M.Kumarasamy College of Engineering
NAAC Accredited Autonomous Institution
Approved by AICTE & Affiliated to Anna University
ISO 9001:2015 Certified Institution
Thalavapalayam, Karur, Tamilnadu.

A Salesforce Project

On

TO SUPPLY LEFTOVER FOOD TO POOR

Submitted by

DHEEPANRAJ M -927622BIT016

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

M.KUMARASAMY COLLEGE OF ENGINEERING

(Autonomous)

KARUR - 639113

JULY 2025

FOOD CONNECT - SUPPLYING LEFTOVER FOOD TO THE POOR

PROJECT OVERVIEW

- FoodConnect is a Salesforce CRM-powered platform developed to simplify the process of collecting and distributing surplus food to people in need. It allows NGOs to work closely with volunteers, organize drop-off points, and monitor food delivery activities with ease. By improving coordination and tracking, the system helps reduce food wastage while promoting fairness and accountability in the redistribution process.
- The core mission is powerful yet straightforward: to help build a hunger-free world by turning excess food into a valuable resource for the underprivileged. Using technology for seamless coordination, encouraging community participation, and fostering a culture of generosity, FoodConnect aspires to ensure that every extra meal finds its way to someone's plate. Acting as a vital link between surplus and scarcity, it makes the act of giving both efficient and impactful.

OBJECTIVES

- The main goal of the FoodConnect CRM system is to address both food wastage and hunger by using technology to link surplus food providers with communities that need support. The platform is designed to make the donation and distribution process smoother, ensuring effective collaboration between donors, volunteers, and non-profit organizations.
- Through a centralized system, it manages donor and recipient information, tracks locations, drop-off points, and delivery routes to guarantee timely distribution. Volunteer activities, task allocations, and live status updates are handled efficiently, improving overall workflow.
- The solution also incorporates analytics and reporting features to maintain transparency and accountability. By automatically matching donations based on proximity, type of food, and urgency, it minimizes waste and ensures maximum benefit from available

resources. Ultimately, FoodConnect fosters stronger community connections, enables informed decision-making, and promotes sustainable resource management for lasting social good.

PHASE 1: REQUIREMENT ANALYSIS & PLANNING

A. Understanding Business Requirements

- FoodConnect's primary mission is to use a technology-enabled CRM platform to make the process of collecting and distributing surplus food to underprivileged individuals more organized and efficient. This phase involves analyzing real-world challenges and translating them into precise system requirements.

Key Stakeholders:

- Non-Governmental and charitable organizations
- Volunteer groups and individuals
- Managers of designated drop-off points
- Food donors (information maintained externally or entered manually)

Identified Core Needs:

- Effective allocation and monitoring of volunteer responsibilities
- Accurate recording and tracking of food pickup and delivery activities
- Administration of specified distribution sites
- Coordination of both temporary and permanent food handover locations
- Transparent tracking of volunteer availability and engagement

Objective:

To develop a CRM solution that facilitates smooth communication and coordination among food donors, volunteers, and NGOs, thereby reducing waste and increasing the positive reach of every donation.

B . Defining Project Scope and Objectives

Based on the requirement analysis, the scope for the **Minimum Viable Product (MVP)** is defined to focus on the core operational flow:

Custom Objects to be Developed:

1. **Task** – Assign and track responsibilities for volunteers, including pickup and delivery duties.
2. **Execution Detail** – Record the completion of each pickup and drop-off activity for accountability.
3. **Venue** – Maintain records of fixed donation or distribution centers.
4. **Drop-Off Point** – Track temporary or mobile food delivery locations for flexible operations.
5. **Volunteer** – Store and manage information about individuals engaged in field activities.

Key Objectives:

- Establish a centralized platform to manage and monitor all donation and distribution activities.
- Simplify volunteer task assignments and follow-ups to improve efficiency.
- Track the real-time status of food delivery events from assignment to completion.
- Maintain detailed records of fixed and temporary food handover locations.

- Provide visibility into volunteer availability and past contributions.
- Enable reporting and dashboards to strengthen transparency and accountability.

C. Designing the Data Model & Security Model

Data Model Overview:

- **Volunteer → Task**: One-to-Many relationship, allowing a single volunteer to handle multiple tasks.
- **Task → Execution Detail**: One-to-One relationship to link each task with its execution record.
- **Venue → Drop-Off Point**: One-to-Many relationship to associate multiple drop-off points with a fixed venue.
- **Drop-Off Point → Task**: One-to-Many relationship for managing multiple tasks at a single drop-off location.

Security Model Plan:

- **Admin**: Full access to all data, configuration, and system settings.
- **NGO Staff**: Restricted Create, Read, Update, Delete (CRUD) permissions for objects such as Task, Execution Detail, and Drop-Off Point, ensuring they can perform operational duties without accessing sensitive system configurations.

  The combination of a well-structured data model and a role-based security plan ensures that FoodConnect's MVP is both functional and secure, supporting efficient operations while safeguarding data integrity

PHASE 2: SALESFORCE DEVELOPMENT – BACKEND & CONFIGURATIONS

Environment Setup & DevOps Workflow

To ensure a secure and reliable development process, the project was built following Salesforce best practices for environment management and version control.

- **Developer Sandbox** was used for coding, customization, and testing without affecting live data.
- **Change Sets** facilitated the safe transfer of configurations from sandbox to production.
- **Version Control** was maintained using Git to track changes, collaborate efficiently, and prevent code conflicts.
- **Deployment** was carried out via Change Sets for smaller configuration moves or **Salesforce CLI (sfdx)** for advanced, automated deployment, ensuring smooth and error-free transitions to the production environment.

Customization & Configurations

The FoodConnect CRM was tailored with custom objects, fields, and relationships to align with operational needs.

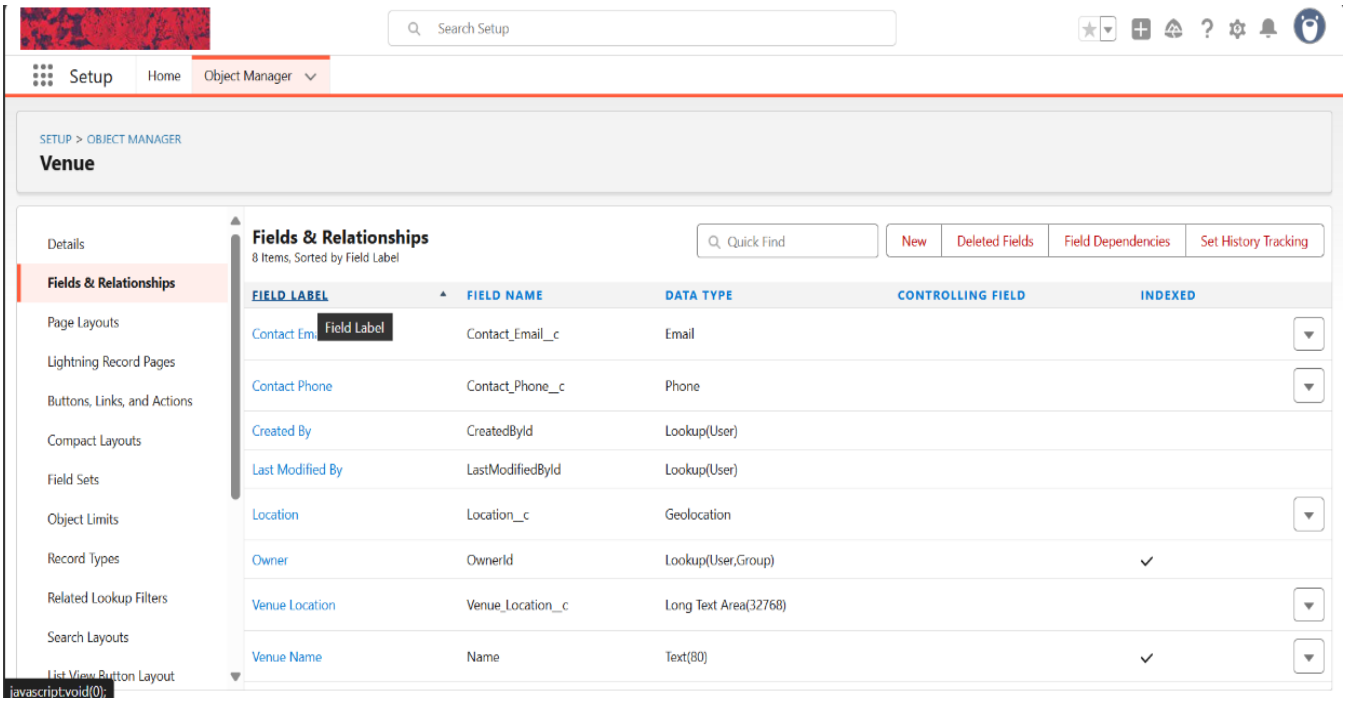
1. Object – Venue

- **Purpose**: Represents fixed donation or distribution centers where food is stored or handed over.
- **Key Fields & Relationships**:
    - **Venue Name** *(Text)* – Unique identifier for each location.

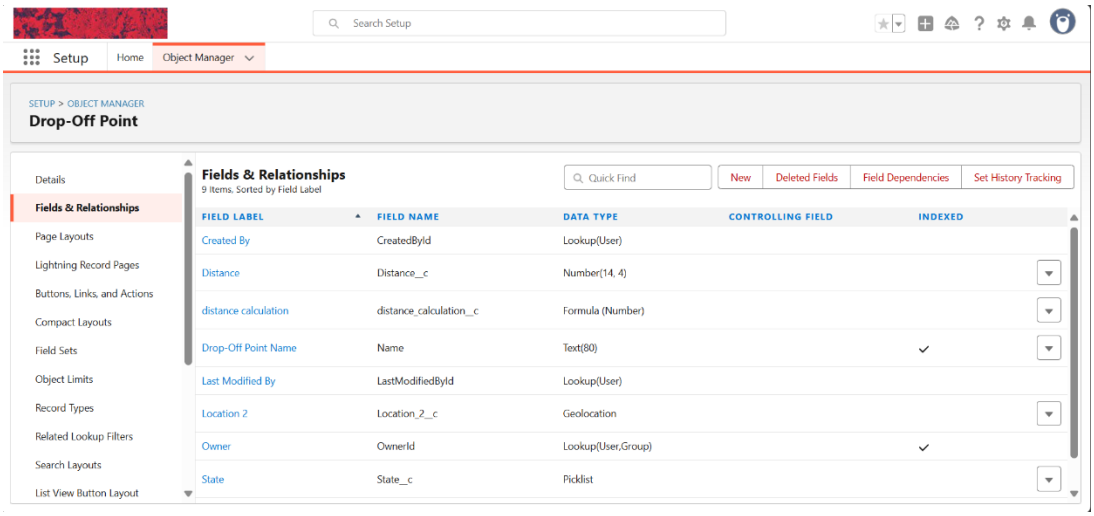
- **Address** *(Text Area)* – Complete location details.
- **Contact Person** *(Lookup to Volunteer)* – Responsible individual managing the venue.
- **Capacity** *(Number)* – Storage or handling capacity of the location.
- **Drop-Off Points** *(Master-Detail Relationship)* – Links multiple drop-off locations to the venue.
- **Status** *(Picklist)* – Active / Inactive state of the venue.

This configuration ensures that each venue can be effectively managed, linked to multiple drop-off points, and monitored for operational readiness.

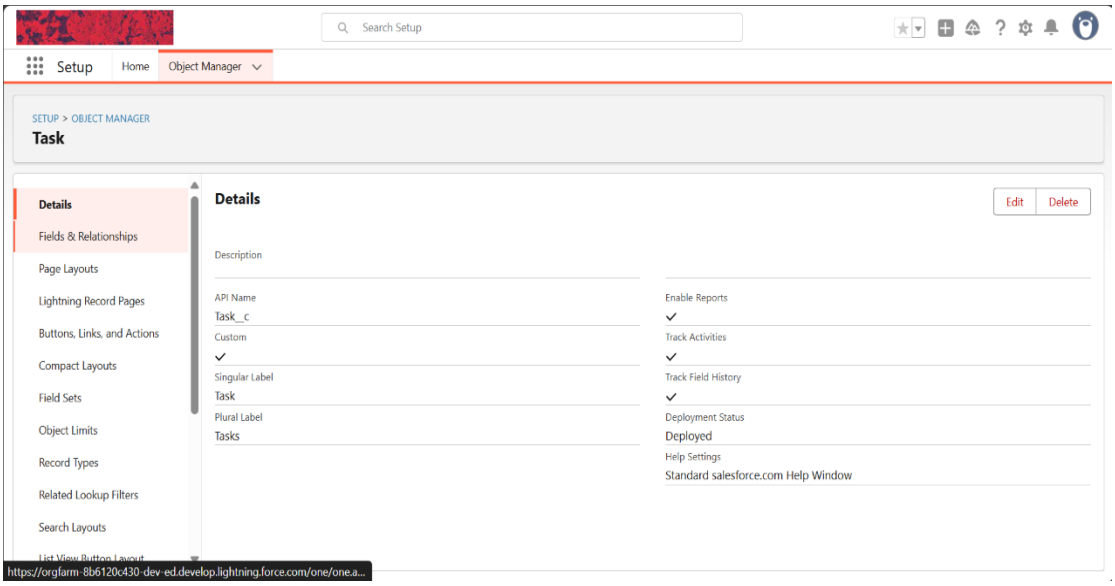
1.Object – Venue and Fields & Relationships



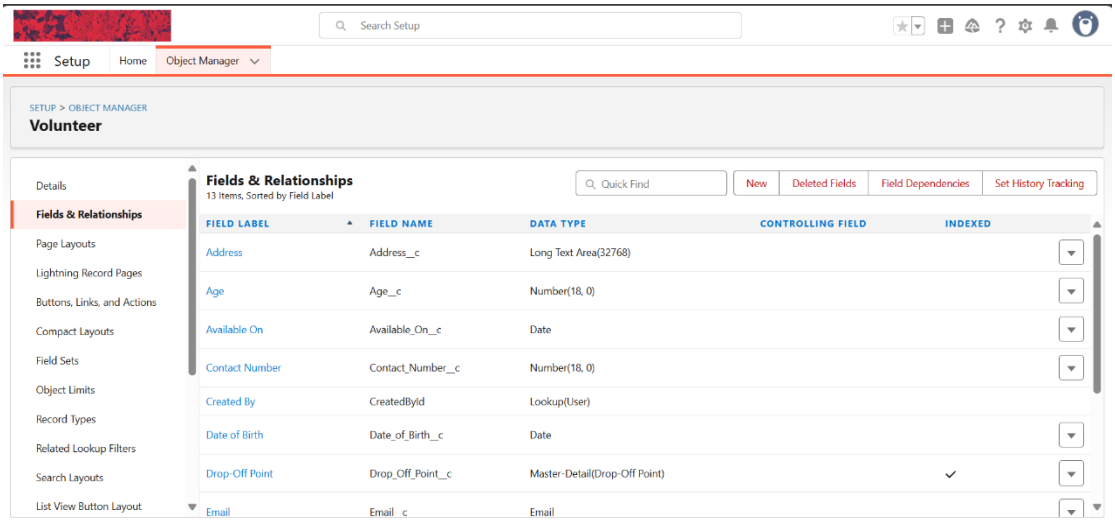
2.Object – Drop-Off Point and Fields & Relationships



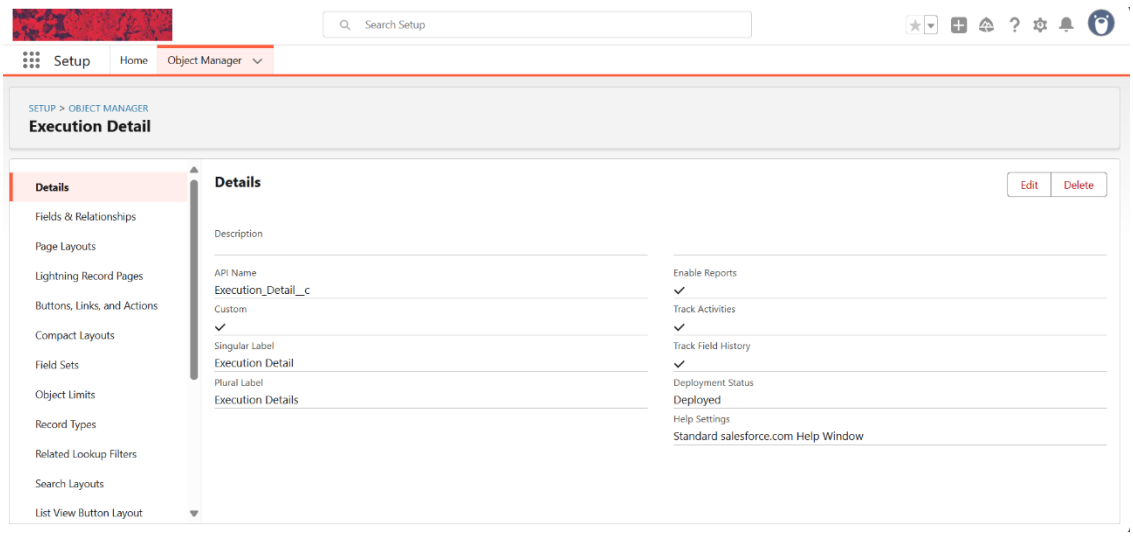
3.Object – Task and Fields & Relationships



4.Object – Volunteer and Fields & Relationships



5.Object – Execution Detail and Fields & Relationships



Field Configuration :

Each custom object in FoodConnect was configured with specific fields to ensure accurate data capture and efficient record management. The fields were designed to reflect real-world operational requirements and maintain data consistency.

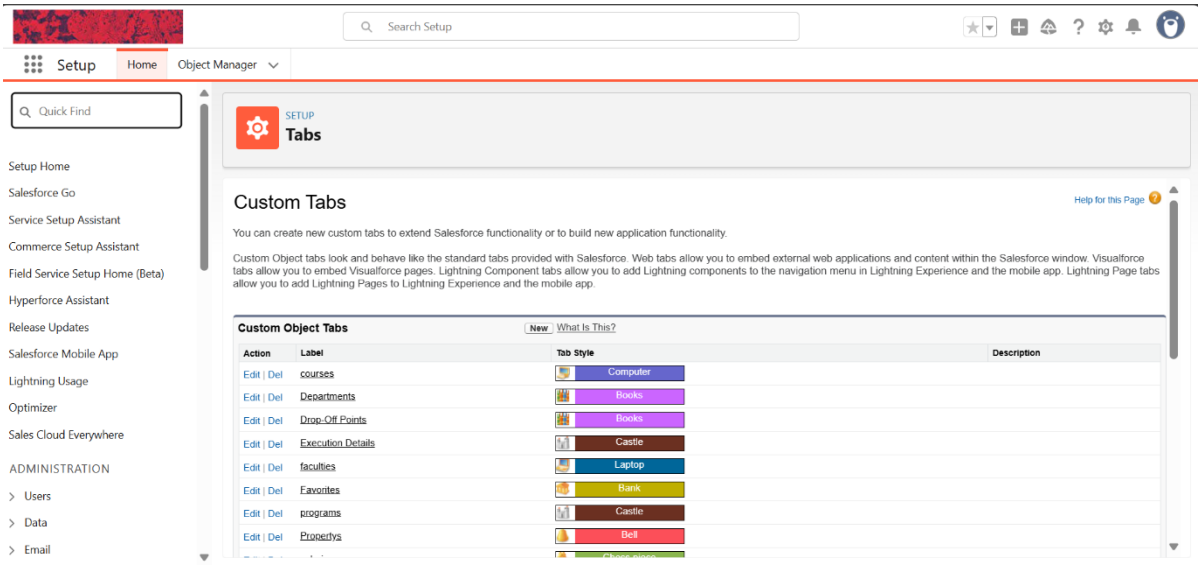
Examples of Configured Fields:

- **Venue:**
  - ***Venue Name*** – Unique identifier for the location**.**
  - ***Address*** – Detailed physical address**.**
  - ***Capacity* –** Storage or handling limit of the venue.
  - ***Contact Information* –** Phone number or email of the venue manager.
- **Task:**
  - ***Description* –** Summary of the assigned duty.
  - ***Due Date* –** Deadline for task completion.
  - ***Assigned Volunteer* –** Lookup to the volunteer responsible for the task.
- **Volunteer:**
  - ***Full Name* –** Volunteer's complete name**.**
  - ***Contact Details* –** Phone number and/or email.
  - ***Availability* –** Days or hours the volunteer is available for assignments.

Similar structured fields were implemented for other objects such as Drop-Off Point and Execution Detail, ensuring that all operational data is captured accurately and is easy to retrieve.

Custom Tabs for Navigation :

Custom tabs were created for each object—such as Venues, Drop-Off Points, Tasks, Volunteers, and Execution Details—to provide quick access and simplify data management. This setup improves navigation, saves time, and makes the system more user-friendly.

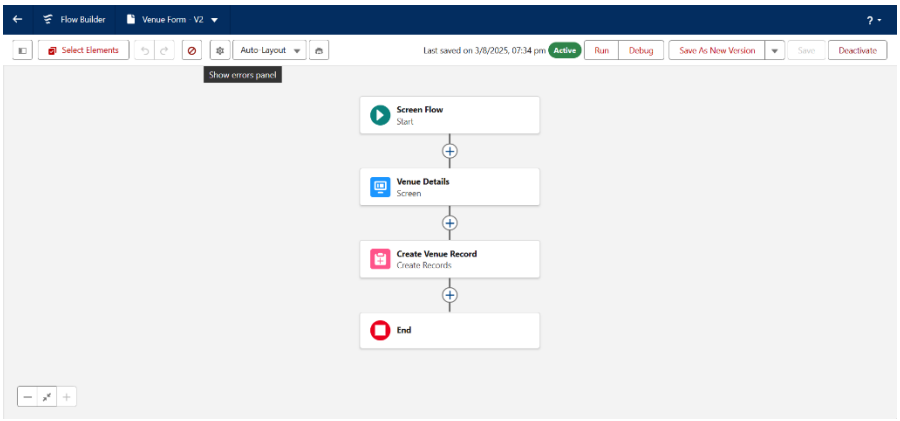


ValidationRules

Implemented rules to maintain data accuracy, such as requiring every task to have an assigned volunteer before saving.

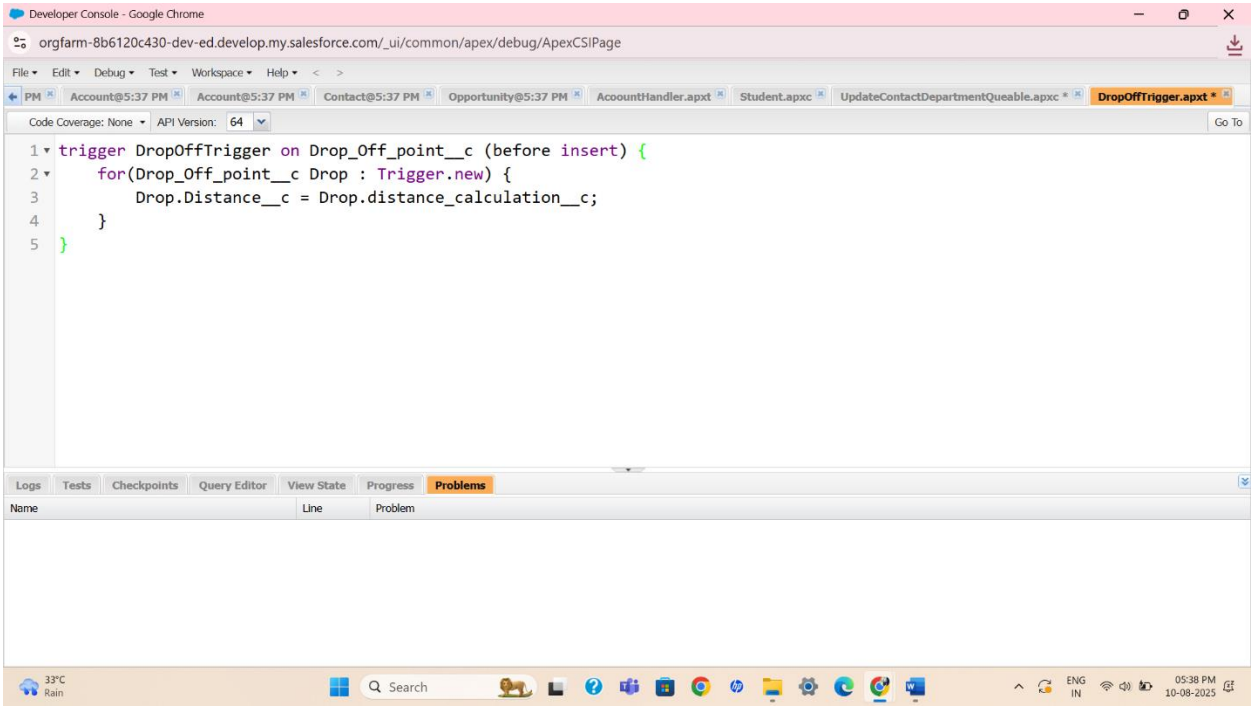
Automation Tools

- **Screen Flow** – Configured to automatically assign tasks based on volunteer availability and handle complex process logic.



Apex Development

- **Apex Classes**: Encapsulated business logic to automate workflows and manage tasks efficiently.

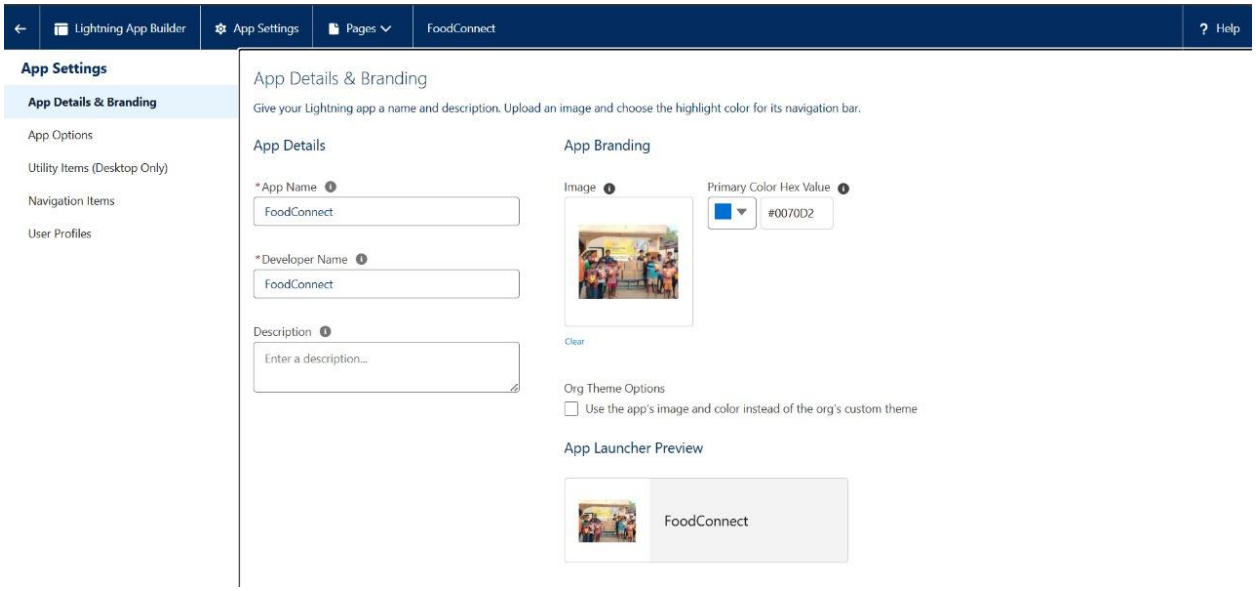


Asynchronous Apex: Utilized Future methods or Queueable Apex for bulk notifications and resource-heavy processes, such as large-scale volunteer record updates.

PHASE 3: UI/UX DEVELOPMENT & CUSTOMIZATION

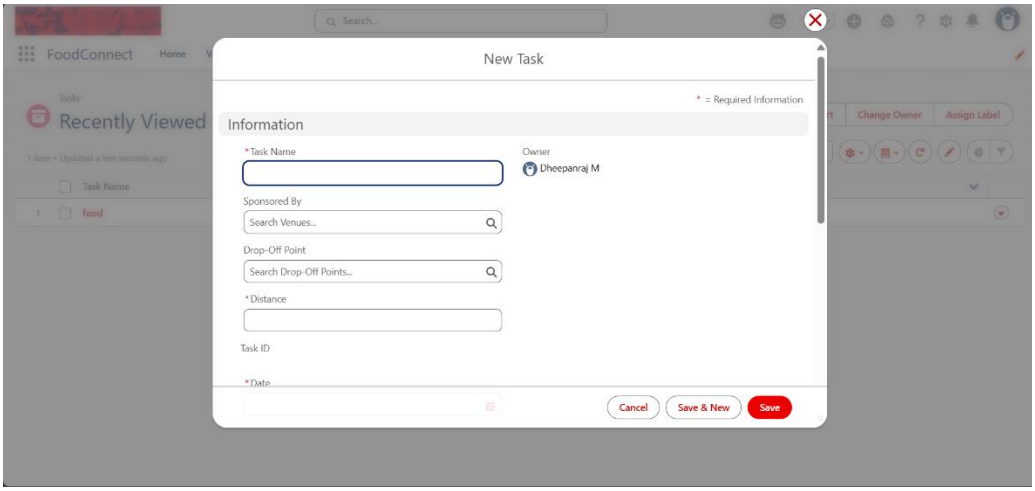
Lightning App Setup

- **Custom Lightning App**: *FoodConnect App* created using **App Manager**.
- **Organized Tabs**: Volunteers, Tasks, Venues, and Execution Details grouped in one workspace.
- **Role-Based Navigation**: Simplified interface tailored to user roles for faster access.



Page Layouts & Dynamic Forms

- **Customized Page Layouts**: Structured for each object so fields are logically grouped, making important details easy to view.
- **Dynamic Forms**: Configured to improve user experience by showing or hiding fields based on set criteria, such as *Task Status*.



User Management

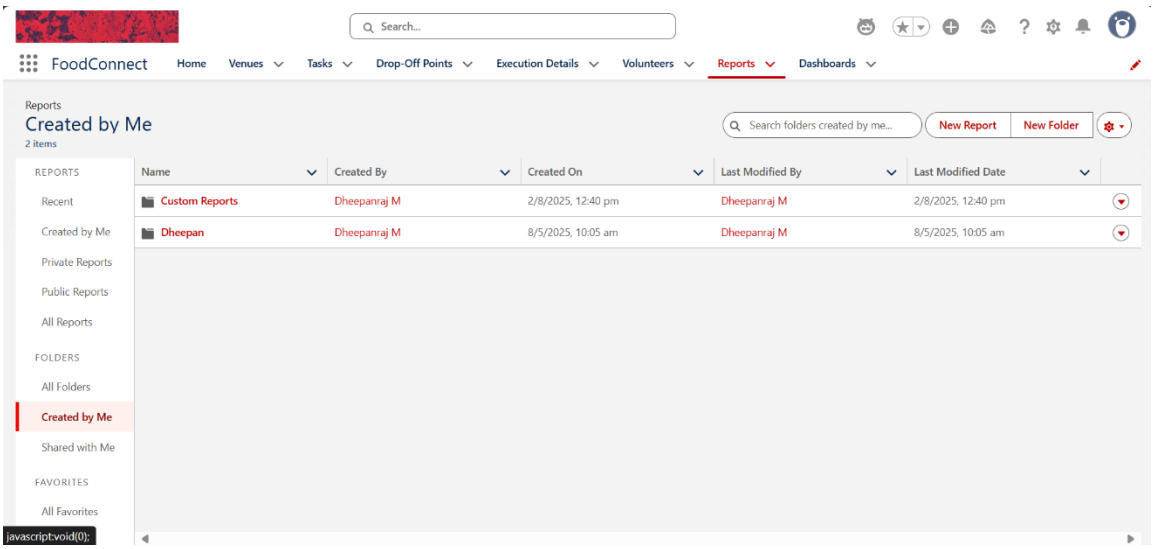
Profiles Created: *Admin*, *NGO Staff*, *Volunteer* to define role-specific access.

- **Permissions & Layouts**: Customized permission sets, page layouts, and tab visibility for each profile.
- **Data Security & Relevance**: Ensures users only access, view, and manage the information necessary for their responsibilities.

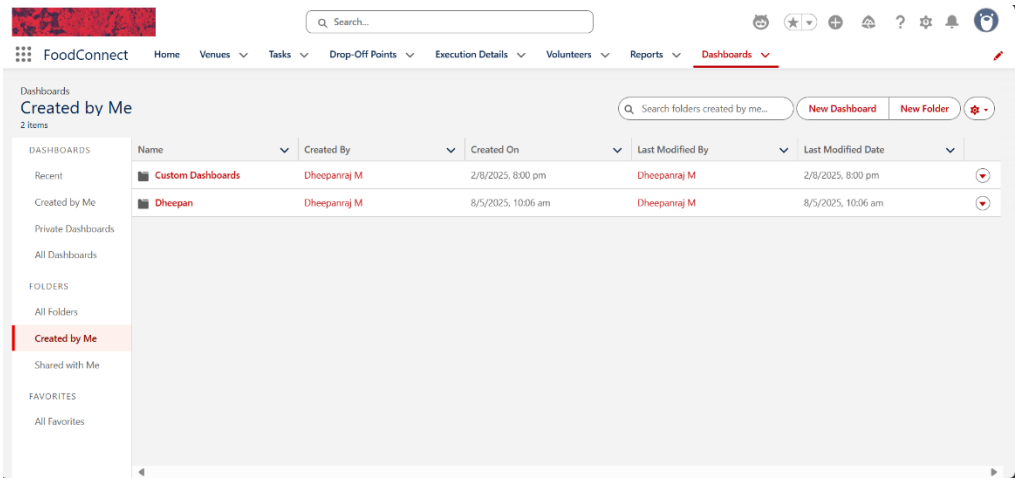
Reports and Dashboards

- **Custom Reports**: Track *task completion*, *volunteer performance*, and *food distribution metrics*.
- **Real-Time Insights**: Enable monitoring of operations and decision-making based on live data.

.

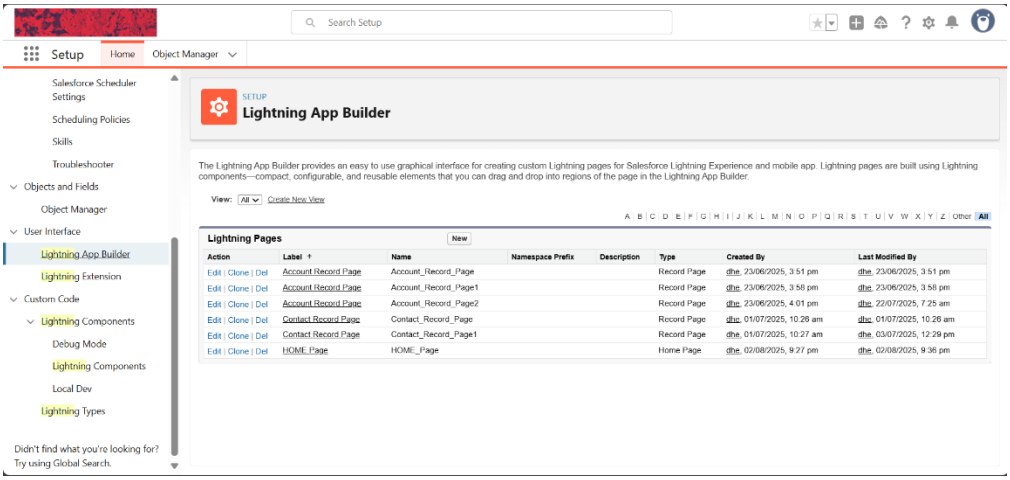


Dashboards: Display key metrics such as *Daily Drop-Off Activity*, *Assigned vs. Completed Tasks*, and *Volunteer Participation Rate* to track performance at a glance.



Lightning Pages

- Used **Lightning App Builder** to customize record pages.
- Built pages for key objects like **Task** and **Volunteer**, including **tabs**, **related lists**, and **report components**.
- Made pages **responsive** and **role-aware**, ensuring users see only the most relevant components.



Phase 4: Data Migration, Testing & Security

Data Migration

Objective:

Ensure that **legacy or pre-prepared records** are migrated into Salesforce **accurately**, with **minimal manual intervention** and **complete traceability** of the process.

Tool Used – Data Import Wizard:

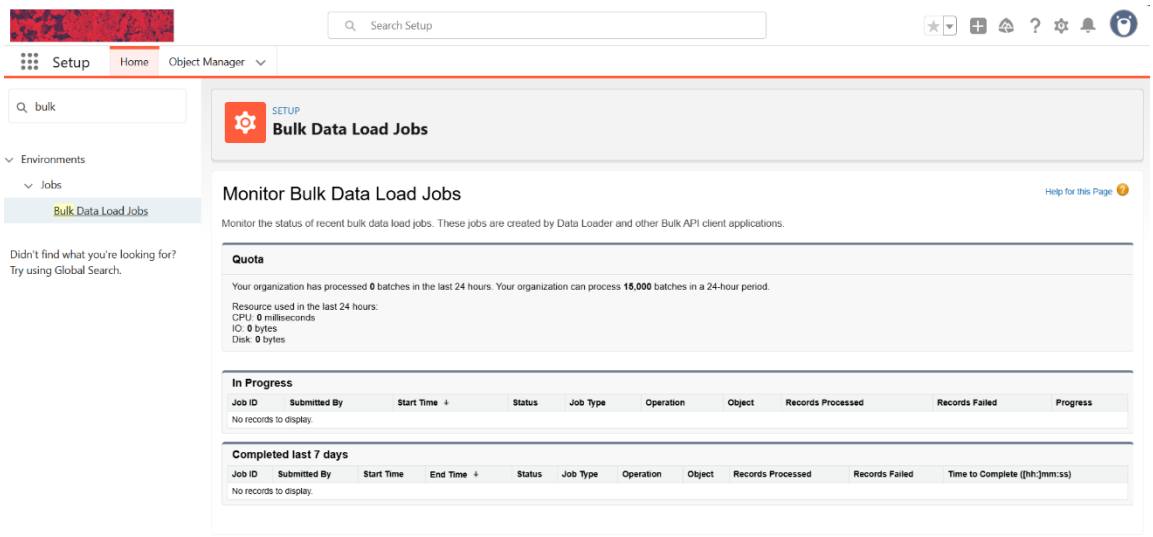
- Used for importing **simple records** into both **custom** and **standard objects** (e.g., *Venue*, *Volunteer*, *Drop-Off Point*).
- Provided an **intuitive, user-friendly interface** for **field mapping** during import.
- Allowed quick validation before import to **reduce errors**.

Example Import – Venue Object:

- Imported fields included: **Venue Name**, **Contact Email**, **Contact Phone**, **Latitude**, **Longitude**.
- Maintained **data consistency** by matching exact field formats.
- Generated **import history logs** for tracking and auditing.

Key Outcomes:

- **Accurate** and **efficient** data migration.
- **Reduced manual data entry** errors.
- Ensured **full traceability** for future audits.



Security Model Implementation

Objective:

To **enforce data security** and **access control**, ensuring that **only authorized users** can view, edit, or manage specific records and system features.

Profiles & Roles:

- **Created distinct Profiles** for key user types:
    - **Admin** – Full system access, including configuration and data management.
    - **Organizer** – Manage event details, volunteers, and drop-off points with restricted administrative capabilities.
    - **Volunteer** – Limited access to assigned tasks, related records, and personal profile data.
- **Defined a Role Hierarchy** to:
    - Reflect the **organizational reporting structure**.

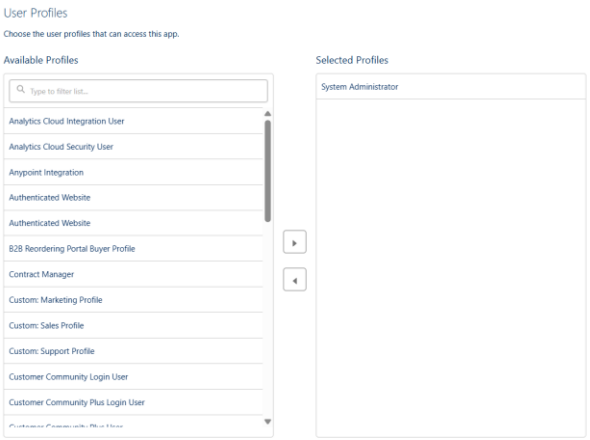
- Ensure **upward data visibility** (e.g., managers can see subordinates' records, but not vice versa).

Sharing Settings:

- Configured **organization-wide defaults (OWD)** to control baseline record visibility.
- Applied **criteria-based sharing rules** to grant additional access where needed (e.g., all organizers can see records tagged with their event ID).

Permission Sets:

- Created **granular permission sets** for specific features (e.g., report creation, dashboard access) without altering base profiles.
- Assigned permission sets dynamically to **reduce administrative overhead**.



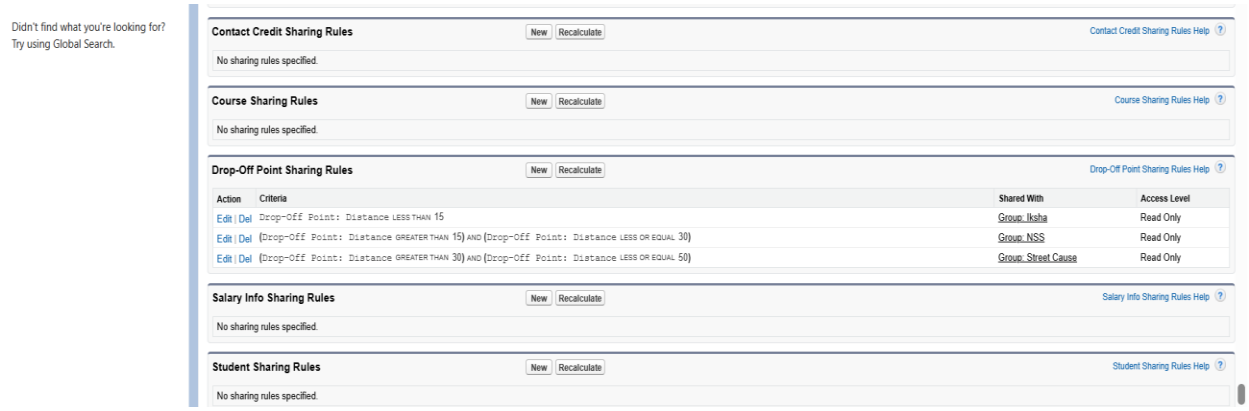
Permission Sets:

- Created to grant specific access without changing profiles.
- Included:
  - **Task Creation Access** – Create/assign tasks.
  - **Report & Dashboard Viewing** – View analytics without edit rights.

- **Flow Trigger Permissions** – Run automated flows.

Sharing Rules:

- **OWD Configured** – Established baseline record access for all objects.
- **Criteria-Based Sharing** – Allowed selective access to Volunteer records based on role or manager assignment.
- **Role Hierarchy Alignment** – Ensured managers automatically inherit access to their team's data.
- **Security Compliance** – Balanced collaboration needs with data privacy requirements.



Audit & Validation

Field History Tracking:

- Enabled on sensitive objects:
    - Booking: Tracks changes in status or time slot.
    - Execution Details: Tracks any update in remarks or completion status.

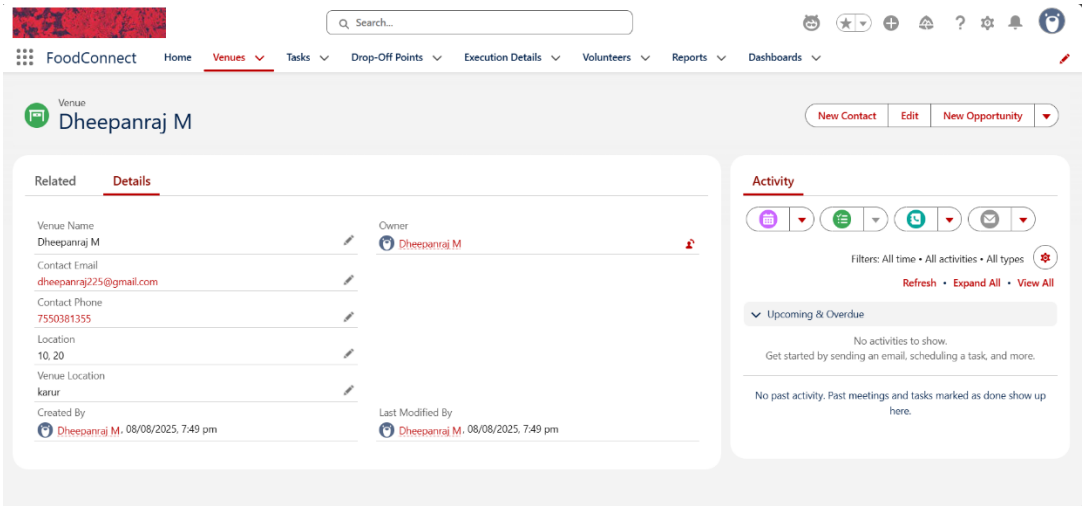
Duplicate Rules & Matching Rules:

- Enabled to avoid duplicate venue entries and repeated volunteer registrations.
- Configured matching rules on:
    - Email
    - Phone Number

Testing Process

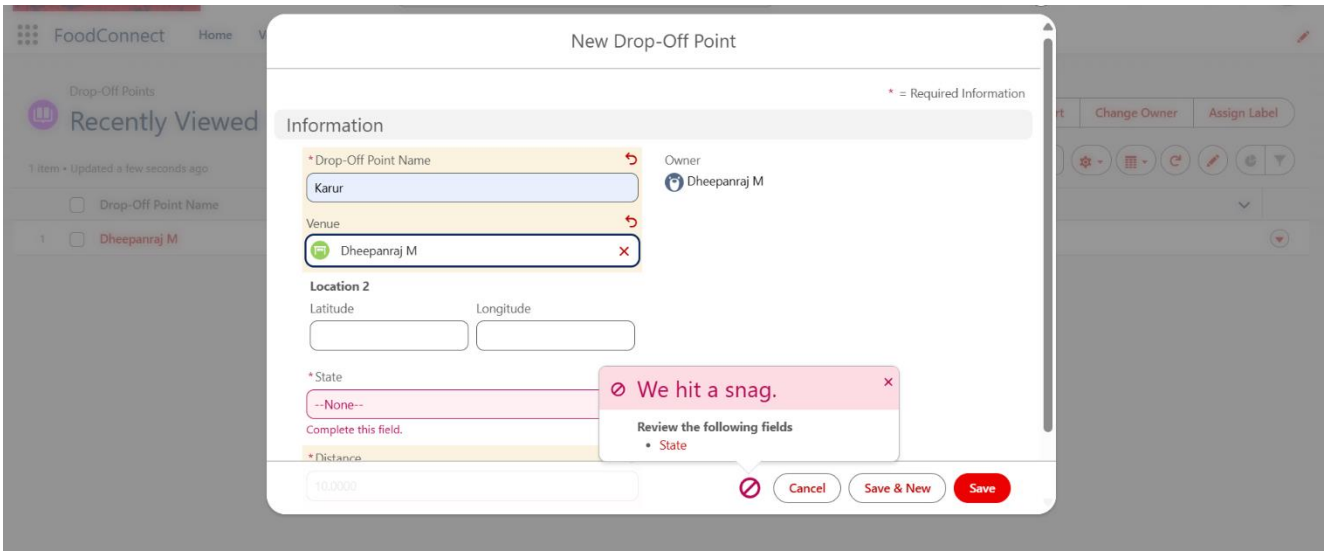
Test Case 1 – New Venue Creation

- Create a **booking** and verify it saves correctly.
- Confirm **auto task creation** (if enabled).



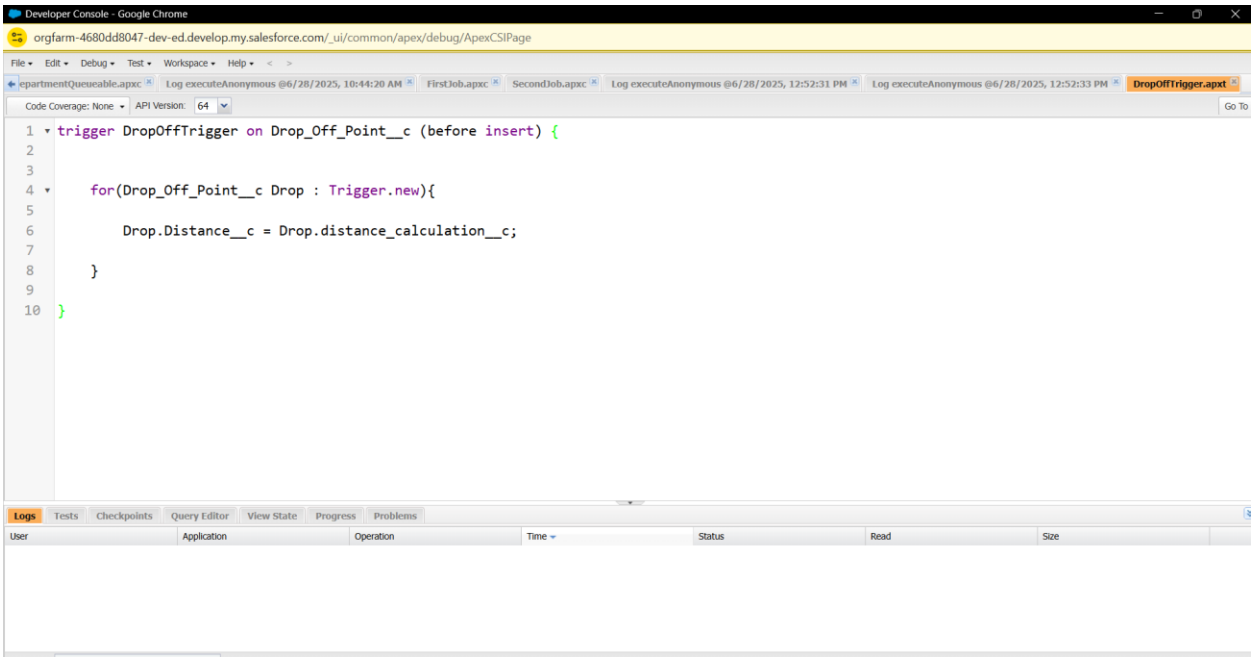
Test Case 2 – New Task Creation

- Verify error triggers if task distance exceeds limit set by sharing rules.
- Confirm system blocks creation to enforce security constraints.



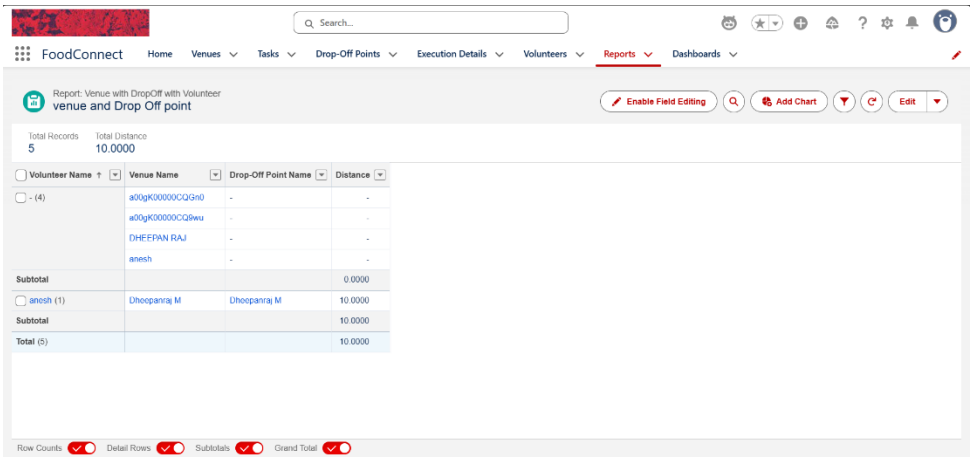
Test Case 3 – Unit Testing (Developer Testing)

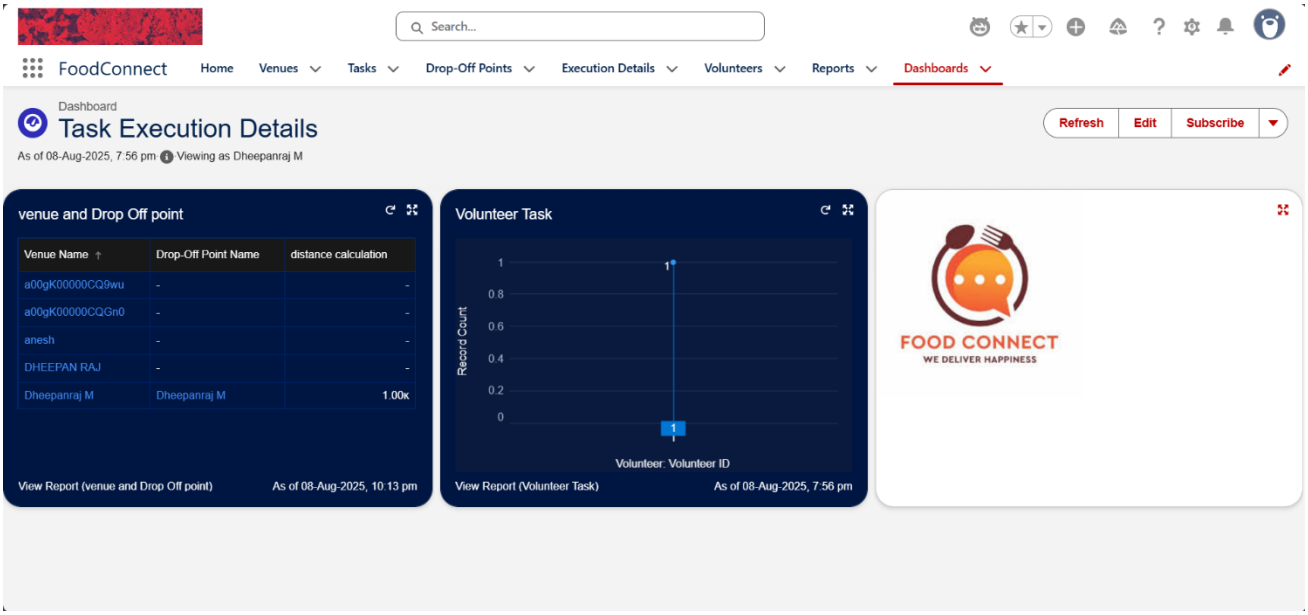
- Perform thorough testing of individual components to ensure correct functionality:
    - **Triggers:** Validate proper execution during record creation, update, or deletion.
    - **Apex Classes:** Test business logic methods for accuracy and error handling.
    - **Flows:** Confirm automated processes run smoothly without errors.
    - **Validation Rules:** Ensure data validation enforces rules correctly and prevents invalid data.
- Identify and fix bugs early in the development cycle for stable system performance.



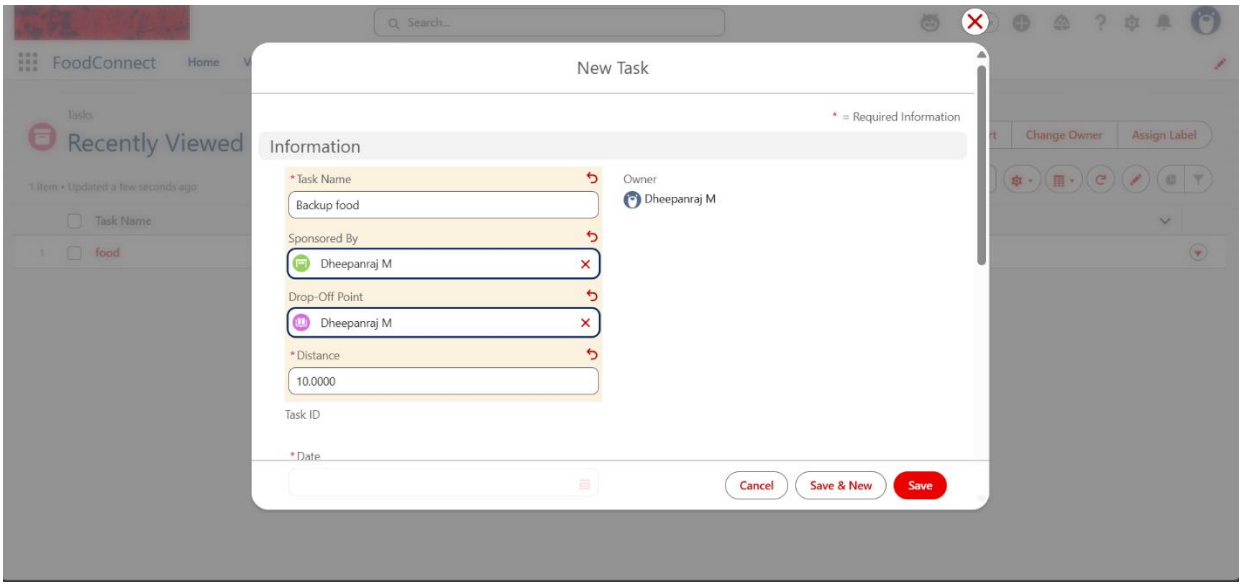
Test Case 4 – Reports and Dashboards Testing

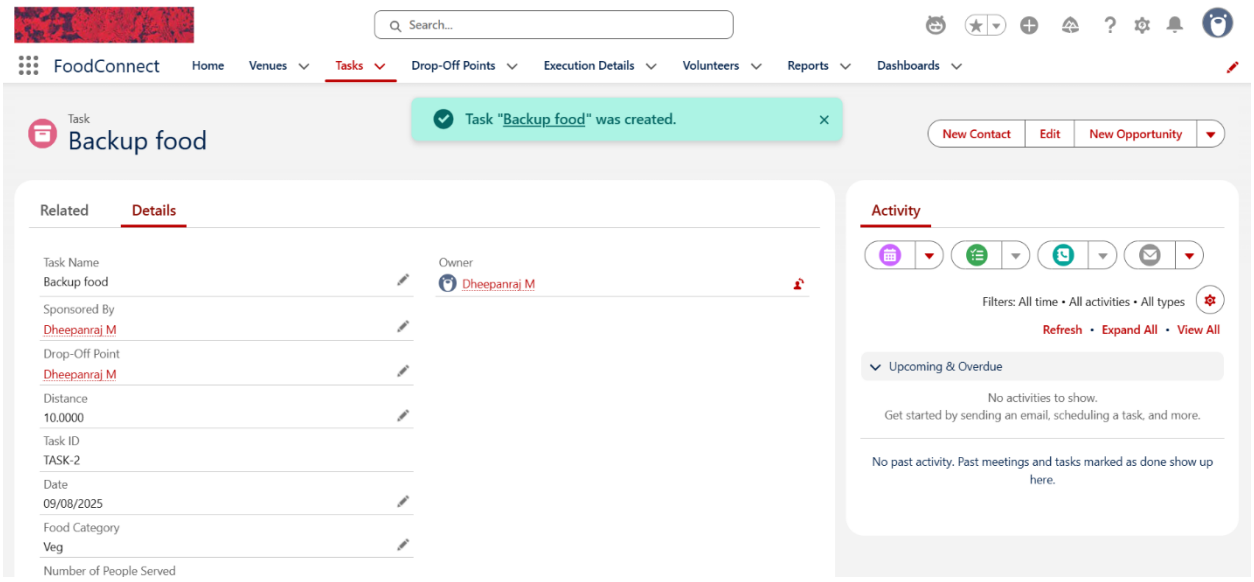
- Validate that reports accurately reflect **up-to-date data** from all relevant objects such as Tasks, Venues, and Volunteers.
- Check that dashboards visualize critical metrics like **Daily Drop-Off Activity**, **Tasks Assigned vs. Completed**, and **Volunteer Participation Rate** correctly.
- Test report filters, grouping, and sorting functionalities for correctness and usability.
- Ensure report export options work as intended and data is consistent across exports.
- Verify that dashboard components refresh data in real-time or on schedule as configured.





Test Case 5: Form Validation in Obj Task





Phase 5: Deployment, Documentation & Maintenance

Deployment Strategy

- Deployed the FoodConnect application using **Salesforce Change Sets**, facilitating a secure and reliable transfer of components from the sandbox to the production environment.
- Included components:

- **Custom Objects**: Venue, Drop-Off Point, Volunteer, Execution Details, Task capturing core data entities.
- **Automation**: Flows, Process Builders, Validation Rules, and Approval Processes — automating business logic and enforcing data integrity.
- **User Interface**: Custom Fields, Page Layouts, Lightning Pages — enhancing user experience and accessibility.
- **Notifications**: Email Alerts and Workflow Rules — ensuring timely communication and task updates.

- Change Sets provided version control, rollback options, and facilitated **collaborative deployment** with the team, reducing risks of errors or downtime.
- Deployment followed best practices including thorough **pre-deployment testing**, scheduled rollout windows, and post-deployment validation to confirm system stability.

System Maintenance & Monitoring

To maintain optimal system performance, stability, and data integrity, ongoing maintenance strategies were implemented:

- **Weekly Scheduled Data Backups** leveraging Salesforce's Data Export feature to secure data snapshots for recovery purposes.
- **Audit Trail** functionality enabled to monitor changes to system configurations, providing an audit log of administrative actions.
- **Field History Tracking** configured on key objects such as *Booking*, *Venue*, and *Volunteer* to track changes on critical fields and support troubleshooting.
- **Scheduled Jobs and Flow Monitoring** performed weekly via Salesforce Setup, ensuring background processes and automations run without errors.

- **Automated Error Notifications** sent to administrators via email alerts when flows or scheduled jobs fail, allowing for prompt issue resolution.

Troubleshooting Approach

The following systematic steps guide troubleshooting efforts

1. **Review Setup Logs:** Examine flow execution logs, Apex debug logs, and system exception logs to pinpoint errors.
2. **Validate User Permissions:** Confirm that users have appropriate access rights through roles, profiles, and permission sets to perform required actions.
3. **Inspect Object Relationships:** Verify that all custom object references (e.g., lookup fields, master-detail relationships) are correctly configured.
4. **Sandbox Testing:** Replicate issues in the sandbox environment to safely diagnose and test fixes before applying them to production.
5. **Data Integrity Checks:** Utilize Salesforce Data Loader or Workbench to identify and resolve data issues such as duplicates, missing lookups, or incorrect records.

Key Salesforce Features Implemented

Validation Rules:

- Enforced required fields like *Venue Email* and *Location Coordinates* to ensure complete and valid data.
- Applied custom logic to prevent duplicate volunteer entries, maintaining data cleanliness.

Approval Processes:

- Automated approval workflows triggered for significant donations or drop-off scheduling, notifying managers to review and approve.

Automation Flows:

- Created flows that automatically generate tasks when a venue record is approved, streamlining operational task management.
- Configured email alerts to notify volunteers upon task assignments, improving communication.
- Implemented record-triggered flows to update related records dynamically, maintaining data consistency.

Custom Objects:

- Developed key custom objects to support the project's data model, including *Venue__c*, *Volunteer__c*, *Execution_Detail__c*, *DropOff__c*, and *Booking__c*.

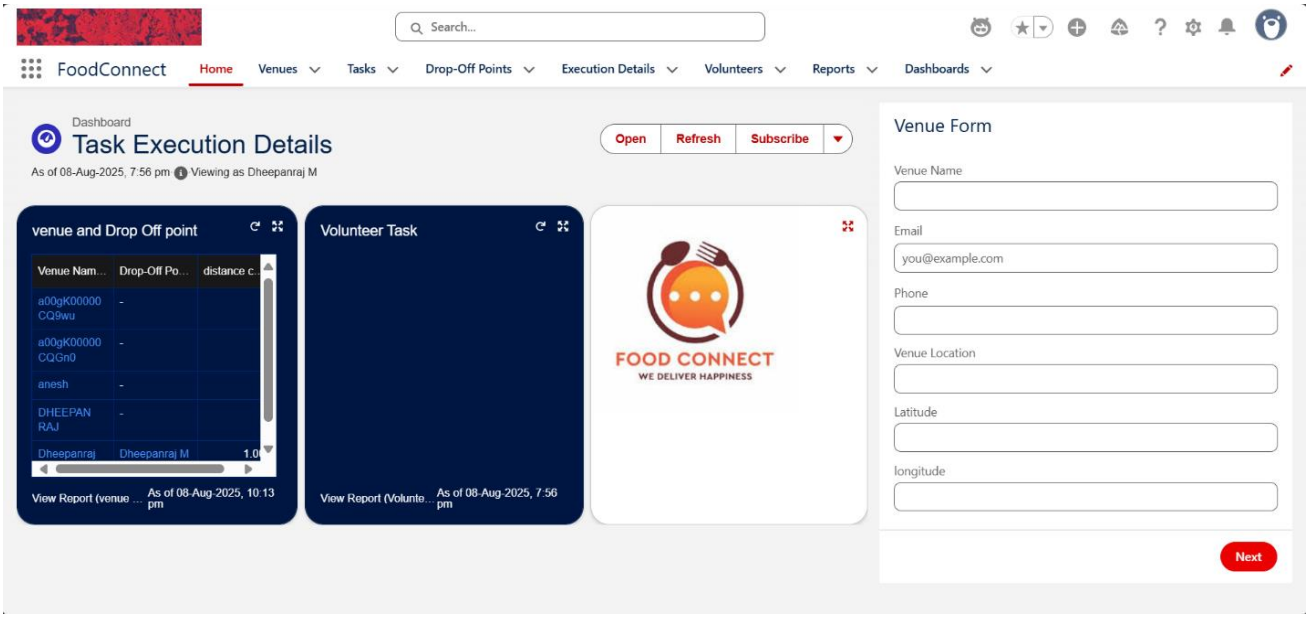
Testing Approach

- **Unit Testing:** Rigorous testing of individual components such as Apex classes, flows, validation rules, and approval processes to ensure they function correctly in isolation.
- **System Testing:** Comprehensive end-to-end testing of key processes including booking creation, task assignments, and volunteer management workflows to validate integrated system behavior.
- **User Acceptance Testing (UAT):** Involving real users (organizers and volunteers) to simulate typical use cases, ensuring the system meets business needs and user expectations before launch.

Future Enhancements

To keep FoodConnect adaptable and efficient as it grows, several future enhancements are planned:

- **Chatbot Integration:** Use Salesforce Einstein Bot to provide users with guided assistance for booking, donations, and volunteer coordination through conversational AI.
- **AI-based Suggestion Engine:** Implement AI algorithms to match volunteers with tasks automatically based on factors like location, availability, and skill set.
- **Mobile App Integration:** Develop mobile applications using Salesforce Mobile SDK for greater accessibility and real-time updates on the go.
- **Community Cloud Site:** Create a self-service portal allowing venue managers and volunteers to update details, view schedules, and communicate independently.
- **Dashboard Enhancements:** Expand dashboards to offer deeper insights into food distribution metrics, volunteer engagement, and operational KPIs for data-driven decision making.



Conclusion :

The Salesforce-based FoodConnect system has been successfully designed, developed, and deployed to effectively address the challenges of coordinating leftover food collection and distribution to underserved communities. By integrating custom objects, automated workflows, approval processes, and comprehensive security measures, the platform ensures smooth management of volunteers, drop-off locations, and task assignments with high accuracy and accountability.

The system's modular and scalable design not only meets the current operational requirements but also lays the groundwork for future growth and feature expansion. This flexibility allows FoodConnect to adapt to evolving community needs, increasing its impact over time.

Beyond technology, the project fosters stronger collaboration between NGOs, volunteers, and donors by simplifying communication and task coordination. It embodies a socially responsible approach to reducing food waste while tackling hunger, demonstrating how innovative technology solutions can drive meaningful community change.

Overall, FoodConnect stands as a robust, scalable, and user-friendly platform that empowers communities to work together toward a hunger-free future.