

---

## FOOD CONNECT

---

To Supply Leftover Food To Poor

## **Project Overview:**

*Food Connect* is a web-based and mobile-accessible platform aimed at minimizing food wastage by connecting individuals or organizations with surplus food to NGOs or volunteers who can distribute it to the poor and needy. The platform provides a simple, location-based system where food donors (like restaurants, event halls, or households) can post available leftover food, and nearby volunteers or NGOs can view, claim, and distribute that food.

By facilitating real-time communication, transparent tracking, and safe handling of food, *Food Connect* serves as a bridge between excess and hunger. It promotes sustainability, social responsibility, and community engagement through technology.

### **Key Goals:**

- Reduce edible food waste.
- Feed the hungry by redirecting surplus food.
- Enable real-time donor-volunteer coordination.
- Ensure food safety and tracking through a user-friendly interface.

### **Target Users:**

- Donors (Restaurants, Caterers, Individuals)
- Volunteers/NGOs
- Platform Administrators

### **Outcome:**

The system helps create a positive social impact by leveraging technology to ensure that no good food goes to waste while supporting communities in need.

## **Objectives:**

The primary objective of the *Food Connect* project is to reduce the wastage of edible leftover food by establishing an efficient system that redistributes it to the poor and needy. This platform aims to connect food donors such as restaurants, households, and event organizers with nearby NGOs and volunteers who can collect and deliver the surplus food to underprivileged communities. By using real-time data and GPS-based location services, the platform ensures quick matching and coordination between donors and recipients.

The project also seeks to promote social responsibility among individuals and organizations, encouraging them to contribute toward reducing hunger and supporting community welfare. An important focus is placed on food safety and traceability, ensuring that donated food is safe for consumption and delivery activities are tracked from pickup to distribution.

Additionally, the system will generate analytical reports to help monitor impact, identify high-demand zones, and guide future improvements. Ultimately, *Food Connect* is designed to be a scalable and sustainable solution that not only minimizes food waste but also fosters volunteer participation and builds a stronger, more compassionate society.

## **Phase 1: Requirement Analysis & Planning:**

### **• Understanding Business Requirements:**

The Food Connect platform aims to tackle two pressing social issues: food wastage and hunger. Donors, such as restaurants and individuals, often discard leftover food due to the absence of a structured system for redistribution. At the same time, many communities lack access to sufficient meals. This project was initiated to provide a centralized platform where donors can quickly report available food, and volunteers or organizations can access this information to arrange pickup and distribution.

During requirement analysis, it was identified that users need a simple, responsive system to post and claim food, real-time visibility of nearby donations, status tracking from pickup to delivery, and a mechanism for communication and updates. The system must be scalable, secure, and accessible across devices.

---

### **• Defining Project Scope and Objectives:**

Scope:

- Build a platform that allows food donors to share details of available surplus food.
- Enable volunteers to view and claim food donations based on their location.
- Allow administrators to oversee all platform activity and user management.
- Automate status updates, alerts, and coordination through workflows.
- Generate insights on donation trends, volunteer performance, and delivery coverage.

Objectives:

- Reduce food wastage by enabling timely redistribution of surplus food.
  - Facilitate a seamless connection between donors and volunteers using real-time data.
  - Simplify food tracking from posting to final delivery.
  - Ensure secure access and data privacy across all user roles.
  - Promote social responsibility and volunteer engagement through a digital medium.
-

- **Design Data Model and Security Model:**

**Data Model Design:**

The platform's structure includes components to manage user information, food donation details, delivery activities, and system communications. These elements are linked together to support operations such as posting, claiming, updating status, and generating reports. The relationships between data entries are defined to ensure a smooth flow of information and traceability of every donation made.

**Security Model Design:**

The system applies role-based access control, ensuring that each user (donor, volunteer, admin) has appropriate permissions. Donors can only manage their own contributions, while volunteers access donation listings and update their own deliveries. Administrators maintain full access to manage users and oversee the platform. Access to sensitive information is limited, and data visibility is controlled through carefully applied security settings. Authentication, validation rules, and secure channels ensure safe and reliable use of the platform.

## **Phase 2: Salesforce Development – Backend & Configurations:**

- **Setup Environment & DevOps Workflow:**

In this phase, the development environment was prepared by setting up Salesforce sandboxes for configuration, testing, and deployment purposes. A structured DevOps workflow was followed to manage the development lifecycle efficiently. Changes were made in a developer sandbox and then promoted to a test environment before final deployment to production. Source control tools were used to track changes, and Salesforce Change Sets or deployment tools like Salesforce CLI were used to migrate configurations and code.

This ensured a clean separation between development, testing, and production environments, minimizing risks and allowing for structured reviews and rollback if necessary.

- **Customization of Objects, Fields, Validation Rules, and Automation:**

**Create Venue Object**

To create an object:

1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
  - Enter the label name >> Venue
  - Plural label name >> Venues
  - Enter Record Name Label and Format

1. Record Name >> Venue Name
2. Data Type >> Text
2. Click on Allow reports and Track Field History, Allow Activities.
3. Allow search >> Save.

The screenshot shows the Salesforce Setup interface for managing objects. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. The main area is titled 'SETUP > OBJECT MANAGER' and shows the 'Venue' object. The left sidebar lists various configuration tabs: 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, Restriction Rules, and Scoping Rules. The 'Details' tab is currently active. In the main content area, there are two sections: 'Description' and 'Settings'. The 'Description' section contains fields for 'API Name' (set to 'Venue\_\_c'), 'Custom' (checked), 'Singular Label' (set to 'Venue'), and 'Plural Label' (set to 'Venues'). The 'Settings' section contains checkboxes for 'Enable Reports' (checked), 'Track Activities' (checked), and 'Track Field History' (checked). Below these settings, 'Deployment Status' is set to 'Deployed' and 'Help Settings' are set to 'Standard salesforce.com Help Window'.

## Create Drop-Off Point Object

To create an object:

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

  - Enter the label name >> Drop-Off Point
  - Plural label name >> Drop-Off Points
  - Enter Record Name Label and Format
    - a. Record Name >> Drop-Off point Name
    - b. Data Type >> Text

2. Click on Allow reports and Track Field History, Allow Activities
3. Allow search >> Save.

The screenshot shows the Salesforce Setup interface under the Object Manager. A new object named 'Drop-Off Point' is being created. The 'Details' tab is selected, showing the following configuration:

- Description:** None
- API Name:** Drop\_Off\_Point\_c
- Custom:** ✓
- Singular Label:** Drop-Off Point
- Plural Label:** Drop-Off Points
- Enable Reports:** ✓
- Track Activities:** ✓
- Track Field History:** ✓
- Deployment Status:** Deployed
- Help Settings:** Standard salesforce.com Help Window

The left sidebar lists various object settings: 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, Restriction Rules, and Scoping Rules.

## Create Task Object

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 >> Task
  2. Plural label name >> Tasks
  3. Enter Record Name Label and Format
    - Record Name >> Task Name
    - Data Type >> Text
2. Click on Allow reports and Track Field History, Allow Activities
3. Allow search >> Save.

The screenshot shows the Salesforce Setup interface under the Object Manager. A new object named 'Task' is being created. The 'Details' tab is selected, showing the following configuration:

- Description:** None
- API Name:** Task\_c
- Custom:** ✓
- Singular Label:** Task
- Plural Label:** Tasks
- Enable Reports:** ✓
- Track Activities:** ✓
- Track Field History:** ✓
- Deployment Status:** Deployed
- Help Settings:** Standard salesforce.com Help Window

The left sidebar lists various object settings: 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, Restriction Rules, and Scoping Rules.

## Create Volunteer Object

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 >> Volunteer
  2. Plural label name >> Volunteers
  3. Enter Record Name Label and Format
    - Record Name >> Volunteer Name
    - Data Type >> Text
2. Click on Allow reports and Track Field History, Allow Activities
3. Allow search >> Save.

The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes 'Setup', 'Home', 'Object Manager', and a search bar labeled 'Search Setup'. Below the header, the breadcrumb trail reads 'SETUP > OBJECT MANAGER' and the specific object name is 'Volunteer'. On the left, a sidebar lists various configuration options: 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, Restriction Rules, and Scoping Rules. The main workspace is titled 'Details' and contains several input fields and checkboxes. The 'Description' field is empty. The 'API Name' field is set to 'Volunteer\_c'. Under the 'Custom' section, there is a checked checkbox. The 'Singular Label' field is set to 'Volunteer' and the 'Plural Label' field is set to 'Volunteers'. To the right of these fields are three checkboxes: 'Enable Reports' (checked), 'Track Activities' (checked), and 'Track Field History' (unchecked). Further down, the 'Deployment Status' is set to 'Deployed' and the 'Help Settings' link points to 'Standard salesforce.com Help Window'. At the bottom right of the workspace are 'Edit' and 'Delete' buttons.

## Create Execution Detail Object

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 >> Execution Detail
  2. Plural label name >> Execution Details
  3. Enter Record Name Label and Format
    - Record Name >> Execution Detail Name

- Data Type >> Text
- Click on Allow reports and Track Field History, Allow Activities
  - Allow search >> Save.

The screenshot shows the Salesforce Setup interface under the Object Manager. The left sidebar lists various configuration options like Fields & Relationships, Page Layouts, and Lightning Record Pages. The main area displays the 'Execution Detail' object's details. The 'Details' tab is selected, showing fields such as API Name (Execution\_Detail\_\_c), Singular Label (Execution Detail), and Plural Label (Execution Details). On the right, there are checkboxes for enabling Reports, Activities, and Field History, all of which are checked. Deployment status is set to 'Deployed'. Help settings point to the 'Standard salesforce.com Help Window'.

## Creation of Relationship fields in objects

Creation of Lookup Relationship Field on Volunteer Object:

1. Go to setup >> click on Object Manager >> type object name (Volunteer) in the search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Master Detail relationship
4. Select the related object “Drop-Off point” and click next.
5. Field Name: Drop\_off point
6. Field label: Auto generated
7. Next >> Next >> Save.

The screenshot shows the Salesforce Setup interface under the 'Object Manager' section for the 'Volunteer' object. A custom field named 'Drop\_Off\_point' is being edited. The 'Field & Relationships' tab is selected. The 'Field Information' section shows the field label as 'Drop\_Off\_point', field name as 'Drop\_Off\_point', and API name as 'Drop\_Off\_point\_c'. The 'Object Name' is set to 'Volunteer' and the 'Data Type' is 'Master-Detail'. The 'Master-Detail Options' section shows the related object as 'Drop-Off Point' and the child relationship name as 'Volunteers'. The 'Lookup Filter' section is collapsed.

### Creation of Master Detail Relationship Field on Execution Details Object:

8. Go to setup >> click on Object Manager >> type object name (Execution Details) in the search bar >> click on the object.
9. Now click on “Fields & Relationships” >> New
10. Select Master Detail relationship
11. Select the related object “Volunteer” and click next.
12. Field Name: Volunteer
13. Field label: Auto generated
14. Next >> Next >> Save.

The screenshot shows the Salesforce Setup interface under the 'Object Manager' section for the 'Execution Detail' object. A custom field named 'Volunteer' is being edited. The 'Field & Relationships' tab is selected. The 'Field Information' section shows the field label as 'Volunteer', field name as 'Volunteer', and API name as 'Volunteer\_\_c'. The 'Object Name' is set to 'Execution Detail' and the 'Data Type' is 'Master-Detail'. The 'Master-Detail Options' section shows the related object as 'Volunteer' and the child relationship name as 'Execution\_Details'. The 'Lookup Filter' section is collapsed.

### Creation of Master Detail Relationship Field on Execution Details Object :

15. Go to setup >> click on Object Manager >> type object name(Execution Details) in the search bar >> click on the object.
16. Now click on “Fields & Relationships” >> New
17. Select Master Detail relationship
18. Select the related object “Task” and click next.
19. Field Name: Task
20. Field label: Auto generated
21. Next >> Next >> Save.

**Execution Detail Custom Field**  
**Task**

**Custom Field Definition Detail**

Field Information		Object Name: Execution_Detail	Data Type: Master-Detail
Field Label:	Task		
Field Name:	Task		
API Name:	Task__c		
Description:			
Help Text:			
Data Owner:			
Field Usage:			
Data Sensitivity Level:			
Compliance Categorization:			
Created By:	Munagapati Mounika, 7/15/2025, 5:41 AM	Modified By:	Munagapati Mounika, 7/15/2025, 5:41 AM

**Master-Detail Options**

Related To:	Task	Child Relationship Name:	Execution_Details
Related List Label:	Execution Details		
Sharing Setting:	Read/Write. Allows users with at least Read/Write access to the Master record to create, edit, or delete related Detail records.		
Representable Master Detail:	<input checked="" type="checkbox"/>		

**Lookup Filter**

#### Creation of Lookup Relationship Field on Drop-Off Point Object:

22. Go to setup >> click on Object Manager >> type object name (Drop-Off Point) in the search bar >> click on the object.
23. Now click on “Fields & Relationships” >> New
24. Select Lookup relationship
25. Select the related object “Venue” and click next.
26. Field Name: Venue
27. Field label: Venue\_\_c
28. Next >> Next >> Save.

The screenshot shows the Salesforce Setup interface under the Object Manager section for the 'Drop-Off Point' object. A custom field named 'Venue\_\_c' is being created. The 'Field Information' section shows the field label as 'Venue\_\_c', field name as 'Venue', and API name as 'Venue\_\_c'. The object name is 'Drop-Off Point' and the data type is 'Lookup'. The 'Lookup Options' section indicates the field is related to the 'Venue' object, with a child relationship name 'Drop\_Off\_Points'. The 'Field & Relationships' sidebar on the left lists various setup options like Page Layouts, Lightning Record Pages, and Field Sets.

### Creation of Lookup Relationship Field on Task Object:

29. Go to setup>> click on Object Manager >> type object name (Task) in the search bar >> click on the object.
30. Now click on “Fields & Relationships” >> New
31. Select Lookup relationship
32. Select the related object “Venue” and click next.
33. Field Name: Sponsored By
34. Field label: Auto generated
35. Next >> Next >> Save.

The screenshot shows the Salesforce Setup interface under the Object Manager section for the 'Task' object. A custom field named 'Sponsored By' is being created. The 'Field Information' section shows the field label as 'Sponsored By', field name as 'Sponsored\_By', and API name as 'Sponsored\_By\_\_c'. The object name is 'Task' and the data type is 'Lookup'. The 'Lookup Options' section indicates the field is related to the 'Venue' object, with a child relationship name 'Tasks'. The 'Field & Relationships' sidebar on the left lists various setup options like Page Layouts, Lightning Record Pages, and Field Sets.

### Creation of Lookup Relationship Field on Task Object:

36. Go to setup>> click on Object Manager >> type object name(Task) in the search bar >> click on the object.
37. Now click on “Fields & Relationships” >> New
38. Select Lookup relationship
39. Select the related object “Drop-Off point” and click next.
40. Field Name: Drop-Off point
41. Field label: Auto generated
42. Next >> Next >> Save.

The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. The main title is 'SETUP > OBJECT MANAGER' followed by 'Task'. On the left, a sidebar lists various configuration options like 'Page Layouts', 'Lightning Record Pages', etc. The main content area is titled 'Task Custom Field Drop-Off point'. It shows the 'Custom Field Definition Detail' section with tabs for 'Edit', 'Set Field-Level Security', 'View Field Accessibility', and 'Where Is This Used?'. Under 'Field Information', fields include 'Field Label' (Drop-Off point), 'Field Name' (Drop\_Off\_point), 'API Name' (Drop\_Off\_point\_c), 'Description', 'Help Text', 'Data Owner', 'Field Usage', 'Data Sensitivity Level', 'Compliance Categorization', and 'Created By' (Munagapati Mounika, 7/15/2025, 5:46 AM). A note indicates 'Modified By' (Munagapati Mounika, 7/15/2025, 5:46 AM). Under 'Lookup Options', it shows 'Related To' (Drop-Off Point), 'Related List Label' (Tasks), and 'Required' (unchecked). A note says 'What to do if the lookup record is deleted?' with an option to 'Clear the value of this field'. At the bottom, there's a 'Lookup Filter' section.

## **Creation of fields for the Venue object**

1. Go to setup>> click on Object Manager >> type object name(Venue) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “Email” and Click on Next
4. Fill the Above as following:
  - Field Label: Contact Email
  - Field Name: Contact Email
  - Click on required check box
  - Click on Next >> Next >> Save and new.

The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. The main title is 'SETUP > OBJECT MANAGER' followed by 'Venue'. On the left, a sidebar lists various setup options under 'Fields & Relationships'. The main content area displays the 'Custom Field Definition Detail' for 'Contact Email'. Key details shown include:

- Field Information:** Field Label: Contact Email, Field Name: Contact\_Email, API Name: Contact\_Email\_c.
- General Options:** Required is checked, Unique is unchecked, External ID is unchecked.
- Validation Rules:** A 'New' button is present.
- Object Details:** Object Name: Venue, Data Type: Email.
- Timestamps:** Created By: Munagapati Mounika, 7/15/2025, 5:52 AM; Modified By: Munagapati Mounika, 7/15/2025, 5:52 AM.

### To create another fields in an object:

5. Go to setup >> click on Object Manager >> type object name(Venue) in search bar >> click on the object.
5. Now click on “Fields & Relationships” >> New
5. Select Data type as a “Phone” and Click on Next

This screenshot shows the continuation of creating a new custom field. The setup process has moved to the 'Fields & Relationships' section under 'Object Manager' for the 'Venue' object. The 'Contact Phone' field is being defined. The 'Field Information' section includes:

- Field Label: Contact Phone
- Field Name: Contact\_Phone
- API Name: Contact\_Phone\_c

The 'General Options' section shows 'Required' is checked. In the 'Validation Rules' section, it notes 'No validation rules defined.' The right side of the screen shows standard Salesforce navigation and help icons.

5. Fill the Above as following:

- Field Label: Contact Phone
- Field Name: Contact Phone
- Click on required check box

- Click on Next >> Next >> Save and new.

**To create another fields in an object:**

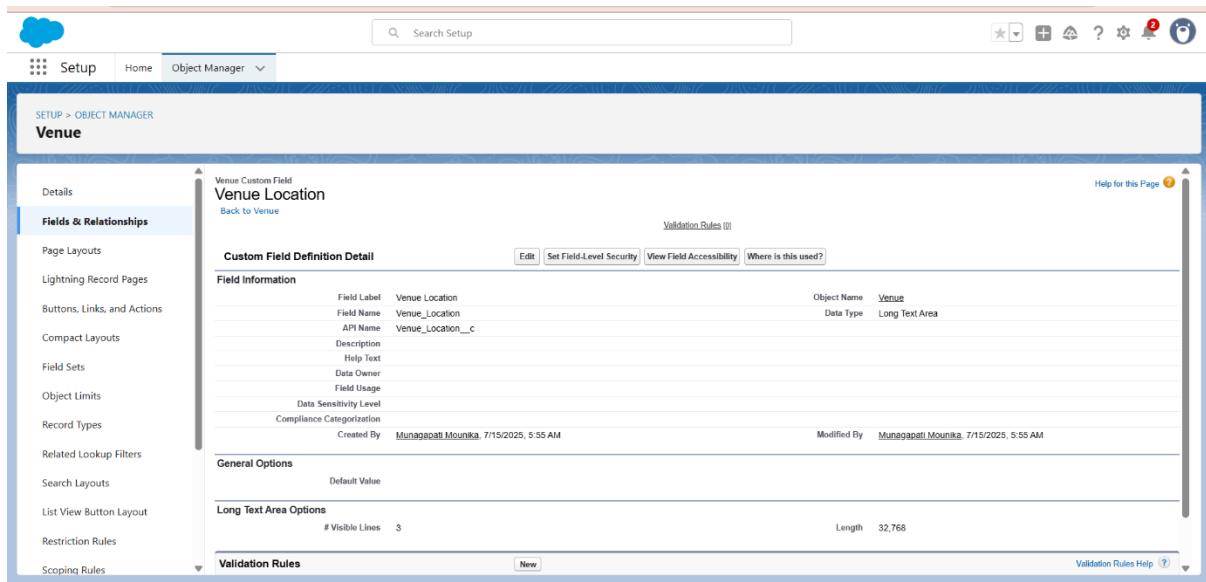
1. Go to setup >> click on Object Manager >> type object name(Venue) in search bar >>click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “Geolocation” and Click on Next

4. Fill the Above as following:

- Field Label: Location
- Decimal Places: 4
- Field Name: Location
- Description: Enter the Geolocation of your Venue
- Click on Next >> Next >> Save and new.

**To create other fields in an object:**

9. Go to setup >> click on Object Manager >> type object name(Venue) in search bar >> click on the object.
9. Now click on “Fields & Relationships” >> New
9. Select Data type as a “Long Text Area” and Click on Next



9. Fill the Above as following:

- Field Label : Venue Location
- Field Name : Venue\_Location
- Click on Next >> Next >> Save and new.

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

Go to setup >> click on Object Manager >> type object name(Drop-Off point) in search bar >> click on the object.

2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “Geolocation” and Click on Next
4. Fill the Above as following:

- Field Label: Location 2
- Field Name: gets auto generated
- Description: Enter the Geolocation of the Drop off Point
- Geolocation Options: select Decimal
- Decimal Places: 4
- Click on Next >> Next >> Save and new.

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  
Restriction Rules  
Scoping Rules

Drop-Off Point Custom Field  
**Location 2**  
Back to Drop-Off Point

Custom Field Definition Detail

Field Information

Field Label	Location 2	Object Name	Drop-Off Point
Field Name	Location_2	Data Type	Geolocation
API Name	Location_2_c		
Description	Enter the Geolocation of the Drop off Point		
Help Text			
Data Owner			
Field Usage			
Data Sensitivity Level			
Compliance Categorization			

Created By: Munagapati Mounika, 7/15/2025, 6:00 AM Modified By: Munagapati Mounika, 7/15/2025, 6:00 AM

General Options

Required	<input type="checkbox"/>
----------	--------------------------

Geolocation Options

Decimal	<input checked="" type="checkbox"/>
Decimal Places	4

To create another fields in an object:

1. Go to setup >> click on Object Manager >> type object name (Drop-Off point) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “Formula” and Click on Next

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  
Restriction Rules  
Scoping Rules

Drop-Off Point Custom Field  
**distance calculation**  
Back to Drop-Off Point

Custom Field Definition Detail

Field Information

Field Label	distance calculation	Object Name	Drop-Off Points
Field Name	distance_calculation		
API Name	distance_calculation_c		
Description			
Help Text			
Data Owner			
Field Usage			
Data Sensitivity Level			
Compliance Categorization			

Created By: Munagapati Mounika, 7/19/2025, 8:27 AM Modified By: Munagapati Mounika, 7/19/2025, 8:27 AM

Formula Options

Data Type	Formula $\text{km}$
Decimal Places	2

DISTANCE(Location\_2\_c, Venue\_\_r.Location\_\_c, 'km')

4. Fill the Above as following:
  - Field Label: distance calculation
  - Field Name: distance\_calculation
  - Formula Return Type: Number

- Formula Options: DISTANCE( Location\_2\_\_c , Venue\_\_r.Location\_\_c , 'km')
- Click on Next >> Next >> Save and new.

To create another fields in an object:

5. Go to setup >> click on Object Manager >> type object name (Drop-Off point) in search bar >> click on the object.

5. Now click on “Fields & Relationships” >> New

5. Select Data type as a “Picklist” and Click on Next

5. Fill the Above as following:

- Field Label: State
- Field Name: State
- Enter values, with each value separated by a new line:

Andhra Pradesh

Arunachal Pradesh

Assam

Bihar

Chhattisgarh

Goa

Gujarat

Haryana

Himachal Pradesh

Jharkhand

Karnataka

Kerala

Maharashtra

Madhya Pradesh

Manipur

Meghalaya

Mizoram

- Nagaland
  - Odisha
  - Punjab
  - Rajasthan
  - Sikkim
  - Tamil Nadu
  - Tripura
  - Telangana
  - Uttar Pradesh
  - Uttarakhand
  - West Bengal
  - Andaman & Nicobar (UT)
  - Chandigarh (UT)
  - Dadra & Nagar Haveli and Daman & Diu (UT)
  - Delhi [National Capital Territory (NCT)]
  - Jammu & Kashmir (UT)
  - Ladakh (UT)
  - Lakshadweep (UT)
  - Puducherry (UT)
- Click on required check box
  - Click on Next >> Next >> Save and new.
  - Click on Next >> Next >> Save and new.

To create another fields in an object:

9. Go to setup >> click on Object Manager >> type object name(Drop-Off point) in search bar >> click on the object.
9. Now click on “Fields & Relationships” >> New
9. Select Data type as a “Number” and Click on Next

10. Fill the Above as following:

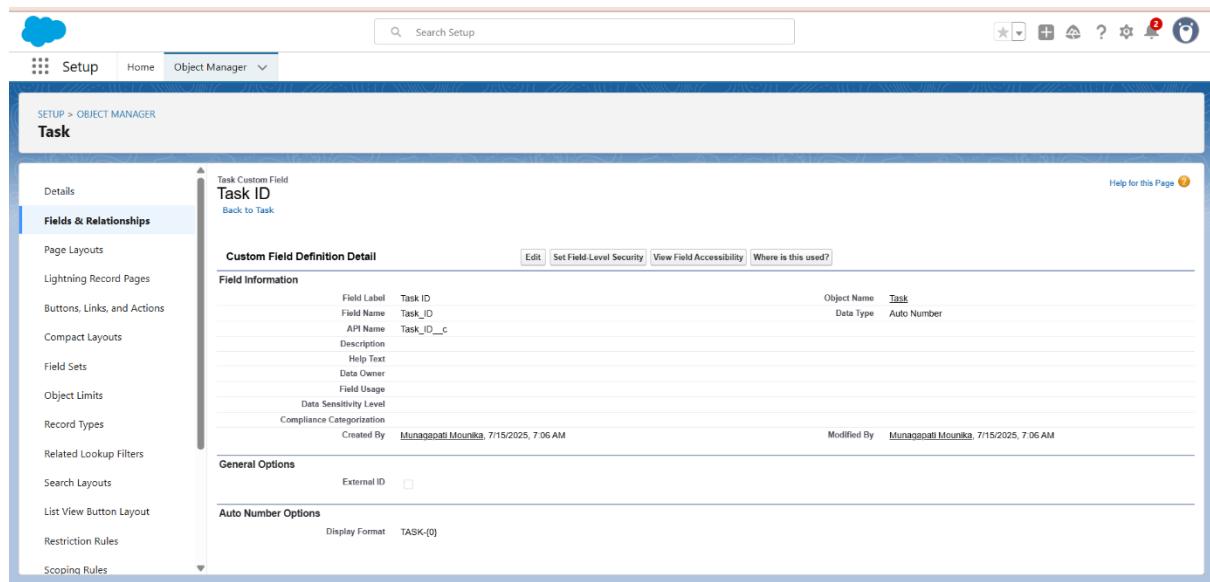
- Field Label: Distance
- Field Name: Distance
- Length: 14

- Decimal Places: 4
- Click on required check box
- Click on Next >> Next >> Save and new.

### Creation of fields for the Task object

Go to setup >> click on Object Manager >> type object name (Task) in search bar >> click on the object.

2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “Auto Number” and Click on Next

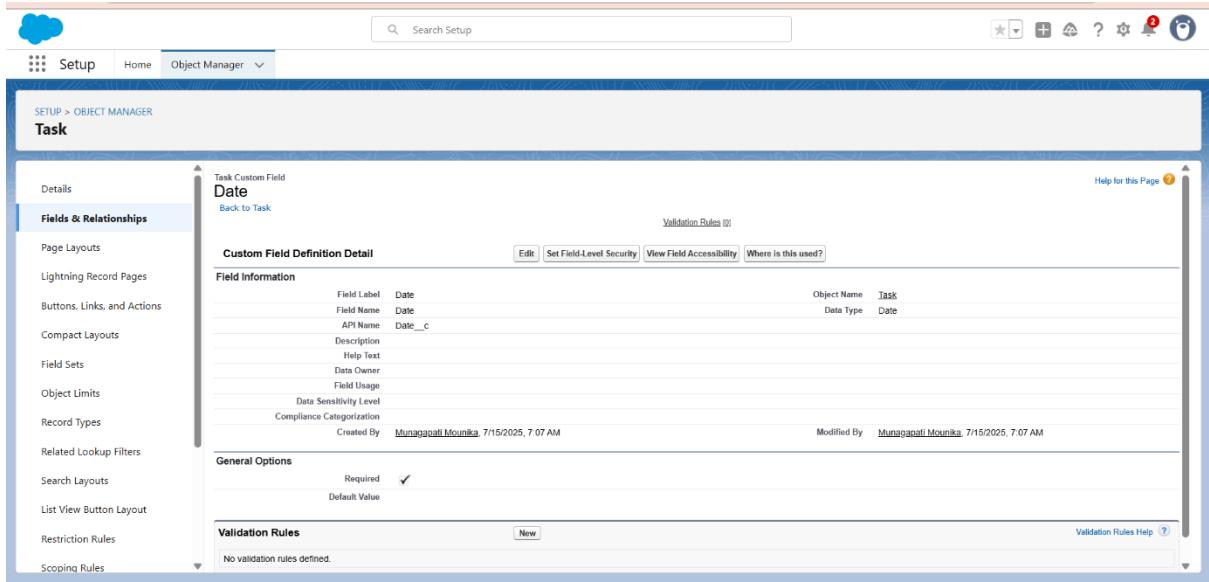


### 4. Fill the Above as following:

- Field Label: Task ID
- Display Format: TASK- {0}
- Starting Number: 1
- Field Name: gets auto generated
- Click on required check box
- Click on Next >> Next >> Save and new.

To create another fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Task) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “Date” and Click on Next



4. Fill the Above as following:
  - Field Label: Date
  - Field Name: Date
  - Click on required check box
  - Click on Next >> Next >> Save and new.

To create another fields in an object:

5. Go to setup >> click on Object Manager >> type object name (Task) in search bar >> click on the object.
5. Now click on “Fields & Relationships” >> New
5. Select Data type as a “Picklist (Multi-Select)” and Click on Next

The screenshot shows the Salesforce Setup interface with the following details:

- Object Manager**: Task
- Custom Field Definition Detail**: Food Category
- Field Information**:
  - Field Label: Food Category
  - Field Name: Food\_Category
  - API Name: Food\_Category\_\_c
  - Description: (empty)
  - Help Text: (empty)
  - Data Owner: (empty)
  - Data Sensitivity Level: (empty)
  - Compliance Categorization: (empty)
- General Options**:
  - Required: checked
  - Default Value: (empty)
- Picklist (Multi-Select) Options**:
  - Restrict picklist to the values defined in the value set: checked
  - Controlled by Field: (empty)
  - Max: (empty)

5. Fill the Above as following:

- Field Label: Food Category
- Field Name: Food Category
- Enter values, with each value separated by a new line :

Veg

Non-Veg

Salad

Snack

- Click on required check box
- Click on Next >> Next >> Save and new.

To create another fields in an object:

9. Go to setup >> click on Object Manager >> type object name(Task) in search bar >> click on the object.
9. Now click on “Fields & Relationships” >> New
9. Select Data type as a “Number” and Click on Next

9. Fill the Above as following:

- Field Label: Number of People Served
- Field Name: Number\_of\_People\_Served
- Click on required check box
- Click on Next >> Next >> Save and new.

To create another fields in an object:

13. Go to setup >> click on Object Manager >> type object name(Task) in search bar >> click on the object.

13. Now click on “Fields & Relationships” >> New

13. Select Data type as a “Text” and Click on Next

13. Fill the Above as following:

- Field Label: Name of the Person
- Field Name: Name\_of\_the\_Person
- Click on Next >> Next >> Save and new.

To create another fields in an object:

17. Go to setup>> click on Object Manager >> type object name(Task) in search bar >> click on the object.
17. Now click on “Fields & Relationships” >> New
17. Select Data type as a “Phone” and Click on Next

17. Fill the Above as following:

- Field Label: Phone
- Field Name: Phone
- Click on Next >> Next>> Save and new.

To create another fields in an object:

21. Go to setup >> click on Object Manager >> type object name(Task) in search bar >> click on the object.
21. Now click on “Fields & Relationships” >> New
21. Select Data type as a “Pick List” and Click on Next
- 1.
- 2.
- 3
- 4
- 5

The screenshot shows the Salesforce Setup interface for creating a custom field named 'Rating' on the 'Task' object. The 'Field Information' section includes the following details:

- Field Label: Rating
- Field Name: Rating
- API Name: Rating\_c
- Description: (empty)
- Help Text: (empty)
- Data Owner: (empty)
- Field Usage: (empty)
- Data Sensitivity Level: (empty)
- Compliance Categorization: (empty)

On the right side, the 'Object Name' is listed as 'Task' and the 'Data Type' is listed as 'Picklist'. The 'General Options' section contains:

- Required:
- Default Value: null

The 'Picklist Options' section contains:

- Restrict picklist to the values defined in the value set:
- Controlling Field: (empty)
- Max: (empty)

21. Fill the Above as following:
  - Field Label: Rating
  - Field Name: Rating
  - Enter values, with each value separated by a new line:
  - Click on Next >> Next >> Save and new.

To create another fields in an object:

25. Go to setup >> click on Object Manager >> type object name (Task) in search bar >> click on the object.

25. Now click on “Fields & Relationships” >> New
25. Select Data type as a “Long Text Area” and Click on Next
25. Fill the Above as following:
  - Field Label: Feedback
  - Field Name: Feedback
  - Click on Next >> Next >> Save and new.

The screenshot shows the Salesforce Setup interface under the Object Manager. A custom field named 'Feedback' is being created for the 'Task' object. The field is defined as a 'Long Text Area'. Other details shown include Field Label ('Feedback'), Field Name ('Feedback'), API Name ('Feedback\_\_c'), and Data Type ('Long Text Area'). The field has a length of 32,768 characters. The 'Validation Rules' section is visible at the bottom.

## Creation of fields for the Volunteer object

1. Go to setup >> click on Object Manager >> type object name(Volunteer) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “Auto Number” and Click on Next
4. Fill the Above as following:
  - Field Label : Volunteer ID
  - Field Name : gets auto generated
  - Click on required check box
  - Click on Next >> Next >> Save and new.
  - Click on Next >> Next >> Save and new.

The screenshot shows the Salesforce Setup interface. In the top navigation bar, there is a cloud icon, the word "Setup", and tabs for "Home" and "Object Manager". A search bar with the placeholder "Search Setup" is located at the top right. Below the navigation, the path "SETUP > OBJECT MANAGER" and the object name "Volunteer" are displayed. On the left, a sidebar lists various setup categories such as Details, Fields & Relationships (which is currently selected), Page Layouts, Lightning Record Pages, etc. The main content area is titled "Volunteer Custom Field" and "Volunteer ID". It shows the "Custom Field Definition Detail" page with fields like "Field Label" (Volunteer ID), "Field Name" (Volunteer\_ID), "API Name" (Volunteer\_ID\_\_c), and "Data Type" (Auto Number). Other sections include "Field Information", "General Options", and "Auto Number Options". Buttons for "Edit", "Set Field-Level Security", "View Field Accessibility", and "Where is this used?" are available at the top right of the detail page.

To create another fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Volunteer) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “Picklist” and Click on Next
4. Fill the Above as following:
  - Field Label : Gender
  - Field Name : Gender
  - Enter values, with each value separated by a new line :

Female

Male

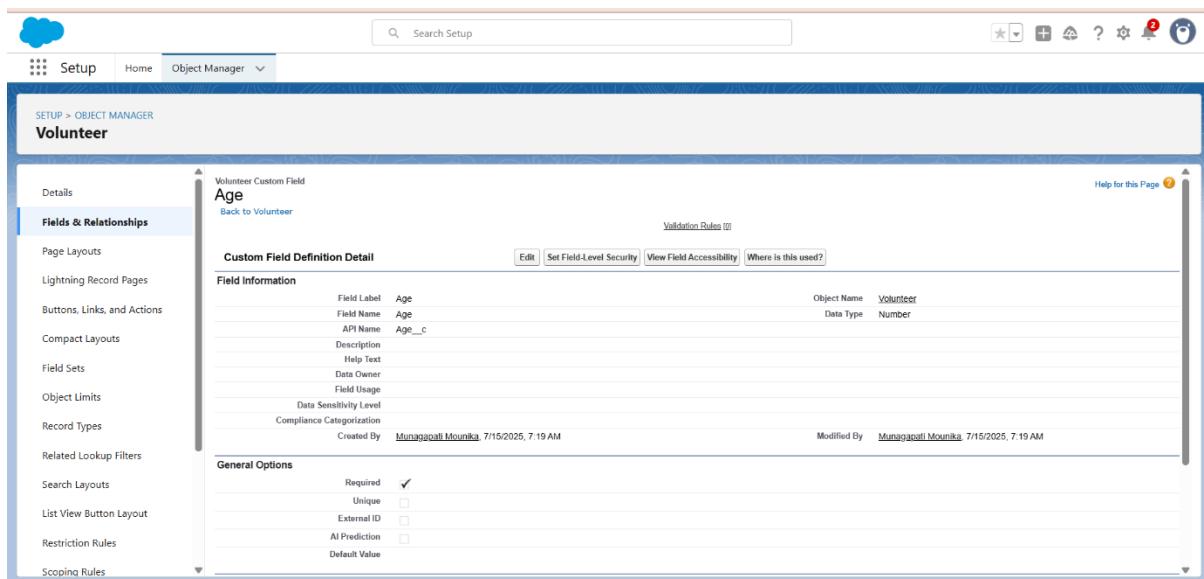
- Click on Next >> Next >> Save and new.

To create another fields in an object:

5. Go to setup >> click on Object Manager >> type object name(Volunteer) in search bar >> click on the object.
6. Now click on “Fields & Relationships” >> New
7. Select Data type as a “Date” and Click on Next
8. Fill the Above as following:
  - Field Label : Available On
  - Field Name : Available On
  - Click on required check box
  - Click on Next >> Next >> Save and new.

To create another fields in an object:

9. Go to setup >> click on Object Manager >> type object name(Volunteer) in search bar >> click on the object.
10. Now click on “Fields & Relationships” >> New
11. Select Data type as a “Number” and Click on Next
12. Fill the Above as following:
  - Field Label : Age
  - Field Name : Age
  - Click on required check box
  - Click on Next >> Next >> Save and new.



To create another fields in an object:

13. Go to setup >> click on Object Manager >> type object name(Volunteer) in search bar >> click on the object.
14. Now click on “Fields & Relationships” >> New
15. Select Data type as a “Email” and Click on Next
16. Fill the Above as following:
  - Field Label : Email

- Field Name : Email
- Click on required check box
- Click on Next>> Next >> Save and new.

The screenshot shows the Salesforce Object Manager interface for creating a custom field. The object being edited is 'Volunteer'. The custom field is named 'Email' with the API name 'Email\_c'. The data type is set to 'Email'. The 'Required' option is checked, while 'Unique' is not. Other settings like 'Data Sensitivity Level' and 'Compliance Categorization' are also visible.

To create another fields in an object:

17. Go to setup >> click on Object Manager >> type object name(Volunteer) in search bar >> click on the object.
18. Now click on “Fields & Relationships” >> New
19. Select Data type as a “Number” and Click on Next
20. Fill the Above as following:
  - Field Label : Contact Number
  - Field Name : Contact\_Number
  - Click on required check box
  - Click on Next >> Next >> Save and new.

Volunteer Custom Field  
**Contact Number**

**Custom Field Definition Detail**

**Field Information**

- Field Label: Contact Number
- Field Name: Contact\_Number
- API Name: Contact\_Number\_c
- Description:
- Help Text:
- Data Owner:
- Field Usage:
- Data Sensitivity Level:
- Compliance Categorization:

Created By: Munagapati.Mounika 7/15/2025, 7:20 AM Modified By: Munagapati.Mounika 7/15/2025, 7:20 AM

**General Options**

- Required:
- Unique:
- External ID:
- AI Prediction:
- Default Value:

To create another fields in an object:

21. Go to setup >> click on Object Manager >> type object name(Volunteer) in search bar >> click on the object.
22. Now click on “Fields & Relationships” >> New
23. Select Data type as a “Text Area (Long)” and Click on Next
24. Fill the Above as following:
  - Field Label : Address
  - Field Name : Address
  - Click on Next >> Next >> Save and new.

Volunteer Custom Field  
**Address**

**Custom Field Definition Detail**

**Field Information**

- Field Label: Address
- Field Name: Address
- API Name: Address\_\_c
- Description:
- Help Text:
- Data Owner:
- Field Usage:
- Data Sensitivity Level:
- Compliance Categorization:

Created By: Munagapati.Mounika 7/15/2025, 7:21 AM Modified By: Munagapati.Mounika 7/15/2025, 7:21 AM

**General Options**

Default Value:

**Long Text Area Options**

# Visible Lines: 3 Length: 32,768

**Validation Rules**

To create another fields in an object:

25. Go to setup >> click on Object Manager >> type object name(Volunteer) in search bar >> click on the object.
26. Now click on “Fields & Relationships” >> New
27. Select Data type as a “Date” and Click on Next
28. Fill the Above as following:
  - Field Label : Date of Birth
  - Field Name : Date\_of\_Birth
  - Click on Next >> Next >> Save and new.

The screenshot shows the Salesforce Object Manager interface for the 'Volunteer' object. The 'Fields & Relationships' tab is active. A custom field named 'Date of Birth' is selected. The 'Field Information' section displays the following details:

- Field Label: Date of Birth
- Field Name: Date\_of\_Birth
- API Name: Date\_of\_Birth\_c
- Description: (empty)
- Help Text: (empty)
- Data Owner: (empty)
- Field Usage: (empty)
- Data Sensitivity Level: (empty)
- Compliance Categorization: (empty)
- Created By: Munagapati Mounika, 7/15/2025, 7:22 AM
- Modified By: Munagapati Mounika, 7/15/2025, 7:22 AM

The 'General Options' section includes a 'Required' checkbox which is checked. The 'Validation Rules' section indicates 'No validation rules defined.'

### Creation of fields for the Execution Details object

1. Go to setup >> click on Object Manager >> type object name(Volunteer) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “Auto Number” and Click on Next
4. Fill the Above as following:
  - Field Label : Execution ID
  - Field Name : gets auto generated
  - Click on required check box

- Click on Next >> Next >> Save and new.

The screenshot shows the Salesforce Setup interface with the following details:

- Object Manager:** Execution Detail
- Custom Field Definition Detail:**
  - Field Information:** Field Label: Execution ID, Field Name: Execution\_ID, API Name: Execution\_ID\_c, Data Type: Auto Number.
  - General Options:** External ID:
  - Auto Number Options:** Display Format: ExecutionID-{0}
- Details:** Object Name: Execution\_Detail, Data Type: Auto Number.
- Created By:** Munagapati Mounika, **Modified By:** Munagapati Mounika, **Created Date:** 7/15/2025, 7:24 AM, **Modified Date:** 7/15/2025, 7:24 AM.

## Create Flow to create a record in Venue object

- Go to setup >> type Flow in quick find box >> Click on the Flow and Select the New Flow.
- Select the Screen flow. Click on create.
- Click on the '+' icon in between start and end and click on screen element.
- Under the Screen Properties:
  - Label: Venue Details
  - API Name: Venue\_Details

The screenshot shows the Salesforce Flow Builder interface with the following details:

- Flow Builder:** Venue Form - V1
- Edit Screen:** Edit Screen
- Components:** Input (53) including Action Button, Add Attendees, Address, Call Script, Cancel Appointment, Checkbox, Checkbox Group, etc.
- Venue Form:** A form with fields for Venue Name, Email, and Phone.
- Screen Properties:**
  - Properties:**
    - Label: Venue Details
    - API Name: Venue\_Details
  - Description: (empty)
  - Stage: Select a stage resource...
- Buttons:** Cancel, Done

5. Now let's add components in this flow. Click on Text Component and name it as:

Label: Venue Name

API Name: Venue\_Name

6. Click on Email Component and name it as:

Label: Email

API Name: Contact\_Email

7. Click on Phone Component and name it as:

Label: Phone

API Name: Contact\_Phone

8. Click on Text Component and name it as:

Label: Venue Location

API Name: Venue\_Location

9. Click on Number Component and name it as:

Label: Latitude

API Name: Latitude

10. Click on Number Component and name it as:

Label: longitude

API Name: longitude

11. Next click on Done. This would like below

12. Click on the '+' icon in between Venue details and end and click on create record element.

12. 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}

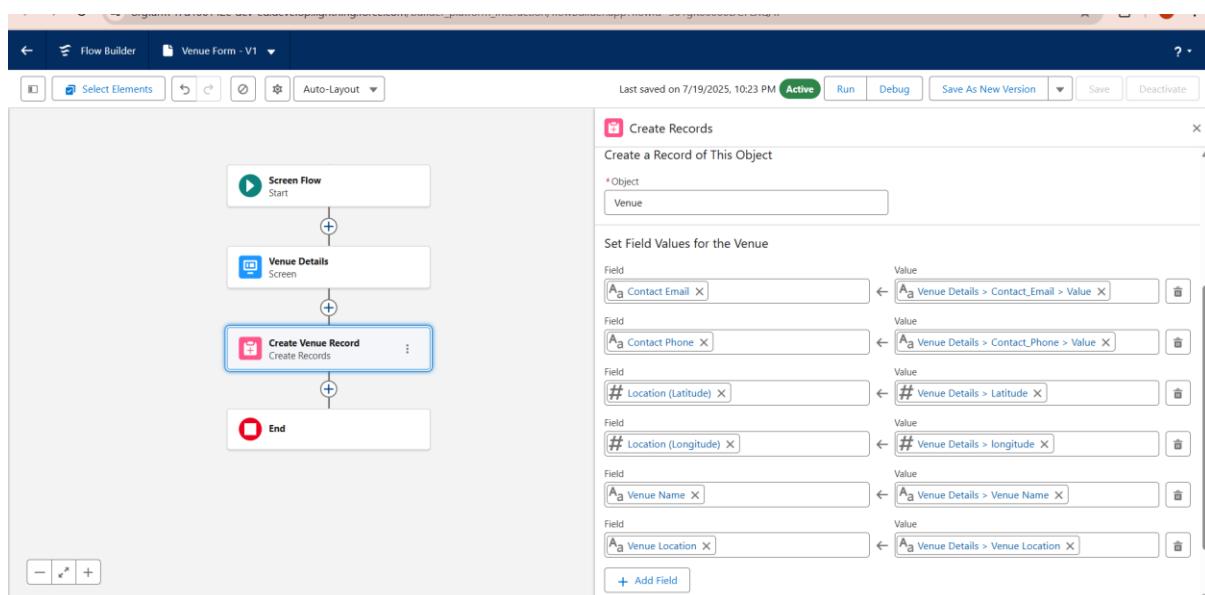
Field: Value = Contact\_Phone\_\_c : {! Contact\_Phone.value}

Field: Value = Name: {! Venue Name}

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

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

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



14. This would look like:

15. Click on Save as:

Flow Label: Venue Form

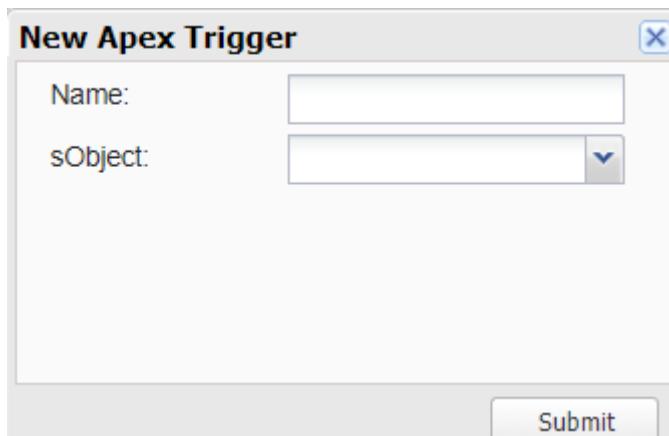
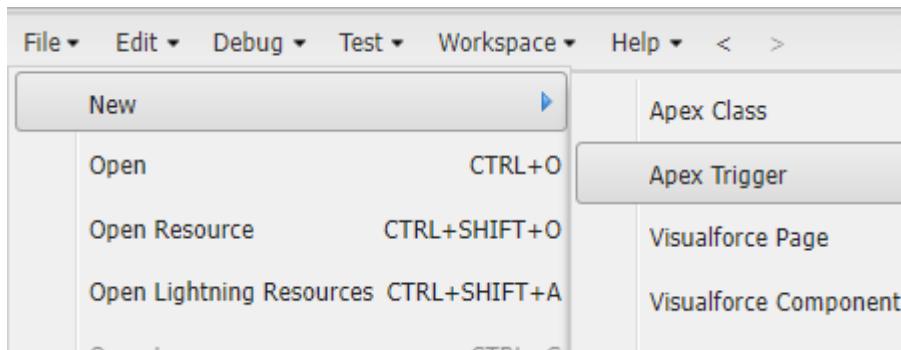
Flow API Name: Venue\_Form

### Apex Classes, Triggers, Asynchronous Apex:

Custom Apex triggers were developed to automatically update stock status and generate distribution records when a pickup is marked completed. Additionally, asynchronous Apex (Batch Apex) was implemented to periodically send summary emails to donors and NGOs highlighting food saved and beneficiaries served over the week. This ensures timely communication and promotes continued engagement without impacting real-time system performance.

## Create a Trigger

1. Log into the trailhead account, navigate to the gear icon in the top right corner.
2. Click on developer console and you will be navigated to a new console window.
3. Click on the File menu in the toolbar and click on new >> Trigger.
4. Enter the trigger name and the object to be triggered.



5. Enter Name: DropOffTriggers
6. Object: Drop-Off Point
6. Click on Submit.

## Trigger Code

(This Trigger is to assign Distance field to the Distance Calculation field. So that we can assign the distance in the sharing rules.)

### 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;
    }
}

```

The screenshot shows the Salesforce Developer Console interface. The title bar says "Developer Console - Google Chrome". The URL is "orgfarm-f7d106142c-dev-ed.develop.my.salesforce.com/\_ui/common/apex/debug/ApexCSIPage". The tabs at the top include File, Edit, Debug, Test, Workspace, Help, and a dropdown for API Version (set to 64). The main area displays the Apex trigger 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;
    }
}

```

Below the code editor is a toolbar with tabs for Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems. The Logs tab is selected. The log table has columns for User, Application, Operation, Time, Status, Read, and Size. There are no entries in the log table.

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

### Create a Lightning App

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. Fill the app name in app details and branding as follow App

Name:FoodConnect

Developer\_Name: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.

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.

## 6. To Add User Profiles:

Search profiles (System administrator) in the search bar >> click on the arrow button >> save & finish.

## **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

3. Open Custom Reports and click on New Report
3. Select Report Type: Venue with DropOff with Volunteer

3. Then click on Start Report.
  3. In GROUP ROWS: Add Volunteer Name
  3. In Columns: Add Venue Name, Drop-Off point Name, Distance.

REPORT ▾

venue and Drop Off point ✓ Venue with DropOff with Volunteer

Fields

☰ Outline Filters

Groups

GROUP ROWS

Add group...

Volunteer Name

X

GROUP COLUMNS

Add group...

Columns

Add column...

Venue Name

X

Drop-off point Name

X

# Distance

X

Previews a limited number of records. Run the report to see everything.

Update Preview Automatically

Volunteer Name	Venue Name	Drop-off point Name	Distance
- (3)	Le Royale banquet Hall.	Jeedimetla	6,902.9950
	Rajampet	-	-
	Ujwala Grand	-	-
	<b>Subtotal</b>		6,902.9950
Bhavika (1)	Paradise Garden Function Hall	Suraram Village	28.2332
	<b>Subtotal</b>		28.2332
Charan (1)	La Royale Banquet Hall	Shapur	5.1161
	<b>Subtotal</b>		5.1161
	<b>Total (5)</b>		6,938.3443

Row Counts  Detail Rows  Subtotals  Grand Total  Conditional Formatting

↶ ↻ Add Chart Save & Run Save Close Run

8. Now click on Save & Run.
  8. Give Label as:
  8. Report Name: venue and Drop Off point
  8. Report Unique Name: Auto Populated
  8. 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.

The screenshot shows the FoodConnect application's reporting feature. A report titled "Tasks with Execution Details and Volunteers" is displayed. The report preview shows the following data:

Volunteer: Volunteer ID	Volunteer: Volunteer Name	Task: Task Name	Execution Detail: Execution Detail Name
VOLUNTEER-1 (1)	Charan	Task 2	Execution 2
VOLUNTEER-2 (1)	Bhavika	Task 1	Execution 1
Total (2)			

The left sidebar shows report configuration options for Groups and Columns. The top navigation bar includes links for Home, Venues, Drop-Off Points, Tasks, Volunteers, Execution Details, Reports, and a search bar.

7. Now click on Save & Run.

7. Give Label as:

Report Name: Volunteer Task

Report Unique Name: Auto Populated

1. Click on Select Folder and select Custom Report, then click on Save.

### **Adding venue and Drop Off point Report to the Dashboard**

1. Go to the app (Food Connect) >> click on the Dashboards tab.

2. Click on New Folder.

Folder Label: Custom Dashboards

Folder Unique Name: Auto Populated

3. Open Custom Dashboards and click on New Dashboards

3. Name: Organization Details

3. Click on Widget and select Chart or Table

3. In Select Report: Select venue and Drop Off point Report.

3. Then click on select

3. In Add Component:

Display As: Select Lightning Table

Component Theme: Select Dark (Optional)

The screenshot shows the FoodConnect application interface with a 'Task Execution Details' dashboard. A modal window titled 'Edit Widget' is open, showing a preview of a report titled 'venue and Drop Off point'. The report table lists venues and their drop-off points with distances:

Venue Name	Drop-Off point Name	Distance
La Royale Banquet Hall	Shapur	5.1161
Le Royale banquet Hall.	Jeedimetta	6.9030k
Paradise Garden Function Hall	Suraram Village	28.2332
Rajampet	-	-
Ujwala Grand	-	-

The 'Edit Widget' dialog also includes sections for 'Report', 'Display As' (set to Table), 'Groups', 'Columns', and 'Preview'.

1. Now click on save.

## Adding Volunteer Task Report to the Dashboard

1. Click on Widget and select Chart or Table
2. In Select Report: Select Volunteer Task Report.
3. Then click on select
4. In Add Component:

Display As: Select Line Chart

Component Theme: Select Dark (Optional)

The screenshot shows the FoodConnect application interface with a 'Task Execution Details' dashboard. A modal window titled 'Edit Widget' is open, showing a preview of a report titled 'Volunteer Task'. The report displays a line chart with 'Record Count' on the Y-axis (0, 0.5, 1) and 'Volunteer: Volunteer ID' on the X-axis (VOLUNTEER-1, VOLUNTEER-2). The chart shows a single data point at (VOLUNTEER-1, 1) and another at (VOLUNTEER-2, 1).

The 'Edit Widget' dialog includes sections for 'Report' (set to 'Volunteer Task'), 'Display As' (set to Line Chart), 'X-Axis' (set to 'Volunteer: Volunteer ID'), 'Y-Axis' (set to 'Record Count'), and 'Preview'.

1. Now click on save.

## Phase 4: Data Migration, Testing & Security:

### Data Loading Process:

Historical data of partner restaurants, NGOs, and past donation records was migrated into Salesforce using the Data Loader, chosen for its capability to handle bulk records and ensure referential integrity across custom objects like Food Donation, Pickup Schedule, and Distribution Record. This ensured a smooth transition from manual tracking systems to the new automated CRM.

### Field History Tracking, Duplicate Rules, Matching Rules:

Field History Tracking was enabled on key objects such as Food Donation and Pickup Schedule to maintain an audit trail of critical changes like pickup time adjustments or status updates. Duplicate Rules and Matching Rules were configured on the Donor and Recipient records to prevent redundant entries and ensure that every partner is uniquely identified, thereby preserving data quality.

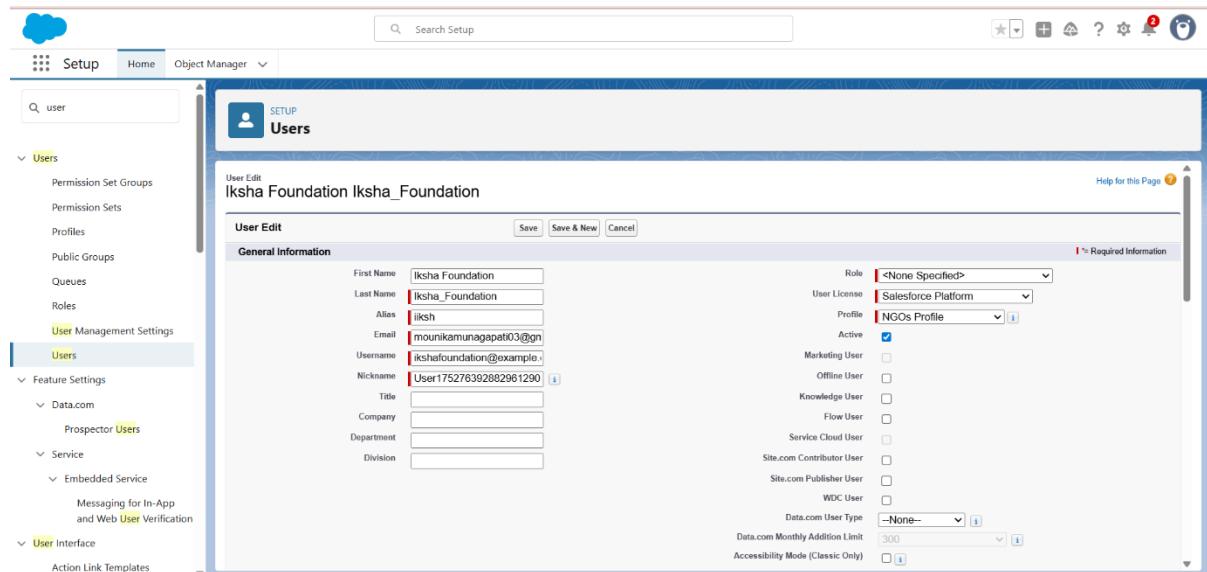
### Profiles, Roles, Role Hierarchy, Permission Sets, Sharing Rules:

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

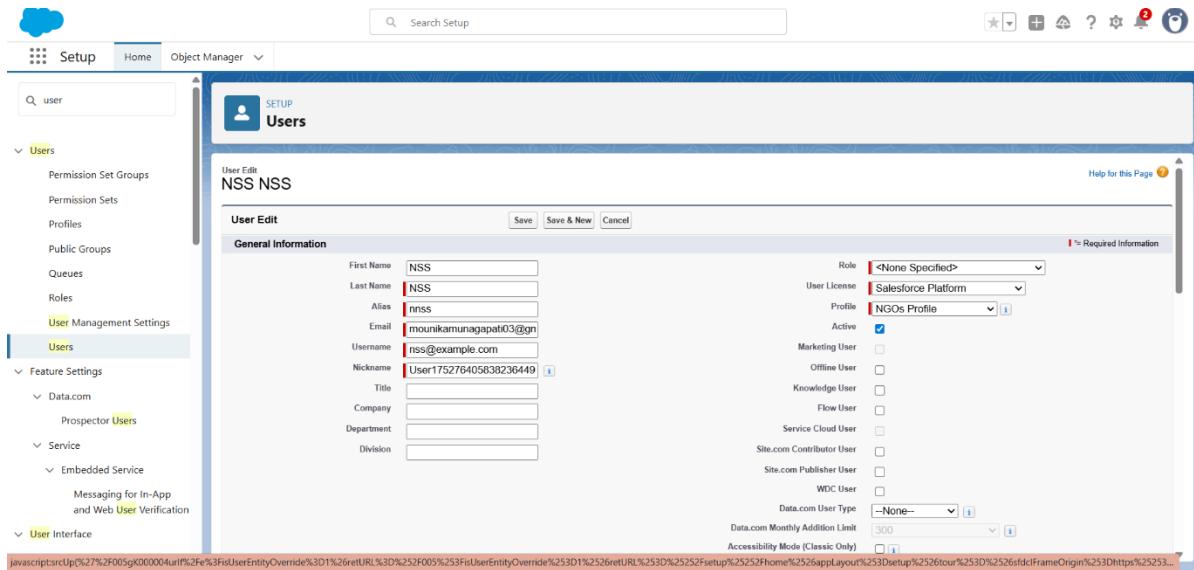
1. Go to setup page >> type Profiles in Quick Find bar >> click on Profiles >> click on 'S'

2. Click on Clone beside Standard Platform User.
  3. Under Clone Profile:  
Profile Name: NGOs Profile
  4. Then click on Save
- 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
  4. Then click on Save

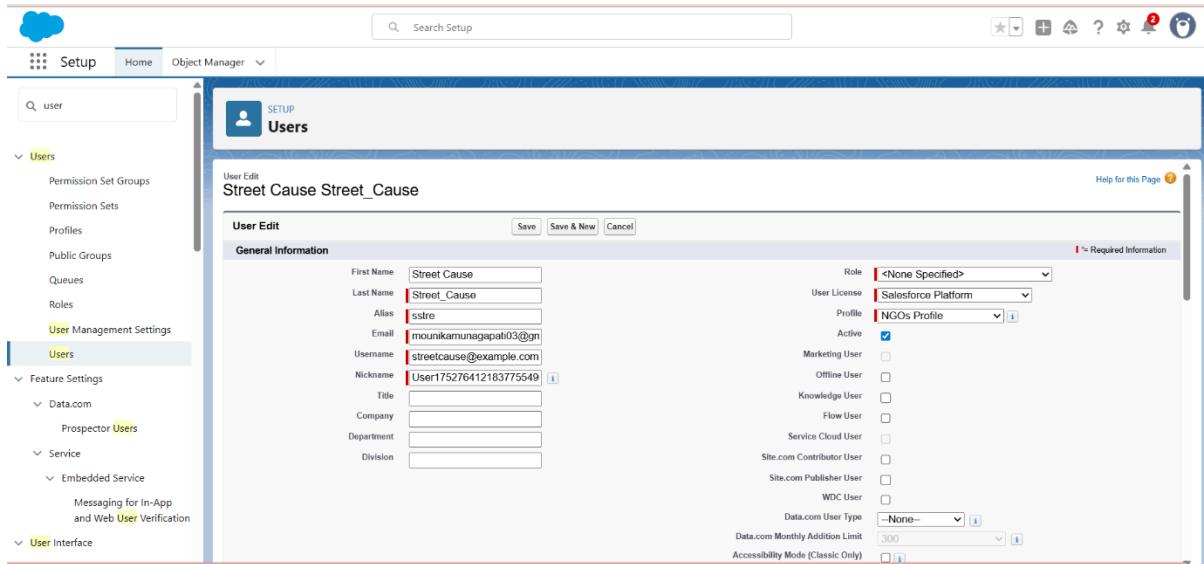


## Creation of User2, User3

1. Create another Two Users by following steps in Activity - 1 with similar User License and Profile.



## 2. Give Different First Name, Last Name based on Different NGO's.



## 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
4. Select your rule type: Select Based on criteria.

4. Select which records to be shared:  
Field: Operator: Value = Distance: less than: 15
6. Select the users to share with: Near Share With  
Public Groups: Iksha
7. Click on Save.
7. Click on new near Drop-Off point Sharing Rules and Name it as:  
Label: Rule 2  
Rule Name: Rule\_2
9. Select your rule type: Select Based on criteria.
9. Select which records to be shared:  
Field: Operator: Value = Distance: greater than: 15  
Field: Operator: Value = Distance: less or equal: 30
11. Select the users to share with: Near Share With  
Public Groups: NSS
12. Click on Save.
12. Click on new near Drop-Off point Sharing Rules and Name it as:  
Label: Rule 3  
Rule Name: Rule\_3
14. Select your rule type: Select Based on criteria.
14. Select which records to be shared:  
Field: Operator: Value = Distance: greater than: 30  
Field: Operator: Value = Distance: less or equal: 50
16. Select the users to share with: Near Share With  
Public Groups: Street Cause
17. Click on Save.

The screenshot shows the 'Sharing Settings' page in Salesforce. It displays several sections for defining sharing rules:

- Work Step Template Sharing Rules**: No sharing rules specified.
- Work Type Sharing Rules**: No sharing rules specified.
- Work Type Group Sharing Rules**: No sharing rules specified.
- Drop-Off Point Sharing Rules**:
 

Action	Criteria	Shared With	Access Level
Edit   Del	Drop-Off Point: Distance LESS OR EQUAL 15	Group: Iksha	ReadWrite
Edit   Del	(Drop-Off Point: Distance GREATER THAN 15) AND (Drop-Off Point: Distance LESS OR EQUAL 30)	Group: NSS	ReadWrite
Edit   Del	(Drop-Off Point: Distance GREATER THAN 30) AND (Drop-Off Point: Distance LESS OR EQUAL 50)	Group: Street Cause	ReadWrite
- Task Sharing Rules**: No sharing rules specified.
- Venue Sharing Rules**: No sharing rules specified.

## 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

The screenshot shows a venue record in the FOODCONNECT CRM. The top navigation bar includes links for Home, Venues, Drop-Off Points, Tasks, Volunteers, Execution Details, Reports, and Task Execution Details. The main view displays the following information:

- Venue Name:** Paradise Garden Function Hall.
- Contact Email:** abcd12@gmail.com
- Contact Phone:** (965) 874-1232
- Location:** Location field (with edit icon).
- Venue Location:** Location field (with edit icon).
- Created By:** Munagapati Mounika (Last modified by Munagapati Mounika on 7/19/2025, 7:59 AM).
- Owner:** Munagapati Mounika.
- Activity:** A feed showing no activities to show, with options to refresh, expand all, or view all.
- Upcoming & Overdue:** A section indicating no past activity.

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

The screenshot shows the FoodConnect application interface. The top navigation bar includes links for Home, Venues, Drop-Off Points, Tasks, Volunteers, Execution Details, Reports, and Task Execution Details. A search bar and various system icons are also present. The main content area displays a 'Drop-Off Point' record for 'Suraram Village'. The 'Details' tab is selected, showing fields such as Drop-Off point Name (Suraram Village), Owner (Munagapati Mounika), Venue (Paradise Garden Function Hall), Location (Location 2), State (Arunachal Pradesh), Distance (28.2332), and Created By (Munagapati Mounika, 7/19/2025, 8:30 AM). The 'Activity' section on the right shows no upcoming or overdue activities.

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

The screenshot shows the FoodConnect application interface, similar to the previous one but with a different record type. The top navigation bar and search bar are identical. The main content area displays a 'Task' record titled 'Task 1'. The 'Details' tab is selected, showing fields such as Task Name (Task 1), Owner (Munagapati Mounika), Sponsored By (Paradise Garden Function Hall), Drop-Off point (Suraram Village), Task ID (TASK-1), Date (7/20/2025), Food Category (Veg/Non-Veg), Number of People Served (10), Name of the Person, Phone, and Rating. The 'Activity' section on the right shows no upcoming or overdue activities.

**FoodConnect**

Volunteer Bhavika

**Details**

Volunteer Name	Bhavika
Drop_Off_point	Suraram Village
Volunteer ID	VOLUNTEER-2
Gender	Female
Available On	7/20/2025
Age	20
Email	bhavika@gmail.com
Contact Number	9,478,623,014
Address	
Date of Birth	

**Activity**

No activities to show.

Get started by sending an email, scheduling a task, and more.

No past activity. Past meetings and tasks marked as done show up here.

**FoodConnect**

Execution Detail Execution 1

**Details**

Execution Detail Name	Execution 1
Volunteer	Bhavika
Task	Task 1
Execution ID	ExecutionID-1
Created By	Munagapati Mounika, 7/19/2025, 10:21 AM
Last Modified By	Munagapati Mounika, 7/19/2025, 10:21 AM

**Activity**

No activities to show.

Get started by sending an email, scheduling a task, and more.

No past activity. Past meetings and tasks marked as done show up here.



FoodConnect

Home Venues

Drop-Off Points

Tasks

Volunteers

Execution Details

Reports

Search...

Report: Venue with DropOff with Volunteer  
venue and Drop Off pointTotal Records 5 Total Distance  
6,936.3443

Volunteer Name	Venue Name	Drop-Off point Name	Distance
□ (3)	Le Royale banquet Hall	Jeedimetta	6,902.9950
	Ujwala Grand	-	-
	Rajampet	-	-
<b>Subtotal</b>			6,902.9950
□ Bhavika (1)	Paradise Garden Function Hall	Suraram Village	28.2332
<b>Subtotal</b>			28.2332
□ Charan (1)	La Royale Banquet Hall	Shapur	5.1161
<b>Subtotal</b>			5.1161
<b>Total (5)</b>			6,936.3443

Row Counts  Detail Rows  Subtotals  Grand Total 

FoodConnect

Home Venues

Drop-Off Points

Tasks

Volunteers

Execution Details

Reports

Search...

Report: Tasks with Execution Details and Volunteers  
Volunteer TaskTotal Records  
2

Volunteer: Volunteer ID	Volunteer: Volunteer Name	Task: Task Name	Execution Detail: Execution Detail Name
□ VOLUNTEER-1 (1)	Charan	Task 2	Execution 2
<b>Subtotal</b>			
□ VOLUNTEER-2 (1)	Bhavika	Task 1	Execution 1
<b>Subtotal</b>			
<b>Total (2)</b>			

Row Counts  Detail Rows  Subtotals  Grand Total

The screenshot shows the Food Connect platform's dashboard. On the left, there is a section titled "Task Execution Details" with a table of venue and drop-off point data. The table includes columns for "Venue Name", "Drop-Off point...", and "Di...". Data rows include "La Royale Banquet Hall, Shapur 5.116", "Le Royale banquet Hall, Jeedimetla 6.903", "Paradise Garden Function Hall, Suraram Village 28.22", and "Rajampet - -". Below this is a "Volunteer Task" section with a chart showing record counts for "VOLUNTEER-1" and "VOLUNTEER-2", both at 1. A photograph of food items (red bell pepper, yellow pear, orange egg, green bean) is displayed next to the chart. On the right, there is a "Venue Form" section with fields for "Venue Name", "Email", "Phone", "Venue Location", "Latitude", and "Longitude", each with an input field and a "Next" button.

## Troubleshooting Approach:

During the development of the *Food Connect* platform using Salesforce, a structured troubleshooting process was followed to ensure system reliability and user satisfaction. Issues were identified through continuous testing, user feedback, and monitoring tools such as debug logs and error notifications. Each problem was categorized as configuration-related, code-related, data-related, UI/UX, or performance-based. Root cause analysis involved reviewing system logs, user permissions, and automation flows to isolate errors. Solutions included fixing validation rules, updating Apex logic, correcting data, optimizing queries, and adjusting user interface settings. Problems were tested and resolved in sandbox environments before deployment. All issues and resolutions were documented, and preventive measures like validation rules, alerts, and enhanced test coverage were implemented to avoid recurrence and maintain platform stability.

## Conclusion:

The *Food Connect* project successfully demonstrates how technology can be leveraged to address critical social challenges such as food wastage and hunger. By building the platform on Salesforce, we were able to utilize powerful tools for data management, automation, and user access control to create an efficient and scalable system. The application enables seamless coordination between food donors and volunteers, ensuring that surplus food is redirected to those in need in a timely and organized manner. Through the use of real-time tracking, notifications, and user-friendly interfaces, the platform enhances transparency, encourages community participation, and promotes social responsibility. The project not only meets its functional and technical objectives but also creates a foundation for future enhancements such as mobile access, AI-driven matching, and integration with larger relief networks. Overall, *Food Connect* stands as a meaningful, technology-driven solution that makes a real-world impact.

