# Functional Requirement

### Chapter 1: Functional Requirements   
  
\*\*1.1 Customer Registration Function\*\*   
Function ID: FR-01   
Description: Enables unregistered users to create new accounts with mandatory personal information and email verification.   
Input: Customer details (Name, Email, Password, PhoneNumber, Address).   
Output: New customer record stored in the database with unique CustomerID and encrypted credentials.   
  
\*\*1.2 Customer Authentication Function\*\*   
Function ID: FR-02   
Description: Validates registered customer credentials to grant system access.   
Input: Email and Password.   
Output: Authenticated session with permission tokens; login timestamp recorded in audit log.   
  
\*\*1.3 Customer Profile Management Function\*\*   
Function ID: FR-03   
Description: Allows authenticated customers to modify personal details (contact, address).   
Input: Updated customer attributes (PhoneNumber, Address).   
Output: Persisted database changes; audit log entry with modification timestamp.   
  
\*\*1.4 Admin Customer Management Function\*\*   
Function ID: FR-04   
Description: Enables administrators to create, update, or deactivate customer accounts.   
Input: Administrator actions (add/edit/deactivate), customer attributes.   
Output: Modified customer records; audit log with admin ID and action details.   
  
\*\*1.5 Product Catalog Management Function\*\*   
Function ID: FR-05   
Description: Allows administrators to add, edit, or delete products with category assignments.   
Input: Product attributes (Name, Description, Price, StockQuantity, CategoryID).   
Output: Updated product records; inventory synchronization; audit log entries.   
  
\*\*1.6 Inventory Adjustment Function\*\*   
Function ID: FR-06   
Description: Modifies stock quantities and triggers reorder alerts.   
Input: ProductID, new StockQuantity, adjustment reason.   
Output: Updated inventory data; low-stock notifications if applicable.   
  
\*\*1.7 Shopping Cart Modification Function\*\*   
Function ID: FR-07   
Description: Handles item additions, quantity updates, and removals in the cart.   
Input: ProductID, new Quantity (or removal flag).   
Output: Updated cart metadata (subtotal, item count); real-time inventory validation.   
  
\*\*1.8 Order Placement Function\*\*   
Function ID: FR-08   
Description: Converts cart contents into orders with payment processing.   
Input: Confirmed shipping address, payment method details.   
Output: New Order record; reduced inventory; order confirmation email; cleared cart.   
  
\*\*1.9 Order Tracking Function\*\*   
Function ID: FR-09   
Description: Retrieves real-time shipment status from carrier APIs.   
Input: OrderID.   
Output: Dynamic tracking data (carrier, location, estimated delivery); cached status timeline.   
  
\*\*1.10 Payment Processing Function\*\*   
Function ID: FR-10   
Description: Authorizes transactions and records payment details.   
Input: PaymentMethod (card details), Amount, OrderID.   
Output: Payment record (TransactionID, Status); order status synchronization.   
  
\*\*1.11 Category Hierarchy Management Function\*\*   
Function ID: FR-11   
Description: Manages product category creation, editing, and deletion.   
Input: Category attributes (Name, Description, ParentCategoryID).   
Output: Updated category tree; synchronized product-category associations.   
  
\*\*1.12 Admin Order Management Function\*\*   
Function ID: FR-12   
Description: Updates order statuses, processes cancellations/refunds.   
Input: OrderID, new OrderStatus, refund amounts.   
Output: Modified Order and Payment records; inventory restocks on cancellations.   
  
\*\*1.13 System Configuration Function\*\*   
Function ID: FR-13   
Description: Adjusts global settings (security, notifications, UI).   
Input: Configuration parameters (session timeout, email templates).   
Output: Persisted system-wide changes; audit log entries.   
  
\*\*1.14 Audit Logging Function\*\*   
Function ID: FR-14   
Description: Automatically records critical actions (logins, data modifications).   
Input: Actor type (Customer/Admin), action type, timestamp.   
Output: Immutable audit log entries stored in the database.   
  
\*\*1.15 Plugin API Integration Function\*\*   
Function ID: FR-15   
Description: Provides interfaces for third-party functionality extensions.   
Input: API requests compliant with documented specifications.   
Output: Executed plugin operations; system state modifications.   
  
\*\*1.16 Data Validation Function\*\*   
Function ID: FR-16   
Description: Enforces input constraints (email format, password strength).   
Input: User-provided data fields.   
Output: Validation errors or sanitized inputs for database operations.   
  
\*\*1.17 Email Notification Function\*\*   
Function ID: FR-17   
Description: Sends transactional emails (order confirmations, password resets).   
Input: Recipient email, template type, dynamic content.   
Output: Delivered emails; delivery status logs.   
  
\*\*1.18 Session Management Function\*\*   
Function ID: FR-18   
Description: Handles session creation/timeout for authenticated users.   
Input: Login credentials or session tokens.   
Output: Active session establishment/termination; timeout redirects.   
  
\*\*1.19 Dependency Validation Function\*\*   
Function ID: FR-19   
Description: Prevents operations violating data integrity (e.g., deleting products in active orders).   
Input: Entity ID (ProductID/CategoryID), action type.   
Output: Blocked action alert or dependency resolution prompts.   
  
\*\*1.20 Conflict Resolution Function\*\*   
Function ID: FR-20   
Description: Manages concurrent data editing conflicts.   
Input: Conflicting data versions from multiple users.   
Output: Merged valid changes; conflict notifications with latest data.   
  
---  
\*\*Compliance Verification\*\*   
- All inputs/outputs reference defined entities/attributes from the provided data model.   
- Functions map directly to use cases (e.g., FR-08 covers UC-12/UC-22; FR-12 covers UC-14).   
- Feasibility ensured: Inputs contain sufficient data for transformations (e.g., address → shipping cost calculation).   
- No undefined terms used; all references rooted in system description/data model.

# External Description

### Chapter 2: External Interfaces  
  
This chapter documents all external interfaces of the system, categorizing them into user interfaces, hardware interfaces, software interfaces, and communication interfaces. Each interface is defined with its role, interaction method, and relevant inputs/outputs.  
  
---  
  
#### \*\*2.1 User Interfaces\*\*  
  
\*\*2.1.1 Customer Registration Screen\*\*  
- \*\*Description:\*\* A form for unregistered users to input personal details.  
- \*\*Inputs:\*\* Name, Email, Password, PhoneNumber, Address.  
- \*\*Outputs:\*\* Confirmation message or error notifications.  
  
\*\*2.1.2 Customer Login Screen\*\*  
- \*\*Description:\*\* Interface for user authentication.  
- \*\*Inputs:\*\* Email, Password.  
- \*\*Outputs:\*\* Session token or error message.  
  
\*\*2.1.3 Customer Profile Screen\*\*  
- \*\*Description:\*\* Allows users to view and update personal details.  
- \*\*Inputs:\*\* Updated PhoneNumber, Address.  
- \*\*Outputs:\*\* Updated profile information or success message.  
  
\*\*2.1.4 Shopping Cart Screen\*\*  
- \*\*Description:\*\* Displays cart contents and allows modifications.  
- \*\*Inputs:\*\* ProductID, Quantity adjustments.  
- \*\*Outputs:\*\* Updated cart subtotal and item count.  
  
\*\*2.1.5 Order Placement Screen\*\*  
- \*\*Description:\*\* Interface for placing orders.  
- \*\*Inputs:\*\* Confirmed shipping address, payment method.  
- \*\*Outputs:\*\* Order confirmation message or error.  
  
\*\*2.1.6 Order Tracking Screen\*\*  
- \*\*Description:\*\* Shows real-time shipment status.  
- \*\*Inputs:\*\* OrderID.  
- \*\*Outputs:\*\* Carrier, location, estimated delivery.  
  
\*\*2.1.7 Data Validation Errors\*\*  
- \*\*Description:\*\* Displays validation errors.  
- \*\*Inputs:\*\* User-provided data.  
- \*\*Outputs:\*\* Error messages indicating issues.  
  
\*\*2.1.8 Session Timeout Notification\*\*  
- \*\*Description:\*\* Alerts user of session expiration.  
- \*\*Inputs:\*\* Session token.  
- \*\*Outputs:\*\* Logout prompt or redirect.  
  
\*\*2.1.9 Dependency Alert\*\*  
- \*\*Description:\*\* Warns of data dependencies.  
- \*\*Inputs:\*\* Entity ID, action type.  
- \*\*Outputs:\*\* Alert message or prompt.  
  
\*\*2.1.10 Conflict Notification\*\*  
- \*\*Description:\*\* Notifies of data conflicts.  
- \*\*Inputs:\*\* Conflicting data versions.  
- \*\*Outputs:\*\* Notification message or resolved data.  
  
---  
  
#### \*\*2.2 Hardware Interfaces\*\*  
  
\*\*2.2.1 Database Interaction\*\*  
- \*\*Description:\*\* Manages storage of customer, product, order, and audit records.  
- \*\*Inputs:\*\* Data to be stored.  
- \*\*Outputs:\*\* Stored records or query results.  
  
---  
  
#### \*\*2.3 Software Interfaces\*\*  
  
\*\*2.3.1 Payment API\*\*  
- \*\*Description:\*\* Processes transactions.  
- \*\*Inputs:\*\* Payment details, OrderID.  
- \*\*Outputs:\*\* Transaction ID, payment status.  
  
\*\*2.3.2 Email Notification System\*\*  
- \*\*Description:\*\* Sends transactional emails.  
- \*\*Inputs:\*\* Recipient email, template, content.  
- \*\*Outputs:\*\* Delivered emails, delivery logs.  
  
\*\*2.3.3 Carrier Tracking API\*\*  
- \*\*Description:\*\* Retrieves shipment status.  
- \*\*Inputs:\*\* OrderID.  
- \*\*Outputs:\*\* Tracking data, cached status.  
  
\*\*2.3.4 Plugin API\*\*  
- \*\*Description:\*\* Integrates third-party functionality.  
- \*\*Inputs:\*\* API requests.  
- \*\*Outputs:\*\* Plugin operations, system updates.  
  
---  
  
#### \*\*2.4 Communication Interfaces\*\*  
  
\*\*2.4.1 Email Notifications\*\*  
- \*\*Description:\*\* Communication via email for order confirmations and password resets.  
- \*\*Inputs:\*\* Recipient, template, content.  
- \*\*Outputs:\*\* Sent emails, delivery status logs.  
  
\*\*2.4.2 API Communication\*\*  
- \*\*Description:\*\* Interacts with external APIs for payment, tracking, and plugins.  
- \*\*Inputs:\*\* API requests.  
- \*\*Outputs:\*\* API responses, system modifications.  
  
---  
  
This structured approach ensures all external interfaces are clearly defined, aiding developers in understanding and implementing the system effectively. Each interface is categorized, described, and linked to its role in the system's functionality.