Cosmetics Online Store

1.system scope

Cosmetics Online Store System

2. Mandatory Objects

- **Admin**
- **Item**
- **Client (Customer)**

3. Software Type & Requirement Gathering Techniques

- **Type**: Customizable system
- **Gather Techniques**: Questionnaires
- **Clarification of Requirements**

Admin Features

1. **Admin Account**: Must be created before any operations.

2. **Login**: Admins will log in using (email and password).

No

3. **Item Management**:
   - 3.1. Add items with details (name, description, image, category, brand, quantity, cost, price, discount flag, and discount amount).
   - 3.2. Explore all items.
   - 3.3. Activate/deactivate items (availability for sale or not).
   - 3.4. Edit item details.
   - 3.5. Add discounts for loyal customers.

4. **Category and Brand Management**:
   - 4.1. Add categories (name, description).
   - 4.2. Add brands (name, description).

5. **Ingredient Management**:
    5.1. Add new ingredients (name, quantity).
    5.2. Explore all ingredients.
    5.3. Update ingredient details.
    5.4. Activate/deactivate ingredients.

6. **Customer Management**:
    6.1. Add customers (full name, email, phone, address).
    6.2. Explore the customer list.
    6.3. Activate/deactivate customers.
    6.4. Edit customer details.

7. **Managing Orders**:updating the status of each order (e.g., "Completed," "Pending," "Canceled")

8.The admin can generate sales reports

Customer Features

1. **Login:** Customers will log in using email and password.

2. **Product Requests:**Customers can send item requests with details (item details, quantity, status, request date, customer info).

3. **Item Exploration:**

    3.1. Customers can explore all items (name, image, price).

    3.2. Customers can view detailed item information (name, description, image, category, brand, quantity, cost, price, discount flag, discount).

4. **Rating:**

    4.1. Customers can rate items in their orders.

4.2. Customers can rate their overall order.

5. **Wish List:**

   5.1. Customers can add favorite items to their wish list (customer, item, added date).

   5.2. Customers can delete items from their wish list that they no longer want.

6. **Order Creation:** Customers can create orders by selecting items, specifying quantities, and completing the purchase process.

7. **Ingredient Exploration:** Customers can explore ingredients associated with items.

8. **Brand Exploration:** Customers can explore item brands.

9.Category Exploration: Customers can explore item categories.

10.Logout: Customers can log out of the system.

4.Summarizing System Objects (Abstract action)

(id,createby,updateby,isActive,updatedate,creationdate)

1. **Admin** : (Email ,Password,Profile image,Phone number).
2. **Customer** :(Email ,Password,Profile image,Phone number)
3. **Item**:(Name,Description,Image,Category,Brand,Quantity,Cost,Price

   ,Discount flag(boolean),Discount amount,rate)

4. **Order:**(totalAmount,Status (e.g., 'Pending', 'Approved', 'Rejected'),)

   5.**OrderItem** (Quantity,ItemPrice,DiscountAmount ,Rating (Optional by Client),subtotal)

5. **Ingredient:**(Name,Quantity)
6. **Wishlist:,**(itemid,Quantity)
7. **itemRequest:(**QuantityRequested,ProductName,Description)

5- Generalization and Specialization (Restructuring)
(person):Admin,customer
(ParentEntity):person,item,order,orderItem,ingredient,wishlist, itemRequest

6- Define Relationships

1)Inheritance

A - Combine every thing in one table (Person)

B - Create child and add the shared element(id,createby,updateby,isActive,updatedate,creationdate) in each child (person,item,order,orderItem,ingredient,wishlist, itemRequest) individually

2)Composition

- **Person**
  1. **One customer creates many orders (1-M)orders.**
  2. **One customer adds many items to their wishlist (1-1)wishList.**
  3. **One customer sends many item requests (1-M)itemRequest**
- **Item**
  1. **One item has many ingredients (1-M)Ingredient**
  2. **One item appears in many order items (1-M)orderItem**

- **Order (1- M)orderItem(One order has many order items).**
- **orderItem (week entity)**
- **Ingredient(week entity)**
- **wishlist(week entity)**

each wishlist have many item) (1-M) item

- **itemRequest(week entity)**

1) OrderItem (Depends on Order and Item)

2)Ingredient (Depends on Item)

3) Wishlist (Depends on Customer and Item)

4)ItemRequest (Depends on Customer)

Multi Dependent :(OrderItem,Wishlist)

Single Dependent:(Ingredient,ItemRequest)

8- Implement Database (Tables and constraints)

/Normalization (Actors , Object , Lookups (status,category,brand))