AgriConnect – Phase 1: Problem Understanding & Industry Analysis

Problem Statement

Agriculture supply chains often face inefficiencies due to: - Lack of centralized farmer and crop data.
- Manual tracking of buyer inquiries and farmer–buyer transactions.
- Poor coordination between farmers, officers, and buyers.
- No real-time insights into crop availability, yield, or sales.
This leads to missed sales opportunities, delays in advisory services and field visits, and limited decision-making for farmers and buyers.

Therefore, a Salesforce CRM solution is needed to digitize farmer support, buyer engagement, and crop-to-market visibility.

Objectives

1. To centralize farmer, crop, and buyer data in Salesforce.
2. To automate capture (farmer inquiries, buyers).
3. To streamline field visit scheduling .
4. To provide real-time dashboards for records and engagement.
5. To improve farmer–buyer engagement through automated notifications and reporting.

Use Cases

1. **Farmer Management:** Register farmers with details like crops grown, and location. Assign officers.
2. **Crop & Inventory Tracking:** Maintain digital records of crops, and availability.
3. **Visit Scheduling:** . Farmers receive emails.
4. **Buyer Inquiry & Deal Closure:** Capture buyer requests, with crop availability
5. **Reporting & Dashboards:** Farmer engagement, crops, officer activity reports.

Stakeholder Analysis

- **Farmers:** Want advisory help, better market access, timely updates.
- **Officers:** Need to track farmers, schedule visits, provide support.
- **Buyers/Distributors:** Need crop availability, pricing, and faster deals.
- **Administrators:** Want dashboards for yield, sales, and performance.

Business Process Mapping (Before Salesforce)

- Farmer calls officer manually Officer notes crops.
- Buyers approach farmers individually Delays in deals.
- No central database Limited visibility into stock & yield.
- Reports prepared manually Time-consuming.

Industry-Specific Use Case Analysis (Agriculture)

- Similar to real estate CRM: crops to buyers.
- visits similar to doctor appointments in healthcare CRM.

Phase 2: Salesforce Org Setup & Configuration

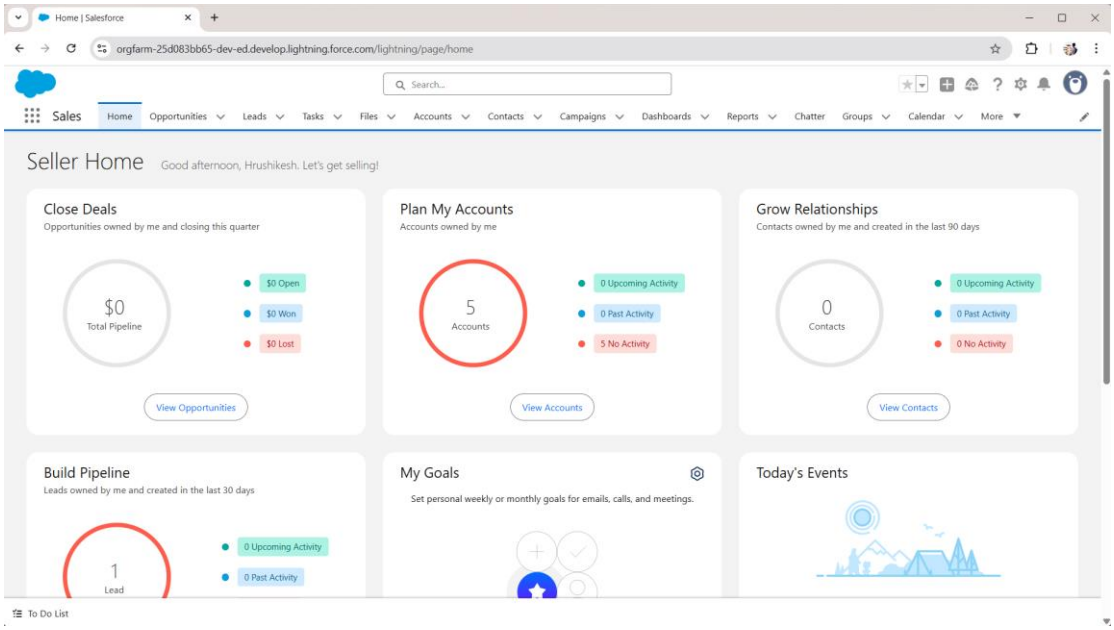
Project: **AgriConnect -Smart Agriculture CRM**

Institute: Prof. Ram Meghe Institute of Technology & Research

---

1. Salesforce Edition

- Developer Org used for implementation



// I HAVE USED SAME ORG THAT WAS USED FOR TRAINING

2. Company Profile Setup

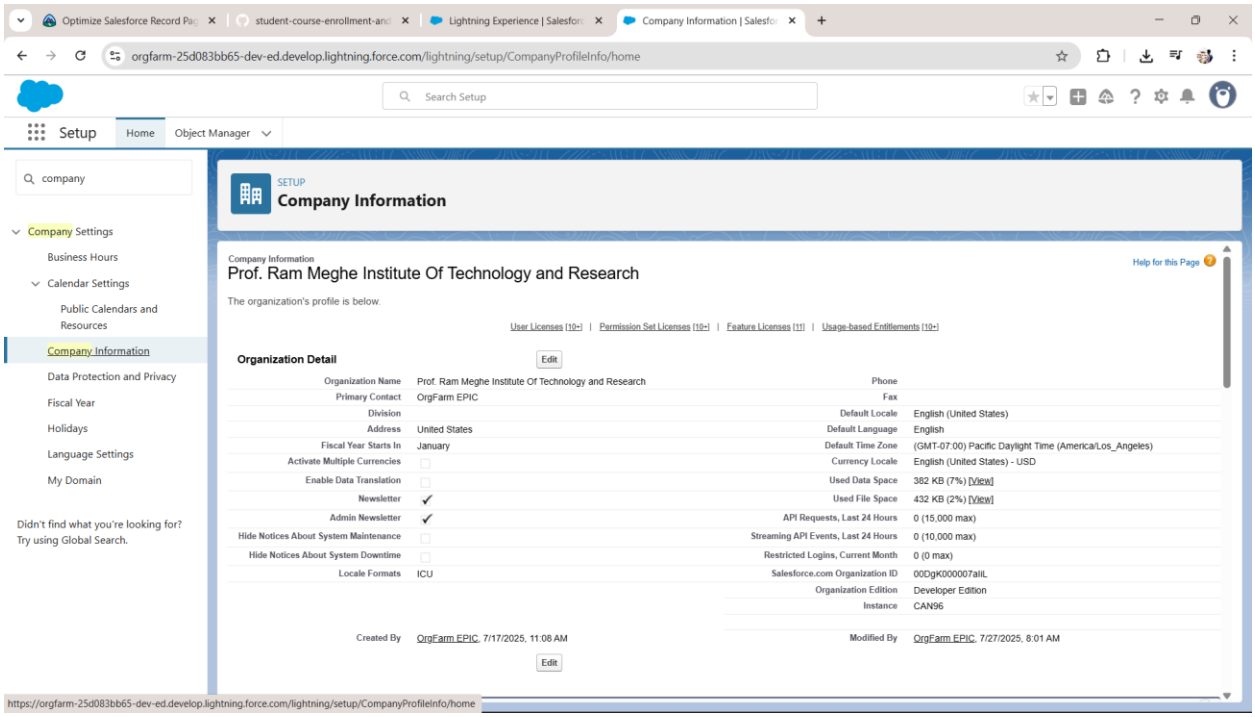
- Organization Name: Prof. Ram Meghe Institute of Technology & Research

- Default Locale: English (India)

- Default Currency: INR

- Default Time Zone: (GMT+05:30) India Standard Time

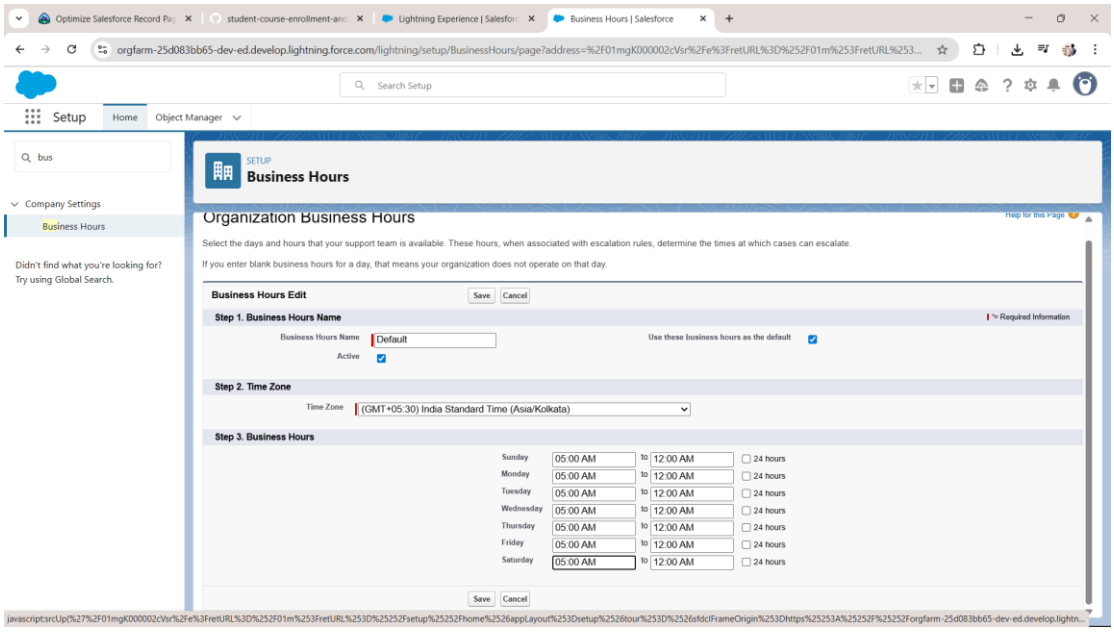
Process Flow: Setup → Company Information → Edit Company profile



3. Business Hours & Holidays

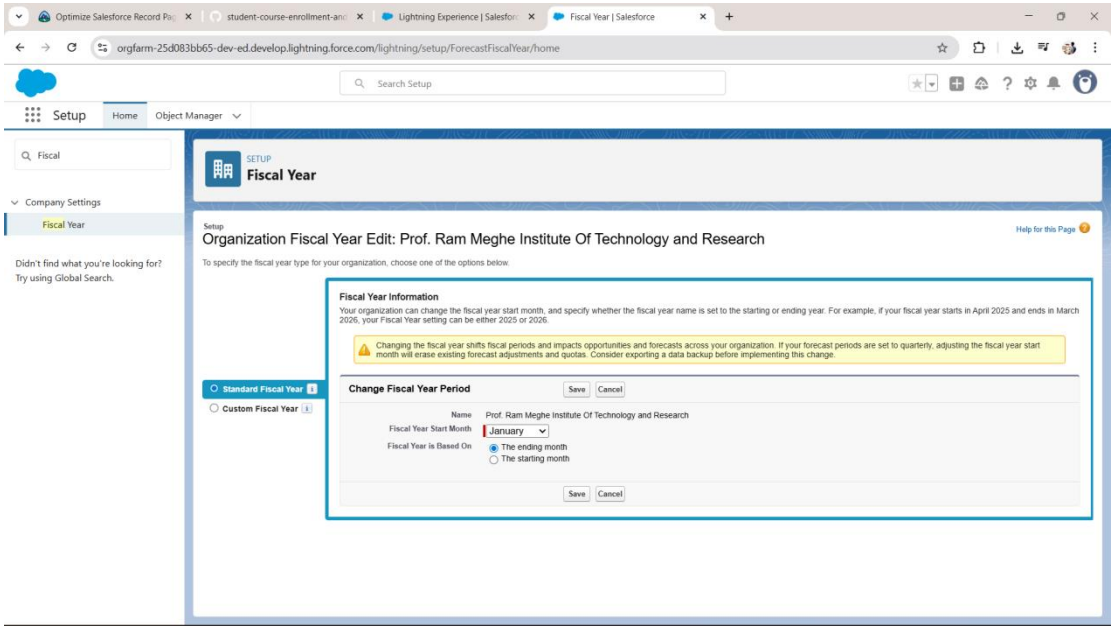
- Business Hours: Default (All days, 05:00 AM – 12:00 AM)

Process Flow: Setup → Company Settings → Business Hours → Define & Assign



4. Fiscal Year Settings

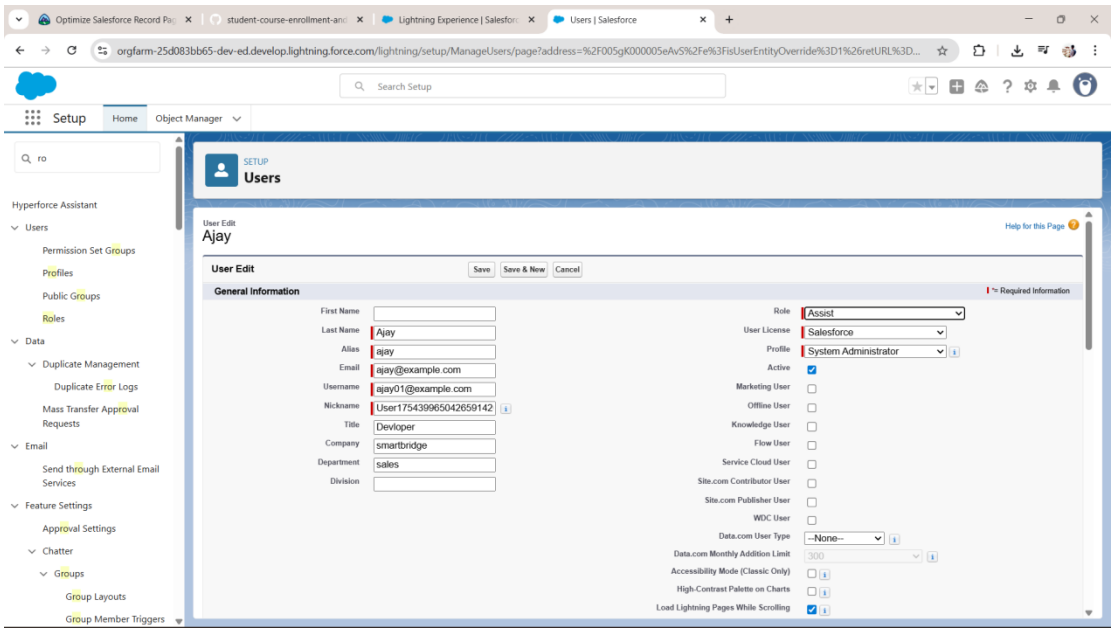
Process Flow: Setup → Company Settings → Fiscal Year → Custom Fiscal Year (if required)



5.User Setup & Licenses

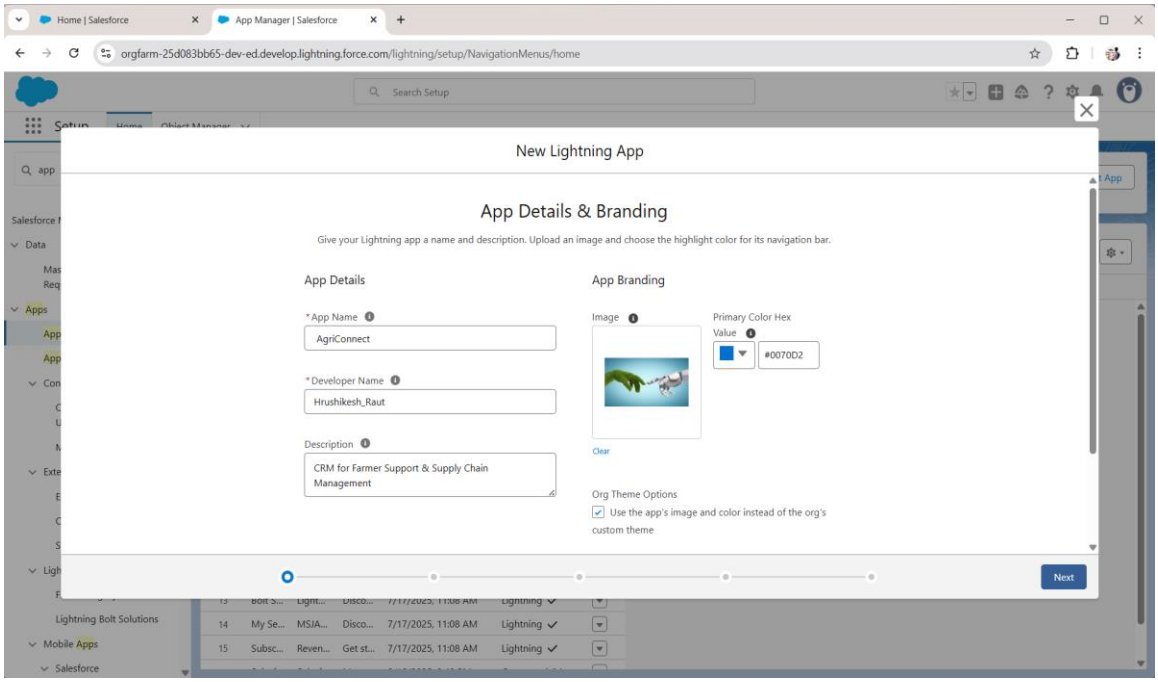
Ajay: License = Salesforce, Profile = System Administrator, Role = Assist

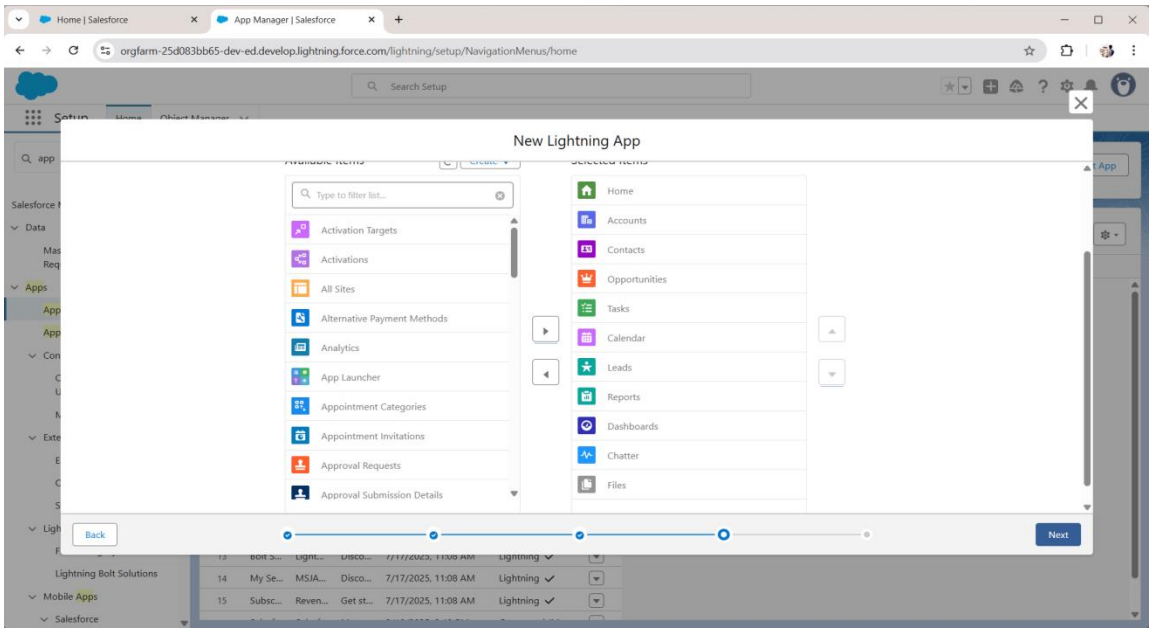
Process Flow: Setup → Users → New User → Assign License & Profile



5. App Creation

- Go to **App Manager** → Click **“New Lightning App”**→ Name the app “**AgriConnect**” →follow steps (ex : navigation items, next,.. save)

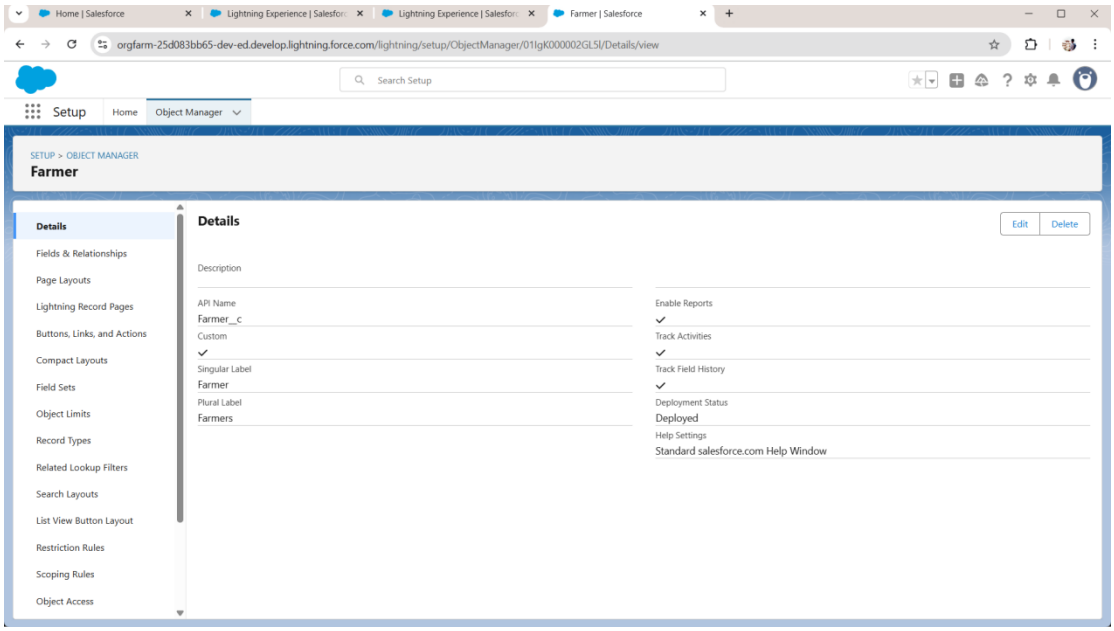




6. Create Objects

- Go to **Object Manager** → Click **“Create”** → **Custom Object**

- Define object name (Ex: “Farmer”, ”Buyer”, “Crop”,”Visit”,”Deal”)



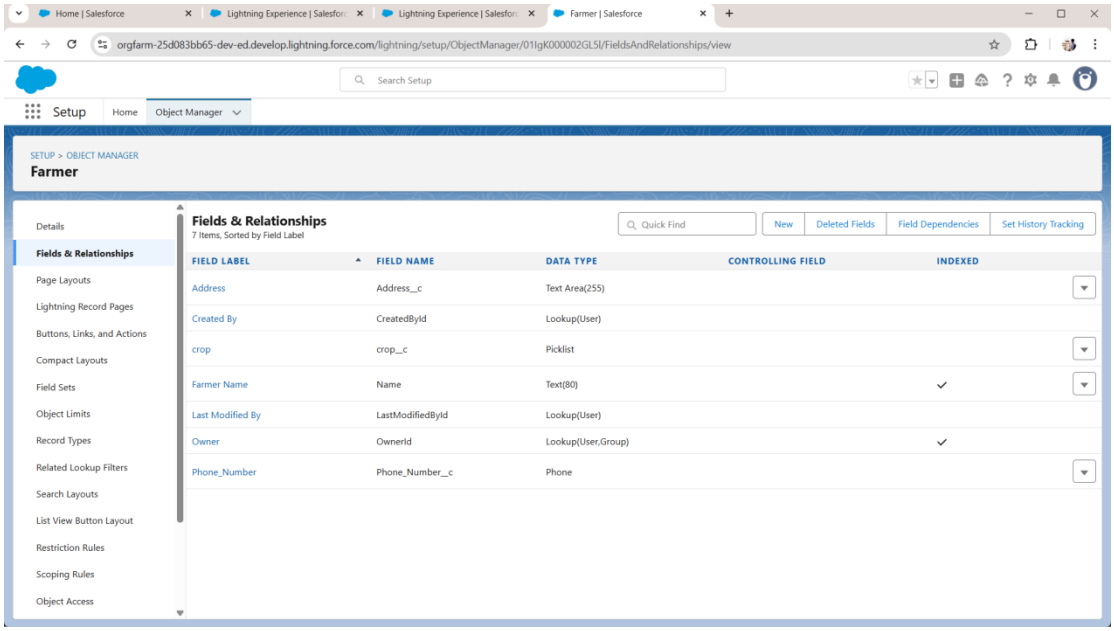
- Add necessary fields (text, date, picklist, etc.)

7. Create Custom feilds

- Inside the object, go to **Fields & Relationships**

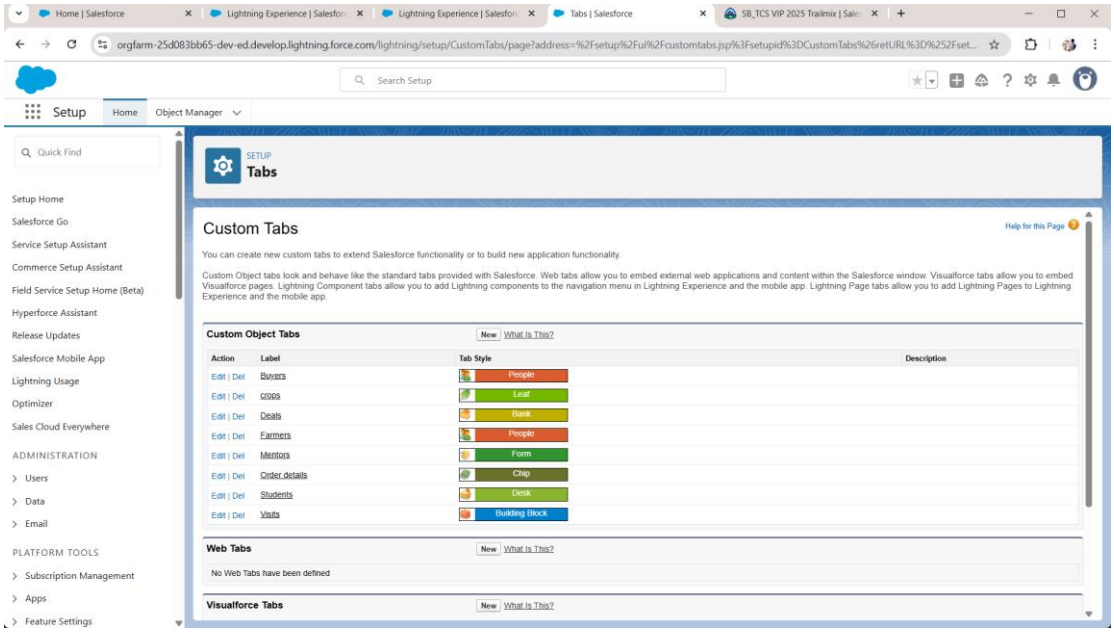
- Add new fields based on the project requirements

- Fields & Relationship → New → Choose data type → next → label name →Fill reuirements →save



8. Create Custom feilds

- Go to **Tabs** in Setup
- Create a new tab for custom object



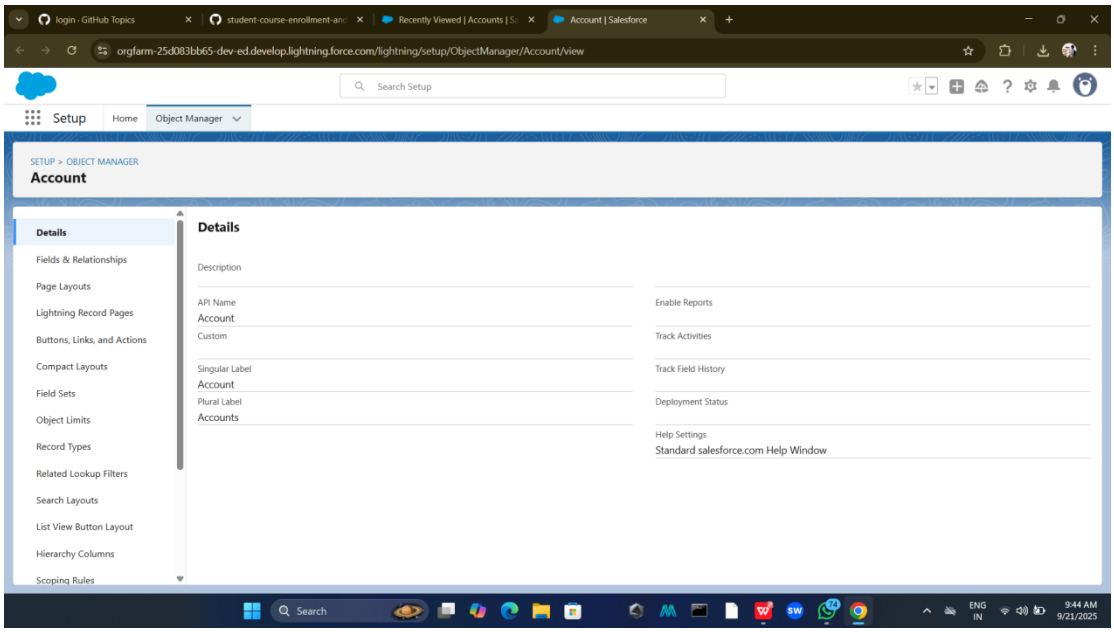
Phase 3 : Data Modeling & Relationships - Completion Report

1. Standard & Custom Objects

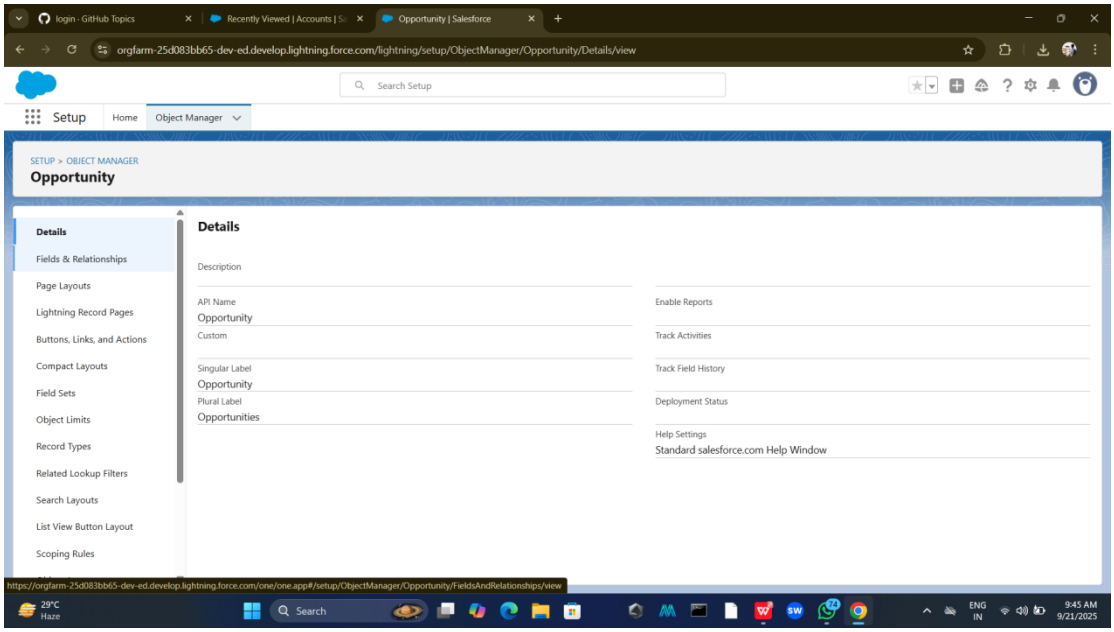
Process Flow: Setup → Object Manager → New Custom Object → Add Fields. All objects configured in a similar manner, with respective fields as listed

Standard Objects :

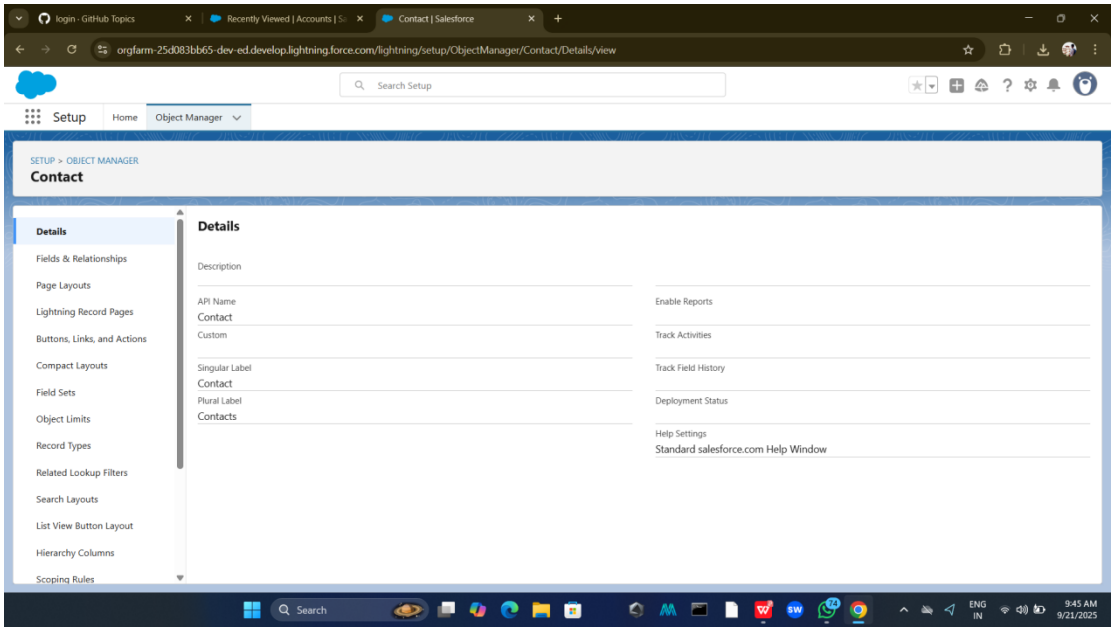
Account (used for buyer or farmer cooperatives)



Opportunity (used for Deals between Farmer & Buyer)

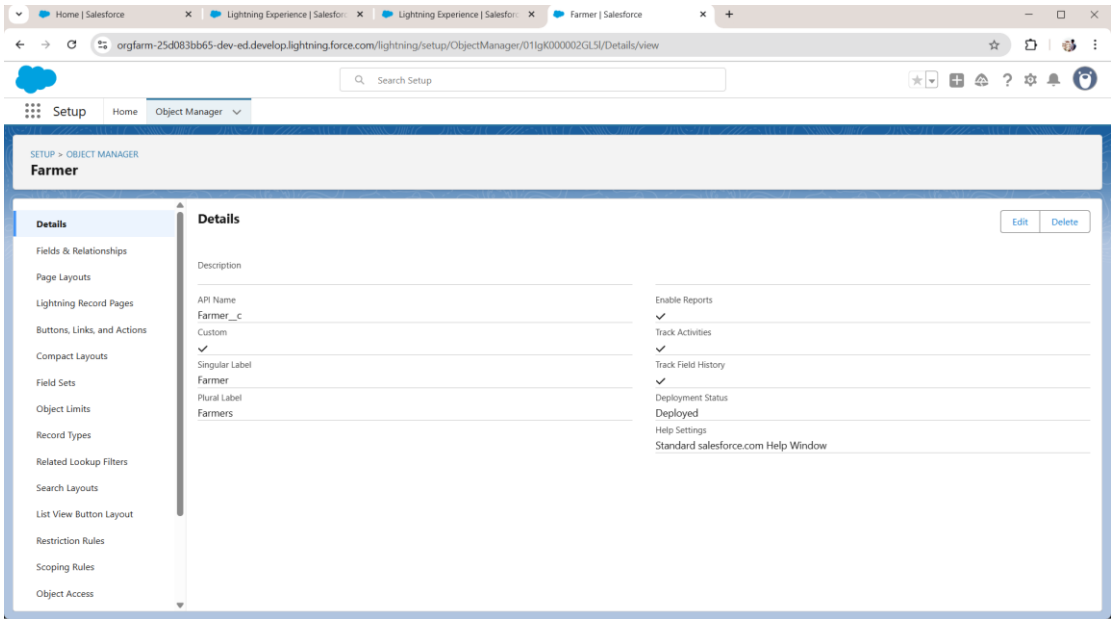


Contact (used for indivisual Farmer and Buyer)



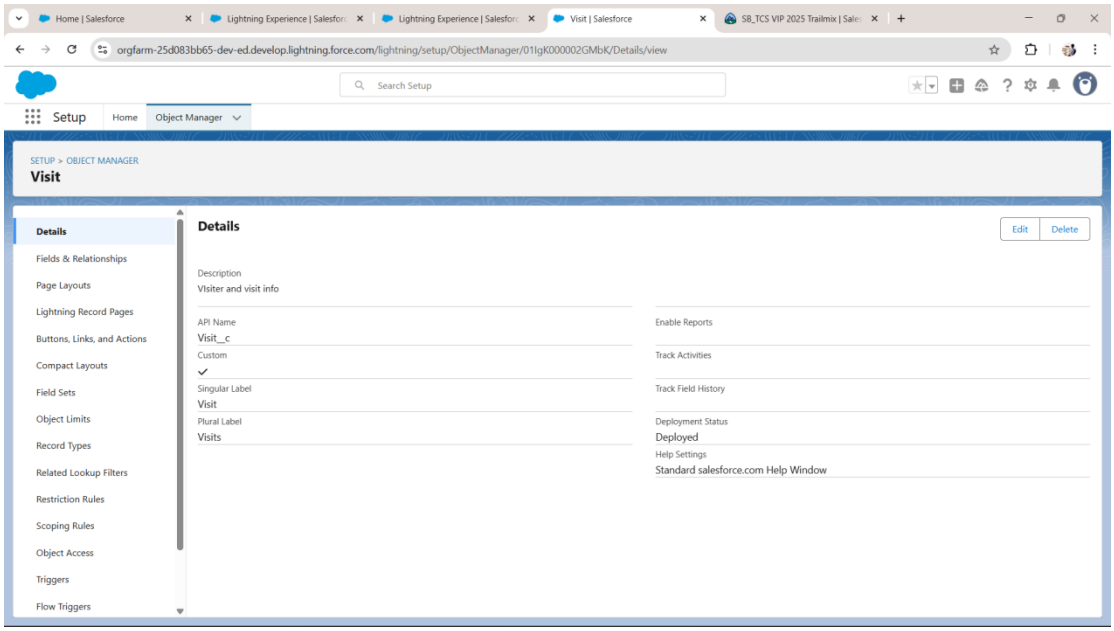
Custom Objects :

Farmer_c



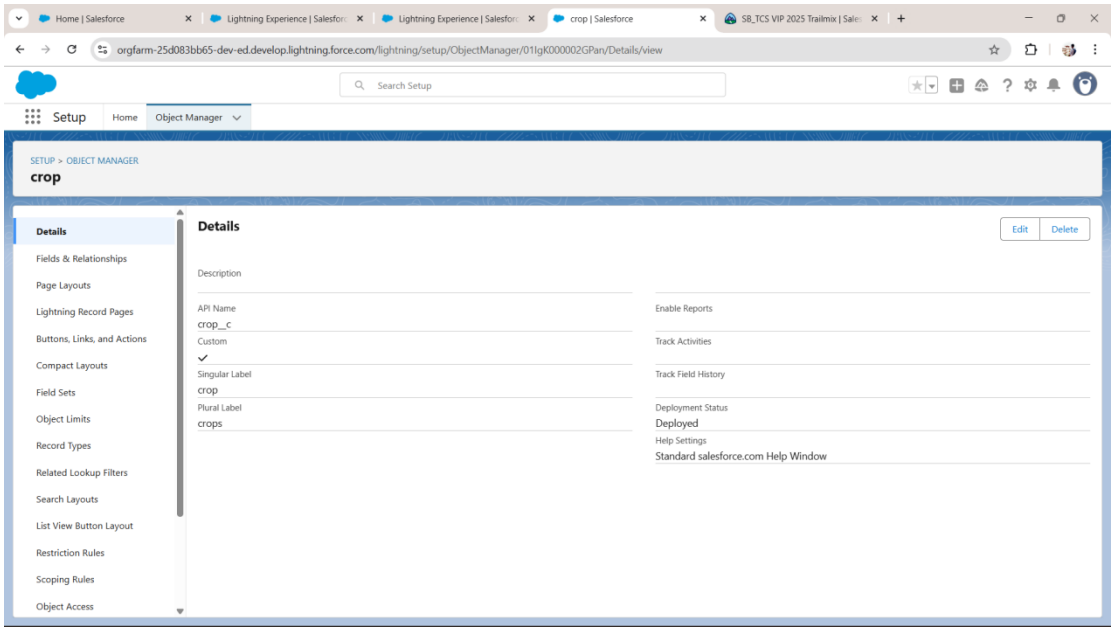
FIELD LABELSORTED ASCENDING	SORTFIELD NAME	SORTDATA TYPE
Address	Address__c	Text Area(255)
Created By	CreatedById	Lookup(User)
crop	crop__c	Picklist
Farmer Name	Name	Text(80)
Last Modified By	LastModifiedById	Lookup(User)
Owner	OwnerId	Lookup(User,Group)
Phone_Number	Phone_Number__c	Phone

Visits_c



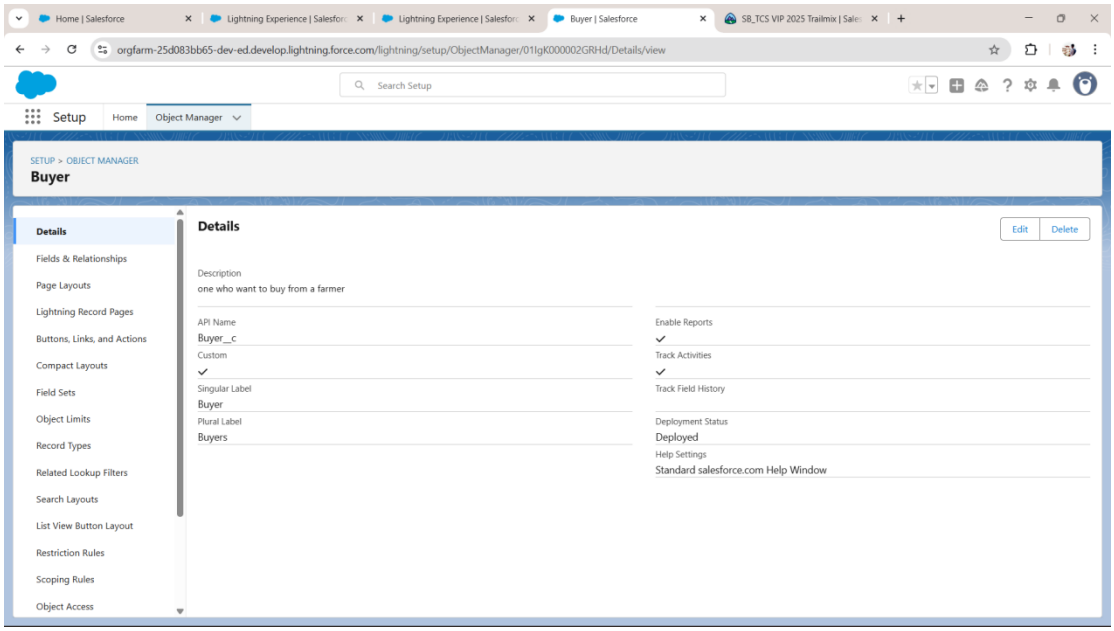
FIELD LABELSORTED ASCENDING	SORTFIELD NAME	SORTDATA TYPE
Created By	CreatedById	Lookup(User)
Last Modified By	LastModifiedById	Lookup(User)
Lookup	Lookup__c	Lookup(Farmer)
Owner	OwnerId	Lookup(User,Group)
Purpose	Purpose__c	Text(150)
Visit Date	Visit_Date__c	Date
Visit Name	Name	Text(80)
Visiter name	Visiter_name__c	Text(150)

Crop_c



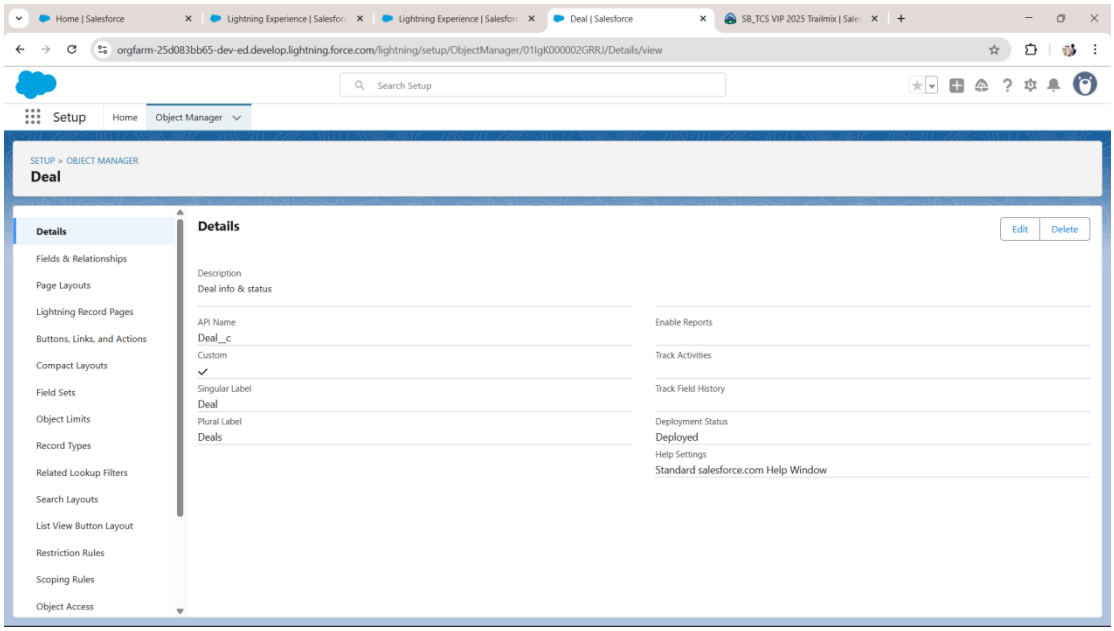
FIELD LABELSORTED ASCENDING	SORTFIELD NAME	SORTDATA TYPE
Created By	CreatedById	Lookup(User)
crop Name	Name	Text(80)
Harvest Time	Harvest_Time__c	Date
Last Modified By	LastModifiedById	Lookup(User)
Master detail	Master_detail__c	Master-Detail(Farmer)
Prize	Prize__c	Currency(18, 0)
Quantity	Quantity__c	Number(18, 0)

Buyer_c



FIELD LABELSORTED ASCENDING	SORTFIELD NAME	SORTDATA TYPE
Buyer Name	Name	Text(80)
Contact	Contact__c	Phone
Created By	CreatedById	Lookup(User)
Last Modified By	LastModifiedById	Lookup(User)
Owner	OwnerId	Lookup(User,Group)

Deal_c



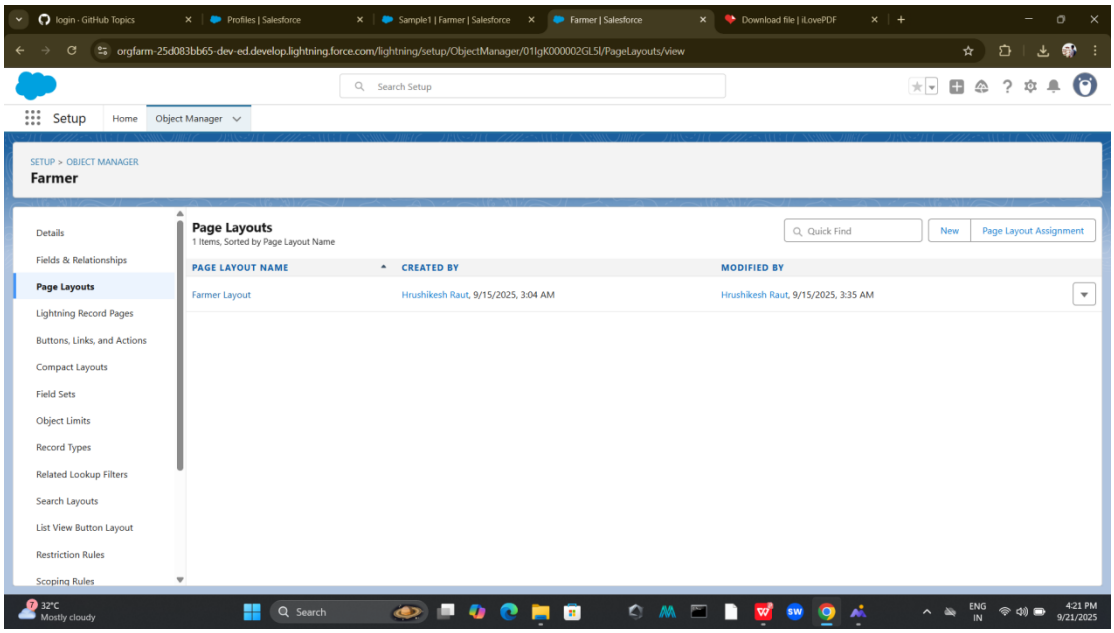
FIELD LABELSORTED ASCENDING	SORTFIELD NAME	SORTDATA TYPE
Created By	CreatedById	Lookup(User)
crop	crop__c	Picklist
Deal Name	Name	Text(80)
Farmer	Farmer__c	Lookup(Farmer)
Last Modified By	LastModifiedById	Lookup(User)
Lookup1	Lookup1__c	Lookup(Buyer)
Owner	OwnerId	Lookup(User,Group)
Quantity	Quantity__c	Text(150)
status	status__c	Picklist

2. Page Layouts

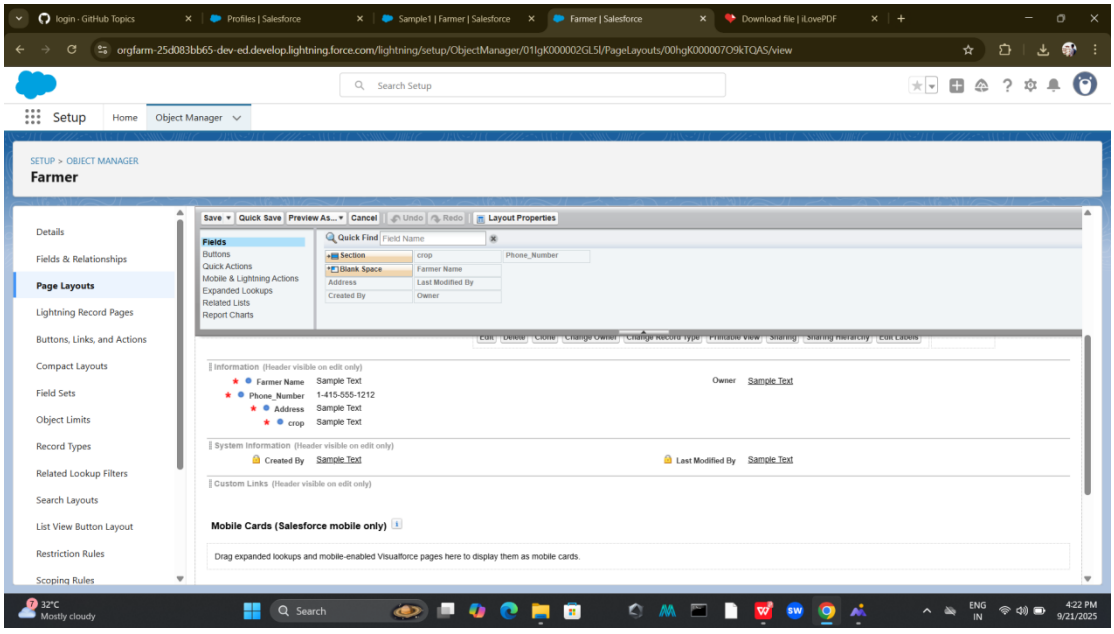
Configured for Farmer_c as an example:

Process Flow: Setup → Object Manager → Farmer → Page Layouts → Farmer Layout

→ Drag fields into sections → Save.

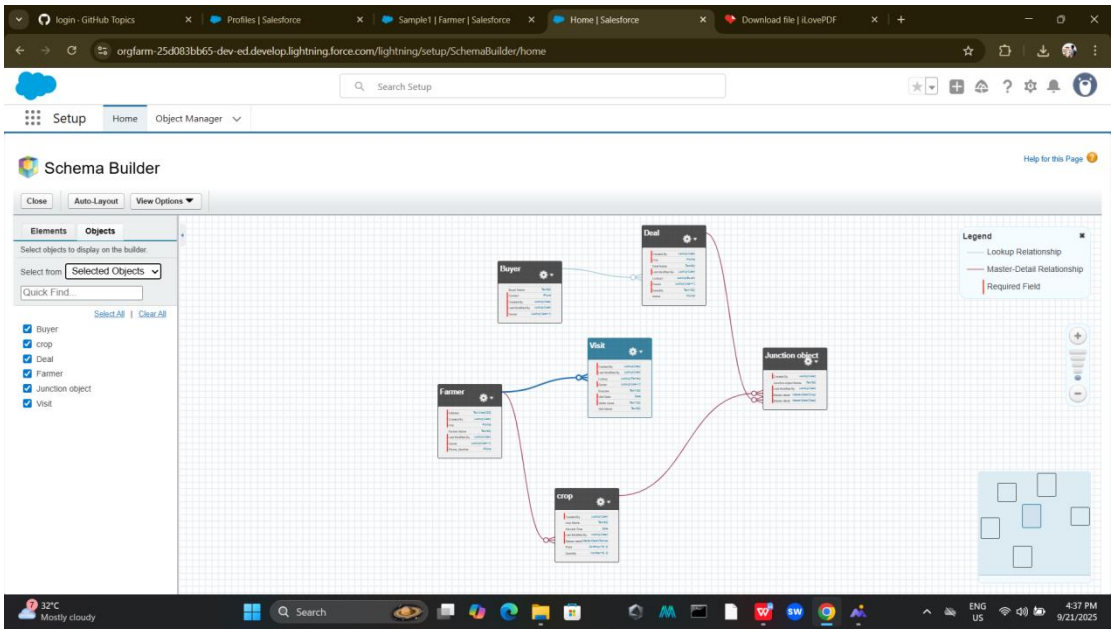


Related Lists (visit, deal, crops)



Other objects (Crop, Visit, Buyer, Deal) had layouts configured similarly with their relevant fields.

3 .Schema Builder



- Farmer (Master) → Crop (Detail).
- Farmer → Visit (Lookup).
- Crop ↔ Buyer (Many-to-Many via Deal)

Phase 4 Completion Report – Process Automation (Admin)

1. Validation Rules

Validation rules were created across different objects:

Steps to Create Validation Rule :

1. Go to Setup → Object Manager → Select the required Object .
2. Navigate to Validation Rules → New Rule.
3. Enter Rule Name and Description.
4. Define the Formula (logic specific to the rule).
5. Enter the Error Message and select Error Location.
6. Save & Activate.

Repeating repeat the steps for each rule as listed above.

Example: Crop Quantity must be greater than 0

Steps:

Setup → **Object Manager** → Select **Crop** → **Validation Rules**.

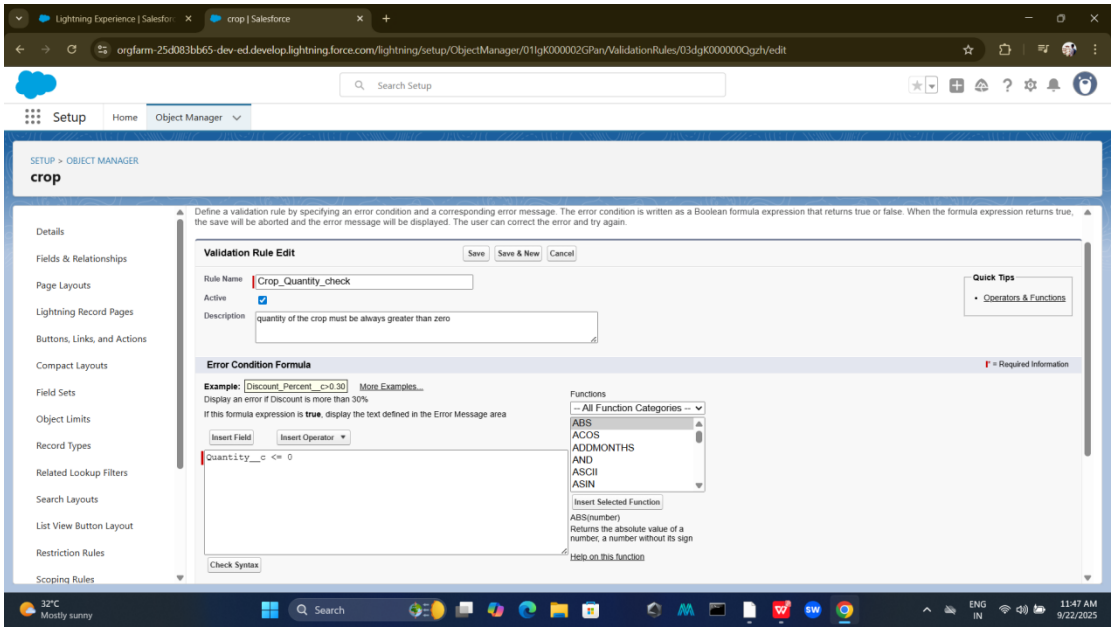
Click **New Rule** → Enter Rule Name: `Crop_Quantity_Check`.

Formula: `Quantity__c <= 0`

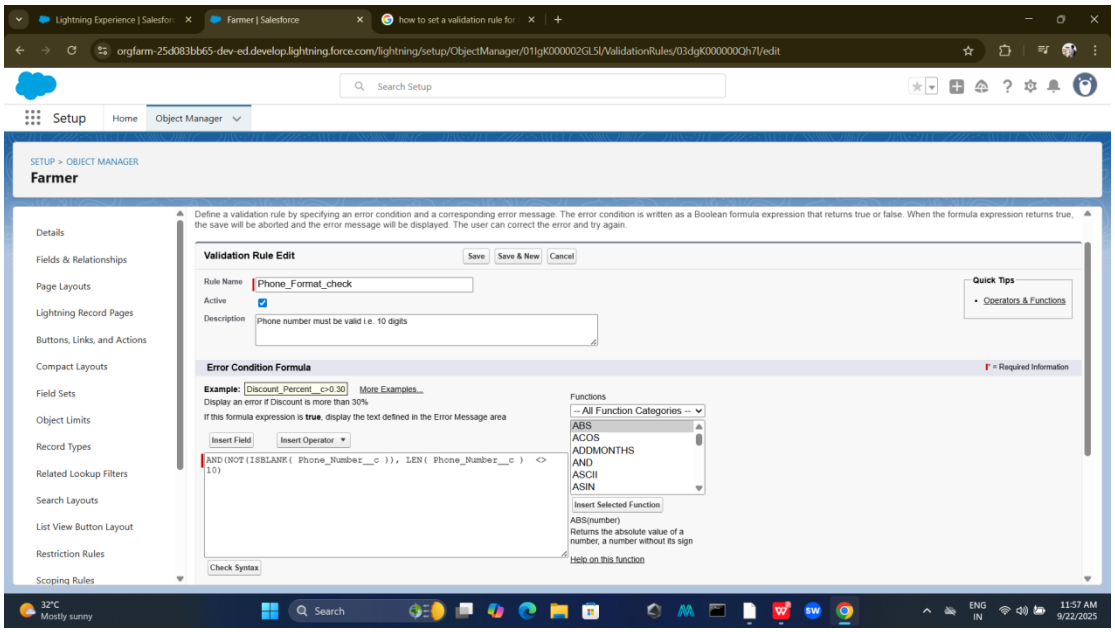
Error Message: "Quantity must be greater than 0."

Choose where to display → *Top of Page* or *Field*.

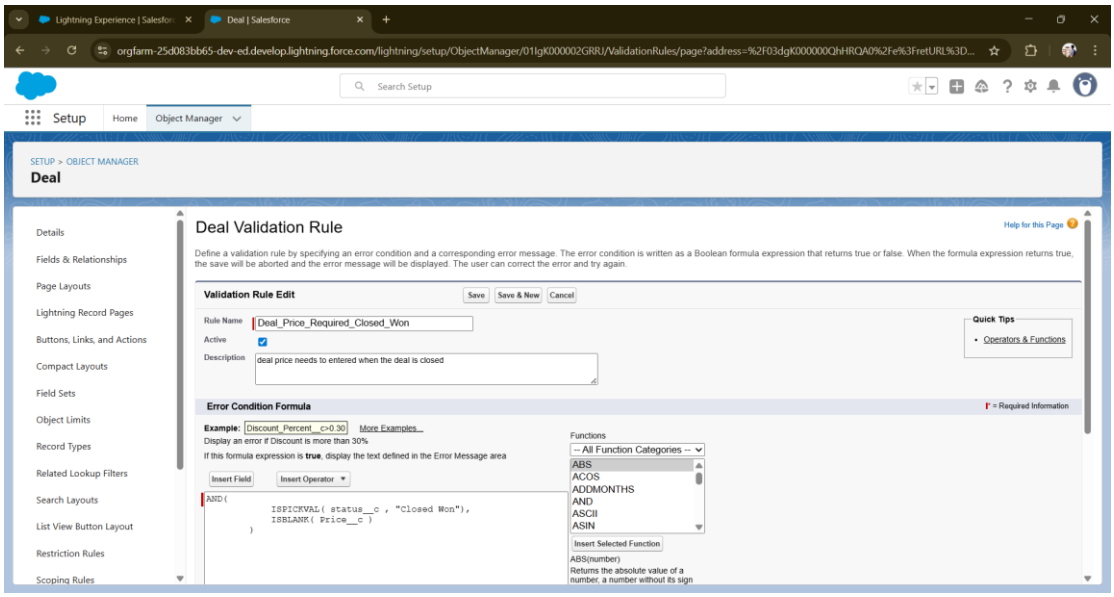
Save & Activate.



Farmer _ c Phone Number Format : (Phone number shouldn't be Null & must have 10 digits)



Deal_c Validation Rule : (Deal Price not empty when Closed Won)



2. WorkFlow Rules

Workflows perform actions when conditions are met.

 Example: Send Email when New Farmer is Created
Steps:

Setup → **Workflow Rules** → New Rule.

Object: **Farmer**

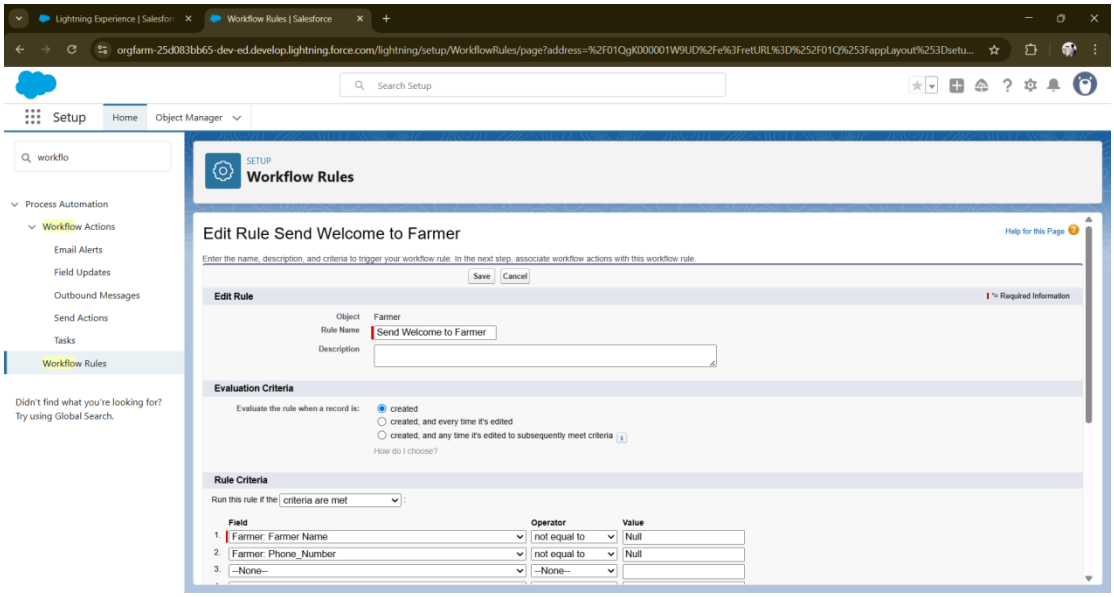
Rule Criteria: `Created = True`

Add Workflow Action → **New Email Alert**

Select Email Template → “Welcome Farmer”

Recipient → Farmer Email

Activate Workflow



3. Flow Builder

Flows replace Workflow & Process Builder.

- **Record-Triggered Flow (Visit Scheduling)**

Trigger: When new Visit record is created.

Action: Send email notification to Farmer.

Steps:

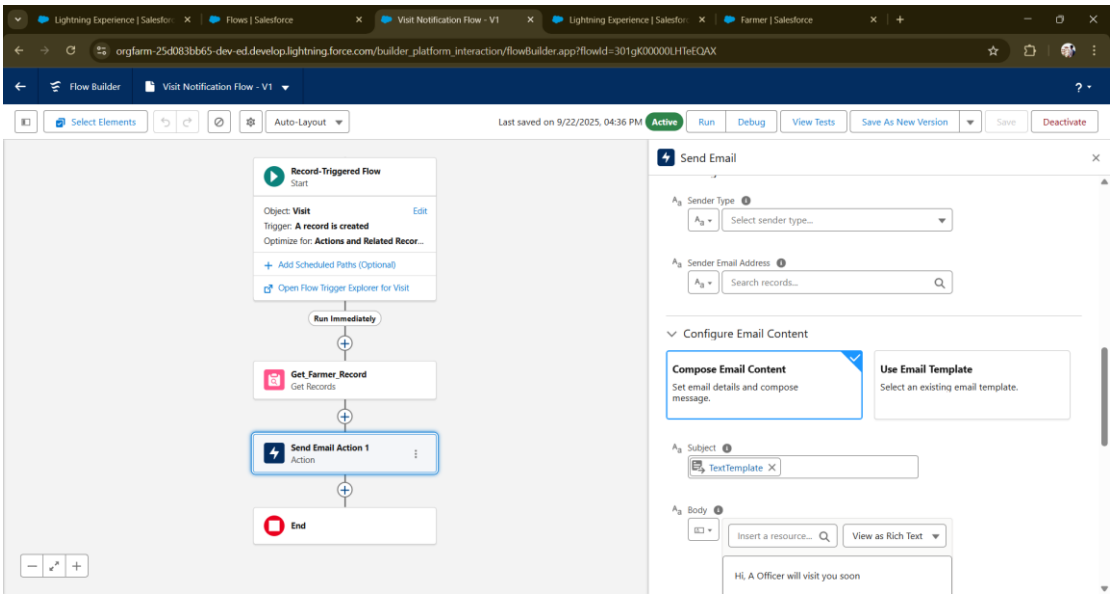
Setup → **Flows** → New Flow.

Choose **Record-Triggered Flow** → Object = Visit.

Trigger = On Create.

Add **Action** → Email Alert

Save & Activate.



➢ Update Crop Stock after Deal Closure

Steps:

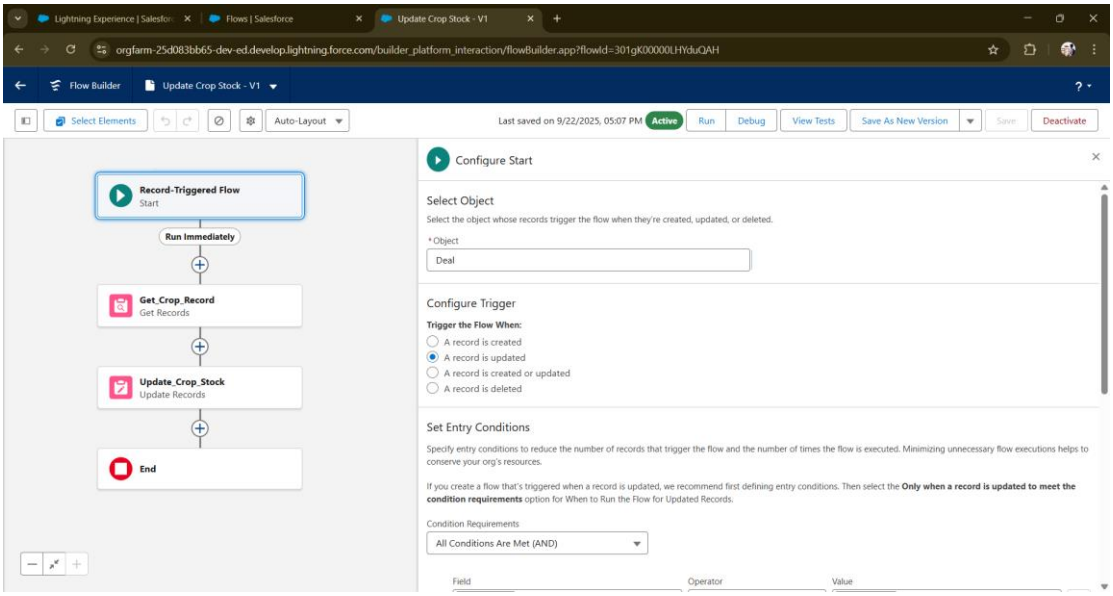
Setup → **Flows** → New Flow.

Type: **Record-Triggered** → Object = Deal.

Trigger = When Deal Status = "Closed Won."

Add Update Records → Reduce Crop.Quantity = Crop.Quantity – Deal.Quantity.

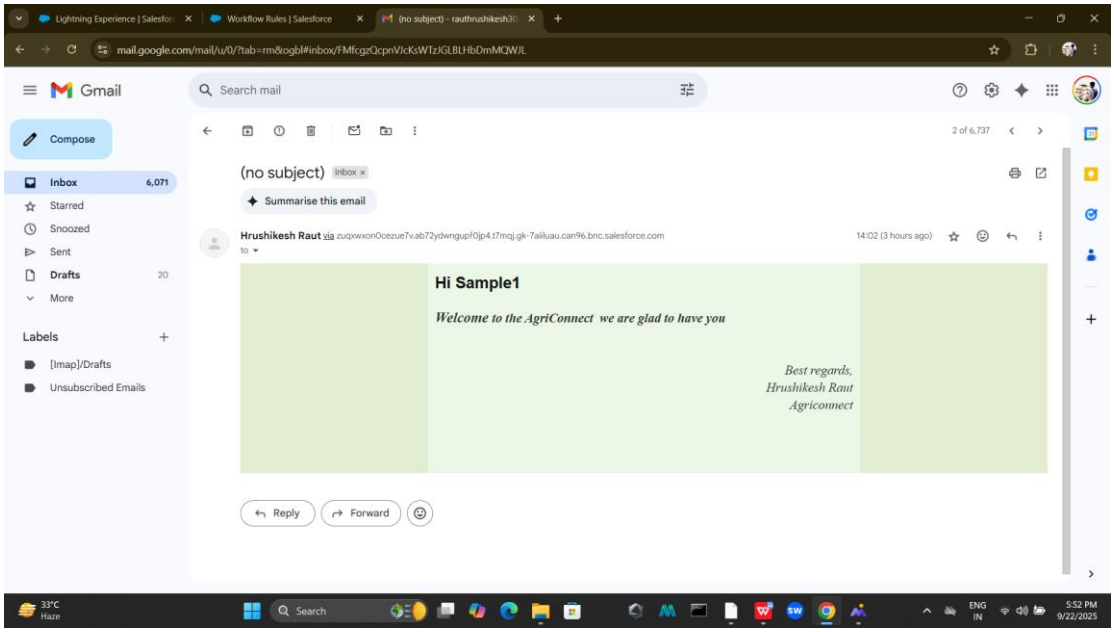
Save & Activate.



4. Email Alerts

Email Alerts were successfully configured and activated :

Example Email : Welcome email for Farmer



Phase 5 Completion Report – Apex Programming (Developer)

1. Classes & Objects

- **Create an Apex Class**

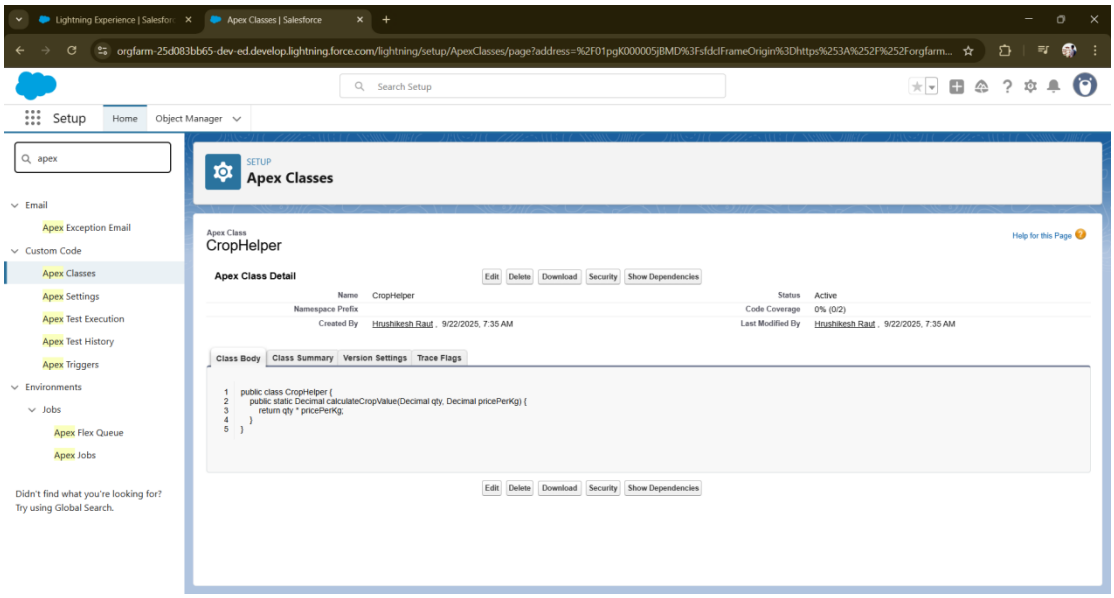
Steps:

1. In Salesforce → Setup → Quick Find → **Apex Classes** → New.

OR Developer Console → File → New → Apex Class.

2. Enter Class Name and click OK.
3. Write Apex code implementing the required logic.
4. Save the class.

Example: Utility class to calculate total crop value.



2. Apex Triggers

Steps to Create Trigger:

1. Developer Console → File → New → Apex Trigger.

2. Select Object

3. Write Trigger code .

4. Save the trigger

✧ **Example** :Let's create a Trigger i.e.

When a Deal is **Closed Won**, automatically update the available Crop quantity.

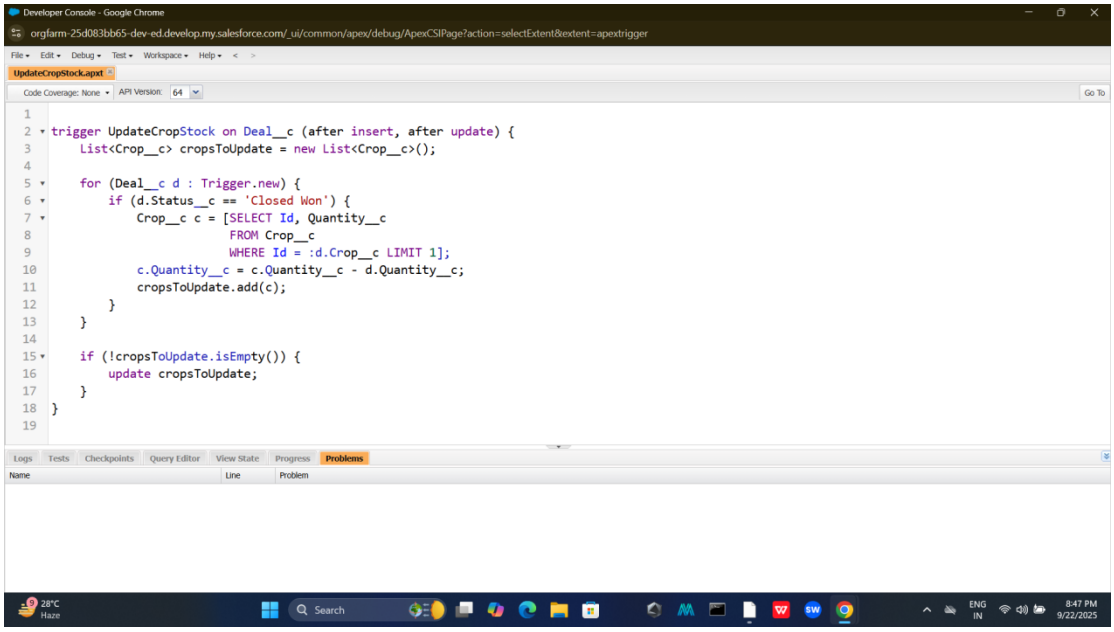
Steps:

Setup → Apex Triggers → New Trigger.

Select **Deal__c** object.

Add Trigger Code as mentioned in screenshot

Save & Test: Close a Deal → Crop stock reduces



3. SOQL (Salesforce Object Query Language)

---- Used to query data inside Apex.

Example Query:

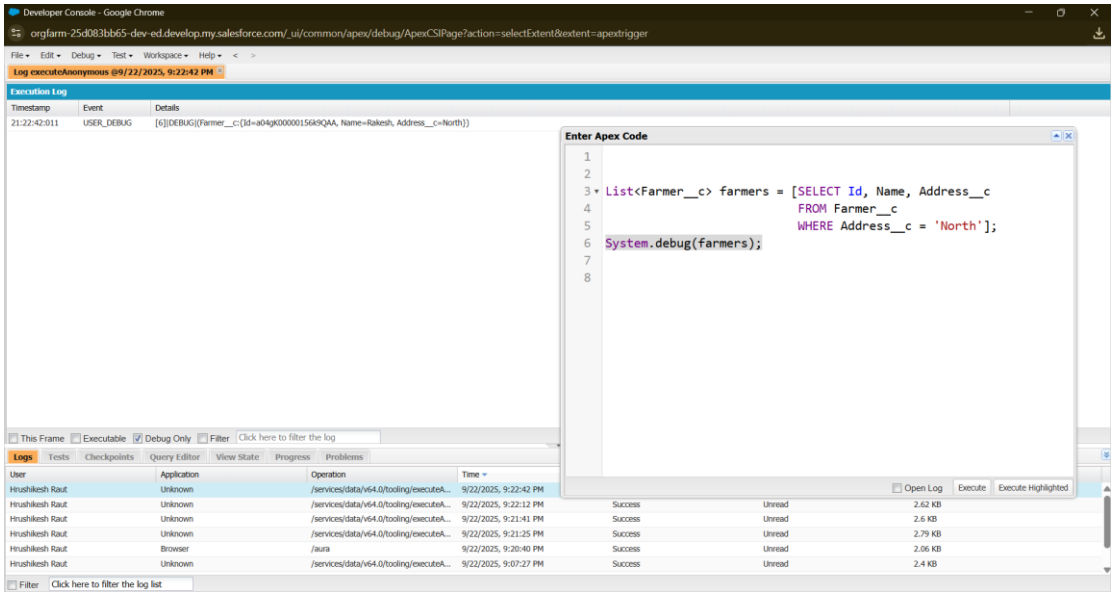
List<Farm__c> farmers = [SELECT Id, Name, Address__c
FROM Farmer__c
WHERE Address_c = 'North'];

'Queries Can Run in developer console'

i. Enter Ctrl + E to open window for code

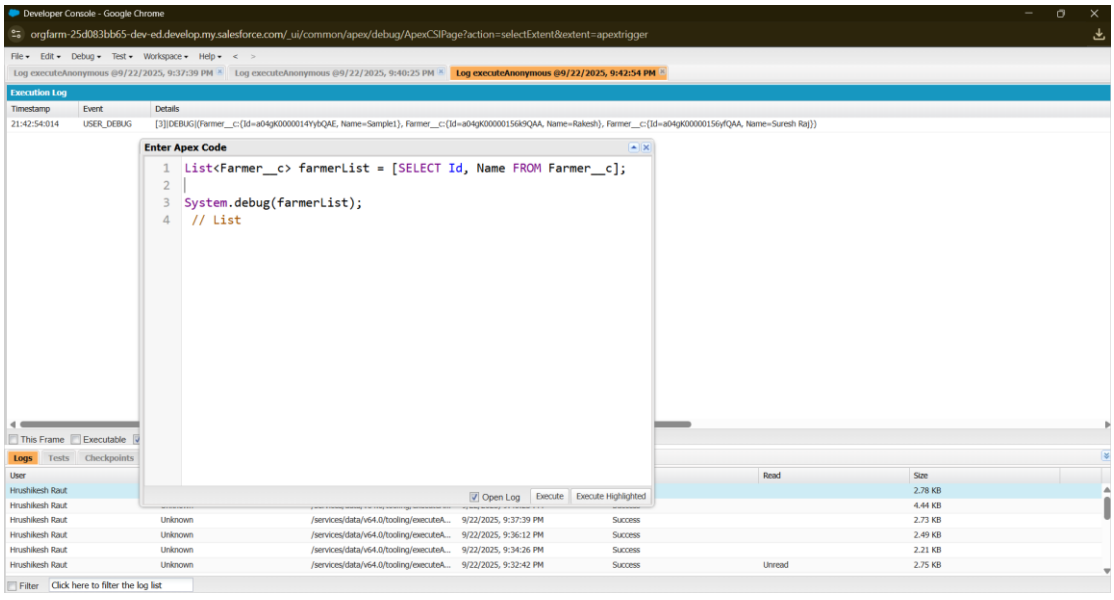
ii. Enter Query followed by Debug Command

iii. Click Execute

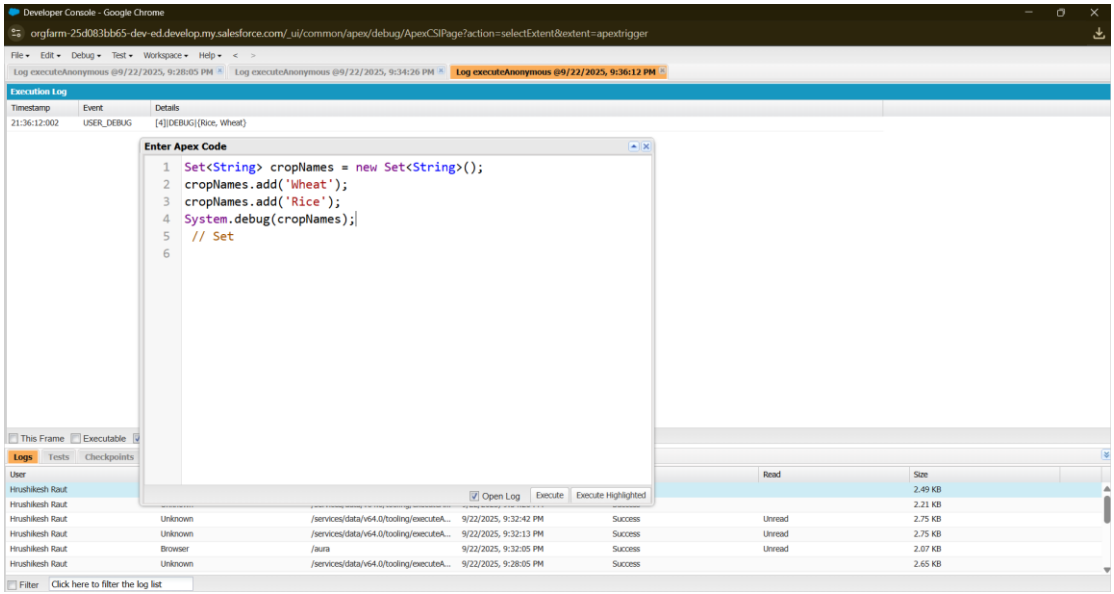


4. Collections: List, Set, Map

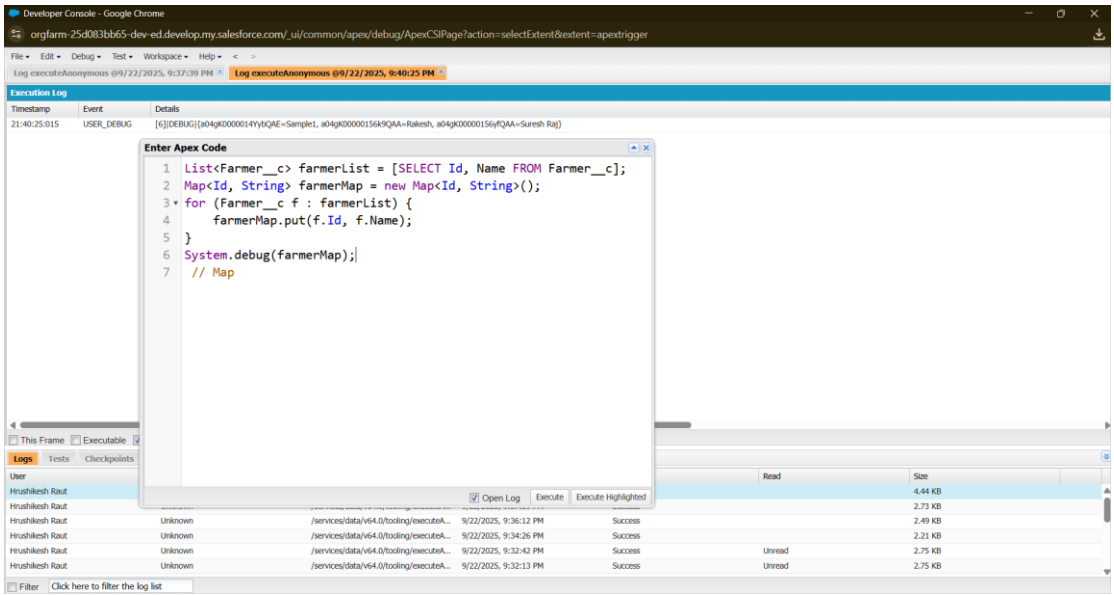
- **List:** Store multiple Farmers



- **Set:** Unique crop names.

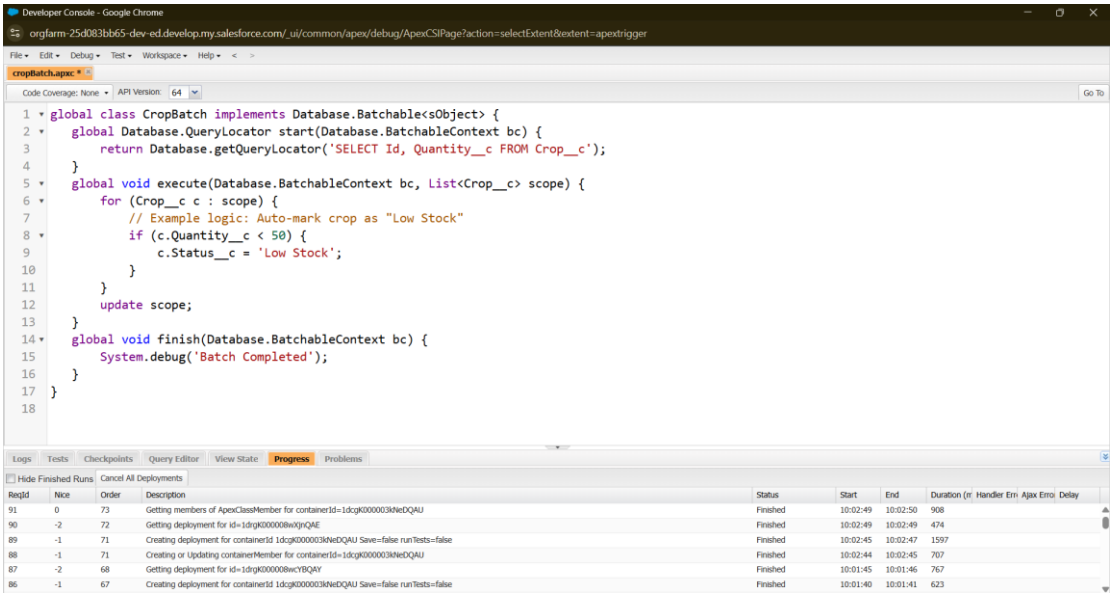


- **Map:** Match Farmer IDs with Names.



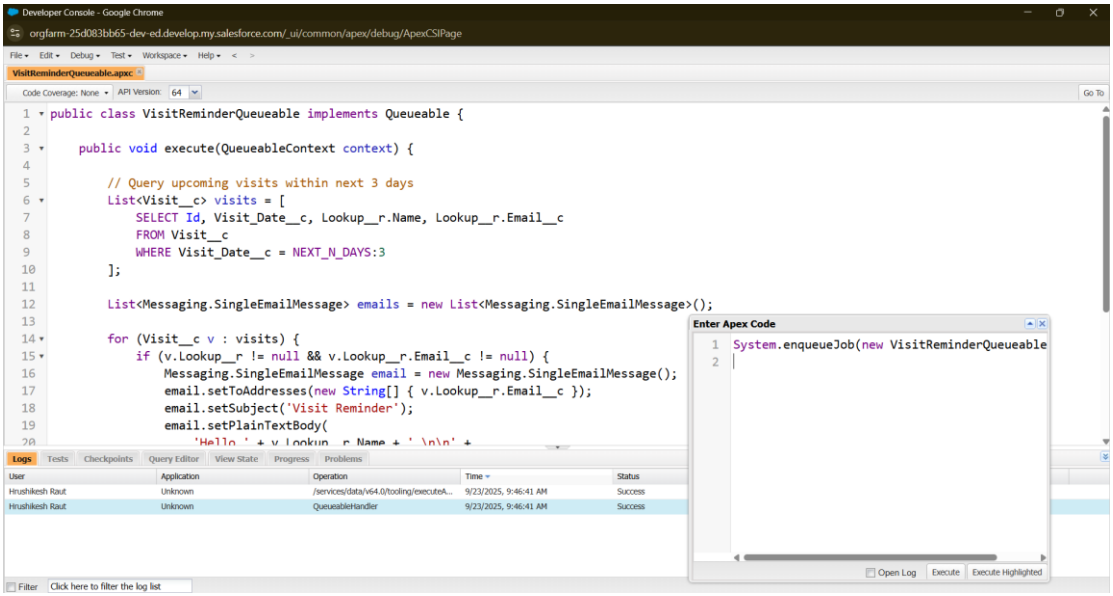
5. Batch Apex

✧ Batch Apex – Weekly Update Crop Reports :



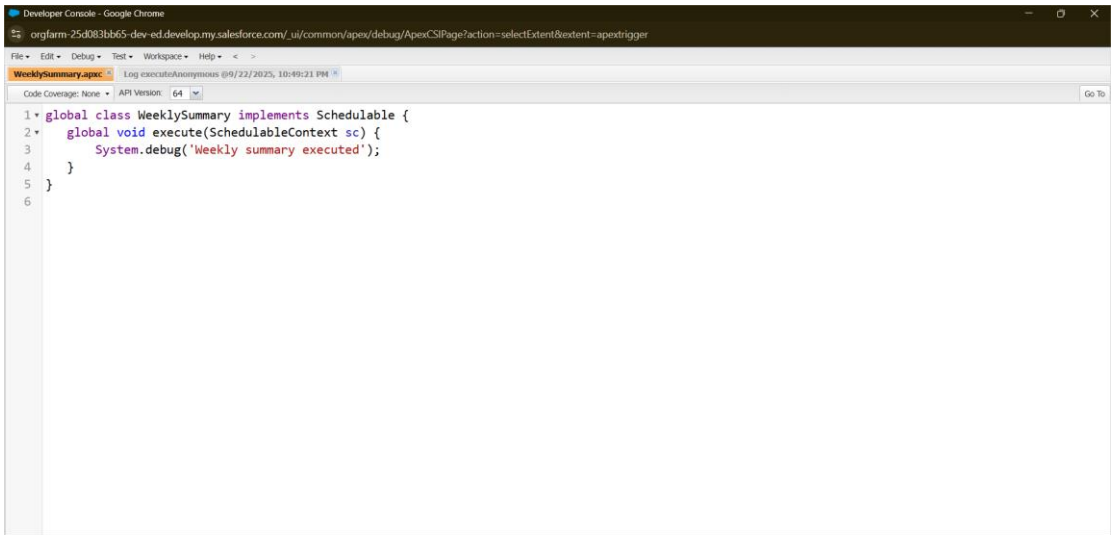
6. Queueable Apex

✧ Queueable Apex –Visit Reminder :

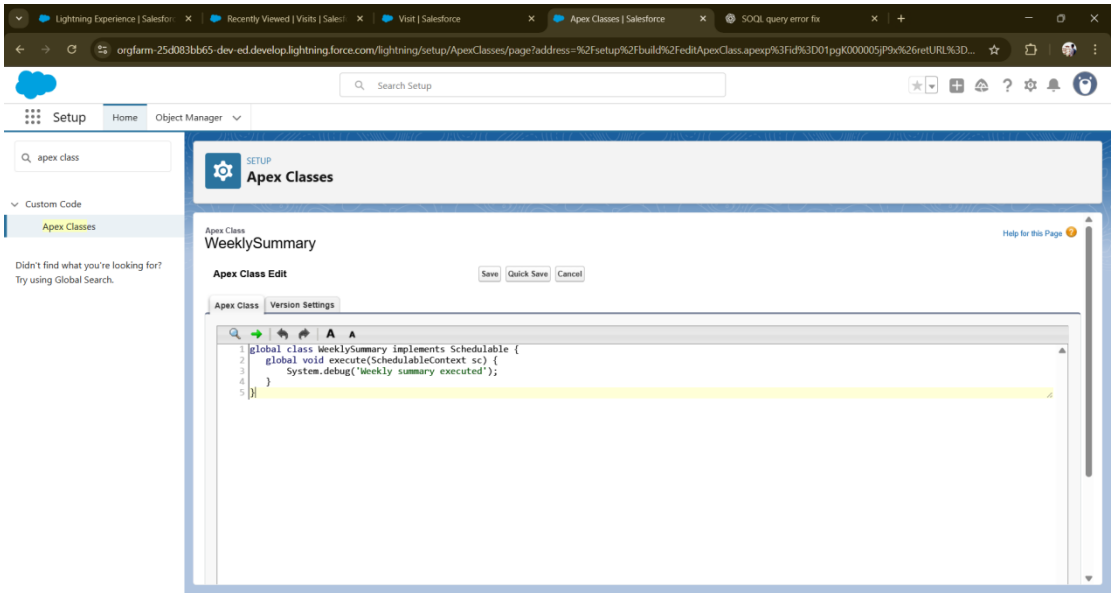


7. Schedule Apex

✧ Scheduled Apex – Run Every Week :



Setup → Apex Classes → Schedule Apex or via Developer Console.



8. Test Classes

Requires 75% code coverage.

Steps:

Setup → Apex Test Execution → New Test.

Enter the test class name

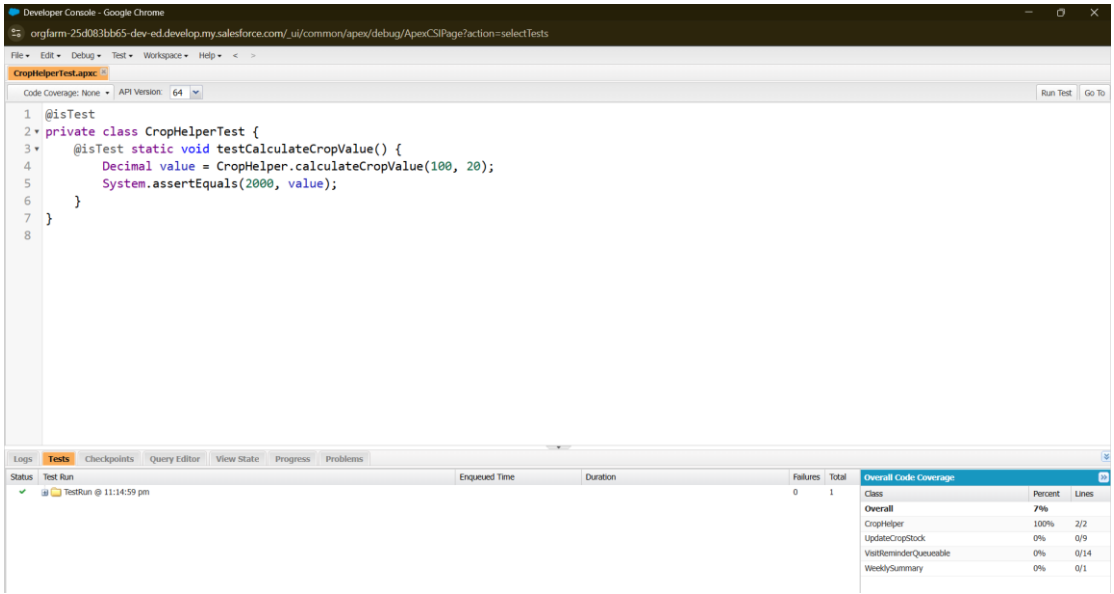
Enter test class code

Save & Run

Steps to run via setup :

- Go to **Setup → Apex Test Execution**.
- Click **Select Tests** → Pick **VisitReminderQueueableTest.**
- Run Test → Wait for result

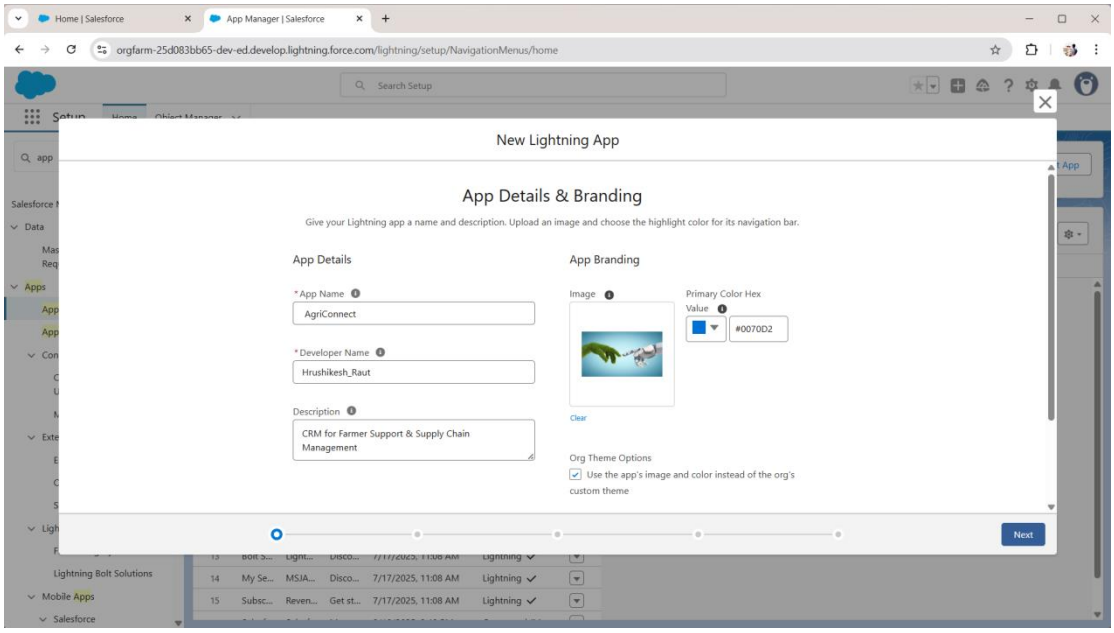
Example : test for CropHelper (sends remainder for visit)



Phase 6 User Interface Development

❖ Lightning App Builder

- This was the primary tool used for all declarative UI design. Used the Lightning App Builder's drag-and-drop canvas to create and modify the custom pages.



❖ Record Pages

Various Record pages can be created :

Steps:

(Same Steps can be applied to all pages)

1. Lightning App Builder → New → **Record Page** → Object = (select your object) eg : **Crop__c** → name eg: Crop Record Page.
2. Template: Choose any Template (Ex : **One Region with Right Sidebar** (or Two Columns).
3. Add Components as per requirement into Canvas (Drag & Drop)
4. Example:

   Top: **Highlights Panel** (compact layout show Crop Name, Quantity, Price).

   Main: **Record Details** (all fields: Variety, Harvest Date, Quantity, Price).

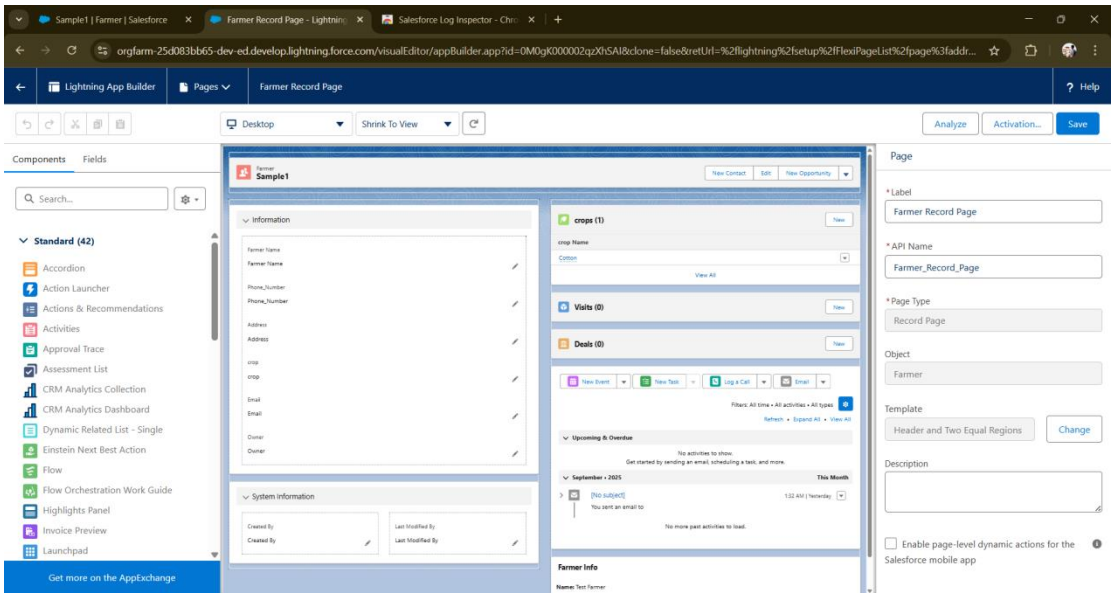
   Right sidebar:

   Related List — Single: select **Deals** (show which deals reference this crop)

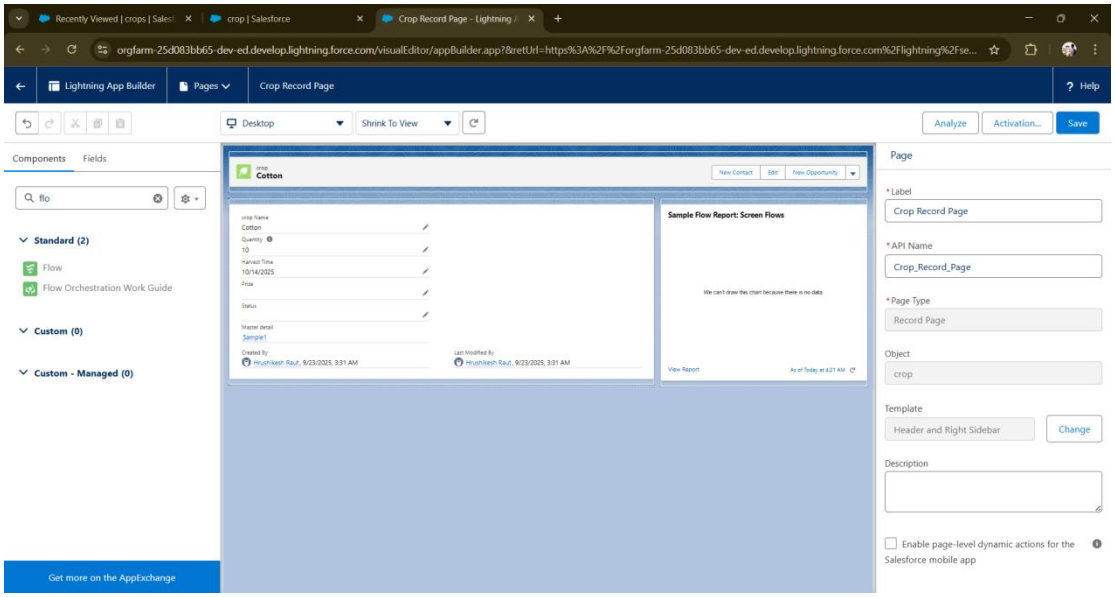
   Report Chart: show a small chart (e.g., Crop sales trend or low-stock chart).

5. Save → Activate → assign app & profiles (eg : AgriConnect app & profiles.)

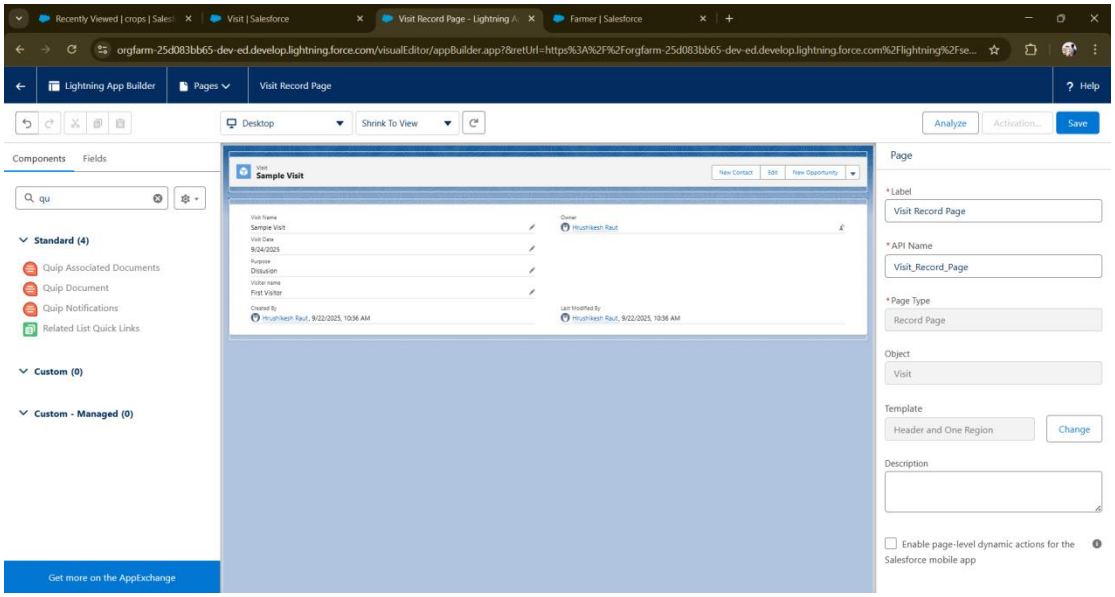
Farmer Record Page



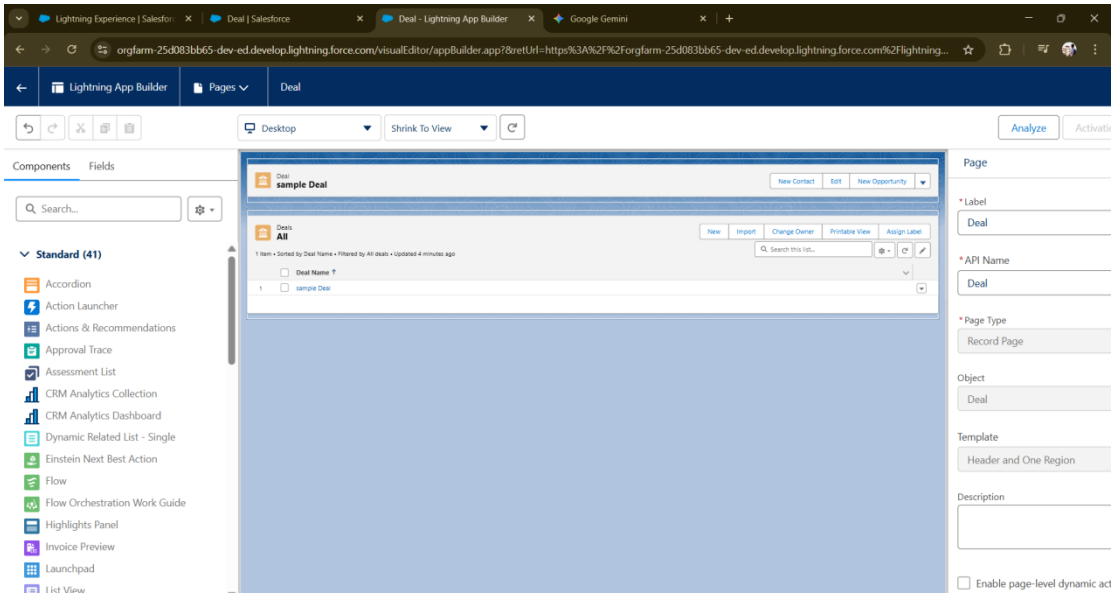
Crop Record Page



Record Page

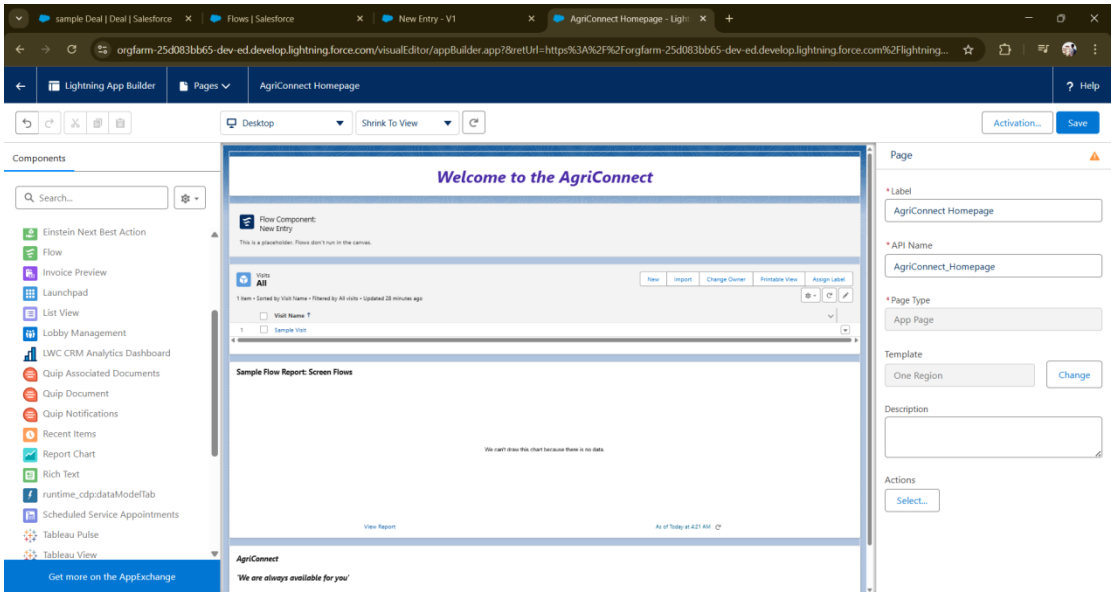


Deal Record Page



AgriConnect HomePage

(This page is a app page not a record page)



❖ Lightning Web Component (LWC)

(Use LWC when you need custom UI/UX. Below are small, clear examples you can use in AgriConnect)

Step 1: Install & Open

Install the **Lightning Studio Chrome extension**.

Log in to your Salesforce Org in Chrome.

Open the Lightning Studio extension (you'll see an editor pop up in the org).

Step 2: Create a New Component

Click **+ New → Lightning Web Component**.

Give your component a name, e.g., farmerCard.

Select where to save → usually under lwc/farmer Card.

The extension will auto-create **3 files**:

farmerCard.html (markup/template)

farmerCard.js (JavaScript logic)

farmerCard.js-meta.xml (exposure config)

Step 3 : Write Necessary Code

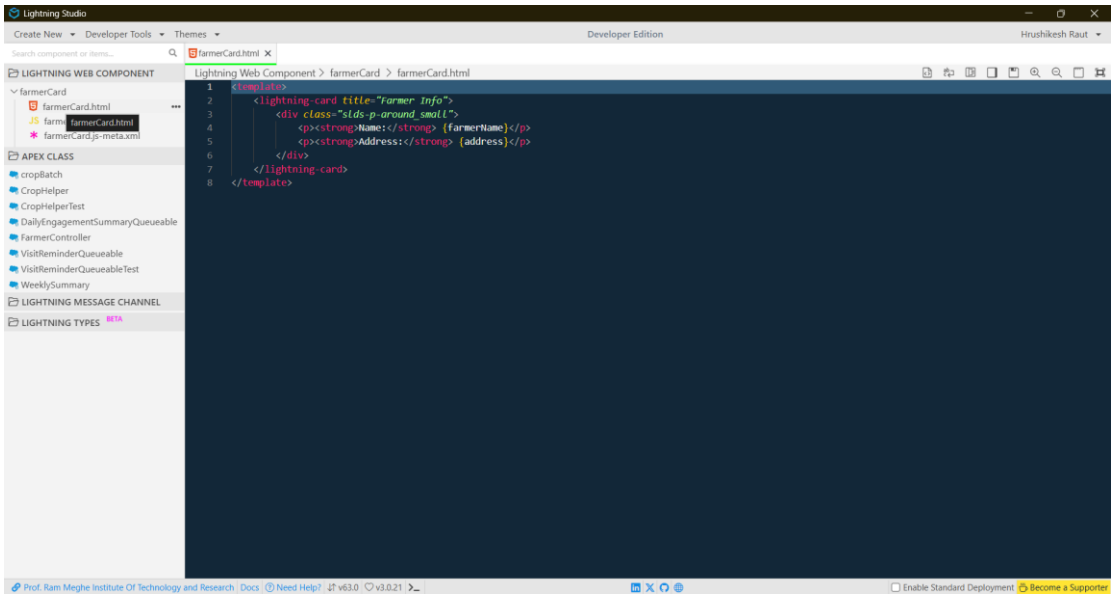
Step 4: Save & Deploy

In Lightning Studio, click **Save** → it auto-deploys to your Salesforce org.

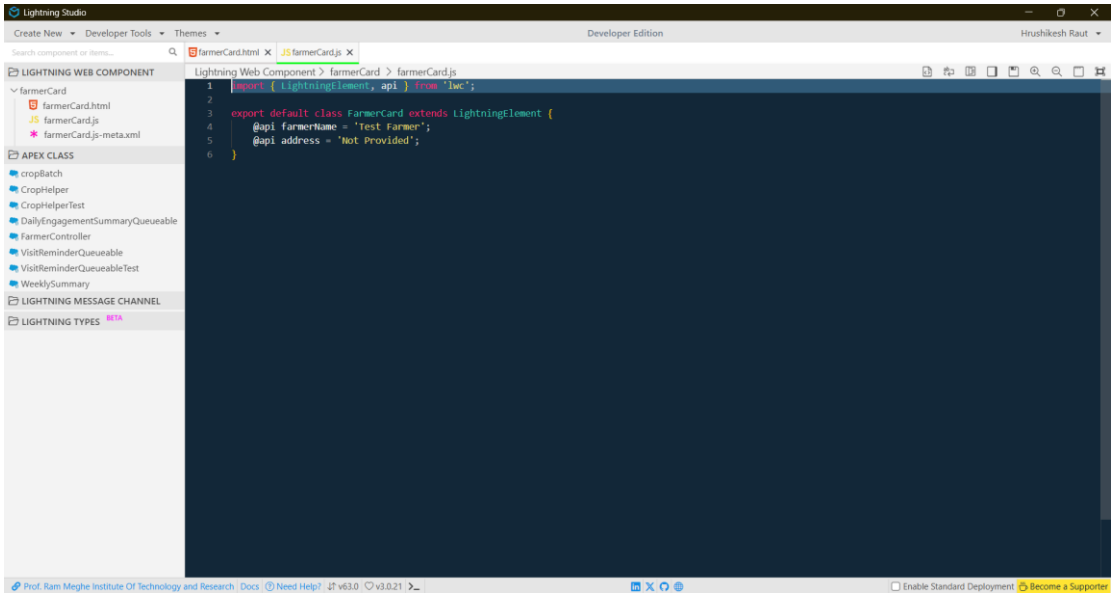
If successful, you'll see a confirmation message.

(This saves the setup and configuration of VS studio or Salesforce CLI)

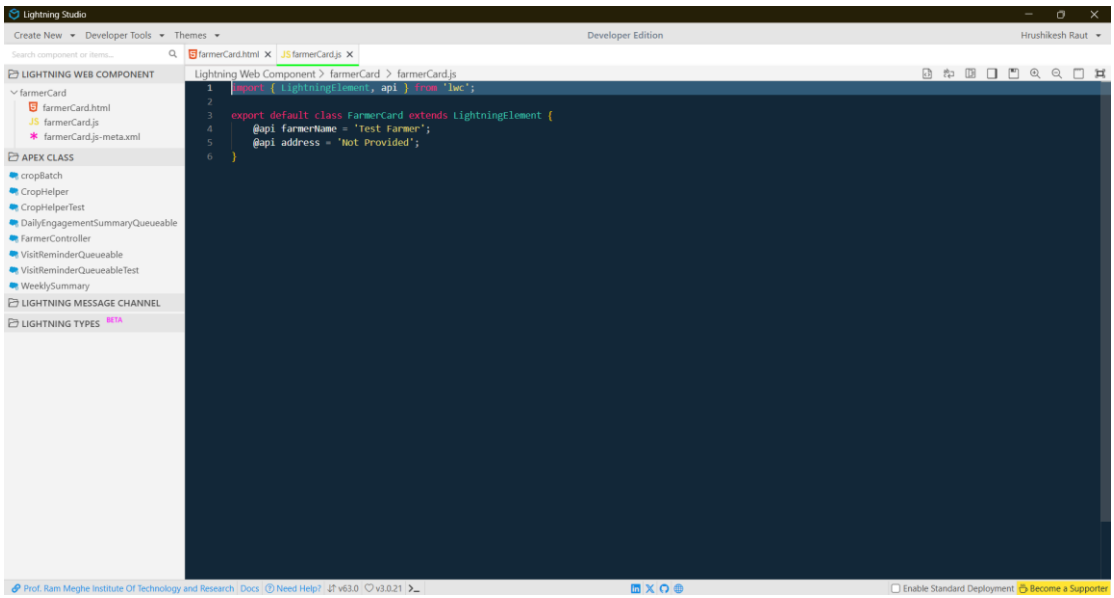
farmerCard.html (markup/template) :



farmerCard.js (JavaScript logic) :



farmerCard.js-meta.xml (exposure config) :



(Above is a sample for UI development using LWC)

OR,

❖ Flows

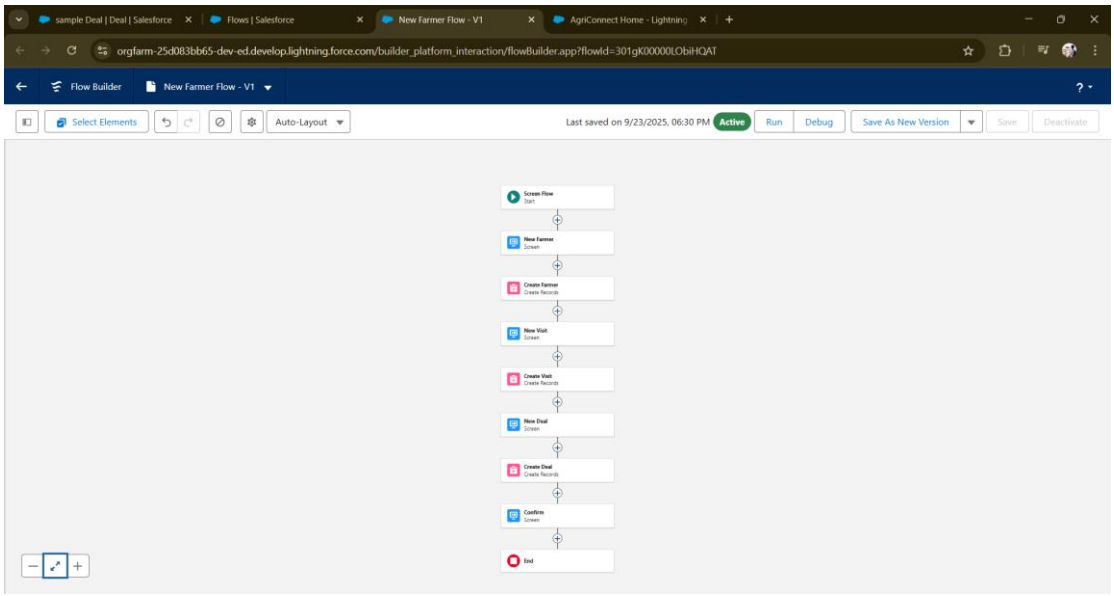
(Flows can be used as an alternative for LWC)

Steps:

1. Navigate to Flows: From the App Launcher, search for Flow
2. Start a New Flow: In the Flows panel, click the New button.
3. Select Flow Type: For Example Choose Screen Flow from the options
4. Add a Screen Element: On the Flow Builder canvas, click the + icon and select the Screen element.
5. Configure the Screen:

6. Give your screen element a unique API Name and a descriptive Label.
7. In the left panel, find the desired components (e.g., Text, Lookup, Email) and drag them onto your screen.
8. For each component, provide a unique API Name and a user-friendly Label.
9. Set components as required if necessary, using the Configure Footer option to customize button labels (like "Next" or "Finish").
10. Add Other Elements (if needed): Use other elements like Create Records or Get Records to interact with Salesforce data and connect them to your screen elements.
11. Save and Activate : Click the Save button in the Flow Builder to save then click Activate

A Screen Flow for new Entry in the records :



Phase 7: Integration & External Access

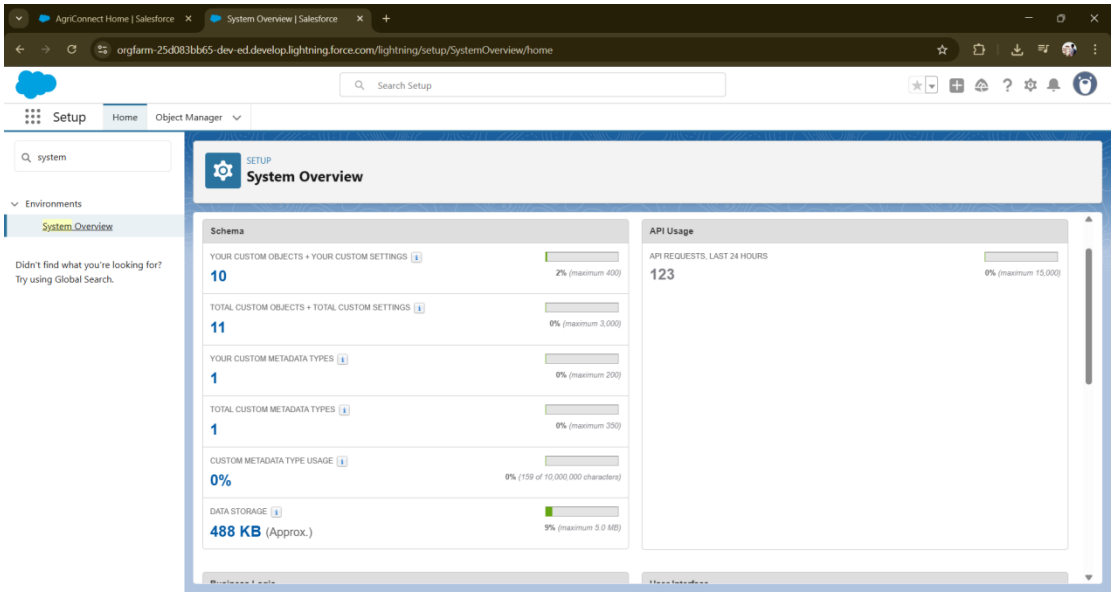
1. API Limits

• Salesforce tracks the number of **API calls** made by external systems or integrations.

• In **AgriConnect**, all major actions (Farmer Registration, Visits, Buyer related, Deals etc) are handled **inside Salesforce**.

• No external APIs are currently being used in this project, so **API limits are not a concern**. (Below screenshots shows 123 API requests that is because System administrator has some other applications like marketing but our CRM don't)

• You can monitor API usage in **Setup → System Overview**, but no special configuration is required.



Phase 8: Data Management & Deployment

❖ Duplicate Management

Prevent duplicate **Farmer** records (by Email/Phone/Name) and surface potential duplicates during data entry or import.

1) Create the Matching Rule

1. Setup → Quick Find → **Matching Rules** → **New Rule**.

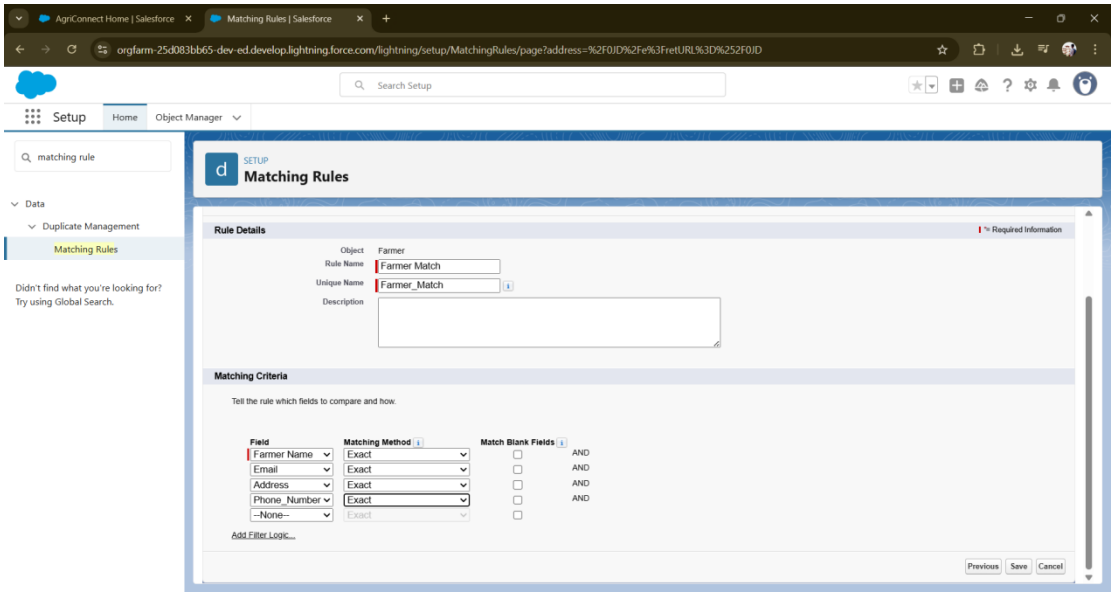
2. **Object: Farmer**

3. **Rule Name:** Farmer Match (or any name).

4. **Add Matching Criteria:**
   - Field = **Farmer Name** → Matching Method = **Exact**.
   - Field = **Email** → Matching Method = **Exact**.
   - Field = **Address** → Matching Method = **Exact**.
   - Field = **Phone** → Matching Method = **Exact**.

5. **Save** the rule.

6. Click **Activate** (only active rules can be used by Duplicate Rules).



2) Create the Duplicate Rule

1. Setup → Quick Find → **Duplicate Rules** → **New Rule**.

2. **Object: Farmer**. (same as that of matching rule)

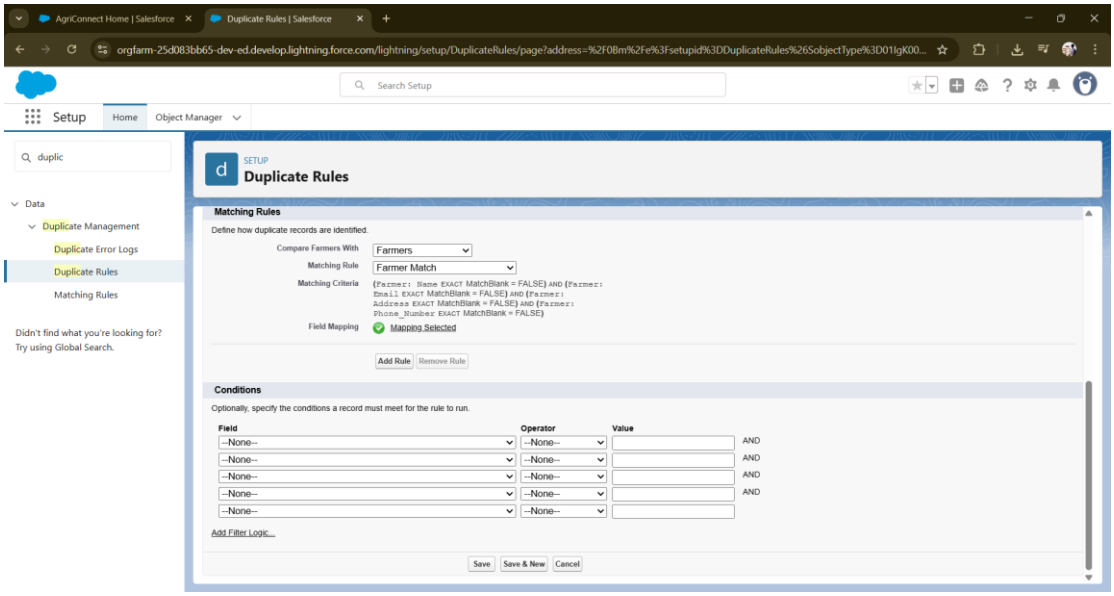
3. **Rule Label:** Farmer Duplication Rule

4. Under **Matching Rules**, click **Add** and select the 'Farmer Match' matching rule you just activated.

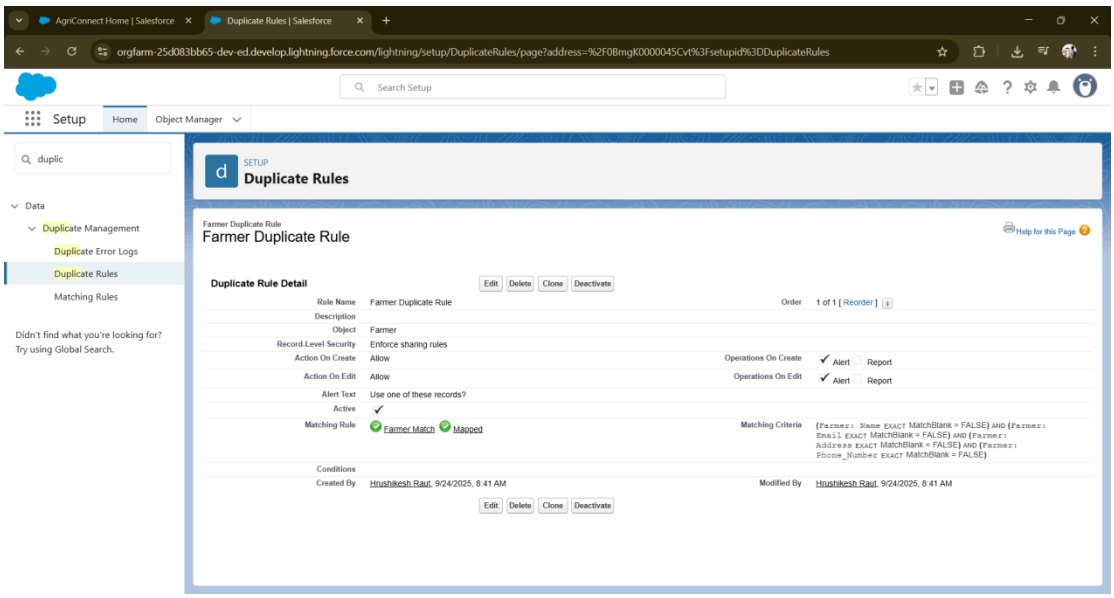
5. **Action on Create:** choose **Alert** (start in Alert mode while testing).

6. **Action on Edit:** choose **Alert**.

7. **Save**, then click **Activate**.



It will look like this after Activation



❖ Data Backup

Steps:

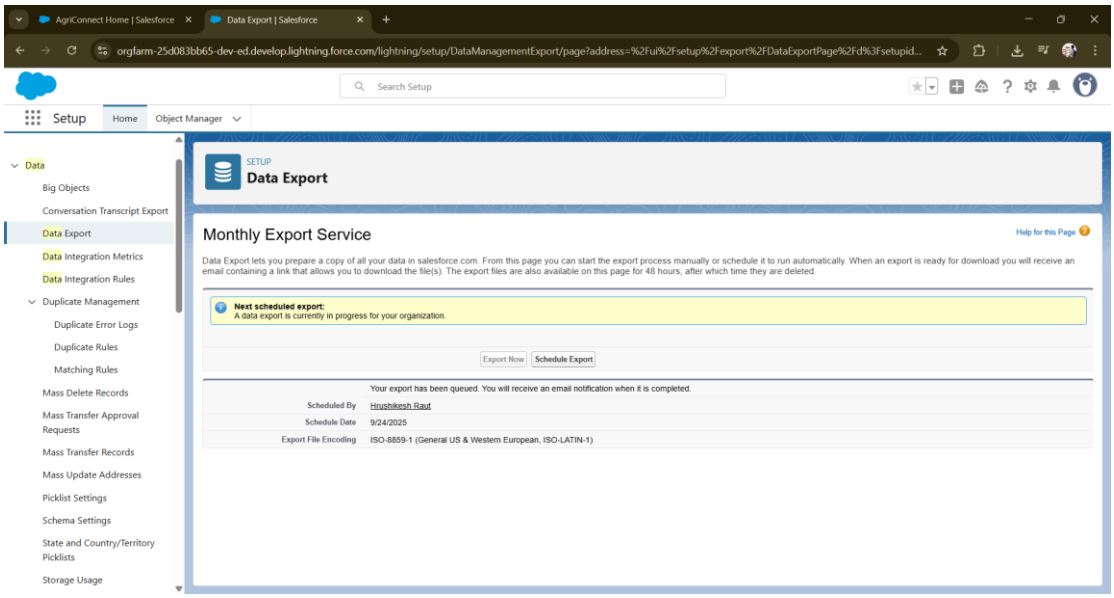
□ Go to **Setup**.

□ In Quick Find, type **Data Export** → click **Data Export**.

□ Choose one:
□ **Export Now** → run a one-time backup.
□ **Schedule Export** → set weekly/monthly backups.

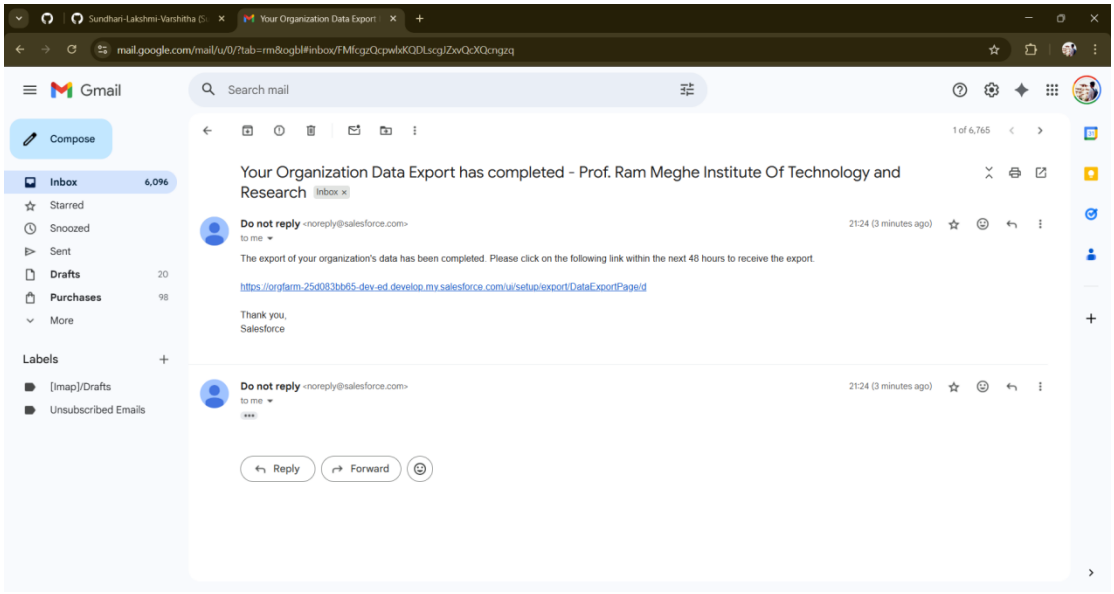
□ Select the objects you want:
□ **Farmer, Visit, crop, Deal, Buyer** and any standard objects you use (e.g., Users).

□ Click **Start Export** (for immediate) or **Schedule Export** (for scheduled).



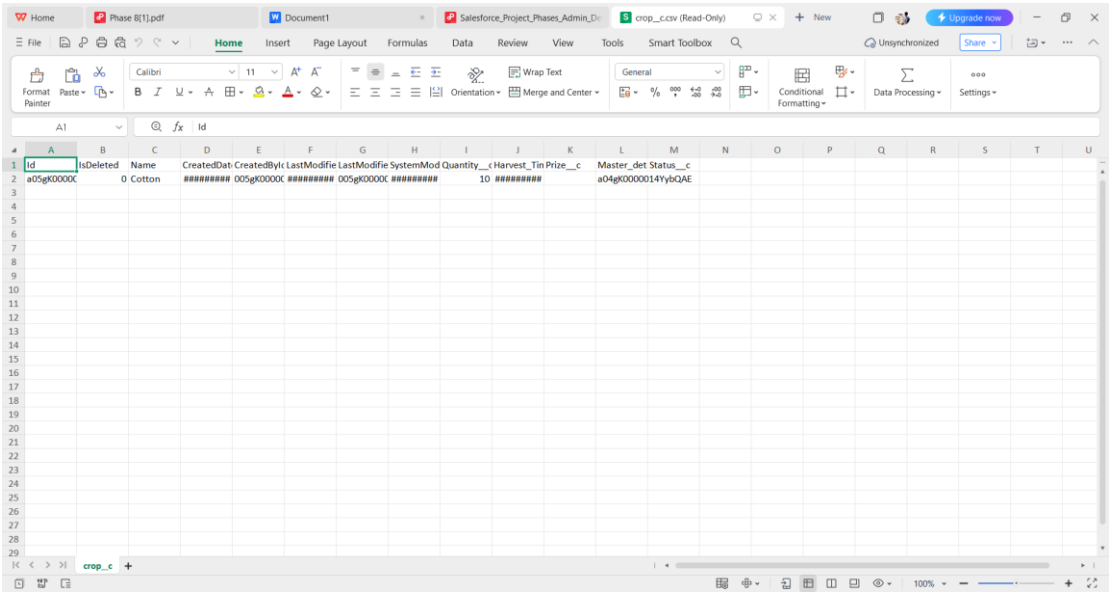
□ Wait → Salesforce emails you when the backup is ready.
□ Download the **.zip file** from the export page → extract CSV files.

□ ***Store the backup securely*** (encrypted drive, company server, cloud storage).

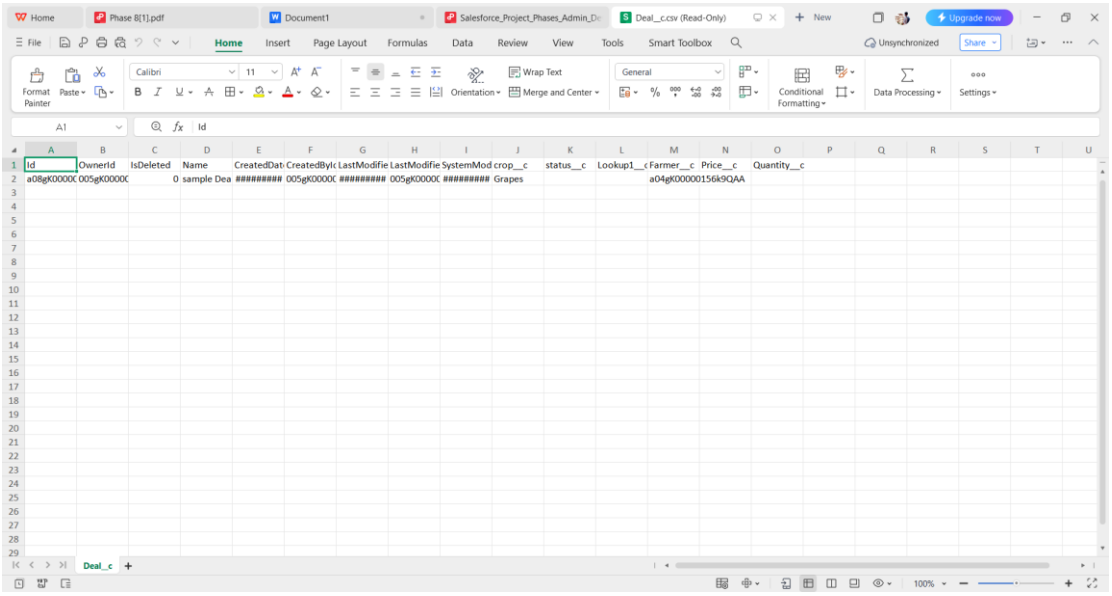


Extracted ‘ CSV ‘ from zip :

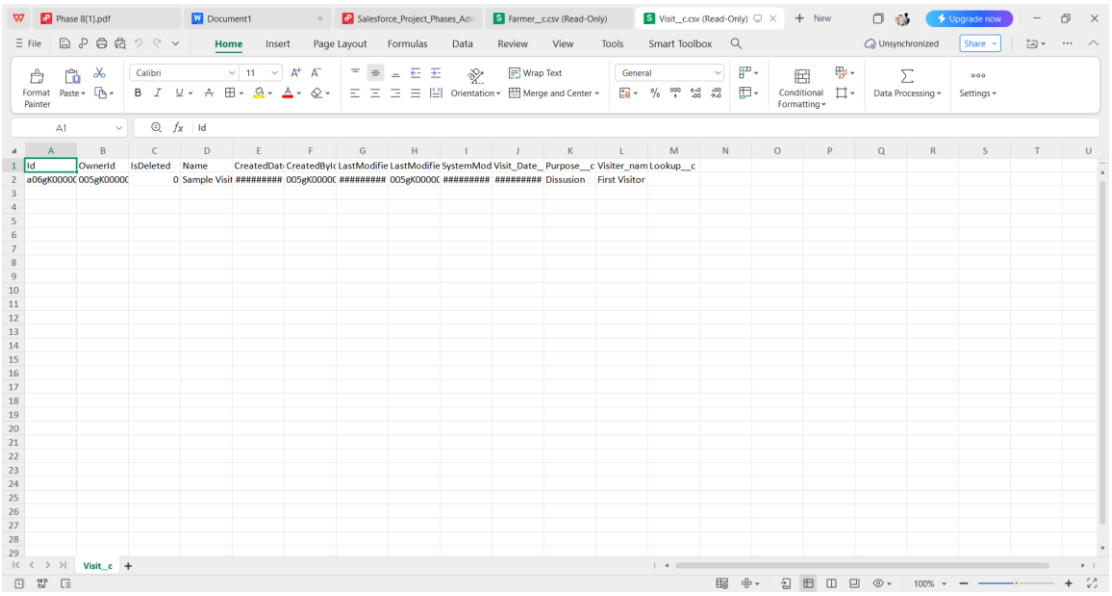
crop_c.csv



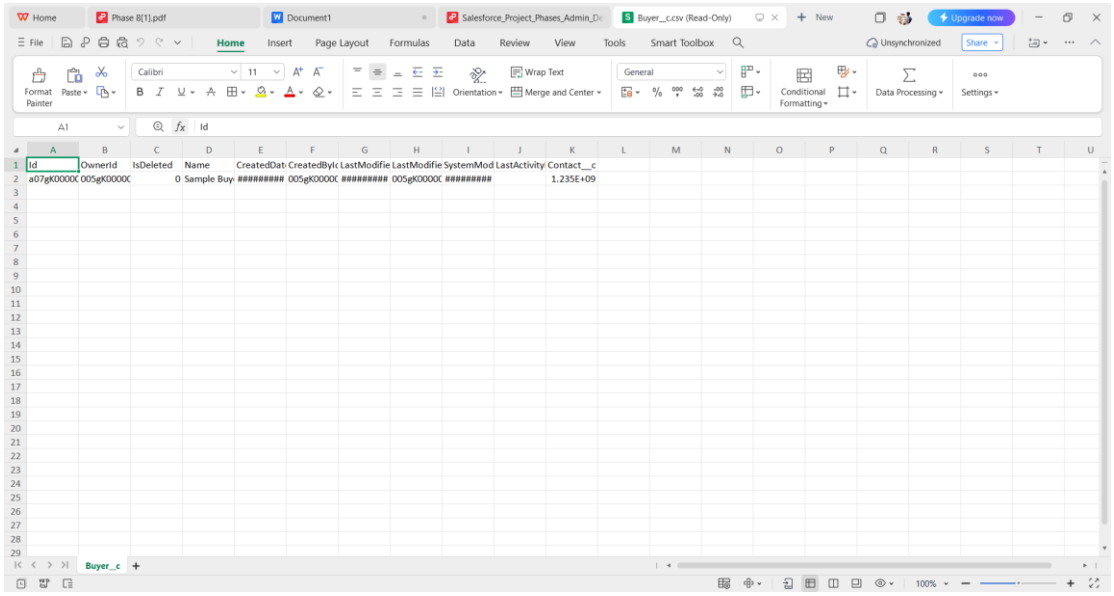
Deal_c.csv



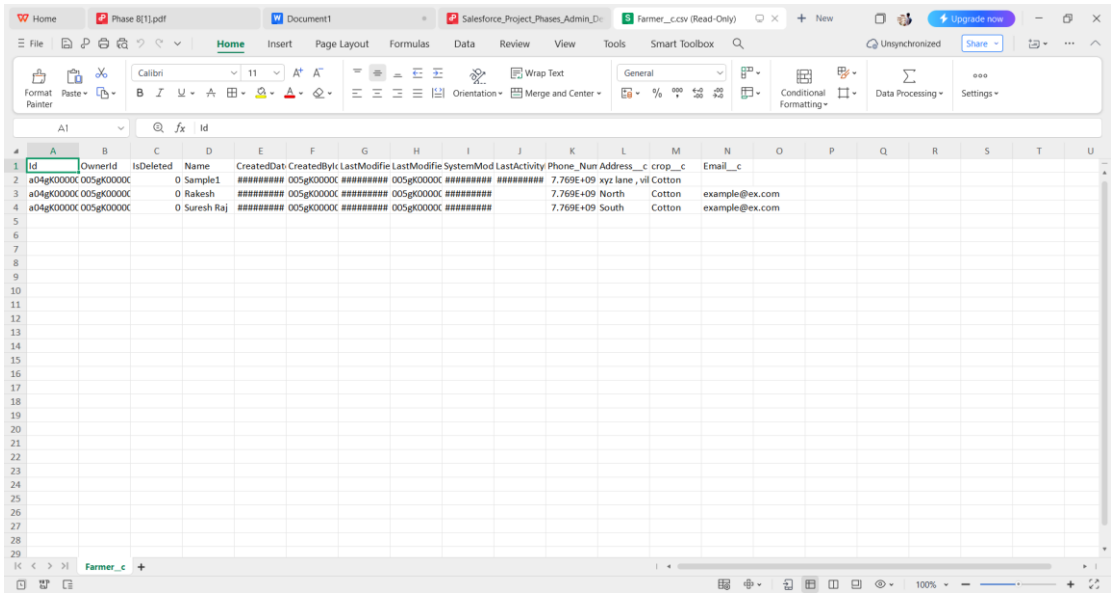
Visit_c.csv



Buyer_c.csv



Farmer_c.csv



Phase 9: Reporting Dashboard & Security review

◆ Report Type

Decide which *custom report types* you need beyond standard types (e.g., Farmer with visit or Deal).

Steps: Setup → Report Types → New Custom Report Type

Primary Object: Farmer

Related Object: as per requirement crop/deal/visit (select "Each 'Farmer' may or may not have related records")

Save & Deploy

Why: custom report types let you show object combinations not available in standard report types.

◆ Reports

Reports helps to track effectively in the AgriConnect

A. Create Reports

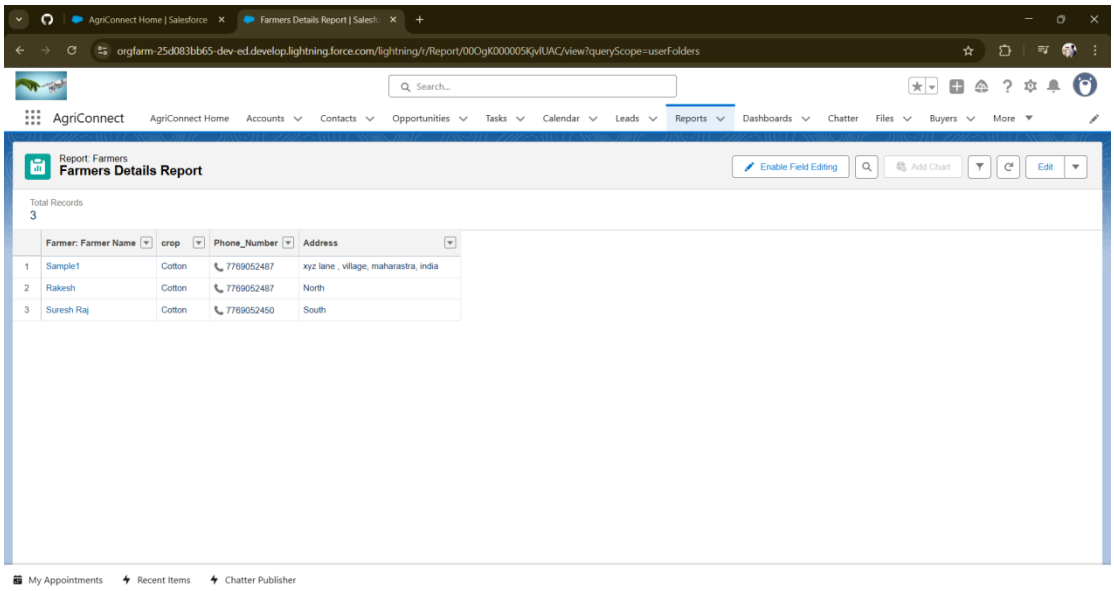
1. Go to **App Launcher → Reports → New Report**.

2. Select the report type: e.g., **Products** or **Orders**.

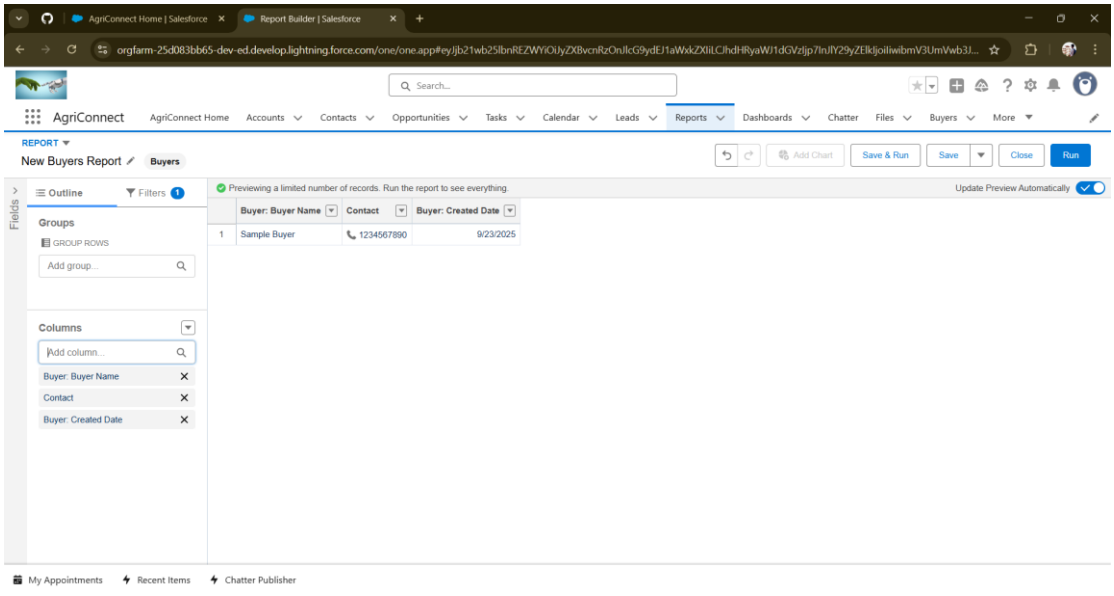
3. Add fields:

- For Farmer Detail Report:
  - ✧ Farmer name
  - ✧ crop

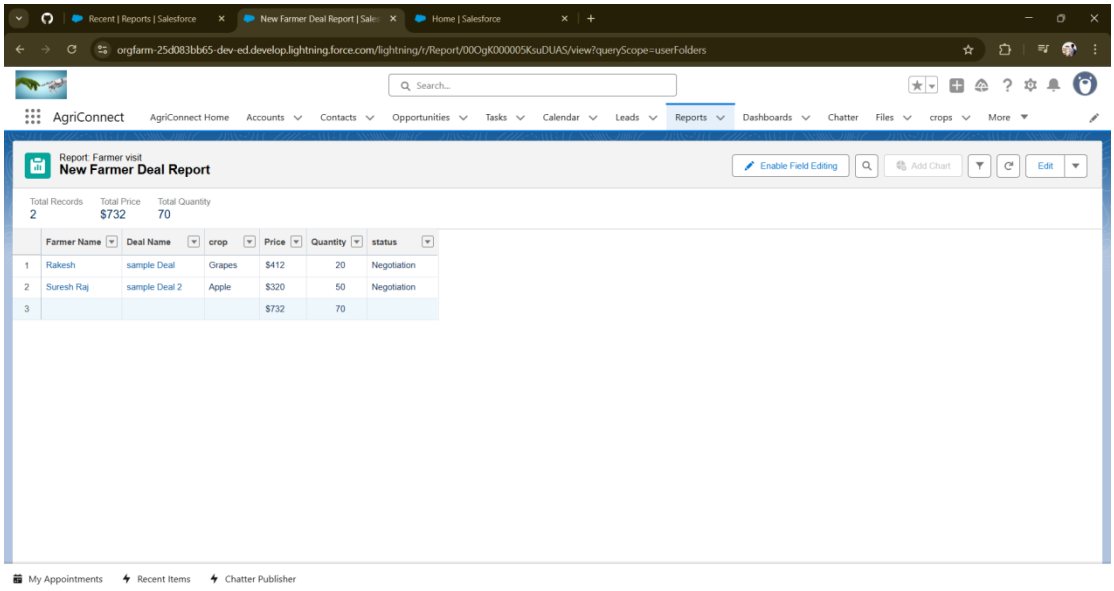
- ✧ Phone_Number
- ✧ Address



- For New Buyer Report:
  - ✧ Buyer Name
  - ✧ Contact
  - ✧ Created Date



- For New Farmer Deal Report :
    - ✧ Farmer Name
    - ✧ Deal Name
    - ✧ Crop
    - ✧ Price
    - ✧ Quantity
    - ✧ status



◆ Dashboard

Dashboards provide the visual insights.

A. Create Dashboard

1. Go to App Launcher → Dashboards → New Dashboard.

2. Enter dashboard name, e.g. AgriConnect Dashboard.

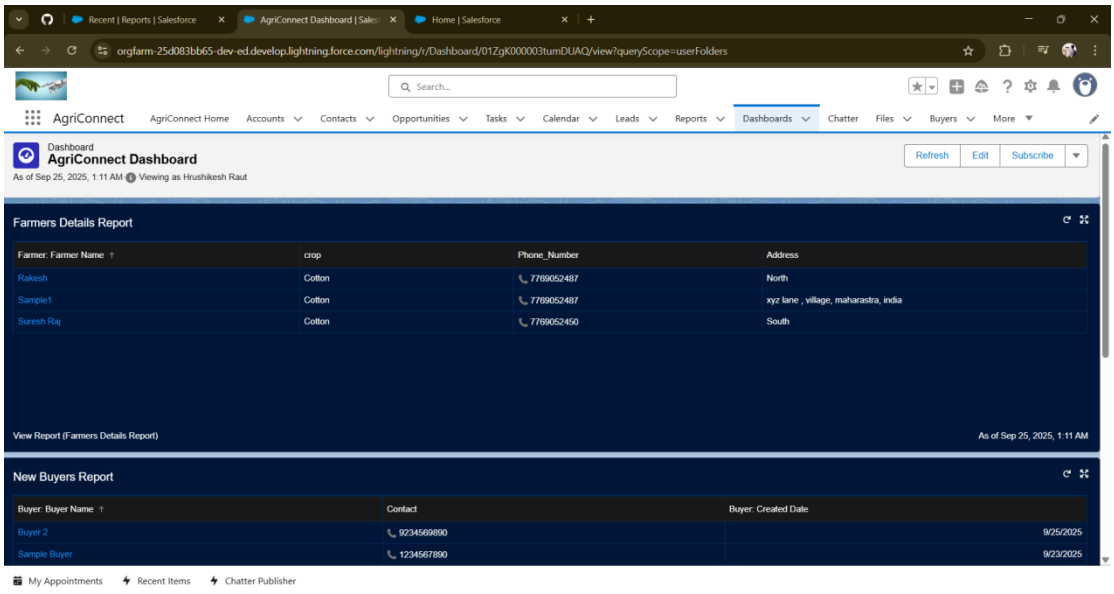
3. Add components using your reports as data sources (charts, tables, graphs).

4. Configure components to show key metrics:

- Farmer Details
- Buyer Details
- New Farmer Deals

5. Arrange components for clear visualization.

6. Save the dashboard



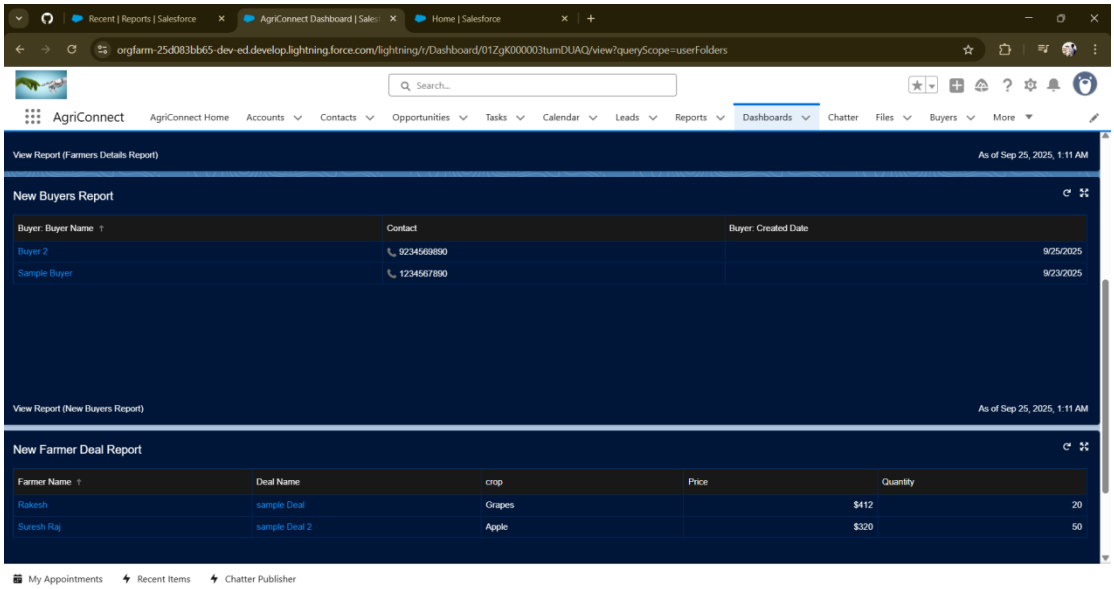
B. Dashboard Benefits

• Easy visualization about new records.

• Helps identify records and issues quickly.

• Supports faster decision-making for officer.

• Enhances Monitoring, eg. Officer can keep an eye.



• Security is very essential, specially a persons data. Sharing Settings provide the feature about data sharing like who can see, what can see etc. E.g For a Farmer it is private