Test Strategy Document

Product: apo.vwo.com (A/B Testing Tool)

Version: 1.1 (Final)

Prepared By: Senior QA Engineer – Bhagya Lakshmi

Reviewed By: QA Lead (20+ Years Experience)

Date: 08th May 2025

1. Objective

To ensure the successful release of apo.vwo.com by verifying its functional, performance, security, usability, and compatibility requirements through a thorough testing strategy that covers key user workflows including login, dashboard navigation, sign-up creation, and A/B test configuration.

2. Scope of Testing

☑ In-Scope

- Core Functional Workflows:
  - User login/logout
  - User sign-up creation (admin/visitor roles)
  - A/B test creation, configuration, targeting, execution, and result viewing
  - Dashboard widgets and navigation
  - Account settings: profile, password, notifications

- Platform Coverage:
  - Web (desktop and responsive mobile browser)

- Integrations:
  - Google Analytics (event tracking)
  - Webhooks
  - Email alerts/notifications

❌ Out-of-Scope

- Native mobile apps (iOS/Android)
- Marketing site, blog, or landing pages

- Internal admin dashboards (non-customer facing)

3. Testing Focus Areas

Area	Goals
Functional	Validate each user-facing feature for correctness
UI/UX	Ensure layout, responsiveness, and navigation flow
Performance	Validate system behavior under load and stress
Security	Protect against vulnerabilities (OWASP Top 10)
Compatibility	Cross-browser and cross-device consistency
Usability	Accessibility and intuitive user experience

4. Test Approach

Testing Types:
- **Black Box Testing**: UI and business logic testing
- **White Box Testing**: Backend validations, if accessible
- **Exploratory Testing**: Rapid testing to identify edge case issues
- **Automation Testing**: Selenium for regression
- **Performance Testing**: JMeter with custom load plans
- **Security Testing**: OWASP ZAP and manual review
- **Cross-Browser Testing**: Chrome, Firefox, Safari, Edge (latest 2 versions)
- **Accessibility Testing**: WCAG 2.1 Level AA compliance

Testing Techniques:
- **Risk-Based Testing**
- **Boundary Value Analysis**
- **Equivalence Partitioning**
- **Session-Based Exploratory Testing**

5. Tools & Environments

Category	Tools
UI Automation	Selenium WebDriver + TestNG
API Testing	Postman, REST Assured (optional)
Performance	Apache JMeter
Security	OWASP ZAP
Test Management	Jira + Xray/TestRail

CI/CD	Jenkins, GitHub Actions
Cross-Browser Grid	BrowserStack / LambdaTest
Documentation	Confluence, Google Docs

6. Deliverables

- Finalized Test Strategy Document
- Functional & Regression Test Cases
- Automated Regression Suite
- Performance Benchmark Report
- Security Audit Report
- Defect Summary and Triage Log
- UAT Report with Sign-off
- Test Coverage Metrics
- Release Readiness Report

7. Team & Schedule

Test Team Structure:
- 2 Manual Test Engineers
- 2 Automation Engineers
- 1 Performance Test Engineer (shared)
- 1 Security Analyst (shared)
- 1 UAT Coordinator

Tentative Timeline:
Phase	Duration	Activities
Week 1–2	Functional, API, and Security Testing
Week 3	Performance and Load Testing
Week 4	Cross-Browser Testing
Week 5	Usability Testing and UAT
Week 6	Regression & Final Review

8. Entry & Exit Criteria

Entry Criteria:
- All user stories reviewed and signed off for QA
- Test environment deployed and stable
- Test data available or scriptable

- APIs fully integrated
- Access provided to required tools

Exit Criteria:
- 100% test case execution completed
- All critical/high-priority bugs fixed and retested
- >95% regression suite pass rate
- No high/critical security issues open
- Final sign-off from QA and business stakeholders

9. Key QA Metrics

Metric	Target
Functional Test Coverage	≥ 95%
Automation Coverage	≥ 80% (Regression)
Defect Leakage	≤ 2% post-release
Performance SLA	< 2s average response under load
UAT Satisfaction	≥ 90% approval rating
Accessibility Compliance	WCAG 2.1 AA

10. Risk Management

Risk	Mitigation
Feature creep during sprint	Enforce freeze dates and backlog grooming
Environment instability	Use Docker-based snapshots or cloud environments
Incomplete integration points	Use mocks and stubs for early validation
Defect reopens due to unclear requirements	Align QA with product for definition reviews
Delayed UAT feedback	Schedule dedicated UAT windows in advance

11. Communication Plan

- **Daily QA Standups** (with Dev/QA/Product)
- **Weekly Defect Triage Meetings**
- **Bi-weekly Sprint Demos & Retrospectives**
- **UAT Review Sessions with Business Users**
- **Final QA Handover Presentation** to Stakeholders

☑ Sign-Off

Role	Name	Signature	Date
QA Lead			
Product Owner			
Engineering Lead			