

Exercício Prático 2 – gRPC

Fundamentos em Sistemas Paralelos e Distribuídos

Daniel Souza de Campos – 2018054664 – 2021/2

1 – Objetivo do Documento

Essa é uma documentação extremamente breve e direta do trabalho citando pontos importantes e decisões de projeto.

2 - Pontos Importantes

- Foram utilizados os exemplos disponíveis no [tutorial básico de gRPC em Python](#) como base para o trabalho. Além disso, para desenvolver o arquivo `.proto` também foram utilizados os [tutoriais explicativos](#) e o [tutorial para compilação](#).
- Só existe um arquivo `.proto` em `./protos/keyValueStore.proto` a partir da pasta raiz do projeto.
- Todas as regras do `makefile` dependem da compilação do arquivo `.proto`. Dessa forma, todas as regras acabam realizando a compilação de acordo com essa dependência. Não consegui montar as regras de forma que essa compilação só se concretizasse caso os arquivos `*_pb2.py` e `*_pb2_grpc.py` já não existissem. O comando de compilação do `.proto` foi impedido de ser impresso na saída padrão ao usar o símbolo “@” no `Makefile`.
- Existem três arquivos principais:
 - `server.py`: Arquivo com o código dos servidores que armazenam os pares de chave-valor
 - `client.py`: Arquivo com o código do cliente
 - `centralServer.py`: Arquivo com o código do servidor central.
- A diferenciação do cliente e do servidor de armazenamento de pares sobre o comportamento da primeira parte e da segunda parte acontece pelo envio de um segundo parâmetro ao executar o arquivo Python.
- O servidor de armazenamento de pares possui duas versões do método de ativação. Para a primeira parte do trabalho, ele apenas retorna uma mensagem que contém uma flag igual a 0. Para a segunda parte do trabalho, ele também retorna uma flag que indica quantas chaves ele enviou para o servidor central.

- O servidor central só salvará as chaves dos 10 primeiros servidores diferentes que enviarem suas chaves. Os servidores são identificados pela sua *string* de identificação.
- Na primeira parte do trabalho, caso um cliente realize a consulta de uma chave que não exista em um servidor de armazenamento, será impresso na saída padrão uma *String* vazia seguida de uma quebra de linha.
- Na segunda parte do trabalho, caso uma situação semelhante a descrita a seguir aconteça, será impresso na saída padrão "*string* de identificação do servidor:*string* vazia"
 - Inserir um par (1, valor1) em um servidor de armazenamento
 - Mandar o servidor de armazenamento enviar suas chaves para o servidor central
 - Desligar o servidor de armazenamento
 - Ligar o servidor de armazenamento na mesma porta e na mesma máquina
 - Consultar o servidor central pela chave "1"
- Não foram tratadas exceções relativas a conexões entre cliente-servidor ou servidor-servidor.
- O trabalho estará disponível publicamente no [Github](#) após a data final de entrega dia 31/01/2022 às 13:00.