

Logo da Orange Talents

# Consumindo um endpoint via GRPC

## Cenário:

Imagine que um desenvolvedor(a) junior do seu time recebeu a tarefa de criar uma API REST responsável por verificar se determinado cliente possui situação regular no sistema do SERESA.

Para se comunicar com o SERASA temos que se integrar à sua API gRPC pública disponível através de um arquivo Protobuf publicado na página de desenvolvedores da empresa. Ao fazer o download do arquivo .proto temos o seguinte conteúdo:

syntax = "proto3";  
  
service SerasaGrpcService {  
 rpc verificarSituacaoDoCliente(SituacaoDoClienteRequest) returns (SituacaoDoClienteResponse) {}  
}  
  
message SituacaoDoClienteRequest {  
 string cpf = 1;  
}  
  
message SituacaoDoClienteResponse {  
 enum Situacao {  
 DESCONHECIDA = 0;  
 REGULAR = 1;  
 IRREGULAR = 2;  
 }  
  
 Situacao situacao = 1;  
}

Dando início ao projeto, o desenvolvedor(a) criou um projeto Micronaut e o configurou como aplicação gRPC fazendo todo o setup necessário (dependências no Gradle, IDE, stubs gerados etc) para consumir uma API gRPC. Em seguida implementou uma API REST com Micronaut que recebe o CPF do cliente para consultar sua situação no SERASA:

@Validated  
@Controller  
class SerasaController {  
  
 @Post("/api/serasa/clientes/verificar-situacao")  
 fun verificar(@Valid @NotBlank @CPF val cpf): HttpResponse<Any> {  
  
 val situacao = // TODO: consumir API gRPC aqui  
   
 return HttpResponse.ok(SituacaoNoSerasaResponse(cpf, situacao))  
 }  
}  
  
data class SituacaoNoSerasaResponse(  
 val cpf: String,   
 val situacao: Situacao  
)  
  
enum class Situacao {  
 SEM\_INFORMACOES,  
 REGULARIZADA,  
 NAO\_REGULARIZADA  
}

Infelizmente o desenvolvedor(a) junior não tem experiência suficiente com consumo de APIs gRPC. Seu papel é ajudá-lo(a) a terminar de implementar a API REST acima para consumir a API gRPC do SERASA retornando a situação atual do cliente no SERASA.

Como vimos, o projeto Micronaut está devidamente configurado com suas dependências, desse modo, como você implementaria o consumo dessa API gRPC?