Full stack engineer assignment (Logs Dashboard)

Overview

Assignment: Logs Dashboard

Create a web-based application that allows users to manage logs, and an analytical dashboard that displays interesting metrics about the logs (with at least 1 chart).

Requirements

The application should have the following features:

- Backend:

    - A REST API that serves data for the frontend to consume.

    - A relational database to store log data with columns such as timestamp, message, severity, and source.

    - Endpoint for performing CRUD operations on logs.

    - Endpoint for querying raw log data by date range, severity, or source.

    - Endpoint for querying aggregated log data by date range, severity, or source.

- Frontend:

    - A log list page that displays a list of logs with the ability to search, filter, sort, and paginate the logs.

    - A log detail page that will display the details of a selected log and allows users to modify or delete it.

    - A log creation page that allows users to create new logs.

    - A dashboard that displays aggregated log data for the selected date range, severity, and source.

- A filter panel that allows users to select a date range, severity, and source to view log data for.
- A chart that displays the trend of log counts over time for the selected severity or source.

Bonus ideas

Feel free to implement any bonus ideas you have or mention them in the README.

For example:

- A feature that allows users to download the log data as a CSV file.
- A feature that displays a histogram of log severity distribution for the selected date range and source.
- Unit or integration tests

Tech stack

- A python-based web framework like Flask, Django, or FastAPI to create the REST API
- A relational database, such as Postgres to store the log data
- A React or NextJS frontend application

Submission

1. The project doesn't need to be deployed, as long as we can easily run it locally to test the functionality (such as via docker-compose or instructions in README)
2. The design is completely up to your preference
3. The code can either be published on GitHub or sent to us via a ZIP file

Final thoughts

Remember to use best practices like input validation, error handling, and logging in mind when building your application. Please use a README including documentation about how to run/test the application, convey your thoughts on

the project and implementation details such as design decisions and technical libraries chosen.

Good luck with your assignment!