

Escola de Artes, Ciências e Humanidades
Universidade de São Paulo

Lucas S. S. Gigante (12691898) e Pietro Zalla (12717606)

DOCUMENTAÇÃO PARA COMPILAR O PROJETO

Versão do Python usada: 3.10.8

Instalação de bibliotecas:

Para o gRPC

```
pip install matplotlib  
pip install numpy  
pip install grpcio  
pip install protobuf
```

Para o JSON-RPC

```
pip install matplotlib  
pip install numpy  
pip install jsonpickle  
pip install jsonrpcserver  
pip install jsonrpclib
```

Preparando o código para execução em uma máquina

No arquivo `server.py` do gRPC: como argumento da função `add_insecure_port`, deve-se colocar `'localhost:50051'`

No arquivo `client.py` do gRPC: como argumento da função `grpc.insecure_channel`, deve-se colocar `'localhost:50051'`

No arquivo `server.py` do JSON-RPC: como argumento da função `serve`, deve-se colocar `('localhost', 8545)`

No arquivo `client.py` do JSON-RPC: como valor da variável `server_ip`, deve-se colocar `'localhost'`

Preparando o código para execução em máquinas diferentes

Considere `{IPv4_address}` como o IP do computador que está executando o `server.py`

No arquivo `server.py` do gRPC: como argumento da função `add_insecure_port`, deve-se colocar `'0.0.0.0:50051'`

No arquivo `client.py` do gRPC: como argumento da função `grpc.insecure_channel`, deve-se colocar `{IPv4_address}:50051'`

No arquivo `server.py` do JSON-RPC: como argumento da função `serve`, deve-se colocar `("0.0.0.0", 8545)`

No arquivo `client.py` do JSON-RPC: como valor da variável `server_ip`, deve-se colocar `{IPv4_address}`

Executando o RPC

Primeiro, inicia-se o servidor (na máquina desejada) ao executar o arquivo `server.py`. Depois, é acionado o cliente com a execução do arquivo `client.py`.

No cliente, surgirá um menu onde será pedido o método a ser testado ou a execução manual de todos os testes.