



وزارة الاتصالات
وتقنياتها المعلوماتية

National Academy
For Science & Skills



مدارس WE
التطبيقية
للتكنولوجيا

الجامعة العربية المفتوحة
جامعة العلوم والتكنولوجيا



Grade 3



IT Department
Web Development

On-site 4

ON-SITE GUIDE

2025 - 2026

Web Development On-Site 4

React Store APP - Practical Workshop for Students

In this hands-on project, you will build a React Product App that fetches data from an external API and displays it beautifully using Bootstrap.

[Home](#)[Docs](#)[GitHub](#)[Buy us a coffee](#)

Fake Store API

Realistic E-Commerce Data. Instantly.

A free RESTful API for developers, ideal for prototyping, testing, or teaching.
No backend setup required.

[Read Docs](#)[View on Github](#)

Learning Objectives:

By the end of this workshop, students will:

- How to use **React Context API** for global state
- Use **Axios** for fetching product data
- How to perform **Routing** with React Router
- How to display alerts using **SweetAlert2**
- How to use **React Helmet** for SEO
- How to handle **offline detection**
- How to add a **loading spinner** while data loads

Part 1 Using Props

1. Create a new React project

```
npm create vite@latest
```

Write Project Name: fake-store

Choose Framework: React

Choose JavaScript

```
cd fake-store
```

```
npm install
```

```
npm run dev
```

2. Install the required packages

```
npm install axios bootstrap react-router-dom
```

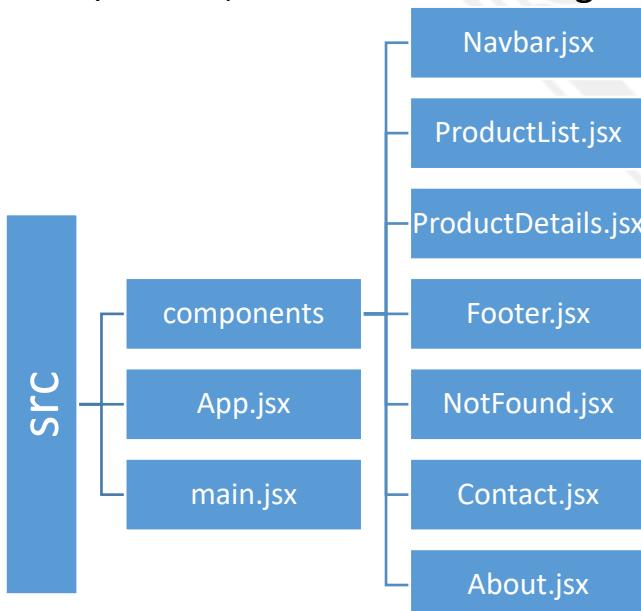
Then add Bootstrap to the project in src/main.jsx:

```
import 'bootstrap/dist/css/bootstrap.min.css';
import 'bootstrap/dist/js/bootstrap.bundle.min.js';
import { BrowserRouter } from 'react-router-dom';
```

```
<BrowserRouter>
  <App />
</BrowserRouter>
```

3. Create folder structure

Inside the src/ folder, create the following:



2. Create components Design

In components/Footer.jsx

```
export default function Footer() {  
  return (  
    <footer className="bg-dark text-light text-center p-3 mt-5">  
      &copy; {new Date().getFullYear()} FakeStore Inc.  
    </footer>  
  );  
}
```

In components/About.jsx

```
export default function About() {  
  return (  
    <div>About</div>  
  )  
}
```

In components/Contact.jsx

```
export default function Contact() {  
  return (  
    <div>Contact</div>  
  )  
}
```

In components/NotFound.jsx

```
export default function NotFound() {  
  return (  
    <div className="text-center mt-5">  
      <h2>404 - Page Not Found</h2>  
      <p>The page you're looking for doesn't exist.</p>  
    </div>  
  );  
}
```

In components/ ProductList.jsx

```
export default function ProductList() {  
  return ( <div>ProductList</div> )  
}
```

In components/Navbar.jsx

```
import { Link, NavLink } from 'react-router-dom';

export default function Navbar() {
  return (
    <nav className="navbar navbar-expand-md navbar-dark bg-dark">
      <div className="container">
        <Link className="navbar-brand" to="/"> FakeStore </Link>
        <button className="navbar-toggler" type="button" data-bs-
          toggle="collapse" data-bs-target="#navbarNavAltMarkup" aria-
          controls="navbarNavAltMarkup" aria-expanded="false" aria-label="Toggle
          navigation">
          <span className="navbar-toggler-icon" />
        </button>
        <div className="collapse navbar-collapse" id="navbarNavAltMarkup">
          <div className="navbar-nav">
            <NavLink className="nav-link" to="/">Home</NavLink>

            <NavLink className="nav-link" to="/products">Products</NavLink>

            <NavLink className="nav-link" to="/contact">Contact</NavLink>

            <NavLink className="nav-link" to="/about">About</NavLink>
          </div>
        </div>
      </div>
    </nav>
  );
}
```



3. Routing Setup

In App.jsx

```
import { Routes, Route } from 'react-router-dom';
import Navbar from './components/Navbar';
import Footer from './components/Footer';
import ProductList from './components/ProductList';
import NotFound from './components/NotFound';
import About from './components/About';
import Contact from './components/Contact';

export default function App() {
  return (
    <>
      <Navbar />
      <div className="container mt-4">
        <Routes>
          <Route path="/" element={<ProductList />} />
          <Route path="/products" element={<ProductList />} />
          <Route path="/about" element={<About />} />
          <Route path="/contact" element={<Contact />} />
          <Route path="*" element={<NotFound />} />
        </Routes>
      </div>
      <Footer />
    </>
  );
}
```



Fetching Data From API

First, try fetching data using `fetch`.

In components/ `ProductList.jsx`

```
import React, { useState , useEffect } from 'react'
```

```
export default function ProductList() {
```

```
  const [products , setProducts] = useState([]);
```

```
  useEffect(() => {
    fetch("https://fakestoreapi.com/products/")
      .then((res) => res.json())
      // .then((json) => console.log(json));
      .then((json) => setProducts(json));
  }, [])
```

```
  return (
    <div className="row">
      {products.map(product => (
        <div className="col-md-4 mb-4" key={product.id}>
          <div className="card h-100">
            <img src={product.image} className="card-img-top p-3"
              alt={product.title} style={{ height: '250px', objectFit: 'contain' }} />
            <div className="card-body">
              <h5 className="card-title">{product.title}</h5>
              <p className="card-text text-truncate">{product.description}</p>
              <p className="card-text fw-bold">${product.price}</p>
            </div>
          </div>
        </div>
      ))}
    </div>
  )
}
```



Second, try fetching data using Axios.

```
import React, { useState , useEffect } from 'react'  
import axios from 'axios'  
export default function ProductList() {  
  const [products , setProducts] = useState([]);
```

```
const getProducts =async ()=>{  
  const response = await axios.get("https://fakestoreapi.com/products/");  
  setProducts(response.data)  
  //console.log(products);  
} //getAllProducts
```

```
useEffect(() => {  
  getProducts()  
, [])
```

```
  return (
```

Note It is better to create a separate component for the product card and call it through the ProductList.

Add In components folder ProductCard.jsx

```
export default function ProductCard() {  
  return (  
    <div>ProductCard</div>  
  ) }
```

In ProductList.jsx

```
return (  
  <div className="row">  
    {products.map(product => (  
      <ProductCard key={product.id} product={product}/>  
    ))}  
  </div>  
)
```

In ProductCard.jsx

```
export default function ProductCard({ product }) {
  return (
    <div className="col-md-4 mb-4" >
      <div className="card h-100">
        <img src={product.image} className="card-img-top p-3" alt={product.title} style={{ height: '250px', objectFit: 'contain' }} />
        <div className="card-body">
          <h5 className="card-title">{product.title}</h5>
          <p className="card-text text-truncate">{product.description}</p>
          <p className="card-text fw-bold">${product.price}</p>
        </div>
      </div>
    </div>
  )
}
```

Displaying details for a single product

1. In component ProductDetails.jsx

```
import { useParams } from 'react-router-dom';
import { useEffect, useState } from 'react';
import axios from 'axios';
```

```
export default function ProductDetails() {
  const { id } = useParams();
  const [product, setProduct] = useState(null);

  const getProduct = async () => {
    try {
      const res = await axios.get(`https://fakestoreapi.com/products/${id}`);
      setProduct(res.data);
    } catch (err) {
      console.error(err);
    }
  };
}
```

```
useEffect(() => {
  getProduct();
}, [id]);
```

```
if (!product) return <p>Loading...</p>;
```

```
return (
<div className="card mb-3 p-3">
  <div className="row">
    <div className="col-md-4">
      <img src={product.image} alt={product.title} className="img-fluid" />
    </div>
    <div className="col-md-8">
      <h3>{product.title}</h3>
      <p className="text-muted">{product.category}</p>
      <p>{product.description}</p>
      <h4>${product.price}</h4>
    </div>
  </div>
</div>
);
}
```

2. Modify App.js to add a product details route.

```
import ProductDetails from './components/ProductDetails';
// Within <Routes>
<Route path="/product/:id" element={<ProductDetails />} />
```

3. Modify a button in ProductCard.jsx to link each product to its details.

Replace the <h5> in the card with:

```
<Link to={`/product/${product.id}`} className="text-decoration-none text-dark">
  <h5 className="card-title">{product.title}</h5>
</Link>
```

And don't forget:

```
import { Link } from 'react-router-dom';
```



Display products based on categories

In Products List.js file

```
function ProductsList() {  
  const [products , setProducts] = useState([]);  
  const [categories , setCategories] = useState([]);  
  const getProducts =async ()=>{  
    const response = await axios.get("https://fakestoreapi.com/products/");  
    setProducts(response.data)  
  } //getAllProducts
```

```
const getCategories =async ()=>{  
  const response = await  
  axios.get("https://fakestoreapi.com/products/categories");  
  setCategories(response.data)  
  // console.log(categories);  
} //getCategories
```

```
useEffect(() => {  
  getProducts();  
  getCategories();  
}, [])
```

```
const getProductsByCategory = async(category)=>{  
  const response = await  
  axios.get(`https://fakestoreapi.com/products/category/${category}`);  
  
  setProducts(response.data)  
  //console.log(products);  
} //getProductsByCategory
```

```
<div className="mb-3">  
  <button className="btn btn-outline-primary me-2"  
  onClick={getProducts}>All</button>  
  {categories.map(cat => (  
    <button key={cat} className="btn btn-outline-secondary me-2"  
    onClick={() => getProductsByCategory(cat)}>
```



```
{cat}  
</button>  
))}  
</div>
```

```
import React, { useState , useEffect } from 'react'  
import axios from 'axios'  
import ProductCard from './ProductCard';  
export default function ProductList() {
```

```
  const [products , setProducts] = useState([]);  
  const [categories , setCategories] = useState([]);
```

```
  const getProducts =async ()=>{  
    const response = await axios.get("https://fakestoreapi.com/products/");  
    setProducts(response.data)  
    //console.log(products);  
  } //getAllProducts
```

```
  const getCategories =async ()=>{  
    const response = await  
    axios.get("https://fakestoreapi.com/products/categories");  
    setCategories(response.data)  
    // console.log(categories);  
  } //getCategories
```

```
  const getProductsByCategory = async(category)=>{  
    const response = await  
    axios.get(`https://fakestoreapi.com/products/category/${category}`);  
    setProducts(response.data)  
    //console.log(products);  
  } //getProductsByCategory
```

```
useEffect(() => {  
  getProducts();  
  getCategories();  
}, [])
```

```
return (
```

<>



مدارس تكنولوجيا
التطبيقية



```
<div className="mb-3">
  <button className="btn btn-outline-primary me-2"
    onClick={getProducts}>All</button>
  {categories.map(cat => (
    <button key={cat} className="btn btn-outline-secondary me-2"
      onClick={() => getProductsByCategory(cat)}>
      {cat}
    </button>
  )))
</div>

<div className="row">
{products.map(product => (
  <ProductCard key={product.id} product={product}/>
))
}
</div> /* <div className="row"> */
</>
)
}
```

Part2 Using Context API

1. Create a new React project

```
npm create vite@latest
Write Project Name: fake-store
Choose Framework: React
Choose JavaScript
cd fake-store
npm install
npm run dev
```

2. Install the required packages

```
npm install axios bootstrap react-router-dom
```

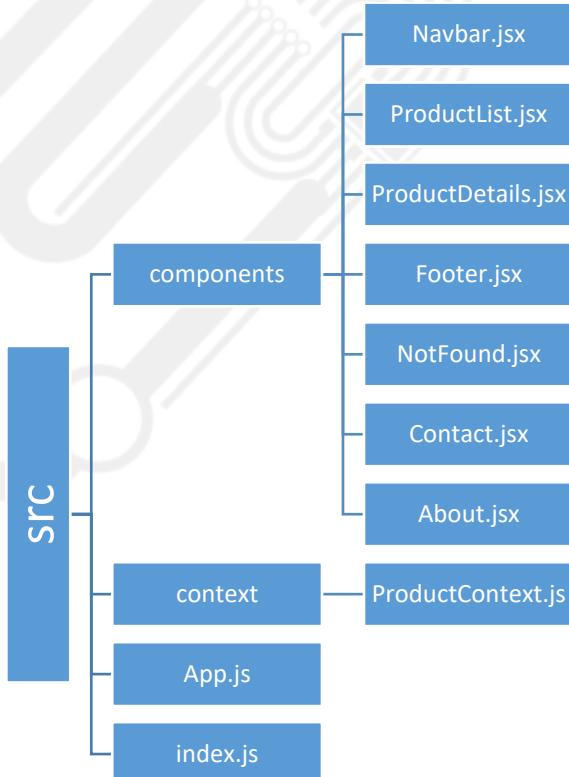
Then add Bootstrap to the project in src/main.js:

```
import 'bootstrap/dist/css/bootstrap.min.css';
import 'bootstrap/dist/js/bootstrap.bundle.min.js';
import { BrowserRouter } from 'react-router-dom';
```

```
<BrowserRouter>
  <App />
</BrowserRouter>
```

3. Create folder structure

Inside the src/ folder, create the following:



2. Create components Design

In components/Footer.jsx

```
export default function Footer() {  
  return (  
    <footer className="bg-dark text-light text-center p-3 mt-5">  
      © {new Date().getFullYear()} FakeStore Inc.  
    </footer>  
  );  
}
```

In components/About.jsx

```
export default function About() {  
  return ( <div>About</div> )  
}
```

In components/Contact.jsx

```
export default function Contact() {  
  return ( <div>Contact</div> )  
}
```

In components/NotFound.jsx

```
export default function NotFound() {  
  return (  
    <div className="text-center mt-5">  
      <h2>404 - Page Not Found</h2>  
      <p>The page you're looking for doesn't exist.</p>  
    </div>  
  );  
}
```

In components/ ProductList.jsx

```
export default function ProductList() {  
  return ( <div>ProductList</div> )  
}
```

In components/Navbar.jsx

```
import { Link, NavLink } from 'react-router-dom';
```

```
export default function Navbar() {
```



```
return (
<nav className="navbar navbar-expand-md navbar-dark bg-dark ">
  <div className="container ">
    <Link className="navbar-brand" to="/"> FakeStore </Link>
    <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavAltMarkup" aria-controls="navbarNavAltMarkup" aria-expanded="false" aria-label="Toggle navigation">
      <span className="navbar-toggler-icon" />
    </button>

    <div className="collapse navbar-collapse" id="navbarNavAltMarkup">
      <div className="navbar-nav">
        <NavLink className="nav-link" to="/">Home</NavLink>
        <NavLink className="nav-link" to="/products">Products</NavLink>
        <NavLink className="nav-link" to="/contact">Contact</NavLink>
        <NavLink className="nav-link" to="/about">About</NavLink>
      </div>
    </div>
  </div>
</nav>
);
}
```

3. Routing Setup

In App.js

```
import { Routes, Route } from 'react-router-dom';

import Navbar from './components/Navbar';
import Footer from './components/Footer';
import ProductList from './components/ProductList';
import NotFound from './components/NotFound';
import About from './components/About';
import Contact from './components/Contact';
```

```
export default function App() {
  return (
    <>
    <Navbar />
    <div className="container mt-4">
```



```
<Routes>
  <Route path="/" element={<ProductList />} />
    <Route path="/products" element={<ProductList />} />
    <Route path="/about" element={<About />} />
    <Route path="/contact" element={<Contact />} />
    <Route path="*" element={<NotFound />} />
</Routes>
</div>
<Footer />
</>
);
}
```

4. Setting up a ProductContext (using the Context API) in context/ProductContext.js

```
import { createContext, useEffect, useState } from 'react';
import axios from 'axios';
// Create a context for the product data
// This context will be used to provide the product data to the components
export const ProductContext = createContext();
```

```
export function ProductProvider({ children }) {
  const [products, setProducts] = useState([]);
```

```
const getAllProducts = async () => {
  try {
    const response = await axios.get('https://fakestoreapi.com/products');
    setProducts(response.data);
  } catch (error) {
    console.error('Error fetching products:', error);
  }
}; // getAllProducts
```

```
useEffect(() => {
  getAllProducts();
}, []);
```

```

return (
  <ProductContext.Provider
    value={{ products }}>
    {children}
  </ProductContext.Provider>
);

```

// ProductProvider



5. Wrap the application with a ProductProvider in index.js

```
import { ProductProvider } from './context/ProductContext';
```

```

<ProductProvider>
  <BrowserRouter>
    <App />
  </BrowserRouter>
</ProductProvider>

```

6. Use Product Context in Product List.js

In components/Product List.js

```
import { useContext } from 'react';
import { ProductContext } from '../context/ProductContext';
```

```

export default function ProductList() {
  const { products } = useContext(ProductContext);
  return (
    <div className="row">
      {products.map(product => (
        <div className="col-md-4 mb-4" key={product.id}>
          <div className="card h-100">
            <img src={product.image} className="card-img-top p-3" alt={product.title}
              style={{ height: '250px', objectFit: 'contain' }} />
            <div className="card-body">
              <h5 className="card-title">{product.title}</h5>
              <p className="card-text text-truncate"> {product.description} </p>
              <p className="card-text fw-bold">${product.price}</p>
            </div> /* cart-body */
          </div>
      ))}
    </div>
  );
}
```

```
</div> /* cart */  
    </div>  
  )}  
  </div>  
);  
}
```

7 Displaying details for a single product

In ProductDetails.jsx component:

```
import { useParams } from 'react-router-dom';  
import { useEffect, useState } from 'react';  
import axios from 'axios';  
  
export default function ProductDetails() {  
  const { id } = useParams();  
  const [product, setProduct] = useState(null);
```

```
const getProduct = async () => {  
  try {  
    const res = await axios.get(`https://fakestoreapi.com/products/${id}`);  
    setProduct(res.data);  
  } catch (err) {  
    console.error(err);  
  }  
}; // getProduct
```

```
useEffect(() => {  
  getProduct();  
}, [id]);  
  
if (!product) return <p>Loading...</p>;  
  
return (  
  <div className="card mb-3 p-3">  
    <div className="row">  
      <div className="col-md-4">  
        <img src={product.image} alt={product.title} className="img-fluid" />  
      </div>
```

```

<div className="col-md-8">
  <h3>{product.title}</h3>
  <p className="text-muted">{product.category}</p>
  <p>{product.description}</p>
  <h4>${product.price}</h4>
</div>
</div>
);
} // ProductDetails

```

Modify App.js to add a product details route.

```
import ProductDetails from './components/ProductDetails';
```

```
// Within <Routes>
```

```
<Route path="/product/:id" element={<ProductDetails />} />
```

In ProductList.jsx Modify a button to link each product to its details. Replace the **<h5>** in the card with:

```
<Link to={`/product/${product.id}`} className="text-decoration-none text-dark">
  <h5 className="card-title">{product.title}</h5>
</Link>
```

And don't forget:

```
import { Link } from 'react-router-dom';
```

Adding Categories

We'll fetch categories from the API and add them to a product filter list.

1. Edit ProductContext.js to include categories.

```
export function ProductProvider({ children }) {
```

```
  const [products, setProducts] = useState([]);
  const [categories, setCategories] = useState([]);
```

```
  const getAllProducts = async () => {
    try {
      const res = await axios.get('https://fakestoreapi.com/products');
      setProducts(res.data);
    } catch (err) {
```



```
        console.error(err);
    }
};


```



```
const getCategories = async () => {
  try {
    const res = await
    axios.get('https://fakestoreapi.com/products/categories');
    setCategories(res.data);
  } catch (err) {
    console.error(err);
  }
};


```

```
const getProductsByCategory = async (category) => {
  try {
    const res = await
    axios.get(`https://fakestoreapi.com/products/category/${category}`);
    setProducts(res.data);
  } catch (err) {
    console.error(err);
  }
};


```

```
useEffect(() => {
  getAllProducts();
  getCategories();
}, []);
```

```
return (
  <ProductContext.Provider value={{ products, categories,
getProductsByCategory, getAllProducts }}>
    {children}
  </ProductContext.Provider>
);
}
```



2. Display the list of categories in ProductList.js

```
export default function ProductList() {  
  const { products, categories, getProductsByCategory, getAllProducts } =  
    useContext(ProductContext);  
  
  return (  
    <>  
      <div className="mb-3">  
        <button className="btn btn-outline-primary me-2"  
          onClick={getAllProducts}>All</button>  
  
        {categories.map(cat => (  
          <button key={cat} className="btn btn-outline-secondary me-2"  
            onClick={() => getProductsByCategory(cat)}>  
            {cat}  
          </button>  
        ))}  
      </div>  
  
      <div className="row">.....</div>  
    </>  
  );  
}
```



Part3 Library Explanations & Usage Notes

1. SweetAlert2 – Beautiful Pop-up Messages

What It Is

SweetAlert2 replaces the ugly built-in JavaScript alert() box with **modern, animated popup messages**.

You can use it for success, error, confirmation, or information messages.

Install:

```
npm install sweetalert2
```

Import the library:

```
import Swal from "sweetalert2";
```

Then show an alert:

```
Swal.fire("Title", "Message text", "success");
```

Why We Use It

- Makes alerts professional and user-friendly
- Customizable icons, buttons, and colors
- Easy to use anywhere in your app

Docs: <https://sweetalert2.github.io>

2. React Detect Offline – Handle No Internet

What It Is

React Detect Offline allows your app to **detect the user's connection status**.

You can show a message or a different layout when they go offline.

Install

```
npm install react-detect-offline
```

Basic Usage

Import the components:

```
import { Online, Offline } from "react-detect-offline";
```

Then use them:

```
<Online>
  <p>You are online!</p>
</Online>
<Offline>
  <p>You are offline!</p>
</Offline>
```

Why We Use It

- Improves user experience during bad connectivity
- Prevents confusion when API requests fail
- Automatically detects internet loss & recovery

Docs: <https://github.com/chrisbolin/react-detect-offline>



3-React Helmet – Dynamic Page Titles & Meta Tags

What It Is:

React Helmet lets you control the **HTML <head>** (title, meta tags) from React components. This improves **SEO** and makes each page unique in the browser tab.

Install

```
npm install react-helmet
```

Basic Usage

Import the component:

```
import { Helmet } from "react-helmet";
```

Then wrap your tags inside <Helmet>:

```
<Helmet>
  <title>Product Details - My Store</title>
  <meta name="description" content="View detailed info about this product." />
</Helmet>
```

Why We Use It

- Makes every page have a custom browser title
- Improves SEO for search engines
- Helps identify pages when multiple tabs are open

Docs: <https://www.npmjs.com/package/react-helmet>

4-React Spinners – Loading Animations

What It Is

React Spinners provides pre-built, animated loaders (spinners, bars, dots) to show when data is loading.

Install

```
npm install react-spinners
```

Basic Usage

Import a spinner:

```
import ClipLoader from "react-spinners/ClipLoader";
```

Show it conditionally:

```
if (loading) {
  return (
    <div className="d-flex justify-content-center align-items-center vh-100">
      <ClipLoader color="#0d6efd" size={60} />
    </div>
  );
}
```

Why We Use It

- Gives visual feedback during loading
- Prevents user confusion when data takes time
- Many animation styles available (Pulse, Ring, BarLoader, etc.)

Docs: <https://www.npmjs.com/package/react-spinners>

Phase 1: Create the Project

Step 1: Create React + Vite Project

```
npm create vite@latest react-product-app --template react
cd react-product-app
npm install
```

Step 2: Install Required Libraries

```
npm install axios react-router-dom bootstrap sweetalert2
react-helmet react-detect-offline react-spinners
```

Step 3: Project Folder Structure

```
src/
  |- context/
  |- pages/
  |- App.jsx
  |- main.jsx
  |- index.css
```

Phase 2: Set Up Context (Global Product State)

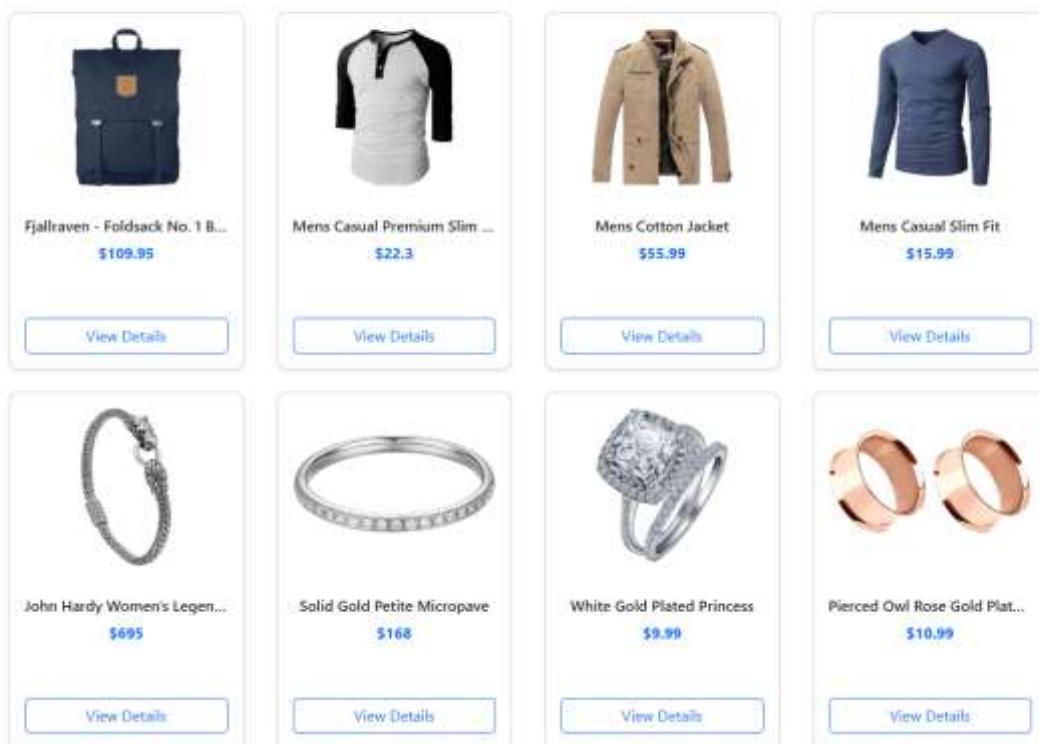
This file uses the Context API to fetch and share product data with the entire app.

src/context/ProductContext.jsx

```
import React, { createContext, useState, useEffect } from 'react';
import axios from 'axios';
export const ProductContext = createContext();
function ProductContextProvider({ children }) {
  const [products, setProducts] = useState([]);
  useEffect(() => {
    getProducts();
  }, []);
  function getProducts() {
    axios.get('https://fakestoreapi.com/products')
      .then(response => setProducts(response.data))
      .catch(error => console.error('Error fetching products:', error));
  }
  return (
    <ProductContext.Provider value={{ products, getProducts }}>
      {children}
    </ProductContext.Provider>
  );
}
export default ProductContextProvider;
```

Phase 3: Product List (Home Page)

Products



src/pages/Product.jsx

```

import React, { useContext } from "react";
import { ProductContext } from "../context/ProductContext";
import { Link } from "react-router-dom";
import "bootstrap/dist/css/bootstrap.min.css";
import ClipLoader from "react-spinners/ClipLoader"; // npm install react-spinners
import "./Product.css";

function Product() {
  const { products } = useContext(ProductContext);

  // Show spinner while products are loading
  if (products.length === 0) {
    return (
      <div className="loading-screen d-flex justify-content-center align-items-center">
        <ClipLoader color="#0d6efd" size={60} />
      </div>
    );
  }
  // Product Grid
}

```

```
return (
  <div className="container my-5
text-center">
  <h1 className="fw-bold mb-5">Products</h1>

  <div className="row justify-content-center align-items-center g-4">
    {products.map((product) => (
      <div
        className="col-sm-6 col-md-4 col-lg-3 d-flex justify-content-center"
        key={product.id}
      >
        <div className="card product-card shadow-sm text-center">
          <img
            src={product.image}
            alt={product.title}

          className="card-img-top p-3"
        />
        <div className="card-body d-flex flex-column justify-content-between">
          <h6 className="card-title text-truncate">{product.title}</h6>
          <p className="fw-bold text-primary">${product.price}</p>
          <Link
            className="btn btn-outline-primary mt-auto"
            to={`/products/${product.id}`}
          >
            View Details
          </Link>
        </div>
      </div>
    )));
  </div>
</div>
);

}

export default Product;
```

src/pages/Product.css

```
/* Perfect centering for loading screen */
.loading-screen {
  width: 100vw;
  height: 100vh;
  display: flex;
  justify-content: center;
  align-items: center;
```



```
background-color: #f8f9fa;  
}  
.product-card {  
width: 100%;  
max-width: 250px;  
height: 380px;  
display: flex;  
flex-direction: column;  
justify-content: space-between;  
border-radius: 10px;  
transition: transform 0.3s ease, box-shadow 0.3s ease;  
}  
.product-card:hover {  
transform: translateY(-5px);  
box-shadow: 0 8px 16px rgba(0,0,0,0.15);  
}  
.product-card img {  
width: 100%;  
height: 200px;  
object-fit: contain;  
}  
.loading-screen {  
background-color: #f8f9fa;  
}
```

Phase 4: Routing Setup

This uses **React Router v6** for page navigation and **React Detect Offline** to show an alert when the user loses internet.

Src/app.jsx

```
import { Offline, Online } from "react-detect-offline";  
import Product from "./pages/Product";  
import ProductDetails from "./pages/ProductDetails";  
import { Routes, Route } from "react-router-dom";  
  
function App() {  
return (  
<>  
<Online>  
  <Routes>  
    <Route path="/" element={<Product />} />  
    <Route path="/products/:id" element={<ProductDetails />} />  
  </Routes>  
</Online>
```



<Offline>

```
<div className="offline-
message d-flex justify-content-center align-items-center">
  <div className="alert alert-danger text-center p-4 rounded shadow-sm">
    You are offline. Please check your internet connection.
  </div>
</div>
</Offline>
</>
);
}
```

```
export default App;
```

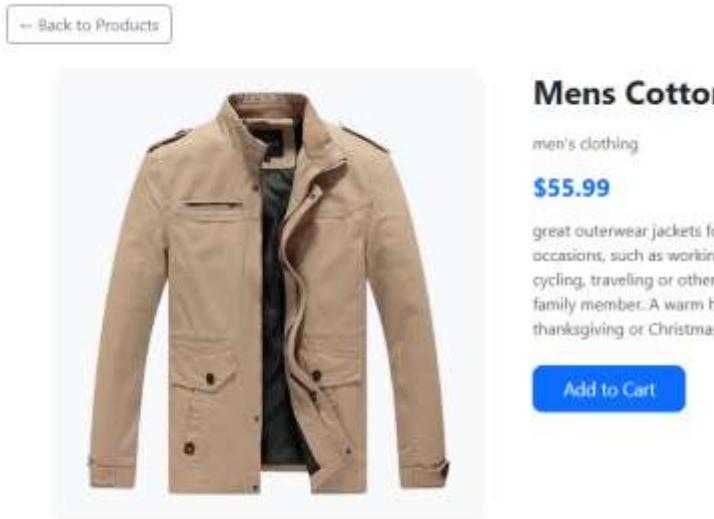
Phase 5: Main Entry Point

Src/main.jsx

```
import { StrictMode } from 'react';
import { createRoot } from 'react-dom/client';
import App from './App.jsx';
import './index.css';
import ProductContextProvider from './context/ProductContext.jsx';
import { BrowserRouter } from 'react-router-dom';

createRoot(document.getElementById('root')).render(
<StrictMode>
  <ProductContextProvider>
    <BrowserRouter>
      <App />
    </BrowserRouter>
  </ProductContextProvider>
</StrictMode>
);
```

Phase 6: Product Details Page



src/pages/ProductDetails.jsx

```
import React, { useEffect, useState } from "react";
import { useParams, Link } from "react-router-dom";
import axios from "axios";
import Swal from "sweetalert2";
import { Helmet } from "react-helmet";
import "bootstrap/dist/css/bootstrap.min.css";
import "./ProductDetails.css";

function ProductDetails() {
  const { id } = useParams();
  const [product, setProduct] = useState(null);
  useEffect(() => {
    axios
      .get(`https://fakestoreapi.com/products/${id}`)
      .then((response) => setProduct(response.data))
      .catch((error) => console.error("Error fetching product details:", error));
  }, [id]);
  if (!product) {
    return (
      <div className="loading-screen d-flex justify-content-center align-items-center vh-100">
        <div className="spinner-border text-primary" role="status">
          <span className="visually-hidden">Loading...</span>
        </div>
      </div>
    );
  }
  return (
    <div>
      <div>
        <img alt="Product image" src={product.image} />
        <div>
          <h3>{product.title}</h3>
          <p>Category: {product.category}</p>
          <p>Price: ${product.price}</p>
          <p>Description:<br/>{product.description}</p>
          <button type="button" onClick={() => Swal.fire("Added to Cart!")}>Add to Cart</button>
        </div>
      </div>
    </div>
  );
}

export default ProductDetails;
```

```
});
```

```
const addToCart = () => {
  Swal.fire({
    icon: "success",
    title: "Added to Cart!",
    text: `${product.title} has been added successfully.`,
    confirmButtonColor: "#0d6efd",
  });
};

return (
  <div className="container product-details-page my-5">
    <Helmet>
      <title>{product.title} - Product Details</title>
    </Helmet>

    <Link to="/" className="btn btn-outline-secondary mb-4">
      ← Back to Products
    </Link>

    <div className="row align-items-start justify-content-center g-5">
      {/* LEFT: Product Image */}
      <div className="col-md-5 col-sm-10 text-center">
        <div className="product-image-wrapper shadow-sm">
          <img
            src={product.image}
            alt={product.title}
            className="img-fluid product-image"
          />
        </div>
      </div>
    </div>

    {/* RIGHT: Product Details */}
    <div className="col-md-6 col-sm-10">
      <h2 className="fw-bold mb-3">{product.title}</h2>
      <p className="text-muted">{product.category}</p>
      <h4 className="text-primary fw-bold mb-3">${product.price}</h4>
      <p className="text-secondary mb-4">{product.description}</p>

      <button onClick={addToCart} className="btn btn-primary btn-lg">
        Add to Cart
      </button>
    </div>
  </div>
);
```



```
</div>
</div>
);
}
```

```
export default ProductDetails;
```

src/pages/ ProductDetalis.css

```
.product-details-page {
  max-width: 1100px;
  margin: 0 auto;
}

/* Product image section */
.product-image-wrapper {
  border-radius: 15px;
  background-color: #f8f9fa;
  padding: 20px;
  transition: transform 0.3s ease;
}

.product-image-wrapper:hover {
  transform: scale(1.02);
}

.product-image {
  max-height: 400px;
  object-fit: contain;
}

/* Details text styling */
h2 {
  line-height: 1.4;
}

.text-secondary {
  font-size: 1rem;
  color: #555 !important;
}

.btn-primary {
  background-color: #0d6efd;
  border: none;
  padding: 10px 30px;
  font-size: 1.1rem;
}
```



مدارس ناس
للتكنولوجيا
التطبيقية



```
border-radius: 10px;  
transition: background-color 0.3s  
ease, transform 0.2s ease;  
}
```

```
.btn-primary:hover {  
background-color: #0b5ed7;  
transform: scale(1.05);  
}
```

```
/* Center loading spinner */
```

```
.loading-screen {  
width: 100vw;  
height: 100vh;  
background-color: #f8f9fa;  
display: flex;  
justify-content: center;  
align-items: center;  
}
```