

Salesforce CRM Project Documentation

HandsMen Threads: Elevating the Art of Sophistication in Men's Fashion

Submitted by:

Chanda Indraja

Institution/Organization:

SMART BRIDGE

Program:

Salesforce Developer with Agentblazer Champion Program

Project Type:

**Salesforce CRM Implementation (Custom Objects, Flows, Apex,
Automation, Reports)**

Guided by:

Smart Bridge Educational Pvt.Lmt

Project Overview

Project Description:

HandsMen Threads is a premium fashion brand dedicated to redefining the modern man's wardrobe through sophistication, quality, and innovation. This project aims to develop a comprehensive digital platform that showcases the brand's offerings, enhances customer engagement, and streamlines backend operations, such as inventory, customer management, and order processing.

The platform will include an e-commerce website integrated with Salesforce CRM to offer personalized experiences, real-time customer support, and data-driven marketing strategies. In addition to showcasing elegant menswear collections, the project will provide features for managing customer interactions, tracking loyalty status, and generating sales reports.

Objectives:

- Create a user-friendly and visually appealing e-commerce website for HandsMen Threads.
- Integrate Salesforce to manage customer data, orders, and support interactions.
- Implement automation using Salesforce Flows to optimize internal processes (e.g., order confirmations, loyalty updates).
- Develop a scalable backend that supports future product expansions and marketing campaigns.

Key Features:

- Product catalog with detailed views and filtering options.
- Secure user registration, login, and account management.
- Seamless checkout and payment integration.
- Admin panel for inventory and order management.
- Salesforce-powered customer relationship management (CRM).
- Loyalty program based on purchase history.
- Email and notification automation for order status and promotions.

Target Audience:

Style-conscious men seeking premium and sophisticated clothing with personalized services and a seamless online shopping experience.

Technology Stack:

- **Frontend:** React.js / Lightning Web Components (LWC)
- **Backend:** Node.js / Salesforce Apex
- **Database:** Salesforce objects / external databases (if applicable)
- **CRM & Automation:** Salesforce Flow, Process Builder, Email Alerts
- **Hosting:** Salesforce Experience Cloud / Web Hosting Services
- **Other Tools:** Visual Studio Code, GitHub, Figma (for UI design), and Rational Rose (for UML design)

Objectives:

The primary objective of the **HandsMen Threads** project is to establish a robust and elegant digital presence for the brand that reflects its commitment to sophistication and quality in men's fashion. This platform will serve as a comprehensive solution for showcasing the brand's product offerings, facilitating online sales, and enhancing customer engagement through modern CRM and automation tools. Below are the detailed objectives of the project:

1. **Develop a Feature-Rich E-Commerce Platform:**

Build a responsive and visually appealing web platform that allows customers to explore, filter, and purchase premium men's clothing with ease. The website will reflect the brand's identity and provide an immersive shopping experience.

2. **Seamless Shopping Experience:**

Ensure that users can register, log in, browse collections, add items to their cart, and complete purchases with minimal friction. Features like secure payments, order tracking, and personalized product recommendations will be prioritized.

3. **Salesforce CRM Integration:**

Integrate Salesforce to manage customer profiles, purchase history, and support cases. This will enable the business to offer tailored services, track customer interactions, and implement data-driven marketing strategies.

4. **Automate Core Business Processes:**

Utilize Salesforce Flow and automation tools to reduce manual effort in order confirmations, loyalty program updates, inventory alerts, and customer communication. This will enhance operational efficiency and minimize errors.

5. **Admin Dashboard for Business Control:**

Develop an intuitive admin dashboard where the HandsMen team can manage inventory, monitor orders, update product listings, and generate reports without technical assistance.

6. **Loyalty Program Implementation:**

Introduce a points-based or tier-based loyalty system that rewards repeat customers. Based on their purchase history, customers will be categorized (e.g., Bronze, Silver, Gold), and rewarded with discounts, early access, or exclusive offers.

7. **Ensure Scalability and Flexibility:**

Design the system architecture to support future growth, such as adding new product categories, launching mobile apps, or expanding to international markets.

8. **Data Security and Compliance:**

Protect sensitive user data using industry-standard encryption, secure login methods, and compliance with data privacy regulations such as GDPR. The system will incorporate user consent handling, password protection, and secure payment gateways.

9. **Enhance Customer Engagement:**

Include dynamic features such as style blogs, customer reviews, email newsletters, and AI-powered product suggestions to increase customer interaction and retention.

10. **Generate Insightful Reports and Analytics:**

Enable business users to access real-time reports on sales, customer behavior, and inventory levels using Salesforce dashboards. This will assist in strategic decisionmaking and marketing optimization.

Scope 1: Software Requirements Specification

1. Purpose

The purpose of this document is to outline the software requirements for the development of the HandsMen Threads web-based fashion platform. This system will provide a seamless online shopping experience for users and streamline business processes for administrators by integrating Salesforce CRM features.

2. System Overview

HandsMen Threads is a fashion brand targeting modern men who seek elegance and sophistication. The proposed system will include:

- A fully responsive e-commerce website
- Admin control panel for managing products and orders
- Integration with Salesforce for customer relationship and order automation
- Automated loyalty program updates based on purchase history

3. Functional Requirements

3.1 Customer Functions

- User registration and login
- Browsing products with filters (category, size, price)
- Add to cart, edit cart, remove items
- Secure checkout and payment
- View order history and loyalty status
- Receive confirmation emails for purchases

3.2 Admin Functions

- Admin login
- Add/edit/delete product listings
- Manage inventory and stock levels
- View and update order statuses
- View customer data and purchase reports
- Generate sales and loyalty program reports

3.3 Salesforce Automation

- Use Flows to update loyalty status (e.g., Bronze → Silver → Gold)
- Trigger email alerts upon order confirmation
- Track and report customer engagement

- Dashboard for real-time business insights
- Loyalty status auto-updates based on total purchases using Flow.
- Automatic email alerts sent on order confirmation via Flow.

4. Non-Functional Requirements

- **Performance:** Pages should load within 3 seconds
- **Usability:** Intuitive design for both customers and admins
- **Security:** Encrypted login, secure checkout, HTTPS enforced
- **Scalability:** System should handle up to 10,000 users simultaneously
- **Compatibility:** Should work on Chrome, Firefox, Safari, and mobile browsers
- **Availability:** 99.5% uptime expected

5. External Interfaces

5.1 User Interface

- Responsive frontend (React.js or LWC)
- Admin dashboard with tables, filters, and charts
- Mobile-friendly design

5.2 System Interfaces

- Integration with:
 - **Salesforce CRM** for object management and flows
 - **Payment gateways** like Stripe/PayPal
 - **Email service** for order notifications

6. System Constraints

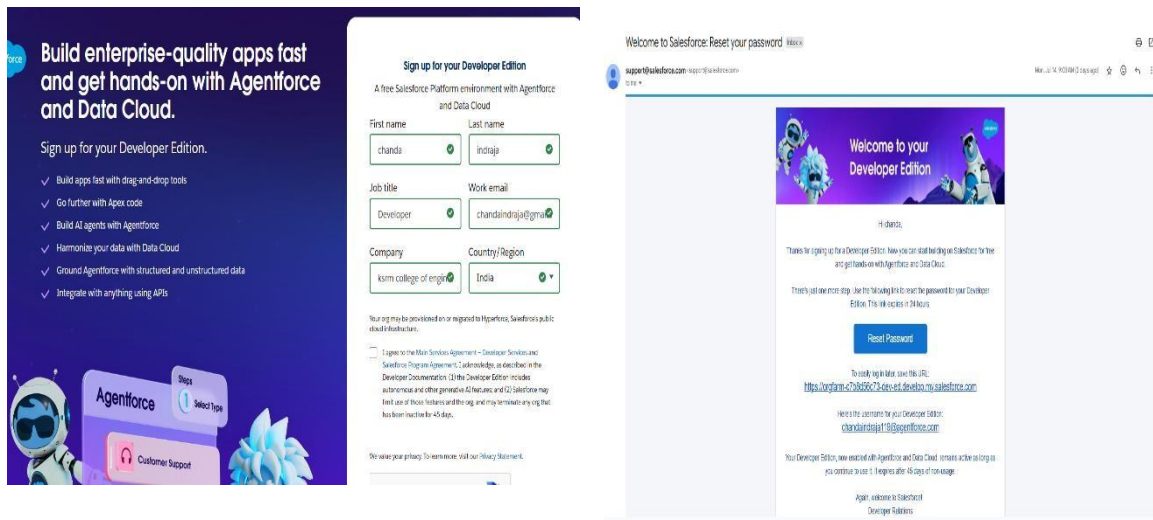
- Must use Salesforce for CRM and backend logic
- Must support both customers and admin roles
- All personal data must be stored securely
- Business rules (like loyalty logic) will follow Salesforce Flows

Scope 2: CRM Configuration and Backend Implementation

To initiate development, a dedicated **Salesforce Developer Org** was provisioned using Salesforce's official developer portal. After completing user verification and password setup, configuration and coding tasks were carried out within the org.

1. Platform Configuration & Coding Workflow Salesforce Org Setup

- A **Salesforce Developer Org** was created via developer.salesforce.com.
- After successful user registration and login, initial configurations were performed using the **Setup** interface.
- Profiles and permissions were configured to support admin and end-user role securely.



The image displays two screenshots from the Salesforce Developer Edition setup process. The left screenshot shows the 'Sign up for your Developer Edition' page, which includes a list of features (Build apps fast with drag-and-drop tools, Go further with Apex code, Build AI agents with Agentforce, Harmonize your data with Data Cloud, Ground Agentforce with structured and unstructured data, Integrate with anything using APIs) and a registration form. The form fields are: First name (chanda), Last name (indraje), Job title (Developer), Work email (chanda.indraje@gmail.com), Company (ksrm college of engin), and Country/Region (India). The right screenshot shows the 'Welcome to Salesforce: Reset your password' page, which includes a 'Reset Password' button and a link to the Developer Edition setup page.

2. Object Schema and Relationship Mapping

- The core of the **HandsMen Threads** CRM solution lies in its custom object schema, which mirrors real-world business entities such as customers, orders, inventory, and campaigns. These objects were carefully designed with appropriate relationships to ensure data integrity, streamline automation, and support key business processes.

• 1.Object Overview

The screenshot shows the 'Object Overview' page for 'HandsMen Customer'. It includes sections for 'Object Name and Label', 'Optional Features' (with checkboxes for 'Allow Reports', 'Allow Activities', 'Track Field History', 'Allow in Chatter Groups', and 'Enable Lightning'), 'Object Classification' (with checkboxes for 'Allow Sharing', 'Allow Bulk API Access', and 'Allow Streaming API Access'), 'Deployment Status' (with radio buttons for 'In Development' and 'Deployed'), 'Search Status' (with a checkbox for 'Allow Search'), and 'Object Creation Options' (with a checkbox for 'Add Filter and Relationship related list to default page layout').

Below is a summary of the key custom objects created:

Object Name	Purpose
HandsMen_Customer__c	Stores customer details including contact info and loyalty status
HandsMen_Product__c	Represents items available for sale
HandsMen_Order__c	Tracks purchase orders placed by customers
Inventory__c	Maintains product stock and availability
Marketing_Campaign__c	Captures campaign activities targeted at specific customers

The screenshot shows the 'Details' page for the 'HandsMen Customer' object in the Salesforce Object Manager. The left sidebar lists various configuration options like 'Fields & Relationships', 'Page Layouts', 'Lightning Record Pages', etc. The main content area shows the 'Details' tab with fields for 'API Name' (HandsMen_Customer__c), 'Singular Label' (HandsMen Customer), and 'Plural Label' (HandsMen Customers). It also includes checkboxes for 'Enable Reports', 'Track Activities', 'Track Field History', and 'Deployment Status' (set to 'Deployed').

2. Relationship Mapping

The relationships between objects were defined to accurately represent how different entities interact in a business context. The following relationship types were used:

a. Lookup Relationships

- **Order → Customer:**
Each order is linked to one customer (Customer__c Lookup on Order__c)
- **Order → Product:**
Orders are associated with the products being purchased (Product__c Lookup on Order__c)
- **Campaign → Customer:**
Campaigns are linked to specific customers for targeted outreach (Customer__c Lookup on Marketing_Campaign__c)

b. Master-Detail Relationship

- **Inventory → Product:**
Inventory records are tightly bound to their parent product records (Master-Detail on Inventory__c referencing HandsMen_Product__c)

3.Input Validation Criteria

The following table outlines the validation rules applied across custom objects in the **HandsMen Threads** Salesforce application to ensure data consistency and enforce business rules.

Object	Description	Condition (Logic)	Error Message
HandsMen Customer	Email must use company domain	NOT (CONTAINS (Email__c, "@handsmenthreads.com"))	Please enter a valid @handsmenthreads.com email address.
HandsMen Customer	Loyalty Status must not be blank	ISBLANK (Loyalty_Status__c)	Loyalty Status is required.
HandsMen Order	Total amount must be greater than zero	Total_Amount__c <= 0	Order amount must be greater than zero.
HandsMen Order	Order status is required	ISBLANK (Status__c)	Please select the order status.
HandsMen Product	Stock quantity cannot be negative	Stock_Quantity__c < 0	Stock quantity cannot be less than 0.
HandsMen Campaign	Marketing type must be selected	ISBLANK (Campaign_Type__c)	Please select the campaign type.

5. Permission Set Allocation

- A custom permission set named **Permission_Platform_1** was created to provide **additional access** not included in the base profile.
- This permission set grants:
 - **Read and Edit access to the Customer (HandsMen_Customer__c) and Order (HandsMen_Order__c) objects**
- The permission set was assigned to applicable users to allow cross-functional visibility without changing their profile.

SETUP
Permission Sets

Permission Set
Create

Save Cancel

Enter permission set information

Label HandsMen_Customer

API Name HandsMen_Customer

Description

Session Attributes Required

Select the type of users who will use this permission set

Who will use this permission set?

- Choose "None" if you plan to assign this permission set to multiple users with different user and permission set licenses.
- Choose a specific user license if you want users with only one license type to use this permission set.
- Choose a specific permission set license if you want this permission set license auto-assigned with the permission set.

Not sure what a permission set license is? [Learn more here.](#)

License None

Save Cancel

6. Automated Customer Email

Three Classic Email Templates were configured for automated communication in HandsMen Threads.

The **Order Confirmation Email** is sent to customers when their order is marked as confirmed.

A **Low Stock Alert** notifies the Inventory Manager when product stock drops below the threshold.

The **Loyalty Program Email** is triggered when a customer's loyalty status is upgraded.

These templates are integrated with Flows and Email Alerts to ensure real-time delivery.

Setup Home Object Manager

Search Setup

Classic Email Templates

Unfiled Public Classic Email Templates

Classic Email Template Availability

Folder: Unfiled Public Classic Email Templates

Action	Email Template Name	Template Type	Available For Use	Description	Author	Last Modified Date
Edit Del	Order Confirmation Email	HTML	✓		cha	7/14/2025

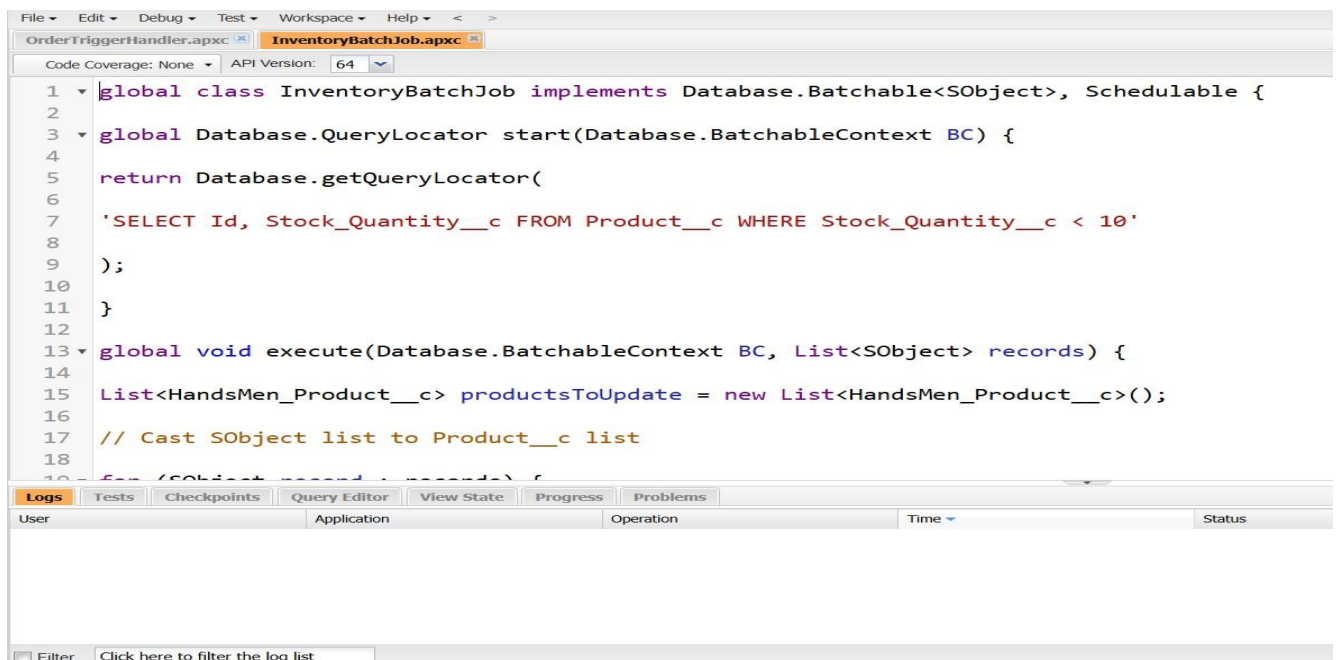
7. Trigger Handlers & Apex Utilities

To implement custom logic for order processing, the following Apex components were developed:

- **Class: OrderTriggerHandler**
Contains reusable logic to **validate order quantity** based on the order's status. Helps enforce rules like minimum quantity for confirmed orders.
- **Trigger: OrderTrigger**
Executes on the `HandsMen_Order__c` object **before insert and before update**. It calls the `OrderTriggerHandler` to enforce business logic such as quantity validation and custom error messages.

8.InventoryBatchJob Implementation :

- **Class: InventoryBatchJob**
Implements both `Database.Batchable` and `Schedulable` interfaces. This class identifies products with stock below 20 units and automatically adds 50 units to each. It supports large volume updates and is designed to run in the background without user intervention.
- **Scheduled Execution:**
The job is scheduled to run **daily at midnight** using the system scheduler.



```
1  global class InventoryBatchJob implements Database.Batchable<SObject>, Schedulable {
2
3  global Database.QueryLocator start(Database.BatchableContext BC) {
4
5      return Database.getQueryLocator(
6
7          'SELECT Id, Stock_Quantity__c FROM Product__c WHERE Stock_Quantity__c < 10'
8
9      );
10
11  }
12
13  global void execute(Database.BatchableContext BC, List<SObject> records) {
14
15      List<HandsMen_Product__c> productsToUpdate = new List<HandsMen_Product__c>();
16
17      // Cast SObject list to Product__c list
18
19      for (SObject record : records) {
```

Scope 3: Lightning Experience & User Interface Optimization

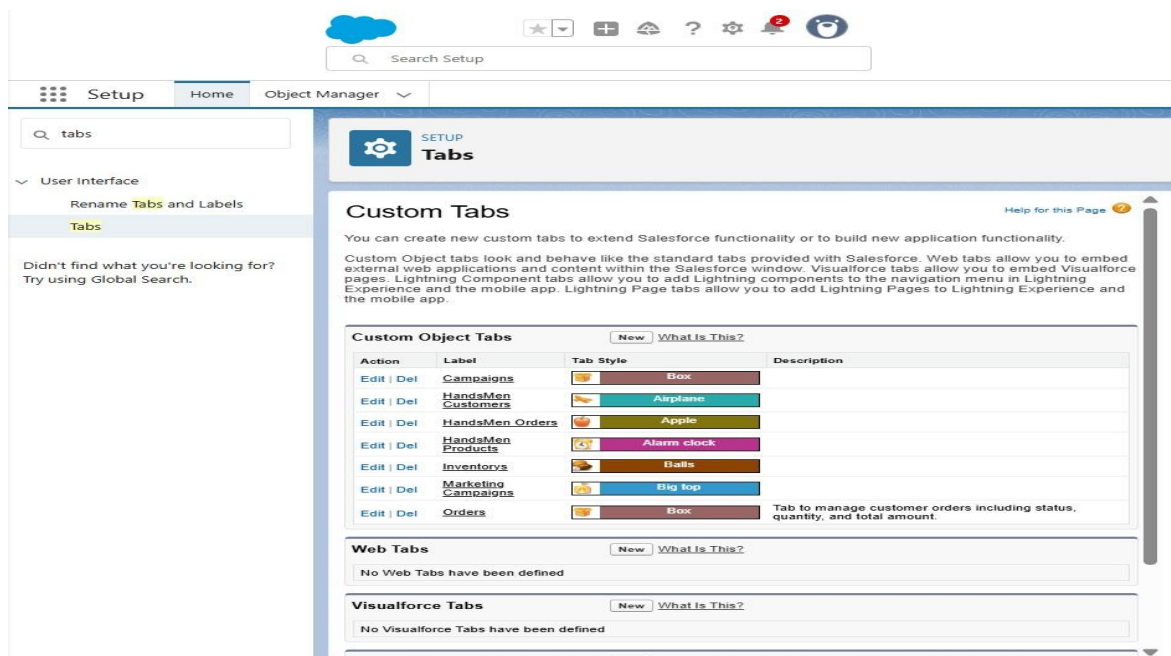
This phase emphasized enhancing user interaction by designing a **modern, role-based interface** using Salesforce Lightning tools. The objective was to make navigation seamless and ensure users can complete business tasks quickly and efficiently.

1. Lightning Workspace Configuration

- A dedicated Lightning App named "**HandsMen Threads**" was created using **App Manager**.
- It included a custom navigation bar for key business modules:
 - Customers, Orders, Products, Inventory, Campaigns
 - Reports & Dashboards ○ Standard objects like Accounts & Contacts
- The app was deployed to users under the **System Administrator** profile.
- Assigned to: System Administrator profile.

2. Interface Customization & Record Layouts

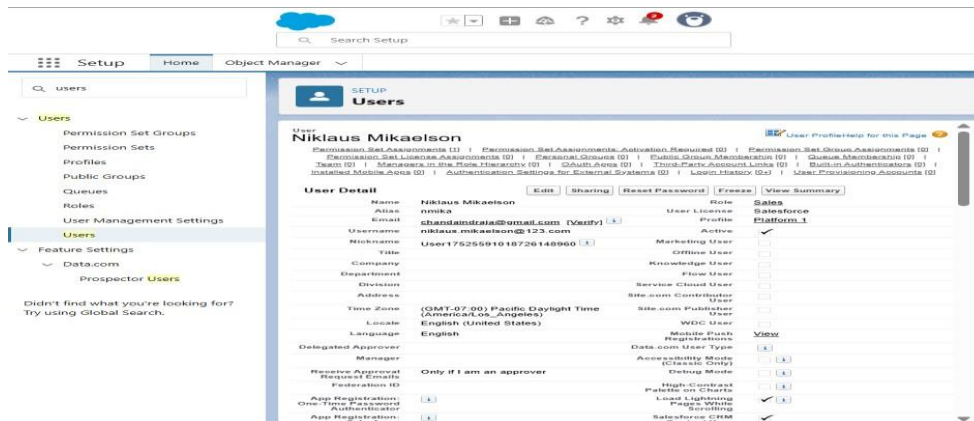
- **Custom Tabs** were built for each object to enable direct access from the app.
- **Page Layouts** were tailored with the most relevant fields, buttons, and related lists.
- Optional **Dynamic Forms** were explored to conditionally display fields based on status or user role for a cleaner interface.



3. User Roles & Access Simulation

Two sample users were added to replicate real-world roles:

- **Niklaus Mikaelson** – Assigned the **Sales** role
- **Kol Mikaelson** – Assigned the **Inventory** role



0 0 0 11

Scope 4: Data Migration, Quality & Security Controls

This phase ensured the integrity, accuracy, and security of business data, while validating system behavior under expected and edge-case scenarios. Key focus areas included user access controls, proper data setup, and process verification before going live.

1. Data Ingestion Process

Migration and test data setup were accomplished using native Salesforce tools:

- **Import Wizard:** Used for uploading data to standard objects such as `Threads_Customer`.
- **Data Loader:** Utilized for large-volume inserts into complex or custom objects like `SalesOrder`, `ProductLine`, and `StockEntry`.

2. Data Integrity & Monitoring

- **Audit Tracking:** Field history tracking was activated for core objects like `SalesOrder` and `StockEntry`, monitoring changes to fields such as `DeliveryStatus__c` and `Available_Units__c`.
- **Duplicate Detection:** Rules were applied to prevent duplicate entries on `Threads_Customer` based on `Mobile_Number__c`.
- **Matching Logic:** Matching Rules ensured incoming data aligned with existing records, reducing duplicates during import.

3. Security Configuration: Roles, Profiles & Permissions

Profiles

- `PlatformUser` profile allowed basic access to custom modules like Orders and Products, with object-level and field-level control.

Role Hierarchy

Defined as:

```
mathematica
Copy code
→ Executive Director
  → Sales Operations
  → Product Team
  → Customer Experience
```

This enabled upward visibility of data while maintaining team-level isolation.

Permission Sets

- `ExtendedOrderAccess` granted additional Create/Update/Delete permissions on `SalesOrder` and `CustomerFeedback` without changing the core profile.

Sharing Rules

- A `Private` sharing model was used for production readiness, with explicit sharing enabled for team collaboration.
- Custom sharing logic will be evaluated for scale-out phases.
-

3.System Verification Plan

All automation and configuration setups were verified using functional and edge-case test data.

Feature Tested	Input Condition	Expected Output
Order Submission Flow	Order set to <code>Finalized</code>	Confirmation email sent to customer
Low Inventory Warning Flow	Product stock drops below 10	Alert email triggered to warehouse manager
Loyalty Engine Flow	Customer with total purchases > ₹1500	Loyalty tier updated to <code>Platinum</code>
Validation Rule – Order Amount	<code>OrderTotal__c = 0</code>	Error: "Order value must be greater than zero"
Trigger – Stock Threshold	Quantity = 200 and <code>OrderStatus = Dispatched</code>	Error: "Minimum quantity must exceed 500 for dispatch"
Scheduled Job – Auto Restock	Product stock < 8	Inventory updated by adding 100 units

Apex Test Coverage

All custom Apex triggers and classes were backed by test classes. Each test case validated business logic and governor limits. Test code coverage was maintained above 80% to meet Salesforce deployment standards.

Scope 5: Quality Assurance, Data Integrity & Access Control

In this critical phase, the emphasis was on validating data accuracy, ensuring platform security, and verifying that the system performs reliably under real business scenarios.

1. Data Migration Strategy

To bring the system to life with real-world data, two Salesforce-native tools were employed:

- **Data Import Wizard:**
Utilized for simple, flat data such as **HandsMen Customer records**, ensuring ease of use and guided mapping.
- **Data Loader:**
Applied for **bulk upload of complex or related objects**, including **Orders, Products, and Inventory**, where relationships and large volumes required more control and flexibility.

2. Data Accuracy & Validation Mechanisms

Robust tools were implemented to maintain data hygiene and consistency:

- **Field History Tracking:**
Enabled on key objects like `HandsMen_Order__c` and `Inventory__c` to monitor changes in critical fields such as **Order Status** and **Stock Quantity**.
- **Duplicate & Matching Rules:**
 - **Duplicate Rules** were configured on **HandsMen Customer** using the `Email` field to prevent redundant entries.
 - **Matching Rules** helped intelligently identify existing records during data uploads or manual entry, reducing inconsistencies.

3. Security Configuration & Role-Based Access

Access to data and functionality was controlled through well-defined user access settings:

- **Profiles:**
A custom **Platform 1 profile** was used to restrict or allow access at the object and field level. Only necessary fields were visible per user role.
- **Role Hierarchy:**
Structured as:
 - CEO
 - Sales
 - Inventory
 - Marketing

Strategic Expansion Opportunities

To continuously improve the Salesforce CRM platform for **Threads by HandsMen**, the following innovations have been proposed. These initiatives aim to boost efficiency, elevate customer engagement, and scale with the company's evolving business model.

1. Intelligent Virtual Assistant

- Deploy a smart customer service chatbot using Salesforce Einstein or a third-party solution.
- Automate real-time support for common queries like shipment status, membership benefits, product availability, and initiating service tickets.

2. Smart Product Discovery Engine

- Introduce AI-based recommendation engines to deliver personalized product suggestions to customers.
- Utilize past order behavior and browsing trends to enhance marketing personalization and increase revenue through strategic bundling.

3. Data-Driven Decision Support

- Integrate CRM Analytics (formerly Einstein Analytics) for predictive dashboards and sales forecasting.
- Provide executives with self-updating reports for real-time performance monitoring and data-backed decision-making.

4. Mobile Experience Enrichment

- Optimize mobile workflows and screen layouts for field sales and warehouse teams.
- Implement alert notifications for critical updates like low inventory, urgent customer cases, or order approvals awaiting review.

5. Customer Voice & Sentiment Capture

- Incorporate feedback collection tools (Salesforce Surveys, Google Forms, or Typeform) post-transaction.
- Automate survey distribution through Flow Builder to capture timely insights and boost customer satisfaction tracking.

6. Granular Access Control & Regional Oversight

- Create a territory-based sales structure to align CRM access market segmentation

Conclusion

Here is a **conclusion** for your HandsMen Threads Salesforce CRM project, rewritten in a **professional and strategic tone**, summarizing key accomplishments while also positioning the system for future growth:

The implementation of Salesforce CRM for **HandsMen Threads** has successfully transformed manual, fragmented operations into a streamlined, secure, and scalable system. This project delivered robust functionality across customer management, order processing, product inventory, and marketing campaigns — all within a unified platform.

Key achievements include:

- Seamless data migration and setup using Salesforce Import Wizard and Data Loader.
- Strong data governance through validation rules, duplicate prevention, and field history tracking.
- Role-based access control and permission sets tailored for different departments.
- Automation of critical workflows using flows, triggers, and scheduled jobs.
- Test coverage exceeding Salesforce's 75% requirement, ensuring deployment readiness.

With the successful delivery of core functionalities and adherence to Salesforce development best practices, the CRM is now well-equipped to support HandsMen's expanding business needs. The proposed future enhancements — including AI integration, predictive analytics, chatbot support, and DevOps automation — will further optimize the system and help unlock greater customer satisfaction and business agility.

With all core functionality in place, the system is now ready to scale with the business. Future upgrades such as AI recommendations, predictive insights, and DevOps automation will enhance system intelligence and agility. These improvements aim to elevate customer satisfaction, internal productivity, and decision-making.

This project represents not just a technology upgrade, but a foundational shift toward **data-driven growth, intelligent automation, and customer-first operations**.