PHASE 3 – Data Modeling & Relationships

HandsMenThreads – Salesforce Retail CRM Data Model

1. Introduction

Data Modeling is the foundation of any Salesforce implementation, and for HandsMenThreads, it is the most crucial step because the entire business revolves around structured product catalogs, customer histories, orders, order items, loyalty tracking, and marketing operations. This phase defines how information is stored, related, retrieved, and utilized across the CRM system.

A robust data model ensures that business processes are streamlined, reports are accurate, automations run efficiently, and user workflows remain intuitive. The HandsMenThreads data model not only captures core retail operations but also supports scalability for future enhancements like returns management, coupons, multi-channel sales integration, and advanced analytics.

2. Objectives of Data Modeling

The primary objectives of building this data model were:

1. **Represent all real-world entities** involved in the fashion retail business, such as products, orders, customers, campaigns, and loyalty.
2. **Define relationships** among these objects to support business flows like product-to-order mapping, customer-to-orders, and orders-to-campaigns.
3. **Ensure data integrity**, preventing inconsistencies like orders without products or stock going negative.
4. **Enable powerful reporting** to analyze sales trends, customer segments, and product performance.
5. **Support automation** through triggers and flows by ensuring the model is relational and easily accessible.

3. Custom Objects Created

HandsMenThreads requires several custom objects due to its specialized workflow. Below is a breakdown of the main objects:

1. Customer__c

Stores all customer-related information.

- Name
- Phone
- Email

- Total_Spend__c
- Loyalty_Points__c
- Loyalty_Tier__c
- First_Order_Date__c
- Last_Order_Date__c

This object serves as the backbone for customer segmentation, loyalty calculations, and relationship management.

2. Product__c

Represents fashion products.

- Product_Name__c
- SKU__c
- Price__c
- Stock__c
- Category__c (Shirts, Trousers, Ethnic Wear, Accessories, etc.)
- Size_Availability__c
- Image_URL__c

This object allows inventory control and supports reporting on top-selling categories.

3. Order__c

Captures order details.

- Customer__c (Lookup)
- Order_Date__c
- Order_Status__c (Placed, Packed, Shipped, Delivered, Cancelled)
- Order_Total__c
- Payment_Mode__c
- Campaign_Influenced__c

Orders are essential for tracking sales performance, loyalty generation, and customer engagement.

4. Order_Item__c (Junction Object)

Represents line items within an order.

- Order__c (Master-Detail)
- Product__c (Lookup)
- Quantity__c
- Line_Total__c

This allows the creation of multi-product orders and supports detailed transactional reporting.

5. Marketing_Campaign__c

Stores marketing efforts.

- Campaign_Name__c
- Start_Date__c
- End_Date__c
- Discount_Offered__c
- Type__c (Seasonal Sale, New Arrival Launch, Loyalty Promotion)

Campaigns support attribution and tracking of marketing performance.

4. Relationships Between Objects

Customer ↔ Order (Lookup Relationship)

- One customer can have multiple orders.
- Customer records display order history, lifetime spend, and visit frequency.

Order ↔ Order Item (Master-Detail)

- An Order cannot exist without at least one Order Item.
- When an order is deleted, related items also get deleted.
- Useful for roll-up calculations (total products ordered, total quantity).

Product ↔ Order Item (Lookup Relationship)

This allows:

- A product to appear in many orders.
- Order items to pull price & stock details dynamically.

Campaign ↔ Order (Lookup Relationship)

This facilitates:

- Linking orders to campaigns for ROI tracking.
- Reporting on campaign-driven revenue.

5. Field-Level Details and Purpose

Loyalty Fields

- Loyalty_Points__c
- Loyalty_Tier__c
- Tier_Upgrade_Date__c

These fields support automated loyalty progression using Apex.

Inventory Fields

- Stock__c
- Reorder_Level__c
- Supplier__c

Inventory accuracy is essential to prevent overselling.

Order Fields

- Shipping_Address__c
- Delivery_Partner__c
- Tracking_ID__c

These will support future logistics integration.

6. Schema Builder Usage

The Schema Builder was used to visually verify:

- Object dependencies
- Relationship integrity
- Cardinality correctness
- Fields grouping

The visual blueprint helped ensure that the model was scalable and clean.

7. Data Integrity Considerations

To avoid data issues:

- Required fields ensure completeness.
- Lookup filters restrict invalid selections.
- Roll-up summary fields handle total calculations automatically.

8. Future-Proofing the Model

The design accommodates:

- Product variants (size, color)
- Multi-store inventory
- Returns and exchanges
- Stock transfers
- Coupon systems

The flexible structure ensures long-term sustainability.

Conclusion

Phase 3 establishes a strong data foundation for all subsequent phases. With this relational model, HandsMenThreads gains the ability to track customer activity, automate loyalty, manage inventory accurately, and generate detailed reports. This data model transforms the CRM into a powerful retail engine supporting every part of the business lifecycle.