

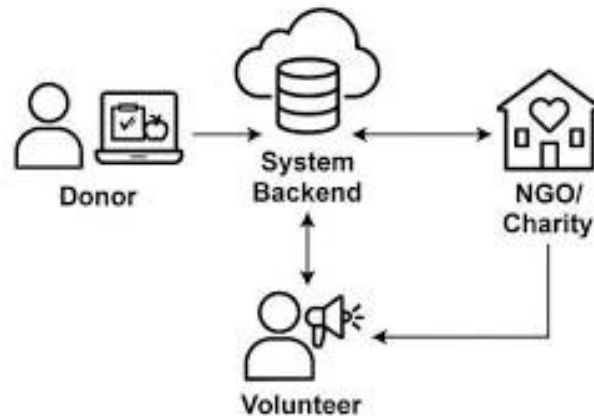
PROJECT DESIGN PHASE-II

TECHNOLOGY STACK (ARCHITECTURE & STACK)

Date:	1 November 2025
Team ID:	NM2025TMID01438
Project Name:	To Supply Leftover Food to Poor
Maximum Marks:	4 Marks

Technical Architecture:

The system helps collect leftover food from restaurants, events, and households, then redistributes it to needy people using a web platform and mobile interface. The architecture consists of user registration, food donation requests, real-time location tracking, and delivery management.



Guidelines Followed:

- Includes all processes (as application logic/technology block)
- Provides infrastructural demarcation (Local/Cloud)
- Indicates external interfaces (third-party APIs like Maps, Payment Gateway, etc.)
- Indicates data storage components/services
- Indicates optional interfaces for AI-based demand prediction (if applicable)

Table-1: Components & Technologies

S.No	Component	Description	Technology
1.	User Interface	Donors, volunteers, and admins interact via a web and mobile dashboard.	ReactJS (Web), Flutter (Mobile)
2.	Application Logic-1	Manages donor registration, food listings, and volunteer assignments.	Node.js / Express.js
3.	Application Logic-2	Validates location, food expiry time, and nearby volunteers.	REST API, Google Maps API
4.	Application Logic-3	Sends real-time notifications to volunteers and NGOs.	Firebase Cloud Messaging
5.	Database	Stores donor info, food details, volunteer data, and delivery records.	MongoDB / MySQL
6.	Cloud Database	Centralized cloud-hosted backend for all data operations.	AWS / Firebase Cloud
7.	File Storage	Stores uploaded food images, reports, and donor verification proofs.	AWS S3 / Firebase Storage
8.	External API-1	Google Maps API for location tracking and navigation.	Google Cloud Services
9.	External API-2 (Optional)	NGO verification via government portal API.	REST API Integration
10.	Machine Learning Model (Optional)	Predicts high-demand areas for food donation.	Python (TensorFlow / scikit-learn)
11.	Infrastructure (Server / Cloud)	Hosted on scalable cloud architecture for reliability and uptime.	AWS EC2 / Firebase Hosting

Table-2: Application Characteristics

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Built using open-source front-end and back-end technologies.	ReactJS, Node.js, MongoDB
2.	Security Implementations	Includes role-based access, data encryption, and secure login.	JWT Authentication, HTTPS
3.	Scalable Architecture	Supports increasing numbers of donors, NGOs, and volunteers.	Microservices Architecture
4.	Availability	Cloud hosting ensures continuous system availability.	AWS / Firebase Cloud
5.	Performance	Fast response via optimized queries and caching.	Redis Cache, Indexed DB Queries