Project Proposal: Grocery Delivery Management System - *AirLyft*

Overview:

The *AirLyft* project is designed to be a comprehensive grocery delivery management system built using C++ and OOP principles. It incorporates core enterprise system concepts such as **Supply Chain Management (SCM)**, **Human Resource Management (HRM)**, **Customer Relationship Management (CRM)**, and **Customer Interaction**. The system is intended to streamline grocery order management, optimize delivery processes, and ensure a seamless customer experience.

Key Enterprise System Concepts:

1. **SCM (Supply Chain Management):** Manages supplier inventory, incoming stock, and grocery products for timely fulfillment of customer orders.
2. **HRM (Human Resource Management)**: Tracks delivery personnel (drivers) and their schedules to ensure timely order fulfillment.
3. **CRM (Customer Relationship Management)**: Stores customer data, handles order history, preferences, and provides personalized services.
4. **Customer Interaction**: Facilitates the interface between the customer and the platform for placing orders, tracking deliveries, and providing feedback.

Class Design:

1. **Class: Product**
   - Attributes:
     - product_id
     - name
     - category
     - price
     - stock_quantity
   - Methods:
     - update_stock()
     - get_price()
     - display_details()
     - check_availability()
2. **Class: Supplier**
   - Attributes:

        - supplier_id
        - supplier_name
        - contact_info
        - supplied_products (list of Product objects)
    - Methods:
        - supply_products()
        - update_supplier_info()
        - view_supplied_products()

3. **Class: Order**
    - Attributes:
        - order_id
        - customer
        - product_list (list of Product objects)
        - order_total
        - delivery_status
    - Methods:
        - calculate_total()
        - update_status()
        - display_order_details()
4. **Class: Customer**
    - Attributes:
        - customer_id
        - name
        - contact_info
        - order_history (list of Order objects)
    - Methods:
        - place_order()
        - view_order_history()
        - update_contact_info()
        - give_feedback()
5. **Class: DeliveryPersonnel (HRM Integration)**
    - Attributes:

        - personnel_id
        - name
        - assigned_orders (list of Order objects)
        - availability_status
    - Methods:
        - assign_order()
        - update_status()
        - view_assigned_orders()

6. **Class: CRM**
    - Attributes:
        - customer_data (stores all Customer objects)
        - feedback_data
    - Methods:
        - add_new_customer()
        - update_customer_info()
        - store_feedback()
        - get_customer_info()
7. **Class: SCM**
    - Attributes:
        - product_inventory (stores all Product objects)
        - suppliers (list of Supplier objects)
    - Methods:
        - check_inventory()
        - replenish_stock()
        - view_supplier_list()
8. **Class: System (Main Controller)**
    - Manages the entire workflow of AirLyft
    - Attributes:
        - orders (list of Order objects)
        - customers (list of Customer objects)
        - delivery_personnel (list of DeliveryPersonnel objects)
        - crm (instance of CRM)

        - scm (instance of SCM)
    - Methods:
        - process_order()
        - manage_inventory()
        - assign_delivery_personnel()
        - customer_interaction()

Work Flow of Operations:

1. **Customer Interaction**:
    - A customer interacts with the system via a user interface (not part of the C++ core, but could be simulated with command-line interaction).
    - The customer selects grocery products from available inventory using the place_order() method.
    - The system checks product availability using the check_availability() method from the Product class.
2. **Order Placement**:
    - The customer places an order by adding selected products to the product_list of the Order class.
    - The Order object calculates the total price using the calculate_total() method and stores the details in the customer's order history.
3. **Inventory and SCM Management**:
    - The system checks inventory levels using the check_inventory() method from the SCM class.
    - If stock is low, the system triggers the replenish_stock() method, which involves contacting the supplier (Supplier class) to restock the items.
4. **Delivery Assignment**:
    - Once the order is confirmed, the system assigns delivery personnel using the assign_order() method from the DeliveryPersonnel class.
    - The availability of personnel is tracked using the availability_status attribute.
    - The assigned personnel can update their status as in_transit or delivered once the order is processed.
5. **Customer Feedback & CRM**:
    - After delivery, the system allows the customer to leave feedback via the give_feedback() method.
    - The feedback is stored in the CRM system (store_feedback() in the CRM class), which helps in maintaining customer satisfaction and personalization.

6. **HRM (Delivery Personnel Management)**:
   - The HRM system tracks delivery personnel performance, availability, and efficiency.
   - Personnel are assigned orders based on availability, tracked using update_status() in the DeliveryPersonnel class.

Detailed Work Flow:

1. **Product Inventory Initialization**:
   - At the start, all available grocery products are stored in the product_inventory attribute of the SCM class. Each product has attributes like product_id, name, price, and stock_quantity.
2. **Customer Browses & Places an Order**:
   - The customer interacts with the System, selects products from inventory, and places an order via the place_order() method. The system verifies stock using check_availability() in the Product class.
   - Once products are selected, an Order object is created, which stores the details of the products, calculates the total, and assigns it to the customer.
3. **SCM & Inventory Check**:
   - The system continuously monitors inventory levels. If stock is low, the replenish_stock() function contacts the supplier to restock.
4. **HRM & Delivery Assignment**:
   - Available delivery personnel are assigned orders based on their current availability using assign_order() in the DeliveryPersonnel class.
5. **Delivery Tracking & Customer Feedback**:
   - The system tracks the order's delivery status through DeliveryPersonnel. After completion, the customer leaves feedback, stored in the CRM module for future reference and customer relationship management.

Proposed Milestones:

1. **Week 1**:
   - Define and implement core classes (Product, Customer, Order).
   - Design the interaction flow between the Customer and Order classes.
2. **Week 2**:
   - Develop SCM and CRM modules.
   - Implement inventory management and customer data management.
3. **Week 3**:

- Add HRM features to manage delivery personnel.
- Integrate customer feedback into the CRM module.

4. **Week 4**:
    - Test and finalize interactions between all components (SCM, CRM, HRM).
    - Optimize code for efficiency and scalability.

This roadmap outlines a structured approach to building *AirLyft* as a functional grocery delivery management system using OOP principles in C++. Each module of the system is tied to an enterprise concept (SCM, HRM, CRM), ensuring a real-world simulation of how such systems operate.