Integration Testing Strategy Document for SaaS Platforms

1. Objective

Define the approach, scope, and processes for integration testing of the SaaS Platform to ensure seamless interaction among modules, third-party APIs, and infrastructure components.

2. Scope

- Core platform modules integration
- Third-party service and API integrations
- Data flow, authentication, and error handling across services
- Multi-tenant scenarios and environment configurations

3. Test Approach

1. Identify integration points and data flows
2. Design integration test scenarios for end-to-end workflows
3. Automate integration test suites where applicable
4. Set up test data and stubs/mocks for external dependencies
5. Conduct regression and exploratory integration tests

4. Roles & Responsibilities

Role	Responsibility
QA Lead	Owns integration test planning and reporting
Developers	Develop and maintain integration test cases and stubs
DevOps	Maintain integration test environments

5. Test Execution

- Schedule integration cycles for every deployment to staging
- Track test cases, results, and defects in the test management tool
- Ensure timely triage and retesting of integration issues

6. Tools

- API Testing: Postman, Insomnia
- Test Automation: Cypress, Selenium
- CI/CD: Jenkins, GitHub Actions

7. Metrics & Reporting

- Number of integration test cases executed
- Integration defects found and fixed per cycle
- Automation coverage for integration flows

8. Risks & Mitigations

- **Unstable external APIs:** Use mocks/stubs or sandbox environments.
- **Complex multi-tenant flows:** Include tenant-aware test scenarios.
- **Non-reproducible environments:** Enforce Infrastructure-as-Code for test setups.

9. Sign-Off

Integration testing is completed when all critical test cases pass and major defects are resolved.