

TEST PLAN – Sauce Demo Web Application

Project Name: Sauce Demo – Web UI Automation & Manual Testing

Version: 1.0

Prepared By: Arslan Jabbar (SQA Engineer)

Creation Date: 16-Feb-2025

Application Type: E-commerce (Web)

Testing Type: Manual + Automation (Playwright)

1. Introduction

The purpose of this Test Plan is to define the testing strategy, scope, objectives, resources, schedule, tools, and deliverables for validating the functionality of the Sauce Demo application. The application simulates an e-commerce platform where users log in, browse products, add items to the cart, complete checkout, and place an order.

This Test Plan ensures the system functions as expected before deployment and supports regression, functional, UI, security, and performance testing.

2. Test Objectives

The primary objectives of this testing effort are:

- Verify the complete **end-to-end purchase flow** from login → products → cart → checkout → order completion.
- Validate **functional accuracy** of all modules.
- Ensure **UI/UX compliance** including navigation, responsiveness, and layout.
- Identify and report **defects** efficiently.
- Validate **security behaviors**, such as unauthorized access and data manipulation prevention.
- Confirm **performance stability** of key flows.

3. Scope

In Scope

The following areas will be tested:

Functional Testing

- Login Module
- Product Listing & Sorting
- Add to Cart / Remove from Cart
- Checkout Step One (User Info Validation)
- Checkout Overview Page
- Order Confirmation
- Logout Functionality

Non-Functional Testing

- UI/UX validation
- Performance testing (basic load and response time)
- Security testing (URL tampering, devtools manipulation)

Automation Scope

- Full checkout flow using Playwright
- Smoke suite (login + add item + logout)
- Regression suite for cart & checkout

Out of Scope

- API Testing (application does not expose APIs publicly)
- Database testing (not provided)
- Mobile app testing (web-only)
- Load testing beyond basic performance metrics
- Backend server validation

4. Test Approach

4.1 Manual Testing Approach

- Execute all functional test cases from each module.
- Validate UI/UX manually on desktop and mobile views.
- Perform edge case testing (invalid inputs, empty fields).
- Execute exploratory testing in unstable or high-risk areas.

4.2 Automation Testing Approach (Playwright)

- Automate high-priority flows such as login, add-to-cart, checkout, and order placement.
- Maintain Page Object Model (POM) design.
- Use Playwright Test Runner for execution and reporting.
- Run tests on multiple browsers (Chromium, Firefox, WebKit when needed).

5. Test Deliverables

Mandatory Deliverables

- Test Plan (this document)
- Test Cases (structured format with dates & execution details)
- Defect Reports
- Daily Test Execution Report
- Automation Test Scripts (Playwright)
- Test Summary Report
- Screenshots, Logs, and Videos (from Playwright)

6. Test Environment

Component

Details

Browser

Chrome, Firefox

Automation Framework

Playwright (POM design)

Language

JavaScript / Node.js

Component

Details

Recommended System

Windows 10+, macOS, or Linux

Test Data Source

Static test data provided by SQA team

7. Tools & Utilities

Purpose

Tool

Functional Automation

Playwright

Test Case Management

Excel / Google Sheets

Defect Tracking

Jira / Notion

Reporting

Allure / Playwright HTML reports

Version Control

GitHub

8. Risks & Mitigation

Risk

Impact

Mitigation

Application updates may break tests

Medium

Maintain modular POM structure

Third-party CDN delays (images, scripts)

Low

Add retry logic in automation

No backend access

Medium

UI-only validation strategy

Performance glitches for certain user types

Low

Capture logs & timestamps

9. Entry & Exit Criteria

Entry Criteria

- Test environment is ready
- All required test data is prepared
- Test cases reviewed and approved
- Access to application provided

Exit Criteria

- All high & medium severity defects are fixed
- All test cases executed
- Test summary report prepared
- Automation scripts run successfully

10. Test Schedule

Phase	Start Date	End Date
Test Planning	16-Feb-2025	17-Feb-2025
Manual Test Case Creation	17-Feb-2025	20-Feb-2025
Test Execution	20-Feb-2025	24-Feb-2025
Automation Development	21-Feb-2025	28-Feb-2025
Test Summary Report	01-Mar-2025	02-Mar-2025

11. Team & Roles

Role	Responsibility	Assigned To
QA Engineer	Test planning, execution, reporting	Arslan Jabbar
Automation Engineer	Build Playwright framework	Arslan Jabbar

Role	Responsibility	Assigned To
QA Lead	Review & approve deliverables	—
Developer	Fix reported bugs	Development Team

12. Approval

This Test Plan is approved by:

Name	Role	Signature
Arslan Jabbar	QA Engineer	_____
QA Lead	QA Lead	_____
Product Owner	Stakeholder	_____