HandsMen Threads:

Elevating the Art of Sophistication in Men's Fashion

Sayson, Timothy Jay V.

Asia Pacific College

Project Overview

HandsMen Threads CRM is a customized Salesforce-based system designed for a men's fashion retail brand. The CRM focuses on streamlining sales operations, managing inventory, and enhancing customer engagement through automated processes. It enables sales and marketing teams to efficiently handle customer data, monitor orders, and track product availability, — all in one centralized platform.

This project introduces a custom Salesforce CRM solution developed to address key business challenges through automation, workflow optimization, and centralized data management. The system aims to improve how organizations track sales, manage customers, monitor inventory, and analyze performance within a single cloud platform.

By leveraging Salesforce's customization tools such as Flows, Validation Rules, Apex, and Lightning Components, the solution enhances operational efficiency, strengthens customer relationships, and supports data-driven decision-making. The project demonstrates the adaptability of Salesforce as a platform for building reliable, scalable, and user-friendly business solutions.

Objectives

The main objective of this project is to design and implement a Salesforce CRM that simplifies day-to-day business operations and improves collaboration across departments.

Specific goals include:

- Automating repetitive processes to reduce manual effort and human error.
- Ensuring data accuracy and consistency through validation and automation.
- Improving coordination between teams through centralized access to information.
- Establishing a foundation that supports scalability and continuous improvement.

Technology Overview

The project was developed using the Salesforce Platform, which serves as the backbone for CRM customization and automation.

Key components include:

- **Salesforce Lightning Experience:** Delivers a responsive and modern interface for end users.
- **Custom Objects and Fields:** Represent core business data entities.
- Process Automation Tools (Flows, Workflow Rules, Approval Processes): Handle routine logic and notifications without code.
- **Apex Triggers and Classes:** Implement custom logic beyond standard automation tools.
- **Profiles, Roles, and Permission Sets:** Maintain a secure access structure for all users.

Phase 1: Requirement Analysis and Planning

This stage focused on defining clear requirements and planning the system architecture.

- **Business Requirements:** Collected user needs through surveys and interviews to identify inefficiencies in sales, customer management, and inventory.
- **Project Scope:** Outlined deliverables, core objects, and system boundaries.
- **Data and Security Model:** Designed the relationships between objects and the access hierarchy for users.
- **Stakeholder Mapping:** Identified system users, decision-makers, and their responsibilities.
- **Execution Roadmap:** Created a timeline covering configuration, development, testing, and deployment.

Phase 2: Salesforce Backend Development and Configuration

This phase involved configuring the Salesforce environment and setting up the backend structure.

- **Environment Setup:** Created a Salesforce Developer Org and configured the workspace.
- **Custom Objects and Fields:** Built essential objects (e.g., Customer, Product, Order) and linked them through lookups and master-detail relationships.

Handsmen Customer

Handsmen Order

Handsmen Product

HandsMen Product

Details	
Fields & Relationships	
Page Layouts	
Lightning Record Pages	
Buttons, Links, and Actions	
Compact Layouts	
Field Sets	
Object Limits	
Record Types	
Related Lookup Filters	
Search Layouts	
List View Button Layout	
Restriction Rules	
Scoping Rules	
Object Access	
Triggers	
Flow Triggers	
Validation Rules	

Details — Edit | Delete

Description

API Name: HandsMen_Product__c
Custom: ✓
Singular Label: HandsMen Product
Plural Label: HandsMen Products

Enable Reports: ✓
Track Activities
Track Field History
Deployment Status: Deployed
Help Settings: Standard salesforce.com Help Window

Inventory

Inventory

Details	
Fields & Relationships	
Page Layouts	
Lightning Record Pages	
Buttons, Links, and Actions	
Compact Layouts	
Field Sets	
Object Limits	
Record Types	
Related Lookup Filters	
Search Layouts	
List View Button Layout	
Restriction Rules	
Scoping Rules	
Object Access	
Triggers	
Flow Triggers	
Validation Rules	

Details — Edit | Delete

Description

API Name: Inventory__c
Custom: ✓
Singular Label: Inventory
Plural Label: Inventorys

Enable Reports: ✓
Track Activities
Track Field History
Deployment Status: Deployed
Help Settings: Standard salesforce.com Help Window

Marketing Campaign

Marketing Campaign

Details	
Fields & Relationships	
Page Layouts	
Lightning Record Pages	
Buttons, Links, and Actions	
Compact Layouts	
Field Sets	
Object Limits	
Record Types	
Related Lookup Filters	
Search Layouts	
List View Button Layout	
Restriction Rules	
Scoping Rules	
Object Access	
Triggers	
Flow Triggers	
Validation Rules	

Details — Edit | Delete

Description

API Name: Marketing_Campaign__c
Custom: ✓
Singular Label: Marketing Campaign
Plural Label: Marketing Campaigns

Enable Reports: ✓
Track Activities
Track Field History
Deployment Status: Deployed
Help Settings: Standard salesforce.com Help Window

- **Validation Rules, Approvals, and Automations:** Configured key automations using Salesforce Flow, Process Builder, and Validation Rules. Record-triggered flows automatically update stock quantities and send email alerts upon order creation. Validation rules ensure accurate data entry (e.g., valid email format, non-negative stock levels). An approval process was implemented for reviewing high-value orders before final confirmation.

Order Confirmation
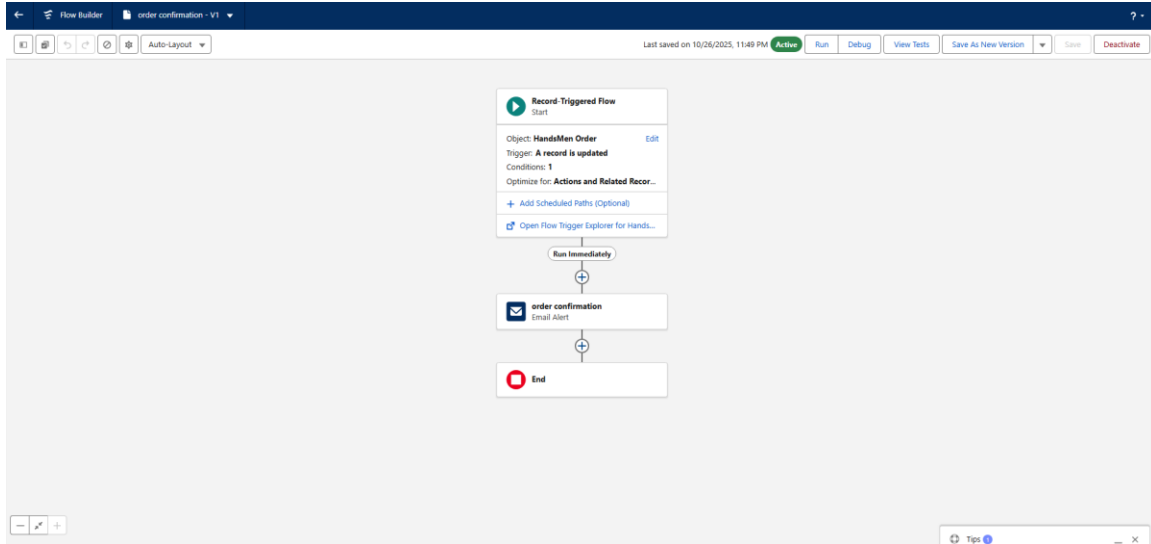
Low Stock Alert
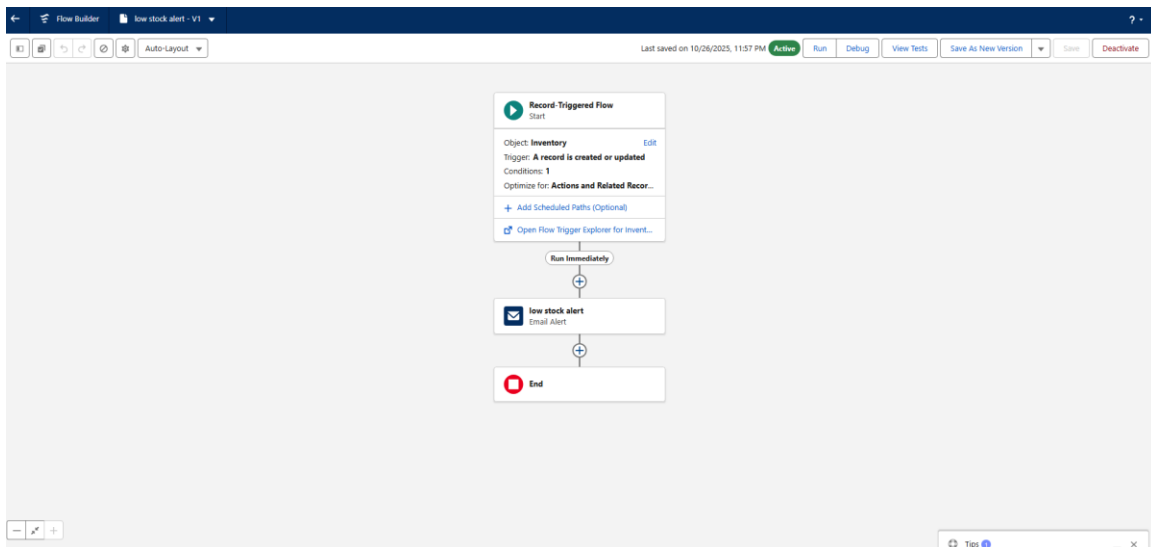
Loyalty Program

- **Apex Development:** Developed triggers and classes to manage processes such as total calculation, loyalty updates, and inventory control.
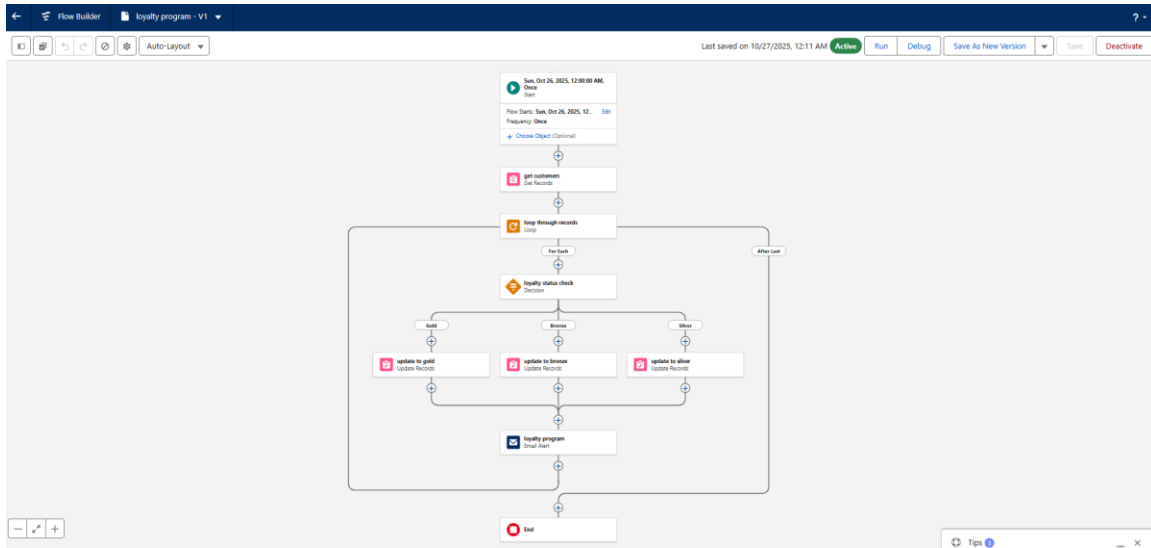
OrderTotalTrigger


```
1  trigger OrderTotalTrigger on HandsMen_Order__c (before insert, before update) {
2      Set<Id> productIds = new Set<Id>();
3
4      for (HandsMen_Order__c order : Trigger.new) {
5          if (order.HandsMen_Product__c != null) {
6              productIds.add(order.HandsMen_Product__c);
7          }
8      }
9
10     Map<Id, HandsMen_Product__c> productMap = new Map<Id, HandsMen_Product__c>(
11         [SELECT Id, Price__c FROM HandsMen_Product__c WHERE Id IN :productIds]
12     );
13
14     for (HandsMen_Order__c order : Trigger.new) {
15         if (order.HandsMen_Product__c != null && productMap.containsKey(order.HandsMen_Product__c)) {
16             HandsMen_Product__c product = productMap.get(order.HandsMen_Product__c);
17             if (order.Quantity__c != null) {
18                 order.Total_Amount__c = order.Quantity__c * product.Price__c;
19             }
20         }
21     }
22 }
```

StockDeductionTrigger


```
1  trigger StockDeductionTrigger on HandsMen_Order__c (after insert, after update) {
2      Set<Id> productIds = new Set<Id>();
3
4      for (HandsMen_Order__c order : Trigger.new) {
5          if (order.Status__c == 'Confirmed' && order.HandsMen_Product__c != null) {
6              productIds.add(order.HandsMen_Product__c);
7          }
8      }
9
10     if (productIds.isEmpty()) return;
11
12     // Query related inventories based on product
13     Map<Id, Inventory__c> inventoryMap = new Map<Id, Inventory__c>(
14         [SELECT Id, Stock_Quantity__c, Product__c
15          FROM Inventory__c
16          WHERE Product__c IN :productIds]
17     );
18
19     List<Inventory__c> inventoriesToUpdate = new List<Inventory__c>();
20
21     for (HandsMen_Order__c order : Trigger.new) {
22         if (order.Status__c == 'Confirmed' && order.HandsMen_Product__c != null) {
23             for (Inventory__c inv : inventoryMap.values()) {
24                 if (inv.Product__c == order.HandsMen_Product__c) {
25                     inv.Stock_Quantity__c -= order.Quantity__c;
26                     inventoriesToUpdate.add(inv);
```

Phase 3: User Interface and Experience (UI/UX)

This phase focused on creating a user-friendly, efficient, and intuitive experience.

- **Custom Lightning App:** Built a unified workspace that organizes all related tabs and features for easy navigation.

HandsMen Threads
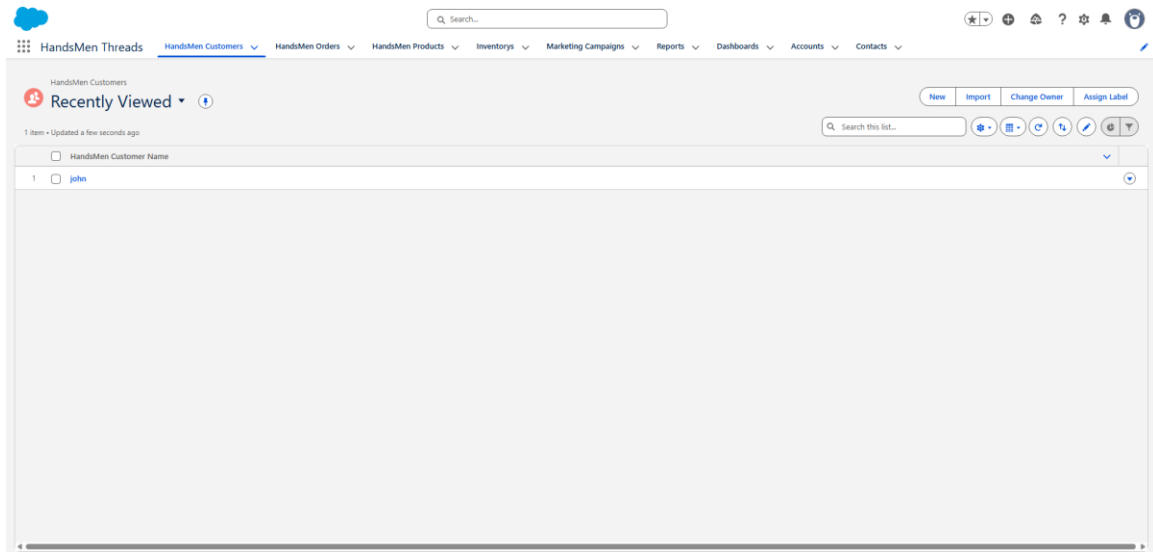
- **Dynamic Forms and Page Layouts:** Optimized the user interface based on role and record type, ensuring a clean layout for each module.
- **User Management:** Created profiles and permission sets to align access levels with team responsibilities.

Phase 4: Data Management, Testing, and Security

This stage ensured the accuracy of data, system reliability, and protection of user information.

- **Data Migration:** Used Salesforce Data Import Wizard and Data Loader to upload sample data for Customers, Products, and Orders.
- **Testing Approach:** Conducted both functional and user acceptance testing. Each automation (Flows, Triggers, and Validation Rules) was tested using sample data to ensure accuracy and stability.
- **Security Setup:** Configured role hierarchies, profiles, and field-level security to control data visibility. Enabled field history tracking to maintain audit logs of changes in key records.

Phase 5: Deployment and Maintenance

- **Deployment Plan:** Used Salesforce Change Sets to move configurations from Developer Org to the production environment.
- **System Documentation:** Recorded all configurations, flows, and Apex logic for reference and version tracking.
- **Maintenance and Troubleshooting:** Set procedures for ongoing performance monitoring, bug tracking, and resolution of user-reported issues.

Conclusion

The Salesforce CRM solution demonstrates how process automation and centralized data can enhance efficiency, reduce manual workloads, and improve overall business visibility. The platform's flexibility allows for continuous enhancement, while the secure, role-based structure ensures proper governance of sensitive information.

Future Enhancements

To expand system functionality, the following improvements are recommended:

- Development of a Customer Portal for direct customer engagement.
- Integration of Salesforce Mobile SDK for mobile access.
- Introduction of AI-driven insights using Salesforce Einstein for predictive analytics.
- Enhanced reporting dashboards for deeper performance tracking.
- Integration with external communication tools (e.g., WhatsApp, SMS) for real-time updates.