Automation Exercise – Test Plan Document

Version: 1.0
Date: February 2026
Author: Senior QA Architect
Status: Draft
Domain: E-Commerce (Automation Exercise – https://automationexercise.com)

Table of Contents

1. Document Control & Version History

Field	Value
Document Title	Automation Exercise Test Plan
Project Name	Automation Exercise E-Commerce Platform
Version	1.0
Date	February 2026
Author	Senior QA Architect
Status	Draft

Version	Date	Author	Changes
1.0	Feb 2026	QA Architect	Initial version – full coverage (functional, negative, edge, security, integration, performance, usability)

2. Executive Summary

2.1 Project Overview

Automation Exercise is a demo e-commerce platform for practicing test automation. It supports user registration/login, product search and browsing, cart management, checkout (with multiple flows), contact form with file upload, newsletter subscription, and various UI behaviors (categories, brands, scroll, reviews).

Application URL: https://automationexercise.com

2.2 Testing Objectives

1. **Functional (Positive):** Validate all features work as per specification.
2. **Negative Testing:** Invalid inputs, wrong credentials, boundary conditions, and error paths.
3. **Edge Cases:** Empty cart, max quantity, duplicate email, session/cart persistence, concurrent actions.
4. **Security:** Input validation, XSS/SQL injection resistance, auth/session handling, file upload restrictions.
5. **Integration:** End-to-end flows (register → cart → checkout → payment), cross-module consistency.
6. **Performance:** Page load times, critical path response times, stability under normal load.
7. **Usability:** Navigation, labels, error messages, responsiveness, accessibility basics.

2.3 Scope Summary

In-Scope:

- User Management (Register, Login, Logout, existing-email registration)
- Product Discovery (Search, Categories, Brands, Product detail, Reviews)
- Cart (Add, Update quantity, Remove, Recommended items)
- Checkout (Register before/during checkout, Login before checkout, Payment, Invoice download)
- Contact Us (Form submit, file upload)
- Subscription (Home page, Cart page)
- UI Behaviors (Scroll, Test Cases page, Address details at checkout)
- Positive, negative, edge, security, integration, performance, and usability test scenarios

Out-of-Scope:

- Direct database testing
- Backend API testing (unless explicitly in scope)
- Load/stress testing beyond basic response-time checks
- Third-party payment gateway internals
- Mobile native app (web only)
- Email delivery verification

2.4 Key Stakeholders

Role	Responsibility
QA Engineer	Test design, execution, automation
QA Lead	Test plan approval, review
Product Owner	Requirements, UAT
Development Team	Bug fixes, build deployment

3. Test Strategy & Approach

3.1 Testing Methodology

Agile testing with test levels aligned to the lifecycle: functional, regression, security, integration, performance, and usability. Automation (Playwright + POM) for regression and CI; manual for exploratory and usability where needed.

3.2 Test Levels

Level	Responsibility	Coverage
Unit	Development	Components/functions (if applicable)
Integration	QA	Module interactions, checkout flow, cart ↔ login
System	QA	E2E workflows, cross-browser
UAT	Product Owner / Business	Business scenarios

3.3 Test Types

Type	Purpose	Coverage
Functional (Positive)	Verify requirements with valid data	All features and user journeys
Negative	Invalid/erroneous inputs and credentials	Login, register, forms, cart, checkout
Edge	Boundaries, empty states, duplicates, persistence	Cart, registration, search, quantity
Security	Auth, injection, file upload, session	Login, contact, file upload, inputs
Integration	Multi-step flows and data consistency	Register → Cart → Checkout → Payment
Performance	Response time and stability	Page load, critical transactions
Usability	UX and clarity	Navigation, messages, layout

3.4 Test Techniques

- **Equivalence Partitioning:** Valid/invalid email, password length, card format.
- **Boundary Value Analysis:** Password min length, quantity 0/1/max, field max length.
- **Decision Table:** Checkout paths (guest vs register vs login), payment methods.
- **State Transition:** Login → Cart → Checkout → Order placed; session expiry.
- **Error Guessing:** SQL injection, XSS, invalid file types, negative quantities.

3.5 Automation Strategy

- **Tool:** Playwright (JavaScript), Page Object Model.
- **Scope:** Regression and smoke for critical paths; expand to negative/edge where maintainable.
- **Reporting:** Allure (history, categories, executor, screenshots/videos/traces).

---

4. Scope – Detailed Test Coverage

4.1 User Management

Scenario Type	Examples

Positive	Register with valid data, login with correct credentials, logout, delete account.
Negative	Login with wrong email/password, empty fields, invalid email format.
Edge	Register with existing email, very long name/address, special characters in fields.
Security	SQL injection in login/register, XSS in name/email, session after logout.
Integration	Register → Login → Add to cart → Checkout → Delete account.
Usability	Clear error messages, visible "Logged in as", logout accessible.

4.2 Product & Search

Scenario Type	Examples
Positive	Search returns results, category/brand filters work, product detail opens.
Negative	Search with no results, invalid category.
Edge	Empty search, very long search string, special characters.
Integration	Search → Add to cart → Login → Cart persisted.
Performance	Search result page load < 3s.
Usability	Results readable, filters obvious.

4.3 Cart

Scenario Type	Examples
Positive	Add/remove/update quantity, view cart, proceed to checkout.
Negative	Quantity 0 or negative (if allowed by UI).
Edge	Empty cart, max quantity, add same product twice.
Integration	Cart persists after login; cart used in checkout.
Usability	Cart icon count, clear totals.

4.4 Checkout & Payment

Scenario Type	Examples
Positive	All checkout paths (register before/during, login before), payment, invoice download.
Negative	Invalid card number, expired date, empty required fields.
Edge	Minimal address, different country/state combinations.
Security	Card data not exposed in DOM/logs, HTTPS.
Integration	Full flow: Register/Login → Cart → Checkout → Payment → Order confirmation.
Performance	Checkout/payment step < 5s.

Usability	Clear steps, order summary visible.

4.5 Contact Us & File Upload

Scenario Type	Examples
Positive	Submit with all fields, attach file, success feedback.
Negative	Empty required fields, invalid email.
Edge	Very long message, large file (if limit exists).
Security	File type restriction, no path traversal; XSS in message.
Usability	"Get in Touch" visible, success message clear.

4.6 Subscription

Scenario Type	Examples
Positive	Subscribe from home and from cart with valid email.
Negative	Invalid email, empty email.
Edge	Already subscribed email.
Usability	Subscription box visible, success message.

4.7 UI & Navigation

Scenario Type	Examples
Positive	Scroll up with/without arrow, Test Cases page, address details at checkout.
Usability	Scroll button visible when needed, links work.

5. Test Environment

Component	Details
Application URL	https://automationexercise.com
Environment	Production (demo)
Browsers	Chrome (primary), Firefox, Edge
OS	Windows 10/11, macOS
Tools	Playwright, Node.js, Allure

6. Risk Analysis

ID	Risk	Impact	Probability	Mitigation

R001	Checkout/payment failure	High	Medium	Full checkout regression, multiple payment scenarios
R002	Cart/session inconsistency after login	High	Medium	Integration tests for cart persistence
R003	Security bypass (injection, file upload)	High	Low	Dedicated security test suite
R004	Poor performance on critical paths	Medium	Medium	Response-time checks, performance suite
R005	Unclear errors (usability)	Medium	Medium	Usability checklist, negative test validation

7. Test Schedule & Milestones

Phase	Duration	Deliverable
Test Planning	1–2 days	Test Plan approved
Test Design	3–5 days	Test cases/suites (all types)
Test Execution	Ongoing	Execution results, defect log
Regression / Automation	Ongoing	Automated suite, Allure reports
Test Closure	1 day	Test summary, metrics

8. Entry & Exit Criteria

Entry: Environment accessible, test data available, test cases reviewed, automation framework ready.
Exit: Planned cases executed, pass rate > 95%, no open P0/P1, summary report and deliverables submitted.

9. Roles & Responsibilities

QA Engineer: design, execution, automation.
QA Lead: plan/case review, defect triage.
Development: fixes, build.
Product Owner: UAT, requirements.

10. Defect Management

Severity: P0 (Critical) – P3 (Low).
Lifecycle: New → Assigned → In Progress → Fixed → Retest → Closed.
Template: ID, Title, Severity, Module, Steps, Expected/Actual, Environment.

11. Test Deliverables

- Test Plan (this document)

- Test Suites (Positive, Negative, Edge, Security, Integration, Performance, Usability)
- Automation suite (Playwright + POM)
- Allure reports
- Defect log and test summary report

12. Assumptions & Dependencies

Stable application URL, test data (e.g. user-data.json) available, no direct DB/API access unless agreed.

13. Traceability Matrix (Summary)

Requirement Area	Positive	Negative	Edge	Security	Integration	Performance	Usability
User Management	✓	✓	✓	✓	✓	-	✓
Product & Search	✓	✓	✓	-	✓	✓	✓
Cart	✓	✓	✓	-	✓	-	✓
Checkout & Payment	✓	✓	✓	✓	✓	✓	✓
Contact & Upload	✓	✓	✓	✓	-	-	✓
Subscription	✓	✓	✓	-	-	-	✓
UI & Navigation	✓	-	-	-	-	-	✓

Document End

(Aligned with IEEE 829 and industry QA best practices.)