TEST PLAN

Product name: New e-commerce website

Prepared By: Hansika Muthumali

Date: 19th Sept,

Introduction

This test plan outlines the approach to validate key functionalities of a new e-commerce website, focusing on critical features like payment methods, shipping options, address management, order summary and review, and order confirmation with receipt generation. The goal is to ensure the system functions as expected and delivers a seamless shopping experience.

Scope

The scope of the project includes testing the following features of the web application

Inclusions

- Multiple payment methods
- Shipping options and address management
- Order summary and review
- Order confirmation and receipt generation

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’.
- Test automation

Test Strategy

As part of functional testing, we will follow the below approach for testing.

Step 01 – Test Scenario and Test Case Development:

- We begin by drafting test scenarios and detailed test cases for each feature within the testing scope.
- During test case creation, we apply several test design techniques to ensure comprehensive coverage:

    - **Equivalence Partitioning** to group inputs and reduce redundancy.
    - **Boundary Value Analysis** to test at the edge of input limits.
    - **Decision Table Testing** to cover all possible combinations of inputs and outputs.
    - **State Transition Testing** to examine the application's behavior under various state changes.
    - **Use Case Testing** to validate real-world user interactions.
- Additionally, we leverage our expertise to craft effective test cases using:
    - **Error Guessing** to anticipate likely problem areas.
    - **Exploratory Testing** to discover issues through hands-on interaction.
- Test cases are then prioritized based on their importance and the risk associated with the feature.

Step 02 – Testing Process for New Application Builds:

- When we receive a new build for testing, the first step is to conduct **Smoke Testing**. This ensures that the application's core and critical functions are operational.
- If the build fails this initial test, it's rejected, and testing is paused until a stable version is provided.
- Once a stable build passes Smoke Testing, we proceed with thorough testing using the predefined test cases.
- Multiple testers will work in parallel across various supported environments to ensure coverage across different setups.
- All identified defects are logged in a bug-tracking tool, and a daily status report, summarizing the day's findings, is sent to the development team.
- As part of our testing efforts, we perform:
    - **Smoke Testing** and **Sanity Testing** to verify critical paths.
    - **Regression Testing** and **Retesting** to check for defect fixes and prevent reintroduction of bugs.
    - **Usability Testing**, along with **Functionality** and **UI Testing**, to ensure a smooth user experience and functional reliability.
- The testing cycles are repeated until the product meets the required quality standards.

Step 03 – Best Practices for Enhanced Testing:

- **Contextual Testing** – We tailor our testing approach based on the specific requirements and nature of the application being tested.
- **Shift Left Testing** – Testing begins early in the development process, rather than waiting for a finished build, allowing early detection of issues.
- **Exploratory Testing** – In addition to executing planned test cases, we perform exploratory testing to uncover additional defects based on tester intuition and expertise.

- **End-to-End Testing** – We simulate real user workflows by testing complete processes that span multiple features, ensuring seamless integration and interaction across functionalities.

Defect Reporting Procedure

During Test Execution,

- Any behavior by the application that deviates from the expected outcome will be recorded. If the deviation isn't classifiable as a defect, it will still be logged as an observation, issue, or raised as a question for further clarification.
- Usability concerns will also be captured and reported.
- When a defect is identified, it will be retested to confirm that the issue can be consistently reproduced. Screenshots along with detailed steps to recreate the problem will be documented.
- At the conclusion of each day's testing, a summary of defects found, along with any notable observations, will be shared.

Note that,

- Defects will be recorded in an Excel sheet for tracking purposes.
- Test scenarios and test cases will be maintained and documented in an Excel file as well.

Roles/Responsibilities

Person	Role	Responsibility
A	Test Manager	Handle escalations
B	Test Lead	- Develop the Test Plan and secure client approval - Engage with the application, design and run test cases - Log defects - Manage the test execution process, ensuring defects are valid - Provide daily updates on issues and send defect summaries to the client - Attend client meetings
C	Senior Test Engineer	- Interact with the application - Create and execute test cases - Log defects
D	Test Engineer	- Work with the application - Run the test cases - Report defects

Test Schedule

Below is the test schedule plan for the project.

Task	Time Duration
Test Plan Creation	20, Sept – 25, Sept
Test Case Creation	26, Sept – 05, Oct
Test Case Execution	06, Oct – 10, Oct
Test Summary Report Submission	11 Oct

Test Deliverables

The following are to be delivered to the client.

Deliverables	Description	Target Completion Data
Test Plan	Details covering project scope, testing strategy, schedule, resources, and deliverables.	19, Sept
Functional Test Cases	Test cases that align with the defined scope.	05, Oct
Defect Reports	A comprehensive defect report with screenshots, steps to reproduce, and defect descriptions, sent daily.	N/A
Summary Reports	Overview report including a list of defects by ID, functional area, and priority.	11, Oct

Entry and Exit Criteria

The below shows the entry and exit criteria for every phase of STLC.

- **Requirement Analysis:**

Entry criteria - Once the testing team receives the requirement documents or details of the project

Exit criteria – List of requirements are explored and understood and doubts are cleared by the testing team.

- **Test Planning:**

Entry criteria – Testable requirements derived from the requirements documents

Exit criteria – Test plan document 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 case documents are signed off by the client and application is ready for testing

Exit criteria –Test case reports and defects reports are ready

- **Test Closure**

Entry criteria – Test case reports and defects reports are ready

Exit criteria – Test summary reports

Suspension and Resumption Criteria

We will pause and restart the project according to the client's decision.
The allocation of resources will be increased or reduced based on the client's requirements.

Tools

The tools we plan to utilize for this project include:
• Jira for bug tracking
• Snipping screenshot tool
• Word and Excel for documentation

Risks and Mitigations

Below are potential risks and their mitigation strategies:

- Risk: Unavailability of a team member
  Mitigation: Implement backup resource planning
- Risk: Build URL is inaccessible
  Mitigation: Team will focus on alternative tasks in the meantime
- Risk: Limited time for testing
  Mitigation: Adjust resource allocation dynamically according to client requirements

Approvals

The team will submit various documents for client approval, such as:

- Test Plan
- Test Scenarios
- Test Cases
- Reports

Progress to the next testing phases will only occur after these approvals are received.