## Teste para desenvolvedor .net

O teste abaixo visa avaliar o perfil técnico dos candidatos a desenvolvedor backend. Leia atentamente as instruções abaixo e responDa as questões da melhor forma possível.

* O teste será composto de três perguntas técnicas e uma avaliação prática.
* Responda as perguntas com suas palavras, com seu entendimento sobre o tema.
* Publique o teste prático com as respostas em um repositório no git-hub e nos envie.

## Perguntas:

## Explique com suas palavras o que é domain driven design e sua importância na estratégia de desenvolvimento de software.

## É um padrão de modelagem de software orientado a objetos. A idéia do ddd é simplesmente programar o software com foco no desenvolvimento da camada de domínio. Em ddd, o design da solução é orientado pelas regras de negócio.

## Explique com suas palavras o que é e como funciona uma arquitetura baseada em microservices. explique ganhos com este modelo e desafios em sua implementação.

## Essa arquitetura tem como objetivo principal tornar a aplicação mais flexível, escalável e com manutenção mais simples. Funciona basicamente em construir uma aplicação com um conjunto de pequenos serviços, cada um sendo desenvolvido em torno de um conjunto de regras de negócio específico, sendo inclusive implementados de forma independente.

## Posso dizer algumas grandes vantagens como: facilitar a escalabilidade do sistema, simplificar o deploy de componentes separados do sistema e possível rapidez com que a solução permite entregar software com qualidade. Já um grande desafio pode estar no acesso a informação entre os serviços.

## explique qual a diferença entre comunicação sincrona e assincrona e qual o melhor cenário para utilizar uma ou outra.

## Basicamente, comunicação síncrona requer a espera por uma resposta para continuar e/ou iniciar uma outra tarefa, já a assíncrona independe de uma resposta.

## Pensando em uma melhor experiência do usuário, a comunicação síncrona não é bem recomendada, pois por exemplo, a aplicação fica aguardando o retorno de uma requisição, enquanto isso o usuário “fica preso” na tela, sem poder avançar, então para este cenário o recomendado é uma comunicação assíncrona. Pensando na experiência do usuário, a comunicação síncrona só deve ser utilizada quando realmente não for possível utilizar a assíncrona, dependerá muito da necessidade da aplicação.

## Teste prático:

## regra de negócio:

## 

## criar um microservices que, através de um http post efetue uma operação de debito (origem) e credito (destino) nas contas correntes.

## entidades: contacorrente, lancamentos (voce pode incrementar com outras entidades se achar necessário)

## Parâmetrosde entrada:

## conta origem

## conta destino

## valor

## Parâmetros de saída:

## http status code sE OK, RETORNA STATUS CODE 200 E BOOLEAN TRUE

## 

## informações adicionais:

## o método “post” devera receber os parametros no body da requisição em formato json

## UTILIZE Domain Driven Design

## serão avaliados critérios de arquitetura como separação de responsabilidade, clean code, segurança e testes

## tecnologias que você pode utilizar .net core 2.X, c#, xunits (testes)

## no término do projeto, publique o código em um repositório no git-hub

## **Utilização recursos adicionais:**

## **autenticação por bearer token**

## inicialmente o usuário deverá se autenticar

## usuário precisará se autenticar para utilizar qualquer método da api

## **swagger** para expor os métodos da api e documentação dos retornos de cada método, desta forma facilitando a leitura dos métodos por quem irá consumí-los

## **implementação de métodos extras:**

## **método para obter saldo**

## **método para obter extrato**

## **método para inserir um lançamento na conta corrente (ex: crédito, débito)**

## **implementação de validações:**

## **validação para que não seja possível:**

## **verificar se contas correntes existem na base**

## **não aplicar transferência para mesma conta corrente**

## **não aplicar transferência de valor o (zero)**

## **verificar se conta de origem possui saldo para realizar a transferência**

## **oBS.: GRAVEI UM VÍDEO PARA DEMONSTRAR O CONSUMO DOS MÉTODOS DA APLICAÇÃO**