Project Title: To Supply Leftover Food to Poor

Category: Salesforce

Skills Required: Salesforce Developer

1. Project Overview

1.1 Project Objectives and Goals

Objective: This project aims to develop a Salesforce-based application designed to manage andstreamline the distribution of leftover food to underserved communities. The primary goal is to enhance the efficiency of food drop-offs, coordinate tasks, manage volunteers, and generate insightful reports.

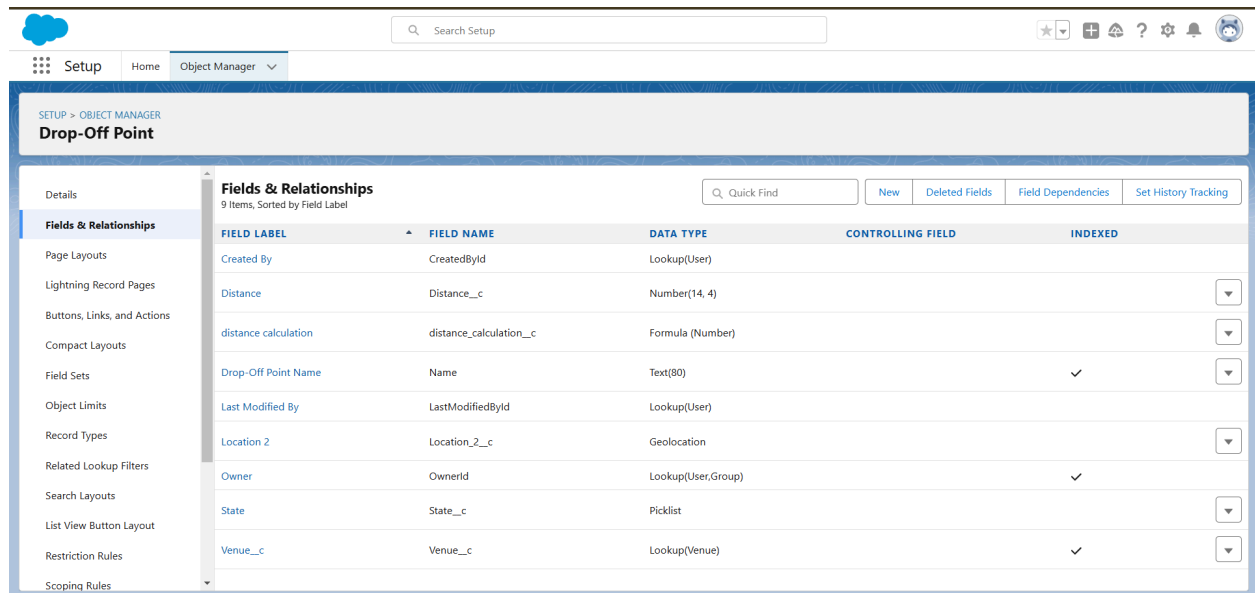2. Detailed Description of Implemented Features

2.1 Custom Objects

1. Drop-Off Point:

■ Purpose: The Drop-Off Point object captures essential details about locations

where food is dropped off. This helps in tracking and managing food distribution

points effectively.

■ Fields:

○ Drop-Off Point Name: A text field representing the name of the drop-off

point.

○ Location Details: A text field for additional details about the drop-off point

location.

○ Distance: A number field that records the distance from the drop-off point

to the delivery area.

○ Venue: A lookup relationship field linking to the Venue object to provide a

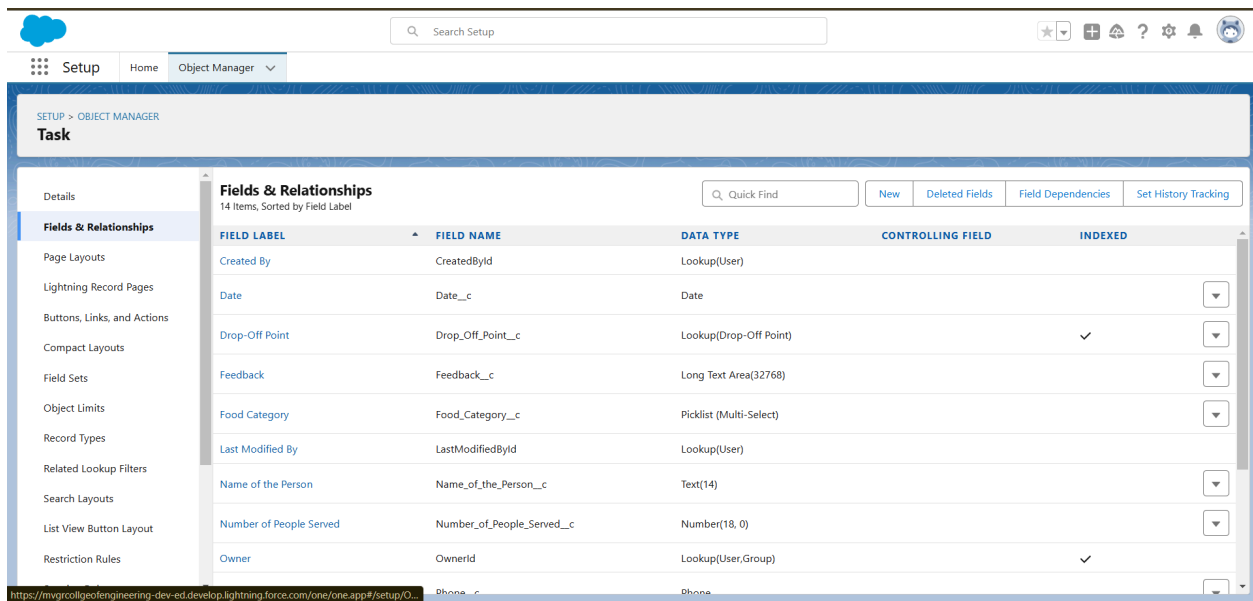connection between drop-off points and venues.

2. Task:

■ Purpose: The Task object is used to manage and track various tasks related to

food distribution, including assigning tasks to volunteers and recording task

details.

■ Fields:

○ Task Name: A text field representing the name of the task.

○ Date: A date field indicating when the task is scheduled or completed.

○ Rating: A numeric field used to rate the task performance.

○ Drop-Off Point: A lookup relationship field linking to the Drop-Off Point

object to associate tasks with specific drop-off points.

○ Sponsored By: A lookup relationship field linking to the Drop-Off Point

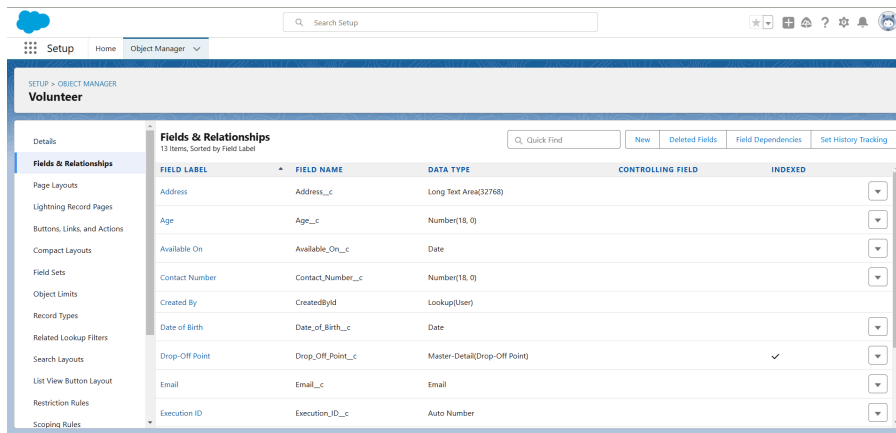object to indicate the sponsor of the task.

3. Volunteer:

■ Purpose: The Volunteer object keeps track of individuals who are involved in the food distribution process, including their contact information and assigned tasks.

■ Fields:

○ Volunteer Name: A text field for the name of the volunteer.

○ Contact Email: An email field for the volunteer's contact email.

○ Contact Phone: A phone field for the volunteer's contact number.

○ Assigned Tasks: A related list showing tasks assigned to the volunteer

.

4. Execution Detail:

■ Purpose: The Execution Detail object captures detailed information about food distribution activities, including which volunteers and task are associated with each distribution.

■ Fields:

○ Execution Detail Name: A text field for naming the execution detail record.

○ Volunteer: A master-detail relationship field linking to the Volunteer object to associate execution details with a specific volunteer.

○ Task: A master-detail relationship field linking to the Task object to associate execution details with a specific task.

○ Details: A text field for additional information about the execution.

2.2 Object Relationships

● Master-Detail Relationships:

■ Execution Detail to Volunteer: Ensures that each execution detail record is linked to a volunteer, making volunteer information essential for each distribution record.

■ Execution Detail to Task: Ensures that each execution detail record is linked to a task, making task information essential for each distribution record.

● Lookup Relationships:

■ Task to Drop-Off Point: Connects tasks to drop-off points, providing context on where each task is executed.

■ Task to Volunteer: Connects tasks to volunteers, enabling tracking of which volunteer is responsible for each task.

■ Drop-Off Point to Venue: Links drop-off points to venues, facilitating a detailed understanding of where food is being dropped off.

2.3 Sharing Rules

To ensure that records are shared appropriately based on distance criteria, the following sharing
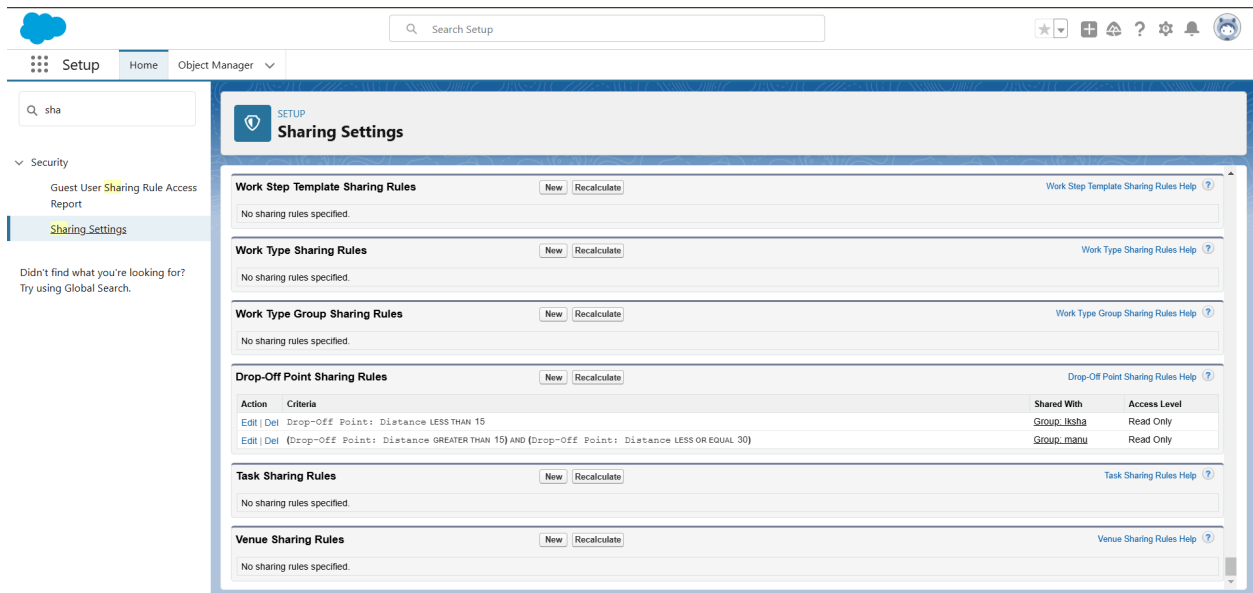
rules were set up:

● Rule 1:

■ Label: Rule 1

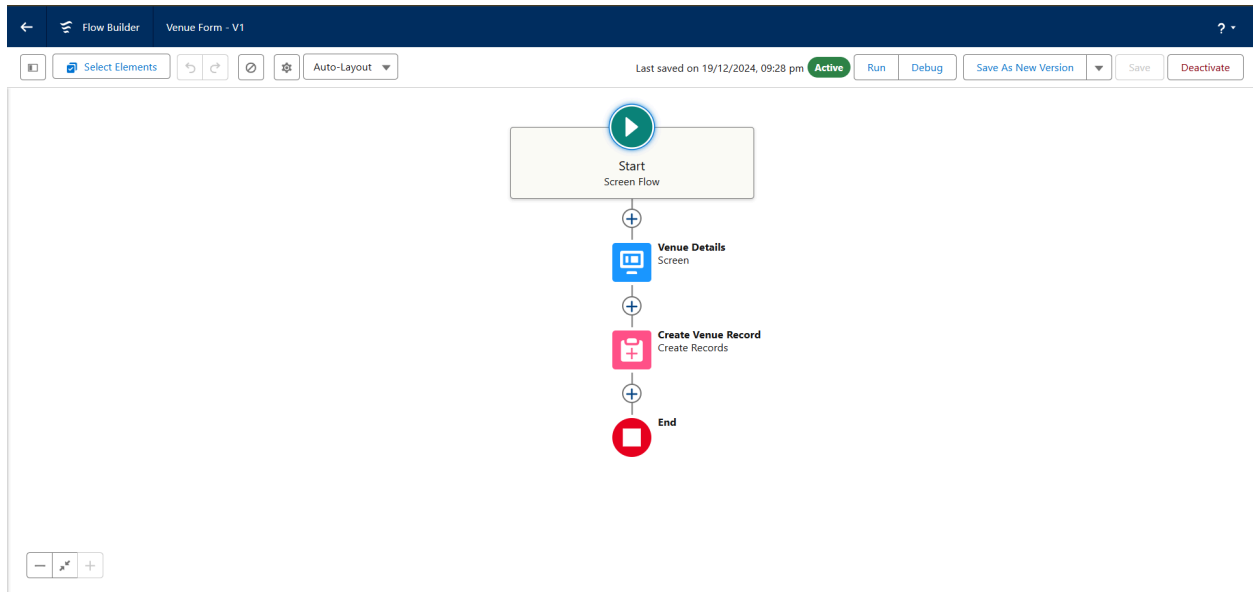■ Criteria: Records where the distance is less than 15 km

■ Access: Shared with Public Group - Iksha

● Rule 2:

■ Label: Rule 2

■ Criteria: Records where the distance is between 15 km and 30 km

■ Access: Shared with Public Group - NSS

2.4 Flow and Dashboard

● Flow: Venue Flow

■ Purpose: To manage and capture detailed information related to food drop-off points.

■ Components:

○ Text Component: For entering the venue name.

○ Email Component: For entering the contact email of the venue.

○ Phone Component: For entering the contact phone number of the venue.

○ Text Component: For entering the venue location.

○ Number Components: For entering latitude and longitude values of the venue.

● Dashboard:

■ Name: Organization Details

■ Components:

○ Lightning Table: Displays detailed information about venues and drop-off

points, facilitating a comprehensive view of the data.

3. Configuration and Setup Instructions

3.1 Creating Custom Objects

1. Drop-Off Point:

■ Navigate to Object Manager > Create > Custom Object.

■ Enter details such as the label name, record name, and other settings for Drop

Off Point.

2. Task:

■ Navigate to Object Manager > Create > Custom Object.

■ Enter details such as the label name, record name, and other settings for Task.

3. Volunteer:

■ Navigate to Object Manager > Create > Custom Object.

■ Enter details such as the label name, record name, and other settings for

Volunteer.

4. Execution Detail:

■ Navigate to Object Manager > Create > Custom Object.

■ Enter details such as the label name, record name, and other settings for

Execution Detail.

3.2 Sharing Rules Setup

● Go to Sharing Settings.

● Define new sharing rules for Drop-Off Point based on distance criteria, ensuring

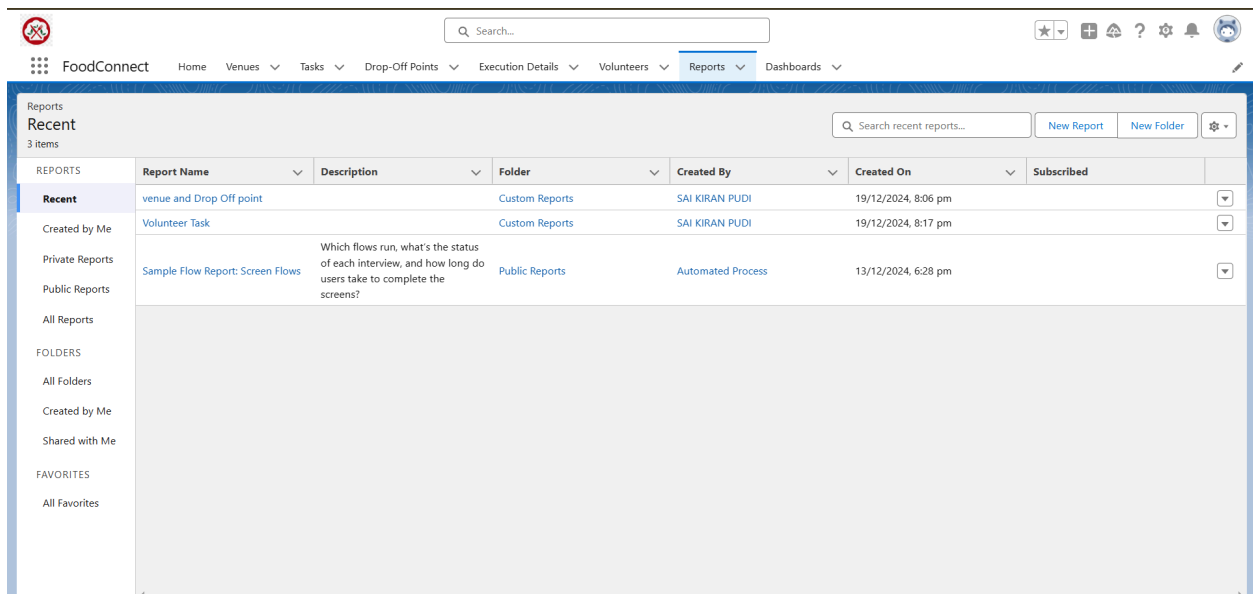appropriate access for different public groups.

3.3 Lightning App Builder

1. Create Home Page:

■ Go to Lightning App Builder > New.

■ Select Home Page and name it "HOME Page".

■ Add Flow and Dashboard components to the page.

2. Save and Activate:

■ Click on Save and Activation.

■ Set the page as the default for the FoodConnect app.

3.4 Report Types and Dashboards

1. Custom Report Types:

■ Create report types for Venue with Drop-Off with Volunteer and Volunteers with

Execution Details and Tasks.

2. Reports:

■ Generate and configure reports using the custom report types to track and

analyse relevant data.

3. Dashboards:

■ Configure dashboards to visualise data from reports and other components,

providing insights into the overall distribution process.

4. Deliverables

● Code Files: Include Apex triggers and any other code implemented.

● Output Screenshots: Provide screenshots of custom objects, flows, dashboards, and

reports to illustrate the setup.

● Project Documentation: Detailed descriptions of implemented features, configurations,

and setup instructions.

● Project Demonstration Video: A video showcasing the functionality of the application,

including key features and processes.