**To Supply Leftover Food to Poor**

**Project Overview:**

The "To Supply Leftover Food to Poor" project in a CRM (Customer Relationship Management) system focuses on coordinating efforts between restaurants, event organizers, and volunteers to collect and distribute leftover food to underprivileged communities. The CRM helps track food donors, manage pickup and delivery schedules, monitor recipient feedback, and ensure food safety and compliance. This initiative aims to reduce food waste while addressing hunger, using CRM tools to streamline communication, logistics, and reporting for greater social impact.

**Objectives:**

The objective of the "To Supply Leftover Food to Poor" project in a CRM system is to efficiently manage and streamline the collection, tracking, and distribution of surplus food from donors (such as restaurants, hotels, and event organizers) to individuals and communities in need. By leveraging CRM tools, the project aims to ensure timely coordination among stakeholders, maintain records of food donations and deliveries, reduce food waste, and ultimately contribute to hunger relief in a structured and accountable manner.

**Requirement Analysis & Planning:**

**Business Requirements:**

In the "To Supply Leftover Food to Poor" project using a CRM, user needs include efficient coordination between food donors, volunteers, and distribution teams, real-time tracking of food availability and delivery, and clear communication among all parties. The project addresses problems such as food waste, lack of a centralized system to manage donations, delays in food distribution, and difficulties in reaching those in need. By using a CRM, the project solves these issues by organizing data, automating tasks, and enabling timely and transparent operations to ensure surplus food is safely and quickly delivered to the poor.

### ****Project Scope:****

The "To Supply Leftover Food to Poor" project aims to create a system, managed through a CRM platform, to collect surplus food from donors (e.g., restaurants, hotels, event venues), coordinate its pickup, and distribute it to underprivileged communities. The scope includes identifying food donors, registering beneficiaries, managing logistics (such as pickup and delivery scheduling), tracking food inventory, and ensuring compliance with safety and hygiene standards. It also includes reporting, volunteer coordination, and real-time communication between all stakeholders.

### ****Project Objectives:****

1. **Reduce Food Waste:** Minimize the amount of edible food discarded by repurposing it for distribution to those in need.
2. **Alleviate Hunger:** Provide timely food assistance to poor and homeless individuals through organized distribution.
3. **Streamline Operations:** Use CRM tools to manage donor data, schedule logistics, and monitor food movements efficiently.
4. **Enhance Communication:** Facilitate clear and real-time communication between donors, volunteers, and coordinators.
5. **Ensure Accountability and Safety:** Maintain proper records and compliance with food safety regulations.
6. **Promote Social Responsibility:** Encourage more organizations to participate in food donation by demonstrating impact through CRM-generated reports.

**Data Model:**

The **data model** for the *To Supply Leftover Food to Poor* project defines the structure and relationships of the key data entities managed in the CRM system.

1. **Entities:**
   * **Donor**: Stores details of food donors (restaurants, hotels, etc.) – name, contact, location, food type, quantity.
   * **Volunteer/Staff**: Details of individuals handling food pickup and distribution – name, contact info, assigned area.
   * **Recipient/Beneficiary**: Information about individuals or organizations receiving the food – name, location, number of people served.
   * **Food Donation**: Tracks each donation – donor ID, food type, quantity, time of donation, expiry.
   * **Pickup/Delivery Schedule**: Contains logistics details – pickup time, assigned volunteer, delivery location, status.
   * **Feedback/Report**: Logs recipient or volunteer feedback on food quality, delivery issues, etc.
2. **Relationships:**
   * One **Donor** can have multiple **Food Donations**.
   * One **Food Donation** links to one or more **Pickup/Delivery Schedules**.
   * One **Volunteer** can manage multiple **Pickups/Deliveries**.
   * One **Recipient** can receive multiple deliveries.

**Security Model:**

The **security model** ensures that sensitive data is protected and access is granted based on roles and responsibilities:

1. **Role-Based Access Control (RBAC):**
   * **Admin**: Full access to all data, reports, and system settings.
   * **Donors**: Can view and manage their own donation records.
   * **Volunteers**: Access to assigned pickups/deliveries and route details only.
   * **Recipients/NGOs**: Limited view access to delivery schedules and feedback tools.
2. **Data Security Features:**
   * **Field-level security**: Sensitive information (e.g., contact details) is hidden based on roles.
   * **Audit trails**: Logs of who accessed or edited data, to ensure accountability.
   * **Encryption**: Secure storage and transfer of sensitive data such as personal info or location.
   * **Authentication**: Secure login with password policies and possibly two-factor authentication.

This structured **data model and security model** ensures efficient management and safe, responsible sharing of information across all participants in the project.

**Salesforce Development - Backend & Configurations:**

**Objects:**

1. Venue:

Label name >> Venue

Plural label name >> Venues

Record Name >> Venue Name

 Data Type >> Text

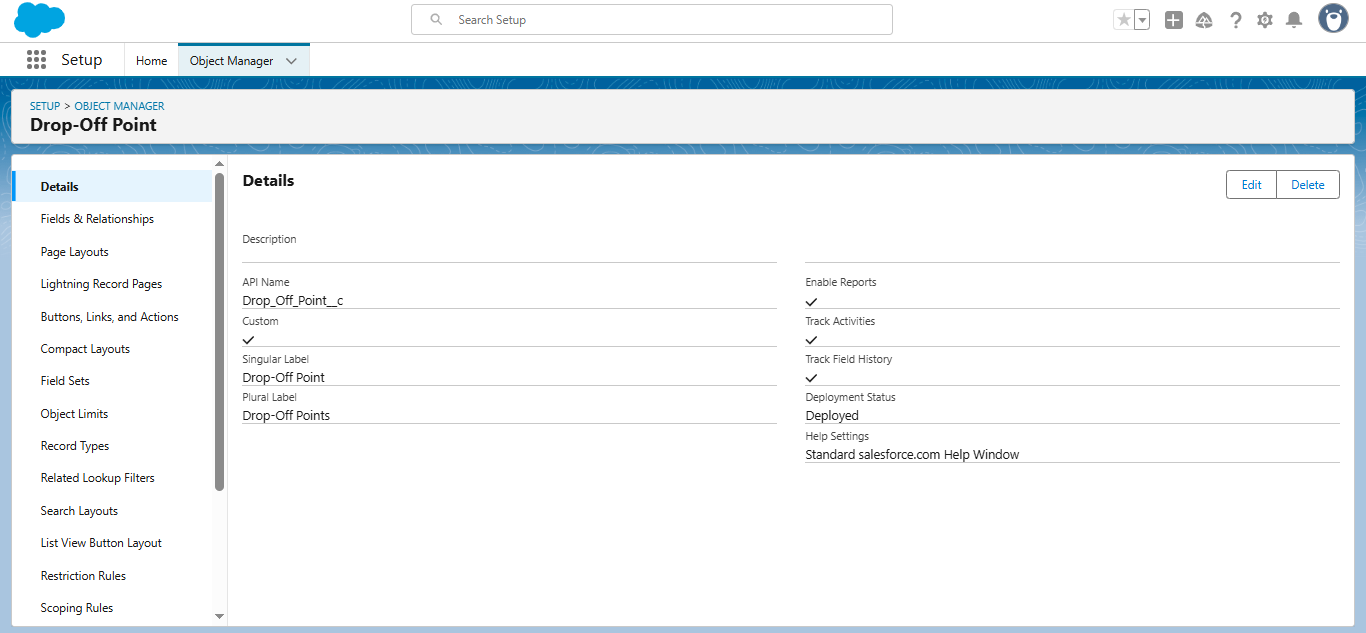
1. Drop-Off Point

label name >> Drop-Off Point

Plural label name>> Drop-Off Points

Record Name >> Drop-Off point Name

Data Type >> Text



1. Task:

label name>> Task

Plural label name>> Tasks

Record Name >> Task Name

Data Type >> Text



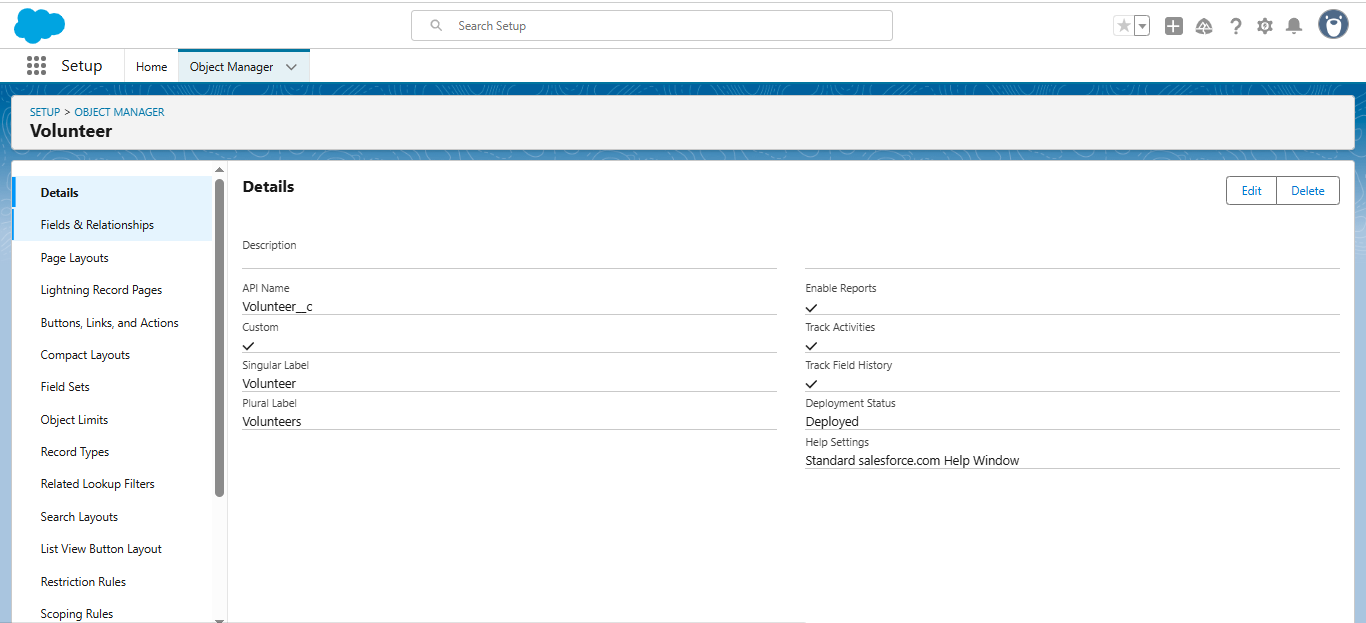
1. Volunteer:

label name>> Volunteer

Plural label name>> Volunteers

Record Name >> Volunteer Name

Data Type >> Text



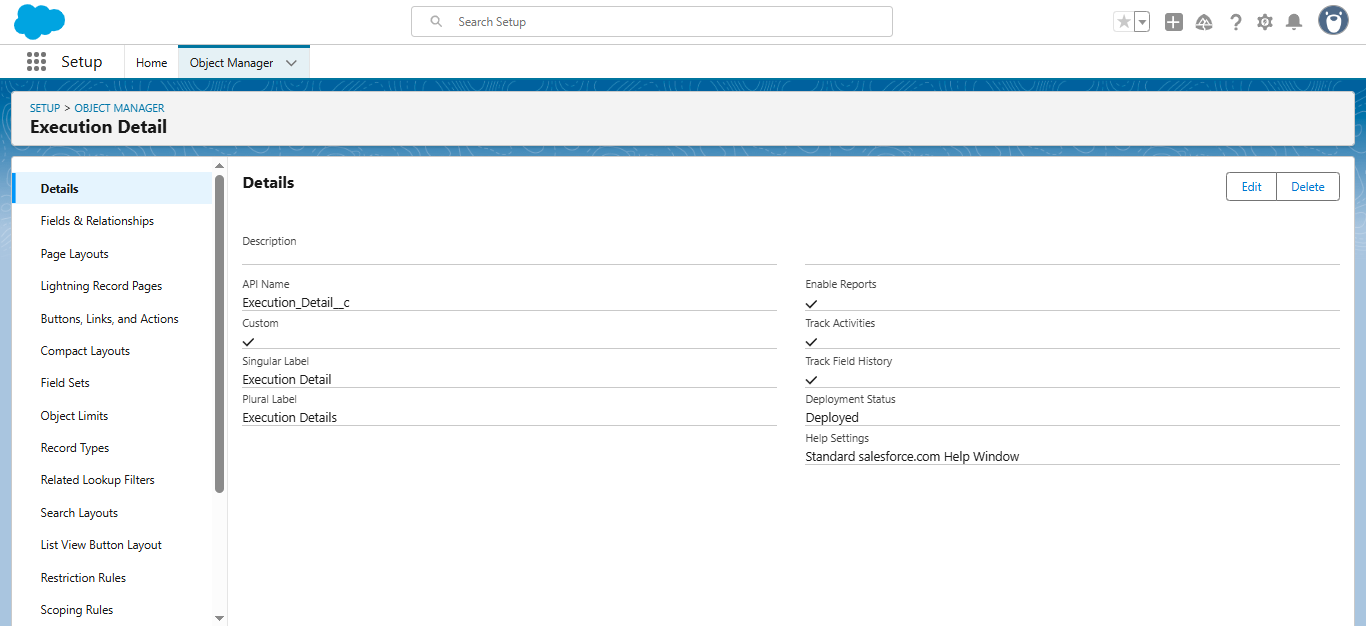
1. Execution Details:

label name >> Execution Detail

Plural label name >> Execution Details

Record Name >> Execution Detail Name

Data Type >> Text



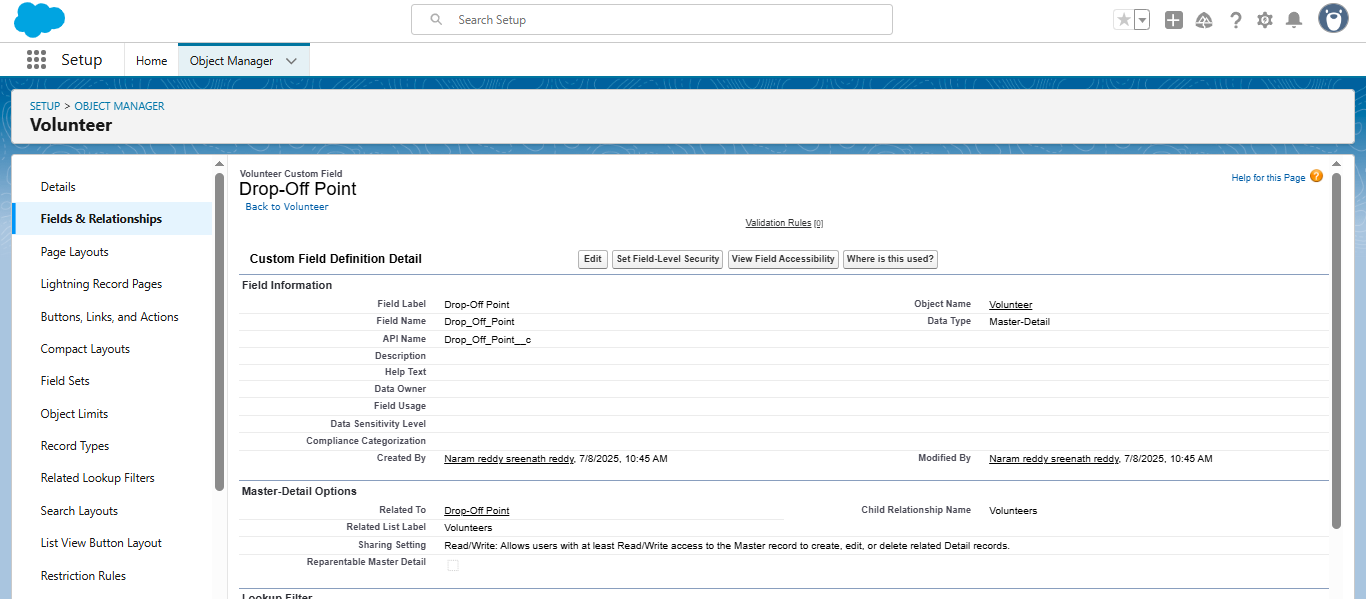
**Creation of Relationship fields in objects:**

Creation of Lookup Relationship Field on Volunteer Object :

1. Select Master Detail relationship

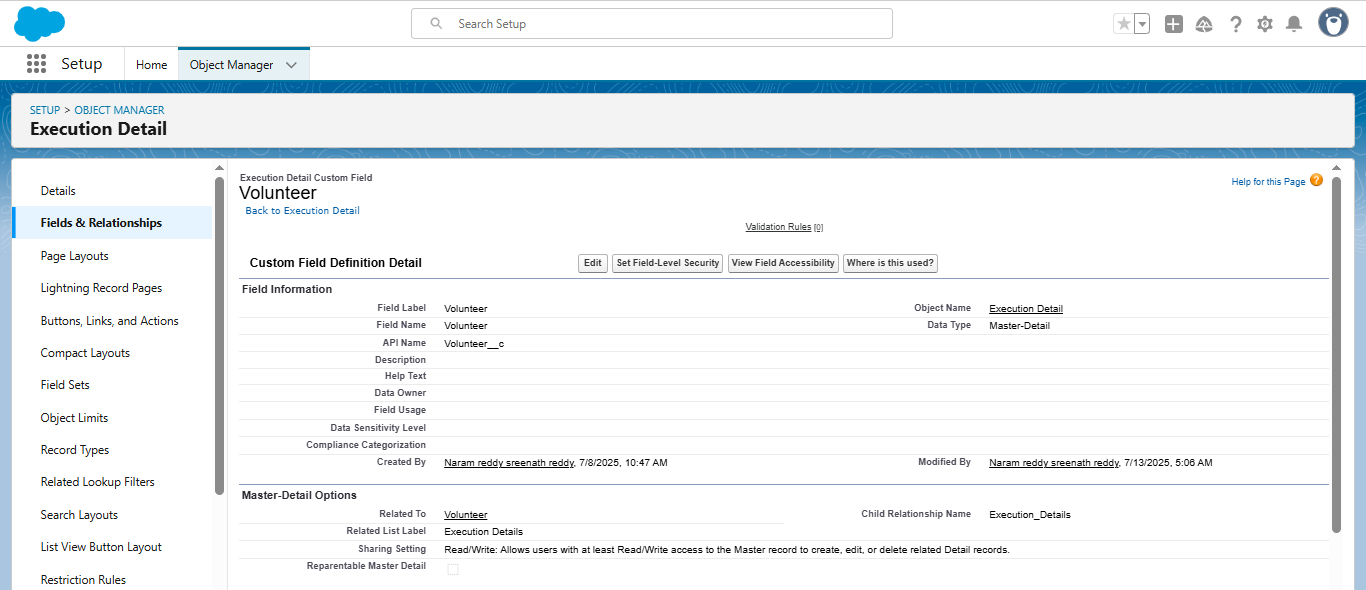
2. Field Name : Drop\_Off\_point

3. Field label : Auto generated

****

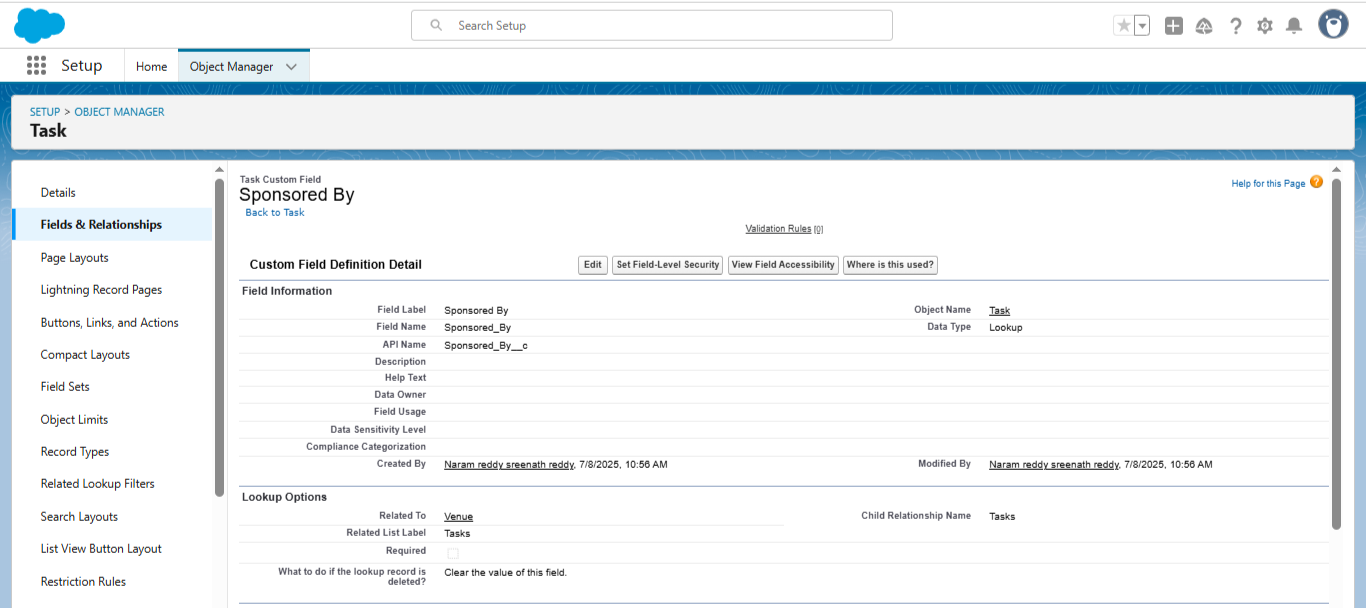
Creation of Master Detail Relationship Field on Execution Details Object :

1. Select Master Detail relationship
2. Field Name : Volunteer
3. Field label : Auto generated



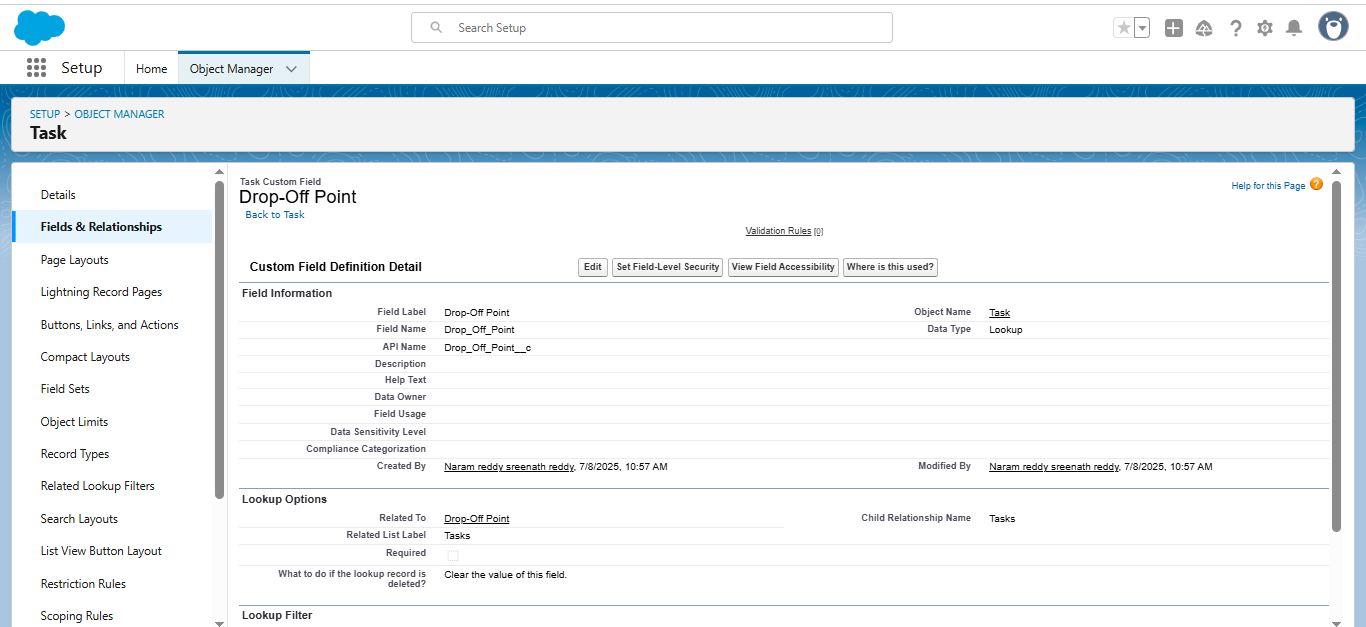
Creation of Lookup Relationship Field on Task Object :

1. Select Lookup relationship
2. Field Name : Sponsored By
3. Field label : Auto generated



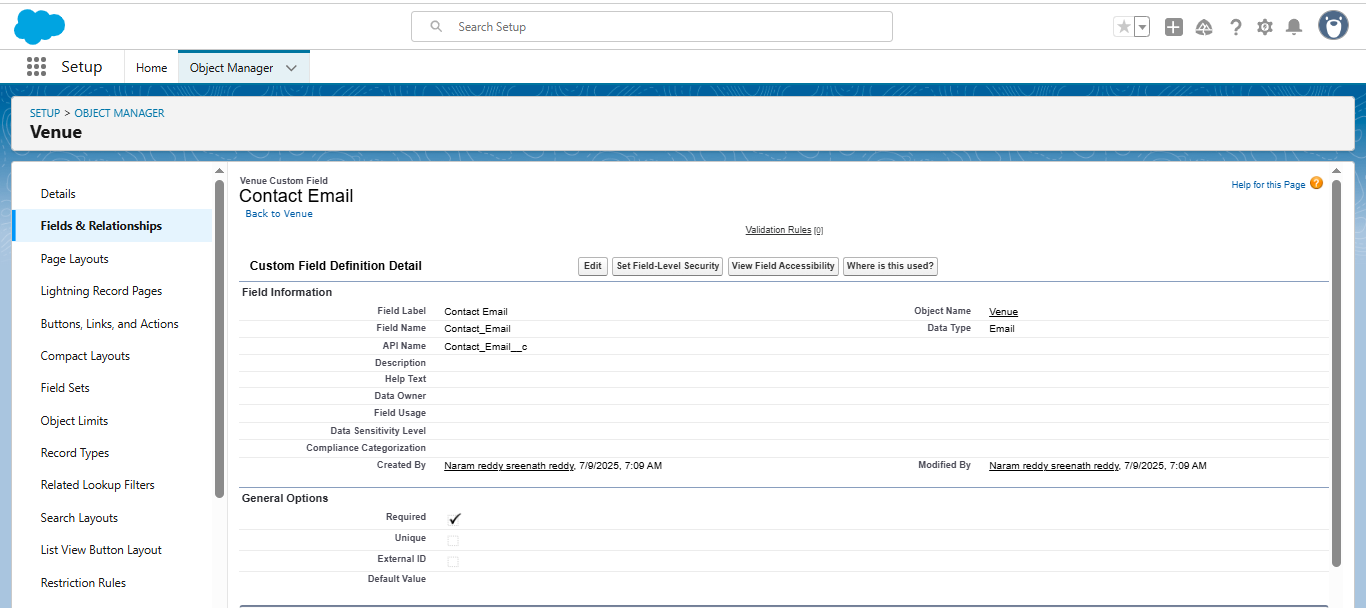
Creation of Lookup Relationship Field on Task Object :

1. Select Lookup relationship
2. Field Name : Drop-Off point
3. Field label : Auto generated



### Creation of fields for the Venue object:

1. Field Label : Contact Email
2. Field Name : Contact Email
3. Click on required check box



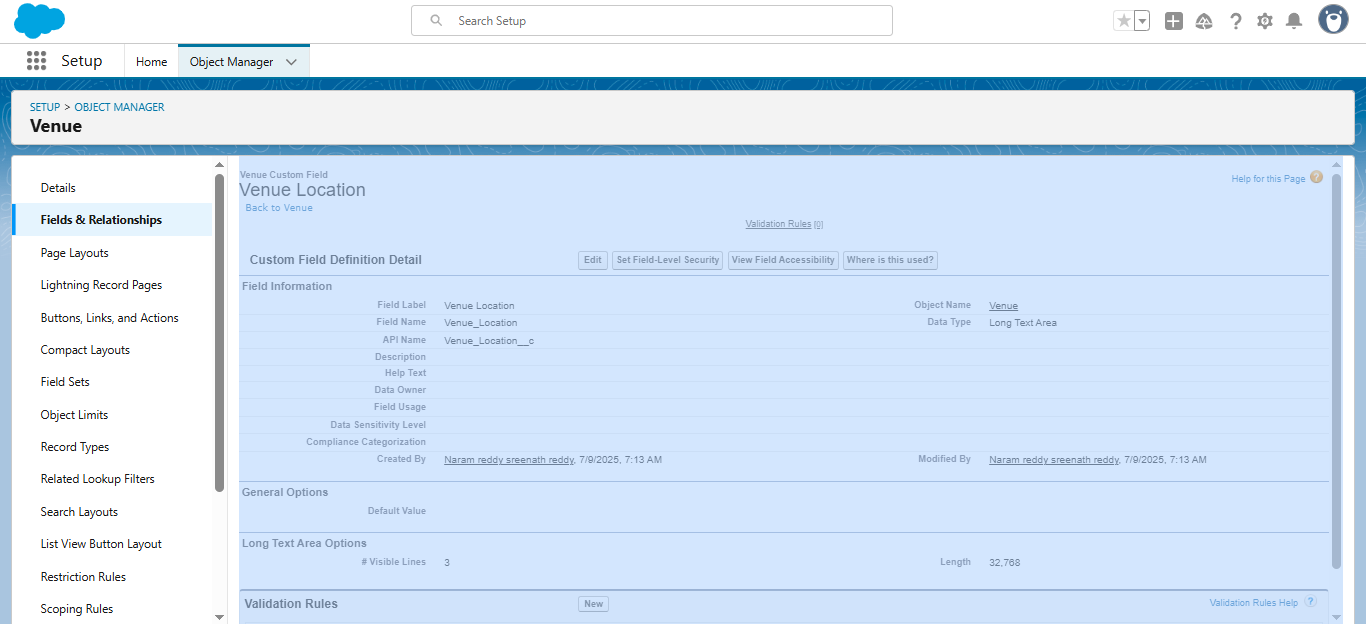
1. Field Label : Contact Phone
2. Field Name : Contact Phone
3. Click on required check box

### 

1. Field Label : Location
2. Decimal Places : 4
3. Field Name : Location
4. Description : Enter the Geolocation of your Venue

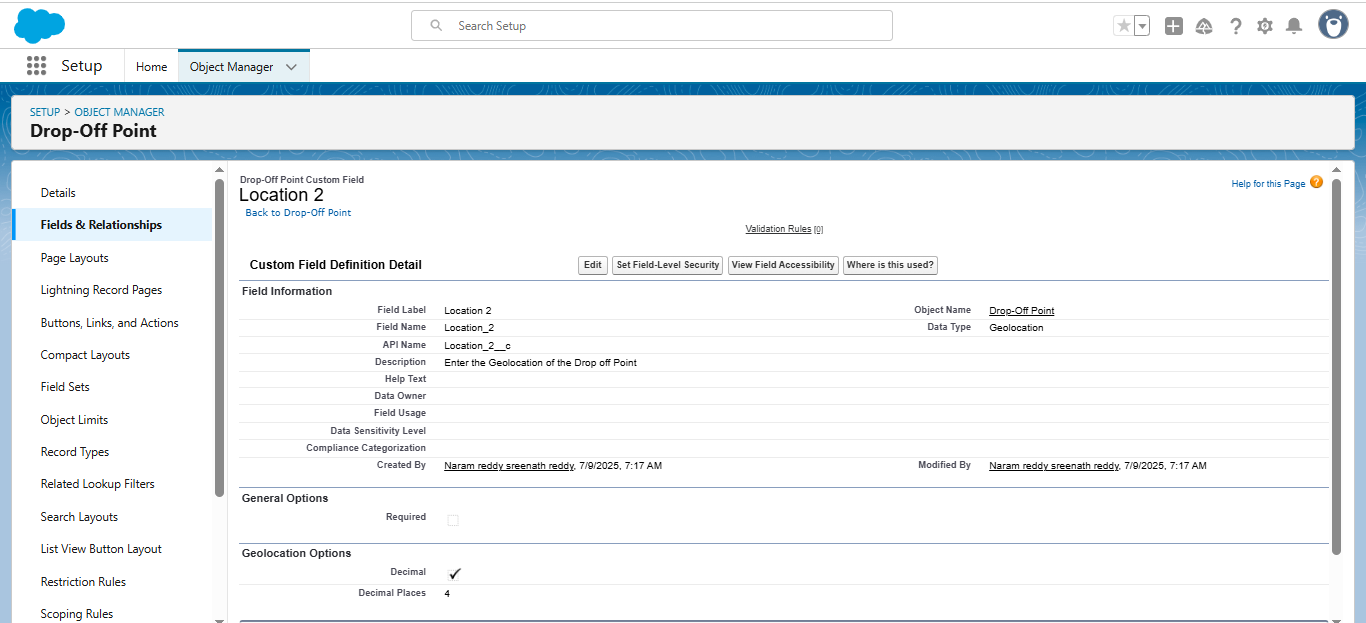
### 

1. Field Label : Venue Location
2. Field Name : Venue\_Location

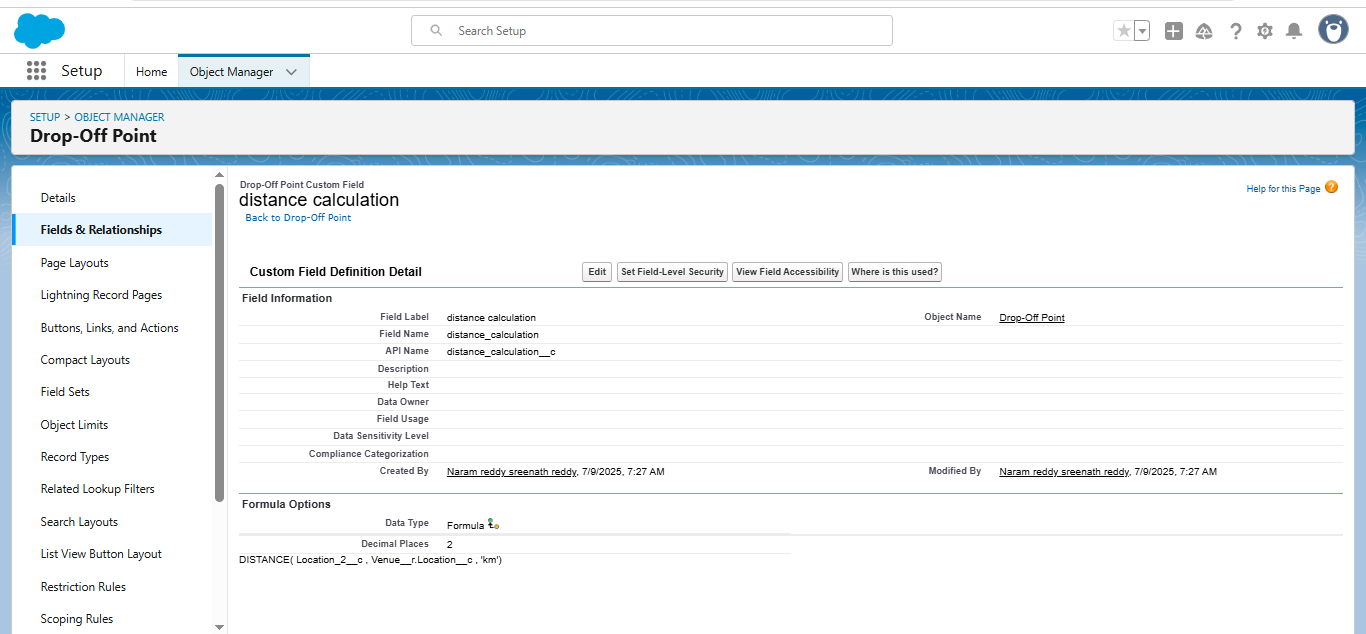
****

### Creation of fields for the Drop-Off point object:

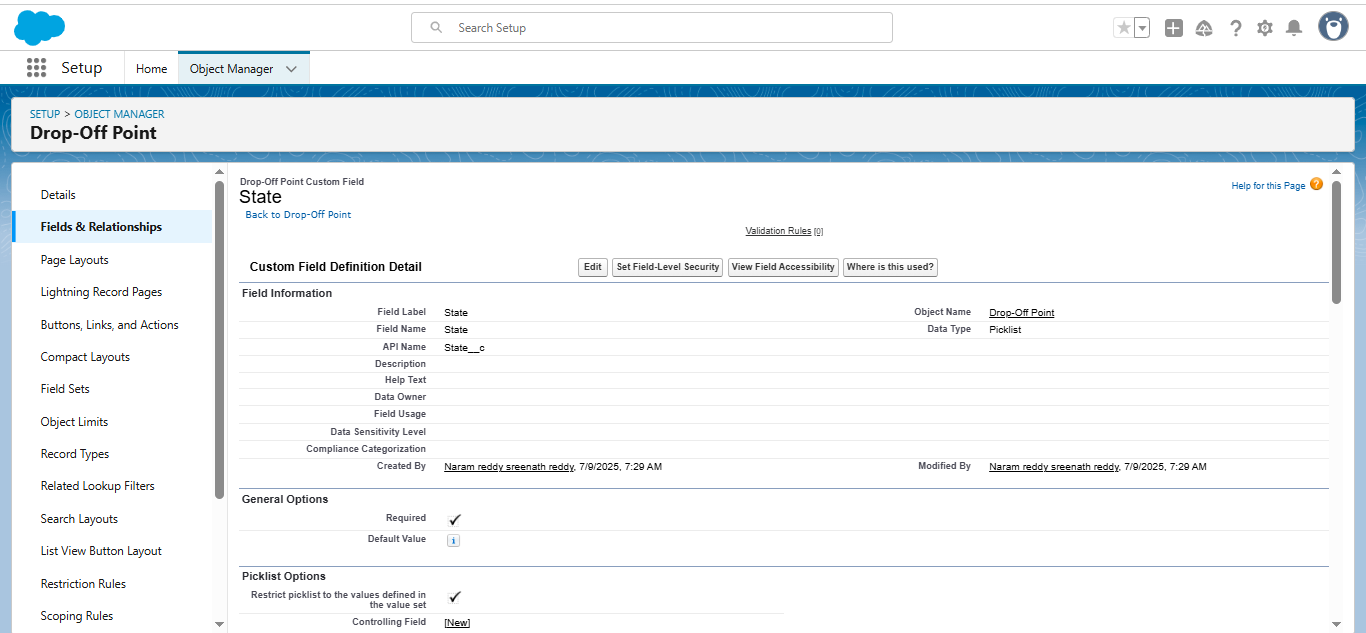
1. Field Label : Location 2
2. Field Name : gets auto generated
3. Description : Enter the Geolocation of the Drop off Point
4. Geolocation Options : select Decimal
5. Decimal Places : 4



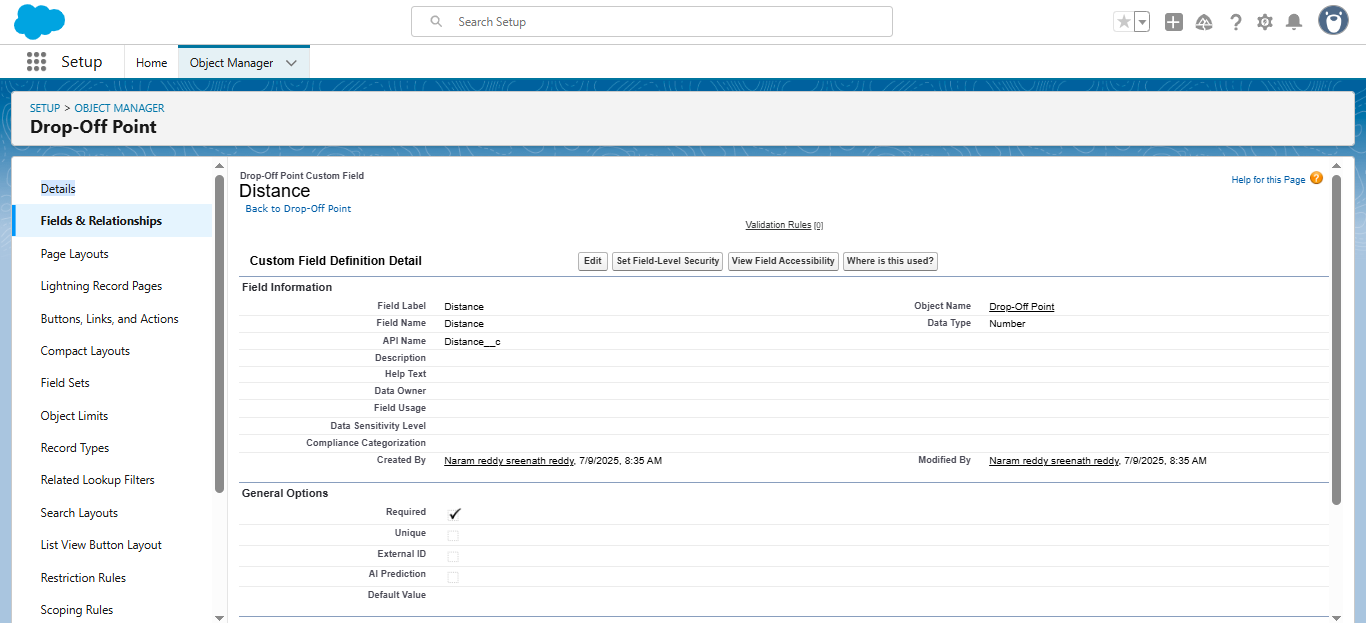
1. Field Label : distance calculation
2. Field Name : distance\_calculation
3. Formula Return Type : Number
4. Formula Options : DISTANCE( Location\_2\_\_c ,  Venue\_\_r.Location\_\_c , 'km')



1. Field Label : State
2. Field Name : State
3. Enter values, with each value separated by a new line

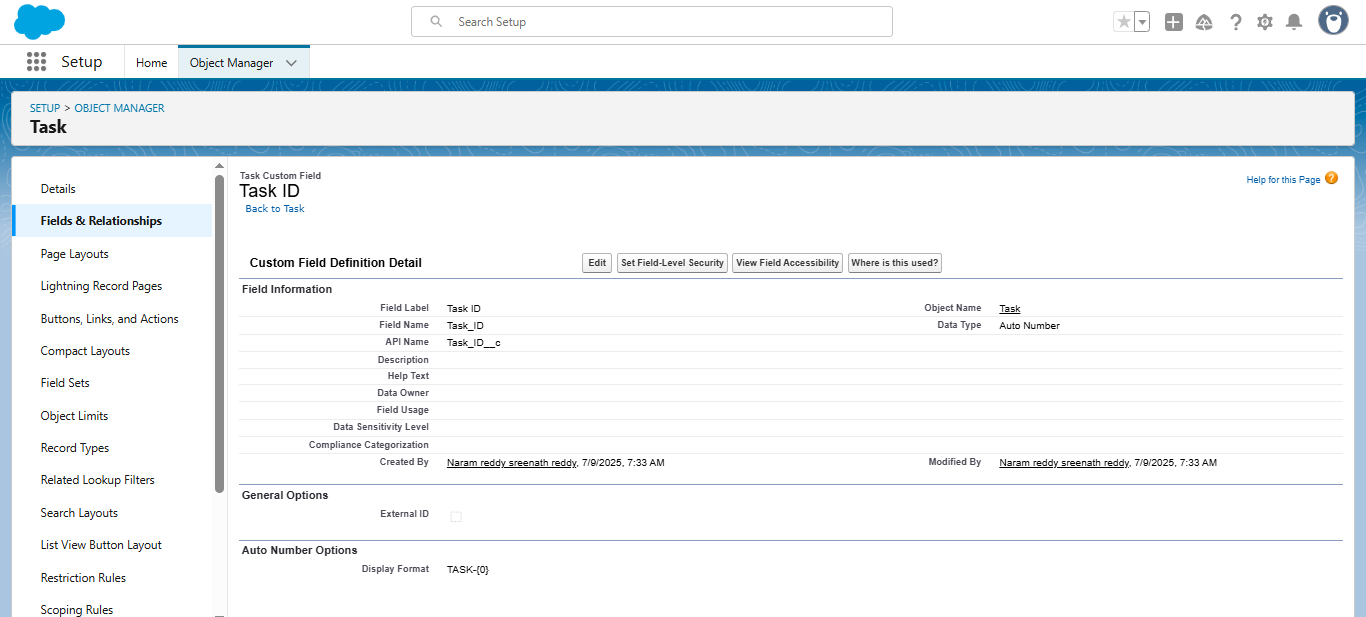


1. Field Label : Distance
2. Field Name : Distance
3. Length : 14
4. Decimal Places : 4
5. Click on required check box

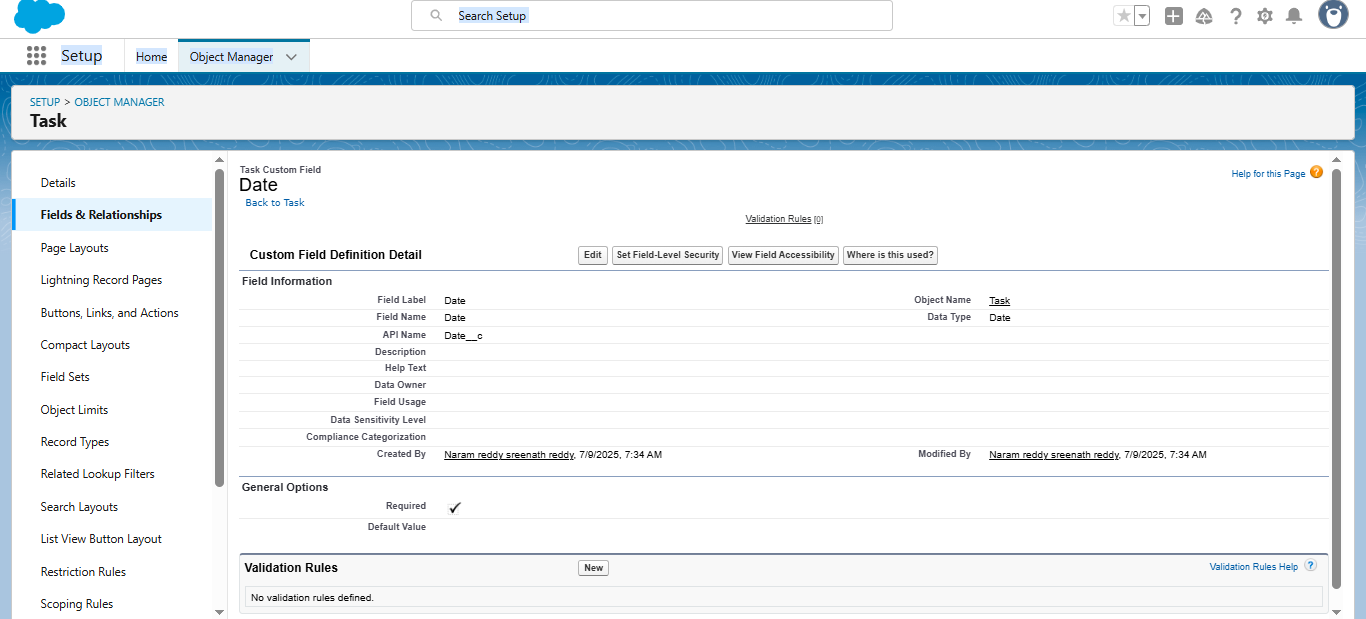


### Creation of fields for the Task object:

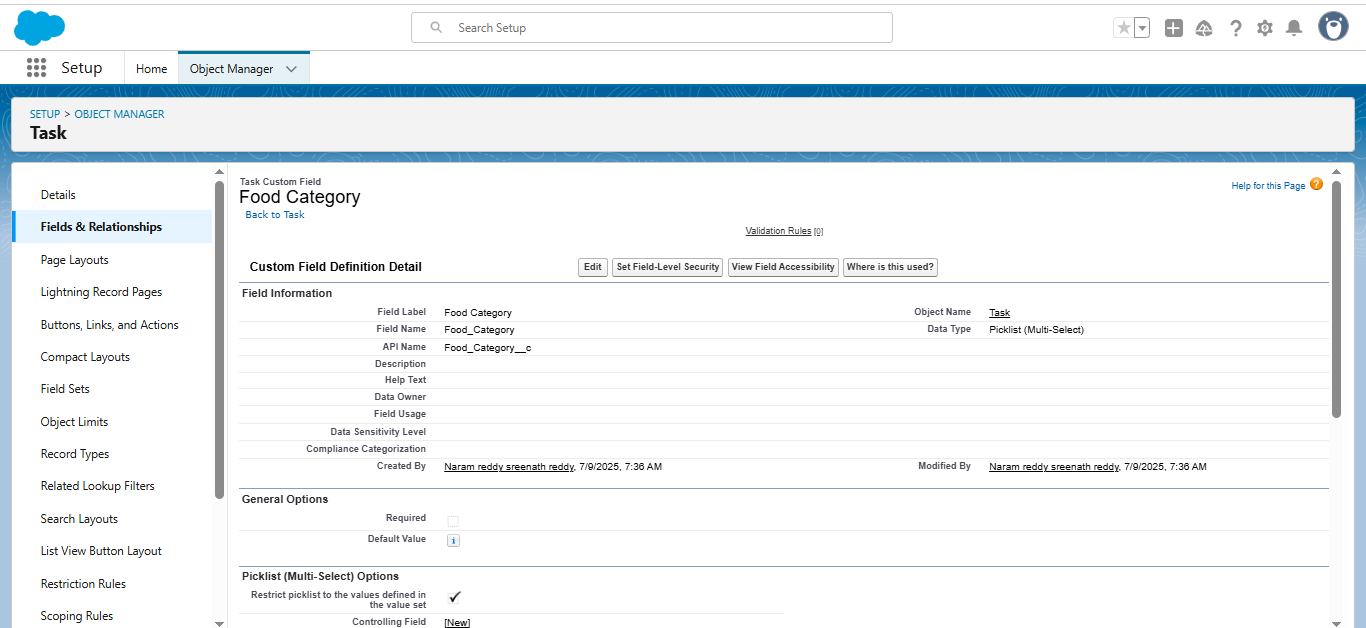
1. Field Label : Task ID
2. Display Format : TASK-{0}
3. Starting Number : 1
4. Field Name : gets auto generated
5. Click on required check box.



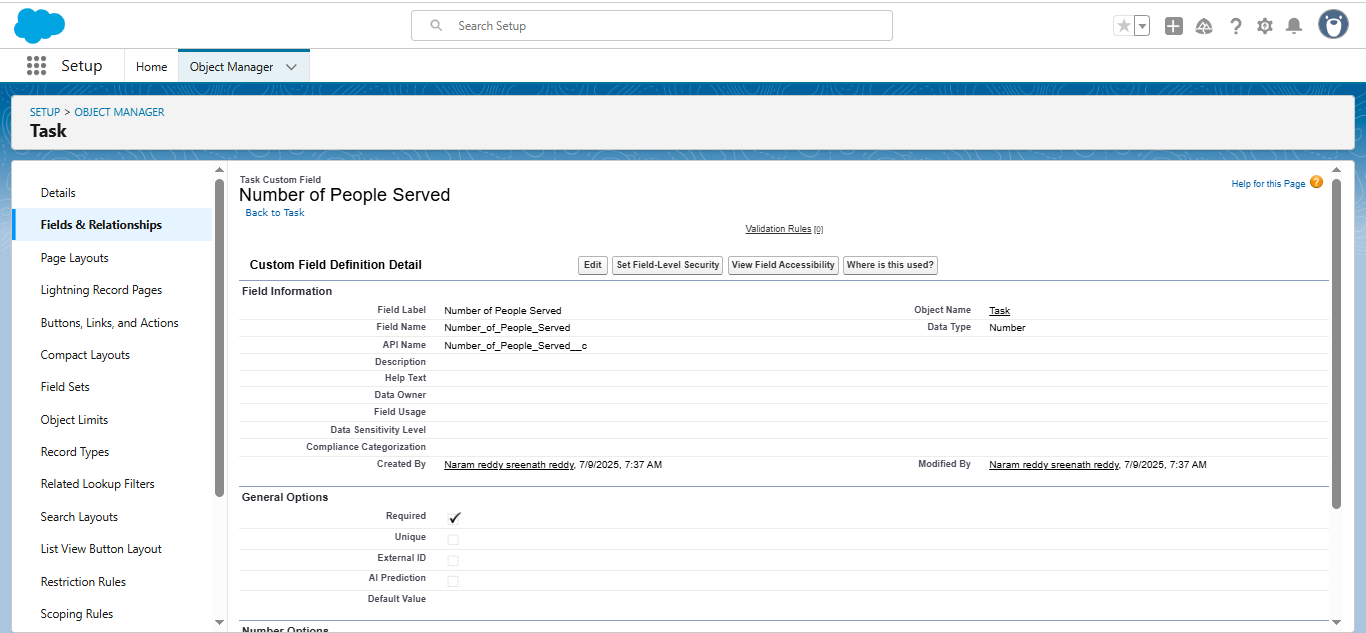
1. Field Label : Date
2. Field Name : Date
3. Click on required check box
4. Click on Next >> Next >> Save and new



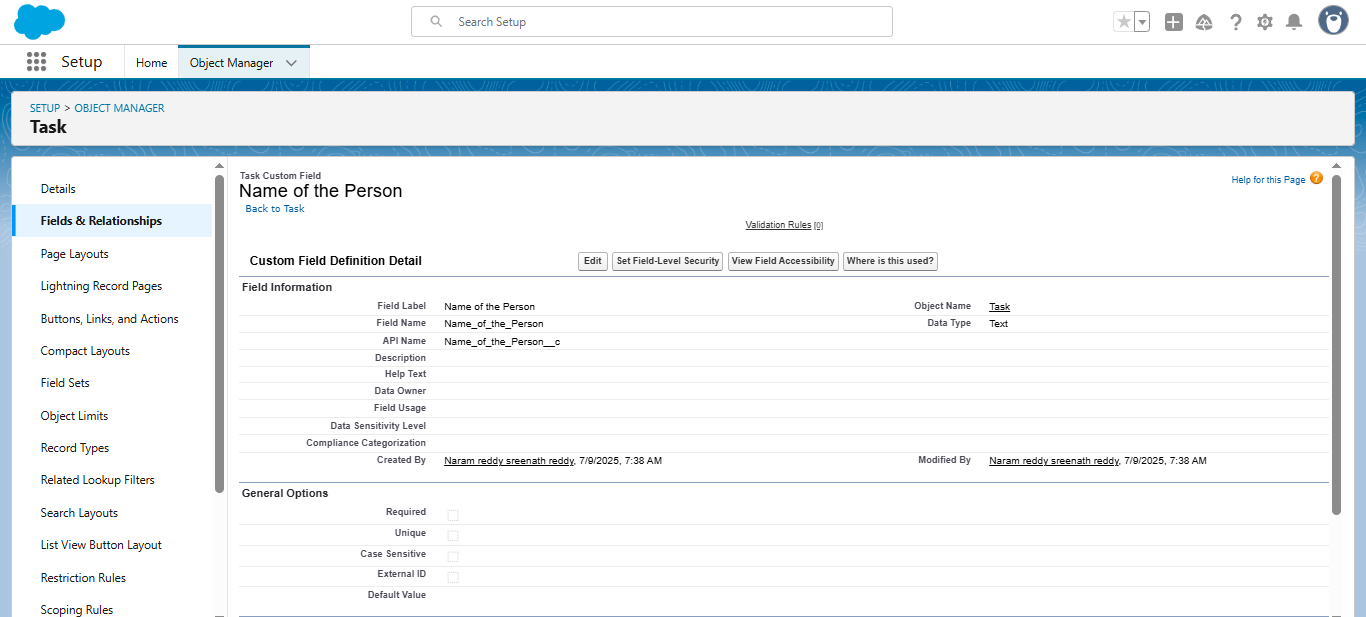
1. Field Label : Food Category
2. Field Name : Food Category
3. Enter values, with each value separated by a new line :
   1. Veg
   2. Non-Veg
   3. Salad
   4. Snack
4. Click on required check box



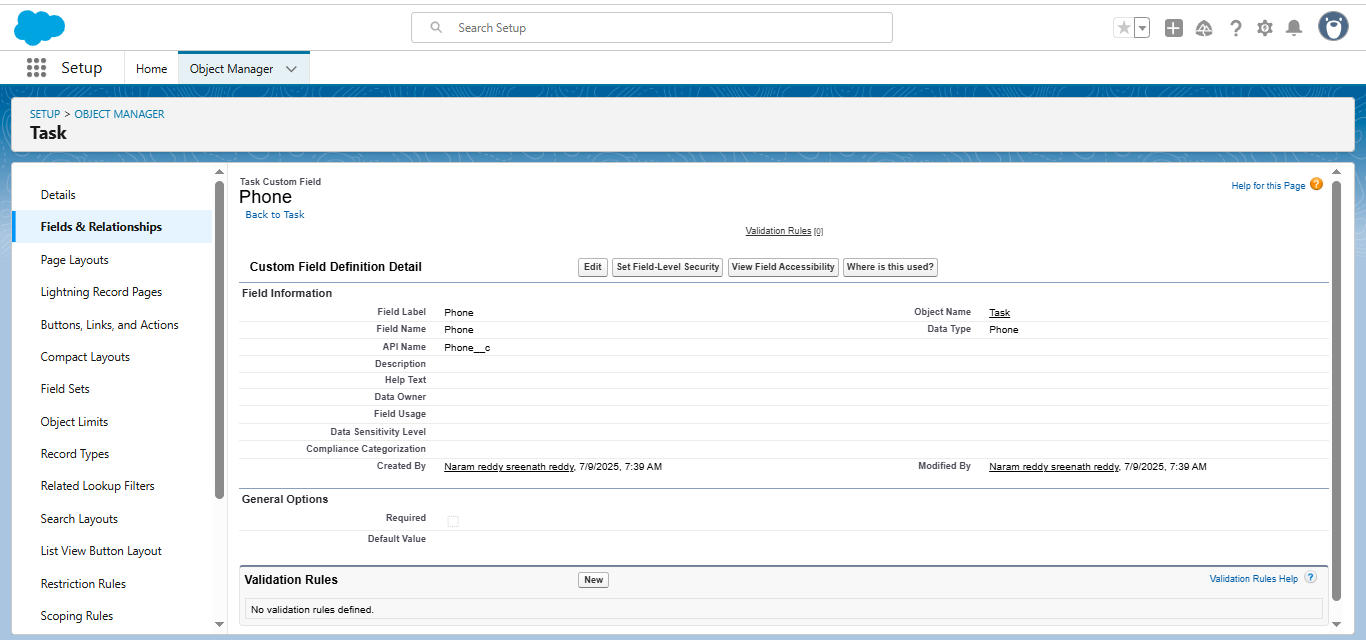
1. Field Label : Number of People Served
2. Field Name : Number\_of\_People\_Served
3. Click on required check box



1. Field Label : Name of the Person
2. Field Name : Name\_of\_the\_Person



1. Field Label : Phone
2. Field Name : Phone



1. Field Label : Rating
2. Field Name : Rating
3. Enter values, with each value separated by a new line :

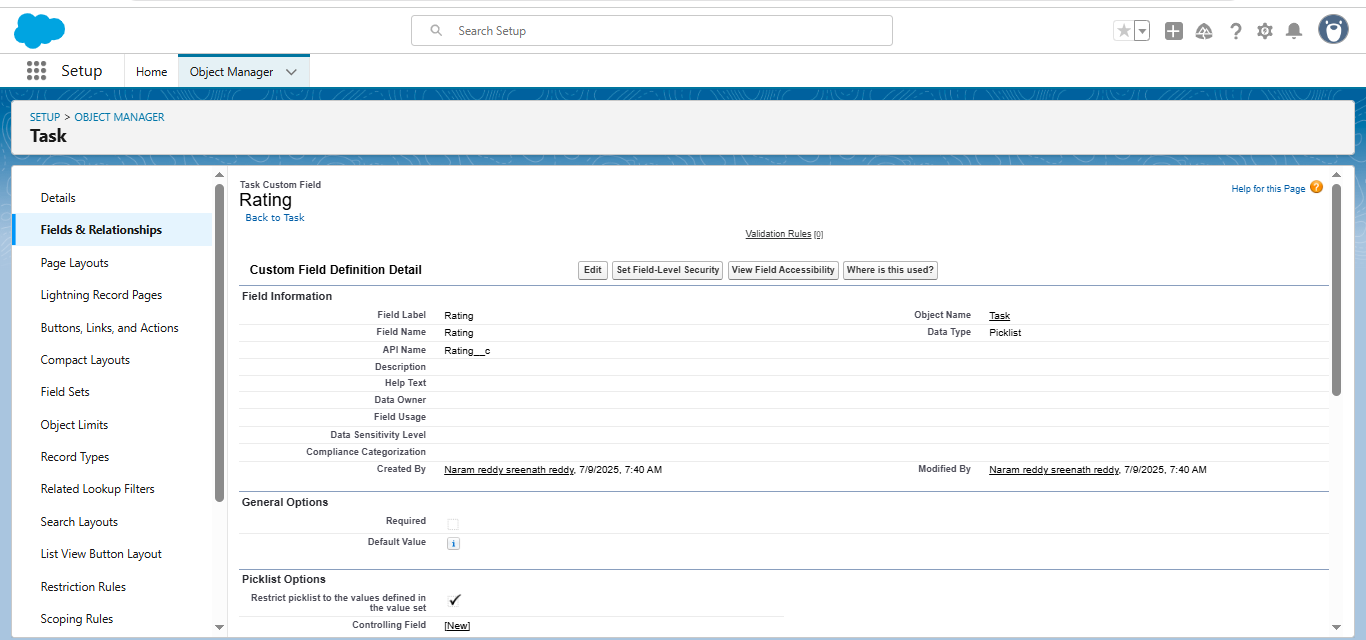
1

2

3

4

5



1. Field Label : Feedback
2. Field Name : Feedback

### 

### Create Flow to create a record in Venue object:

### 1.Select the Screen flow. Click on create.

### 2.Under the Screen Properties:

Label : Venue Details

API Name : Venue\_Details

3.Add components in this flow:

Label : Venue Name

API Name : Venue\_Name

1. Click on Email Component and name it as:

Label : Email

API Name : Contact\_Email

2.Click on Phone Component and name it as:

Label : Phone

API Name : Contact\_Phone

3.Click on Text Component and name it as:

Label : Venue Location

API Name : Venue\_Location

4.Click on Number Component and name it as:

Label : Latitude

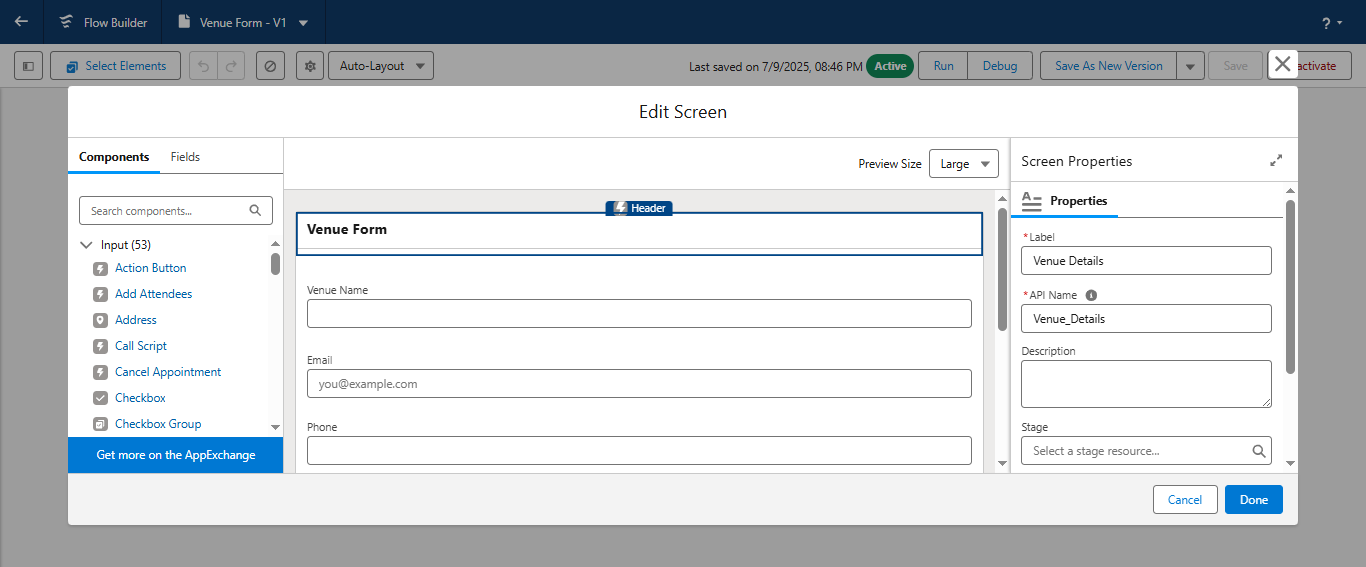
API Name : Latitude

5.Click on Number Component and name it as:

Label : longitude

API Name : longitude

Next click on Done. This would like below



Now label it as

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 the Venue : Click on ‘Add Field’ 5 times

Field : Value = Contact\_Email\_\_c : {!Contact\_Email.value}

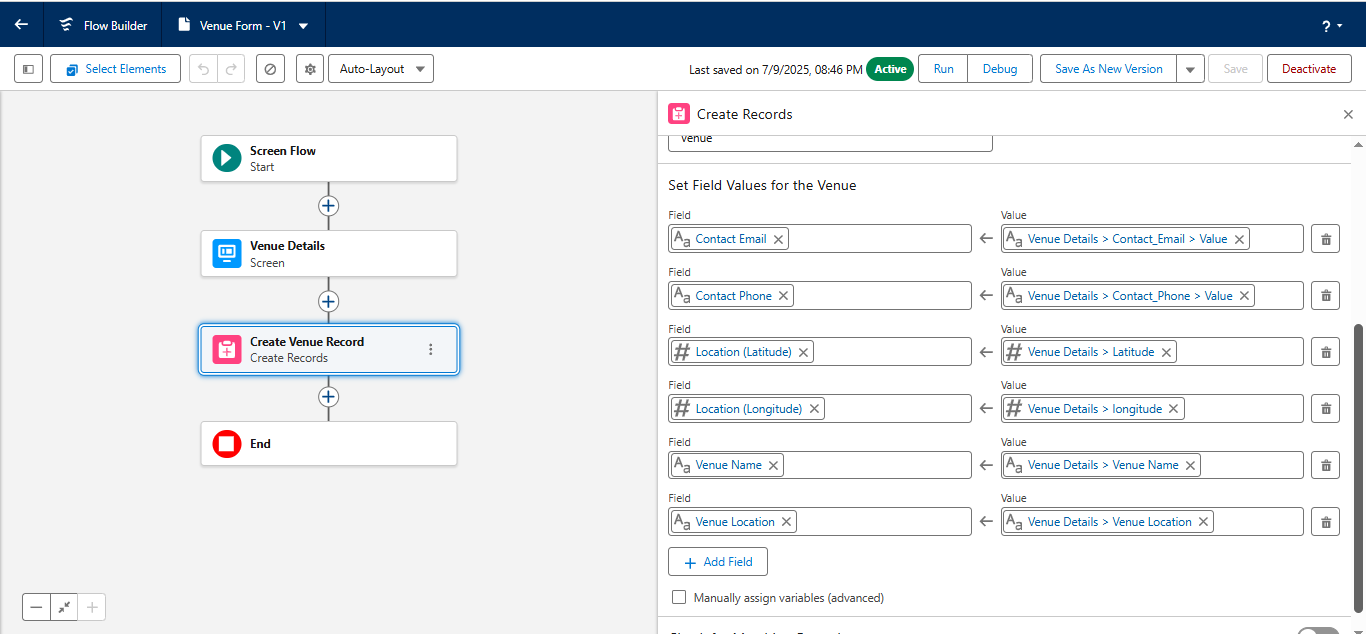
Field : Value = Contact\_Phone\_\_c : {!Contact\_Phone.value}

Field : Value = Name : {!Venue\_Name}

Field : Value = Venue\_Location\_\_c : {!location}

Field : Value = Location\_\_Latitude\_\_s : {!latitude}

Field : Value = Location\_\_Longitude\_\_s : {!longitude}



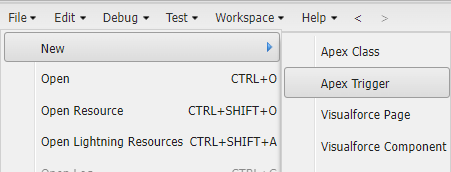
Click on Save as:

Flow Label : Venue Form

Flow API Name : Venue\_Form

### creation of triggers:

1. Click on the File menu in the toolbar, and click on new >> Trigger.
2. Enter the trigger name and the object to be triggered.



**Code:**

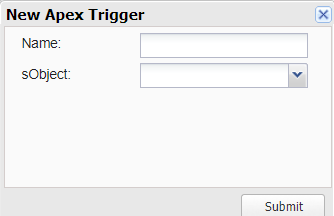
trigger DropOffTrigger on Drop\_Off\_point\_\_c (before insert) {

    for(Drop\_Off\_point\_\_c Drop : Trigger.new){

        Drop.Distance\_\_c = Drop.distance\_calculation\_\_c;

    }

}



Enter Name : DropOffTrigger

sObject: Drop-Off Point

Click on Submit.

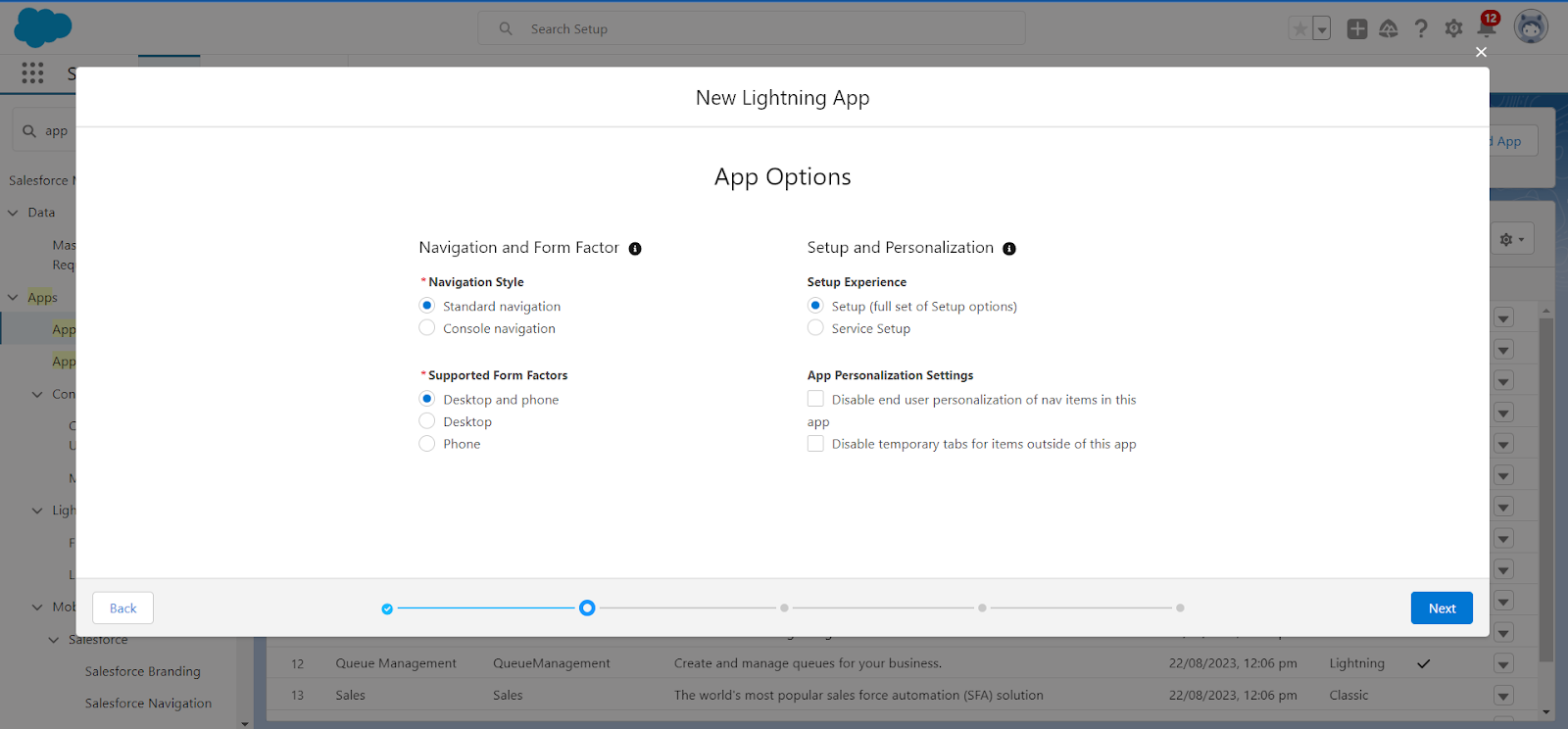
### Phase 3:UI/UX Development & Customization:

**To create a lightning app page:**

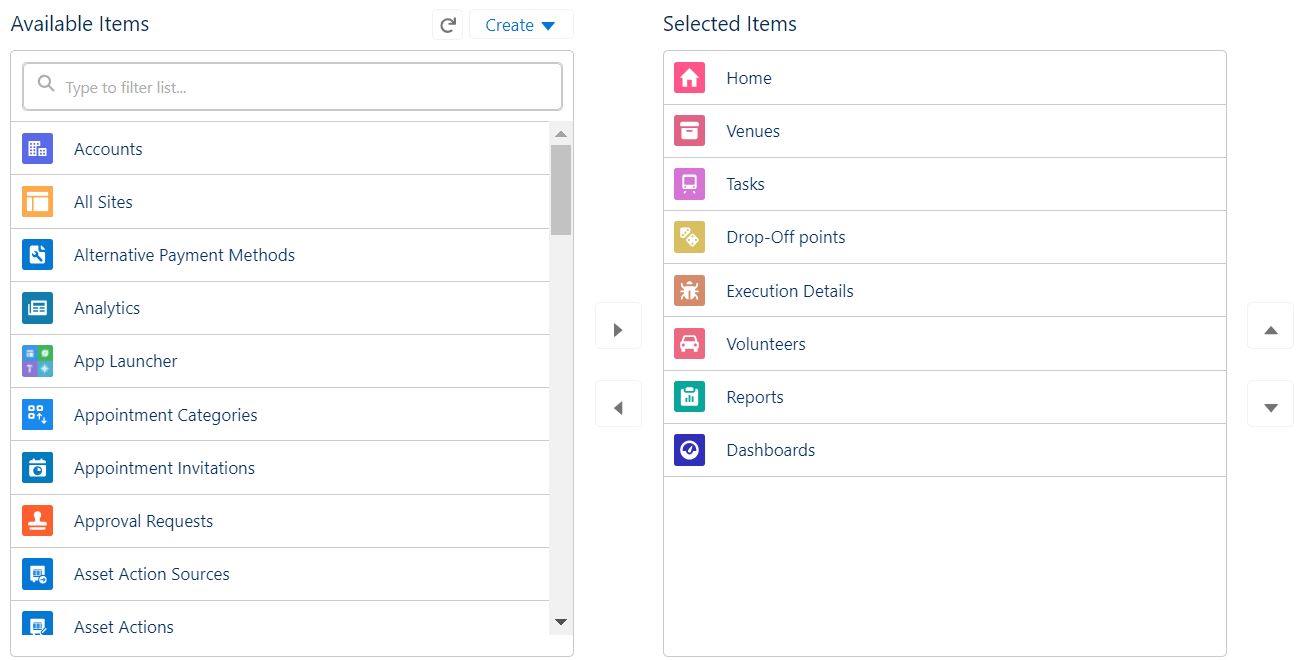
1. Go to setup page >> search “app manager” in quick find >> select “app manager” >> click on New lightning App.

2.AppName:FoodConnect  
DeveloperName:Thiswillautopopulated  
Image : optional (if you want to give any image you can otherwise not mandatory) Primary color hex value : keep this default.

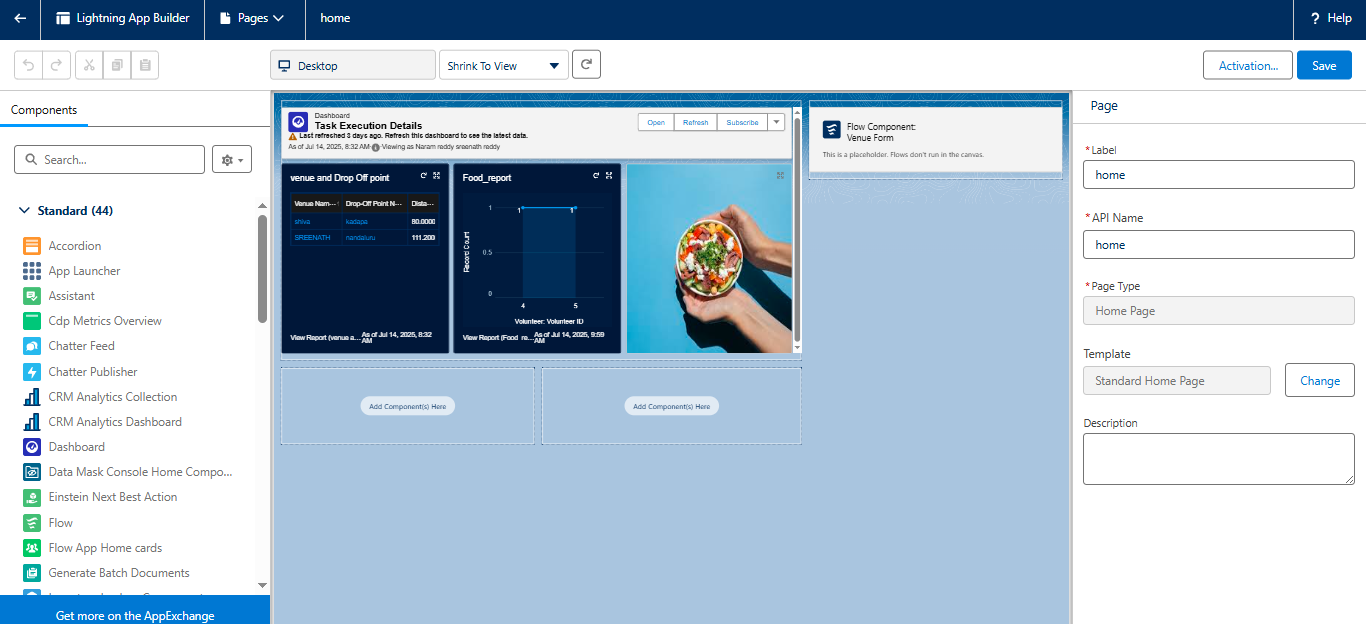
3.Then click Next  >> (App option page)Set Navigation Style as Standard Navigation >> Next.



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.



### User Management:

1. Go to setup page >> type users in Quick Find bar >> click on users>> New user.
2. In General Information give details as: (Note : create users as per your wish NGO’s)

First Name : Iksha Foundation

Last Name : Iksha\_Foundation

Alias : iiksh

Email : Give Your Email

Username : [ikshafoundation@sb.com](mailto:ikshafoundation@sb.com) (give the username different)

Nickname : Auto Populated

User License : Salesforce Platform

Profile : NGOs Profile

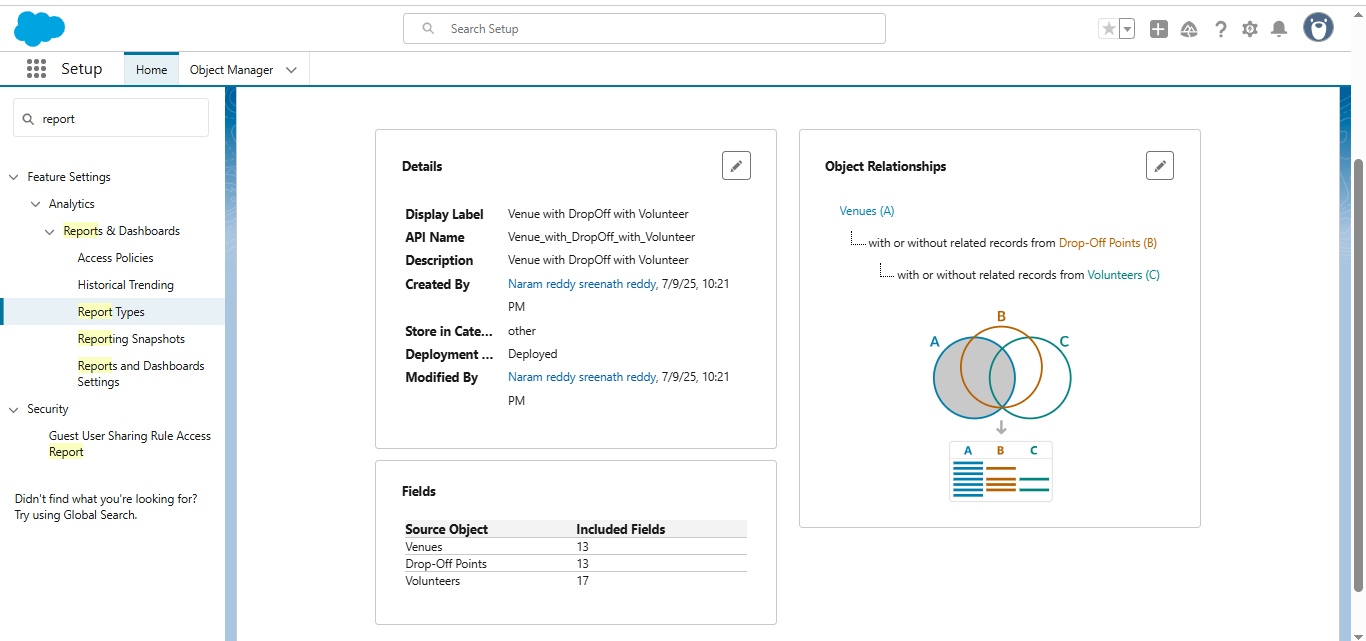
Active : Check

1. Click on Save

### 

**Reports:**

1. Go to setup page  >>  type Report Types in Quick Find bar  >> click on Report Types  >>  click on Continue   >>  Click on New Custom Report Type.
2. In Define the Custom Report Type:
3. Primary Object : Select Venues
4. Report Type Label : Venue with DropOff with Volunteer
5. Report Type Name : Venue\_with\_DropOff\_with\_Volunteer
6. Description : Venue with DropOff with Volunteer
7. Store in Category : Select Other Reports
8. Deployment Status : Deployed
9. Click on Next
10. Near Click to relate another Object Select Drop-Off Points.
11. And also select "A" records may or may not have related "B" records.
12. Now again Near Click to relate another Object Select Volunteers.
13. Now click on Save.



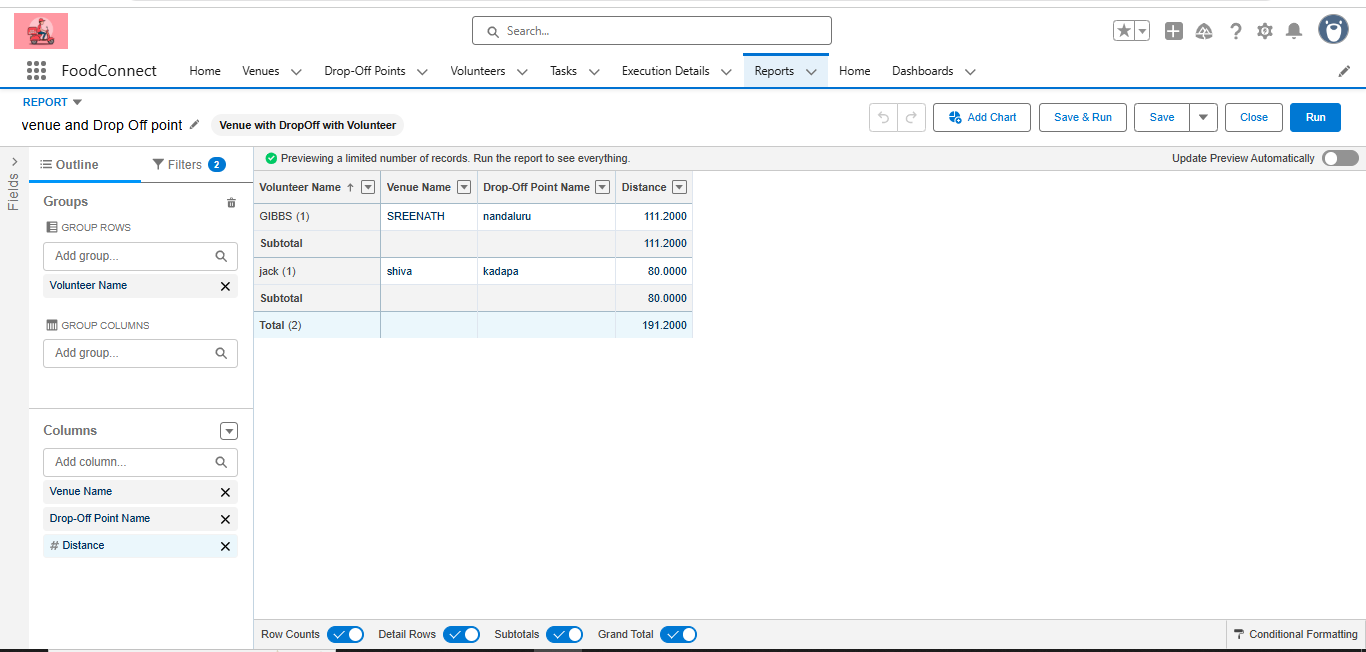
### Creation of Report :

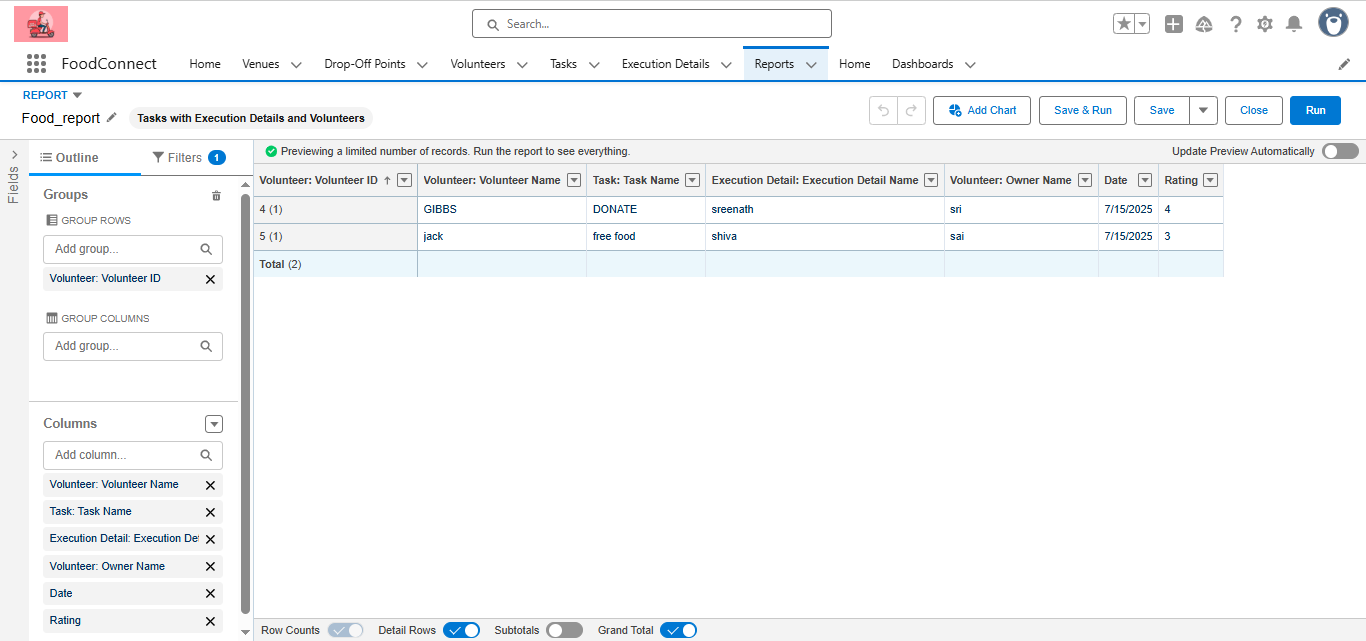
1. Click on New Folder.

Folder Label : Custom Reports

Folder Unique Name : CustomReports

1. Open Custom Reports and click on New Report
2. Select Report Type : Venue with DropOff with Volunteer
3. Then click on Start Report.
4. In GROUP ROWS : Add Volunteer Name
5. In Columns : Add Venue Name, Drop-Off point Name, Distance.
6. Now click on Save & Run.
7. Give Label as :
8. Report Name : venue and Drop Off point
9. Report Unique Name : Auto Populated
10. Click on Select Folder and select Custom Report, then click on Save.





### Dashboards:

### Creation of Report on Venue with DropOff with Volunteer:

1. Go to the app(FoodConnect)  >>  click on the reports tab
2. Click on New Folder.

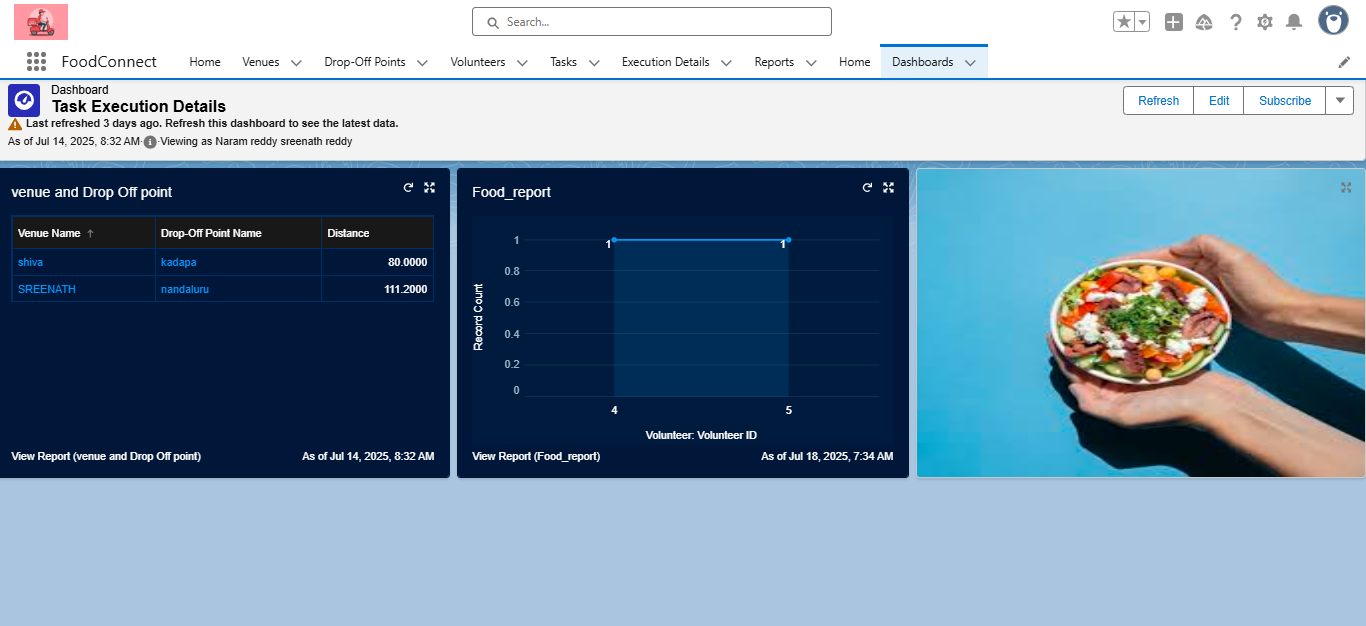
Folder Label : Custom Reports

Folder Unique Name : CustomReports

1. 3.Open Custom Reports and click on New Report
2. Select Report Type : Venue with DropOff with Volunteer
3. Then click on Start Report.
4. In GROUP ROWS : Add Volunteer Name
5. In Columns : Add Venue Name, Drop-Off point Name, Distance.
6. Now click on Save & Run.
7. Give Label as :
8. Report Name : venue and Drop Off point
9. Report Unique Name : Auto Populated
10. Click on Select Folder and select Custom Report, then click on Save.

### Creation of Report on Volunteers with Execution Details and Tasks

1. Go to the app(FoodConnect)  >>  click on the reports tab
2. Click on Custom Reports Folder and click on New Report
3. Select Report Type : Volunteers with Execution Details and Tasks.
4. Then click on Start Report.
5. In GROUP ROWS : Volunteer ID
6. In Columns : Add Volunteer : Volunteer Name, Task : Task Name, Execution Detail : Execution Detail Name, Volunteer: Owner Name, Task: Date, Task : Rating.
7. Now click on Save & Run.
8. Give Label as :
   * + 1. Report Name : Volunteer Task
       2. Report Unique Name : Auto Populated
9. Click on Select Folder and select Custom Report, then click on Save.



**Profiles, Roles, Role Hierarchy, Permission Sets, Sharing Rules:**

A robust security model was implemented using Salesforce Profiles and Roles. Profiles controlled baseline access for different users like restaurant managers, NGO coordinators, and volunteer drivers. A Role Hierarchy was established to allow NGO managers to view records of volunteers under them while keeping data compartmentalized across different NGOs. Permission Sets were used to grant additional privileges, such as access to dashboards for analytics teams, while Sharing Rules ensured specific records (like a Pickup Schedule) could be shared with relevant volunteers automatically.

**Profiles:**

1. Go to setup page >> type Profiles in Quick Find bar >> click on Profiles >> click on ‘S’
2. Click on Clone beside Standard Platform User.
3. Under Clone Profile:

Profile Name: NGOs Profile

1. Then click on Save

**Creation of sharing rules**

1. Go to setup >> type Sharing Settings in quick find box >>  Click on the Sharing Settings.
2. Scroll down and find Drop-Off point Sharing Rules.
3. Click on new near Drop-Off point Sharing Rules and Name it as:

Label: Rule 1

Rule Name: Rule\_1

1. Select your rule type: Select Based on criteria.
2. Select which records to be shared:

Field: Operator: Value = Distance: less than: 15

1. Select the users to share with: Near Share With

Public Groups: Iksha

1. Click on Save.
2. Click on new near Drop-Off point Sharing Rules and Name it as:

Label: Rule 2

Rule Name: Rule\_2

1. Select your rule type: Select Based on criteria.
2. Select which records to be shared:

Field: Operator: Value = Distance: greater than: 15

Field: Operator: Value = Distance: less or equal: 30

1. Select the users to share with: Near Share With

Public Groups: NSS

1. Click on Save.
2. Click on new near Drop-Off point Sharing Rules and Name it as:

Label: Rule 3

Rule Name: Rule\_3

1. Select your rule type: Select Based on criteria.
2. Select which records to be shared:

Field: Operator: Value = Distance: greater than: 30

Field: Operator: Value = Distance: less or equal: 50

1. Select the users to share with: Near Share With

Public Groups: Street Cause

1. Click on Save.

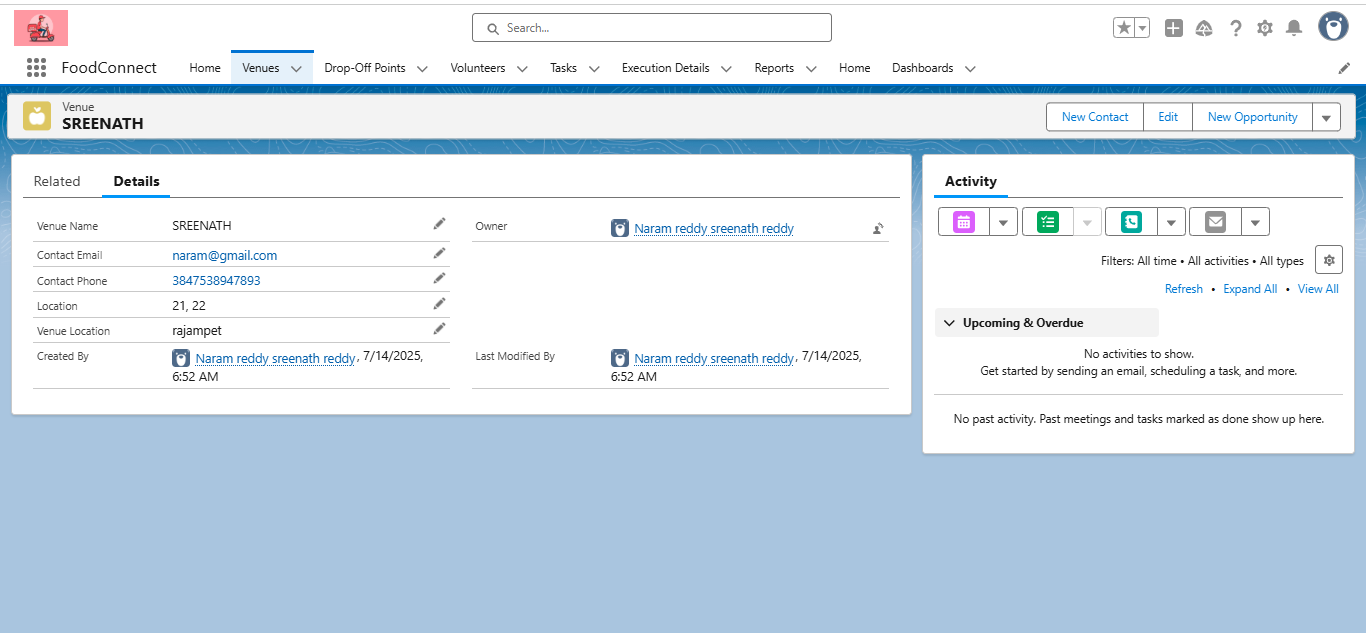
**Phase 5: Deployment, Documentation & Maintenance**

**Deployment Strategy:**

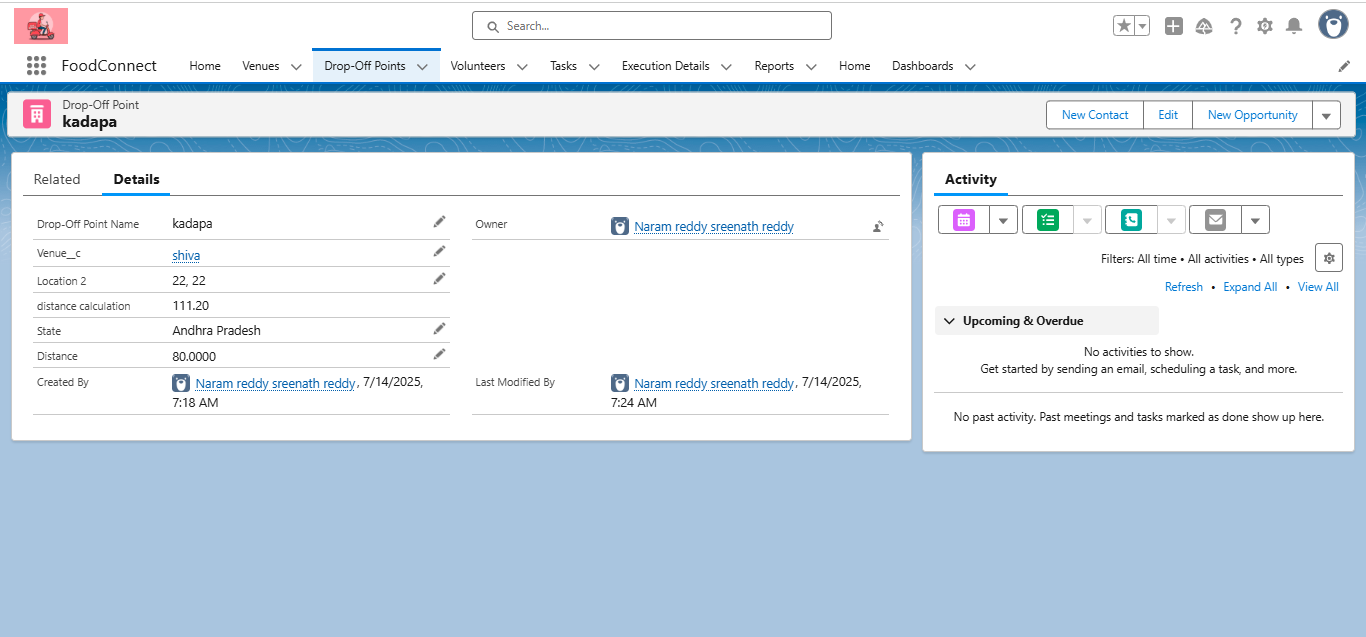
The deployment of the FOODCONNECT CRM from the sandbox environment to production was carried out using **Change Sets**, which facilitated secure and organized migration of metadata components including custom objects, fields, validation rules, flows, and Apex code. This method ensured that all dependencies were properly packaged and tested before final deployment, minimizing the risk of disruptions in the live environment.

**System Maintenance and Monitoring:**

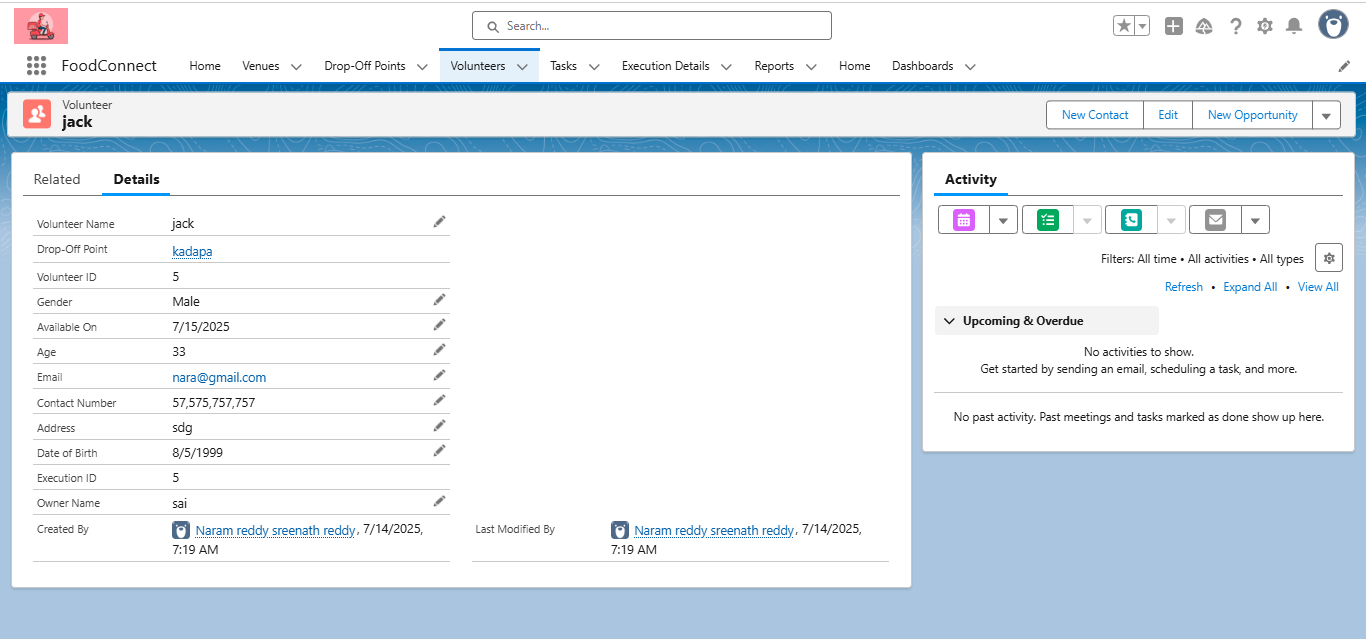
The CRM is designed for ease of ongoing maintenance. Scheduled reports and dashboards help monitor daily operations, highlighting metrics like pending pickups and distributions completed. Regular data quality reviews are performed to identify duplicates or

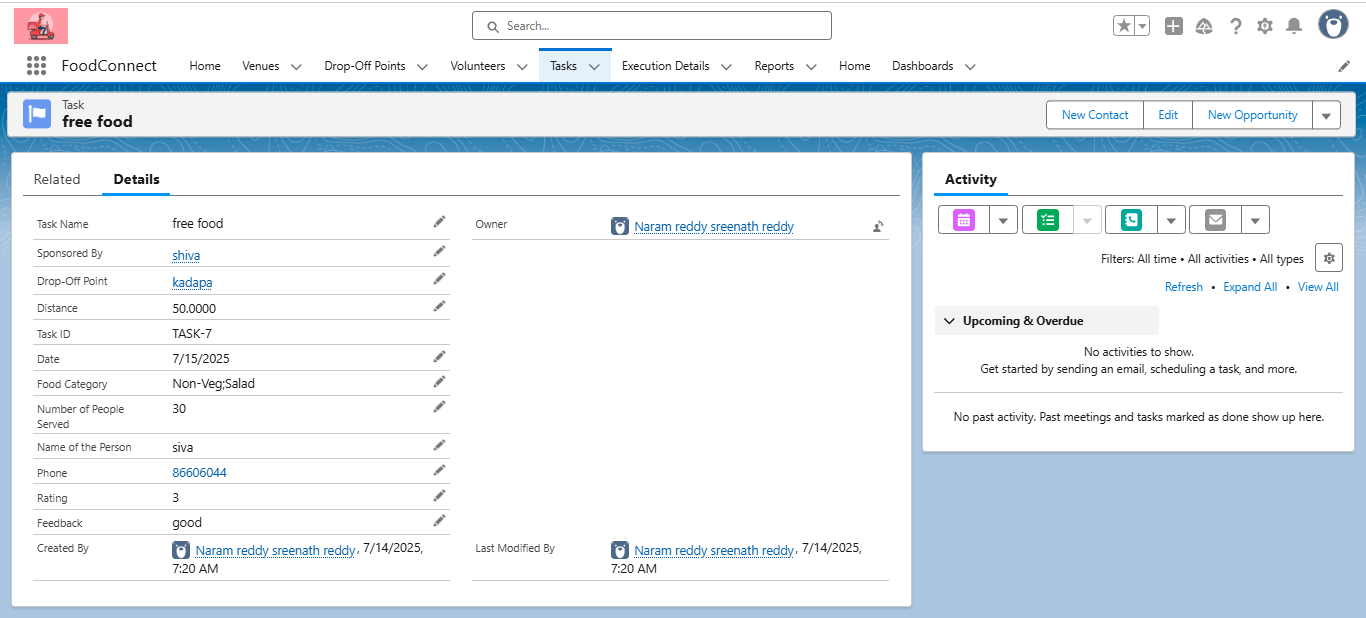


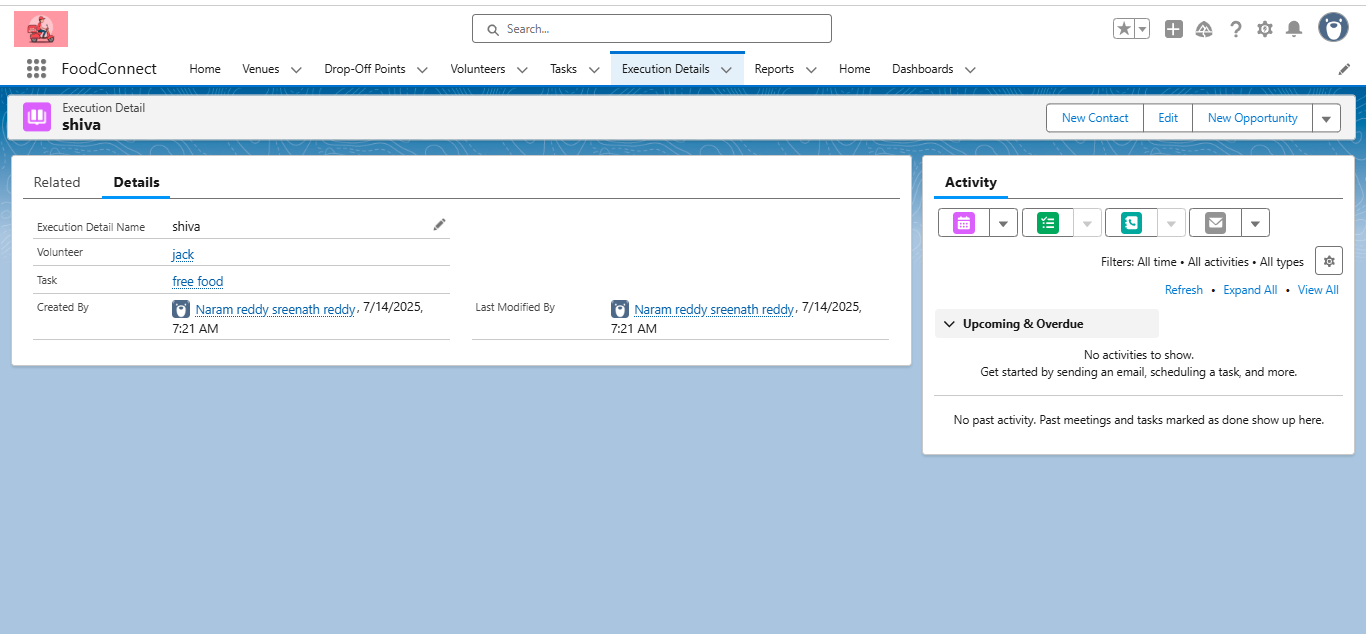
inconsistencies. Admins are responsible for periodic reviews of automation processes and ensuring that validation rules and flows continue to align with evolving business



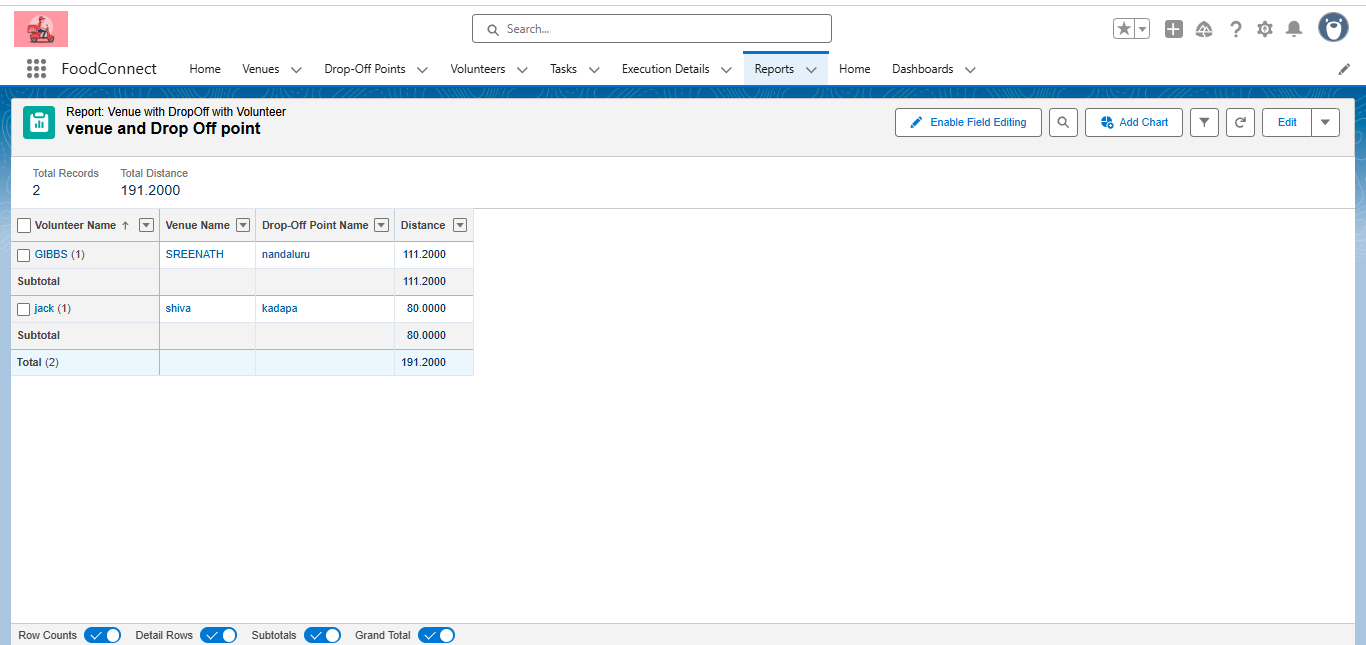
requirements. Any required enhancements or new features will be safely developed and tested in sandbox environments before being pushed to production.











A screenshot of a computer

AI-generated content may be incorrect.

**Troubleshooting Approach:**

A systematic troubleshooting guide is maintained to resolve common issues such as failed pickups, notification errors, or data mismatches. Debug logs are reviewed to trace problems in Apex triggers or automation flows. Additionally, the CRM documentation includes details of object relationships, business logic, and error messages to assist technical teams in quickly diagnosing and fixing problems. This structured approach ensures system stability and minimizes downtime, supporting the mission of timely delivery of leftover food to the needy.

**Conclusion:**

The FOODCONNECT project successfully leverages Salesforce CRM to build a robust platform that bridges the gap between surplus food sources and underprivileged communities. By automating the process of collecting, scheduling, and distributing leftover food, the system minimizes manual coordination and reduces food wastage while ensuring timely delivery to the needy. The project enhances transparency, data tracking, and donor engagement through dashboards and automated communications. In addition, it establishes a scalable framework that can be easily extended to onboard more donors, NGOs, and volunteers in the future. Looking ahead, the system offers opportunities for further enhancement, such as integrating AI-based demand forecasting and chatbot-driven donor support, ultimately strengthening the mission to fight hunger and support the underprivileged.