

# DHT Service

Vinícius de Oliveira Campos dos Reis

11041416

## 1 O projeto

O projeto é uma implementação em Kotlin que utiliza o Gradle como compilador e gerenciador de dependências. A arquitetura de cada componente da aplicação é multi-módulos, seguindo ao máximo uma arquitetura limpa e orientada a domínios.

## 2 Vídeo

O link para o vídeo de apresentação e demonstração pode ser encontrado [neste link](#).

## 3 Como executar

Todos módulos neste projeto podem ser executados através do Makefile na raiz do projeto, cada um com seu target. Confira abaixo como cada um deles pode ser executado:

- Servidor DHT: *make dht*
- Sample - Cofre de senhas: *make sample\_vault*

## 4 Implementação

### 4.1 Servidor

A lógica principal do funcionamento do servidor está implementada na classe *DHTServiceServerGrpcImpl*.

O funcionamento do servidor é baseado no algoritmo de tabelas de hash distribuídas (DHT) em um formato de anel, como proposto para o projeto. O contrato implementado por esta classe é muito semelhante ao que foi proposto no enunciado.

A maior dificuldade encontrada no desenvolvimento foi criar testes unitários para validação e coordenar mais de dois nós na rede. Gerenciar diferentes chamadas via gRPC foi um desafio superado, pois os canais que são criados precisam ser fechados após o uso. Isso por que estes canais podem ser coletados pelo garbage collector caso fiquem inativos por longos períodos.

### 4.2 Sample - Cofre de Senhas (Key Vault)

Para demonstração do funcionamento da DHT, foi implementado uma aplicação simples que funciona como um cofre de senhas.

Esta aplicação implementa um cliente gRPC que se conecta ao servidor DHT e armazena senhas como pares de chave e valor.

Dado o funcionamento da DHT, a aplicação é capaz de armazenar, deletar e recuperar senhas de forma distribuída.