

REACT - SPRINGBOOT

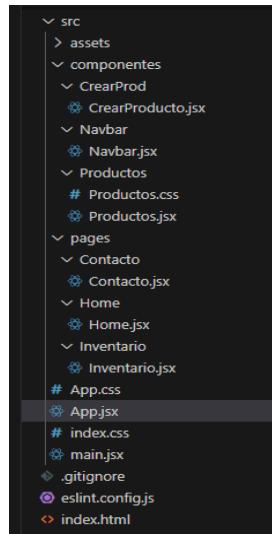
Crearemos proyecto nuevo en react:

- Proyecto: myprojectBackend
- React
- Javascript
- Verificar que la carpeta node_modules este creada.
- Iniciar servidor en su carpeta de proyecto: npm run dev

Estructura del proyecto:

- src/componentes
 - /CrearProd: componente, css.
 - /Navbar: componente, css.
 - /Productos: componente, css.
- src/pages:
 - /Contacto: componente, css.
 - /Home: componente, css.
 - /Inventario: componente.css

Instalar: `npm install react-router-dom`



Configurando Proyecto:

- En Spring boot, en los controladores rest, añadir: `@CrossOrigin`, evitar bloque de cors.

```

CategoríaRestController.java | J ProductoRestController.java | J CategoríaRestController.java
> main > java > com > vivitasol > projectbackend > controllers > J ProductoRestController.java
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.PutMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

import com.vivitasol.projectbackend.entities.Producto;
import com.vivitasol.projectbackend.services.ProductoServices;

@CrossOrigin(origins = "http://localhost:5173")
@RestController
@RequestMapping("/api/productos")

```

b. Componente App.jsx

```
App.jsx
src > App.jsx > App
1 import { BrowserRouter as Router, Routes, Route } from 'react-router-dom';
2 import { Home } from './pages/Home/Home';
3 import { Contacto } from './pages/Contacto/Contacto';
4 import { Inventario } from './pages/Inventario/inventario';
5 import { CrearProducto } from './componentes/CrearProd/CrearProducto';
6
7
8 import './App.css'
9
10 function App() {
11
12
13   return [
14     <Router>
15       <Routes>
16         <Route path="/" element={<Home />} />
17         <Route path="/contacto" element={<Contacto />} />
18         <Route path="/inventario" element={<Inventario />} />
19         <Route path="/crear-producto" element={<CrearProducto />} />
20       </Routes>
21     </Router>
22   ]
23 }
24
25 export default App
26
```

- Import {}
- Router
- Routes
- element

c. Componente Navbar:

- Import {Link}
- Link to="/"

d. Componente Home:

```
Home.jsx
src > pages > Home > Home.jsx > Home
1 import { Navbar } from "../../componentes/Navbar/Navbar";
2
3 export function Home(){
4
5   return(
6     <>
7       <div className="container">
8         <Navbar/>
9       </div>
10    </>
11  )
12}
13
```

```
Navbar.jsx
src > components > Navbar > Navbar.jsx > ...
1 import { Link } from 'react-router-dom';
2
3 export function Navbar(){
4
5   return (
6     <nav className="navbar navbar-expand-lg navbar-light bg-light">
7       <div className="container">
8         <Link className="navbar-brand" to="/">Mi Página</Link>
9         <button className="navbar-toggler" type="button" data-bs-toggle="collapse"
10            data-bs-target="#menuNav">
11           <span className="navbar-toggler-icon"></span>
12         </button>
13
14         <div className="collapse navbar-collapse" id="menuNav">
15           <ul className="navbar-nav ms-auto">
16             <li className="nav-item">
17               <Link className="nav-link active" to="/">Home</Link>
18             </li>
19             <li className="nav-item">
20               <Link className="nav-link" to="/inventario">Inventario</Link>
21             </li>
22             <li className="nav-item">
23               <Link className="nav-link" to="/contacto">Contacto</Link>
24             </li>
25           </ul>
26         </div>
27       </div>
28     );
29   }
30 }
```

e. Componente Inventario:

```
Inventario.jsx
src > pages > Inventario > Inventario.jsx > ...
1 import { Navbar } from "../../componentes/Navbar/Navbar";
2 import { Productos } from "../../componentes/Productos/Productos";
3
4
5 export function Inventario(){
6
7   return (
8     <div className="container">
9       <Navbar/>
10      <Productos/>
11     </div>
12   )
13 }
14
15 }
```

f. Componente Productos:

```
Productos.jsx ×
src > componentes > Productos > Productos.jsx > Productos > useEffect() callback
1 import { useEffect, useState } from 'react';
2 import { Link } from 'react-router-dom';
3 import './Productos.css';
4
5 export function Productos() {
6
7   const [productos, setProductos] = useState([]);
8
9   useEffect(() => {
10     fetch('http://localhost:8080/api/productos')
11       .then(response => {
12         if (!response.ok) {
13           throw new Error('Error en la respuesta del servidor');
14         }
15         return response.json();
16       })
17       .then(data => {
18         console.log("Respuesta del backend:", data);
19         setProductos(data);
20       })
21       .catch(error => {
22         console.error('Error al obtener las categorías:', error);
23       });
24   }, []);
25
26   return (
27     <div>
28       <h1>Inventario de productos</h1>
29       <table border="1">
30         <thead>
31           <tr>
32             <th>Id Producto</th>
33             <th>Nombre Producto</th>
34           </tr>
35         </thead>
36         <tbody>
37           <tr>
38             <td>1</td>
39             <td>Monitor</td>
40           </tr>
41         </tbody>
42       </table>
43     </div>
44   );
45 }
```

- useState
- useEffect
- fetch()
- spring-boot en ejecución.

```
Productos.jsx ×
src > componentes > Productos > Productos.jsx > Productos
5 export function Productos() {
6
7   return [
8     <>
9       <div className="container mi-tabla">
10         <h3 style={{ marginBottom: '20px' }}>Inventario de productos</h3>
11         <div className="row mb-3">
12           <div className="col-12 text-end">
13             <Link className="btn btn-primary" to="/crear-producto">
14               Crear Producto
15             </Link>
16           </div>
17         </div>
18
19         <div className="row">
20           <div className="col-md">
21             <table className="table table-hover">
22               <thead>
23                 <tr>
24                   <th>Id Producto</th>
25                   <th>Nombre Producto</th>
26                 </tr>
27               </thead>
28               <tbody>
29                 <tr>
30                   <td>1</td>
31                   <td>Monitor</td>
32                 </tr>
33               </tbody>
34             </table>
35           </div>
36         </div>
37       </div>
38     </>
39   ];
40 }
```

- Link to = "/crear-producto"

```
Productos.jsx ×
src > componentes > Productos > Productos.jsx > Productos
export function Productos() {
1
2   return [
3     <>
4       <div className="row">
5         <div className="col-md">
6           <table className="table table-hover">
7             <thead>
8               <tr>
9                 <th>Id Producto</th>
10                <th>Nombre Producto</th>
11                <th>Descripción</th>
12                <th>Precio Producto</th>
13                <th>Editar Producto</th>
14                <th>Eliminar Producto</th>
15              </tr>
16            </thead>
17            <tbody>
18              <tr key={prod.id}>
19                <td>{prod.id}</td>
20                <td>{prod.nombre}</td>
21                <td>{prod.descripcion}</td>
22                <td>{prod.precio}</td>
23                <td><Link className="nav-link active enlaceEdit" to="/>Editar/</Link></td>
24                <td><Link className="nav-link active enlaceDel" to="/>Eliminar/</Link></td>
25              </tr>
26            </tbody>
27          </table>
28        </div>
29      </div>
30    </>
31  ];
32 }
```

- {productos.map(prod => ())}
- Link to="/"

g. CrearProducto: (segunda clase de la semana)

```
src > componentes > CrearProd > CrearProducto.jsx > CrearProducto
  1 import React, { useState, useEffect } from 'react';
  2 import { useNavigate } from 'react-router-dom';
  3 import { Navbar } from "../../componentes/Navbar/Navbar";
  4
  5 const API_BASE_URL = 'http://localhost:8080/api';
  6
  7 export function CrearProducto() {
  8   const navigate = useNavigate();
  9
 10   const [producto, setProducto] = useState([
 11     nombre: '',
 12     descripcion: '',
 13     precio: '',
 14     categoria: ''
 15   ]);
 16
 17   const [categorias, setCategorias] = useState([]);
 18   const [loading, setLoading] = useState(false);
 19   const [error, setError] = useState(null);
 20   const [loadingCategorias, setLoadingCategorias] = useState(true);
 21
 22   // Cargar categorías
 23   useEffect(() => {
 24     fetch(`http://localhost:8080/api/categorias`)
 25       .then(response => response.json())
 26       .then(data => setCategorias(data))
 27       .catch(error => setError(error));
 28   }, []);
 29
 30   const handleInputChange = (e) => {
 31     const { name, value } = e.target;
 32     const newProducto = { ...producto };
 33     newProducto[name] = value;
 34     setProducto(newProducto);
 35   }
 36
 37   const handleFormSubmit = (e) => {
 38     e.preventDefault();
 39
 40     if (!newProducto.nombre || !newProducto.descripcion || !newProducto.precio || !newProducto.categoria) {
 41       setError('Todos los campos son obligatorios');
 42       return;
 43     }
 44
 45     const requestOptions = {
 46       method: 'POST',
 47       headers: { 'Content-Type': 'application/json' },
 48       body: JSON.stringify(newProducto)
 49     };
 50
 51     fetch(`${API_BASE_URL}/productos`, requestOptions)
 52       .then(response => response.json())
 53       .then(data => {
 54         navigate(`/detalle/${data.id}`);
 55       })
 56       .catch(error => setError(error));
 57   }
 58
 59   const handleCategoriaSelect = (e) => {
 60     const newCategoria = e.target.value;
 61     const newProducto = { ...producto };
 62     newProducto.categoria = newCategoria;
 63     setProducto(newProducto);
 64   }
 65
 66   const handlePrecioInput = (e) => {
 67     const newPrecio = e.target.value;
 68     const newProducto = { ...producto };
 69     newProducto.precio = newPrecio;
 70     setProducto(newProducto);
 71   }
 72
 73   const handleNombreInput = (e) => {
 74     const newName = e.target.value;
 75     const newProducto = { ...producto };
 76     newProducto.nombre = newName;
 77     setProducto(newProducto);
 78   }
 79
 80   const handleDescripcionInput = (e) => {
 81     const newDescripcion = e.target.value;
 82     const newProducto = { ...producto };
 83     newProducto.descripcion = newDescripcion;
 84     setProducto(newProducto);
 85   }
 86
 87   const handleLoadingCategorias = () => {
 88     setLoadingCategorias(false);
 89     setLoading(true);
 90   }
 91
 92   const handleLoadingProductos = () => {
 93     setLoading(false);
 94     setError(null);
 95   }
 96
 97   const handleResetForm = () => {
 98     setProducto({
 99       nombre: '',
100       descripcion: '',
101       precio: '',
102       categoria: ''
103     });
104   }
105
106   const handleLogout = () => {
107     localStorage.removeItem('token');
108     localStorage.removeItem('username');
109     navigate('/login');
110   }
111
112   const handleNuevoProducto = () => {
113     navigate('/nuevo');
114   }
115
116   const handleDetalle = (id) => {
117     navigate(`/detalle/${id}`);
118   }
119
120   const handleActualizar = (id) => {
121     navigate(`/actualizar/${id}`);
122   }
123
124   const handleBorrar = (id) => {
125     const requestOptions = {
126       method: 'DELETE',
127       headers: { 'Content-Type': 'application/json' }
128     };
129
130     fetch(`${API_BASE_URL}/productos/${id}`, requestOptions)
131       .then(response => response.json())
132       .then(data => {
133         setError(`El producto ${data.nombre} ha sido eliminado`);
134         handleLoadingProductos();
135       })
136       .catch(error => setError(error));
137   }
138
139   const handleNuevoCategoria = () => {
140     navigate('/nuevaCategoria');
141   }
142
143   const handleActualizarCategoria = (id) => {
144     navigate(`/actualizarCategoria/${id}`);
145   }
146
147   const handleBorrarCategoria = (id) => {
148     const requestOptions = {
149       method: 'DELETE',
150       headers: { 'Content-Type': 'application/json' }
151     };
152
153     fetch(`${API_BASE_URL}/categorias/${id}`, requestOptions)
154       .then(response => response.json())
155       .then(data => {
156         setError(`La categoría ${data.nombre} ha sido eliminada`);
157         handleLoadingCategorias();
158       })
159       .catch(error => setError(error));
160   }
161
162   const handleNuevoUsuario = () => {
163     navigate('/nuevoUsuario');
164   }
165
166   const handleActualizarUsuario = (id) => {
167     navigate(`/actualizarUsuario/${id}`);
168   }
169
170   const handleBorrarUsuario = (id) => {
171     const requestOptions = {
172       method: 'DELETE',
173       headers: { 'Content-Type': 'application/json' }
174     };
175
176     fetch(`${API_BASE_URL}/usuarios/${id}`, requestOptions)
177       .then(response => response.json())
178       .then(data => {
179         setError(`El usuario ${data.username} ha sido eliminado`);
180         handleLoadingProductos();
181       })
182       .catch(error => setError(error));
183   }
184
185   const handleNuevoRol = () => {
186     navigate('/nuevoRol');
187   }
188
189   const handleActualizarRol = (id) => {
190     navigate(`/actualizarRol/${id}`);
191   }
192
193   const handleBorrarRol = (id) => {
194     const requestOptions = {
195       method: 'DELETE',
196       headers: { 'Content-Type': 'application/json' }
197     };
198
199     fetch(`${API_BASE_URL}/roles/${id}`, requestOptions)
200       .then(response => response.json())
201       .then(data => {
202         setError(`El rol ${data.nombre} ha sido eliminado`);
203         handleLoadingProductos();
204       })
205       .catch(error => setError(error));
206   }
207
208   const handleNuevoPermiso = () => {
209     navigate('/nuevoPermiso');
210   }
211
212   const handleActualizarPermiso = (id) => {
213     navigate(`/actualizarPermiso/${id}`);
214   }
215
216   const handleBorrarPermiso = (id) => {
217     const requestOptions = {
218       method: 'DELETE',
219       headers: { 'Content-Type': 'application/json' }
220     };
221
222     fetch(`${API_BASE_URL}/permisos/${id}`, requestOptions)
223       .then(response => response.json())
224       .then(data => {
225         setError(`El permiso ${data.nombre} ha sido eliminado`);
226         handleLoadingProductos();
227       })
228       .catch(error => setError(error));
229   }
230
231   const handleNuevoRolPermiso = () => {
232     navigate('/nuevoRolPermiso');
233   }
234
235   const handleActualizarRolPermiso = (id) => {
236     navigate(`/actualizarRolPermiso/${id}`);
237   }
238
239   const handleBorrarRolPermiso = (id) => {
240     const requestOptions = {
241       method: 'DELETE',
242       headers: { 'Content-Type': 'application/json' }
243     };
244
245     fetch(`${API_BASE_URL}/rolPermiso/${id}`, requestOptions)
246       .then(response => response.json())
247       .then(data => {
248         setError(`El rol permiso ${data.nombre} ha sido eliminado`);
249         handleLoadingProductos();
250       })
251       .catch(error => setError(error));
252   }
253
254   const handleNuevoRolUsuario = () => {
255     navigate('/nuevoRolUsuario');
256   }
257
258   const handleActualizarRolUsuario = (id) => {
259     navigate(`/actualizarRolUsuario/${id}`);
260   }
261
262   const handleBorrarRolUsuario = (id) => {
263     const requestOptions = {
264       method: 'DELETE',
265       headers: { 'Content-Type': 'application/json' }
266     };
267
268     fetch(`${API_BASE_URL}/rolUsuario/${id}`, requestOptions)
269       .then(response => response.json())
270       .then(data => {
271         setError(`El rol usuario ${data.nombre} ha sido eliminado`);
272         handleLoadingProductos();
273       })
274       .catch(error => setError(error));
275   }
276
277   const handleNuevoUsuarioPermiso = () => {
278     navigate('/nuevoUsuarioPermiso');
279   }
280
281   const handleActualizarUsuarioPermiso = (id) => {
282     navigate(`/actualizarUsuarioPermiso/${id}`);
283   }
284
285   const handleBorrarUsuarioPermiso = (id) => {
286     const requestOptions = {
287       method: 'DELETE',
288       headers: { 'Content-Type': 'application/json' }
289     };
290
291     fetch(`${API_BASE_URL}/usuarioPermiso/${id}`, requestOptions)
292       .then(response => response.json())
293       .then(data => {
294         setError(`El usuario permiso ${data.nombre} ha sido eliminado`);
295         handleLoadingProductos();
296       })
297       .catch(error => setError(error));
298   }
299
300   const handleNuevoUsuarioRol = () => {
301     navigate('/nuevoUsuarioRol');
302   }
303
304   const handleActualizarUsuarioRol = (id) => {
305     navigate(`/actualizarUsuarioRol/${id}`);
306   }
307
308   const handleBorrarUsuarioRol = (id) => {
309     const requestOptions = {
310       method: 'DELETE',
311       headers: { 'Content-Type': 'application/json' }
312     };
313
314     fetch(`${API_BASE_URL}/usuarioRol/${id}`, requestOptions)
315       .then(response => response.json())
316       .then(data => {
317         setError(`El usuario rol ${data.nombre} ha sido eliminado`);
318         handleLoadingProductos();
319       })
320       .catch(error => setError(error));
321   }
322
323   const handleNuevoPermisoRol = () => {
324     navigate('/nuevoPermisoRol');
325   }
326
327   const handleActualizarPermisoRol = (id) => {
328     navigate(`/actualizarPermisoRol/${id}`);
329   }
330
331   const handleBorrarPermisoRol = (id) => {
332     const requestOptions = {
333       method: 'DELETE',
334       headers: { 'Content-Type': 'application/json' }
335     };
336
337     fetch(`${API_BASE_URL}/permisoRol/${id}`, requestOptions)
338       .then(response => response.json())
339       .then(data => {
340         setError(`El permiso rol ${data.nombre} ha sido eliminado`);
341         handleLoadingProductos();
342       })
343       .catch(error => setError(error));
344   }
345
346   const handleNuevoPermisoUsuario = () => {
347     navigate('/nuevoPermisoUsuario');
348   }
349
350   const handleActualizarPermisoUsuario = (id) => {
351     navigate(`/actualizarPermisoUsuario/${id}`);
352   }
353
354   const handleBorrarPermisoUsuario = (id) => {
355     const requestOptions = {
356       method: 'DELETE',
357       headers: { 'Content-Type': 'application/json' }
358     };
359
360     fetch(`${API_BASE_URL}/permisoUsuario/${id}`, requestOptions)
361       .then(response => response.json())
362       .then(data => {
363         setError(`El permiso usuario ${data.nombre} ha sido eliminado`);
364         handleLoadingProductos();
365       })
366       .catch(error => setError(error));
367   }
368
369   const handleNuevoPermisoRolUsuario = () => {
370     navigate('/nuevoPermisoRolUsuario');
371   }
372
373   const handleActualizarPermisoRolUsuario = (id) => {
374     navigate(`/actualizarPermisoRolUsuario/${id}`);
375   }
376
377   const handleBorrarPermisoRolUsuario = (id) => {
378     const requestOptions = {
379       method: 'DELETE',
380       headers: { 'Content-Type': 'application/json' }
381     };
382
383     fetch(`${API_BASE_URL}/permisoRolUsuario/${id}`, requestOptions)
384       .then(response => response.json())
385       .then(data => {
386         setError(`El permiso rol usuario ${data.nombre} ha sido eliminado`);
387         handleLoadingProductos();
388       })
389       .catch(error => setError(error));
390   }
391
392   const handleNuevoPermisoRolPermiso = () => {
393     navigate('/nuevoPermisoRolPermiso');
394   }
395
396   const handleActualizarPermisoRolPermiso = (id) => {
397     navigate(`/actualizarPermisoRolPermiso/${id}`);
398   }
399
400   const handleBorrarPermisoRolPermiso = (id) => {
401     const requestOptions = {
402       method: 'DELETE',
403       headers: { 'Content-Type': 'application/json' }
404     };
405
406     fetch(`${API_BASE_URL}/permisoRolPermiso/${id}`, requestOptions)
407       .then(response => response.json())
408       .then(data => {
409         setError(`El permiso rol permiso ${data.nombre} ha sido eliminado`);
410         handleLoadingProductos();
411       })
412       .catch(error => setError(error));
413   }
414
415   const handleNuevoPermisoRolUsuarioPermiso = () => {
416     navigate('/nuevoPermisoRolUsuarioPermiso');
417   }
418
419   const handleActualizarPermisoRolUsuarioPermiso = (id) => {
420     navigate(`/actualizarPermisoRolUsuarioPermiso/${id}`);
421   }
422
423   const handleBorrarPermisoRolUsuarioPermiso = (id) => {
424     const requestOptions = {
425       method: 'DELETE',
426       headers: { 'Content-Type': 'application/json' }
427     };
428
429     fetch(`${API_BASE_URL}/permisoRolUsuarioPermiso/${id}`, requestOptions)
430       .then(response => response.json())
431       .then(data => {
432         setError(`El permiso rol usuario permiso ${data.nombre} ha sido eliminado`);
433         handleLoadingProductos();
434       })
435       .catch(error => setError(error));
436   }
437
438   const handleNuevoPermisoRolUsuarioPermisoRol = () => {
439     navigate('/nuevoPermisoRolUsuarioPermisoRol');
440   }
441
442   const handleActualizarPermisoRolUsuarioPermisoRol = (id) => {
443     navigate(`/actualizarPermisoRolUsuarioPermisoRol/${id}`);
444   }
445
446   const handleBorrarPermisoRolUsuarioPermisoRol = (id) => {
447     const requestOptions = {
448       method: 'DELETE',
449       headers: { 'Content-Type': 'application/json' }
450     };
451
452     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRol/${id}`, requestOptions)
453       .then(response => response.json())
454       .then(data => {
455         setError(`El permiso rol usuario permiso rol ${data.nombre} ha sido eliminado`);
456         handleLoadingProductos();
457       })
458       .catch(error => setError(error));
459   }
460
461   const handleNuevoPermisoRolUsuarioPermisoRolPermiso = () => {
462     navigate('/nuevoPermisoRolUsuarioPermisoRolPermiso');
463   }
464
465   const handleActualizarPermisoRolUsuarioPermisoRolPermiso = (id) => {
466     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermiso/${id}`);
467   }
468
469   const handleBorrarPermisoRolUsuarioPermisoRolPermiso = (id) => {
470     const requestOptions = {
471       method: 'DELETE',
472       headers: { 'Content-Type': 'application/json' }
473     };
474
475     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermiso/${id}`, requestOptions)
476       .then(response => response.json())
477       .then(data => {
478         setError(`El permiso rol usuario permiso rol permiso ${data.nombre} ha sido eliminado`);
479         handleLoadingProductos();
480       })
481       .catch(error => setError(error));
482   }
483
484   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermiso = () => {
485     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermiso');
486   }
487
488   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermiso = (id) => {
489     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermiso/${id}`);
490   }
491
492   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermisoPermiso = (id) => {
493     const requestOptions = {
494       method: 'DELETE',
495       headers: { 'Content-Type': 'application/json' }
496     };
497
498     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermisoPermiso/${id}`, requestOptions)
499       .then(response => response.json())
500       .then(data => {
501         setError(`El permiso rol usuario permiso rol permiso permiso ${data.nombre} ha sido eliminado`);
502         handleLoadingProductos();
503       })
504       .catch(error => setError(error));
505   }
506
507   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermiso = () => {
508     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermiso');
509   }
510
511   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermiso = (id) => {
512     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermiso/${id}`);
513   }
514
515   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermiso = (id) => {
516     const requestOptions = {
517       method: 'DELETE',
518       headers: { 'Content-Type': 'application/json' }
519     };
520
521     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermiso/${id}`, requestOptions)
522       .then(response => response.json())
523       .then(data => {
524         setError(`El permiso rol usuario permiso rol permiso permiso permiso ${data.nombre} ha sido eliminado`);
525         handleLoadingProductos();
526       })
527       .catch(error => setError(error));
528   }
529
530   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermiso = () => {
531     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermiso');
532   }
533
534   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
535     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`);
536   }
537
538   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
539     const requestOptions = {
540       method: 'DELETE',
541       headers: { 'Content-Type': 'application/json' }
542     };
543
544     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`, requestOptions)
545       .then(response => response.json())
546       .then(data => {
547         setError(`El permiso rol usuario permiso rol permiso permiso permiso permiso ${data.nombre} ha sido eliminado`);
548         handleLoadingProductos();
549       })
550       .catch(error => setError(error));
551   }
552
553   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = () => {
554     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso');
555   }
556
557   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
558     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`);
559   }
560
561   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
562     const requestOptions = {
563       method: 'DELETE',
564       headers: { 'Content-Type': 'application/json' }
565     };
566
567     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`, requestOptions)
568       .then(response => response.json())
569       .then(data => {
570         setError(`El permiso rol usuario permiso rol permiso permiso permiso permiso ${data.nombre} ha sido eliminado`);
571         handleLoadingProductos();
572       })
573       .catch(error => setError(error));
574   }
575
576   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = () => {
577     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso');
578   }
579
580   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
581     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`);
582   }
583
584   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
585     const requestOptions = {
586       method: 'DELETE',
587       headers: { 'Content-Type': 'application/json' }
588     };
589
590     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`, requestOptions)
591       .then(response => response.json())
592       .then(data => {
593         setError(`El permiso rol usuario permiso rol permiso permiso permiso permiso ${data.nombre} ha sido eliminado`);
594         handleLoadingProductos();
595       })
596       .catch(error => setError(error));
597   }
598
599   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = () => {
600     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso');
601   }
602
603   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
604     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`);
605   }
606
607   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
608     const requestOptions = {
609       method: 'DELETE',
610       headers: { 'Content-Type': 'application/json' }
611     };
612
613     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`, requestOptions)
614       .then(response => response.json())
615       .then(data => {
616         setError(`El permiso rol usuario permiso rol permiso permiso permiso permiso ${data.nombre} ha sido eliminado`);
617         handleLoadingProductos();
618       })
619       .catch(error => setError(error));
620   }
621
622   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = () => {
623     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso');
624   }
625
626   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
627     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`);
628   }
629
630   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
631     const requestOptions = {
632       method: 'DELETE',
633       headers: { 'Content-Type': 'application/json' }
634     };
635
636     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`, requestOptions)
637       .then(response => response.json())
638       .then(data => {
639         setError(`El permiso rol usuario permiso rol permiso permiso permiso permiso ${data.nombre} ha sido eliminado`);
640         handleLoadingProductos();
641       })
642       .catch(error => setError(error));
643   }
644
645   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = () => {
646     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso');
647   }
648
649   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
650     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`);
651   }
652
653   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
654     const requestOptions = {
655       method: 'DELETE',
656       headers: { 'Content-Type': 'application/json' }
657     };
658
659     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`, requestOptions)
660       .then(response => response.json())
661       .then(data => {
662         setError(`El permiso rol usuario permiso rol permiso permiso permiso permiso ${data.nombre} ha sido eliminado`);
663         handleLoadingProductos();
664       })
665       .catch(error => setError(error));
666   }
667
668   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = () => {
669     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso');
670   }
671
672   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
673     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`);
674   }
675
676   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
677     const requestOptions = {
678       method: 'DELETE',
679       headers: { 'Content-Type': 'application/json' }
680     };
681
682     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`, requestOptions)
683       .then(response => response.json())
684       .then(data => {
685         setError(`El permiso rol usuario permiso rol permiso permiso permiso permiso ${data.nombre} ha sido eliminado`);
686         handleLoadingProductos();
687       })
688       .catch(error => setError(error));
689   }
690
691   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = () => {
692     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso');
693   }
694
695   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
696     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`);
697   }
698
699   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
700     const requestOptions = {
701       method: 'DELETE',
702       headers: { 'Content-Type': 'application/json' }
703     };
704
705     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`, requestOptions)
706       .then(response => response.json())
707       .then(data => {
708         setError(`El permiso rol usuario permiso rol permiso permiso permiso permiso ${data.nombre} ha sido eliminado`);
709         handleLoadingProductos();
710       })
711       .catch(error => setError(error));
712   }
713
714   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = () => {
715     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso');
716   }
717
718   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
719     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`);
720   }
721
722   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
723     const requestOptions = {
724       method: 'DELETE',
725       headers: { 'Content-Type': 'application/json' }
726     };
727
728     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`, requestOptions)
729       .then(response => response.json())
730       .then(data => {
731         setError(`El permiso rol usuario permiso rol permiso permiso permiso permiso ${data.nombre} ha sido eliminado`);
732         handleLoadingProductos();
733       })
734       .catch(error => setError(error));
735   }
736
737   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = () => {
738     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso');
739   }
740
741   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
742     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`);
743   }
744
745   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
746     const requestOptions = {
747       method: 'DELETE',
748       headers: { 'Content-Type': 'application/json' }
749     };
750
751     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`, requestOptions)
752       .then(response => response.json())
753       .then(data => {
754         setError(`El permiso rol usuario permiso rol permiso permiso permiso permiso ${data.nombre} ha sido eliminado`);
755         handleLoadingProductos();
756       })
757       .catch(error => setError(error));
758   }
759
760   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = () => {
761     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso');
762   }
763
764   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
765     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`);
766   }
767
768   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
769     const requestOptions = {
770       method: 'DELETE',
771       headers: { 'Content-Type': 'application/json' }
772     };
773
774     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`, requestOptions)
775       .then(response => response.json())
776       .then(data => {
777         setError(`El permiso rol usuario permiso rol permiso permiso permiso permiso ${data.nombre} ha sido eliminado`);
778         handleLoadingProductos();
779       })
780       .catch(error => setError(error));
781   }
782
783   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = () => {
784     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso');
785   }
786
787   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
788     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`);
789   }
790
791   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
792     const requestOptions = {
793       method: 'DELETE',
794       headers: { 'Content-Type': 'application/json' }
795     };
796
797     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`, requestOptions)
798       .then(response => response.json())
799       .then(data => {
800         setError(`El permiso rol usuario permiso rol permiso permiso permiso ${data.nombre} ha sido eliminado`);
801         handleLoadingProductos();
802       })
803       .catch(error => setError(error));
804   }
805
806   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = () => {
807     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso');
808   }
809
810   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
811     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermiso/${id}`);
812   }
813
814   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermiso = (id) => {
815     const requestOptions = {
816       method: 'DELETE',
817       headers: { 'Content-Type': 'application/json' }
818     };
819
820     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermiso/${id}`, requestOptions)
821       .then(response => response.json())
822       .then(data => {
823         setError(`El permiso rol usuario permiso rol permiso permiso ${data.nombre} ha sido eliminado`);
824         handleLoadingProductos();
825       })
826       .catch(error => setError(error));
827   }
828
829   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = () => {
830     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermiso');
831   }
832
833   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermiso = (id) => {
834     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermiso/${id}`);
835   }
836
837   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermiso = (id) => {
838     const requestOptions = {
839       method: 'DELETE',
840       headers: { 'Content-Type': 'application/json' }
841     };
842
843     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermisoPermisoPermiso/${id}`, requestOptions)
844       .then(response => response.json())
845       .then(data => {
846         setError(`El permiso rol usuario permiso rol permiso ${data.nombre} ha sido eliminado`);
847         handleLoadingProductos();
848       })
849       .catch(error => setError(error));
850   }
851
852   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermisoPermiso = () => {
853     navigate('/nuevoPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermiso');
854   }
855
856   const handleActualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermisoPermiso = (id) => {
857     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermisoPermisoPermisoPermiso/${id}`);
858   }
859
860   const handleBorrarPermisoRolUsuarioPermisoRolPermisoPermiso = (id) => {
861     const requestOptions = {
862       method: 'DELETE',
863       headers: { 'Content-Type': 'application/json' }
864     };
865
866     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRolPermisoPermiso/${id}`, requestOptions)
867       .then(response => response.json())
868       .then(data => {
869         setError(`El permiso rol usuario permiso rol ${data.nombre} ha sido eliminado`);
870         handleLoadingProductos();
871       })
872       .catch(error => setError(error));
873   }
874
875   const handleNuevoPermisoRolUsuarioPermisoRolPermisoPermiso = () => {
876     navigate('/nuevoPermisoRolUsuarioPermisoRolPermiso');
877   }
878
879   const handleActualizarPermisoRolUsuarioPermisoRolPermiso = (id) => {
880     navigate(`/actualizarPermisoRolUsuarioPermisoRolPermiso/${id}`);
881   }
882
883   const handleBorrarPermisoRolUsuarioPermisoRol = (id) => {
884     const requestOptions = {
885       method: 'DELETE',
886       headers: { 'Content-Type': 'application/json' }
887     };
888
889     fetch(`${API_BASE_URL}/permisoRolUsuarioPermisoRol/${id}`, requestOptions)
890       .then(response => response.json())
891       .then(data => {
892         setError(`El permiso rol usuario ${data.nombre} ha sido eliminado`);
893         handleLoadingProductos();
894       })
895       .catch(error => setError(error));
896   }
897
898   const handleNuevoPermisoRolUsuario = () => {
899     navigate('/nuevoPermisoRolUsuario');
900   }
901
902   const handleActualizarPermisoRolUsuario = (id) => {
903     navigate(`/actualizarPermisoRolUsuario/${id}`);
904   }
905
906   const handleBorrarPermisoRol = (id) => {
907     const requestOptions = {
908       method: 'DELETE',
909       headers: { 'Content-Type': 'application/json' }
910     };
911
912     fetch(`${API_BASE_URL}/permisoRol/${id}`, requestOptions)
913       .then(response => response.json())
914       .then(data => {
915         setError(`El permiso ${data.nombre} ha sido eliminado`);
916         handleLoadingProductos();
917       })
918       .catch(error => setError(error));
919   }
920
921   const handleNuevoPermisoUsuario = () => {
922     navigate('/nuevoPermisoUsuario');
923   }
924
925   const handleActualizarPermisoUsuario = (id) => {
926     navigate(`/actualizarPermisoUsuario/${id}`);
927   }
928
929   const handleBorrarPermisoUsuario = (id) => {
930     const requestOptions = {
931       method: 'DELETE',
932       headers: { 'Content-Type': 'application/json' }
933     };
934
935     fetch(`${API_BASE_URL}/permisoUsuario/${id}`, requestOptions)
936       .then(response => response.json())
937       .then(data => {
938         setError(`El permiso ${data.nombre} ha sido eliminado`);
939         handleLoadingProductos();
940       })
941       .catch(error => setError(error));
942   }
943
944   const handleNuevoRolUsuario = () => {
945     navigate('/nuevoRolUsuario');
946   }
947
948   const handleActualizarRolUsuario = (id) => {
949     navigate(`/actualizarRolUsuario/${id}`);
950   }
951
952   const handleBorrarRolUsuario = (id) => {
953     const requestOptions = {
954       method: 'DELETE',
955       headers: { 'Content-Type': 'application/json' }
956     };
957
958     fetch(`${API_BASE_URL}/rolUsuario/${id}`, requestOptions)
959       .then(response => response.json())
960       .then(data => {
961         setError(`El rol ${data.nombre} ha sido eliminado`);
962         handleLoadingProductos();
963       })
964       .catch(error => setError(error));
965   }
966
967   const handleNuevoRol = () => {
968     navigate('/nuevoRol');
969   }
970
971   const handleActualizarRol = (id) => {
972     navigate(`/actualizarRol/${id}`);
973   }
974
975   const handleBorrarRol = (id) => {
976     const requestOptions = {
977       method: 'DELETE',
978       headers: { 'Content-Type': '
```

```
// Cargar categorías
useEffect(() => {
  const fetchCategorias = async () => {
    try {
      const response = await fetch(`${API_BASE_URL}/categorias`);
      if (!response.ok) throw new Error('Error al cargar categorías');
      const data = await response.json();
      setCategorias(data);
    } catch (err) {
      console.error('Error al cargar categorías:', err);
      setError('No se pudieron cargar las categorías');
    } finally {
      setLoadingCategorias(false);
    }
  };
  fetchCategorias();
}, [ ]);
```

```
    fetchCategorias();
}, []);  
  
// Manejar cambios en los campos
const handleChange = (e) => {
  const { name, value } = e.target;  
  
  setProducto(prev => ({
    ...prev,
    [name]: name === 'precio' ? (value === '' ? 0 : value) : value
  }));
};  
  
// Enviar formulario
const handleSubmit = (e) => {
```

Cargar en index.html:

Biblioteca de Bootstrap para icons