

# Integrantes

- Cesar Hideki Imai - 10402758
- André Franco Ranieri - 10390470

## Visão geral

Foi usado uma aplicação de análise de relação semântica com grafos desenvolvida no Jupyter Notebook, quando é obtida a versão do código .py, todos os métodos e classes são agrupados num único arquivo. A aplicação usada como referência pode ser acessada neste [link: https://github.com/Cehiim/TeoriaDosGrafos/blob/main/ProjetoFinal/ProjetoFinal.ipynb](https://github.com/Cehiim/TeoriaDosGrafos/blob/main/ProjetoFinal/ProjetoFinal.ipynb). A aplicação converte as palavras listadas no arquivo .txt para representações vetoriais numéricas com um *Large Language Model* (LLM) *Encoder*, importado através do pacote [SentenceTransformers](#). Dessa forma é medido a aproximação semântica entre as palavras por meio da busca de similaridade por cosseno. A partir disso é criado um grafo usando a biblioteca [PyVis](#), em que cada vértice representa uma palavra listada e cada aresta se refere a distância semântica entre duas palavras.

## Padrão empregado

Foi feita a decomposição funcional da arquitetura monolítica seguindo o padrão arquitetural de orquestração para refatorar toda a aplicação em módulos com responsabilidades únicas. Essa abordagem foi escolhida, porque ainda era necessário manter o fluxo de execução das chamadas de cada serviço para que a aplicação funcionasse corretamente. Portanto foi usado um *Controller / Enterprise Service Bus* (ESB) para administrar a ordem de chamada de cada função.

## Resultados

A refatoração contribuiu para encontrar trechos de código que precisavam de correções e otimizações. Também com a segmentação de microsserviços tornou a aplicação viável para a adição de novas funcionalidades.