# Change log

|  |  |  |  |
| --- | --- | --- | --- |
| **version** | **date** | **description** | **Author** |
| **1.0** | **10/16/2024** | **Document created** | **Milena Tsonkova** |
|  |  |  |  |
|  |  |  |  |

# 1. Introduction

This test plan outlines the testing approach, scope, objectives, resources, schedule, and responsibilities for the e-commerce project. The goal of testing is to validate that the e-commerce platform functions correctly, provides a secure and seamless user experience, and meets business requirements.

# 2. Objectives

The objectives of testing are:

* To ensure that the core e-commerce functionality (search, browsing, adding to cart, checkout, payment) works as expected.
* To verify that the system meets performance, security, and scalability requirements.
* To ensure a smooth and consistent user experience across different devices and browsers.
* To identify and fix bugs before deployment to production.

# 3. Scope of Testing

## 3.1. In-Scope

* **Functional Testing**: Product browsing, search, add-to-cart, checkout, and payment processing.
* **Performance Testing**: Testing the website’s load capacity, stress limits, and scalability.
* **Security Testing**: Ensuring that sensitive information (e.g., personal and payment data) is secure.
* **API Testing**: Testing third-party service integrations (payment gateways, shipping providers).
* **Cross-browser Testing**: Validating that the e-commerce platform works across different browsers and devices.
* **Regression Testing**: Ensuring new updates do not break existing functionality.

## 3. 2. Out-of-Scope

* Non-e-commerce features (e.g., content management system pages).
* Back-end infrastructure configurations not directly related to the e-commerce functionality.

# 4. Features to be Tested

## 4.1. Core Functionalities

* **Product Catalog Management**: Adding, updating, deleting products.
* **Search and Filtering**: Ensuring users can search for products and apply filters based on categories, price, etc.
* **Shopping Cart**: Add/remove products, update quantity, display totals, apply discounts.
* **Checkout Process**: Billing/shipping details, payment method selection, order summary.
* **Payment Gateway**: Validation of credit card, PayPal, and other payment methods.
* **Order Management**: Placing orders, viewing order history, order status tracking.
* **User Authentication**: Sign-up, login, password reset, social media logins.
* **Discount and Promotions**: Application of discount codes and promotional offers.

## 4.2 Non-Functional Testing

* **Performance**: Load testing for peak traffic scenarios, stress testing.
* **Security**: Data encryption, vulnerability testing, secure login, and payment handling.
* **Usability**: User interface validation, mobile responsiveness, user experience consistency.
* **Cross-Browser Compatibility**: Testing on major browsers (Chrome, Firefox, Safari, Edge).
* **Cross-Device Compatibility**: Ensuring functionality on mobile, tablet, and desktop devices.

# 5. Features Not to Be Tested

* Third-party plugin performance beyond basic functionality integration.
* Content (text, blogs, static information) unrelated to e-commerce transactions.
* Non-production server performance.

# 6. Test Approach

## 6.1. Functional Testing

* **Method**: Manual and automated testing.
* **Tools**: Selenium, Cypress for UI automation; Postman for API testing.
* **Coverage**: End-to-end coverage of all core e-commerce features.

## 6.2 Usability Testing

* **Method**: Manual testing with a focus on user interaction.
* **Tools**: BrowserStack, mobile devices for device and browser compatibility.
* **Coverage**: Test the interface on both desktop and mobile devices for ease of navigation, clear visuals, and intuitive workflows.

## 6.3 Performance Testing

* **Method**: Automated load testing.
* **Tools**: JMeter or LoadRunner.
* **Coverage**: Simulate multiple users performing transactions to verify response times, server loads, and website stability during peak usage.

## 6.4 Security Testing

* **Method**: Manual and automated penetration tests.
* **Tools**: OWASP ZAP, Burp Suite for vulnerability scanning.
* **Coverage**: Testing the security of login, payment, and order processes. This includes checking for SQL injections, XSS vulnerabilities, data leakage, and secure payment encryption.

## 6.5 Regression Testing

* **Method**: Automated regression testing following each code deployment.
* **Tools**: Selenium, Jenkins for continuous integration and testing.
* **Coverage**: Re-testing the entire system to ensure new updates don't break existing functionality.

## 6.6 Integration Testing

* **Method**: Combination of manual and automated API tests.
* **Tools**: Postman for API testing.
* **Coverage**: Test interactions between different modules (e.g., product catalog, order management, payment gateways, and shipping APIs).

# 7. Test Deliverables

* **Test Plan**: This document outlines the approach and scope.
* **Test Cases**: Detailed manual and automated test cases covering functional, performance, security, and usability tests.
* **Test Scripts**: Automation scripts for functional and regression testing.
* **Test Data**: Data sets for positive and negative testing scenarios (valid/invalid user inputs, various payment options, etc.).
* **Test Reports**: Daily and final test reports summarizing progress, defects, and outcomes.
* **Defect Reports**: Bug tracking through Jira or similar tools, including defect details and status.

# 8. Testing Schedule

| **Phase** | **Start Date** | **End Date** | **Description** |
| --- | --- | --- | --- |
| Test Planning | Oct 16, 2024 | Oct 20, 2024 | Preparation of test strategy, test plan, and test cases. |
| Environment Setup | Oct 20, 2024 | Oct 22, 2024 | Setting up test environment, test data, and tools. |
| Functional Testing | Oct 23, 2024 | Nov 10, 2024 | Manual and automated functional testing. |
| Usability and UI Testing | Nov 11, 2024 | Nov 15, 2024 | Testing user interface, navigation, and responsiveness. |
| Performance Testing | Nov 16, 2024 | Nov 20, 2024 | Load and stress testing for peak traffic scenarios. |
| Security Testing | Nov 21, 2024 | Nov 25, 2024 | Security checks and vulnerability testing. |
| Regression Testing | Continuous | Ongoing | Automated regression testing after each build or release. |
| Final Test Report Submission | Nov 30, 2024 |  | Final testing summary and closure report. |

# 9. Resources

## 9.1 Personnel

* **Test Manager**: Oversees the testing process, ensures deliverables are met.
* **QA Engineers**: Responsible for executing manual and automated tests.
* **Performance Test Engineer**: Focuses on load and performance testing.
* **Security Test Engineer**: Conducts security vulnerability assessments.
* **Developers**: Assist in fixing defects identified during testing.

## 9.2 Tools

* **Automation**: Selenium, Cypress for functional and regression testing.
* **API Testing**: Postman for validating API requests and responses.
* **Performance Testing**: JMeter or LoadRunner.
* **Security Testing**: OWASP ZAP, Burp Suite.
* **Bug Tracking**: Jira or Bugzilla.
* **Cross-Browser Testing**: BrowserStack or Sauce Labs.

# 10. Risk Management

| **Risk** | **Likelihood** | **Impact** | **Mitigation Strategy** |
| --- | --- | --- | --- |
| Payment gateway failures | Medium | High | Early testing of all payment options and services. |
| Performance degradation under load | High | High | Conduct load testing early and optimize server performance. |
| Security vulnerabilities | High | Critical | Regular security testing and code reviews. |
| Cross-browser/device issues | Medium | Medium | Comprehensive cross-browser/device testing using real devices and simulators. |

# 11. Entry and Exit Criteria

**Entry Criteria**

* Test environment is fully set up.
* Functional requirements and design documents are finalized.
* Code has passed unit testing by the development team.

**Exit Criteria**

* All test cases have been executed with a pass rate of at least 95%.
* All critical and high-priority bugs have been resolved.
* Performance and security tests meet the agreed-upon benchmarks.
* No major issues are present in usability and cross-browser tests.

# 12. Test Reporting

* **Daily Reports**: Progress reports showing tests executed, tests passed, and defects logged.
* **Final Test Summary**: A comprehensive report detailing test coverage, test case results, outstanding defects, and risk assessment before go-live.