Nogot, Josephine Claire G. Polytechnic University of the Philippines - San Juan Campus
BSIT 4-1

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

Project Overview:

The Handsmen Threads CRM project transforms the traditional, manual operations of our bespoke tailoring business into a centralized, cloud-based system. By leveraging Salesforce, we aim to streamline client measurement tracking, inventory management, and appointment scheduling, ensuring a seamless experience from the initial fitting to the final suit delivery.

Objectives:

1. **Operational Efficiency:** Automate manual booking and order tracking to reduce human error.
2. **Data Integrity:** Securely store customer body measurements and preferences in a single digital location.
3. **Inventory Control:** Real-time tracking of fabric usage to prevent material shortages.

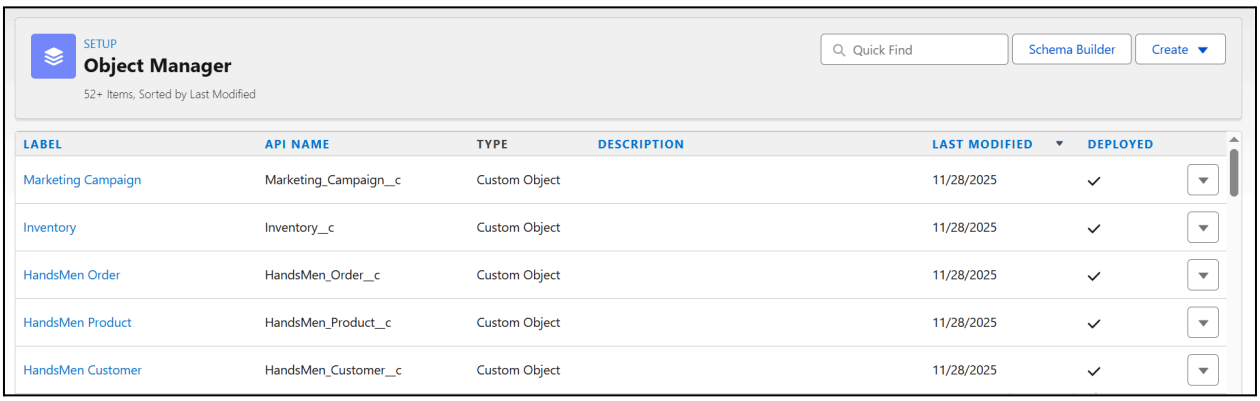
Phase 1:Requirement Analysis & Planning

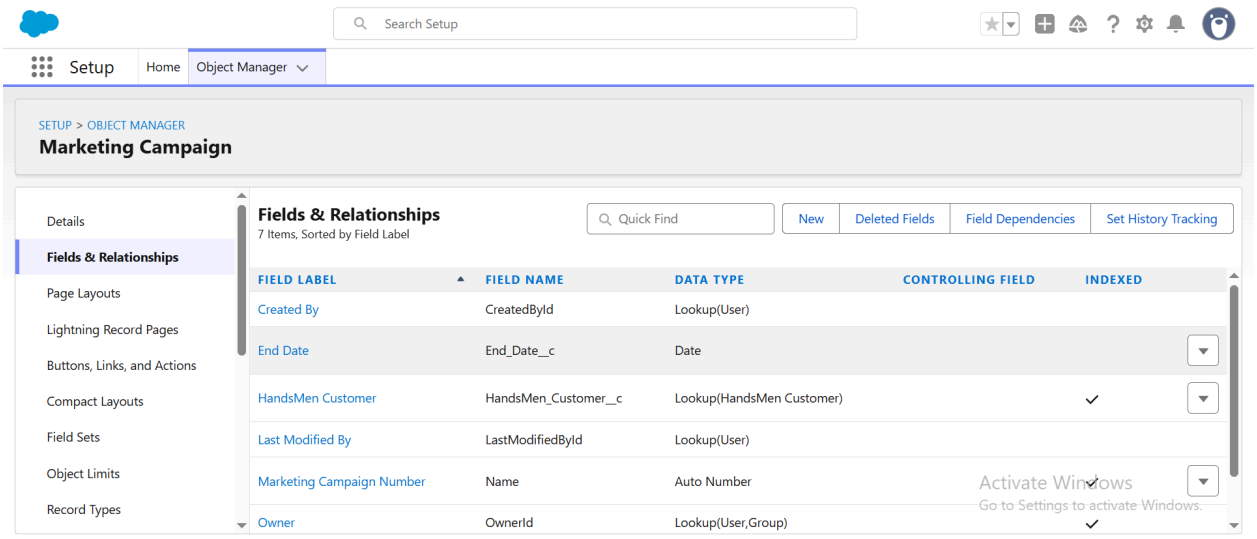
In this phase, I focused on mapping out the system architecture before implementation. I defined the data structure, business logic, and communication strategies needed to support Handsmen Threads' operations.

1. Define Objects, Fields, and Relationships (ERD)

I designed a data model that extends standard Salesforce capabilities. I analyzed the business process to determine necessary Custom Objects.

- **Custom Objects:** HandsMen_Customer__c (to store customer data), HandsMen_Order__c (order tracking), HandsMen_Product__c (product listing), Marketing_Campaign__c, and Inventory__c.
- **Relationships:** Established Lookup Relationships and Master-Detail relationships between different custom objects to ensure data cascading.





2. Establish Business Logic Strategy

I planned the automation logic required to enforce business rules:

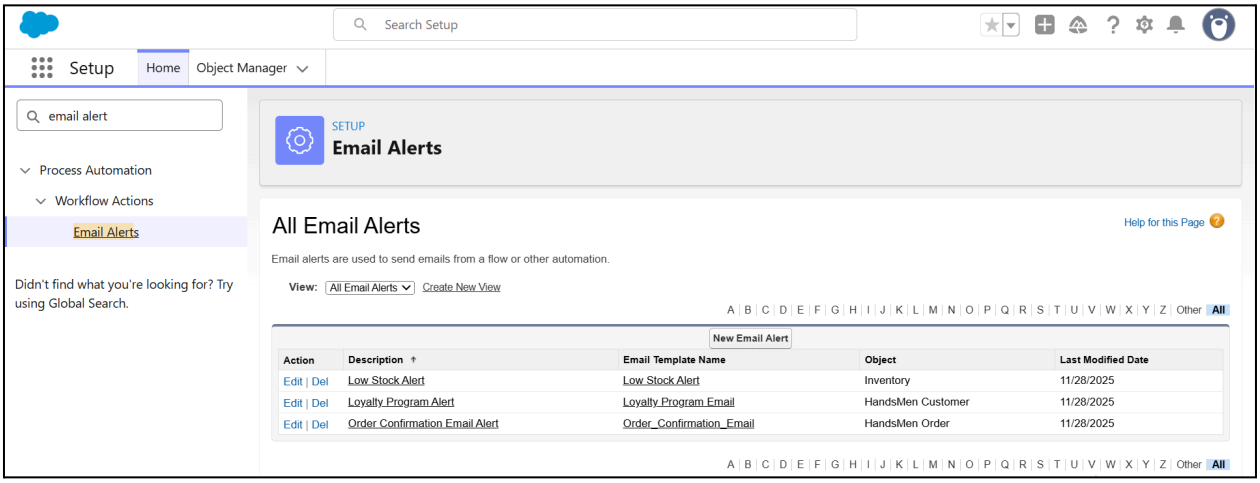
- **Validation Rules:** defined to prevent errors, such as entering negative values for measurements.
- **Flows & Triggers:** Designed workflows to auto-update inventory when an order is placed.
- **Batch Jobs:** Planned a nightly batch job to scan for orders pending delivery for more than 30 days.

3. Design Email Templates

I drafted communication templates to keep customers informed.

- *Order Confirmation Email:* Order Confirmation.
- *Low Stock Alert:* “Low Stock” Notification.

- *Loyalty Program Email:* Either Bronze, Silver, or Gold member based on purchases.

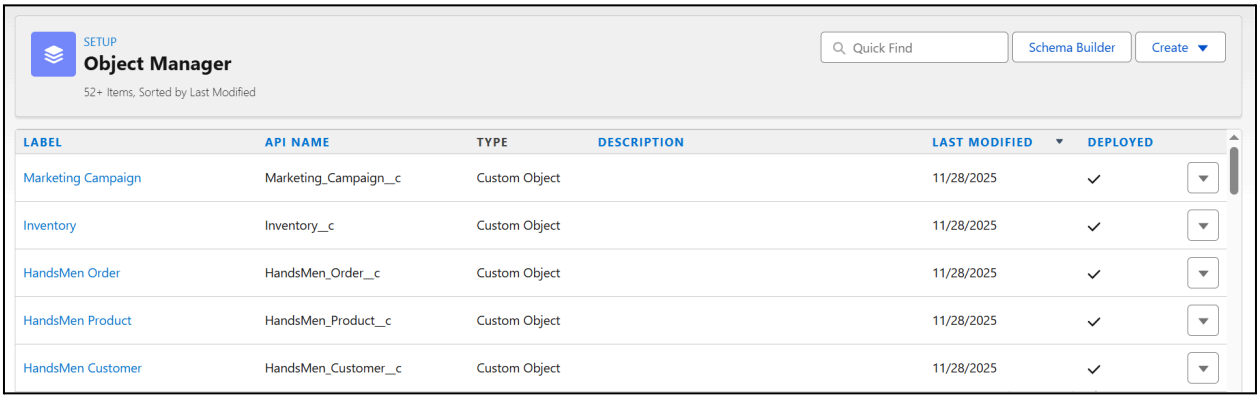


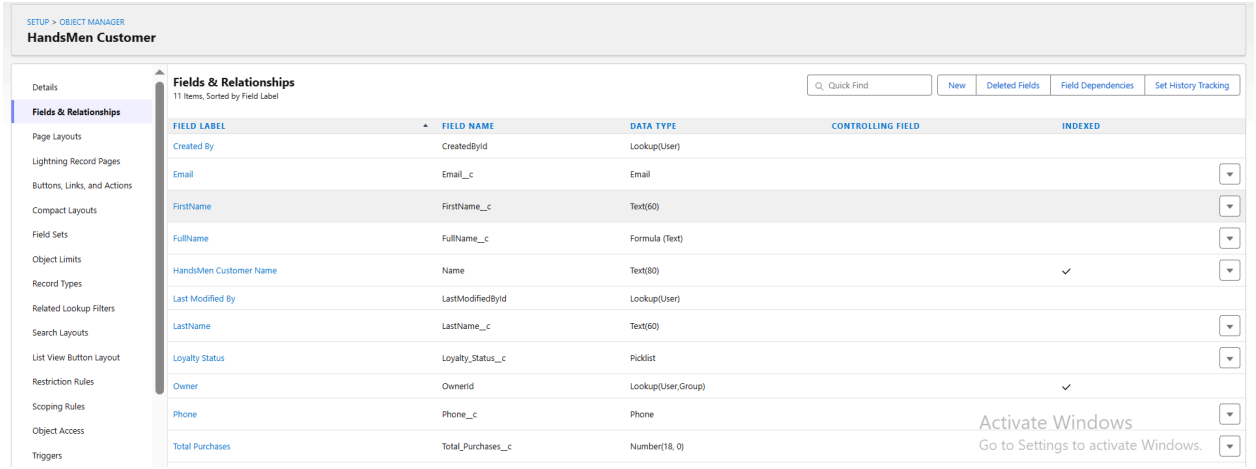
Phase 2:Salesforce Development - Backend & Configurations

This phase involved the actual construction and configuration of the Salesforce environment based on the architecture defined in Phase 1.

1. Object and Field Creation

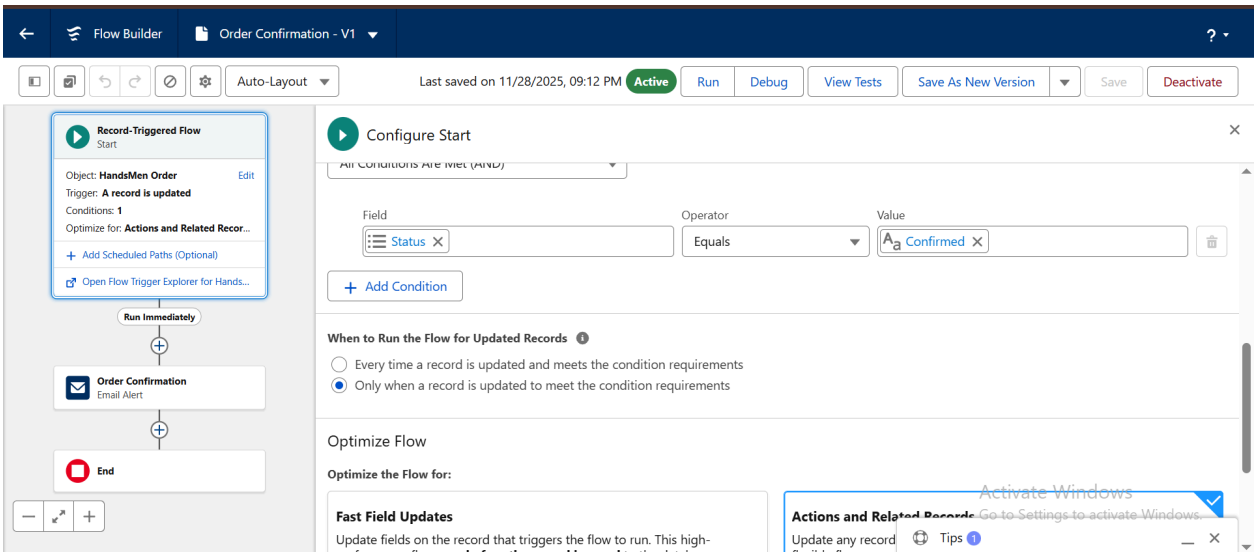
I built the custom objects and fields in the Object Manager. I configured page layouts to ensure the interface is user-friendly for the tailors and store managers.

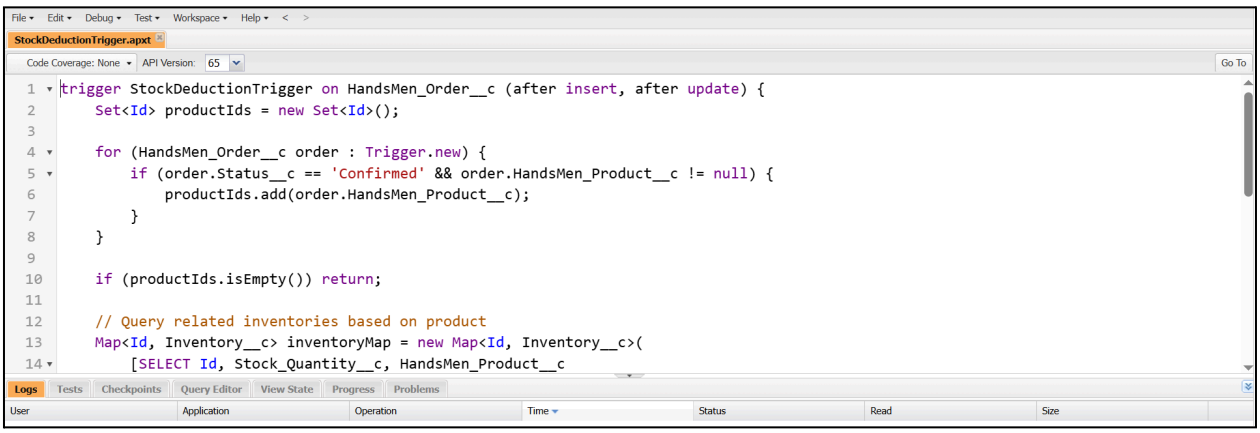




2. Implement Automation

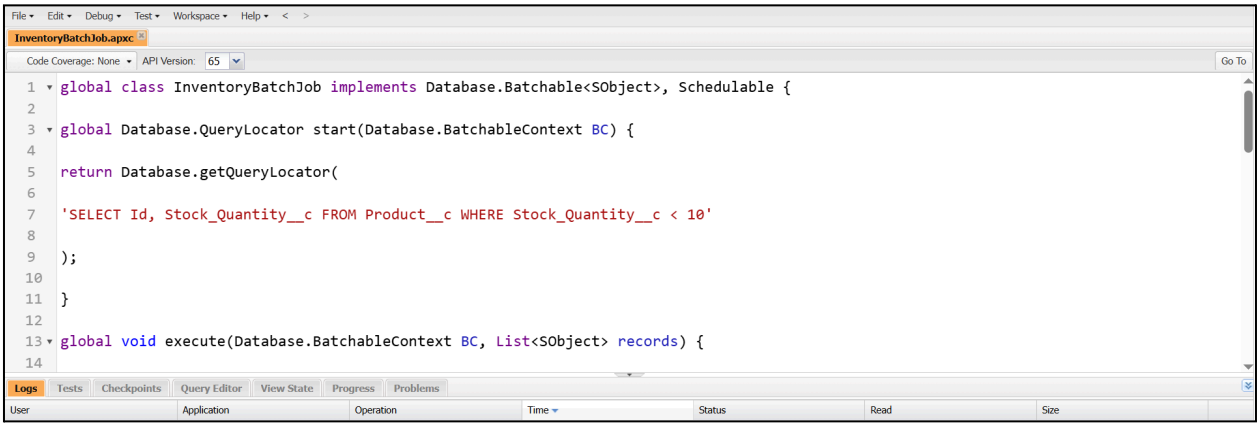
- **Flows:** I utilized Salesforce Flow to handle email scheduling. When a customer confirms an order, the flow automatically sends an email confirmation and updates the customer record.
- **Apex Triggers:** Developed an Apex Trigger on the Handsmen_Order__c object. When a new order is placed, the trigger checks the available quantity in the Handsmen_Product__c inventory. It automatically deducts the ordered quantity from the inventory levels to keep stock accurate, and blocks the transaction if the stock is insufficient.





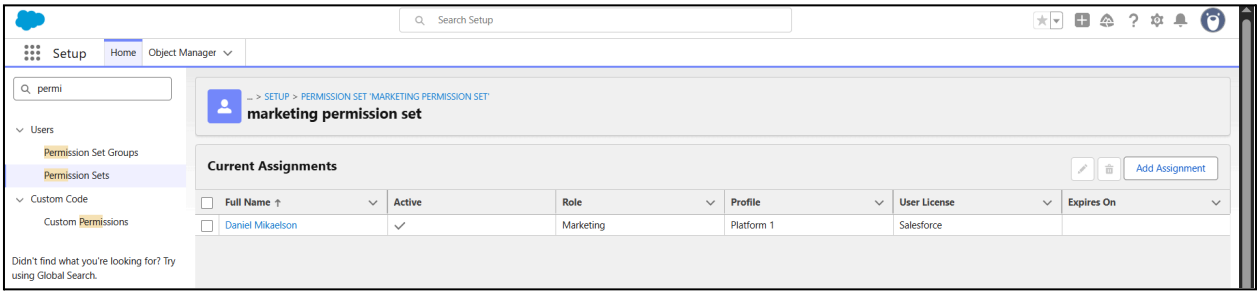
3. Develop Batch Jobs

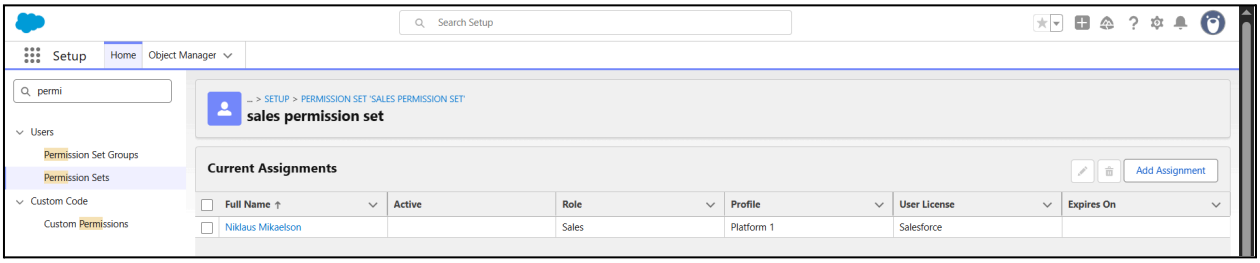
I developed a Batch Apex class named InventoryBatchJob that implements the Schedulable interface to ensure inventory stability. This job is scheduled to run periodically to scan for products with low stock levels (less than 10 units). When identified, the system automatically simulates a restocking process by increasing the Stock_Quantity__c by 50 units, ensuring that popular fabrics are always available for new orders.



4. Data Security and Sharing Rules

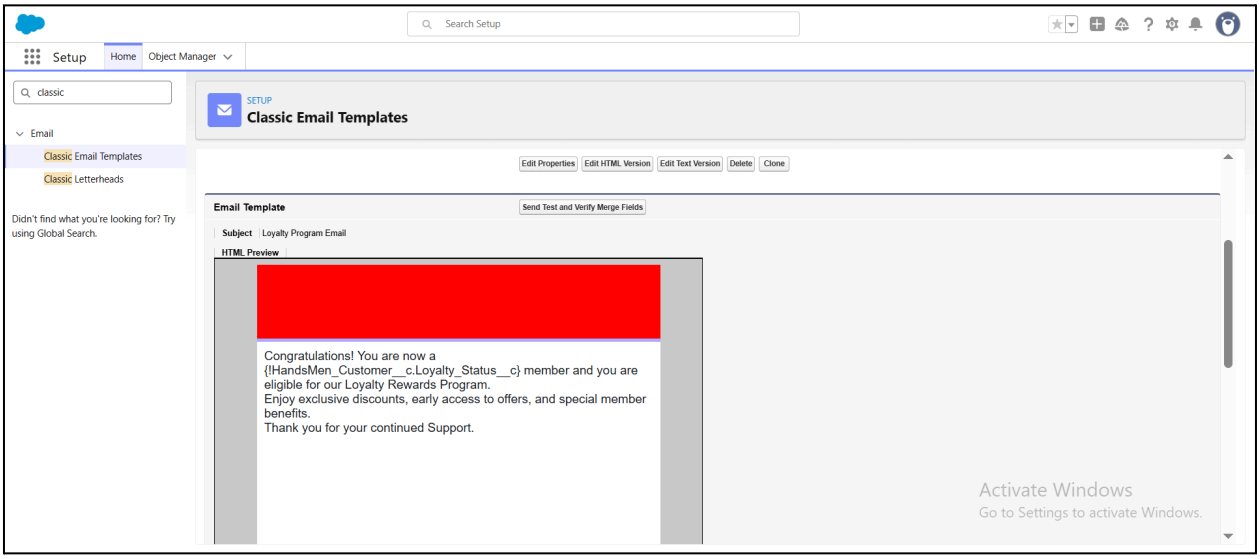
- Sharing Rules: Created rules to assign “Sales Permission Set” with the “Sales” role, while “Marketing Permission Set” are only assigned with the “Marketing” role.

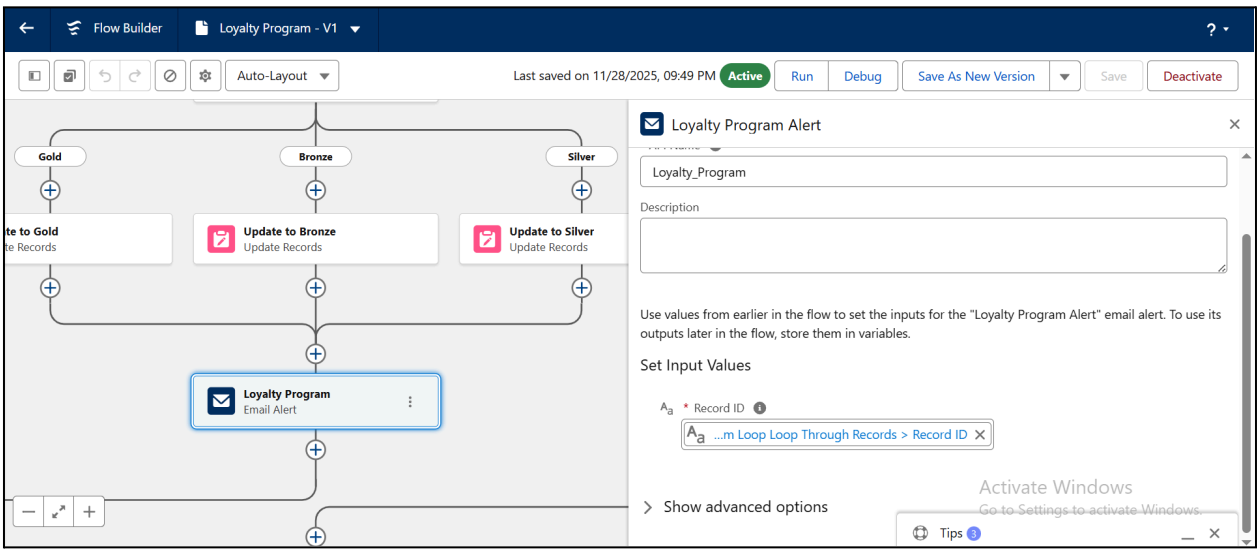




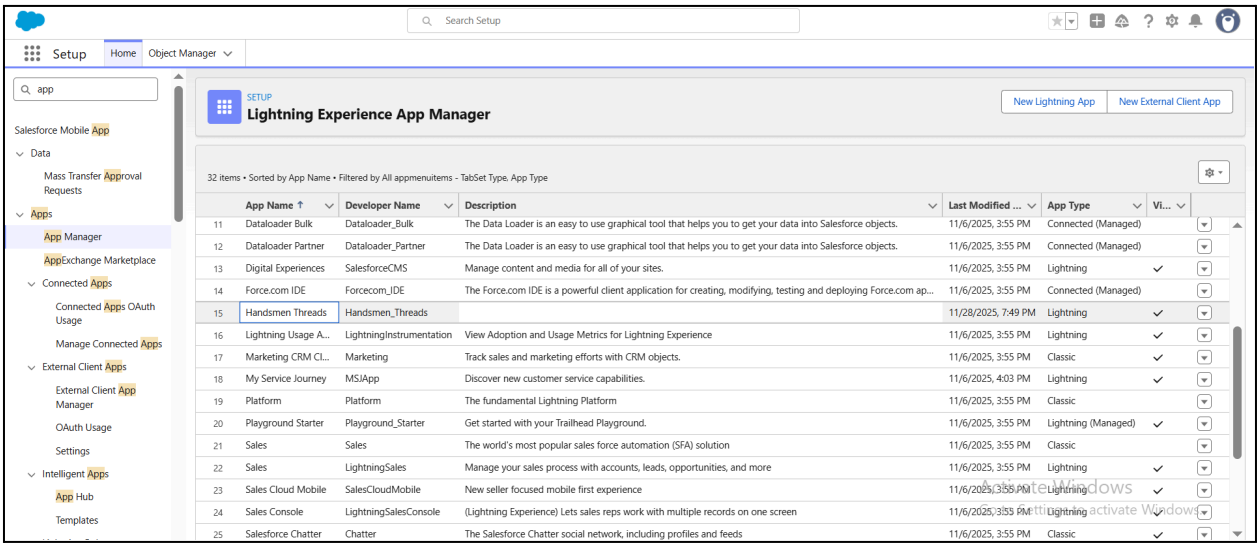
5. Configure Email Templates & Notifications

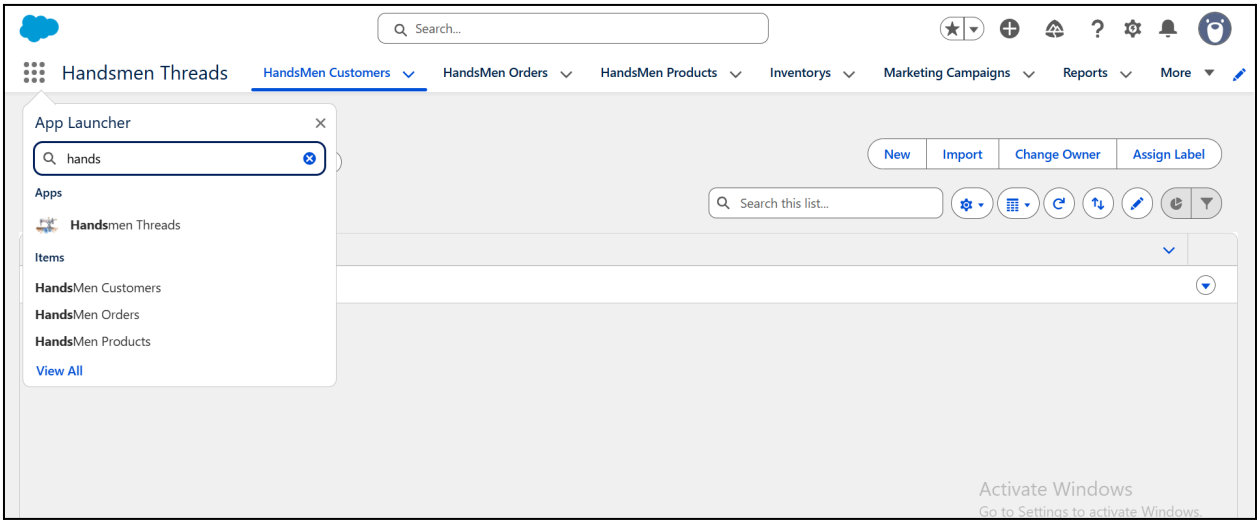
I implemented the HTML email templates and set up Email Alerts within our Flows to send them automatically upon specific triggers.





Phase 3:UI/UX Development & Customization





Phase 4:Data Migration, Testing & Security

In this phase, I validated the system architecture to ensure that the custom objects (HandsMen Customer__c, HandsMen Product__c, etc.) function correctly and that data integrity is maintained through formulas and automation.

1. Data Quality & Integrity Implementation

To automate data formatting and provide instant visual feedback on inventory levels, we implemented the following formula fields:

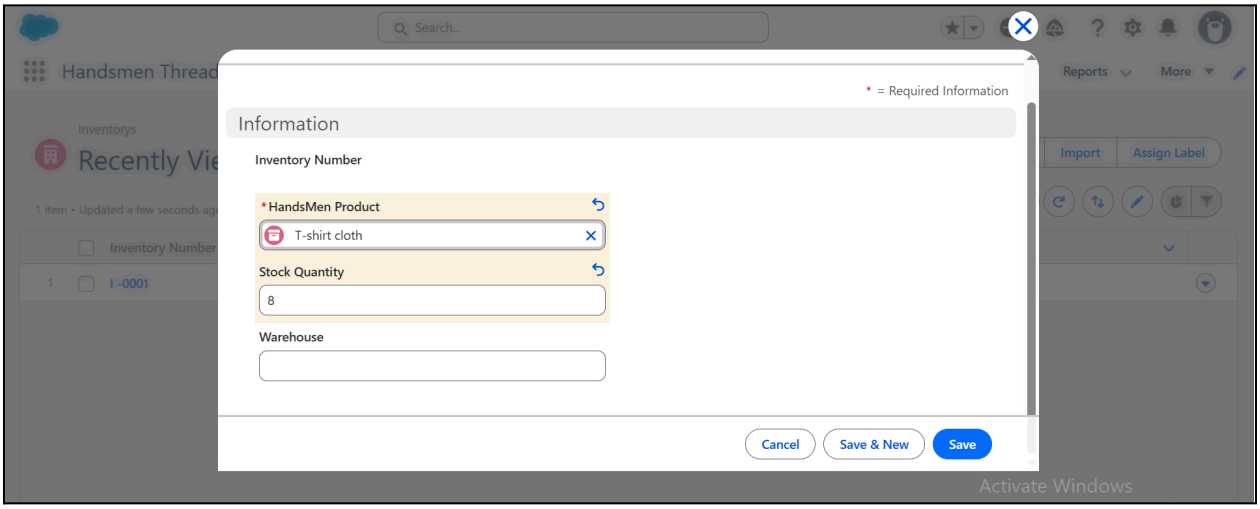
- **Inventory Status Indicator:** I created a formula field Stock_Status__c on the Inventory__c object. This text formula automatically flags items as “Low Stock” if the quantity drops to 10 or below, allowing the warehouse manager to spot shortages immediately without running a report.
    - *Formula Logic:* IF(Stock_Quantity__c > 10, “Available”, “Low Stock”)
- **Customer Name Concatenation:** On the HandsMen Customer__c object, I created the Full_Name__c formula to combine the First and Last names into a single readable field for easier search and reporting.
    - *Formula Logic:* FirstName & " " & LastName

3. Functional Test Cases & Results

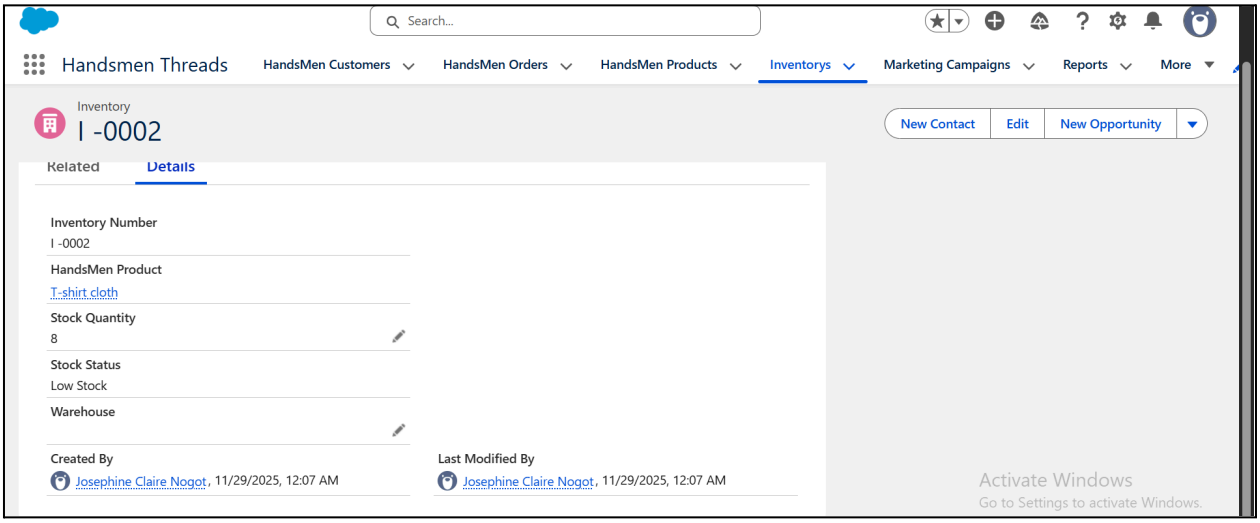
I executed functional tests to verify the formulas and object behaviors defined in the schema.

Test Case 1: Inventory Status Formula Logic

- **Objective:** Verify that the Stock_Status__c field on the **Inventory__c** object correctly updates based on the Stock_Quantity__c.
- **Step 1 (Input):** User creates a new Inventory record and enters 8 in the Stock_Quantity__c field.



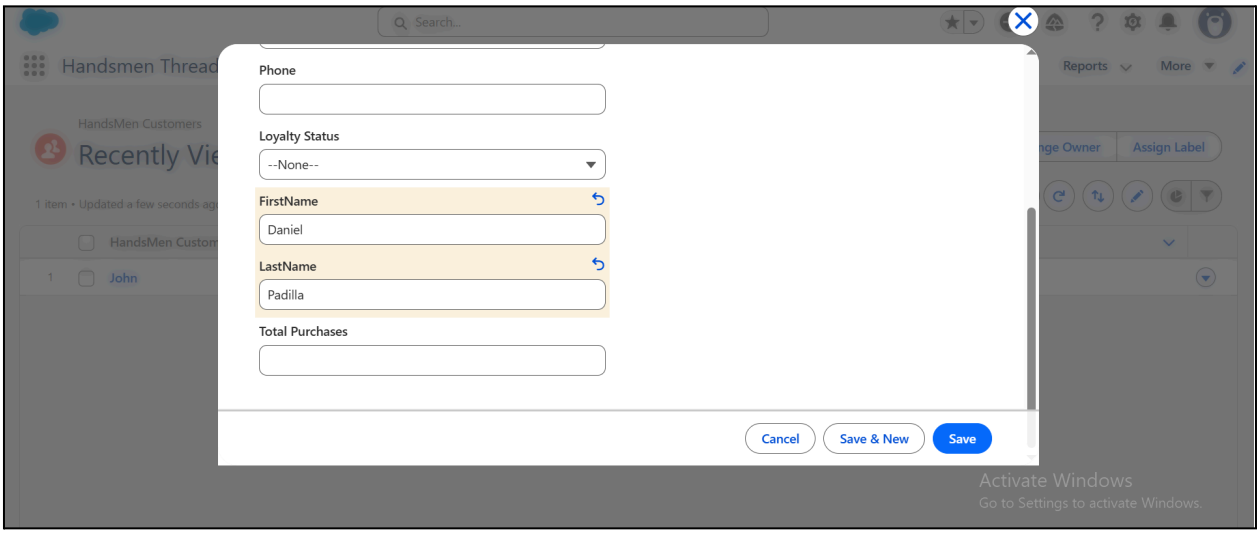
- **Step 2 (Output):** Upon saving, the Stock_Status__c field automatically displays **"Low Stock"**.



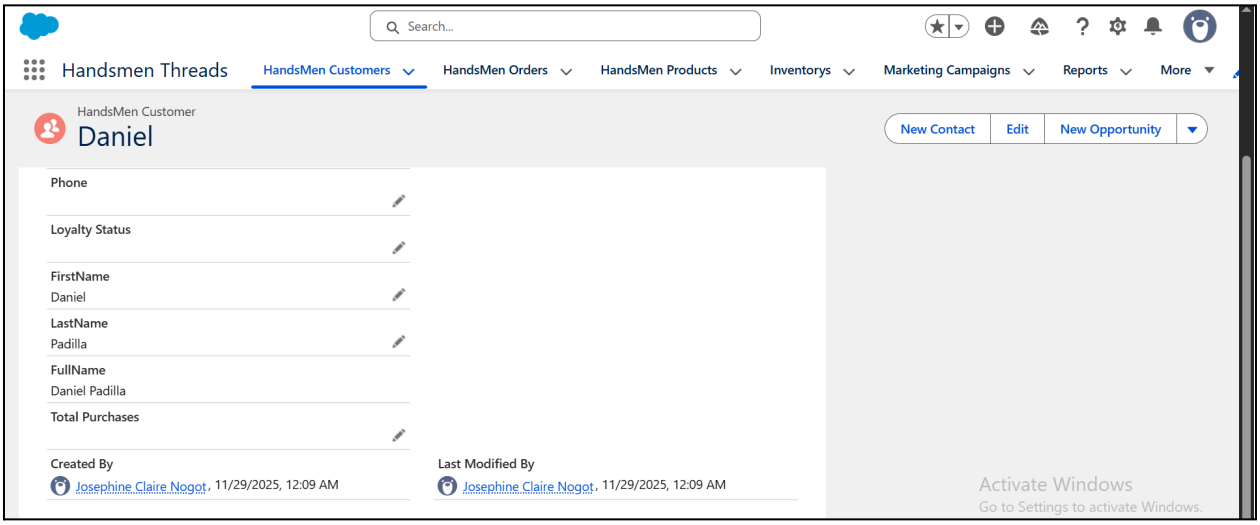
Test Case 2: Customer Name Concatenation

- **Objective:** Verify that the Full_Name__c formula on **HandsMen Customer__c** works as expected.
- **Step 1 (Input):** User creates a new customer record with First Name: **"Daniel"** and Last

Name: **“Padilla”**.

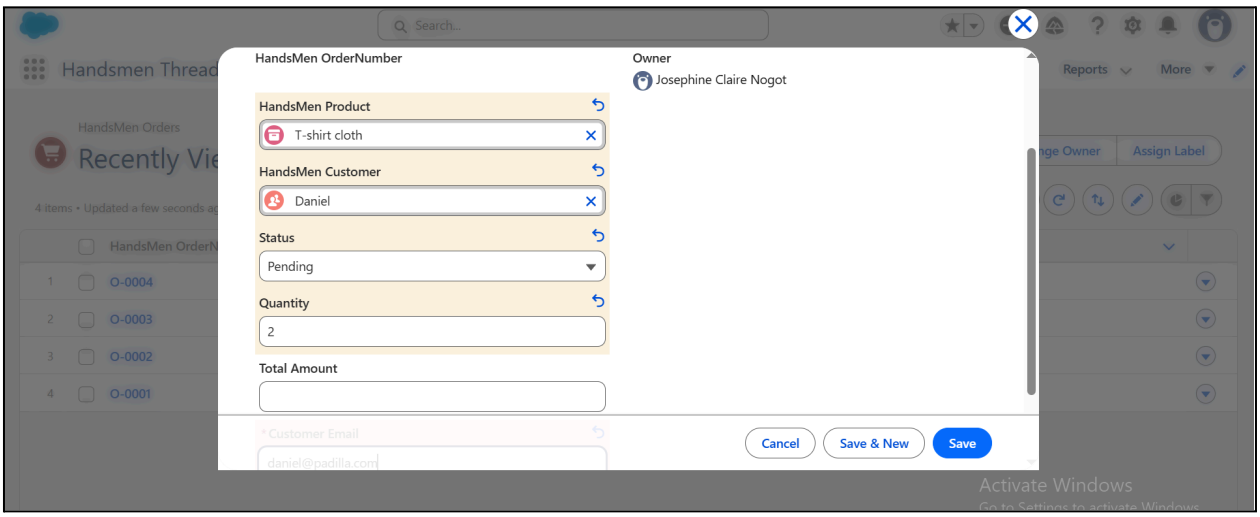


- **Step 2 (Output):** The system saves the record, and the Full_Name__c field displays **“Daniel Padilla”**.

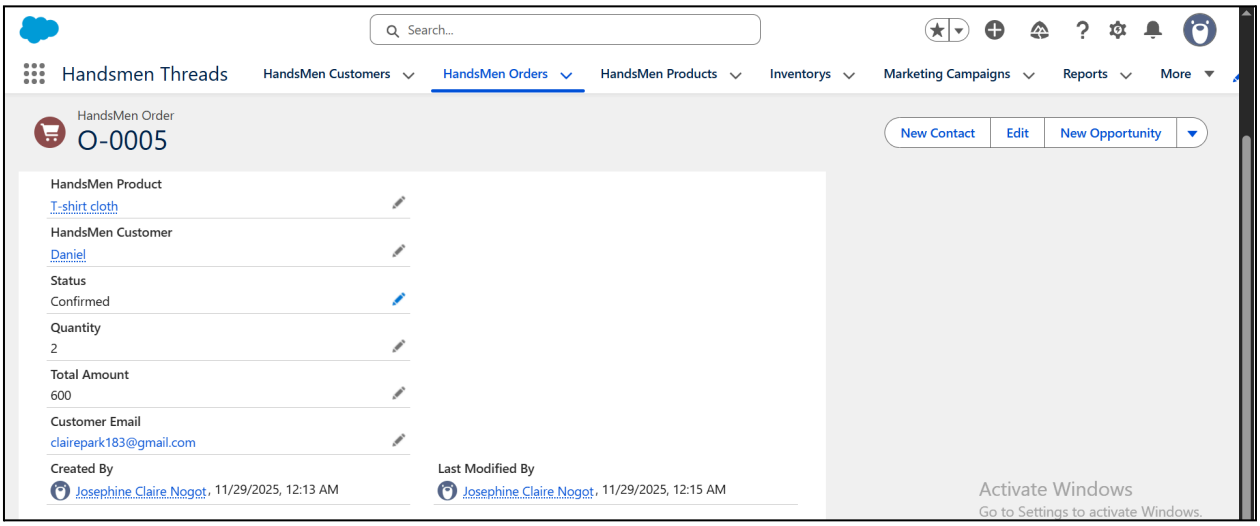


Test Case 3: Order Creation & Status Flow

- **Objective:** Verify that a **HandsMen Order__c** can be created and tracked via the Status picklist.
- **Step 1 (Input):** User creates a new order, linking it to a customer, setting Quantity__c to 2, and Status to **“Pending”**.



- **Step 2 (Output):** The record is saved successfully. Later, the user edits the Status to **"Confirmed".**



Phase 5:Deployment, Documentation & Maintenance

When a user reports an error (e.g., "I can't save this Order"), I follow this standardized troubleshooting workflow:

1. **Reproduce the Issue:** Log in as the user (using "Login As") to see if the error is replicable.
2. **Check Validation Rules:** 90% of save errors are due to Validation Rules. We check if the data input violates rules like the "Low Stock" trigger.
3. **Review Debug Logs:** If the error is technical (like an Apex Trigger failure), we turn on Debug Logs for that user to trace the specific line of code causing the exception.
4. **Flow Fault Emails:** We configured the system to send an email to the Admin whenever a

Flow fails, providing immediate context on what went wrong.

Conclusion

The Handsmen Threads Capstone Project successfully achieved its primary objective: transforming a traditional, manual bespoke tailoring operation into a modernized, digital-first business.

By implementing Salesforce, we addressed the critical pain points of data redundancy and inventory mismanagement. The custom architecture, specifically the HandsMen Customer__c and HandsMen Order__c objects, now provides a "Single Source of Truth" for client measurements and transaction history. The automation we built does more than just save time; the Stock Validation Trigger actively prevents revenue loss by stopping orders for out-of-stock fabrics, while the InventoryBatchJob ensures the warehouse is proactive rather than reactive.

This project demonstrates that Salesforce is not just for large enterprises but can be effectively tailored to small businesses. The system is scalable, secure, and ready to support Handsmen Threads as they expand their customer base, providing a solid foundation for future enhancements like customer self-service portals and automated marketing journeys.