# **Test Plan Document for MintQA.com**

## **1. Test Plan for MintQA.com**

### **1.1 Introduction**

This test plan defines the testing scope, approach, resources, and schedule for MintQA.com, a SaaS platform. The focus is on API, Mobile, UI testing with additional emphasis on compliance, performance, and accessibility.

### **1.2 Objectives**

* Ensure MintQA.com meets functional and non-functional requirements.
* Validate compliance with WCAG and ADA accessibility standards.
* Verify database integrity and security.
* Conduct regression, unit, performance, beta, alpha, and white-box testing.
* Leverage AI tools where applicable for enhanced testing efficiency.

### **1.3 Scope**

* Functional testing of APIs, Mobile apps, and UI components.
* Accessibility testing (WCAG, ADA).
* Database testing for data accuracy and security.
* Performance testing under load and stress conditions.
* User accessibility and usability testing.
* Defect management via Jira; Test case management via TestRail.

### **1.4 Testing Types**

|  |  |  |
| --- | --- | --- |
| **Testing Type** | **Description** | **Tools/Notes** |
| API Testing | Validate endpoints, data responses | Postman, REST Assured |
| Mobile Testing | Functional and UI testing on mobile devices | Appium, BrowserStack |
| UI Testing | Cross-browser and device UI validation | Selenium, Cypress |
| Accessibility Testing | WCAG 2.1 and ADA compliance checks | Axe, WAVE |
| Database Testing | Data validation, schema testing | SQL Queries, DB monitoring tools |
| Regression Testing | Re-test after fixes/updates | Automated via TestRail scripts |
| Unit Testing | Code-level testing by developers | JUnit, NUnit |
| Performance Testing | Load, stress testing | JMeter, LoadRunner |
| Beta & Alpha Testing | Pre-release feedback cycles | Internal & external user groups |
| White-box Testing | Code coverage and path testing | Static code analysis tools |
| AI-assisted Testing | Use GenAI or similar for test generation/review | Optional, experimental |

### **1.5 Roles and Responsibilities**

|  |  |
| --- | --- |
| **Role** | **Responsibility** |
| QA Manager | Oversee test strategy and progress |
| Test Lead | Prepare test plans, coordinate testing |
| Test Engineers | Execute tests, report defects |
| Developers | Unit testing, fix defects |
| Accessibility Specialist | WCAG/ADA compliance validation |

### **1.6 Schedule**

|  |  |
| --- | --- |
| **Phase** | **Dates** |
| Test Planning | YYYY-MM-DD to YYYY-MM-DD |
| Test Case Design | YYYY-MM-DD to YYYY-MM-DD |
| Test Execution | YYYY-MM-DD to YYYY-MM-DD |
| Regression Testing | YYYY-MM-DD to YYYY-MM-DD |
| Beta Testing | YYYY-MM-DD to YYYY-MM-DD |
| Test Closure | YYYY-MM-DD |

### **1.7 Environment**

* Cloud-based test environments replicating production.
* Devices: iOS and Android mobiles, multiple browsers.
* Databases: Production-like database instances.

### **1.8 Risk & Mitigation**

|  |  |
| --- | --- |
| **Risk** | **Mitigation** |
| Delays in environment setup | Early provisioning, backup environments |
| Non-compliance with accessibility | Continuous monitoring, expert reviews |
| Defects backlog buildup | Daily defect triage, prioritization |
| AI tools integration challenges | Pilot projects, manual fallback |

### **1.9 Defect Management**

* Jira for defect tracking, prioritization, and resolution workflows.
* Defects categorized by severity and status updated regularly.

### **1.10 Test Deliverables**

* Test Plan Document
* Test Cases and Scripts (TestRail)
* Defect Reports (Jira)
* Test Summary Reports