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

Use Case

HandsMen Threads is a dynamic organization in the fashion industry embarking on a Salesforce CRM implementation designed to revolutionize data management and enhance customer relations.

The project aims to build a robust data model to store essential business data, ensuring smooth information flow across departments. A core focus is on maintaining data integrity directly from the user interface, ensuring accurate, consistent, and reliable records to support effective decision-making and efficient operations.

The system also integrates automation processes to improve customer service and operational efficiency.

Key Features

• **Automated Order Confirmations** – Customers receive email confirmations after order placement, improving communication.

• **Dynamic Loyalty Program** – Loyalty status updates automatically based on purchase history.

• **Stock Alerts –** The warehouse team receives alerts when stock levels drop below threshold.

• **Scheduled Bulk Order Updates –** Nightly updates ensure inventory and financial data accuracy.

User Story

The Salesforce project aims automate core business processes for HandsMen Threads. By building a data model integrating automation, the system will efficiently operate customer service, inventory management, and financial operations. This ensures efficiency, reliability and customer services.

Project Overview

Phase 1: Architecture s Planning

- Define objects, fields, relationships, formula fields.
- Establish validation rules, flows, Apex triggers, batch jobs.
- Design email templates for notifications and customer communication.

Phase 2: Development

- Object and field creation.

Created By: Antonio Miguel Orias

- Set up data security and sharing rules.
- Implement automation such as (flows, process builders, Apex triggers).
- Develop batch jobs for scheduled processing.
- Configure email templates and notifications.

Phase 3: Testing s QA

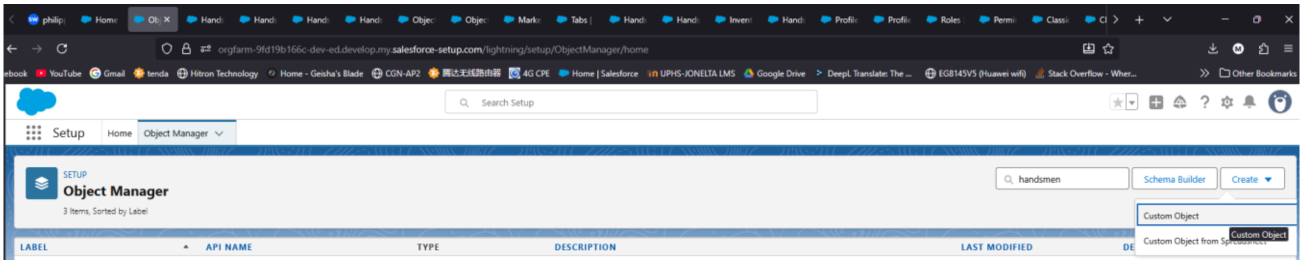
- Unit testing of objects and automation.
- End-to-end testing with sample data.
- Performance testing and security checks.

Phase 4: Deployment s Training

- Deploy to production.
- Train users on new functionality.
- live support and monitoring.

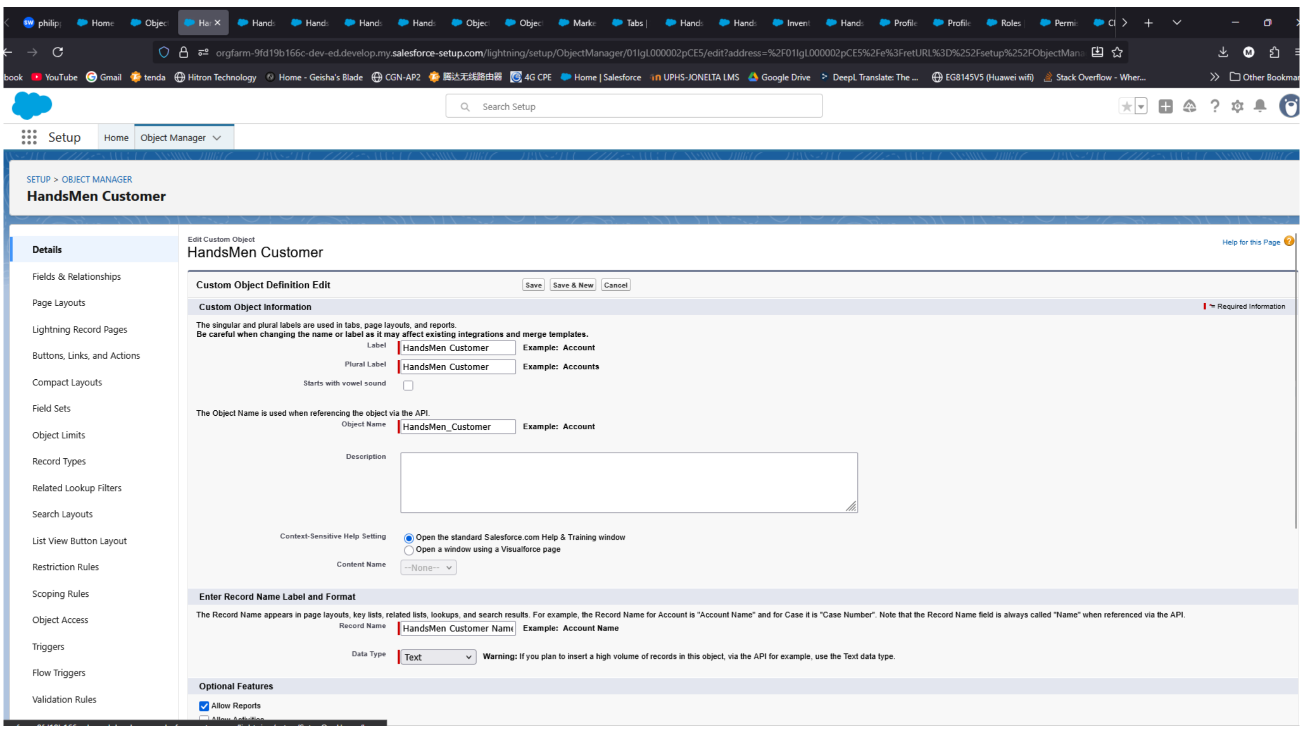
Data Management

Objects:

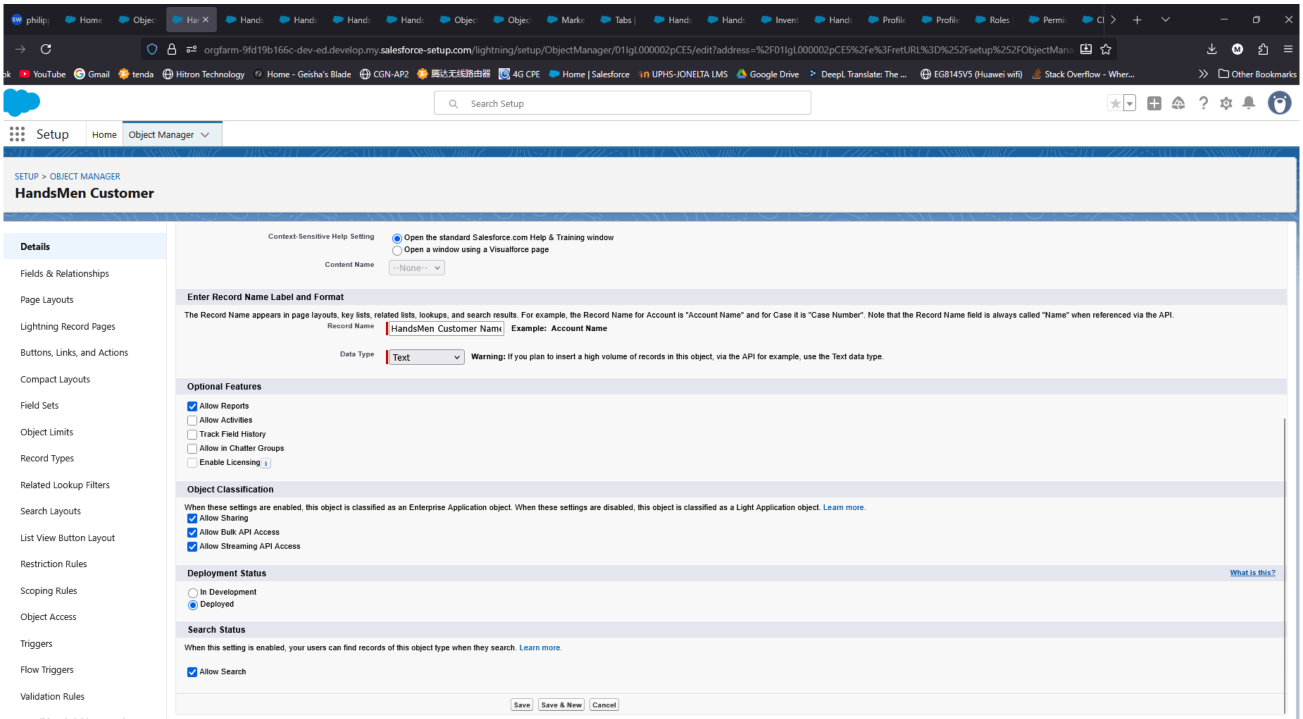


In the **Object Manager** tab, click the **Create** button and select **Custom Object**.

1. HandsMen Customer



Created By: Antonio Miguel Orias



In the Custom Object Information section:

- **Label:** HandsMen Customer
- **Plural Label:** HandsMen Customer
- **Object Name:** HandsMen_Customer

In the Record Name Label and Format section:

- **Record Name:** HandsMen Customer Name
- **Data Type:** Text

In the Optional Features section:

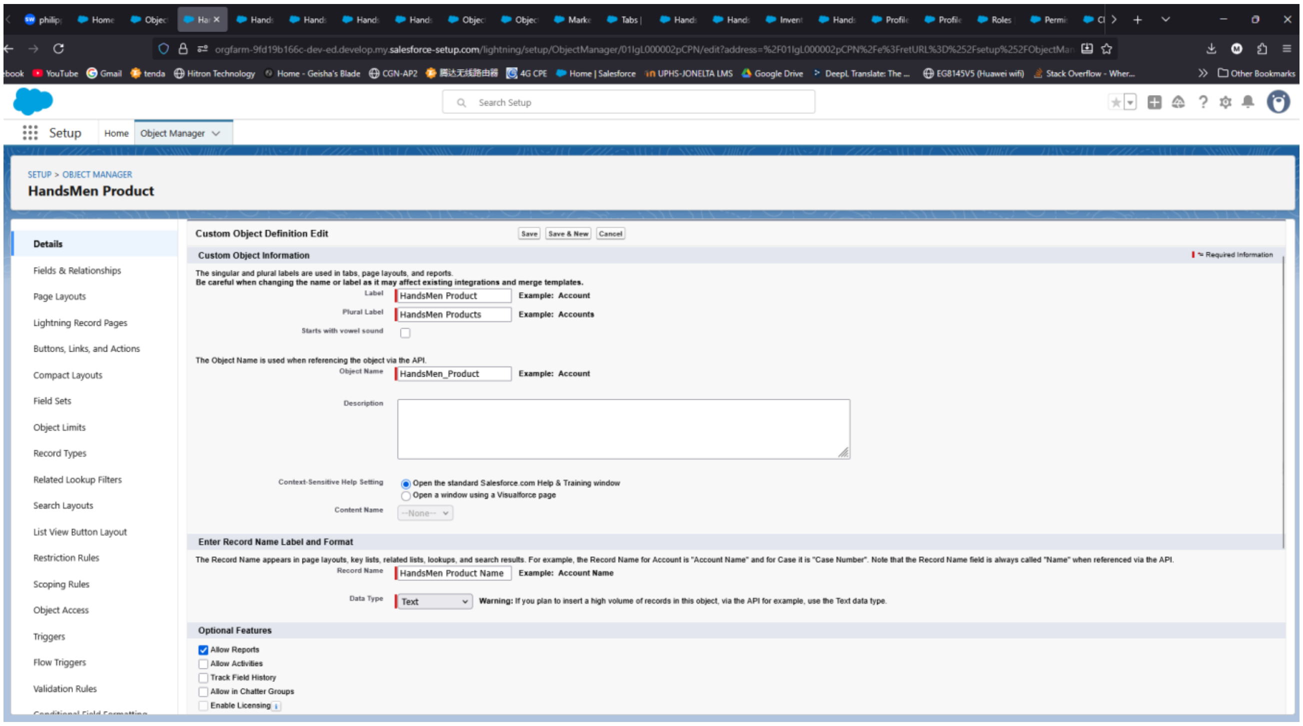
- Select **Allow Reports**

In the Search Status section:

- Select **Allow Search**

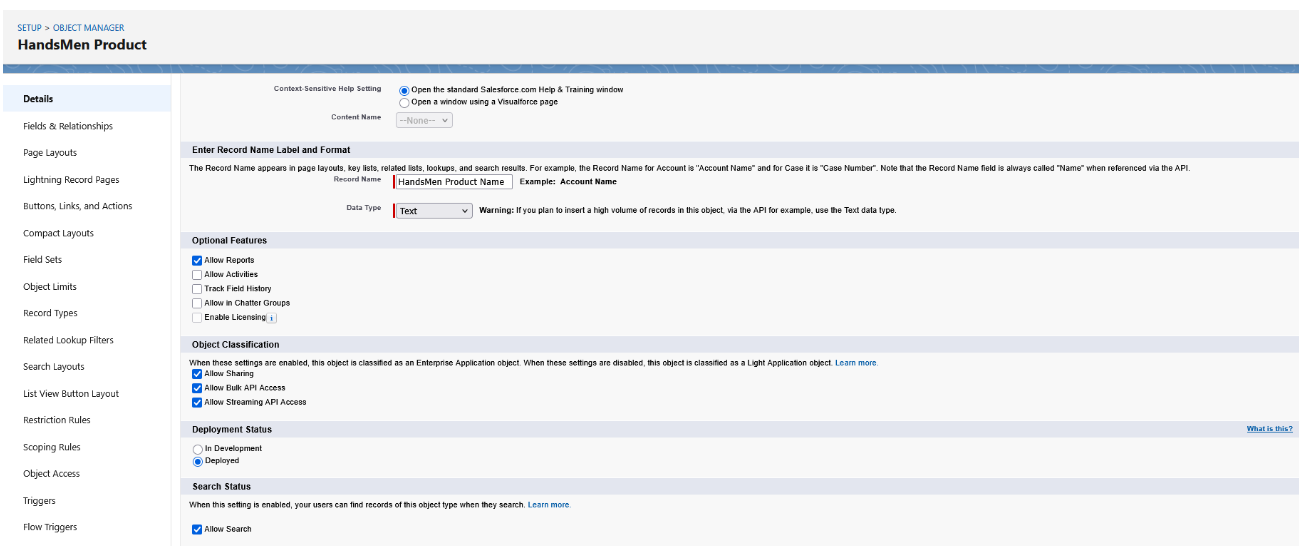
Click the **Save** button to create the custom object.

2. HandsMen Product



3.

In the Custom Object Information section:



- **Label:** HandsMen Product
- **Plural Label:** HandsMen Products
- **Object Name:** HandsMen_Product

In the Record Name Label and Format section:

- **Record Name:** HandsMen Product Name
- **Data Type:** Text

In the Optional Features section:

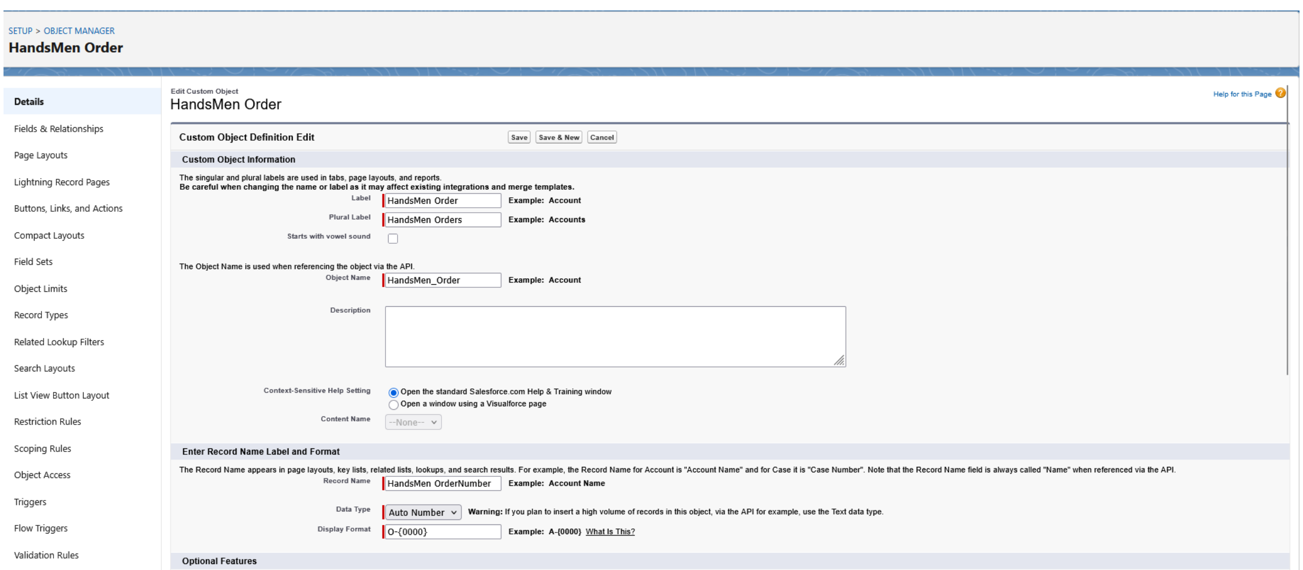
- Select **Allow Reports**

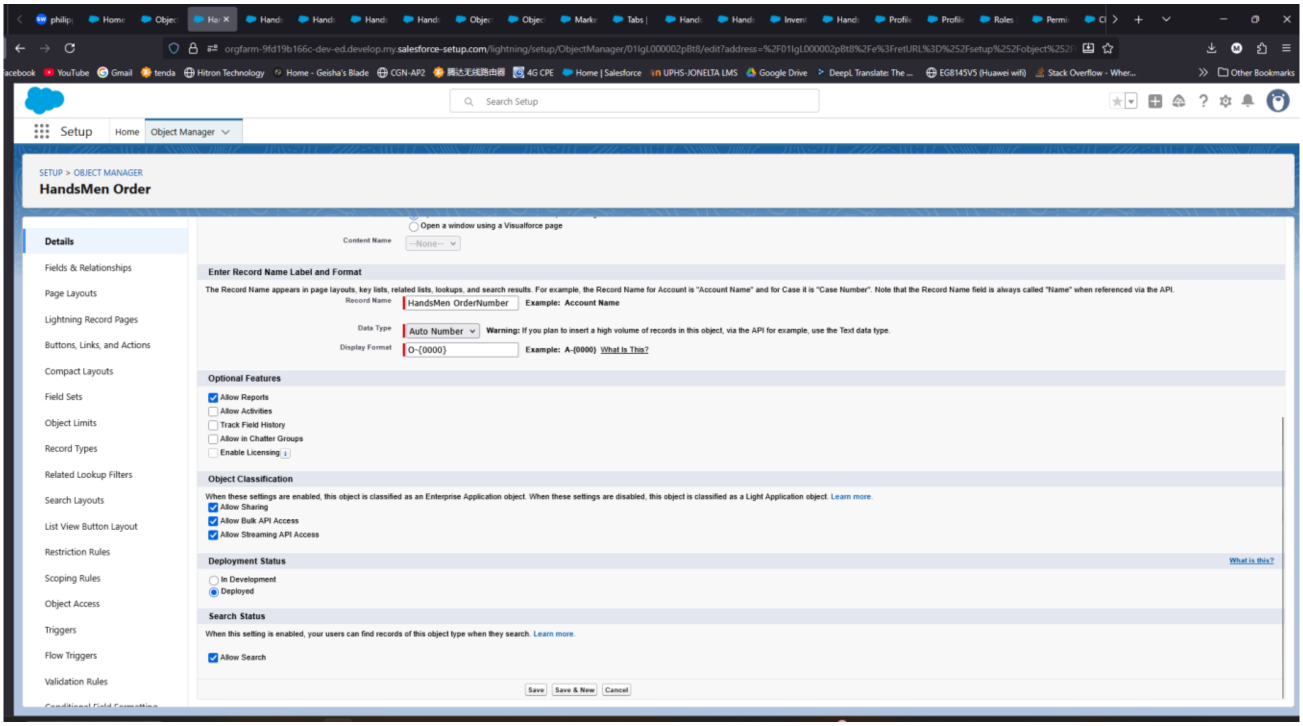
In the Search Status section:

- Select **Allow Search**

Click the **Save** button to create the custom object.

4. HandsMen Order





In the Custom Object Information section:

- **Label:** HandsMen Order
- **Plural Label:** HandsMen Orders
- **Object Name:** HandsMen_Order

In the Record Name Label and Format section:

- **Record Name:** HandsMen OrderNumber
- **Data Type:** Auto Number
  - **Display Format:** O-{0000}
  - **Starting Number**: 001

In the Optional Features section:

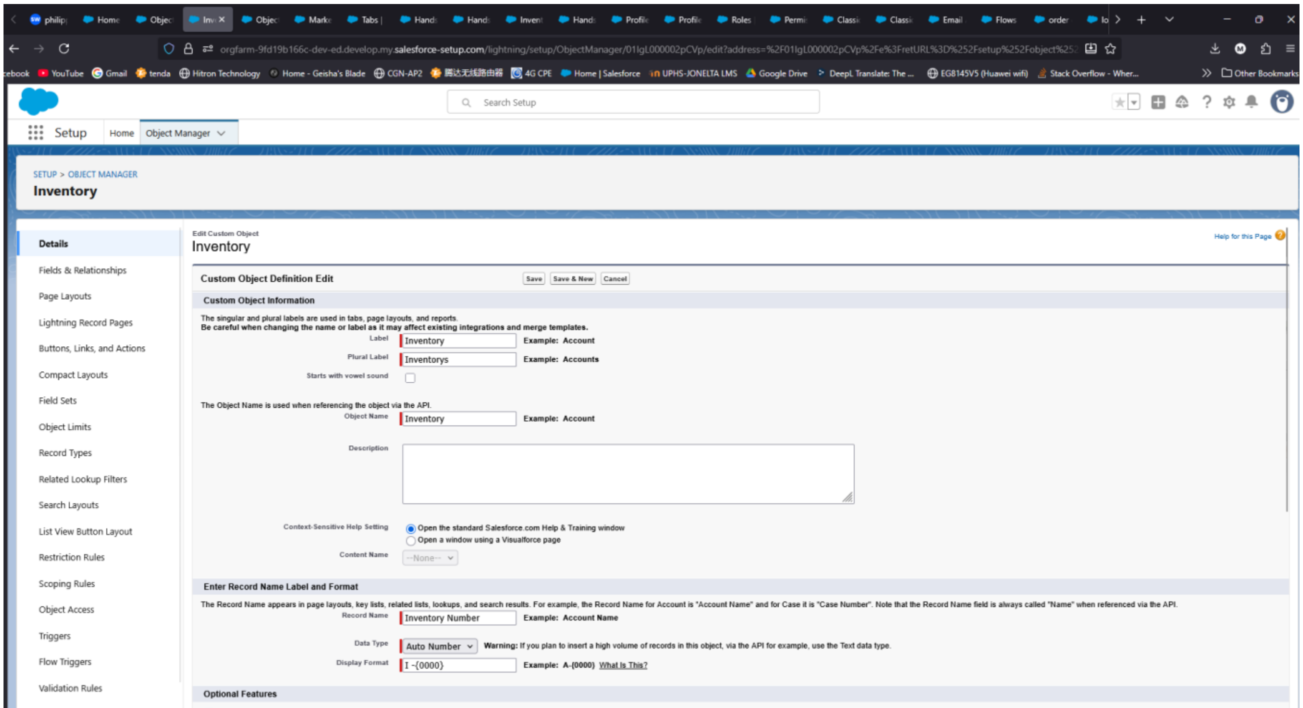
- Select **Allow Reports**

In the Search Status section:

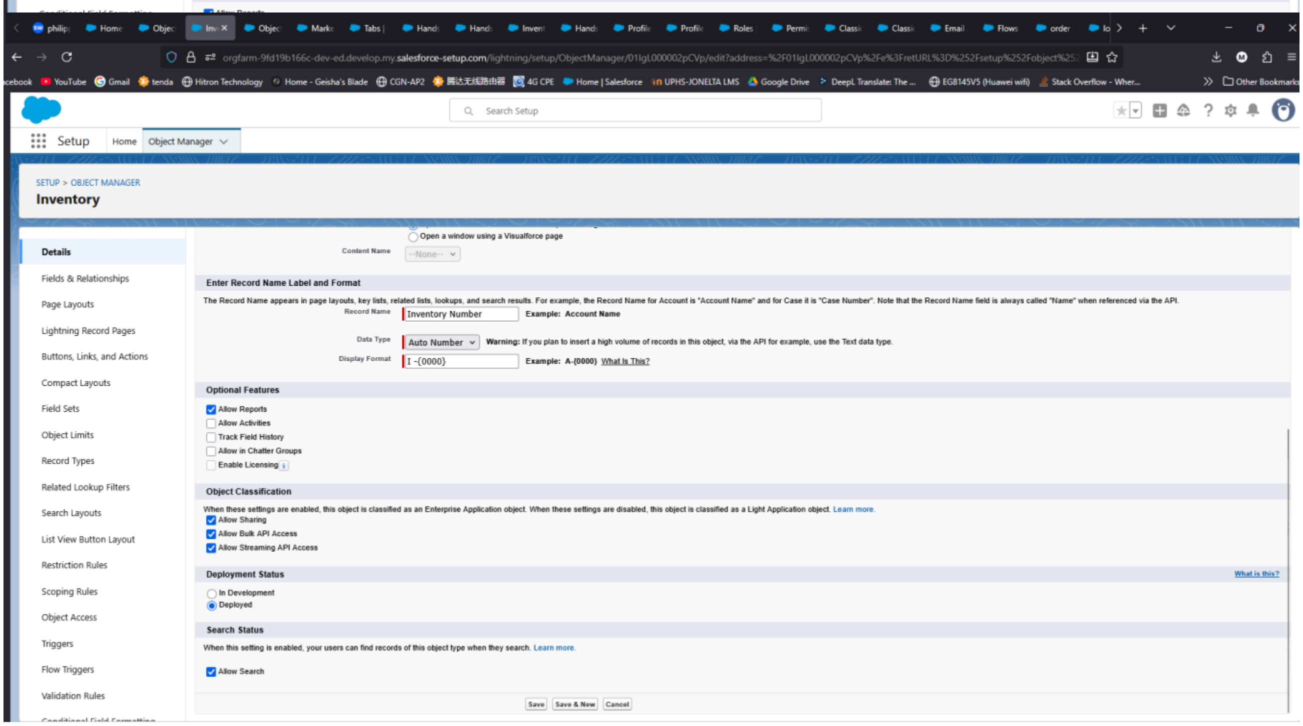
- Select **Allow Search**

Click the **Save** button to create the custom object.

5. Inventory



6.



In the Custom Object Information section:

- **Label:** Inventory
- **Plural Label:** Inventorys
- **Object Name:** Inventory

In the Record Name Label and Format section:

- **Record Name:** Inventory Number
- **Data Type:** Auto Number
  - **Display Format:** I-{0000}
  - **Starting Number**: 001

In the Optional Features section:

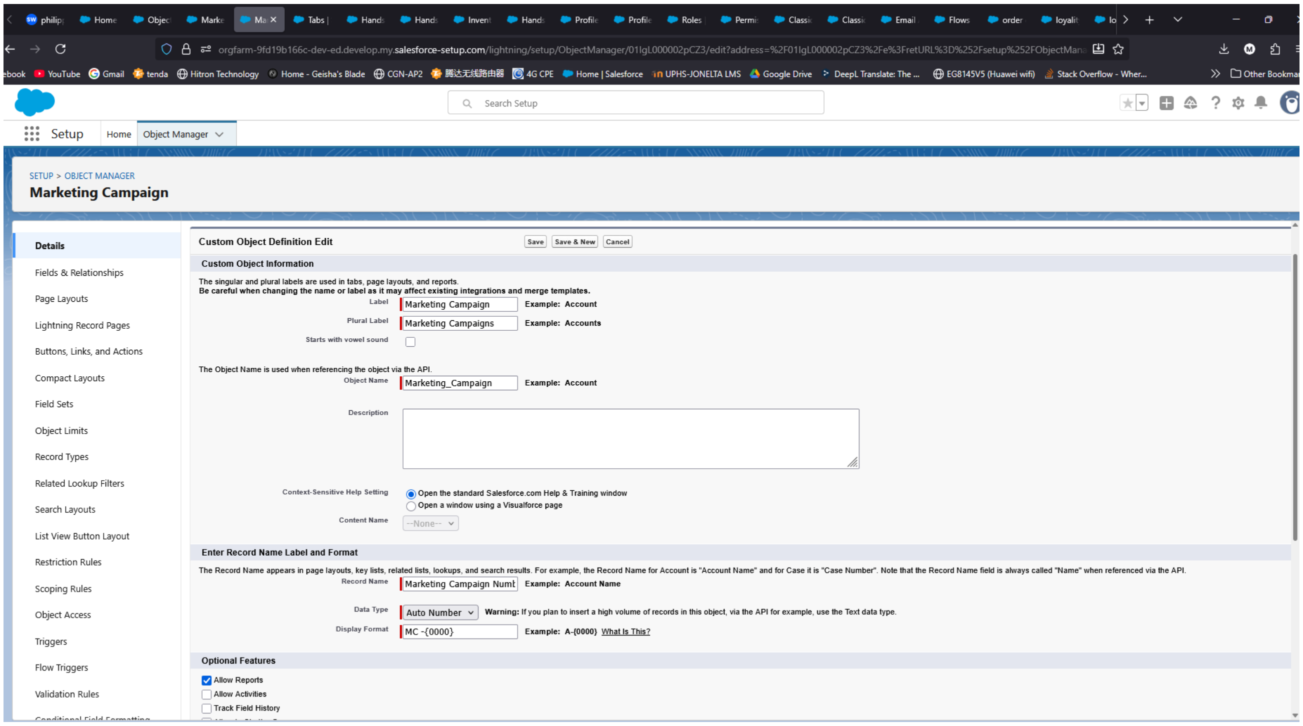
- Select **Allow Reports**

In the Search Status section:

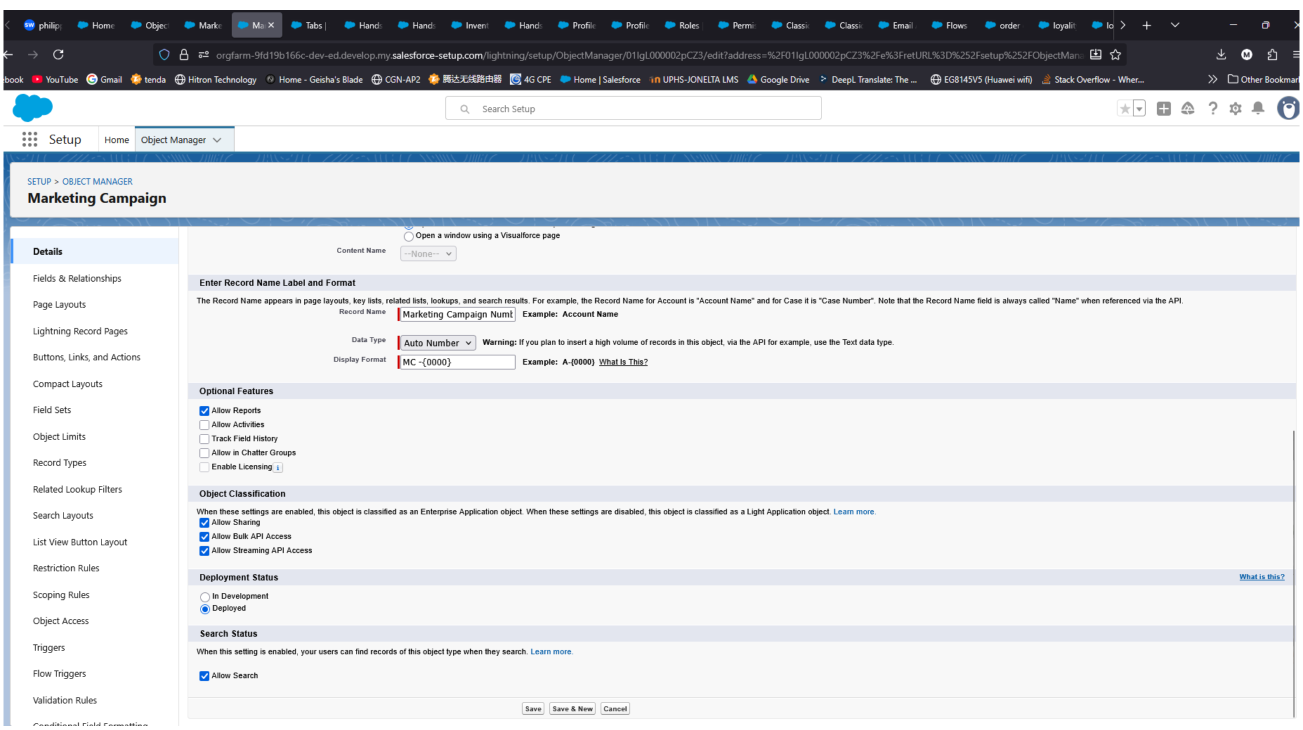
- Select **Allow Search**

Click the **Save** button to create the custom object.

7. Marketing Campaign



8.



In the Custom Object Information section:

- **Label:** Marketing Campaign
- **Plural Label:** Marketing Campaigns
- **Object Name:** Marketing_Campaign

In the Record Name Label and Format section:

- **Record Name:** Marketing Campaign Number
- **Data Type:** Auto Number
    - **Display Format:** MC-{0000}
    - **Starting Number**: 001

In the Optional Features section:

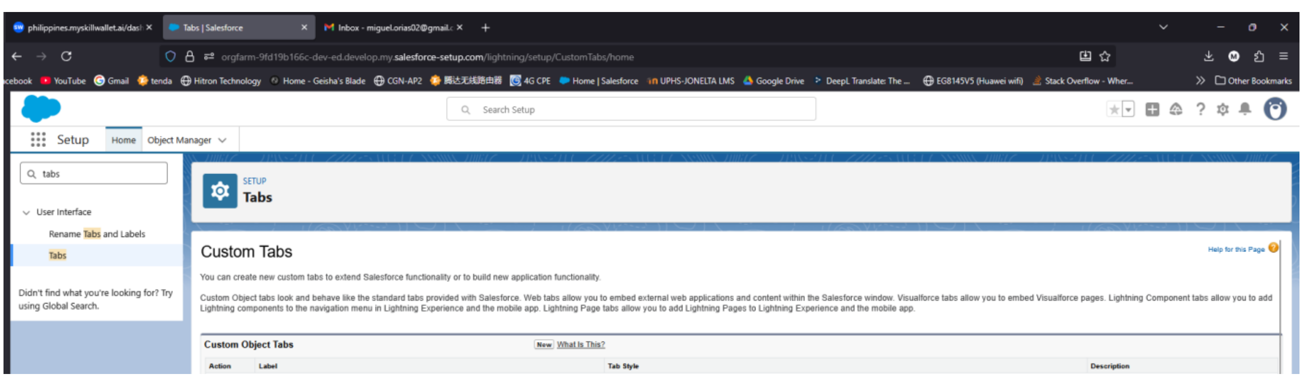
- Select **Allow Reports**

In the Search Status section:

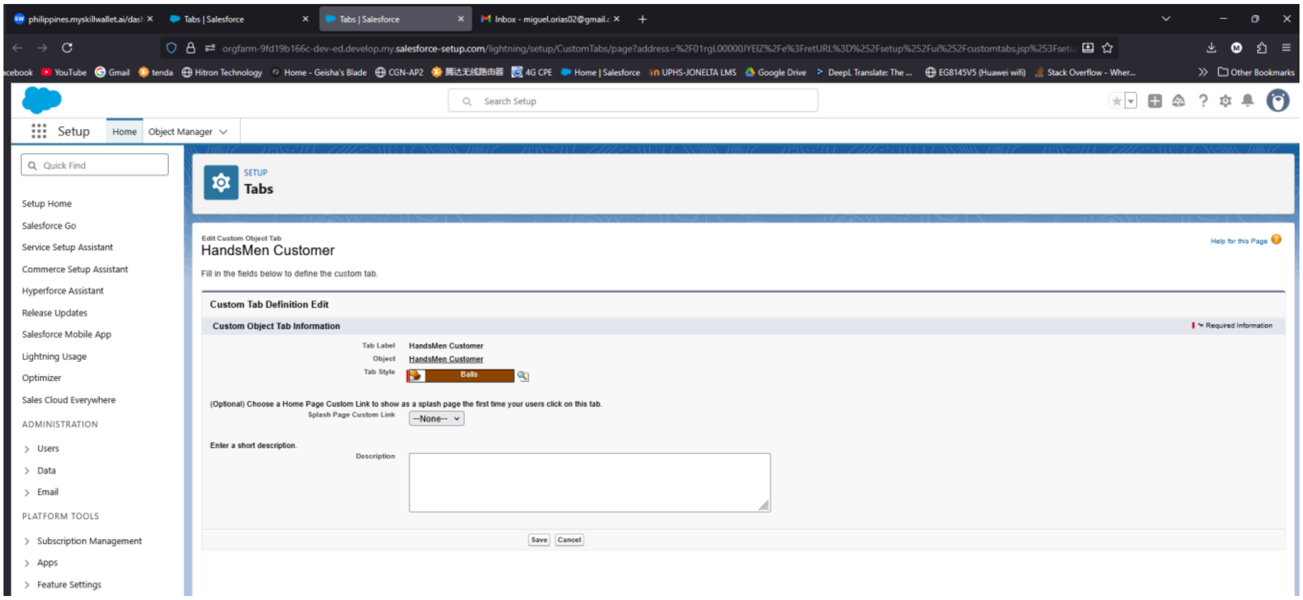
- Select **Allow Search**

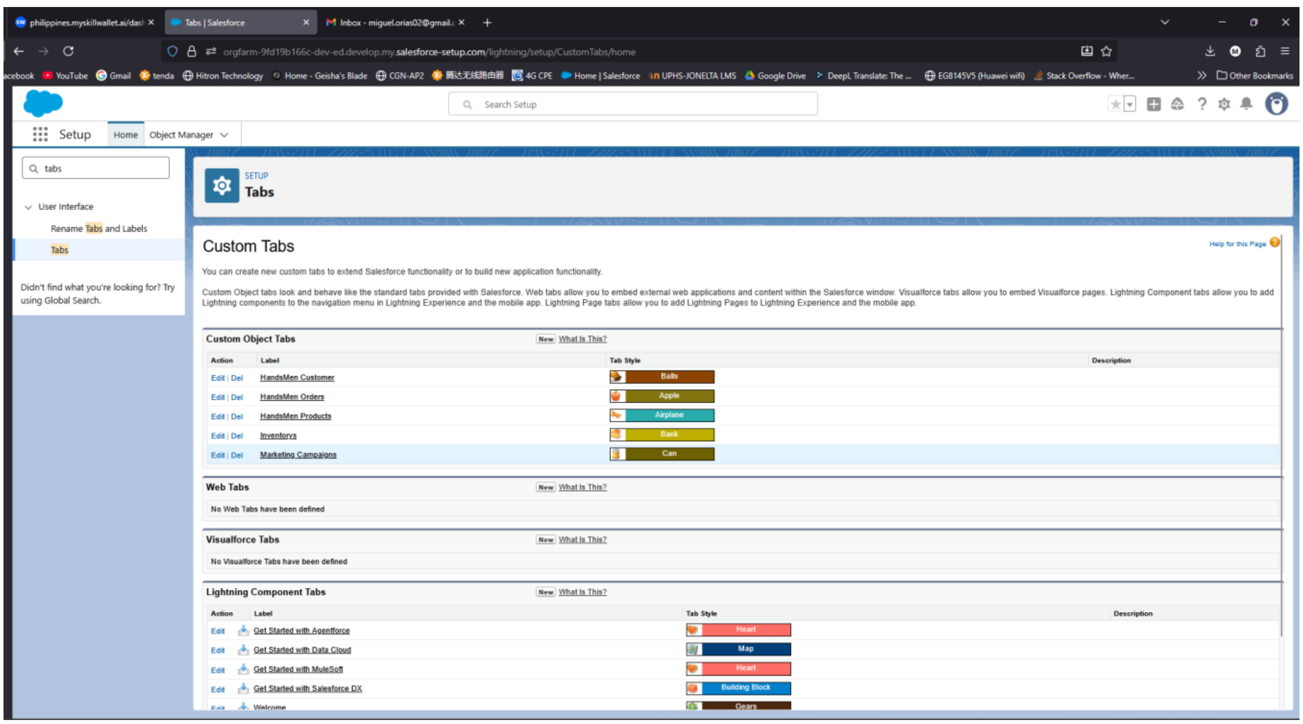
Click the **Save** button to create the custom object.

Tabs:



In **Quick Find**, search for **Tabs**, then in the Tabs setup, click the **New** button under Custom Object Tabs.





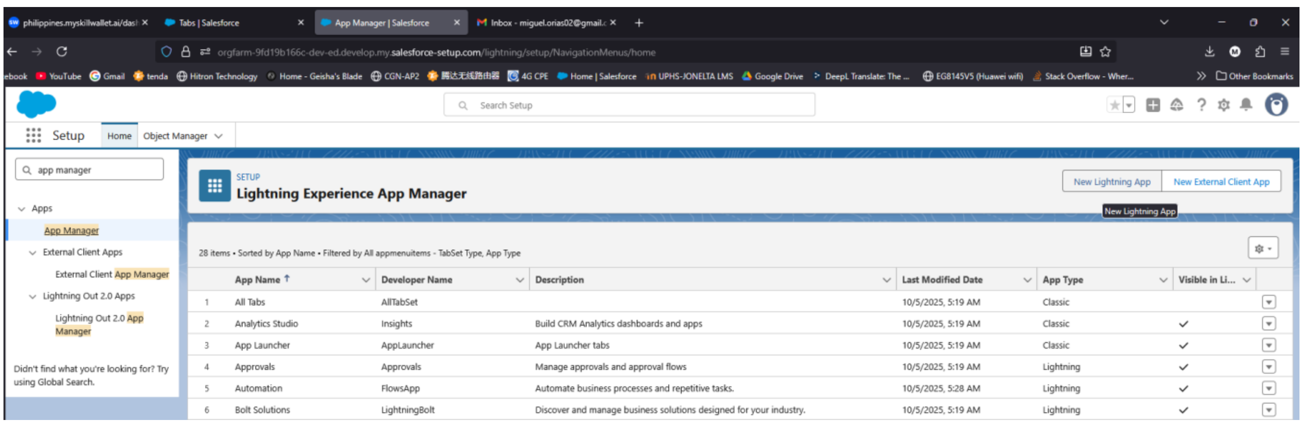
In Step 1, select the object **HandsMen Customer**. For Tab Style, choose any style, then proceed to Step 3 and click the **Save** button.

Repeat the same process for the following objects:

- HandsMen Order
- HandsMen Product
- Inventory
- Marketing Campaign

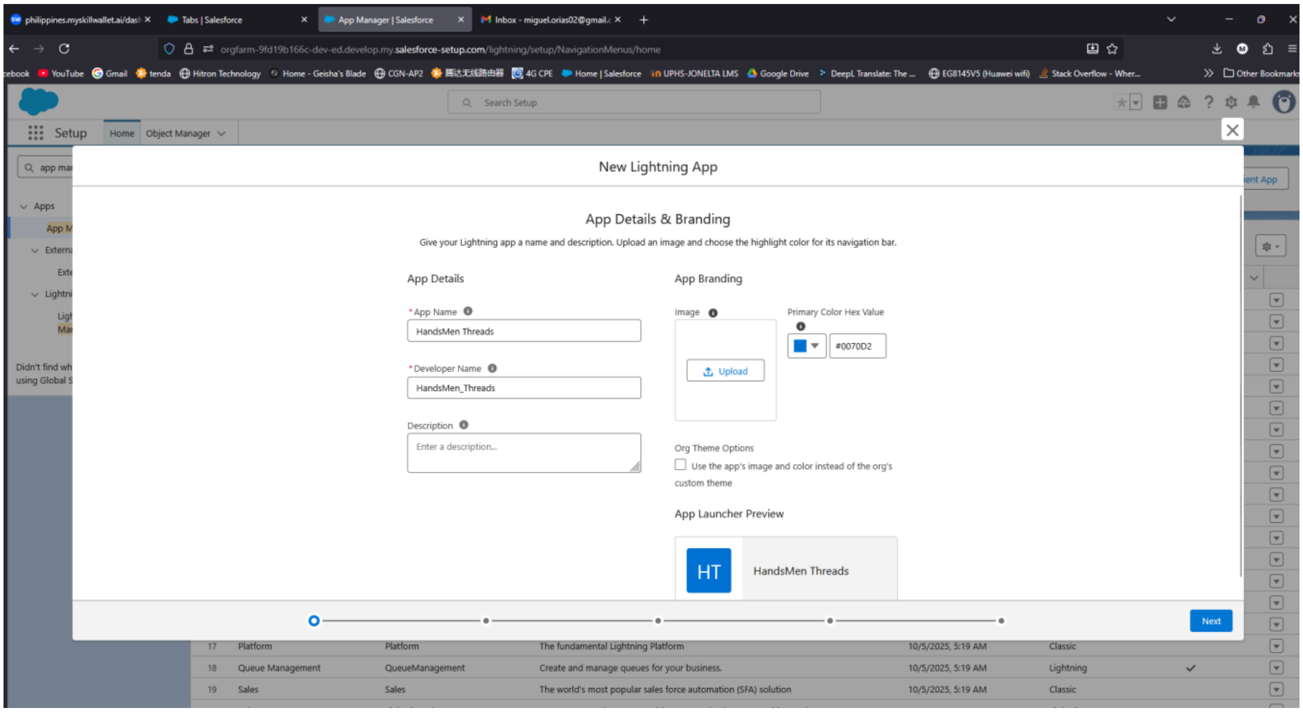
App Manager:

HandsMen Threads



In **Quick Find**, search for **App Manager**, and in the Lightning Experience App Manager setup, click the **New Lightning App** button.

1. **Step 1**

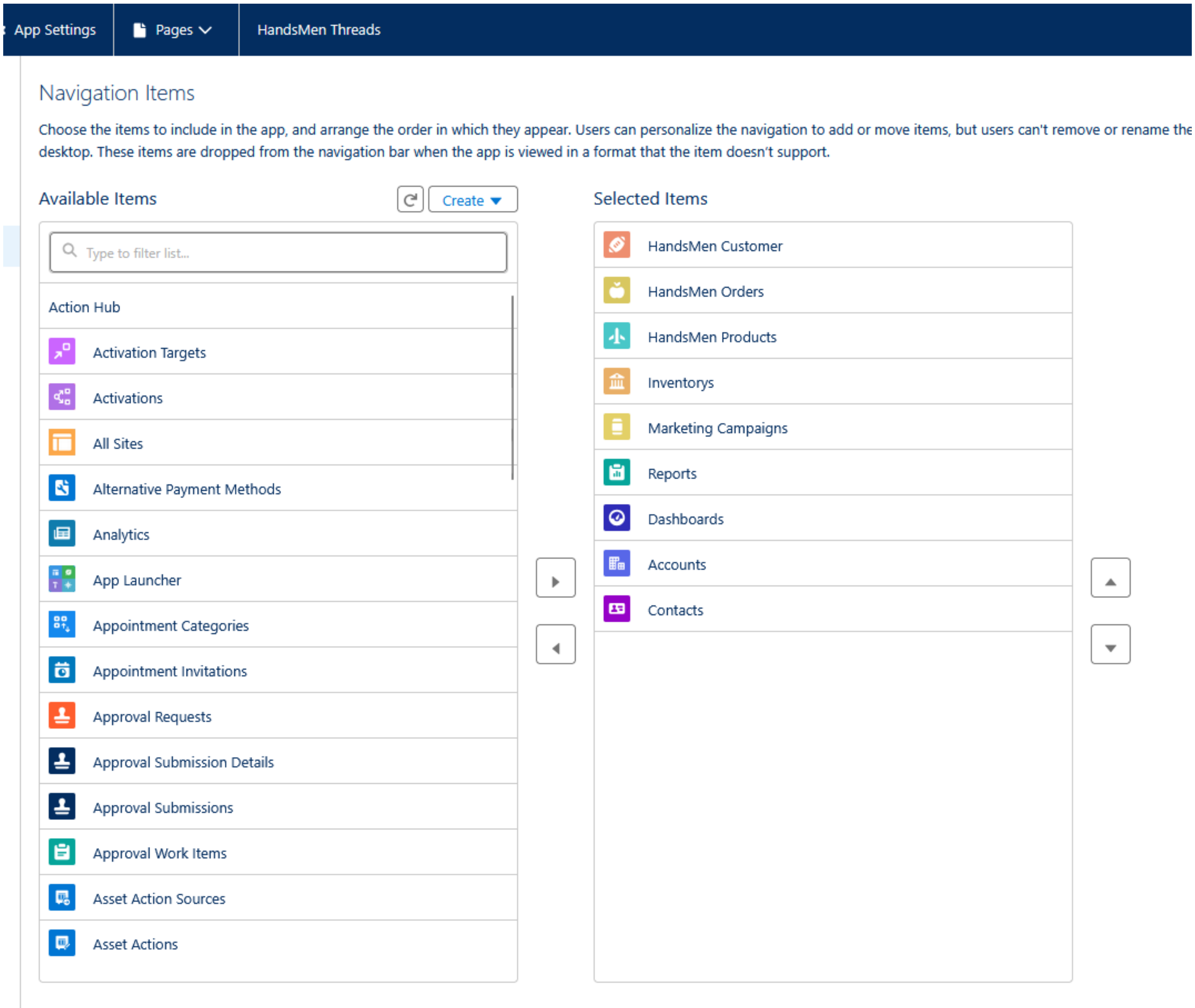


In **App Details**, enter the following:

- **App Name:** HandsMen Threads
- **Developer Name:** HandsMen_Threads
- **Description:** Enter any description you prefer.

After that, click **Next** button until you reach Step 4.

2. Step 4

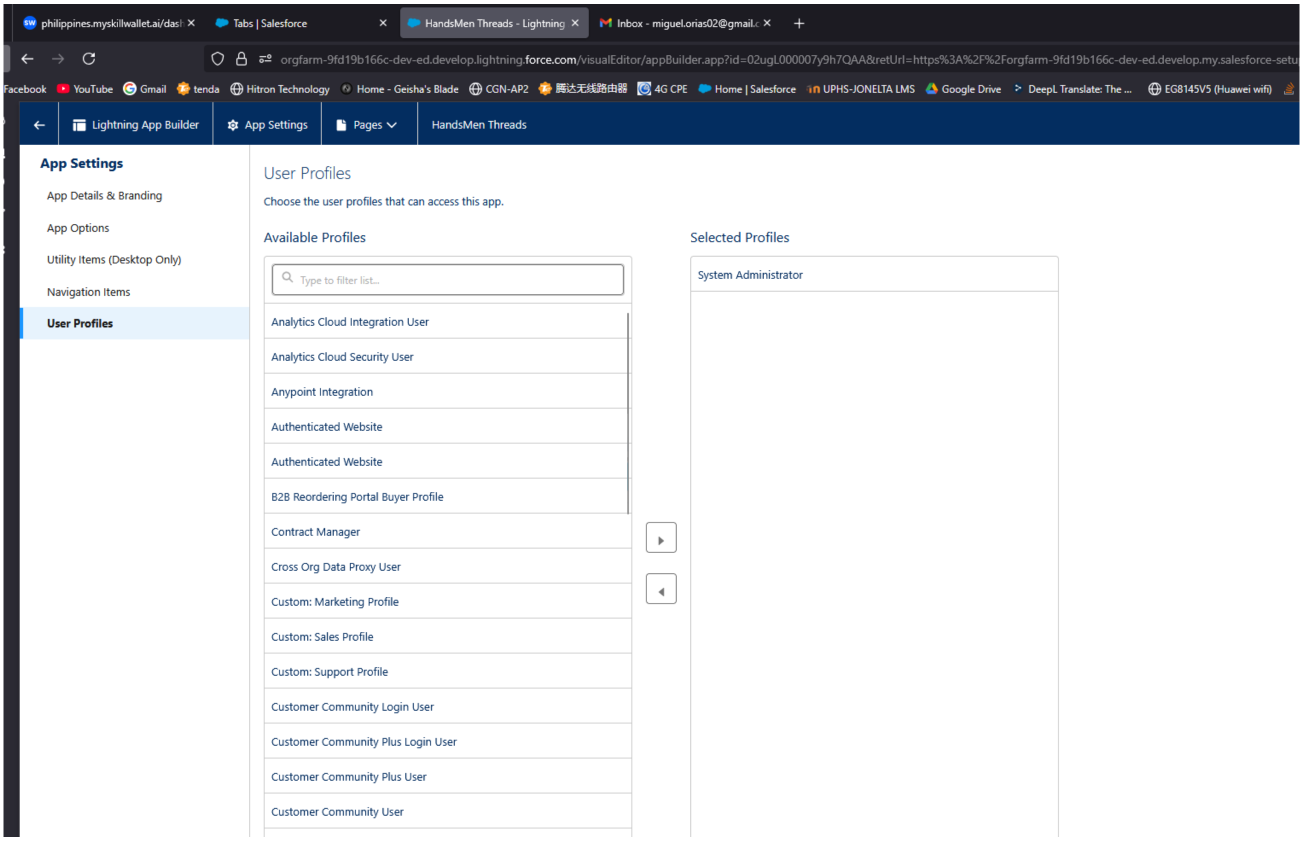


Add the following items to the app:

- **HandsMen Customer**
- **HandsMen Orders**
- **HandsMen Products**
- **Inventorys**
- **Reports**
- **Dashboards**
- **Account**
- **Contacts**

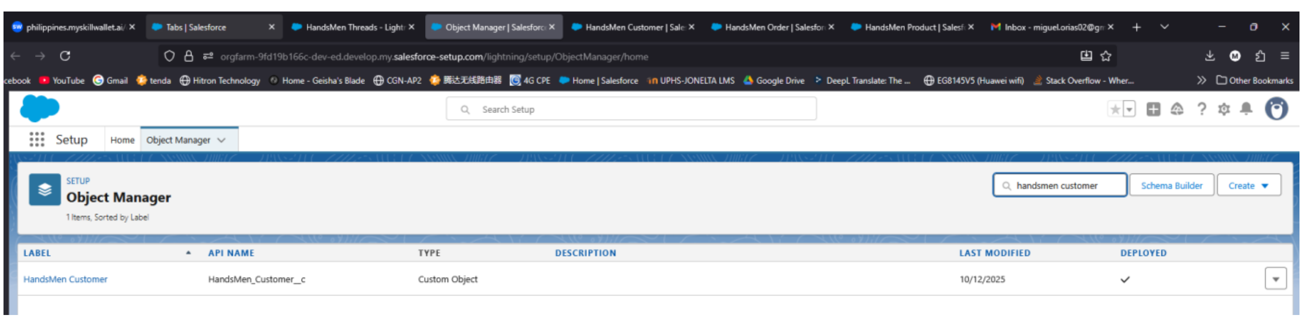
- **Marketing Campaigns**

3. Step 5

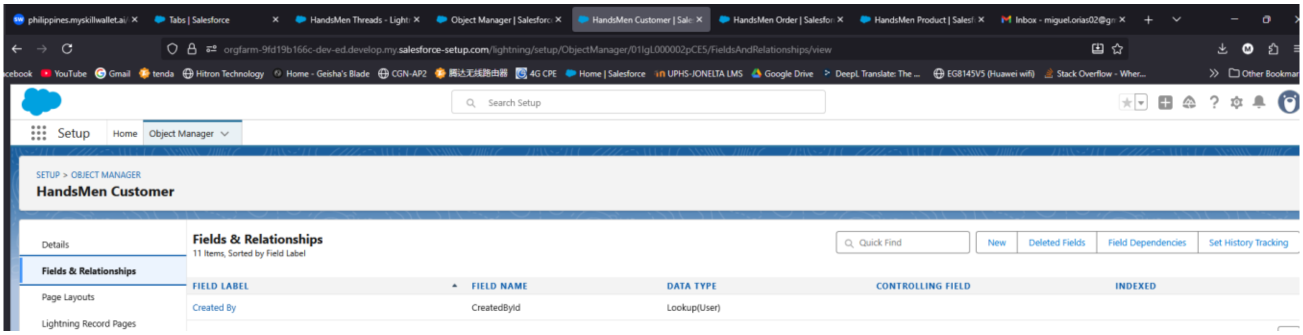


Add **System Administrator** then click **Save s Finish**.

Fields:

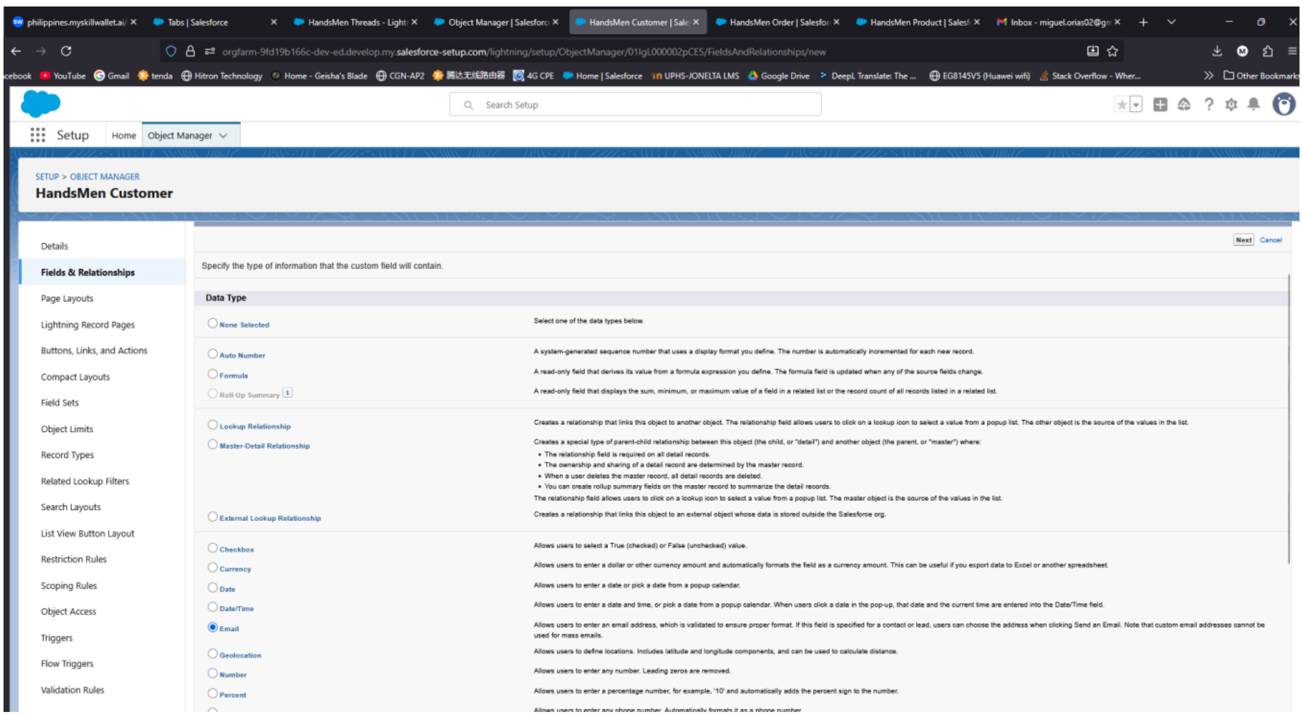


In the **Object Manager** tab, search for **HandsMen Customer**, then click on it to open the object settings.

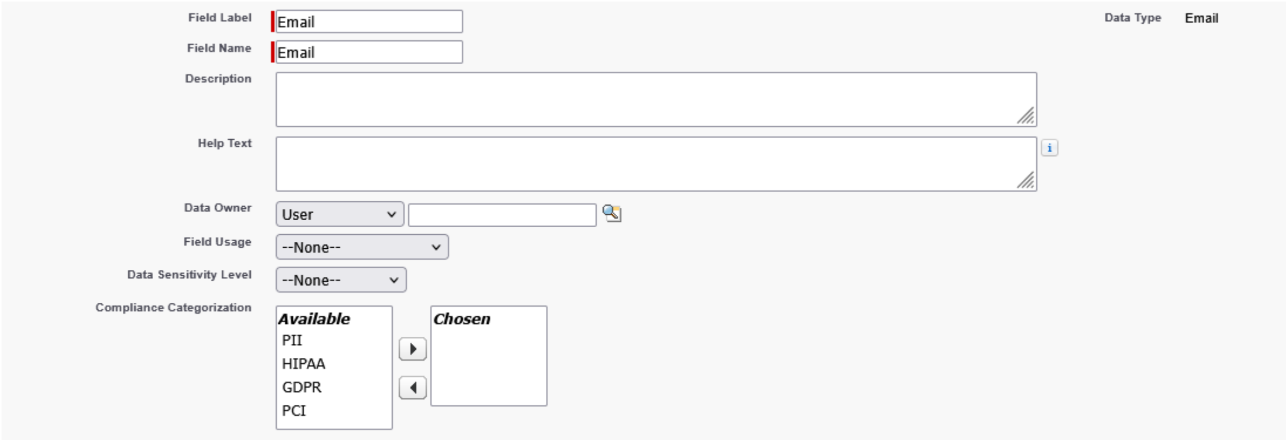


In the HandsMen Customer object settings, click **Fields s Relationships**, then click the **New** button to create a new custom field.

1. Email

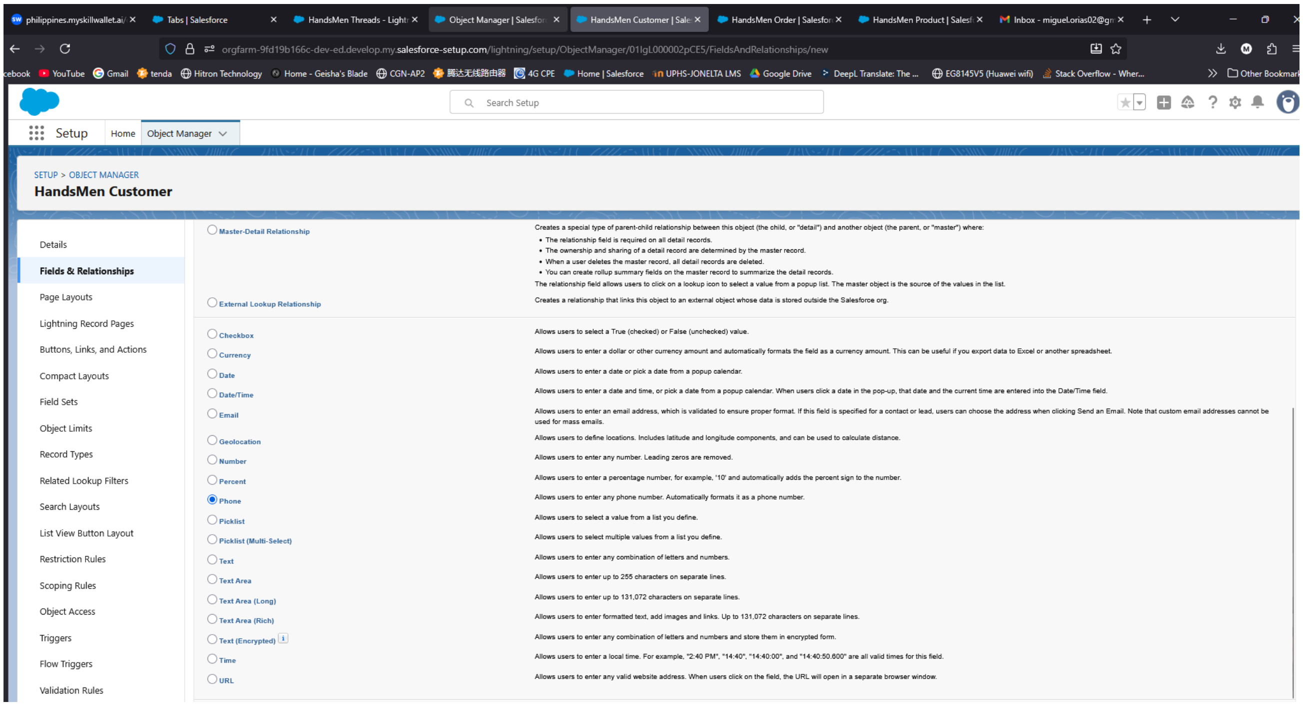


In **Step 1**, select the **Email** data type, then click the **Next** button.

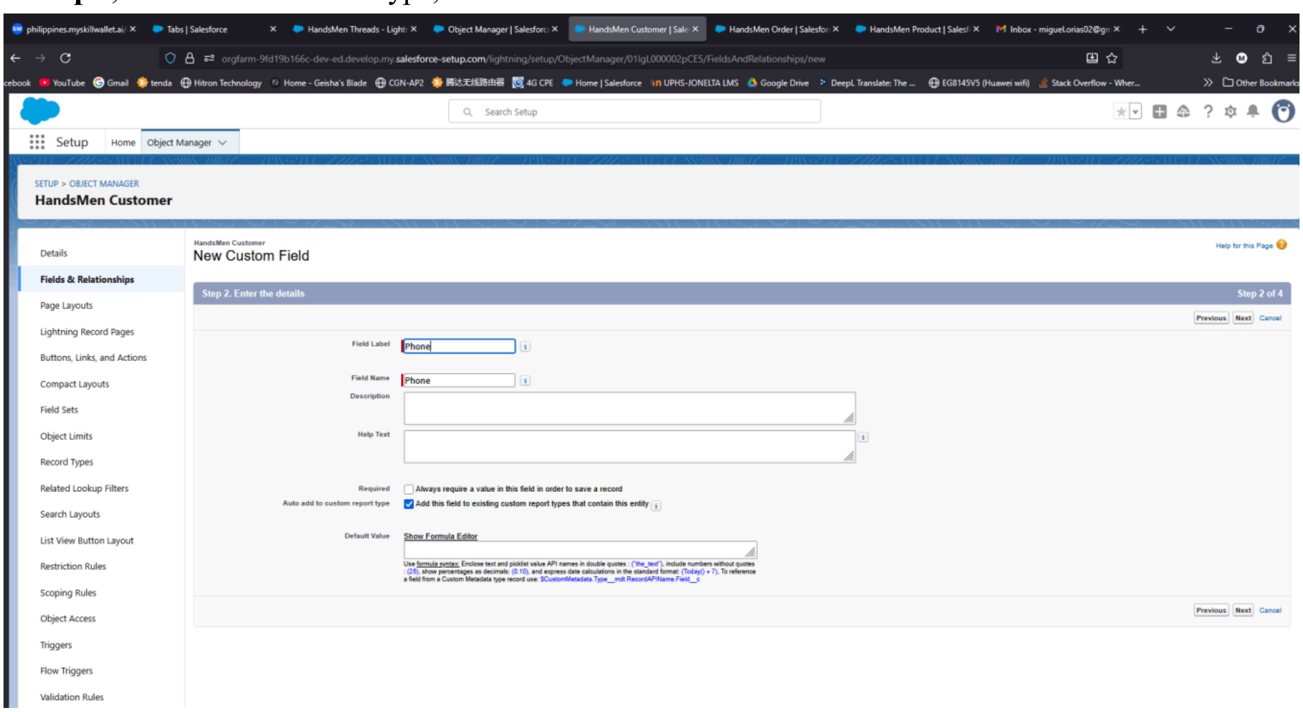


In **Step 2**, put **Email** in Field Label and in Field Name, then proceed to the last step and click the **Save** button.

2. Phone

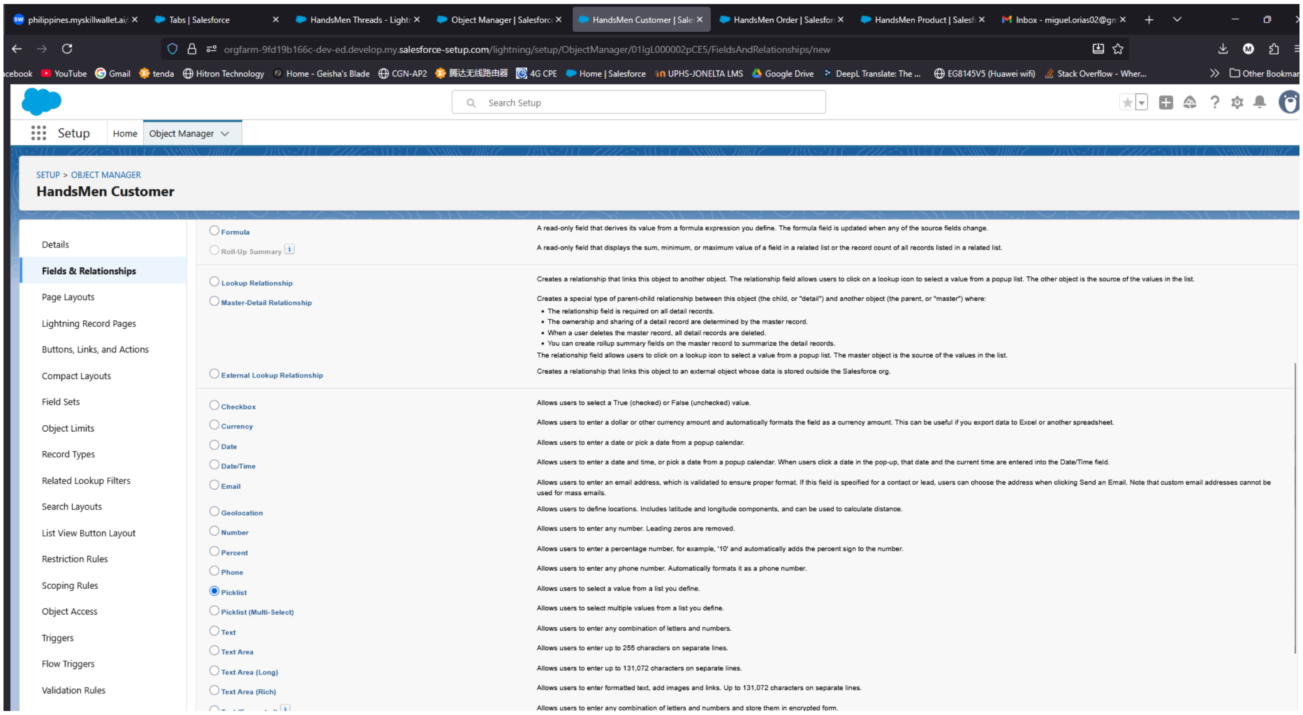


In **Step 1**, select the **Phone** data type, then click the **Next** button.

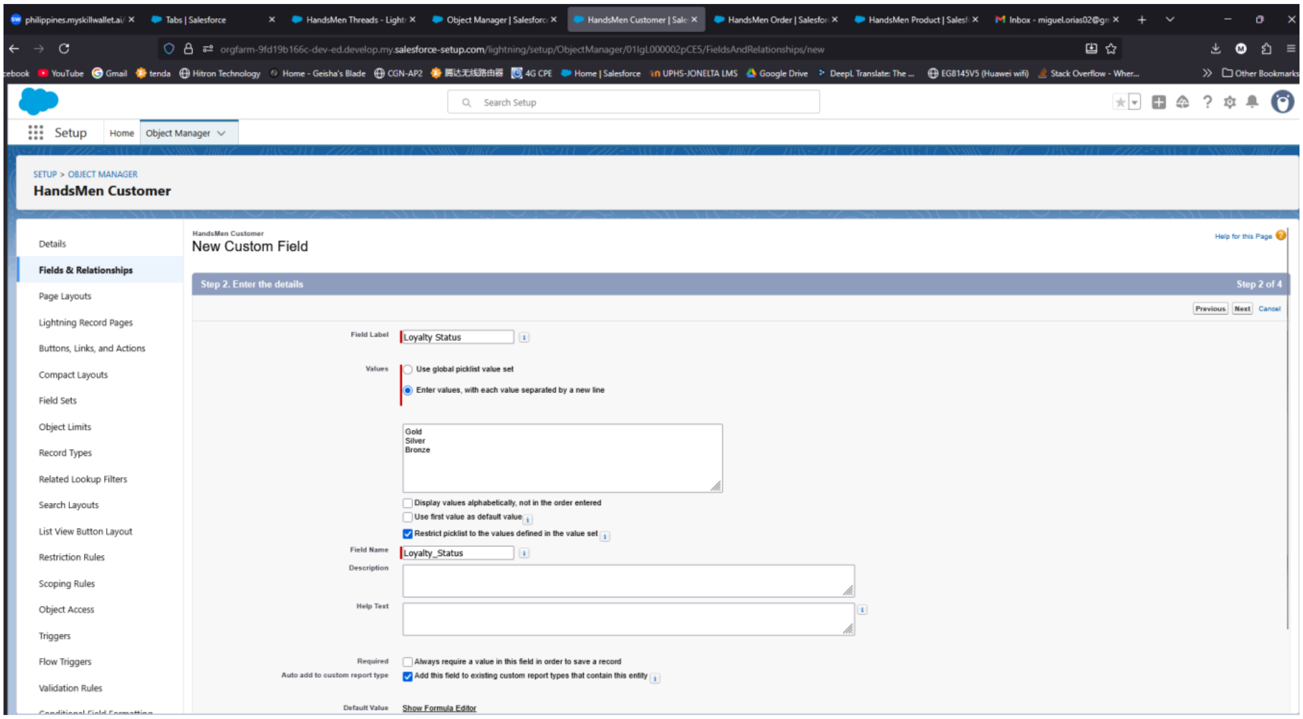


In **Step 2**, put **Phone** in Field Label and in Field Name, then proceed to the last step and click the **Save** button.

3. Loyalty Status



In **Step 1**, select the **Picklist** data type, then click the Next button.



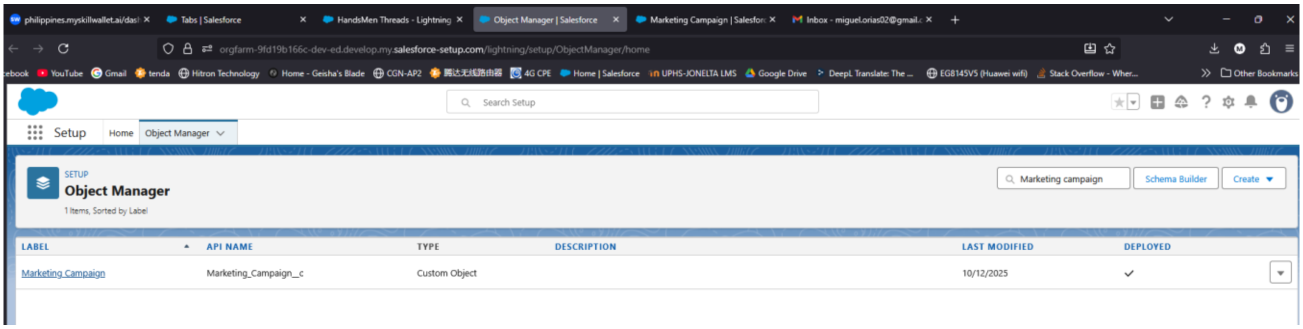
In Step 2, configure the following as follows:

- **Field Label:** Loyalty Status
- **Field Name:** Loyalty_Status
- **Values:** Enter values, with each value separated by a new line:
    - **Gold**
    - **Silver**
    - **Bronze**

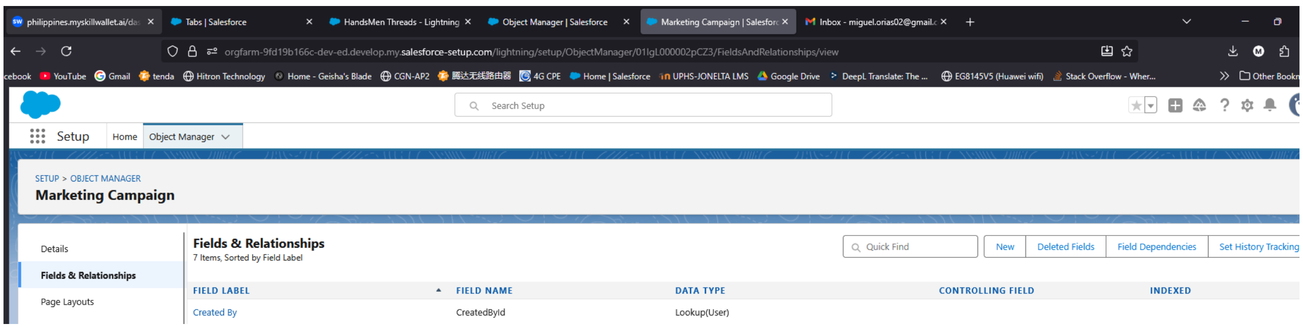
Then, proceed to the last step and click the **Save** button.

4. Create Lookup Relationship

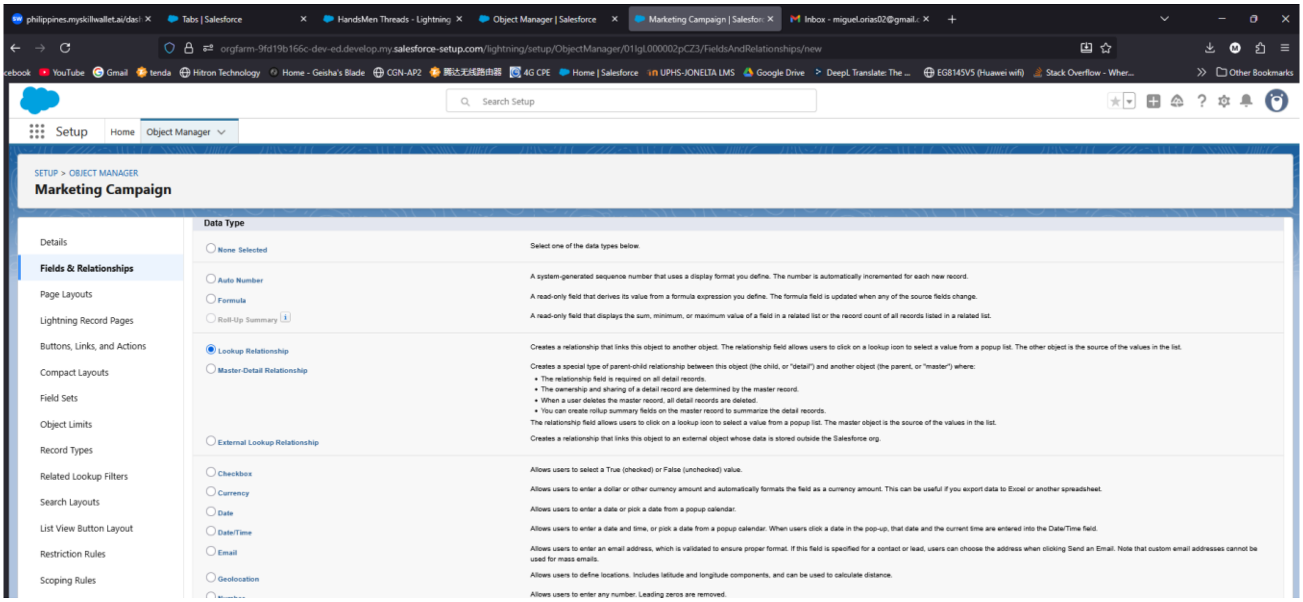
- **Creating Lookup Relationship between Marketing Campaign and HandsSome Customer:**



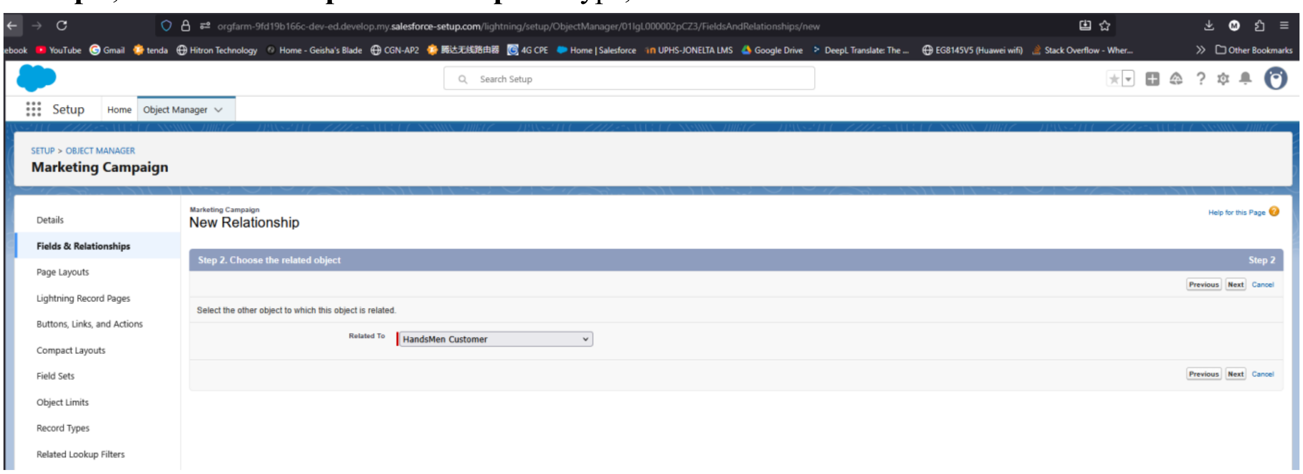
In **Object Manager**, search for **Marketing Campaign**, then select it to open the object settings.



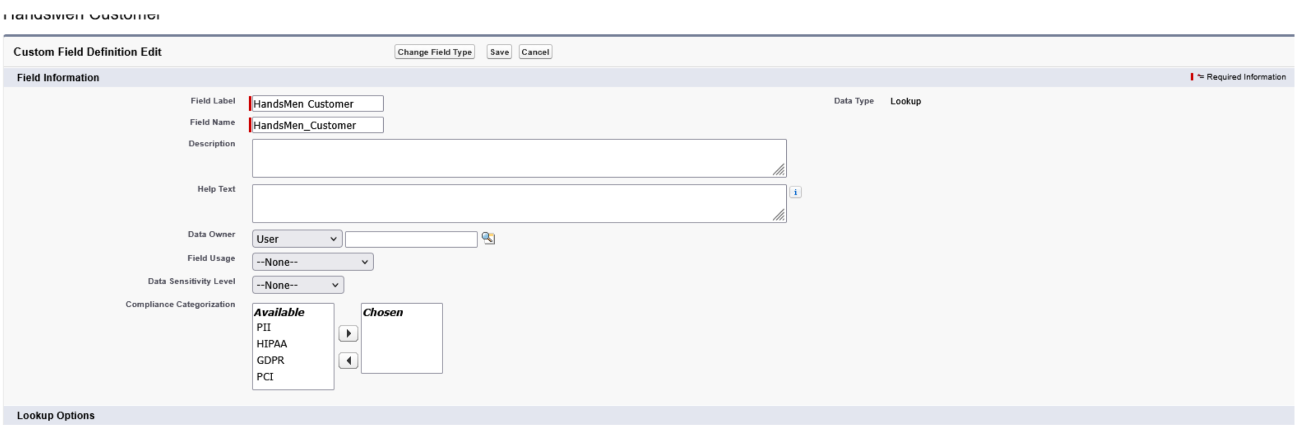
In the **Marketing Campaign** object settings, click **Fields s Relationships**, then click the **New** button to create a new custom field.



In **Step 1**, select the **Lookup Relationship** data type, then click the **Next** button.



In **Step 2**, set **Related To** as **HandsMen Customer**.

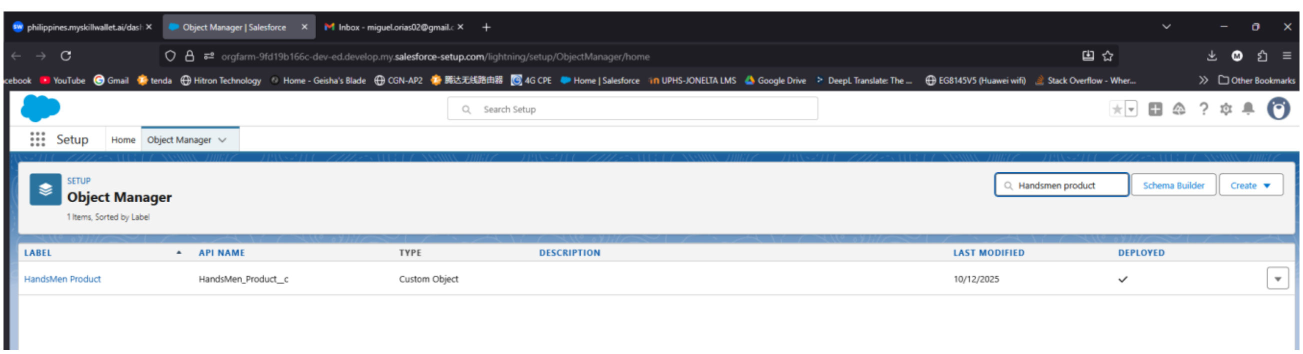


In **Step 3**, configure the relationship field as follows:

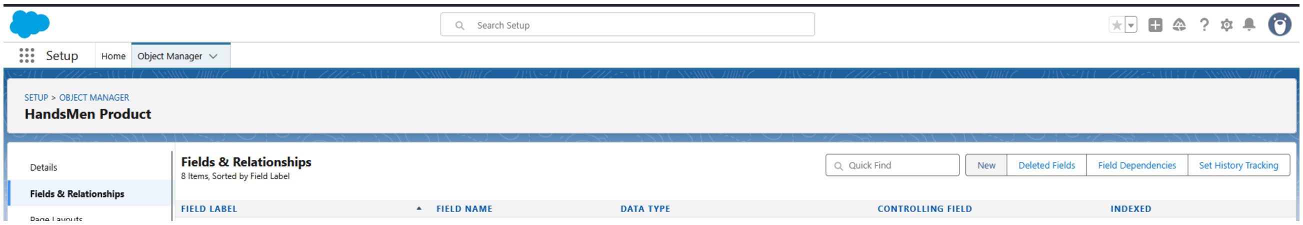
- **Field Label:** HandsMen Customer
- **Field Name:** HandsMen_Customer
- **Child Relationship Name:** Marketing_Campaigns

Then, click the **Next** button through each step until the final screen, and click the **Save** button.

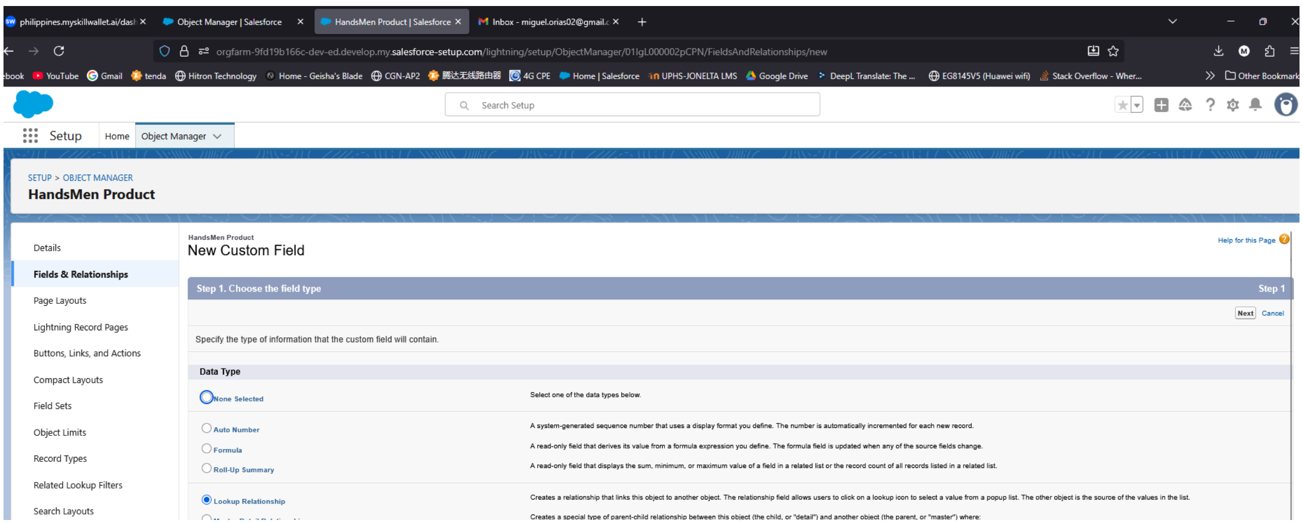
- **Creating Lookup Relationship between HandsMen Product and HandsMen Order:**



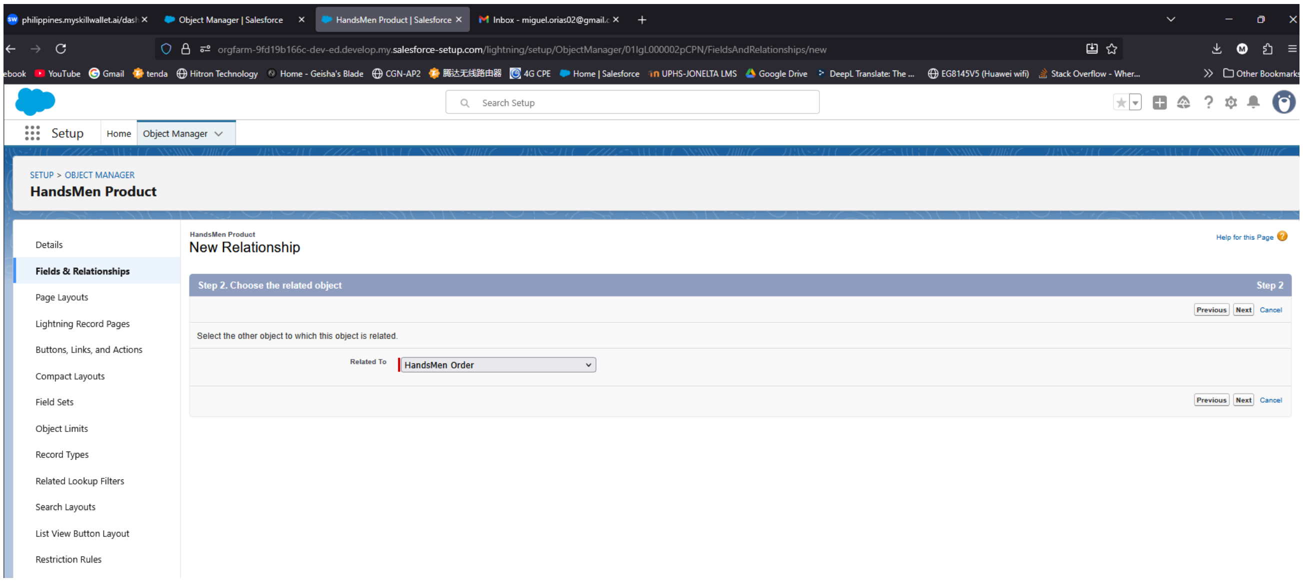
In **Object Manager**, search for **HandsMen Product**, then select it to open the object settings.



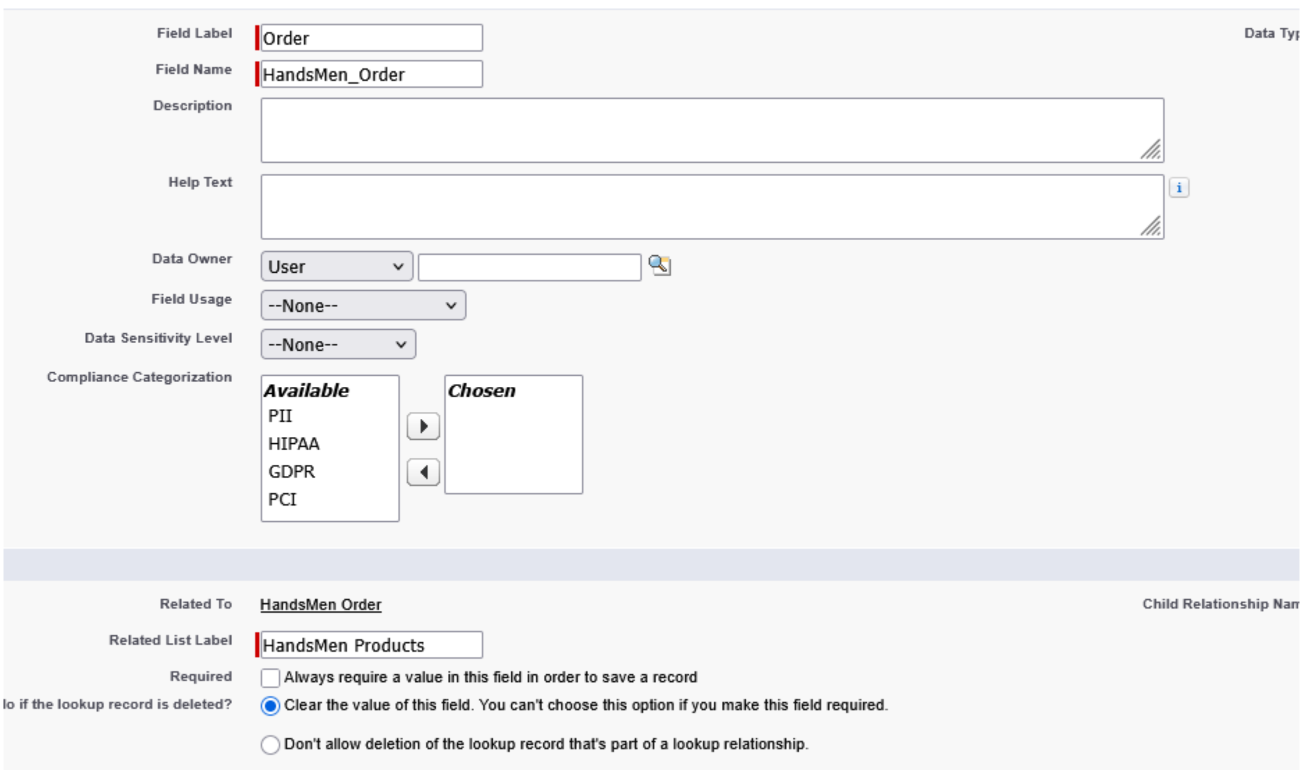
In the **HandsMen Product** object settings, click **Fields s Relationships**, then click the **New** button to create a new custom field.



In **Step 1**, select the **Lookup Relationship** data type, then click the **Next** button.



In **Step 2**, set **Related To** as **HandsMen Order**.

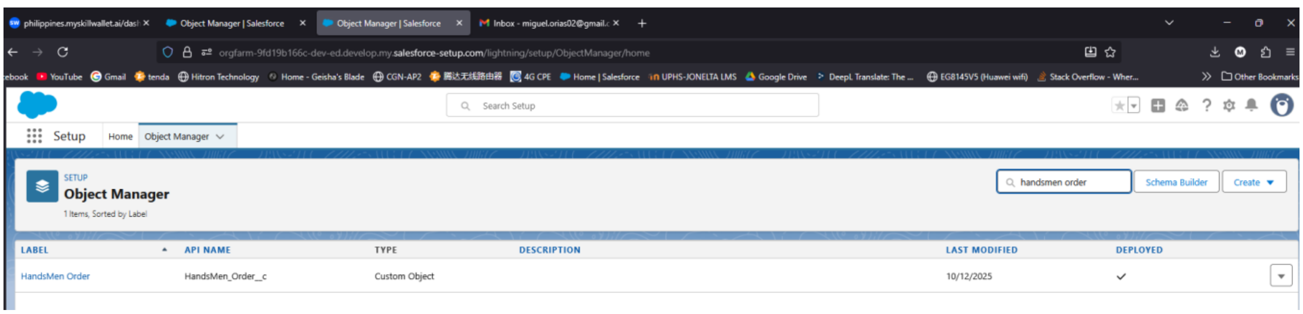


In **Step 3**, configure the relationship field as follows:

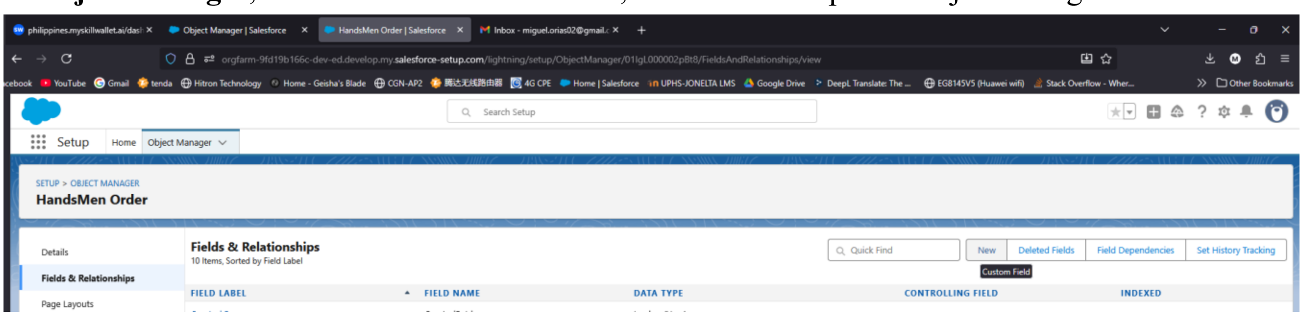
- **Field Label:** Order
- **Field Name:** HandsMen_Order
- **Child Relationship Name:** HandsMen_Products

Then, click the **Next** button through each step until the final screen, and click the **Save** button.

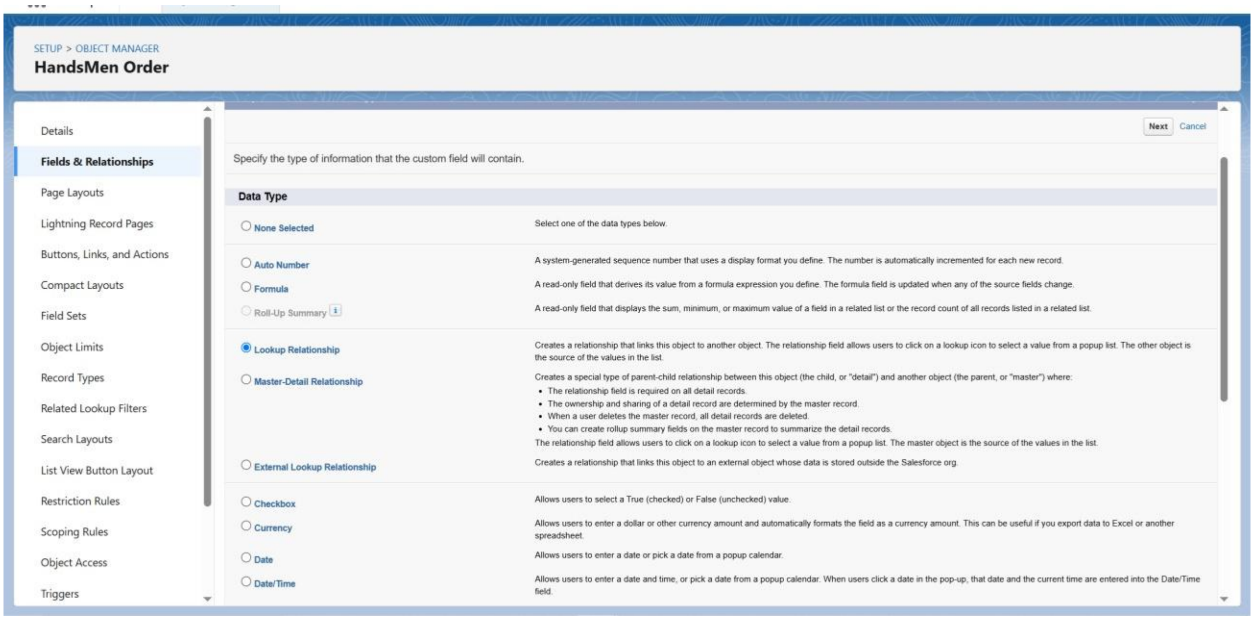
- **Creating Lookup Relationship between HandsMen Order and HandsMen Customer:**



In **Object Manager**, search for **HandsMen Order**, then select it to open the object settings.

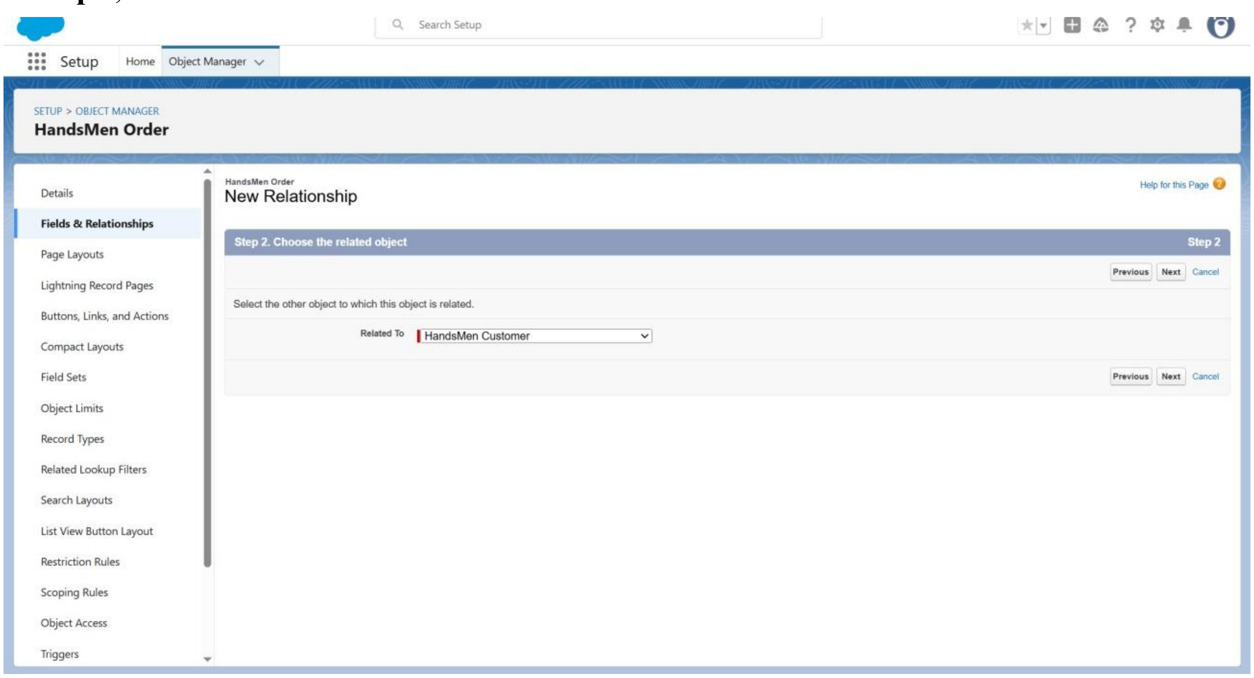


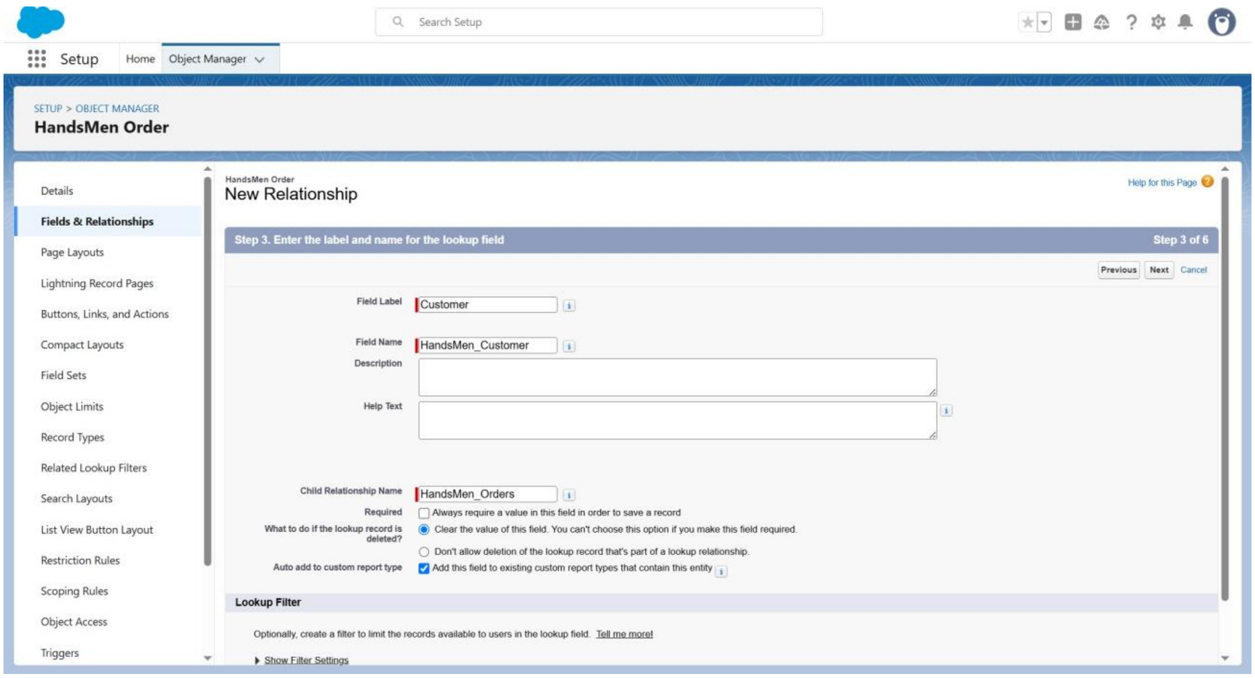
In the **HandsMen Order** object settings, click **Fields s Relationships**, then click the **New** button to create a new custom field.



In **Step 1**, select the **Lookup Relationship** data type, then click the Next button.

In **Step 2**, set **Related To** as **HandsMen Customer**.

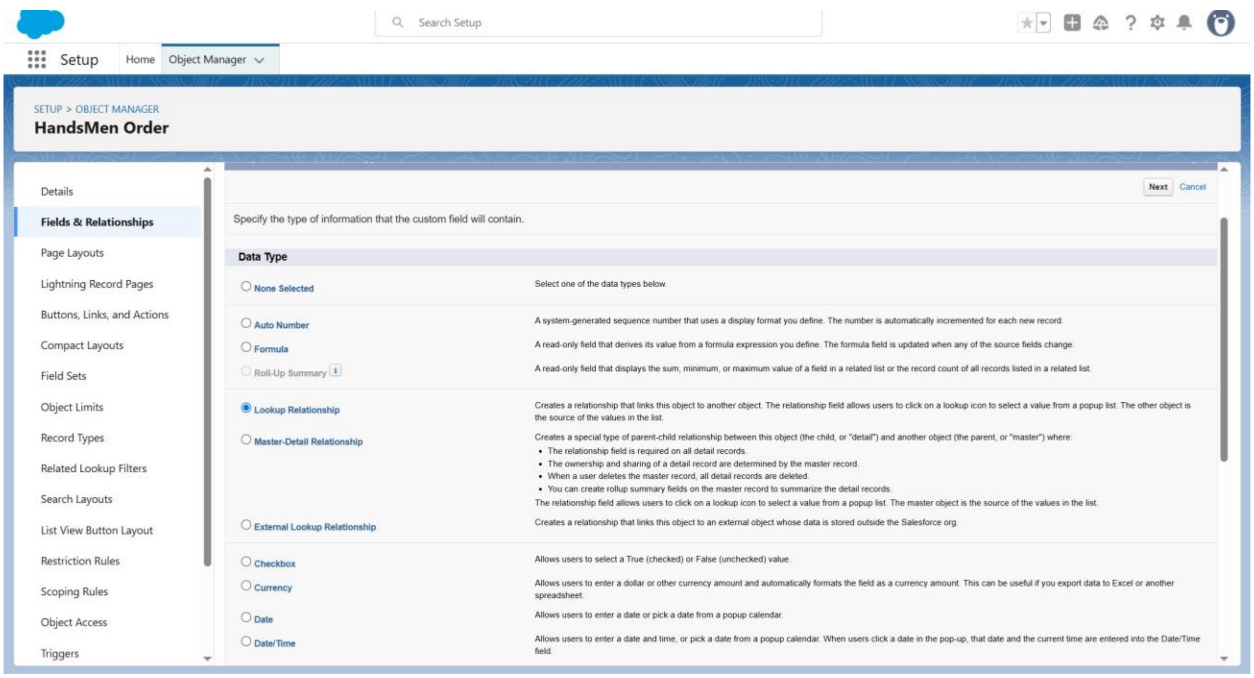




In **Step 3**, configure the relationship field as follows:

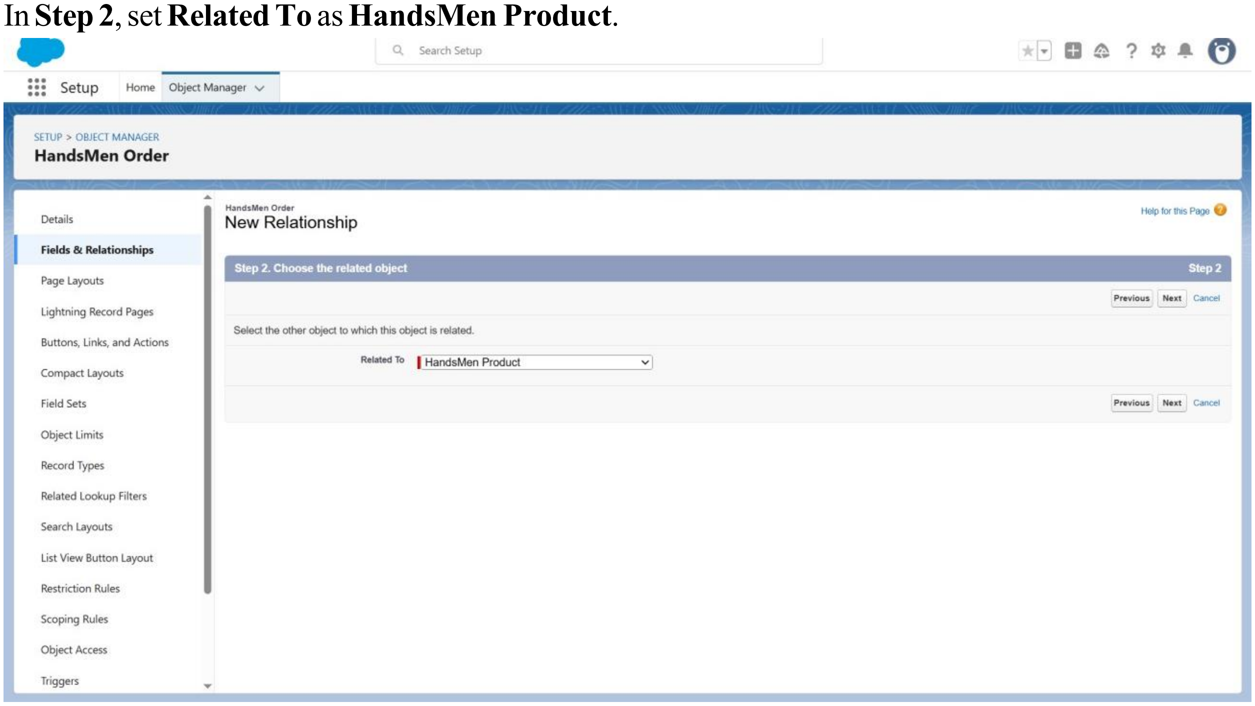
- **Field Label:** Customer
- **Field Name:** HandsMen_Customer
- **Child Relationship Name:** HandsMen_Orders

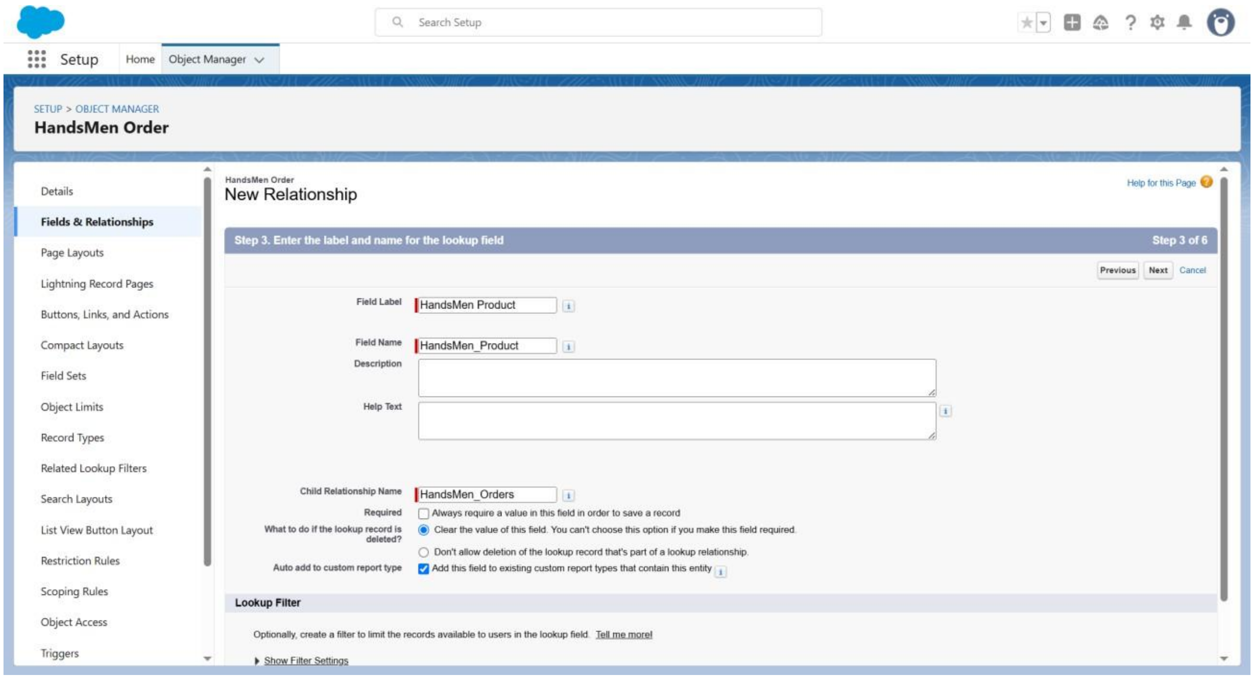
Then, click the **Next** button through each step until the final screen, and click the **Save s New** button.



In **Step 1**, select the **Lookup Relationship** data type, then click the Next button.

In **Step 2**, set **Related To** as **HandsMen Product**.



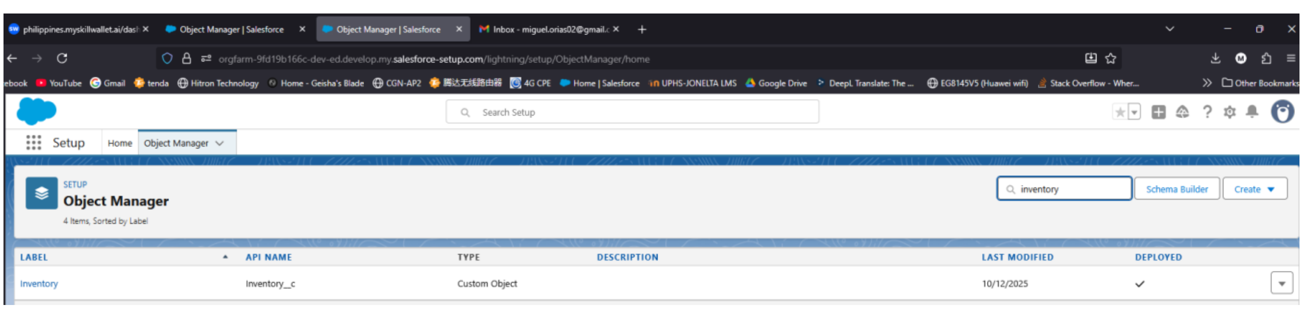


In **Step 3**, configure the relationship field as follows:

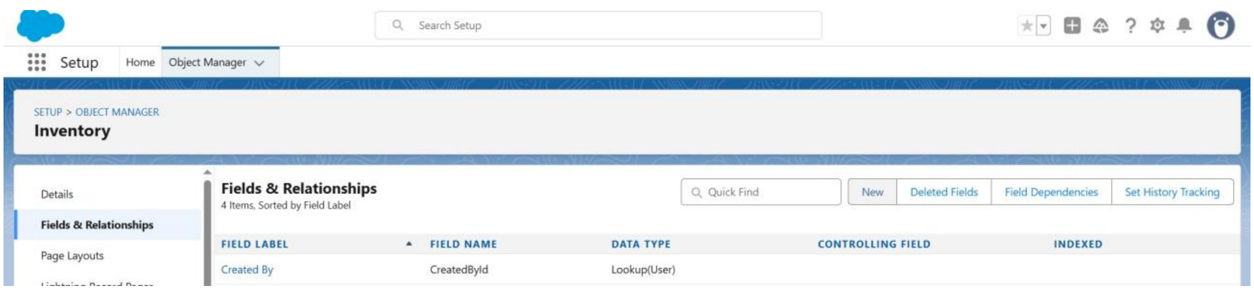
- **Field Label:** HandsMen Product
- **Field Name:** HandsMen_Product
- **Child Relationship Name:** HandsMen_Orders

Then, click the **Next** button through each step until the final screen, and click the **Save** button.

- **Creating Master-Detail Relationship between Inventory and HandsSome Product:**

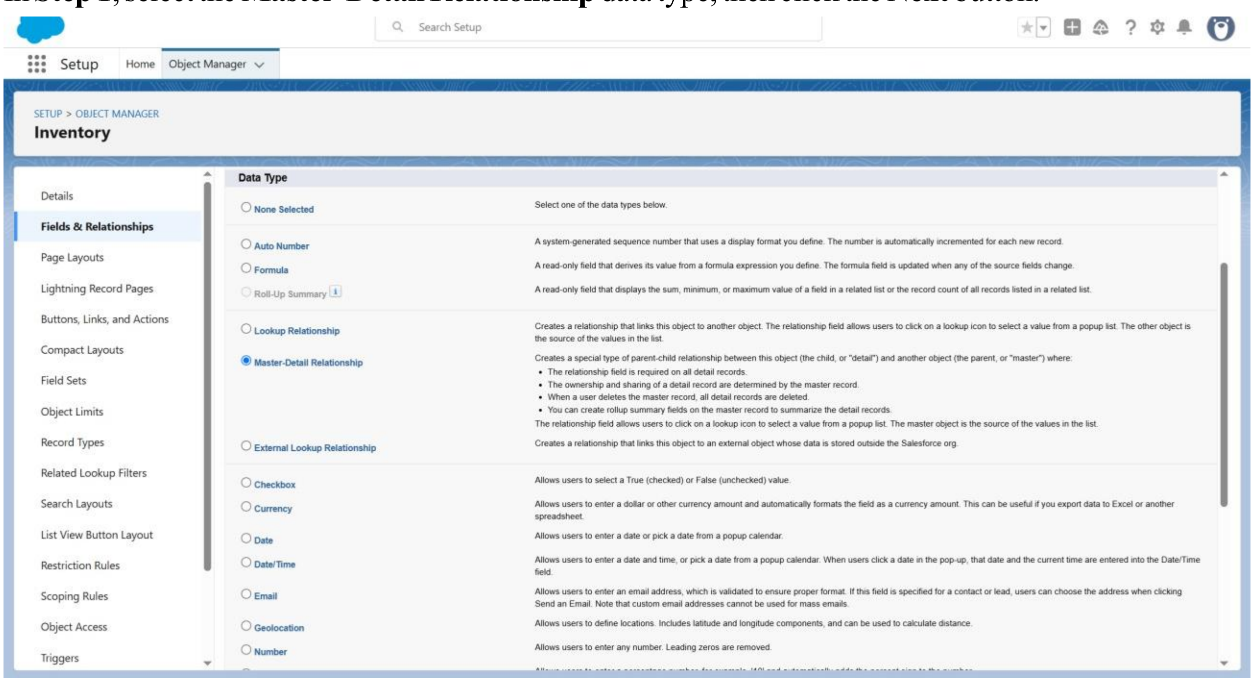


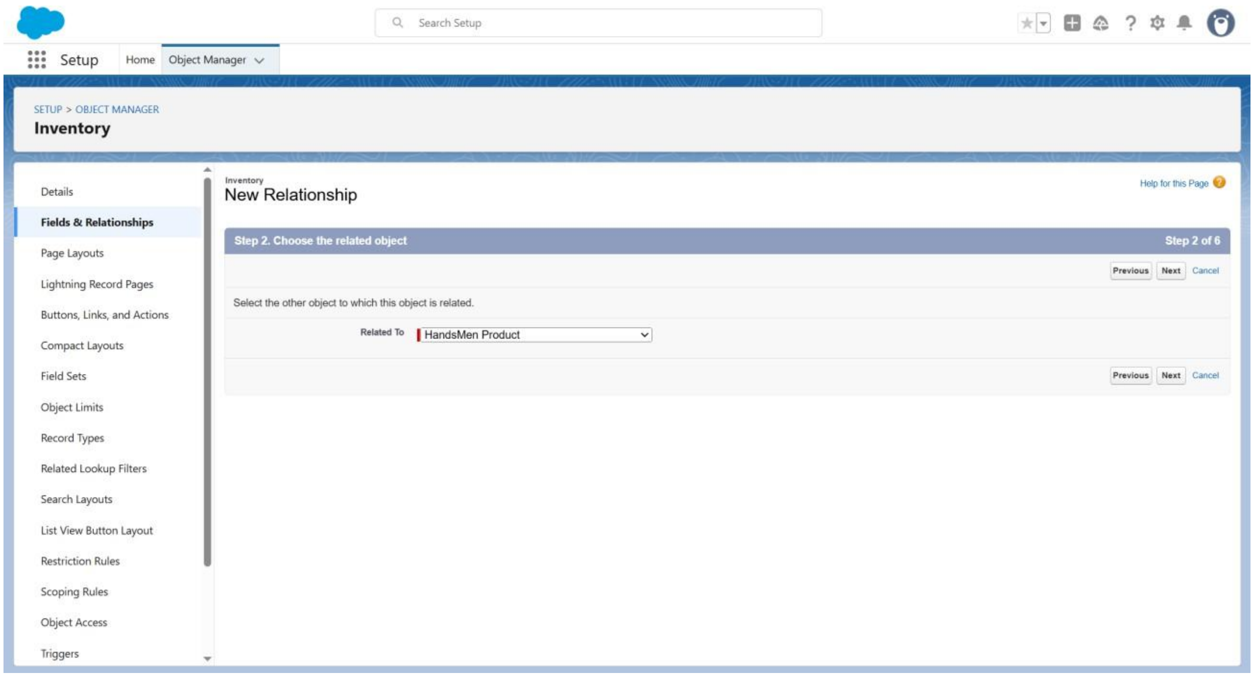
In **Object Manager**, search for **Inventory**, then select it to open the object settings.



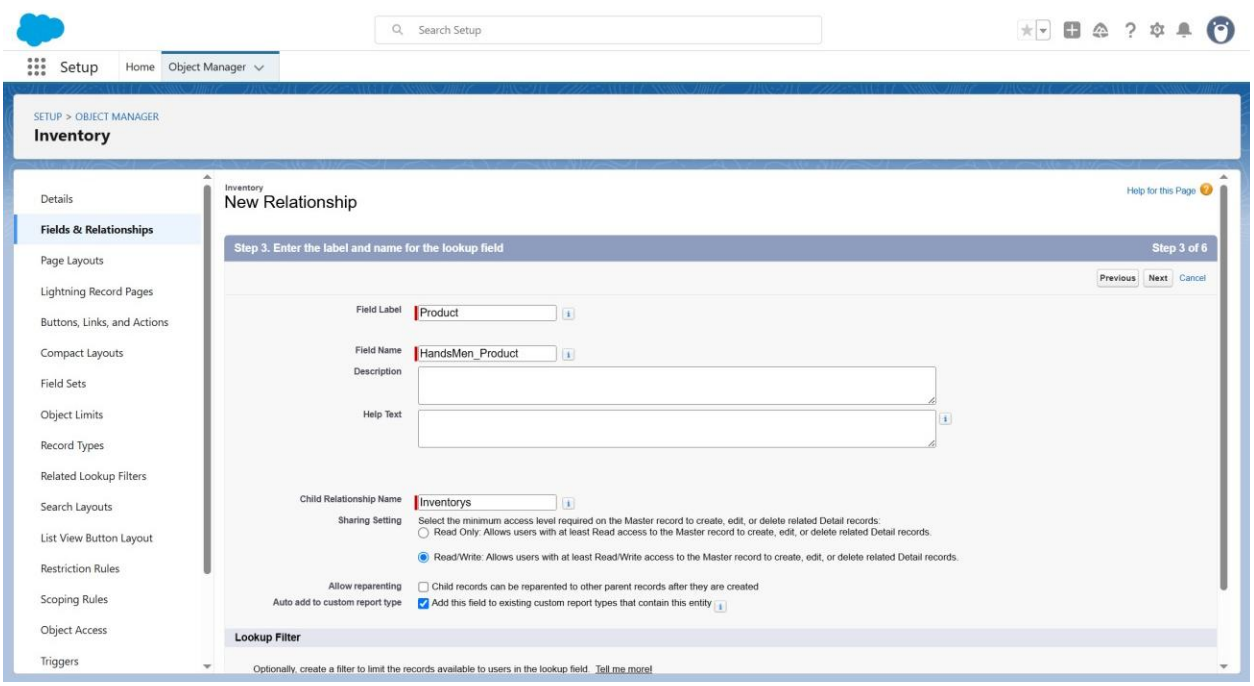
In the **Inventory** object settings, click **Fields s Relationships**, then click the **New** button to create a new custom field.

In **Step 1**, select the **Master-Detail Relationship** data type, then click the **Next** button.





In **Step 2**, set **Related To** as **HandsMen Product**.



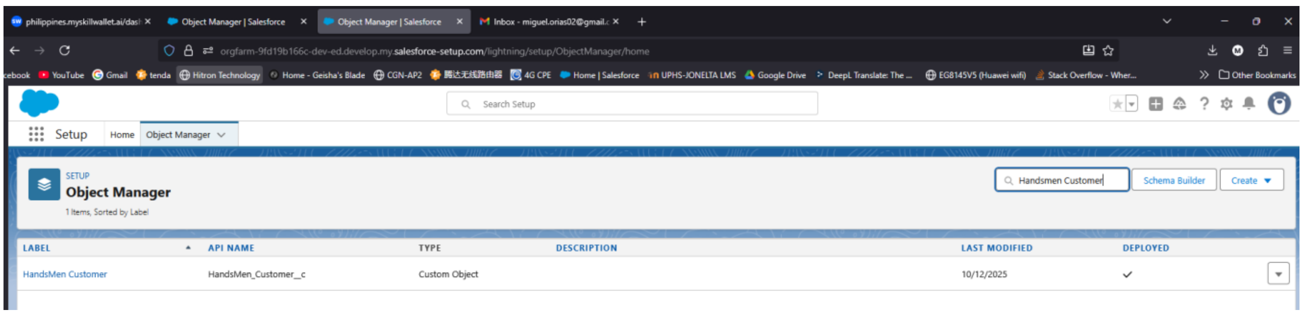
In **Step 3**, configure the relationship field as follows:

- **Field Label:** Product
- **Field Name:** HandsMen_Product
- **Child Relationship Name:** Inventorys

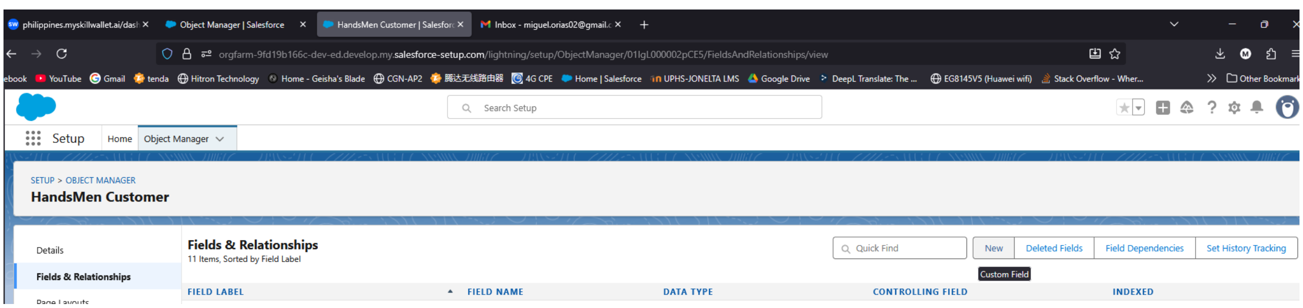
Then, click the **Next** button through each step until the final screen, and click the **Save** button.

5. Formula Fields

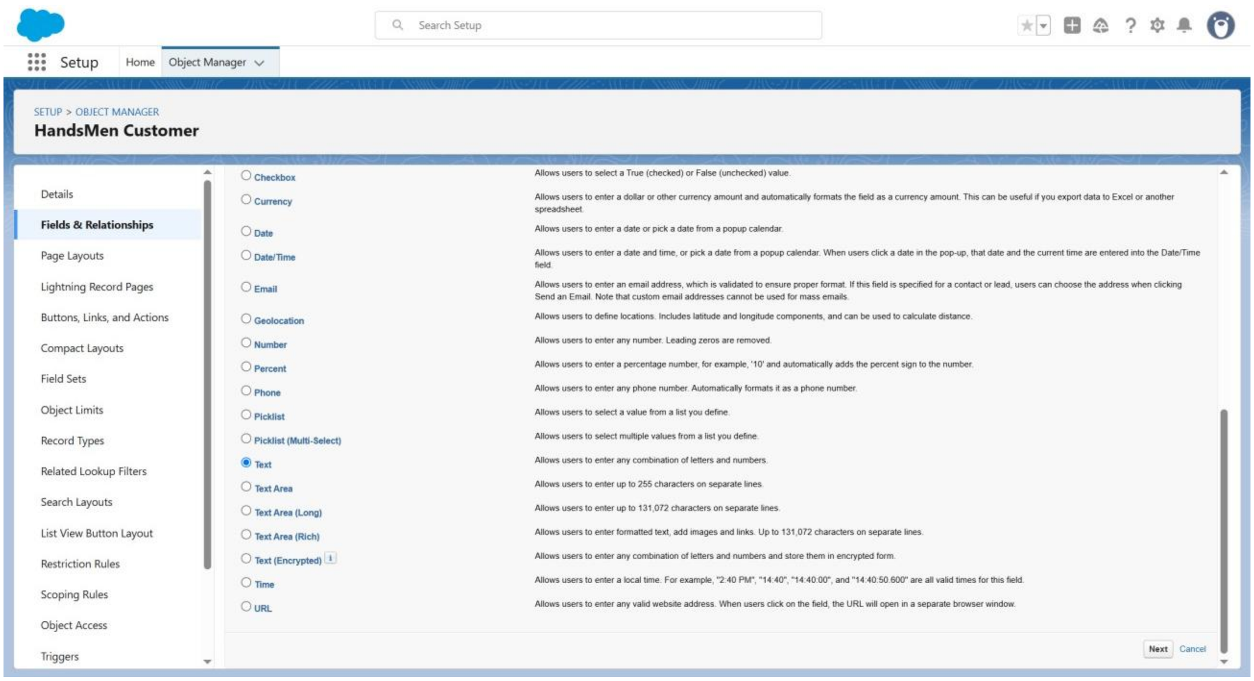
- **Full Name:**



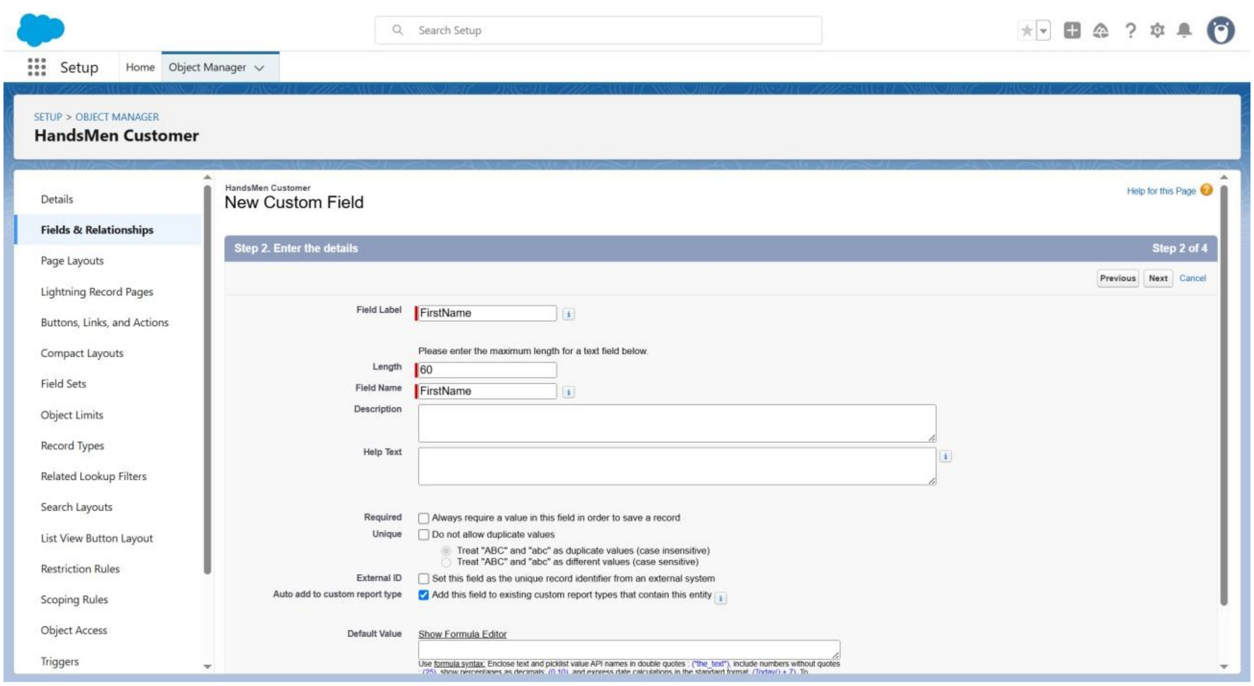
In Object Manager, search for **HandsMen Customer**, then select it to open the object settings.



In the **HandsMen Customer** object settings, click **Fields s Relationships**, then click the **New** button to create a new custom field.



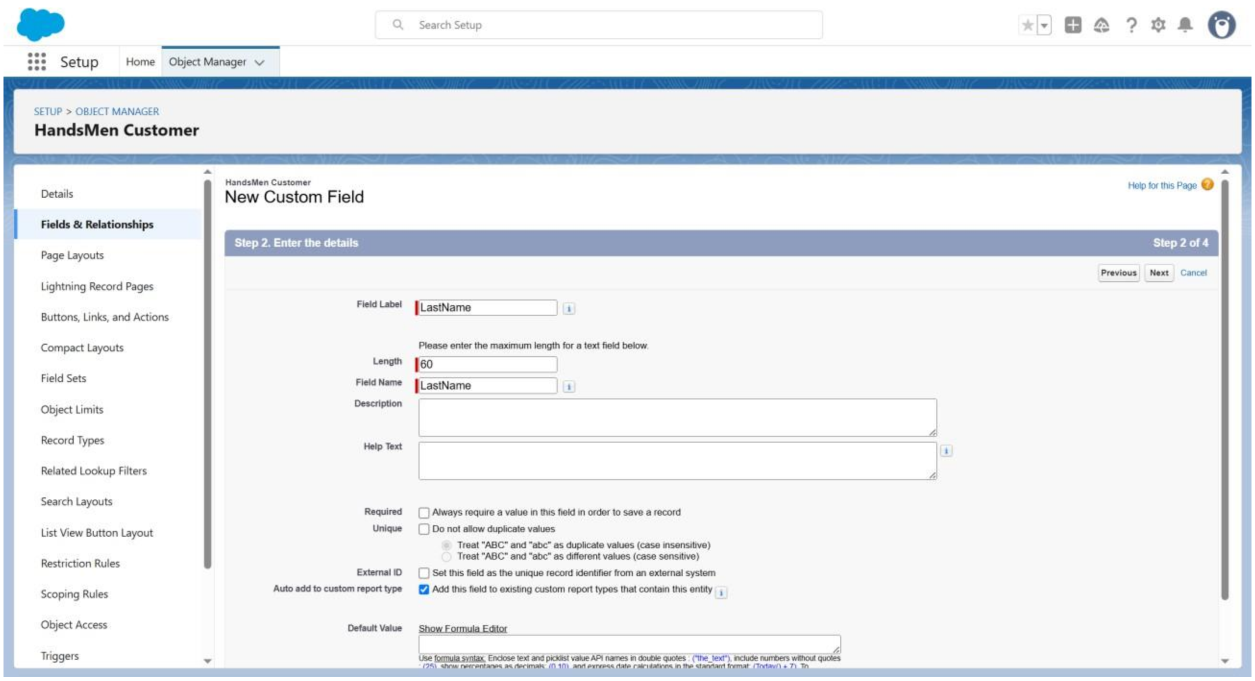
In **Step 1**, select the **Text** data type, then click the **Next** button.



In **Step 2**, configure the field as follows:

- **Field Label:** FirstName
- **Field Name:** FirstName
- **Length:** 60

Then, click **Next** through each step until the final screen, and click the **Save** button.

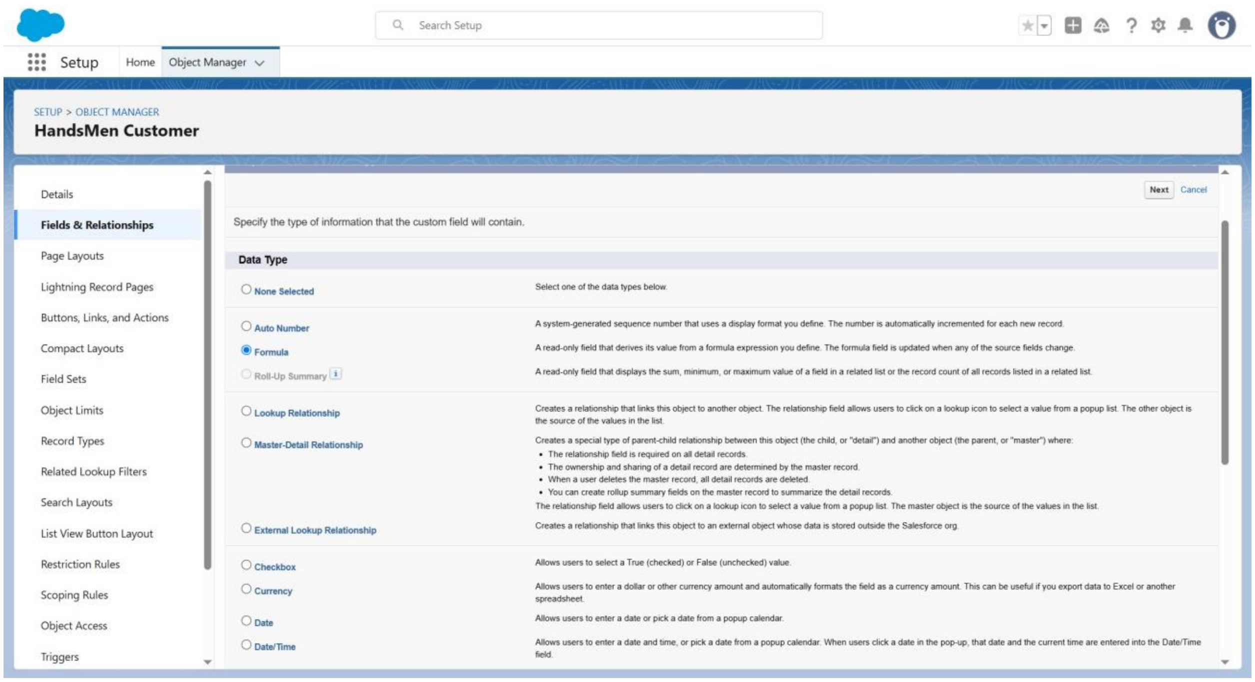


Create a new field with the same data type used for FirstName (**Text**). In **Step 2**,

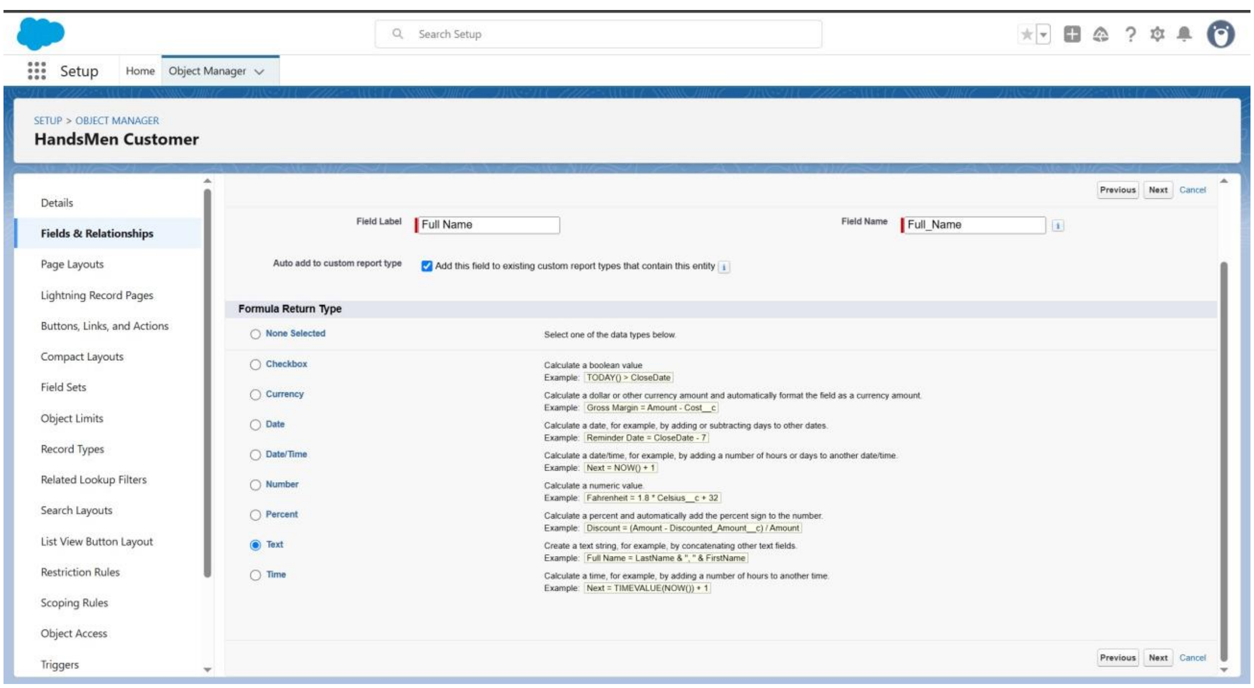
configure the field as follows:

- **Field Label:** LastName
- **Field Name:** LastName
- **Length:** 60

Then, click **Next** through each step until the final screen, and click the **Save** button.



Create a new field, and in **Step 1**, select the **Formula** data type.

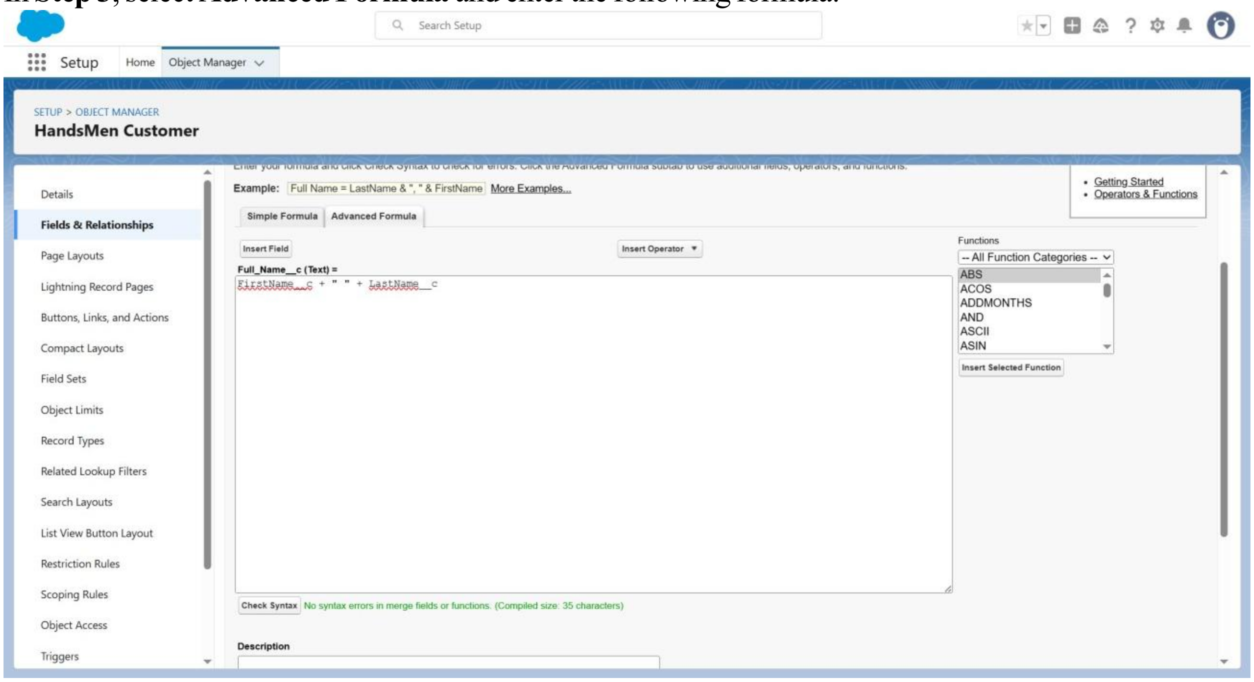


In **Step 2**, configure the formula field as follows:

- **Field Label:** Full Name
- **Field Name:** Full_Name
- In the **Formula Return Type** section, select **Text**.

Then, click the **Next** button to proceed.

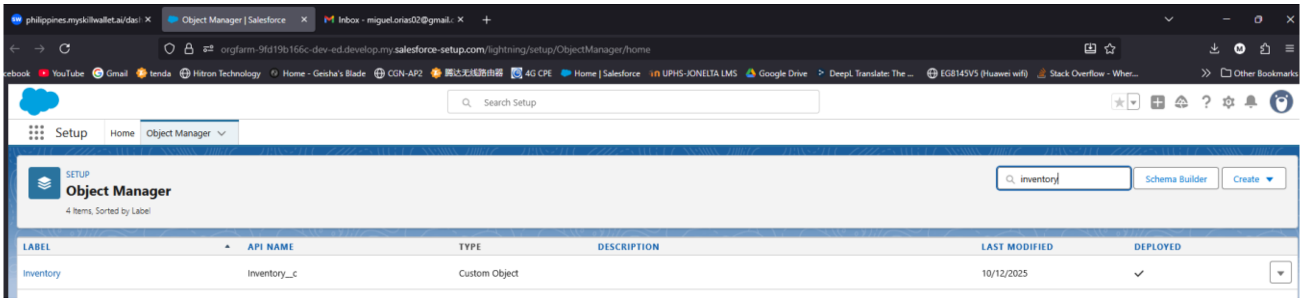
In **Step 3**, select **Advanced Formula** and enter the following formula:



- **FirstName_c + " " + LastName_c**

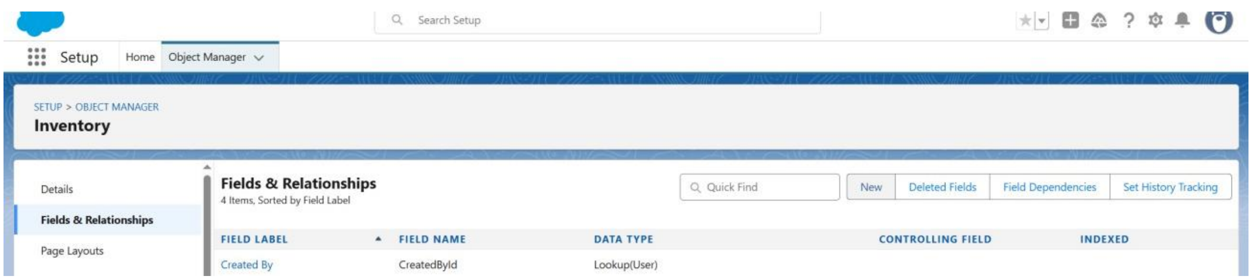
Then, click **Next** through each step until the final screen, and click the **Save** button.

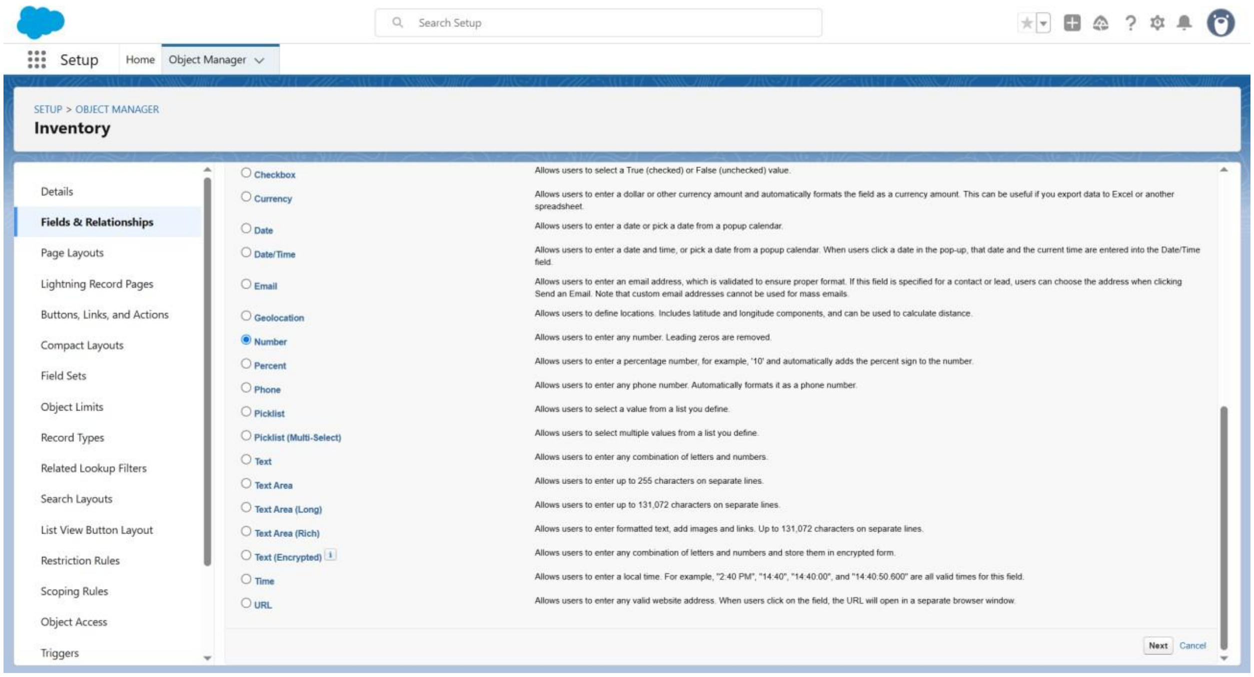
- **Stock Quantity:**



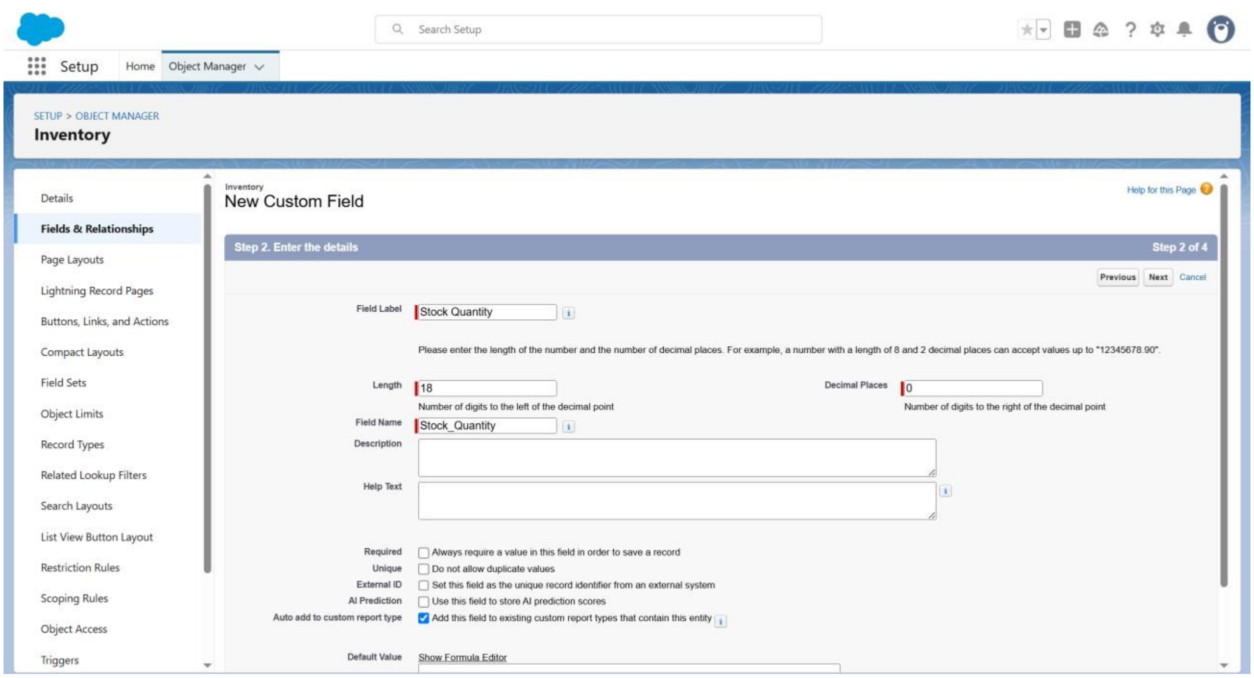
In **Object Manager**, search for **Inventory**, then select it to open the object settings.

In the **Inventory** object settings, click **Fields s Relationships**, then click the **New** button to create a new custom field.





In **Step 1**, select the **Number** data type, then click the **Next** button.



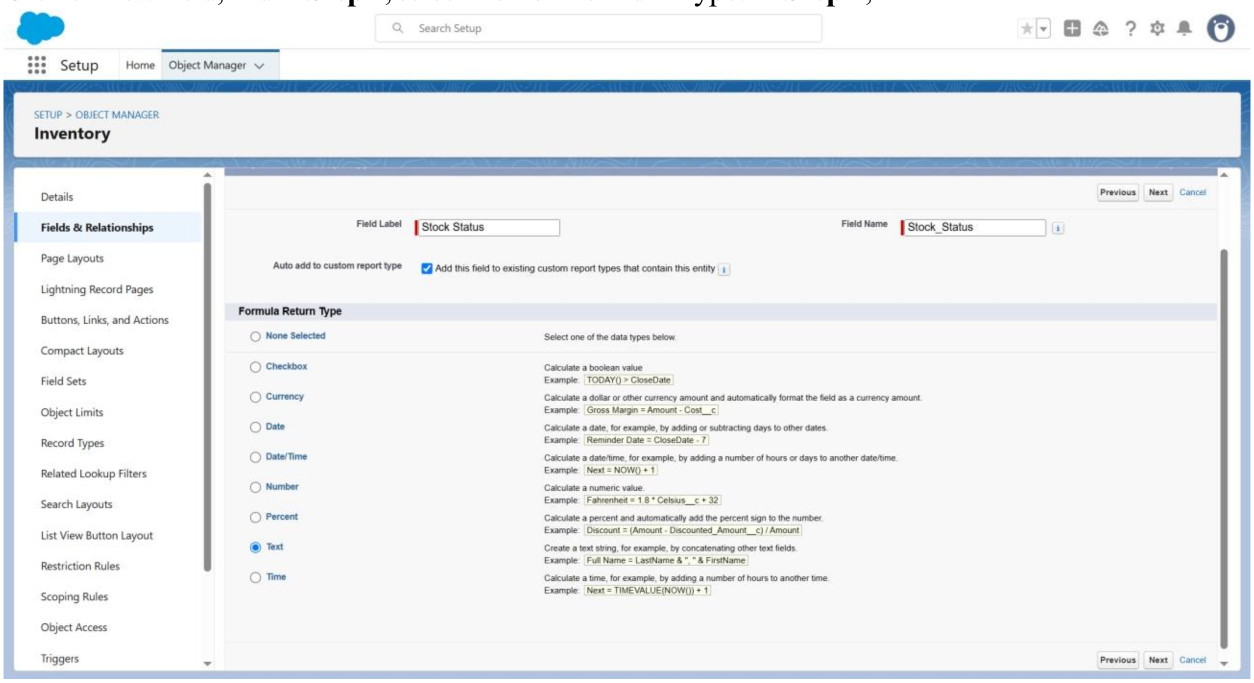
In **Step 2**, configure the field as follows:

- **Field Label:** Stock Quantity
- **Field Name:** Stock_Quantity
- **Length:** 18

Then, click **Next** through each step until the final screen, and click the **Save** button.

- **Stock Status:**

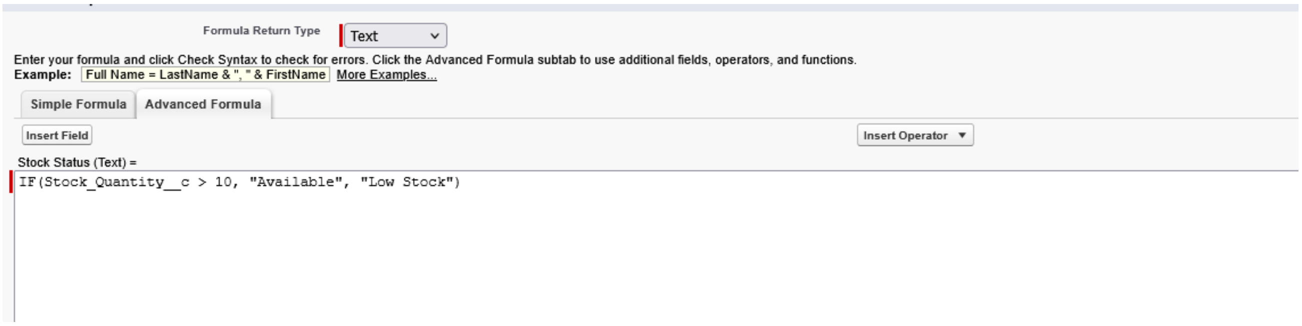
Create a new field, and in **Step 1**, select the **Formula** data type. In **Step 2**,



configure the field as follows:

- **Field Label:** Stock Status
- **Field Name:** Stock_Status
- In the **Formula Return Type** section, select **Text**

Then, click the **Next** button to proceed.



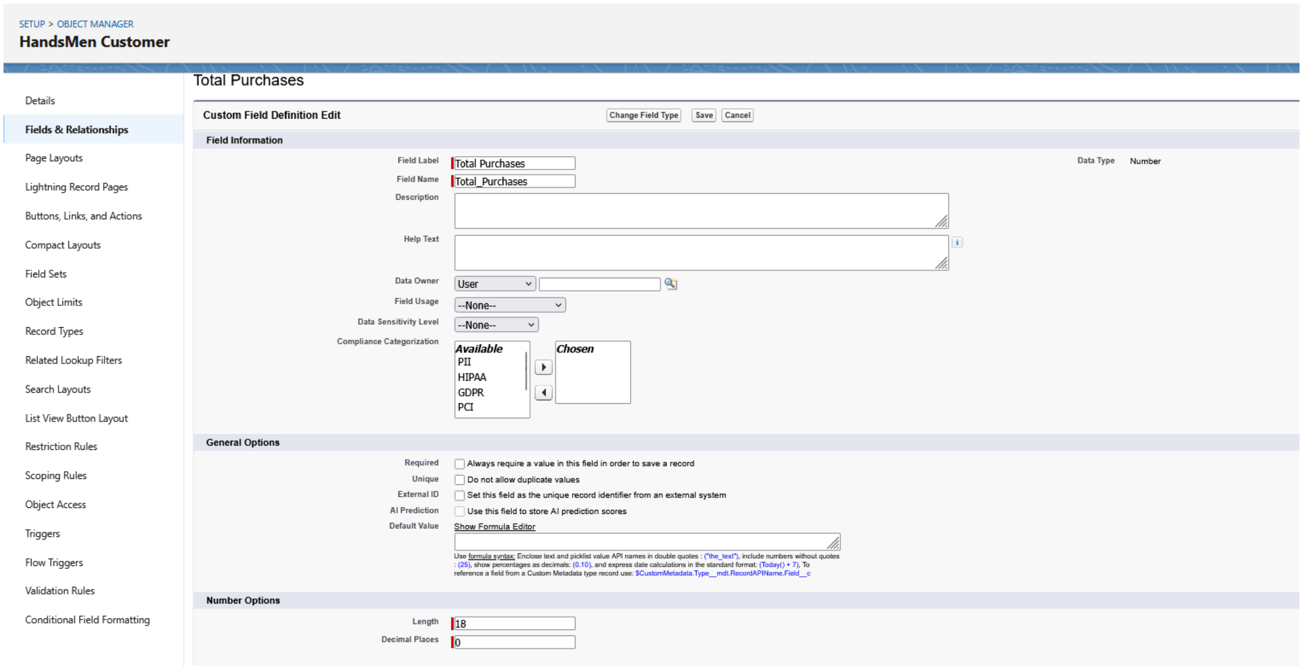
In **Step 3**, select **Advanced Formula** and enter the following formula:

- **IF(Stock_Quantity_c > 10, "Available", "Low Stock")**

Then, click **Next** through each step until the final screen, and click the **Save** button.

6. Create remaining fields for the objects

- **HandsMen Customer:**

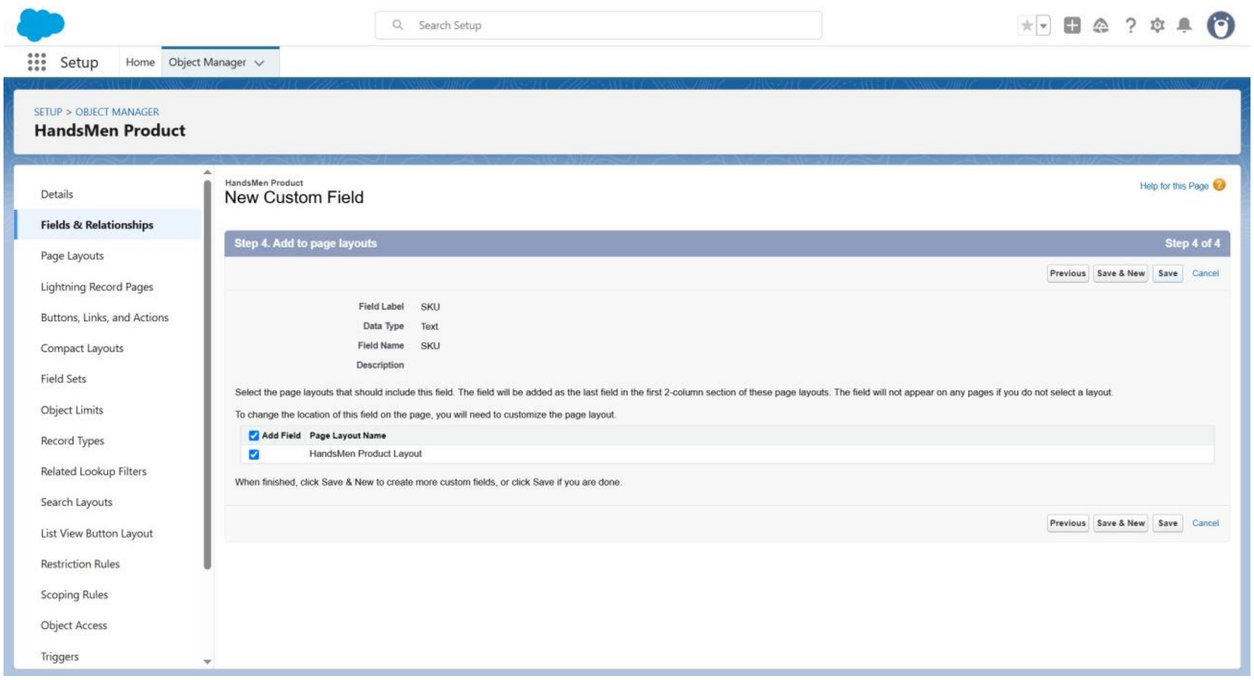


Create a new field in the **HandsMen Customer** object.

- **Step 1:** Select the **Number** data type.
- **Step 2:** Configure the field as follows:
  - **Field Label:** Total Purchases
  - **Field Name:** Total_Purchases
  - **Length:** 18

Then, proceed through the remaining steps and click the **Save** button on the final screen.

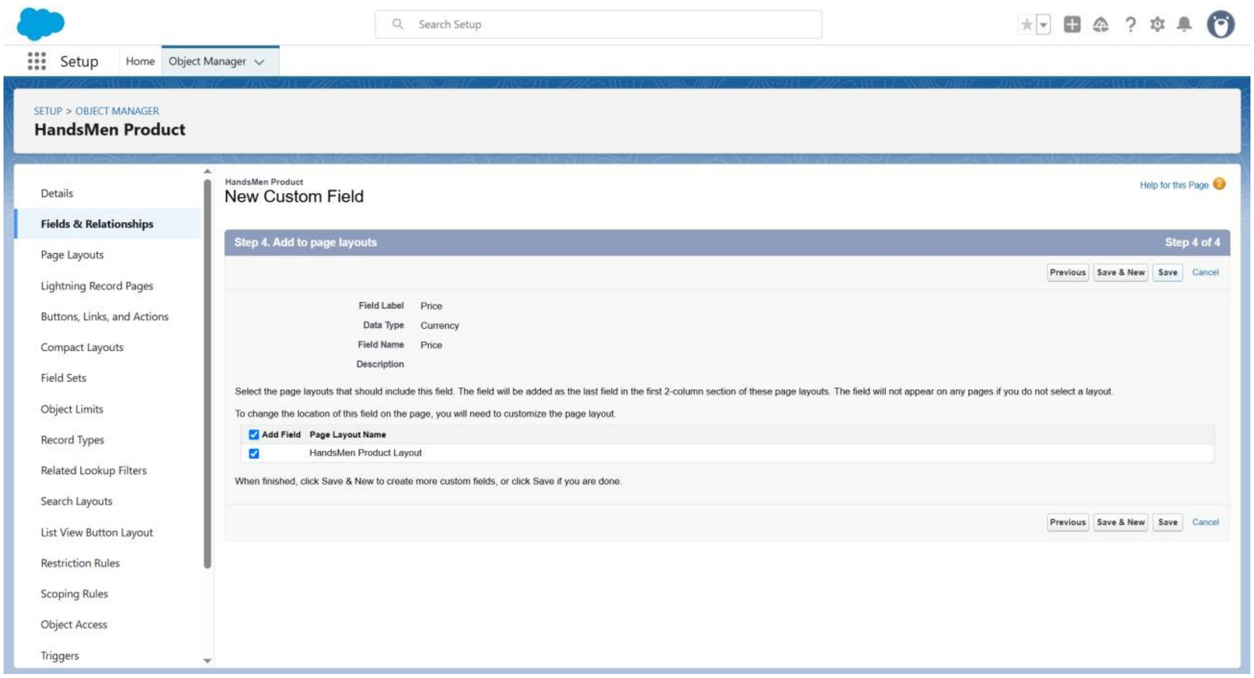
- **HandsMen Product:**



Create a new field in the **HandsMen Product** object.

- **Step 1:** Select the **Text** data type.
- **Step 2:** Configure the field as follows:
    - **Field Label:** SKU
    - **Field Name:** SKU
    - **Length:** 60

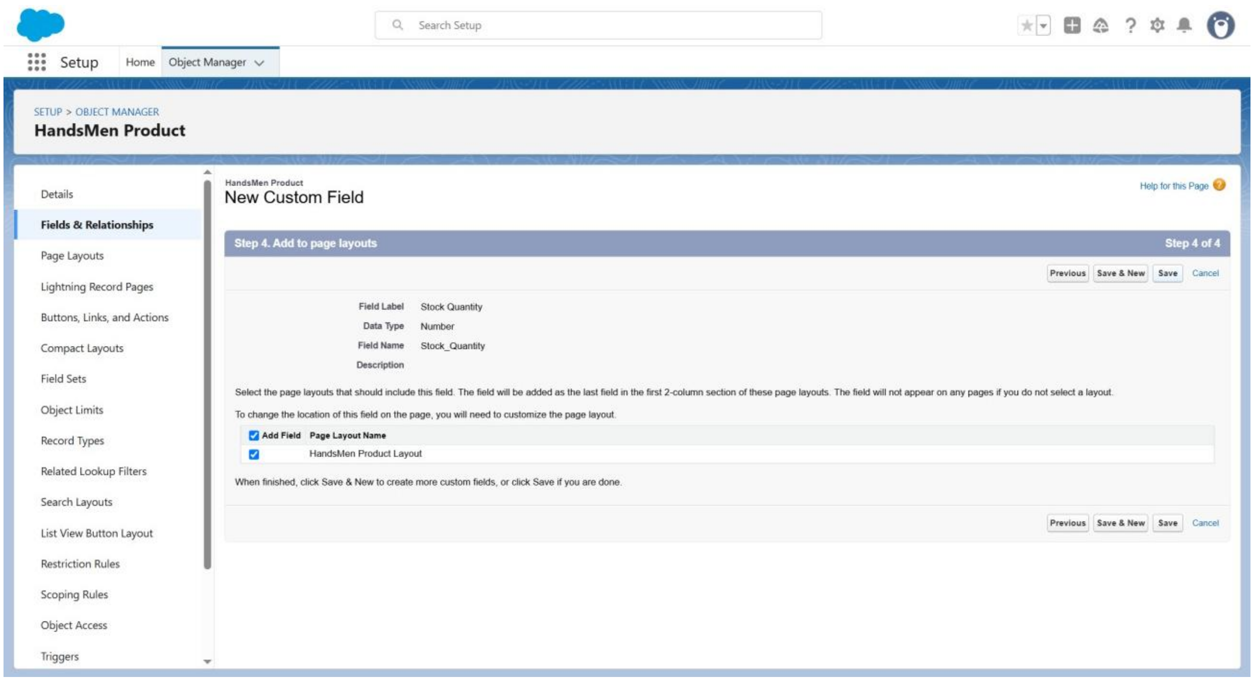
Then, proceed through the remaining steps and click the **Save** button on the final screen.



Create a new field in the **HandsMen Product** object.

- **Step 1:** Select the **Currency** data type.
- **Step 2:** Configure the field as follows:
  - **Field Label:** Price
  - **Field Name:** Price
  - **Length:** 18

Then, proceed through the remaining steps and click the **Save** button on the final screen.

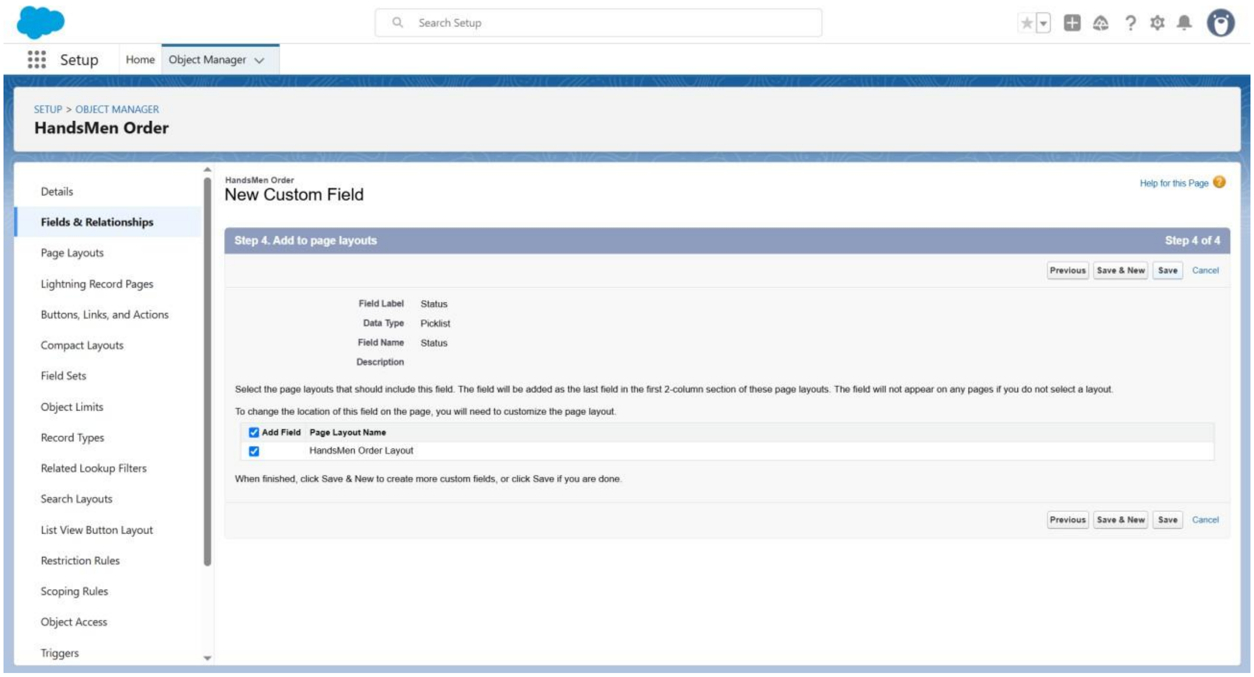


Create a new field in the **HandsMen Product** object.

- **Step 1:** Select the **Number** data type.
- **Step 2:** Configure the field as follows:
    - **Field Label:** Stock Quantity
    - **Field Name:** Stock_Quantity
    - **Length:** 18

Then, proceed through the remaining steps and click the **Save** button on the final screen.

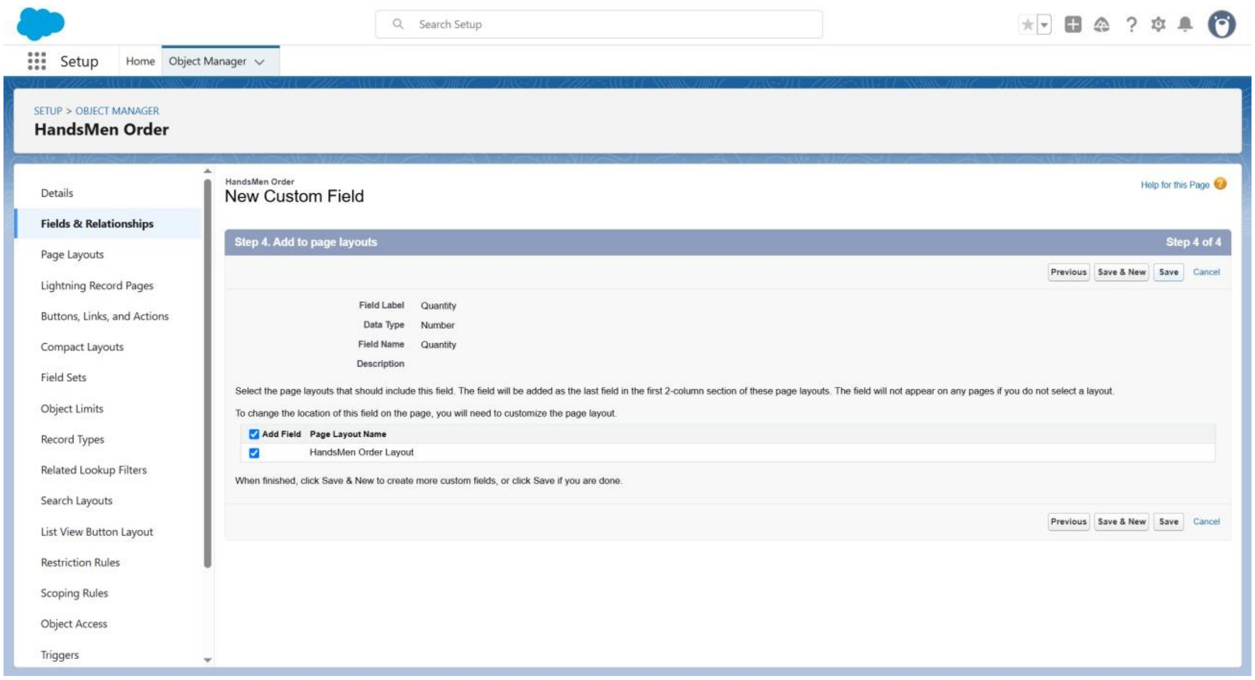
- **HandsMen Order:**



Create a new field in the **HandsMen Order** object.

- **Step 1:** Select the **Picklist** data type.
- **Step 2:** Configure the field as follows:
  - **Field Label:** Status
  - **Field Name:** Status
  - **Values:** Enter values manually, each separated by a new line:
    - **Pending**
    - **Confirmed**
    - **Rejection**

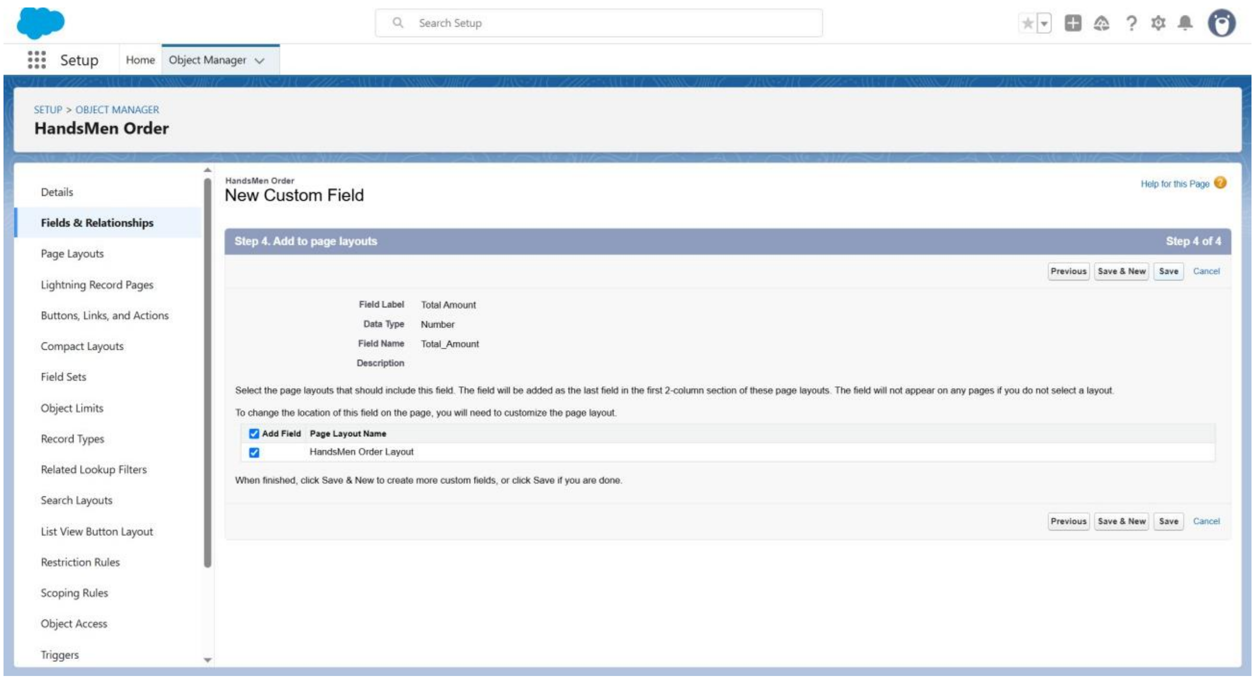
Then, proceed through the remaining steps and click the **Save** button on the final screen.



Create a new field in the **HandsMen Order** object.

- **Step 1:** Select the **Number** data type.
- **Step 2:** Configure the field as follows:
  - **Field Label:** Quantity
  - **Field Name:** Quantity
  - **Length:** 18

Then, proceed through the remaining steps and click the **Save** button on the final screen.

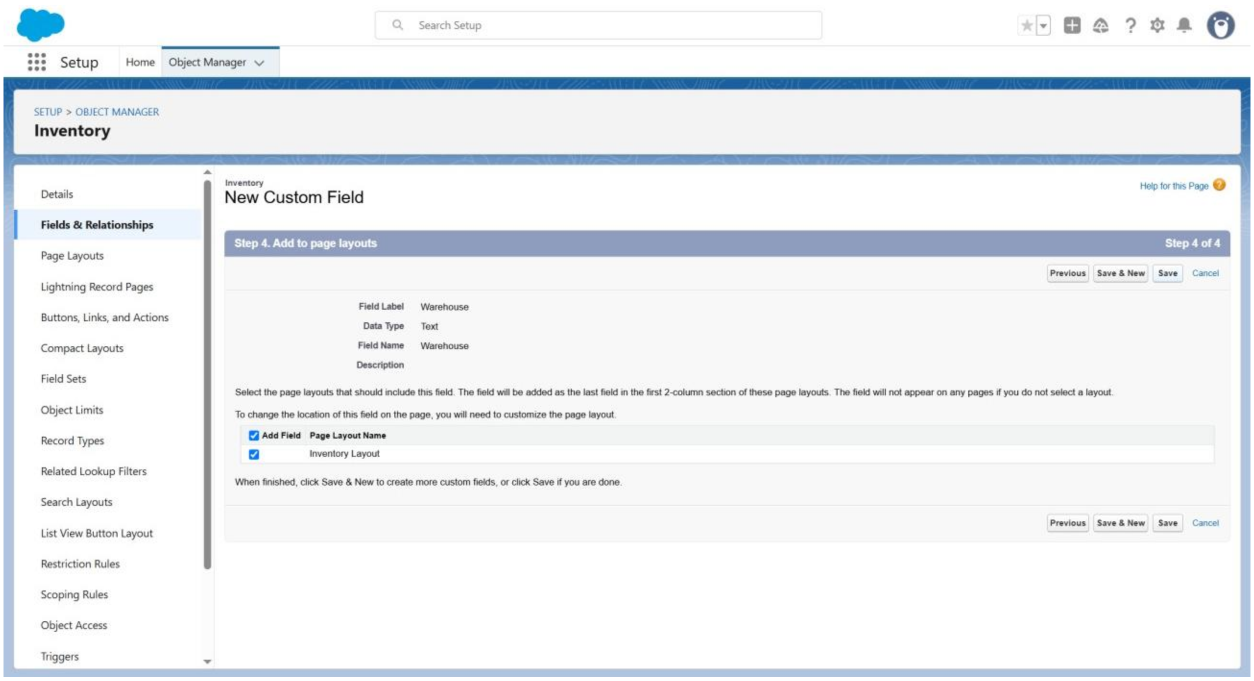


Create a new field in the **HandsMen Order** object.

- **Step 1:** Select the **Number** data type.
- **Step 2:** Configure the field as follows:
    - **Field Label:** Total Amount
    - **Field Name:** Total_Amount
    - **Length:** 18

Then, proceed through the remaining steps and click the **Save** button on the final screen.

- **Inventory:**

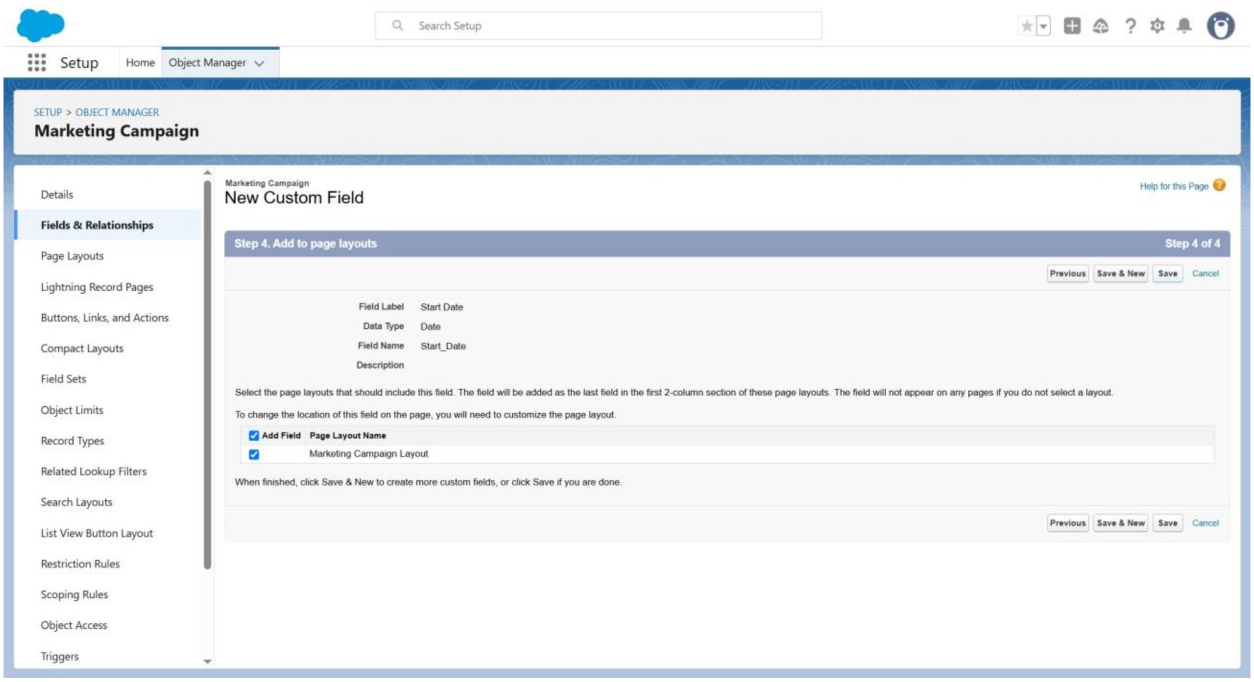


Create a new field in the **Inventory** object.

- **Step 1:** Select the **Text** data type.
- **Step 2:** Configure the field as follows:
  - **Field Label:** Warehouse
  - **Field Name:** Warehouse
  - **Length:** 60

Then, proceed through the remaining steps and click the **Save** button on the final screen.

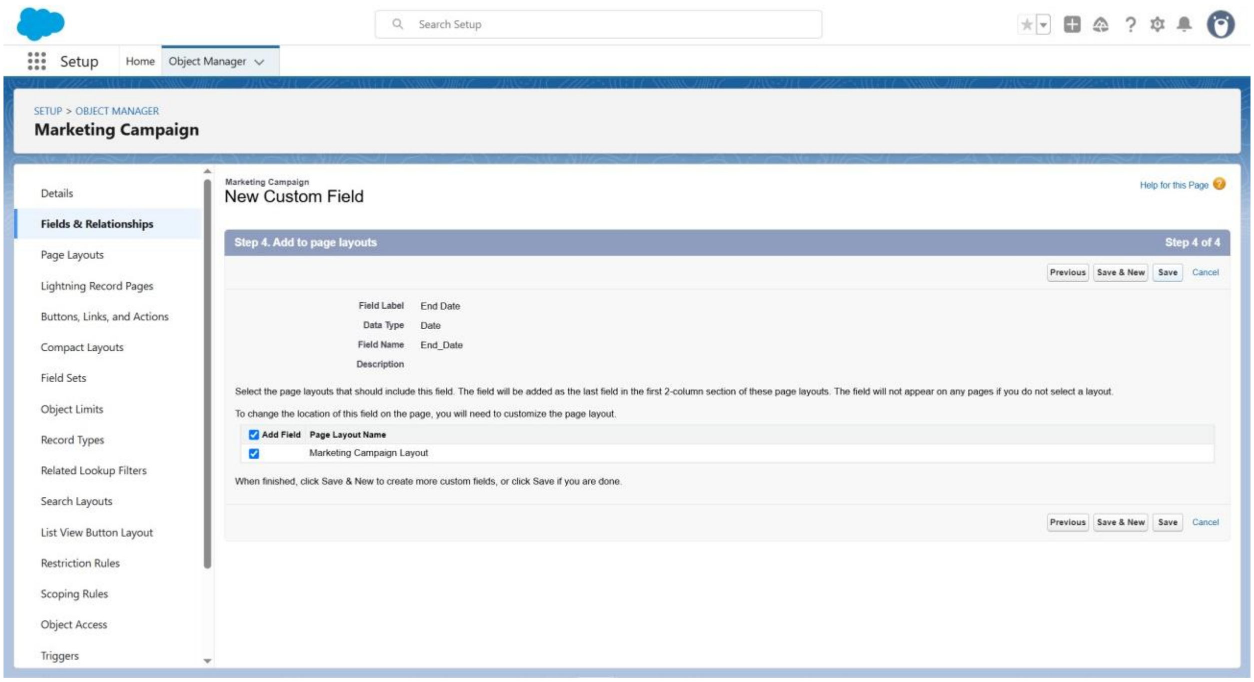
- **Marketing Campaign:**



Create a new field in the **Market Campaign** object.

- **Step 1:** Select the **Date** data type.
- **Step 2:** Configure the field as follows:
    - **Field Label:** Start Date
    - **Field Name:** Start_Date

Then, proceed through the remaining steps and click the **Save** button on the final screen.



Create a new field in the **Market Campaign** object.

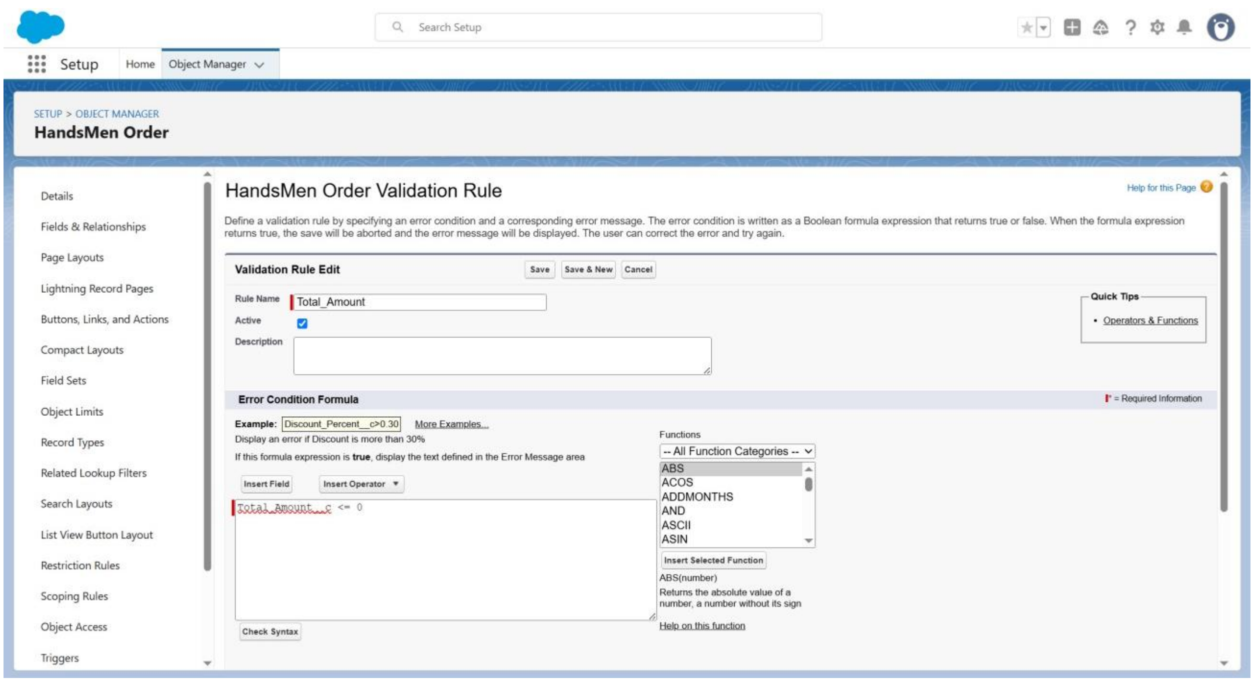
- **Step 1:** Select the **Date** data type.
- **Step 2:** Configure the field as follows:
  - **Field Label:** End Date
  - **Field Name:** End_Date

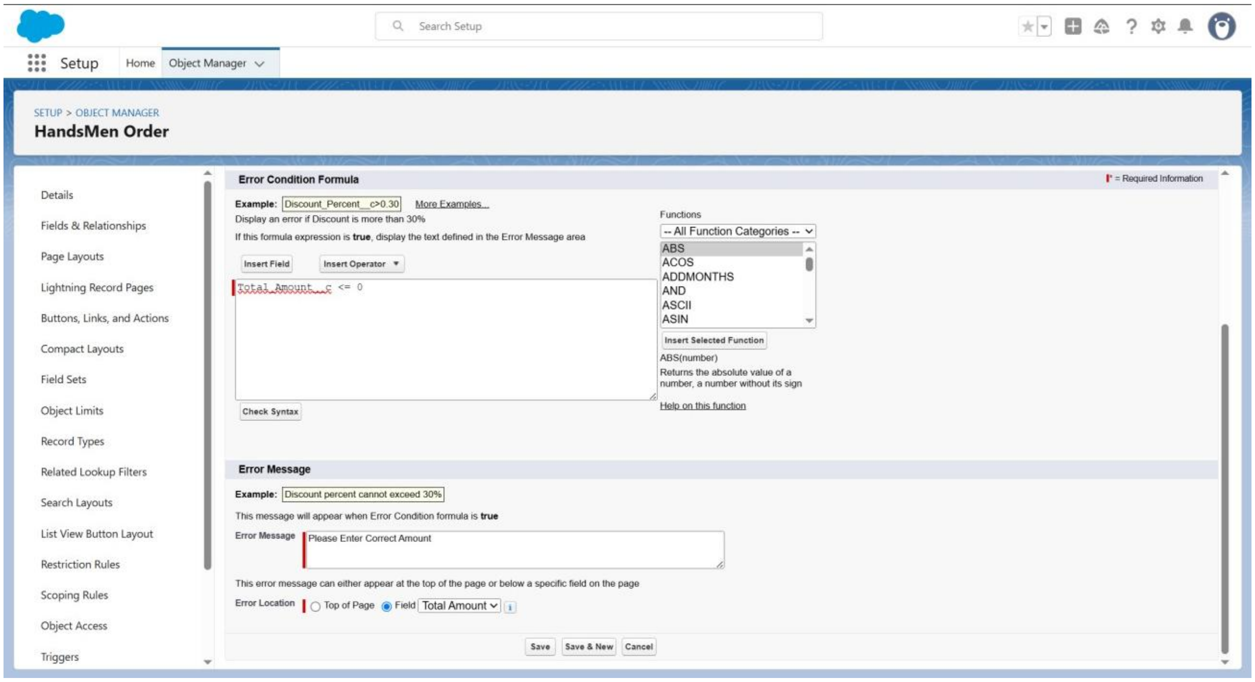
Then, proceed through the remaining steps and click the **Save** button on the final screen.

Data Configuration

Validation Rules:

- **HandsMen Order:**



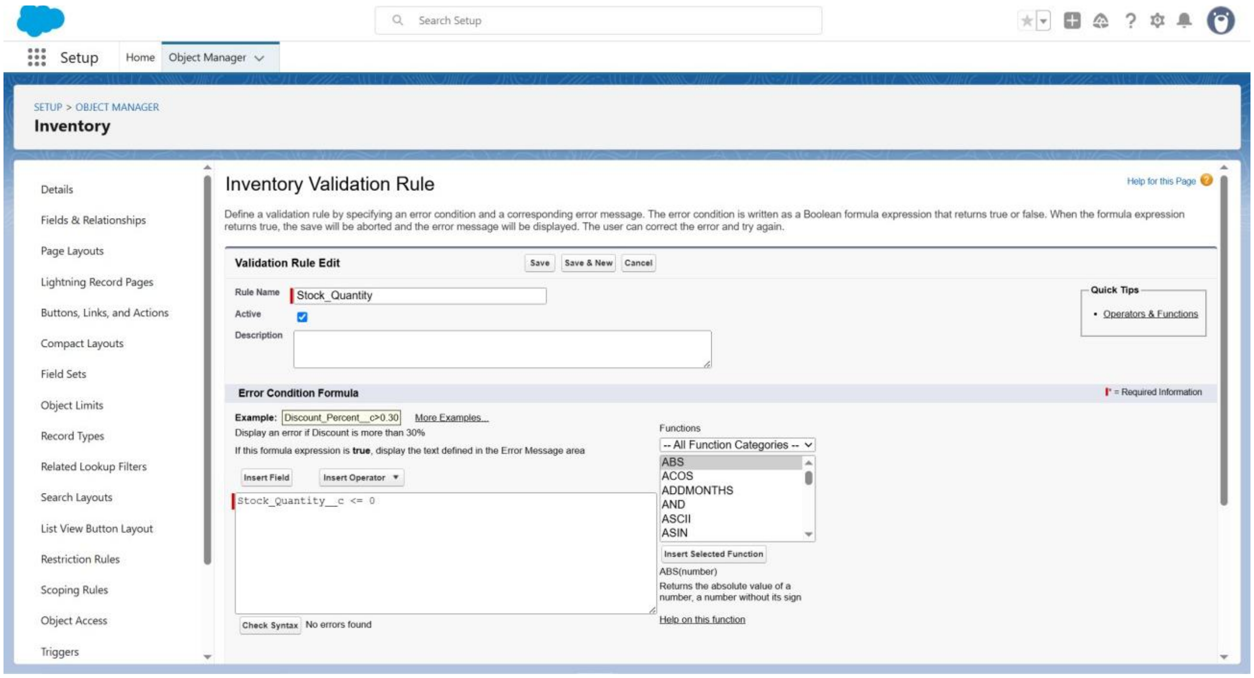


In the **HandsMen Order** object settings:

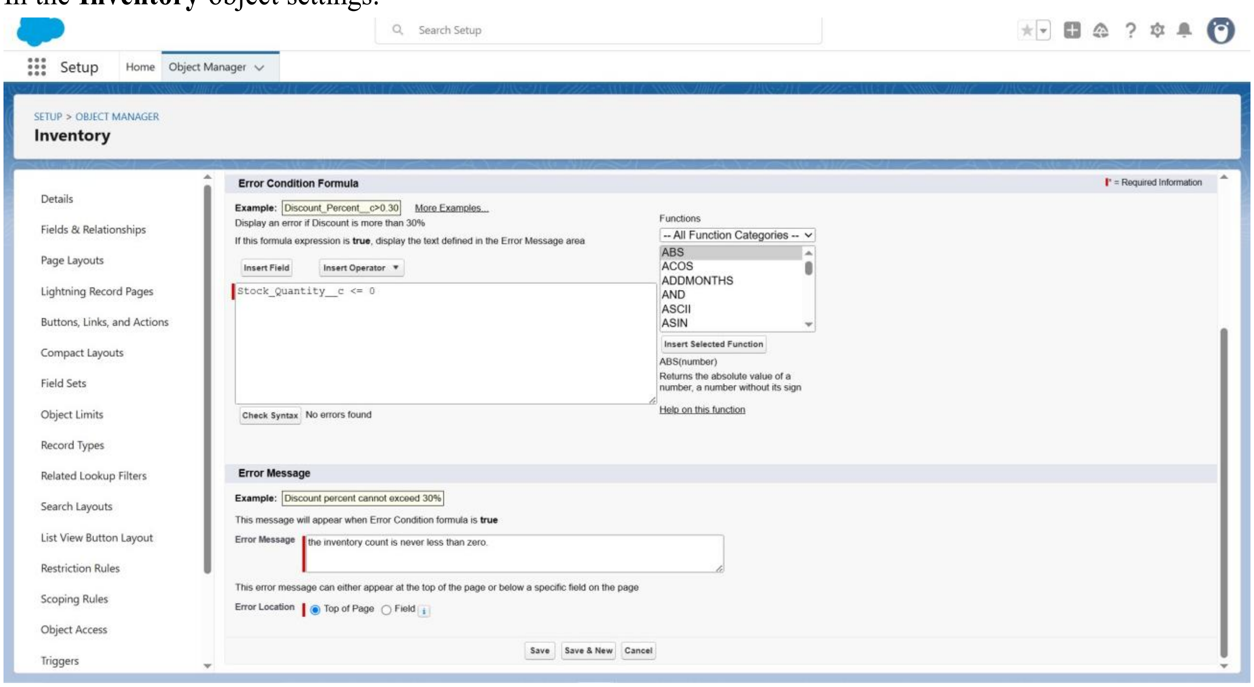
1. Click **Validation Rules**.
2. Click the **New** button.
3. Configure the rule as follows:
   - **Rule Name:** Total_Amount
   - **Error Condition Formula:**
     - **Total_Amount _ c<=0**
   - **Error Message:** Please Enter Correct Amount
   - **Error Location:**
     - Select **Field**
     - In **Field**, choose **Total Amount**

Click the **Save** button to activate the rule.

- **Inventory:**



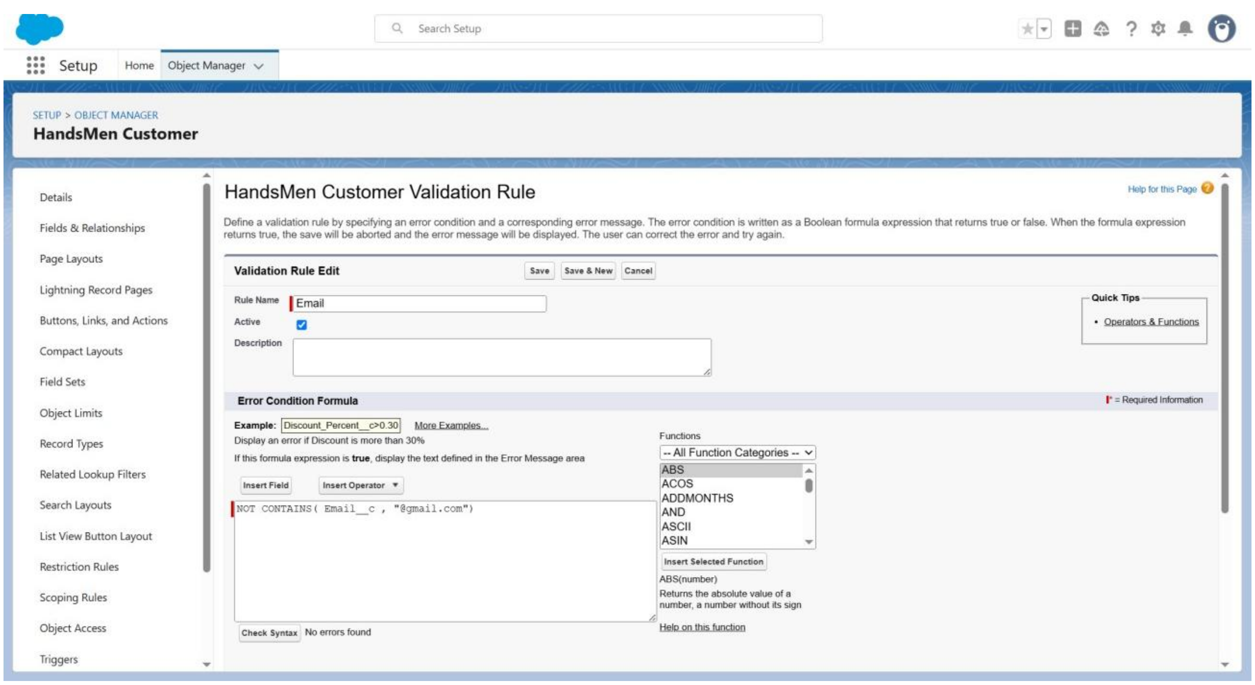
In the **Inventory** object settings:

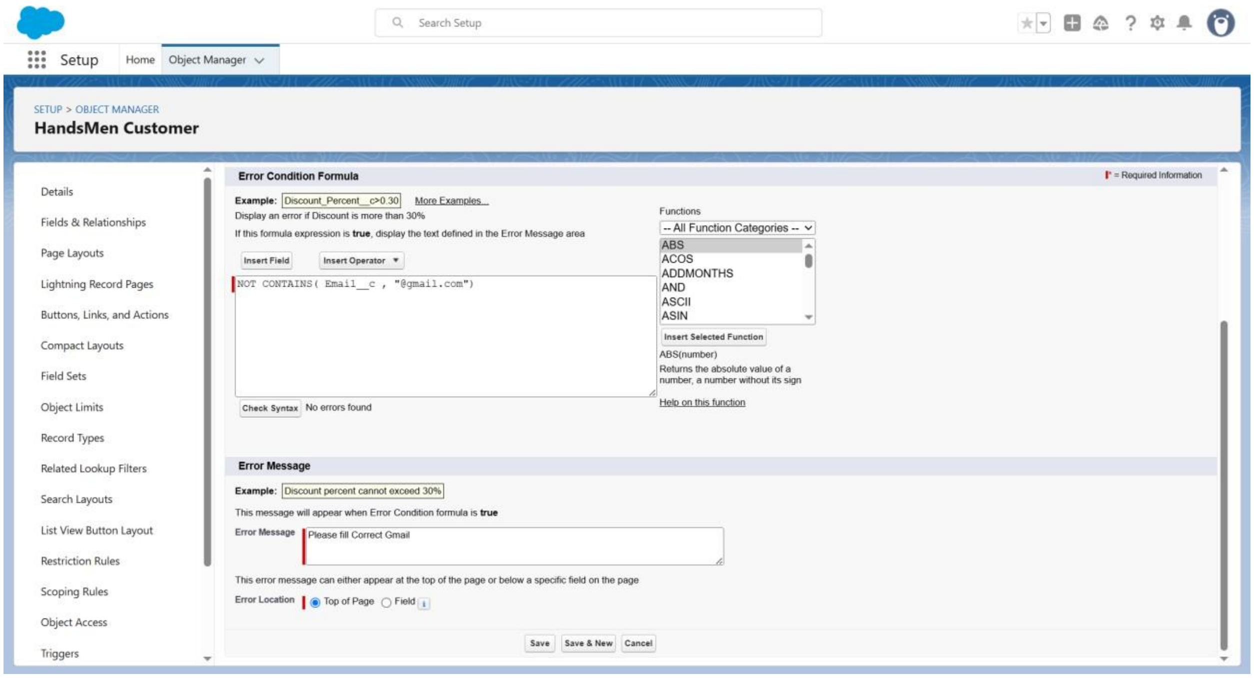


1. Click **Validation Rules**.
2. Click the **New** button.
3. Configure the rule as follows:
    - **Rule Name:** Stock_Quantity
    - **Error Condition Formula:**
        - **Stock_Quantity _ c<=0**
    - **Error Message:** The inventory count is never less than zero
    - **Error Location:** Select Top of Page

Click the **Save** button to activate the rule.

- **HandsMen Customer:**





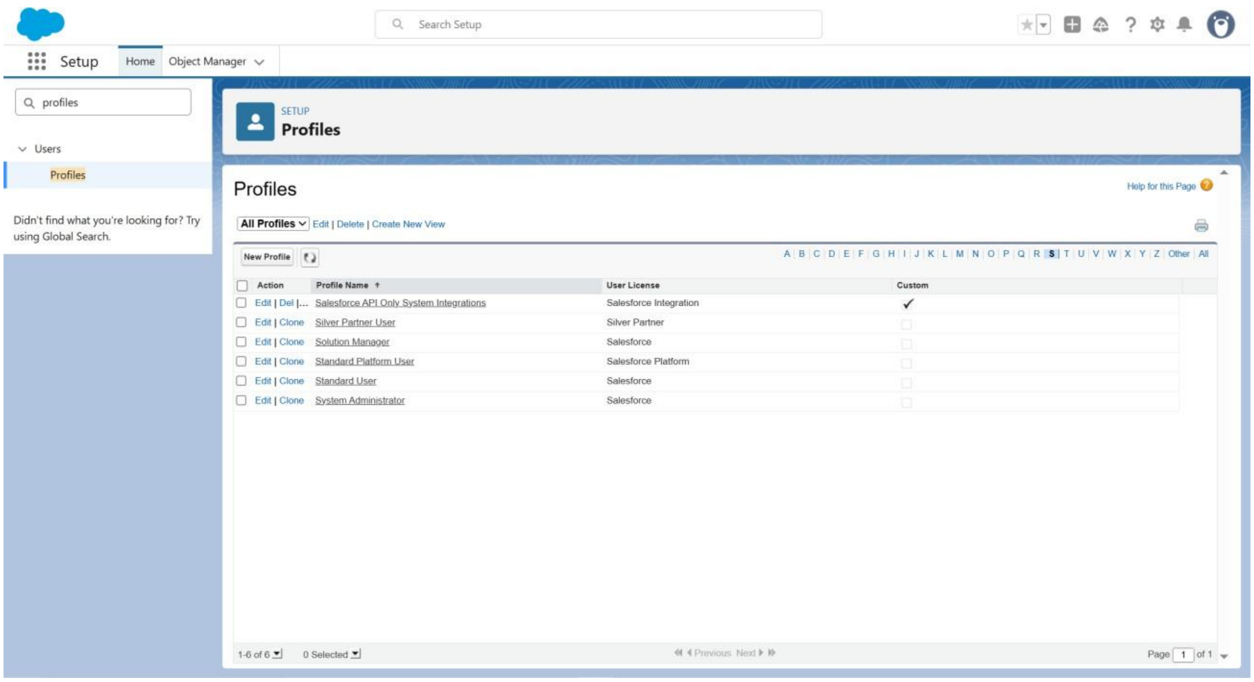
In the **HandsMen Customer** object settings:

1. Click **Validation Rules**.
2. Click the **New** button.
3. Configure the rule as follows:
   - **Rule Name:** Email
   - **Error Condition Formula:**
     - **NOT(CONTAINS(Email__c, "@gmail.com"))**
   - **Error Message:** Please fill correct Gmail
   - **Error Location:** Select Top of Page

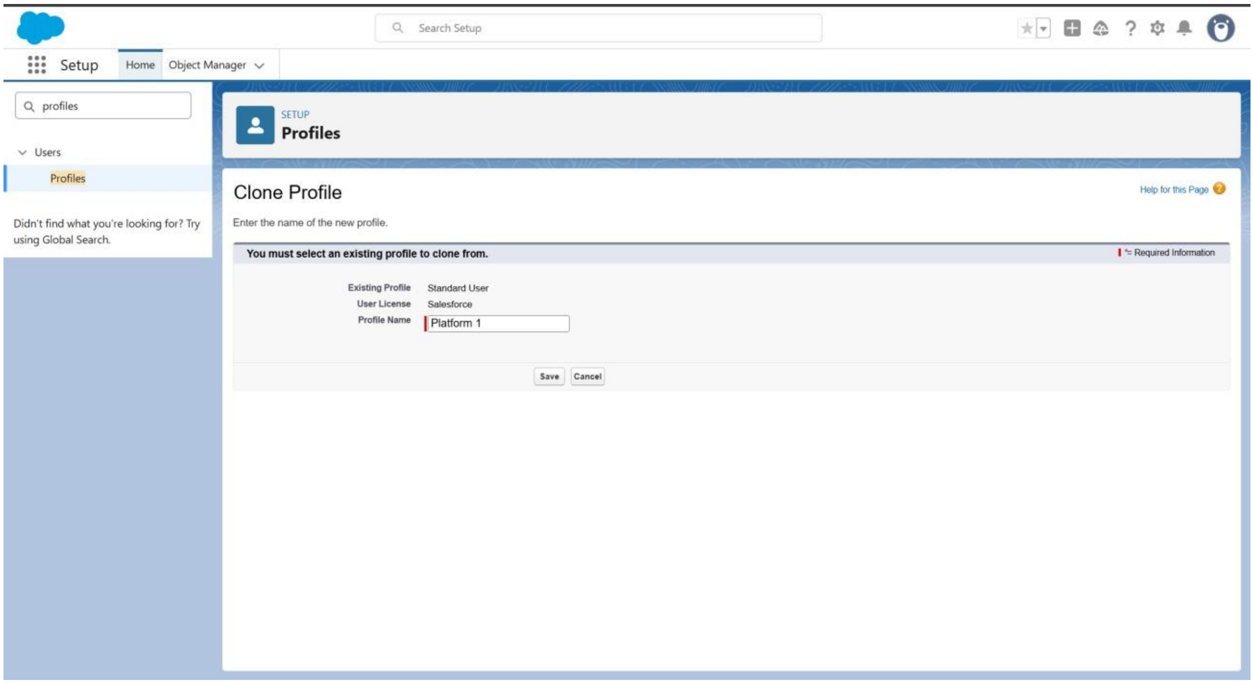
Click the **Save** button to activate the rule.

Data Security

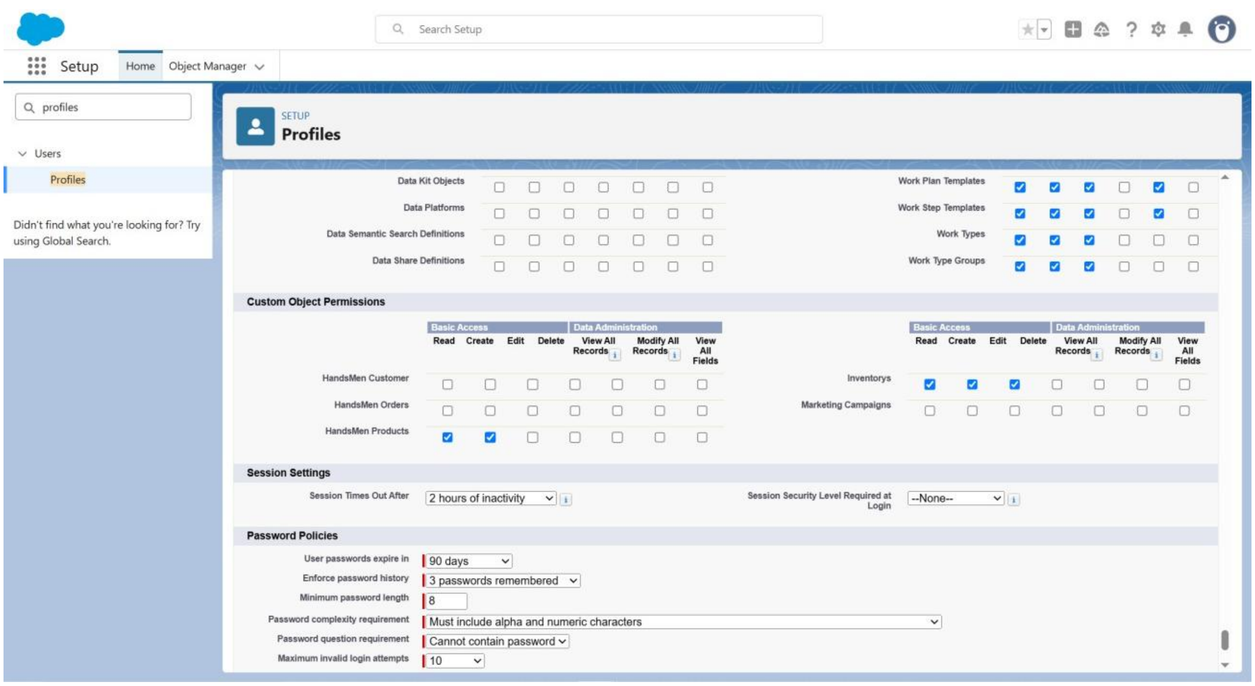
Profiles:



In **Home**, use **Quick Find** to search for **Profiles**. Locate and select the **Standard User** profile, then click the **Clone** button to create a duplicate profile.



In the **Platform Name** field, enter **Platform 1.** Then, click the **Save** button to apply the changes.

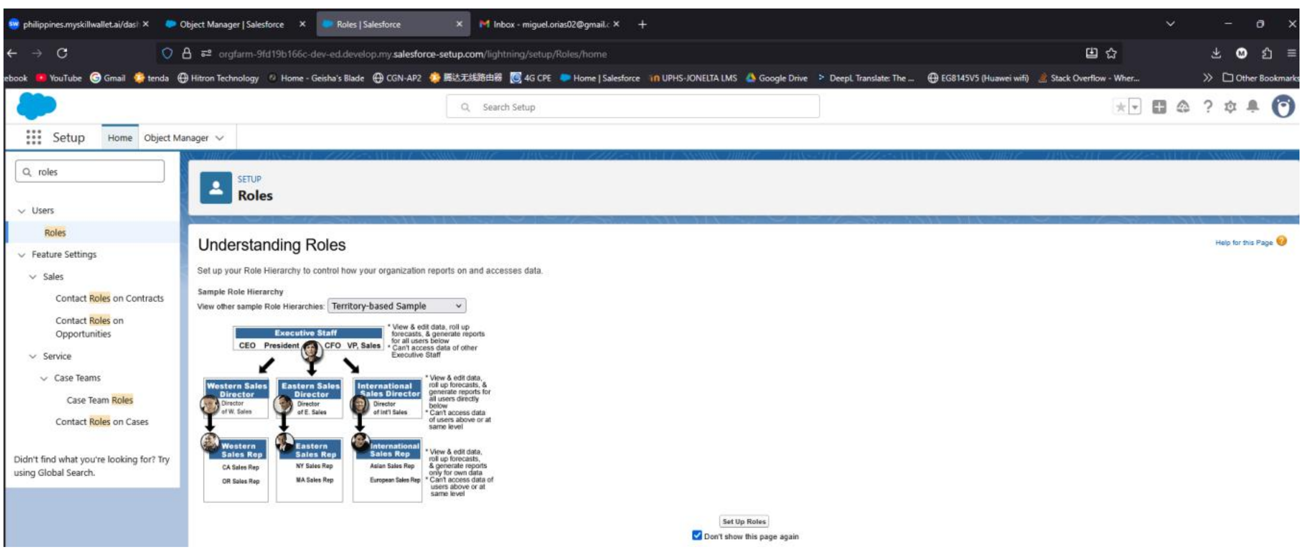


Edit the **Platform 1** profile and navigate to the **Custom Object Permissions** section.

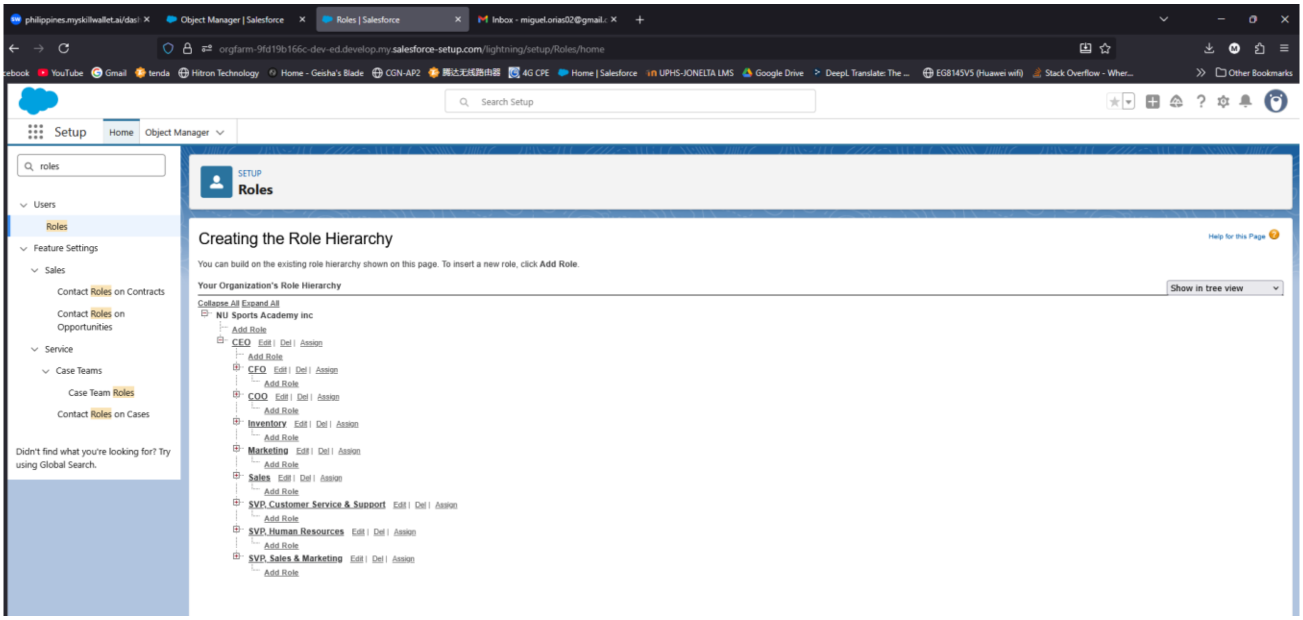
- For **HandsMen Products**, select:
    - **Read**
    - **Create**
- For **Inventorys**, select:
    - **Read**
    - **Create**
    - **Edit**

Then, click **Save** to apply the updated permissions.

Roles:

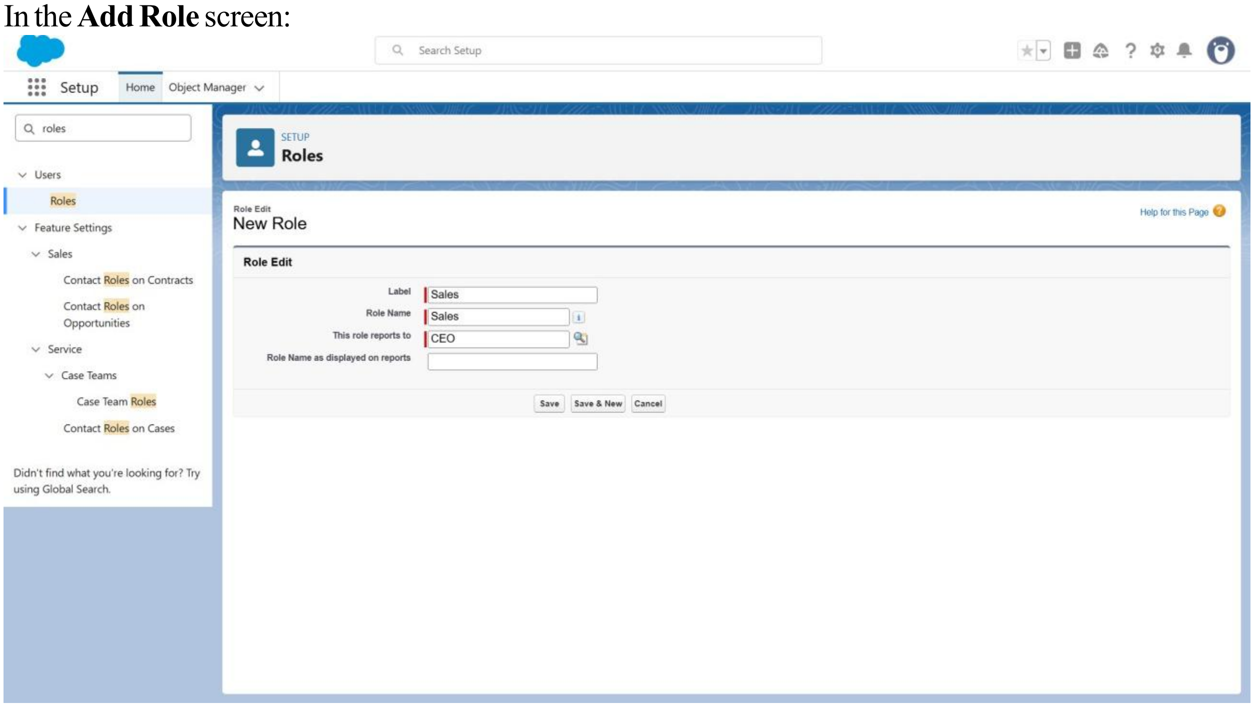


In **Home**, use **Quick Find** to search for **Roles**, then click the **Set Up Roles** button on the Roles setup page.



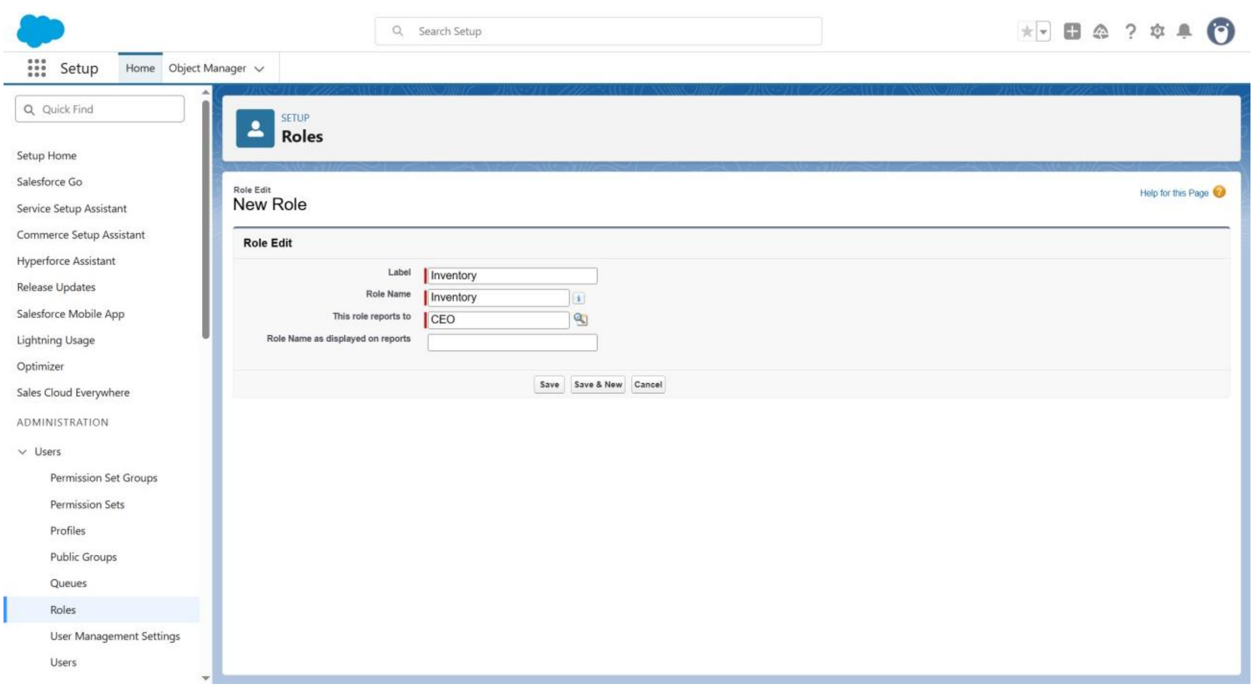
Click **Expand All** in the role hierarchy, then click **Add Role** under the **CEO** node.

In the **Add Role** screen:



- **Label:** Sales
- **Role Name:** Sales
- **This role reports to:** CEO

Then, click the **Save** button to create the role.

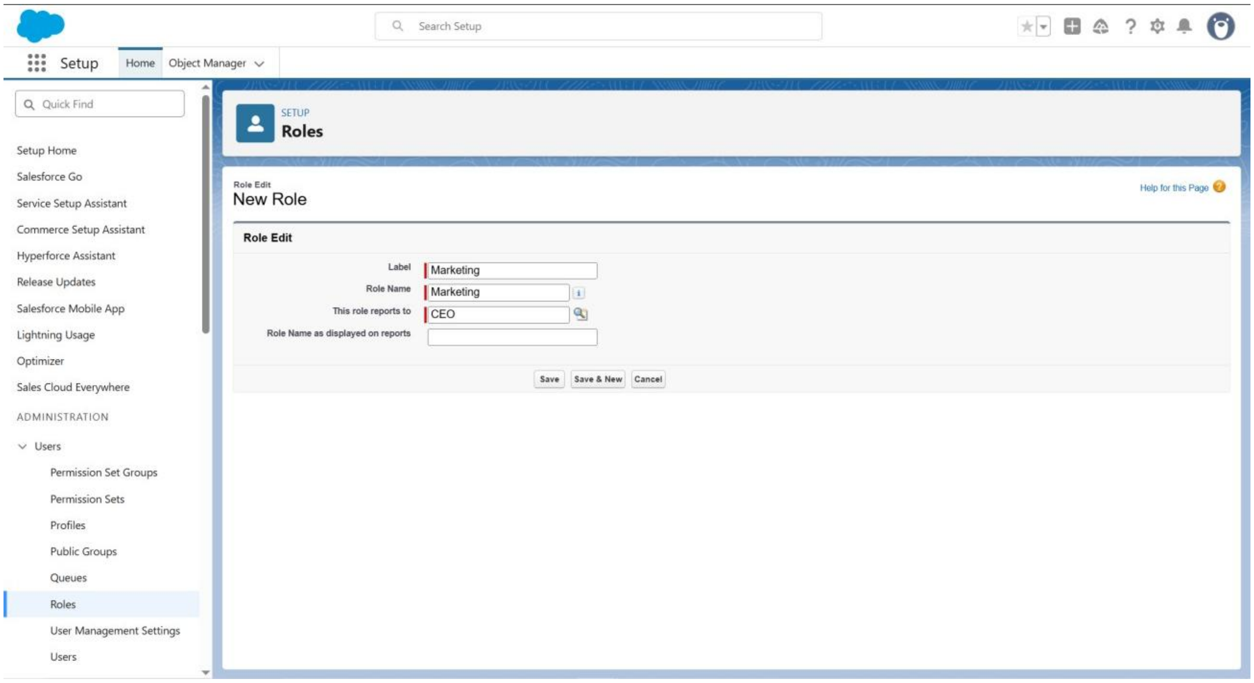


Add another role under the **CEO** node:

- In the **Add Role** screen, enter the following:
    - **Label:** Inventory
    - **Role Name:** Inventory
        - **This role**

  reports to: CEO Click the

  Save button to create the role.

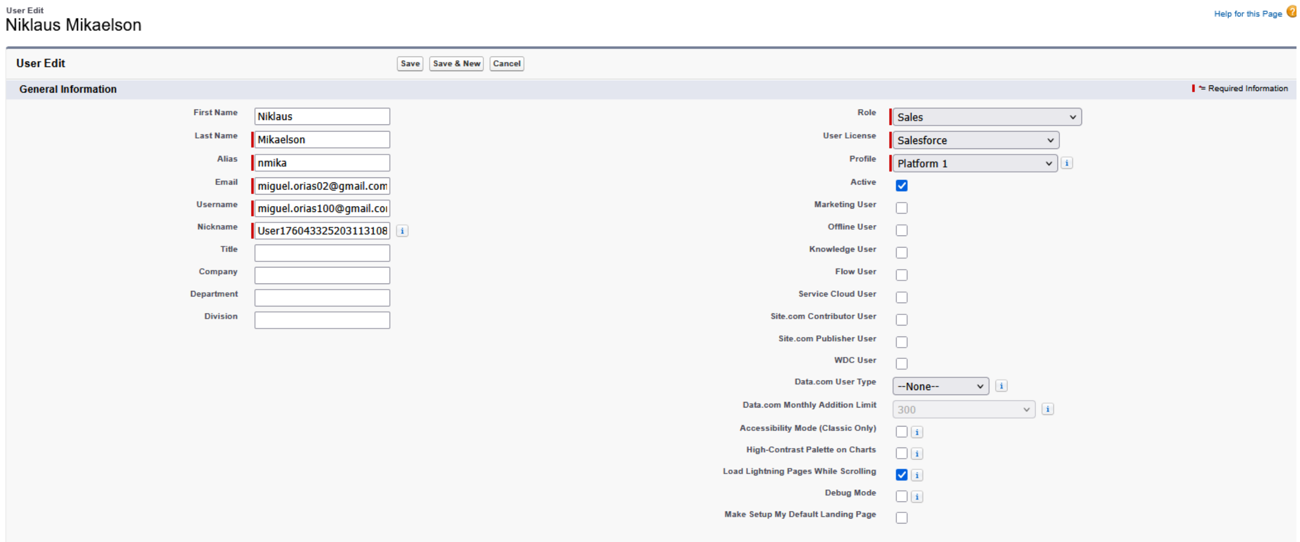


Add another role under the **CEO** node:

- In the **Add Role** screen, enter the following:
    - **Label:** Marketing
    - **Role Name:** Marketing
    - **This role reports to:** CEO Click the **Save** button to create the role.

Users:

- **Niklaus:**

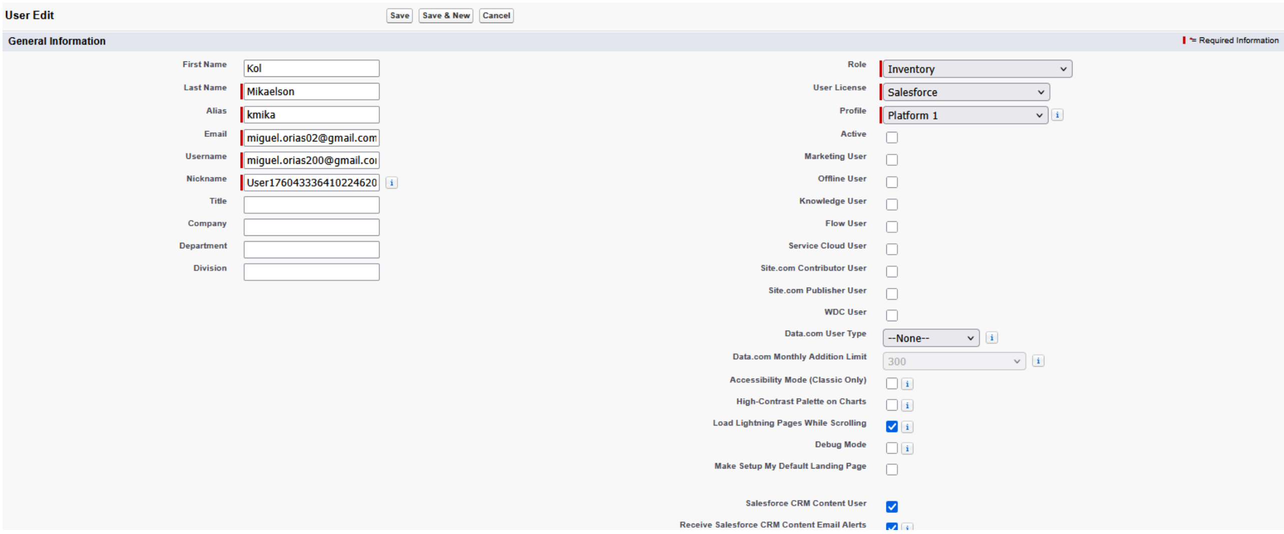


In **Home**, use **Quick Find** to search for **Users**.

1. Click the **New** User button.
2. On the Add User page, enter the following details:
   - **First Name:** Niklaus
   - **Last Name:** Mikaelson
   - **Alias:** [your preferred alias]
   - **Email:** [your preferred email]
   - **Username:** [your preferred username]
   - **Nickname:** [your preferred nickname]
   - **Role:** Sales
   - **User License:** Salesforce
   - **Profile:** Platform 1

Click the **Save** button to create the user.

- **Kol:**

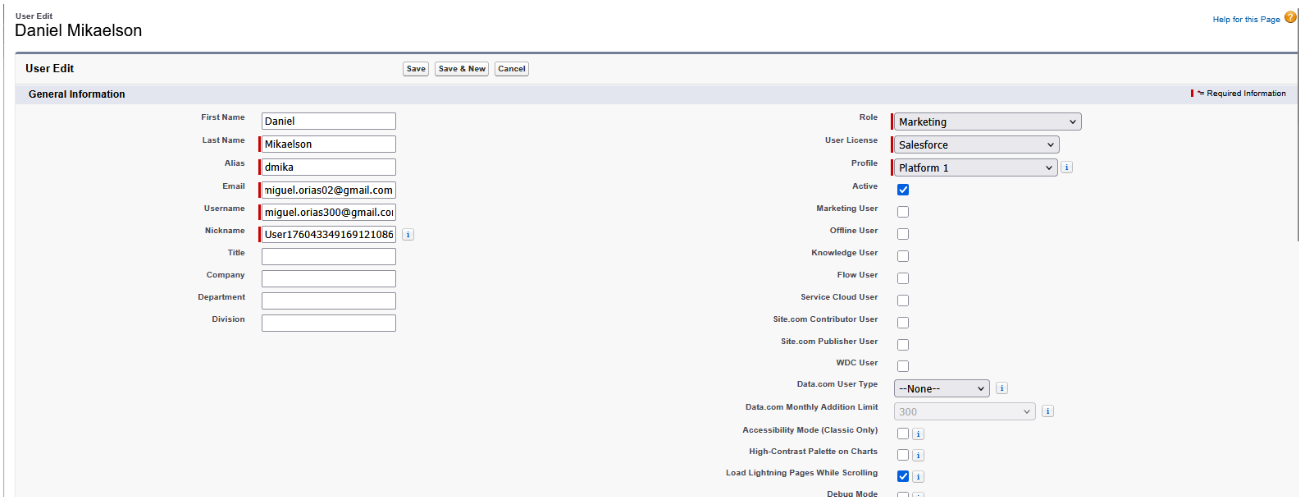


In **Home**, use **Quick Find** to search for **Users**.

1. Click the **New** User button.
2. On the Add User page, enter the following details:
   - **First Name:** Kol
   - **Last Name:** Mikaelson
   - **Alias:** [your preferred alias]
   - **Email:** [your preferred email]
   - **Username:** [your preferred username]
   - **Nickname:** [your preferred nickname]
   - **Role:** Inventory
   - **User License:** Salesforce
   - **Profile:** Platform 1

Click the **Save** button to create the user.

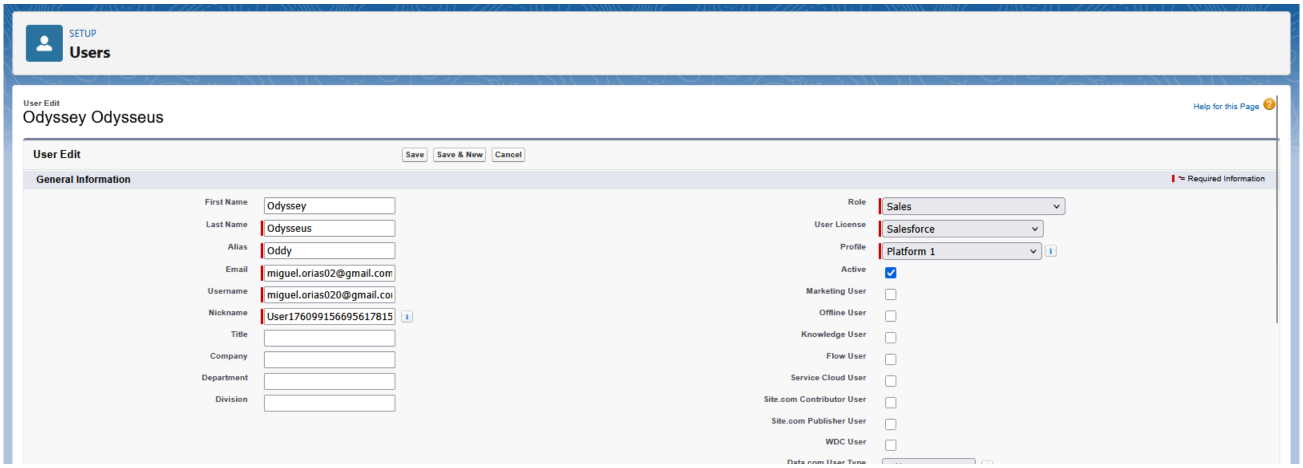
- **Create 2 more user:**



In **Home**, use **Quick Find** to search for **Users**.

1. Click the **New** User button.
2. On the Add User page, enter the following details:
    - **First Name:** [your preferred first name]
    - **Last Name:** [your preferred last name]
    - **Alias:** [your preferred alias]
    - **Email:** [your preferred email]
    - **Username:** [your preferred username]
    - **Nickname:** [your preferred nickname]
    - **Role:** Marketing
    - **User License:** Salesforce
    - **Profile:** Platform 1

Click the **Save** button to create the user.



In **Home**, use **Quick Find** to search for **Users**.

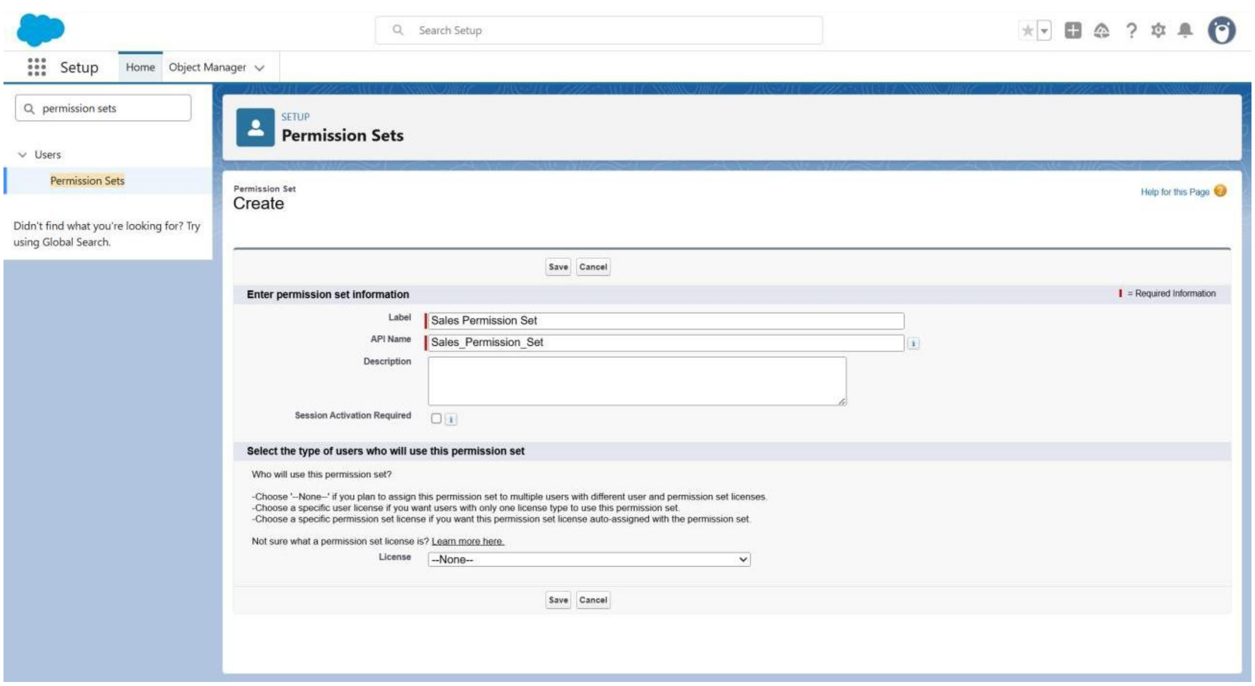
1. Click the **New** User button.
2. On the Add User page, enter the following details:
   - **First Name:** [your preferred first name]
   - **Last Name:** [your preferred last name]
   - **Alias:** [your preferred alias]
   - **Email:** [your preferred email]
   - **Username:** [your preferred username]
   - **Nickname:** [your preferred nickname]
   - **Role:** Sales
   - **User License:** Salesforce
   - **Profile:** Platform 1

Click the **Save** button to create the user.

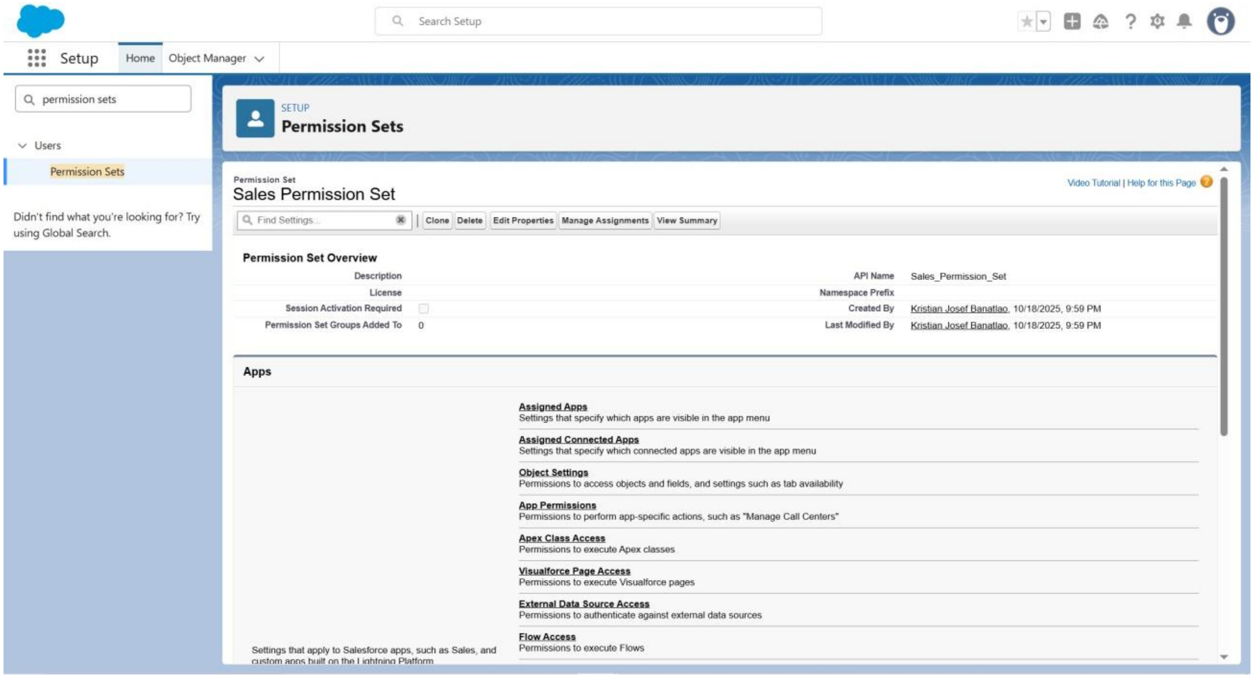
Permission Set:

- **Sales Permission Set:**

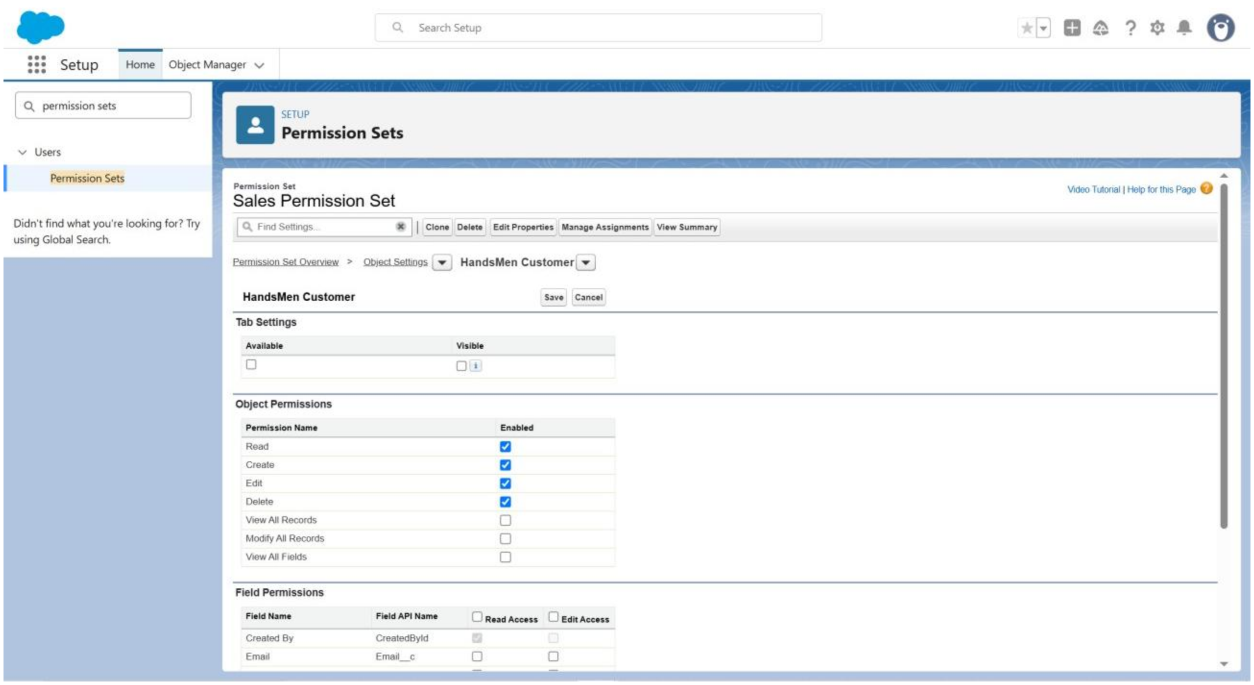
In **Home**, use **Quick Find** to search for **Permission Sets**, then click the **New** button to create a new permission set.



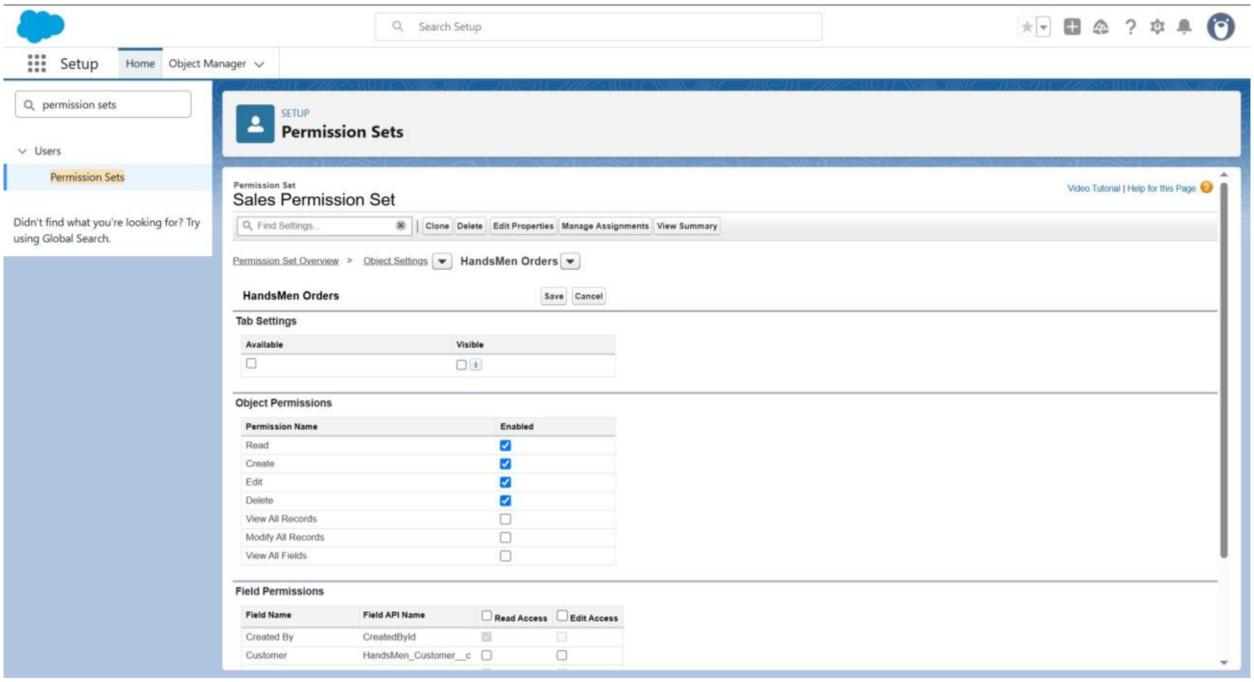
In the **Add Permission Set** screen, enter **Label: Sales Permission** Set and **API Name: Sales_Permission_Set**, then click the Save button.



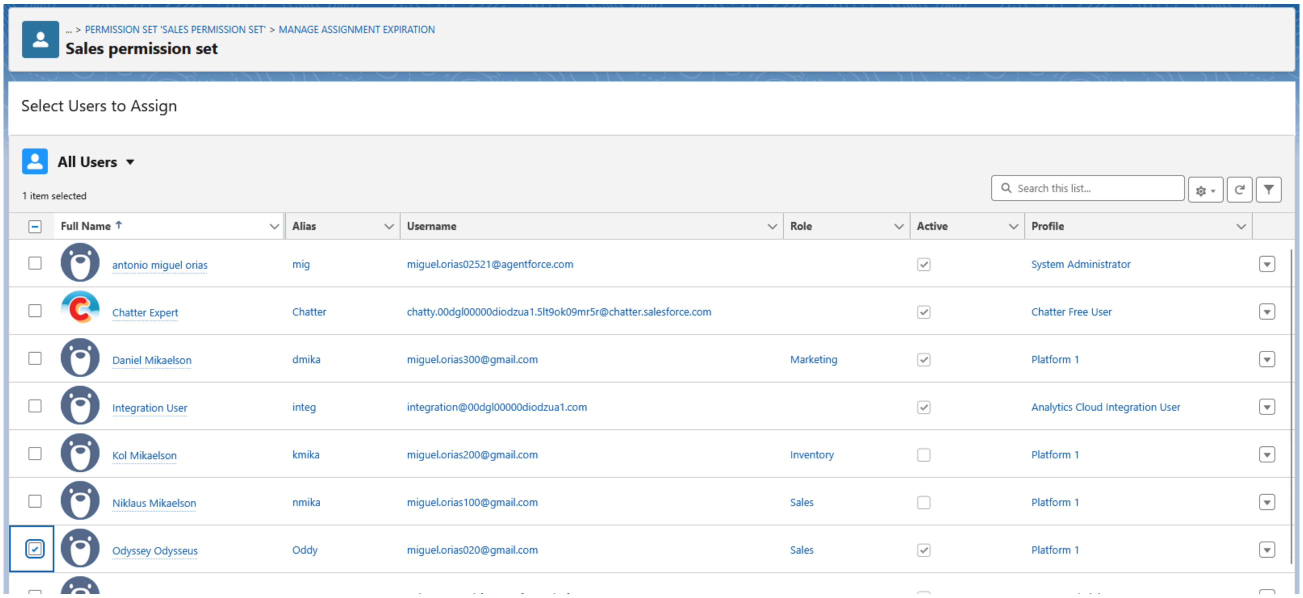
In the **Sales Permission Set**, go to **Find Settings**, search for **HandsMen Customer**, then click the **Edit** button to modify its permissions.

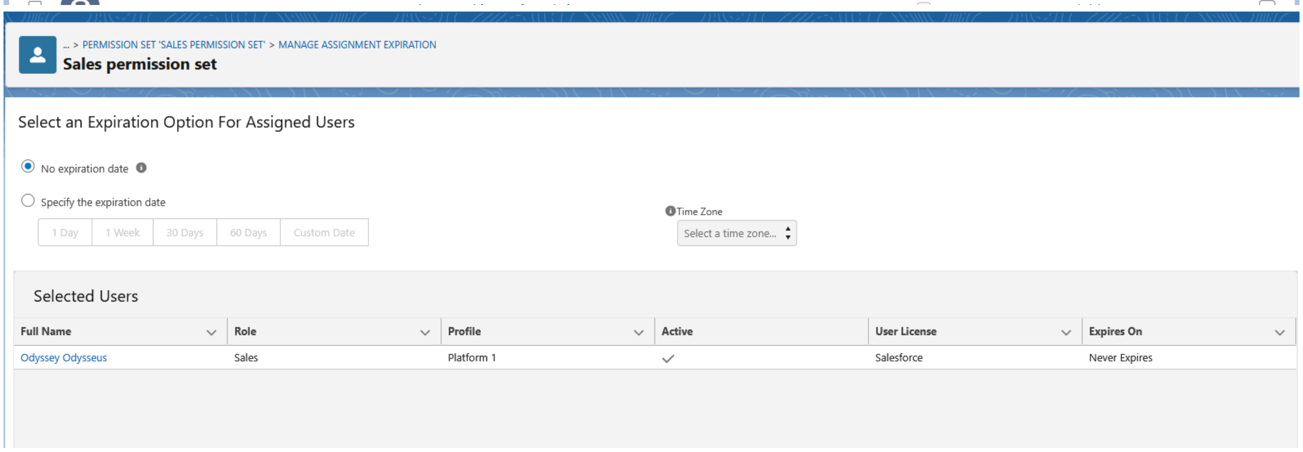


In the **Object Permissions** section, enable **Read**, **Create**, **Edit**, and **Delete**, then click the **Save** button to apply the changes.



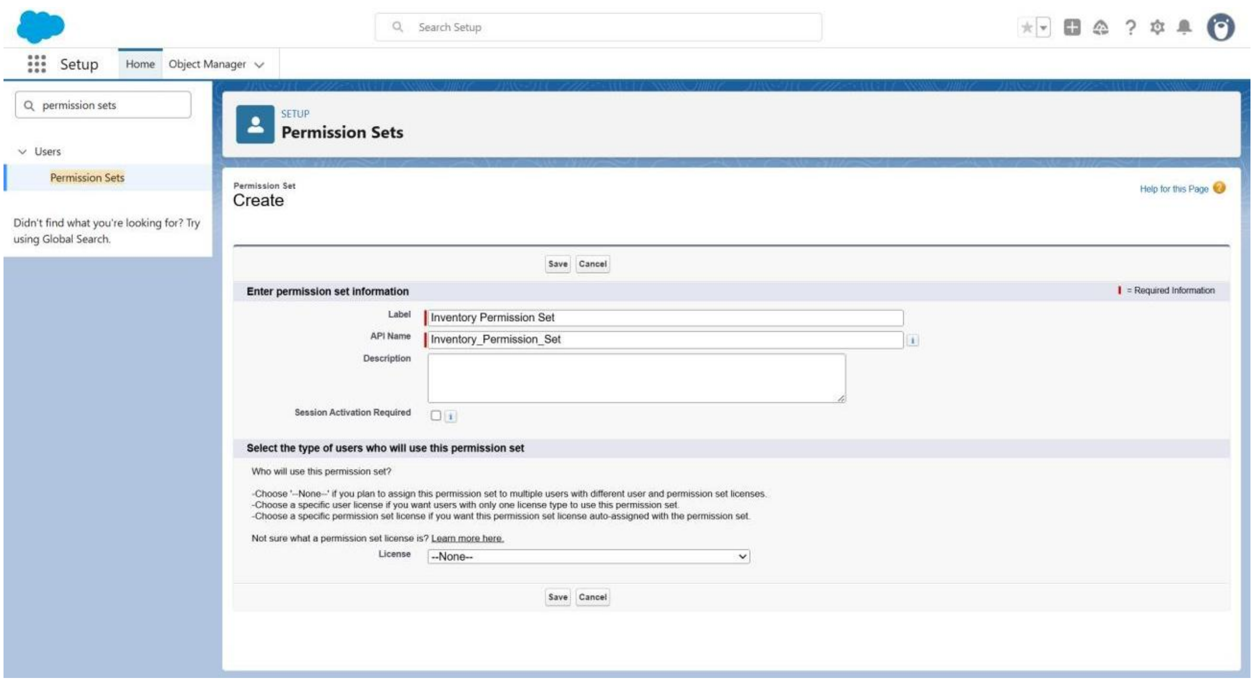
In the **Sales Permission Set**, go to **Find Settings**, search for **HandsMen Orders**, then click the **Edit** button. In the **Object Permissions** section, enable **Read**, **Create**, **Edit**, and **Delete**, then click the **Save** button.


... > PERMISSION SET 'SALES PERMISSION SET' > MANAGE ASSIGNMENT EXPIRATION
Sales permission set
Select Users to Assign
All Users
1 item selected
Search this list...
Full Name
Alias
Username
Role
Active
Profile
antonio miguel orias
mig
miguel.orias02521@agentforce.com
System Administrator
Chatter Expert
Chatter
chatty.00dgl00000diodzua1.5lt9ok09mr5r@chatter.salesforce.com
Chatter Free User
Daniel Mikaelson
dmika
miguel.orias300@gmail.com
Marketing
Platform 1
Integration User
integ
integration@00dgl00000diodzua1.com
Analytics Cloud Integration User
Kol Mikaelson
kmika
miguel.orias200@gmail.com
Inventory
Platform 1
Niklaus Mikaelson
nmika
miguel.orias100@gmail.com
Sales
Platform 1
Odyssey Odysseus
Oddy
miguel.orias020@gmail.com
Sales
Platform 1

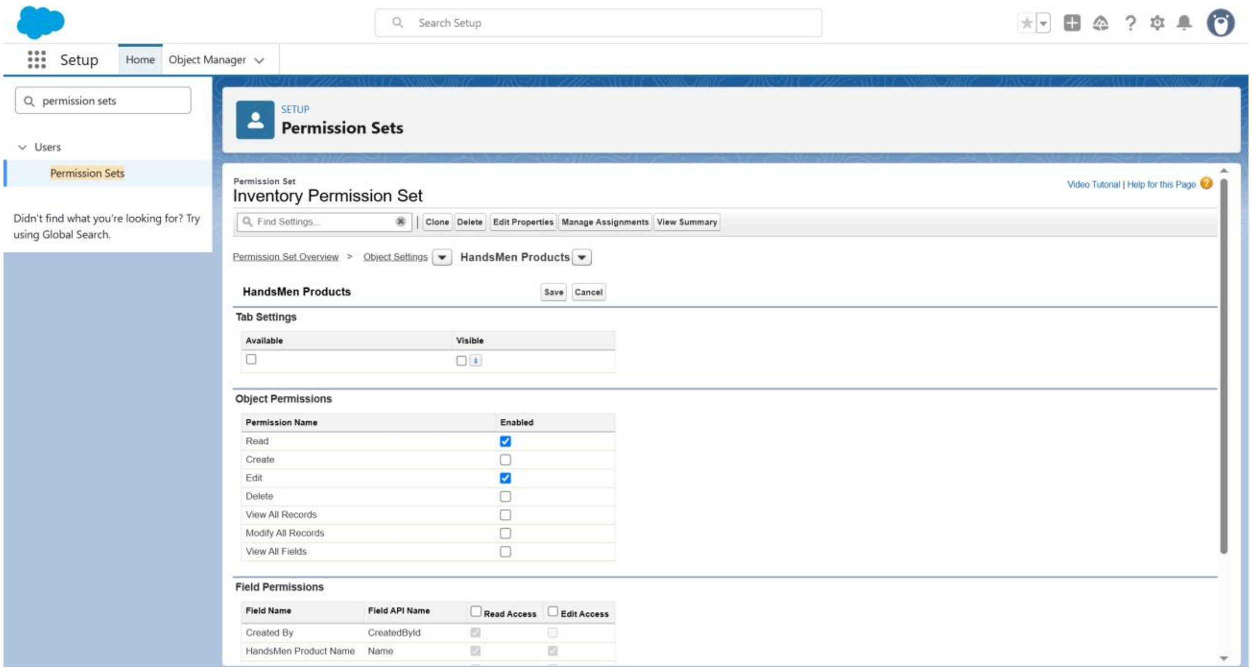

... > PERMISSION SET 'SALES PERMISSION SET' > MANAGE ASSIGNMENT EXPIRATION
Sales permission set
Select an Expiration Option For Assigned Users
No expiration date
Specify the expiration date
1 Day
1 Week
30 Days
60 Days
Custom Date
Time Zone
Select a time zone...
Selected Users
Full Name
Role
Profile
Active
User License
Expires On
Odyssey Odysseus
Sales
Platform 1
Salesforce
Never Expires

In the **Sales Permission Set**, click the **Manage Assignments** button, then click **Add Assignment**. Select the user with the **Sales** role, click **Next**, and then click **Assign** to complete the process.

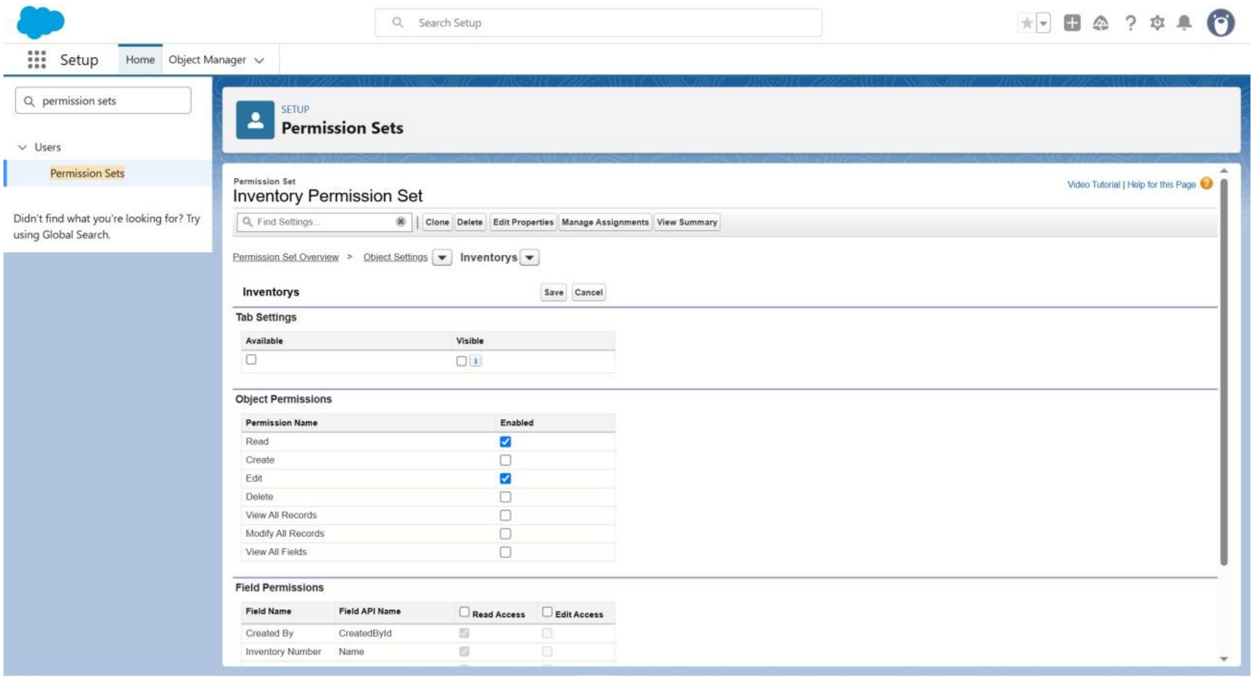
- **Inventory Permission Set:**



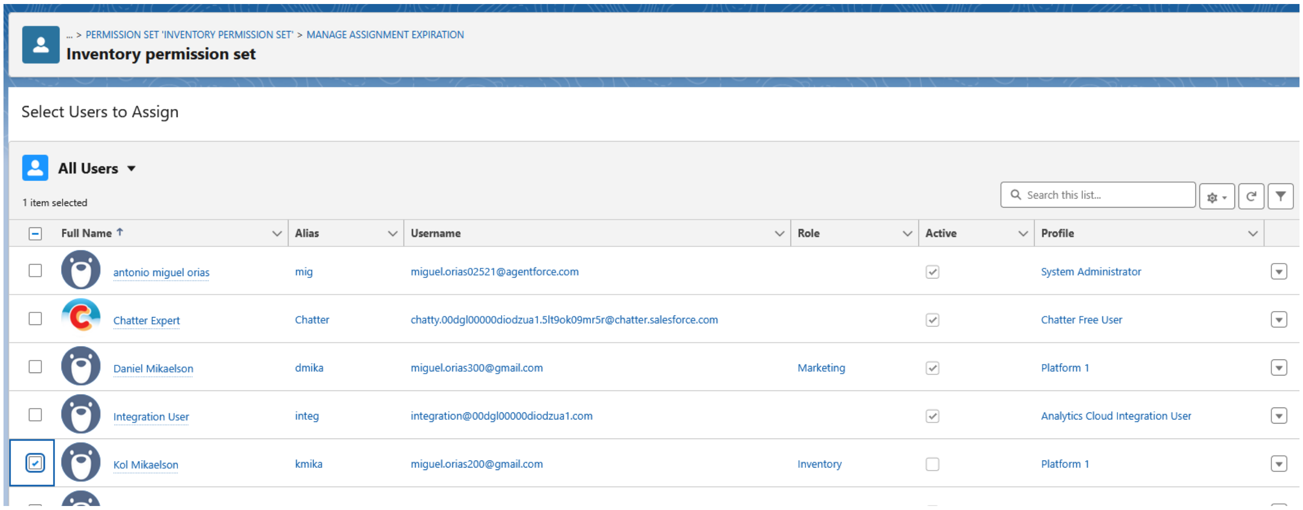
Create a new permission set with the Label **Inventory Permission Set** and API Name **Inventory_Permission_Set**, then click the **Save** button.

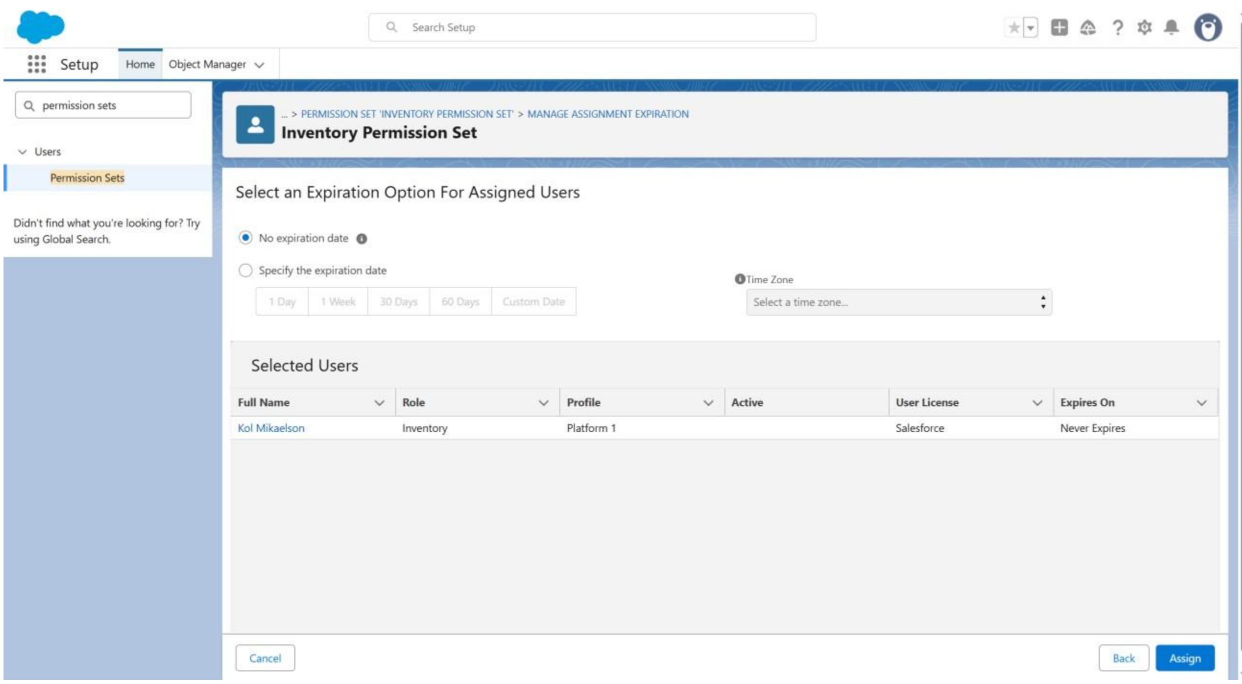


In the **Inventory Permission Set**, use the **Find Settings** search bar to locate **HandsMen Products**. Click the **Edit** button, then enable the **Read** and **Edit** permissions under **Object Permissions**.



In the **Inventory Permission Set**, use the **Find Settings** search bar to locate **Inventorys**. Click the **Edit** button, then enable the **Read** and **Edit** permissions under **Object Permissions**.

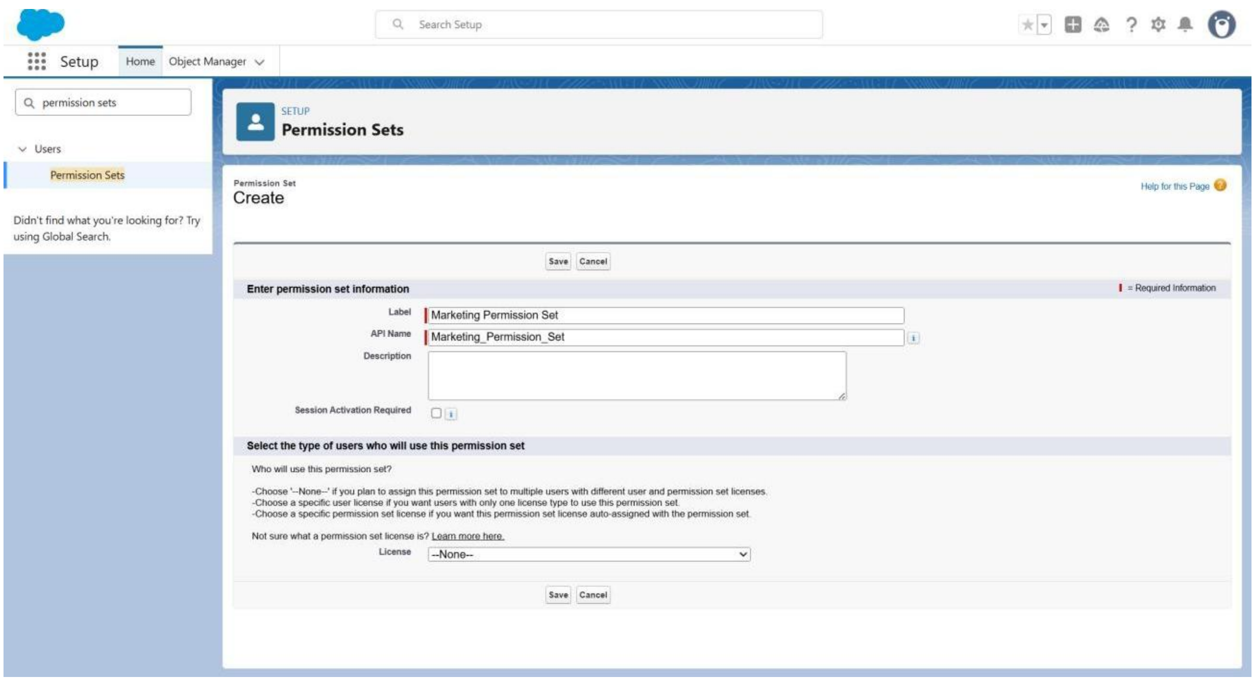




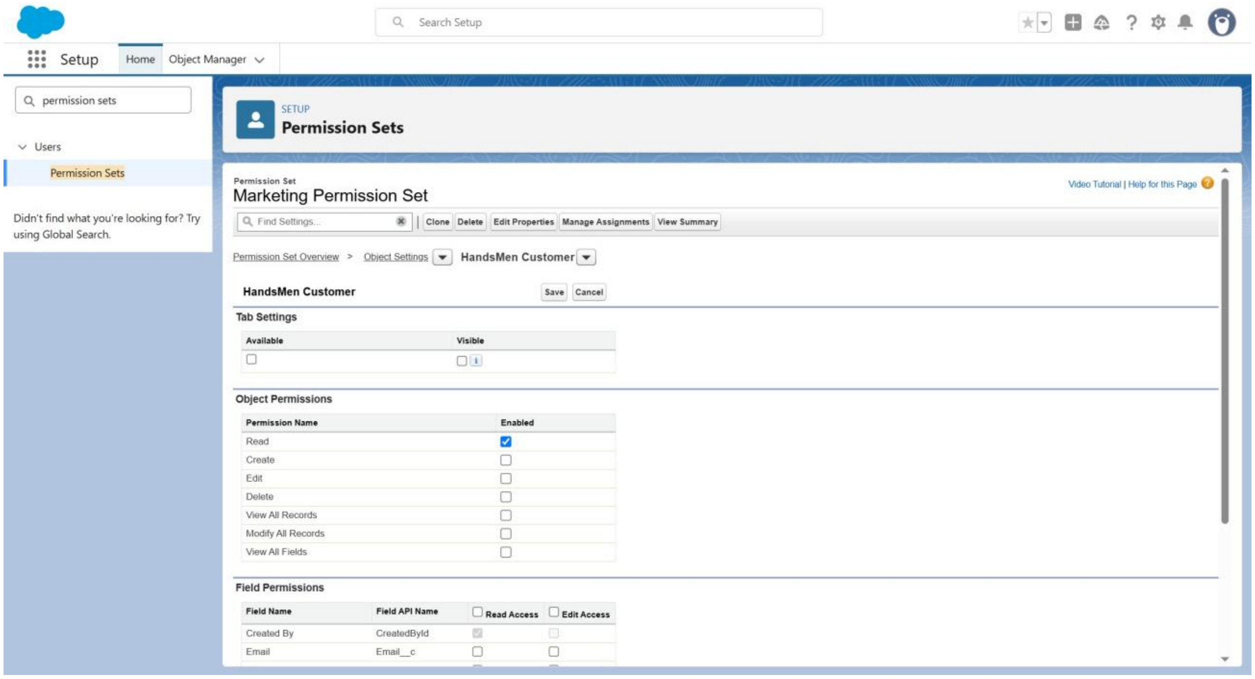
Created By: Antonio Miguel Orias

In the **Inventory Permission Set**, click the **Manage Assignments** button, then click **Add Assignment**. Select the user with the **Inventory** role, click **Next**, and then click **Assign** to complete the process.

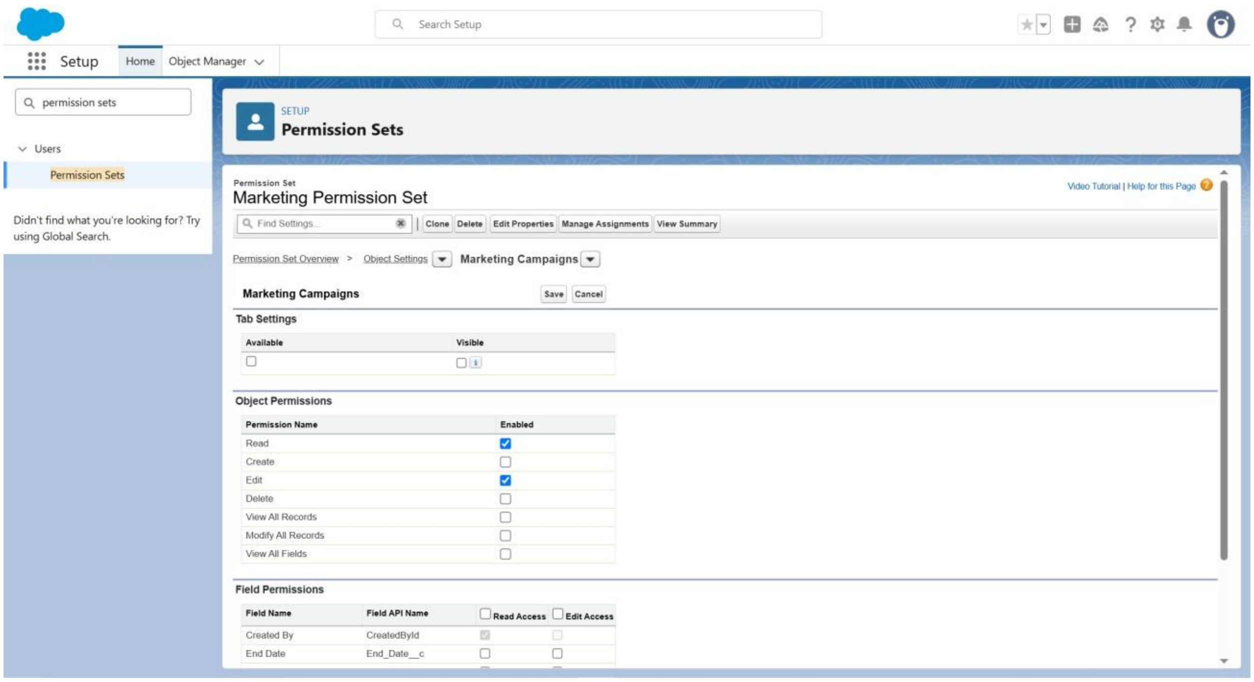
- **Marketing Permission Set:**



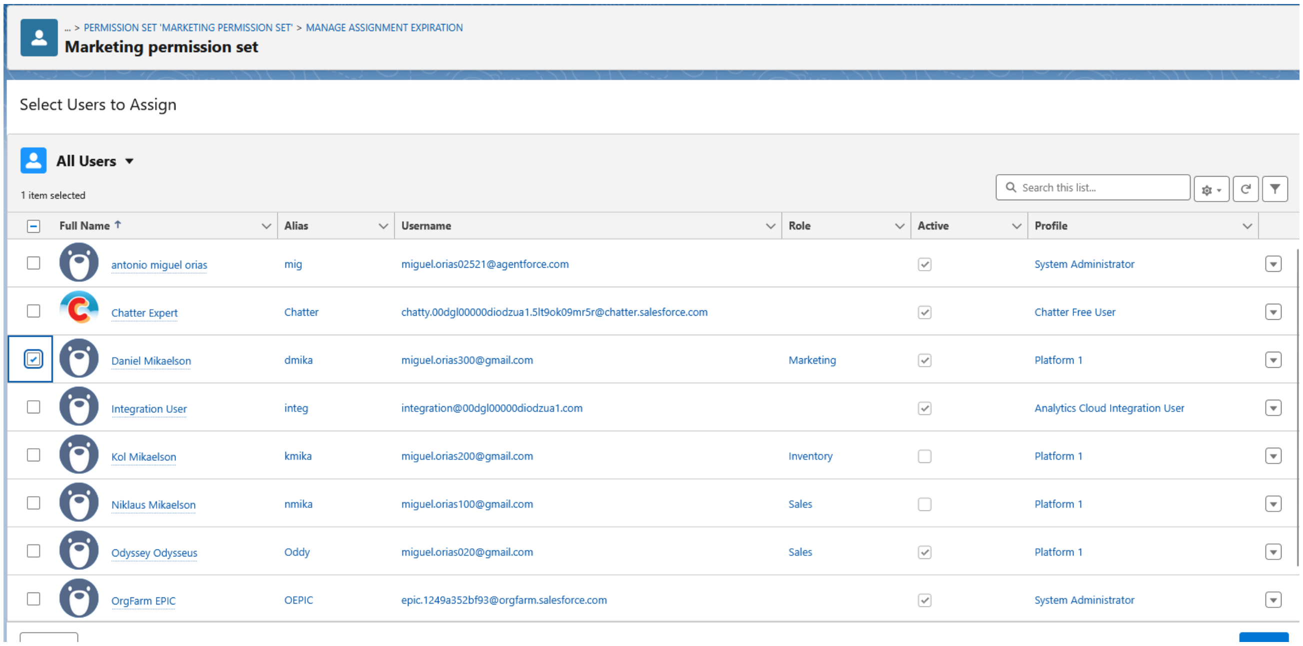
Create a new permission set with the Label **Marketing Permission Set** and API Name **Marketing_Permission_Set**, then click the **Save** button.

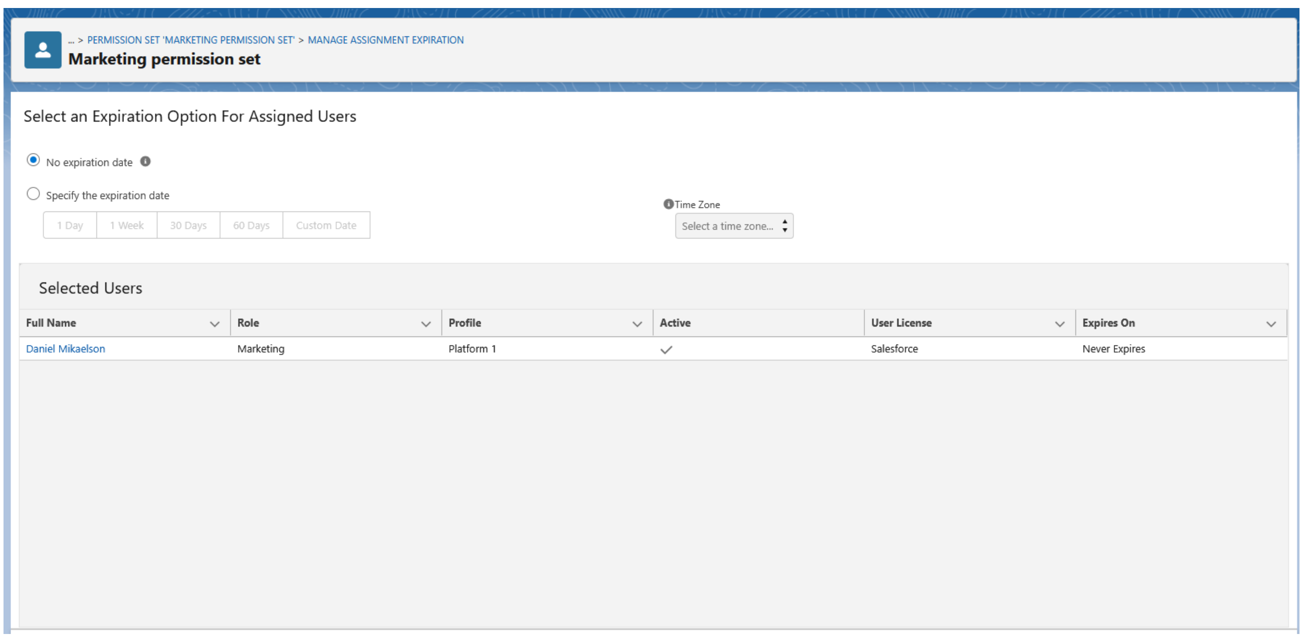


In the **Marketing Permission Set**, use the **Find Settings** search bar to locate **HandsMen Customer**. Click the **Edit** button, then enable the **Read** permission under **Object Permissions**.



In the **Marketing Permission Set**, use the **Find Settings** search bar to locate **Marketing Campaigns**. Click the **Edit** button, then enable the **Read** and **Edit** permission under **Object Permissions**.





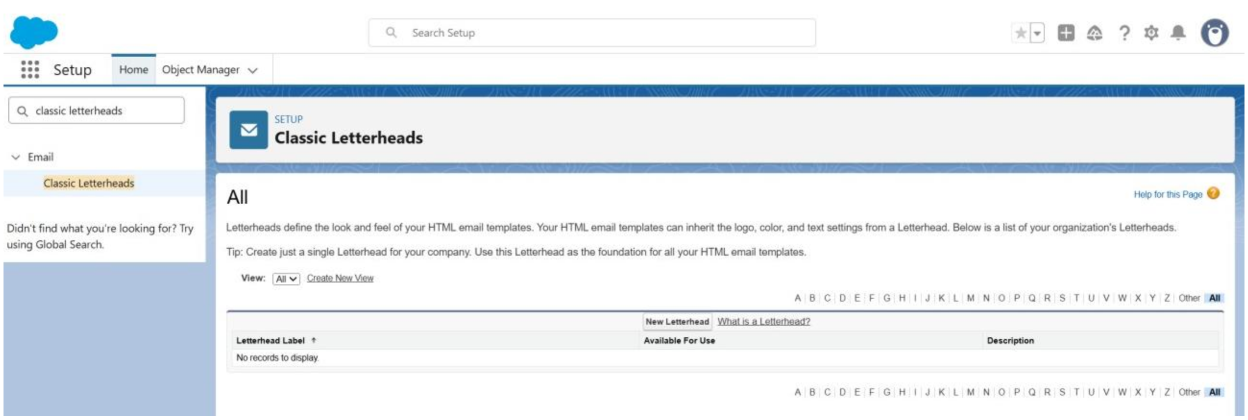
Created By: Antonio Miguel Orias

In the **Marketing Permission Set**, click the **Manage Assignments** button, then click **Add Assignment**. Select the user with the **Marketing** role, click **Next**, and then click **Assign** to complete the process.

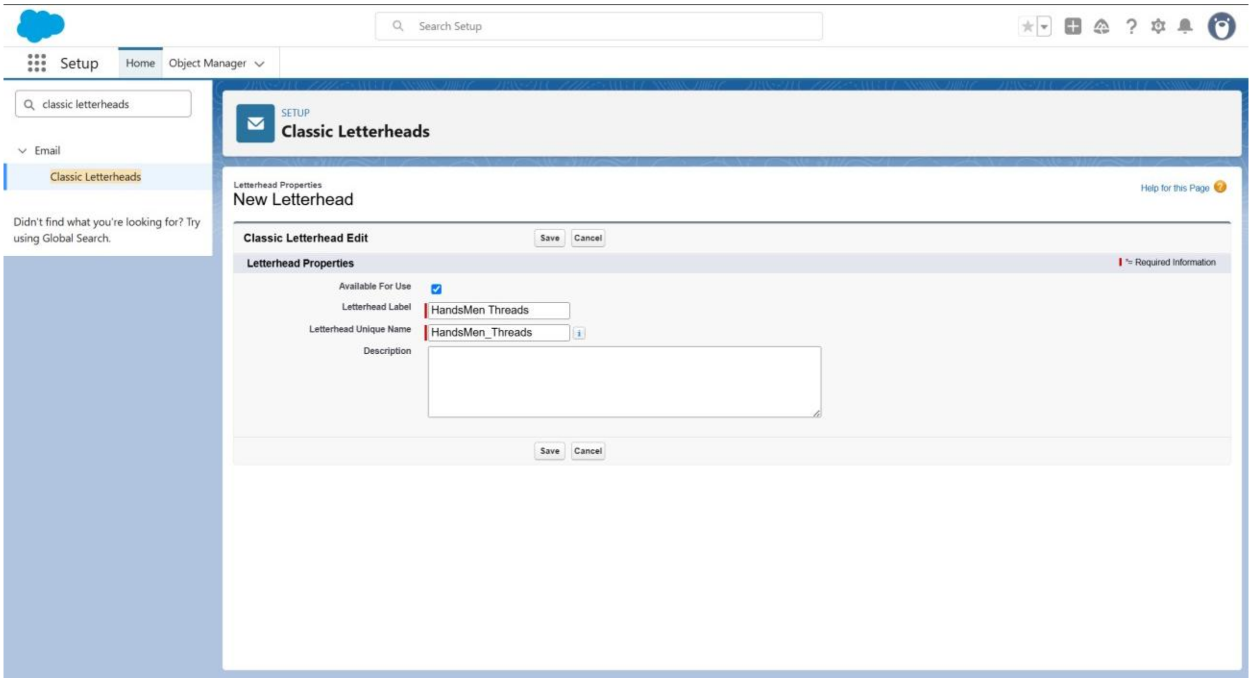
Email Templates

Classic Letterheads:

In **Home**, use **Quick Find** to search for **Classic Letterheads**, then click the **Next** button to proceed.



After clicking the Next button in Classic Letterheads, click the **New Letterhead** button to begin creating a new letterhead.

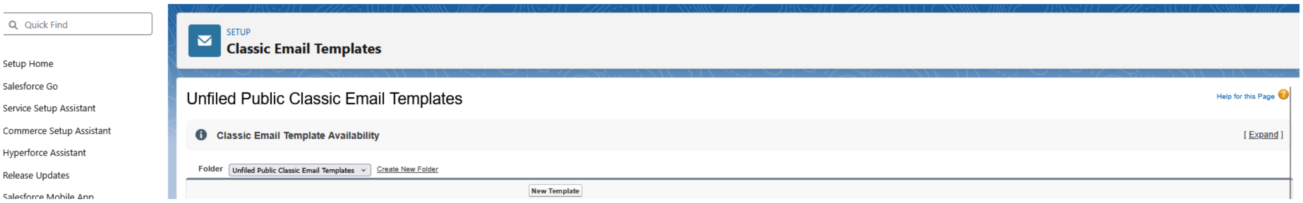


Select **Available For Use**, enter **Letterhead Label: HandsMen Threads** and **Letterhead Unique Name: HandsMen_Threads**, then click the **Save** button.

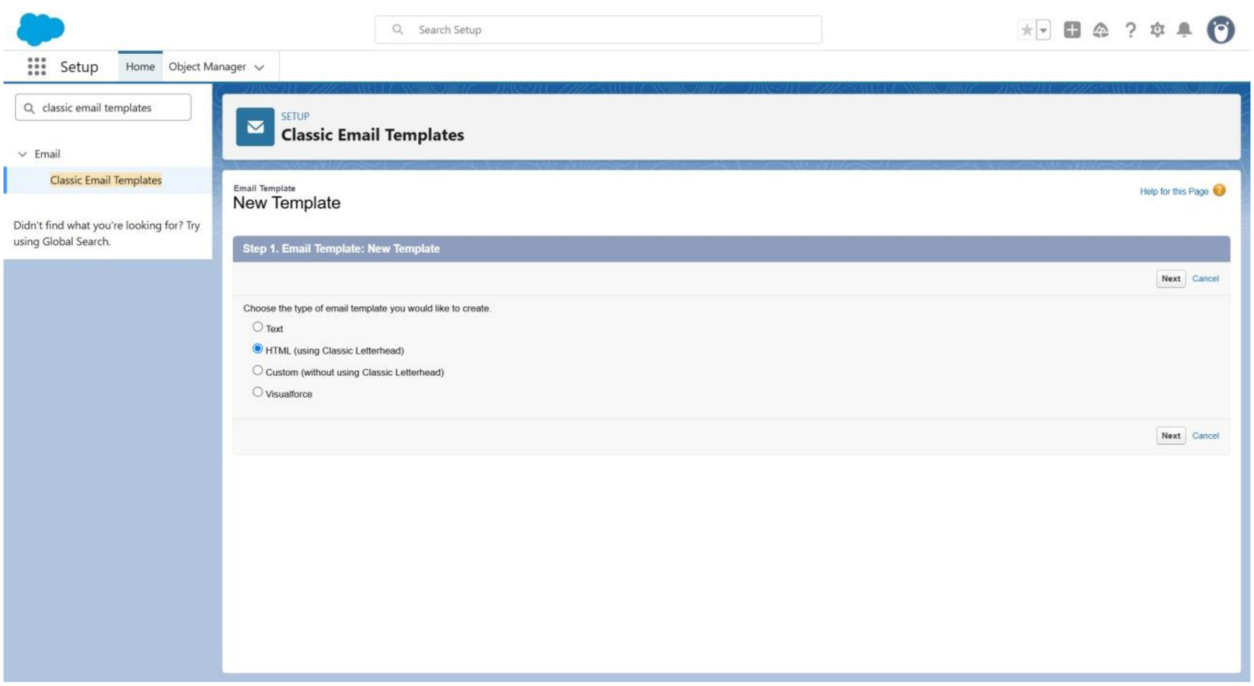

Edit Background Color
Edit Header Properties
Select Logo
Remove Logo
Edit Top Line
Edit Body Colors
Edit Middle Line
Edit Footer Properties
Select Logo
Remove Logo

To edit your letterhead, select the letterhead you created, then click the **Edit Letterhead** button.

Order Confirmation Email Template:



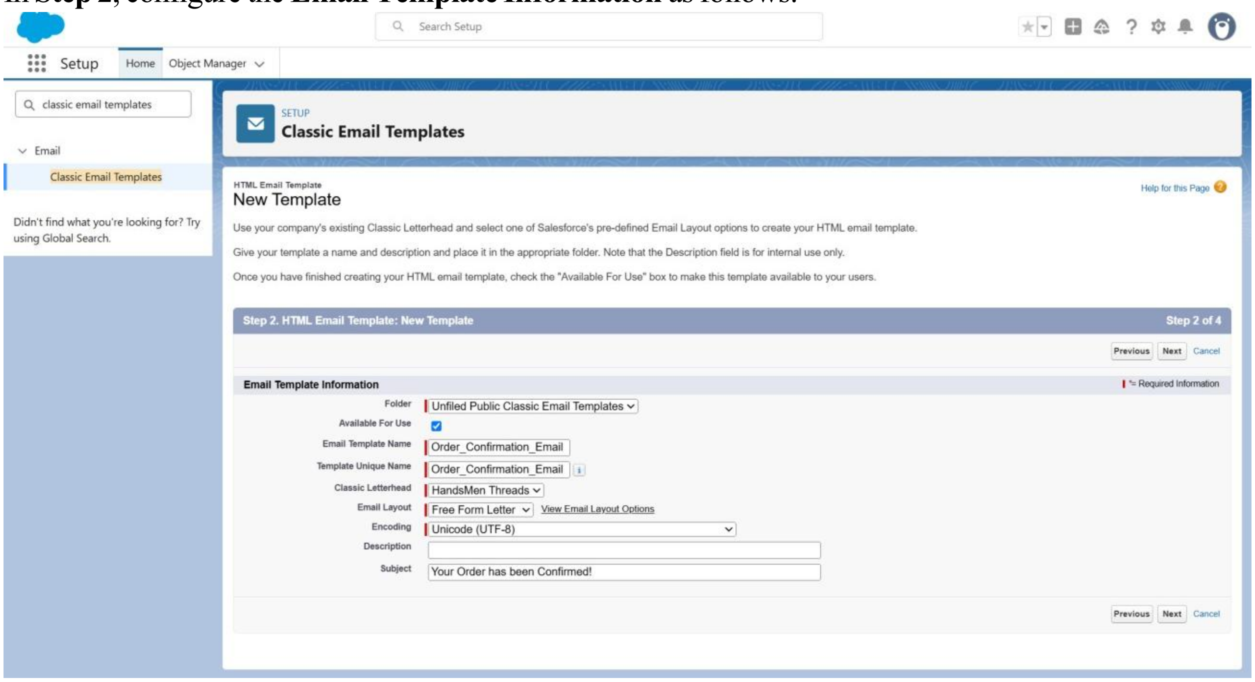
In **Home**, use **Quick Find** to search for **Classic Email Templates**, then click the **New Template** button to create a new email template.



Created By: Antonio Miguel Orias

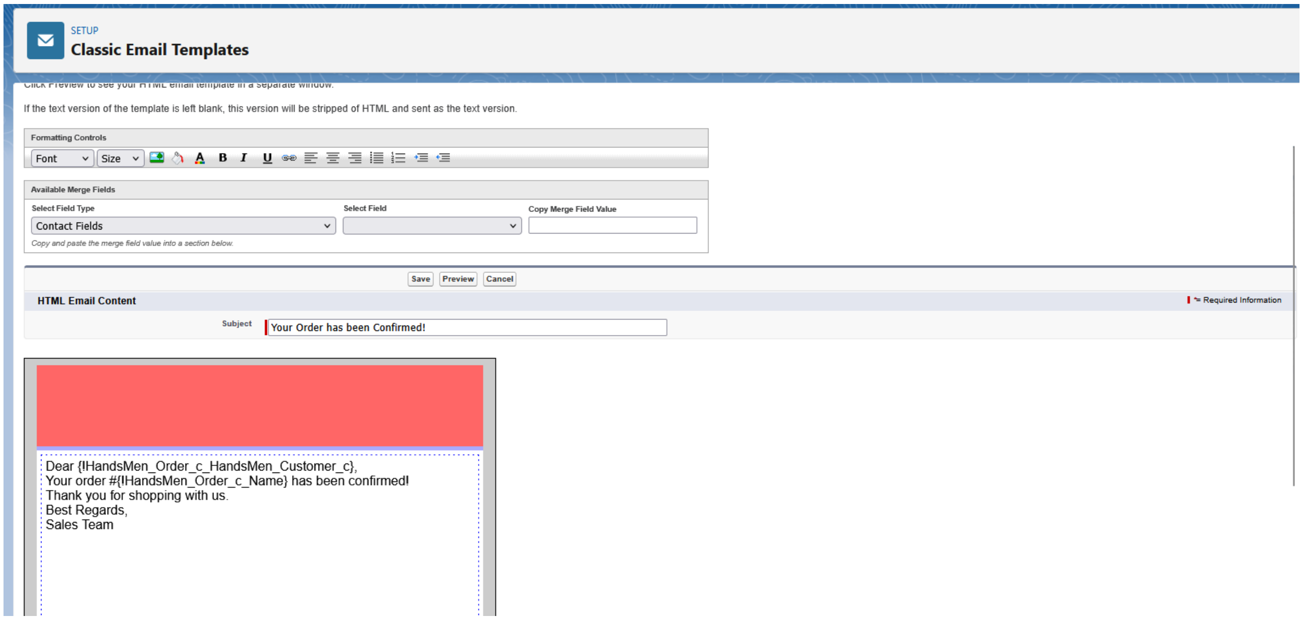
In **Step 1**, select the email template type as **HTML (using Classic Letterhead)**, then click the **Next** button to proceed.

In **Step 2**, configure the **Email Template Information** as follows:

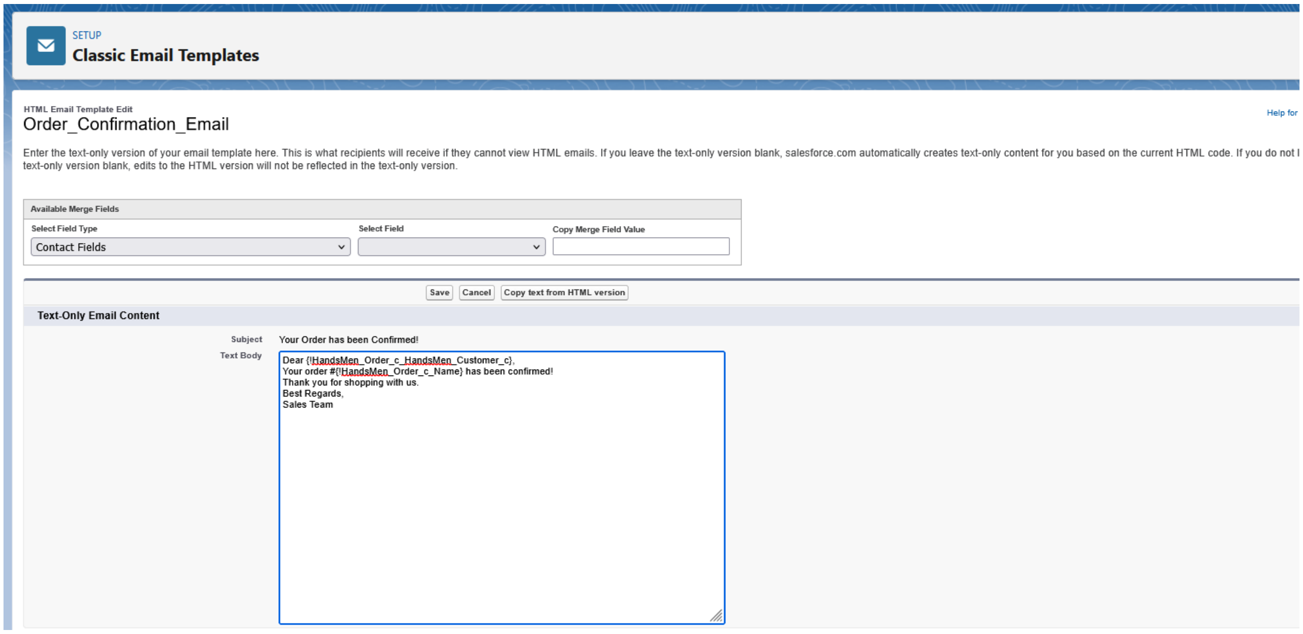


- **Folder:** Unified Public Classic Email Templates
- **Available for Use:** Selected
- **Email Template Name:** Order_Confirmation_Email
- **Template Unique Name:** Order_Confirmation_Email
- **Classic Letterhead:** HandsMen Threads
- **Email Layout:** Free Form Letter
- **Encoding:** Unicode (UTF-8)
- **Description:** Optional
- **Subject:** [Enter your preferred subject]

Then, click the **Next** button to proceed.

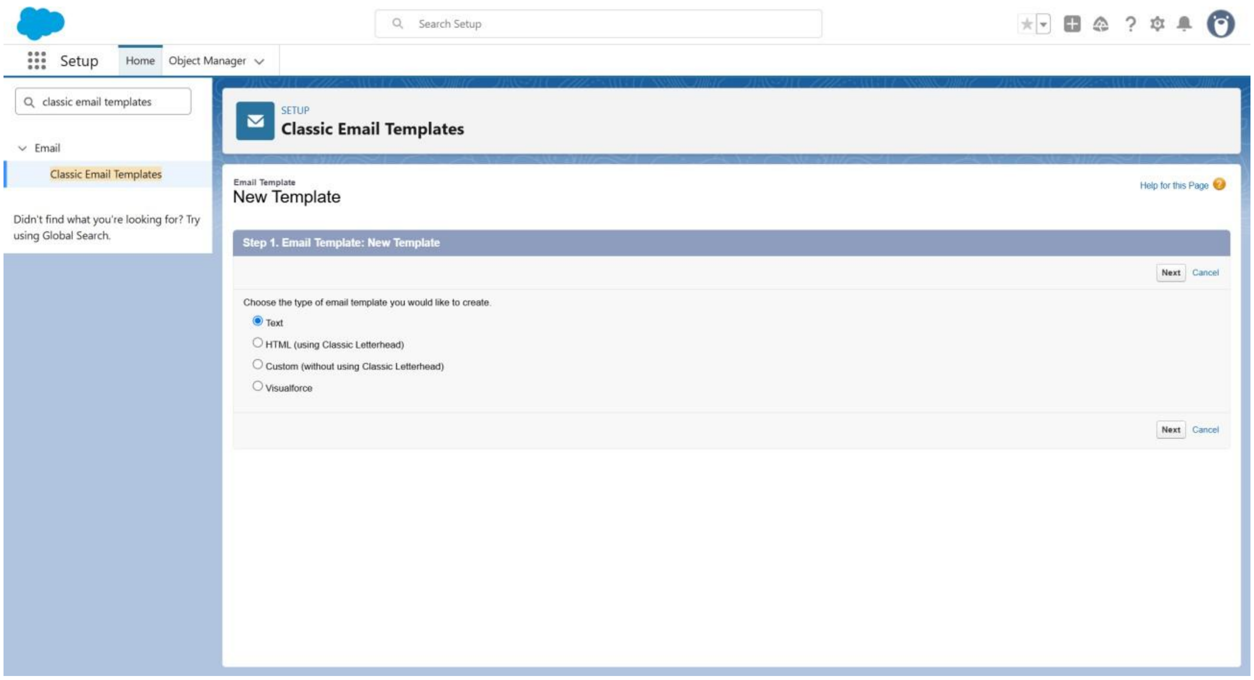


In **Step 3**, add the **content** of your letter in the body section of the email template. This is where you can write your message, include merge fields if needed, and format the text according to your preferred layout and tone. Once you're satisfied with the content, click the **Next** button to proceed.

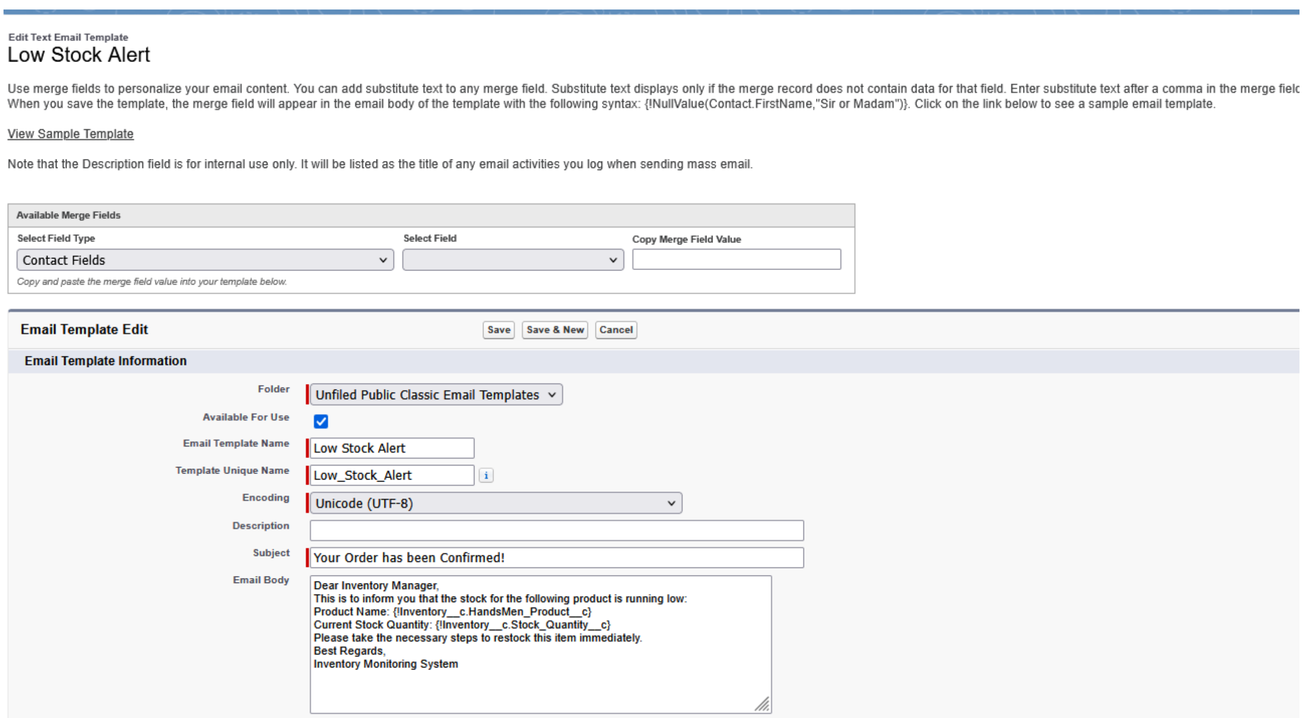


In **Step 4**, click the **Copy text from HTML version** button, confirm by clicking **OK**, then click the **Save** button to finalize the email template.

Low Stock Alert Email Template:



Create another **new template** and in **Step 1**, select the email template type as **Text**, then click the **Next** button to proceed.

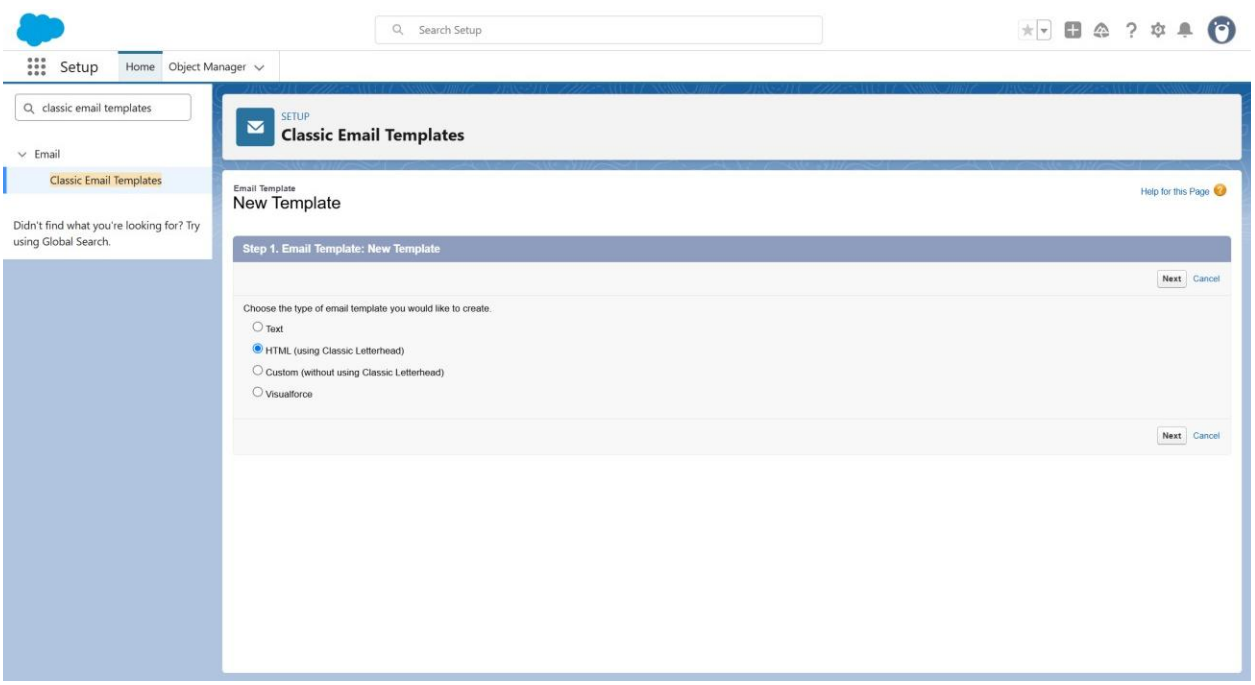


In **Step 2**, configure the **Email Template Information** as follows:

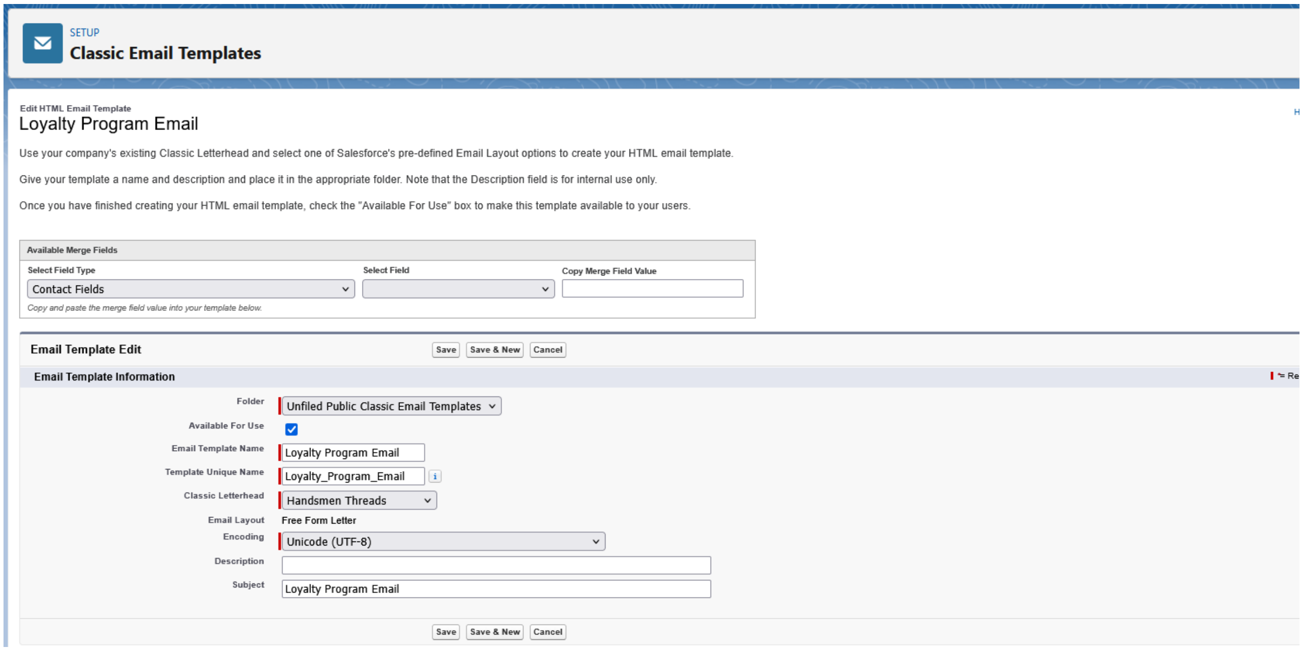
- **Folder:** Unified Public Classic Email Template
- **Available For Use:** Selected
- **Email Template Name:** Low Stock Alert
- **Template Unique Name:** Low_Stock_Alert
- **Encoding:** Unicode (UTF-8)
- **Description:** Optional
- **Subject:** [Your preferred subject]
- **Email Body:** [Your preferred email message]

Then, click the **Save** button to finalize the template. **Loyal**

Program Email Template:



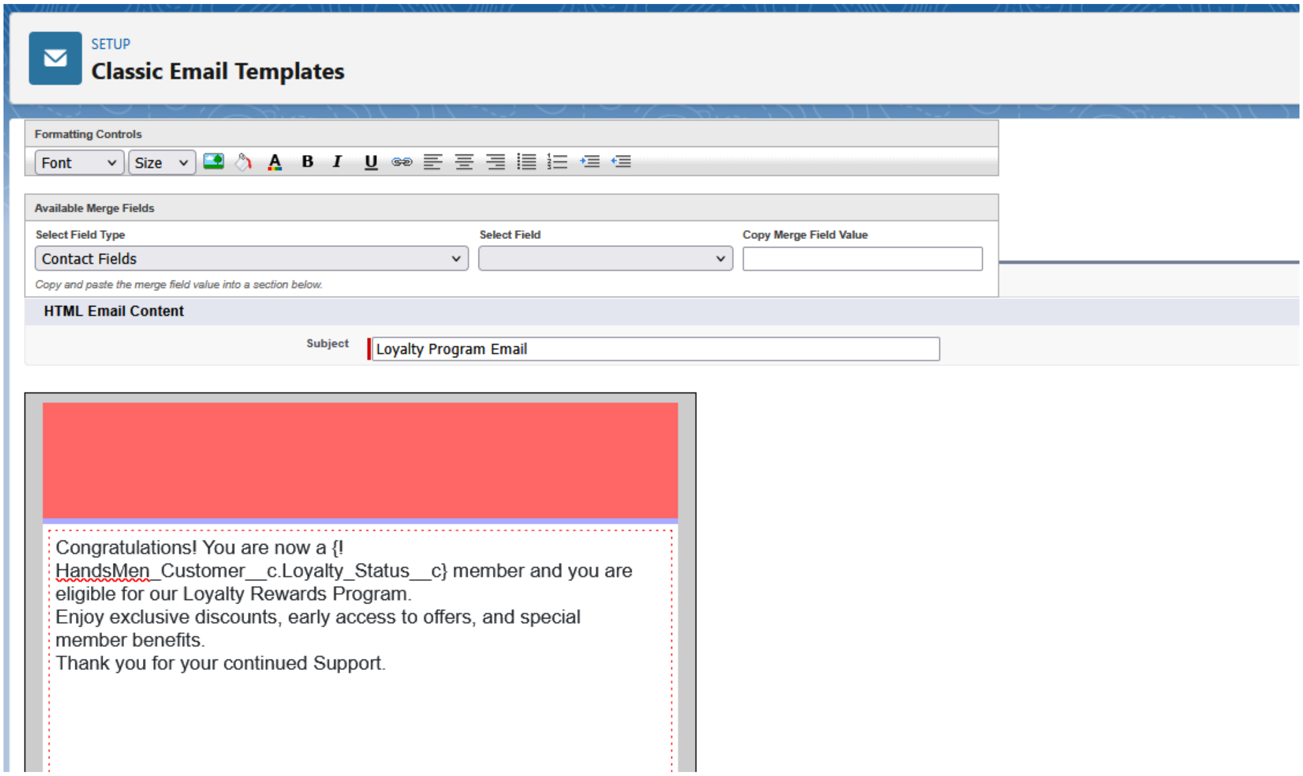
Create another **new template** and in **Step 1**, select the email template type as **HTML (using Classic Letterhead)**, then click the **Next** button to proceed.



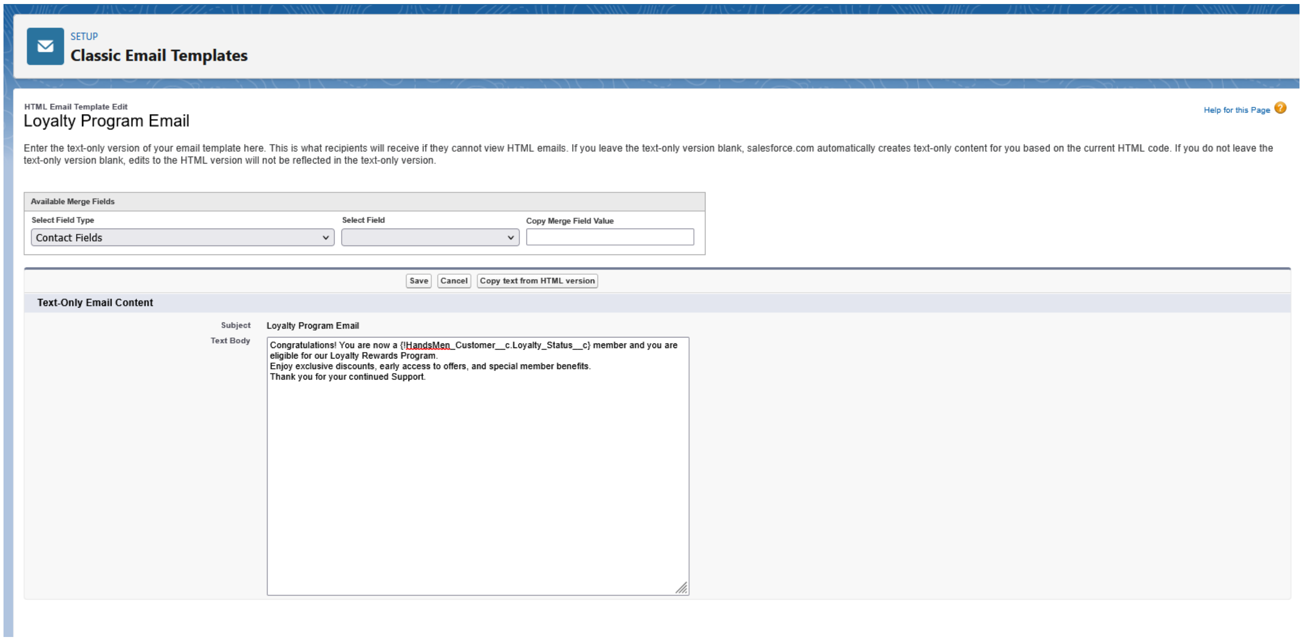
In **Step 2**, configure the **Email Template Information** as follows:

- **Folder:** Unified Public Classic Email Templates
- **Available for Use:** Selected
- **Email Template Name:** Loyalty Program Email
- **Template Unique Name:** Loyalty_Program_Email
- **Classic Letterhead:** HandsMen Threads
- **Email Layout:** Free Form Letter
- **Encoding:** Unicode (UTF-8)
- **Description:** Optional
- **Subject:** [Enter your preferred subject]

Then, click the **Next** button to proceed.

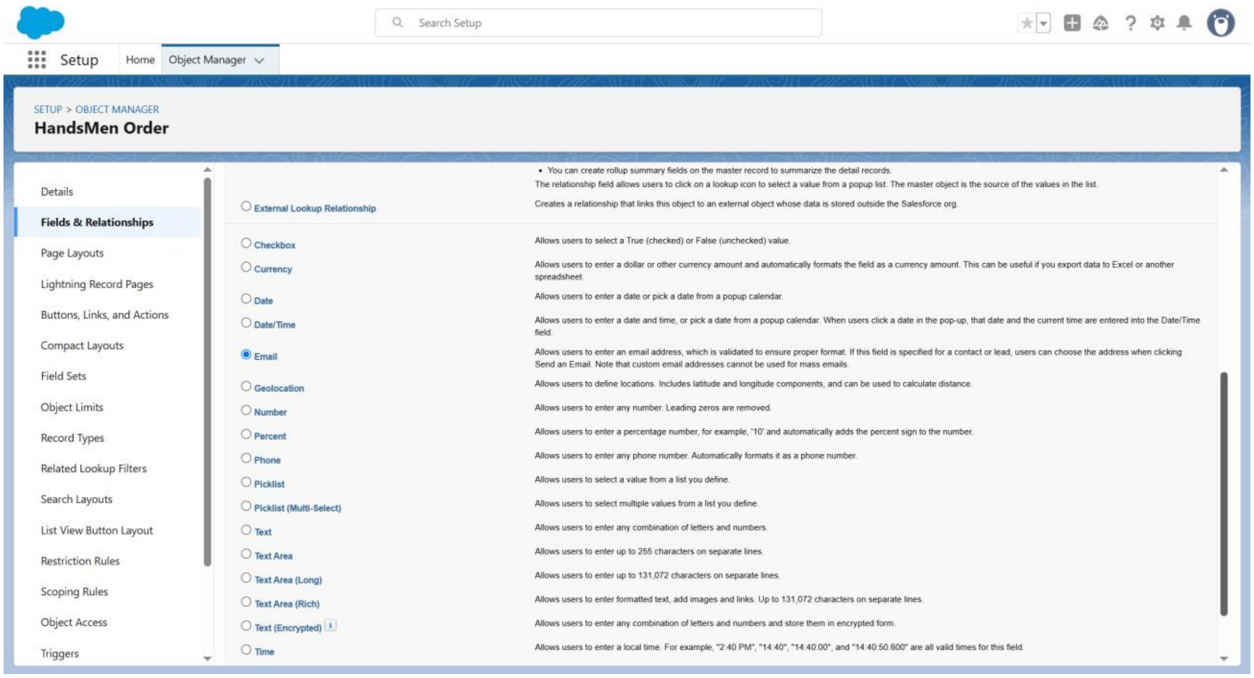


In **Step 3**, add the **content** of your letter in the body section of the email template. This is where you can write your message, include merge fields if needed, and format the text according to your preferred layout and tone. Once you're satisfied with the content, click the **Next** button to proceed.

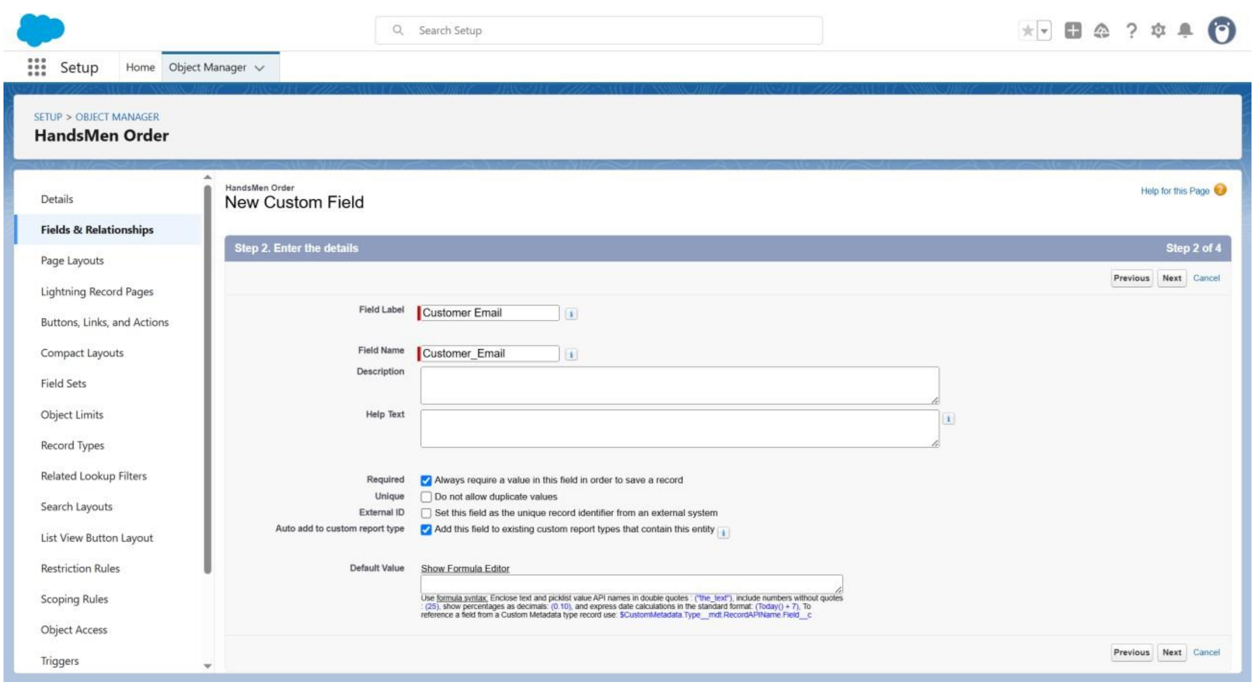


In **Step 4**, click the **Copy text from HTML version** button, confirm by clicking **OK**, then click the **Save** button to finalize the email template.

- **Before creating the Email Alert, first create the Customer Email by following these steps:**



In the **HandsMen Order** object settings, click **Fields s Relationships**, then click the **New** button. In **Step 1**, select the **Email** data type to create a new email field.

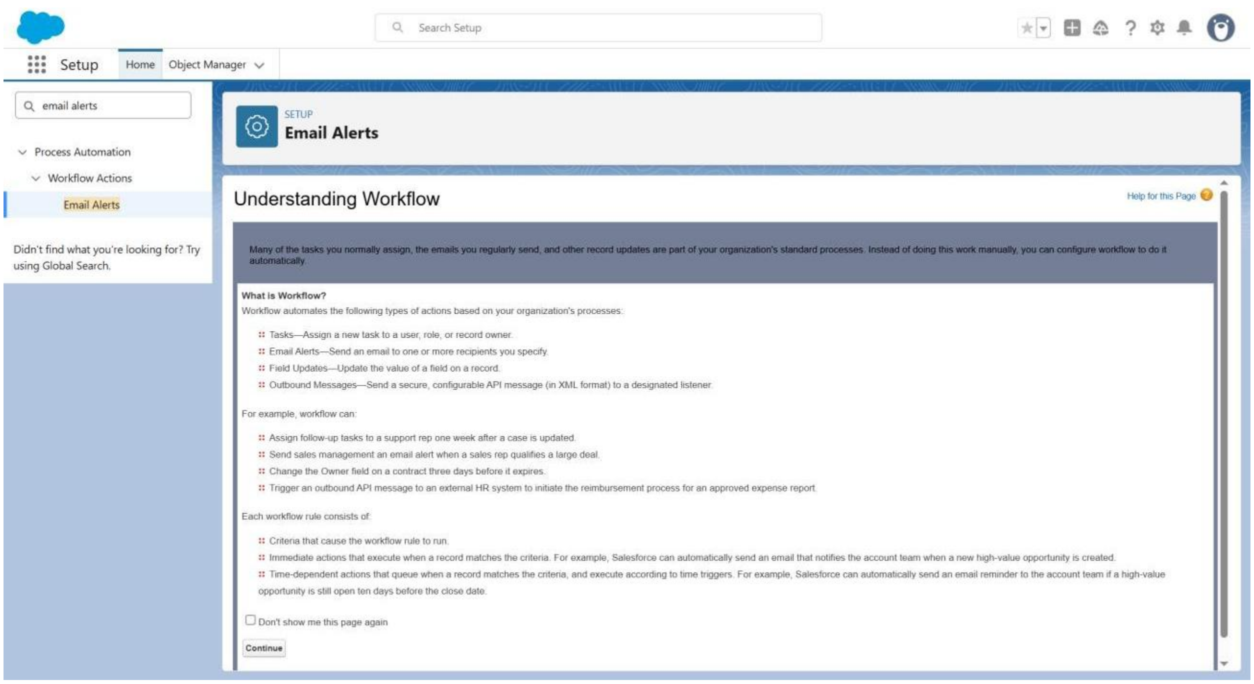


In **Step 2** of creating the new field on the **HandsMen Order** object, configure the field as follows:

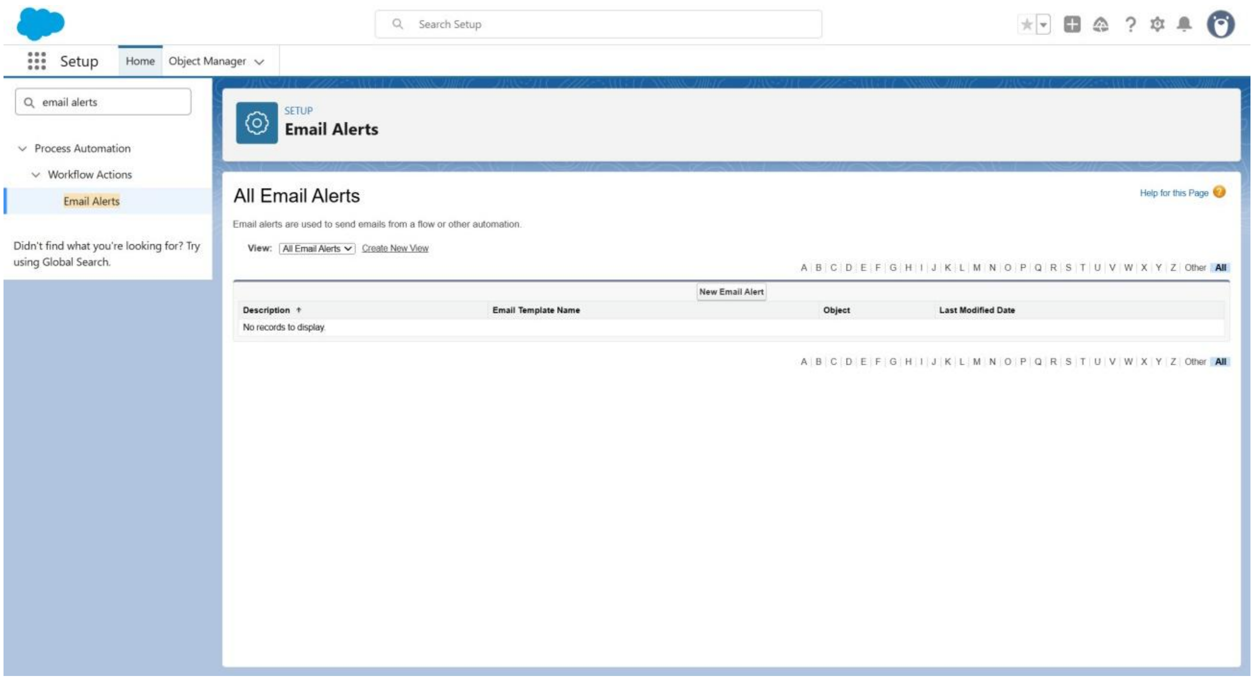
- **Field Label:** Customer Email
- **Field Name:** Customer_Email (auto-populated based on the label)
- **Required:** Selected (check the box to make this field mandatory)

Then, click **Next** through the remaining steps, accepting the default settings or adjusting as needed, and finally click the **Save** button to create the field.

Email Alert setup:

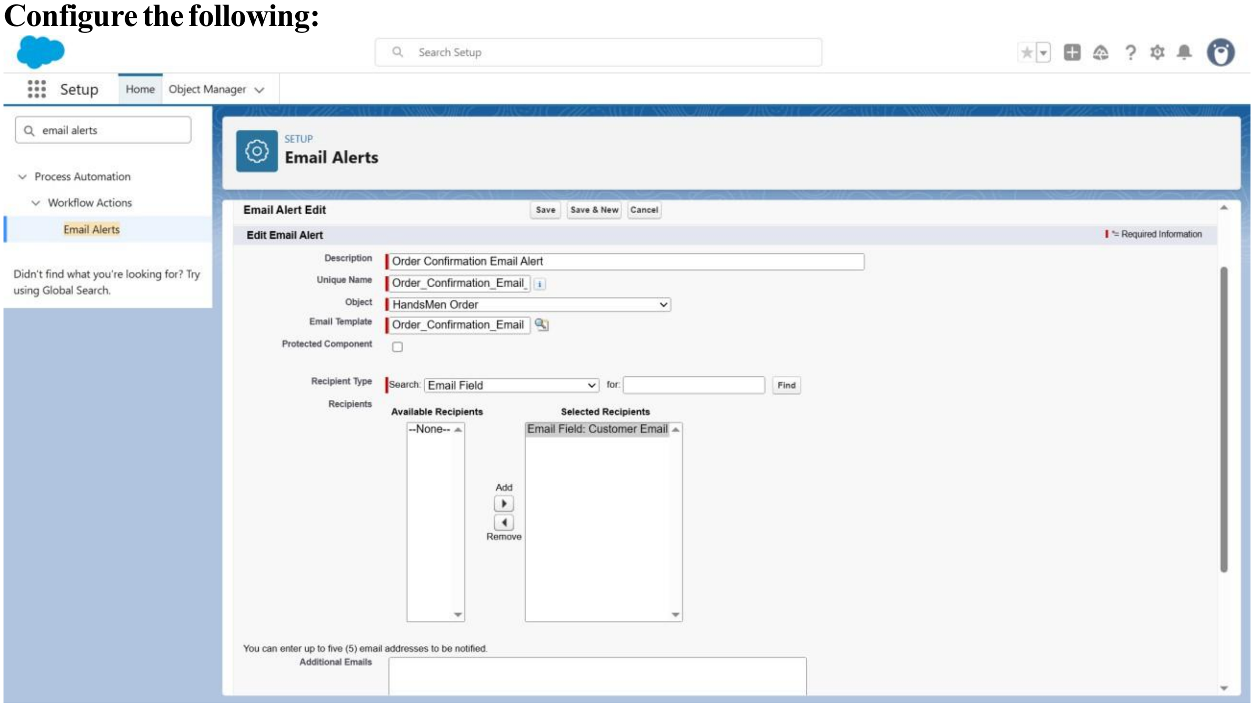


In the **Home** tab, use the **Quick Find** search bar to search for **Email Alerts**, then click the **Continue** button.



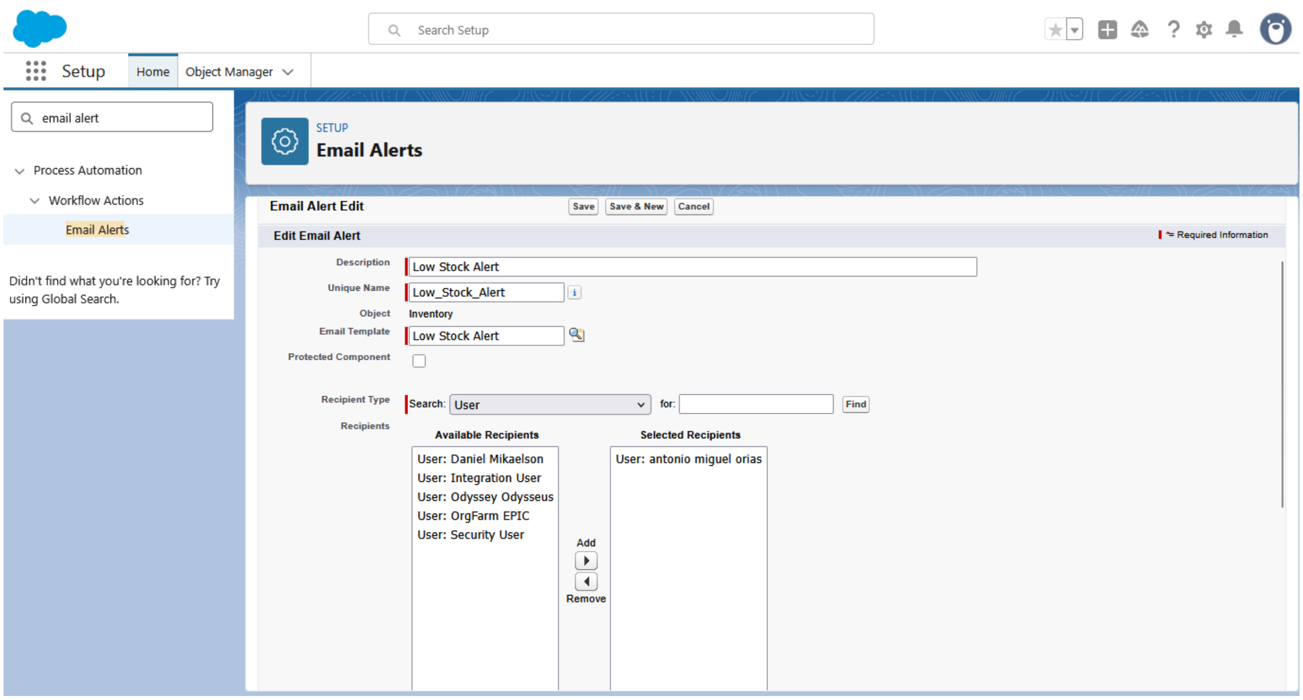
Click the **New Email Alert** button.

Configure the following:



- **Description:** Order Confirmation Email Alert
- **Unique Name:** Order_Confirmation_Email_Alert
- **Object:** HandsMen Order
- **Email Template:** Order_Confirmation_Email
- **Recipient Type:** Email Field
- **Recipients:** Email Field: Customer Email

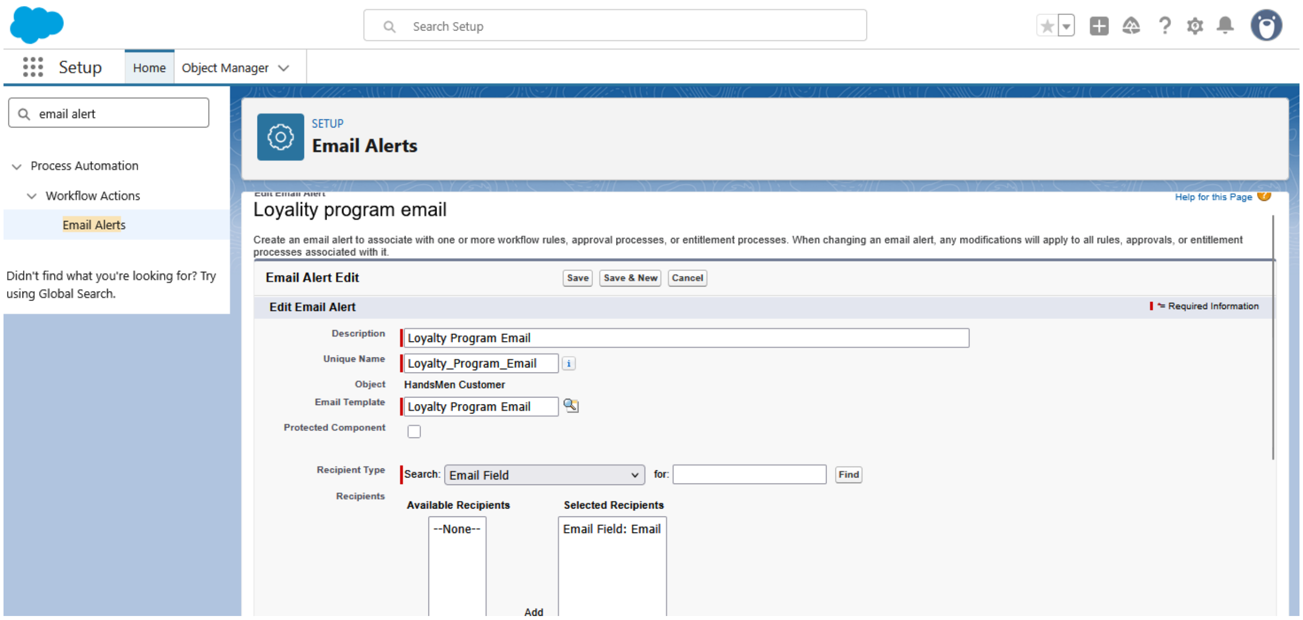
Then click **Save s New** button.



Create a new email alert and configure it as follows:

- **Description:** Low Stock Alert
- **Unique Name:** Low_Stock_Alert
- **Object:** Inventory
- **Email Template:** Low Stock Alert
- **Recipient Type:** User
- **Recipient:** [your name]

Then click the **Save s New** button.

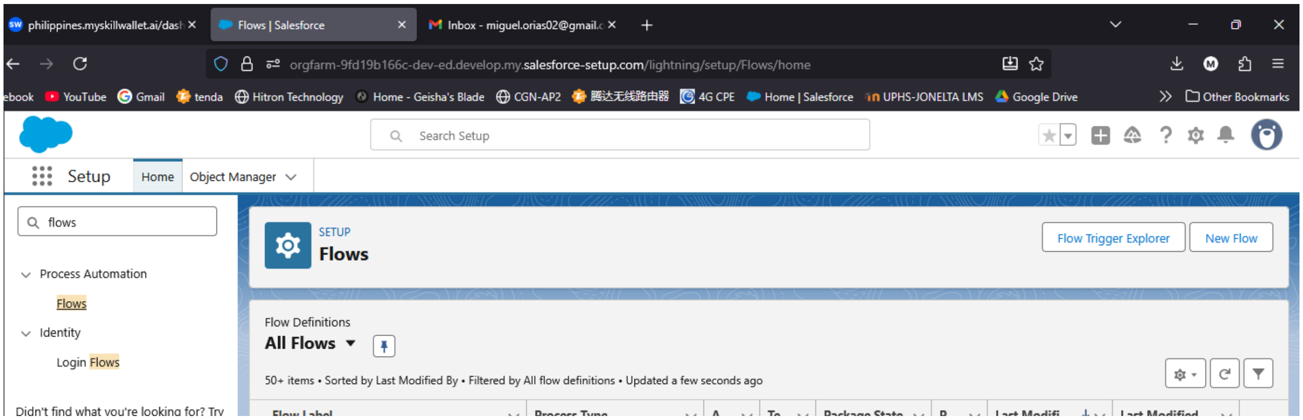


Create a new email alert and configure it as follows:

- **Description:** Loyalty Program Email
- **Unique Name:** Loyalty_Program_Email
- **Object:** HandsMen Customer
- **Email Template:** Loyalty Program Email
- **Recipient Type:** Email Field
- **Recipient:** Email Field: Email

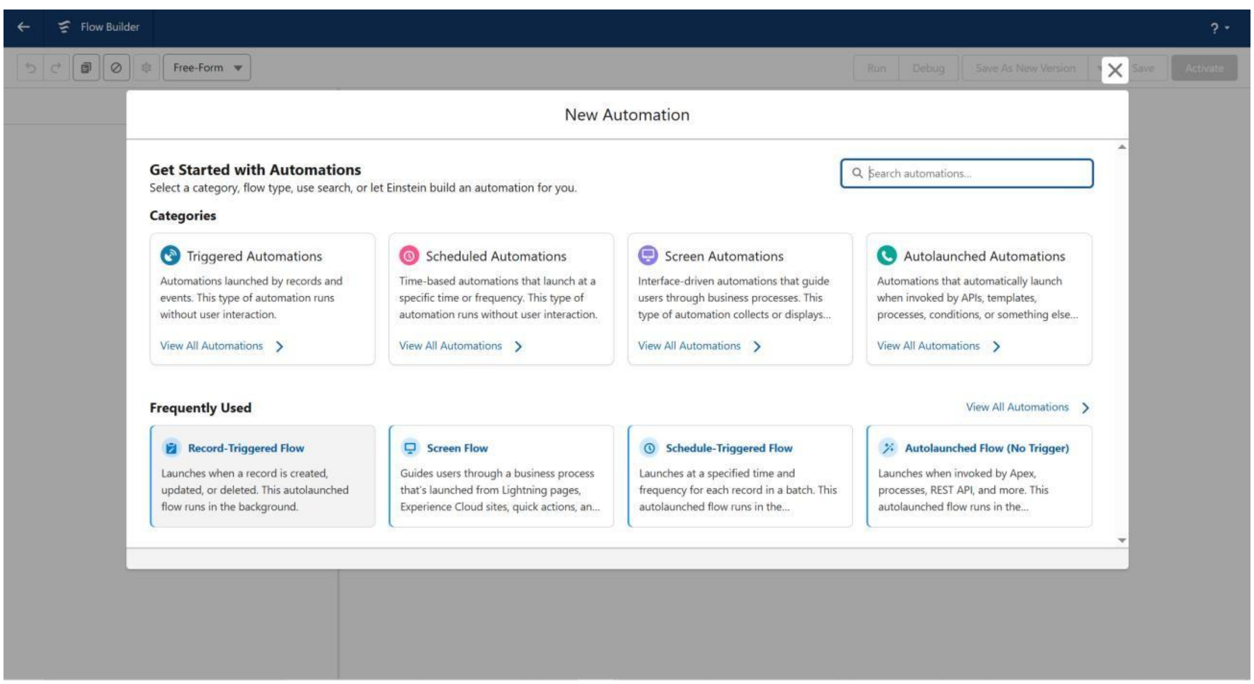
Then click the **Save** button.

Flows

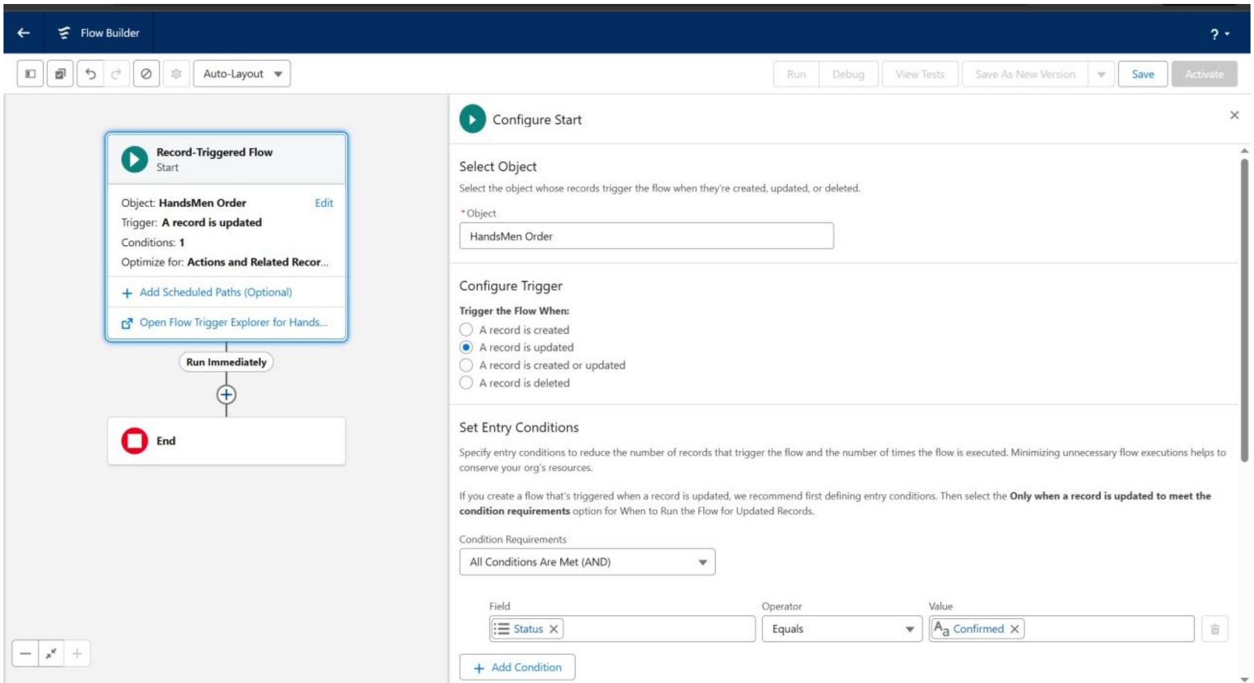


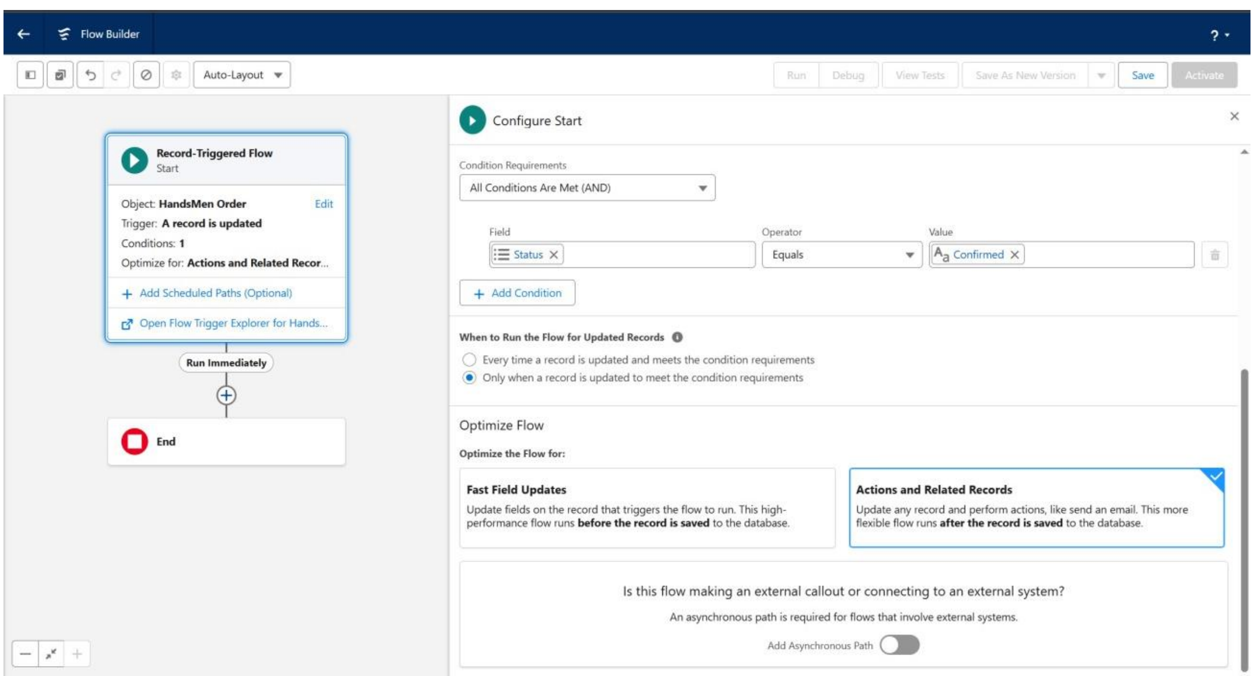
In **Home**, use **Quick Find** to search for **Flows**, then click the **New Flow** button to begin creating a new flow.

Record Triggered:



In **Flow Builder**, select **Record-Triggered Flow** as the flow type to begin automating processes based on record changes.

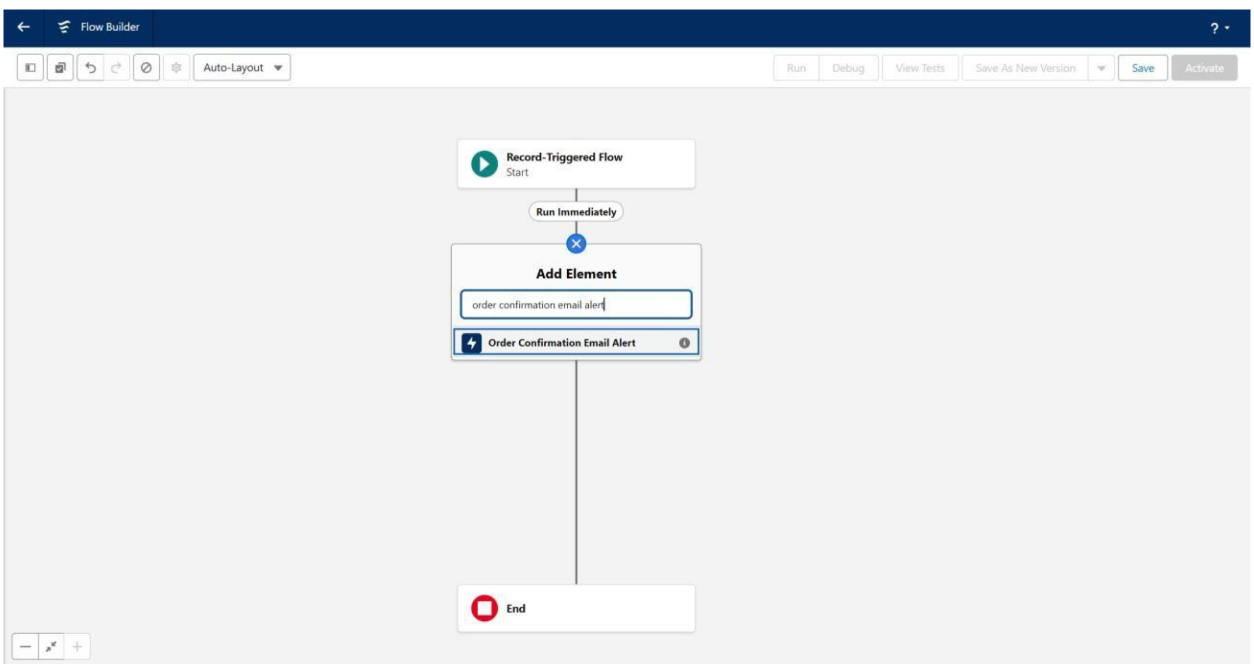




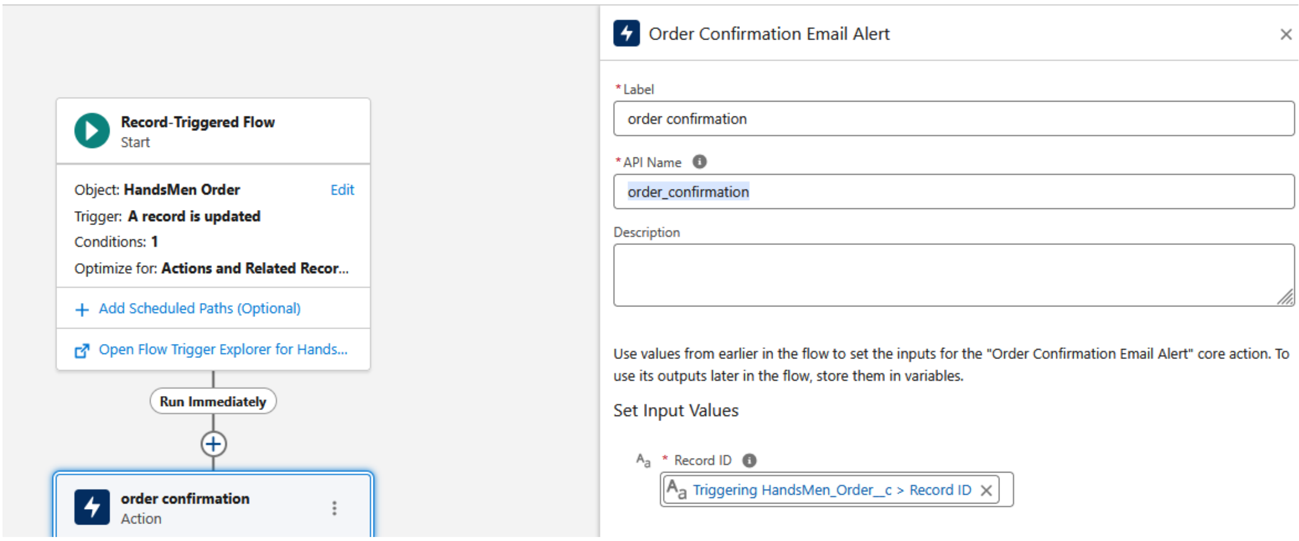
Configure the flow as follows:

- **Select Object Type**
    - Object: HandsMen Order
- **Configure Trigger**
    - **Trigger the Flow When:** A record is updated
- Set Entry Conditions
    - **Conditions Requirements:** All Conditions Are Met (AND)
        - **Field:** Status
        - **Operator:** Equals
        - **Value:** Confirmed
    - **When to Run the Flow for Updated Records:** Only when a record is updated to meet the condition requirements
- **Optimize the Flow**
    - Actions and Related Records

Then click the **X** button to close the configuration panel.



Under **Start** element, click the + button and add the **Order Confirmation Email Alert** element.



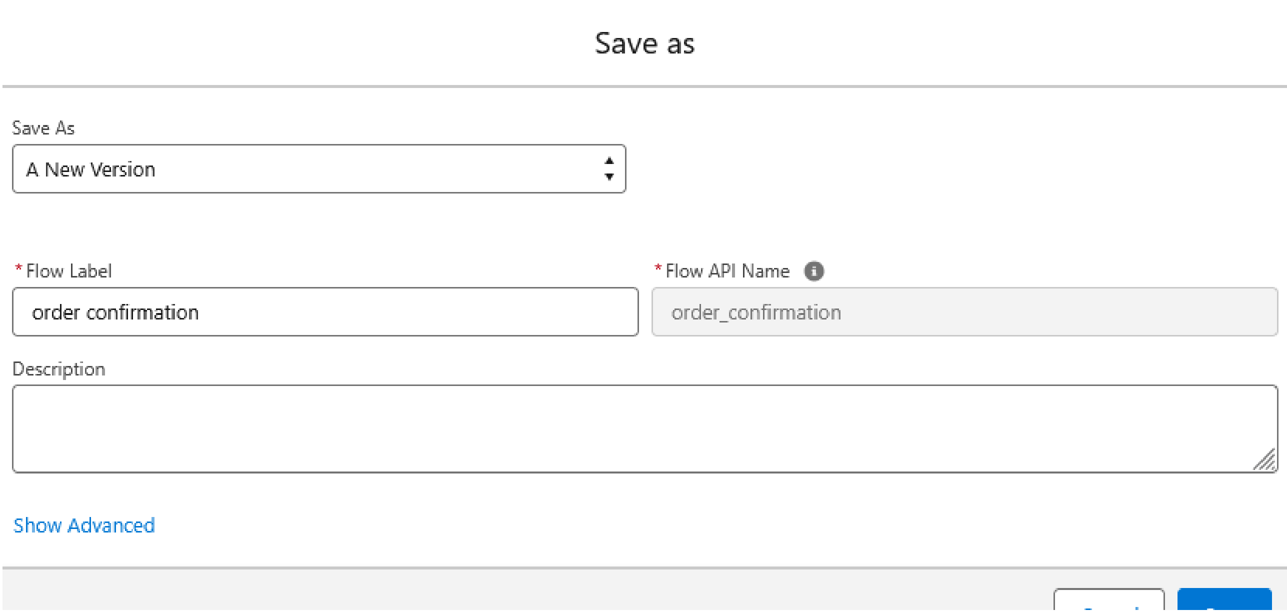
Configure the **Order Confirmation Email Alert** element as follows:

- **Label:** order confirmation
- **API Name:** order_confirmation
- **Description:** Optional

Set Input Values:

- **Record ID:** Triggering HandsMen_Order_c > Record ID

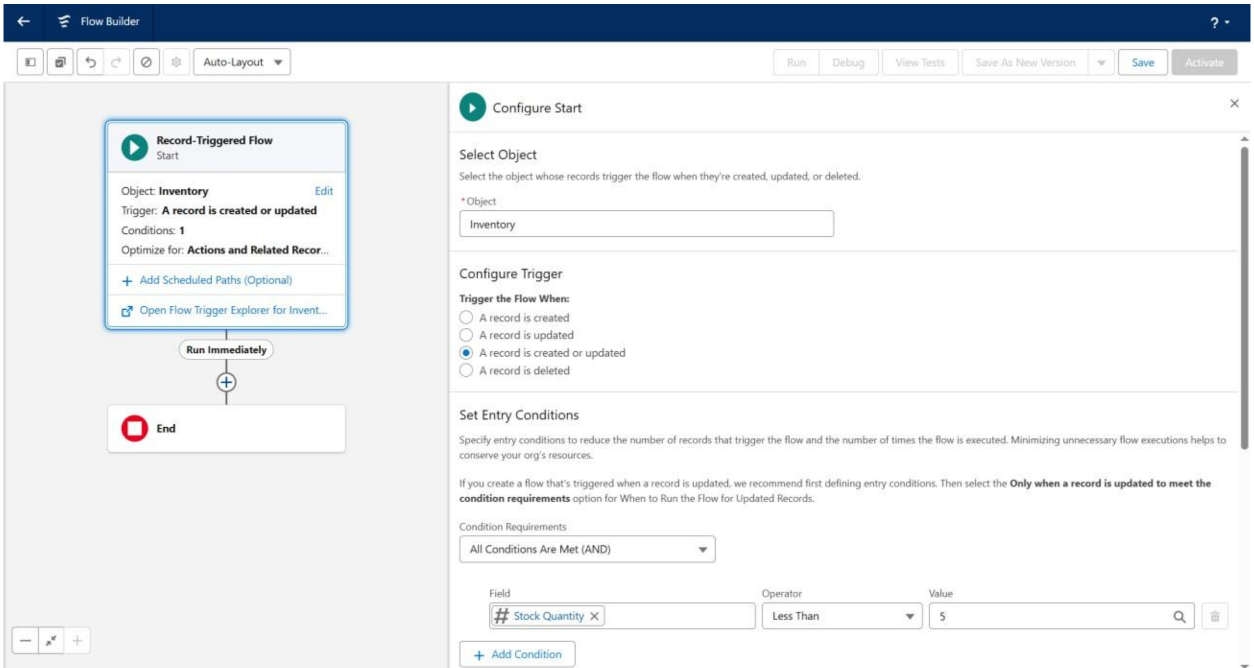
Then click the **X** button to close the configuration panel.



Click the **Save** button and enter the following:

- **Flow Label:** order confirmation
- **Flow API Name:** order_confirmation

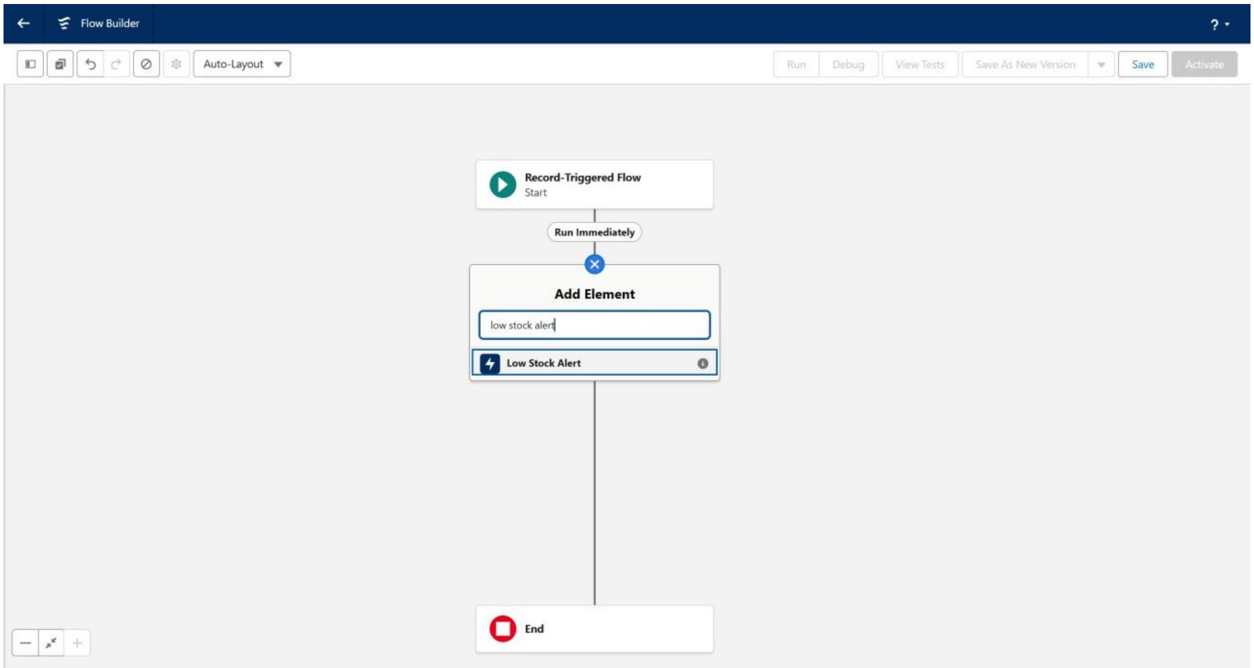
Click the **Save** button, then click **Activate** to enable the flow.



Create a new **Record-Triggered Flow** and configure it as follows:

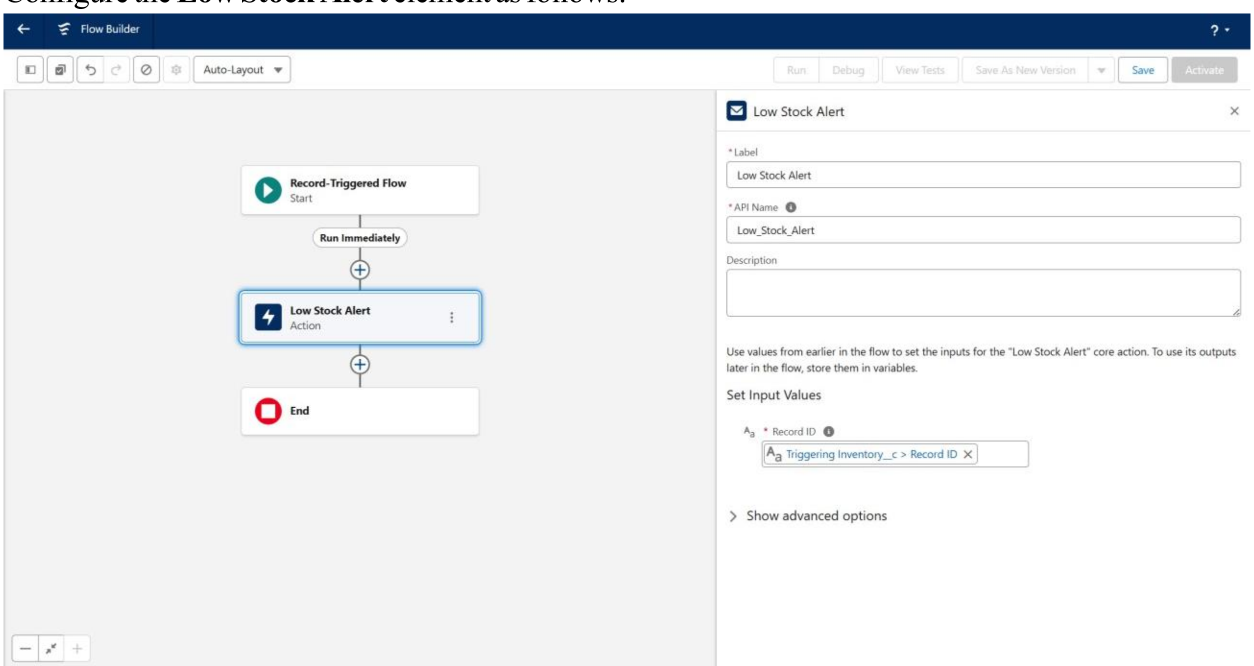
- **Select Object Type**
  - **Object:** Inventory
- **Configure Trigger**
  - **Trigger the Flow When:** A record is created or updated
- **Set Entry Conditions**
  - **Conditions Requirements:** All Conditions Are Met (AND)
    - **Field:** Stock Quantity
    - **Operator:** Less Than
    - **Value:** 5
  - **When to Run the Flow for Updated Records:** Every time a record is updated and meets the condition requirements
- **Optimize the Flow**
  - **Select:** Actions and Related Records

Then click the **X** button to close the configuration panel.



Under **Start** element, click the + button and add the **Low Stock Alert** element.

Configure the **Low Stock Alert** element as follows:

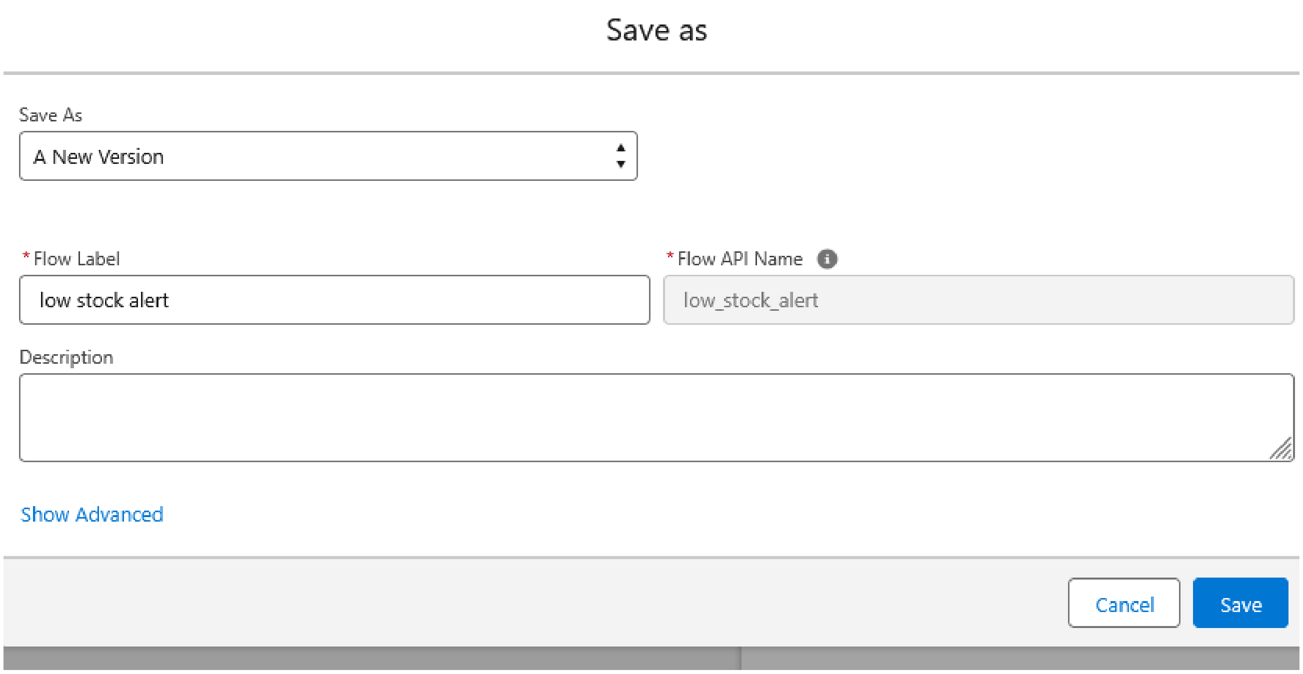


- **Label:** Low Stock Alert
- **API Name:** Low_Stock_Alert
- **Description:** Optional

Set Input Values:

- **Record ID:** Triggering Inventory > Record ID

Then click the **X** button to close the configuration panel.

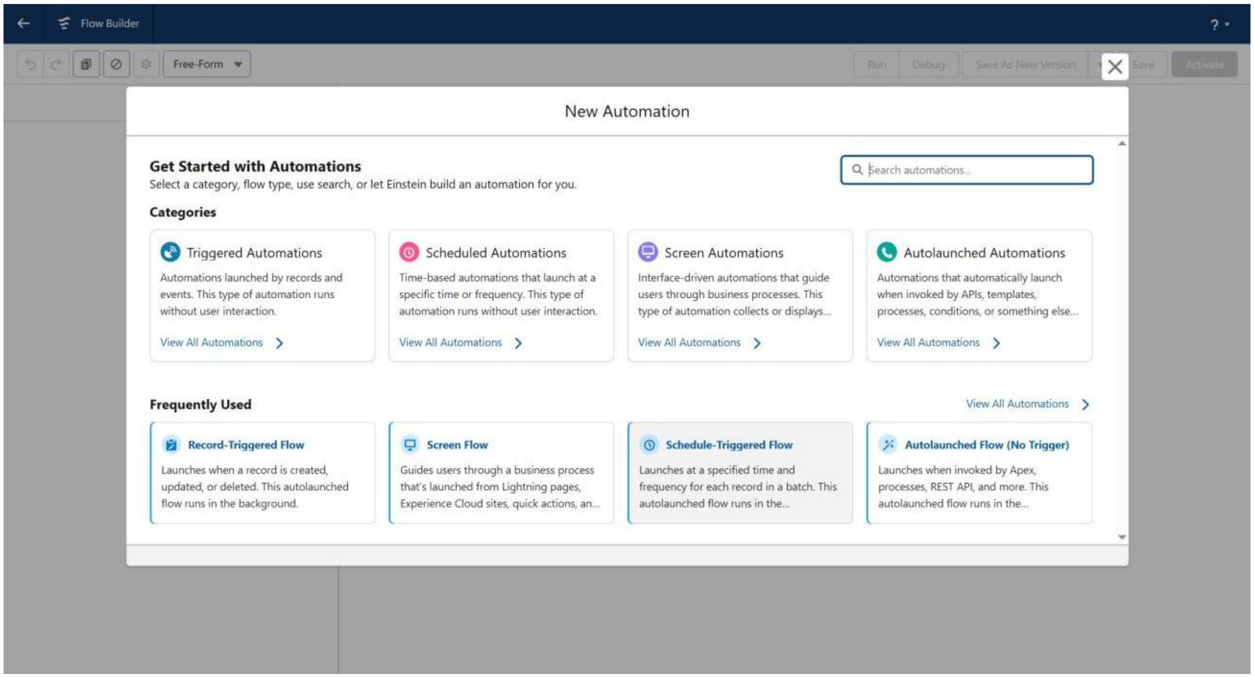


Click the **Save** button and enter the following:

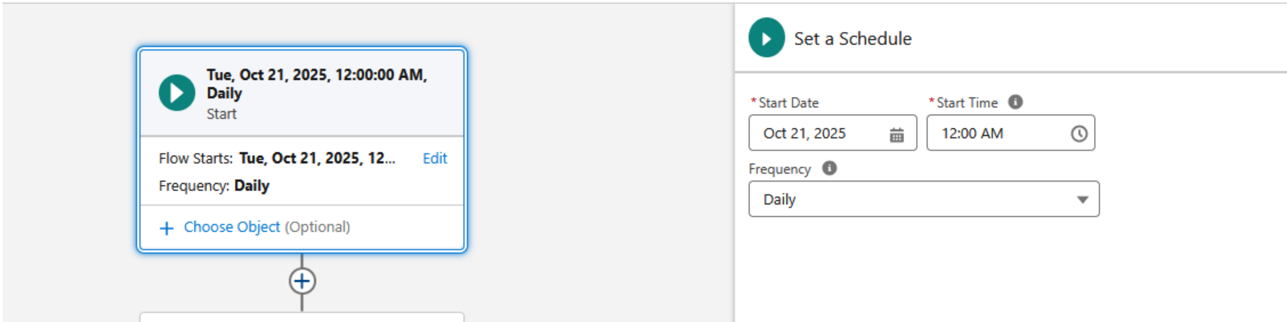
- **Flow Label:** low stock alert
- **Flow API Name:** low_stock_alert

Click the **Save** button, then click **Activate** to enable the flow.

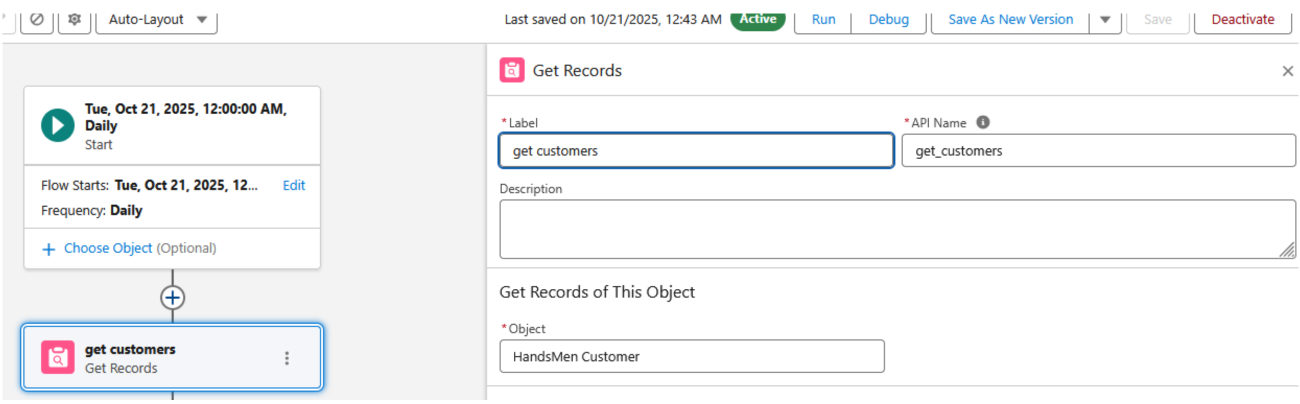
Scheduled Flow:



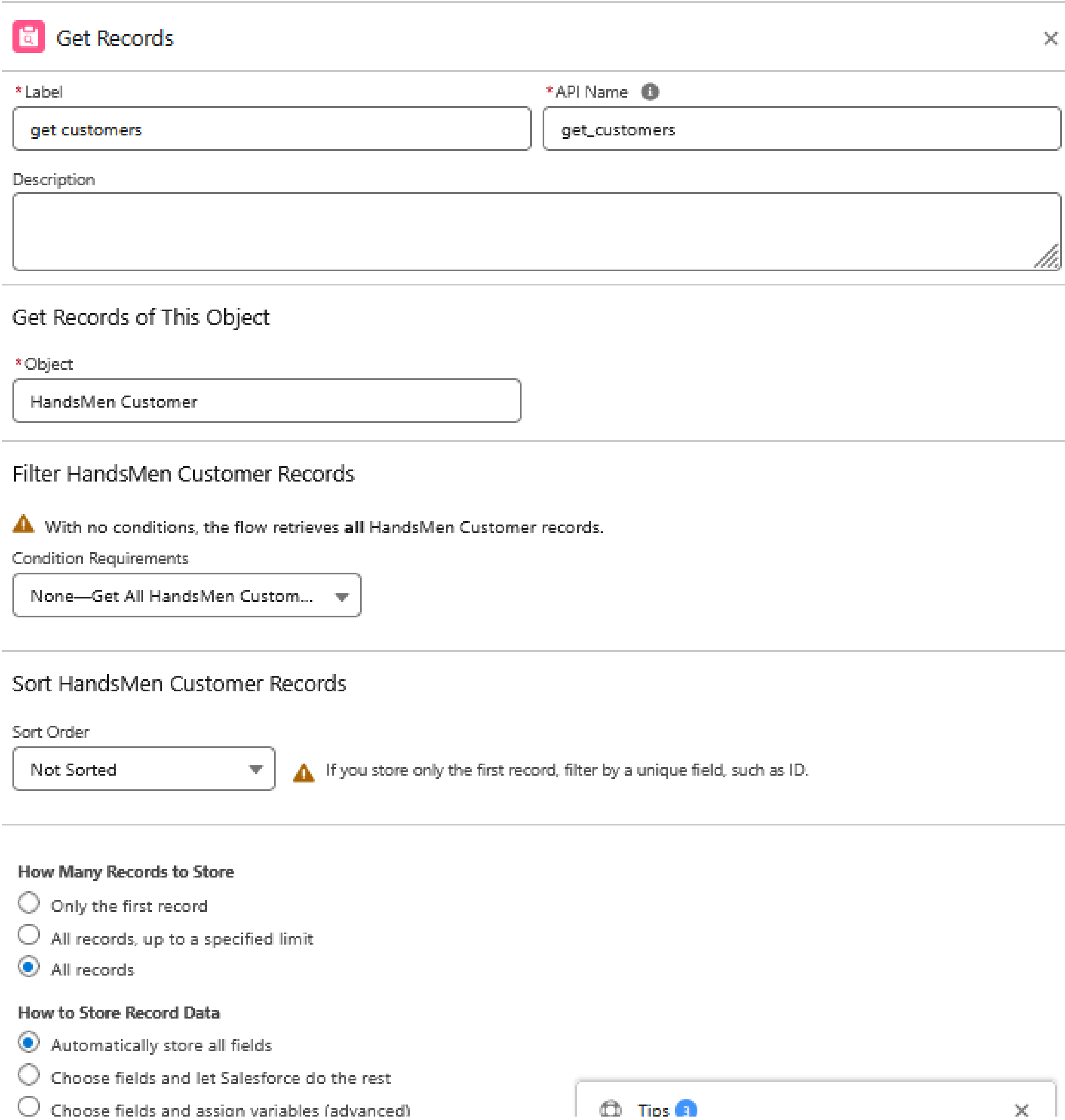
In **Flow Builder**, select **Scheduled-Triggered Flow** as the flow type to begin automating processes based on a scheduled time.



In **Set a Schedule**, you can configure the timing based on your needs. Then click the **X** button to close the configuration panel.



Under **Start** element, click the + button and add the **Get Records** element.



Configure the **Get Records** element as follows:

- **Label:** get customers
- **API Name:** get_customers
- **Description:** Optional

Get Records of This Object:

- **Object:** HandsMen Customer

Filter HandsMen Customer Records:

- **Condition Requirements:** None – Get All HandsMen Customer Records

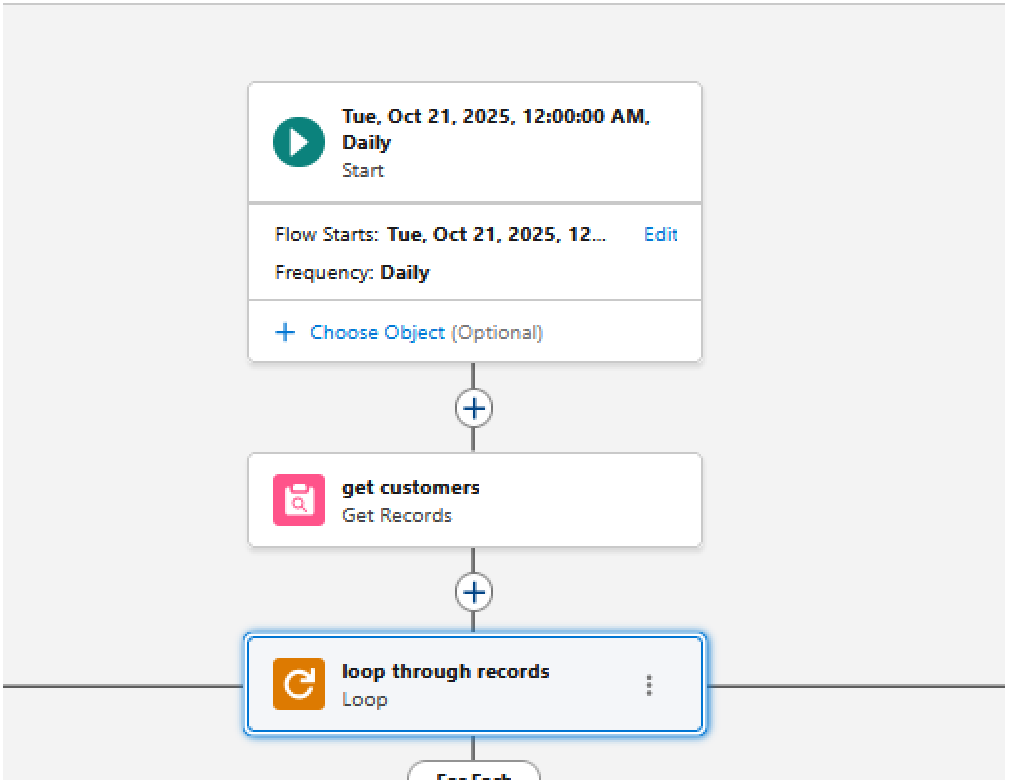
Sort HandsMen Customer Records:

- **Sort Order:** Not Sorted

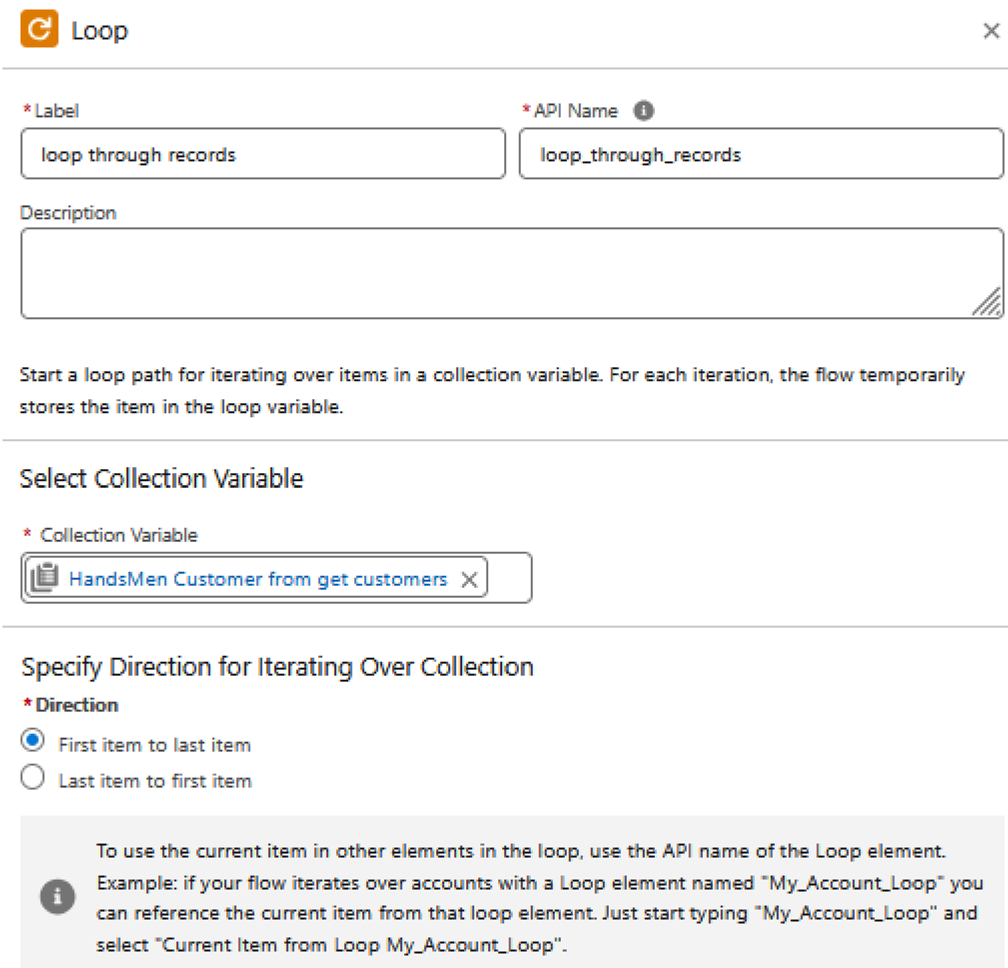
How Many Records to Store: All Records

How to Store Record Data: Automatically store all fields Then

click the **X** button to close the configuration panel.



Under **Get Records** element, click the + button and add the **Loop** element.



Configure the **Loop** element as follows:

- **Label:** loop through records
- **API Name:** loop_through_records
- **Description:** Optional

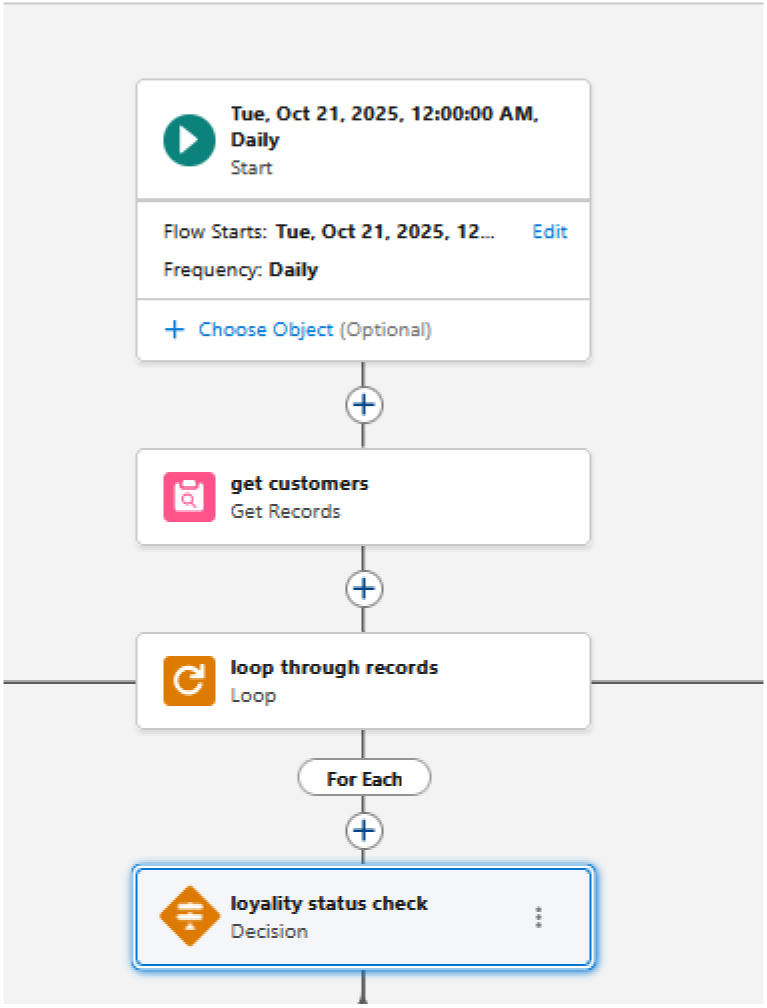
Select Collection Variable:

- **Collection Variable:** HandsMen Customer from Get Customers

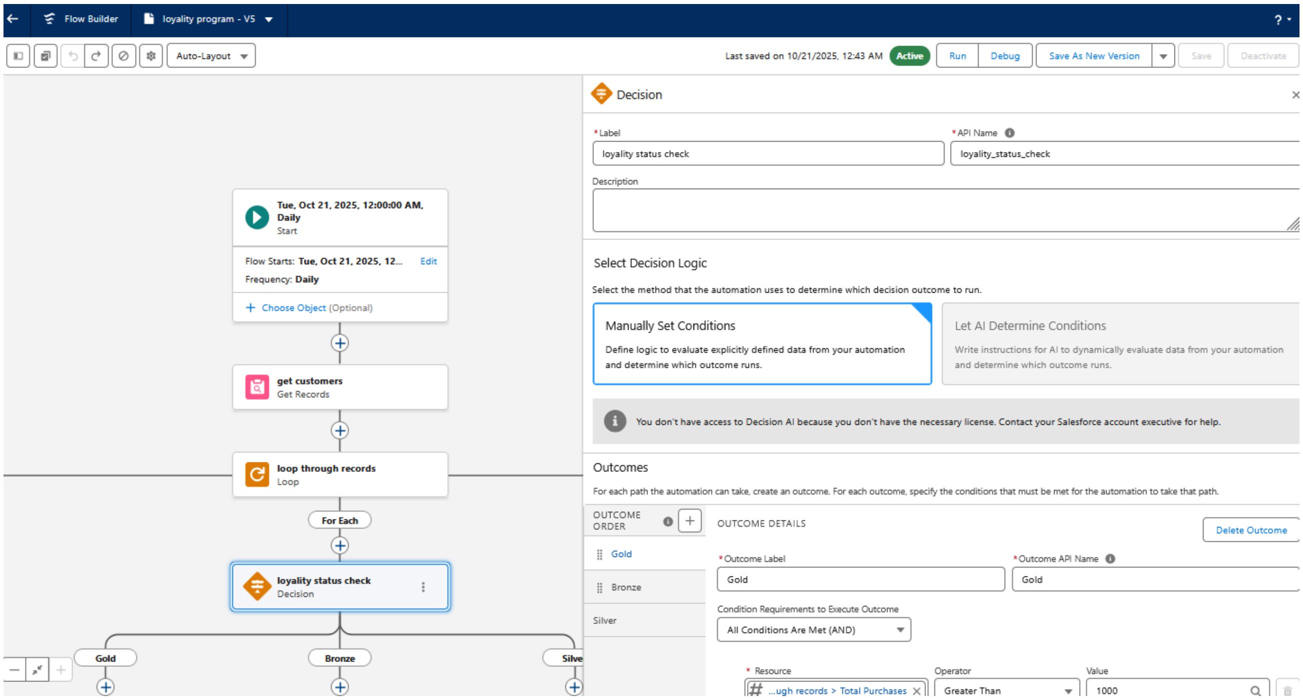
Specify Direction for Iterating Over Collection:

- **Direction:** First item to last item

Then click the **X** button to close the configuration panel.



Under **Loop (For Each)** element, click the + button and add the **Decision** element.



Created By: Antonio Miguel Orias

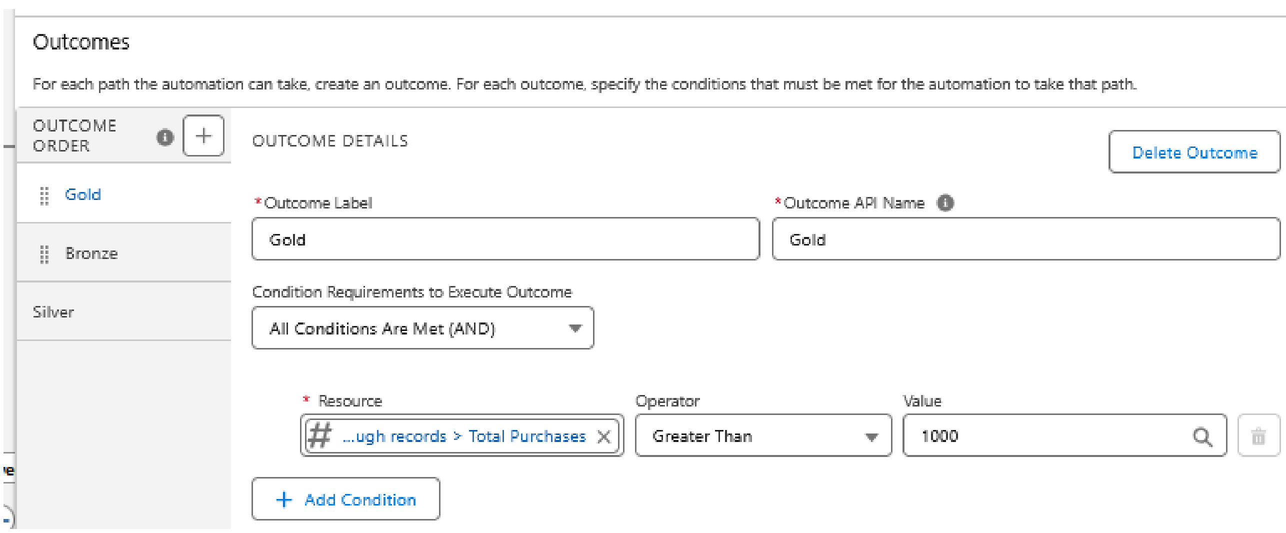
Configure the **Decision** element as follows:

- **Label:** loyalty status check
- **API Name:** loyalty_status_check
- **Description:** Optional

Select Decision Logic: Manually Set Conditions

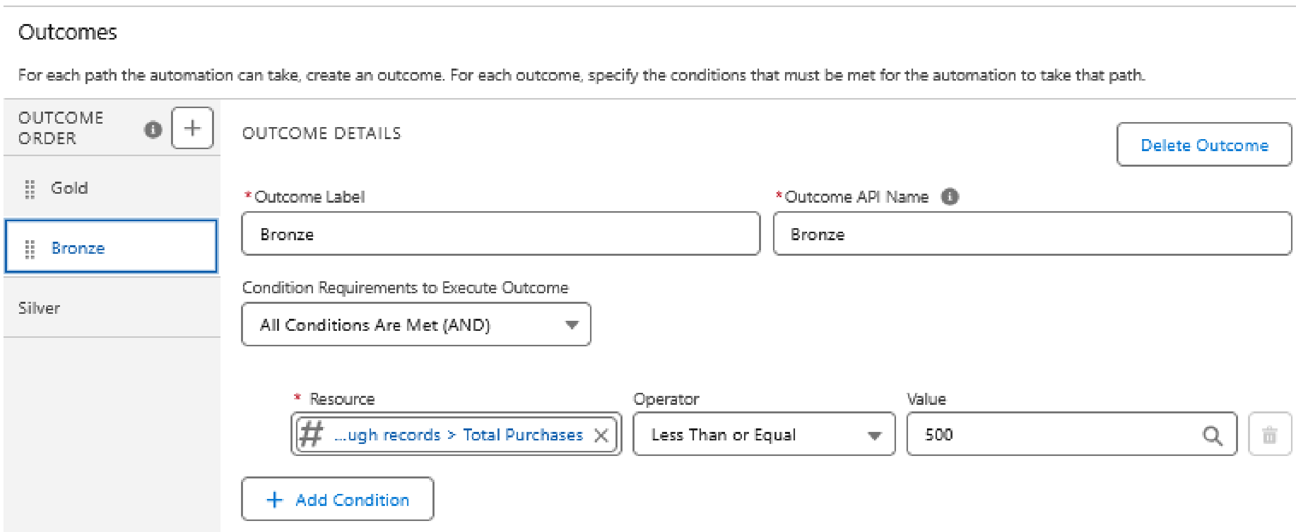
Outcome:

- **Outcome Label:** Gold
- **Outcome API Name:** Gold
- **Condition Requiredments to Execute Outcome:** All Conditions Are Met (AND)
- **Resources:** Current Item from Loop Loop Through Records > Total Purchases
- **Operator:** Greater Than
- **Value:** 1000

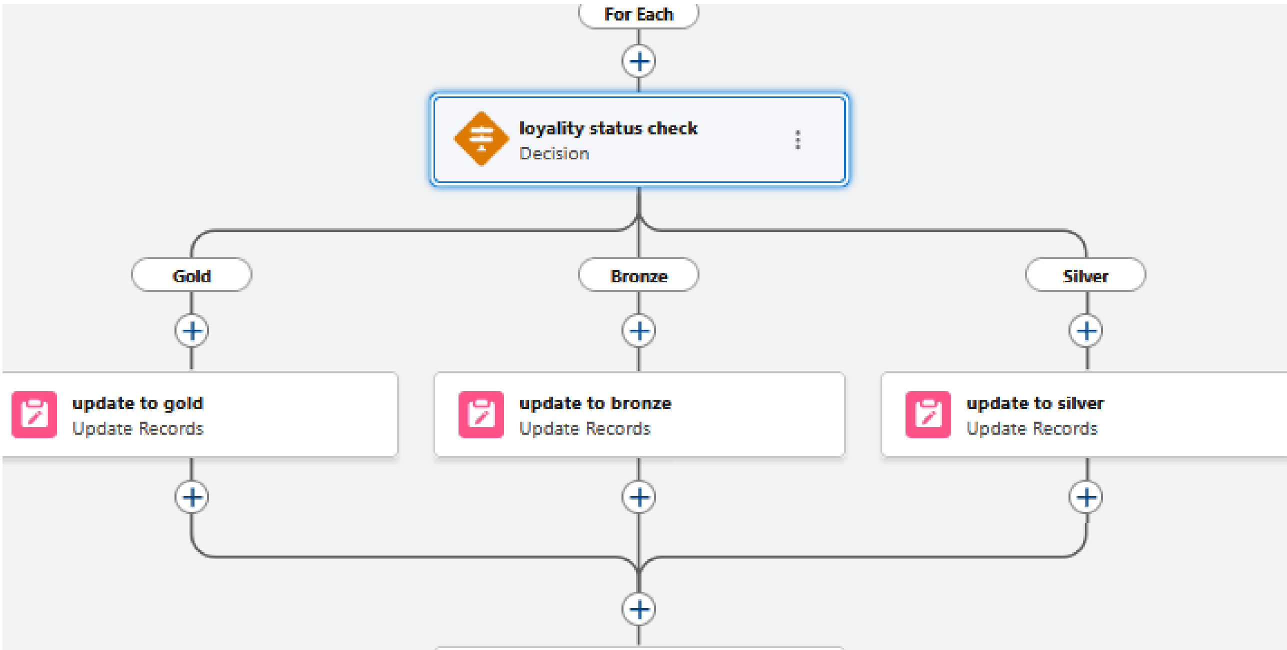


Add another outcome:

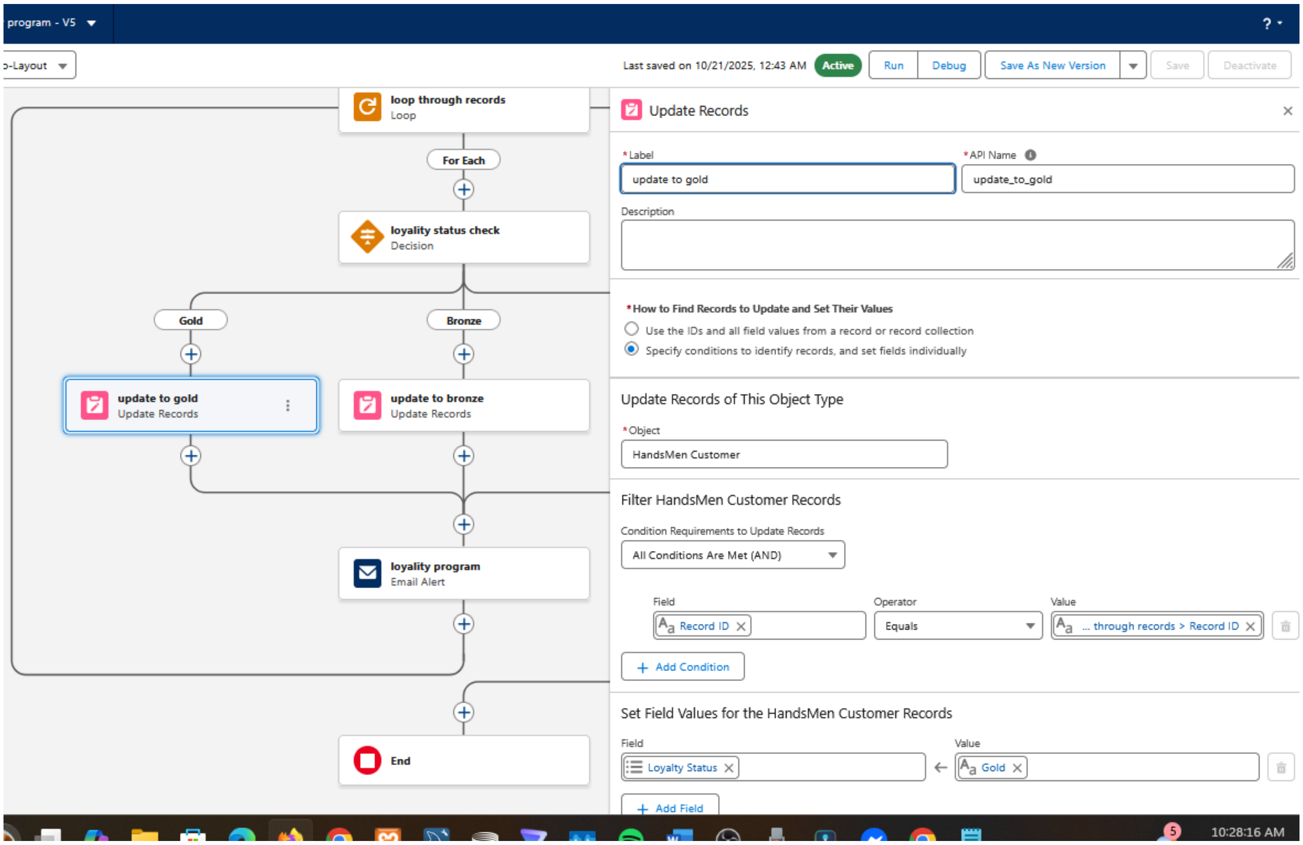
- **Outcome Label:** Bronze
- **Outcome API Name:** Bronze
- **Condition Requiredments to Execute Outcome:** All Conditions Are Met (AND)
- **Resources:** Current Item from Loop Loop Through Records > Total Purchases
- **Operator:** Less Than or Equal
- **Value:** 500



Set the **Default Outcome** to **Silver**. Then click the **X** button to close the panel.



Under **Decision (Gold)** element, click the + button and add the **Update Records** element.



Configure the **Update Records** element as follows:

- **Label:** update to gold

- **API Name:** update_to_gold
- **Description:** Optional

How to Find Records to Update and Set Ther Values: Specify conditions to identify records, and set fields individually

Update Records of This Object Type:

- **Object:** HandsMen Customer

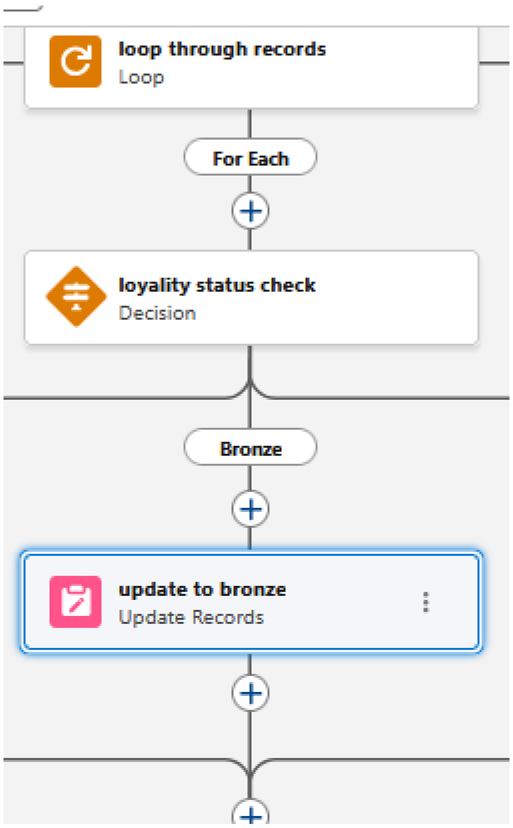
Filter HandsMen Customer Records:

- **Conditions Requirements to Update Records:** All Conditions Are Met (AND)
- **Field:** Records ID
- **Operator:** Equals
- **Value:** Current Item from Loop Loop Through Records > Record ID

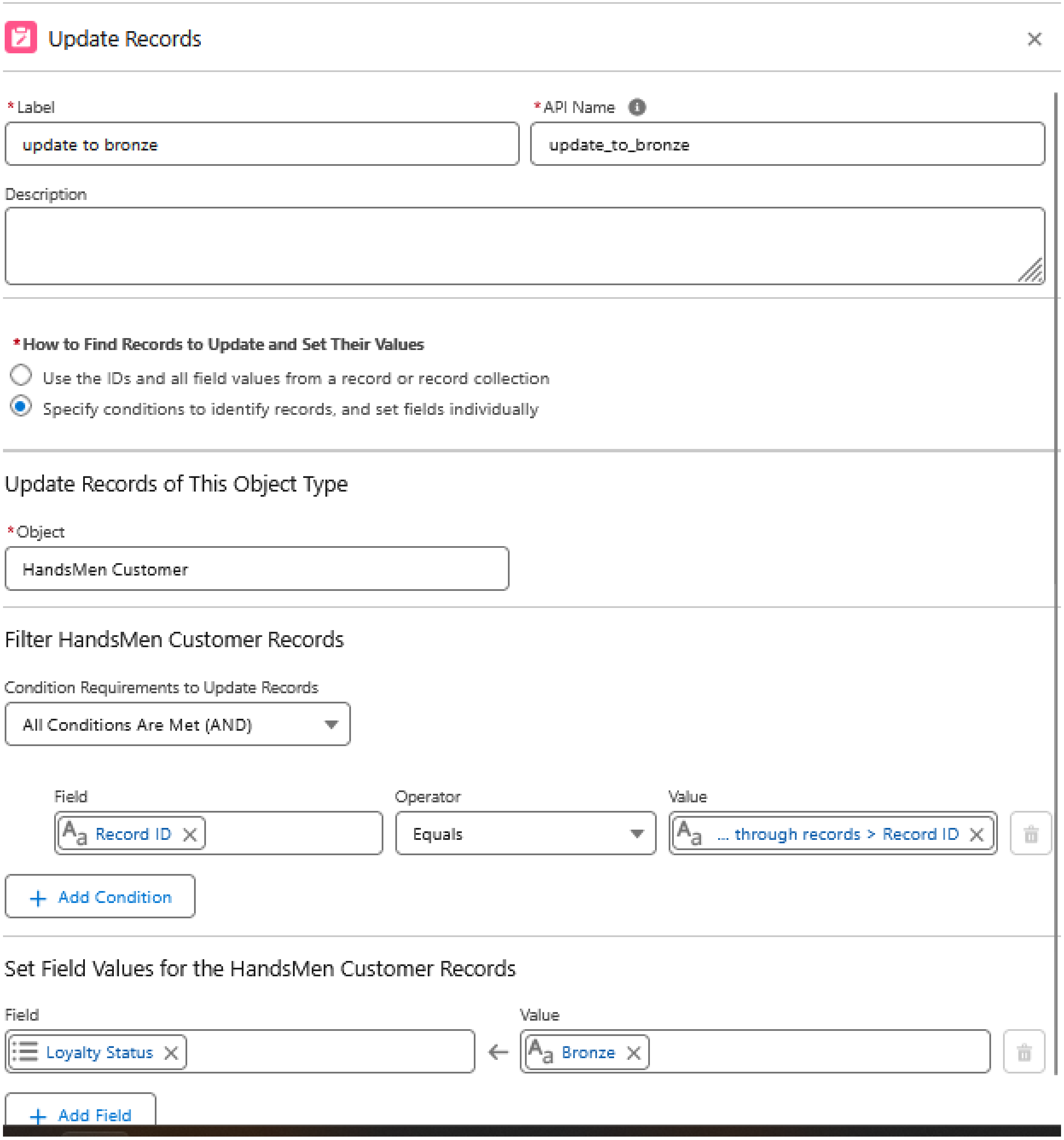
Set Field Value for the HandsMen Customer Records:

- **Field:** Loyalty Status
- **Value:** Gold

Then click the **X** button to close the configuration panel.



Under **Decision (Bronze)** element, click the + button and add the **Update Records** element.


Update Records
*Label
update to bronze
*API Name
update_to_bronze
Description
*How to Find Records to Update and Set Their Values
Use the IDs and all field values from a record or record collection
Specify conditions to identify records, and set fields individually
Update Records of This Object Type
*Object
HandsMen Customer
Filter HandsMen Customer Records
Condition Requirements to Update Records
All Conditions Are Met (AND)
Field
Record ID
Operator
Equals
Value
... through records > Record ID
Add Condition
Set Field Values for the HandsMen Customer Records
Field
Loyalty Status
Value
Bronze
Add Field

Configure the **Update Records** element as follows:

- **Label:** update to bronze
- **API Name:** update_to_bronze
- **Description:** Optional

How to Find Records to Update and Set Ther Values: Specify conditions to identify records, and set fields individually

Update Records of This Object Type:

- **Object:** HandsMen Customer

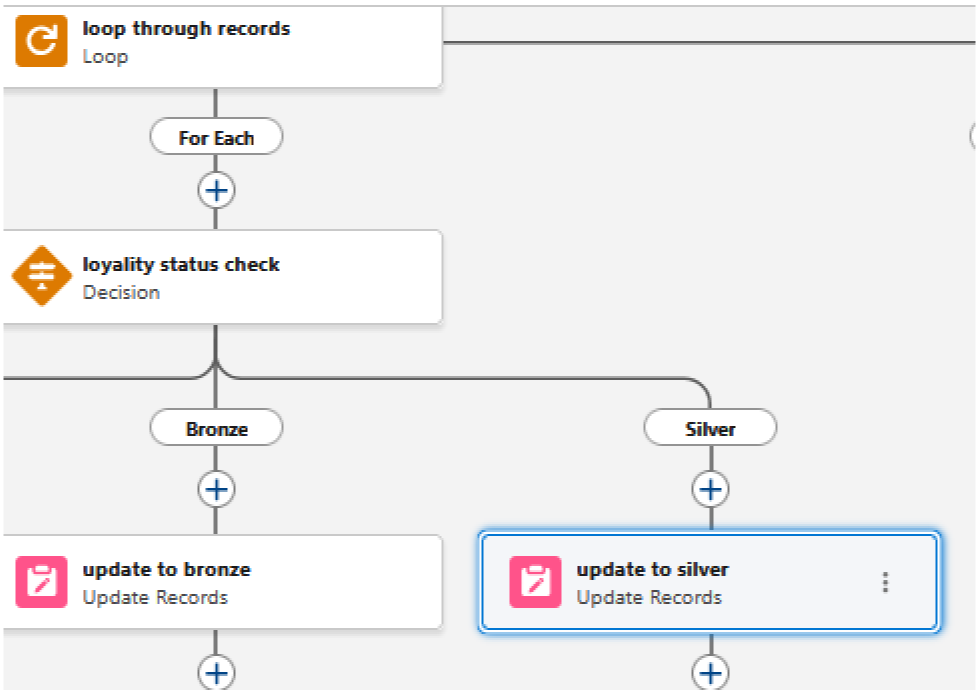
Filter HandsMen Customer Records:

- **Conditions Requirements to Update Records:** All Conditions Are Met (AND)
- **Field:** Records ID
- **Operator:** Equals
- **Value:** Current Item from Loop Loop Through Records > Record ID

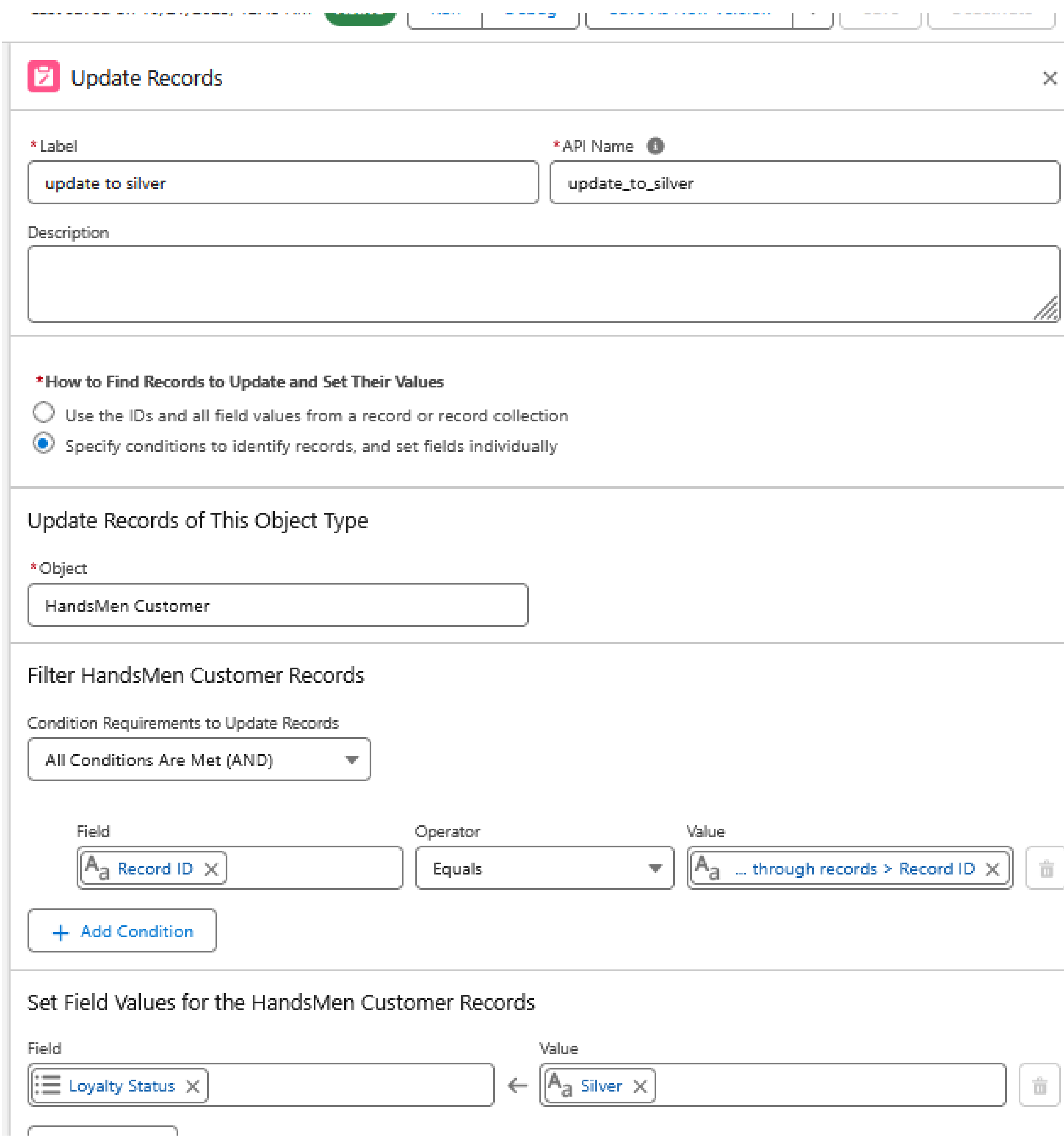
Set Field Value for the HandsMen Customer Records:

- **Field:** Loyalty Status
- **Value:** Bronze

Then click the **X** button to close the configuration panel.



Under **Decision (Silver)** element, click the + button and add the **Update Records** element.



Configure the **Update Records** element as follows:

- **Label:** update to silver

- **API Name:** update_to_silver
- **Description:** Optional

How to Find Records to Update and Set Ther Values: Specify conditions to identify records, and set fields individually

Update Records of This Object Type:

- **Object:** HandsMen Customer

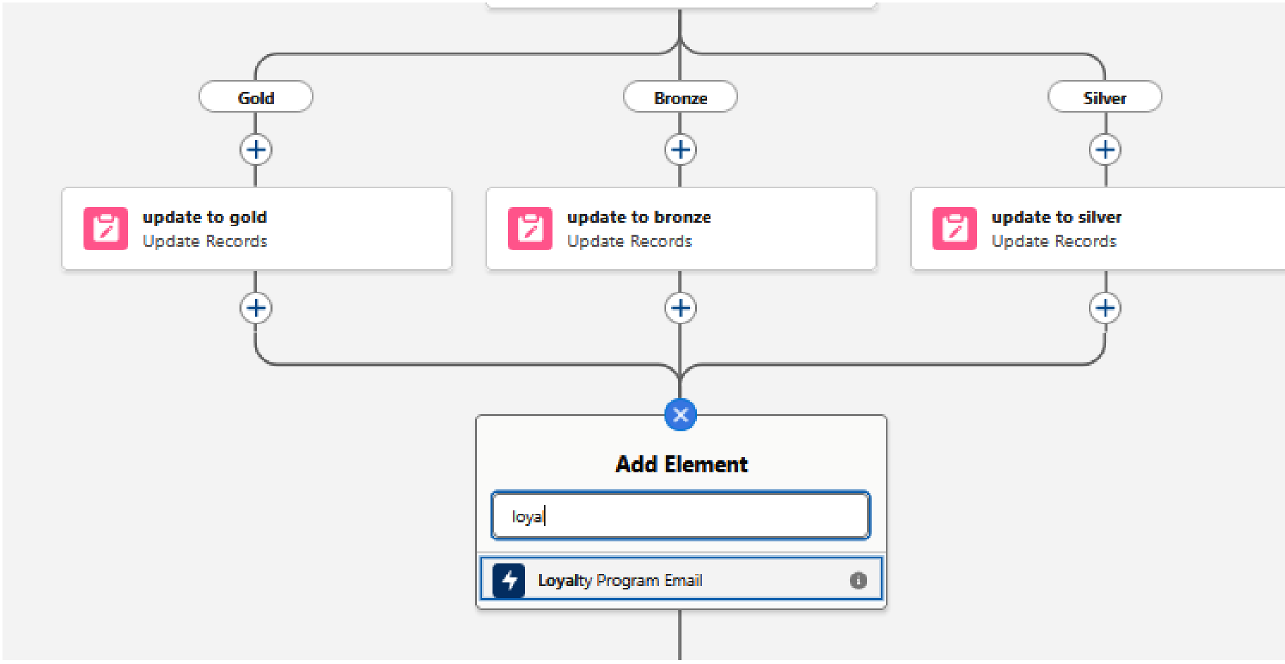
Filter HandsMen Customer Records:

- **Conditions Requirements to Update Records:** All Conditions Are Met (AND)
- **Field:** Records ID
- **Operator:** Equals
- **Value:** Current Item from Loop Loop Through Records > Record ID

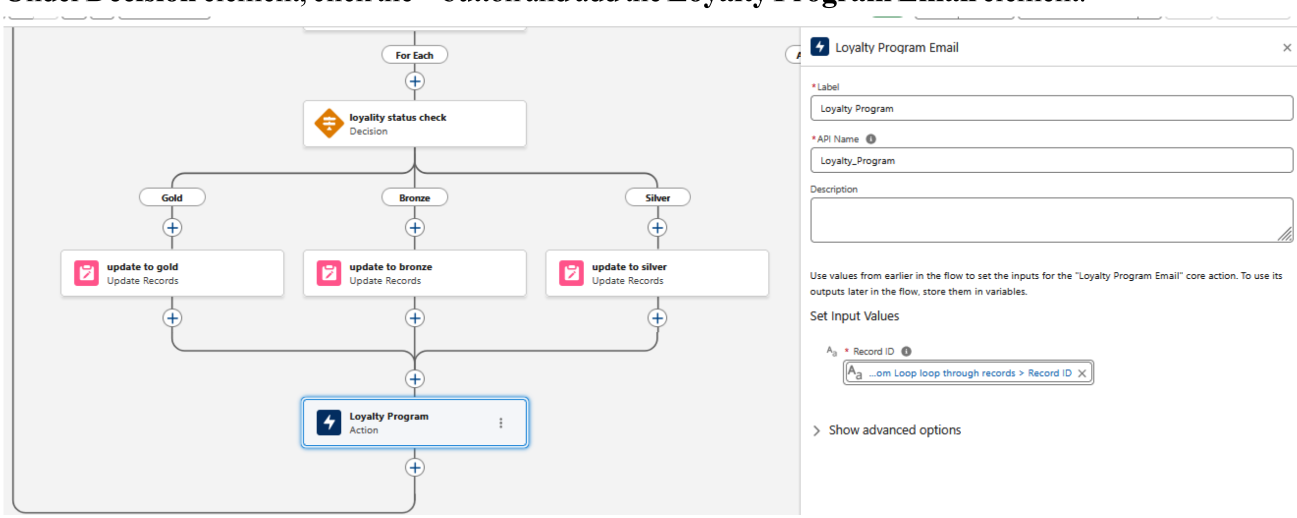
Set Field Value for the HandsMen Customer Records:

- **Field:** Loyalty Status
- **Value:** Silver

Then click the **X** button to close the configuration panel.



Under **Decision** element, click the + button and add the **Loyalty Program Email** element.



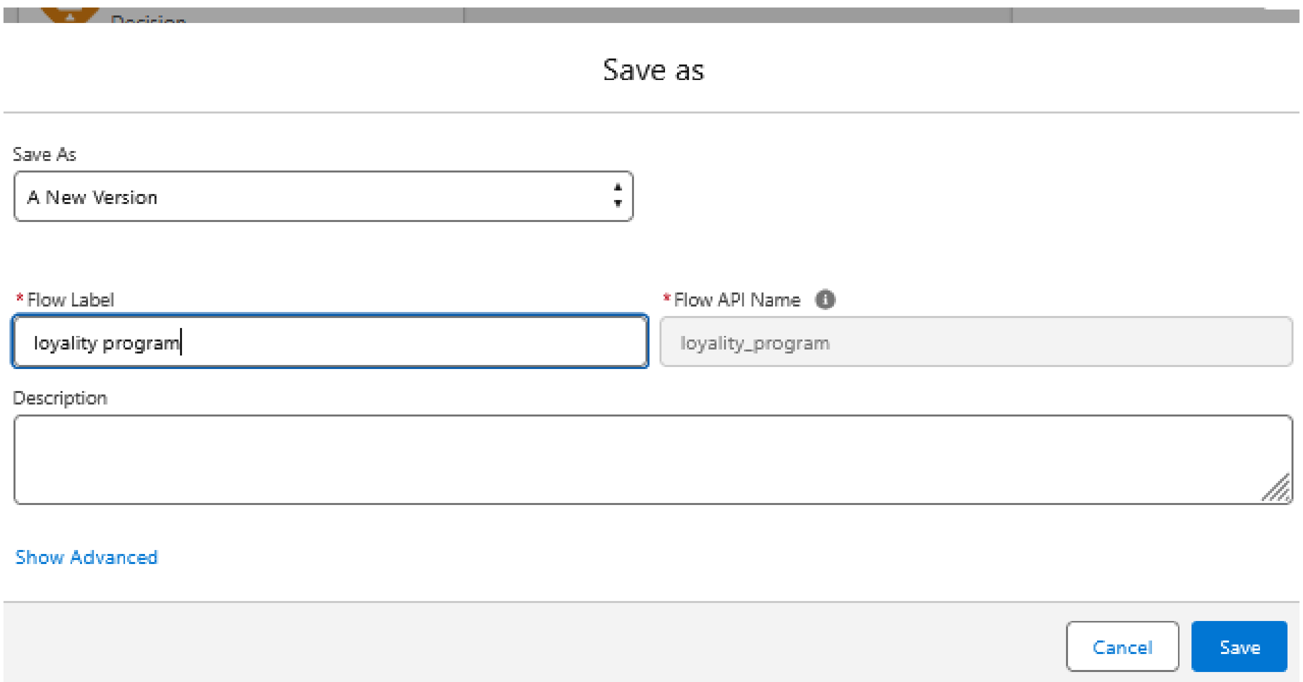
Configure the **Loyalty Program Email** element as follows:

- **Label:** Loyalty Program
- **API Name:** Loyalty_Program
- **Description:** Optional

Set Input Values:

- **Record ID:** Current Item from Loop Loop Through Records > Record ID

Then click the **X** button to close the configuration panel.

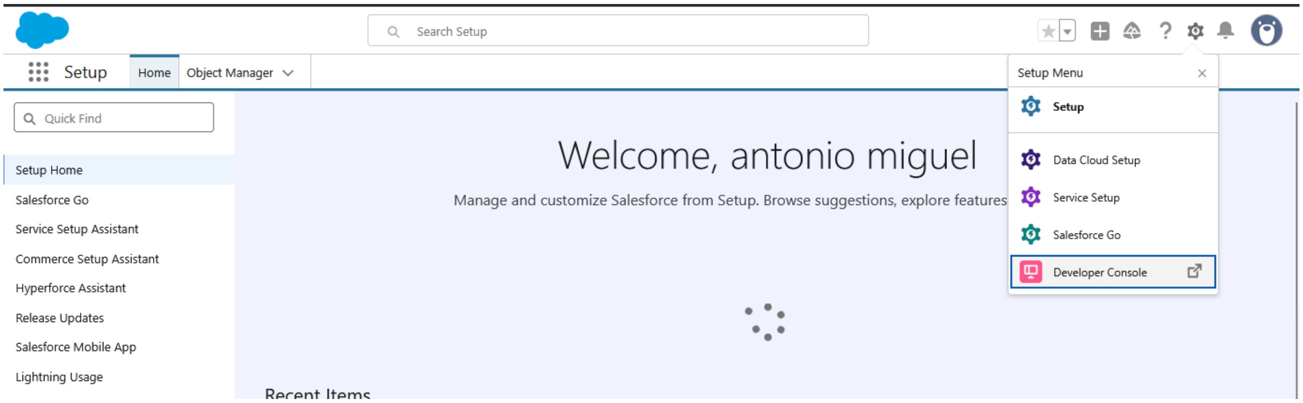


Click the **Save** button and enter the following:

- **Flow Label:** loyality program
- **Flow API Name:** loyality_program

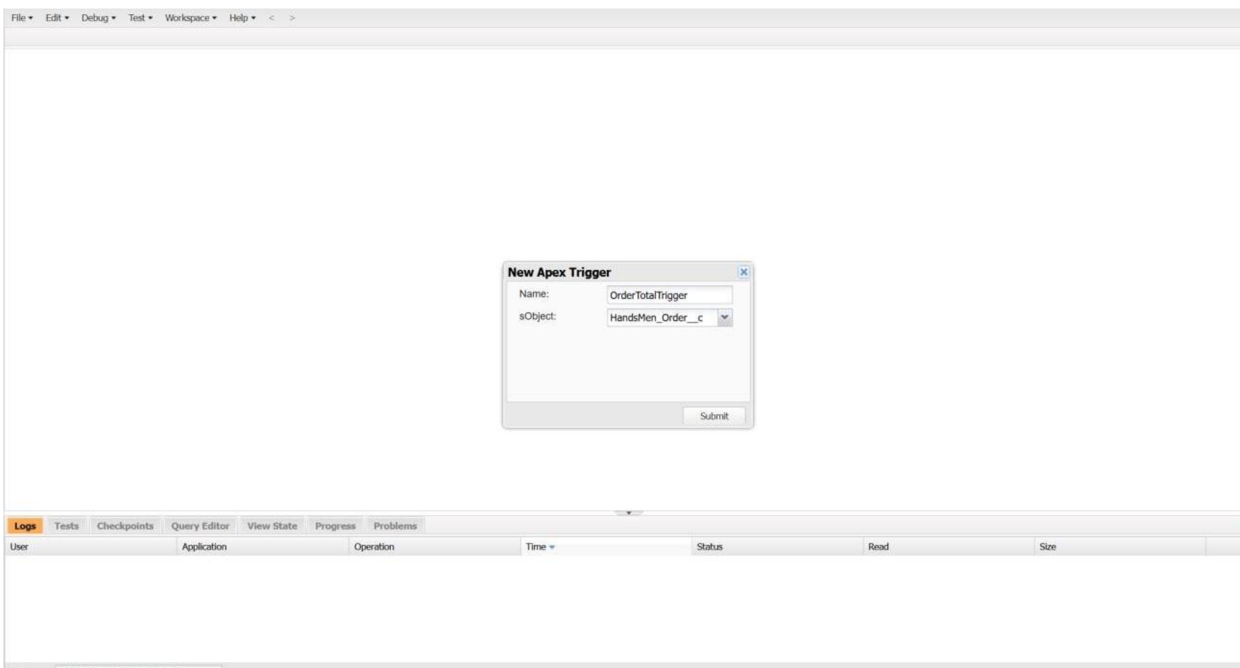
Click the **Save** button, then click **Activate** to enable the flow.

Automation using Apex



In the **Home** tab, click the **Setup** icon, then select **Developer Console**.

- **Order Total Trigger:**



In **Developer Console**, follow these steps:

1. Click **File** > **New** > **Apex Trigger**
2. In the dialog box, enter:
   - **Name**: OrderTotalTrigger

- **sObject**: HandsMen_Order__c

Click **Submit** to create the trigger.

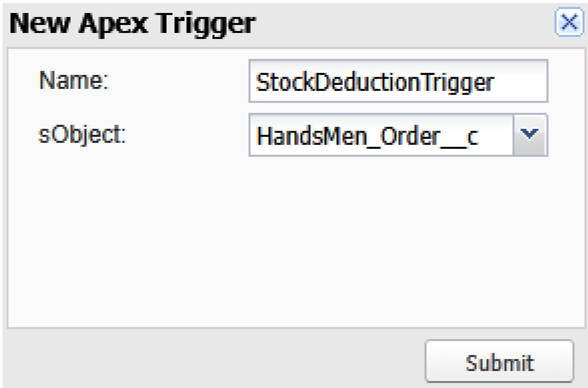
```
trigger OrderTotalTrigger on HandsMen_Order__c (before insert, before update) {
    Set<Id> productIds = new Set<Id>();

    for (HandsMen_Order__c order : Trigger.new) {
        if (order.HandsMen_Product__c != null) {
            productIds.add(order.HandsMen_Product__c);
        }
    }

    Map<Id, HandsMen_Product__c> productMap = new Map<Id, HandsMen_Product__c>(
        [SELECT Id, Price__c FROM HandsMen_Product__c WHERE Id IN :productIds]
    );

    for (HandsMen_Order__c order : Trigger.new) {
        if (order.HandsMen_Product__c != null && productMap.containsKey(order.HandsMen_Product__c)) {
            HandsMen_Product__c product = productMap.get(order.HandsMen_Product__c);
            if (order.Quantity__c != null) {
                order.Total_Amount__c = order.Quantity__c * product.Price__c;
            }
        }
    }
}
```

-

OrderTotalTrigger.apxt (CODE): After Adding the code save it and run.

```
trigger OrderTotalTrigger on HandsMen_Order__c (before insert, before update) {
  Set<Id> productIds = new Set<Id>();

  for (HandsMen_Order__c order : Trigger.new) {
    if (order.HandsMen_Product__c != null) {
      productIds.add(order.HandsMen_Product__c);
    }
  }

  Map<Id, HandsMen_Product__c> productMap = new Map<Id, HandsMen_Product__c>(
    [SELECT Id, Price__c FROM HandsMen_Product__c WHERE Id IN :productIds]
  );

  for (HandsMen_Order__c order : Trigger.new) {
    if (order.HandsMen_Product__c != null && productMap.containsKey(order.HandsMen_Product__c))
{
```

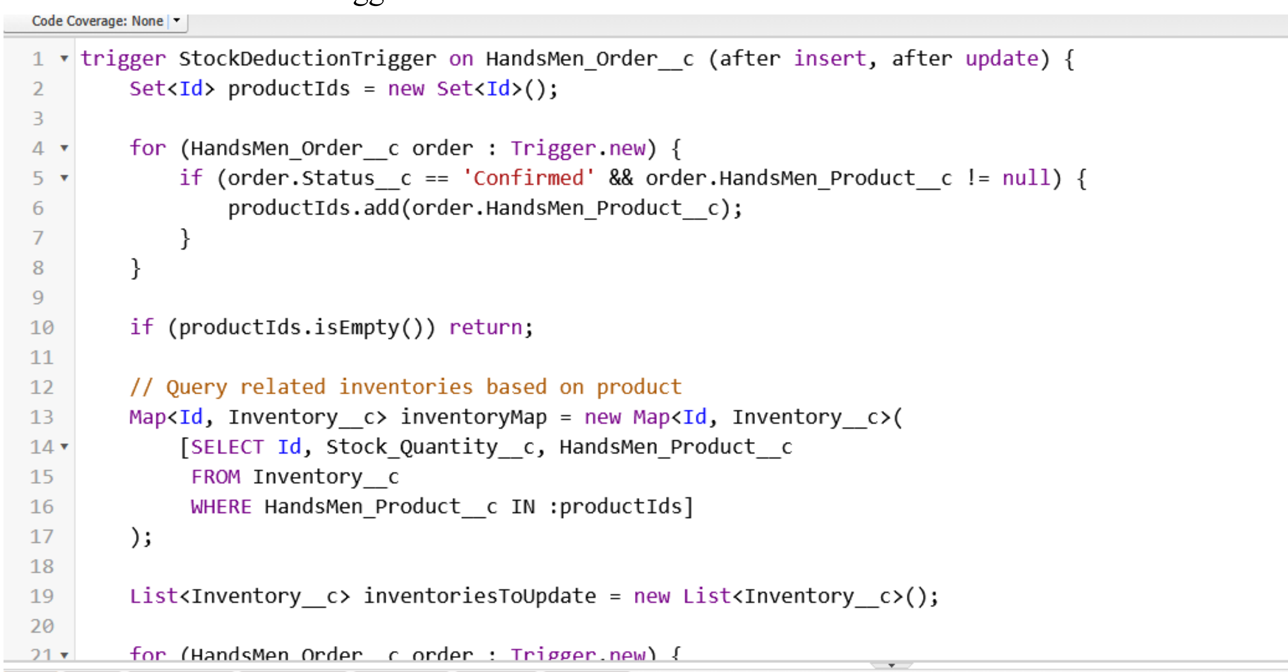
```
            HandsMen_Product__c product = productMap.get(order.HandsMen_Product__c);
            if (order.Quantity__c != null) {
                order.Total_Amount__c = order.Quantity__c * product.Price__c;
            }
        }
    }
}
```



To create a new Apex Trigger in **Developer Console**, follow these steps:

1. Click **File** > **New** > **Apex Trigger**
2. In the dialog box, enter:
    - **Name**: StockDeductionTrigger
    - **sObject**: *HandsMen_Order__c*

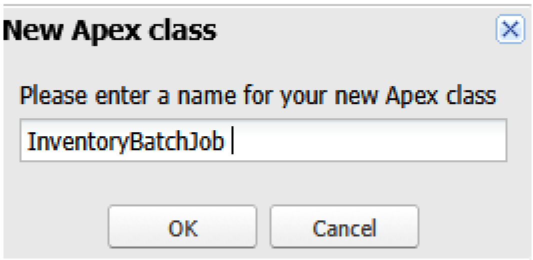
Click **Submit** to create the trigger.



StockDeductionTrigger.apxt (CODE): After Adding the code save it and run.

```
trigger StockDeductionTrigger on HandsMen_Order__c (after insert, after update) {
    Set<Id> productIds = new Set<Id>();

    for (HandsMen_Order__c order : Trigger.new) {
```

```
        if (order.Status__c == 'Confirmed' && order.HandsMen_Product__c != null) {
            productIds.add(order.HandsMen_Product__c);
        }
    }

    if (productIds.isEmpty()) return;

    // Query related inventories based on product
    Map<Id, Inventory__c> inventoryMap = new Map<Id, Inventory__c>(
        [SELECT Id, Stock_Quantity__c, HandsMen_Product__c
         FROM Inventory__c
         WHERE HandsMen_Product__c IN :productIds]
    );

    List<Inventory__c> inventoriesToUpdate = new List<Inventory__c>();

    for (HandsMen_Order__c order : Trigger.new) {
        if (order.Status__c == 'Confirmed' && order.HandsMen_Product__c != null) {
            for (Inventory__c inv : inventoryMap.values()) {
                if (inv.HandsMen_Product__c == order.HandsMen_Product__c) {
                    inv.Stock_Quantity__c -= order.Quantity__c;
                    inventoriesToUpdate.add(inv);
                    break;
                }
            }
        }
    }

    if (!inventoriesToUpdate.isEmpty()) {
        update inventoriesToUpdate;
    }
}
```



In **Developer Console**, follow these steps:

1. Click **File** > **New** > **Apex Class**
2. In the dialog box, enter:
   - **Name**: InventoryBatchJob

Click **Ok** to create the class.

```
global class InventoryBatchJob implements Database.Batchable<SObject>, Schedulable {
global Database.QueryLocator start(Database.BatchableContext BC) {
return Database.getQueryLocator(
'SELECT Id, Stock_Quantity__c FROM Product__c WHERE Stock_Quantity__c < 5'
);
}
global void execute(Database.BatchableContext BC, List<SObject> records) {
List<HandsMen_Product__c> productsToUpdate = new List<HandsMen_Product__c>();
// Cast SObject list to Product__c list
for (SObject record : records) {
HandsMen_Product__c product = (HandsMen_Product__c) record;
```

InventoryBatchJob.apxc (CODE): After Adding the code save it and run.

```
global class InventoryBatchJob implements Database.Batchable<SObject>, Schedulable {

global Database.QueryLocator start(Database.BatchableContext BC) {

return Database.getQueryLocator(

'SELECT Id, Stock_Quantity__c FROM Product__c WHERE Stock_Quantity__c < 5'

);

}

global void execute(Database.BatchableContext BC, List<SObject> records) {

List<HandsMen_Product__c> productsToUpdate = new List<HandsMen_Product__c>();

// Cast SObject list to Product__c list

for (SObject record : records) {

HandsMen_Product__c product = (HandsMen_Product__c) record;

product.Stock_Quantity__c += 50; // Restock logic

productsToUpdate.add(product);
```

```
}
if (!productsToUpdate.isEmpty()) {
try {
update productsToUpdate;
} catch (DmlException e) {
System.debug('Error updating inventory: ' + e.getMessage());
}
}
}
global void finish(Database.BatchableContext BC) {
System.debug('Inventory Sync Completed');
}
// Scheduler Method
global void execute(SchedulableContext SC) {
InventoryBatchJob batchJob = new InventoryBatchJob();
Database.executeBatch(batchJob, 200);
}
}
```

Enter Apex Code

```
System.schedule('Daily Inventory Sync', '0 0 0 * * ?', new InventoryBatchJob());
```

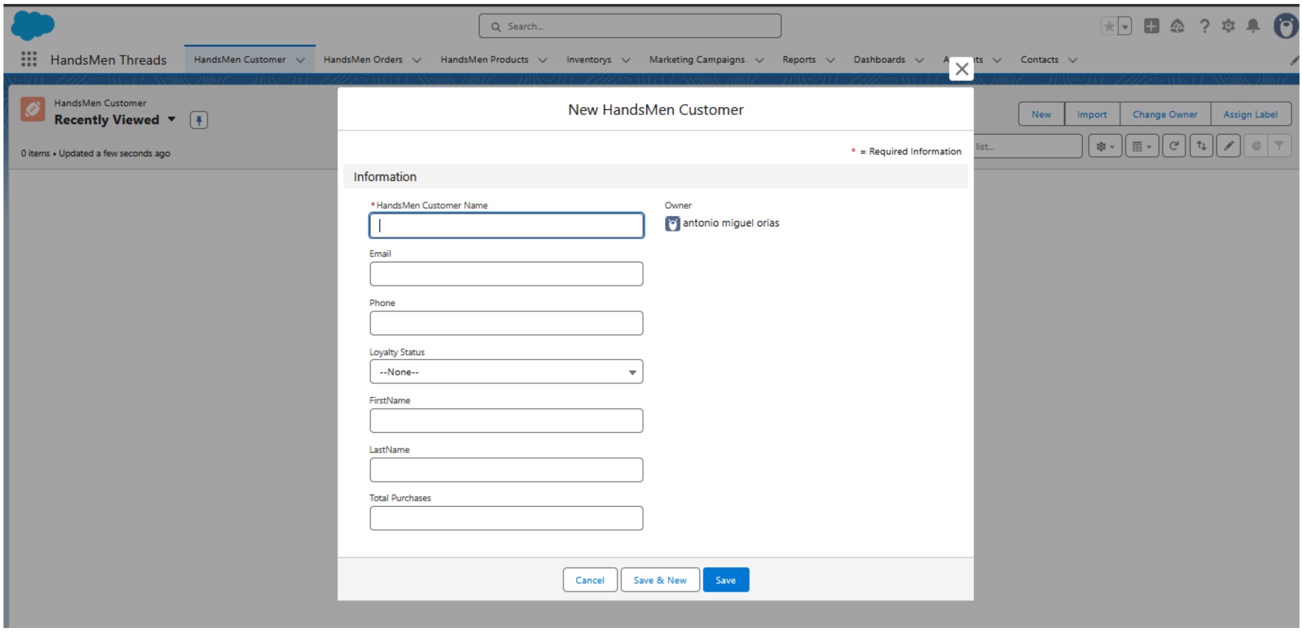
In **Developer Console**, follow these steps:

1. Click **Debug** > **Open Execute Anonymous Window**
2. In the window that appears, paste the following code:

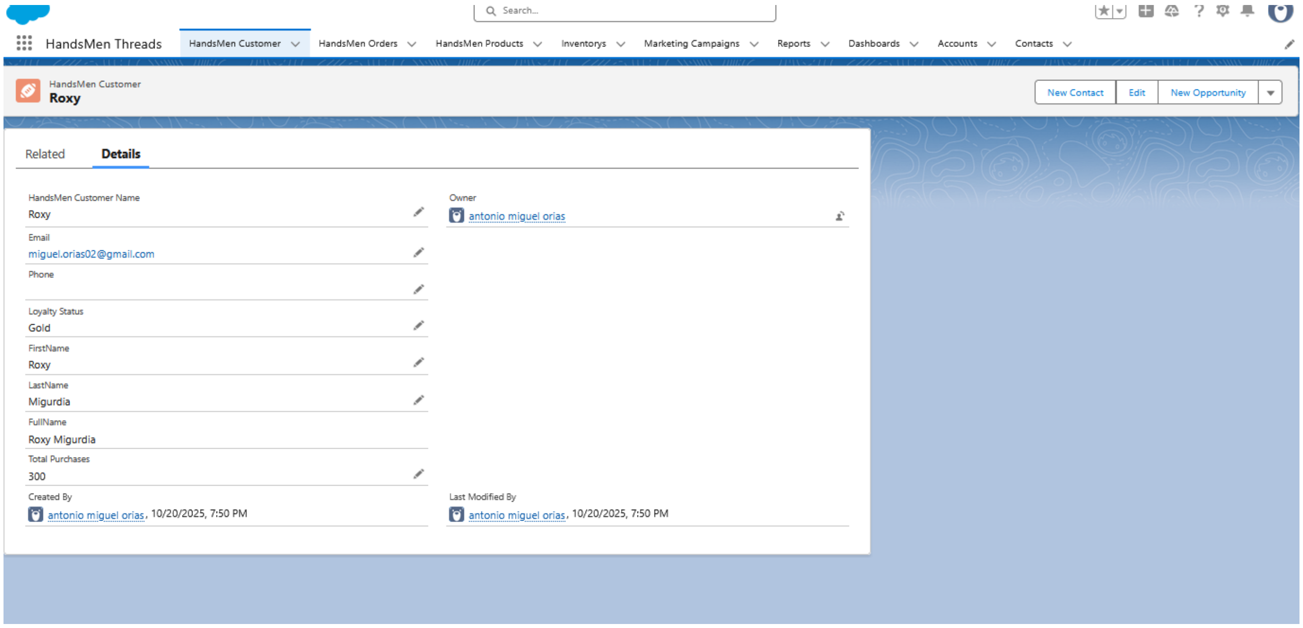
System.schedule('Daily Inventory Sync', '0 0 0 * * ?', new InventoryBatchJob());
Click **Execute** to schedule the batch job.

Testing

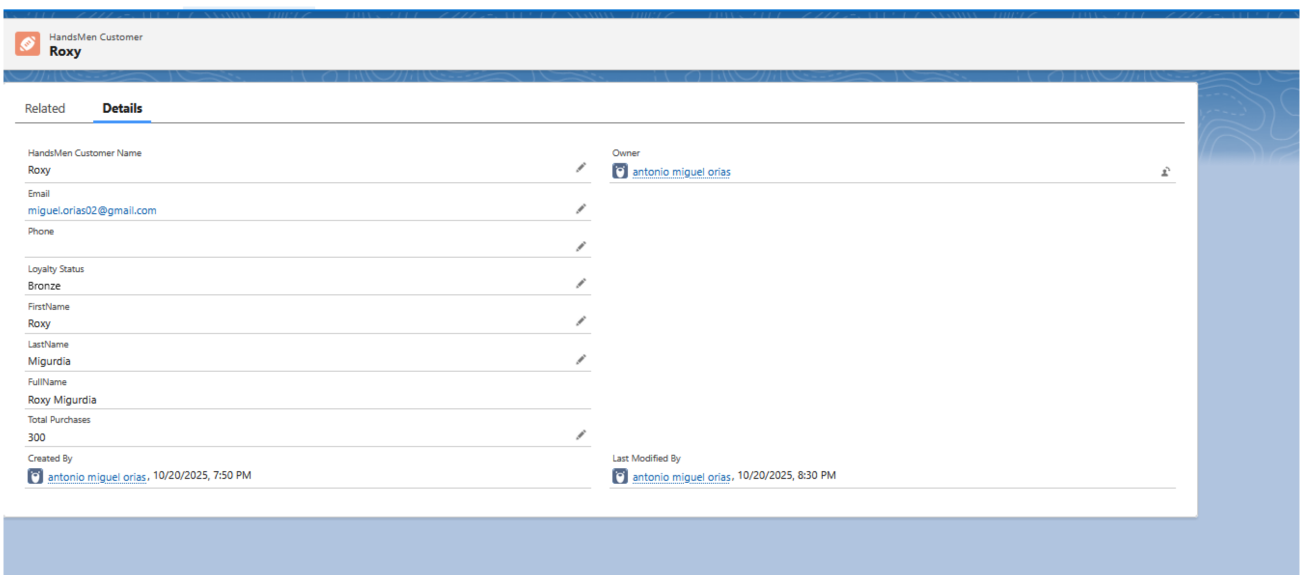
HandsMen Customer:

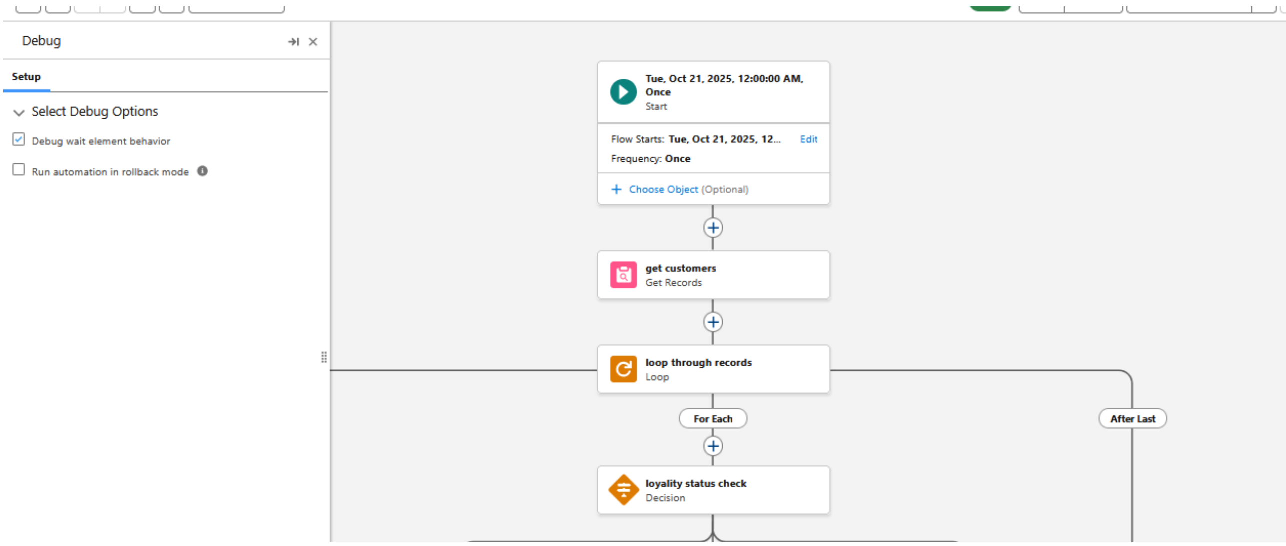


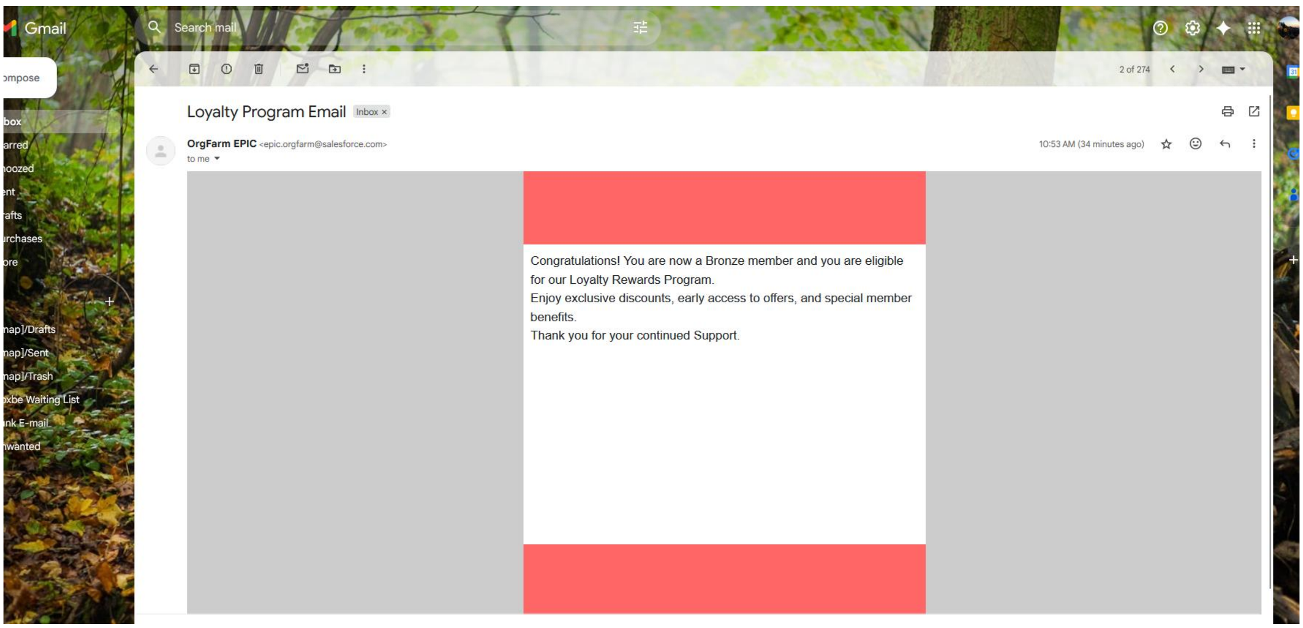
Check if the fields **HandsMen Customer Name**, **Email**, **Phone, Loyalty Status**, **FirstName**, **LastName**, and **Total Purchases** are displayed when adding a customer.



Test if the **Full Name** field correctly combines the values from the **FirstName** and **LastName** fields.

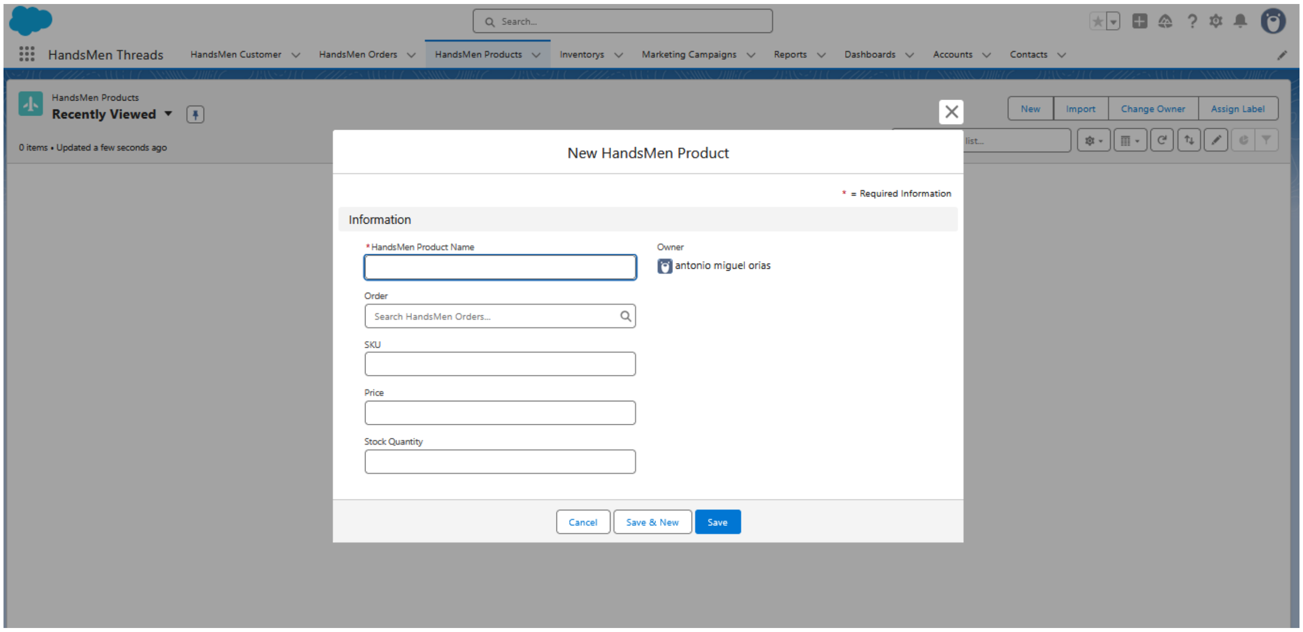






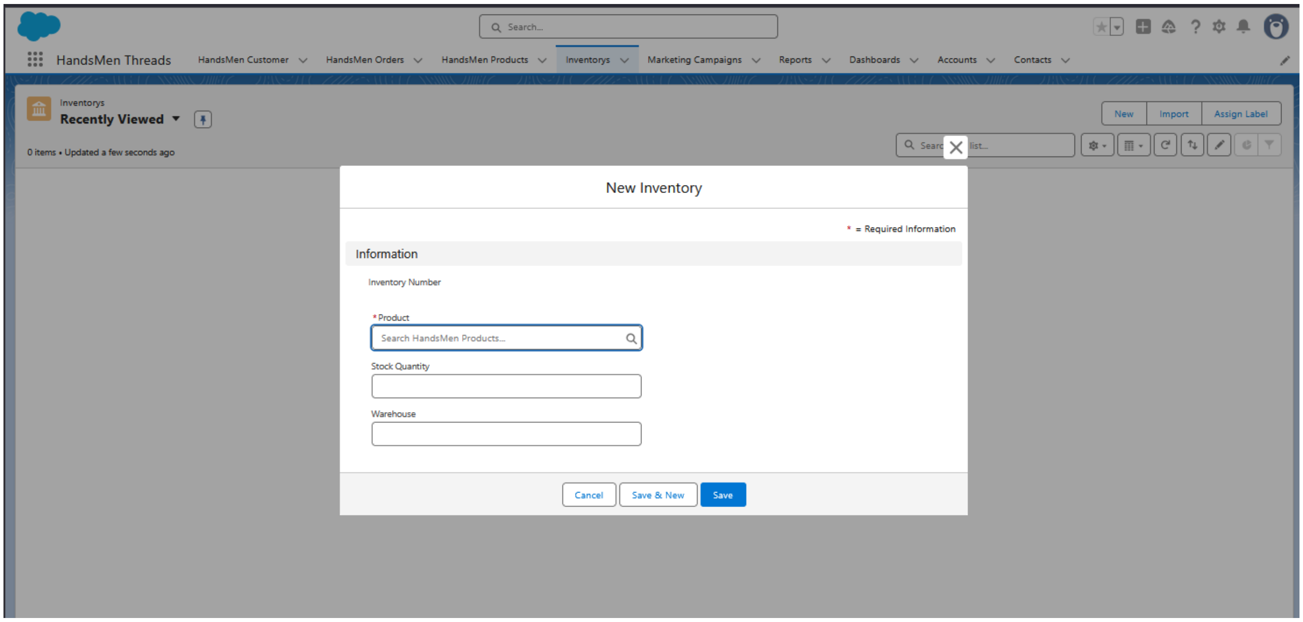
Test if the **Loyalty Status** is changing based on **Total Purchases** and check if you received the email notification. Use debug in Loyalty Program flow.

HandsMen Product:

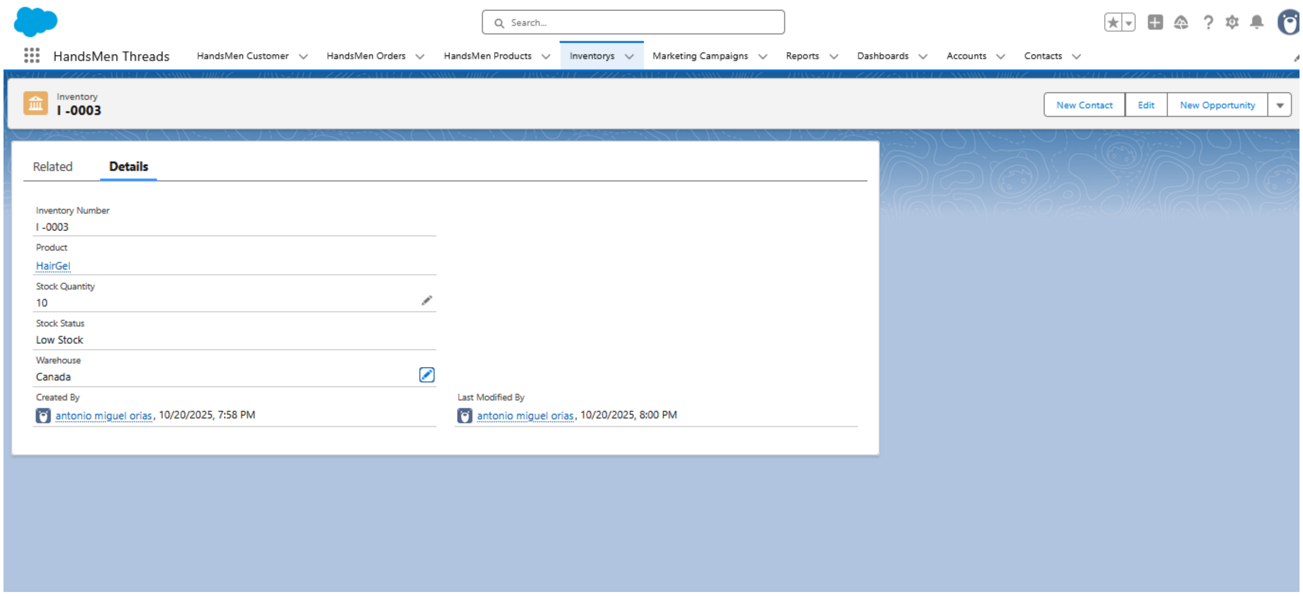


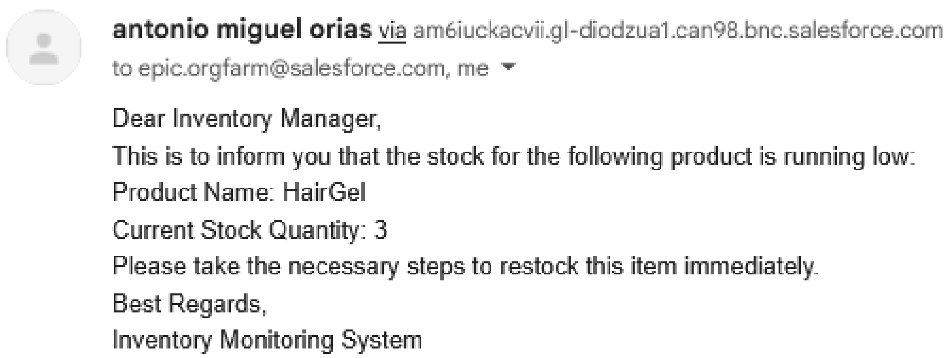
Check if the fields **HandsMen Product Name**, **Order**, **SKU**, **Price**, **Stock Quantity** are displayed when adding a product.

Inventory:



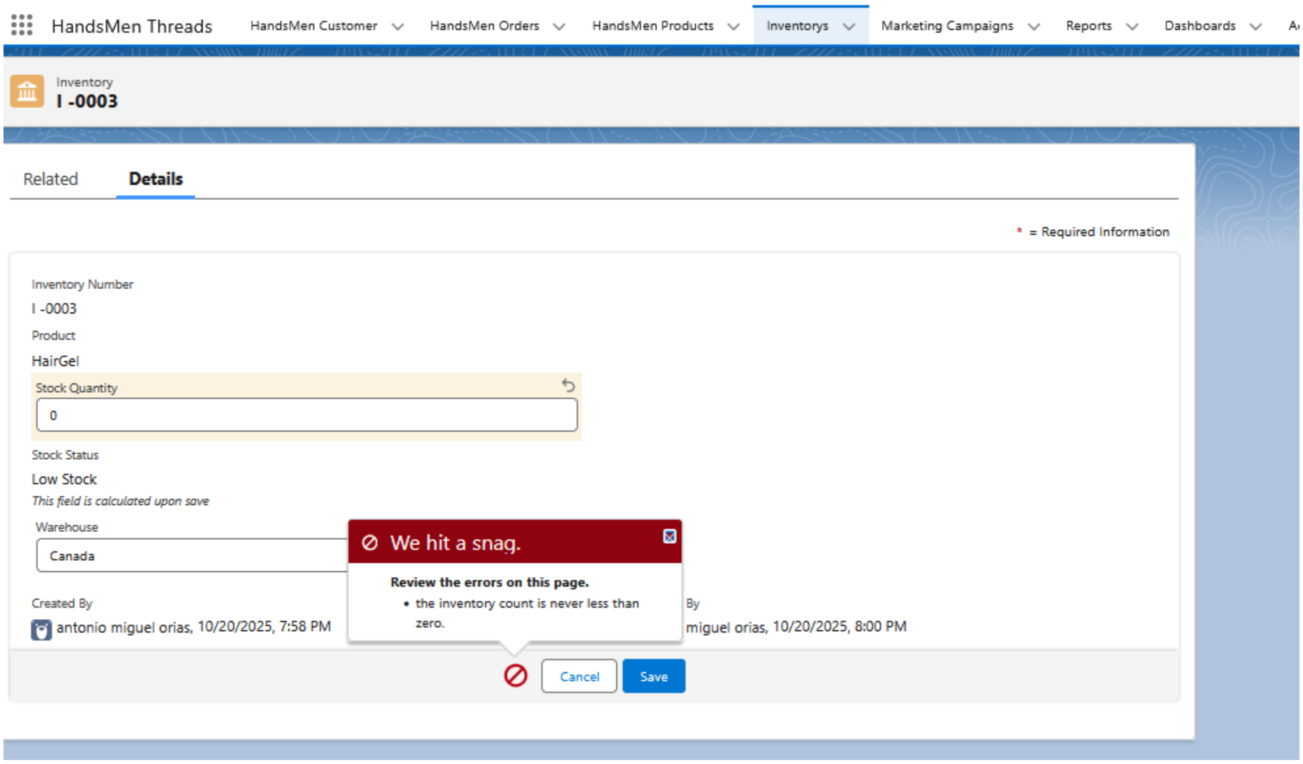
Check if the fields **Product**, **Stock Quantity**, and **Warehous** are displayed when adding an Inventory.





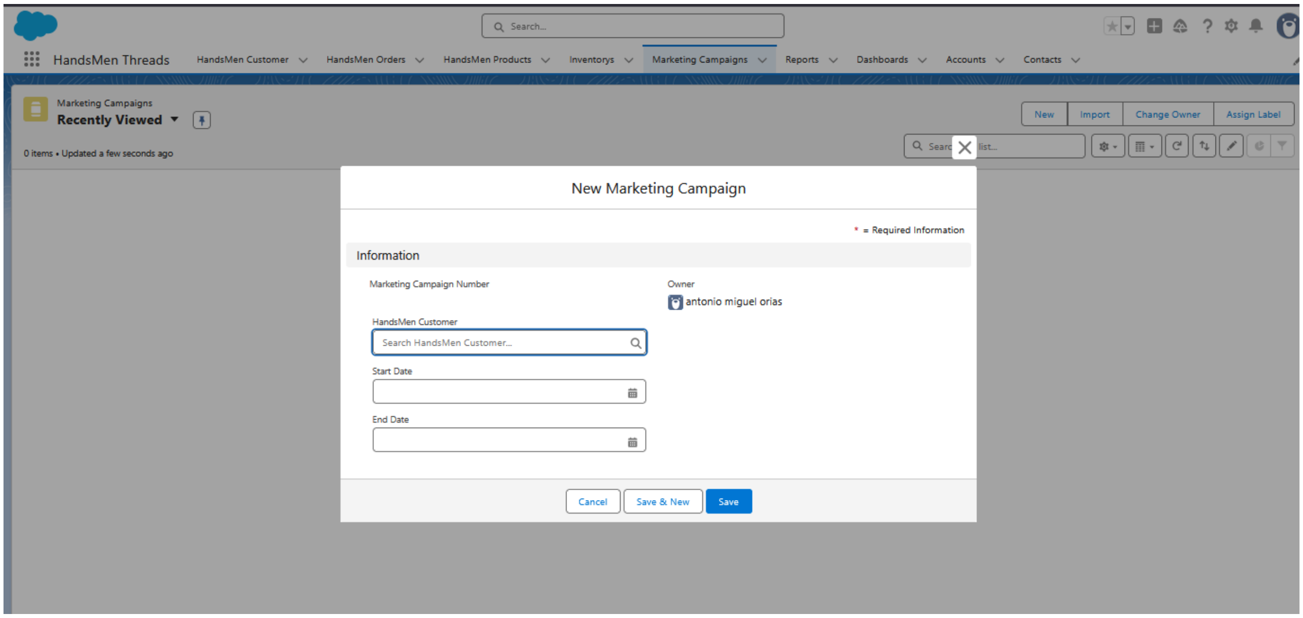
Test if the **Stock Quantity** updates when an order is confirmed, and verify that if the **Stock Quantity** is less than or equal to 10, the **Stock Status** field correctly displays either **Low**

Stock or **Available**. Also, check if you received the **Low Stock Alert** email when the **Stock Quantity** drops below 5.



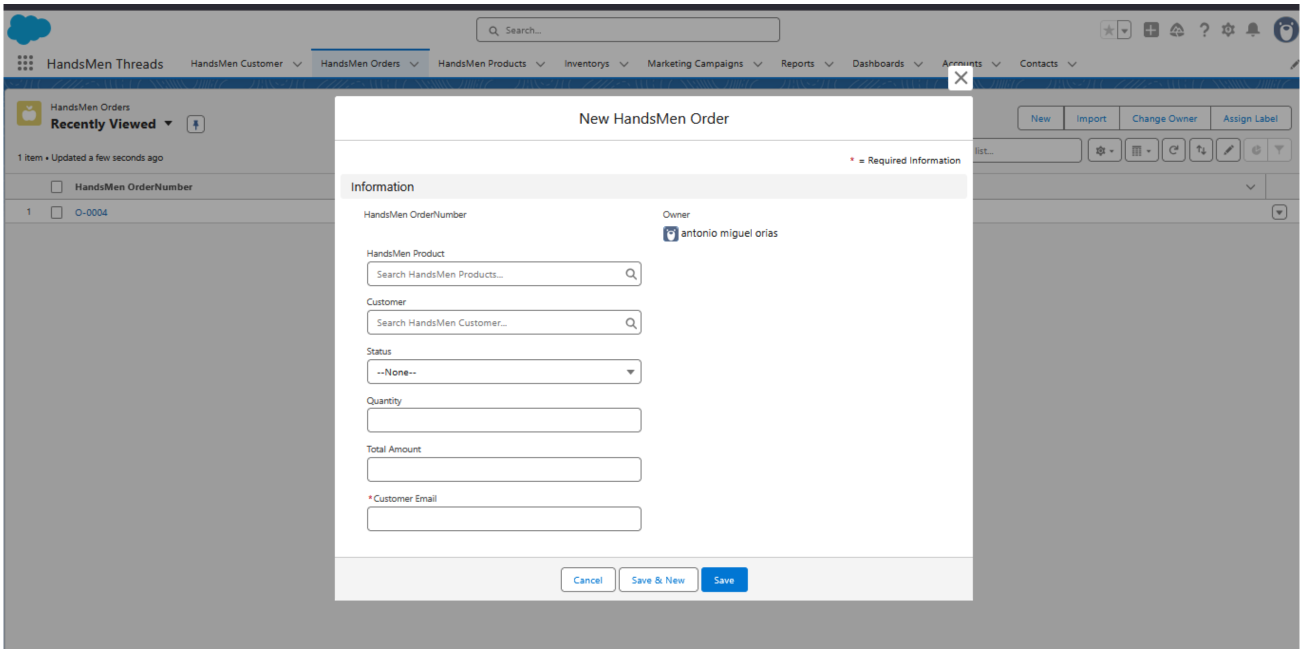
Test if you receive an error message when the **Stock Quantity** is set to **0**. This should trigger the validation rule in the **Inventory** object and display the message: "**The inventory count is never less than zero.**"

Marketing Campaign:

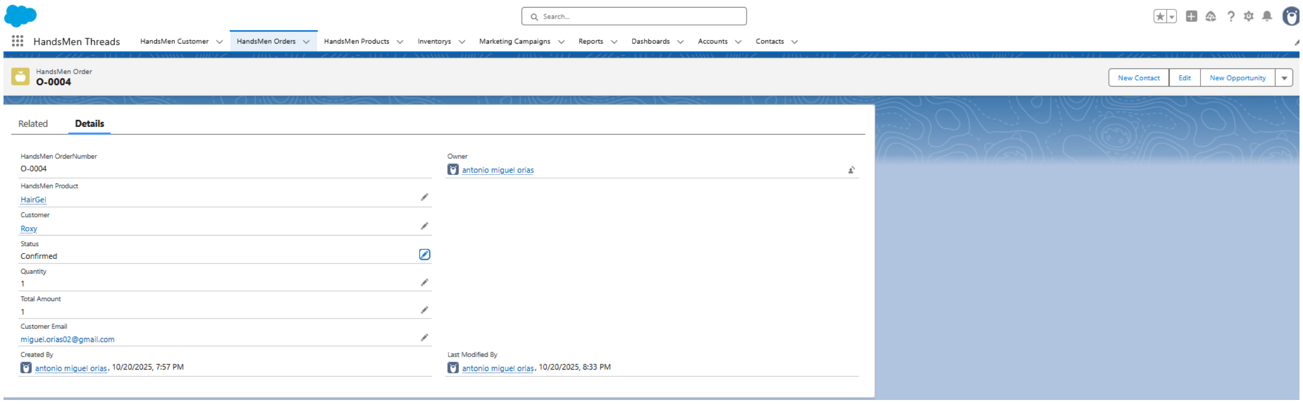


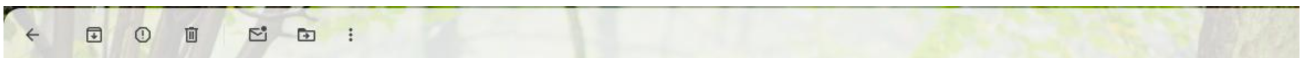
Check if the fields **HandsMen Customer**, **Start Date**, and **End Date** are displayed when adding a Marketing Campaign.

HandsMen Order:



Check if the fields **Customer**, **Status**, **Quantity, Total Amount**, **Customer Email** and **HandsMen Product** are displayed when adding an order.





Your Order has been Confirmed! Inbox ×

antonio miguel orias via bg5nhah26g2t7b.gl-diodzua1.can98.bnc.salesforce.com

to me

Dear Inventory Manager,

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

Product Name:

Current Stock Quantity:

Please take the necessary steps to restock this item immediately.

Best Regards,

Inventory Monitoring System

Created By: Antonio Miguel Orias

Conclusion

In Conclusion, By implementing HandsMen Threads CRM in salesforce marks an innovation for company's goal of success and succeed in customer needs. By integrating salesforce CRM creating; custom objects, automation flows, Apex triggers, validation rules, and permission sets, As the system now operates in combined, scalable framework managing inventory, customer relationships, orders, and marketing campaigns.

This solution resolves problem for HandsMen Threads by:

1. Deliver timely and personalized communication through automated email alerts
2. Maintain data integrity and enforce business logic via validation and formula fields
3. Streamline inventory and order processing with scheduled flows and batch jobs
4. Enhance user experience and security through role-based access and permission sets

With this, HandsMen Threads is able to adapt to innovation, improve customer loyalty, and compete its brand in the fashion landscape.