Project Design Phase-II Technology Stack (Architecture & Stack)

Date	02 November 2025
Team ID	NM2025TMID04225
Project Name	To supply leftover food to poor
Maximum Marks	4 Marks

Technical Architecture:

The deliverable includes the architectural diagram representing the process of connecting donors, volunteers, and receivers through a centralized digital platform.
This system ensures safe, efficient, and transparent redistribution of leftover food. It integrates a web and mobile interface**,** real-time database**,** and notification services to coordinate the process end to end.

Example: Food Redistribution and Donation System using Cloud Integration

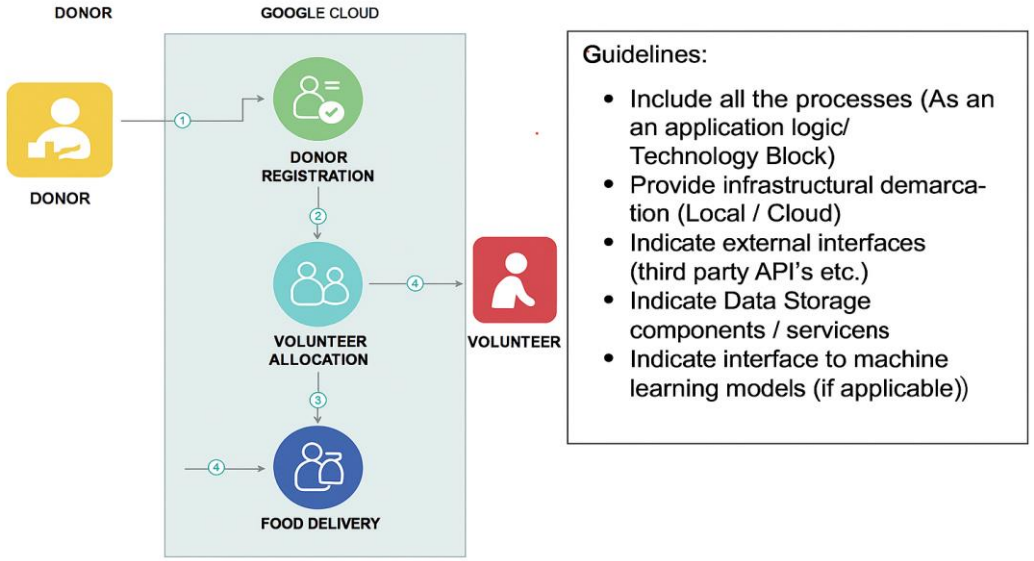

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1	**User Interface**	Donors, Volunteers, and Receivers interact through a unified web/mobile platform	React.js (Web), Flutter (Mobile)
2	**Application Logic-1**	Manages donor registration and leftover food uploads	Node.js, Express.js
3	**Application Logic-2**	Allocates nearest volunteers for pickup using geo-location	Google Maps API, Firebase Cloud Functions
4	**Application Logic-3**	Sends notifications and updates on food collection and delivery	Firebase Cloud Messaging (FCM), Twilio
5	**Database**	Stores donor, food, volunteer, and receiver details	Firebase Realtime Database / MySQL
6	**Cloud Database**	Ensures secure storage and synchronization	Google Cloud Firestore
7	**File Storage**	Used for uploading food images and verification proofs	Firebase Storage / AWS S3
8	**External API-1**	Map integration for distance and location tracking	Google Maps API
9	**External API-2 (Optional)**	NGO registration verification via government API	REST API Integration
10	**Machine Learning Model**	Predicts volunteer availability and delivery time (optional)	TensorFlow Lite (future enhancement)
11	**Infrastructure (Server / Cloud)**	Hosted on scalable cloud infrastructure	Google Cloud Platform (GCP) / AWS

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1	**Open-Source Frameworks**	Application built using open-source frameworks	React.js, Node.js, Flutter
2	**Security Implementations**	Secure authentication and role-based access	Firebase Auth, OAuth 2.0
3	**Scalable Architecture**	Cloud-based scalable backend supporting real-time sync	Google Cloud, Firebase
4	**Availability**	High availability via load-balanced servers	GCP Load Balancer
5	**Performance**	Optimized data queries and asynchronous operations	Cloud Functions, Indexed Database