

Test Strategy Document

Create by sakil ahmed

Table of Contents

- 1. Objective**
- 2. Scope of Testing**
- 3. Testing Types**
 - 3.1 Functional Testing
 - 3.2 Usability Testing
 - 3.3 Performance Testing
 - 3.4 Security Testing
 - 3.5 Regression Testing
 - 3.6 Compatibility Testing
- 4. Test Approach**
 - 4.1 Smoke Testing
 - 4.2 Depth Testing
 - 4.3 Automated Testing
- 5. Entry and Exit Criteria**
 - 5.1 Entry Criteria
 - 5.2 Exit Criteria
- 6. Test Deliverables**
- 7. Risk Analysis**
- 8. Tools and Resources**

1. Objective

The testing strategy ensures that the Academy LMS Plugin is robust, reliable, secure, and user-friendly. The goal is to validate that it meets functional and non-functional requirements across various platforms and devices.

2. Scope of Testing

The testing process will cover:

- **Functional Testing:** Ensures all features like registration, course creation, and payment integration work as expected.
 - **Non-Functional Testing:** Validates performance, compatibility, and security aspects.
 - **Regression Testing:** Ensures no existing functionality is broken after updates.
 - **Usability Testing:** Verifies that the plugin provides an intuitive user experience.
-

3. Testing Types (In Detail)

3.1 Functional Testing

- Validate workflows for registration, login, course creation, and purchase.
- Verify quiz creation and automated grading.
- Test course tracking and progress reporting for accuracy.

3.2 Usability Testing

- Evaluate navigation ease and UI design consistency.

- Test responsiveness on different screen sizes.
- Verify accessibility for users with disabilities.

3.3 Performance Testing

- Use **JMeter** to test response times and stability under varying loads (e.g., 100, 500, 1000 users).
- Identify and fix bottlenecks during peak usage.

3.4 Security Testing

- Test secure login/logout mechanisms.
- Validate encryption for sensitive data (e.g., passwords, payments).
- Ensure proper implementation of HTTPS protocol.

3.5 Regression Testing

- Execute previously tested scenarios to verify that updates haven't introduced new defects.

3.6 Compatibility Testing

- Test on multiple browsers (Chrome, Firefox, Edge, Safari).
 - Validate functionality on various devices (desktop, tablet, mobile).
-

4. Test Approach

4.1 Smoke Testing

- Perform initial testing to ensure the plugin is stable and ready for detailed testing.

4.2 Depth Testing

- Conduct in-depth testing of individual modules after passing smoke tests.

4.3 Automated Testing

- Automate repetitive tasks such as login, course creation, and basic functionality checks using Selenium.
-

5. Entry and Exit Criteria

5.1 Entry Criteria

- Test environment is set up and verified.
- Test data is prepared and validated.
- Approved test plan and test cases are in place.

5.2 Exit Criteria

- All test cases are executed with a pass rate of 95% or higher.
 - All critical and high-priority defects are resolved.
 - Test reports are reviewed and signed off by stakeholders.
-

6. Test Deliverables

- **Test Plan:** High-level testing strategy and objectives.
 - **Test Cases:** Detailed test cases for all scenarios.
 - **Defect Reports:** List of identified bugs with detailed reproduction steps.
 - **Test Summary Report:** Consolidated results of all test cycles.
-

7. Risk Analysis

Risk	Impact	Mitigation
Incompatibility with WordPress themes	Could break the site or UI.	Test the plugin on multiple WordPress themes.
Performance issues during high loads	Sluggish user experience.	Conduct load and stress testing.
Security vulnerabilities	Could lead to data breaches.	Perform regular security assessments.

8. Tools and Resources

- **Bug Tracking:** Jira
- **Automation Testing:** Selenium
- **Performance Testing:** JMeter
- **Browser Compatibility:** BrowserStack
- **Documentation:** MS Word, Excel