

Desenvolvedor Back End Jr (Codeigniter 4)

Prova Técnica

Olá, caro(a) desenvolvedor(a), nesse teste analisaremos seu conhecimento geral e inclusive velocidade de desenvolvimento. Abaixo explicaremos tudo o que será necessário.

Instruções

O desafio consiste em implementar uma **API REST** utilizando o framework **PHP Codeigniter 4**, e um banco de dados relacional **MySQL**.

Você vai criar uma API de cadastro de pedidos de compra, com as seguintes funcionalidades (endpoints):

Desafio 1 *(Caráter eliminatório)*

- Endpoints de CRUD (Create, Read, Update e Delete) de clientes com os campos (CPF e/ou CNPJ, nome e/ou Razão social)
- Endpoints de CRUD (Create, Read, Update e Delete) de produtos
- Endpoints de CRUD (Create, Read, Update e Delete) de pedidos de compra, com status (Em Aberto, Pago ou Cancelado).

Desafio 2 *(Caráter não eliminatório)*

- Implementar paginação e filtro de dados nos endpoints de listagem dos dados. Os dados poderão ser filtrados por qualquer campo.
- Implementar validação de token JWT com data de expiração. Este token pode ser gerado através do link <https://jwt.io/>. O token deve ser enviado para API através do header **Authentication**

A API deverá receber os dados no formato JSON seguindo o padrão abaixo:

```
{
  "parametros": {
    // dados utilizados para o CRUD
  }
}
```

O retorno da API deverá ser no formato JSON seguindo o padrão abaixo:

```
{
  "cabecalho": {
    "status": 200, // Protocolos HTTP (200, 404, 403, 401 e
etc...)
    "mensagem": "Dados retornados com sucesso" // mensagem de
retorno para informar o usuário que acessou a api
  },
  "retorno": { // todos os dados consultados e/ou cadastrados devem
ser retornados neste campo
  }
}
```

OBS:

- A modelagem de dados, bem como os campos nela contidos ficará a seu critério. Faça da forma que julgar melhor.
- A modelagem do banco de dados deverá ser desenvolvida de forma que possamos ver os relacionamentos entre as tabelas.
- O banco de dados pode ser criado utilizando Migrations do framework Codeigniter.
- Fique à vontade para fazer a implementação das validações necessárias na camada que julgar melhor.

Devem ser utilizadas as seguintes tecnologias:

1. Framework Codeigniter 4(PHP)
2. Banco de dados relacional (MySQL)

É importante que todo o código seja 100% seu.

Prazo de Entrega: **7 dias**.

Entrega:

- O desafio 2 é de caráter **não** eliminatório, caso não consiga fazê-lo, tudo bem!
- Ao finalizar o teste, os seguintes itens devem ser enviados por e-mail ou disponibilizados em um repositório público no GITHUB:
 - Código fonte da API e do banco de dados do teste realizado em formato **zip** ou **rar**
 - Documentação explicando o processo de instalação e teste

Demonstre seu conhecimento! Caso opte pelo uso do GITHUB, capriche no README!

Uma boa prova para você! Esperamos que você se divirta e aprenda com ela! Bom trabalho!