

Práctica Guiada 2: Creación de Componentes (Cabecera, Cuerpo y Pie) con Bootstrap

Objetivo: Diseñar tres componentes principales (Cabecera, Cuerpo y Pie) para la aplicación de productos informáticos en React, integrando Bootstrap para el estilo y mostrando un listado de productos (código, descripción, precio y disponible).

Paso 0: Requisitos previos

- Proyecto React creado (puede ser el de la práctica anterior).
- **Bootstrap instalado:**
npm install bootstrap
- Importar Bootstrap en src/index.js:

```
import 'bootstrap/dist/css/bootstrap.min.css';  
import 'bootstrap/dist/js/bootstrap.bundle.min.js';
```

Paso 1: Crear la estructura de carpetas

Dentro de src/ crea la carpeta components:

```
src/  
├── components/  
│   ├── Header.js  
│   ├── Footer.js  
│   └── ProductList.js  
├── data/  
│   └── catalog.js  
└── App.js
```

Paso 2: Crear el componente Cabecera (Header.js)

Ubicación: src/components/Header.js

```
import React from 'react';
```

```
export default function Header() {  
  return (  

```

```

<header className="bg-dark text-white p-3 mb-4">
  <div className="container">
    <h1 className="mb-0">Tienda de Productos Informáticos</h1>
    <nav className="mt-2">
      <ul className="nav">
        <li className="nav-item">
          <a className="nav-link text-white" href="#">Inicio</a>
        </li>
        <li className="nav-item">
          <a className="nav-link text-white" href="#">Productos</a>
        </li>
        <li className="nav-item">
          <a className="nav-link text-white" href="#">Contacto</a>
        </li>
      </ul>
    </nav>
  </div>
</header>
);
}

```

Paso 3: Crear el componente Pie (Footer.js)

Ubicación: src/components/Footer.js

```

import React from 'react';

export default function Footer() {
  return (
    <footer className="bg-light text-center p-3 mt-4">
      <div className="container">
        <p className="mb-0">© 2025 Tienda Informática - Todos los derechos reservados</p>
      </div>
    </footer>
  );
}

```

Paso 4: Crear el componente Cuerpo (ProductList.js)

Ubicación: src/components/ProductList.js

```

import React from 'react';

export default function ProductList({ products }) {
  if (!products || products.length === 0) {
    return <p className="text-center text-muted mt-3">No hay productos disponibles.</p>;
  }

  return (
    <div className="container">
      <h2 className="mb-3">Listado de Productos Informáticos</h2>
      <table className="table table-striped table-bordered">
        <thead className="table-dark">
          <tr>
            <th>Código</th>
            <th>Descripción</th>
            <th>Precio (€)</th>
            <th>Disponible</th>
          </tr>
        </thead>
        <tbody>
          {products.map((p) => (
            <tr key={p.id}>
              <td>{p.id}</td>
              <td>{p.nombre}</td>
              <td>{p.precio.toFixed(2)}</td>
              <td>
                <span className={`badge ${p.enStock ? 'bg-success' : 'bg-danger'}>
                  {p.enStock ? 'Sí' : 'No'}
                </span>
              </td>
            </tr>
          ))}
        </tbody>
      </table>
    </div>
  );
}

```

Paso 5: Integrar todo en App.js

Ubicación: src/App.js

```
import React from 'react';
import Header from './components/Header';
import Footer from './components/Footer';
import ProductList from './components/ProductList';
import { catalog } from './data/catalog';

export default function App() {
  return (
    <>
      <Header />
      <main className="mb-4">
        <ProductList products={catalog} />
      </main>
      <Footer />
    </>
  );
}
```

Paso 6: Ejecutar y verificar

1. Ejecuta el proyecto:

npm start

2. Abre <http://localhost:3000> y comprueba:

- Cabecera con navegación.
- Tabla de productos estilizada con Bootstrap.
- Pie con información legal.

Entregable

- Capturas de: código de los componentes y pantalla mostrando cabecera, listado y pie.
- (Opcional) archivo comprimido del proyecto.

Extensión opcional

- 1) Añadir botón “Añadir al carrito” en cada fila.
- 2) Crear componente Carrito y mostrar productos añadidos.
- 3) Usar Context API para gestionar estado global del carrito.