8 - Criando uma loja com rotas dinâmicas

E aí, programador! o/

Nessa aula vamos criar uma página usando rotas dinâmicas para exibir um produto específico.

1. Primeiro de tudo, vamos criar o componente ProductDetails.tsx, que servirá para exibir as informações completas do produto na página:

```
// src/components/ProductDetails.tsx

import Image from "next/image";
import React, { useState } from "react";
import { Button, Col, Row } from "reactstrap";
import { ProductType } from "../services/products";
import SuccessToast from "./SuccessToast";

type ProductDetailsProps = {
  product: ProductType
}

const ProductDetails: React.FC<ProductDetailsProps> = ({ product }) => {
  const [toastIsOpen, setToastIsOpen] = useState(false)

  return (
    <Row>
      <Col lg={6}>
        <Image
          src={product.imageUrl}
          alt={product.name}
          height={500}
          width={600}
        />
      </Col>

      <Col lg={6}>
        <h1>{product.name}</h1>

        <h2 className="text-muted">R$ {product.price}</h2>

        <p className="my-3">
          <span className="d-block font-weight-bold">Descrição:</span>
          {product.description}
        </p>
```

```
        <p className="text-muted">Em estoque: {product.inStock}</p>

        <Button
          color="dark"
          className="my-3 pb-2"
        >
          Compre agora
        </Button>

        <SuccessToast toastIsOpen={toastIsOpen} setToastIsOpen={setToastIsOpen} />
      </Col>
    </Row>
  )
}

export default ProductDetails
```

2. Agora já podemos criar nossa página com rotas dinâmicas. Crie uma pasta “products” dentro de “pages” e então um arquivo chamado [id].tsx, semelhante ao que fizemos com a API. A primeira coisa que faremos aqui será obter o produto da API através do getStaticProps:

```
// pages/products/[id].tsx

import { GetStaticPaths, GetStaticProps, NextPage } from "next";
import Head from "next/head";
import { ReactNode } from "react";
import { Container } from "reactstrap";
import Header from "../../src/components/Header";
import ProductDetails from "../../src/components/ProductDetails";
import { fetchProduct, fetchProducts, ProductType } from "../../src/services/products";

export const getStaticProps: GetStaticProps = async (context) => {
  const id = context.params?.id

  if (typeof id === 'string') {
    const product = await fetchProduct(id)

    return { props: { product }, revalidate: 10  }
  }

  return { redirect: { destination: '/products', permanent: false } }
}
```

3. Depois disso vamos utilizar outra função do Next.js chamada getStaticPaths. Ela é usada em conjunto com o getStaticProps em rotas dinâmicas para que o Next.js saiba quais são todos as rotas possíveis. Só precisamos retornar aqui uma propriedade paths, que é um array de objetos contendo uma propriedade params, que é um objeto com todos os parâmetros da rota. Também utilizamos a propriedade fallback: false para dizer que se o id não for encontrado o Next.js pode retornar uma página com erro 404, ou seja, não há nenhum fallback nesse caso:

```
// pages/products/[id].tsx

// ...

export const getStaticPaths: GetStaticPaths = async () => {
  const products = await fetchProducts()

  const paths = products.map(product => {
    return { params: { id: product.id.toString() } }
  })

  return { paths, fallback: false }
}
```

4. Por fim, adicionamos o nosso componente da página:

```
// pages/products/[id].tsx

// ...

const Product: NextPage = (props: {
  children?: ReactNode
  product?: ProductType
}) => {
  return (
    <div>
      <Head>
        <title>{props.product!.name}</title>
        <meta name="description" content={props.product!.description} />
        <link rel="icon" href="/favicon.ico" />
      </Head>

      <Header />

      <Container className="mt-5">
```

```
        <ProductDetails product={props.product!} />
      </Container>
    </div>
  )
}

export default Product
```

5. Agora já podemos testar e ver que temos as páginas de detalhes de todos os nossos produtos.
6. Também já podemos subir a aplicação para o github e ver as mudanças na Vercel.