Test Plan for Daraz.com.bd

1. Introduction

The purpose of this test plan is to define the testing approach, objectives, and scope for the Daraz.com.bd website. It outlines a structured framework for evaluating its functionality, usability, compatibility, performance, and security. As a crucial platform for technology product showcase and sales, ensuring high-quality standards for a seamless user experience is vital.

1.1 Web Page Overview

Daraz.com.bd is an online platform dedicated to offering a wide range of technology products, including Health & Beauty,cosmetics, TV & Home Appliances, and Automotive & Motorbike. The webpage serves as a hub for users to explore products, compare specifications, place orders, and access customer support. It is crucial for the web page to perform flawlessly across various browsers and devices while ensuring the security and confidentiality of user data.

1.2 Purpose

The primary purpose of this test plan is to ensure the quality and reliability of the Daraz.com.bd webpage by thoroughly assessing its various components and functionalities. The testing process aims to verify that the webpage functions as intended, is user-friendly, performs well under different conditions, and adheres to security best practices.

2. Scope of Testing Approach

2.1 Scope of Testing

The testing approach for the Daraz.com.bd webpage will encompass a comprehensive evaluation of its features, functionalities, and performance across various dimensions. The testing scope includes, but is not limited to:

1. **Functionality Testing:**
    - Verification of core functionalities such as navigation, search, product browsing, filtering and sorting. Some explanation are given below-

    1.Navigation: Test how users move between different sections or pages of the website.

    2.Search: Test the search bar's accuracy in returning relevant results.

    3.Product Browsing: Verify that users can easily browse through products.

    4.Filtering: Ensure that filters work correctly and refine search results.

    • Testing of user account features including registration, login, and password recovery. • Assessment of shopping cart management, including adding or removing products and updating quantities.
    - Validation of the checkout process, including address selection, payment methods, and order confirmation.

2. **Usability Testing:**
    1. Involve real users in usability testing to evaluate how easy it is for them to navigate, search, browse, filter, and sort products.
    2. Gather feedback from users to identify any pain points or areas of improvement.
    3. Testing of responsiveness across different screen sizes and resolutions.
3. **Compatibility Testing:**
    1. Validation of the webpage's rendering and functionality across popular browsers such as Chrome, Firefox, Safari, and Edge.
    2. Testing on various devices, including desktops, laptops, tablets, and mobile phones, to ensure a consistent user experience.

4. **Performance Testing:**

    1. Assess the performance of these functionalities, especially for large catalogs or high traffic scenarios.

    2. Check for latency in search results, load times, and responsiveness during filtering and sorting
    3. Measurement of page load times, ensuring acceptable performance even under varying network conditions.
    4. Evaluation of the webpage's responsiveness to user actions and interactions.

5. **Security Testing:**
    1. Validate that user data is secure, especially when users provide personal information during navigation or search.
    2. Testing of login and logout processes to ensure secure user authentication.

6. **Error Handling and Recovery Testing:**
    1. Assessment of how the webpage handles errors, such as invalid inputs, unavailable products, and server errors.
    2. Validation of error messages for clarity, accuracy, and helpfulness in guiding users to resolve issues.

7. **Automated Testing :**
    1. Consider automating tests for repetitive tasks, such as regression testing and cross-browser testing.
    2. Tools like Selenium, Appium, or Cypress can help automate these tests.

8. **Accessibility Testing:**
    1. Ensure that core functionalities are accessible to individuals with disabilities by testing with assistive technologies, like screen readers.

9.Feedback Loop:
    1. Establish a feedback loop with development teams to address and resolve any issues found during testing promptly.

3. Testing method:

 GUI Testing Functional Testing

i. Unit Testing.
ii. Integration Testing.
iii. System Testing.
iv. Smoke Testing.
v. Interface Testing.
vi. Regression Testing.

 Non-functional Testing

I. Performance Testing
II. Load Testing
III. Security Testing.
IV. usability testing.
V. Environmental Testing

 API Testing

A collection of steps that ensure the backend and front end work together properly and behave as intended in terms of reliability, security, and functionality. within a short period of time, the build architecture's logic must be verified. The test actions listed below are included in each API test. The API Test Plan will go into further depth regarding API Testing.

- Verify the URL accordingly.
- Verify required request headers and their correct values.
- Verify response payload.
- Verify the correct HTTP status code and response headers.
- Verify the expected result and correct application state.
- Verify correct performance sanity.

Automated Testing:

- Automated testing uses automated scripts and testing tools to perform repetitive test cases quickly and accurately. Automated tests are beneficial for regression testing, load testing, and performing a large number of test iterations.

- Use automation frameworks and tools like Selenium, Appium, Cypress, or TestComplete to create automated test scripts for these core functionalities.

Security Testing:

- ❖ Security testing focuses on identifying vulnerabilities, weaknesses, and potential security threats in an application. It ensures data protection and secure access.
- ❖ Techniques include penetration testing, vulnerability scanning, and code review for security

4. Testing Need Environment Setup

Like Browser Setup, Test Data Preparation and Devices with different models and versions.

5. Flow Diagram for Testing

Start Test Planning Test Design Test Environment Setup Test Execution Result Analysis Defect Reporting ---> (back to Test Design/Execution if defects found) Regression Testing Test Closure End.

6. Reporting

- A list of the test cases that were executed.
- A description of the testing environment.
- The names of the testers who participated in the testing.
- The date and time of the testing.

8. Fixing Bugs

Bug fixing is an integral part of the software development and quality assurance process. It involves identifying, diagnosing, and resolving defects or issues (commonly referred to as "bugs") in software to ensure that the product functions as expected and meets its quality standards.

- Steps to reproduce the defect
- Expected behavior
- Actual behavior
- Severity and priority levels

Defect Classification and Prioritization

According to the severity of each issue, there are overall benefits to fixing them and overall risks to leaving them unfixed in the current release..

Severity Level

Severity 1 - Crash or High impact problems that often prevent a user/host from correctly completing an experience/booking.

Severity 2 - Moderate to high-frequency problems with the functionality/UI or UX impact

Severity 3 - Either moderate problems with low frequency or low problems with moderate

frequency; these are minor annoyance problems faced by a number of participants. Severity

4 - Low impact problems faced by few participants; there is a low risk of not resolving these

problems.

Priority Level

1.Critical:-Definition: Defects that cause a complete failure oa critical functionality or severely impact the overall functionality of the application.

2. High: Definition: Defects that significantly impact key functionalities or have the potential to affect a large portion of users.

3. Medium: Definition: Defects that impact non-critical functionalities or affect a small portion of users.

4. Low: Definition: Defects that have minor impacts or are considered cosmetic in nature.

9. Not fix Bugs

The issues are not considered a priority, and the development team is focusing on other tasks or features instead.

There may be a lack of resources or time to address the bugs at the moment.

The team may have decided to leave certain bugs in place because they do not significantly impact the overall functionality of the software or because fixing them could introduce new issues.

In any case, not fixing bugs can have various implications depending on the specific situation and the reasons behind this decision. It's generally considered important to address and fix bugs, especially if they affect the usability, security, or stability of a software application**.**

10. Test Resources

10.1 Human Resources

The following human resources will be involved in testing the Daraz.com.bd webpage:

Testing Team:
QA Analysts: Responsible for designing and executing test cases, and reporting defects. Test Automation Engineers: Develop and maintain automated test scripts for critical functionalities.
Usability Testers: Conduct usability evaluations to ensure an intuitive user experience.
Performance Testers: Execute performance tests to assess page load times and responsiveness.

Security Testers: Identify and report security vulnerabilities and data protection issues.

Development Team:
Developers: Responsible for addressing defects and issues identified during testing. Frontend and Backend Developers: Collaborate to fix UI and functional defects.

Test Data:

> Test data comprises the input values, datasets, and scenarios used in test cases to evaluate the software's behavior. Test data should encompass various test cases, including valid and invalid input, edge cases, and boundary conditions.

Stakeholders:
Product Owners: Provide requirements, clarify user stories, and validate acceptance criteria.
Project Managers: Oversee testing progress, prioritize defect fixes, and ensure timely releases.
Designers: Assist in evaluating UI/UX aspects and design alignment.

10.2 Testing Tools and Environments

The following tools and environments will be used for testing the startech.com.bd webpage:

Testing Tools:
Test Management Tool: Jira for test case management and defect tracking.
Automation Tools: Selenium WebDriver with Java for UI automation.
Performance Testing Tool: Apache JMeter for load and performance testing. Security Testing Tools: OWASP ZAP for security assessments.

10.3 Infrastructure

Test Environments: Dedicated testing environments with duplicate production architecture for precise testing are part of the necessary infrastructure for testing the startech.com.bd website. Workstations for manual testing, creating test cases, and documenting defects are known as "testing workstations."

10.4 Documentation and References

The following documentation and references will be used for testing:

Test Plan Document: Comprehensive test plan document outlining testing approach, scope, and methodologies.
Test Cases: Detailed test cases covering various scenarios including navigation, search, cart management, and checkout.
Defect Reports: Detailed defect reports with descriptions, steps to reproduce, and attached screenshots.
User Stories/Requirements: Business requirements and user stories to guide testing efforts. Design Specifications: Design documents outlining UI/UX expectations and layouts.

11. Verification and Validation

11.1 Verification

Verification is the process of ensuring that the implemented system follows the design requirements and fulfills them. Verification operations for the daraz.com.bd website will include:

Requirements Review:

QA analysts will review the business requirements and user stories to ensure a clear understanding of the expected functionalities.

Design Review:

The testing team will review design specifications and mockups to validate that the user interface matches the intended design.

Test Case Design:

Test cases will be designed to verify each requirement and user story. These cases will serve as a blueprint for validation.

Code Review:

Developers' code will be reviewed by peers to ensure it aligns with design specifications and coding standards.

11.2 Validation

Validation involves assessing whether the Daraz.com.bd webpage meets user expectations and functions as intended. Validation activities will include:
Functional Testing: - QA analysts will execute functional test cases to validate that features, such as search, product details, cart management, and checkout, work as specified.

Usability Testing: -Usability testers will evaluate the user interface for ease of use, intuitiveness, and alignment with user expectations.

Compatibility Testing: -Cross-browser and cross-device testing will be conducted to validate that the webpage functions correctly on different platforms.

Performance Testing: -Performance testers will measure page load times and responsiveness to ensure optimal user experience.

Security Testing: -Security testers will assess the webpage for vulnerabilities and validate that user data is adequately protected.

Successful validation ensures that the software is not only built correctly (as determined by verification) but is also the right product for its intended users and purpose.

11.3 Acceptance Criteria

Validation will be based on acceptance criteria derived from user stories and requirements. These criteria outline the conditions that must be met for each feature to be considered acceptable. The testing team will ensure that these criteria are fulfilled through rigorous testing and verification.

12. User Acceptance Testing (UAT)

Upon successful completion of validation testing, user acceptance testing (UAT) will be conducted. In this phase, stakeholders and actual end-users will perform tests to validate that the Daraz.com.bd webpage meets their expectations and serves their needs.

13 Tools and Defect Tracking

Jira will be used for defect reporting and issue bugs/defects management and traceability.

14. Conclusion

The testing efforts for the Daraz.com.bd webpage have been comprehensive and systematic, aiming to ensure the quality, reliability, and usability of the platform. Through diligent testing across various dimensions, valuable insights have been gained regarding the webpage's performance, functionality, security, and user experience.

14.1 Achievements and Milestones

Throughout the testing lifecycle, several key achievements and milestones were reached:

Successful completion of all testing phases, including functionality, usability, compatibility, performance, and security testing. Identification and resolution of various defects across different severity levels, enhancing the overall stability and user experience. Validation of critical user scenarios, ensuring the smooth flow of key processes such as product selection, cart management, and checkout. A thorough assessment of security mechanisms leads to the identification and mitigation of potential vulnerabilities. Collaboration and engagement with stakeholders, including developers, designers, and product owners, to ensure alignment with project goals.

14.2 Readiness for Deployment

Based on the comprehensive testing efforts, the startech.com.bd webpage is deemed ready for deployment with the following considerations:

Functionality: The core functionalities of the webpage have been thoroughly tested and validated, demonstrating their reliability and accuracy.

Usability: Usability testing has confirmed that the user interface is intuitive and user-friendly, providing a positive browsing and shopping experience.

Compatibility: Cross-browser and cross-device testing has ensured that the webpage performs consistently across various platforms and devices.

Performance: Performance tests have indicated acceptable page load times and responsiveness, even under varying user loads.

Security: Security assessments have led to the identification and mitigation of potential security vulnerabilities, enhancing user data protection.

14.3 Recommendations

The Daraz.com.bd website is rated ready for deployment, although the following suggestions are made for further improvements and ongoing work:

Maintaining the best user experience during peak usage requires constant monitoring and performance metric optimisation. Conduct security assessments frequently to address new risks and keep a strong security posture. After deployment, take into account getting user feedback to improve the user experience based on actual usage..

14.4 Acknowledgments

Without the coordinated efforts of the testing team, development team, stakeholders, and all other involved parties, the testing process would not have been able to be completed successfully. The improved quality of the Daraz.com.bd website is a result of the effort, skill, and dedication shown by each team member.