Initial Thoughts & Design

Problem Statement
A non-profit organization has requested to build a software application that will help manage and optimize their volunteer activities. The application should help the organization efficiently allocate volunteers to different events and tasks based on their preferences, skills, and availability. The application should consider the following criteria:

Volunteer's location
Volunteer's skills and preferences
Volunteer's availability
Event requirements and location
Task urgency and priority

The software must include the following components:

Login (Allow volunteers and administrators to register if not already registered)
User Registration (Initially only username and password, followed by email verification)
User Profile Management (After registration, users should log in to complete their profile, including location, skills, preferences, and availability)
Event Management (Administrators can create and manage events, specifying required skills, location, and urgency)
Volunteer Matching (A module that matches volunteers to events/tasks based on their profiles and the event requirements)
Notification System (Send notifications to volunteers for event assignments, updates, and reminders)
Volunteer History (Track volunteer participation history and performance)

Initial Thoughts
Consider user experience: How will users (volunteers and administrators) interact with the application?
- The application will have two types of users: volunteers and administrators. The design must ensure ease of navigation for both user types. For Volunteers, we all agreed the user flow will focus on quick registration, profile completion, and accessing their matched events or tasks.
- For administrators, the flow will prioritize event creation task assignment, and volunteer management (Scale to match).
- Background Check

Identify the key functionalities: What are the essential features the application must have?

- Login and Registration: Secure user auth
- Profile Management: Enable users to update location/skills
- Location Geo: Enable user to update location and matching location
- Event Detail/ Management: Allow admin to create events with detailed requirement
- Volunteer Matching: Automate the process of matching volunteers
- Notification System: Send automated notifications
- Volunteer History: Maintain a detailed history of volunteer activities

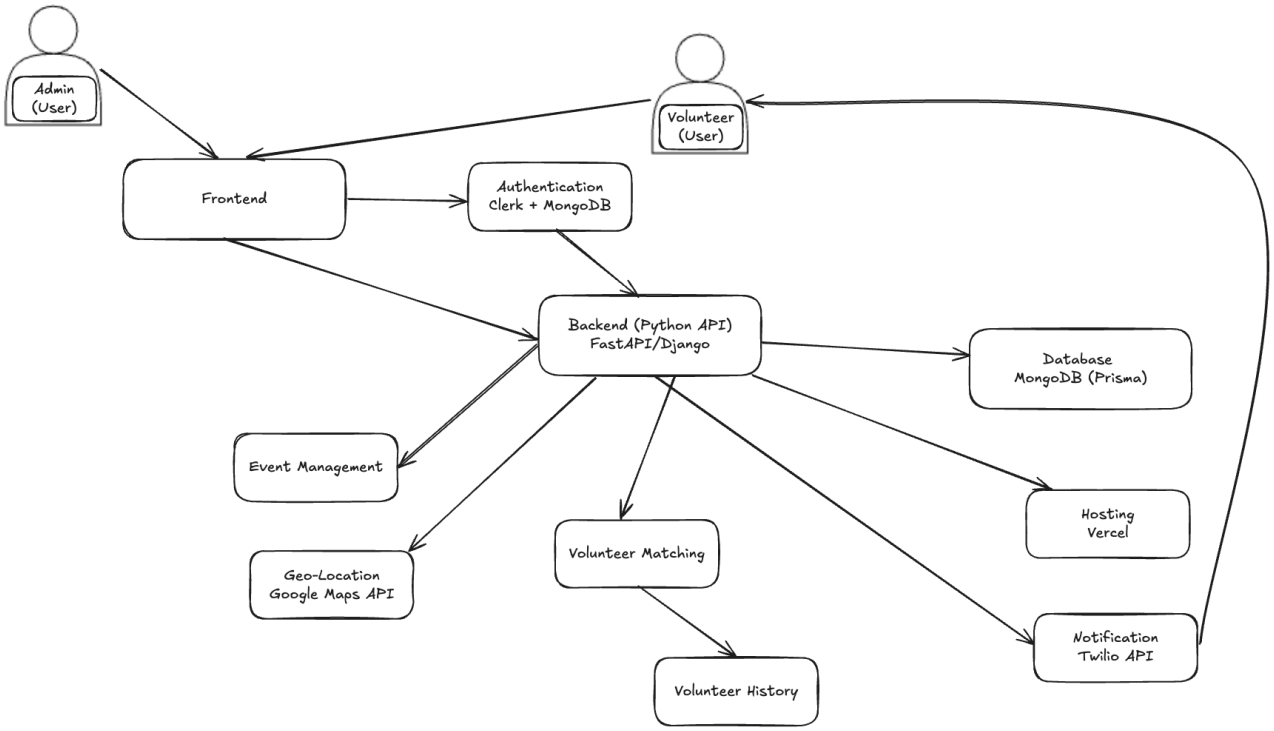
Technology stack: What technologies might you use for front-end, back-end, database, and other components?
- **Frontend:** Next.js (React (web applications)
- **Backend:** Python application logic and API endpoints
- **Database:** Prisma's integration with Mongo DB
- **Authentications:** Clerk integration with MongoDB (Store all user registration), ensure we have code, Google, MFA, email verification
- **Notifications:** Twillio
- **Hosting:** Vercel

Development Methodology
Explain why you would choose a particular development methodology (e.g., Agile, Waterfall, DevOps).

Chosen Methodology: Agile Development
- Why Agile?
1. **Respond to Change Quickly**: Agile prioritizes adaptability over rigid planning, enabling teams to adjust to evolving requirements or unforeseen challenges.
2. **Enhance Collaboration**: Agile fosters close communication between developers, testers, and non-profit organizations ensuring alignment on goals and priorities.
3. Improve Quality: Continuous testing and integration ensure issues are identified and resolved early, reducing the risk of major defects.

Discuss how this methodology will help manage the project effectively.
- Sprint cycles of 2 weeks, focusing on delivering specific features (user reg, event managment)
- Regular meetings for feedback
- Continuous testing to ensure the application remains functional

High-Level Design / Architecture
Create a diagram to illustrate the overall structure of your application.



Identify the main components (e.g., front-end, back-end, database).
Describe how these components will interact with each other.
- Volunteer register and verify using Clerk Auth
- After Logging in, the volunteer completes their profile
- An Administrator creates an event
- The backend marches volunteers
- Volunteers accept the assignment
- System tracks their participation

Mention any third-party services or APIs you plan to integrate.

Fill in this table, and provide as many details as possible:

Group Member Name	What is your contribution?	Discussion Notes
1. David	Gave ideas about the frontend development part and the database (third-party services) will be easier. I also gave ideas on the functionality features and worked on the high-level design.	
2. Ayo	Architecture design Pipeline, description of functionality of the application, and development methodology	
3. Afoma	Contributed to brainstorming essential features for the volunteer platform, including location implementation and event details/management	
4. Summer	Contributed to general UI design ideas. Discussed potential admin/volunteer view features.	