# Task 1

Imagine that you have one monolithic platform where you can register new users. Every time you register a new user, the data is persisted into a database, and an email and an SMS are sent to the user indicating registration is completed. Now we want to transform this into a microservice structure.
Create a diagram/use case to explain what's the better solution to accomplish this.

# Task 2

We want to have an API to register and manage jobs. We have two types of users (regular and manager). The job has a little title (max 100 chars) and a description.
The regular user is only able to see, create and update his jobs. The manager can see all tasks. When a new job is created, the managers should be notified.

## Features

- create two API endpoints to save a new job and another one to list jobs
- notify the manager every time a job is created. This notification should not block any HTTP request

## Requirements

- use docker to create a local environment for this service with a MySQL database to persist data
- all features should have at least unit tests to guarantee that everything is working as expected
- use a message broker to decouple the notification system from the rest of the logic (bonus)
- create Kubernetes file needed to deploy this service (bonus)