Software Requirements Specification

E-Commerce Web Application

Version 1.0

Document Date: December 27, 2025

Attribute	Value
Document Owner	Product Management Team
Stakeholders	Engineering, QA, Product, Business
Classification	Internal Use
Status	Final

Table of Contents

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) document provides comprehensive functional and non-functional requirements for the E-Commerce Web Application. This document serves as the authoritative reference for development, quality assurance, and product management teams throughout the software development lifecycle. It is specifically designed to facilitate automated test case generation, Playwright-based UI automation, API testing, and RAG pipeline integration for intelligent test synthesis.

1.2 Scope

The E-Commerce Web Application is a full-featured B2C online retail platform providing end-to-end e-commerce functionality. The system includes user registration and authentication, product catalog with advanced search, shopping cart management, multi-step checkout with payment integration, order tracking, customer reviews, and administrative tools. The platform supports guest checkout, registered users, and administrative roles with appropriate access controls.

1.3 Definitions and Acronyms

Term	Definition
API	Application Programming Interface
MFA	Multi-Factor Authentication
SKU	Stock Keeping Unit - Unique product identifier
JWT	JSON Web Token
OTP	One-Time Password
PCI DSS	Payment Card Industry Data Security Standard
GDPR	General Data Protection Regulation
RAG	Retrieval-Augmented Generation
RBAC	Role-Based Access Control

2. Overall Description

2.1 Product Perspective

The system is a standalone web application following microservices architecture with clear separation between frontend (React/Vue.js), backend API (Node.js/Python), and data persistence (PostgreSQL/MySQL) layers. External integrations include payment gateways (Stripe, PayPal), email services (SendGrid), SMS providers (Twilio), and shipping APIs.

2.2 User Classes

User Class	Characteristics	Primary Functions
Guest User	Unauthenticated visitor	Browse products, search catalog, view details
Registered Customer	Authenticated account holder	Purchase, track orders, write reviews, manage wishlist
Administrator	System admin with elevated privileges	Manage products, process orders, moderate content
Support Agent	Customer service representative	Handle support tickets, process refunds, view order details

2.3 Operating Environment

Frontend: Modern browsers (Chrome 90+, Firefox 88+, Safari 14+, Edge 90+), responsive design supporting desktop (1920x1080), tablet (768x1024), and mobile (375x667) viewports.
Backend: Node.js 18+ or Python 3.10+, PostgreSQL 14+/MySQL 8+, Redis 6+ for caching, Linux OS (Ubuntu 22.04 LTS).

3. Functional Requirements

3.1 User Registration Module

3.1.1 Description

Multi-step registration process with email verification. System creates user account with encrypted credentials using bcrypt hashing (cost factor 10+).

3.1.2 Input Requirements

Field	Type	Required	Validation
first_name	String	Yes	2-50 chars, letters only
last_name	String	Yes	2-50 chars, letters only
email	String	Yes	RFC 5322 format, unique, MX validation
password	String	Yes	8-128 chars, complexity requirements
phone_number	String	No	E.164 format, 10-15 digits
date_of_birth	Date	No	YYYY-MM-DD, minimum age 13

3.1.3 Password Complexity Rules

• Minimum 8 characters, maximum 128
• At least one uppercase letter (A-Z)
• At least one lowercase letter (a-z)
• At least one digit (0-9)
• At least one special character: !@#$%^&*()_+-=[]{}|;:,.<>?
• Not in OWASP top 10,000 common passwords list

3.1.4 Success Flow

1. User submits registration form → 2. System validates inputs → 3. System checks email uniqueness → 4. System creates account (status: pending_verification) → 5. System generates UUID v4 verification token → 6. System sends verification email → 7. User clicks link within 24 hours → 8. System activates account → 9. Redirect to login with success message

3.1.5 Error Scenarios

Error	Message	HTTP Status
Email exists	Email already registered. Please login.	409 Conflict
Invalid email	Please enter valid email address.	400 Bad Request
Weak password	Password must meet complexity requirements.	400 Bad Request
Token expired	Verification link expired. Request new email.	410 Gone
Underage user	Must be 13+ years old to register.	400 Bad Request

3.2 Authentication Module

3.2.1 Login Flow

User enters credentials → System validates format → Rate limiting check (5 attempts/15 min) → Retrieve user by email → Verify password hash (bcrypt) → Check account status → Send OTP if MFA enabled → Generate JWT access token (15 min expiry) and refresh token (30 day expiry) → Store refresh token in HTTP-only cookie → Log event → Redirect to dashboard.

3.2.2 Session Management

Access Token: JWT with HS256 signature, 15-minute expiration, payload includes user_id, email, role, issued_at, expires_at.
Refresh Token: Opaque token stored in database, 30-day expiration with automatic rotation on use.
Security: Device fingerprinting, session termination on password change, remember-me extends refresh to 90 days.

3.2.3 Authentication Errors

Scenario	Response	HTTP
Invalid credentials	Generic error, increment counter	401
Account not verified	Verification required, resend option	403
Account suspended	Suspension message, contact support	403
Rate limit exceeded	Lockout message with retry time	429
Invalid OTP	OTP error, 3 retries max	401

3.3 Product Catalog Module

3.3.1 Product Listing Features

Paginated grid layout (12/24/48/96 items per page). Each card shows: thumbnail, name, price, discount %, rating (1-5 stars), review count, stock status. Sorting options: Relevance, Price (Low/High), Rating, Newest, Best Sellers.

3.3.2 Search and Filtering

Feature	Implementation	Details
Search	Full-text with Elasticsearch	Autocomplete (2+ chars), spell check, synonyms, fuzzy matching
Category Filter	Multi-select checkboxes	3-level hierarchy (e.g., Electronics > Laptops > Gaming)
Price Range	Dual slider	Min/max inputs, dynamic range
Brand Filter	Multi-select	Searchable list with product counts
Rating Filter	Star selection	4+ stars, 3+ stars, etc.
Availability	Radio buttons	In Stock, Out of Stock, Pre-order

3.4 Shopping Cart Module

3.4.1 Cart Operations

Operation	Validation	Business Rules
Add Item	Stock check	Max 10 per item, merge if exists, guest: session, logged: DB
Update Qty	Stock validation	Real-time price update, show max available if exceeded
Remove Item	Confirmation	Undo option (10 sec), recalculate totals
Apply Coupon	Code validation	Check expiry, usage limits, min purchase, applicable products
Save for Later	Login required	Move to saved list, keep 90 days

3.4.2 Cart Display

Product thumbnail, name, variant, SKU, unit price, quantity, subtotal, discount amount, stock status (In Stock/Low Stock/Out of Stock), estimated delivery. Cart Summary: Subtotal, Shipping (calculated at checkout), Tax estimate, Coupon discount, Order Total, Total Savings.

3.5 Checkout Process

3.5.1 Checkout Steps

Multi-step wizard with progress bar:
1. Authentication (Login or Guest)
2. Shipping Address (with validation)
3. Shipping Method Selection
4. Payment Information
5. Order Review and Confirmation

3.5.2 Shipping Address Fields

Field	Type	Required	Validation
Full Name	String	Yes	2-100 characters
Phone	String	Yes	E.164 format, 10-15 digits
Address Line 1	String	Yes	5-200 characters
Address Line 2	String	No	Max 200 characters
City	String	Yes	2-100 characters
State/Province	Dropdown	Yes	From predefined list
ZIP/Postal	String	Yes	Format varies by country
Country	Dropdown	Yes	ISO 3166-1 alpha-2

3.5.3 Shipping Methods

Method	Delivery Time	Cost	Availability
Standard	5-7 business days	Weight/flat rate	All regions
Express	2-3 business days	Weight + destination	Major cities

Next Day	1 business day	Premium rate	Select ZIP codes
Free Shipping	7-10 business days	$0	Orders over $50

3.5.4 Payment Processing

Supported Methods: Credit/Debit Cards (Visa, MC, Amex, Discover), Digital Wallets (PayPal, Apple Pay, Google Pay), Buy Now Pay Later (Afterpay, Klarna), Bank Transfer (ACH), Gift Cards.

3.5.5 Credit Card Validation

Field	Format	Validation
Card Number	13-19 digits	Luhn algorithm, card type detection, real-time masking
Expiration	MM/YY	Future date, month 01-12, year within 20 years
CVV/CVC	3-4 digits	3 for Visa/MC/Discover, 4 for Amex, never stored
Cardholder Name	String	As on card, 2-100 chars, letters and spaces
Billing ZIP	5-10 chars	AVS verification, format by country

3.5.6 Payment Flow

Select method → Load gateway iframe/SDK → Enter details (never touches server) → Gateway validates and tokenizes → Return token to application → User reviews order → Create order (status: pending_payment) → Authorize payment → 3DS if required → Update status to confirmed on success → Send confirmation email → Capture payment on shipment.

3.6 Order Management Module

3.6.1 Order Status Workflow

Status	Description	Next States
Pending Payment	Awaiting payment	Confirmed, Cancelled (30 min timeout)
Confirmed	Payment authorized	Processing (send email, reduce inventory)
Processing	Being prepared	Shipped (can cancel)
Shipped	Dispatched	In Transit, Delivered (capture payment, no cancel)
Delivered	Customer received	Final state (enable review, 30-day return)
Cancelled	Order cancelled	Final state (refund, restore inventory)
Returned	Return processed	Final state (refund after inspection)

3.6.2 Return Process

Eligibility: Within 30 days of delivery, original condition with tags, packaging intact. Non-returnable: perishables, intimate items, custom products, digital downloads.
Workflow: Request return → System validates eligibility → Select items and reason → Upload photos (optional) → Generate RMA number → Provide prepaid label → User ships within 7 days → Warehouse inspects → Approve: refund in 3-5 days; Reject: return to customer with explanation.

4. Non-Functional Requirements

4.1 Performance Requirements

Metric	Requirement	Measurement
Page Load Time	< 2 seconds	95th percentile, 3G connection
API Response Time	< 200ms	95th percentile for GET requests
Transaction Throughput	1000 TPS	Peak concurrent transactions
Search Response	< 500ms	Full-text search results
Concurrent Users	10,000	Simultaneous active users

4.2 Security Requirements

- **Authentication:** Bcrypt password hashing (cost 10+), JWT-based sessions, MFA support
- **Authorization:** RBAC with principle of least privilege
- **Data Protection:** TLS 1.3 for all connections, PCI DSS Level 1 compliance
- **Input Validation:** Server-side validation for all inputs, SQL injection prevention
- **CSRF Protection:** Token-based CSRF prevention on state-changing operations
- **Rate Limiting:** IP-based throttling (100 req/min per IP)

4.3 Scalability Requirements

Horizontal scaling with load balancing across multiple application servers. Database read replicas for query distribution. CDN for static assets. Stateless API design enabling auto-scaling. Caching layer (Redis) for session and frequently accessed data. Message queue (RabbitMQ/Kafka) for asynchronous processing.

4.4 Availability and Reliability

- **Uptime:** 99.9% SLA (max 43.2 min downtime/month)
- **Backup:** Daily automated backups with 30-day retention
- **Disaster Recovery:** RPO 1 hour, RTO 4 hours
- **Monitoring:** Real-time monitoring with alerting (Prometheus, Grafana)
- **Logging:** Centralized logging (ELK stack) with 90-day retention

5. Data Validation Rules

5.1 Email Validation

• RFC 5322 compliant format
• Maximum length 254 characters
• Domain must have valid MX records
• Reject disposable email providers
• Case-insensitive uniqueness check
• Whitespace trimming

5.2 Password Validation

• Length: 8-128 characters
• Character requirements: uppercase, lowercase, digit, special character
• No common passwords (OWASP top 10K list)
• No sequential characters (abc, 123)
• No username/email inclusion
• Password history check (last 5 passwords)

5.3 Input Sanitization

Input Type	Validation Rules	Sanitization
User Input Text	Max length enforcement	HTML entity encoding, XSS prevention
Numeric Fields	Type checking, range validation	Parse and validate as number
Dates	ISO 8601 format	Parse and validate date range
File Uploads	Extension whitelist, MIME check	Virus scan, size limit (10MB)
URLs	Valid URL format, HTTPS only	URL encoding, malicious link check

6. Error Handling and Messages

6.1 Error Response Format

All API errors follow consistent JSON structure:

```
{"error": {"code": "ERROR_CODE", "message": "Human-readable message", "details": [],
"timestamp": "ISO-8601"}}
```

6.2 HTTP Status Codes

Status	Usage	Example
200 OK	Successful GET/PUT	Product retrieved successfully
201 Created	Successful POST	User account created
400 Bad Request	Invalid input	Invalid email format
401 Unauthorized	Authentication required	Invalid credentials
403 Forbidden	Insufficient permissions	Admin access required
404 Not Found	Resource not found	Product ID does not exist
409 Conflict	Duplicate resource	Email already registered
422 Unprocessable	Validation failed	Password too weak
429 Too Many Requests	Rate limit exceeded	Try again in 15 minutes
500 Internal Error	Server error	Contact support

6.3 User-Friendly Error Messages

• Avoid technical jargon in user-facing messages
• Provide actionable guidance (e.g., 'Please check your email and try again')
• Never expose sensitive information (stack traces, DB errors)
• Include error codes for support reference
• Log detailed technical errors server-side with correlation ID

7. Security Requirements

7.1 Authentication Security

• Password hashing: Bcrypt with salt rounds >= 10
• Session tokens: Cryptographically secure random generation (32+ bytes)
• Token expiration: Access 15 min, refresh 30 days
• Failed login tracking: Account lockout after 5 failed attempts
• MFA support: TOTP (RFC 6238) and SMS-based OTP

7.2 Data Protection

• Encryption at rest: AES-256 for sensitive data (PII, payment info)
• Encryption in transit: TLS 1.3 with strong cipher suites
• PII handling: GDPR compliance, data minimization, right to erasure
• Payment data: Never store CVV, tokenize card numbers, PCI DSS Level 1
• Audit logging: All access to sensitive data logged with user ID and timestamp

7.3 API Security

Security Control	Implementation	Purpose
Rate Limiting	100 requests/min per IP	Prevent brute force and DoS
CORS Policy	Whitelist approved origins	Prevent unauthorized cross-origin requests
Input Validation	Schema validation (JSON Schema)	Prevent injection attacks
SQL Injection Prevention	Parameterized queries/ORM	Database security
XSS Prevention	Output encoding, CSP headers	Prevent script injection
CSRF Protection	Token validation on mutations	Prevent cross-site forgery

7.4 Compliance Requirements

PCI DSS: Level 1 compliance for card data handling, annual audit, quarterly vulnerability scans.
GDPR: Data protection officer, privacy policy, consent management, data portability, breach notification within 72 hours.
WCAG 2.1: Level AA accessibility, keyboard navigation, screen reader support, color contrast ratios.
SOC 2: Type II certification for security, availability, confidentiality.

8. Appendices

8.1 Testing Scope

This SRS document provides comprehensive requirements suitable for:
• Automated test case generation using AI/RAG pipelines
• End-to-end UI automation with Playwright/Selenium

• API testing with Postman/REST Assured
• Performance testing with JMeter/Gatling
• Security testing with OWASP ZAP/Burp Suite
• Accessibility testing with axe-core/Pa11y

8.2 Document Revision History

Version	Date	Author	Changes
1.0	Dec 27, 2025	Product Team	Initial release

--- End of Document ---