

## **EX-5**

### **Test Plan Documentation for Simple Library Management System**

(As per IEEE 829-1998 standard format)

**AIM:- To prepare a test document for simple Library Management System**

#### **1. Test Plan Identifier**

SLMS\_TestPlan\_2024\_v1.0

#### **2. Introduction**

##### **2.1 Purpose**

The purpose of this test plan is to outline the testing strategy, objectives, resources, schedule, and deliverables for the Simple Library Management System (SLMS). This document ensures that the system meets its functional and non-functional requirements before deployment.

##### **2.2 Scope**

This test plan applies to all features and functionalities of the SLMS, including User Management, Book Management, Lending System, Report Generation, and Notifications. The focus will be on verifying that the system operates according to the specified requirements.

##### **2.3 References**

- SLMS System Requirements Specification (SRS)
- SLMS User Manual
- Previous Test Reports (if any)

##### **2.4 Definitions/Acronyms/Abbreviations**

- SLMS: Simple Library Management System
- UAT: User Acceptance Testing
- QA: Quality Assurance
- SRS: System Requirements Specification

#### **3. Test Items**

##### **3.1 Items to be Tested**

- **User Management:** Includes user registration, login, and profile management.
- **Book Management:** Adding, updating, deleting book records, and searching for books.
- **Lending System:** Issuing and returning books, tracking due dates.
- **Report Generation:** Creating reports on book inventory, user activity, and lending history.

- **Notifications:** Alerts for overdue books and upcoming due dates.

### 3.2 Items Not to be Tested

- Integration with external systems (if beyond scope).
- Performance and load testing (if not within scope).
- Security penetration testing (unless specified).

## 4. Features to be Tested

This section lists the specific features of SLMS to be tested:

### 4.1 User Management

- User registration and profile management.
- Login and authentication process.

### 4.2 Book Management

- Adding new books to the inventory.
- Searching for books by title, author, or category.
- Updating and deleting book records.

### 4.3 Lending System

- Issuing books to users.
- Managing book returns and updating inventory.
- Tracking due dates and overdue books.

### 4.4 Report Generation

- Generating reports on current book inventory.
- Generating user activity reports.

### 4.5 Notifications

- System-generated alerts for overdue books.
- Notifications for upcoming book return dates.

## 5. Features Not to be Tested

This section lists the features of SLMS that are outside the testing scope:

- **Integration Testing:** Connections with external systems are not within the current testing scope.
- **Performance Testing:** Load and stress testing is excluded unless specified.
- **Security Testing:** Penetration testing and security assessments are beyond the current scope.

## 6. Approach

## 6.1. Testing Levels

**6.1.1 Unit Testing** :Performed by developers on individual components to ensure each unit functions as expected.

**6.1.2 Integration Testing** : Testing the integration of different modules and components within SLMS to ensure they work together seamlessly.

**6.1.3 System Testing** : Comprehensive testing of the entire SLMS in a simulated production environment to ensure all components function together correctly.

**6.1.4 User Acceptance Testing (UAT)** : End-user testing to validate that the system meets the user's requirements and expectations.

## 6.2 Testing Types

**6.2.1 Functional Testing** :Testing all functionalities of SLMS to ensure they operate according to the specifications.

**6.2.2 Usability Testing** :Ensuring the system is intuitive and user-friendly.

**6.2.3 Regression Testing** :Re-testing of the system after changes to ensure no existing functionalities are broken.

**6.2.4 Automation Testing** : Identifying and automating critical test cases for efficient regression testing.

## 7. Pass/Fail Criteria

**7.1 Entry Criteria** : All necessary resources (test environment, test data) are available.

The system is fully developed and ready for testing.

**7.2 Exit Criteria :** All test cases have been executed, and no high-priority defects remain unresolved.

- UAT has been successfully completed and signed off by the end-users.

**7.3 Suspension Criteria :**Testing will be suspended if critical defects are found that prevent further testing.

**7.4 Resumption Criteria :**Testing will resume once critical defects are resolved and validated.

## **8. Test Deliverables**

### **8.1 Deliverables**

- Test Plan Document
- Test Cases Document
- Test Scripts (for automation)
- Test Data
- Test Summary Reports
- Defect Reports

### **8.2 Non-Deliverables**

- Source Code
- Deployment Scripts

## **9. Testing Tasks**

### **9.1 Task Identification**

- Develop test cases based on the SRS.
- Prepare test data and environment.
- Execute test cases and log defects.
- Automate critical test cases.
- Prepare test summary and defect reports.

### **9.2 Task Assignment**

- Test Case Design: QA Team
- Test Execution: QA Team
- Test Automation: QA Team
- UAT Coordination: Project Manager
- Defect Reporting: QA Lead

## 10. Environmental Needs

### 10.1 Hardware Requirements

- Servers: [List specifications]
- Client Machines: [List specifications]

### 10.2 Software Requirements

- Operating System: Windows/Linux/macOS
- Database: MySQL/PostgreSQL/SQLite
- Browsers: Chrome, Firefox, Safari (for web-based systems)
- Testing Tools: Selenium (for automation), JIRA (for defect tracking)

### 10.3 Test Data Requirements

- Dummy data sets for user profiles, books, and lending transactions to simulate real-world scenarios.

## 11. Responsibilities

### 11.1 Test Team

- QA Lead: [Name] - Responsible for overseeing the testing process.
- Testers: [Names] - Responsible for executing test cases and logging defects.
- Test Automation Engineer: [Name] - Responsible for creating and maintaining automated test scripts.

**11.2.1 Management** :Project Manager: [Name] - Responsible for overall project delivery and coordinating UAT.

**11.3 Client Representatives** :Client Representative: [Name] - Responsible for providing requirements and sign-off on UAT.

## 12. Staffing and Training Needs

- Adequate number of test engineers and developers.
- Training sessions on the LMS and testing tools for the testing team.

## 13. Schedule

Activity	Start Date	End Date
Test planning		
Test Case Design		
Test Environment Setup		
UAT		
Test Report Preparation		

## 14. Risks and Contingencies

### 14.1 Risks

- Potential delay in setting up the test environment.
- Limited availability of resources for testing.
- Unclear or changing requirements during the testing phase.

### 14.2 Contingencies

- Schedule buffer time to accommodate potential delays.
- Ensure clear communication channels with stakeholders for quick resolution of issues.
- Implement a change management process to handle requirement changes.

## 15. Approvals

Name	Role	Signature	Date
[Test Lead Name]	Test Lead		
[Project Manager Name]	Project manger		
[Client Representative]	client		

**Result:** - Test plan document prepared for a simple library management system as per the IEEE standard format