**TEST STRATEGY DOCUMENT**

**1. Document ID**

TS-001

**2. Project Name**

Automated Expense Tracker

**3. Test Strategy Overview**

The test strategy defines the overall approach and methodology for testing the Automated Expense Tracker application, ensuring quality and reliability.

**4. Testing Scope**

**In-Scope:**

* Functional testing of all application modules.
* User authentication and authorization testing.
* Data processing and validation.
* API testing for mock transaction synchronization.
* UI and UX usability testing.
* Security and access control verification.
* Regression testing to ensure stable releases.

**Out-of-Scope:**

* Integration with live banking APIs.
* Performance testing beyond expected load.
* Mobile application testing (only web application covered).

**5. Testing Approach**

* **Manual Testing:** Primary functional validation using predefined test cases.
* **Automated Testing:** Automating regression and repetitive test cases.
* **Black-Box Testing:** Validating system behavior without accessing code.
* **Risk-Based Testing:** Prioritizing test cases based on risk assessment.
* **Regression Testing:** Ensuring new changes do not break existing functionality.

**6. Test Levels**

1. **Component Testing:** Individual module testing.
2. **Integration Testing:** Verifying interactions between integrated modules.
3. **System Testing:** End-to-end validation of system behavior.
4. **User Acceptance Testing (UAT):** Final validation by business stakeholders.

**7. Test Environment**

* **OS:** Windows/Linux
* **Browser:** Chrome, Firefox, Edge
* **Database:** SQLite/PostgreSQL
* **Tools:** Selenium (automation), Postman (API testing)

**8. Test Data Management**

* **Synthetic Test Data:** Used for transaction validation.
* **Mock APIs:** Simulated data sources for testing integrations.
* **Data Privacy Considerations:** No real customer data will be used.

**9. Defect Management**

1. Defects logged in the defect tracking tool (e.g., JIRA).
2. Defects prioritized based on severity and impact.
3. Fixes verified in regression cycles.
4. Defects closed upon successful validation.

**10. Entry & Exit Criteria**

**Entry Criteria:**

* Requirements finalized.
* Test environment setup is complete.
* Test cases are approved.

**Exit Criteria:**

* All critical test cases executed.
* Major defects resolved.
* Test summary report submitted.

**11. Risk & Mitigation Plan**

| **Risk** | **Mitigation Strategy** |
| --- | --- |
| Unclear requirements | Conduct requirement clarification sessions. |
| Environment unavailability | Maintain backup test environments. |
| High defect leakage | Implement rigorous regression cycles. |

**12. Test Metrics & Reporting**

* **Test Execution Progress:** (Executed Test Cases / Total Test Cases) \* 100
* **Defect Density:** (Defects Found / Test Cases Executed)
* **Test Coverage:** (Executed Test Cases / Total Requirements) \* 100
* **Pass/Fail Rate:** (Passed vs. Failed Test Cases)

**13. Approval & Sign-off**

**Approval By:**

* QA Lead: \_\_\_\_\_\_\_\_\_\_\_ (Signature & Date)
* Project Manager: \_\_\_\_\_\_\_\_\_\_\_ (Signature & Date)
* Product Owner: \_\_\_\_\_\_\_\_\_\_\_ (Signature & Date)

**End of Test Strategy Document**