

# Desafio do Módulo 3

**Entrega** 15 abr em 21:00

**Pontos** 40

**Perguntas** 14

**Disponível** até 15 abr em 21:00

**Limite de tempo** Nenhum

## Instruções

O Desafio do Módulo 3 está disponível!

### 1. Instruções para realizar o desafio

Consulte a data de entrega no teste e em seu calendário.

Reserve um tempo para realizar a atividade, leia as orientações e enunciados com atenção. Em caso de dúvidas utilize o "Fórum de dúvidas do Desafio do Módulo 3".

Para iniciá-lo clique em "Fazer teste". Você tem somente **uma** tentativa e não há limite de tempo definido para realizá-lo. Caso precise interromper a atividade, apenas deixe a página e, ao retornar, clique em "Retomar teste".

Clique em "Enviar teste" **somente** quando você concluí-lo. Antes de enviar confira todas as questões.

Caso o teste seja iniciado e não enviado até o final do prazo de entrega, a plataforma enviará a tentativa não finalizada automaticamente, independente do progresso no teste. Fique atento ao seu teste e ao prazo final, pois novas tentativas só serão concedidas em casos de questões médicas.

O gabarito será disponibilizado partir de sexta-feira, **16/04/2021**, às 23h59.

Bons estudos!

### 2. O arquivo abaixo contém o enunciado do desafio

[Enunciado do Desafio - Módulo 3 - Bootcamp Arquiteto\(a\) de Software.pdf](#)

## Histórico de tentativas

	Tentativa	Tempo	Pontuação
MAIS RECENTE	<a href="#">Tentativa 1</a>	17 minutos	40 de 40

⚠ As respostas corretas estarão disponíveis em 16 abr em 23:59.

Pontuação deste teste: **40** de 40

Enviado 15 abr em 17:19

Esta tentativa levou 17 minutos.

## Pergunta 1

2,86 / 2,86 pts

Qual tipo de arquitetura é a mais indicada para o sistema da Silicon Sandwiches?

☐ Micros serviços.

☐ Distribuído.

☐ Misto.

☒ Monolítico.

## Pergunta 2

2,86 / 2,86 pts

A respeito de como os dados do sistema da Silicon Sandwiches serão armazenados, tomando como premissa a adoção da arquitetura Microkernel, podemos afirmar, EXCETO:

☐ É possível utilizar um banco de dados centralizado para armazenar informações que são comuns do núcleo central da aplicação.

☐ É importante que cada parte customizada do sistema esteja desacoplada uma da outra, ou seja, que cada uma mantenha seus próprios dados.

☐ Devido as diferentes customizações, podemos ter diferentes banco de dados para cada uma delas.



Não é recomendado o uso de banco de dados relacional, já que esse tipo de tecnologia não pode ser utilizado em sistemas que exigem muita personalização.

### Pergunta 3

2,86 / 2,86 pts

Qual estilo de comunicação seria o mais adequado para a loja virtual da Silicon Sandwiches?



RPC.



Assíncrona.



Síncrona.



Rest.

### Pergunta 4

2,86 / 2,86 pts

Sobre os possíveis estilos de arquiteturas que poderiam ser candidatos para a loja da Silicon Sandwiches, podemos afirmar, EXCETO:



Uma solução monolítica com uma interface web simples e um banco de dados centralizado seria uma alternativa, especialmente por causa do custo.



A arquitetura microkernel não poderia ser uma opção, já que não é indicada para sistemas web.



Microserviços não é uma solução viável por causa de custos e de sua complexidade.



Uma opção seria algum tipo de estilo arquitetural que pudesse ser ao mesmo tempo particionado do ponto de vista técnico ou por domínio.

### Pergunta 5

2,86 / 2,86 pts

Sobre a arquitetura de microserviços, foram feitas as seguintes afirmações:

- a. Microserviços permitem que cada time de desenvolvimento tenha seu próprio fluxo de liberação (release).
- b. Microserviços devem compartilhar o mesmo banco de dados, por questões de segurança.
- c. Dentre outras vantagens, microserviços oferecem independência de tecnologia, toleram falhas parciais e oferecem uma menor latência na comunicação.
- d. Microserviços devem ser fortemente coesos e fracamente acoplados.

Quantas/ quais dessas afirmações são VERDADEIRAS?



Todas são verdade, exceto o item (b).



Todas elas.



Nenhuma delas.



Duas delas.

### Pergunta 6

2,86 / 2,86 pts

O product owner do sistema da Silicon Sandwiches se disse preocupado em como lidar com as diferentes regras de negócio, tendo em vista que poderia haver regras diferentes para cada franquia da empresa. Sobre esse tema, é CORRETO afirmar.

☐

Todas as regras devem estar divididas entre cada componente da aplicação, mesmo que isso cause algum tipo de duplicidade, seria mais seguro.

☐

Trata-se de um requisito do negócio que não é capaz de ser atendido por qualquer arquitetura.

☒

As regras específicas poderiam ficar nos módulos que possuem personalizações distintas, sendo que as regras gerais ficariam centralizadas no núcleo da aplicação.

☐

As regras deveriam ser centralizadas e as personalizações tratadas como exceções.

## Pergunta 7

2,86 / 2,86 pts

Uma preocupação que surgiu durante a decisão do melhor estilo para a Silicon Sandwiches é como o sistema poderia se comportar com o aumento do número de usuários. Sobre esse tema, escolha a alternativa mais adequada considerando todos os requisitos do sistema:

☐

O sistema deveria utilizar o estilo microsserviços de modo a provisionar os recursos necessários para cada módulo conforme o necessário.

☐

Dada essa preocupação o ideal é utilizar uma arquitetura distribuída.



Esse tipo de problema deveria ser tratado pela equipe de DevOps que ficaria responsável por definir a infra que melhor atenda as necessidades do negócio, tendo em vista que isso não é um tema para a equipe de arquitetura.



Um dos desafios para o sistema é como lidar com um grande número de usuários simultâneos sem grave degradação do desempenho. O problema poderia ser tratado com um maior provisionamento de máquinas, contudo, dado que o custo é um requisito do projeto, tal abordagem deve ser analisada com base em seus trade-offs.

### Pergunta 8

2,86 / 2,86 pts

Quais das alternativas a seguir NÃO representa uma maneira de determinar se você tem o nível certo de granularidade em um microserviço?



Transação.



Propósito.



Custo.



Coreografia.

### Pergunta 9

2,86 / 2,86 pts

A respeito do isolamento das camadas em uma arquitetura de camadas, é CORRETO afirmar:



Por facilitar a inclusão de novas camadas, devemos sempre optar por camadas abertas.



Devemos sempre adotar uma camada fechada, independente do cenário.



Uma camada fechada significa que uma solicitação não pode saltar nenhuma camada. Na camada aberta, uma solicitação pode ser atendida sem o uso dessa camada.



O conceito de camadas de isolamento que exige que as camadas sejam abertas.

### Pergunta 10

2,86 / 2,86 pts

A arquitetura em camadas é uma opção para os seguintes cenários:

- i. Aplicações pequenas e simples ou websites.
- ii. Aplicações de linha de comando
- iii. Para situações com orçamento muito apertado
- iv. Projetos com restrições de tempo

Sobre esses cenários, é possível afirmar que a quantidade CORRETA de opções é a seguinte:

☒ 3.☐ 4.☐ 1.☐ 2.

### Pergunta 11

2,86 / 2,86 pts

A respeito da arquitetura pipeline, é possível afirmar, EXCETO:

- ☐ Cada duto (pipe) é tipicamente unidirecional.
- ☐ Os dutos (pipes) formam o canal de comunicação entre os filtros.
- ☒ Os filtros são autocontidos, independentes uns dos outros e não devem armazenar o estado do item processado.
- ☐ Os filtros deveriam realizar apenas uma tarefa, de modo que tarefas mais complexas deveriam ser organizadas como uma sequência de filtros.

### Pergunta 12

2,86 / 2,86 pts

São exemplos de aplicação que utilizam a estrutura de microkernel, EXCETO:

- ☐ Eclipse IDE.
- ☐ Editor VScode.
- ☒ Kernel do Linux.
- ☐ Navegador Firefox.

### Pergunta 13

2,86 / 2,86 pts



Um plug-in pode ser adicionado ao sistema core em tempo de compilação ou em tempo de execução. Sobre essa característica da arquitetura microkernel, podemos afirmar, EXCETO:



Os plug-ins alterados em tempo de execução necessitam de um novo “deploy” da aplicação.



Caso reiniciar a aplicação seja um problema, devemos utilizar plug-ins que permitam ser trocados em tempo de execução.



Os plug-ins adicionados em tempo de compilação são mais simples.



São exemplos de framework que ajudam no gerenciamento de plug-ins em tempo de execução: OSGi e Prim.

### Pergunta 14

2,82 / 2,82 pts

A respeito da arquitetura orientada a eventos, podemos afirmar, EXCETO:



A arquitetura orientada a eventos é tecnicamente dividida.



A topologia mediator foi proposta para tratar alguns problemas relacionados com a topologia broker.



Na topologia broker há uma entidade central responsável por mediar os eventos.



A arquitetura orientada a eventos possui duas topologias primárias: mediator e broker.

Pontuação do teste: **40** de 40