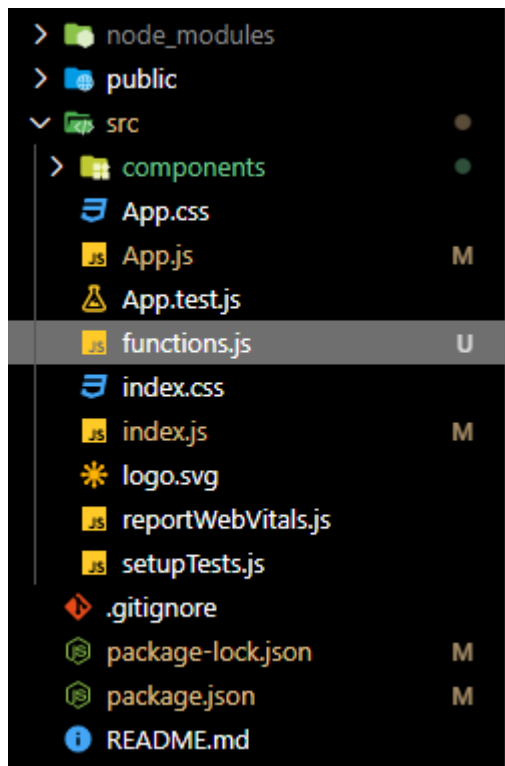


```
npm install react-router-dom axios bootstrap sweetalert2 sweetalert2-react-content
@fortawesome/fontawesome-free
```

index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
import 'bootstrap/dist/css/bootstrap.min.css';
import 'bootstrap/dist/js/bootstrap.bundle';
import '@fortawesome/fontawesome-free/css/all.min.css';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);
```



functions.js

```
import Swal from 'sweetalert2';
import withReactContent from 'sweetalert2-react-content';

export function show_alerta(mensaje, icono, foco='') {
  onFocus(foco);
  const MySwal = withReactContent(Swal);
  MySwal.fire({
    title: mensaje,
    icon: icono
  });
}

function onFocus(foco) {
  if(foco !== '') {
    document.getElementById(foco).focus();
  }
}
```

ShowProducts.js

```
import React, {useEffect, useState} from 'react';
import axios from 'axios';
import Swal from 'sweetalert2';
import withReactContent from 'sweetalert2-react-content';
import { show_alerta } from '../functions';

const ShowProducts = () => {
  const url = 'http://localhost:8000';
  const [products, setProducts] = useState([]);
  const [id, setId] = useState('');
  const [name, setName] = useState('');
  const [description, setDescription] = useState('');
  const [price, setPrice] = useState('');
  const [operation, setOperation] = useState(1);
  const [title, setTitle] = useState('');

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

  const getProducts = async () => {
    const respuesta = await axios.get(url);
```

```

    setProducts(respuesta.data);
  }
const openModal = (op, id, name, description, price) => {
  setId('');
  setName('');
  setDescription('');
  setPrice('');
  setOperation(op);
  if(op === 1){
    setTitle('Registrar Producto');
  }
  else if(op === 2){
    setTitle('Editar Producto');
    setId(id);
    setName(name);
    setDescription(description);
    setPrice(price);
  }
  window.setTimeout(function(){
    document.getElementById('nombre').focus();
  }, 5000);
}

const validar = () => {
  var parametros;
  var metodo;
  if(name.trim() === ''){
    show_alerta('Escribe el nombre del producto', 'warning')
  }else if(description.trim() === ''){
    show_alerta('Escribe la descripción del producto', 'warning')
  }else if(price.trim() === ''){
    show_alerta('Escribe el precio del producto', 'warning')
  }else{
    if(operation === 1){
      parametros =
{name:name.trim(),description:description.trim(),price:price}
      metodo='POST';
    }
    else{
      parametros =
{id:id,name:name.trim(),description:description.trim(),price:price}
      metodo='PUT';
    }
  }
}

```

```

        enviarSolicitud(metodo, parametros);
    }
}

const enviarSolicitud = async(metodo, parametros) => {
    await axios({ method:metodo, url:url,
data:parametros}).then(function(respuesta) {
        var tipo = respuesta.data[0];
        var msj = respuesta.data[1];
        show_alerta(msj,tipo);
        if(tipo === 'success'){
            document.getElementById('btnCerrar').click();
            getProducts();
        }
    })
    .catch(function(error) {
        show_alerta('Error en la solicitud', 'error');
        console.log(error)
    });
}

const deleteProduct= (id, name,) =>{
    const MySwal = withReactContent(Swal);
    MySwal.fire({
        title:'¿ Seguro de eliminar el producto '+name+' ?',
        icon: 'question', text:'Esta acción es irreversible',
        showCancelButton:true, confirmButtonText:'Si, eliminar',
cancelButtonText:'Cancelar'
    }).then((result) => {
        if(result.isConfirmed) {
            setId(id);
            enviarSolicitud('DELETE', {id:id});
        }
        else{
            show_alerta('El producto NO fue eliminado', 'info');
        }
    })
}

return (
    <div className='App'>

```

```

    <div className='container-fluid'>
      <div className='row mt-3'>
        <div className='col-md-4 offset-md-4'>
          <div className='d-grid mx-auto'>
            <button onClick={() => openModal(1)} className ='btn
btn-dark' data-bs-toggle='modal' data-bs-target='#modalProducts'>
              <i className='fa-solid fa-circle-plus'></i> Añadir
            </button>

          </div>
        </div>
      </div>

    </div>

    <div className='row mt-3'>
      <div className ='col-12 col-lg-8 offset-0 offset-lg-2'>
        <div className='table-responsive'>
          <table className='table table-bordered'>
            <thead>

          <tr><th>#</th><th>Producto</th><th>Descripcion</th><th>Precio</th></tr>
          </thead>
          <tbody className='table-group-divider'>
            {products.map((product, i) => (
              <tr key={product.id}>
                <td>{ (i+1) }</td>
                <td>{product.name}</td>
                <td>{product.description}</td>

                <td>${new
Intl.NumberFormat('es-mx').format(product.price)}</td>
                <td>
                  <button onClick={() =>
openModal(2,product.id,product.name,product.description,product.price)}
className='btn btn-warning' data-bs-toggle='modal'
data-bs-target='#modalProducts'>
                    <i className='fa-solid fa-edit'></i>
                  </button>
                  <button onClick={() =>deleteProduct(product.id,
product.name)} className='btn btn-danger m-2'>
                    <i className='fa-solid fa-trash'></i>
                  </button>
                </td>
              </tr>
            )}
          </tbody>
        </table>
      </div>
    </div>
  </div>

```

```

    ))
  }
</tbody>
</table>
</div>
</div>

</div>
</div>
<div id='modalProducts' className='modal fade'
aria-hidden='true'>
  <div className='modal-dialog'>
    <div className='modal-content'>
      <div className='modal-header'>
        <label className='h5'>{title}</label>
        <button type='button' className='btn-close'
data-bs-dismiss='modal' aria-label='Close'></button>
      </div>
      <div className='modal-body'>
        <input type='hidden' id='id'></input>
        <div className='input-group mb-3'>
          <span className='input-group-text'><i
className='fa-solid fa-gift'></i></span>
          <input type='text' id='nombre' className='form-control'
placeholder='Nombre' value={name}
onChange={(e) => setName(e.target.value)}></input>
        </div>

        <div className='input-group mb-3'>
          <span className='input-group-text'><i
className='fa-solid fa-dollar-sign'></i></span>
          <input type='text' id='precio' className='form-control'
placeholder='Precio' value={price}
onChange={(e) => setPrice(e.target.value)}></input>
        </div>

        <div className='input-group mb-3'>
          <span className='input-group-text'><i
className='fa-solid fa-comment'></i></span>
          <input type='text' id='descripcion'
className='form-control' placeholder='Descripcion' value={description}
onChange={(e) => setDescription(e.target.value)}></input>
        </div>
        <div className='d-grid col-6 mx-auto'>

```

```

        <button onClick={ () => validar() } className='btn
btn-success'>
            <i className='fa-solid fa-floppy-disk'></i> Guardar
        </button>
    </div>
</div>
    <div className='modal-footer'>
<button type='button' id='btnCerrar' className='btn btn-secondary'
data-bs-dismiss='modal'>Cerrar</button>
    </div>
</div>
</div>
</div>
)
}

export default ShowProducts

```