

- 1) Roles & Permissions

- Inventory Manager: Manages the inventory, creates orders, approves supplier quotations, and tracks inventory levels.
- Supplier: Receives order requests, submits quotations, and delivers approved orders.
- Admin: Oversees the entire system, manages users, and configures system settings.

## 2) Tables & Database Structure

- Supplier Table: Stores supplier details and contact information.
- Inventory Table: Tracks all items in stock, including quantity, price, and status.
- Inventory Orders Table: Records orders created by the inventory manager, including the items, quantities, and selected suppliers.
- Quotations Table: Stores quotations submitted by suppliers in response to inventory orders.
- Categories Table: Organizes inventory items into categories, such as meats, vegetables, dairy, etc.
- Notifications Table: Logs all notifications sent via email, WhatsApp, and the portal.

## System Workflow

### 1. Inventory Order Creation

- The Inventory Manager identifies the need for new inventory based on current stock levels.
- An order is created in the Inventory Orders Table, specifying the required items, quantities, and the desired delivery time.
- Notifications are sent to all relevant suppliers within the item's category via email, WhatsApp.

### 2. Supplier Quotation Submission

- Suppliers receive the order request and submit their quotations via the system.
- Quotations include price, delivery time, and any additional terms.

### 3. Quotation Approval

- The Inventory Manager reviews all submitted quotations in the Quotations Table.
- They can approve one or multiple suppliers for the order.
- Once approved, notifications are sent to the selected suppliers, confirming the order details, including the items, quantities, prices, and delivery instructions.

### 4. Order Acceptance by Supplier

- Suppliers receive the order approval notification.
- They can either accept or reject the order. If accepted, the system updates the order status to "Accepted."
- If rejected, the Inventory Manager is notified, and they can choose another supplier or reissue the order.

## 5. Order Delivery & Inventory Update

- Upon delivery, the supplier updates the order status to "Delivered."
- The system automatically updates the Inventory Table to reflect the new stock levels.
- The delivered items are added to the inventory, and the system checks if any items fall below the reorder threshold, triggering the cycle to repeat if necessary.

## Statuses & Notifications

### • Order Statuses:

- Pending: Order created, awaiting supplier quotations.
- Quotation Submitted: Supplier has submitted a quotation for the order.
- Approved: Inventory Manager has approved one or more supplier quotations.
- Rejected: Supplier has rejected the order request.
- Accepted: Supplier has accepted the order and will deliver the items.
- Delivered: Items have been delivered and added to the inventory.

### • Notifications:

- Order Request: Sent to suppliers upon creation of an order.
- Quotation Approval: Sent to suppliers when their quotation is approved.
- Order Acceptance: Sent to Inventory Manager when a supplier accepts or rejects an order.
- Delivery Confirmation: Sent to Inventory Manager when an order is marked as delivered.

## Reports & Analytics

- Inventory Reports: Track inventory levels, including items nearing reorder points.
- Supplier Performance: Analyze supplier reliability, delivery times, and pricing trends.
- Order History: Review past orders, including supplier quotations and final costs.

## Relationships and Workflow

1. Inventory & Categories: Each inventory item is categorized, allowing suppliers to be linked to specific categories.
2. Inventory Orders: Created by the inventory manager when an item in the inventory falls

below a certain threshold.

3. Quotations: Submitted by suppliers in response to orders. Multiple suppliers can submit quotations for a single order.
4. Notifications: Alerts are sent to inventory managers and suppliers throughout the order process, including when stock is low, when quotations are approved, and when orders are delivered.

## Database Tables

```
CREATE TABLE `suppliers` (  
  `supplier_id` INT(11) NOT NULL AUTO_INCREMENT,  
  `supplier_name` VARCHAR(255) NOT NULL,  
  `email` VARCHAR(255) DEFAULT NULL,  
  `phone_no` VARCHAR(20) DEFAULT NULL,  
  `category` VARCHAR(255) NOT NULL, -- Category of items the supplier provides  
  `about_supplier` TEXT DEFAULT NULL,  
  `status` ENUM('Active','Inactive') DEFAULT 'Active',  
  `created_at` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,  
  `updated_at` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE  
  CURRENT_TIMESTAMP,  
  PRIMARY KEY (`supplier_id`)  
);
```

////////////////////////////////////

```
CREATE TABLE `inventory` (  
  `inventory_id` INT(11) NOT NULL AUTO_INCREMENT,  
  `supplier_id` INT(11) NOT NULL,  
  `item_name` VARCHAR(255) NOT NULL,  
  `quantity` INT(11) NOT NULL,  
  `unit` VARCHAR(255) NOT NULL,  
  `status` ENUM('Active','Inactive') DEFAULT 'Active',  
  `created_at` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,  
  `updated_at` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE  
  CURRENT_TIMESTAMP,  
  PRIMARY KEY (`inventory_id`),  
  FOREIGN KEY (`supplier_id`) REFERENCES `suppliers` (`supplier_id`)  
);
```

```

`inventory_id` INT(11) NOT NULL AUTO_INCREMENT,
`item_name` VARCHAR(255) NOT NULL,
`barcode` VARCHAR(100) NOT NULL UNIQUE,
`category` VARCHAR(255) NOT NULL, -- Category of the item, related to supplier category
`type` VARCHAR(100) NOT NULL, -- Type of item (e.g., meat, vegetable, dairy)
`price` DECIMAL(10,2) NOT NULL,
`quantity` DECIMAL(10,2) NOT NULL, -- Quantity in stock
`unit` VARCHAR(20) NOT NULL, -- Unit of measurement (e.g., kg, liter)
`image` TEXT DEFAULT NULL,
`status` ENUM('Available','Out of Stock') DEFAULT 'Available',
`created_at` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
`updated_at` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP,
PRIMARY KEY (`inventory_id`)
);

```

////////////////////////////////////

```

CREATE TABLE `inventory_orders` (
`order_id` INT(11) NOT NULL AUTO_INCREMENT,
`order_number` VARCHAR(100) NOT NULL UNIQUE,
`inventory_id` INT(11) NOT NULL, -- References the inventory item
`quantity_ordered` DECIMAL(10,2) NOT NULL, -- Quantity ordered
`unit` VARCHAR(20) NOT NULL, -- Unit of measurement (e.g., kg, liter)
`status` ENUM('Pending','Approved','Rejected','Accepted','Delivered') DEFAULT 'Pending',
`created_at` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,

```

```
`updated_at` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE  
CURRENT_TIMESTAMP,  
  
PRIMARY KEY (`order_id`),  
  
FOREIGN KEY (`inventory_id`) REFERENCES `inventory` (`inventory_id`) ON DELETE CASCADE  
);
```

////////////////////////////////////

```
CREATE TABLE `quotations` (  
  
  `quotation_id` INT(11) NOT NULL AUTO_INCREMENT,  
  
  `order_id` INT(11) NOT NULL, -- References the inventory order  
  
  `supplier_id` INT(11) NOT NULL, -- References the supplier  
  
  `price_per_unit` DECIMAL(10,2) NOT NULL, -- Price quoted per unit  
  
  `total_price` DECIMAL(10,2) NOT NULL, -- Total price for the order  
  
  `status` ENUM('Submitted','Approved','Rejected') DEFAULT 'Submitted',  
  
  `created_at` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,  
  
  `updated_at` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE  
CURRENT_TIMESTAMP,  
  
  PRIMARY KEY (`quotation_id`),  
  
  FOREIGN KEY (`order_id`) REFERENCES `inventory_orders` (`order_id`) ON DELETE CASCADE,  
  
  FOREIGN KEY (`supplier_id`) REFERENCES `suppliers` (`supplier_id`) ON DELETE CASCADE  
);
```

////////////////////////////////////

```
CREATE TABLE `categories` (  
  `category_id` INT(11) NOT NULL AUTO_INCREMENT,  
  `category_name` VARCHAR(255) NOT NULL UNIQUE,  
  `description` TEXT DEFAULT NULL,  
  `created_at` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,  
  `updated_at` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE  
  CURRENT_TIMESTAMP,  
  PRIMARY KEY (`category_id`)  
);
```

```
////////////////////////////////////
```

```
CREATE TABLE `notifications` (  
  `notification_id` INT(11) NOT NULL AUTO_INCREMENT,  
  `recipient_type` ENUM('Inventory Manager', 'Supplier') NOT NULL,  
  `recipient_id` INT(11) NOT NULL, -- References either inventory manager or supplier based on  
  recipient_type  
  `message` TEXT NOT NULL,  
  `status` ENUM('Sent', 'Failed') DEFAULT 'Sent',  
  `created_at` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,  
  PRIMARY KEY (`notification_id`)  
);
```