

# Publish/Subscribe Service

Vinícius de Oliveira Campos dos Reis

11041416

## 1 O projeto

O projeto é uma implementação principal em Kotlin (cliente secundário em Python) 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 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: *make server*
- Cliente Kotlin: *make client*
- Cliente Python: *make py.client*

Note que o servidor deve ser executado antes dos clientes para o correto funcionamento dos clientes.

## 3 Implementação

Da forma mais direta possível, veja a função das principais implementações da aplicação (note que removemos da referência o início comum entre elas, que é o pacote *io.github.vinicareis.pubsub.server*):

### 3.1 Servidor

- *core.grpc.service.QueueServiceGrpc*: Ponto de entrada do serviço gRPC. Implementação de classes geradas pela biblioteca oficial do gRPC, que delega a essa classe o que é feito em cada operação do servidor.
- *core.data.database.postgres.repository.QueueRepositoryDatabase*: repositório de filas ativas armazenadas num banco de dados.
- *core.data.database.postgres.repository.TextMessageRepositoryDatabase*: repositório de mensagens pendentes armazenadas num banco de dados.
- *core.data.database.postgres.repository.EventRepositoryDatabase*: repositórios de eventos pendentes, como a criação e remoção de filas e mensagens postadas. É válido destacar que, visando manter a máxima consistências, os eventos são salvos na mesma transação da operação que gera o evento em si.
- *core.grpc.service.SubscriberManagerServiceImpl*: serviço de gerenciamento de inscrições. Responsável por monitorar os eventos ocorridos no repositório de eventos e gerenciar filas de acordo com seu tipo, selecionando quais inscritos devem ser notificados de dados eventos.

### 3.2 Clientes

Os clientes, tanto em Kotlin quanto em Python, são parecidos tanto em implementação quanto em funcionamento. Estes são interativos via linha de comando, executando um laço onde o usuário seleciona uma das operações para ser realizada ou encerra a aplicação.