

# C# Full Stack Developer

## Parte 1:

Vamos exercitar o desenvolvimento de uma arquitetura Back-End, ok?

Crie um sistema Back-End, que mantém os dados de uma Entidade "Produto". Essa entidade possui os atributos nome, descrição e valor. Esse Back-End deve prover toda a manutenção dessa Entidade "Produto", como incluir Produto, Remover Produto, etc.

Deve ser usada a stack DotNet C# (DotNet Core, Clássico) para a construção deste Back-End de APIs.

---

## Parte 2:

Vamos exercitar o desenvolvimento de uma arquitetura Front-End, ok?

Crie um sistema Front-End, que apresente os dados de sua Entidade Produto, da Parte 1.

Esse Front-End deve prover toda a interface gráfica para manutenção da Entidade Produto, como incluir Produto, Remover Produto etc.

Deve ser usada a stack ReactJS para a construção deste Front-End. Caso não tenha conhecimento em ReactJS pode ser APS NET, ASP NET Core, MVC etc.

---

## Parte 3:

Crie uma infraestrutura para esses sistemas, com as ferramentas Docker e Docker Compose.

Nessa infraestrutura deverão existir 3 servidores: Front-End-Server, Back-End-Server e DB-Server.

No servidor Back-End-Server deve ser instalado o sistema da Parte 1.

No servidor Front-End-Server deve ser instalado o sistema da Parte 2.

No servidor DB-Server deve ser instalado o banco de dados dos sistemas.

O Banco de Dados deve ser PostgreSQL ou MySQL.

Crie um [README.md](#) com instruções para instalação e inicialização dos sistemas em modo desenvolvimento, ou seja, na máquina local.

---

## Observações:

Consulte as documentações oficiais das ferramentas , todas elas possuem Tutoriais muito bons para serem seguidos e servirem como base.

Sugerimos enriquecer seu resultado com mecanismos e middleware de Autenticação, Autorização e Logging etc.

O resultado deste desafio deve ser publicado em um repositório público em seu Github, para que possamos analisar seu código fonte.

Um bom Desafio para você ! Esperamos que você se divirta e aprenda com ele! Bom Trabalho!