



Republic of the Philippines
POLYTECHNIC UNIVERSITY OF THE PHILIPPINES
Office of the Vice President for Branches and Satellite Campuses
LOPEZ QUEZON CAMPUS
Bachelor of Science in Information Technology

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

Prepared Documentation By:

Elaija Jasmin C. Yap

BSIT-3

PUP-LOPEZ CAMPUS



ABSTRACT

The proposed project is a complete Salesforce CRM that was designed to work with HandsMen Threads, which is a high-end men fashion and men tailoring company. The key points of the system are to simplify the business processes, improve the interaction with customers, and increase the reliability of the data in various departments.

Five major custom objects, Customer, Order, Product, Inventory and Marketing Campaign were used in designing a comprehensive data model to handle information that is critical to the business. Record-Triggered Flows, Scheduled Flows, Email Alerts, and Apex Triggers were used to automate business processes to aid in confirming orders, computing loyalty status, and proactive management of stock.

Sales, Inventory and Marketing teams were set up with validation rules and role based security model to keep a check on data integrity/sensitive data. There is a planned Apex batch job that also deals with automated stock tracking.

Overall, this CRM solution would greatly enhance customer experience in terms of personal communication, operational efficiency due to automation, and scalability to facilitate further development of HandsMen Threads in Salesforce platform.

OBJECTIVE

This project aims at designing and deploying a customized Salesforce CRM, which will centralize the customer, order, inventory, products and marketing activities of HandsMen Threads. The purposes of this system are to simplify everyday business processes and advance quality, efficient and customer-oriented operations.

The CRM is planned to accomplish the following:

- Automate key business functions like order verifications, loyalty programs, inventory alerts and communication triggers.
- Ensure proper and standard records by the validation rules and structured data entry.
- Improve interdepartmental coordination with role-based access, where each team works on the right privileges.
- Enhance the interaction with the customers with targeted emails and automated loyalty programs.
- Offer time-sensitive access to inventory, customer relations and sales to make informed decisions.



TECHNOLOGY DESCRIPTION

Salesforce Platform

Salesforce is a cloud-based CRM platform enabling organizations to manage customer data, automate business processes, and support sales, marketing, and service operations. It includes both declarative (click-based) and programmatic (Apex-based) tools for custom process development.

Custom Objects

Custom Objects store unique business data. They function similarly to database tables. Examples:

- **Customer__c** – Customer information such as name, contact details, and loyalty status
- **Product__c** – Product details including price, SKU, and category
- **Order__c** – Order information such as quantity, price, and status
- **Inventory__c** – Stock levels and warehouse locations
- **Marketing_Campaign__c** – Campaign schedules and promotional activities

Tabs

Tabs provide easy navigation to view and manage records for each custom object.

Custom App

A Salesforce Lightning App groups tabs and tools used for the HandsMen Threads CRM, allowing users to access all relevant objects in one workspace.

Profiles

Profiles define user permissions such as object access, field visibility, and system capabilities.

Roles

Roles control hierarchical data visibility and determine which records users can view or manage.

Permission Sets

Permission Sets grant additional access to users without changing their assigned profile.

Validation Rules



Validation Rules enforce correct data entry and ensure compliance with business requirements.

Examples:

- Email must contain “@gmail.com”
- Stock quantity must not drop below zero

Email Templates & Alerts

Predefined email templates are used for automated messaging such as order confirmations and loyalty updates. Email Alerts trigger these templates through flows or automation.

Flows

Flows automate business logic such as sending emails, updating records, and validating conditions.

Example: Send order confirmation email upon new order creation.

Apex

Apex is Salesforce’s programming language used for creating complex business logic.

Examples:

- Trigger to update Total Amount on orders
- Trigger to adjust inventory stock
- Trigger to compute customer loyalty status

DETAILED EXECUTION OF PROJECT PHASES

Phase 1: Requirement Analysis & Planning

Understanding Business Requirements

Analyzed operational needs of HandsMen Threads including customer tracking, order processing, personalized communication, and inventory monitoring.

Defining Project Scope & Objectives

Defined CRM modules covering customers, products, inventory, orders, and marketing.

Data Model & Security Model Design

Prepared custom objects, fields, relationships, validation rules, and security components including profiles, roles, and permission sets.



Stakeholder Mapping

Identified key stakeholder groups: Sales Team, Inventory Manager, Marketing Department, and System Administrator.

Execution Roadmap

Planned sequential development from data modeling → backend automation → UI customization → testing → deployment.

Phase 2: Salesforce Development — Backend & Configurations

Environment Setup

Created a Salesforce Developer Org and configured initial admin settings.

Custom Objects, Fields & Validation Rules

Developed five custom objects and relevant fields. Applied validation rules for email format and total amount checks.

Automation Tools (Flows, Processes, Approvals)

- Record-Triggered Flows for order confirmation, loyalty updates.
- Scheduled Flows for periodic stock alerts.
- Email Alerts for customer notifications.

Apex Development

- **Order Total Trigger** – Calculates total order amount based on quantity and unit price.
- **Stock Update Trigger** – Deducts product stock when an order is placed.
- **Loyalty Status Trigger** – Automatically updates customer loyalty tier based on lifetime purchase amount.

Screenshots:



Republic of the Philippines
POLYTECHNIC UNIVERSITY OF THE PHILIPPINES
Office of the Vice President for Branches and Satellite Campuses
LOPEZ QUEZON CAMPUS
Bachelor of Science in Information Technology

TABLE	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
HandsMen Customer	HandsMen_Customer_v1	Custom Object		11/24/2025	✓
HandsMen Order	HandsMen_Order_v1	Custom Object		11/24/2025	✓
HandsMen Product	HandsMen_Product_v1	Custom Object		11/24/2025	✓

Fig. 1 Created Objects

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedBy	Lookup(User)		
Email	Email_v1	Email		
Last Name	LastName_v1	Text(50)		
Full Name	FullName_v1	Formula (Text)		
HandsMen Customer Name	Name	Text(50)	✓	
Last Modified By	LastModifiedBy	Lookup(User)		
Last Name	LastName_v1	Text(50)		
Loyalty Status	LoyaltyStatus_v1	Picklist		
Owner	OwnerId	Lookup(Owner Group)	✓	
Phone	Phone_v1	Phone		
Order Number	OrderNumber_v1	Number(18, 0)		

Fig. 2 Fields in Handsmen Customer

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedBy	Lookup(User)		
Customer Email	CustomerEmail_v1	Email		
HandsMen Customer	HandsMen_Customer_v1	Lookup(HandsMen Customer)	✓	
HandsMen Order Number	Name	Auto Number	✓	
HandsMen Product	HandsMen_Product_v1	Lookup(HandsMen Product)	✓	
Last Modified By	LastModifiedBy	Lookup(User)		
Owner	OwnerId	Lookup(Owner Group)	✓	
Quantity	Quantity_v1	Number(18, 0)		
Status	Status_v1	Picklist		
Total Amount	TotalAmount_v1	Number(18, 0)		

Fig. 3 Fields in Handsmen Order



Republic of the Philippines
POLYTECHNIC UNIVERSITY OF THE PHILIPPINES
Office of the Vice President for Branches and Satellite Campuses
LOPEZ QUEZON CAMPUS
Bachelor of Science in Information Technology

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedBy	Lookup(User)		
HandsMen Product Name	Name	Text(80)		✓
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(Owner/Shop)		✓
Price	Price_s	Currency(16, 0)		
SKU	SKU_s	Text(60)		
Stock Quantity	Stock_Quantity_s	Number(16, 0)		

Fig. 4 Fields in Handsmen Product

HandsMen Customer Validation Rule

Validation Rule Detail

Rule Name: Email

Error Condition Formula: NOT (CONTAINS>Email_s, "@gmail.com")

Error Message: Please Use Correct Email

Error Location: Top of Page

Created By: Edith J. J. 11/24/2025, 3:34 AM

Modified By: Edith J. J. 11/24/2025, 3:34 AM

Fig. 5 Handsmen Customer Validation Rule

HandsMen Order Validation Rule

Validation Rule Detail

Rule Name: Total_Amount

Error Condition Formula: Total_Amount_s >= 3

Error Message: Please Enter Correct Amount

Error Location: Total Amount

Created By: Edith J. J. 11/24/2025, 3:31 AM

Modified By: Edith J. J. 11/24/2025, 3:31 AM

Fig 6. Handsmen Order Validation Rule

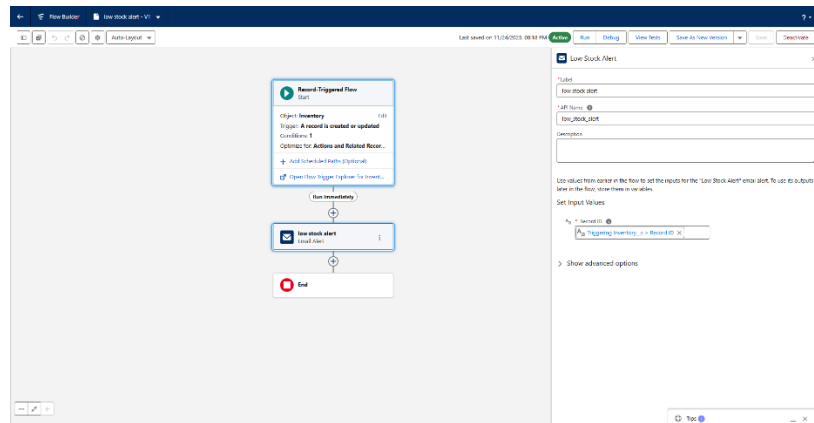


Fig. 7 Low Stock Alert Flow

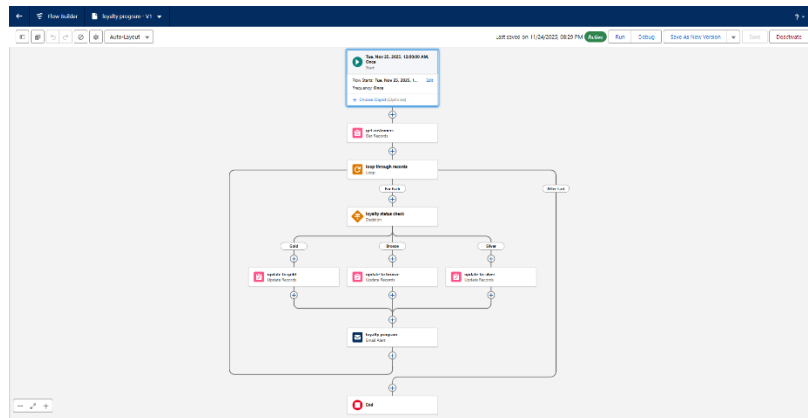


Fig. 8 Loyalty Program Flow

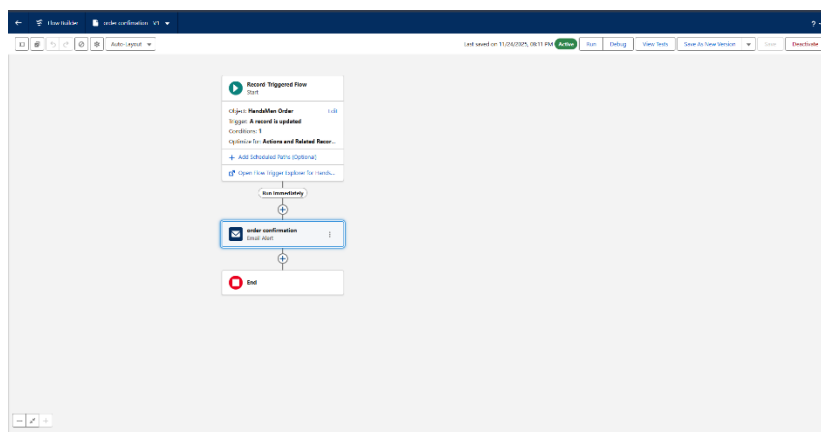


Fig. 9 Order Confirmation Flow



```
File Edit Debug Test Workspace Help < >
StockDeductionTrigger.aspx OrderTotalTrigger.aspx
Code Coverage: None API Version: 65

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, HandsMen_Product__c
15         FROM Inventory__c
16         WHERE HandsMen_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.HandsMen_Product__c == order.HandsMen_Product__c) {
25                    inv.Stock_Quantity__c -= order.Quantity__c;
26                    inventoriesToUpdate.add(inv);
27                    break;
28                }
29            }
30        }
31    }
32
33    if (!inventoriesToUpdate.isEmpty()) {
34        update inventoriesToUpdate;
35    }
36 }
```

Fig. 10 Stock Deduction Trigger

```
File Edit Debug Test Workspace Help < >
StockDeductionTrigger.aspx OrderTotalTrigger.aspx
Code Coverage: None API Version: 65

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 }
```

Fig. 11 Order Total Trigger



```
File • Edit • Debug • Test • Workspace • Help • < • >
StockDeductionTrigger.aspx • OrderTotalTrigger.aspx • InventoryBatchJob.aspx
Code Coverage: None • API Version: ES • No
1 • global class InventoryBatchJob implements Database.Batchable<SObject>, Schedulable {
2
3 • global Database.QueryLocator start(Database.BatchableContext BC) {
4
5 • return Database.getQueryLocator( 'SELECT Id, Stock_Quantity__c FROM Product__c WHERE Stock_Quantity__c < 10'); }
6
7 • global void execute(Database.BatchableContext BC, List<SObject> records) {
8
9 • List<HandsMen_Product__c> productsToUpdate = new List<HandsMen_Product__c>();
10
11 • // Cast SObject list to Product__c list
12
13 • for (SObject record : records) {
14
15 • HandsMen_Product__c product = (HandsMen_Product__c) record;
16
17 • product.Stock_Quantity__c += 50; // Restock logic
18
19 • productsToUpdate.add(product); }
20
21 • if (!productsToUpdate.isEmpty()) {
22
23 • try { update productsToUpdate;
24
25 • } catch (DmlException e) { System.debug('Error updating inventory: ' + e.getMessage()); } } }
26
27 • global void finish(Database.BatchableContext BC) { System.debug('Inventory Sync Completed'); }
28
29 • // Scheduler Method
30
31 • global void execute(SchedulableContext SC) {
32
33 • InventoryBatchJob batchJob = new InventoryBatchJob();
34
35 • Database.executeBatch(batchJob, 200); } }
```

Fig. 12 Inventory Batch Job Class

Phase 3: UI/UX Development & Customization

Lightning App Setup

Created the *HandsMen Threads* Lightning App with all relevant tabs.

Page Layouts & Dynamic Forms

Customized layouts to display customer information, loyalty status, order details, and inventory summaries.

User Management

Created users and assigned appropriate profiles and roles (Sales, Inventory, Marketing).

Reports & Dashboards

Set up initial reports for order tracking and inventory monitoring.

Screenshots:

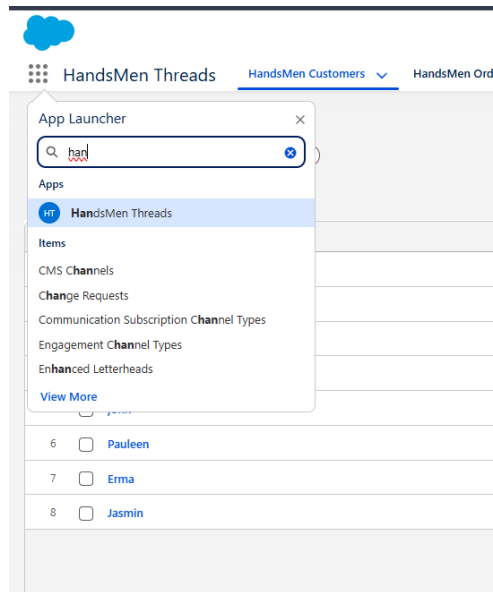


Fig. 13 Lightning App Setup

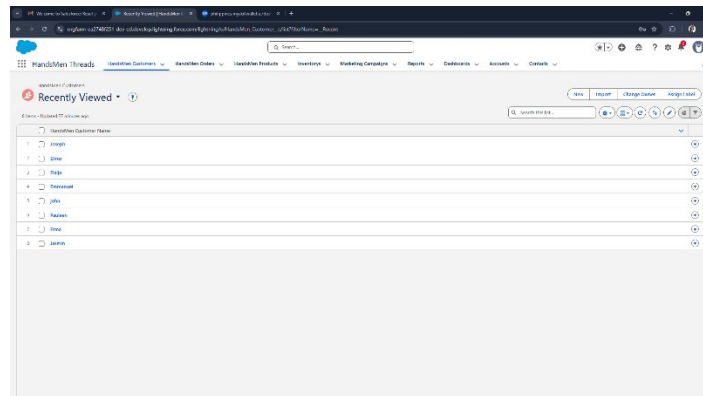


Fig. 14 Page Layout

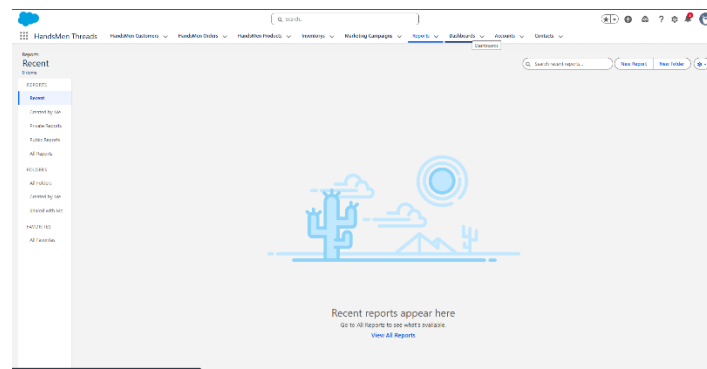


Fig. 15 Reports and Dashboards



Republic of the Philippines
POLYTECHNIC UNIVERSITY OF THE PHILIPPINES
Office of the Vice President for Branches and Satellite Campuses
LOPEZ QUEZON CAMPUS
Bachelor of Science in Information Technology

Fig. 16. User Niklaus Mikaelson

Fig. 17 User Daniel Mikaelson

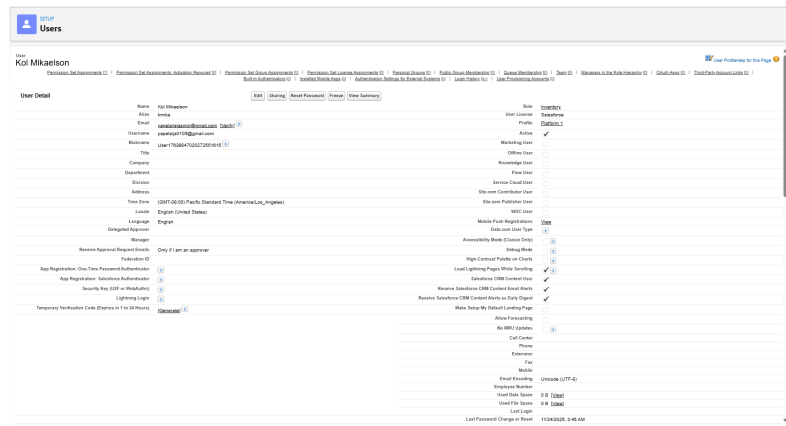


Fig. 18 Kol Mikaelson

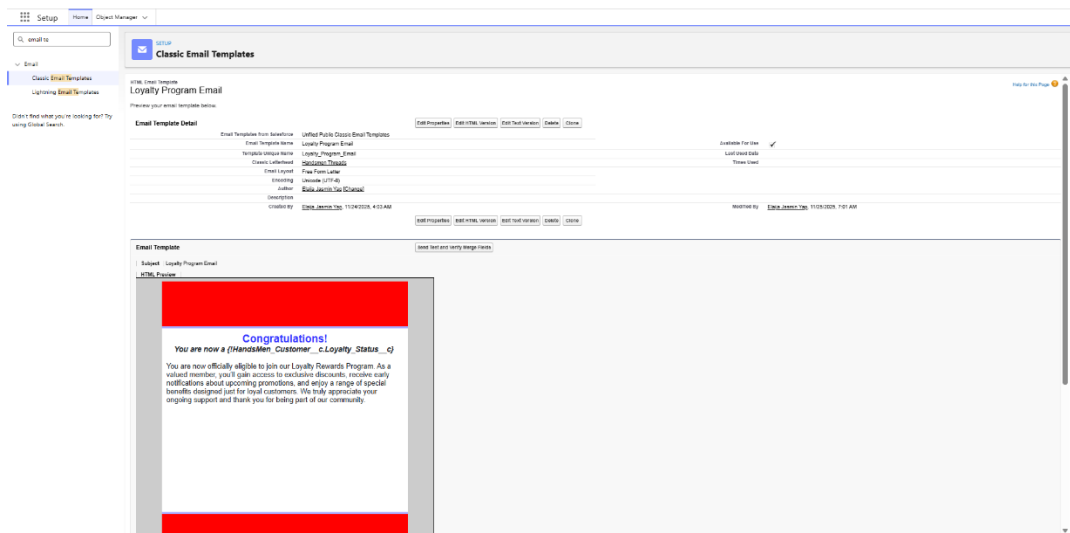


Fig. 19 Classic Email Template

Phase 4: Data Migration, Testing & Security

Data Migration Tools

Utilized Data Import Wizard to load customer, product, and inventory data.

Security Configurations

Implemented:

- Profiles & Permission Sets
- Roles & Role Hierarchy
- Sharing Rules



- Field History Tracking
- Duplicate & Matching Rules

Test Classes

Created Apex Test Classes to ensure reliability of triggers and logic.

Testing Approach

Prepared and executed test cases for:

- Booking/order creation
- Approval processes
- Flow executions
- Triggered automation
- Inventory deductions
- Loyalty updates

Screenshots:

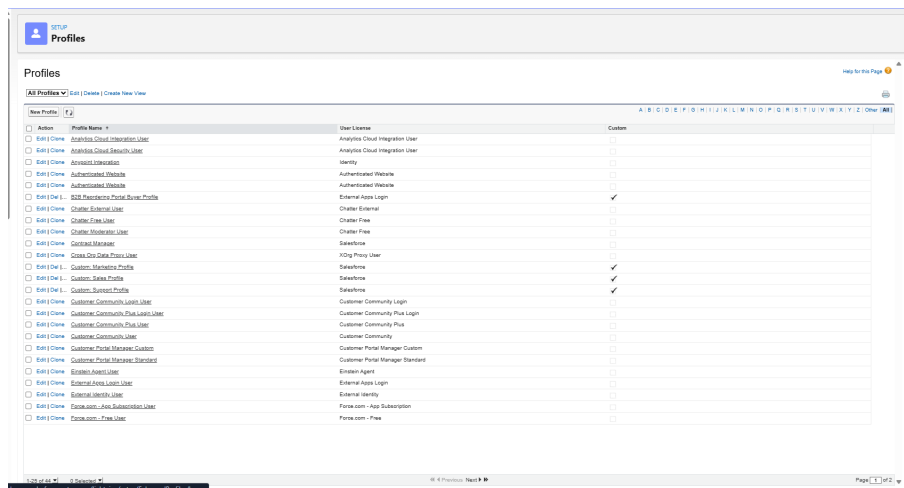


Fig. 20 Profiles

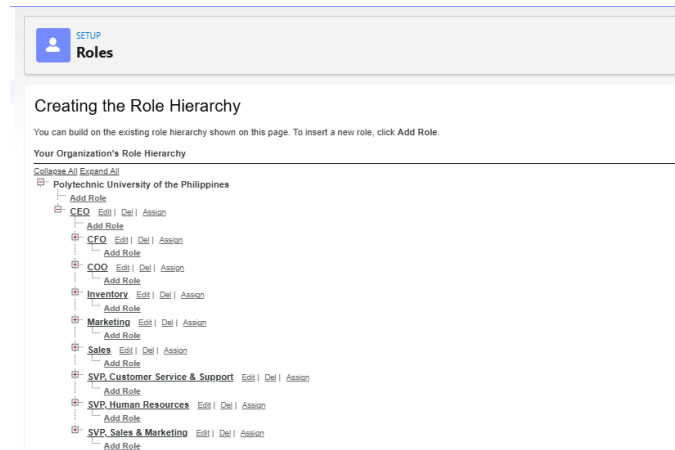


Fig. 21 Roles

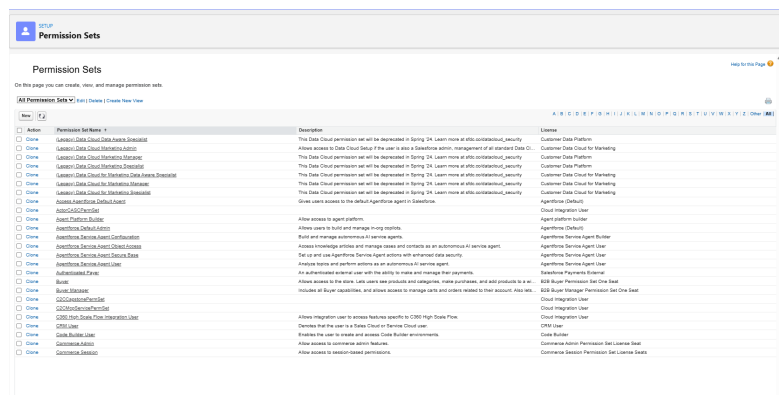


Fig. 22 Permission Sets

New HandsMen Customer

* = Required Information

Information

HandsMen Customer Name Owner Elajja Jasmin Yap

Email

Phone

Loyalty Status

FirstName

LastName

Total Purchases

Fig. 23 Customer Validation Rule



Phase 5: Deployment, Documentation & Maintenance

Deployment Strategy

Used Change Sets to move components from development to production.

Maintenance Plan

Outlined procedures for monitoring automation, reviewing error logs, updating security roles, and performing regular data checks.

Troubleshooting Approach

Documented steps to diagnose issues involving flows, triggers, validation rules, and user permissions.

PROJECT EXPLANATION (REAL-WORLD USE CASE)

1. Customer Registration

Store staff enters customer information. Validation rules ensure correct email format. Loyalty status initializes as Bronze.

2. Product & Inventory Management

Admin adds new products. Inventory tracks available stock.

3. Order Processing

Customer places an order. System calculates total automatically through Apex.

4. Inventory Adjustment

Stock is updated immediately when orders are confirmed.

5. Loyalty Program Automation

System updates customer loyalty levels (Bronze → Silver → Gold) based on accumulated purchases.

6. Email Notifications

Customers receive automated emails for order confirmations and loyalty changes.

7. User Roles in the System



Sales users handle customer and order data; inventory managers track stock; marketing users manage campaigns.

CONCLUSION

HandsMen Threads CRM is an appropriate Salesforce-based tool used to enhance the management of customers, process sales, and monitor inventory as well as carry out a loyalty program. With the collection of declarative tools and Apex automation, the system provides operations faster, proper data processing, and improved communication with clients. It is also scalable due to its modular design.

Future Enhancements:

Order tracking and Loyalty portal Customer Community Portal.

Mobile App Salesforce Mobile SDK.

Complex analytics dashboards.

Product recommendations based on Salesforce Einstein, with AI.

WhatsApp/SMS features of automated, customer notifications.