TEST PLAN

Product Name: OpenCart (Frontend)

Table of Contents

* Overview
* Scope
* Inclusions
* Test Environments
* Exclusions
* Test Strategy
* Defect Reporting Procedure
* Roles/Responsibilities
* Test Schedule
* Test Deliverables
* Entry and Exit Criteria
* Tools
* Risks and Mitigations
* Approvals

Overview

As part of the project, 'OpenCart' has requested testing of various functionalities of the 'https://demo.opencart.com/' web application. This document outlines the high-level test planning with details on the project scope, test strategy, schedule, resource requirements, deliverables, and approval process.

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

 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

we need to perform Functional Testing of all the functionalities mentioned in the above Scope section.

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.

 Create test scenarios and test cases with an emphasis on boundary value analysis, equivalence class partitioning, and exploratory testing.

 We prioritizes the Test Cases for efficient testing.

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

 Begin with Smoke Testing to verify critical functionalities.

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.

 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.

 Execute a range of tests, including Functional, Regression

Usability, and UI testing.

 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

 Checklist based Testing – We will do the checklist based testing which to get new defects

Defect Reporting Procedure: During the test execution –

 Behavious during test cases execution are noted

 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.

 use different techniques to solve the defects.

Note:

 Defects will be documented in a excel file.

 Test scenarios and Test cases will be documented in an excel document.

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 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 |  |
| ▪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 |  |
| Functional Test Cases | Test Cases created for the scope defined |  |
| Defect Reports | Detailed description of the defects identified along with screenshots and steps to reproduce on a daily basis. |  |
| Summary Reports | Summary Reports – Bugs by Bug#,  Bugs by Functional Area and Bugs by Priority |  |

Entry and Exit Criteria

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

**Requirement** **Analysis**

Entry Criteria:

* Initially, Testing team reviews Requirements Documents or details about the Project.

Exit Criteria:

 List of Requirements are explored and understood by the Testing team  requirements are analyzed by the team.

**Test** **Planning**

Entry Criteria:

 by using the given document, the testable requirement are derived.

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 closed 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:

 Analyze and test Summary Reports

Tools

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

 Bug Tracking Tool

 Word and Excel documents

Risks and Mitigations

The following are the list of risks possible 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 like below:

 Test Plan

 Test Scenarios  Test Cases

 Reports

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