# Linki:

<https://github.com/gzub04/ERSMS-project>

# 26.02.2024

Rzeczy do zrobienia na za tydzień:

* ~~Znaleźć Alkaratusa~~
* ~~Zapisać grupę~~
* ~~Napisać do kruka -> najlepiej przyklepać temat~~
* ~~Github repo założyć~~

Co wiemy:

Skład drużyny

| **Rola** | **Osoba (w kolejności od najbardziej do najmniej)** |
| --- | --- |
| Leader | kuhyx |
| Developer | Olhado, michal\_sar, Gzub, Alkaratus |
| System Architect | Alkaratus |
| Tester | Gzub |
| Presentation Producer | michal\_sar |
| Librarian | kuchy/? Ktoś nowy |

**toDo:**

Idea of the project:

1. Gierka co zrobił ją Alkaratus (jakie 4 mikroserwisy?)
2. Movie Recommendations, ktoś musi założyć konto i wybrać filmy:
   1. Wyszukiwanie w bazie danych
   2. OAuth
   3. AI do rekomendacji
   4. (awaryjnie) Płatność za korzystanie

Terraform?

Olhado:  
 Ansible (małe doświadczenie)

Rest API (python flask)

Troooooooszkę JS-a

Uczelniane AI python

Gzub:

Testy w perlu/python

Trochę w pipeline githubów

Coś tam w dockerze

Uczelniane AI python

Michal\_sar:

Data bricks

SQL

MSSMS

Proste Rest API w **fast api**

Uczelniane AI python

Azure trochę

Frontend: React, trochę Next

Stack:

Do deployowania docker?

Backend w pythonie?  
 OAuth pythonowy

Bazka danych -> toDecide by michal\_sar

Ewentualny frontend w React

toWrite:

* Software:
  + Script to deploy
  + Control version -> git
  + Authorization and session management
  + Identities
  + Use oAuth
  + Create GUI if you want
  + Monitoring service, collecing logs
  + Risk analysis with threat model of the service (with mitigations)
  + Reliable -> any single part of solution may fail and the service will survive it -> replication, duplication, recovery
  + 3-4 microservices
* Documentation:
  + System architecture (choose standard and follow it)
  + Automated management solution
  + All in one file
  + Film for presentation