**Desafio Sicredi**

**Autor: Everton Morais**

**O que é necessário para testar o Projeto?**

1. Docker
2. Postman

**Como testar o projeto?**

1. Todos os serviços estão configurados para subir via docker-compose: Kafka, MongoDB, Postgres, assembleia-service e votacao-service.
2. Acessar a pasta raiz desafio-sicredi via cmd.
3. Executar docker-compose up
4. Aguardar a subida dos serviços.
5. Importar a collection Desafio-Sicredi.postman\_collection no Postman, onde existem todas as chamadas de APIs configuradas.
   1. A collection se encontra na Raiz do Projeto.
6. Cadastrar 3 Associados.
7. Cadastrar uma Pauta.
8. Cadastrar uma Sessão de Votação, uma pauta pode ter várias sessões.
9. Registrar votos para uma sessão de votação.
10. Consultar o Resultado.
11. O Sistema encerra automaticamente a Sessão de Votação e registra o resultado.
    1. A verificação pelo encerramento ocorre a cada 1 minuto.

**Características**

O sistema possui dois serviços:

* **assembleia-service**
  + Este serviço utiliza banco de dados relacional Postgres e Kafka.
  + É responsável pelo gerenciamento de Associados, Pautas, Sessões de Votação.
  + Possui dois Schedulers:
    - Scheduler para verificar Sessões de Votação que tiveram o tempo esgotado, e passá-las para Em Análise.
    - Scheduler para Apurar a votação das Sessões de Votação e passá-las para Encerrada.
      * Neste momento ela irá receber um dos Resultados:
        + Aprovada
        + Não Aprovada
        + Empate
* **votacao-service**
  + Este serviço utiliza Spring WebFlux, Mongo DB e Kafka.
  + Ele foi pensado para permitir o Registro de votos em grande quantidade, e a possibilidade do escalonamento horizontal.
  + Este serviço é responsável pelo registro de votos e a contagem dos mesmos.
  + Para ganhar tempo no registro de votos, este serviço possui cache dos Ids de Associados, bem como a autorização para votação, possui também cache para as datas de Início e Fim de Sessões de Votação (CQRS). Desta forma não é necessária a consulta de outros serviços para o registro do Voto.
  + Os caches são atualizados utilizando Kafka, no cadastro e alteração dos registros no serviço assembleia-service.

**Observações**

Não foi possível utilizar a API citada, pois o serviço não está mais disponível. Apesar disso a integração foi implementada.

