

**Netshoes**

**Teste de Arquitetura**

**OLÁ**

Caro candidato, você está participando do processo seletivo Netshoes. Abaixo você encontrará algumas perguntas relacionadas ao perfil de *Software Engineer / Architect Java / Solution Architect*.

Que Força esteja com você! e boa sorte!

**1) Arquitetura e Designer (10 pontos)**

Você, sendo um dos arquitetos da principal plataforma de e-commerce da Companhia, é chamado para uma reunião técnica para falar sobre algumas metas que as lojas precisam ter para atingir o target de 300k usuários simultâneos, considerando primeiro acesso, pesquisas, detalhamento de pedido e finalização de compra. Para tanto, o Diretor da área decide que não devemos mais evoluir a plataforma atual e pede para que o Arquiteto desenhe uma nova solução de Arquitetura contendo os seguintes requisitos:

* Seja escalável;
* Seja Segura;
* Seja Flexível para atender o negócio, principalmente ao front;
* Rode em ambiente de Cloud Computing;
* Tenha resiliência nas conectividades e integrações;
* Seja Evolutiva na parte tecnológica, e para manutenção e Testes;
* Tenha Baixa latência para a camada de Banco de Dados;

**Proposta**

Desenhe uma plano arquitetural/ Arquitetura de referencia envolvendo todas as camadas da aplicação, forma de integração entre as camadas e forma de integração para outros sistemas. Descreva ainda quais tecnologias devem ser colocadas em cada uma destas camadas e porque deve ser utilizadas cada uma delas tecnologias.

**Entrega**

Os desenhos da solução, podem ser entregues via Google Docs, Office 365, Git, ArchimateTool, Gliffy etc. So não esqueçam de autorizar o acesso.

**2) Limitação Backoffice (10 pontos)**

Mais uma vez, você é chamado para uma reunião de tecnologia para discutir uma grande problema que está impactando a finalização de vários pedidos. Este problema está relacionado ao Back office onde é feito todo o processamento do pedido. Atualmente o pedido é iniciado na loja e finalizado no back office, mas, a versão atual do back office é um produto de empresa XPTO onde existe um limite de processamento de 5000 pedidos hora. Porém a meta de venda da Loja é de 40k e esta integração está aferindo diretamente esta meta pois o back office não processo em tempo hábil a cada integração. Você, como arquiteto deve resolver esse problema.

**Proposta**

Elabore um arquitetura de integração que resolva este problema. Nesta arquitetura, descreva as tecnologias a serem utilizadas e porque esta mudança resolve os problemas. Por fim, é necessário que seja utilizado um designer Pattern de Integração conhecido. Não esqueça de descreve-lo.

**Entrega**

Os desenhos da solução, podem ser entregues via Google Docs, Office 365, Git, ArchimateTool, Gliffy etc. So não esqueçam de autorizar o acesso.

**3) Um pouco além do MicroServiço (5 pontos)**

Como a ascensão de modularização e segregação de aplicações por contexto, explique com suas palavras:

* que é Micro Serviço?
* Porque devemos utilizar?
* Quais São os Prós e Contras.

Por fim, imagine que você, como arquiteto direcionou o time a construção de um micro serviço de CEP, por exemplo, e o mesmo é altamente acessado como, você como arquiteto, escalaria este micro service? Como resolveria a questão do "S.E.P" (Single entry point) ?

**O que espera-se como resposta - Dicas e direcionamentos:**

* Descreva detalhadamente todas as questões se forma organizada

**4) Latência de Rede (10 pontos)**

Você, como um arquiteto de Soluções é chamado pelo time de Devops/ Infraestrutura para desenhar uma solução que resolva o problema de latência de rede relacionada a comunicação entre a aplicação e o Banco de dados, pois a aplicação é fortemente dependente de um banco de dados, principalmente nas pesquisas e geração de relatórios.

**Para tanto, algumas perguntas foram feitas na reunião e que você precisa responder:**

* que é Latência de REDE?
* Qual estratégia deve ser implementada para diminuir essa latência?

**5) Concorrência (10 pontos)**

Durante uma reunião de negócio, um dos diretores de Back office levanta um problema sério de sincronismo de estoque, onde, a loja está vendendo produtos com estoque zerado. De forma complementar, ele explica que, existe 3 canais de vendas de produtos, a loja principal, o call center e as lojas parceiras. Com isto, temos 3 canais que manipulam estoque, porém o back office é responsável por avisa-los da alteração do estoque. Com isto, como cada canal cuida do seu estoque, a probabilidade de vender produtos sem estoque é alta.

Sendo assim, a equipe de arquitetura de solução foi acionada e, você, como arquiteto responsável por esta frente, precisa:

* Desenhar uma solução que resolva o problema de venda de estoque zerado;
* A solução deve se atentar a situações de estorno/devolução de compra;
* Deve ser performática e escalável, pensando em situações de sazonalidade;

**Entrega**

Os desenhos da solução, podem ser entregues via Google Docs, Office 365, Git, ArchimateTool, Gliffy etc. So não esqueçam de autorizar o acesso.

**6) Dada uma stream, encontre o primeiro caracter Vogal, após uma consoante, onde a mesma é antecessora a uma vogal e que não se repita no resto da stream. O termino da leitura da stream deve ser garantido através do método hasNext(), ou seja, retorna falso para o termino da leitura da stream. Voce tera acesso a leitura da stream através dos métodos de interface fornecidos ao termino do enunciado. (10 pontos)**

***\*Leia todo o enunciado***

**Premissas:**

Uma chamada para hasNext() ir retornar se a stream ainda contem caracteres para processar.

Uma chamada para getNext() ir retornar o proximo caracter a ser processado na stream.

Não será possível reiniciar o fluxo da leitura da stream.

Não poderá ser utilizado nenhum framework Java, apenas código nativo.

Exemplo:

Input:  aAbBABac**afe**

Output: e

No exemplo, ‘e’ é o primeiro caracter Vogal da stream que não se repete após a primeira Consoante ‘f’o qual tem uma vogal ‘a’ como antecessora.

*Segue o exemplo da interface em Java:*

*public interface Stream{*

*public char getNext();*

*public boolean hasNext();*

*}*

*public static char firstChar(Stream input) {*

*}*

**O que espera-se como resultado - Dicas e direcionamentos:**

* Tente criar sua implementação pensando em performance;
* Efetuar casos de teste para diversos cenários e uma boa pratica;
* Documentar o código seguindo as boas práticas de mercado;
* Os códigos mais simples tendem a serem mais elegantes;
* Em caso de não localização do caracter, o sistema deve informar uma mensagem amigável;
* Ao finalizar o desenvolvimento você pode compartilhar o código pelo Github ou de outra maneira que preferir (como arquivo compactado). Se possível, em caso de arquivo compacto, envie o mesmo para um virtual drive e compartilha o link na prova;
* Caso utilize o Git/Bitbucket, não esqueça de criar o .gitignore
* Não copie os códigos gerados na ferramenta.
* Não utilizar o nome da **Netshoes** nos projetos ou packages da prova.

**7)** **Quando você digita a URL de um site (http://www.netshoes.com.br) no browser e pressiona enter, explique da forma que preferir, o que ocorre nesse processo do protocolo HTTP entre o Client e o Server. (5 pontos)**

**O que espera-se como resposta - Dicas e direcionamentos:**

* Detalhe sua linha de raciocínio;
* Elabore um plano de entendimento, por exemplo, lista, de forma a elencar os passos;
* Não copie conteúdo da internet, responda com suas palavras.