**Test Plan Document**

**1. Introduction**

**1.1 Background**

This test plan outlines the approach and strategy for automating tests on the RFID-based Attendance and Payment System. The goal is to ensure functional stability, data integrity, and user experience across key features such as user management, RFID attendance simulation, transaction handling, and wallet management.  
**1.2 Testing Objectives**

The primary objectives of the testing process are:

 Validate core functionalities: Admin Login, CRUD operations on users, RFID scan simulation, wallet funding, transaction recording, and dashboard analytics.

 Detect and log any system defects via automated and manual tests.

 Ensure high confidence in the system’s reliability and correctness before delivery.

**1.3 Document Audience**

This Test Plan is intended for:

 QA Engineers

 Project Supervisors

 Backend/Frontend Developers

 University Evaluation Committee

**1.4 References**

* Test Strategy Document
* Test Case Template
* Defect Triage Matrix
* Bug Report Template

**2. Testable Items**

**2.1 In Scope**

The following functionalities of the Amazon website are in scope:

 Admin Login (authentication)

 User CRUD operations (create, edit, view, delete)

 RFID scan simulation (attendance tracking)

 Wallet system (create, fund, deduct)

 Transactions and purchase recording

 Role-based dashboards (admin, student, teacher, parent)

 Dashboard charts and statistics (spending, attendance, lectures)

**2.2 Out of Scope**

 Performance and load testing on the deployment server

 Non-critical UI enhancements

**3. Test Approach**

**Types:**

* Functional Testing
* Regression Testing
* Data-Driven Testing (user data, RFID events)

**Tools**:

* Katalon Studio (for automation)
* Selenium WebDriver (integrated)
* Excel (test data, defect log)

**Test Execution**:

* Katalon test cases will simulate login and CRUD flows.
* Selenium WebDriver will handle dropdowns, modals, and dynamic components.

**4. Test Environment**

| **Component** | **Details** |
| --- | --- |
| Operating System | Windows 10/11 |
| Browsers | Chrome |
| Test Server | http://localhost:3000 |
| Automation Tool | Katalon Studio, Selenium Driver |
| DB | MongoDB (local instance) |

**5. Test Schedule**

| **Phase** | **Activity** | **Timeline** |
| --- | --- | --- |
| Phase 1 | Setup and configuration |  |
| Phase 2 | Test Case Design and Object Repository |  |
| Phase 3 | Scripting Katalon Test Cases |  |
| Phase 4 | Test Execution & Bug Reporting |  |
| Phase 5 | Test Summary Report and Documentation |  |

**6. Resources**

| **Role** | **Responsibility** |
| --- | --- |
| QA Engineer | Develop and execute Katalon scripts |
| QA Lead | Review scripts and test results |
| Defect Manager | Track, log, and prioritize defects |
| Documentation Specialist | Maintain test artifacts and reports |
| Developer | Fix bugs discovered during testing |
| Supervisor | Review and approve testing outcomes |
|  |  |

**7. Pass/Fail Criteria**

* **Pass**: Test case executes successfully, and expected result matches’ actual result.
* **Fail**: Test case does not produce the expected outcome or has defects.
* All critical and high-severity defects must be fixed before sign-off.

**8. Risks and Mitigation**

| **Risk** | **Mitigation Plan** |
| --- | --- |
| Dynamic content causing test failures | Use dynamic locators and robust waits |
| Limited access to specific features | Focus on accessible stable workflows |
| Browser compatibility issues | Execute tests across multiple browsers |
| Modal layers interfere with clicks | Ensure modal focus or switch context |
| UI timing causes flaky test results | Add robust waitUntilVisible/sleep |

**9. Deliverables**

The following deliverables will be provided:

1. Test Cases (Katalon)
2. Automated Test Scripts (Katalon Studio)
3. Defect Reports (Excel)
4. Test Execution Reports
5. Test Summary Report

**10. Approval**

| **Name** | **Role** | **Signature** | **Date** |
| --- | --- | --- | --- |
|  | Project Manager |  |  |
|  | QA Lead |  |  |
|  | Automation Tester |  |  |