Assignment 1

Gian Alfonso
Rohan Kancherla
Karish Surani
Justin Tran

Description

In this assignment we will come up with an initial design for a software application that you will build in this semester.
We will not be writing any code in this assignment, but only looking at some initial design ideas and high-level design.

Answer these Questions

Initial Thoughts *(2 Points)*

Discuss your initial thought in detail on how you will design this application.

- **Consider user experience:**
    - How will users (volunteers and administrators) interact with the application?

Both administrators and volunteers start by registering an account. After completing that, volunteers can customize their portfolios. This includes providing their location, skills, preferences, and availability. Administrators have the ability to create and manage events. Once events are created, volunteers will be matched to them based on the requirements. Volunteers will also be notified about event assignments, updates, and reminders. They will also have access to their participation history and performance.

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

The application must include a user authentication system that supports registration from both volunteers and administrators. It should have a profile manager that allows users to specify their location, skills, preferences, and availability. Administrators must also be able to create and manage events. There should be a matching system in place that matches volunteers based on their profile data and event requirements. Another core feature is a notification system that alerts volunteers of their assignments, changes, and reminders.

Lastly, the application should track the volunteers participation history and performance over time.

- **Technology Stack:**
    - What technologies might you use for:

Front-end	• HTML • JS • CSS
Back-end	• Node.js • Rust
Database	• PostgreSQL
Other components +	• PostGIS • Napi-RS

Development Methodology *(2 Points)*

Discuss which development methodology you will use and why.

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

   [Review lecture 2: 2-software_dev.pdf from the *Course Contents (Notes)* Module]

We're tentatively choosing a spiral development methodology for our application.

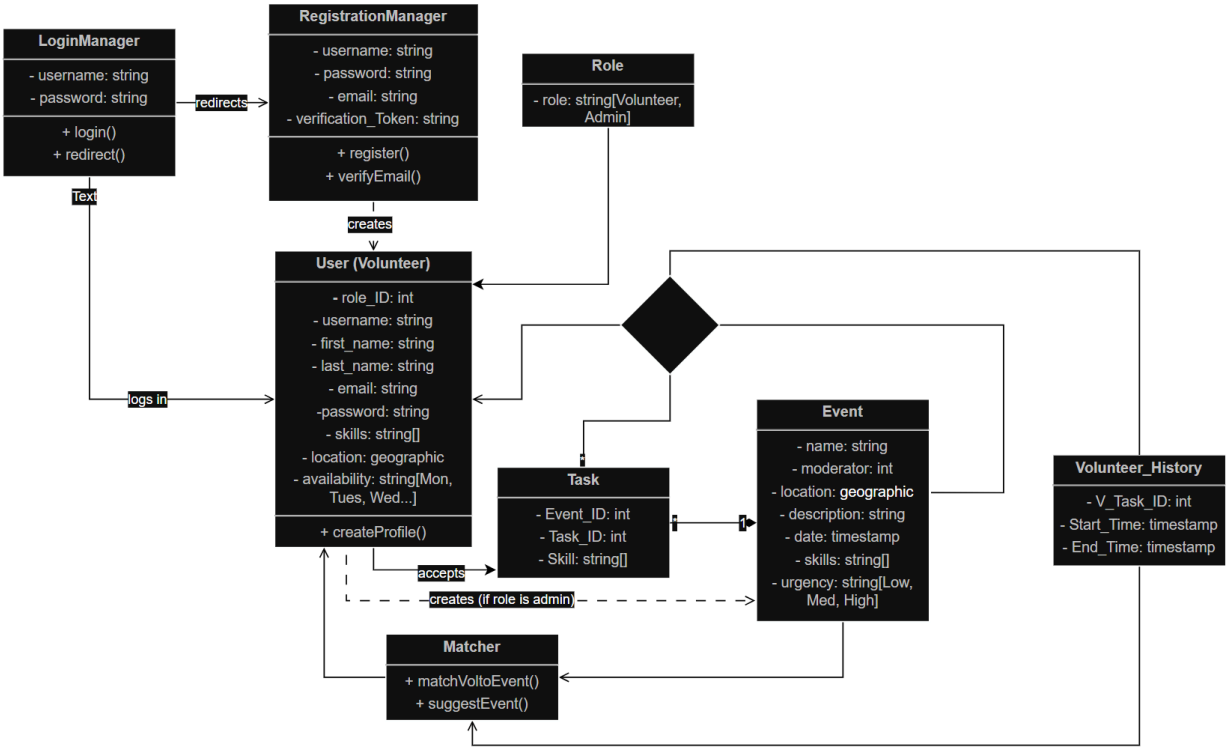
2. Discuss how this methodology will help manage the project effectively.

Software development is, in some ways, a cyclical process, and we will need to repeat several steps in the process many times over. The spiral methodology will accommodate this need better than the others.

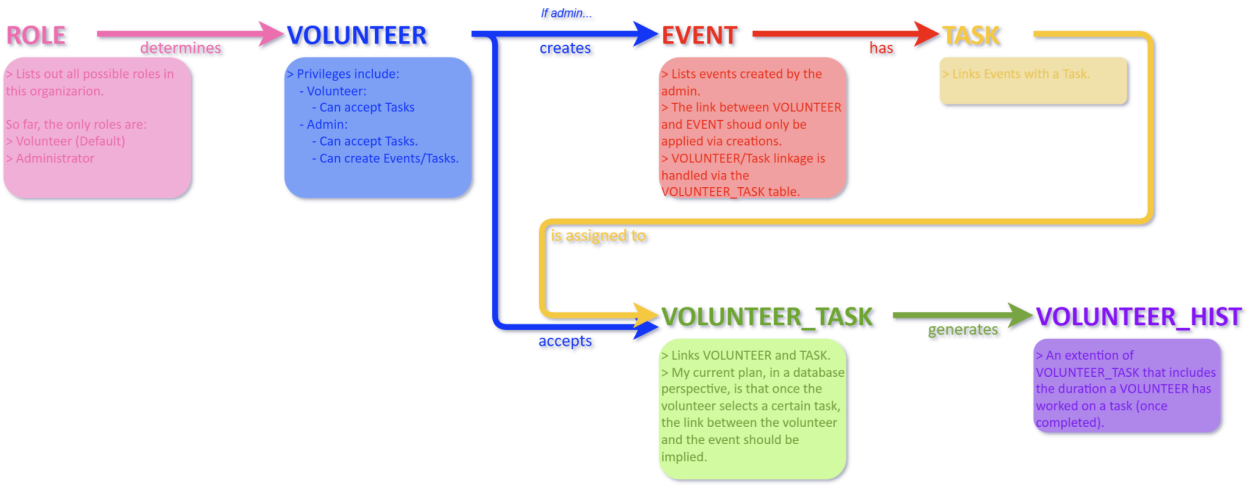
High-Level Design / Architecture *(6 points)*

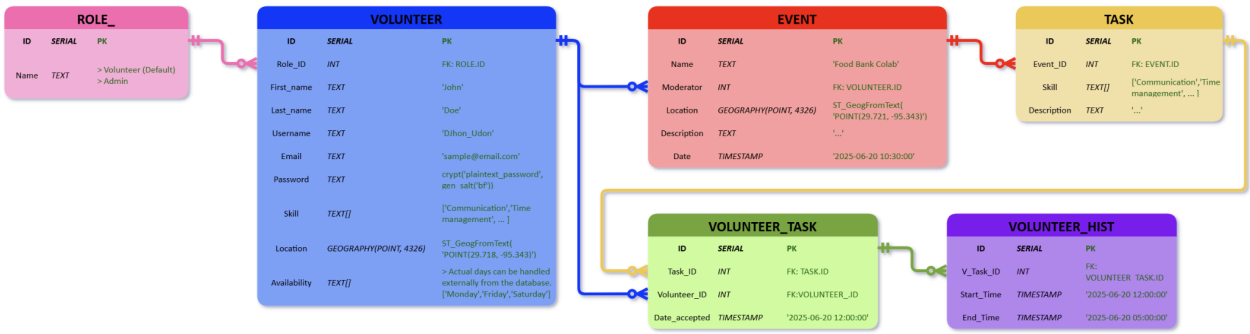
Provide a high-level design or architecture of the solution you are proposing.

1. Create a diagram to illustrate the overall structure of your application.
    - Application Structure



- Database Structure:





2. Identify the main components
    - Front-end
    - Back-end
    - Database

3. Describe how these components will interact with each other.

User-Side/Database

- When volunteers/admins register an account and customize their portfolios, the attributes (first name, last name, username, email, password, skills, location, and availability) should populate the USER relation.
    - USER.ID in the database is stored as a *SERIAL datatype*. No input is required for this attribute since *SERIAL* tells the database to autoincrement that attribute for each new row.
- At the moment, in the database side, the locations of events/volunteers are strictly handled as coordinates (global).
    - To accommodate for this, the user can enter their address which will then be converted to coordinates for the database.
    - PostGIS has implementation for geocoding functions.

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

- PostGIS is an extension of PostgreSQL that allows it to work with geospatial data. We were looking into ways to store user locations, and this seemed perfect.
- We will be using Napi-RS to integrate Rust with Node.js, since they are otherwise two separate languages

---

Resources for Architectural Diagrams

Educative | Software Architecture Diagramming and Patterns

Edraw | Software Architecture Examples and Templates