OpenCart
- Test Plan -

Revision History

Date	Description	Author	Comments
Date of issue	Test Plan for OpenCart version 4.0.0.0	*Author Name*	Test plan version 1.0 (initial version)
10.05.2024	Test Plan for Catalog	Proj. Mngr: TBD	Version 4.0.0.0

Table of Content:

1. Introduction

OpenCart is free open source e-commerce platform for online merchants. OpenCart provides a professional and reliable foundation from which to build a successful online store. This foundation appeals to a wide variety of users; ranging from seasoned web developers looking for a user-friendly interface to use, to shop owners just launching their business online for the first time. OpenCart has an extensive amount of features that gives you a strong hold over the customization of your store. With OpenCart's tools, you can help your online shop live up to its fullest potential.

1. Project Objective

The objective of this application is to increase the trust in the quality of the application, to find defects in the live software application, to evaluate the product risks of the application, and to send back to the stakeholders information with regards to the testing process and the level of quality of the application after the improvement process.

2. Functionalities in scope

For this version of the application the functionalities in the scope of testing are:

For the Admin Module, the functionalities in the field are related to the management of all products, including the total number of products sold, customers, orders and the number of people visiting the site, as well as general settings related to organization and configuration.

For this version of the app, functionality in the test domain of the Catalog Module differs based on user rights.

The administrator can:

- View total orders placed, total sales, total customers, number of people online
- Generates reports on the total number of products in stock
- Configure the list of manufacturers and products
- View the details of each order
- Can add or delete the specifications of each product

The ESS user can:

- View each product category, their price and other specifications
- Generate orders
- View detailed product information
- Place your order

- Throughout the testing process we will perform functional testing and some types of non-functional testing (like usability testing), positive testing and negative testing and also, as needed, we will perform retesting and regression testing.
- Some other types of testing that might be performed if necessary are smoke testing and sanity testing.
- Some other types and techniques of testing will be decided accordingly after requirement analysis.
- Testing will be performed at a system testing level.

1.3 Functionalities and tests out of scope

- All the other modules except **Catalog Module** will not be tested throughout this project (for which this the test plan is being done)

- Non-functional testing like performance (stress testing, load testing, volume testing, scalability testing, spike testing) and security is beyond the scope of this project.
- No QA support for mobile applications developed. Only web applications will be tested.
- Automation testing is beyond scope

2. **Test process**

1. **Test planning**

Roles and responsibilities

Role 1 - Name (TBD)	Test Lead - Will monitor the proper functionality of the test process, the involvement of the teams and the reach of the defined deadlines
Role 2 - Name (TBD)	will test: Dashboard -> Catalog-> Categories
Role 3 - Name (TBD)	will test: Dashboard -> Catalog-> Products
Role 4 - Name (TBD)	will test: Dashboard -> Catalog -> Subscription Plans
Role 5 - Name (TBD)	will test: Dashboard -> Catalog -> Filters
Role 6 - Name (TBD)	will test: Dashboard -> Catalog -> Attributes->Attributes
Role 7 - Name (TBD)	will test: Dashboard -> Catalog -> Attributes-> Attribute Groups
Role 8 - Name (TBD)	will test: Dashboard -> Catalog -> Options
Role 9 - Name (TBD)	will test: Dashboard -> Catalog -> Manufacturers
Role 10 - Name (TBD)	will test: Dashboard -> Catalog -> Downloads
Role11 - Name (TBD)	will test: Dashboard -> Catalog -> Reviews
Role12 - Name (TBD)	will test: Dashboard -> Catalog-> Information

Entry criteria:

- Testing environment is up and running (being an already live application, we will have the environment ready even before the implementation step)
- Business requirements are completed by the analysis team and are delivered to the appropriate testing team for evaluation
- Potential project risks are detected and mitigated
- Roles and responsibilities are allocated
- Test plan should be finalized before entering the next phase of testing
- Define the objectives of testing and the accepted level of quality
- Developed test cases: Detailed test cases should be created that describe the steps to verify each functionality. These test cases should cover both positive and negative scenarios to ensure thorough testing.
- Prepared test data: A variety of test data sets simulating real-world scenarios should be prepared for testing. This includes product data, customer accounts and orders with different variants (payment methods, shipping options, etc.).

Exit criteria:

- 90% or more of the tests are passed
- No critical issues have status open
- All detected errors have been reported
- The budget was reached
- The deadline was reached
- The objectives were fulfilled
- The product usage documentation has been finalized with the scenarios evaluated during the testing phase
- Test completion report has been created and sent to the stakeholders
- Product risks have been identified and mitigated
- Not finding bugs of major severity in a specific period of time
- The existing bugs that were reported must have been fixed and followed by retesting and regression testing

Test scope

Tests in scope:

In order to meet our testing goals, we will only focus on the Catalog Module that has been scoped for testing and targeted for improvement over the next two months.

In terms of testing techniques, we will mostly use black box testing with the following test design techniques:

- equivalent partitioning

- boundary analysis
- decision table

In terms of type of testing, we will use non-functional testing where we will only cover usability testing and compatibility testing. Positive and negative testing should also be performed, and (as needed) retesting and regression testing will be done when defects are fixed or changes of any kind are made to the code.

Tests not in scope:

We are not going to cover during the testing process any techniques related to whitebox testing.

Also, performance and security testing will not be performed during this session of testing.

From the perspective of the modules covered, any other functionality that is located outside of the login or register module are not to be tested.

Project risks:

- The team does not have the proper knowledge or experience in order to guarantee the desired level of quality for the application
- Not enough time has been allocated in order to properly test and cover all the functionalities in scope
- All that the data that is going to be used will have to be created explicitly in the scope of testing, which will cut off from the time allocated for testing, generating a risk of not reaching the deadline

Product risks:

- All the data that is going to be used will be test data, which will not give us an experience of the application close enough to the ones that the user will experience
- Taking into account that only two modules are in the scope of testing, the rest of the modules will still be at risk of not fulfilling the user needs

2.2 Test analysis

- In this phase we will analyze the business requirements that were provided and we will create test conditions based on the received requirements.

2.3 Test design

- In this phase we will create the test cases based on the previously defined test conditions to ensure that we will be covering all the functionalities that are in scope of this project.
- The test data that will be needed will be identified in this phase based on the identified data necessities from the created test cases

2.4 Test implementation

- We make sure that all the test data is available and reviewed (test data = email examples, password examples, different type of currency, different types of credit cards)
- We make sure that the setup environment is up and running
- We make sure that we have all the needed access and permissions to all the systems involved in the validation process
- We prioritize the tests based on risks (if known) and business priority

2.5 Test execution

- The tests will be executed on the top 4 used browsers: Chrome, Mozilla Firefox, Microsoft Edge, Apple Safari. If time will be available we will extend tests on Opera and Brave browsers
- We will create bug reports when the expected results that were defined in the test cases are different from the actual results
- We will perform retesting and regression testing to make sure that all the bugs have been fixed and no previously working functionality was not affected by the changes
- We will generate the test status reports once a week and send them to the management team in order to provide them with means to monitor the testing process and take measures in case new risks are identified

2.6 Test closure

- We will evaluate the exit criteria and we will make sure that it was fulfilled in order to green light the launching of the new product version into the production environment
- We will generate the test completion report and send it to stakeholders in order to inform them about the testing process results and enhance them with the ability to make informed decisions with regards to the product launching
- All the product risks will be detected and mitigated (a solution was found in order to reduce the probability of them to arise) or a contingency plan has been set in place
- All the performed test cases will be executed in the test management tool and all the remaining bugs that have been retested and fixed have been closed
- Regression testing will be performed and no other issues have been detected
- Learned lessons will be gathered and collected into a common improvement plan in order to enhance the improvement of the next testing processes

2.7 Test monitoring and control

- We will evaluate the test status reports and monitor them all throughout the testing process in order to ensure a smooth testing and

team collaboration and in order to make sure that new risks are identified in time and managed accordingly

- In case new risks will appear they will be mitigated or a contingency plan will be set in place to make sure that the negative effects will not stop us from fulfilling the testing objectives that were defined in the planning phase

3. **Test deliverables**

The following test deliverables will be provided by the end of the testing process and sent to the stakeholders in order to create the basis of informed decision:

- Test plan
- Test conditions
- Test cases
- Daily test summary report
- Traceability matrix
- Test case results
- Bugs report
- Test completion report

This document outlines the test plan for the OpenCart-based e-commerce shop application. It details the testing process, deliverables, and resources required to ensure a high-quality application.

Test Objectives

- Verify core functionalities across various modules (Categories, Products, Subscriptions, Filters, etc.).
- Ensure a seamless user experience for both customers and administrators.
- Identify and report bugs, defects, and usability issues.
- Validate the application meets the defined business requirements.

Test Scope

The testing scope will encompass functionalities for both customer and administrator users. Key areas include:

- **Customer-facing functionalities:**
  - Product browsing and search with filters (category, attributes, price etc.)
  - Product details including images, descriptions, attributes, options, and reviews.
  - User account creation, login, and management.
  - Shopping cart functionality (adding, removing, modifying items).
  - Secure checkout process with payment gateway integration.
  - Order tracking and history.
- **Administrator functionalities:**

- Product management (add, edit, delete) with categories, options, attributes, and downloads.
- Subscription plan definition and management.
- Customer account management and review moderation.
- Order management system (processing, fulfilling, tracking orders).
- Content management system for informative pages.

Test Deliverables

- **Test Plan:** This document outlines the overall testing strategy.
- **Test Cases:** Detailed procedures for testing each functionality.
- **Test Data:** Data sets used for testing various scenarios.
- **Daily Test Summary Report:** Summarizes daily testing activities and findings.
- **Traceability Matrix:** Links test cases to specific business requirements.
- **Test Case Results:** Documents the outcome (pass/fail) of each test case.
- **Bug Report:** Lists identified bugs with details for reproduction and severity.
- **Test Completion Report:** Summarizes overall testing results and recommendations.

Test Cases

Test cases will be developed for each functionality listed in the test scope. Here are some examples:

- **Customer:**
  - User can successfully register for a new account.
  - User can search for products by category, attribute, and price range.
  - User can add products to the shopping cart and modify quantities.
  - User can proceed through checkout securely using a valid payment method.
  - User can track the status of their orders.
- **Administrator:**
  - Admin can create new product categories with a hierarchical structure.
  - Admin can add a new product with detailed information (description, images, attributes, options).
  - Admin can define subscription plans with pricing and benefits.
  - Admin can process customer orders and update order status.
  - Admin can edit and moderate customer reviews.

Test Data

A variety of test data will be created to simulate real-world scenarios. This may include:

- Products with different categories, attributes, and options.
- Customer accounts with varying levels of activity.
- Orders with different payment methods and shipping options.

Test Environment

Testing will be conducted in a dedicated test environment that mirrors the production environment as closely as possible.

Pass/Fail Criteria

Each test case will have defined pass/fail criteria to determine if the functionality is working as expected. Examples of pass/fail criteria include:

- The system performs the action as intended without errors.
- Expected data is displayed accurately on the screen.
- The user experience is smooth and intuitive.

Test Tools

- Manual testing will be the primary method for functional testing.
- Automated testing tools can be explored for repetitive tasks (regression testing).
- Bug tracking software will be used to document and manage identified issues.

Schedule

A detailed test schedule will be created outlining the timeline for test case execution and reporting.

Resources

- A team of qualified testers with experience in e-commerce applications.
- Access to the OpenCart platform and testing environment.
- Test case management and bug tracking tools.

Conclusion

This test plan provides a framework for a comprehensive testing process for the OpenCart e-commerce shop application. By following this plan and utilizing the defined deliverables, we can ensure the application functions as intended, delivers a positive user experience, and meets the business needs.

4. Schedule

- The testing process will take place over a period of 1.5 months and will involve all the activities defined in the previous section
- All the resources will be adapted accordingly in case new testing resources are detected as necessary