Teste para desenvolvedor .net

O teste abaixo visa avaliar o perfil técnico dos candidatos a desenvolvedor backend na Superdigital. 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 envie o endereço, (BRUNO.COUTO@superdigital.com.br)

Perguntas:

* Explique com suas palavras o que é domain driven design e sua importância na estratégia de desenvolvimento de software.  
    
  Domain driven design é um princípio de desenvolvimento que guia o desenvolvedor a criar suas aplicações/soluções de modo a trazerem a maior parte possível da complexidade pra o domínio da aplicação, ou seja, criar dentro dele as entidades (models), os contratos (iterfaces) contendo todos os métodos que deverão ser implementados posteriormente, etc... Isso, além de tudo, permite que você torne o projeto pouco acoplado, pois toda a complexidade e as dependências do projeto estarão vinculadas ao domínio, e não espalhados por todas as demais camadas do projeto.
* 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.  
    
  Uma arquitetura baseada em microserviços usa como base endpoints http (api’s), que não necessariamente estão localizados na mesma solution ou no mesmo projeto de desenvolvimento. deste modo, um processo pode ser dividido em partes separadas que se completam, o que deixa o processo como um todo mais leve, pois cada microsserviço fica responsável por fornecer a parte do processo que lhe cabe. Além disso, do ponto de vista de infraestrutura, os microserviços são independentes, o que torna o gerenciamento muito pouco centralizado. em outras palavras, a baixa dependência de um microsserviço para outro permite até que sejam escritos em linguagens diferentes uma das outras e as informações são transmitidas entre as camadas de apresentação através de requisições http que devolvem dados (normalmente em formato json) que podem ser também interpretados e deserializados em objetos facilmente por qualquer linguagem de programação.
* explique qual a diferença entre comunicação sincrona e assincrona e qual o melhor cenário para utilizar uma ou outra.

a comunicação síncrona acontece quando duas ou mais usuários se comunicam ao mesmo tempo, ao exemplo de uma conferência via chamada de vídeo. Nesse caso, todos os usuários podem se manifestar ao mesmo tempo. já no caso da comunicação assíncrona, acontece quando a comunicação acontece em tempos distintos, ao exemplo de um envio de e-mail. Nesse caso, os usuários envolvidos na comunicação se manifestam acerca de um assunto individualmente e independentemente uns dos outros.

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)

Parâmetros de entrada:

conta origem

conta destino

valor

Parâmetros de saída:

http status code

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