



UNIVERSIDADE FEDERAL DO CEARÁ – CAMPUS SOBRAL

QUALIDADE DE SOFTWARE

Aluno: Gabriel Vasconcelos Santos

Matrícula: 497688

Sobral - CE

2022.02

Parte 1 - Exercícios de Fixação

1. Dada a sua complexidade, sistemas de bancos de dados são componentes relevantes na arquitetura de qualquer tipo de sistema. Verdadeiro ou falso?

Justifique a sua resposta.

R: Verdadeiro, pois um SBD (Sistema de banco de dados) responsável pelo armazenamento de informações importantes em um projeto.

2. Descreva três vantagens de arquiteturas MVC.

R: Permite testabilidade, permite que as classes do modelo sejam usadas por visões diferentes, favorece a especialização do trabalho de desenvolvimento.

3. Qual a diferença entre classes Controladoras em uma Arquitetura MVC tradicional e classes Controladoras de um sistema Web implementado usando um framework MVC como Ruby on Rails?

R:

MVC: As classes controladoras tratam e interpretam eventos gerados pelos dispositivos de entradas, e em resposta podem solicitar alterações nos estados de divisão e modelo.

MVC em Ruby on Rails: Além de ser responsável por tratar solicitações externas, lida com URLs, gerência cache e módulos auxiliares do sistema.

4. Descreva resumidamente quatro vantagens de microsserviços.

R:

1) Quando se usa um monolito, as falhas são totais. Se o banco de dados cair, todos os serviços ficam fora do ar. Por outro lado, em arquiteturas baseadas em microsserviços podemos ter falhas parciais.

2) Como os microsserviços são autônomos e independentes eles podem ser implementados em tecnologias diferentes, incluindo linguagens de programação, frameworks e bancos de dados.

3) Permitem escalar um sistema em um nível de granularidade mais fino do que é possível com monólitos.

4) Permitem a evolução mais rápida e independente de um sistema, permitindo que cada time tenha seu próprio regime de liberação de novas releases.

5. Por que microsserviços não são uma bala de prata? Isto é, descreva pelo menos três desvantagens do uso de microsserviços.

R:

1) Complexibilidade: Um sistema formado por partes autônomas e especializadas forma um todo bem complexo, distribuído.

2) Governança: Com vários componentes completamente distintos trabalhando juntos em um único aplicativo, a governança pode deixar a desejar.

3) Segurança: Algumas vulnerabilidades dos microsserviços são resultados diretos de sua natureza aberta, com uma ampla superfície sujeita a ataques.

6. Explique a relação entre a Lei de Conway e os microsserviços.

R:

Empresas tendem a adotar arquiteturas de software que são cópias de suas estruturas organizacionais. Em outras palavras, a arquitetura dos sistemas de uma empresa tende a espelhar seu organograma. Por isso, não é coincidência que microsserviços sejam usados, principalmente, por grandes empresas de Internet que possuem centenas de times de desenvolvimento distribuídos em diversos países. Além de descentralizados, esses times são autônomos e sempre incentivados a produzir inovações.

7. Explique o que significa desacoplamento no espaço e desacoplamento no tempo. Por que arquiteturas baseadas em filas de mensagens e arquiteturas Publish/Subscribe oferecem essas formas de desacoplamento?

R:

A comunicação entre clientes e servidores é assíncrona e cria um acoplamento fraco entre clientes e servidores.

Desacoplamento no espaço: Clientes não precisam conhecer os servidores e vice-versa. Em outras palavras, o cliente é exclusivamente um produtor de informações. Mas ele não precisa saber quem vai consumir essa informação. O raciocínio inverso vale para os servidores.

Desacoplamento no tempo: Clientes e servidores não precisam estar simultaneamente disponíveis para se comunicarem. Se o servidor estiver fora do ar, os clientes podem continuar produzindo mensagens e colocá-las na fila. Quando o servidor voltar a funcionar, ele irá processar essas mensagens.

8. Quando uma empresa deve considerar o uso de uma arquitetura baseada em filas de mensagens ou uma arquitetura publish/subscribe?

R:

Fila de mensagens: Quando um aplicativo tiver apenas alguns consumidores que precisam de informações consideravelmente diferentes do aplicativo de produção ou quando exigir interação quase em tempo real com os consumidores.

Publish/Subscribe: Quando um aplicativo precisar transmitir informações para um número considerável de consumidores ou quando um aplicativo precisar se comunicar com um ou mais aplicativos ou serviços desenvolvidos de forma independente, que podem usar plataformas, linguagens de programação e protocolos de comunicação diferentes.

9. Explique o objetivo do conceito de tópicos em uma arquitetura publish/subscribe.

R:

Em alguns sistemas publish/subscribe, eventos são organizados em tópicos, que funcionam como categorias de eventos. Quando um publicador produz um evento, ele deve informar seu tópico. Assim, clientes não precisam assinar todos os eventos que ocorrem no sistema, mas apenas eventos de um determinado tópico.

10. (POSCOMP, 2019, adaptado) Marque V ou F.

(F) O padrão MVC é uma adaptação do padrão arquitetural Camadas. A Camada

Visão lida com a apresentação e a manipulação da interface, a Camada Modelo organiza os objetos específicos da aplicação, e a Camada Controle posiciona-se entre estas duas com as regras do negócio.

(V) O padrão Broker é voltado a problemas de ambientes distribuídos. Sugere uma arquitetura na qual um componente (broker) estabelece uma mediação que permite um desacoplamento entre clientes e servidores.

(V) Mesmo que um dado padrão arquitetural ofereça uma solução para o problema sendo resolvido, nem sempre ele é adequado. Fatores como contexto e o sistema de forças que afeta a solução fazem também parte do processo de avaliação e da escolha de padrões adequados.

Parte 2 - Atividades Práticas

1) Exemplo (didático) de um sistema MVC Web:

<https://replit.com/@engsoftmoderna/ExemploArquiteturaMVC>

Navegue no link ``Show files ``para visualizar a fonte do projeto.

Indique quais arquivos devem ser alterados para que o sistema de exemplo possibilite a consulta pelo “isbn text” do livro.

R: ControladorPesquisaLivros.java e ServicoPesquisaLivro.java