项目文档

# Functional Requirement

# 1. Functional Requirements  
  
## 1.1 Customer Management Function   
Function ID: FR-01   
Description: Administrators can create, manage, view, and delete customer records. This includes editing customer information and checking for existing records.   
Input: Customer details (name, email, address, phone number), customer ID for management operations, confirmation inputs for deletion.   
Output: Updated or newly created customer records in the database, confirmation messages, error messages for invalid operations.  
  
## 1.2 Product Management Function   
Function ID: FR-02   
Description: Administrators can add, update, and delete product records. The system ensures product details are valid and checks for existing products.   
Input: Product details (name, description, price, stock, category), product ID for update and deletion, confirmation inputs for deletion.   
Output: Updated or newly added product records in the database, confirmation messages, error messages for invalid operations.  
  
## 1.3 Product Catalog Display Function   
Function ID: FR-03   
Description: The system displays a list of available products to users, with the ability to search and filter using built-in or plugin-based functionality.   
Input: User search or filter criteria, product data from the database.   
Output: Displayed product catalog with filtered or full product list, error messages for invalid input, warning for plugin unavailability.  
  
## 1.4 Order Placement Function   
Function ID: FR-04   
Description: Customers can place orders by submitting products in their cart, selecting a payment method, and confirming the order. The system validates cart data and processes payments.   
Input: Customer-selected products, payment method, confirmation to place order.   
Output: A new order record in the database, updated product stock, a new payment record, a notification to the customer, and a confirmation message.  
  
## 1.5 Order Management Function   
Function ID: FR-05   
Description: Administrators or customers can modify existing orders, including product quantities or shipping details. The system recalculates totals and updates payment and notification status as needed.   
Input: Selected order, modified order details (product quantities, shipping address), confirmation for modifications.   
Output: Updated order record in the database, updated payment status, a notification to the customer, and a confirmation message.  
  
## 1.6 Order Cancellation Function   
Function ID: FR-06   
Description: Customers or administrators can cancel an order if it is in a cancellable state. The system updates the order status and initiates a refund if a payment has been made.   
Input: Selected order, confirmation for cancellation.   
Output: Updated order status (canceled), refund or voided payment if applicable, a notification to the customer, and a confirmation message.  
  
## 1.7 Payment Processing Function   
Function ID: FR-07   
Description: The system processes payments for orders using available payment methods. It validates the payment method, calculates fees, and updates the order and payment status accordingly.   
Input: Selected order, selected payment method, plugin-based validation and calculation data.   
Output: Updated payment status in the database, updated order status, a notification to the customer, and a confirmation message.  
  
## 1.8 Payment Status View Function   
Function ID: FR-08   
Description: Customers or administrators can view the status of a payment associated with an order. The system retrieves and displays the current payment status.   
Input: Selected order or payment, user authentication status.   
Output: Displayed payment status (e.g., pending, completed, failed), a notification if the status has changed, and error messages for technical failures.  
  
## 1.9 Payment Record Deletion Function   
Function ID: FR-09   
Description: Administrators can delete a payment record from the system. The system checks for dependencies and confirms the deletion.   
Input: Selected payment record, confirmation for deletion.   
Output: Deleted payment record from the database, confirmation message, and error messages for invalid deletions.  
  
## 1.10 Order Detail Management Function   
Function ID: FR-10   
Description: Administrators or customers can add, edit, or remove order details (itemized product information) for an order. The system recalculates totals if needed and sends notifications for changes.   
Input: Selected order, modified or new order detail information (product ID, quantity, unit price), confirmation for modifications.   
Output: Updated order detail records in the database, updated order total, a notification to the customer if needed, and a confirmation message.  
  
## 1.11 Plugin Management Function   
Function ID: FR-11   
Description: Administrators can install, update, and uninstall plugins. The system validates plugin compatibility and manages configurations.   
Input: Plugin file or identifier, confirmation for uninstallation or update.   
Output: Installed, updated, or uninstalled plugin in the system, a notification to the administrator, and a confirmation message.  
  
## 1.12 Notification Management Function   
Function ID: FR-12   
Description: Administrators can send, view, and delete notifications. The system ensures notifications are delivered to the correct recipients and logs the activity.   
Input: Notification content, recipient information, notification ID for viewing or deletion, confirmation for deletion.   
Output: Sent or deleted notification, displayed notification list, a log of notification attempts, and a confirmation message.  
  
## 1.13 Administrator Management Function   
Function ID: FR-13   
Description: Administrators can manage other administrator accounts, including editing roles and permissions or removing accounts. The system ensures at least one administrator remains active.   
Input: Selected administrator, modified administrator details (role, username), confirmation for removal.   
Output: Updated or deleted administrator records, confirmation messages, and error messages for invalid operations.  
  
## 1.14 Administrator Log Viewing Function   
Function ID: FR-14   
Description: Administrators can view logs of previous administrative actions. The system retrieves and displays logs in a structured format with filtering options.   
Input: Log filter or search criteria, request to view logs.   
Output: Displayed administrator log entries, error messages for invalid input, and a prompt if no logs are selected.  
  
## 1.15 Order History Viewing Function   
Function ID: FR-15   
Description: Customers or administrators can view the history of orders placed by a customer. The system retrieves and displays the order details without making any changes.   
Input: Selected customer or order, request to view order history.   
Output: Displayed order history with details, error messages if the order is not found, and a prompt if no order is selected.

# External Description

# 2. External Interfaces  
  
This chapter defines the external interfaces of the system, including user interfaces, hardware interfaces, software interfaces, and communication interfaces. These interfaces describe how the system interacts with external entities such as users, hardware, other software systems, and communication protocols.  
  
## 2.1 User Interface  
  
The system provides a user interface for both administrators and customers to interact with the functionalities. The interface includes forms, buttons, tables, and modals to manage customer records, product records, orders, and notifications.  
  
### 2.1.1 Customer Management UI   
- \*\*Description\*\*: A form-based interface for administrators to create, view, edit, and delete customer records.   
- \*\*Inputs\*\*: Customer details (name, email, address, phone number), customer ID, and confirmation inputs for deletion.   
- \*\*Outputs\*\*: Updated or newly created customer records, confirmation messages, and error messages.   
  
### 2.1.2 Product Management UI   
- \*\*Description\*\*: A form-based interface for administrators to add, update, and delete product records.   
- \*\*Inputs\*\*: Product details (name, description, price, stock, category), product ID, and confirmation inputs for deletion.   
- \*\*Outputs\*\*: Updated or newly added product records, confirmation messages, and error messages.   
  
### 2.1.3 Product Catalog UI   
- \*\*Description\*\*: A searchable and filterable interface for users to browse available products.   
- \*\*Inputs\*\*: Search or filter criteria provided by the user.   
- \*\*Outputs\*\*: A list of products based on the search or filter, error messages for invalid input, and a warning if plugins are unavailable.   
  
### 2.1.4 Order Placement UI   
- \*\*Description\*\*: A user interface for customers to submit their orders, select a payment method, and confirm the transaction.   
- \*\*Inputs\*\*: Selected products, payment method, and confirmation to place order.   
- \*\*Outputs\*\*: A confirmation message, a summary of the order, and a notification to the customer.   
  
### 2.1.5 Order Management UI   
- \*\*Description\*\*: A form-based interface for administrators or customers to modify existing orders, such as adjusting product quantities or updating shipping details.   
- \*\*Inputs\*\*: Selected order, modified order details, and confirmation for modifications.   
- \*\*Outputs\*\*: Updated order details, confirmation messages, and notifications to the customer.   
  
### 2.1.6 Order Cancellation UI   
- \*\*Description\*\*: An interface for administrators or customers to cancel orders that are in a cancellable state.   
- \*\*Inputs\*\*: Selected order and confirmation for cancellation.   
- \*\*Outputs\*\*: Updated order status, confirmation messages, and notifications for cancellations or refunds.   
  
### 2.1.7 Payment Status View UI   
- \*\*Description\*\*: A display interface for users to view the current status of a payment associated with an order.   
- \*\*Inputs\*\*: Selected order or payment.   
- \*\*Outputs\*\*: Displayed payment status (e.g., pending, completed, failed), notifications for status changes, and error messages for technical failures.   
  
### 2.1.8 Notification Management UI   
- \*\*Description\*\*: A dashboard interface for administrators to send, view, and delete notifications.   
- \*\*Inputs\*\*: Notification content, recipient information, notification ID, and confirmation for deletion.   
- \*\*Outputs\*\*: Sent or deleted notifications, displayed notification list, and a log of notification attempts.   
  
### 2.1.9 Administrator Management UI   
- \*\*Description\*\*: A user interface for administrators to manage other administrator accounts, including editing roles and permissions or removing accounts.   
- \*\*Inputs\*\*: Selected administrator, modified details (role, username), and confirmation for removal.   
- \*\*Outputs\*\*: Updated or deleted administrator records, confirmation messages, and error messages for invalid operations.   
  
### 2.1.10 Administrator Log Viewing UI   
- \*\*Description\*\*: A structured interface for administrators to view and filter logs of previous administrative actions.   
- \*\*Inputs\*\*: Log filter or search criteria and request to view logs.   
- \*\*Outputs\*\*: Displayed log entries, error messages for invalid input, and a prompt if no logs are selected.   
  
### 2.1.11 Order History Viewing UI   
- \*\*Description\*\*: A read-only interface for administrators or customers to view the history of orders placed by a specific customer.   
- \*\*Inputs\*\*: Selected customer or order and a request to view order history.   
- \*\*Outputs\*\*: Displayed order history with detailed information, error messages if the order is not found, and a prompt if no order is selected.   
  
## 2.2 Hardware Interface  
  
The system does not require direct interaction with any specific hardware devices. All hardware dependencies are abstracted through the software and communication interfaces. Therefore, no hardware interface is defined for this system.  
  
## 2.3 Software Interface  
  
The system interacts with various software components, including the database and external plugins used for payment processing and product catalog filtering.  
  
### 2.3.1 Database Interface   
- \*\*Description\*\*: The system communicates with a relational database to store and retrieve data related to customers, products, orders, and notifications.   
- \*\*Inputs\*\*: Data to be stored (e.g., customer details, product information, order records), queries to retrieve data (e.g., order history, product catalog).   
- \*\*Outputs\*\*: Updated or retrieved records in the database, error messages for invalid database operations.   
- \*\*Interaction Method\*\*: SQL queries or ORM (Object-Relational Mapping) commands are used to interact with the database.   
  
### 2.3.2 Payment Plugin Interface   
- \*\*Description\*\*: The system supports plugin-based payment processing. Administrators can install, update, or uninstall payment plugins.   
- \*\*Inputs\*\*: Selected payment method, plugin file or identifier, confirmation for uninstallation or update.   
- \*\*Outputs\*\*: Updated payment method status, installed or uninstalled plugin, confirmation messages, and error messages for invalid plugin operations.   
- \*\*Interaction Method\*\*: The system uses a plugin architecture to load and execute external payment processing plugins.   
  
### 2.3.3 Product Catalog Plugin Interface   
- \*\*Description\*\*: The system supports plugin-based functionality for product catalog filtering and searching.   
- \*\*Inputs\*\*: User-defined filter or search criteria, plugin-based logic for filtering.   
- \*\*Outputs\*\*: Filtered product catalog results, warnings for plugin unavailability, and error messages for invalid filtering logic.   
- \*\*Interaction Method\*\*: The system dynamically loads plugins to perform filtering and searching based on user input.   
  
## 2.4 Communication Interface  
  
The system uses communication interfaces to interact with external systems or users through network-based methods.  
  
### 2.4.1 Email Notification Interface   
- \*\*Description\*\*: The system sends email notifications to users regarding order status, payment confirmation, and other relevant updates.   
- \*\*Inputs\*\*: Notification content and recipient email addresses.   
- \*\*Outputs\*\*: Email sent to the recipient, a log of the notification attempt, and confirmation message for the administrator.   
- \*\*Interaction Method\*\*: The system uses an email service API to send messages via SMTP or other email protocols.   
  
### 2.4.2 Web Communication Interface   
- \*\*Description\*\*: The system is accessible through web browsers, and all user interactions occur via HTTP/HTTPS protocols.   
- \*\*Inputs\*\*: User requests from the web browser, including form submissions and API calls.   
- \*\*Outputs\*\*: Web page responses, JSON or XML data for API calls, and error messages for invalid requests.   
- \*\*Interaction Method\*\*: RESTful API or web pages are served through a web server using HTTP/HTTPS.   
  
### 2.4.3 Plugin Communication Interface   
- \*\*Description\*\*: The system communicates with external plugins to perform additional functionalities, such as payment processing and filtering.   
- \*\*Inputs\*\*: Plugin identifiers, configuration data, and execution commands.   
- \*\*Outputs\*\*: Plugin responses, updated system state, and error messages if the plugin fails.   
- \*\*Interaction Method\*\*: The system uses a defined plugin architecture to load, execute, and communicate with plugins using standard data formats (e.g., JSON).   
  
---  
  
This section has defined all external interfaces required by the system, ensuring consistency with the functional requirements and providing clear descriptions of each interface and its role. Developers can use this information to correctly implement the system and its interactions with external entities.

# Use Case

Use Case Name: Create Customer   
Use Case ID: UC-01   
Actors: Administrator   
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to manage customer data.   
  
Postconditions:   
- A new customer record is created in the system.   
- The system provides confirmation of the successful creation.   
  
Main Flow:   
1. The Administrator navigates to the customer management section of the system.   
2. The Administrator selects the "Create Customer" option.   
3. The system displays a form for entering customer details (e.g., name, email, address, contact information).   
4. The Administrator fills in the required customer information.   
5. The Administrator submits the form.   
6. The system validates the input data (e.g., checks for missing fields, valid email format).   
7. The system creates a new Customer entity with the provided details.   
8. The system assigns a unique identifier to the new customer.   
9. The system displays a confirmation message indicating the customer was successfully created.   
  
Alternative Flow:   
1. If the Administrator leaves a required field empty, the system displays an error message prompting them to fill in the missing information.   
2. If the email format is invalid, the system displays an error message and prevents submission until a valid email is provided.   
3. If a customer with the same unique identifier already exists, the system displays a message indicating the customer already exists and suggests editing the existing record instead.  
  
Use Case Name: Manage Customer Information   
Use Case ID: UC-02   
Actors: Administrator   
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to manage customer data.   
- At least one customer record exists in the system.   
  
Postconditions:   
- The customer information is updated or deleted in the system.   
- The system provides confirmation of the successful update or deletion.   
  
Main Flow:   
1. The Administrator navigates to the customer management section of the system.   
2. The Administrator selects the "Manage Customer" option.   
3. The system displays a list of existing customer records.   
4. The Administrator selects a specific customer to edit or delete.   
5. If editing, the system displays the customer's current information in an editable form.   
6. The Administrator modifies the necessary customer details.   
7. The Administrator submits the updated information.   
8. The system validates the updated data.   
9. The system updates the Customer entity with the new information.   
10. The system displays a confirmation message indicating the customer information was successfully updated.   
11. If deleting, the system prompts the Administrator to confirm the deletion.   
12. The Administrator confirms the deletion.   
13. The system removes the selected Customer entity from the database.   
14. The system displays a confirmation message indicating the customer was successfully deleted.   
  
Alternative Flow:   
1. If the Administrator attempts to delete a customer who has associated Orders or Payments, the system displays a warning and prevents deletion until these associations are resolved.   
2. If the Administrator enters invalid data during an update (e.g., incorrect email format), the system displays an error message and prompts for valid information.   
3. If the Administrator cancels the deletion confirmation, the system returns to the customer list without making any changes.   
4. If no customer is selected, the system displays an error message prompting the Administrator to choose a customer to manage.  
  
Use Case Name: View Customer Profile   
Use Case ID: UC-03   
Actors: Administrator   
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to view customer data.   
- At least one customer record exists in the system.   
  
Postconditions:   
- The Administrator can view the detailed information of a selected customer.   
- No changes are made to the customer data.   
  
Main Flow:   
1. The Administrator navigates to the customer management section of the system.   
2. The Administrator selects the "View Customer Profile" option.   
3. The system displays a list of existing customer records.   
4. The Administrator selects a specific customer to view.   
5. The system retrieves the selected Customer entity from the database.   
6. The system displays the customer's detailed information (e.g., name, email, address, contact information).   
  
Alternative Flow:   
1. If the selected customer has no record in the database, the system displays an error message indicating the customer does not exist.   
2. If the Administrator does not select a customer, the system displays a prompt to choose a customer from the list.   
3. If the system fails to retrieve the customer profile due to a technical issue, an error message is displayed, and the Administrator is redirected to the customer list.  
  
Use Case Name: Delete Customer   
Use Case ID: UC-04   
Actors: Administrator   
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to manage customer data.   
- At least one customer record exists in the system.   
  
Postconditions:   
- The selected customer record is removed from the system.   
- The system provides confirmation of the successful deletion.   
  
Main Flow:   
1. The Administrator navigates to the customer management section of the system.   
2. The Administrator selects the "Delete Customer" option.   
3. The system displays a list of existing customer records.   
4. The Administrator selects a specific customer to delete.   
5. The system prompts the Administrator to confirm the deletion.   
6. The Administrator confirms the deletion.   
7. The system removes the selected Customer entity from the database.   
8. The system displays a confirmation message indicating the customer was successfully deleted.   
  
Alternative Flow:   
1. If the selected customer has associated Orders or Payments, the system displays a warning and prevents deletion until these associations are resolved.   
2. If the Administrator cancels the deletion confirmation, the system returns to the customer list without making any changes.   
3. If the system fails to delete the customer due to a technical issue, an error message is displayed, and the Administrator is redirected to the customer list.  
  
Use Case Name: Add Product   
Use Case ID: UC-05   
Actors: Administrator   
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to manage product data.   
  
Postconditions:   
- A new product record is created in the system.   
- The system provides confirmation of the successful addition.   
  
Main Flow:   
1. The Administrator navigates to the product management section of the system.   
2. The Administrator selects the "Add Product" option.   
3. The system displays a form for entering product details (e.g., product name, description, price, category, and stock quantity).   
4. The Administrator fills in the required product information.   
5. The Administrator submits the form.   
6. The system validates the input data (e.g., checks for missing fields, valid price format).   
7. The system creates a new Product entity with the provided details.   
8. The system assigns a unique identifier to the new product.   
9. The system displays a confirmation message indicating the product was successfully added.   
  
Alternative Flow:   
1. If the Administrator leaves a required field empty, the system displays an error message prompting them to fill in the missing information.   
2. If the price format is invalid, the system displays an error message and prevents submission until a valid price is provided.   
3. If a product with the same unique identifier already exists, the system displays a message indicating the product already exists and suggests editing the existing record instead.  
  
Use Case Name: Update Product Details   
Use Case ID: UC-06   
Actors: Administrator   
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to manage product data.   
- At least one product record exists in the system.   
  
Postconditions:   
- The selected product's details are updated in the system.   
- The system provides confirmation of the successful update.   
  
Main Flow:   
1. The Administrator navigates to the product management section of the system.   
2. The Administrator selects the "Update Product Details" option.   
3. The system displays a list of existing product records.   
4. The Administrator selects a specific product to update.   
5. The system displays the product's current information in an editable form.   
6. The Administrator modifies the necessary product details.   
7. The Administrator submits the updated information.   
8. The system validates the updated data (e.g., checks for missing fields, valid price format).   
9. The system updates the Product entity with the new information.   
10. The system displays a confirmation message indicating the product details were successfully updated.   
  
Alternative Flow:   
1. If the Administrator leaves a required field empty, the system displays an error message prompting them to fill in the missing information.   
2. If the price format is invalid, the system displays an error message and prevents submission until a valid price is provided.   
3. If the Administrator cancels the update process, the system returns to the product list without making any changes.   
4. If the system fails to update the product due to a technical issue, an error message is displayed, and the Administrator is redirected to the product list.  
  
Use Case Name: View Product Catalog   
Use Case ID: UC-07   
Actors: Administrator, Customer, Plugin   
Preconditions:   
- The system is operational and accessible.   
- The Administrator or Customer is authenticated and authorized to view product data.   
- At least one product record exists in the system.   
  
Postconditions:   
- The product catalog is displayed to the user.   
- The user can search or filter products.   
- No changes are made to the product data.   
  
Main Flow:   
1. The user navigates to the product catalog section of the system.   
2. The system retrieves all Product entities from the database.   
3. The system displays the list of products in a catalog format (e.g., product name, price, description, and availability).   
4. The user can optionally use search or filter functionality provided by the system or Plugin to narrow down the catalog.   
5. The system updates the displayed catalog based on the user's search or filter input.   
  
Alternative Flow:   
1. If no product records exist in the database, the system displays a message indicating the catalog is empty.   
2. If the search or filter input is invalid, the system displays an error message and resets the catalog view.   
3. If the Plugin is unavailable or malfunctions, the system displays a warning and continues to show the catalog without the enhanced search/filter functionality.   
4. If the system fails to retrieve the product data due to a technical issue, an error message is displayed, and the user is redirected to the main dashboard.  
  
Use Case Name: Delete Product   
Use Case ID: UC-08   
Actors: Administrator   
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to manage product data.   
- At least one product record exists in the system.   
  
Postconditions:   
- The selected product record is removed from the system.   
- The system provides confirmation of the successful deletion.   
  
Main Flow:   
1. The Administrator navigates to the product management section of the system.   
2. The Administrator selects the "Delete Product" option.   
3. The system displays a list of existing product records.   
4. The Administrator selects a specific product to delete.   
5. The system prompts the Administrator to confirm the deletion.   
6. The Administrator confirms the deletion.   
7. The system removes the selected Product entity from the database.   
8. The system displays a confirmation message indicating the product was successfully deleted.   
  
Alternative Flow:   
1. If the selected product has associated Orders, the system displays a warning and prevents deletion until these associations are resolved.   
2. If the Administrator cancels the deletion confirmation, the system returns to the product list without making any changes.   
3. If the system fails to delete the product due to a technical issue, an error message is displayed, and the Administrator is redirected to the product list.  
  
Use Case Name: Place Order   
Use Case ID: UC-09   
Actors: Customer, Plugin, Notification, Payment   
Preconditions:   
- The system is operational and accessible.   
- The Customer is authenticated and has at least one Product in their cart.   
- The Payment system is available and configured.   
- The Plugin is available for any additional processing (e.g., discounts, shipping options).   
  
Postconditions:   
- A new Order is created in the system.   
- The associated Payment is processed or initiated.   
- A confirmation message is displayed to the Customer.   
- A Notification is sent to the Customer regarding the order confirmation.   
  
Main Flow:   
1. The Customer navigates to the cart section of the system.   
2. The system displays the list of selected Products with their details.   
3. The Customer selects the "Place Order" option.   
4. The system processes the cart using the Plugin for any additional calculations (e.g., taxes, discounts).   
5. The system displays an order summary including total price and shipping details.   
6. The Customer selects a payment method.   
7. The system initiates the Payment process for the order.   
8. The system creates a new Order entity with the selected Products and Customer information.   
9. The system sends a Notification to the Customer confirming the order placement.   
10. The system displays a confirmation message to the Customer.   
  
Alternative Flow:   
1. If the cart is empty, the system displays an error message and redirects the Customer to the product catalog.   
2. If the Plugin is unavailable or malfunctions, the system displays a warning and proceeds with default order processing.   
3. If the Payment fails, the system displays an error message and allows the Customer to retry or select an alternative method.   
4. If the system fails to create the Order due to a technical issue, an error message is displayed, and the Customer is redirected to the cart.   
5. If the Customer cancels the order placement, the system returns to the cart without creating the Order.  
  
Use Case Name: Modify Order   
Use Case ID: UC-10   
Actors: Administrator, Customer, Plugin, Notification, Payment   
Preconditions:   
- The system is operational and accessible.   
- The Administrator or Customer is authenticated and authorized to modify orders.   
- At least one Order record exists in the system.   
- The Payment system is available for any changes involving payment.   
- The Plugin is available for any additional processing (e.g., discounts, shipping adjustments).   
  
Postconditions:   
- The selected Order is updated in the system.   
- Any changes to the Order are reflected in the associated Payment or Notification.   
- The system provides confirmation of the successful modification.   
  
Main Flow:   
1. The Administrator or Customer navigates to the order management section of the system.   
2. The system displays a list of existing orders associated with the user.   
3. The user selects a specific Order to modify.   
4. The system displays the Order's current information in an editable form (e.g., product list, shipping details, payment status).   
5. The user modifies the necessary order details (e.g., update product quantities, change shipping address).   
6. The user submits the updated order information.   
7. The system validates the updated data (e.g., checks for valid product quantities, valid shipping address).   
8. If applicable, the system uses the Plugin to recalculate order totals, taxes, or shipping costs.   
9. The system updates the Order entity with the new information.   
10. If the modification affects the Payment, the system updates the Payment status accordingly.   
11. The system sends a Notification to the Customer regarding the order modification.   
12. The system displays a confirmation message indicating the order was successfully modified.   
  
Alternative Flow:   
1. If the user selects an Order that cannot be modified due to its current status (e.g., completed, canceled), the system displays an error message and prevents the modification.   
2. If the Plugin is unavailable or malfunctions during recalculation, the system displays a warning and proceeds with default calculations.   
3. If the updated data is invalid (e.g., negative quantity, missing shipping information), the system displays an error message and prompts the user to correct the input.   
4. If the system fails to update the Order due to a technical issue, an error message is displayed, and the user is redirected to the order list.   
5. If the user cancels the modification process, the system returns to the order list without making any changes.  
  
Use Case Name: View Order History   
Use Case ID: UC-11   
Actors: Customer, Administrator   
Preconditions:   
- The system is operational and accessible.   
- The Customer or Administrator is authenticated and authorized to view order data.   
- At least one Order record exists in the system associated with the Customer.   
  
Postconditions:   
- The order history for the Customer is displayed.   
- No changes are made to the Order or Payment data.   
  
Main Flow:   
1. The Customer or Administrator navigates to the order history section of the system.   
2. The system displays a list of Orders associated with the Customer.   
3. The user selects a specific Order to view its details.   
4. The system retrieves the selected Order entity from the database.   
5. The system displays the Order details (e.g., date, products ordered, total amount, payment status, and shipping information).   
  
Alternative Flow:   
1. If the selected Order does not exist in the database, the system displays an error message indicating the order cannot be found.   
2. If the system fails to retrieve the Order due to a technical issue, an error message is displayed, and the user is redirected to the order list.   
3. If the user does not select an Order, the system displays a prompt to choose an order from the list.  
  
Use Case Name: Cancel Order   
Use Case ID: UC-12   
Actors: Customer, Administrator, Notification, Payment   
  
Preconditions:   
- The system is operational and accessible.   
- The Customer or Administrator is authenticated and authorized to cancel orders.   
- At least one Order record exists in the system associated with the Customer.   
- The selected Order is in a cancellable state (e.g., pending, processing).   
  
Postconditions:   
- The selected Order is marked as canceled in the system.   
- The associated Payment is refunded or marked as voided if applicable.   
- A Notification is sent to the Customer confirming the cancellation.   
- The system provides confirmation of the successful cancellation.   
  
Main Flow:   
1. The Customer or Administrator navigates to the order management section of the system.   
2. The system displays a list of Orders associated with the Customer.   
3. The user selects a specific Order to cancel.   
4. The system prompts the user to confirm the cancellation.   
5. The user confirms the cancellation.   
6. The system checks the current status of the Order to ensure it is cancellable.   
7. The system updates the Order entity to reflect the canceled status.   
8. If the Order includes a Payment, the system initiates a refund or voids the Payment.   
9. The system sends a Notification to the Customer informing them of the cancellation.   
10. The system displays a confirmation message indicating the order was successfully canceled.   
  
Alternative Flow:   
1. If the selected Order is not in a cancellable state (e.g., already shipped or completed), the system displays an error message and prevents cancellation.   
2. If the system cannot process a refund for the associated Payment, an error message is shown, and the user is prompted to contact support.   
3. If the Notification system is unavailable, the system displays a warning and still confirms the cancellation but may delay the Notification.   
4. If the user cancels the cancellation confirmation, the system returns to the order list without making any changes.   
5. If the system fails to update the Order status due to a technical issue, an error message is displayed, and the user is redirected to the order list.  
  
Use Case Name: Process Payment   
Use Case ID: UC-13   
Actors: Customer, Payment, Plugin, Notification   
Preconditions:   
- The system is operational and accessible.   
- The Customer is authenticated and has an active Order in the system.   
- The selected Order contains Products with available stock.   
- The Payment system is available and properly configured.   
- The Plugin is available for any additional processing (e.g., payment validation, discounts).   
  
Postconditions:   
- The Payment for the selected Order is processed or initiated.   
- The Order status is updated to reflect the payment status.   
- A Notification is sent to the Customer confirming the payment result.   
- The system provides confirmation of the successful or failed payment processing.   
  
Main Flow:   
1. The Customer navigates to the checkout section of the system.   
2. The system displays the Order summary and available payment methods.   
3. The Customer selects a payment method.   
4. The system uses the Plugin to validate the payment method and calculate any additional fees or discounts.   
5. The system initiates the Payment process with the selected method.   
6. The system receives the result of the Payment (e.g., success, failure).   
7. The system updates the Order status to reflect the payment result.   
8. The system sends a Notification to the Customer confirming the payment result.   
9. The system displays a confirmation message to the Customer.   
  
Alternative Flow:   
1. If the selected payment method is invalid or not supported, the system displays an error message and prompts the Customer to choose a different method.   
2. If the Plugin is unavailable or malfunctions during validation, the system displays a warning and proceeds with default payment processing.   
3. If the Payment fails, the system displays an error message and allows the Customer to retry or select an alternative method.   
4. If the system fails to update the Order status due to a technical issue, an error message is displayed, and the Customer is redirected to the order details.   
5. If the Notification system is unavailable, the system displays a warning and confirms the payment result without sending the Notification.  
  
Use Case Name: View Payment Status   
Use Case ID: UC-14   
Actors: Customer, Administrator, Payment, Notification   
Preconditions:   
- The system is operational and accessible.   
- The Customer or Administrator is authenticated and authorized to view payment data.   
- At least one Payment record exists in the system associated with the Customer.   
- The Payment system is available for status retrieval.   
  
Postconditions:   
- The payment status for the selected Order is displayed.   
- No changes are made to the Payment or Order data.   
- A Notification may be sent if the payment status has been recently updated.   
  
Main Flow:   
1. The Customer or Administrator navigates to the order or payment section of the system.   
2. The system displays a list of Orders associated with the Customer.   
3. The user selects a specific Order to view its payment status.   
4. The system retrieves the Payment entity linked to the selected Order.   
5. The system displays the current payment status (e.g., pending, completed, failed).   
6. If the payment status has changed recently, the system sends a Notification to the Customer.   
  
Alternative Flow:   
1. If the selected Order does not have an associated Payment, the system displays a message indicating no payment has been made.   
2. If the Payment system is unavailable, the system displays a warning and may show cached or last-known payment status.   
3. If the system fails to retrieve the Payment status due to a technical issue, an error message is displayed, and the user is redirected to the order list.   
4. If no Order is selected, the system displays a prompt to choose an order from the list.  
  
Use Case Name: Delete Payment Record   
Use Case ID: UC-15   
Actors: Administrator   
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to manage payment data.   
- At least one Payment record exists in the system.   
  
Postconditions:   
- The selected Payment record is removed from the system.   
- The system provides confirmation of the successful deletion.   
  
Main Flow:   
1. The Administrator navigates to the payment management section of the system.   
2. The Administrator selects the "Delete Payment" option.   
3. The system displays a list of existing Payment records.   
4. The Administrator selects a specific Payment record to delete.   
5. The system prompts the Administrator to confirm the deletion.   
6. The Administrator confirms the deletion.   
7. The system removes the selected Payment entity from the database.   
8. The system displays a confirmation message indicating the payment was successfully deleted.   
  
Alternative Flow:   
1. If the selected Payment is associated with an Order, the system displays a warning and prevents deletion until the association is resolved.   
2. If the Administrator cancels the deletion confirmation, the system returns to the payment list without making any changes.   
3. If the system fails to delete the Payment due to a technical issue, an error message is displayed, and the Administrator is redirected to the payment list.  
  
Use Case Name: Install Plugin   
Use Case ID: UC-16   
Actors: Administrator, Plugin, Notification   
  
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to install plugins.   
- A Plugin is available for installation (e.g., uploaded or accessible via repository).   
  
Postconditions:   
- The selected Plugin is installed and configured in the system.   
- A Notification is sent to the Administrator confirming the installation result.   
- The system provides confirmation of the successful installation or failure.   
  
Main Flow:   
1. The Administrator navigates to the plugin management section of the system.   
2. The Administrator selects the "Install Plugin" option.   
3. The system displays a list of available Plugins or provides an upload interface.   
4. The Administrator selects a specific Plugin to install or uploads a new one.   
5. The system validates the Plugin (e.g., checks for compatibility, correct format).   
6. The system initiates the installation process for the selected Plugin.   
7. The system configures the Plugin based on default or user-defined settings.   
8. The system sends a Notification to the Administrator confirming the Plugin was successfully installed.   
9. The system displays a confirmation message indicating the Plugin is now active and ready to use.   
  
Alternative Flow:   
1. If the selected Plugin is incompatible with the current system version, the system displays an error message and prevents installation.   
2. If the Plugin file is corrupted or in an invalid format, the system displays an error message and prompts the Administrator to re-upload or select a valid Plugin.   
3. If the system fails to install the Plugin due to a technical issue, an error message is displayed, and the Administrator is redirected to the plugin list.   
4. If the Administrator cancels the installation process, the system returns to the plugin list without making any changes.   
5. If the Notification system is unavailable, the system displays a warning and confirms the installation result without sending the Notification.  
  
Use Case Name: Update Plugin   
Use Case ID: UC-17   
Actors: Administrator, Plugin, Notification   
  
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to manage plugins.   
- At least one Plugin is installed in the system.   
- The Plugin to be updated is available (e.g., new version uploaded or accessible via repository).   
  
Postconditions:   
- The selected Plugin is updated to the new version in the system.   
- A Notification is sent to the Administrator confirming the update result.   
- The system provides confirmation of the successful update or failure.   
  
Main Flow:   
1. The Administrator navigates to the plugin management section of the system.   
2. The Administrator selects the "Update Plugin" option.   
3. The system displays a list of installed Plugins along with their current versions and update availability.   
4. The Administrator selects a specific Plugin to update.   
5. The system validates the new Plugin version (e.g., checks for compatibility, correct format).   
6. The system initiates the update process for the selected Plugin.   
7. The system applies the new configuration or settings if required.   
8. The system sends a Notification to the Administrator confirming the Plugin was successfully updated.   
9. The system displays a confirmation message indicating the Plugin is now updated and active.   
  
Alternative Flow:   
1. If the new Plugin version is incompatible with the current system version, the system displays an error message and prevents the update.   
2. If the Plugin file is corrupted or in an invalid format, the system displays an error message and prompts the Administrator to re-upload or select a valid Plugin.   
3. If the system fails to update the Plugin due to a technical issue, an error message is displayed, and the Administrator is redirected to the plugin list.   
4. If the Administrator cancels the update process, the system returns to the plugin list without making any changes.   
5. If the Notification system is unavailable, the system displays a warning and confirms the update result without sending the Notification.  
  
Use Case Name: Uninstall Plugin   
Use Case ID: UC-18   
Actors: Administrator, Plugin, Notification   
  
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to manage plugins.   
- At least one Plugin is installed in the system.   
  
Postconditions:   
- The selected Plugin is uninstalled from the system.   
- A Notification is sent to the Administrator confirming the uninstallation result.   
- The system provides confirmation of the successful uninstallation or failure.   
  
Main Flow:   
1. The Administrator navigates to the plugin management section of the system.   
2. The Administrator selects the "Uninstall Plugin" option.   
3. The system displays a list of installed Plugins.   
4. The Administrator selects a specific Plugin to uninstall.   
5. The system prompts the Administrator to confirm the uninstallation.   
6. The Administrator confirms the uninstallation.   
7. The system removes the selected Plugin from the system.   
8. The system sends a Notification to the Administrator confirming the Plugin was successfully uninstalled.   
9. The system displays a confirmation message indicating the Plugin is no longer active.   
  
Alternative Flow:   
1. If the selected Plugin is in use (e.g., by an Order or Notification process), the system displays a warning and prevents uninstallation until the usage is resolved.   
2. If the Administrator cancels the uninstallation confirmation, the system returns to the plugin list without making any changes.   
3. If the system fails to uninstall the Plugin due to a technical issue, an error message is displayed, and the Administrator is redirected to the plugin list.   
4. If the Notification system is unavailable, the system displays a warning and confirms the uninstallation result without sending the Notification.  
  
Use Case Name: View Plugin List   
Use Case ID: UC-19   
Actors: Administrator   
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to manage plugins.   
  
Postconditions:   
- The list of installed and available Plugins is displayed.   
- No changes are made to the Plugin entities.   
  
Main Flow:   
1. The Administrator navigates to the plugin management section of the system.   
2. The Administrator selects the "View Plugin List" option.   
3. The system retrieves all Plugin entities from the database.   
4. The system displays the list of installed Plugins, including their names, versions, and statuses.   
5. The system also displays a section for available Plugins that can be installed.   
6. The Administrator can optionally filter or search for specific Plugins.   
7. The system updates the displayed Plugin list based on the Administrator's input.   
  
Alternative Flow:   
1. If no Plugins are installed, the system displays a message indicating the plugin list is empty.   
2. If no available Plugins exist, the system displays a message that no new Plugins can be installed at the moment.   
3. If the system fails to retrieve the Plugin data due to a technical issue, an error message is displayed, and the Administrator is redirected to the main dashboard.   
4. If the search or filter input is invalid, the system displays an error message and resets the view to the full Plugin list.  
  
Use Case Name: Send Notification   
Use Case ID: UC-20   
Actors: Administrator, Notification   
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to send notifications.   
- At least one Notification configuration or method is available in the system.   
- The target recipient (e.g., Customer) has a valid contact method (e.g., email, phone number).   
  
Postconditions:   
- A Notification is successfully sent to the intended recipient.   
- The system logs the Notification activity.   
- The Administrator receives confirmation of the notification being sent.   
  
Main Flow:   
1. The Administrator navigates to the notification management section of the system.   
2. The Administrator selects the "Send Notification" option.   
3. The system displays a form for entering the notification details (e.g., recipient, message, subject, delivery method).   
4. The Administrator fills in the required notification information.   
5. The Administrator submits the form.   
6. The system validates the input (e.g., checks for valid recipient information and message content).   
7. The system uses the Notification component to send the message via the selected delivery method.   
8. The system logs the notification attempt and sends a confirmation message to the Administrator.   
  
Alternative Flow:   
1. If the recipient information is invalid (e.g., incorrect email, missing phone number), the system displays an error message and prompts for valid details.   
2. If the Notification component is unavailable, the system displays a warning and prevents sending the notification until it is restored.   
3. If the system fails to send the Notification due to a technical issue, an error message is displayed, and the Administrator is redirected to the notification list.   
4. If the Administrator cancels the notification sending process, the system returns to the notification list without sending the message.  
  
Use Case Name: View Notification   
Use Case ID: UC-21   
Actors: Administrator, Customer   
Preconditions:   
- The system is operational and accessible.   
- The Administrator or Customer is authenticated and authorized to view notifications.   
- At least one Notification record exists in the system associated with the user.   
  
Postconditions:   
- The Notification details are displayed to the user.   
- No changes are made to the Notification data.   
- The user is informed of the latest notifications.   
  
Main Flow:   
1. The Administrator or Customer navigates to the notification section of the system.   
2. The system retrieves all Notification entities associated with the user.   
3. The system displays the list of notifications, including details such as date, message, and sender.   
4. The user selects a specific Notification to view its full content.   
5. The system displays the detailed information of the selected Notification.   
  
Alternative Flow:   
1. If the selected Notification does not exist in the database, the system displays an error message indicating the notification cannot be found.   
2. If the system fails to retrieve the Notification data due to a technical issue, an error message is displayed, and the user is redirected to the main dashboard.   
3. If no Notification is selected, the system displays a prompt to choose a notification from the list.  
  
Use Case Name: Delete Notification   
Use Case ID: UC-22   
Actors: Administrator   
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to manage notifications.   
- At least one Notification record exists in the system.   
  
Postconditions:   
- The selected Notification record is removed from the system.   
- The system provides confirmation of the successful deletion.   
  
Main Flow:   
1. The Administrator navigates to the notification management section of the system.   
2. The Administrator selects the "Delete Notification" option.   
3. The system displays a list of existing Notification records.   
4. The Administrator selects a specific Notification to delete.   
5. The system prompts the Administrator to confirm the deletion.   
6. The Administrator confirms the deletion.   
7. The system removes the selected Notification entity from the database.   
8. The system displays a confirmation message indicating the notification was successfully deleted.   
  
Alternative Flow:   
1. If the selected Notification is linked to an Order or Payment process, the system displays a warning and prevents deletion until the dependency is resolved.   
2. If the Administrator cancels the deletion confirmation, the system returns to the notification list without making any changes.   
3. If the system fails to delete the Notification due to a technical issue, an error message is displayed, and the Administrator is redirected to the notification list.  
  
Use Case Name: Manage Administrator   
Use Case ID: UC-23   
Actors: Administrator   
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to manage other administrators.   
- At least one administrator account exists in the system.   
  
Postconditions:   
- The selected administrator's account is updated or deleted in the system.   
- The system provides confirmation of the successful modification or deletion.   
  
Main Flow:   
1. The Administrator navigates to the administrator management section of the system.   
2. The Administrator selects the "Manage Administrator" option.   
3. The system displays a list of existing administrator accounts.   
4. The Administrator selects a specific administrator to edit or delete.   
5. If editing, the system displays the administrator's current information in an editable form (e.g., username, role, permissions).   
6. The Administrator modifies the necessary details.   
7. The Administrator submits the updated information.   
8. The system validates the updated data (e.g., checks for valid role, unique username).   
9. The system updates the Administrator entity with the new information.   
10. The system displays a confirmation message indicating the administrator was successfully modified.   
11. If deleting, the system prompts the Administrator to confirm the deletion.   
12. The Administrator confirms the deletion.   
13. The system removes the selected Administrator entity from the database.   
14. The system displays a confirmation message indicating the administrator was successfully deleted.   
  
Alternative Flow:   
1. If the Administrator attempts to delete an account that is currently active or has critical system responsibilities, the system displays an error message and prevents the deletion.   
2. If the updated username is already taken by another administrator, the system displays an error and prompts for a unique username.   
3. If the Administrator cancels the deletion confirmation, the system returns to the administrator list without making any changes.   
4. If the system fails to update or delete the administrator due to a technical issue, an error message is displayed, and the Administrator is redirected to the administrator list.   
5. If no administrator is selected, the system displays an error message prompting the Administrator to choose an entry to manage.  
  
Use Case Name: View Administrator Log   
Use Case ID: UC-24   
Actors: Administrator   
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to view system logs.   
- At least one log entry related to administrator actions exists in the system.   
  
Postconditions:   
- The Administrator can view the detailed log entries related to their or other administrators’ actions.   
- No changes are made to the log data.   
  
Main Flow:   
1. The Administrator navigates to the system log section of the system.   
2. The system displays a list of available log categories, including administrator actions.   
3. The Administrator selects the "View Administrator Log" option.   
4. The system retrieves all log entries related to administrator activities from the database.   
5. The system displays the logs in a structured format, including details such as timestamp, user, action performed, and additional context.   
6. The Administrator can optionally filter or search the logs for specific entries.   
7. The system updates the displayed logs based on the Administrator's input.   
  
Alternative Flow:   
1. If no administrator log entries exist, the system displays a message indicating the log is empty.   
2. If the search or filter input is invalid, the system displays an error message and resets the log view.   
3. If the system fails to retrieve the log entries due to a technical issue, an error message is displayed, and the Administrator is redirected to the main dashboard.   
4. If the Administrator does not select the correct option, the system displays a prompt to choose "View Administrator Log" from the available options.  
  
Use Case Name: Remove Administrator   
Use Case ID: UC-25   
Actors: Administrator   
Preconditions:   
- The system is operational and accessible.   
- The Administrator is authenticated and authorized to manage other administrator accounts.   
- At least one administrator account exists in the system besides the one being removed.   
  
Postconditions:   
- The selected administrator account is permanently removed from the system.   
- The system provides confirmation of the successful removal.   
  
Main Flow:   
1. The Administrator navigates to the administrator management section of the system.   
2. The Administrator selects the "Remove Administrator" option.   
3. The system displays a list of existing administrator accounts.   
4. The Administrator selects a specific administrator to remove.   
5. The system prompts the Administrator to confirm the removal.   
6. The Administrator confirms the removal.   
7. The system removes the selected Administrator entity from the database.   
8. The system displays a confirmation message indicating the administrator was successfully removed.   
  
Alternative Flow:   
1. If the selected administrator is the last remaining administrator, the system displays an error message and prevents removal.   
2. If the system fails to remove the administrator due to a technical issue, an error message is displayed, and the Administrator is redirected to the administrator list.   
3. If the Administrator cancels the removal confirmation, the system returns to the administrator list without making any changes.   
4. If the selected administrator has active sessions or pending actions, the system displays a warning and prevents removal until these are resolved.  
  
Use Case Name: Manage Order Detail   
Use Case ID: UC-26   
Actors: Administrator, Customer, Plugin, Notification   
  
Preconditions:   
- The system is operational and accessible.   
- The Administrator or Customer is authenticated and authorized to manage order details.   
- At least one Order record exists in the system associated with the Customer.   
- The selected Order contains at least one Order Detail (i.e., itemized product information).   
- The Plugin is available for any additional processing if required (e.g., recalculations).   
  
Postconditions:   
- The selected Order Detail is updated, removed, or added to the Order.   
- The system updates the Order entity to reflect the changes in details.   
- A Notification may be sent to the Customer if the modification affects their order.   
- The system provides confirmation of the successful modification.   
  
Main Flow:   
1. The Administrator or Customer navigates to the order management section of the system.   
2. The system displays a list of Orders associated with the user.   
3. The user selects a specific Order to manage its details.   
4. The system displays the Order’s current details (e.g., product list, quantity, price per item, and total for each item).   
5. The user selects an option to add, edit, or remove an Order Detail.   
6. If adding, the system provides a form to input new Order Detail information (e.g., product ID, quantity, price).   
7. If editing, the system displays the selected Order Detail in an editable form.   
8. The user modifies or inputs the necessary information for the Order Detail.   
9. The user submits the changes.   
10. The system validates the input (e.g., checks for valid product ID, positive quantity, and correct price format).   
11. The system updates the Order entity with the modified Order Details.   
12. If applicable, the system uses the Plugin to recalculate the total price or apply any relevant adjustments.   
13. The system sends a Notification to the Customer if the modification impacts their order (e.g., price change, item removal).   
14. The system displays a confirmation message indicating the Order Detail was successfully managed.   
  
Alternative Flow:   
1. If the user attempts to modify an Order Detail for an Order that is in a non-editable state (e.g., shipped or canceled), the system displays an error message and prevents the modification.   
2. If the Plugin is unavailable or malfunctions during recalculation, the system displays a warning and proceeds with default calculations or skips the recalculation.   
3. If the input data for the Order Detail is invalid (e.g., negative quantity, missing product ID), the system displays an error message and prompts the user to correct the input.   
4. If the system fails to update the Order due to a technical issue, an error message is displayed, and the user is redirected to the order list.   
5. If the Notification system is unavailable, the system displays a warning and confirms the modification without sending the Notification.   
6. If the user cancels the modification process, the system returns to the Order details list without making any changes.