

# Test Plan for [www.xyz.com](http://www.xyz.com)

**Created by:** Pushkar Kumar

**Reviewed and Refined by:** Senior QA

**Version:** 1.2

**Date:** [Insert Date]

---

## 1. Objective

This test plan outlines the strategy and scope for validating the **XYZ project** ([www.xyz.com](http://www.xyz.com)) — a web-based platform. The focus is to ensure the platform performs as expected from both technical and user perspectives across supported environments through structured and exploratory testing methods.

---

## 2. Scope

### In-Scope Testing Areas:

- **Login Page:** Authentication, security, and session handling
- **Dashboard Page:** Data presentation, performance, and user navigation
- **Create Account Page:** Form validation, email handling, and user creation workflows
- **Core Features and Workflows:** Covering complete user flows and business logic

### Types of Testing:

- Functional Testing
- Black Box Testing
- Sanity Testing
- Regression Testing
- UI Responsiveness Testing
- Cross-Browser Testing
- API Testing (RESTful)
- Database Testing (Data validation, CRUD operations, constraints)

### Exclusions:

- Third-party service testing (e.g., payment gateways, CDNs)
  - Security Penetration Testing (covered in a separate security review)
  - Performance testing beyond load and responsiveness
- 

## 3. Objectives

- Validate key user workflows function as intended
  - Ensure database interactions reflect accurate state transitions
  - Confirm APIs work independently and in integration
  - Detect regressions across builds/releases
  - Verify cross-browser compatibility and responsive design behavior
  - Provide a release recommendation based on test outcomes and defect metrics
- 

4. Exclusions

- Deep security audits (e.g., penetration testing, vulnerability scanning)
  - Load testing at scale (beyond defined responsiveness validations)
  - Multi-language and regionalization testing
- 

5. Test Environments

Category	Details
Operating Systems	Windows 10/11, macOS, Ubuntu 22.04+
Browsers	Chrome, Firefox, Edge, Safari
Devices	Desktop, Laptop, iPad, Mobile Phones
APIs	RESTful, JSON, Auth (Token-based)
Databases	MySQL/PostgreSQL (Staging mirror)
Network Types	Wi-Fi, 4G, corporate VPN, throttled 3G (UX testing)
Security Protocols	HTTPS, JWT/OAuth2-based authentication

---

6. Defect Reporting Procedure

- **Defect Identification:**
  - Functional mismatches, broken APIs, UI inconsistencies, data errors
- **Reporting Standards:**
  - Tool: JIRA
  - Includes: Summary, Environment, Steps, Expected vs Actual, Evidence
- **Severity Levels:**
  - Blocker, High, Medium, Low

- **Priority Assignment:**
  - Based on business risk and user impact
- **Communication:**
  - Slack, daily QA calls, and weekly stakeholder sync
- **Metrics:**
  - Defect density, resolution time, test execution coverage

---

## 7. Test Strategy

### Test Design Techniques:

- Equivalence Partitioning
- Boundary Value Analysis
- Use Case and State Transition Testing
- Decision Table Testing
- Exploratory Testing (Time-boxed sessions)

### Test Types and Approach:

Test Type	Approach
Functional Testing	Positive and negative tests for all features
Black Box Testing	Input/output behavior validation without internal code knowledge
Sanity Testing	Quick checks post-deployment to ensure system stability
Regression Testing	Re-executing tests to verify no new issues introduced
UI Responsiveness	Grid-based responsive testing on multiple resolutions
Cross-Browser Testing	Use of BrowserStack/Sauce Labs or manual multi-browser verification
API Testing	Using Postman/Newman to validate endpoints, headers, responses, and error codes
Database Testing	Data integrity, query validation, constraint enforcement, transaction flow testing

### Automation Scope:

- API Testing (Postman + Newman, or REST Assured)
- Sanity and Smoke Scenarios (Selenium/TestNG)
- Cross-browser Regression (Optional based on project timelines)

---

## 8. Test Schedule

Activity	Start Date	End Date	Owner
Test Plan & Review	[Date]	[Date]	QA Lead
Test Case Design & Approval	[Date]	[Date]	QA Team
Environment Setup	[Date]	[Date]	DevOps
Test Execution	[Date]	[Date]	QA Team
API and Database Testing	[Date]	[Date]	QA Team
Regression & Exploratory Testing	[Date]	[Date]	QA Team
Test Closure Report	[Date]	[Date]	QA Lead

---

## 9. Test Deliverables

- Reviewed Test Plan
  - Test Scenarios & Detailed Test Cases
  - API Test Collection & Database Validation Scripts
  - Test Execution Reports
  - Daily/Weekly QA Dashboards
  - Final QA Sign-Off Document
- 

## 10. Entry and Exit Criteria

### Entry Criteria:

- Approved BRD/FSD
- Test environment ready with stable build
- Access to API documentation, DB schema
- Test data available

### Exit Criteria:

- 95–100% test case execution
- All high and critical bugs resolved
- API and DB validations complete
- Test Summary Report submitted

- QA sign-off from QA Lead and PO

---

11. Tools	
Tool Category	Tools Used
Test Management	Excel, TestRail
Bug Tracking	JIRA
API Testing	Postman, Newman
UI Automation	Selenium, TestNG
Screenshot Evidence	Lightshot, Snagit
Responsive Testing	Chrome DevTools, BrowserStack
Database Validation	MySQL Workbench, DBeaver
Collaboration	Slack, Confluence

---

12. Risks and Mitigations		
Risk	Impact	Mitigation
Incomplete API documentation	Medium	Early collaboration with backend team
Test data unavailability	High	Seed data scripts and synthetic data generation
Database schema changes mid-sprint	High	Lock schema post-approval, use migration logs
Browser rendering inconsistencies	Medium	Early cross-browser testing on priority pages
QA resource crunch	Medium	Cross-training and flexible QA assignment

---

13. Approvals		
Document	Approver	Status
Test Plan	Product Manager	[Pending/Approved]
Test Cases	QA Lead	[Pending/Approved]
Test Summary Report	QA Lead, PO	[Pending/Approved]