Report on Assignment 2

Monitoring.

In terms of monitoring the system I created in the assignment has both centralized logging using Seq and centralized tracing using Zipkin. All traces bring with them relevant information like which service they are coming from and of course when cross communicating, I am using the propagation class to send tracing information in the headers of the requests.

Sustainability

One sustainability improvement that I can suggest is that the server on which this system is running can be scheduled to turn off every evening and turn on every morning during business hours. I can recommend it since the software is going to be running only in Denmark and can be expected not to be needed during the night.

The frontend can also be improved to use state management, so it doesn’t fetch data constantly but just once and then updates the local state mainly. For example, you fetch ones all patients and then if you send a delete request, instead of fetching the updated list of patients on successful response we can just update the client state assuming it is the same as in the database.

Design to be disabled.

In terms of design to be disabled I have two features designed to be turned on and off since they do not have old versions. Those features are the same as the suggested Patient DELETE and Patient POST. Sadly, I didn’t had time to implement unavailability message to patient outside of Denmark.

I don’t use FeatureHub but rather have the activated using HTTP request.

Automation over people

I attempted to build my CI/CD pipeline however while working with Jenkins I had too many issues some of which I couldn’t resolve. I still wrote a JenkinsFile with two build steps, one for building the images and one for uploading them to Docker Hub.

Culture

In terms of culture since I worked alone, I cannot say that much of it was implemented. But I can quickly describe the differences.

The Pathological is the one where people are power-oriented and do not communicate much. In such corporations people are blamed for bad news and they do not want to take responsibilities.

The Bureaucratic is where there are rules established and you are not supposed to break them. As an example, I can say that a lot of government organizations have that culture. Here communication is a bit more tolerated compared to Pathological culture.

The best one in terms of performance is the Generative one. There the focus is the performance and error handling. It is where communication is encouraged and people bringing bad news are trained.

Docker

Running “docker compose up” will bring the whole system up in my project.