

# PROJECT DESIGN PHASE

---

## 1. Introduction

The **Project Design Phase** defines the architecture, interface, and data flow for the *To Supply Leftover Food to Poor* system.

This Salesforce-based platform is designed to connect **food donors**, **NGOs**, and **volunteers** under one cloud environment.

The design phase focuses on how the identified requirements will be translated into a structured, efficient, and user-friendly system using Salesforce tools such as **Objects**, **Apex Classes**, **Visualforce/Lightning Pages**, and **Automation Workflows**.

The goal is to ensure the application is scalable, secure, and intuitive while maintaining seamless interaction among all stakeholders.

---

## 2. Design Objectives

The main objectives of the project design phase are:

- To develop a **modular system architecture** for efficient data handling.
  - To create **user interfaces** that are intuitive for different roles (Donor, NGO, Volunteer, Admin).
  - To define **relationships among Salesforce objects** ensuring smooth data flow.
  - To establish **automated workflows** and **approval processes** for timely operations.
  - To design **reports and dashboards** for effective monitoring and analytics.
-

### **3. System Architecture**

The *To Supply Leftover Food to Poor* system follows a **multi-layer architecture** comprising the following layers:

#### **a) Presentation Layer**

This is the user interface accessible through Salesforce Lightning or Visualforce pages. It provides different views based on user roles:

- Donors: Food donation form, status tracking, and feedback section.
- NGOs: List/map of available donations, acceptance option, and delivery records.
- Volunteers: Delivery tasks, route details, and update forms.
- Admin: Dashboard with full system analytics and controls.

#### **b) Application Layer**

This layer includes all business logic implemented through **Apex classes, triggers, and flows**. It handles donation creation, assignment logic, notifications, and delivery confirmation.

#### **c) Data Layer**

This layer consists of Salesforce custom objects and relationships. It ensures all food, donor, NGO, and volunteer data are securely stored and linked for real-time access.

---

### **4. Data Flow Design**

The **data flow** in the system moves as follows:

1. **Donor** logs in and submits a new food donation record.
2. The system notifies nearby **NGOs** using automation flows.

3. An **NGO** accepts the donation and assigns a **volunteer** for pickup.
4. The **volunteer** collects and delivers the food to beneficiaries.
5. The **delivery record** is updated, and confirmation is sent to the donor and admin.
6. Data from all transactions are stored and displayed in **reports and dashboards** for monitoring.

This flow ensures real-time coordination between stakeholders and minimizes delays in food redistribution.

---

## 5. Database Design

Salesforce's object-oriented data model allows efficient data storage and management.

### Key Custom Objects and Relationships

Object Name	Purpose	Relationships
<b>Donor</b>	Stores donor details (restaurants, hotels, event organizers).	One-to-Many with Food Donations.
<b>Food Donation</b>	Holds information about donated food items.	Master-Detail with Donor; Lookup to NGO and Volunteer.
<b>NGO</b>	Contains NGO details and their food distribution capacity.	One-to-Many with Deliveries.

Object Name	Purpose	Relationships
Volunteer	Tracks volunteer details and delivery availability.	Lookup to Delivery object.
Delivery	Manages pickup, delivery, and confirmation details.	Lookup to Volunteer, NGO, and Food Donation.

#### Relationships Used:

- **Master-Detail Relationship** between Donor and Food Donation.
  - **Lookup Relationships** between Food Donation, Volunteer, and NGO.
  - **Hierarchical Relationship** for Admin to manage all users.
- 

## 6. User Interface Design

User interfaces are designed with Salesforce **Lightning Experience** for clarity and accessibility:

### Donor Interface

- “Create Donation” form with fields for food type, quantity, expiry time, and pickup address.
- Donation history and live status updates.

### NGO Interface

- “Available Donations” dashboard with location filters.
- “Accept Donation” and “Assign Volunteer” features.

### Volunteer Interface

- Task list showing assigned pickups and deliveries.
- Real-time update buttons for “Picked Up” and “Delivered.”

## **Admin Interface**

- Central dashboard summarizing all donations, deliveries, and active users.
  - Access to reports, analytics, and manual overrides.
- 

## **7. Process Design and Automation**

Automation is a critical part of the Salesforce design.

### **Automated Flows and Processes**

- **Auto Notification Flow:** Sends email/SMS alerts to NGOs when a donation is posted.
- **Assignment Rule:** Automatically assigns volunteers based on location and availability.
- **Delivery Confirmation Flow:** Updates donation and delivery status upon completion.
- **Report Scheduler:** Generates weekly reports on food collected and distributed.

### **Validation Rules**

- Ensures that expiry date of food is valid.
  - Prevents duplicate donation entries.
  - Confirms that delivery cannot be marked complete without NGO confirmation.
- 

## **8. Security Design**

To ensure data protection and controlled access, Salesforce's **role-based security** model is used:

## Role Access Level

Admin	Full access to all objects and settings.
Donor	Create and view own donations only.
NGO	View and accept donations; manage assigned deliveries.
Volunteer	Update assigned delivery tasks only.

Field-level security and sharing rules are configured to maintain data confidentiality between different user types.

---

## 9. Reports and Dashboards

To measure the project's impact, Salesforce **Reports and Dashboards** provide:

- Total number of donations created, accepted, and delivered.
  - Quantity of food redistributed by time period.
  - Performance summary of NGOs and volunteers.
  - Visual charts for active donors and successful deliveries.
- 

## 10. Conclusion

The **Project Design Phase** establishes a solid technical foundation for the *To Supply Leftover Food to Poor* system. By leveraging Salesforce's cloud architecture, automation tools, and security model, the design ensures a scalable, reliable, and user-friendly solution.

This design guarantees real-time coordination between donors, NGOs, and volunteers — reducing food wastage, improving efficiency, and supporting the broader mission of eliminating hunger through technology.