documents or Project details
- Doubts are cleared

Exit Criteria:

- The Test Plan document, which encompasses the Test Strategy, is approved 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

- As per the Client's determination, we will either pause or recommence the project activities.
- We are prepared to scale up or scale down the resources in accordance with the Client's requirements.

Tools

Here is a compilation of the Tools that will be utilized throughout this Project:

- XYZ Bug Tracking Tool
- Mind map Tool
- Snipping Screenshot Tool
- Word and Excel document

Risks and Mitigations

Below is a compilation of potential risks along with corresponding mitigation strategies:

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

The team will submit various document types for Client approval, as illustrated below:

- Test Plan
- Test Scenarios
- Test Cases
- Reports

Testing will proceed to the next steps only upon successful completion of this approval