# Functional Requirement

# 1. Functional Requirements   
  
## 1.1 Customer Registration Function   
\*\*Function ID:\*\* FR-01   
\*\*Description:\*\* Allows a customer to register an account by providing their personal information. The system validates the input data and sends an email confirmation to activate the account.   
\*\*Input:\*\*   
- Customer name (string)   
- Email address (string, must be valid and unique)   
- Password (string, must meet strength requirements)   
- Optional: Address (string), Phone number (string)   
\*\*Output:\*\*   
- A new customer account in the database with an assigned CustomerID   
- A confirmation email sent to the customer's email address   
- Customer redirected to login page or homepage   
  
---  
  
## 1.2 Customer Login Function   
\*\*Function ID:\*\* FR-02   
\*\*Description:\*\* Enables a customer to log in using their registered email and password. The system authenticates the user and generates a session token to maintain login state.   
\*\*Input:\*\*   
- Customer's email (string)   
- Customer's password (string)   
\*\*Output:\*\*   
- A session token assigned to the customer   
- Customer redirected to dashboard or homepage   
- Login status updated in the system   
  
---  
  
## 1.3 View Product Details Function   
\*\*Function ID:\*\* FR-03   
\*\*Description:\*\* Displays detailed information about a product, including name, description, price, and images. The system logs the customer's view of the product.   
\*\*Input:\*\*   
- ProductID (integer, unique identifier of the product)   
- CustomerID (integer, optional, for tracking views)   
\*\*Output:\*\*   
- Product details displayed on the product page   
- A log entry in the system indicating the customer viewed the product   
- Option to add product to cart   
  
---  
  
## 1.4 Add Product to Cart Function   
\*\*Function ID:\*\* FR-04   
\*\*Description:\*\* Adds a selected product and quantity to the customer's shopping cart. The system checks inventory availability and updates the cart's total items and cost.   
\*\*Input:\*\*   
- ProductID (integer)   
- Quantity (integer, must be greater than 0 and within inventory limits)   
- CustomerID (integer, unique identifier of the customer)   
\*\*Output:\*\*   
- Updated ShoppingCart record in the database with the added item   
- Updated cart display with new total items and cost   
- Confirmation message displayed to the customer   
  
---  
  
## 1.5 Modify Cart Contents Function   
\*\*Function ID:\*\* FR-05   
\*\*Description:\*\* Allows a customer to adjust the quantity of a product in the cart or remove it. The system verifies inventory availability and recalculates the cart total.   
\*\*Input:\*\*   
- CartItemID (integer)   
- New quantity (integer, optional)   
- CustomerID (integer)   
- Action type: "update quantity" or "remove item"   
\*\*Output:\*\*   
- Updated CartItem record in the database   
- Updated ShoppingCart with new total items and cost   
- Confirmation message displayed to the customer   
  
---  
  
## 1.6 Remove Product from Cart Function   
\*\*Function ID:\*\* FR-06   
\*\*Description:\*\* Removes a specific product from the customer's shopping cart. The system recalculates the cart total and confirms the removal.   
\*\*Input:\*\*   
- CartItemID (integer)   
- CustomerID (integer)   
\*\*Output:\*\*   
- Updated ShoppingCart with reduced total items and cost   
- CartItem removed from the database   
- Confirmation message displayed to the customer   
  
---  
  
## 1.7 Checkout Function   
\*\*Function ID:\*\* FR-07   
\*\*Description:\*\* Processes the customer's cart to create an order and handle payment. The system updates inventory, clears the cart, and sends an email confirmation.   
\*\*Input:\*\*   
- CustomerID (integer)   
- Shipping address (string)   
- Billing address (string)   
- Payment method (string, e.g., credit card, PayPal)   
- Cart content (list of CartItems)   
\*\*Output:\*\*   
- A new Order record in the database with OrderID and associated OrderItems   
- Inventory updated with reduced stock for purchased items   
- Payment record created with PaymentID and transaction details   
- Email confirmation sent to customer   
- Shopping cart cleared of the purchased items   
  
---  
  
## 1.8 Confirm Order Function   
\*\*Function ID:\*\* FR-08   
\*\*Description:\*\* Finalizes an order after checkout by confirming the payment and updating the order status. The system ensures inventory and payment are processed correctly.   
\*\*Input:\*\*   
- OrderID (integer)   
- CustomerID (integer)   
- Payment confirmation status (boolean)   
- Inventory status (boolean)   
\*\*Output:\*\*   
- Order status updated to "confirmed" or "completed"   
- Inventory updated with reduced stock   
- Payment status updated to "processed"   
- Email confirmation sent to the customer   
- Customer redirected to order confirmation page   
  
---  
  
## 1.9 View Purchase History Function   
\*\*Function ID:\*\* FR-09   
\*\*Description:\*\* Displays a list of all previous orders made by a customer, along with order details such as items, total cost, and status. The system allows the customer to request an email confirmation for a specific order.   
\*\*Input:\*\*   
- CustomerID (integer)   
- Optional: OrderID (integer, for viewing a specific order)   
\*\*Output:\*\*   
- List of orders displayed in the "Purchase History" section   
- Order details (products, quantities, prices, status) retrieved and displayed   
- Email confirmation sent to the customer (if requested)   
- Log entry indicating access to purchase history   
  
---  
  
## 1.10 Administrator Login Function   
\*\*Function ID:\*\* FR-10   
\*\*Description:\*\* Authenticates an administrator using their username and password, and grants access to the admin dashboard. The system generates a session token and logs the login attempt.   
\*\*Input:\*\*   
- Admin username (string)   
- Admin password (string)   
\*\*Output:\*\*   
- A session token assigned to the administrator   
- Administrator redirected to the admin dashboard   
- Login status updated in the system   
  
---  
  
## 1.11 Manage Inventory Function   
\*\*Function ID:\*\* FR-11   
\*\*Description:\*\* Enables the administrator to add, edit, or remove products from the inventory. The system ensures that all associated data (e.g., category, stock levels) are synchronized and logs the action.   
\*\*Input:\*\*   
- ProductID (integer, optional)   
- Product name (string)   
- Description (string)   
- Price (decimal)   
- QuantityInStock (integer)   
- CategoryID (integer)   
- Action type: "add", "edit", or "remove"   
\*\*Output:\*\*   
- Updated Inventory record in the database   
- Product details updated or removed   
- Plugin triggered (if applicable) to notify stakeholders   
- Confirmation message displayed to the administrator   
  
---  
  
## 1.12 Update Product Information Function   
\*\*Function ID:\*\* FR-12   
\*\*Description:\*\* Allows the administrator to modify product details such as name, description, price, and category. The system validates the input and updates the inventory accordingly.   
\*\*Input:\*\*   
- ProductID (integer)   
- New product name (string, optional)   
- New description (string, optional)   
- New price (decimal, optional)   
- New category ID (integer, optional)   
- New stock level (integer, optional)   
- Action type: "update"   
\*\*Output:\*\*   
- Updated Product record in the database   
- Updated Inventory record if stock level changed   
- Plugin triggered (if applicable) to notify stakeholders   
- Confirmation message displayed to the administrator   
  
---  
  
## 1.13 Manage Categories Function   
\*\*Function ID:\*\* FR-13   
\*\*Description:\*\* Allows the administrator to create, edit, or delete product categories. The system ensures that product records are updated if a category is changed and logs the action.   
\*\*Input:\*\*   
- CategoryID (integer, optional)   
- Category name (string)   
- Description (string, optional)   
- ParentCategoryID (integer, optional)   
- Action type: "add", "edit", or "delete"   
\*\*Output:\*\*   
- Updated Category record in the database   
- Updated Product records if their category is changed   
- Plugin triggered (if applicable) to notify stakeholders   
- Confirmation message displayed to the administrator   
  
---  
  
## 1.14 Plugin Installation Function   
\*\*Function ID:\*\* FR-14   
\*\*Description:\*\* Enables the administrator to install and activate plugins that extend system functionality. The system validates plugin compatibility and updates the configuration.   
\*\*Input:\*\*   
- Plugin name (string)   
- Plugin version (string)   
- Plugin description (string)   
- Plugin file (compatible plugin package)   
- Action type: "install"   
\*\*Output:\*\*   
- Plugin installed and activated   
- Plugin record added to the Plugin table   
- System configuration updated   
- Confirmation message displayed to the administrator   
  
---  
  
## 1.15 Email Confirmation Function   
\*\*Function ID:\*\* FR-15   
\*\*Description:\*\* Sends a confirmation email to the customer for specific actions such as registration, order confirmation, or account verification. The system logs the email sending action and confirms delivery.   
\*\*Input:\*\*   
- CustomerID (integer)   
- OrderID (integer, optional)   
- Action type: "registration", "order confirmation", or "account verification"   
- Email template (string, dynamically generated based on action)   
\*\*Output:\*\*   
- EmailConfirmation record created in the database   
- Confirmation email sent to the customer   
- System logs the email sending attempt   
- Customer redirected to confirmation page or shown success message   
  
---  
  
## 1.16 View Order Status Function   
\*\*Function ID:\*\* FR-16   
\*\*Description:\*\* Displays the current status of a customer’s order (e.g., pending, shipped, delivered). The system retrieves the latest order data and provides options for further actions if applicable.   
\*\*Input:\*\*   
- OrderID (integer)   
- CustomerID (integer)   
\*\*Output:\*\*   
- Order status displayed (e.g., "pending", "shipped", "delivered")   
- Order details (products, quantities, dates) displayed   
- Option to cancel or contact support (if applicable)   
- Log entry for order status access   
  
---  
  
## 1.17 Manage Cart Item Function   
\*\*Function ID:\*\* FR-17   
\*\*Description:\*\* Allows a customer to modify individual items in the shopping cart, including updating quantity, replacing product, or removing the item. The system recalculates the cart total and validates against inventory.   
\*\*Input:\*\*   
- CartItemID (integer)   
- New quantity (integer, optional)   
- New ProductID (integer, optional)   
- CustomerID (integer)   
- Action type: "update", "replace", or "remove"   
\*\*Output:\*\*   
- Updated CartItem record in the database   
- Updated ShoppingCart with new total items and cost   
- Confirmation message displayed to the customer   
- Inventory status updated if replacement product is added   
  
---  
  
## 1.18 Manage Order Item Function   
\*\*Function ID:\*\* FR-18   
\*\*Description:\*\* Enables the administrator to modify individual items in an order, such as changing quantity or replacing a product. The system updates the order and inventory accordingly and may notify the customer.   
\*\*Input:\*\*   
- OrderItemID (integer)   
- New quantity (integer, optional)   
- New ProductID (integer, optional)   
- OrderID (integer)   
- AdministratorID (integer)   
- Action type: "update", "replace", or "remove"   
\*\*Output:\*\*   
- Updated OrderItem record in the database   
- Updated Order with new total cost and items   
- Inventory updated if product stock is affected   
- Email notification sent to customer if order item is modified   
- Log entry of the administrator's action   
  
---  
  
## 1.19 Manage Product Update Record Function   
\*\*Function ID:\*\* FR-19   
\*\*Description:\*\* Allows the administrator to view, edit, or delete records of product updates. The system ensures data integrity and logs any modifications or deletions.   
\*\*Input:\*\*   
- UpdateRecordID (integer)   
- ProductID (integer)   
- AdminID (integer)   
- OldInfo (string or JSON, optional)   
- NewInfo (string or JSON, optional)   
- Action type: "view", "edit", or "delete"   
\*\*Output:\*\*   
- Updated or deleted ProductUpdateRecord in the database   
- Log entry of the administrator's action   
- Plugin triggered (if applicable) to notify stakeholders   
- Confirmation message displayed to the administrator   
  
---  
  
以上是 \*\*GAMMA-J Web Store\*\* 系统中 \*\*功能需求（Functional Requirements）\*\* 的完整描述。每个功能均基于提供的用例和数据模型提取并定义，确保输入输出清晰、可实现，并与系统上下文一致。

# External Description

# 2. External Interfaces  
  
This chapter describes the external interfaces of the \*\*GAMMA-J Web Store\*\* system. These interfaces define the system's interactions with external entities such as users, hardware, software, and communication channels. The interfaces are derived from the functional requirements and are organized into four categories: \*\*User Interfaces\*\*, \*\*Hardware Interfaces\*\*, \*\*Software Interfaces\*\*, and \*\*Communication Interfaces\*\*.  
  
---  
  
## 2.1 User Interface Output  
  
The system interacts with users through web-based interfaces. These interfaces include web pages, buttons, input fields, and modal windows that allow customers and administrators to perform actions and receive visual feedback.  
  
### 2.1.1 Customer Registration Page   
- \*\*Description:\*\* A web page where customers can provide their personal information to register a new account.  
- \*\*Inputs:\*\* Customer name, email address, password, optional address and phone number.  
- \*\*Outputs:\*\*   
 - Success or error messages based on input validation.  
 - Confirmation message and redirection to the login page or homepage.  
 - Email confirmation sent to the user (see 2.4.1).  
  
### 2.1.2 Customer Login Page   
- \*\*Description:\*\* A web page where registered customers can log in using their email and password.  
- \*\*Inputs:\*\* Email and password.  
- \*\*Outputs:\*\*   
 - Success or error messages.  
 - Redirection to the dashboard or homepage upon successful login.  
  
### 2.1.3 Product Detail Page   
- \*\*Description:\*\* A web page that displays detailed product information including name, description, price, and images.  
- \*\*Inputs:\*\* ProductID and optionally CustomerID for tracking.  
- \*\*Outputs:\*\*   
 - Product information displayed to the user.  
 - Log entry in the system (see 2.4.2).  
 - Option to add the product to the cart.  
  
### 2.1.4 Shopping Cart Interface   
- \*\*Description:\*\* A web interface where customers can view and modify their selected items for purchase.  
- \*\*Inputs:\*\* ProductID, Quantity, CustomerID for adding items; CartItemID, new quantity, or action type for modifying items.  
- \*\*Outputs:\*\*   
 - Updated cart display with total items and cost.  
 - Confirmation messages for add, update, or remove actions.  
 - Inventory status updates if applicable.  
  
### 2.1.5 Checkout Page   
- \*\*Description:\*\* A web page where customers can review their cart and enter shipping and billing details to complete the purchase.  
- \*\*Inputs:\*\* CustomerID, shipping address, billing address, payment method, and cart content.  
- \*\*Outputs:\*\*   
 - Order confirmation message.  
 - Redirection to the order confirmation page.  
 - Email confirmation sent to the customer (see 2.4.3).  
  
### 2.1.6 Order Confirmation Page   
- \*\*Description:\*\* A web page displayed to the customer after a successful checkout and payment confirmation.  
- \*\*Inputs:\*\* OrderID and CustomerID.  
- \*\*Outputs:\*\*   
 - Order status confirmation (e.g., confirmed or completed).  
 - Summary of purchased items and total cost.  
 - Redirection to the homepage or purchase history.  
  
### 2.1.7 Purchase History Page   
- \*\*Description:\*\* A web page where customers can view all their previous orders with details such as items, total cost, and status.  
- \*\*Inputs:\*\* CustomerID and optionally OrderID.  
- \*\*Outputs:\*\*   
 - List of orders with their details displayed.  
 - Option to request an email confirmation for a specific order.  
 - Log entry indicating access (see 2.4.4).  
  
### 2.1.8 Admin Login Page   
- \*\*Description:\*\* A web page where administrators can log in using their username and password.  
- \*\*Inputs:\*\* Admin username and password.  
- \*\*Outputs:\*\*   
 - Session token assigned to the admin.  
 - Redirection to the admin dashboard.  
 - Log entry of login attempt (see 2.4.5).  
  
### 2.1.9 Admin Dashboard Interface   
- \*\*Description:\*\* A web interface for administrators to manage inventory, product information, categories, and order items.  
- \*\*Inputs:\*\* ProductID, CategoryID, OrderItemID, etc., along with new information and action types.  
- \*\*Outputs:\*\*   
 - Confirmation messages for admin actions (e.g., product or category updates).  
 - Updated data displayed in real time or after refresh.  
 - Plugin notification if triggered (see 2.4.6).  
  
### 2.1.10 Order Status Page   
- \*\*Description:\*\* A web page where customers can view the current status of a specific order.  
- \*\*Inputs:\*\* OrderID and CustomerID.  
- \*\*Outputs:\*\*   
 - Order status (e.g., pending, shipped, delivered).  
 - Order details displayed.  
 - Option to cancel or contact support if applicable.  
 - Log entry of access (see 2.4.7).  
  
---  
  
## 2.2 Hardware Interface Output  
  
The \*\*GAMMA-J Web Store\*\* system does not require direct interaction with specific hardware devices for its core functionalities. However, the following hardware-related considerations may apply to the system's operation:  
  
### 2.2.1 Web Browsers   
- \*\*Description:\*\* The system must be compatible with standard web browsers such as Chrome, Firefox, Safari, and Edge.  
- \*\*Interaction Method:\*\* HTML5, CSS3, JavaScript, and responsive design principles.  
- \*\*Purpose:\*\* Ensures users can access and interact with the system regardless of the browser they are using.  
  
---  
  
## 2.3 Software Interface Output  
  
The system interacts with various software components, including databases, plugins, and internal modules, to provide functionality and maintain data integrity.  
  
### 2.3.1 Customer Database   
- \*\*Description:\*\* A database storing customer account information such as CustomerID, name, email, password (hashed), address, and phone number.  
- \*\*Interaction Method:\*\* SQL queries or ORM (Object-Relational Mapping) tools.  
- \*\*Purpose:\*\* Provides persistent storage for customer data and supports registration, login, and purchase history functions.  
  
### 2.3.2 Product and Inventory Database   
- \*\*Description:\*\* A database storing product details (ProductID, name, description, price, image URLs) and inventory information (QuantityInStock).  
- \*\*Interaction Method:\*\* SQL queries or ORM tools.  
- \*\*Purpose:\*\* Ensures accurate tracking of product availability and supports product management, cart updates, and order processing.  
  
### 2.3.3 Order and Payment Database   
- \*\*Description:\*\* A database storing order records (OrderID, CustomerID, status, date) and payment records (PaymentID, transaction details, amount).  
- \*\*Interaction Method:\*\* SQL queries or ORM tools.  
- \*\*Purpose:\*\* Manages order lifecycle and payment processing, supporting checkout, order confirmation, and purchase history.  
  
### 2.3.4 Session Management Module   
- \*\*Description:\*\* A module responsible for generating and managing session tokens for both customers and administrators.  
- \*\*Interaction Method:\*\* Internal API calls or direct database access.  
- \*\*Purpose:\*\* Maintains user authentication state and supports secure access to user-specific data.  
  
### 2.3.5 Plugin Framework   
- \*\*Description:\*\* A software framework allowing administrators to install and activate plugins to extend system functionality.  
- \*\*Interaction Method:\*\* Plugin installation via file upload and activation through admin interface.  
- \*\*Purpose:\*\* Enhances system flexibility and supports notifications or additional features.  
  
---  
  
## 2.4 Communication Interface Output  
  
The \*\*GAMMA-J Web Store\*\* system communicates with external systems and services through various communication channels such as email and logging mechanisms.  
  
### 2.4.1 Email Confirmation Service   
- \*\*Description:\*\* An external email service used to send confirmation emails to customers for account registration, order confirmation, and account verification.  
- \*\*Interaction Method:\*\* SMTP or REST API to send emails, with dynamic email templates.  
- \*\*Inputs:\*\*   
 - CustomerID  
 - OrderID (optional)  
 - Action type (e.g., "registration", "order confirmation", "account verification")  
- \*\*Outputs:\*\*   
 - Email confirmation sent to the customer.  
 - EmailConfirmation record added to the database.  
 - System logs the email sending attempt.  
  
### 2.4.2 Inventory Logging Service   
- \*\*Description:\*\* A logging system that records when a product is viewed or when inventory is modified.  
- \*\*Interaction Method:\*\* Internal logging framework or external logging service.  
- \*\*Inputs:\*\*   
 - ProductID  
 - CustomerID (optional)  
 - Action type (e.g., "view", "add", "remove")  
- \*\*Outputs:\*\*   
 - Log entry created with timestamp and action details.  
  
### 2.4.3 Order Status Notification System   
- \*\*Description:\*\* A communication module that provides updates on order status to customers and logs these interactions.  
- \*\*Interaction Method:\*\* Internal system calls and email notifications.  
- \*\*Inputs:\*\*   
 - OrderID  
 - CustomerID  
- \*\*Outputs:\*\*   
 - Order status displayed to the customer.  
 - Email notification sent if the order is modified.  
 - Log entry of access or modification.  
  
### 2.4.4 Admin Action Logging System   
- \*\*Description:\*\* A system that logs all administrator actions, such as inventory management, product updates, and plugin installation.  
- \*\*Interaction Method:\*\* Internal logging system or external audit logging service.  
- \*\*Inputs:\*\*   
 - AdminID  
 - Action type (e.g., "add product", "delete category", "install plugin")  
- \*\*Outputs:\*\*   
 - Log entry created with timestamp, admin ID, and action details.  
  
### 2.4.5 Plugin Notification System   
- \*\*Description:\*\* A communication system that notifies stakeholders or external services when a plugin is installed or activated.  
- \*\*Interaction Method:\*\* REST API or message queue (e.g., RabbitMQ).  
- \*\*Inputs:\*\*   
 - Plugin name  
 - Plugin version  
 - Plugin description  
- \*\*Outputs:\*\*   
 - Plugin status updated in the system.  
 - Notification sent to stakeholders.  
 - Confirmation message displayed to the admin.  
  
---  
  
## Summary of External Interfaces  
  
| Interface Name | Type | Description |  
|------------------------------------|---------------------|-----------------------------------------------------------------------------|  
| Customer Registration Page | User Interface | Web page for user registration with input validation and email confirmation |  
| Customer Login Page | User Interface | Web page for logging in with authentication and session token generation |  
| Product Detail Page | User Interface | Displays product details and logs user views |  
| Shopping Cart Interface | User Interface | Allows users to add, update, or remove items from the cart |  
| Checkout Page | User Interface | Process order and payment with confirmation and email notification |  
| Order Confirmation Page | User Interface | Displays final order confirmation details |  
| Purchase History Page | User Interface | Displays a list of past orders and allows email reconfirmation |  
| Admin Login Page | User Interface | Secure login for administrators |  
| Admin Dashboard Interface | User Interface | Allows admins to manage products, categories, and orders |  
| Order Status Page | User Interface | Displays current status and details of a specific order |  
| Web Browsers | Hardware Interface | Compatibility with standard web browsers for user access |  
| Customer Database | Software Interface | Stores and retrieves customer account data |  
| Product and Inventory Database | Software Interface | Stores and updates product and stock information |  
| Order and Payment Database | Software Interface | Stores and updates order and payment records |  
| Session Management Module | Software Interface | Manages session tokens for user authentication |  
| Plugin Framework | Software Interface | Supports plugin installation and activation |  
| Email Confirmation Service | Communication Interface | Sends confirmation emails to customers |  
| Inventory Logging Service | Communication Interface | Logs product view and inventory modification events |  
| Order Status Notification System | Communication Interface | Sends order status updates to customers |  
| Admin Action Logging System | Communication Interface | Logs admin actions for audit and traceability |  
| Plugin Notification System | Communication Interface | Notifies stakeholders of plugin-related changes |  
  
---  
  
This section ensures that all external interfaces required for the \*\*GAMMA-J Web Store\*\* system are clearly defined and consistent with the functional requirements. Developers can use this section as a reference for implementing and integrating the system with external components.