

# **Software Design & Construction**

## **Project Proposal: ReliefTrack – AI-Powered Smart Ration & Relief Distribution System**

### **1. Executive Summary**

ReliefTrack is a comprehensive, cloud-integrated web application designed to modernize and streamline disaster relief distribution. During crises, traditional manual tracking leads to transparency issues, inventory mismanagement, and delayed aid. ReliefTrack solves this by providing a dual-portal system: a **Seeker Portal** for affected families to check their balances and a **Staff Portal** for relief workers to manage inventory and registration.

The system is built on a high-performance **AWS-based architecture**. It utilizes a **Smart DB Proxy** to optimize data storage: structured data like family profiles and inventory are managed via **MySQL** on an **EC2 instance**, while high-frequency audit logs are stored in **Amazon DynamoDB** for serverless scalability. Additionally, a **Gemini AI Assistant** is integrated to provide compassionate, real-time guidance to users.

### **2. Introduction**

In the aftermath of natural disasters, the rapid and equitable distribution of essential resources—such as food, water, and medicine—is critical. Existing systems often struggle with real-time updates and accessible communication, leaving many vulnerable populations without clear information. This project leverages modern cloud computing and AI to bridge the gap between relief organizations and those in need, ensuring that no family is left behind due to administrative inefficiency.

### **3. Problem Statement**

Current relief distribution mechanisms face three primary challenges:

- **Lack of Transparency:** Families often do not know what resources are available or if they have reached their allocated limit.
- **Inventory Silos:** Manual tracking at individual relief centers leads to stockouts in some areas and surpluses in others.
- **Communication Barriers:** Standard portals can be difficult for stressed users to navigate, leading to confusion and slower aid delivery.

### **4. Aim & Objectives**

The primary **aim** of this project is to deploy a scalable, AI-driven relief management system on AWS that ensures transparent and efficient aid delivery.

#### **Key Objectives:**

- **Cloud Deployment:** Host the backend API on an **AWS EC2** instance to ensure 24/7 availability.
- **Hybrid Database Strategy:** Implement a **Smart DB Proxy** that routes critical assets (Inventory/Families) to SQL and high-volume logs (Transactions) to NoSQL (**DynamoDB**).
- **AI Integration:** Deploy a **Google Gemini-powered** chat assistant to assist users in both the Seeker and Staff portals.
- **Real-Time Tracking:** Provide instant inventory updates across multiple relief centers using a centralized database.

## 5. Proposed AWS Architecture

- **Compute:** **AWS EC2** (t2.micro) running a Node.js/Express backend.
- **Storage (Relational):** **MySQL** hosted on the EC2 instance for structured data (Families, Inventory).
- **Storage (NoSQL):** **Amazon DynamoDB** for serverless transaction logging to ensure high availability and auditability.
- **Security:** **AWS IAM Roles** for secure backend-to-DynamoDB communication and **Security Groups** to manage traffic on ports 3000 (API) and 3306 (DB).