

Alif Ahsanil Satria
1606882540

Server setelah modifikasi :

```
class ExampleServicer(example_pb2_grpc.ExampleServiceServicer):  
    def SayHello(self, request, context):  
        response = example_pb2.HelloResponse()  
        response.reply = "Hello " + request.greeting + "!"  
        return response  
  
    def DoAddition(self, request, context):  
        response = example_pb2.NumberResponse()  
        response.result = request.first_number + request.second_number  
        return response  
  
    def DoSubtraction(self, request, context):  
        response = example_pb2.NumberResponse()  
        response.result = request.first_number - request.second_number  
        return response  
  
    def CustomFunction(self, request, context):  
        response = example_pb2.ThreeArgumentResponse()  
        response.argresp1 = request.argreq1  
        response.argresp2 = request.argreq2  
        response.argresp3 = request.argreq3  
        return response
```

Client setelah modifikasi :

```
client.CustomFunction(  
    { argreq1: "a", argreq2: "b", argreq3: "c", argreq4: "d", argreq5: "e" },  
    (error, response) => {  
        if (!error) {  
            console.log("First response : " + response.argresp1);  
            console.log("Second response : " + response.argresp2);  
            console.log("Third response : " + response.argresp3);  
        }  
        else {  
            console.log("Error : " + error.message);  
        }  
    }  
);
```

example.proto setelah modifikasi :

```
service ExampleService {  
    rpc SayHello(HelloRequest) returns (HelloResponse);  
    rpc DoAddition(NumberRequest) returns (NumberResponse);  
    rpc DoSubtraction(NumberRequest) returns (NumberResponse);  
    rpc CustomFunction(FiveArgumentRequest) returns (ThreeArgumentResponse);  
}
```

```
message FiveArgumentRequest {  
    string argreq1 = 1;  
    string argreq2 = 2;  
    string argreq3 = 3;  
    string argreq4 = 4;  
    string argreq5 = 5;  
}  
  
message ThreeArgumentResponse {  
    string argresp1 = 1;  
    string argresp2 = 2;  
    string argresp3 = 3;  
}
```

pemanggilan server.py :

```
alif@alif:~/Documents/kuliah/law/alifahsanilsatria-labreport12/grpc-example/python$ python3 server.py  
Starting server. Listening on port 50050.  
█
```

pemanggilan client.js :

```
alif@alif:~/Documents/kuliah/law/alifahsanilsatria-labreport12/grpc-example/node$ node client.js  
First response : a  
Second response : b  
Third response : c  
Substraction: -1  
Addition: 3  
Hello Prakash!
```

A. Bagaimana Client-Server bekerja ?

client.js, sebagai client yang terintegrasi dengan gRPCstub, mengirimkan proto request ke server.py yang terintegrasi dengan gRPC server. Lalu, server.py mengirimkan proto response ke client.js untuk ditampilkan ke client hasil response nya. Pengiriman request ini sendiri diimplementasikan dengan cara memanggil fungsi yang menerima sejumlah argumen request dan memberikan sejumlah argumen response yang sudah didefinisikan pada example.proto, di-compile oleh

generate_pb.py, lalu implementasi fungsi tersebut dilakukan pada class ExampleServices di server.py

- B. Apa yang menjadi kendala dalam implementasi pada bahasa yang berbeda ?
Library yang menjadi dependensinya berbeda sehingga membutuhkan waktu dan energi ekstra untuk menginstall keduanya.
- C. Apa bedanya pemanggilan gRPC ini dengan pemanggilan web service ?
Dalam pemanggilan request, web service menggunakan HTTP API sehingga cara pengirimannya di