

Goals of the Architecture:

- Create a reliable system to connect **food donors, volunteers, and receivers** (NGOs or shelters).
 - Ensure **safe and timely redistribution** of leftover food to minimize waste.
 - Maintain **data integrity** between donors, available food, and delivery operations.
 - Enable **real-time notifications** for nearby food donations and pickup updates.
 - Reduce manual coordination and increase efficiency using automation.
-

Key Components:

- **Food Donor Module:** Allows restaurants, hotels, and event halls to register and post leftover food details.
 - **Volunteer Management Module:** Tracks and assigns volunteers for food collection and delivery.
 - **NGO/Receiver Module:** Displays available food for pickup and records receipt confirmations.
 - **Food Inventory Database:** Stores donation details, quantity, donor info, and pickup records.
 - **Notification & Tracking System:** Sends real-time alerts for food availability and delivery status.
 - **Quality Check Module:** Verifies food safety before distribution to ensure hygiene
 - **Analytics & Reporting Dashboard:** Provides statistics on food saved, meals distributed, and donor impact.
-

Development Phases:

- **Donor Registration:** Restaurants, hotels, and canteens register on the platform.
 - **Food Listing:** Donors post leftover food details (type, quantity, and pickup window).
 - **Volunteer Assignment:** The system automatically notifies available volunteers for pickup.
 - **Food Verification:** Collected food is checked for quality by the NGO or coordinator.
 - **Distribution:** Volunteers deliver the food to poor people or shelters.
 - **Feedback & Reporting:** Receivers confirm delivery; the system logs data for reports.
-

Solution Architecture Description:

The *To Supply Leftover Food to Poor* solution architecture is designed to create a **sustainable food redistribution network** using technology-driven coordination between food donors, NGOs, and volunteers.

The system collects leftover food details from registered donors and sends automated notifications to nearby NGOs and volunteers through a cloud-based platform. Once a volunteer accepts a pickup request, real-time tracking and updates are enabled to ensure quick delivery.

All data — including food details, delivery status, and recipient feedback — is stored in a **centralized database** for transparency and reporting. The architecture ensures safe, traceable, and efficient food distribution, reducing waste and hunger simultaneously.

This architecture improves community welfare by connecting those who have surplus food with those who need it, leveraging automation and cloud technology for scalability and reliability.

Example - Solution Architecture Diagram:

(Illustrative architecture may include the following components)

- **Frontend:** Web & Mobile Interface (React, Flutter, HTML5, CSS3)
- **Backend:** Node.js / Python Flask API
- **Database:** MySQL / Firebase
- **Cloud Services:** AWS / Azure (for hosting and notifications)
- **External APIs:** Twilio (SMS alerts), Google Maps (pickup tracking)

Reference:

<https://aws.amazon.com/blogs/industries/voice-applications-in-clinical-research-powered-by-ai-on-aws-part-1-architecture-and-design-considerations/>