



Desafio Delphi

TOTVS Hospitalidade

CONFIDENCIAL





Questão 1

O que você entende por arquitetura MVC?

Model: Regras de negócio / dados;

View: Interface com usuário (envia eventos e recebe respostas da controller);

Controller: Camadas intermediária de controle.

Questão 2

Escolha um padrão de projeto e escreva seu conceito, objetivo e um cenário de uso.

Padrão: MVC;

Conceito:

- Arquitetura em três camadas - Model, View, Controller;
- Divisão das responsabilidades
- Organização e manutenção otimizados;

Cenário:

- Cadastro de Hóspedes onde o endereço precisa ser validado através do CEP, sendo a camada Model a mais importante, pois pode ser reutilizada em outras interfaces compartilhando tecnologias, como Angular no Front End, por exemplo.

Questão 3

Temos um novo projeto na empresa, onde precisaremos disponibilizar ao usuário do nosso sistema acessar informações de endereço de um determinado CEP.

Para isso, podemos contar com a api de viacep (<https://viacep.com.br/>) que nos retorna essa informação.

Porém, não podemos contar que o viacep esteja online em 100% do tempo e pra não correremos o risco de darmos um erro ao nosso cliente, podemos contar também com a apicep (<https://apicep.com/api-de-consulta/>) para o caso do viacep estar fora do ar.

Para não darmos o azar de pegarmos os 2 serviços fora do ar, contaremos com a awesomeapi (<https://docs.awesomeapi.com.br/api-cep>) para diminuirmos ainda mais o risco do nosso cliente receber uma resposta de erro.

Você consegue resolver esse problema pra gente?

O desenvolvimento é livre, porém devem ser utilizados recursos nativos da IDE (recursos de terceiros devem ser utilizados apenas via Boss. Não use componentes de terceiros que precise ser instalado na IDE).

A solução pode ser uma aplicação VCL, FMX ou pode ser uma api desenvolvida em Horse, Datasnap ou DelphiMVC.

Diferencias (mas não obrigatório);

- Aplicação de testes unitários será um diferencial interessante.
- Uso de padrões de projetos e sua criatividade.
- Desenvolver uma api e um aplicação para consumir essa api.
- Capricho em documentação tanto pra desenvolvedor quanto para usuário final.