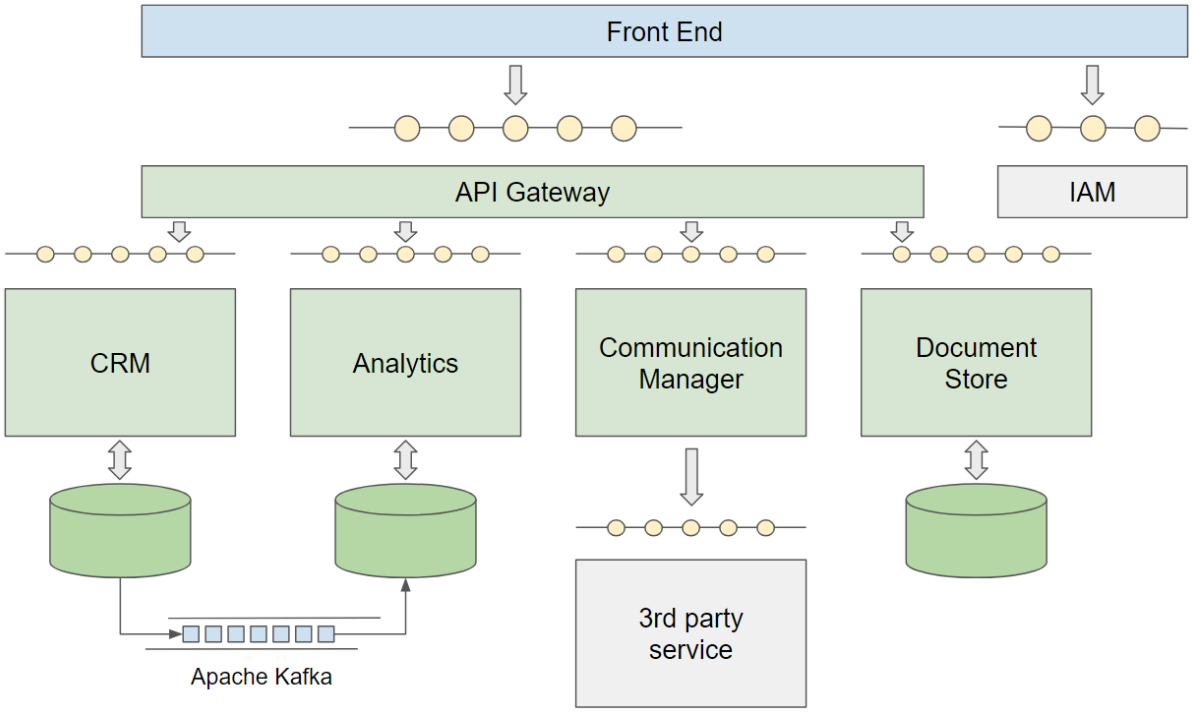
A system for temporary job placement services

Final project – A complete working system

For the final project, you have to complete and consolidate the work done during the five labs, resulting in a locally running system based on containerized microservices. This approach ensures adequate scalability and availability.

The system must include a user-friendly front-end that provides seamless access to all required and offered functionalities.

You can expand the system outlined during the labs, adding any features or components you find valuable for the proposed case. In any case, it must meet the following description and final requirements.



Description

A company provides temporary job placement services. It relies on a Customer Relationships Management tool to streamline its operations as well as improve its effectiveness and efficiency.
This tool acts as a centralized platform to manage and enhance interactions with candidates, clients, and other stakeholders involved in the hiring process by providing a centralized hub for managing candidates, job offers, and client relationships. It helps recruiters and HR professionals to work more effectively, make data-driven decisions, and ultimately improve the overall quality of placements.

Using this tool, through a user-friendly web-app interface, **recruiters** working for the company can perform the following actions:

1. Candidate Management:
    - Store and organize candidate profiles, resumes, and relevant information
    - Easily search and filter candidates based on skills, experience, and other criteria (e.g., their location)
2. Customer Relationship Management:
    - Maintain client profiles with details such as job offer requirements, preferences, and communication and placement history
    - Create, manage, and track the status of job offers within the system
    - Follow the progress of candidates through the recruitment pipeline

Customers (companies or individuals searching for personnel to be temporarily employed) and **Professionals** (people looking for a temporary job) interact with the company operators. The system should support operators performing the following actions:

3. Registration:
    - Create a profile for a customer or a professional and provide a set of basic information about himself as well as a contact point (email address or telephone number) to be used by the recruiters to get in touch with them
    - Access and maintain the profile information.
4. Job Offer:
    - Create an initial description of a job offer, providing the basic requirements. The offer can be further detailed by recruiters who will contact the customer and run a more structured interview in order to collect all relevant information.

    - Verify the current state of the job offer, as it progresses through the candidate selection phase (how many potential candidates have been contacted so far? How many declined? Who was selected for the placement?)
    - Access and maintain professional resumes as well as their current availability/employment state
5. Job Proposal:
    - Review the details of a proposed job and access relevant information (work contract proposal, public profile of the employer, ...)

The system also provides the following functionalities to **managers** and other stakeholders:

6. Analytics and Reporting:
    - Generate detailed reports on recruitment metrics, performance, and key insights (purpose: analyze data to improve recruitment strategies, track KPIs, and make data-driven decisions)

General purpose functionalities:

7. Document Management:
    - Centralize storage for contracts, resumes, and other important documents
    - Maintain version control and track document changes within the CRM system
8. Security and Compliance:
    - Ensure the security of sensitive candidate and client information through robust security measures.
9. (Optional - Observability and Monitoring stack with Grafana, Loki, Prometheus):
    - Collect comprehensive system metrics such as API usage, response times, error rates, and other critical performance indicators
    - Aggregate and store logs from different microservices to provide a unified view of system health and performance
    - Dashboard to visualize these metrics in real-time, allowing developers and managers to monitor system performance and quickly identify and address issues

Evaluation

Once the system is ready, please contact us to schedule a presentation.
You will present your system and discuss the technical implementation you chose.
Up to 5 points will be assigned to the project.