Salesforce Virtual Internship Program

SmartInternz

To supply leftover food to poor

NAME : BALLIKURI SHERLIE ANGEL
E-MAIL : 322103382002@gvpce.ac.in
COLLEGE NAME : GAYATRI VIDYA PARISHAD COLLEGE OF ENGINEERING

Project Title: Food Connect - To Supply Leftover Food to the Poor

1. Project Overview:

This project focuses on the development of a Salesforce-based application, "Food Connect," aimed at facilitating the ecient distribution of leftover food from donors to underprivileged communities. By leveraging Salesforce's robust platform, the application optimizes daily operations, improves data accuracy, and provides actionable insights into food donation and distribution processes.

Designed to address the pressing issue of food wastage and hunger, this application automates critical processes such as tracking food donations, inventory management, volunteer coordination, and generating real-time reports.

The primary challenge addressed by this project is the inecient handling of food donations, which can lead to delays and wastage. By providing a comprehensive, user-friendly solution, Food Connect ensures effective resource management, strengthened donor relationships, and seamless reporting.

Through this project, Food Connect aims to achieve:

● Operational Excellence: Automating routine processes to ensure timely collection and distribution of food.

● Data-Driven Decision Making: Equipping stakeholders with real-time insights into donations, distributions, and volunteer activities.

● Scalability and Eciency: Supporting long-term growth with a secure, scalable, and exible solution.

2. Objectives

Business Goals:

1. Streamlining Operations: Automating processes such as food donation tracking, inventory updates, and volunteer task allocation.

2. Improved Decision-Making: Delivering detailed reports and dashboards for real-time analytics, enabling better allocation of food resources and strategic planning.

3. Enhancing Donor Relationships: Providing personalized insights into donor contributions and feedback.

4. Ensuring Data Security: Implementing role-based access controls to restrict sensitive information to authorized users.

Specific Outcomes:

● A centralized platform to monitor and manage food donations and distributions effectively.

● Real-time automated reports on food collection, distribution, and beneciary statistics.
● Reduction of manual errors in data entry and calculations.

● User-friendly dashboards to visualize key metrics and insights.

3. Salesforce Key Features and Concepts Utilized

1. Reports and Dashboards:

■ Automated generation of daily, weekly, and monthly reports on food donations, inventory levels, and distributions.

■ Dashboards displaying critical metrics such as most active donors, top distribution locations, and food wastage trends.

2. Rollup Summary Fields:

■ Summarizes data from child records to parent records in master-detail relationships.
■ Examples:

○ Total food donated by each donor.

○ Total meals distributed in a specic location.

3. Cross-Object Formula Fields:

■ Enables calculations across related objects.

■ Example: Total beneciaries reached calculated using Quantity of Food Distributed × Average Meals per Unit.

4. Validation Rules:

■ Ensures data accuracy and completeness.

■ Example: The ISBLANK formula prevents saving records with missing mandatory elds, such as food quantity or beneciary details, and displays error messages to guide users.

5. Permission Sets and Organization Wide Defaults (OWD):

■ Congures access levels based on roles:

○ Admin: Complete access to all records.

○ Volunteer Coordinator: Access restricted to volunteer-related records.

○ Volunteer: Limited access based on assigned tasks.

■ Ensures sensitive data is protected while enabling collaboration.

4. Detailed Steps to Solution Design:

Requirement Gathering:

● Conducted discussions with stakeholders, including donors, volunteers, and community leaders, to understand operational pain points, reporting needs, and goals.

Data Model Design:

● Created custom objects for "Food Inventory," "Donor," "Distribution," and "Beneciary."

● Defined relationships:

■ Master-detail relationship between "Food Inventory" and "Donor."

■ Lookup relationship between "Distribution" and "Beneciary."

User Interface (UI) Design:

● Developed intuitive Lightning Pages tailored to different user roles (e.g., Admin Dashboard, Donation Entry Form).

● Included custom components to facilitate data entry and quick access to reports.

Business Logic Implementation:

● Automated workows for low inventory alerts and donor notications.

● Developed Apex classes and triggers for advanced calculations and inventory updates.

Reports and Dashboards:

● Congured reports to highlight:

■ Daily food donations and distributions.

■ Inventory levels and wastage trends.

■ Volunteer activity metrics.

● Dashboards provide real-time visualizations for quick decision-making.

Documentation and Screenshots:

● Documented all components, congurations, and workows with accompanying screenshots for clarity and reference.

Object : Salesforce objects are database tables that permit you to store data that is specic to an organization.

Creating required objects :

To Navigate to Setup page To create an object:

1. From the setup page > Click on Object Manager > Click on Create > Click on Custom Object.

2. On Custom object dening page:

3. Enter the label name, plural label name, click on Allow reports, Allow search.

4. Click on Save.

To create an object:

1. From the setup page >> Click on Object Manager >> Click on Create >>Click on Custom Object. 1. Enter the label name >> Venue

2. Plural label name >> Venues

3. Enter Record Name Label and Format

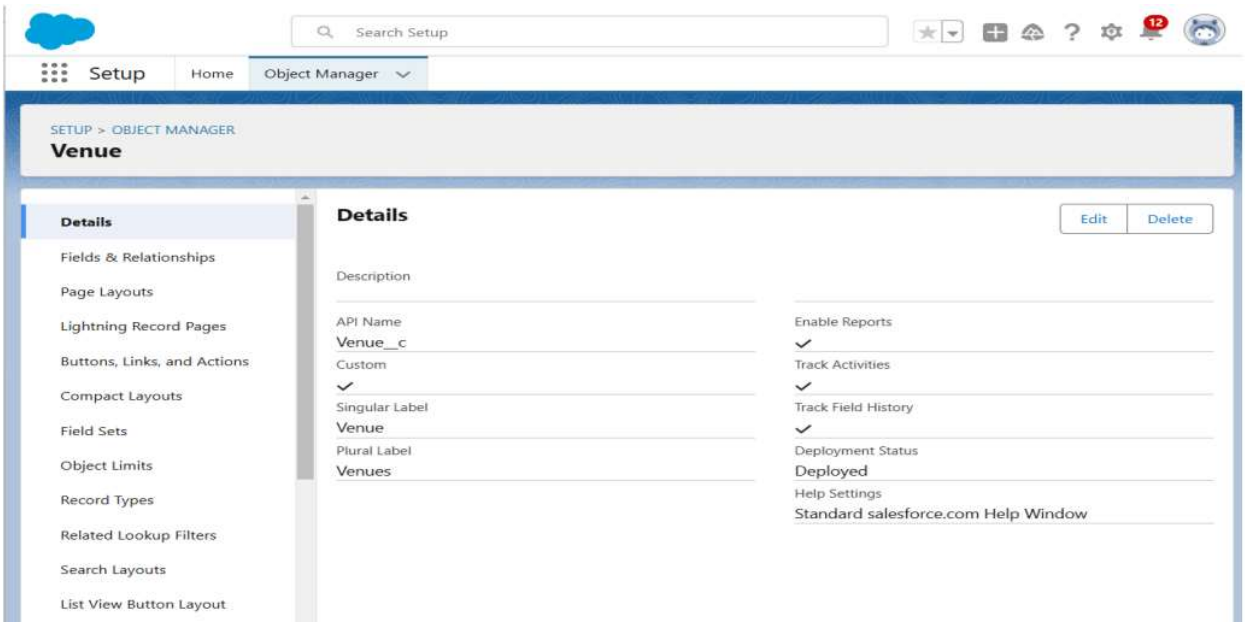
● Record Name >> Venue Name

● Data Type >> Text

4. Click on Allow reports and Track Field History,Allow Activities.

5. Allow search >> Save

Like the below:



Tabs

A tab is like a user interface that is used to build records for objects and to view the records in the objects .

Types of Tabs:

Custom Tabs: Custom object tabs are the user interface for custom applications that you build in salesforce.com. They look and behave like standard salesforce.com tabs such as accounts, contacts, and opportunities.

Web Tabs Web: Tabs are custom tabs that display web content or applications embedded in the salesforce.com window. Web tabs make it easier for your users to quickly access content and applications they frequently use without leaving the salesforce.com application.

Visualforce Tabs: Visualforce Tabs are custom tabs that display a Visualforce page. Visualforce tabs look and behave like standard salesforce.com tabs such as accounts, contacts, and opportunities.

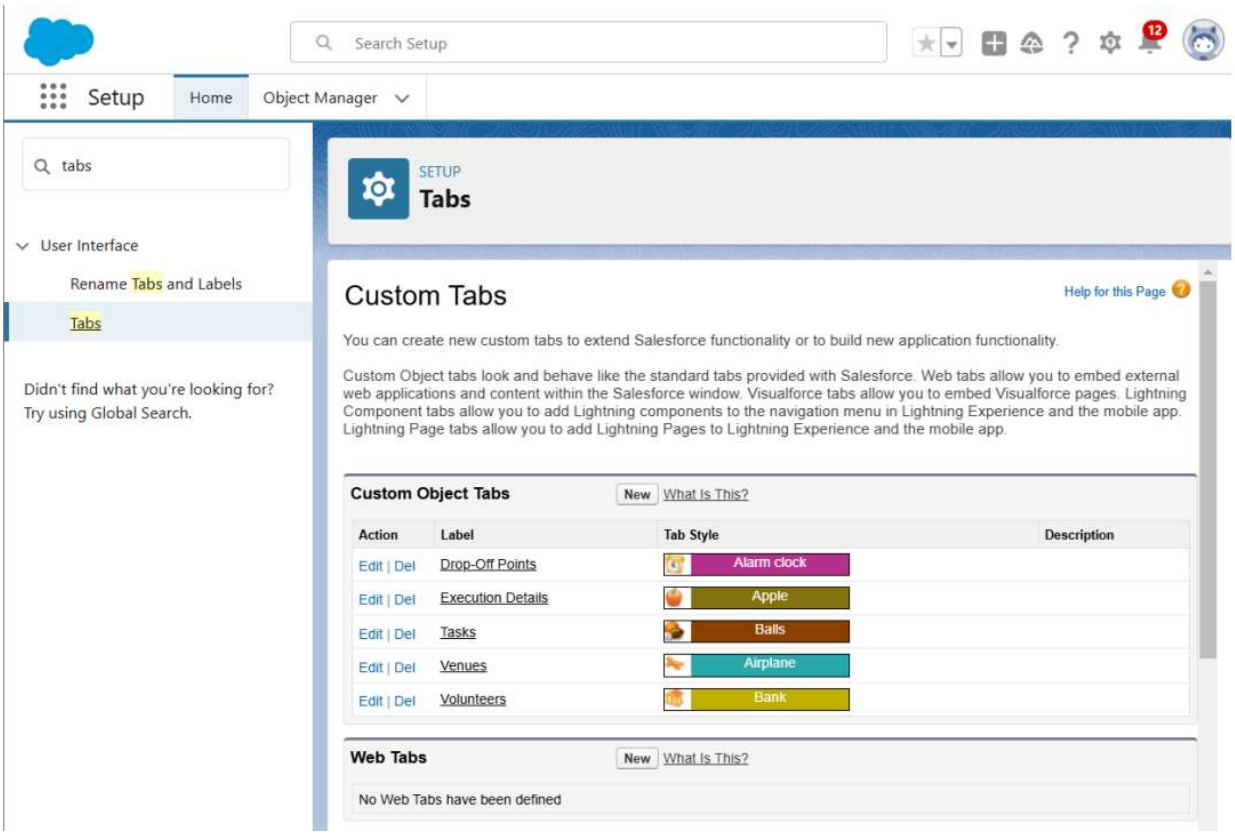
Lightning Component Tab:s Lightning Component tabs allow you to add Lightning components to the navigation menu in Lightning Experience and the mobile app. Lightning Page Tabs Lightning Page Tabs let you add Lightning Pages to the mobile app navigation menu. Lightning Page tabs don't work like other custom tabs. Once created, they don't show up on the All Tabs page when you click the Plus icon that appears to the right of your current tabs. Lightning Page tabs also don't show up in the Available Tabs list when you customize the tabs for your apps.

To create a Tab:

1. Go to setup page >> type Tabs in Quick Find bar >> click on tabs >> New (under custom object tab)

2. Select Object(Venue) >> Select the tab style >> Next (Add to proles page) keep it as default >> Next (Add to Custom App) uncheck the include tab .

3. Make sure that the Append tab to users' existing personal customizations is checked.
4. Click save



The Lightning App: An app is a collection of items that work together to serve a particular function. In Lightning Experience, Lightning apps give your users access to sets of objects, tabs, and other items all in one convenient bundle in the navigation bar. Lightning apps let you brand your apps with a custom color and logo. You can even include a utility bar and Lightning page tabs in your Lightning app. Members of your org can work more eciently by easily switching between apps.

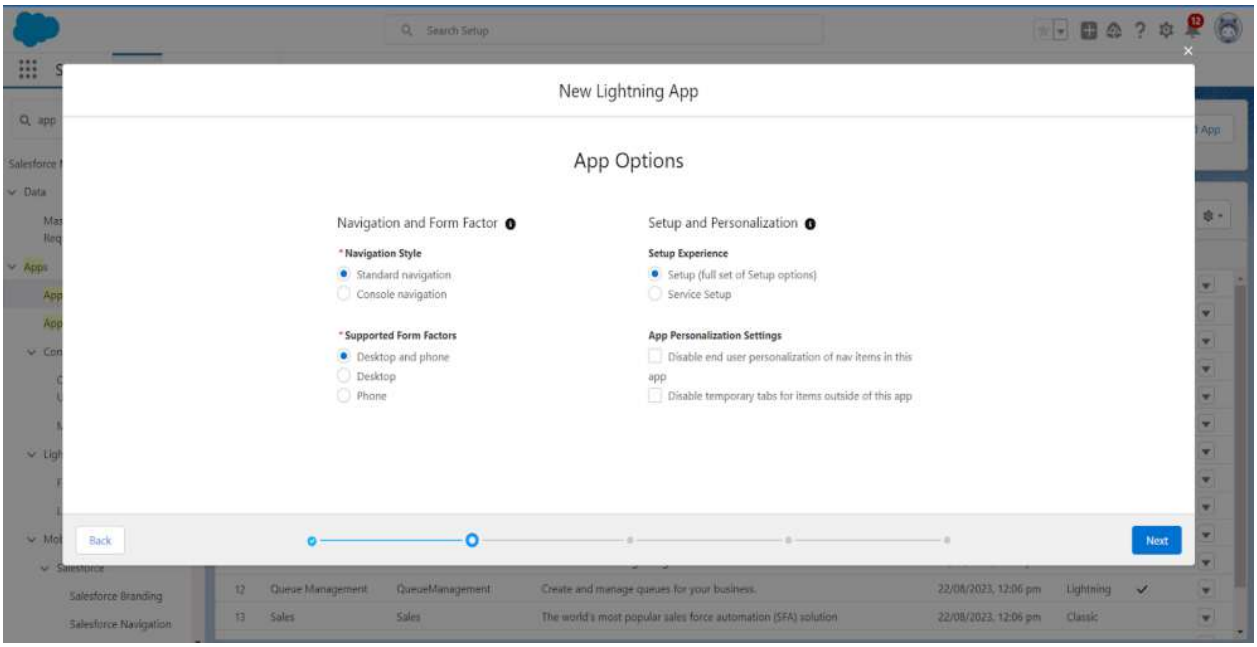
To create a lightning app page:

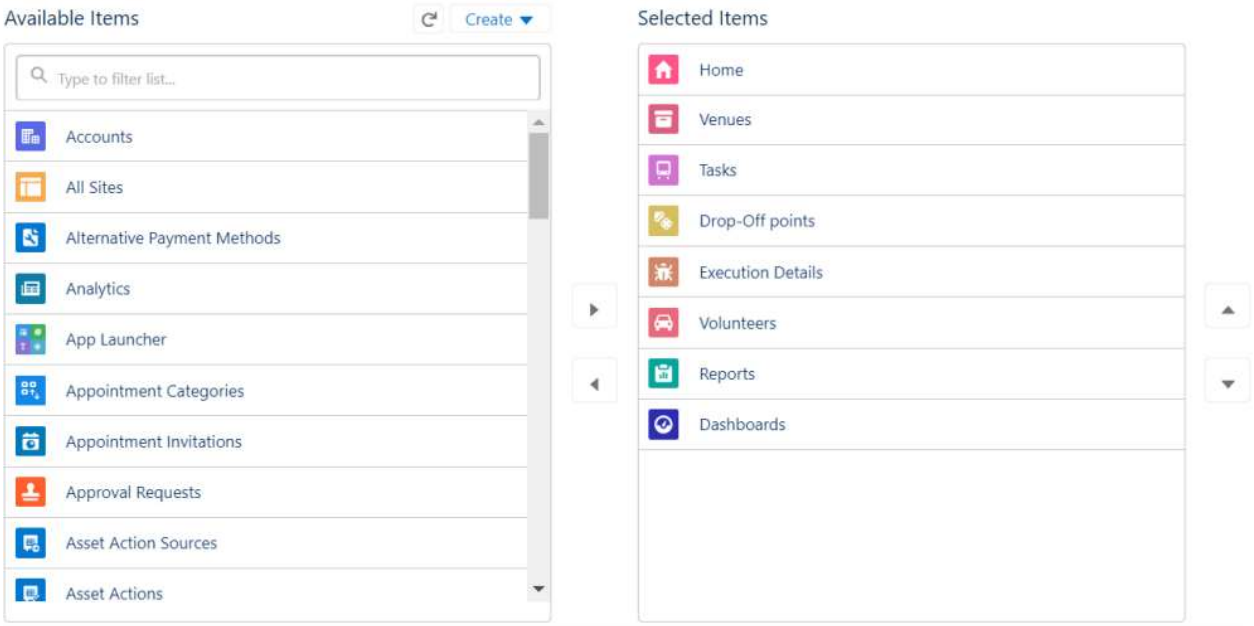
1.Go to setup page >> search “app manager” in quick nd >> select “app manager” >> click on New lightning App

2. Fill the app name in app details and branding as follow App Name : FoodConnect Developer Name : This will auto populated Image : optional (if you want to give any image you can otherwise not mandatory) Primary color hex value : keep this default.
3.Then click Next >> (App option page)Set Navigation Style as Standard Navigation >> Next.

4. (Utility Items) keep it as default >> Next.

5. To Add Navigation Items: Search for the item in the (Home, Venue, Drop-Off Point, Task, Volunteer, Execution Details, Reports) from the search bar and move it using the arrow button >> Next >> Next.





Salesforce Fields Overview

In Salesforce, **Fields** represent the data stored in the columns of a relational database. These fields hold valuable information required for a specific object, simplifying searching, deleting, and editing records.

Types of Fields

1. **Standard Fields**:
   These are predefined fields in Salesforce that perform standard tasks. Standard Fields cannot be deleted unless they are non-required fields. Some common Standard Fields include:
   - **Created By**
   - **Owner**
   - **Last Modified**
   - **Fields created during object creation**
2. **Custom Fields**:
   Custom Fields are flexible and can be modified to meet specific requirements. Unlike Standard Fields, their inclusion in records is optional, giving users complete control to add or remove them as needed.

Creating Relationship Fields in Objects

1. Lookup Relationship Field on Volunteer Object

- Navigate to **Setup > Object Manager** and search for the object "Volunteer."
- Go to **Fields & Relationships > New** and select **Master-Detail Relationship.**
- Choose the related object **Drop-Off Point** and click **Next.**
- Set the **Field Name** as `Drop_Off_Point` (Field Label is auto-generated).
- Click **Next > Next > Save.**

2. Master-Detail Relationship Field on Execution Details Object

- Navigate to **Setup > Object Manager** and search for the object "Execution

Details."

- Go to **Fields & Relationships > New** and select **Master-Detail Relationship.**
- Choose the related object **Volunteer** and click **Next.**
- Set the **Field Name** as `Volunteer` (Field Label is auto-generated).
- Click **Next > Next > Save.**

3. Additional Master-Detail Relationship Field on Execution Details Object

- Repeat the above steps, selecting the related object **Task.**
- Set the **Field Name** as `Task` (Field Label is auto-generated).
- Click **Next > Next > Save.**

4. Lookup Relationship Field on Drop-Off Point Object

- Navigate to **Setup > Object Manager** and search for the object "Drop-Off Point."
- Go to **Fields & Relationships > New** and select **Lookup Relationship.**
- Choose the related object **Venue** and click **Next.**
- Set the **Field Name** as `Venue` (Field Label is `Venue__c`).
- Click **Next > Next > Save.**

5. Lookup Relationship Fields on Task Object

a. Sponsored By:

- Navigate to **Setup > Object Manager** and search for the object "Task."
- Go to **Fields & Relationships > New** and select **Lookup Relationship.**
- Choose the related object **Venue** and click **Next.**
- Set the **Field Name** as `Sponsored By` (Field Label is auto-generated).
- Click **Next > Next > Save.**

b. Drop-Off Point:

- Repeat the above steps, selecting the related object **Drop-Off Point.**
- Set the **Field Name** as `Drop-Off Point` (Field Label is auto-generated).
- Click **Next > Next > Save.**

Creation of Fields for the Venue Object

1. **Contact Email Field**:

- Navigate to **Setup > Object Manager** and search for the object "Venue."
- Go to **Fields & Relationships > New** and select **Email** as the data type.
- Fill in:
  - **Field Label**: Contact Email
  - **Field Name**: Contact_Email
  - **Required**: Checked
- Click **Next > Next > Save & New.**

2. **Contact Phone Field**:
   - Repeat the above steps, selecting **Phone** as the data type.
   - Fill in:
     - **Field Label**: Contact Phone
     - **Field Name**: Contact_Phone
     - **Required**: Checked
   - Click **Next > Next > Save & New.**
3. **Location Field**:
   - Select **Geolocation** as the data type.
   - Fill in:
     - **Field Label**: Location
     - **Decimal Places**: 4
     - **Field Name**: Location
     - **Description**: Enter the Geolocation of your Venue
   - Click **Next > Next > Save & New.**
4. **Venue Location Field**:
   - Select **Long Text Area** as the data type.
   - Fill in:
     - **Field Label**: Venue Location
     - **Field Name**: Venue_Location
   - Click **Next > Next > Save & New.**

Creation of Fields for the Drop-Off Point Object

1. **Location 2 Field**:
   - Navigate to **Setup > Object Manager** and search for "Drop-Off Point."
   - Go to **Fields & Relationships > New** and select **Geolocation** as the data type.
   - Fill in:
     - **Field Label**: Location 2
     - **Field Name**: Auto-generated

        - **Description**: Enter the Geolocation of the Drop-Off Point
        - **Decimal Places**: 4
    - Click **Next > Next > Save & New.**
2. **Distance Calculation Field**:
    - Select **Formula** as the data type.
    - Fill in:
        - **Field Label**: Distance Calculation
        - **Field Name**: Distance_Calculation
        - **Formula Return Type**: Number
        - **Formula**: `DISTANCE(Location_2__c, Venue__r.Location__c, 'km')`
    - Click **Next > Next > Save & New.**
3. **State Field**:
    - Select **Picklist** as the data type.
    - Fill in:
        - **Field Label**: State
        - **Field Name**: State
        - **Values**: Enter all state/UT names separated by new lines (e.g., Andhra Pradesh, Arunachal Pradesh, etc.).
        - **Required**: Checked
    - Click **Next > Next > Save & New.**

Creation of Fields for the Task Object

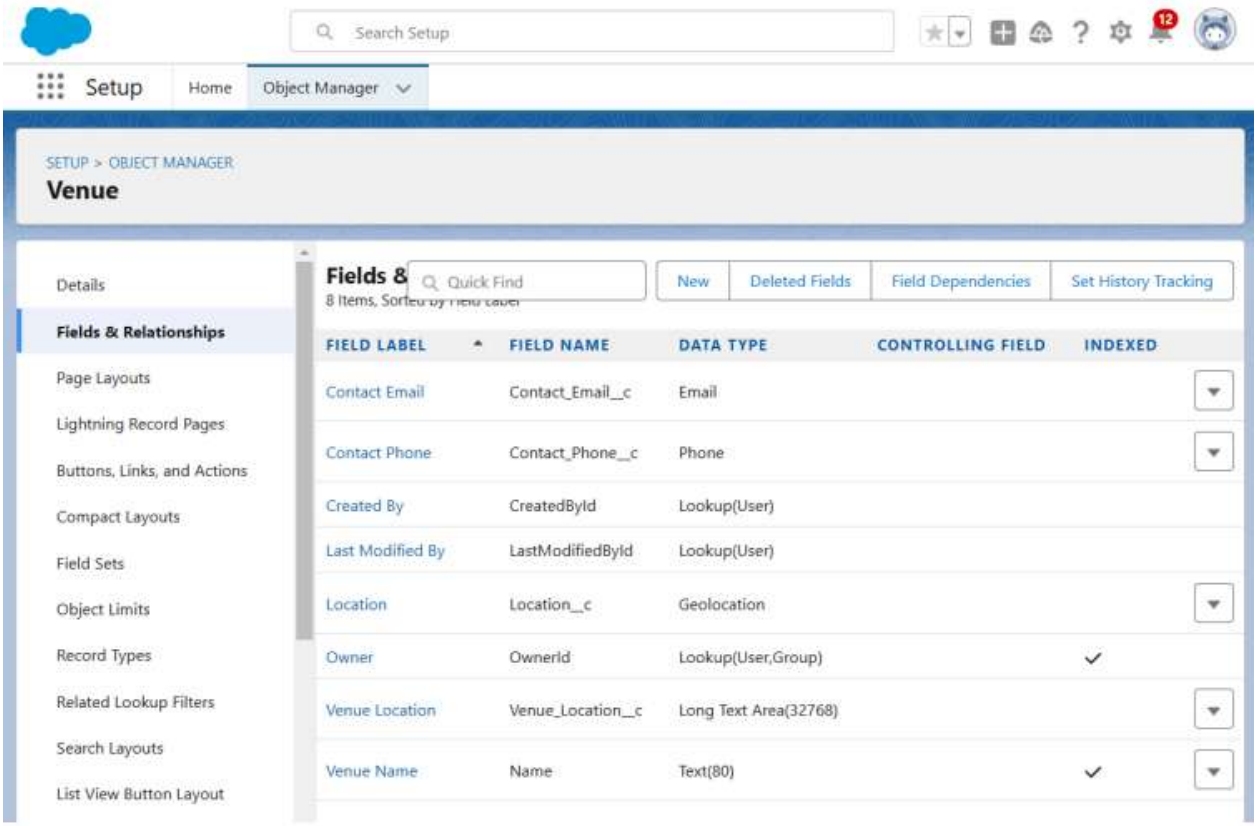
1. **Distance Field**:
    - Navigate to **Setup > Object Manager** and search for "Task."
    - Go to **Fields & Relationships > New** and select **Number** as the data type.
    - Fill in:
        - **Field Label**: Distance
        - **Field Name**: Distance
        - **Length**: 14
        - **Decimal Places**: 4
        - **Required**: Checked
    - Click **Next > Next > Save & New.**

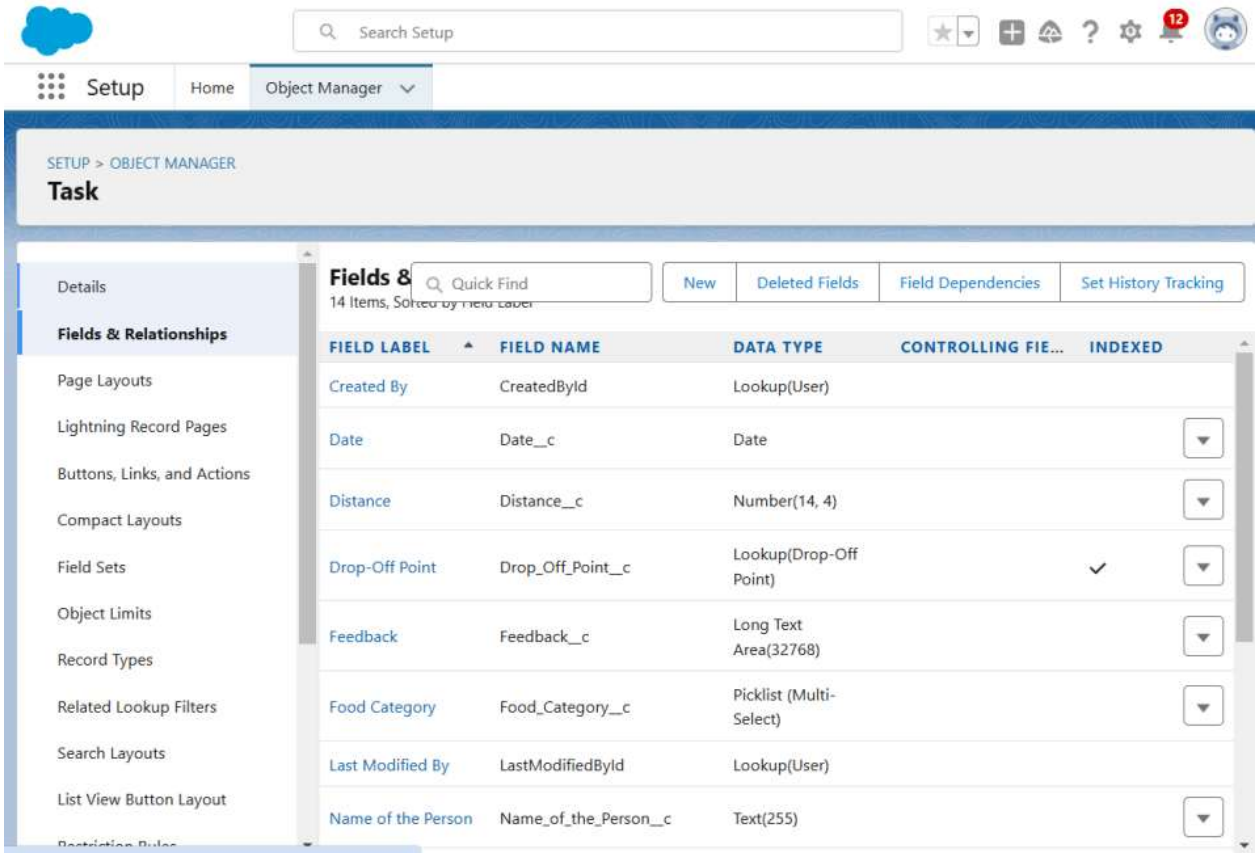
Creation of Fields for the Volunteer Object

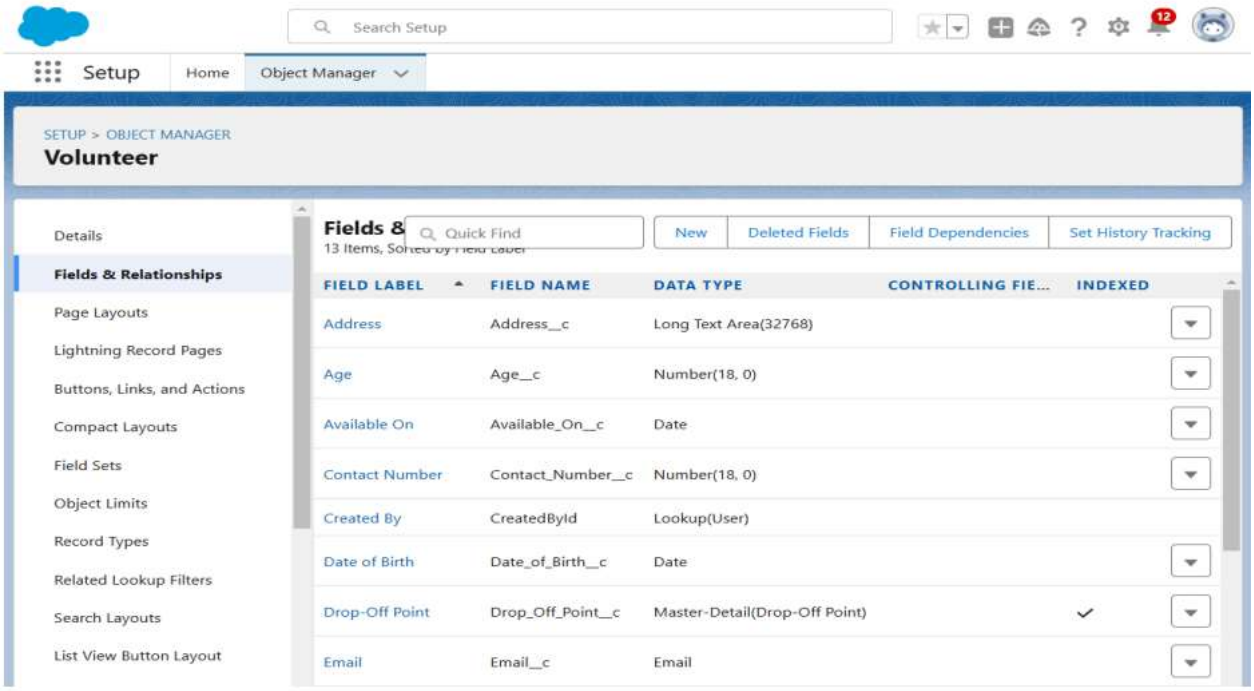
1. **Volunteer ID Field**:
    - Navigate to **Setup > Object Manager** and search for the object "Volunteer."
    - Go to **Fields & Relationships > New** and select **Auto Number** as the data type.
    - Fill in:
        - **Field Label**: Volunteer ID
        - **Field Name**: Auto-generated
        - **Required**: Checked
    - Click **Next > Next > Save & New.**
2. **Gender Field**:
    - Select **Picklist** as the data type.
    - Fill in:
        - **Field Label**: Gender
        - **Field Name**: Gender
        - **Values**:
            - Female
            - Male
    - Click **Next > Next > Save & New.**
3. **Available On Field**:
    - Select **Date** as the data type.
    - Fill in:
        - **Field Label**: Available On
        - **Field Name**: Available_On
        - **Required**: Checked
    - Click **Next > Next > Save & New.**
4. **Age Field**:
    - Select **Number** as the data type.
    - Fill in:
        - **Field Label**: Age
        - **Field Name**: Age
        - **Required**: Checked
    - Click **Next > Next > Save & New.**
5. **Email Field**:
    - Select **Email** as the data type.

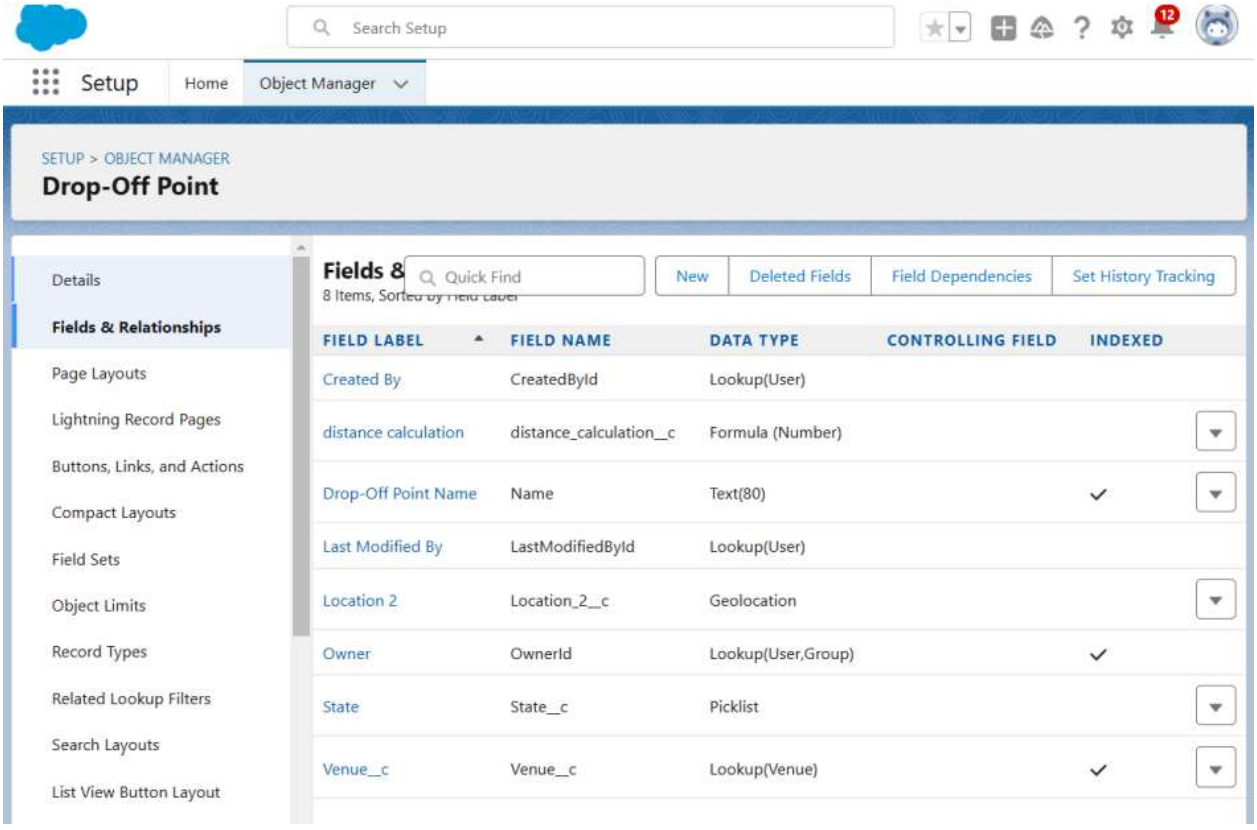
- Fill in:
  - **Field Label**: Email
  - **Field Name**: Email
  - **Required**: Checked
- Click **Next > Next > Save & New.**

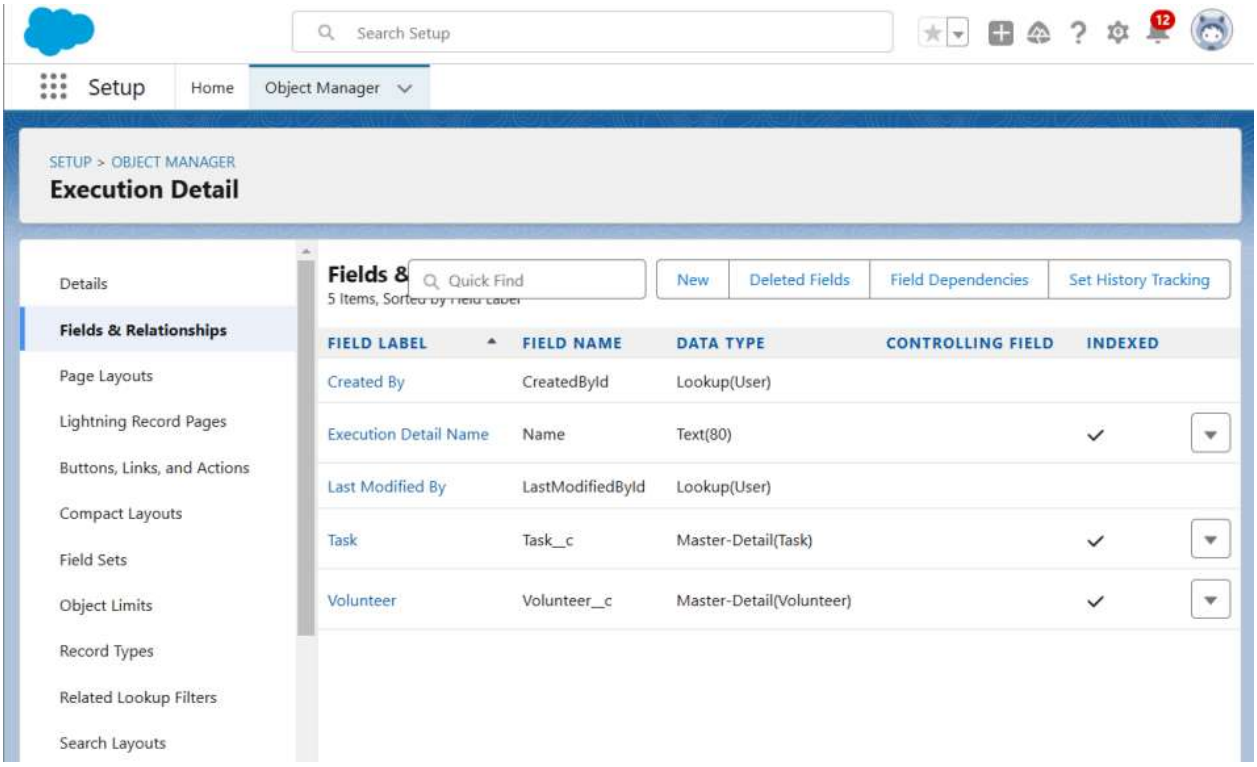
6. **Contact Number Field**:
   - Select **Number** as the data type.
   - Fill in:
     - **Field Label**: Contact Number
     - **Field Name**: Contact_Number
     - **Required**: Checked
   - Click **Next > Next > Save & New.**
7. **Address Field**:
   - Select **Text Area (Long)** as the data type.
   - Fill in:
     - **Field Label**: Address
     - **Field Name**: Address
   - Click **Next > Next > Save & New.**
8. **Date of Birth Field**:
   - Select **Date** as the data type.
   - Fill in:
     - **Field Label**: Date of Birth
     - **Field Name**: Date_of_Birth
   - Click **Next > Next > Save & New.**


Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Venue
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Fields &
Quick Find
New
Deleted Fields
Field Dependencies
Set History Tracking
FIELD LABEL
FIELD NAME
DATA TYPE
CONTROLLING FIELD
INDEXED
Contact Email
Contact_Email__c
Email
Contact Phone
Contact_Phone__c
Phone
Created By
CreatedById
Lookup(User)
Last Modified By
LastModifiedById
Lookup(User)
Location
Location__c
Geolocation
Owner
OwnerId
Lookup(User,Group)
Venue Location
Venue_Location__c
Long Text Area(32768)
Venue Name
Name
Text(80)


Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Task
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Fields &
Quick Find
New
Deleted Fields
Field Dependencies
Set History Tracking
FIELD LABEL
FIELD NAME
DATA TYPE
CONTROLLING FIE...
INDEXED
Created By
CreatedById
Lookup(User)
Date
Date__c
Date
Distance
Distance__c
Number(14, 4)
Drop-Off Point
Drop_Off_Point__c
Lookup(Drop-Off Point)
Feedback
Feedback__c
Long Text Area(32768)
Food Category
Food_Category__c
Picklist (Multi-Select)
Last Modified By
LastModifiedById
Lookup(User)
Name of the Person
Name_of_the_Person__c
Text(255)


Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Volunteer
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Fields &
13 Items, Sorted by Field Label
Quick Find
New
Deleted Fields
Field Dependencies
Set History Tracking
FIELD LABEL
FIELD NAME
DATA TYPE
CONTROLLING FIE...
INDEXED
Address
Address__c
Long Text Area(32768)
Age
Age__c
Number(18, 0)
Available On
Available_On__c
Date
Contact Number
Contact_Number__c
Number(18, 0)
Created By
CreatedById
Lookup(User)
Date of Birth
Date_of_Birth__c
Date
Drop-Off Point
Drop_Off_Point__c
Master-Detail(Drop-Off Point)
Email
Email__c
Email


Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Drop-Off Point
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Fields &
8 Items, Sorted by Field Label
Quick Find
New
Deleted Fields
Field Dependencies
Set History Tracking
FIELD LABEL
FIELD NAME
DATA TYPE
CONTROLLING FIELD
INDEXED
Created By
CreatedById
Lookup(User)
distance calculation
distance_calculation__c
Formula (Number)
Drop-Off Point Name
Name
Text(80)
Last Modified By
LastModifiedById
Lookup(User)
Location 2
Location_2__c
Geolocation
Owner
OwnerId
Lookup(User,Group)
State
State__c
Picklist
Venue__c
Venue__c
Lookup(Venue)

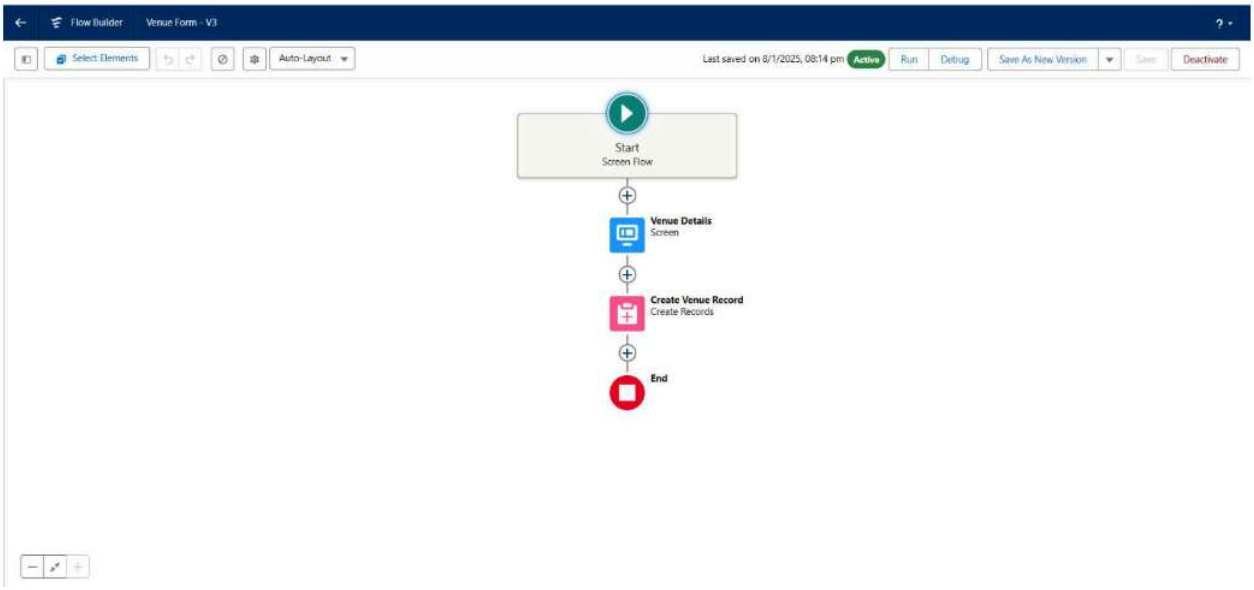


Steps to Create a Flow for Adding a Record to the Venue Object

1. **Create a New Flow**
   - Go to **Setup > Quick Find Box** and type **Flow.**
   - Click **Flow** and select **New Flow.**
   - Choose **Screen Flow** and click **Create.**
2. **Add a Screen Element**
   - Click the **'+'** icon between **Start** and **End** and select **Screen Element.**
   - Set **Screen Properties**:
     - **Label**: Venue Details
     - **API Name**: Venue_Details
3. **Add Components to the Flow Screen**
   - **Text Component**:
     - **Label**: Venue Name
     - **API Name**: Venue_Name
   - **Email Component**:
     - **Label**: Email
     - **API Name**: Contact_Email

- **Phone Component**:
  - **Label**: Phone
  - **API Name**: Contact_Phone
- **Text Component**:
  - **Label**: Venue Location
  - **API Name**: Venue_Location
- **Number Component** (Latitude):
  - **Label**: Latitude
  - **API Name**: Latitude
- **Number Component** (Longitude):
  - **Label**: Longitude
  - **API Name**: Longitude
- Click **Done** when all components are added.

4. **Add a Create Record Element**
   - Click the **'+'** icon between **Venue Details** and **End.**
   - Select **Create Record Element** and configure as follows:
     - **Label**: Create Venue Record
     - **API Name**: Create_Venue_Record
     - **How Many Records to Create**: One
     - **How to Set the Record Fields**: Use separate resources and literal values
     - **Object**: Venue
     - **Set Field Values for Venue**:
       - Field: **Contact_Email__c** | Value: **{!Contact_Email.value}**
       - Field: **Contact_Phone__c** | Value: **{!Contact_Phone.value}**
       - Field: **Name** | Value: **{!Venue_Name}**
       - Field: **Venue_Location__c** | Value: **{!Venue_Location}**
       - Field: **Location__Latitude__s** | Value: **{!Latitude}**
       - Field: **Location__Longitude__s** | Value: **{!Longitude}**
5. **Save the Flow**
   - Click **Save As** and enter the following details:
     - **Flow Label**: Venue Form
     - **Flow API Name**: Venue_Form
   - Click **Save** and activate the flow.



Trigger Creation

1. **Log into Trailhead**
   - Navigate to the gear icon in the top-right corner and click on **Developer Console.**
2. **Create a New Trigger**
   - Go to **File > New > Trigger.**
   - Enter the following details:
     - **Trigger Name**: DropOffTrigger
     - **sObject**: Drop-Off Point
   - Click **Submit.**

Profiles

Definition:
Profiles are groups of settings and permissions that define user access to objects, fields, and various other features in Salesforce. Examples include **System**

Administrator, Developer, Sales Representative.

Steps to Create a Profile

1. Go to **Setup > Quick Find** and type **Profiles.**
2. Click on **Profiles** and locate **Standard Platform User.**
3. Click **Clone** beside the Standard Platform User.
4. Under **Clone Profile**, set the following:
    - **Profile Name**: NGOs Profile
    - Click **Save.**

User Creation

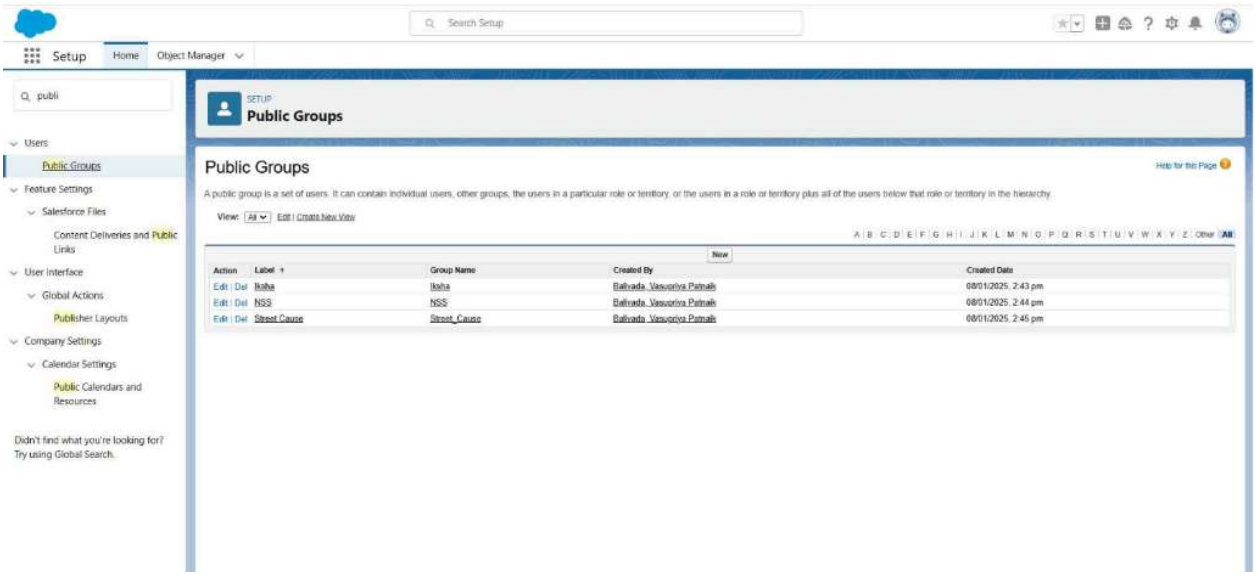
Steps to Create User 1

1. Go to **Setup > Quick Find** and type **Users.**
2. Click on **Users > New User.**
3. Fill in the following details:
    - **First Name**: Iksha Foundation
    - **Last Name**: Iksha_Foundation
    - **Alias**: iiksh
    - **Email**: Your email address
    - **Username**: ikshafoundation@sb.com (must be unique)
    - **Nickname**: Auto-populated
    - **User License**: Salesforce Platform
    - **Profile**: NGOs Profile
    - **Active**: Check the box
4. Click **Save.**

Repeat for other NGOs, using unique First Name, Last Name, and Username.

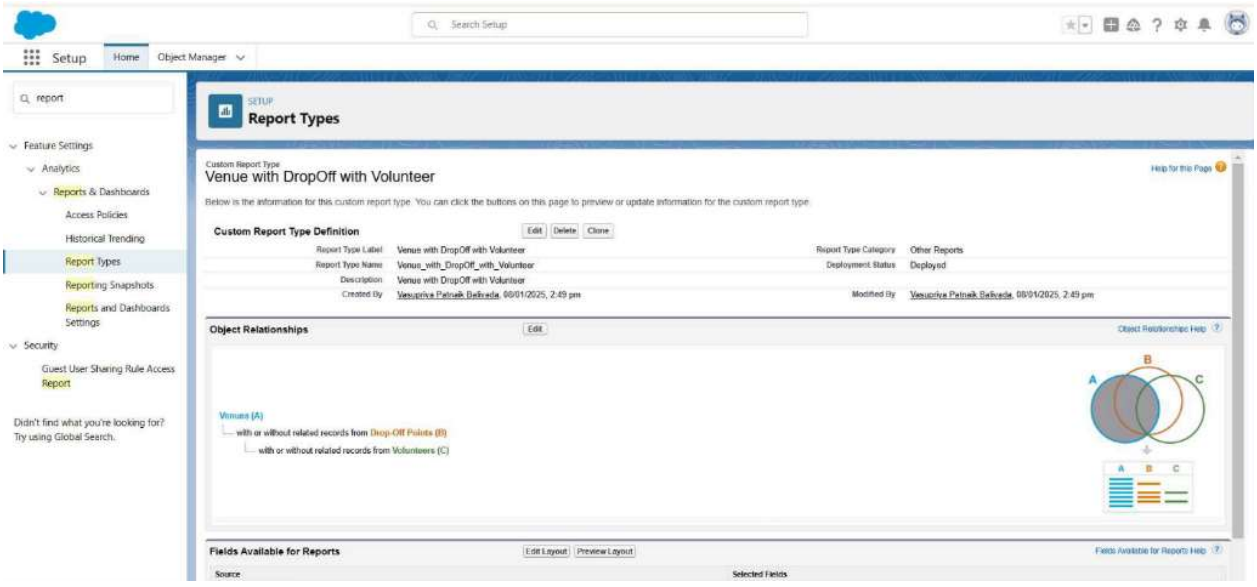
Public Group Creation

1. Go to **Setup > Quick Find** and type **Public Groups.**
2. Click on **Public Groups > New.**
3. Under **Group Information**, set the following:
    - **Label**: Iksha
    - **Group Name**: Iksha
    - **Grant Access Using Hierarchies**: Check the box
4. In **Search**, select **Users** and add the following to **Selected Members**:
    - Iksha Foundation
    - System Administrator
5. Repeat for the remaining NGOs.



Report Type Creation

1. Go to **Setup > Quick Find** and type **Report Types.**
2. Click on **Report Types > Continue > New Custom Report Type.**
3. Under **Define the Custom Report Type**, set the following:
    - **Primary Object**: Venues
    - **Report Type Label**: Venue with DropOff with Volunteer
    - **Report Type Name**: Venue_with_DropOff_with_Volunteer
    - **Description**: Venue with DropOff with Volunteer
    - **Store in Category**: Other Reports
    - **Deployment Status**: Deployed
4. Click **Next.**
5. Relate another object:
    - Select **Drop-Off Points** and set "A records may or may not have related B records."
    - Select **Volunteers** and repeat the relation.
6. Click **Save.**



Report on Venue with Drop-Off and Volunteer

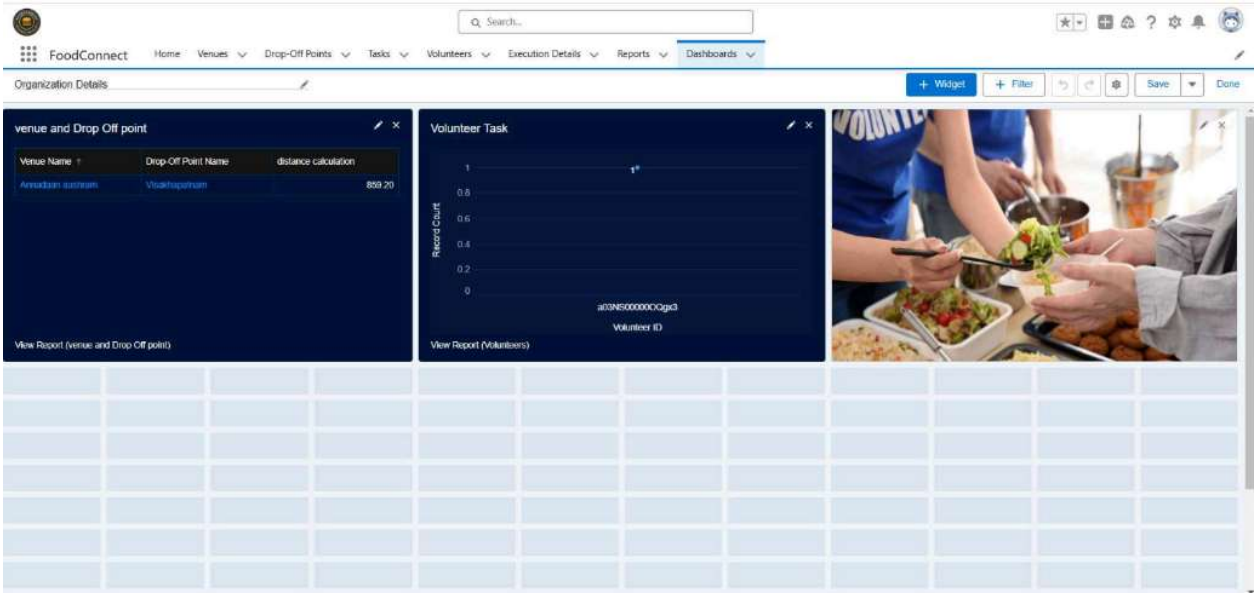
1. Open the **FoodConnect App** and go to the **Reports Tab.**
2. Click **New Folder** and set:
    - **Folder Label**: Custom Reports
    - **Folder Unique Name**: CustomReports
3. Open the **Custom Reports** folder and click **New Report.**
4. Select **Report Type**: Venue with DropOff with Volunteer.
5. Click **Start Report** and configure:
    - **Group Rows**: Volunteer Name
    - **Columns**: Venue Name, Drop-Off Point Name, Distance
6. Click **Save & Run** and set the following:
    - **Report Name**: Venue and Drop-Off Point
    - **Report Unique Name**: Auto-populated
    - **Select Folder**: Custom Reports
7. Click **Save.**

Report on Volunteers with Execution Details and Tasks

1. Open the **FoodConnect App** and go to the **Reports Tab.**
2. Open the **Custom Reports** folder and click **New Report.**
3. Select **Report Type**: Volunteers with Execution Details and Tasks.
4. Click **Start Report** and configure:
    - **Group Rows**: Volunteer ID
    - **Columns**: Volunteer Name, Task Name, Execution Detail Name, Owner Name, Task Date, Task Rating
5. Click **Save & Run** and set the following:
    - **Report Name**: Volunteer Task
    - **Report Unique Name**: Auto-populated
    - **Select Folder**: Custom Reports
6. Click **Save.**

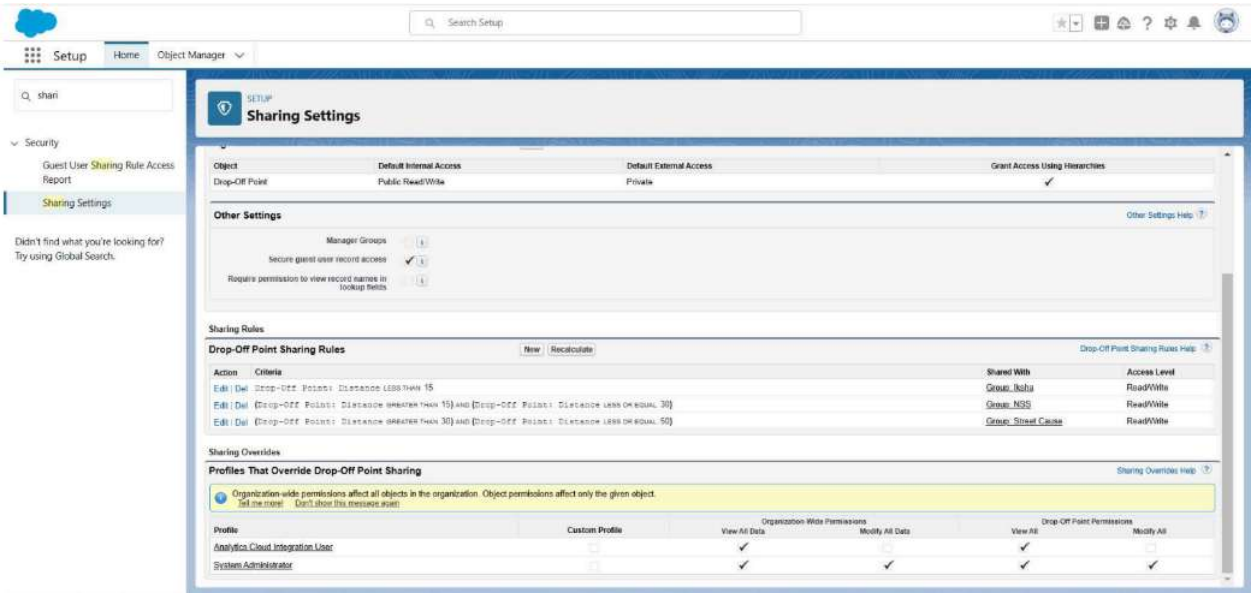
Adding a Picture to the Dashboard (Optional):

1. Click on Widget and select Image. Click on Browse Files and select the image.
2. Click Save As:
    - Name: Task Execution Details
    - Folder: Custom Dashboards
3. Save.



Creation of Sharing Rules:

1. Go to Setup >> Sharing Settings.
2. Scroll to Drop-Off Point Sharing Rules. Click New.
    - Label: Rule 1
    - Rule Name: Rule_1
    - Type: Based on criteria
    - Field: Distance < 15
    - Share with: Iksha (Public Group)
    - Save.
3. Repeat for Rule 2 (Distance > 15, <= 30) with NSS, and Rule 3 (Distance > 30, <= 50) with Street Cause.



Creation of Home Page:

1. Go to Setup >> Lightning App Builder.
2. Select Home Page and name it “HOME Page.”
3. Drag Flow (Venue Flow) and Dashboard components to the page.
4. Save and activate the page for the FoodConnect app.

Conclusion:

The project leveraged Salesforce to streamline surplus food donations, enhancing coordination with volunteers and addressing food insecurity effectively.

HERE IS COde