Test Plan

Version 1.0

(Created By: MAYANK RAGHAV)

Table of Contents

Team Members

Role	Name
Project Manager	SANKAR
Development Team	ANU, XYZ
QA Team	RAHUL MAYANK, SUMIT

Document Log

	Name	Designation	Version Number	Date
Created By	MAYANK RAGHAV	QA	Version 1.0	11-1-2024
Reviewed By	XYZ	QA Head	Version 1.0	11-02-2024

1. Introduction

1.1 Overview of Open cart Web Application

Open cart is a free, open source e-commerce platform that gives a foundational support for online merchants. This web application can be used by web developers to shop owners who want to build their own websites and try different features for their websites. Open cart consists of two parts.

1: Open cart Frontend (Web Application)

2: Open cart Backend. (Admin Dashboard)

Open cart frontend will be discussed in this test plan. The basic features of Open cart (frontend) web application are given below:

1. Gives foundational support for anyone who wants to build their own website.
2. Users can freely register in this web application.
3. This is a free open source ecommerce web application.
4. It has more features home page with top header more informative information like (contact no, my account, wish list, shopping cart, checkout) and header second ,footer top (pages list)and footer second (copyright information) and a menu bar with a search option.
5. It has a product display page.
6. Users can register and login to this website and search for a product.
7. It has a shopping cart and users can add products to the shopping cart.
8. Checkout page is available with payment options and users can buy products internationally using their payment option.
9. Users can give reviews to the products.
10. Gift vouchers are also available and users can buy it..

1.2 Purpose of Test Plan

Test plan document served as a draft for the Open cart. It covers the overall test strategy, scopes of testing, resources that are required and methods and processes used to test the release. Also this document will give the overall idea of the Open cart product to any new team member joining the team.

2. Scope

2.1 In Scope

The testing of the following features of in scope is included in the scopes of this project.

Web Application (Frontend)

The basic features of the web application are given below:

1. Home

2. Header and Footer
3. Menu bar
4. Search
5. Register
6. Login
7. Forgot Password
8. Logout
9. My Account
10. Edit Account
11. Change Password
12. Address Book
13. Wish list
14. Payment
15. Order History
16. Downloads
17. Reward Points
18. Product Returns
19. Transactions
20. Affiliate Account
21. Newsletter
22. Gift Certificate
23. Shopping Cart
24. Checkout
25. Contact Us
26. Specials
27. Site Map
28. Brands
29. Product Compare
30. Product Display

QA Resources Allocated:

1. MAYANK RAGHAV
2. RAHUL
3. SUMIT

QA Backup Resources:

ABC

2.2 Out of Scope

1. Test automation.
2. All the features that doesn't mentioned in scope.
3. Any third party features.

2.3 Test Environments

- **Operating System :** Windows 10
- **Browser:** Chrome, Firefox, Safari, Edge, I9, I10, I11

2.4 Testing

- **Manual:** Unit testing, functional testing, cross browser testing, mobile and web application testing .
- **Automation:** automation with Cypress Automation JavaScript tool.

2.5 Testing Types

1. Functional Testing
2. Non Functional Testing
3. Cross Browser Testing
4. Positive Testing
5. Negative Testing
6. Happy Testing
7. Black Box Testing
8. GUI Testing
9. Exploratory Testing
10. Smoke and Sanity Testing
11. Regression and Retesting

2.6 Entrance/Exit Criteria

2.6.1 Entrance Criteria

- All developed code must be unit tested.
- Test Environment is set up.
- All credentials are provided (Test user accounts etc.)

2.6.2 Exit Criteria

- All high priority errors from functional test must be fixed and tested.
- Full project team must be comfortable with the quality of the project before going to the production stage.
- If any medium or low priority errors are outstanding the implementation risk must be acceptable by business representatives.
- Final sign off by stakeholders and business IT personnel.

3. Test Strategy

After communicating with the 'Open Cart' client it was understandable that functional testing has to be done in the project in the features of in scope section.
The below approach will be follow as part of functional testing:

Step1 : Creation of Test Scenarios and Test Cases for the different features in Scope.

1. We will apply several Test Designing techniques while creating Test Cases
   - Equivalence Class Partition
   - Boundary Value Analysis
   - State Transition Testing
2. We also use our expertise in creating Test Cases by applying the below:
   - Error Guessing
   - Exploratory Testing
3. We prioritize 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.
- We then report the bugs in bug report document and send it to the PM
- As part of the Testing, we will perform the below types of Testing:
  1. Smoke Testing and Sanity Testing
  2. 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.

4. Deployment Test Plan

On Test, Stage, Production (will be shared later).

5. Deliverables

The test deliverables are given below.

Deliverables	Description	Targeted End date
Test Plan	Detailed planning document of testing process, deliverables, resources required for testing, test schedule and scope of the project.	08.09.2023
Test Case Scenarios and Test Case development	Test cases developed on the features of in scope.	15.10.2023
Bug Report	Bugs that are found during testing are reported with screenshots and videos and will be provided weekly at the PM discretion.	12.11.2023
RTM	RTM will be given to the PM after the bug report finishes to get the entire view of overall testing.	13.11.2023

6. Testing Process

The testing processes from test schedule to the way the test will be done are discussed in this section.

6.1 Test Schedule

The Test schedule for the overall testing process is given below.

Task	Start date	Finish Date
Test Plan Creation	11.01.2024	15.01.2024
Test Scenarios and Test Cases Creation	17.01.2024	21.01.2024
Test Execution	22.01.2024	25.01.2024
Bug Report Submission	3.02.2024	16.02.2024
RTM Submission	13.01.2024	13.02.2024

6.2 Testing

- QA will develop test scenarios.
- QA will develop test cases based on the test scenarios.
- QA will execute test cases.

6.3 Defect Reporting

- QA will make bug reports using Microsoft Excel and Jira .
- QA will make RTM (Requirement Traceability Matrix) based on the test case Execution and bug reports.
- QA will assign bugs to Project Manager.

6.4 Fixing

- Developer will fix the assigned bug and assign it to QA.

6.5 Verification

- QA will verify the fix on assigned bugs.

6.6 Closure

- If bug is fixed QA will close the bug.

6.7 Not Fixed

- If bug is not fixed QA will re-assign the bug to the developer.

7. Error Management and Configuration Management

In time of functional test if any bugs are found it will be mentioned in bug report and if any duplicate bugs remain, PM shall close the bug to avoid re-work.

Bugs which are agreed as valid will be categorized by the error review team. The category are given below:

High: Serious errors that prevents system tests of a particular function continuing or serious data type errors

Medium: Serious or missing data errors that will not prevent implementation.

Low: Minor errors that do not hinder any functionality.

8. Issues, Risks and Assumptions

8.1 Testing Needs

- Testing Server for Web application test.
- Cypress for taking screenshots.
- Microsoft Word for creating test plan.
- Microsoft Excel to design test cases and write bug reports.
- Jira for bug tracking.
- Click Up for project management.

8.2 Issues/Risks

No further changes or inclusion will be considered for inclusion in this release except

1. Where there is the express permission and agreement of the Project Manager and Business Representative.
2. Where the changes will not require significant effort on behalf of the test team and will not adversely affect the schedule.

This is a serious issue, as any major changes to design will entail additional time to re-plan testing and to create or amend test conditions.

8.3 Assumptions

- Required resources available.
- Project is of required quality.
- Project will be delivered on time.
- All documentation will be up to date and delivered to the functional test team.