

Software Requirements Specification (SRS) Template

Project: E-Commerce Software(SnapBuy)

Version: 1.0

Authors: <Neha PM (PES1UG23AM183)> <Raksha (PES1UG23AM229)> <Rakshitha M (PES1UG23AM231)>

Date: 31/08/2025

Status: For Review

1. Introduction

1.1 Purpose

The purpose of this document is to define the functional and non-functional requirements for the E-Commerce Software(SnapBuy). The system allows customers to browse products, add them to a cart, place orders, make secure payments, and track their orders. Administrators will manage inventory, view reports, and maintain the system.

1.2 Scope

- **In-scope:**
 - User registration/login
 - Product catalog & search
 - Cart management
 - Checkout & payments
 - Order tracking & history
 - Admin functions (inventory, sales reports)
- **Out-of-scope:**
 - Third-party logistics systems
 - Warehouse automation
 - Vendor onboarding

1.3 Audience

- Developers (for building the system)
- Testers (for validating requirements)
- Customers & Admins (end users)
- Faculty/Evaluators (project reviewers)

1.4 Definitions

- **SKU:** Stock Keeping Unit
- **COD:** Cash on Delivery
- **OTP:** One Time Password
- **UI:** User Interface
- **API:** Application Programming Interface

2. Overall Description

2.1 Product Perspective

SnapBuy is a web-based e-commerce application. It connects to a backend database for product and order management, integrates with APIs for payment, and uses email/SMS services for notifications.

2.2 Major Product Functions (Detailed)

1. **User Registration & Authentication**
 - New users register with email/phone.
 - Login/Logout functions.
 - Password reset via OTP/email.
2. **Product Catalog & Search**
 - Browse by categories (Electronics, Fashion, etc.).
 - Search with filters (price, brand, rating).
 - Product detail page with description, images, price, stock status.
3. **Shopping Cart**
 - Add/remove products.
 - Update quantity.
 - Apply discount coupons.
 - Auto calculation of subtotal, taxes, discounts, final total.
4. **Order Management**
 - Place secure orders through checkout.
 - Payment via UPI, debit/credit card, COD.
 - Order confirmation with unique Order ID.
 - Track order status (Pending → Shipped → Delivered).
 - Cancellation/refunds (if allowed).
5. **Notifications & Communication**
 - Email/SMS order confirmation.
 - Notifications for shipping & delivery.
 - Promotional messages.
6. **Admin Functions**
 - Add/update/remove products.
 - Manage stock/inventory.
 - View sales statistics & revenue reports.
 - Manage registered users.

2.3 User Roles & Characteristics

- **Customer:**
 - Basic computer/mobile knowledge.
 - Browses catalog, orders, makes payments, tracks deliveries, gives reviews.
- **Administrator:**
 - Experienced system operator.
 - Manages inventory, sales reports, and user accounts.

- Requires high security clearance.
- **Seller (Optional):**
 - Adds and manages products.
 - Views orders for their products.
- **Delivery Staff (Optional):**
 - Updates order delivery status.
 - Needs mobile-friendly interface.

2.4 Operating Environment

- Frontend: Modern browsers (Chrome, Firefox, Edge).
- Backend: Web server + relational database.
- OS: Windows/Linux server environment.

2.5 Constraints

- Limited development time and resources.
- Must comply with data privacy/security standards.
- Must integrate with secure payment APIs.

3. External Interfaces

1. User Interfaces (UI)

These are what the **end users directly interact with**.

- **Login page** → Customers/Admins authenticate.
- **Product catalog** → Browse/search products.
- **Cart** → Add/remove items.
- **Checkout** → Payment & address input.
- **Order tracking** → Track delivery status.
- **Admin dashboard** → Manage products, orders, customers.

2. Hardware Interfaces

- **Server hosting** → Web/application/database servers.
- **Client devices** → PCs, smartphones, tablets.

3. Software Interfaces

Interaction with **third-party or internal software**.

- **Payment gateway APIs** (e.g., Razorpay, PayPal sandbox).
- **Email/SMS APIs** (e.g., SMTP server, Twilio, Firebase).

4. Communications Interfaces

How systems **exchange data securely**.

- **HTTPS with TLS/SSL** → Encrypts client–server communication.

4. Functional Requirements

Req ID	Requirement Description	Type	Priority	Source/Stakeholder	Acceptance Criteria
FR-001	System shall allow users to register and log in	Functional	High	Customer/Admin	User logs in successfully
FR-002	System shall allow browsing products by category	Functional	High	Customer	Category shows correct products
FR-003	System shall provide product search with filters	Functional	High	Customer	Search returns correct results
FR-004	System shall allow adding/removing items from cart	Functional	High	Customer	Cart updates accurately
FR-005	System shall calculate cart totals with discounts/taxes	Functional	High	Customer	Totals are correct
FR-006	System shall allow secure checkout	Functional	High	Customer	Checkout completes successfully
FR-007	System shall support multiple payment modes	Functional	High	Customer	UPI/Card/COD available
FR-008	System shall generate order confirmation	Functional	High	Customer	Confirmation displayed
FR-009	System shall track order status	Functional	High	Customer	Status updates correctly
FR-010	System shall allow users to view past orders	Functional	Medium	Customer	Past orders listed
FR-011	System shall send email/SMS notifications	Functional	Medium	Customer	Notifications received
FR-012	Admin shall add/update/remove products	Functional	High	Admin	Inventory changes saved
FR-013	Admin shall manage inventory	Functional	High	Admin	Stock levels update correctly

Req ID	Requirement Description	Type	Priority	Source/Stakeholder	Acceptance Criteria
FR-014	System shall allow users to rate/review products	Functional	Medium	Customer	Reviews visible
FR-015	System shall generate sales reports for admin	Functional	Medium	Admin	Reports generated

5. Non-Functional Requirements

Req ID	Requirement	Category	Priority	Acceptance Criteria
NFR-001	Search results < 2 seconds	Performance	High	90% of searches return <2s
NFR-002	System uptime 99% monthly	Reliability	High	Logs show uptime ≥99%
NFR-003	Role-based access enforced	Security	High	Unauthorized access blocked
NFR-004	Passwords encrypted	Security	High	Stored passwords are hashed
NFR-005	UI responsive on devices	Usability	Medium	Works on desktop & mobile

5.1.1. Security Requirements

Req ID	Requirement	Type	Priority	Acceptance Criteria
SR-001	Passwords shall be hashed before storage	Security	High	No plaintext passwords
SR-002	Transactions shall use TLS/SSL	Security	High	All traffic over HTTPS
SR-003	System shall enforce OTP for high-value purchases	Security	Medium	OTP required for >X amount
SR-004	Automatic logout after inactivity	Security	Medium	Session expires after 10 mins
SR-005	Log all login attempts with timestamps	Security	High	Logs record all attempts

5.1.2. Security Objectives

1. Confidentiality

- Ensure that sensitive data (user credentials, payment details, personal information) is always protected through encryption (HTTPS/TLS, password hashing).

2. Integrity

- Guarantee that transactions, orders, and payment data cannot be altered or tampered with during transmission or storage.

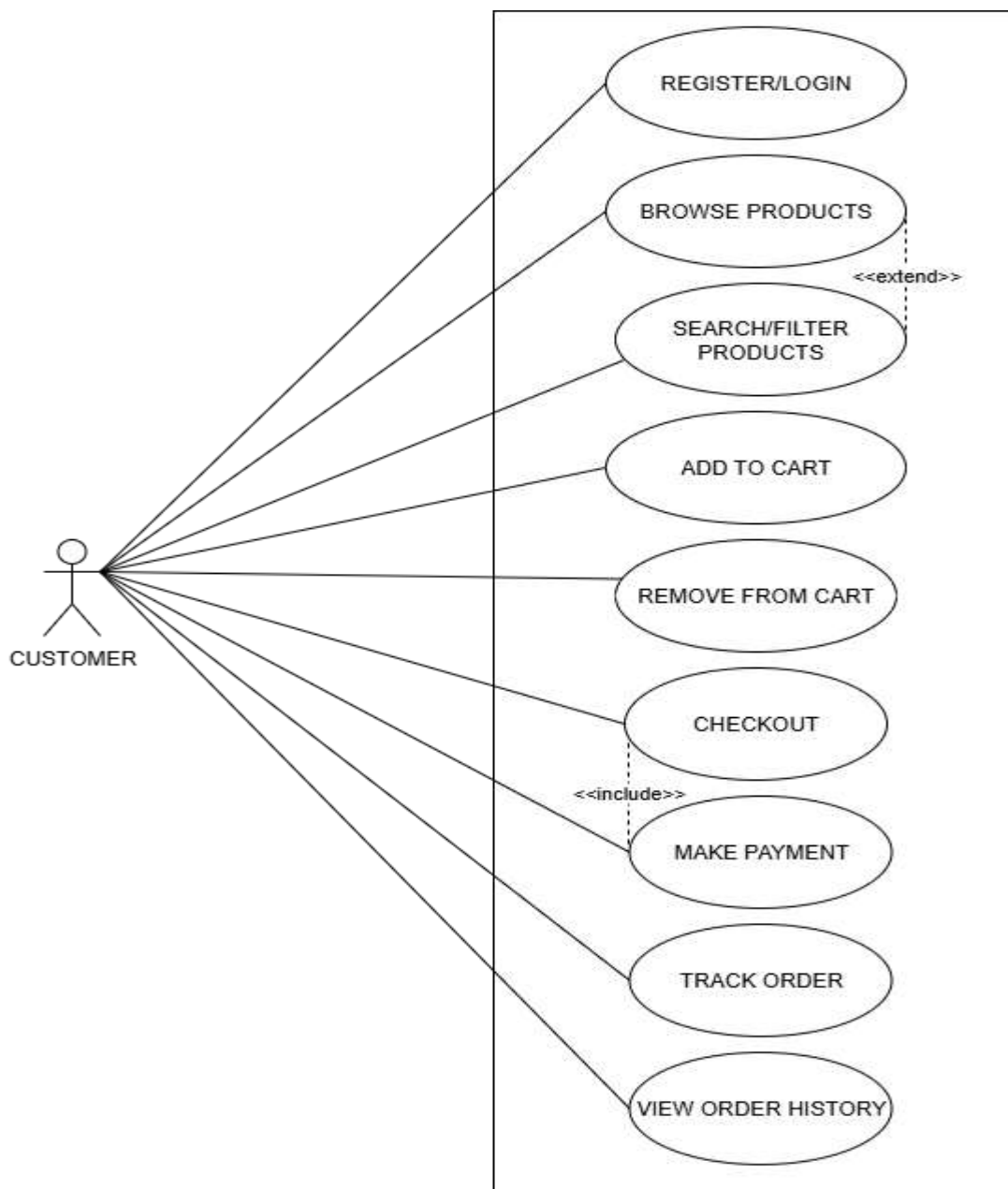
3. Authentication & Authorization

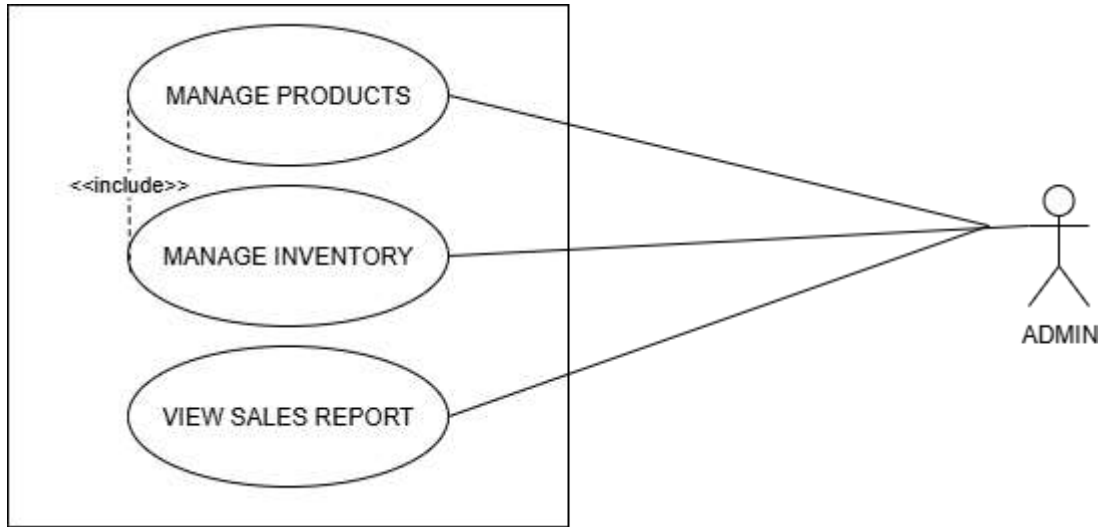
- Verify the identity of users through secure login, OTP for high-value transactions, and enforce role-based access (Customer vs. Admin).

6. Quality Attributes & Acceptance Tests

- **Exit criteria for acceptance:** All high-priority functional requirements implemented and verified, no critical NFR failures, and RTM shows all test cases passed.
- **Acceptance test suites:** Authentication, Catalog & Cart, Checkout & Payment, Order Management, Admin Management, Performance, Security, and Accessibility tests.

7. UML Use-Case Diagrams





8. Requirements Traceability Matrix (RTM)

Req ID	Requirement	Section Ref	Module	Test Case	Status
FR-001	Register/Login	4.1	Auth Module	Login-01	N
FR-004	Add to Cart	4.1	Cart Module	Cart-01	N
FR-009	Track Order	4.1	Order Module	Order-03	N
NFR-001	Response time	5	UI Module	Perf-01	N