Graduation Project – Sprint 1 Test Plan

OpenCart E-Commerce Testing Project

Prepared by: Software Testers Team (6 Members)
Date: October 2025
Tools Used: Postman, Selenium (Java), MySQL
Testing Types: Manual, API, Automation, Database

Project Team

1) Youssef Mohamed Mahmoud (QA Lead)
2) Abanoub Nabil Bishay (QA Tester)
3) Omar Saeed Othman (QA Tester)
4) Shrouq Rafaat (QA Tester)
5) Ereny Yacoub (QA Tester)
6) Ahmed Fathy (QA Tester)

Table of Contents

1. Introduction

This document presents the overall **Test Plan** for the **OpenCart E-Commerce Web Application**. It defines the testing scope, objectives, methodologies, responsibilities, and schedule for the QA team during **Sprint 1** of the Graduation Project.
The primary goal of this project is to ensure the **functionality, reliability, and quality** of OpenCart's key features through comprehensive **Manual Testing**, **API Testing**, **Automation Testing**, and **Database Validation**.

URL: https://demo.opencart.com/

2. Scope

2.1 In Scope

The following testing activities are included within the scope of the OpenCart Graduation Project across all sprints:

- **Functional Testing:**
  Verification of major OpenCart functionalities, including User Registration, Login, Add to Cart, Checkout, and Order History.
- **API Testing:**
  Validation of API endpoints related to User Authentication, Product Management, and Order Processing to ensure proper request/response behavior.
- **Automation Testing:**
  Implementation of Regression Test Scenarios using Selenium (Java) to ensure system stability after updates or bug fixes.
- **Database Validation:**
  Testing the backend using MySQL to confirm data integrity, consistency, and proper data flow between the UI and database.
- **Regression Testing:**
  Re-execution of previously designed test cases to ensure that recent changes do not introduce new defects.
- **User Acceptance Testing (UAT):**
  Validation of the website from the end-user perspective to confirm that it meets business and usability expectations.

2.2 Out of Scope

The following items are excluded from the testing activities across all sprints of the OpenCart Graduation Project:

- **Performance Testing:**
  Load, stress, and scalability testing are not included in the scope of this project.
- **Security and Penetration Testing:**
  Vulnerability assessments and penetration tests are outside the current testing scope.
- **Third-Party Payment Gateway Integration:**
  Validation of external payment systems and transaction-level integration is excluded.
- **Mobile Application Testing:**
  Testing on native or hybrid mobile platforms is not covered in this project.
- **Unit Testing:**
  Individual code component testing is not performed as the website is pre-developed for the graduation project.

3. Quality Objectives

The primary objective of testing is to ensure that the OpenCart E-Commerce Website meets all defined functional and non-functional requirements with a high level of quality and reliability throughout the project lifecycle.

The **Software Testers Team** aims to:

- Identify and report defects early to minimize project risks and rework costs across all sprints.
- Ensure that all core website functionalities perform as expected under defined conditions.
- Deliver a stable, user-friendly, and high-performing website at the end of each sprint.
- Maintain **consistent test coverage** through manual, API, automation, and database testing.
- Facilitate **continuous improvement** in testing processes and defect management across all sprints.

4. Test Methodologies

4.1 Overview

Testing activities for this project will be conducted based on the **Agile testing principles**, but adapted to the context of a **graduation project** where the QA team works independently on a pre-developed website.

All testing will focus on validating the existing functionalities of the **OpenCart E-Commerce Website** provided by the project team.
Test cases will be designed from the available **requirements and user flows**, and executed across multiple environments to verify functionality, stability, and reliability.

4.2 Testing Strategy

The testing approach for the **OpenCart E-Commerce Website** will combine **Manual**, **API**, **Automation**, and **Database** testing techniques to ensure full coverage of all functionalities and workflows.

- **Manual Testing:**
  Used for the initial validation of all website features such as registration, login, cart management, and checkout processes. It will help identify UI, usability, and functional issues early.
- **API Testing:**
  Performed using **Postman** to validate the correctness and reliability of backend API endpoints related to user authentication, product management, and order handling.
- **Automation Testing:**
  Regression scenarios will be automated using **Selenium WebDriver (Java)** to verify that existing features remain stable after updates or bug fixes.
- **Database Testing:**
  MySQL will be used to perform backend validation, ensuring data integrity, accuracy, and synchronization between the user interface and database records.

4.3 Bug Triage

All reported defects will be **reviewed, prioritized, and tracked** by the Software testers team based on their **severity** and **impact** on the system.
During bug triage sessions, each issue will be evaluated to determine:

- **Severity:** The technical seriousness of the defect and its effect on system functionality.
- **Priority:** The urgency with which the defect should be fixed based on business or user impact.

All bugs will be documented and managed through a **bug tracking system** (e.g., Excel Sheet) to ensure visibility and proper follow-up until resolution.

4.4 Suspension Criteria and Resumption Requirements

Testing Suspension:
Testing activities will be temporarily suspended if **critical or blocker defects** are found that prevent further test execution.
Examples include:

- Website or environment downtime.
- Inaccessibility of critical modules (e.g., checkout or login).
- Broken test environment configurations or unavailable test data.

Testing Resumption:
Testing will resume once the **blocking issues are resolved**, verified by the Software testers team, and the test environment is confirmed to be stable for continued execution.

4.5 Test Completeness

Testing will be considered **complete** when the following conditions are met:

- All **planned test cases** have been designed, executed, and documented.
- All **critical and major defects** have been fixed, re-tested, and closed successfully.
- The **exit criteria** defined in this plan have been achieved.
- **Regression testing** confirms that recent fixes have not introduced new defects.
- The **Software Testers Team** confirms overall system stability and readiness for release.

5. Roles and Responsibilities

All testing activities in this project will be carried out collectively by the **Software Testers Team**.
Each team member will participate in all testing phases to ensure shared understanding, full coverage, and consistent quality.

- Design and execute **manual test cases** for all website functionalities.
- Develop and maintain **automation scripts** using **Selenium (Java)** for regression scenarios.
- Perform **API testing** using **Postman** to validate backend endpoints.
- Conduct **database validation** in **MySQL** to ensure data integrity.
- Log defects, retest fixes, and report progress in a shared tracking document.
- Collaborate as a team to review results and ensure overall test coverage and quality.

6. Entry and Exit Criteria

6.1 Entry Criteria

Testing will begin only when the following conditions are met:

- All **requirements and user stories** are reviewed and approved.
- The **test environment** is properly configured and ready for execution.
- Required **test data** has been created and validated.
- The **Software Testers Team** has access to all necessary tools, environments, and credentials.

6.2 Exit Criteria

Testing will be considered complete when the following conditions are met:

- All **planned test cases** have been executed and results documented.
- All **major and critical defects** have been fixed, re-tested, and closed successfully.
- A detailed **Test Summary Report** has been prepared and reviewed.
- Final **approval** for test completion has been confirmed by the **Software Testers Team**.

7. Test Deliverables

Sprint 1: Planning

- Test Plan Document
- Project Roadmap
- Defined Entry and Exit Criteria
- Sprint Schedule and Timeline

Sprint 2: Analysis & Design

- Test Scenarios and Test Cases (Manual + Automated)
- Test Data Sets
- Initial BDD Feature Files (Cucumber)
- Jira Dashboard with Test Cases and Defects

Sprint 3: Implementation

- GitHub Repository with Framework and Documentation
- Postman Collections (API Testing)
- SQL Queries for Database Validation
- JMeter Scripts for Performance Testing

Sprint 4: Test Execution & Completion

- Executed Test Results (UI, API, DB, Performance)
- Bug Reports and Fix Verification Logs
- Final Test Summary Report
- Optimized Automation Framework
- CI/CD Pipeline Integration (Jenkins)
- Final Test Completion Report

8. Resources & Environment Needs

8.1 Resources

The Software Testers Team consists of 6 QA Engineers working collaboratively on all testing activities.
Each member is responsible for manual, API, automation, and database testing tasks.
The team will require full access to the OpenCart demo environment and all necessary testing tools.

8.2 Testing Tools

The following tools will be used by the **Software Testers Team** across all sprints:

- **Postman:** For API testing and validation of request/response behavior.
- **Selenium (Java):** For automation of regression test scenarios.
- **MySQL:** For backend database validation and data integrity checks.
- **Excel and trello:** To manage test cases, track defects, and monitor testing progress.
- **GitHub:** For version control, storing automation scripts, and project documentation.

8.3 Configuration Management

 All **test scripts, test cases, and related documents** will be version-controlled using **GitHub**.

 Any modifications or updates to test cases or automation scripts will be tracked to maintain **version history** and ensure **traceability**.

 The **Software Testers Team** will review and approve changes to maintain consistency and prevent conflicts across all sprints.

 This ensures that every update, addition, or fix is documented and can be referenced for regression testing, audits, or future enhancements.

8.4 Test Environment

Testing will be performed using the following environments:

- **Operating System:** Windows 11
- **Web Browsers:** Google Chrome (latest version), Mozilla Firefox (latest version), **Microsoft Edge (latest version)** for cross-browser validation
- **Database Environment:** MySQL database configured to mirror the OpenCart demo data structure for backend validation
- **Automation Environment:** Selenium WebDriver (Java) configured with IDE and integration with GitHub for version control
- **API Testing Environment:** Postman configured with all required endpoints and test credentials
- **Test Data:** Prepared datasets reflecting real-world scenarios for functional, API, and database testing

This environment setup ensures that all manual, automation, API, and database testing activities can be executed consistently across all sprints.

9. Test Schedule

Sprint 1: Planning (Week 1)

- Define project scope and objectives.
- Prepare Test Plan Document.
- Define Entry & Exit Criteria.
- Plan Sprint schedule and timelines.
  Deliverables: Test Plan Document, Project Roadmap, Defined Entry & Exit Criteria, Sprint Schedule & Timeline.

Sprint 2: Analysis & Design (Week 2)

- Conduct exploratory testing to identify key functionalities.
- Design Manual Test Cases.
- Design API Test Cases using Postman.
- Design Database Test Cases using MySQL.
- Plan initial Automation Test Cases using Selenium.
- Prepare BDD Feature Files (Cucumber).
- Track test cases and defects in Excel.
  Deliverables: Initial Manual & Automated Test Cases, Test Data Sets, Initial BDD Feature Files, Test Case Tracking in Excel.

Sprint 3: Implementation (Week 3)

- Setup GitHub repository for version control.
- Implement Automation Framework using Selenium.
- Execute API tests in Postman.
- Prepare SQL Queries for database validation.
- Setup JMeter for performance testing (optional).
  Deliverables: GitHub Repository with framework & documentation, Postman Collections & API results, SQL Queries for Database Validation, Initial JMeter Scripts (optional).

Sprint 4: Test Execution & Completion (Week 4)

- Execute all planned test cases (Manual, API, Automation).
- Perform Database Validation.
- Conduct Regression Testing.
- Log defects, retest fixes, and track resolutions.
- Prepare Test Summary Report.
- Optimize Automation Scripts.
- Integrate CI/CD pipeline using Jenkins (optional).
  Deliverables: Executed Test Results (UI, API, DB, Automation), Bug Reports & Fix Verification Logs, Final Test Summary Report, Optimized Automation Framework, CI/CD Integrated Pipeline, Final Test Completion Report.

10. Risks & Assumptions

Potential Risks:

- **Environment Instability:** Temporary unavailability or misconfiguration of the OpenCart demo environment may delay test execution.
- **API Response Issues:** Unexpected behavior, downtime, or incorrect data from API endpoints may affect testing accuracy.
- **Delayed Requirement Updates:** Late or unclear requirements can impact test case design and execution.
- **Defect Fix Delays:** Delays in resolving critical or major defects may postpone regression testing and final delivery.
- **Resource Constraints:** Limited access to team members or tools may affect testing progress.

Assumptions:

- All testing environments (Web, Database, Automation, API) will be available and properly configured throughout the sprints.
- The Software Testers Team has the necessary access, credentials, and permissions to execute all planned testing activities.
- Test data required for functional, API, and database validation will be prepared in advance.
- All core functionalities of the OpenCart demo website are stable enough to allow meaningful testing.

11. Terms / Acronyms

- **QA – Quality Assurance:** Ensuring the quality of the product through systematic testing and process control.
- **API – Application Programming Interface:** A set of rules and protocols for interacting with backend services.
- **UAT – User Acceptance Testing:** Testing performed from the end-user perspective to validate business and usability requirements.
- **SIT – System Integration Testing:** Testing to verify that different modules and components of the system work together correctly.
- **SQL – Structured Query Language:** A programming language used for managing and querying relational databases.
- **BDD – Behavior Driven Development:** A testing approach using plain language (e.g., Gherkin) to define test scenarios.
- **CI/CD – Continuous Integration / Continuous Deployment:** Automated process for building, testing, and deploying code changes.
- **Jenkins:** A tool for automating CI/CD pipelines.
- **Regression Testing:** Re-executing tests to ensure that recent changes have not broken existing functionality.

12. Approvals

This Test Plan has been reviewed and approved by the following personnel:

1) Youssef Mohamed Mahmoud (QA Lead)
2) Abanoub Nabil Bishay (QA Tester)
3) Omar Saeed Othman (QA Tester)
4) Shrouq Rafaat (QA Tester)
5) Ereny Yacoub (QA Tester)
6) Ahmed Fathy (QA Tester)

Note: All approvals confirm that the Test Plan is reviewed, understood, and agreed upon by the responsible parties.

13. Conclusion

This Test Plan establishes the foundation for **all QA activities across Sprint 1 to Sprint 4** of the OpenCart Graduation Project. It ensures structured planning, design, execution, and reporting aligned with best practices in software testing.

The focus of this plan is to:

- Deliver a **stable and high-quality OpenCart website**.
- Ensure **comprehensive test coverage** across manual, API, automation, and database testing.
- Provide clear **traceability** of test cases, defects, and results across all sprints.
- Support a **collaborative workflow** for the Software Testers Team, ensuring transparency and accountability.
- Facilitate **continuous improvement** through sprint retrospectives and optimization of testing activities.

By following this plan, the team aims to minimize risks, ensure consistency, and prepare the system for successful deployment and future enhancements.