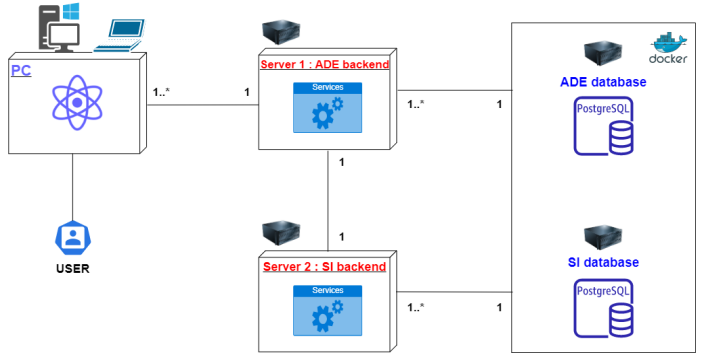
Architecture

In the first layer, users use their PC to access the platform. An interface is displayed and will allow them to interact with the application while requesting the backend server.



The second layer contains the business logic of the application and is invisible to the users. The ADE backend server will receive client requests, process them and return responses to users, which makes possible the user interaction with the application interface. The SI backend is part of the project that already existed and was developed by an intern, but it does not intervene in the business logic of our application. On the other hand, this server contains important data that the ADE backend needs when processing user requests. For this, a communication has been established between the ADE backend server and the SI backend server in order to retrieve data when necessary instead of duplicating them in the ADE backend.

The third layer contains the databases related to the ADE backend and SI backend.

Your daily schedule !

Context

Polytech Grenoble is an engineering school currently using ADE as a scheduling tool. ADE is a private service and there is no way to access and manage the timetable by Polytech Grenoble staff. This means that each time a teacher for example needs to modify the timetable, he/she will have to send an email to the Polytech manager in charge of the communication with the ADE administrator, who in turn will send a request to this administrator to make the changes. The procedure is therefore complicated, inefficient and time consuming. The project focuses on the development of an alternative to ADE.

Description

The prototype will consist of two parts : a backend extension of a previous Python-based project using FastAPI and SQLAlchemy, and a frontend component that may use React and display calendars using FullCalendar.

FastAPI

SQLAlchemy

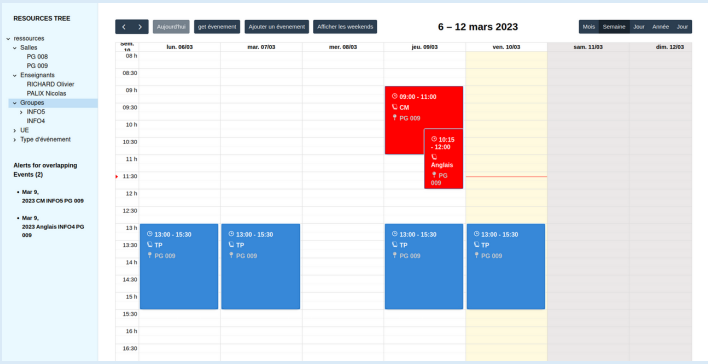
PostgreSQL

React

docker

ALEMBIC

Graphical User Interface



- Our Graphical User Interface (GUI) allows users to schedule and manage events easily and provides a customizable calendar that displays events in different views such as day, week and month.
- The app can then detect any overlapping events and alert the user if there is a conflict (red events and the overlapping events list on the sidebar). This feature helps to prevent double booking of rooms or teachers and ensures that events are scheduled at appropriate times.
- The GUI may also have filters or search options to help users find specific events based on various criteria such as the room, teacher, group, or event type. Users can view the events in different colors or categories to easily identify the type of event or group it belongs to.