

IntelliHack 2025 Final Round Challenge

Challenge Title: AI-Powered Disaster Response Coordination Web App

Problem Statement

Natural disasters such as floods, earthquakes, and wildfires strike communities worldwide, often overwhelming local response systems. Effective disaster response requires rapid coordination of resources, real-time communication, and actionable insights from diverse data sources (e.g., text reports, images, and videos from affected areas). However, many disaster response systems lack the ability to intelligently process multimodal data, prioritize tasks, and provide user-friendly interfaces for volunteers, first responders, and affected individuals.

Your challenge is to build a web application powered by an AI-agentic workflow that enhances disaster response coordination. The app should leverage Machine Learning and AI to process and generate natural language responses, with bonus points for integrating vision-language models (VLMs) to analyze images. The app should solve a real-world problem by enabling efficient coordination, resource allocation, and communication during a disaster scenario.

Use Case

Disaster Response Coordination Platform

The web app, called [Name of your application], will serve as a centralized platform for disaster response coordination. It will assist four primary user groups:

1. **First Responders:** Receive prioritized tasks, analyze incoming reports, and coordinate resource deployment.
2. **Volunteers:** Register, receive assignments, and report observations from the field (via text or images).
3. **Affected Individuals:** Submit requests for help (e.g., food, shelter, medical aid) via text or images and receive updates.
4. **Government Help Centre:** Administer the platform and oversee a centralized coordination dashboard.

Core Functionalities

1. Request Intake, Task Prioritization, Resource Tracking, and Assignment:

- Users (affected individuals or volunteers) can submit help requests or field observations via text, images, or optionally voice.
- Use an ML or AI to parse text inputs, extract key information (e.g., location, urgency, type of need), and categorize requests.
- Track available resources (e.g., food, medical kits, vehicles) and update their status as tasks are completed.
- Prioritize requests based on urgency, location, and resource availability using AI agent workflow.
- Assign tasks to first responders or volunteers and provide step-by-step instructions.
- **Main Workflow for Model Selection:** Build a decision-making pipeline to determine whether to use a custom model (e.g., rules-based or lightweight on-device models) or a general-purpose LLM depending on the task, urgency, and context.

2. Real-Time Coordination Dashboard

- User-friendly dashboard for first responders to view prioritized tasks, real-time updates, resource availability, and incoming reports.
- Optional map-based interface showing geolocated help requests, ongoing missions, and deployed teams.
- **Evaluation Metrics:** Include a metrics system to measure AI response time, request fulfillment efficiency, and user satisfaction.

3. Communication Hub

- Real-time communication channel among all user roles (volunteers, responders, affected individuals).
- Powered by an LLM for task clarification, dynamic instructions, and natural language responses.

4. On-Device Processing and Synchronization

- Recognizing that users may be offline during disasters, design the system to support **on-device request generation and temporary local storage**.
- When the device regains internet access, automatically **sync stored requests** with the central platform.
- May use **Agent-to-Agent (A2A) protocol** or similar techniques to maintain state and ensure data consistency.

System-Wide AI Considerations

- **AgentOps Integration:** Integrate AgentOps for observability, monitoring, and dynamic updates to AI agents.
- **Guardrails:** Implement safety and compliance checks to ensure AI outputs remain aligned with ethical standards and emergency protocol guidelines.
- **Cost Effectiveness:** Optimize resource allocation and AI workload balance to reduce computational and operational costs.

Technical Requirements

- **AI Agentic Workflow:** Implement an intelligent agent system that autonomously processes requests, prioritizes tasks, and communicates with users.
- **Web Application:** Mobile responsive web application.
- **Data Handling:** Use synthetic or open-source datasets (e.g., Kaggle, Open Images) to simulate disaster scenarios.
- **Security:** Ensure privacy, security, and integrity of data.

Bonus Features

- **Vision-Language Integration:** Use VLMs (e.g., CLIP, LLaVA) to extract actionable insights from uploaded images.
- **Multimodal Input Handling:** Support text, image, or voice inputs in communication or reporting interfaces.
- **Expose an MCP:** Facilitate structured communication between agents and external LLMs or VLMs.
- **Disaster-Specific Features:** Custom interfaces or workflows for specific disaster types (e.g., wildfire heat maps, flood depth analysis).

Evaluation Criteria

1. Functionality – 30%
2. AI Agentic Workflow – 20%
3. Innovation & Bonus Features – 15%
4. Cost Effectiveness & Practicality – 10%
5. User Experience – 15%
6. Code Quality & Documentation – 10%

Submission Guidelines

- Submit a GitHub repo with:
 - Source code for the frontend and backend
 - README with setup instructions, AI workflow explanation, and assumptions made
- Include a 10-minute demo video (technical walkthrough)
- Submit a short 1–2 page report outlining:
 - Problem tackled
 - AI technologies used
 - Architecture and agent workflow
 - Challenges faced

Deadline: May 31, 2025, 11:59 PM

Why This Matters

Disasters impact millions every year. With intelligent systems for coordination and communication, lives can be saved, and communities can recover faster. Build a tool that makes a real difference.