**Daniel Marmitt**

**MBA em arquitetura de software**

**DBA Official SQL Server**

**Graduado em Análise e desenvolvimento de sistemas**

**Exercícios**

22/07/2017

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

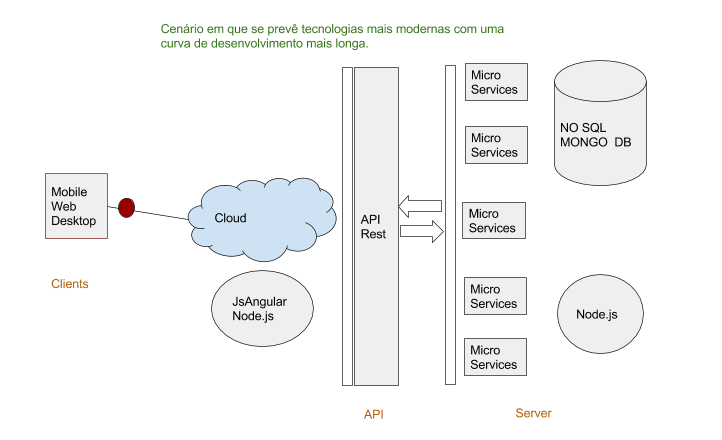
Você, sendo um dos arquitetos da principal plataforma de e-commerce da Companhia, é chamado para uma reunião técnica para falar sobre algumas metas que as lojas precisam ter para atingir o target de 300k usuários simultâneos, considerando primeiro acesso, pesquisas, detalhamento de pedido e finalização 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;

**R.: Proposta**

A dois cenários que precisam ser considerados, já que, software é sinônimo de evolução gradativa, então cada cenário influi em prazos, então tenho a sensibilidade de medir qual a melhor solução pertinente para curto, médio ou longo prazo.

**1º Cenário**



As API’s proporcionam a integração com outros sistemas, as tecnologias modernas como Node.js e Mongo DB prometem suportar muitas requisições sem a necessidade de investimento de hardwares para realizar escalabilidade.

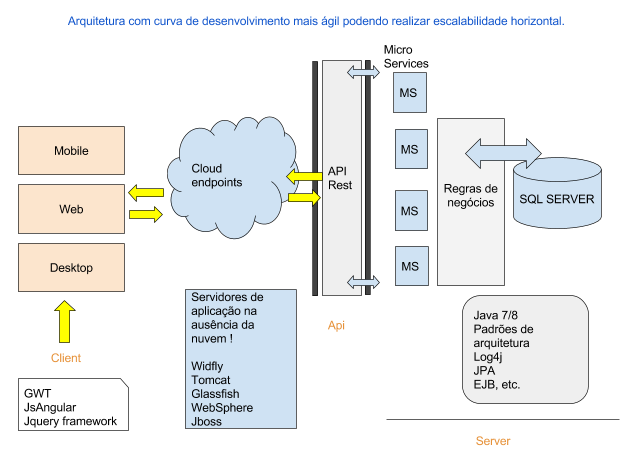
**2º Cenário**

A integração com outros sistemas acontece pela API, o ambiente de desenvolvimento é Java, na camada de Server é possível desenvolver um projeto bem estruturado com padrões de arquitetura, ou ainda, aplicar os novos conceitos de micro serviços que foram ilustrados na imagem.

Toda empresa possui tecnologias classificadas como padrões, neste documento eu determino para o front-end uso de GWT, JsAngular, Jquery Framework, todas tecnologias já carimbadas para desenvolvimento em cloud.

No Server, as Api’s em base rest podem rodar sobre servidores de aplicação como Widfly, Glassfish, WebSphere , por fim, o espaço cloud sugerido seria a tecnologia da Google.

Por ser um banco de dados robusto a qual tenho vivência e certificação oficial, o ilustro como escolha pessoal desconhecendo se algum padrão dentro da corporação, de qualquer modo, trabalhando com a especificação JPA/Hibernate a tecnologia de banco se torna abstrata.

****

**2) LimitaçãoBackoffice (10 pontos)**

Mais uma vez, você é chamado para uma reunião de tecnologia para discutir um 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.

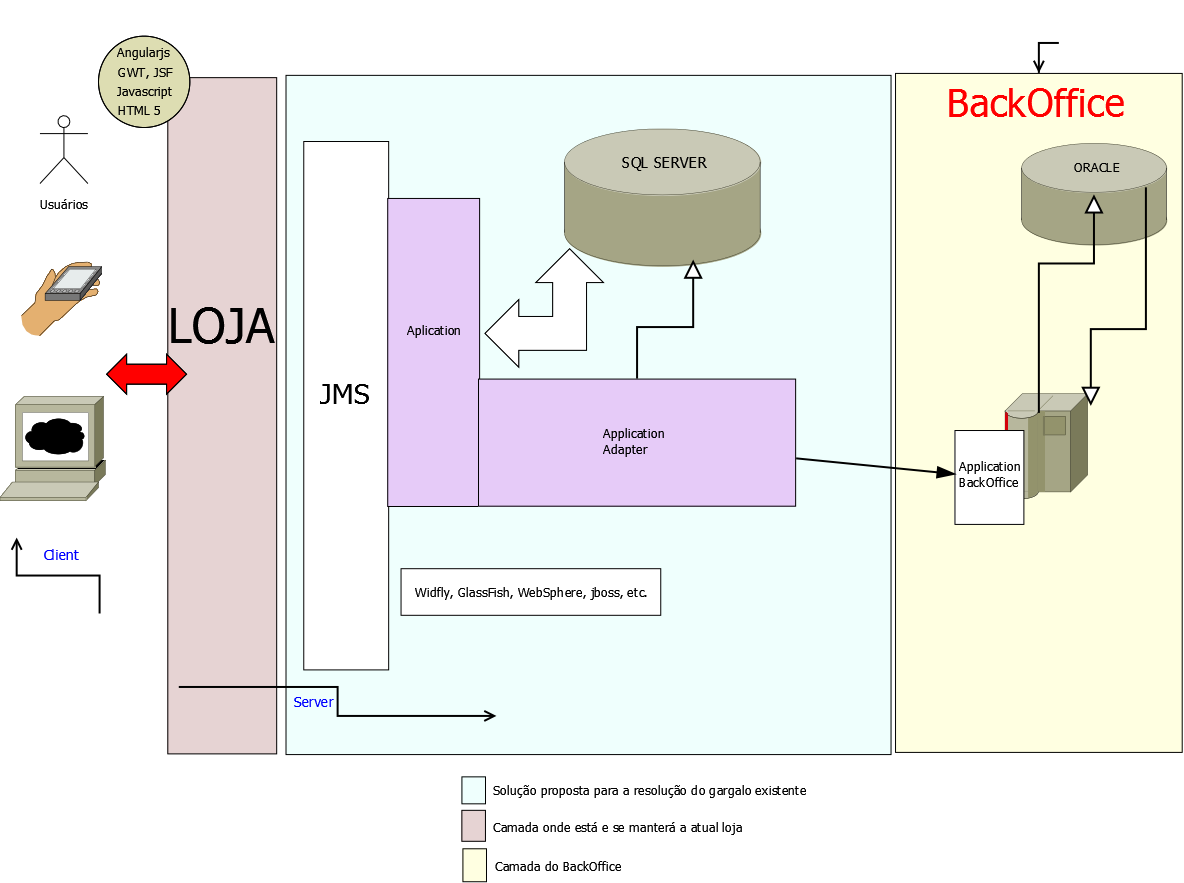
**R.:Proposta**

A criação de uma nova aplicação na camada de Server é uma solução, já que a nova aplicação utilizará JMS (Filas, ambiente Java), terá autonomia de armazenar e controlar a lista de requisições ao BackOffice, não perdendo vendas submetidas pela loja, desta forma a aplicação que pode vir a utilizar tantos Pattern, quanto necessários, vai implementar o Pattern Adapter”.

Na nova aplicação também é concebido um banco de dados visando gravar os pedidos que estiverem na fila e aguardam o backoffice se recuperar dos limites que lhe são impostos.

*Tecnologias a serem utilizadas*

Java7/8(EJB, JPA, JMS, Log4j, NHibernate, etc.), SQL Server(Ou outro banco padrão)



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

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

* **O que é Micro Serviço?**

Pode se considerar que micro serviços são pedaços, fragmentos de programas autônomos que formam um grande programa, a idéia é que ele seja coeso, de baixo acoplamento inclusive facilitando a manutenção e rotinas de testes do departamento de QA.

* **Porque devemos utilizar?**

Em grandes projetos quando houver uma parada crítica ou um possível problema mais sério em um determinado micro serviço, apenas parte do programa deixa de funcionar, e não todo um programa fazendo uso desta arquitetura.

É importante mensurar também que a estrutura de hardwares e softwares pode proporcionar cenários que o conceito não é 100% aplicável.

É importante frisar, microserviço é uma tendência e tem provado ser eficaz na prática em projetos que tem entre as características, ser muito acessado e de tamanho significativo.

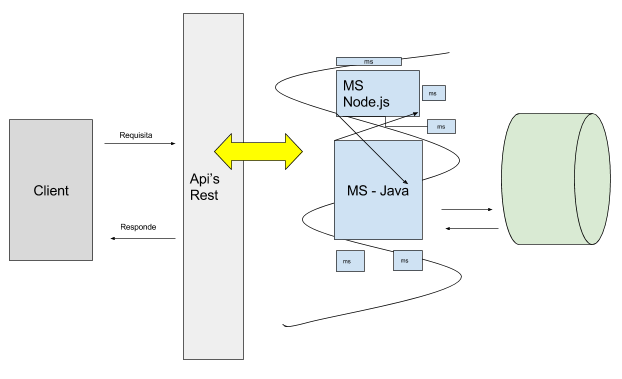
* **Quais São os:**

**Prós;**

- Coeso, multi-tecnologias (Java, Node.js, etc), facilitador na manutenção na grande maioria dos cenários.

**Contras;**

- Versionamento, documentação enorme, demanda maior tempo no desenvolvimento dependendo do cenário do projeto.



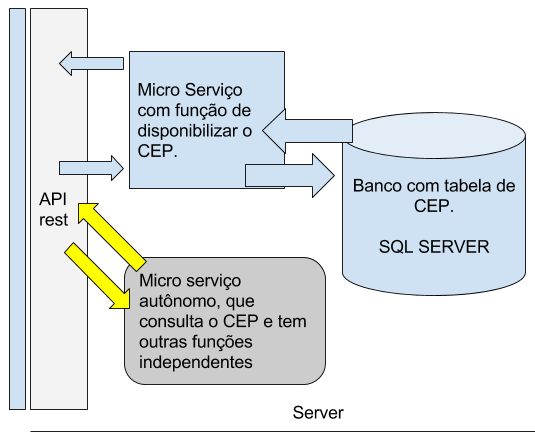
Por fim, imagine que você, como arquiteto direcionou o time a construção de um micro serviço de CEP, por exemplo, e o mesmo é altamente acessado como, você como arquiteto, escalaria este micro service? Como resolveria a questão do "S.E.P" (Single entry point) ?

**R.:Proposta**  
Inicialmente até imaginei que os correios disponibilizassem um webservices para consulta de CEPS, no entanto pude perceber que não a este serviço e sim ocorre a disponibilização para venda do banco de dados.

Com este material em mãos o procedimento natural é importar para uma tabela em banco de dados todos os ceps e seus respectivos dados.

Construindo o micro serviço e o expondo na API padrão para consultas, tomo como base a utilização de rest.

Segue o **desenho da arquitetura**.



**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:

**- O que é Latência de REDE?**  
É o tempo de envio e retorno de um pacote, um exemplo simples seria o ato do Browser navegar até um determinado host e retornar.

**- Qual estratégia deve ser implementada para diminuir essa latência?**

No primeiro momento é fazer com que as aplicações que estejam no Server armazenem em memória valores globais, que não se tenha o custo de um novo processamento.

Em um segundo momento que a estrutura de hardware suporte realmente a quantidade de requisições que chegam, levando em consideração inclusive escalabilidade horizontal.

**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;

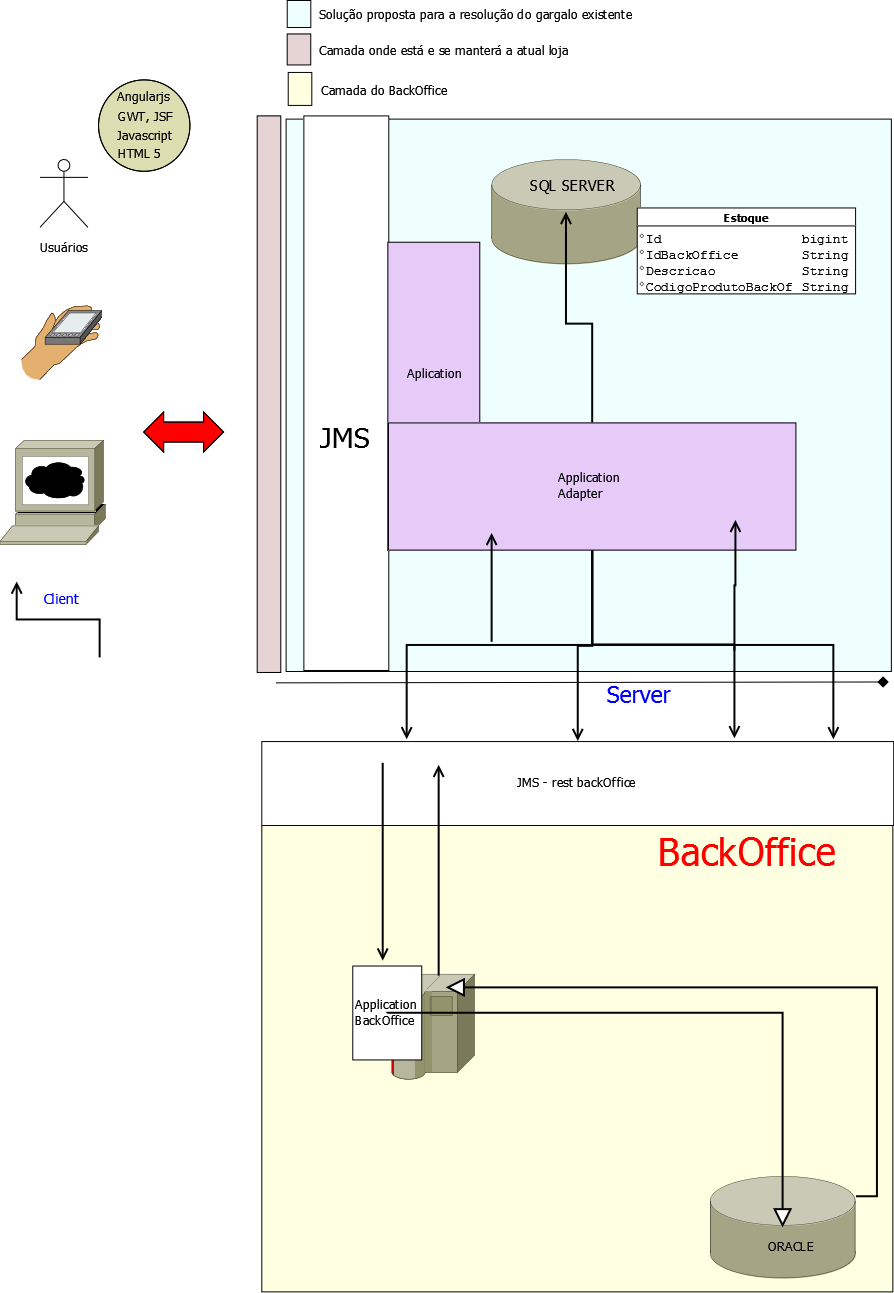
**R.:Proposta**

A solução que resolve de uma vez por todas é o desenvolvimento de uma classe especialista, estou partindo do princípio que tem um banco de dados relacional na solução de BackOffice, onde o campo quantidade em estoque já é existente.

Estaria errado atualizar o banco do BackOffice ignorando as regras de negócios e processos para a conclusão de uma transação, assim, no nível da aplicação adapter proposta no exercício 2 se faz a inclusão de uma tabela com 2 ou 3 dos principais campos referenciados vindos do BackOffice e diariamente atualizando suas quantidades atuais.

A cada venda nos 3 canais a aplicação adapter, decrementa as quantidades na tabela em tempo real mesmo que um determinado pedido seja encaminhado para uma fila (JMS).

Disponível como um serviço na API, a classe proposta tem as funções de dar baixa no estoque quando uma venda acontecer, como também consultar a quantidade de estoque atual, assim uma regra de negócio poderia ser implementada nos 3 canais obtendo a quantidade real de produtos em estoque evitando inclusive a necessidade de futuros estornos.



**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)**

Vamos começar identificando quem são os atores, o client é o navegador ou o dispositivo que faz a requisição, já o Server é quem recebe e retorna uma resposta.

O protocolo Http é baseado em requisições e respostas, o client e o Server trocam cabeçalhos (headers) entre si através de URI, nestes cabeçalhos informações importantes são transportadas, como por exemplo, o status de êxito, 200 ou de referência não encontrada 400.

O protocolo HTTP é stateless, ou seja, não tem a capacidade de armazenar informações entre requisições diferentes, para isso cookies, sessões, formulários ou variáveis na própria URI são utilizados.

Os 5 métodos mais utilizados são, GET, POST, PUT, DELETE, HEAD

