# **Test Plan Document**

**Project Name:** OpenCart E-commerce Web Application  
  
  
**Test Team:** ALX2\_SWD6\_G1 (NHA)  
  
  
**Technical Instructor:** Hussein Ali

**GitHub Repository:** [ALX2\_SWD6\_G1\_GROUP1](https://github.com/alaa20khaled/ALX2_SWD6_G1_GROUP1.git)**Project Website:**<https://demo.opencart.com/>  
  
  
**Job Profile:** Software Tester  
  
  
  
**Team Members:**

| **Name** | **Email** | **Phone No.** |
| --- | --- | --- |
| Alaa Khaled Mohamed Gaber | alaa20khaledd@gmail.com | 01120367824 |
| Fatma ElZhraa Mahmoud Mohammed | fatmaelzhraamahmoud19@gmail.com | 01201479103 |
| Mira Samy Louiz Botros | Mirasamy15@gmail.com | 01229622788 |
| Nour Raafat Elbaz | elbaznour993@gmail.com | 01004129079 |
| Mayar Mohamed Abd Elhalem | mayarmuhammad799@gmail.com | 01286020391 |

## 

## **1. Introduction**

### **1.1 Purpose**

To define a structured test plan that ensures the functionality, performance, usability, and security of the OpenCart web application, using an automated testing framework integrated with CI/CD pipelines.

### **1.2 Scope**

* Validate core functionalities: login, registration, product browsing, search, cart, checkout.
* Cross-browser compatibility: Chrome, Firefox, Edge.
* Performance and security testing.
* Generate automated reports (HTML/PDF) for stakeholders.

### **1.3 Deliverables**

* Test Plan Document
* Automated & Manual Test Cases
* Test Execution Report
* Defect/Bug Report
* Performance & Security Test Report
* CI/CD Pipeline Integration Results

## 

## 

## **2. Test Strategy**

### **2.1 Testing Approach**

* **Black-box Testing**
* **Automation Tools**: Selenium WebDriver with TestNG (Java)
* **API Testing**: REST Assured / Postman
* **Security Testing**: OWASP ZAP (optional)
* **Performance Testing**: Apache JMeter (optional)
* **Manual exploratory testing** for UI/UX

### **2.2 Test Environment**

| **Browser** |  | **Version** |
| --- | --- | --- |
| Chrome |  | 132.0.6834.160 |
| Firefox |  | 132.0 |
| Microsoft Edge |  | Latest Stable |
| Mobile Browser |  | Chrome Android |

**OS:** Windows 10 / Ubuntu  
**CI/CD Tools:** GitHub Actions, Jenkins  
**Reporting:** Allure Reports / ExtentReports

### **2.3 Test Data**

* Valid and invalid test accounts
* Sample products and categories
* Encrypted credentials for security testing

## 

## 

## **3. Test Planning**

### **3.1 Test Schedule**

| **Activity** | **Duration** |
| --- | --- |
| Requirement Analysis | 3 - 4 Days |
| Test Case Design | 4 - 5 Days |
| Script Development | 7 - 10 Days |
| CI/CD Integration | 3 - 4 Days |
| Test Execution | 5 - 6 Days |
| Performance & Security Testing | 3 - 4 Days |
| Bug Fixing & Re-testing | 3 - 5 Days |
| Report Generation & Review | 2 - 3 Days |

**4. Functional Requirements (Test Coverage)**

| **Area** | **Test Cases Include** |
| --- | --- |
| User Authentication | Login/Logout, Registration, Forgotten Password |
| Product Browsing | Category navigation, product listing, filtering, and search |
| Product Details | Images, pricing, description, and stock status |
| Cart Management | Add/remove/update items, view cart |
| Checkout Process | Billing/shipping details, payment simulation, and order confirmation |
| Account Management | View/edit profile, view orders, change password |
| Admin Panel (optional) | Add/edit/delete products, view sales/orders, and user roles |

**5. Non-Functional Requirements**

| **Category** | **Requirement** |
| --- | --- |
| Performance | All test scripts must be completed within 10 minutes for a medium-sized store |
| Scalability | Should support 100+ concurrent test executions |
| Usability | Reports should be understandable by non-technical users |
| Security | Sensitive data should be encrypted; test for vulnerabilities |
| Compatibility | Works across desktop and mobile browsers |

## **6. Test Execution**

### **6.1 Procedure**

* Develop test cases based on functional requirements
* Automate key flows using Selenium
* Execute test cases manually and through the CI/CD pipeline
* Log defects and generate detailed reports

## **7. Defect Tracking**

* Tool: Excel sheet / GitHub Issues / Jira
* Fields: ID, Title, Description, Severity, Priority, Status, Assignee, Resolution Date

## **8. Test Reporting**

### **8.1 Summary Metrics**

| **Metric** | **Count** |
| --- | --- |
| Total Test Cases | 30 |
| Executed Test Cases | 30 |
| Passed Test Cases | 25 |
| Failed Test Cases | 5 |
| Bugs Identified | 5 |

**8.2 Bug Severity Breakdown**

| **Severity** | **Count** |
| --- | --- |
| High | 2 |
| Medium | 1 |
| Low | 2 |

## **9. Risk Management**

### **9.1 Risk Identification**

* Performance Risks: Response delays under load
* Security Risks: Data breaches, SQL injection, XSS
* Compatibility Risks: Inconsistent rendering on browsers/devices
* Project Risks: Schedule delays, miscommunication

### **9.2 Risk Assessment & Mitigation**

| **Risk** | **Severity** | **Mitigation Strategy** |
| --- | --- | --- |
| Performance | Medium | Use JMeter to simulate load |
| Security | Medium | Integrate OWASP ZAP for vulnerability scanning |
| Compatibility | Medium | Test across multiple browsers/devices |
| Schedule | Critical | Follow Agile sprints, assign tasks per module |
| Communication | Critical | Weekly check-ins, shared documentation (Google Drive, GitHub, Trello) |

### 

### **9.3 Risk Monitoring**

* Risks reassessed bi-weekly; mitigation plans adjusted accordingly.