Rescue Recipes: Turning Surplus into Smiles

By

Yasaswini Poluri

21501a05e1@pvpsit.ac.in

Project Abstract

In this Salesforce project, I developed an application titled **"To SupplyLeftover Food to the Poor,"** which aims to streamline the process of distributing surplus food to those in need. The application is built with a focus on efficient logistics and volunteer coordination. Several custom objects have been created to facilitate this process:

1. **Venue Object:** This object storesinformation about locations where leftover food is available for collection.

2. **Drop-off Point Object:** This object identifies specific locations where volunteers can deliver the collected food for distribution to the poor.

3. **Volunteer Object:** This object contains details about the individuals responsible for transporting the food from venues to drop-off points.

4. **Task Object:** This object manages all assignments givento each volunteer, ensuring clear communication and accountability.

5. **Execution Detail Object:** This object records information about the completed tasks, including feedbackand ratings, to monitor performance and improve processes.

The application features multiple tabs, such as Venue, Drop-off Point, Task, Volunteer, Execution Details, Reports, and Dashboards. These tabs provide comprehensive insights and analytics through dynamic reports and dashboards. The homepage is designed usingthe Lightning App Builder, offeringan intuitive and user-friendly interface for managing the entire food supply chain. This project demonstrates the potential of leveraging Salesforce's capabilities to address food insecurity through organized and effective resource distribution

INDEX Page

Introduction

The "To Supply Leftover Food to the Poor" application is an innovative solution aimed at bridging the gap between food surplus and those in need. Built on the Salesforce platform, this app helps communities efficiently collect and distribute leftover food to fight hunger and reduce waste. At its core, the application coordinates the efforts of volunteers who transport food from various venues to drop-off points whereit can reach those who need it most.

Through a series of intuitive features, such as tracking collection venues, managing volunteer tasks, and monitoring food deliveries, the app providesa seamless way to ensure that no food goes to waste. With easy-to-use dashboards and real-time reports, volunteers and coordinators can quickly assess and improve their operations. This project showcases how technology can be harnessed to create a positive social impact, turning a simple idea into a practical tool for change.

Task 1: Object Creation
Venue:

- Go to Object Manager> Create > Custom Object.

- Label: Venue, Plural: Venues.

- Record Name: Venue Name, Data Type: Text.

- Enable Reports,Field History Tracking,Activities, Search.

- Click Save.

Drop-Off Point:

- Go to Object Manager> Create > Custom Object.

- Label: Drop-Off Point,Plural: Drop-Off Points.

- Record Name: Drop-Off Point Name, Data Type: Text.

- Enable Reports,Field History Tracking,Activities, Search.

- Click Save.

Task:
- Go to Object Manager> Create > Custom Object.

- Label: Task, Plural: Tasks.

Record Name: Task Name, Data Type: Text

- Enable Reports, Field History Tracking,Activities, Search.
- Click Save.

Volunteer:
- Go to Object Manager> Create > Custom Object.
- Label: Volunteer, Plural: Volunteers.
- Record Name: Volunteer Name, Data Type: Text.
- Enable Reports,Field History Tracking,Activities, Search.
- Click Save.

Execution Detail:
- Go to Object Manager> Create > Custom Object.
- Label: ExecutionDetail, Plural: ExecutionDetails.
- Record Name: ExecutionDetail Name, Data Type: Text.
- Enable Reports,Field History Tracking,Activities, Search.
- Click Save.

Task 2: Creating multiple tabs
To create a Tab (Venue):
- Go to Setup and search for Tabs.
- Click New under Custom ObjectTabs.
- Choose Venue, pick a tab style, and click Next.

- Keep the default settings for profiles and click Next.
- Uncheck the box for adding to Custom App, but check Append tab to users' existing personal customizations.
- Click Save.

For other tabs:

Followed the same steps to create tabs for Drop-Off Point, Task, Volunteer, and Execution Detail.

Task 3: The lightening app
To create a Lightning App Page:

Go to the Setup page and search for "App Manager" in the Quick Find bar. Select "App Manager" and click New Lightning App.

Fill in the app details:

- App Name: FoodConnect

- Developer Name: Auto-populated
- Image: Optional (add if desired)
- Primary Color Hex Value: Default
- Click Next:

On the App Options page, set Navigation Style to StandardNavigation and click Next.

On the UtilityItems page, keep the defaults and click Next. Add Navigation Items:

- Search and add the following items: Home, Venue, Drop-Off Point, Task, Volunteer, Execution Details, Reports. Use the ar button to move them and click Next.

Add User Profiles:

- Search for System Administrator in the profiles search bar, add it using the ar button, and click Save & Finish.

Task 4: Creating Fields
Creation of Relationship fields in objects:

Creation of LookupRelationship Field on Volunteer Object:

- Go to Setup > Object Manager.
- Search for and selectthe Volunteer object.
- Click Fields & Relationships > New.
- Choose Lookup Relationship and select Drop-OffPoint as the related object.
- Click Next.
- Set Field Name to Drop_Off_point and let the Field Labelauto-generate.
- Click Next > Next > Save.

Creation ofMaster-Detail Relationship Field on Execution Details Object:

- Go to Setup > Object Manager.
- Search for and select the Execution Details object.
- Click Fields & Relationships > New.
- Choose Master-Detail Relationship and select Volunteer as the relatedobject.
- Click Next.
- Set Field Name to Volunteer and let the Field Labelauto-generate.
- Click Next > Next > Save.

Creation ofMaster-Detail Relationship Field on Execution Details Object:

- Go to Setup > Object Manager.
- Search for and select the Execution Details object.
- Click Fields & Relationships > New.
- Choose Master-Detail Relationship and select Task as the related object.
- Click Next.
- Set Field Name to Task and let the Field Label auto-generate.
- Click Next > Next > Save.

Creationof Lookup RelationshipField on Drop-Off Point Object:

- Go to Setup > Object Manager.

- Search for and select the Drop-OffPoint object.
- Click Fields & Relationships > New.
- Choose Lookup Relationship and select Venueas the related object.
- Click Next.
- Set Field Name to Venue and Field Label to Venue__c.
- Click Next > Next > Save.

Creation of Lookup Relationship Field on Task Object:
- Go to Setup > Object Manager.
- Search for and select the Task object.
- Click Fields & Relationships > New.
- Choose Lookup Relationship and select Venueas the related object.
- Click Next.
- Set Field Name to Sponsored By and let the Field Label auto-generate.
- Click Next > Next > Save.

Creation of Lookup Relationship Field on Task Object:
- Go to Setup > Object Manager.
- Search for and select the Task object.
- Click Fields & Relationships > New.
- Choose Lookup Relationship and select Drop-OffPoint as the related object.
- Click Next.
- Set Field Name to Drop-Off Point and let the FieldLabel auto-generate.
- Click Next > Next > Save.

Creation of fields for the Venue object:
- To add fields to the Venue object, start by going to Setup and clicking on Object Manager.Search for "Venue" and select it. Then, navigateto "Fields & Relationships" and click "New."
- First, create an Email field by selecting "Email" as the data type and clicking Next. Fill in the field label as "Contact Email" and the field name as "Contact Email," and check the "Required" box. Click Next twice, then

Save & New to add another field.

- Next, create a Phone field. Choose "Phone" as the data type and click Next. Use "Contact Phone" for both the field label and field name, mark it as required, and proceed with Next, Next, then Save & New.

- Then, add a Geolocation field. Select "Geolocation" as the data type, click Next, and fill in "Location" for the field label and name. Set the decimalplaces to 4, and add a description like "Enter the Geolocation of your Venue." Click through Next, Next, and Save & New.

- Finally, createa Long Text Area field.Select "Long Text Area" as the data type, click Next, and use "Venue Location" for the field label and "Venue_Location" for the field name. Complete the process with Next, Next, and Save & New.

Creation of fields for the Drop-Off point object:

- To create fields for the Drop-Off Point object, go to Setup, click on Object Manager, search for "Drop-Off Point," and select it. Then, navigate to "Fields & Relationships" and click "New."

- First, add a Geolocation field. Choose "Geolocation" as the data type and click Next. Set the field label to "Location 2" and provide a description like "Enter the Geolocation of the Drop-Off Point." Choose "Decimal" for Geolocation Optionsand set DecimalPlaces to 4. Click Next twice, then Save &New.

- Next, create a Formula field. Select "Formula" as the data type, click Next, and use "distance calculation" for the field label and "distance_calculation" for the field name. Set the FormulaReturn Type to "Number" and enter the formula

    DISTANCE(Location_2_c, Venuer.Location_c, 'km'). Proceed with Next, Next, then Save & New.

- Then, add a Picklist field. Choose "Picklist" as the data type, click Next,

and use "State" for both the field label and name. Enter the state names as values, each on a new line. Mark the field as required and click Next twice, then Save & New.

- Finally, for the Task object, repeat the process by searching for "Task" in Object Manager. Add a Number field by selecting "Number" as the data type and clicking Next. Set the field label to "Distance," the field name to "Distance," the length to 14, and decimal places to 4. Make it required, then click Next twice, and finish with Save & New.

Creation of fields for the Task object:

To create various fields in the Task object, start by navigating to Setup and clicking on Object Manager.Search for "Task" and select it. Then, go to "Fields & Relationships" and click "New."

- Begin by adding an Auto Number field. Select "Auto Number" as the data type, click Next, and set the Field Label to "Task ID." Use the display format "TASK-{0}" and start the numbering from 1. Make it required and proceed with Next twice, then Save & New.

- Next, add a Date field. Choose "Date" as the data type, click Next,and set the Field Label to "Date." Mark it as required, then click Next twice, followed by Save & New.

- Then, create a Picklist(Multi-Select) field. Select "Picklist (Multi-Select)" as the data type, click Next, and use "Food Category" for the Field Label and Name. Enter values such as "Veg," "Non-Veg," "Salad," and "Snack," with each value on a new line. Make it required, then click Next twice and Save & New.

- Add a Number field next. Choose "Number" as the data type, click Next, and set the Field Label to "Number of People Served." Mark it as required and proceed with Next twice, then Save & New.

- Next, create a Text field. Select "Text" as the data type, click Next, and set the FieldLabel to "Name of the Person." Click Next twice, then Save & New.

- Add a Phone field.Choose "Phone" as the data type, click Next, and set the

    Field Label to "Phone." Proceedwith Next twice, then Save & New.

- Then, create a Picklistfield for ratings.Select "Picklist" as the data type, click Next, and set the Field Labelto "Rating." Entervalues from 1 to 5, each on a new line. Click Next twice, then Save & New.

- Finally, add a Long Text Area field. Choose "Long Text Area" as the data type, click Next, and set the Field Label to "Feedback." Completethe process with Next twice, and finish with Save & New.

Creation of fields for the Volunteerobject:

To create fields in the Volunteer object, start by going to Setup, clicking on Object Manager, and searching for "Volunteer." Select it and navigate to "Fields & Relationships," then click "New."

- First, create an Auto Number field to uniquely identify each volunteer. Choose "Auto Number" as the data type, click Next, and set the Field Label to "Volunteer ID." Make it required, then click Next twice and Save & New.

- Next, add a Picklist field for gender selection. Select "Picklist" as the data type, click Next, and set the Field Label to "Gender." Enter the options "Female" and "Male," each on a new line.Click Next twiceand Save &New.

- Then create a lookup relation field for VolunteerOwner Name. Select "lookup relation" as the data type, Click Next , and set the related to field as " Users". Enter the field label to "Owner Name" . Click next twice and Save & New.

- Then, create a Date field for availability. Choose "Date" as the data type, click Next, and set the Field Label to "Available On." Mark it as required, then click Next twice and Save & New.

- Add a Number fieldfor age. Select "Number" as the data type, click Next, and set the Field Label to "Age." Make it required, then click Next twice and Save & New.

- Create an Email field to store the volunteer's email address. Choose "Email" as the data type, click Next, and set the Field Label to "Email." Make it required, then click Next twice and Save & New.

- Next, add a Number field for the contact number. Select "Number" as the data type, click Next, and set the Field Labelto "Contact Number." Make it required,then click Next twice and Save & New.

- Then, create a Text Area (Long) field for the address.Choose "Text Area

  (Long)" as the data type, click Next, and set the Field Label to "Address." Click Next twice and Save & New.

- Finally, add another Date field for the date of birth. Select "Date" as the data type, click Next, and set the Field Label to "Date of Birth." Click Next twice and Save & New.

Creation of fields for the ExecutionDetails object:

- To add a new field to the Execution Details object, begin by navigating to Setup, then click on Object Manager. In the search bar, type "Execution Details" to locate and select the object. This action will take you to the object details page, where you can customize various fields and relationships.

- Next, navigate to the "Fields & Relationships"and click on "New" to start creating a new field. Choose the Auto Number data type for this field. The AutoNumber data type is ideal for generating unique identifiers automatically for each execution record, which can help maintainorderly and precise tracking of each task or action.

- In the subsequent step, fill out the field details. Set the Field Label to "Execution ID" to clearly indicate the purpose of this field.The Field Name will be auto-generated based on the label. Ensure that you check the required checkbox, which mandates that each record will have this field filled, ensuring that every execution detail is properly identified and logged in the system.

- Finally, complete the process by clicking through the prompts: Next, Next again, and then Save and New if you wish to continue creating additional fields. This step finalizesthe creation of the Execution ID field, which plays a crucial role in keeping execution records organized and easily accessible for review and reportingpurposes within your Salesforce environment.

Task 5:Creating Flows

Create Flow to create a record in Venue object:

To create a new Flow, go to Setup, type "Flow" in the Quick Find box, and select "New Flow." Choose "Screen Flow" and click "Create."

Add a screen element by clicking the '+' icon, and set the ScreenProperties with the

label"Venue Details" and API Name "Venue_Details." Add components to the screen:

- **Text Component** for "Venue Name" (API Name: "Venue_Name")
- **Email Component** for "Email" (API Name: "Contact_Email")
- **Phone Component** for "Phone" (API Name: "Contact_Phone")
- **Text Component** for "Venue Location" (API Name: "Venue_Location")
- **Number Components** for "Latitude" (API Name: "Latitude") and "Longitude" (API Name: "Longitude")

Click "Done."

Add a "Create Record" element by clickingthe '+' icon. Set the label to "Create Venue Record" (API Name: "Create_Venue_Record"), select "One" record to create, and map fields using values from the screen components:

- Contact_Email__c : {!Contact_Email.value}
- Contact_Phone__c : {!Contact_Phone.value}
- Name : {!Venue_Name}
- Venue_Location__c : {!Venue_Location}
- Location__Latitude__s : {!Latitude}
- Location__Longitude__s : {!Longitude}

Save the flow with the label "Venue Form" and API Name "Venue_Form."

Task 6: Creating Trigger

Create a Trigger:

- Logged into my Trailhead account and clickedon the gear icon in the

top right corner.
- Chose "Developer Console" from the dropdown menu, which opened a new console window.
- In the Developer Console, went to "File" > "New" > "Trigger."
- Named the trigger "DropOffTrigger" and selected "Drop-Off Point" as the sObject.
- Clicked "Submit" to create the trigger.

Trigger Code:

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;
    }
}
```

Task 7: Creating Profiles
- Went to the setup page and typed"Profiles" in the Quick Find bar.
- Clicked on "Profiles" from the search results and then found the profile list.
- Clicked on "S" to find the Standard PlatformUser profile.
- Hit the "Clone" button next to Standard PlatformUser.
- In the Clone Profilesettings, named the new profile"NGOs Profile" and clicked "Save."

Task 8: Creation of Users

Creation of User1:

Navigated to the setup page and typed "Users" in the Quick Find bar.
Clicked on "Users" and then selected "New User."
Filled in the General Information with:

- First Name: Iksha Foundation
- Last Name: Iksha_Foundation
- Alias: iiksh
- Email: (Entered my email address)
- Username: ikshafoundation@sb.com (used a unique username)
- Nickname: Auto populated
- User License: Salesforce Platform
- Profile: NGOs Profile
- Active: Checked

Clicked "Save" to create the user.

Creation of User2, User3:

- Created two more usersby repeating the steps from the previousactivity.
- For each user, used a unique combination of first and last names based on different NGOs.
- Ensured both users had the Salesforce Platform license and were assigned the "NGOs Profile."
- Filled in the detailswith the same email format and different usernames for each new user.
- Checked the "Active" box and clicked"Save" to completethe user creation.

Task9:Creating Public Groups

Creation of Public Group 1:

Went to the setup page and typed "Public Groups" in the Quick Find
bar. Clicked on "Public Groups" and then hit "New" to create a new

group.

Under Group Information, entered the following details:

- Label: Iksha
- Group Name: Iksha
- Checked the box for "Grant Access Using Hierarchies." In the Search , selected "Users."

Added"Iksha Foundation" and "System Administrator" to the SelectedMembers list.

Clicked "Save" to finalize the new publicgroup.

Creation of Public Group 2:

- Created two additional public groups followingthe same steps as before.
- Named each new group based on the other two users from the previous activities.
- For each group, entered a unique label and group name, and checked "Grant Access Using Hierarchies."
- Selected "Users" in the Search and added the respective users to the Selected Members list.
- Clicked "Save" to completethe creation of these publicgroups.

Task 10: Creating Report Types

Creation of ReportTypes:

Headed to the setup page and searched for "Report Types" in the Quick Find bar.
Clicked on "Report Types," then hit "Continue," and selected "New Custom Report Type."

In the "Define the Custom ReportType" , filledout the detailsas follows:

- Primary Object: Venues
- Report Type Label:Venue with DropOffwith Volunteer
- Report Type Name: Venue_with_DropOff_with_Volunteer
- Description: Venue with DropOff with Volunteer
- Store in Category: Other Reports
- Deployment

Status:Deployed
Clicked "Next" to
proceed.

Under"Click to relate another Object," selected "Drop-Off Points" and chose the option "A records may or may not have related B records."

Addedanother related objectby selecting "Volunteers."

Clicked "Save" to complete the creation of the customreport type.

Task 11: Creating Reports

Creation of Report on Venue with Drop-Off Point with Volunteer:

Openedthe FoodConnect app and navigatedto the Reports tab.
Clicked on New Folder and entered:

- Folder Label: Custom Reports
- Folder Unique Name: CustomReports

Opened the Custom Reports folder and clickedon New Report.

Selected the Report Type: Venue with DropOff with Volunteer, then clicked Start Report.
In GROUPS, added Volunteer Name.
In Columns, addedVenue Name, Drop-Offpoint Name, and Distance.
Clicked on Save & Run and entered:

- Report Name: venue and Drop Off point

- Report Unique Name: Auto Populated

Clicked on Select Folder,chose Custom Reports,and clicked Save to finish.

Creation of Report on Tasks with ExecutionDetails and Volunteers:

Openedthe FoodConnect app and navigatedto the Reports tab.
Opened the Custom Reports folder and clicked on New Report.
Selected the Report Type: Volunteers with Execution Detailsand Tasks, then clicked Start Report.

In GROUP S, added
Volunteer ID. In Columns,
added the following:

- Volunteer: VolunteerName
- Task: Task Name
- Execution Detail:Execution Detail Name
- Volunteer: Owner Name
- Task: Date
- Task: Rating

Clicked on Save & Run and entered:

- Report Name: Volunteer Task
- Report Unique Name: Auto Populated

Clicked on Select Folder, chose Custom Reports, and clicked Save to complete
the report setup.

Task 12: Creating Dashboard

Adding venue and Drop Off point Report to the Dashboard:

Openedthe FoodConnect app and navigatedto the Dashboards tab.

Created a New Folder with the following details:

- Folder Label: Custom Dashboards

- Folder Unique Name: Auto Populated

Openedthe Custom Dashboards folder and clickedon New Dashboard.

Entered the Name as Organization Details.

Clicked on Widget and selected Chart or Table .

In Select Report,chose the venue and Drop Off point Report, then clicked Select.

In Add Component, configured the following:

- Display As: Lightning

- Component Theme: Dark (Optional)

Clicked on Save to completethe dashboard setup.

Adding Volunteer Task Report to the Dashboard:

I clicked on Widget and chose Chartor to add a new component.

In Select Report, I selected the Volunteer Task Report and then clicked

Select. In Add Component, I configured it to display as a Line Chart.

I opted for a Dark component theme (this is optional).

Finally, I clicked on Save to complete the process and add the line chart to the dashboard.

Adding a Picture to the Dashboard (Optional):

Clicked on Widget and

chose . Browsed for the

picture to upload.

Afterselecting the picture,saved it as Task

ExecutionDetails. Selected the Custom Dashboards

folder and clicked Save.

Task 13: Apply ing Sharing Rules

Creation of sharing rules:

Navigated to setup and typed SharingSettings in the Quick Find box. Clicked on Sharing Settings.

Scrolled down to find Drop-Off PointSharing Rules and clicked on New next to it. Named the rule:

- Label: Rule 1
- Rule Name: Rule_1

Selected "Based on criteria" as the rule type. Set the criteria:

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

Chose to share with Public Groups and selectedIksha. Clicked Save.

Next, created anotherrule:

- Label: Rule 2
- Rule Name: Rule_2

Selected "Based on criteria" for the rule type. Set the criteria:

- Field: Distance
- Operator: greaterthan
- Value: 15

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

Shared with Public Groups and chose NSS. Clicked

Save. Finally, set up a third rule:

- Label: Rule 3
- Rule Name: Rule_3

Selected "Based on criteria" for the rule type. Set the criteria:

- Field: Distance
- Operator: greaterthan
- Value: 30
- Field: Distance
- Operator: less or equal

- Value: 50

Shared with Public Groups and chose Street Cause. ClickedSave.

Task 14: Creating Home Page

Creation of HomePage:

- Went to setup and typed Lightning App Builder in the QuickFind box. Clicked on Lightning App Builder and selected New.

- Chose Home Page and labeled it as HOME Page. Opted for StandardHome Page.

- In the Components , searched for Flow and dragged it to the right-side . Set the Flow to Venue Flow.

- Next, searched for Dashboard, dragged it to the first on the right side.

- Clicked on Save and Activation, then selected App Default and clicked Add Assignments. Added the FoodConnect App and saved the changes.

The FoodConnect Home Page now looks like this
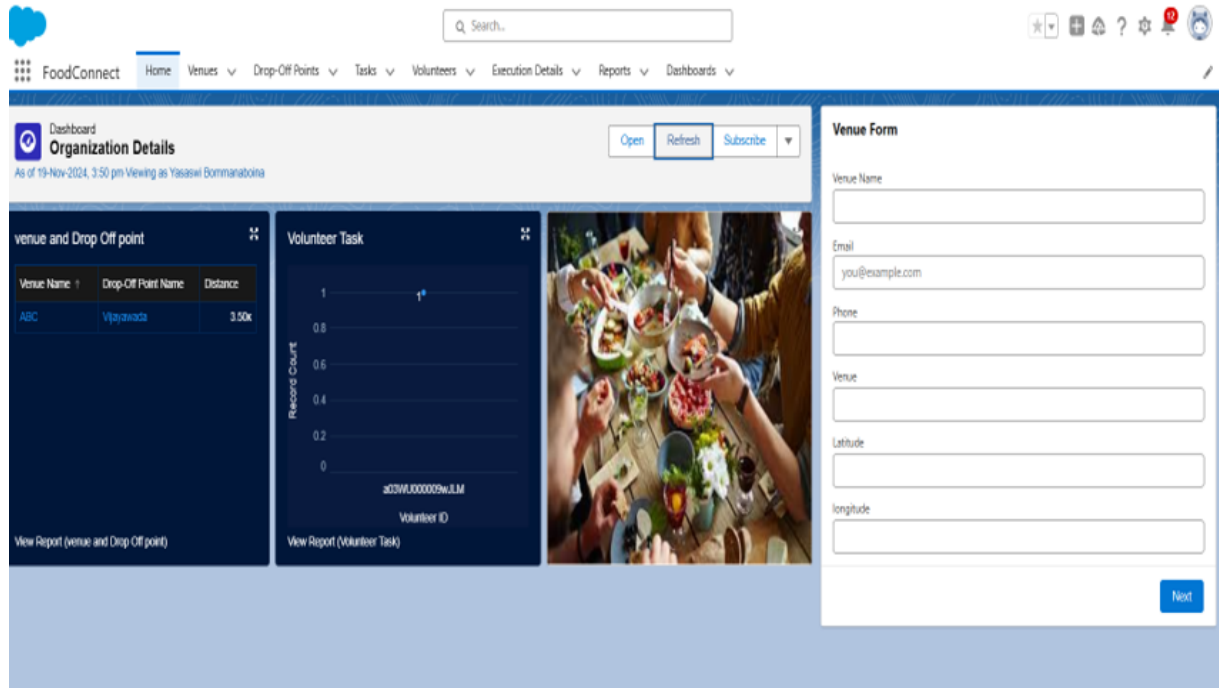
Conclusion:

By leveraging the Salesforce platform, the project successfully established a streamlined and transparent system for managing surplus food donations. It facilitated efficient coordination with volunteers and ensured timely delivery to beneficiaries, effectively addressing food insecurity and maximizing the utilization of available resource.