# Functional Requirement

# 1. Functional Requirements  
  
## 1.1 Customer Registration Function   
\*\*Function ID\*\*: FR-01   
\*\*Description\*\*: A customer can create a new account by providing personal information, including a valid email address and phone number. The system must validate the input data and send a confirmation message.   
\*\*Input\*\*: Name, email, password, phone number.   
\*\*Output\*\*: A new customer account is created, and the customer receives a confirmation message.   
  
## 1.2 Customer Login Function   
\*\*Function ID\*\*: FR-02   
\*\*Description\*\*: A customer can log in to the system using their registered email and password. The system must validate the credentials and initiate a session.   
\*\*Input\*\*: Registered email and password.   
\*\*Output\*\*: A successful login and session initiation, with the customer redirected to the homepage or dashboard.   
  
## 1.3 View Product Details Function   
\*\*Function ID\*\*: FR-03   
\*\*Description\*\*: A customer can view detailed information about a product, such as price, description, and availability. The system must retrieve the data from the Product data entity and update the view count if applicable.   
\*\*Input\*\*: Product ID or search query.   
\*\*Output\*\*: Displayed product details, including price, description, availability, and view count.   
  
## 1.4 Add Product to Cart Function   
\*\*Function ID\*\*: FR-04   
\*\*Description\*\*: A customer can add a product to their shopping cart. The system must verify product availability and update the cart accordingly.   
\*\*Input\*\*: Product ID, quantity (optional), and custom options (e.g., size, color).   
\*\*Output\*\*: Updated shopping cart with the added product and a confirmation message.   
  
## 1.5 Update Cart Contents Function   
\*\*Function ID\*\*: FR-05   
\*\*Description\*\*: A customer can modify the contents of their shopping cart, including changing the quantity or removing items. The system must update the cart and recalculate the total cost.   
\*\*Input\*\*: Cart item ID, action (increase quantity, decrease quantity, remove item), and new quantity (if applicable).   
\*\*Output\*\*: Updated cart summary and confirmation message of the cart change.   
  
## 1.6 Remove Product from Cart Function   
\*\*Function ID\*\*: FR-06   
\*\*Description\*\*: A customer can remove a product from their shopping cart. The system must verify the product exists in the cart, remove it, and update the cart summary.   
\*\*Input\*\*: Cart item ID.   
\*\*Output\*\*: Updated cart summary with the removed product and confirmation message.   
  
## 1.7 Checkout Order Function   
\*\*Function ID\*\*: FR-07   
\*\*Description\*\*: A customer can proceed with the checkout process by confirming shipping and billing details and selecting a payment method. The system must create an order, process the payment, update the inventory, and clear the cart.   
\*\*Input\*\*: Cart items, shipping and billing information, selected payment method.   
\*\*Output\*\*: A new order is created, inventory is updated, cart is cleared, and a confirmation message is sent to the customer.   
  
## 1.8 View Purchase History Function   
\*\*Function ID\*\*: FR-08   
\*\*Description\*\*: A customer can view their purchase history, including order dates, total amounts, and order status. The system must retrieve the data from the Order data entity and display it in a user-friendly format.   
\*\*Input\*\*: Customer account (logged-in session).   
\*\*Output\*\*: A list of orders with detailed information for each.   
  
## 1.9 Receive Order Confirmation Email Function   
\*\*Function ID\*\*: FR-09   
\*\*Description\*\*: After a successful checkout, the system must generate and send an order confirmation email to the customer via a configured Plugin. The email must include the order number, items, total amount, and shipping address.   
\*\*Input\*\*: Order details from the Order data entity.   
\*\*Output\*\*: A confirmation email sent to the customer and a log of the email delivery status.   
  
## 1.10 Administrator Login Function   
\*\*Function ID\*\*: FR-10   
\*\*Description\*\*: An administrator can log in to the system using their registered username and password. The system must validate the credentials and initiate a session.   
\*\*Input\*\*: Registered username and password.   
\*\*Output\*\*: A successful login and session initiation, with the administrator redirected to the admin dashboard.   
  
## 1.11 Manage Inventory Function   
\*\*Function ID\*\*: FR-11   
\*\*Description\*\*: An administrator can update the stock levels of products. The system must validate the input data and update the Inventory and Product data entities accordingly.   
\*\*Input\*\*: Product ID, new stock quantity.   
\*\*Output\*\*: Updated inventory and product availability status, along with a confirmation message.   
  
## 1.12 Update Product Information Function   
\*\*Function ID\*\*: FR-12   
\*\*Description\*\*: An administrator can modify product details such as name, price, description, and stock quantity. The system must validate and update the Product and Inventory data entities.   
\*\*Input\*\*: Product ID, updated attributes (name, price, description, stock quantity).   
\*\*Output\*\*: Updated product information in the catalog and cart interfaces, along with a confirmation message.   
  
## 1.13 Categorize Products Function   
\*\*Function ID\*\*: FR-13   
\*\*Description\*\*: An administrator can assign one or more categories to a product. The system must validate the categories and update the Product data entity.   
\*\*Input\*\*: Product ID, list of categories.   
\*\*Output\*\*: Updated product categories, and the product becomes searchable and browsable under those categories.   
  
## 1.14 Delete Product Function   
\*\*Function ID\*\*: FR-14   
\*\*Description\*\*: An administrator can remove a product from the system. The system must verify the product exists and ensure it is not referenced in any active order or cart before deletion.   
\*\*Input\*\*: Product ID.   
\*\*Output\*\*: The product is removed from the Product and Inventory data entities, and a confirmation message is displayed.   
  
## 1.15 Manage Plugin Function   
\*\*Function ID\*\*: FR-15   
\*\*Description\*\*: An administrator can modify the configuration or code of an existing plugin. The system must validate the updates and store the changes in the Plugin data entity.   
\*\*Input\*\*: Plugin ID, updated plugin name, description, configuration, or code.   
\*\*Output\*\*: Updated plugin information and confirmation message.   
  
## 1.16 View Plugin Documentation Function   
\*\*Function ID\*\*: FR-16   
\*\*Description\*\*: An administrator can view the documentation for a plugin, including configuration, usage, and integration instructions. The system must retrieve the documentation from the Plugin data entity.   
\*\*Input\*\*: Plugin ID.   
\*\*Output\*\*: Displayed plugin documentation and a log of the access.   
  
## 1.17 Manage Customer Function   
\*\*Function ID\*\*: FR-17   
\*\*Description\*\*: An administrator can view, edit, or delete customer accounts. The system must validate the input data and update the Customer data entity accordingly.   
\*\*Input\*\*: Customer ID, action (view, edit, delete), and updated customer information (if applicable).   
\*\*Output\*\*: Updated or deleted customer information, along with a confirmation message and system logs.   
  
## 1.18 Manage Product Function   
\*\*Function ID\*\*: FR-18   
\*\*Description\*\*: An administrator can add, update, or delete products in the system. The system must synchronize changes with the Product and Inventory data entities.   
\*\*Input\*\*: Product ID, action (add, update, delete), and product details (if applicable).   
\*\*Output\*\*: Updated or deleted product information in the catalog and inventory, with confirmation and logs.   
  
## 1.19 Manage Order Function   
\*\*Function ID\*\*: FR-19   
\*\*Description\*\*: An administrator can update the status, cancel, or initiate a refund for an order. The system must update the Order, Inventory, and Payment data entities accordingly and notify the customer.   
\*\*Input\*\*: Order ID, action (update status, cancel, refund).   
\*\*Output\*\*: Updated order status, inventory levels, and payment records, with confirmation and logs.   
  
## 1.20 Manage Payment Function   
\*\*Function ID\*\*: FR-20   
\*\*Description\*\*: An administrator can update the status, cancel, or initiate a refund for a payment. The system must validate the payment state and update the Payment and Order data entities.   
\*\*Input\*\*: Payment ID, action (update status, cancel, refund).   
\*\*Output\*\*: Updated payment status and related order information, with confirmation and logs.   
  
## 1.21 Manage Inventory Function (Extended)   
\*\*Function ID\*\*: FR-21   
\*\*Description\*\*: An administrator can perform stock adjustments (add, deduct, or set stock levels) for products. The system must validate the quantity and update both the Inventory and Product data entities.   
\*\*Input\*\*: Product ID, action (add, deduct, set stock), and quantity.   
\*\*Output\*\*: Updated inventory levels and product availability status, along with a confirmation message and logs.   
  
## 1.22 Manage Administrator Function   
\*\*Function ID\*\*: FR-22   
\*\*Description\*\*: An administrator can add, update, or delete other administrator accounts. The system must validate the input data and update the Administrator data entity accordingly.   
\*\*Input\*\*: Administrator ID, action (add, update, delete), and administrator details (if applicable).   
\*\*Output\*\*: Updated or deleted administrator account, with confirmation message and logs.

# External Description

# 2. External Interfaces   
  
## 2.1 User Interface Output   
  
The system interacts with customers and administrators through a user interface. These interactions include but are not limited to:   
  
- \*\*Customer Registration Form\*\*: A form allowing customers to input personal information such as name, email, password, and phone number.   
- \*\*Customer Login Page\*\*: A login interface where customers enter their registered email and password to authenticate and initiate a session.   
- \*\*Product Detail Page\*\*: A page displaying detailed product information including price, description, availability, and view count.   
- \*\*Shopping Cart Interface\*\*: An interface allowing customers to add, modify, or remove products from their cart. It provides a summary of the cart contents, including total cost and quantity.   
- \*\*Checkout Form\*\*: A form where customers confirm shipping and billing details and select a payment method.   
- \*\*Purchase History Page\*\*: A page displaying a list of the customer's previous orders, including order date, total amount, and status.   
- \*\*Order Confirmation Screen\*\*: A screen shown after a successful checkout, displaying the order confirmation and key details.   
- \*\*Administrator Login Page\*\*: A login interface for administrators to access the admin dashboard.   
- \*\*Admin Dashboard Interface\*\*: A comprehensive interface for administrators to manage inventory, products, plugins, customer accounts, and orders.   
- \*\*Product Management Interface\*\*: A tool for administrators to add, update, or delete products.   
- \*\*Inventory Management Interface\*\*: A tool for administrators to adjust stock levels and manage product availability.   
- \*\*Plugin Management Interface\*\*: An interface for administrators to modify plugin configurations or code.   
- \*\*Plugin Documentation Viewer\*\*: A user interface for administrators to view plugin documentation.   
- \*\*Customer Management Interface\*\*: A tool for administrators to view, edit, or delete customer accounts.   
- \*\*Order Management Interface\*\*: A tool for administrators to update order status, cancel orders, or process refunds.   
- \*\*Payment Management Interface\*\*: A tool for administrators to manage payment status, cancel payments, or process refunds.   
  
These interfaces must ensure user-friendly navigation and provide clear feedback to users regarding actions taken or errors encountered.   
  
## 2.2 Hardware Interface Output   
  
No hardware interfaces are required for this system. The system does not interact with external hardware devices and is designed to operate solely through software-based interactions.   
  
## 2.3 Software Interface Output   
  
The system interacts with several external software components and data entities to fulfill its functional requirements. These interfaces are described below:   
  
### 2.3.1 Product Data Entity   
- \*\*Description\*\*: This data entity stores product-related information such as product ID, name, price, description, stock quantity, and categories.   
- \*\*Interaction Method\*\*: The system retrieves and updates product data via database queries or stored procedures.   
- \*\*Inputs/Outputs\*\*:   
 - Input: Product ID, updated attributes (name, price, description, stock quantity).   
 - Output: Updated product information for display in the catalog and cart.   
  
### 2.3.2 Inventory Data Entity   
- \*\*Description\*\*: This data entity tracks the stock levels and availability status of products.   
- \*\*Interaction Method\*\*: The system updates and retrieves inventory data via database queries or stored procedures.   
- \*\*Inputs/Outputs\*\*:   
 - Input: Product ID, new stock quantity, or action (add, deduct, set stock).   
 - Output: Updated inventory levels and product availability status.   
  
### 2.3.3 Order Data Entity   
- \*\*Description\*\*: This data entity stores order information including order ID, customer ID, product IDs, quantities, total amount, shipping address, and order status.   
- \*\*Interaction Method\*\*: The system retrieves and updates order data via database queries or stored procedures.   
- \*\*Inputs/Outputs\*\*:   
 - Input: Order details (cart items, shipping and billing information, payment method).   
 - Output: Updated order status and related information (e.g., inventory levels, payment records).   
  
### 2.3.4 Payment Data Entity   
- \*\*Description\*\*: This data entity contains payment information such as payment ID, order ID, amount, payment method, status, and transaction logs.   
- \*\*Interaction Method\*\*: The system updates and retrieves payment data via database queries or stored procedures.   
- \*\*Inputs/Outputs\*\*:   
 - Input: Payment ID, action (update status, cancel, refund).   
 - Output: Updated payment status and related order information.   
  
### 2.3.5 Customer Data Entity   
- \*\*Description\*\*: This data entity stores customer account information such as customer ID, name, email, password, phone number, and account status.   
- \*\*Interaction Method\*\*: The system retrieves and updates customer data via database queries or stored procedures.   
- \*\*Inputs/Outputs\*\*:   
 - Input: Customer ID, action (view, edit, delete), and updated customer information.   
 - Output: Updated or deleted customer account information.   
  
### 2.3.6 Administrator Data Entity   
- \*\*Description\*\*: This data entity stores administrator account information including administrator ID, name, username, password, and role.   
- \*\*Interaction Method\*\*: The system retrieves and updates administrator data via database queries or stored procedures.   
- \*\*Inputs/Outputs\*\*:   
 - Input: Administrator ID, action (add, update, delete), and administrator details.   
 - Output: Updated or deleted administrator account information.   
  
### 2.3.7 Plugin Data Entity   
- \*\*Description\*\*: This data entity stores plugin-related information such as plugin ID, name, description, configuration, code, and documentation.   
- \*\*Interaction Method\*\*: The system retrieves and updates plugin data via database queries or stored procedures.   
- \*\*Inputs/Outputs\*\*:   
 - Input: Plugin ID, updated plugin name, description, configuration, or code.   
 - Output: Updated plugin information and confirmation message.   
  
### 2.3.8 Email Plugin (Third-Party Service)   
- \*\*Description\*\*: A configured third-party email service plugin used to send confirmation emails to customers after successful checkout.   
- \*\*Interaction Method\*\*: The system triggers the plugin via API calls or event-driven mechanisms.   
- \*\*Inputs/Outputs\*\*:   
 - Input: Order details (order number, items, total amount, shipping address).   
 - Output: A confirmation email sent to the customer and a log of the email delivery status.   
  
### 2.3.9 Session Management Module   
- \*\*Description\*\*: A software module responsible for managing user sessions (both customer and administrator).   
- \*\*Interaction Method\*\*: The system interacts with this module to initiate, maintain, or terminate sessions upon login or logout.   
- \*\*Inputs/Outputs\*\*:   
 - Input: Credentials (email and password for customers; username and password for administrators).   
 - Output: Session initiation or termination, along with user authentication status.   
  
## 2.4 Communication Interface Output   
  
The system uses network-based communication to interact with external services and users. The following communication interfaces are required:   
  
### 2.4.1 Email Notification Service   
- \*\*Description\*\*: The system communicates with an external email service plugin to send order confirmation emails to customers.   
- \*\*Interaction Method\*\*: The system sends HTTP requests or uses an API to deliver email content, including order details.   
- \*\*Inputs/Outputs\*\*:   
 - Input: Order details from the Order data entity.   
 - Output: A confirmation email sent to the customer with the order number, items, total amount, and shipping address.   
  
### 2.4.2 Web Browsing (HTTP/HTTPS)   
- \*\*Description\*\*: The system communicates with users through web browsers via HTTP or HTTPS protocols.   
- \*\*Interaction Method\*\*: Standard web protocols are used to load user interfaces and process user input (e.g., form submissions, AJAX requests).   
- \*\*Inputs/Outputs\*\*:   
 - Input: User inputs from web interfaces (e.g., login form, product search, cart updates).   
 - Output: Dynamic content updates and responses to user actions.   
  
### 2.4.3 Session Communication (Web Cookies/Token)   
- \*\*Description\*\*: The system maintains user sessions using cookies or token-based authentication (e.g., JWT).   
- \*\*Interaction Method\*\*: Cookies or tokens are stored in the user’s browser and sent with each request to authenticate the session.   
- \*\*Inputs/Outputs\*\*:   
 - Input: Session cookies or tokens.   
 - Output: Session validation and access control to secure areas of the system.   
  
### 2.4.4 Plugin Documentation Access (Internal Web Service)   
- \*\*Description\*\*: The system provides access to plugin documentation through an internal web service or API.   
- \*\*Interaction Method\*\*: The system retrieves plugin documentation using HTTP requests and displays it within the user interface.   
- \*\*Inputs/Outputs\*\*:   
 - Input: Plugin ID.   
 - Output: Plugin documentation content and access logs.   
  
## Summary of External Interfaces   
  
| Interface Type | Interface Name | Description | Key Interaction Method |  
|----------------|----------------|-------------|------------------------|  
| User Interface | Customer Registration Form | Allows customers to create new accounts | Form submission with validation |  
| User Interface | Customer Login Page | Authenticates customer credentials and initiates a session | Form submission with session creation |  
| User Interface | Product Detail Page | Displays product information and updates view count | Dynamic data loading based on product ID |  
| User Interface | Shopping Cart Interface | Allows customers to manage their cart | Cart updates and confirmation messages |  
| User Interface | Checkout Form | Processes order details and payment method | Form submission with order creation |  
| User Interface | Purchase History Page | Displays customer order history | Data retrieval from Order data entity |  
| User Interface | Administrator Login Page | Authenticates administrator credentials | Form submission with session creation |  
| User Interface | Admin Dashboard Interface | Central control for administrators | Dynamic data loading and updates |  
| Software Interface | Product Data Entity | Stores and retrieves product information | Database queries |  
| Software Interface | Inventory Data Entity | Tracks product stock levels | Database queries |  
| Software Interface | Order Data Entity | Stores and retrieves order information | Database queries |  
| Software Interface | Payment Data Entity | Tracks payment status and details | Database queries |  
| Software Interface | Customer Data Entity | Stores and retrieves customer account details | Database queries |  
| Software Interface | Administrator Data Entity | Stores and retrieves administrator account details | Database queries |  
| Software Interface | Plugin Data Entity | Stores and retrieves plugin information | Database queries |  
| Communication Interface | Email Notification Service | Sends order confirmation emails | HTTP API or event-based trigger |  
| Communication Interface | Web Browsing (HTTP/HTTPS) | Loads and processes web-based user interfaces | Standard web protocols |  
| Communication Interface | Session Communication | Maintains user authentication state | Cookie or token-based authentication |  
| Communication Interface | Plugin Documentation Access | Retrieves plugin documentation | Internal web service call |   
  
This section ensures that all external interfaces required for the system are clearly defined, making it easier for developers to implement and integrate the system components.