SWEET SHOP

Project Report

Submitted by:

NEELIMA T

TO

SOFTWARE TESTING

Luminar Techno Lab

Calicut

DECEMBER 2024

ABSTRACT

The SweetShop website is an e-commerce platform designed to provide customers with a seamless online shopping experience for various sweets, chocolates, and baked goods. It is a user-friendly online platform designed to make buying sweets quick, easy, and enjoyable while ensuring a smooth digital shopping experience. It serves as an excellent platform for testing fundamental web functionalities due to its core features and real-world application scenarios. The manual testing process aims to ensure the website operates as intended, providing a smooth experience for users to purchase sweets.

Key areas of focus during testing include functional validation, usability, security, and compatibility. Functional testing involves verifying the accuracy and responsiveness of core features such as user authentication, admin management, account management, product browsing & search, shopping basket & wishlist, checkout & payment processing, order management & tracking, user profile & account settings, notifications & emails, customer support & feedback. Usability testing assesses the intuitive design, ease of navigation, and clarity of information provided to users. Compatibility testing ensures the website performs consistently across different browsers and devices.

Specific test cases include verify that new users can register successfully with valid credentials, checking login functionality with valid and invalid credentials, ensure products are managed and displayed correctly, verify search functionality, verify basket functionality, ensure users can enter, update, and save shipping addresses, verify successful and failed payments using different payment methods, check for secure transactions and proper error messages for failed payments,ensure order confirmation emails/messages are sent correctly, ensure users can track orders and get returns, verify that users can manage profile,manage session expiry etc.

Furthermore, performance factors like page load speed and system responsiveness under different user loads are carefully evaluated. Security assessments, including the safe handling of payment information, are also integral to the testing process.

By conducting thorough manual testing, critical issues like user authentication failures, inadequate input validation, and checkout errors can be detected and addressed, enhancing the website's reliability and user experience. The Sweet Shop's straightforward design provides an ideal platform for refining testing skills on a functional yet easy-to-navigate web application.

CONTENTS

INTRODUCTION

The manual testing project focuses on the Sweet Shop website, a simple e-commerce platform for purchasing sweets. The primary objective of this project is to ensure the website functions as intended by identifying and reporting any defects that may hinder user experience or operational efficiency.

Manual testing was performed to thoroughly analyze the website's functionalities, usability, and performance. Various testing methods, such as functional testing, usability testing, compatibility testing, and security testing were applied to validate the core processes including user management, product browsing and searching,product pricing and stock management, shopping basket management, checkout & payment processing, order management and tracking.

This project highlights the importance of manual testing in real-world scenarios. Unlike automated testing, manual testing allows a tester to approach the system as an end-user, identifying subtle issues that automated tools might overlook. It also ensures a deeper understanding of the application’s behavior and helps in uncovering usability flaws, design inconsistencies, and critical bugs.

Through this project,improper form validations,error handling, authentication issues and other functional issues were identified and documented. These findings contribute to enhancing the reliability, functionality, and overall user experience of the SweetShop website. This manual testing initiative underscores the value of human insight in delivering a robust, user-friendly application.

TEST ENVIRONMENT

HARDWARE REQUIREMENT :

DEVICE : LENOVO
PROCESSOR : AMD Ryzen 5 7520U
NETWORK : Stable internet connection

SOFTWARE REQUIREMENTS :

OPERATING SYSTEM : Windows 11
WEB BROWSERS : Google Chrome

TEST DOCUMENTATION TOOL : Google SpreadSheet

SWEET SHOP

A **Sweet Shop e-commerce Website** is an online platform where customers can browse, order, and purchase a variety of sweets, desserts, and confectionery products. It provides a seamless shopping experience, enabling users to explore different sweet categories, view product details, add items to their basket, and complete purchases through a secure checkout process.

The **Sweet Shop** Website offers a user-friendly interface designed to provide a pleasant browsing and shopping experience for sweet enthusiasts.The website's design is clean and straightforward, focusing on ease of navigation and clear presentation of products, making it simple for users to find and purchase their favorite sweets. Users can easily login, browse sweets and add items to basket.Every order is saved and can be viewed as previous orders.The website provides a simple UI for checkout and payment process .

The website is particularly ideal for Sweet Lovers & Enthusiasts, as it focuses on providing a convenient online shopping experience for sweets with user-friendly navigation & accessibility.

The **Sweet Shop** website can emphasize security and customer satisfaction with **s**trong authentication, safe browsing & HTTPS protocol, user-friendly shopping experience. Its intuitive design, combined with efficient functionality, makes it a best platform for sweet browsing. By implementing these security measures and customer satisfaction strategies, the Sweet Shop website can build trust, reliability, and a loyal customer base while ensuring a secure online shopping experience.

MANUAL TESTING METHODS USED IN PROJECT

Functional Testing

- **Smoke testing:**
  Verifies that all features and main functionalities of the website work as expected.Verify the stability of build.
- **Sanity Testing:**
  Checks all the sub functionalities deeply
- Testing forms, buttons, links, navigation, and user workflows (e.g., registration, login, payment).
- **Integration testing**
- **End to end testing** to check flow of functionality.

Usability Testing

- Focuses on the user experience to ensure the website is intuitive, easy to navigate, and user-friendly.
- Evaluates page layout, readability, navigation structure, and content clarity.

Interface Testing

- Tests the interaction between the website, validates the communicatin and data exchange between different software components,modules,ensure they interact correctly and without errors.

Compatibility Testing

- Checks the website's functionality across different browsers (e.g., Chrome, Firefox, Microsoft Edge)
- Check if all third-party integrations work smoothly without errors that includes payment gateways

Performance Testing

- Measures website speed, response time, and stability under different load conditions.
- Includes load testing, stress testing, spike testing and database performance testing.

Security Testing

- Identifies vulnerabilities such as unauthorized access, data breaches, and security loopholes.
- Verifies encryption, authentication mechanisms, and session management

Database Testing

- Verifies the database integrity, data storage, and retrieval.
- Ensures data consistency and security.

TEST SCENARIOS

SWEET SHOP REPORTS

TEST CASE

SWEET SHOP REPORTS

BUG REPORT

SWEET SHOP REPORTS

CONCLUSION

Manual testing plays an important role in ensuring the quality and reliability of any website. Through the comprehensive testing of the **Sweet Shop** e-commerce website, several defects were identified that could impact user experience and security. These include issues such as acceptance of invalid login data and payment data with improper error handling, validation errors allowing invalid payment data including credit card numbers, and cvv. Critical security issues with incorrect passwords, unencrypted CVV, and no validation for address and zip codes. Essential features such as an admin panel, product detail page, wish lists, reviews and ratings, order tracking, estimated delivery dates, gift options, privacy/security policies, and contact pages are missing. These defects highlight the critical importance of manual testing in uncovering functional and usability issues that automated testing might overlook.

The acceptance of invalid data and the absence of error messages highlight weaknesses in input validation and user feedback mechanisms. These essential elements contribute to a seamless user experience and play a crucial role in maintaining the platform’s credibility. The acceptance of invalid passwords in login forms and invalid credit card numbers in payment pages poses significant security risks, as it undermines the integrity of financial transactions. Additionally, it allows invalid email/password combinations, accepts random addresses, and fails to remove purchased items from the basket.

Manual testing played a crucial role in uncovering these issues, as it enables testers to replicate real user interactions, navigate unconventional workflows, and evaluate the website's overall functionality and usability. This hands-on method helps detect subtle defects that automated testing may overlook.

The findings from this testing project highlight the critical role of comprehensive testing in ensuring a website's quality throughout its development lifecycle. By addressing the identified defects, The Sweet Shop website is functional. Still, it requires critical fixes in security, form validation, user authentication, checkout process, and usability improvements to ensure a smooth

and secure shopping experience. Further functional, usability, and security testing is recommended before deployment.

REFERENCES

- https://sweetshop.netlify.app/
- https://www.browserstack.com/guide/