

# Checkpoint Integrador

## Introdução

Olá Houser, como você está? Ao longo das aulas de Back-end, Front-end e Infraestrutura, temos trabalhado no desenvolvimento de aplicações onde as funcionalidades são construídas de maneira evolutiva. Contudo, a esta altura, uma pergunta que pode surgir é: Como eu estabeleço conexão entre o Back-end e o Front-end? Ou ainda: Como eu faço para publicar esta aplicação em um servidor na web?

Fique tranquilo, pois vamos desvendar juntos este mistério, guiando-o até a solução. Portanto, acompanhe-nos nesta jornada e seja bem-vindo à **CTD Commerce**.

Durante os últimos dois anos, especialmente em virtude do isolamento social, as pessoas modificaram seus hábitos de consumo. Em geral, muitas pessoas têm optado por fazer suas compras pelo comércio eletrônico. Para termos uma ideia, o e-commerce brasileiro cresceu cerca de 72% no primeiro trimestre de 2021 (comparado com o mesmo período de 2020) e o ticket médio de compras subiu 9,4% chegando a R\$447,90, segundo o relatório da empresa de consultorias NeoTrust<sup>1</sup>.

A **CTD Commerce** é uma empresa que atua neste segmento de mercado e desenvolve soluções em comércio eletrônico. Neste projeto, temos que desenvolver um protótipo de e-commerce para um dos clientes da CTD Commerce.

Atentem-se para o briefing que o cliente da CTD Commerce passou para o desenvolvimento deste protótipo.

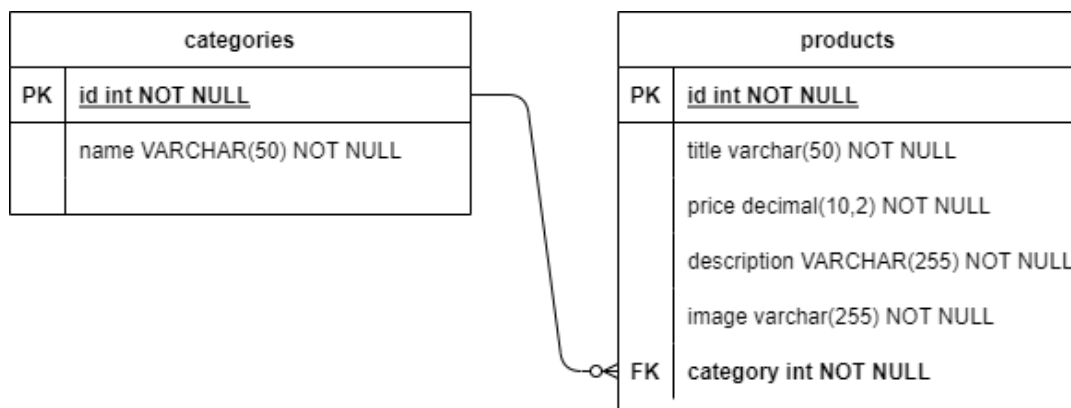
## Requisitos gerais:

- O trabalho será desenvolvido em grupo de 4 ou 5 pessoas (**Requisito indispensável**);
- Tanto o front quanto o back-end deverão constar em um (ou dois) repositórios (Github);
- Prazo: o trabalho deverá ser entregue **até o dia 10/12/2021**.  
{Dica importante: dividam as responsabilidades entre os integrantes do grupo, de forma que as atividades sejam feitas em paralelo};
- A entrega deverá ser realizada pelo formulário a seguir, de forma que sejam indicados os papéis executados pelos integrantes do grupo:

<https://forms.gle/ESCVYezHdDECMX676>

## Requisitos Back-end:

1. Você deverá fazer o back-end de nosso e-commerce em Java com o Framework Spring no modelo MVC.
2. O banco de dados deverá ser o H2.
3. Modelo de dados: O banco de dados terá duas entidades principais, **products** e **categories**. Onde cada produto tem apenas uma categoria e cada categoria pode ter vários ou nenhum produto. Segue o modelo para o banco de dados:



#### 4. API - O projeto deverá disponibilizar uma API com quatro end-points:

- a. **Método GET** - '<https://ctdcommerce.com/products>': este end-point deverá disponibilizar os dados de todos os produtos cadastrados em um JSON com o formato que segue:

```
//output
[
  {
    id:1,
    title:'...',
    price:'...',
    category:'...',
    description:'...',
    image:'...'
  },
  /*...*/
]
```

- b. **Método GET** - <https://ctdcommerce.com/products/1>: este end-point deverá disponibilizar os dados de um produto específico em um JSON com o formato que segue:

```
//output
{
  id:1,
  title:'...',
  price:'...',
  category:'...',
  description:'...',
  image:'...'
}
```

- c. **Método GET** - <https://ctdcommerce.com/products/categories>: este end-point deverá disponibilizar uma lista de categorias cadastradas em um JSON com o formato que segue:

```
//output
[
  "electronics",
  "jewelery",
  "men's clothing",
  "women's clothing"
]
```

- d. **Método GET** - <https://ctdcommerce.com/products/category/jewelery>: este end-point deverá disponibilizar os produtos de uma determinada categoria, em um JSON com o formato a seguir:

```
//output
[
  {
    id:5,
    title:'...',
    price:'...',
    category:'jewelery',
    description:'...',
    image:'...'
  }
]
/*...*/
```

## Requisitos Front-end:

1. Todo o site deve estar responsivo.
2. O entregável será uma página em HTML5, SASS, React-Bootstrap e [React](#).
3. Utilize a API criada no trabalho de Back-end acima para fornecer os dados necessários pro site.
4. O head deve variar de acordo com a página atual. Ex: Na página do produto leite ficaria: nomeDoProjeto | Leite
5. Header deve conter 4 itens além da logo. Cada item deve linkar com uma parte do site. Exemplo: Home, Produtos, Carrinho, Sobre nós.
6. Na home, faça um carousel de duas páginas com 3 produtos cada, ao clicar no produto, vai para a página do mesmo. Uma seção que exibe as categorias dos produtos que quando clicada, vai para a página de produto, filtrada por categoria.
7. Dentro da página de produtos, haverá uma opção para filtrar os produtos de acordo com a categoria.
8. Seu site tem de ter uma página de detalhes para cada item listado onde haverá uma explicação sobre o mesmo. Ex: A página de uma bolsa teria um título, descrição, categoria, imagem, avaliação, etc.
9. Na tela do produto, deve ser possível adicioná-lo num carrinho, mas esse carrinho ficará em um estado dentro da contextAPI e vai ser compartilhado pela aplicação inteira. Posteriormente, quando o usuário clicar na aba de carrinho, os dados serão buscados da contextAPI e a lista de produtos será listada.
10. Não se esqueça de colocar todos os membros da equipe na página de sobre nós.

## Requisitos Infraestrutura:

1. A API deverá ser implementada e disponibilizada por meio de um ambiente de cloud computing EC2 da Amazon Web Services.
2. O front-end deverá ser hospedado em um ambiente [Vercel](#).

## Referências:

1. **Comércio eletrônico brasileiro cresce 72,2% no primeiro trimestre de 2021**

Por Roseli Andrion | 02 de Julho de 2021

<https://canaltech.com.br/e-commerce/comercio-eletronico-brasileiro-cresce-722-no-primeiro-trimestre-de-2021-188932/>

## Referências técnicas:

[Introdução – React documentation](#)

[React-Bootstrap · React-Bootstrap Documentation](#)

[Spring Initializr](#)