

Test Plan for app.vwo.com

Created by: Aniket Mishra

1. Objective	1
2. Scope	2
3. Inclusions	3
4. Exclusions	3
5. Test Environments	3
6. Defect Reporting Procedure	3
7. Test Strategy	4
8. Test Schedule	5
9. Test Deliverables	5
10. Entry and Exit Criteria	6
11. Tools	6
12. Risks and Mitigation	6
13. Requirements Traceability Matrix (RTM)	7
14. Automation Strategy	7
15. Security and Compliance Testing	7
16. Accessibility Testing	7
17. Metrics	1
18. Approvals	1

1. Objective

This test plan outlines the quality assurance approach for the A/B testing platform app.vwo.com. The primary goal is to validate critical functionality including login, dashboard, user management, and A/B campaign creation. The objective is to ensure all features meet functional, performance, usability, security, and accessibility expectations, enabling end-users to optimize their website conversions effectively.

2. Scope

Features to be Tested:

- Login Page
- Dashboard Page
- Create Account Page

- A/B Testing Campaign Management

Testing Types:

- Manual Testing
- Automated Testing
- Performance Testing
- Accessibility Testing
- Security Testing

Platforms and Environments:

- Browsers: Chrome, Firefox, Edge
- OS: Windows 10, macOS Monterey, Ubuntu 22.04
- Devices: Desktop, Laptop, Tablet, Smartphone

Evaluation Criteria:

- Defect Density
- Defect Leakage
- Test Case Coverage
- Automation Coverage
- Performance Benchmarks
- User Feedback Ratings

Team Roles:

- Test Lead: Strategy, planning, approvals
- Testers: Test case creation, execution, defect logging
- Developers: Defect resolution

- Product Owner: Review & business feedback
 - Automation Engineer: CI/CD setup and maintenance
-

3. Inclusions

- End-to-end validation of all core and peripheral features
- Integration points with backend services
- Validation against business KPIs and requirements

4. Exclusions

- Third-party integration performance metrics
 - Mobile-native app (if separate)
 - Deep analytics of visitor segmentation algorithms
-

5. Test Environments

- **Operating Systems:** Windows 10, macOS, Ubuntu 22.04
 - **Browsers:** Chrome, Firefox, Edge (latest stable versions)
 - **Devices:** Desktops, Laptops, Smartphones (responsive testing)
 - **Network Conditions:** Wi-Fi, 4G, simulated slow networks
 - **Security Protocols:** SSL validation, token expiry tests
 - **Access Levels:** Admin, Standard User, Viewer
 - **URL Configuration:** QA, Staging, and Production URLs
-

6. Defect Reporting Procedure

- **Criteria:** Any deviation from expected behavior, security flaws, or UX failures
 - **Steps:** Log in JIRA with reproducible steps, screenshots, logs
 - **Triage:** Defect severity, priority tagging by QA Lead
 - **Tools:** JIRA, Slack, Email, Excel Reports
 - **Metrics:** Defect count, severity distribution, MTTR, defect rejection ratio
-

7. Test Strategy

Step 1: Test Case Design

- Techniques: Equivalence Partitioning, BVA, Use Case Testing, Decision Tables, State Transition
- Additional: Exploratory Testing, Error Guessing
- Include positive, negative, edge cases

Step 2: Testing Approach

- Smoke Testing → Functional Testing → Regression → Sanity → Exploratory
- Environment-based parallel execution
- Defect logging and retesting on daily cycles

Step 3: Best Practices

- Shift Left Testing
- End-to-End Scenario Testing
- Automation for regression
- Continuous Feedback Loop

Negative Testing Areas:

- Invalid login credentials

- Campaigns with missing variants
 - Improper access permissions
-

8. Test Schedule

Task	Duration
Test Plan Preparation	1 Day
Test Case Design	2 Days
Environment Setup	0.5 Day
Test Execution	3 Days
Bug Retesting & Regression	2 Days
Final Report Preparation	1 Day

9. Test Deliverables

- Approved Test Plan
 - RTM Document
 - Test Case Documents
 - Test Execution Report
 - Defect Logs
 - Final Test Summary Report
-

10. Entry and Exit Criteria

Requirement Analysis

- Entry: Reviewed and signed-off requirements
- Exit: All queries resolved, testable artifacts ready

Test Execution

- Entry: Signed-off test cases, stable build deployed
- Exit: All high/critical test cases executed, Sev-1/2 bugs resolved

Test Closure

- Entry: Final test reports and defect logs ready
 - Exit: Stakeholder sign-off, defect metrics submitted
-

11. Tools

- JIRA – Defect Management
 - TestRail / Excel – Test Case Management
 - Postman – API Testing
 - Selenium / Playwright – Automation
 - Lighthouse / Axe – Accessibility
 - XMind – Mind Mapping
 - Slack / Email – Communication
-

12. Risks and Mitigation

Risk	Likelihood	Impact	Mitigation
Test Environment not stable	High	High	Pre-smoke test, regular check-ins
Sudden resource unavailability	Medium	Medium	Cross-training backups
Requirement change mid-cycle	High	High	Change control + RTM traceability

13. Requirements Traceability Matrix (RTM)

Requirement ID	Description	Test Case ID(s)
REQ-001	User Login	TC-001, TC-002
REQ-002	Dashboard Load	TC-010
REQ-003	Create A/B Campaign	TC-020, TC-021, TC-022

14. Automation Strategy

- **Scope:** Smoke tests, Regression pack, API tests
 - **Tools:** Selenium, Playwright, Postman + Newman
 - **CI/CD:** Jenkins/GitHub Actions pipeline
 - **Approach:** Modular, versioned automation with tagging (e.g., @smoke, @regression)
-

15. Security and Compliance Testing

- Validate JWT/session expiration
 - RBAC testing: Role manipulation, URL tampering
 - Cookie attributes: HttpOnly, Secure, SameSite
 - OWASP Top 10: Basic checklist (SQLi, XSS, CSRF)
-

16. Accessibility Testing

- **Standard:** WCAG 2.1 Level AA
 - **Tools:** Axe, Lighthouse, NVDA
 - **Areas Covered:** Keyboard navigation, color contrast, screen reader support
-

17. Metrics

- Test Case Execution %
 - Defect Density = Total Defects / KLOC
 - Defect Leakage Rate = Defects missed in QA but found in UAT/Production
 - Test Coverage = (Executed / Planned) * 100
 - Automation Coverage = % of critical flows automated
 - MTTD & MTTR
-

18. Approvals

Document	Approved By
Test Plan	Test Lead, PM
Test Cases	QA Lead
Test Summary Report	QA Lead, PO
