

# Desafio Técnico Back-end

Last updated by | Luiz Felipe dos Santos | 20/06/2023 at 17:43 GMT-3

---

## Tecnologias/Padrões

Devem ser utilizadas as seguintes tecnologias :

- Escolha um dos frameworks: .NET 6 para C#, SpringBoot para Java e Nest.JS para Typescript;
- Utilizar MariaDB/MySQL como banco de dados;
- Lembre de utilizar orientação objeto;
- Documentar os endpoints utilizando Swagger;
- Testes unitários são **desejáveis**;
- Qualquer tecnologia ou padrão complementar as citadas anteriormente são permitidas com as devidas justificativas no README.

## O desafio






Crie uma API Rest de uma loja de quadrinhos bem semelhante a API Da Marvel.

API: <https://developer.marvel.com> 

### Modo básico do App consiste

Implementar os seguintes endpoint:

- Listar quadrinhos;
- Excluir quadrinhos;
- Cadastrar quadrinhos;
- Atualizar um quadrinho;
- Consultar um quadrinho por Id

-  1. Qualquer funcionalidade extra é bem vinda para agregar na solução básica proposta.
-  2. Os response não precisa ser exatamente o mesmo da API da Marvel
-  3. A arquitetura é por sua conta, seja esperto.
-  4. Você será avaliado pela qualidade do código, pela modularidade, pela legibilidade, pela criatividade, pela quantidade de funcionalidades básicas e extra.
-  5. Otimização de velocidade do app e experiência do usuário são **diferenciais** importantes.

### Modo avançado (Diferencial):

- Implementar uma forma de autenticação para API, de preferência, utilizando JWT token;
- Implementar um endpoint para modificar a raridade dos quadrinhos:  
Podemos ter quadrinhos do tipo, **raro** e **comum**.
- Implementar um endpoint para obter cupons de desconto:  
Aleatoriamente gerar os cupons do tipo raro e comum;  
Cupons comuns, só terão validade em quadrinhos comuns;
- Implementar um endpoint para submeter uma encomenda de quadrinhos  
Neste endpoint, é possível utilizar um cupom gerado. Cada cupom só pode ser utilizado uma única vez por usuário.

Não existe prazo.

## Arquitetura da solução/O que fazer

No arquivo README do projeto explique o funcionamento e a solução adotada na sua implementação do desafio.

## Avaliação

Entre os critérios de avaliação estão:

- Usabilidade
- Criatividade
- Código limpo e organização
- Documentação de código
- Documentação do projeto (readme)
- Performance

Qualquer dúvida ou sugestão, pode entrar em contato conosco.