**Test Plan for Style Haven – Fashion E-commerce Platform**

**1. Objective**

To validate the functionality, usability, security, and performance of the Style Haven platform ensuring a seamless shopping experience for end users and sellers.

**2. Scope**

* **In Scope**:
  + Functional Testing (User Accounts, Product Catalog, Cart, Checkout, Seller Dashboard, Customer Support)
  + Non-Functional Testing (Performance, Security, Scalability, Cross-browser, Cross-platform)
  + Optional Features (Wishlists, Promotions, Loyalty, Social Integration)
* **Out of Scope**:
  + Backend infrastructure scalability testing beyond projected traffic
  + Payment gateway certification (handled by provider)

**3. Test Strategy**

**Types of Testing**

1. **Functional Testing** – Validate all modules (registration, login, cart, checkout, seller dashboard).
2. **UI/UX Testing** – Consistency across browsers (Chrome, Safari) and OS (Mac, Windows).
3. **Cross-Browser/Platform** – Chrome (latest -2 versions), Safari (latest), Mac & Windows.
4. **Integration Testing** – Payment gateway, order confirmation emails, seller dashboard.
5. **Security Testing** – Authentication, authorization, SQL injection, XSS, CSRF.
6. **Performance Testing** – Load test with JMeter for 500 concurrent users.
7. **Regression Testing** – Automated via Cypress/Playwright for key workflows.
8. **UAT Support** – Validate with pilot users before go-live.

**4. Test Environment**

* **OS**: macOS Ventura, Windows 11
* **Browsers**: Chrome (v120+), Safari (latest)
* **Devices**: Desktop/laptops only (mobile optional if time permits)
* **Test Data**: Dummy user accounts, test payment details (sandbox), sample seller products

**5. Roles & Responsibilities**

* **Tester 1**: User workflows (Accounts, Catalog, Cart, Checkout)
* **Tester 2**: Seller dashboard, Inventory, Order fulfillment, Customer support
* **Tester 3**: Non-functional testing (Performance, Security, Cross-browser/platform)

**6. Test Deliverables**

* Test Scenarios & Test Cases (in Excel/TestRail/Jira)
* Test Data Sheets
* Bug Reports (tracked in Jira/Azure DevOps)
* Daily Test Execution Report
* Final Test Summary Report

**7. Test Schedule (45 Days)**

| **Phase** | **Duration** | **Activities** |
| --- | --- | --- |
| Day 1–3 | Planning | Requirement review, test plan, environment setup |
| Day 4–15 | Test Design | Write test cases, prepare test data |
| Day 16–35 | Test Execution | Functional, integration, UI/UX, regression |
| Day 36–40 | Non-Functional | Security, performance, cross-browser |
| Day 41–43 | UAT Support | Assist business users in validation |
| Day 44–45 | Closure | Test summary report, sign-off |

**8. Entry & Exit Criteria**

* **Entry**: Requirements finalized, test environment ready, build deployed.
* **Exit**: All high-priority test cases executed, critical defects resolved, regression passed.

**9. Risks & Mitigation**

* Limited browser coverage → Prioritize Chrome/Safari latest.
* Short timeline (45 days) → Parallel execution across 3 testers.
* Dependencies on 3rd party (payment) → Use sandbox/stubs if delay.

**10. Tools**

* **Functional Automation**: Cypress / Playwright
* **Performance Testing**: JMeter
* **Bug Tracking**: Jira / Azure DevOps
* **Security**: OWASP ZAP, Burp Suite (lightweight scan)