

Software Requirements Specification (SRS) for Automation Exercise Website

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) document outlines the functional and non-functional requirements for the Automation Exercise website (<https://automationexercise.com/>), an e-commerce platform focused on apparel and clothing products. The site serves as a demo environment for testing and automation, providing core features for product browsing and cart management. This document is based on requirements gathered as of July 10, 2025, and updated with site analysis as of October 08, 2025. It defines the system's capabilities to ensure alignment with user interactions, emphasizing simplicity for testing purposes while supporting basic e-commerce workflows.

1.2 Scope

The system includes a product catalog for browsing apparel items (tops, t-shirts, dresses, jeans, sarees, kids' wear), add-to-cart functionality, and implied shopping cart management. It supports user registration, login, account management, and contact features. Exclusions include advanced analytics, payment integrations, or multi-vendor support, as the site is primarily a testing platform. Future enhancements could add search filters or detailed product views.

1.3 Definitions, Acronyms, and Abbreviations

- **SRS:** Software Requirements Specification
- **User:** End-user (e.g., customer browsing or purchasing)
- **Admin:** System administrator (limited visibility on site)
- **FR:** Functional Requirement
- **NFR:** Non-Functional Requirement
- **Cart:** Shopping cart for holding selected products

1.4 References

- IEEE Std 830-1998: Recommended Practice for Software Requirements Specifications
- Requirements Gathering Notes (dated July 10, 2025)
- Site Analysis: <https://automationexercise.com/> (accessed October 08, 2025)

1.5 Overview

Section 2 details functional requirements based on observed site features. Section 3 covers non-functional requirements inferred from the site's structure. Section 4 describes expected system behaviour.

2. Functional Requirements

Functional requirements are derived from the site's core elements, including the product catalog and basic user interactions. The site emphasizes a simple, demo-style e-commerce flow with apparel-focused products priced in Indian Rupees (Rs.).

2.1 User Authentication

- **FR-1.1:** The system shall allow users to register an account via a form collecting name, email, password, and optional details (e.g., title, DOB), with email confirmation.

- **FR-1.2:** The system shall provide a login form accepting email and password, with logout functionality and "delete account" option for registered users.
- **FR-1.3:** The system shall manage sessions, requiring login for account-specific actions like order history.

2.2 Product Catalog

- **FR-2.1:** The system shall display a homepage catalog of 40+ apparel products (e.g., Blue Top at Rs. 500, Men T-shirt at Rs. 400, Sleeveless Dress at Rs. 1000) in a grid or list format, including name, price, and "Add to Cart" button per item.
- **FR-2.2:** The system shall categorize products implicitly (tops, t-shirts, dresses, jeans, sarees, kids' wear) without advanced filters or search, allowing sequential browsing.
- **FR-2.3:** The system shall support viewing individual product details (inferred from standard flows), including images and basic descriptions.

2.3 Shopping Cart and Checkout

- **FR-3.1:** The system shall enable users to add products to the cart via buttons on the catalog, supporting multiple additions and quantity updates.
- **FR-3.2:** The system shall provide a cart view page to display added items, totals, and options to remove or proceed to checkout.
- **FR-3.3:** The system shall facilitate checkout with address entry (billing/shipping), coupon application, and payment simulation (as a demo site), generating order confirmation and email.

2.4 Forms and User Interactions

- **FR-4.1:** The system shall include a contact form on a dedicated page with fields for name, email, subject, and message, submitting inquiries via email.
- **FR-4.2:** The system shall support subscription forms for newsletters, capturing email addresses.
- **FR-4.3:** The system shall provide user account forms for editing profile, viewing/deleting accounts, and accessing order history.

2.5 Additional Features

- **FR-5.1:** The system shall include a test cases page listing predefined scenarios (e.g., registration, add to cart) for automation testing purposes.
- **FR-5.2:** The system shall support basic navigation via implied top menu (home, products, cart, signup/login, contact).

3. Non-Functional Requirements

Non-functional requirements address the site's demo nature, ensuring reliability for testing while maintaining basic e-commerce standards.

3.1 Performance

- **NFR-1.1:** The homepage shall load within 2 seconds, displaying 40+ product listings without pagination delays.
- **NFR-1.2:** The system shall handle up to 500 concurrent users (testing load) with response times under 1 second for cart additions.
- **NFR-1.3:** Form submissions (e.g., registration) shall process in under 3 seconds.

3.2 Security

- **NFR-2.1:** The system shall use HTTPS for all pages to secure data transmission.
- **NFR-2.2:** Passwords shall be hashed during registration/login, with basic validation to prevent weak credentials.
- **NFR-2.3:** The system shall protect against injection attacks in forms (e.g., contact, registration) via sanitization.
- **NFR-2.4:** As a demo site, it shall log suspicious activities without full PCI compliance, but simulate secure checkout.

3.3 Usability

- **NFR-3.1:** The interface shall be responsive for desktop and mobile, with clear "Add to Cart" buttons and repetitive product listings for easy scanning.
- **NFR-3.2:** The site shall adhere to basic accessibility (e.g., alt text for product images, keyboard-navigable links) per WCAG 2.1 AA.
- **NFR-3.3:** Error messages shall be concise (e.g., "Email already exists" on registration) with retry guidance.
- **NFR-3.4:** English as primary language, with simple layout suitable for automation scripting.

3.4 Reliability and Maintainability

- **NFR-4.1:** The system shall maintain 99% uptime, with quick recovery from form submission errors.
- **NFR-4.2:** Code shall be modular for easy updates, supporting Selenium/WebDriver automation without anti-bot measures.

4. Expected Behaviour

4.1 System Behaviour

- The system shall handle duplicate product additions gracefully, updating quantities instead of creating multiples.
- Invalid form inputs (e.g., missing fields) shall trigger inline validation without page reloads.
- Cart persistence shall last for the session, clearing on logout or explicit empty.

4.2 Assumptions and Dependencies

- Assumes integration with email services for confirmations (e.g., registration, contact).
- Dependencies: Backend for cart state (e.g., session-based storage); frontend for dynamic updates.

4.3 Constraints

- Site designed for automation testing; no real payments or complex security.
- Limited to basic features; enhancements constrained by demo scope.
- Initial release/maintenance targeted for ongoing testing support.