

HandsMen Threads

Elevating the Art of Sophistication in Men's Fashion
Capstone Project — Salesforce Demo Documentation

Author: Principio, Angelo M.

Date: 11/18/2025

Environment: Salesforce Developer Org / Lightning Experience

Table of Contents

1. Project Overview
2. Custom App & Architecture
3. Data Model and Key Fields
4. Creating Data Records Step by Step
5. Business Automations Flows and Scheduled Processes
6. Apex Triggers and Batch Jobs Summaries
7. Email Templates Placeholders
8. Testing Validation and QA Scenarios
9. Reports and Dashboards Placeholders
10. Deployment Notes and Permissions
11. Future Scope and Recommendations
12. Appendix Sample SOQL and Test Checklist

Project Overview

Project name: HandsMen Threads

Description: A custom Salesforce CRM to manage bespoke men's tailoring: customers, products, inventory, orders, loyalty tiers, and automated communication.

Business goals

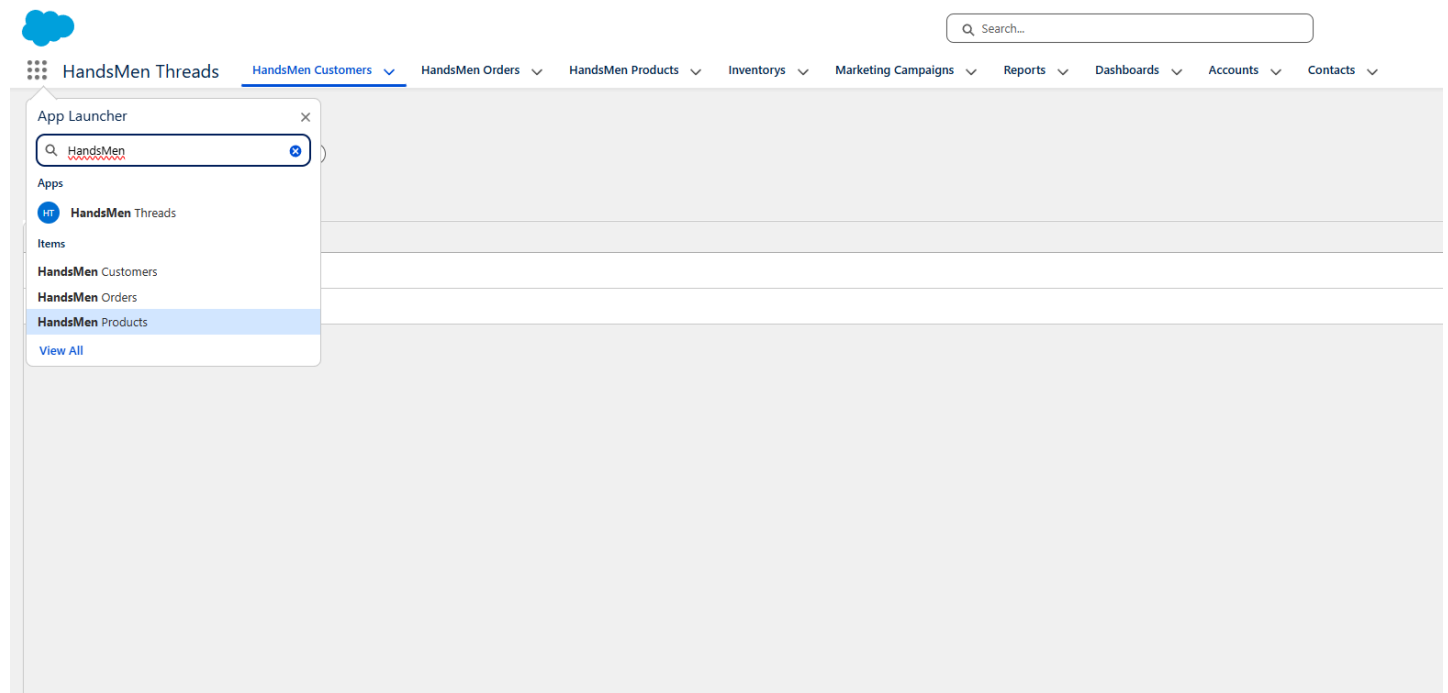
- Streamline tailoring workflows from inquiry to delivery
- Improve customer experience with personalized interactions and loyalty rewards
- Ensure accurate inventory and order management to avoid stock issues
- Automate routine processes to reduce manual work and accelerate fulfillment

Salesforce tools used :

- Custom Objects and Tabs
- Lightning Custom App
- Record-Triggered Flows and Scheduled Flows
- Validation Rules and Email Alerts
- Apex Triggers and Batch Apex
- Reports and Dashboards

Script for Project Overview tab

HandsMen Threads is a custom Salesforce CRM solution designed to elevate men's bespoke fashion management. The goal of this project is to streamline custom tailoring operations, enhance customer experience with personalized interactions, optimize order and inventory management, and automate business processes using Salesforce capabilities. This platform was built using Custom Objects, Flows, Validation Rules, Apex Triggers, Email Alerts, and Reports & Dashboards. This demo walks through the design, configuration, automation, and end-to-end functionality of the CRM.



Custom App and Architecture

Custom App: HandsMen Threads Lightning App — central hub for operations and admin

tasks **Assigned profiles:** System Administrator, Sales, Service, Inventory Manager

Object architecture

Custom Objects

- Handsman Customer — customer name, email, phone, loyalty status, total purchases
- Handsman Orders — order number, product, customer, status, quantity, total amount, customer email
- Handsman Products — SKU, price, name, stock quantity
- Inventory — product, stock quantity, warehouse location
- Marketing Campaigns — campaign number, customer, start date, end date

Standard Objects used

- Accounts and Contacts
- Reports and Dashboards

Data flow

Customer → Orders → Product → Inventory. Orders update Inventory when confirmed; scheduled processes reconcile pending orders with stock.

HandsMen Threads

HandsMen Customers

HandsMen Orders

HandsMen Products

Inventorys

Marketing Campaigns

Reports

Dashboards

Accounts

Contacts

HandsMen Customer

John Dela Cruz

New ContactEditNew Opportunity

RelatedDetails

HandsMen Customer Name

John Dela Cruz

Owner

Angelo Principio

Email

johnde@gmail.com

Phone

09123456789

Loyalty Status

Silver

FirstName

John

LastName

Dela Cruz

FullName

John Dela Cruz

Total Purchases

Created By

Angelo Principio, 11/17/2025, 9:05 AM

Last Modified By

Angelo Principio, 11/17/2025, 9:20 AM

HandsMen Threads

HandsMen Customers

HandsMen Orders

HandsMen Products

Inventorys

Marketing Campaigns

Reports

Dashboards

Accounts

Contacts

HandsMen Order

O-0007

RelatedDetails

HandsMen OrderNumber

O-0007

Owner

Angelo Principio

HandsMen Product

Shorts

HandsMen Customer

John Dela Cruz

Status

Confirmed

Quantity

600

Total Amount

1,800

Customer Email

principioangelo24@gmail.com

Created By

Angelo Principio, 11/17/2025, 9:07 AM

Last Modified By

Angelo Principio, 11/17/2025, 9:08 AM

HandsMen Threads

HandsMen Customers

HandsMen Orders

HandsMen Products

Inventorys

Marketing Campaigns

Reports

Dashboards

Accounts

Contacts

HandsMen Product

Shorts

RelatedDetails

HandsMen Order Name

Shorts

Owner

Angelo Principio

SKU

0002

Price

\$3

Stock Quantity

1,000

Created By

Angelo Principio, 11/17/2025, 9:06 AM

Last Modified By

Angelo Principio, 11/17/2025, 9:06 AM

Inventory
I -0006

Related Details

Inventory Number

I -0006

HandsMen Product

[Shorts](#)

Stock Quantity

1,400

Stock Status

Available

Warehouse

Warehouse Shorts

Created By

[Angelo Principio](#), 11/17/2025, 9:06 AM

Last Modified By

[Angelo Principio](#), 11/17/2025, 9:08 AM

Data Model and Key Fields

Handsman Customer

- FullName
- Email
- Phone
- Loyalty_Status__c
- Total_Purchases__c

Handsman Products

- SKU__c
- Name
- Price__c
- Stock_Quantity__c

Inventory

- HandsMen_Product__c (Master-Detail)
- CreatedById
- Stock_Quantity__c
- Stock_Status__c
- Warehouse__c

Handsman Orders

- Name (Order Number)
- HandsMen_Customer__c (lookup)
- Customer_Email__c
- HandsMen_Product__c (lookup)
- Quantity__c
- Status__c (Pending Confirmed Cancelled)

- Total_Amount__c
- Order_Date__c

Validation rules examples

Email must contain "gmail.com" → Error text: Please fill correct Gmail

The screenshot shows the 'HandsMen Customer Validation Rule' configuration page in Salesforce. The page is titled 'HandsMen Customer Validation Rule' and includes a 'Validation Rule Edit' section. The 'Rule Name' is 'Email'. The 'Error Condition Formula' is 'NOT CONTAINS(Email_c, "gmail.com")'. The 'Error Message' is 'Please fill Correct Gmail'. The 'Error Location' is set to 'Top of Page'. The page also includes a 'Functions' list and a 'Check Syntax' button.

Creating Data Records Step by Step

1 Create Handsman Customer

Steps: App Launcher → HandsMen Threads → New Handsman Customer → fill fields → Save Demo note: Enter invalid email "johndc@xyz.com" → validation error "Please fill correct Gmail" → change to "johndc@gmail.com" → Save success.

The screenshot shows the 'New HandsMen Customer' form. The form includes fields for 'HandsMen Customer Name' (John Dela Cruz), 'Email' (johndc@xyz.com), 'Phone', 'Loyalty Status' (None), 'FirstName', 'LastName', and 'Total Purchases'. A validation error message is displayed: 'We hit a snag. Review the errors on this page. Please fill Correct Gmail'. The 'Save' button is disabled, and the 'Cancel' button is visible.

2 Create Handsman Product

Fields: Order Name: Shorts; SKU 0002; Price \$3; Quantity 1000;

HandsMen Threads

HandsMen Customers

HandsMen Orders

HandsMen Products

Inventorys

Mar

HandsMen Product

Shorts

Related

Details

HandsMen Order Name

Shorts

SKU

0002

Price

\$3

Stock Quantity

1,000

Created By

Angelo Principio, 11/17/2025, 9:06 AM

Owner

Anc

Last Mo

Anc

3 Create Inventory record

New Product → HandsMen Product Shorts → Stock Quantity 1400 → Warehouse Shorts.

HandsMen Threads

HandsMen Customers

HandsMen Orders

HandsMen Products

Inventorys

Inventory

I -0006

Related

Details

Inventory Number

I -0006

HandsMen Product

[Shorts](#)

Stock Quantity

1,400

Stock Status

Available

Warehouse

Warehouse Shorts

Created By

Angelo Principio, 11/17/2025, 9:06 AM

La

c

4 Create an Order

Handsman Orders → New → select Customer Product Quantity → Save → Total Amount auto-calculates via Apex.

HandsMen ThreadsHandsMen CustomersHandsMen OrdersHandsMen ProductsInventorysMarketing CampaignsReportsDashboardsAccountsContacts

HandsMen OrderO-0007

RelatedDetails

HandsMen OrderNumberO-0007

HandsMen ProductShorts

HandsMen CustomerJohn Dela Cruz

StatusConfirmed

Quantity600

Total Amount1,800

Customer Emailprincipioangelo24@gmail.com

Created ByAngelo Principio · 11/17/2025, 9:07 AM

OwnerAngelo Principio

Last Modified ByAngelo Principio · 11/17/2025, 9:08 AM

Business Automations Flows and Scheduled Processes

This section details automations to demonstrate system intelligence.

Automated Order Confirmation Record Triggered Flow

Trigger: Order.Status changes Pending → Confirmed
Actions: Send confirmation email to customer; update Inventory.Stock_Available minus Quantity
Demo example: John DC orders 600 shorts; stock 2000 → new stock 1400; confirmation email sent.

Flow Builder

order confirmation - V1

Auto-Layout

Record-Triggered Flow

Start

Object: HandsMen Order

Trigger: A record is updated

Conditions: 1

Optimize for: Actions and Related Records

+ Add Scheduled Paths (Optional)

Open Flow Trigger Explorer for Hands...

Run Immediately

+

order confirmation

Email Alert

+

End

Last saved on 11/17/2025, 12:13 AM

Active

Run

Debug

View Tests

Save As New Version

Save

Deactivate

Configure Start

Select Object

Select the object whose records trigger the flow when they're created, updated, or deleted.

* Object

HandsMen Order

Configure Trigger

Trigger the Flow When:

☐ A record is created

☒ A record is updated

☐ A record is created or updated

☐ A record is deleted

Set Entry Conditions

Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed. Minimizing unnecessary flow executions helps to conserve your org's resources.

If you create a flow that's triggered when a record is updated, we recommend first defining entry conditions. Then select the **Only when a record is updated to meet the condition requirements** option for When to Run the Flow for Updated Records.

Condition Requirements

All Conditions Are Met (AND)

Field

Status

Operator

Equals

Value

Is Confirmed

+ Add Condition

When to Run the Flow for Updated Records

☐ Every time a record is updated and meets the condition requirements

☒ Only when a record is updated to meet the condition requirements

Optimize Flow

Optimize the Flow for:

Fast Field Updates

Update fields on the record that triggers the flow to run. This high-performance flow runs **before the record is saved** to the database.

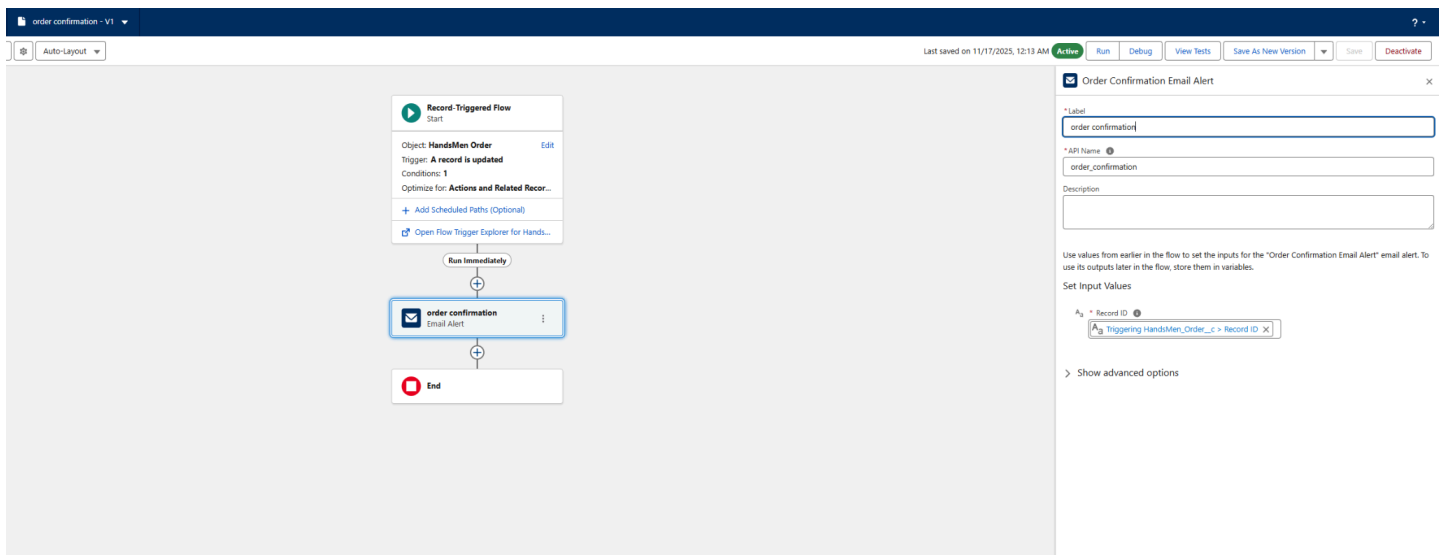
Actions and Related Records

Update any record and perform actions, like send an email. This more flexible flow runs **after the record is saved** to the database.

Is this flow making an external callout or connecting to an external system?

An asynchronous path is required for flows that involve external systems.

Add Asynchronous Path



Low Stock Alert Record Triggered Flow

Trigger: Inventory update

Condition: Stock_Quantity < 5

Actions: Send low-stock email to inventory manager

Sample email:

Dear Inventory Manager,

This is to inform you that the stock for the following product is running low:

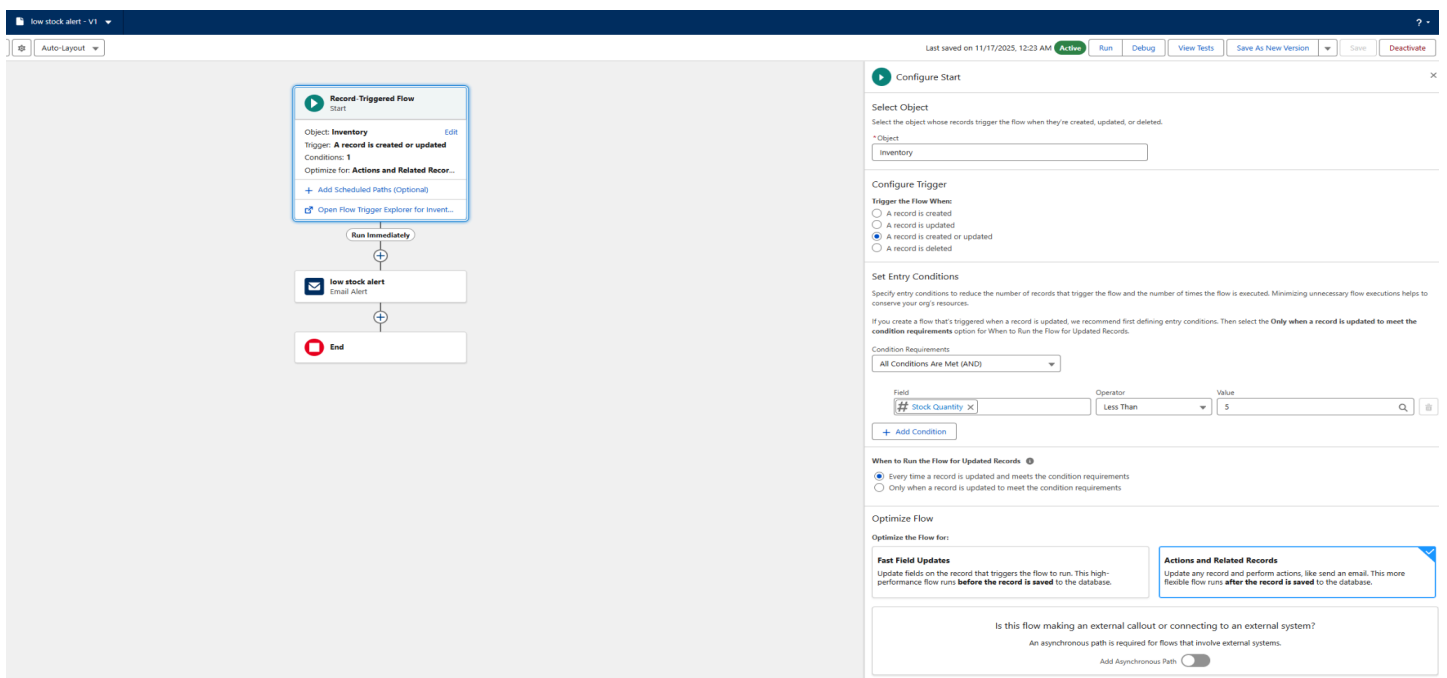
Product Name: {!Inventory__c.HandsMen_Product__c}

Current Stock Quantity: {!Inventory__c.Stock_Quantity__c}

Please take the necessary steps to restock this item immediately.

Best Regards,

Inventory Monitoring System





SETUP

Classic Email Templates

Edit Text Email Template

Low Stock Alert

Use merge fields to personalize your email content. You can add substitute text to any merge field. Substitute text displays only if the merge record does not contain data for that field. Enter substitute text after a comma in the merge field {!NullValue(Contact.FirstName,"Sir or Madam")}. Click on the link below to see a sample email template.

[View Sample Template](#)

Note that the Description field is for internal use only. It will be listed as the title of any email activities you log when sending mass email.

Available Merge Fields		
Select Field Type	Select Field	Copy Merge Field Value
<input type="text" value="Contact Fields"/>	<input type="text"/>	<input type="text"/>

Copy and paste the merge field value into your template below.

Email Template Edit

Save

Save & New

Cancel

Email Template Information

Folder	<input type="text" value="Unfiled Public Classic Email Templates"/>
Available For Use	<input checked="" type="checkbox"/>
Email Template Name	<input type="text" value="Low Stock Alert"/>
Template Unique Name	<input type="text" value="Low_Stock_Alert"/>
Encoding	<input type="text" value="Unicode (UTF-8)"/>
Description	<input type="text"/>
Subject	<input type="text" value="Low Stock Alert Email"/>
Email Body	<div>Dear Inventory Manager, This is to inform you that the stock for the following product is running low: Product Name: {!Inventory__c.HandsMen_Product__c} Current Stock Quantity: {!Inventory__c.Stock_Quantity__c} Please take the necessary steps to restock this item immediately. Best Regards, Inventory Monitoring System</div>

Save

Save & New

Cancel

Loyalty Program Scheduled Flow

Schedule: Daily

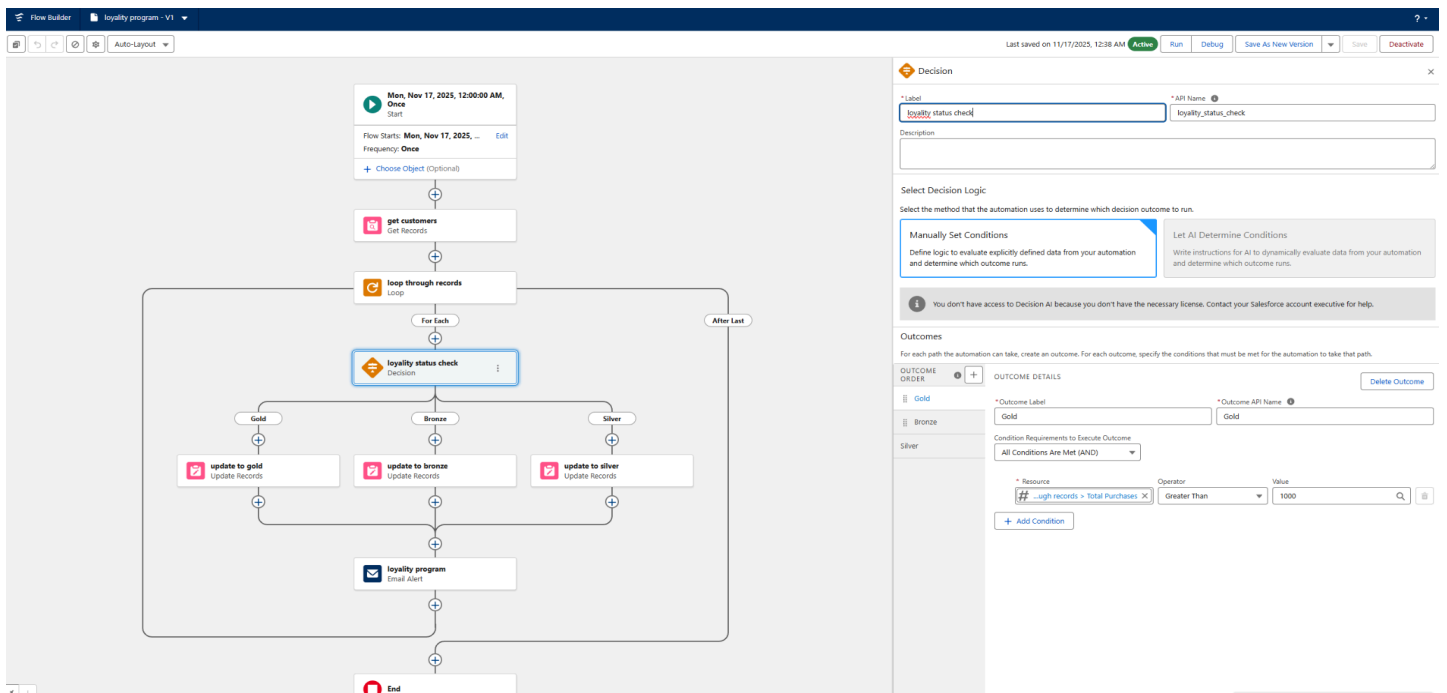
Logic: Total_Purchases thresholds set Loyalty_Status

1000 Gold

500–1000 Silver

≤ 500 Bronze

Actions: Update Loyalty_Status__c and send status update email



Apex Triggers and Batch Jobs Summaries

Order Total Calculation Trigger

When before insert before update on Handsman Orders

Function: $\text{order.Total_Amount_c} = \text{order.Quantity_c} * \text{product.Price_c}$; $\text{Total_Amount_c} = \text{Product.Price_c} \times \text{Quantity_c}$

```
StockDeductionTrigger.apxt | OrderTotalTrigger.apxt
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 }
```

Stock Deduction Trigger

Use-case: Automatic deduction of stocks

Function: `inv.Stock_Quantity__c -= order.Quantity__c;`

```
StockDeductionTrigger.apxt | OrderTotalTrigger.apxt
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 }
```

Email Templates Placeholders

Order Confirmation Email

Subject: Your order #{!HandsMen_Order__c.Name} has been confirmed!

Body sample:

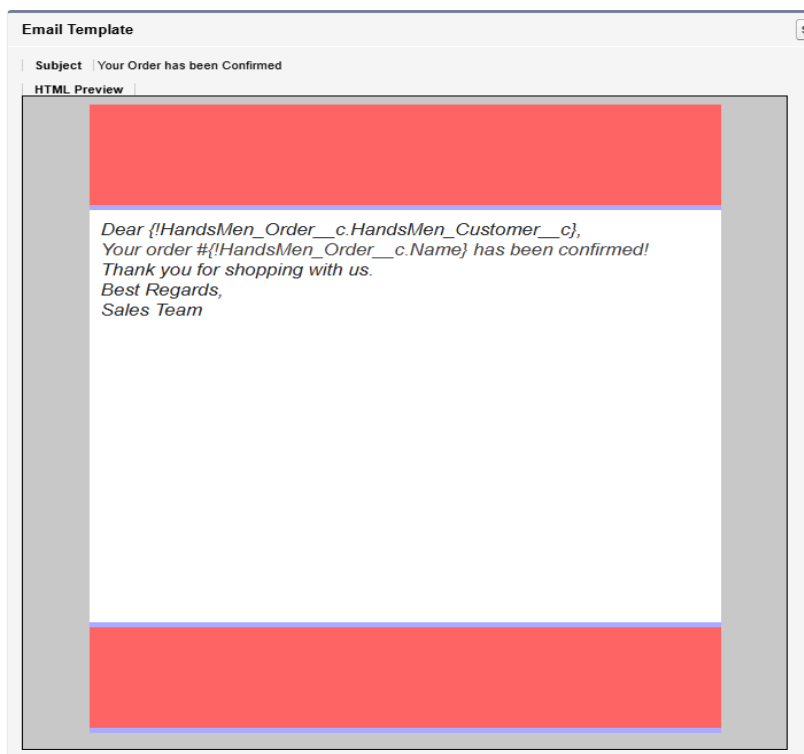
Dear {!HandsMen_Order__c.HandsMen_Customer__c},

Your order #{!HandsMen_Order__c.Name} has been confirmed!

Thank you for shopping with us.

Best Regards,

Sales Team



Low Stock Alert Email

Subject: Current Stock Quantity: {!Inventory__c.Stock_Quantity__c}

Body sample:

Dear Inventory Manager,

This is to inform you that the stock for the following product is running low:

Product Name: {!Inventory__c.HandsMen_Product__c}

Current Stock Quantity: {!Inventory__c.Stock_Quantity__c}

Please take the necessary steps to restock this item immediately.

Best Regards,

Inventory Monitoring System

Email Template

Subject

Low Stock Alert Email

Plain Text Preview

Dear Inventory Manager,
This is to inform you that the stock for the following product is running low:
Product Name: {!Inventory__c.HandsMen_Product__c}
Current Stock Quantity: {!Inventory__c.Stock_Quantity__c}
Please take the necessary steps to restock this item immediately.
Best Regards,
Inventory Monitoring System

Loyalty Status Update Email

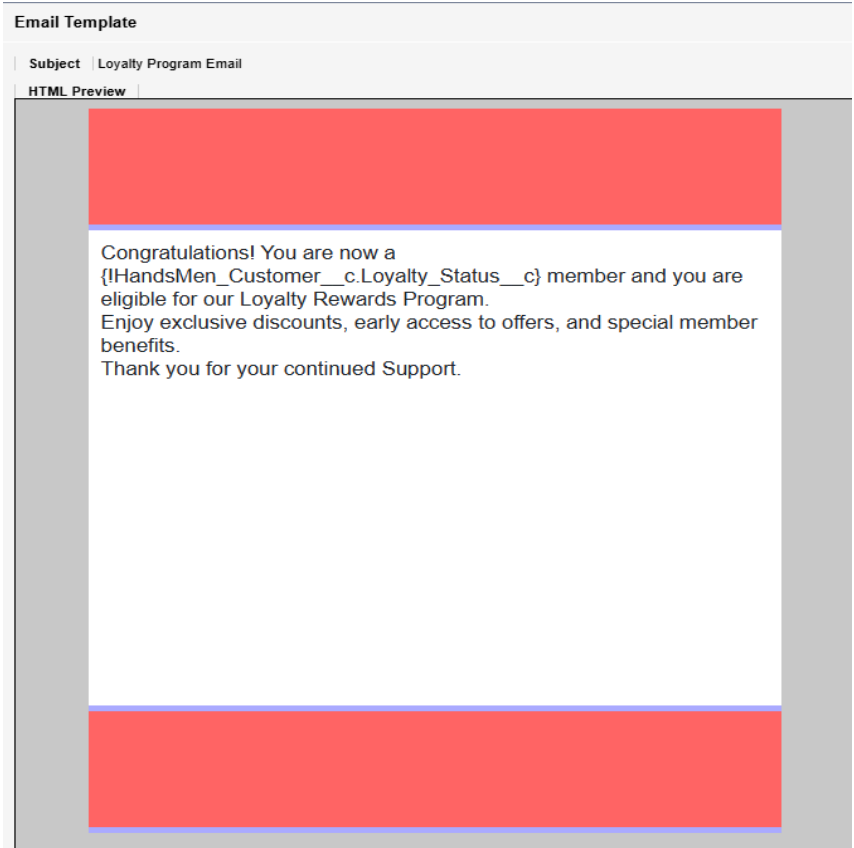
Subject: Congratulations! You are now a {!HandsMen_Customer__c.Loyalty_Status__c} member

Body sample:

Congratulations! You are now a {!HandsMen_Customer__c.Loyalty_Status__c} member and you are eligible for our Loyalty Rewards Program.

Enjoy exclusive discounts, early access to offers, and special member benefits.

Thank you for your continued support.



Conclusion

The HandsMen Threads CRM successfully integrates customer management, order lifecycle, inventory tracking, loyalty programs, smart automation, email communication, and Apex logic for advanced processing.

This project showcases how Salesforce CRM can streamline bespoke tailoring operations, improve customer experience, and provide real-time data insights for the men's fashion industry.