



Teste Técnico – Desenvolvedor Backend

Olá,

Obrigado por se candidatar à posição de Desenvolvedor(a) Backend em nossa empresa! Este teste técnico foi projetado para avaliar suas habilidades e conhecimentos em desenvolvimento backend.

Objetivo

O objetivo deste teste é avaliar sua capacidade de escrever código limpo, escalável e bem estruturado. Você será avaliado tanto pela qualidade do código quanto pela solução final apresentada.

Instruções

Este teste consiste em várias tarefas que abrangem diferentes áreas do desenvolvimento backend. Leia cada tarefa cuidadosamente e forneça uma solução completa para cada uma delas. Utilize Node.js, TypeScript e NestJS para o desenvolvimento do projeto. Você pode usar qualquer ferramenta ou IDE de sua preferência. Se tiver alguma dúvida sobre uma tarefa, não hesite em nos contatar.

Tempo Estimado

Esperamos que você termine o teste em aproximadamente 3 dias. No entanto, não se preocupe muito com o tempo. É mais importante fornecer soluções de alta qualidade do que terminar rapidamente.

Endpoints Requeridos

Usuários

- Cadastro de Usuário**
 - Método:** POST
 - Rota:** /user/signup
 - Descrição:** Cria um novo usuário.
 - Requisitos:** Implementar a lógica para criar um novo usuário
- Login de Usuário**
 - Método:** POST

- **Rota:** /user/login
- **Descrição:** Realiza o login do usuário e retorna um token de autenticação.
- **Requisitos:** Implementar a lógica para autenticação de usuário e geração de um token de autenticação. O tipo de autenticação fica ao seu critério.

Tutoriais

1. Listar Tutoriais

- **Método:** GET
- **Rota:** /tutorial
- **Descrição:** Retorna uma lista de tutoriais.
- **Requisitos:**
 - Deve ser implementado filtros:
 - Por Título: Permita que os usuários filtrem tutoriais por título.
 - Por Data: Permita filtrar tutoriais por data de criação ou atualização.
 - O título não poderá se repetir.
 - A lista precisa ser paginada
 - Uso de cache

2. Adicionar Novo Tutorial

- **Método:** POST
- **Rota:** /tutorial
- **Descrição:** Adiciona um novo tutorial.
- **Requisitos:** Implementar a lógica para criação de um novo tutorial. Campos:

3. Editar Tutorial

- **Método:** PATCH
- **Rota:** /tutorial/{id}
- **Descrição:** Atualiza um tutorial existente.
- **Requisitos:** Implementar a lógica para atualização de um tutorial com base no ID.

4. Remover Tutorial

- **Método:** DELETE
- **Rota:** /tutorial/{id}
- **Descrição:** Remove um tutorial existente.
- **Requisitos:** Implementar a lógica para remoção de um tutorial com base no ID.

O que será um Diferencial

- Manutenibilidade do Código
- Uso de Docker
- Uso de Design Patterns e princípios SOLID.
- Documentação

- Uma cobertura de testes consistente
- Modelagem de Dados
- Cuidado com a segurança
- Tratamento de erros
- Arquitetura (estruturar o pensamento antes e descrever na entrega o motivo de a ter utilizado)
- Preocupação em desacoplar componentes (outras camadas, por ex: service, repository)

Desafio - Code Review

Dê uma olhada no seguinte trecho de código e forneça feedback sobre possíveis melhorias:

```
function process(items: any) {  
  var total = 0;  
  for (var i = 0; i < items.length; i++) {  
    if (items[i] > 0) {  
      total = total + items[i];  
    }  
  }  
  
  for (var j = 0; j < items.length; j++) {  
    console.log(items[j] + ' is a positive number');  
  }  
  
  return total;  
}  
  
// Example usage  
var numbers = [1, -2, 3, -4, 5];  
var result = process(numbers);  
console.log('Total:', result);
```

Entrega

Quando finalizar, envie um e-mail com os dados a seguir, junto com uma breve explicação das suas escolhas e abordagens.



Remetentes:

-

- guilherme.carvalho@b2bti.com

philype.castelo@sga.bet

Corpo do Email:

- Seu Nome
- Nome do recrutador
- Link do repositório
- Link do LinkedIn

Referências

[NestJS](#)

[Typescript](#)