Volunteer and Donation Coordinator

Team: Smart Programming Baddies
Chidi Azuike
Evelyn Adair Martinez
Bryan Granda
Jason Alejandro Mendoza Montoya

Language: Java
Our GitHub repository has not changed.

Service Overview:

Our target demographic group is the homeless. We are looking to aid them by providing a donation and volunteering coordinator service to provide essential services and goods they need. This service will facilitate donation events where organizations can host events, sign up volunteers for these events, find recruitment, and donate goods if necessary. This project will also implement inventory tracking of essential items such as food, clothes, and toiletries.

We propose a service to streamline volunteering operations at donation events, which includes handling scheduling, sign-ups, and role assignments. Organizations can create and oversee events, tracking essential details such as event type, volunteer needs and volume, location, and time. The service also includes a database for inventory management, allowing efficient tracking of donations by category. Clients can subscribe to event notifications, storing their preferences for seamless future access. Key features include volunteer coordination, event hosting, and inventory tracking, ensuring smooth operations for volunteers and organizations.

- The API will have a system for maintaining volunteering operations at a donation event, including scheduling, sign-up, and role assignment.
- Organizations can interact with the API to create and host events, and this service will track the type of event, volunteering requirements, location, and time.
- This API will provide database storage functionality where organizations can store their inventory of goods and services.
- For the donation aspect, we would utilize a database to track and coordinate items in their respective categories.
- Clients will be able to sign up for subscriptions, and notifications of upcoming events will be stored within the database. The next time they retrieve their information, the events will be on the return data.

Our service also provides volunteer scheduling and personnel management features that allow our clients to track the number of volunteers available and the time they can volunteer. Also, different roles amongst the volunteers will be maintained so that it is known what the volunteer is working on. Organizations hosting events can use this feature to request the volunteering power that best suits their event. With this feature, our API will facilitate event hosting and management since an organization can set and request a personnel requirement and assign different roles to the volunteers. This will help ensure the event operations are all accounted for.

Clients will also be able to opt into a subscription feature. These notifications will be stored in an organization class, so when data is retrieved from these organizations, notifications of upcoming events will be provided for them.

Logging operations and transactions is business-critical for any service to ensure sustainability and efficiency. Our volunteer and donation service will also provide an accurate understanding of past and ongoing donation processes and volunteer processes. Our clients can interact with our API to retrieve past information such as donation transactions, previous or upcoming events, and past, current, and projected inventory of donation centers such as food pantries or shelters. Our service in providing these reports can aid in logistical coordination for various clients if they are interested and allow them to work effectively. We will maintain volunteer hours, history, and other relevant information.

Client Overview:

Our clients include food pantries, supermarkets, institutions, social workers, hospitals, and educational institutions.

Food Pantry Operations

- Use the service to manage inventory and ensure proper food distribution.
- Interact with the API to log food donations and track expiration dates.
- Interact with the API to plan and log all intended food drop-offs so that food pantry owners know what is expected.
- Use the service to locate and view the inventory of each food pantry nearby.

Social Workers

- Can use the service to view essential goods such as food donations and/or non-perishables that could be useful to need-based clients.
- Could use the service to help sign up volunteers who need to do community service to any type of donation events in request of volunteers.
- Can have notifications of upcoming events added to their database profile.

Supermarkets

- Supermarkets with charitable endeavors could use this service to donate food or toiletries, volunteer at events, and track their overall impact.
- Use API to check a goods donation center's inventory so they can determine which goods are most beneficial to donate.

Hospitals

- Healthcare workers can use this service to assist people who have food insecurity and have been discharged from the hospital. They can also use the service to view the availability of goods that could be useful to the neediest patients, such as clothes and food. This service could be utilized to host blood drives and get volunteers, such as scheduling events.

Educational Institutions

- Educational institutions can organize donation events where students can participate and receive school credits.
    - Students can volunteer at events to collect donations.
    - Students can volunteer and help organizations with a specific cause at these events.
- Educational institutions can also host or direct donation services to students who lack necessities.

- Provide students with access to get involved in volunteering activities, utilizing the API to view volunteer capacity and sign up for volunteering opportunities.

Testing Overview:

For testing, we plan on utilizing various forms of testing to ensure quality release. Those forms of testing include **unit, integration,** and **load testing.** Unit tests allow us to verify the functionality of independent components, ensuring they work as expected in isolation. Integration tests to ensure that different components of the API work together as expected. Load testing to observe how the API deals with varying data loads (low/high usage). Through exhaustive testing, we can assess the performance of our service. **This first iteration will ensure robustness and functionality without clients.**

Unit Testing

- Relational Database
    - Ensure that **CRUD** operations work
- Auth
    - Managing registration, login/logout, and capabilities of users
- Component Testing
    - Verifying operations are performed correctly
- API Testing
    - Verify that (**put post get delete**) requests are functional.
        - Endpoints are functional
    - Ensure handling of invalid requests.
        - Missing parameters
        - Invalid request format
        - Invalid data
        - Unauthorized
- Methods for all Unit Testing (Robustness)
    - Verify the Happy Path works
    - Verify that intended behavior occurs for edge cases
    - Verify that intended behavior occurs for invalid cases
    - Verify that implemented security systems work as intended
    - Verify the responsiveness of our system.

Integration Testing

- Verify successful interaction between modular components.

Load Testing

- Ensure Performant Database
- Identify load breakpoint of API
    - Stress Testing

Our test plans aim to ensure our service is robust and functions as expected. How we populate our data is tentative, likely a mock database and we will make use of existing/preliminary logging in our testing process. For **unit tests**, we plan on using JUnit, the prototypical unit testing framework for Java, and

Mockito, a framework for mocking dependencies. Postman will also be used to test our endpoints manually. We plan on using JUnit and Spring together to perform **integration tests**. We also set up CI on GitHub to run integration tests on our codebase whenever changes are made. For **load tests** and **logging**, we can opt for open-source APIs. We also plan on assessing **reliability**.