# Functional Requirement

### Chapter 1: Functional Requirements   
#### 1.1 Customer Registration   
- \*\*Function ID\*\*: FR-01   
- \*\*Description\*\*: Enables unregistered customers to create accounts by submitting required personal information. Validates inputs, checks email uniqueness, and stores profiles.   
- \*\*Input\*\*: Customer details (name, email, password, address).   
- \*\*Output\*\*: New customer profile stored in Customer data entity; confirmation message/email.   
  
#### 1.2 User Registration   
- \*\*Function ID\*\*: FR-02   
- \*\*Description\*\*: Allows unregistered users to create accounts. Validates inputs, checks email uniqueness, and stores user profiles.   
- \*\*Input\*\*: User details (name, email, password, contact information).   
- \*\*Output\*\*: New user profile stored in User data entity; confirmation message/email.   
  
#### 1.3 View Customer Information   
- \*\*Function ID\*\*: FR-03   
- \*\*Description\*\*: Permits administrators to retrieve and display customer profiles, including personal details and order history.   
- \*\*Input\*\*: Customer ID or selection from list.   
- \*\*Output\*\*: Customer details (name, email, address, order history); audit log entry.   
  
#### 1.4 Update Customer Information   
- \*\*Function ID\*\*: FR-04   
- \*\*Description\*\*: Enables customers or administrators to modify existing customer profiles (e.g., address, phone). Validates updates before saving.   
- \*\*Input\*\*: Modified customer fields (address, phone, email).   
- \*\*Output\*\*: Updated customer profile in Customer data entity; confirmation message; audit log entry.   
  
#### 1.5 Deactivate Customer Account   
- \*\*Function ID\*\*: FR-05   
- \*\*Description\*\*: Allows administrators to mark customer accounts as inactive. Verifies account status before deactivation.   
- \*\*Input\*\*: Customer ID selection.   
- \*\*Output\*\*: "Inactive" status in Customer data entity; confirmation message; audit log entry.   
  
#### 1.6 User Login   
- \*\*Function ID\*\*: FR-06   
- \*\*Description\*\*: Authenticates users/administrators via credentials. Creates sessions upon successful validation.   
- \*\*Input\*\*: Credentials (email, password).   
- \*\*Output\*\*: Active session; authentication status update in User Authentication data entity; audit log entry.   
  
#### 1.7 User Logout   
- \*\*Function ID\*\*: FR-07   
- \*\*Description\*\*: Terminates active user/administrator sessions and clears session data.   
- \*\*Input\*\*: "Logout" command.   
- \*\*Output\*\*: Session termination; "logged out" status in User Authentication data entity; audit log entry.   
  
#### 1.8 Add Product   
- \*\*Function ID\*\*: FR-08   
- \*\*Description\*\*: Allows administrators to add new products to inventory. Validates inputs (e.g., price format).   
- \*\*Input\*\*: Product details (name, description, price, quantity, category).   
- \*\*Output\*\*: New product stored in Product data entity; confirmation message; audit log entry.   
  
#### 1.9 View Product Details   
- \*\*Function ID\*\*: FR-09   
- \*\*Description\*\*: Displays product attributes (e.g., name, price, availability) to customers or administrators.   
- \*\*Input\*\*: Product selection or search query.   
- \*\*Output\*\*: Product details; audit log entry.   
  
#### 1.10 Update Product Information   
- \*\*Function ID\*\*: FR-10   
- \*\*Description\*\*: Enables administrators to modify product details (e.g., price, description). Validates inputs before saving.   
- \*\*Input\*\*: Modified product fields (price, description, quantity).   
- \*\*Output\*\*: Updated product in Product data entity; confirmation message; audit log entry.   
  
#### 1.11 Remove Product   
- \*\*Function ID\*\*: FR-11   
- \*\*Description\*\*: Allows administrators to delete products from inventory. Removes product from all shopping carts.   
- \*\*Input\*\*: Product ID selection.   
- \*\*Output\*\*: Product removed from Product data entity and Shopping Cart data entities; confirmation message; audit log entry.   
  
#### 1.12 Add Product to Shopping Cart   
- \*\*Function ID\*\*: FR-12   
- \*\*Description\*\*: Enables customers to add in-stock products to their cart. Updates quantities if product exists.   
- \*\*Input\*\*: Product selection.   
- \*\*Output\*\*: Product added/updated in Shopping Cart data entity; confirmation message; audit log entry.   
  
#### 1.13 View Shopping Cart   
- \*\*Function ID\*\*: FR-13   
- \*\*Description\*\*: Displays cart contents (items, quantities, total price) to authenticated customers.   
- \*\*Input\*\*: Access request to cart section.   
- \*\*Output\*\*: Cart items and total price; audit log entry.   
  
#### 1.14 Modify Shopping Cart   
- \*\*Function ID\*\*: FR-14   
- \*\*Description\*\*: Allows customers to adjust cart contents (e.g., update quantities, remove items). Recalculates totals.   
- \*\*Input\*\*: Modification action (e.g., "remove item," "update quantity").   
- \*\*Output\*\*: Updated Shopping Cart data entity; recalculated total price; confirmation message; audit log entry.   
  
#### 1.15 Checkout and Process Order   
- \*\*Function ID\*\*: FR-15   
- \*\*Description\*\*: Processes cart checkout: verifies item availability, collects shipping/payment details, processes payment, creates order, and clears cart.   
- \*\*Input\*\*: Shipping address, payment method, cart contents.   
- \*\*Output\*\*: New order in Order data entity; payment record in Payment Information data entity; cleared cart; confirmation message/email; audit log entry.   
  
#### 1.16 View Order History   
- \*\*Function ID\*\*: FR-16   
- \*\*Description\*\*: Displays transaction history (e.g., order IDs, dates, amounts) to customers or administrators. Supports filtering/sorting.   
- \*\*Input\*\*: Access request to transaction history.   
- \*\*Output\*\*: List of orders with details (date, status, amount); audit log entry.   
  
#### 1.17 View Order Details   
- \*\*Function ID\*\*: FR-17   
- \*\*Description\*\*: Shows specific order information (items, payment status, shipping address) upon selection.   
- \*\*Input\*\*: Order ID or selection.   
- \*\*Output\*\*: Order details; audit log entry.   
  
#### 1.18 Update Order Status   
- \*\*Function ID\*\*: FR-18   
- \*\*Description\*\*: Allows administrators to change order statuses (e.g., "Shipped," "Cancelled"). Notifies customers if applicable.   
- \*\*Input\*\*: New status selection.   
- \*\*Output\*\*: Updated status in Order data entity; customer notification; confirmation message; audit log entry.   
  
#### 1.19 Cancel Order   
- \*\*Function ID\*\*: FR-19   
- \*\*Description\*\*: Permits customers/administrators to cancel eligible orders (e.g., "Pending" status). Restocks inventory.   
- \*\*Input\*\*: Cancellation confirmation.   
- \*\*Output\*\*: "Cancelled" status in Order data entity; inventory restocked in Product data entity; customer notification; audit log entry.   
  
#### 1.20 View Payment Information   
- \*\*Function ID\*\*: FR-20   
- \*\*Description\*\*: Displays payment details (e.g., method, masked card numbers) to customers or administrators.   
- \*\*Input\*\*: Access request to payment section.   
- \*\*Output\*\*: Payment details; audit log entry.   
  
#### 1.21 Add Plugin   
- \*\*Function ID\*\*: FR-21   
- \*\*Description\*\*: Enables administrators to install new plugins by providing configuration details.   
- \*\*Input\*\*: Plugin details (name, description, version, configuration).   
- \*\*Output\*\*: New plugin stored in Plugin data entity; confirmation message; audit log entry.   
  
#### 1.22 Update Plugin   
- \*\*Function ID\*\*: FR-22   
- \*\*Description\*\*: Allows administrators to modify plugin configurations or metadata.   
- \*\*Input\*\*: Updated plugin fields (e.g., configuration settings).   
- \*\*Output\*\*: Updated plugin in Plugin data entity; confirmation message; audit log entry.   
  
#### 1.23 Remove Plugin   
- \*\*Function ID\*\*: FR-23   
- \*\*Description\*\*: Permits administrators to uninstall plugins.   
- \*\*Input\*\*: Plugin selection.   
- \*\*Output\*\*: Plugin removed from Plugin data entity; confirmation message; audit log entry.   
  
#### 1.24 Create Administrator Account   
- \*\*Function ID\*\*: FR-24   
- \*\*Description\*\*: Allows authenticated administrators to create new admin accounts with assigned roles.   
- \*\*Input\*\*: Administrator details (name, email, password, role).   
- \*\*Output\*\*: New administrator account in User Authentication data entity; confirmation message; audit log entry.   
  
#### 1.25 Update Administrator Account   
- \*\*Function ID\*\*: FR-25   
- \*\*Description\*\*: Enables administrators to modify existing admin accounts (e.g., roles, passwords).   
- \*\*Input\*\*: Updated administrator fields (role, password).   
- \*\*Output\*\*: Updated administrator account in User Authentication data entity; confirmation message; audit log entry.   
  
#### 1.26 Delete Administrator Account   
- \*\*Function ID\*\*: FR-26   
- \*\*Description\*\*: Permits administrators to remove admin accounts.   
- \*\*Input\*\*: Administrator ID selection.   
- \*\*Output\*\*: Administrator account removed from User Authentication data entity; confirmation message; audit log entry.   
  
#### 1.27 Add Authentication Method   
- \*\*Function ID\*\*: FR-27   
- \*\*Description\*\*: Allows administrators to configure new authentication mechanisms (e.g., OAuth).   
- \*\*Input\*\*: Authentication details (type, parameters).   
- \*\*Output\*\*: New authentication method in User Authentication data entity; confirmation message; audit log entry.   
  
#### 1.28 Update Authentication Method   
- \*\*Function ID\*\*: FR-28   
- \*\*Description\*\*: Enables administrators to modify existing authentication configurations.   
- \*\*Input\*\*: Updated authentication fields (parameters).   
- \*\*Output\*\*: Updated authentication method in User Authentication data entity; confirmation message; audit log entry.   
  
#### 1.29 Remove Authentication Method   
- \*\*Function ID\*\*: FR-29   
- \*\*Description\*\*: Permits administrators to delete authentication methods.   
- \*\*Input\*\*: Authentication method selection.   
- \*\*Output\*\*: Authentication method removed from User Authentication data entity; confirmation message; audit log entry.   
  
---   
\*Note: All functions adhere to referenced use cases, ensure input-output feasibility, and maintain terminological consistency.\*

# External Description

### Chapter 2: External Interfaces  
  
#### 2.1 User Interface   
The system will interact with users through a graphical user interface (GUI) that provides access to all functional requirements. The interface will include:   
- \*\*Input\*\*: User interactions such as form submissions, button clicks, and search queries.   
- \*\*Output\*\*: Screen displays showing customer information, product details, shopping cart contents, order history, and confirmation messages.   
- \*\*Description\*\*: The GUI will be designed to be user-friendly, with clear navigation and intuitive controls. It will support both customer and administrator roles, with role-based access to features.   
  
---  
  
#### 2.2 Hardware Interface   
The system will not directly interact with any external hardware devices beyond standard computing peripherals (e.g., keyboards, mice, displays).   
  
---  
  
#### 2.3 Software Interface   
The system will interact with the following external software components:   
  
1. \*\*Customer Data Entity\*\*   
 - \*\*Description\*\*: A database storing customer profiles, including personal details and order history.   
 - \*\*Interaction\*\*: The system will read and write customer data for registration, updates, and deactivation.   
 - \*\*Input/Output\*\*: Customer details (name, email, address) and order history.   
  
2. \*\*User Data Entity\*\*   
 - \*\*Description\*\*: A database storing user profiles, including administrators.   
 - \*\*Interaction\*\*: The system will read and write user data for registration, login, and account management.   
 - \*\*Input/Output\*\*: User details (name, email, password) and authentication status.   
  
3. \*\*Product Data Entity\*\*   
 - \*\*Description\*\*: A database storing product information, including inventory levels.   
 - \*\*Interaction\*\*: The system will read and write product details for adding, updating, and removing products.   
 - \*\*Input/Output\*\*: Product details (name, description, price, quantity).   
  
4. \*\*Shopping Cart Data Entity\*\*   
 - \*\*Description\*\*: A database storing temporary shopping cart contents for customers.   
 - \*\*Interaction\*\*: The system will read and write cart data for adding, modifying, and clearing items.   
 - \*\*Input/Output\*\*: Cart contents (products, quantities, totals).   
  
5. \*\*Order Data Entity\*\*   
 - \*\*Description\*\*: A database storing completed orders and their status.   
 - \*\*Interaction\*\*: The system will read and write order data for processing, updating, and viewing orders.   
 - \*\*Input/Output\*\*: Order details (items, status, shipping address).   
  
6. \*\*Payment Information Data Entity\*\*   
 - \*\*Description\*\*: A database storing payment details, including masked card numbers.   
 - \*\*Interaction\*\*: The system will read payment details for order processing and display.   
 - \*\*Input/Output\*\*: Payment details (method, masked card numbers).   
  
7. \*\*Plugin Data Entity\*\*   
 - \*\*Description\*\*: A database storing installed plugins and their configurations.   
 - \*\*Interaction\*\*: The system will read and write plugin data for adding, updating, and removing plugins.   
 - \*\*Input/Output\*\*: Plugin details (name, description, version, configuration).   
  
8. \*\*User Authentication Data Entity\*\*   
 - \*\*Description\*\*: A database storing authentication methods and administrator accounts.   
 - \*\*Interaction\*\*: The system will read and write authentication data for login, account management, and authentication method configuration.   
 - \*\*Input/Output\*\*: Authentication details (email, password, roles) and method configurations.   
  
9. \*\*Third-Party Authentication Services\*\*   
 - \*\*Description\*\*: External authentication services (e.g., OAuth) used for user login.   
 - \*\*Interaction\*\*: The system will integrate with these services to validate credentials and create sessions.   
 - \*\*Input/Output\*\*: Authentication tokens and user credentials.   
  
10. \*\*Email Service\*\*   
 - \*\*Description\*\*: An external email service used to send confirmation emails and notifications.   
 - \*\*Interaction\*\*: The system will send emails for account registration, order confirmations, and status updates.   
 - \*\*Input/Output\*\*: Email content and recipient addresses.   
  
---  
  
#### 2.4 Communication Interface   
The system will use the following communication interfaces to interact with external systems:   
  
1. \*\*Email Notifications\*\*   
 - \*\*Description\*\*: The system will send emails to users and administrators for confirmations, updates, and notifications.   
 - \*\*Interaction\*\*: Emails will be sent via an SMTP server.   
 - \*\*Input/Output\*\*: Email content (e.g., confirmation messages, order updates).   
  
2. \*\*API Calls\*\*   
 - \*\*Description\*\*: The system will interact with external APIs for authentication (e.g., OAuth) and payment processing.   
 - \*\*Interaction\*\*: RESTful API calls will be used to validate credentials and process payments.   
 - \*\*Input/Output\*\*: API requests (e.g., authentication tokens, payment details) and responses (e.g., success/failure status).   
  
3. \*\*Web Browser Communication\*\*   
 - \*\*Description\*\*: The system will interact with web browsers to display the user interface and handle user input.   
 - \*\*Interaction\*\*: The system will use HTTP/HTTPS protocols to communicate with the client-side web application.   
 - \*\*Input/Output\*\*: HTTP requests (e.g., form submissions) and responses (e.g., HTML pages, JSON data).   
  
---  
  
### Summary   
The external interfaces described in this chapter ensure seamless interaction between the system and its users, databases, third-party services, and communication channels. Each interface is clearly defined to support the functional requirements outlined in Chapter 1.