# ProxyFy Of Test Plan

## 1. Introduction

Project Name: Website Quality Assurance & Developer Collaboration  
Test Engineer: Abhishek Maurya  
Testing Scope: Customer Portal, Admin Dashboard, Responsive Design  
Testing Type: Functional, UI/UX, Responsiveness, and Automation

## 2. Objectives

- Ensure core functionalities work as expected.  
- Validate UI consistency across multiple devices.  
- Identify and document high-impact bugs.  
- Improve communication with developers through structured bug reports.  
- Implement basic automation for key user flows.

## 3. Testing Scope

### Customer Portal

- User Registration & Login  
- Product Browsing & Search  
- Shopping Cart & Checkout Process  
- Payment Gateway Integration  
- Order Confirmation & Tracking  
- UI/UX Consistency (Desktop, Mobile, Tablet)

### Admin Dashboard

- User Management (Create, Edit, Delete)  
- Product Inventory Management  
- Order Processing & Status Updates  
- Reporting & Analytics  
- Role-Based Access Control

## 4. Test Approach

- \*\*Manual Testing:\*\* Execute test scenarios and document bugs.  
- \*\*Automation Testing:\*\* Create 5 Selenium scripts for critical flows.  
- \*\*Developer Collaboration:\*\* Categorize bugs and participate in triage meetings.

## 5. Test Scenarios

### Customer Portal

1. User should be able to register and log in.  
2. Product search should return relevant results.  
3. Adding/removing items from the cart should function correctly.  
4. Checkout process should validate payment details.  
5. Responsive UI should display properly.

### Admin Dashboard

1. Admin should be able to log in and manage products.  
2. Order details should be displayed accurately.  
3. User role permissions should be enforced correctly.  
4. Editing product details should reflect on the customer portal.  
5. Analytics and reporting features should work as expected.

## 6. Bug Reporting Guidelines

- Title:Clear and concise  
- Steps to Reproduce Step-by-step instructions  
- Expected vs Actual Result: Explain the deviation  
- Screenshots/Logs:Provide necessary evidence  
- Priority & Severity: Assign appropriate impact levels

## 7. Automation Plan

- Test Cases to Automate:  
 1. User Login Validation  
 2. Product Search Functionality  
 3. Add to Cart & Checkout Process  
 4. Order History Verification  
 5. Admin Login & Inventory Update

## 8. Risk & Mitigation Plan

| Risk | Mitigation |
| --- | --- |
| Tight deadline | Prioritize critical tests |

## 9. Deliverables

- Test Plan Document  
- Bug Reports in GitHub Issues  
- Selenium Test Scripts  
- Analysis Document  
- Video Demonstration

## 10. Conclusion

This test plan ensures comprehensive testing, bug reporting, and automation while improving collaboration with developers.