

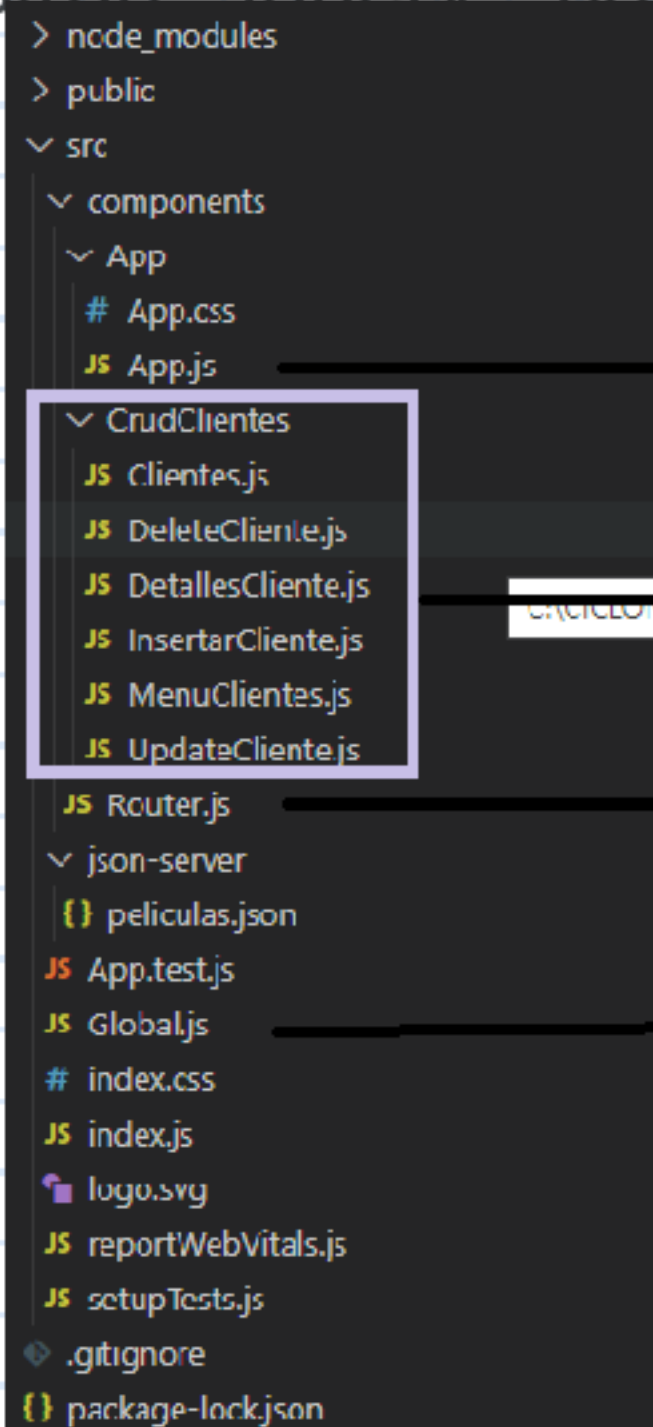
# Consumir RESTAPI con React - Axios

## 1. Crear la estructura del proyecto

`npx create-react-app nombre`

## 2. Agregar las dependencias

`axios - bootstrap - jquery - popper.js - react-router-dom`



Ejecuta el inicio del Proyecto

Componentes necesarios para el CRUD

Navegación del Proyecto

Guarda la URL del API Rest

### 3. Ajuste de App.js

Se movió a una carpeta dentro de components llamada App

Se agrega la importación de bootstrap - Router

```
src > components > App > JS App.js > ...
1  import './App.css';
2  import "bootstrap/dist/css/bootstrap.min.css";
3  import "bootstrap/dist/js/bootstrap.bundle.min";
4  import Router from '../Router';
5
6
7  function App() {
8    return (
9      <div className="App">
10        <Router />
11      </div>
12    );
13  }
14
15  export default App;
16
```

## Componente Router:

Se importan las dependencias o componentes a utilizar  
react-router-dom: Versión 5.2.0

```
src > components > JS Router.js > Router > render > <function>
  1  import React, { Component } from 'react';
  2  import { BrowserRouter, Route, Switch } from 'react-router-dom';
  3  import MenuClientes from './CrudClientes/MenuClientes';
  4  import Clientes from './CrudClientes/Clientes';
  5  import DetallesCliente from './CrudClientes/DetallesCliente';
  6  import UpdateCliente from './CrudClientes/UpdateCliente';
  7  import DeleteCliente from './CrudClientes/DeleteCliente';
  8  import InsertarCliente from './CrudClientes/InsertarCliente';
  9
```

```

10 export default class Router extends Component {
11   render() {
12     return (
13       <div>
14         <BrowserRouter>
15           <MenuClientes />      Estático
16           <Switch>
17             <Route exact path="/" component={Clientes} />
18             <Route exact path="/create" component={InsertarCliente} />
19             <Route exact path="/detalles/:_id" render={props => {
20               var id = props.match.params._id;
21               return <DetallesCliente _id={id} />
22             }} />
23             <Route exact path="/update/:_id" render={props => {
24               var id = props.match.params._id;
25               return <UpdateCliente _id={id} />
26             }} />
27             <Route exact path="/delete/:_id" render={props => {
28               var id = props.match.params._id;
29               return <DeleteCliente _id={id} />
30             }} />
31           </Switch>
32         </BrowserRouter>
33       </div>
34     )
35   }
36 }

```

Dinámica  
 path=url  
 component={nombre}

```
1  import React, { Component } from 'react';
2  import { NavLink } from 'react-router-dom';
3
4  export default class MenuClientes extends Component {
5      render() {
6          return (
7              <nav className="navbar navbar-expand-lg navbar-dark bg-primary">
8                  <a className="navbar-brand" href="/">Navbar</a>
9                  <button className="navbar-toggler" type="button" data-toggle="collapse"
10                     data-target="#navbarNavDropdown" aria-controls="navbarNavDropdown"
11                     aria-expanded="false" aria-label="Toggle navigation">
12                      <span className="navbar-toggler-icon"></span>
13                  </button>
14                  <div className="collapse navbar-collapse" id="navbarNavDropdown">
15                      <ul className="navbar-nav">
16                          <li className="nav-item active">
17                              <NavLink className="nav-link" to="/">Clientes <span className="sr-only">
18                                  (current)</span></NavLink>
19                          </li>
20                          <li className="nav-item">
21                              <NavLink className="nav-link" to="/create">Nuevo Cliente</NavLink>
22                          </li>
23                      </ul>
24                  </div>
25              </nav>
26          )
27      }
28  }
```

```
src > JS Global.js > [🔗] default
1   var Global = {
2     urlclientes: "http://localhost:8082/api/clientes"
3   }
4
5   export default Global;
```

Se definen las URLs que se van a usar en el proyecto

## Componente Clientes

### Mostrar la lista de Clientes

#### 1. Importar las librerías y componentes a usar

```
src > components > CrudClientes > JS Clientes.js > 🗑️ Clientes > 📦 render
1   import React, { Component } from 'react';
2   import axios from 'axios';
3   import Global from '../Global';
4   import { NavLink } from 'react-router-dom';
5
6   export default class Clientes extends Component {
7
```



```
state = {  
  clientes: []  
  , status: false  
}
```

Inicializar Variables  
Arreglo clientes  
Estado: status

```
cargarClientes = () => {  
  var url = Global.urlclientes;  
  var request = "/clientes";  
  axios.get(url + request).then(res => {  
    this.setState({  
      clientes: res.data  
      , status: true  
    });  
  });  
}
```

Método que genera la lista de  
clientes con axios y get  
Enlazar backend con front

```
componentDidMount = () => {  
  this.cargarClientes();  
}
```

Ejecutamos el método  
cargarClientes

```
render() {  
  return (  
    <div>  
      <h1>Listado de Clientes</h1>  
      <table className="table table-striped">  
        <thead className="thead-dark">  
          <tr>  
            <th>Id Cliente</th>  
            <td> Cédula</td>  
            <td> Nombre</td>  
            <td> Dirección</td>  
            <td> Correo</td>  
            <td> Teléfono</td>  
            <td> Ver</td>  
            <td> Editar</td>  
            <td> Borrar</td>  
          </tr>  
        </thead>  
      </table>  
    </div>  
  );  
}
```



```
    <tbody>
      {this.state.status === true &&
      (
        this.state.clientes.map(cliente => {
          return(
            <tr key={cliente._id}>
              <td> {cliente._id}</td>
              <td> {cliente.cedula}</td>
              <td> {cliente.nombre}</td>
              <td> {cliente.direccion}</td>
              <td> {cliente.email}</td>
              <td> {cliente.telefono}</td>
              <td>
                <NavLink to={"/detalles/" + cliente._id}>Detalles</NavLink>
              </td>
              <td>
                <NavLink to={"/update/" + cliente._id}>Modificar</NavLink>
              </td>
              <td>
                <NavLink to={"/delete/" + cliente._id}>Eliminar</NavLink>
              </td>
            </tr>
          );
        })
      )}
    </tbody>
  </table>
</div>
)
```

## Componente InsertarClientes

### 1. Importar librerías y componentes a usar

```
src > components > CrudClientes > JS InsertarCliente.js > InsertarCliente
1  import React, { Component } from 'react';
2  import axios from 'axios';
3  import Global from '../../Global';
4  import { Redirect } from 'react-router-dom';
5
6  export default class InsertarCliente extends Component {
7
```

### 2. Definimos las variables a usar en el componente (lectura) - state

```
8      cajaCedRef = React.createRef();
9      cajaNomRef = React.createRef();
10     cajaDirRef = React.createRef();
11     cajaEmaRef = React.createRef();
12     cajaTelRef = React.createRef();
13
14     state = { status: false }
```

### 3. Definir el método que permita crear el nuevo cliente

Recibe el objeto con los datos y almacena en una variable cada dato

Define un objeto tipo cliente con los datos escritos en el formulario

Llamar al método que permita crear el cliente enviando la url y el objeto cliente

```
nuevoCliente = (e) => {  
  e.preventDefault();  
  var ced = this.cajaCedRef.current.value;  
  var nom = this.cajaNomRef.current.value;  
  var dir = this.cajaDirRef.current.value;  
  var ema = this.cajaEmaRef.current.value;  
  var tel = this.cajaTelRef.current.value;  
  var cliente = {  
    cedula: ced,  
    nombre: nom,  
    direccion: dir,  
    email: ema,  
    telefono: tel  
  };  
  var url = Global.urlclientes + '/clientes';  
  axios.post(url, cliente).then(res => {  
    this.setState({ status: true });  
  });  
}
```

Según el valor de status muestra la lista o el formulario para capturar los datos del nuevo cliente

```
render() {
  if(this.state.status === true){
    return <Redirect to="/" />
  }
  return (
    <div>
      <h1>Nuevo Cliente</h1>
      <form onSubmit={this.nuevoCliente} style={{width: "50%", margin: "auto"}}>
        <label>Cédula: </label>
        <input type="text" name="cajaced" className="form-control" ref={this.cajaCedRef} />
        <label>Nombre: </label>
        <input type="text" name="cajanom" className="form-control" ref={this.cajaNomRef} />
        <label>Dirección: </label>
        <input type="text" name="cajadir" className="form-control" ref={this.cajaDirRef} />
        <label>Correo: </label>
        <input type="text" name="cajaema" className="form-control" ref={this.cajaEmaRef} />
        <label>Teléfono: </label>
        <input type="text" name="cajatel" className="form-control" ref={this.cajaTelRef} />
        <br />
        <button className="btn btn-success">Añadir</button>
      </form>
    </div>
  )
}
```

## Componente DetallesCliente

### 1. Importar librerías y componentes

-- creamos la clase tipo componente

-- Valores iniciales cliente { } y status: false

```
src > components > CrudClientes > JS DetallesCliente.js > DetallesCliente > render
1  import React, { Component } from 'react';
2  import axios from 'axios';
3  import Global from '../../Global';
4  import { NavLink } from 'react-router-dom';
5
6  export default class DetallesCliente extends Component {
7
8      state = {
9          cliente: {}
10         , status: false
11     }
12
```

### 2. Construir el método que muestre los datos del cliente con el \_id

```
mostrarCliente = () => {
    var request = "/clientes/" + this.props._id;
    var url = Global.urlclientes + request;
    axios.get(url).then(res => {
        this.setState({
            cliente: res.data
            ,status: true
        });
    });
}

componentDidMount = () => {
    this.mostrarCliente();
}
```

3. Dibujar en pantalla los datos del usuario, permitiendo navegar para poder ejecutar las acciones de editar, eliminar o listar

```
render() {
  return (
    <div>
      <br /><br />
      <h1><u>Detalles del Cliente {this.props._id}</u></h1>
      {this.state.status === true &&
      (
        <React.Fragment>
          <br />
          <NavLink to="/" className="btn btn-sm btn-dark">Listado</NavLink>
          <br /><br />
          <h3>Nombre: <span style={{color: "green", fontWeight: "bold"}}>
            {this.state.cliente.nombre}</span></h3>
          <h3>Cédula: <span style={{color: "green", fontWeight: "bold"}}>
            {this.state.cliente.cedula}</span></h3>
          <h3>Correo: <span style={{color: "green", fontWeight: "bold"}}>
            {this.state.cliente.email}</span></h3>
          <NavLink to={"/update/" + this.state.cliente._id} className="btn btn-primary">
            Modificar</NavLink> &nbsp;&nbsp;&nbsp;
          <NavLink to={"/delete/" + this.state.cliente._id} className="btn btn-danger">
            Borrar</NavLink>
        </React.Fragment>
      )
    }
    </div>
  )
}
```



## Componente UpdateCliente

1. Importar librerías y componentes
2. Crear el componente inicializar las variables a usar en el formulario e inicializamos los valores de cliente y de status

```
src > components > CrudClientes > JS UpdateCliente.js > UpdateCliente > mo
1  import React, { Component } from 'react';
2  import axios from 'axios';
3  import Global from '../../Global';
4  import { Redirect, NavLink } from 'react-router-dom';
5
6  export default class UpdateCliente extends Component {
7
8      cajaCedRef = React.createRef();
9      cajaNomRef = React.createRef();
10     cajaDirRef = React.createRef();
11     cajaEmaRef = React.createRef();
12     cajaTelRef = React.createRef();
13
14     state = {
15         cliente: {},
16         status: false }
```



3. Crear el método que busque el cliente y guarde el valor en la variable cliente  
Se ejecuta este método gracias al componentDidMount

```
mostrarCliente = () => {  
  var request = "/clientes/" + this.props._id;  
  var url = Global.urlclientes + request;  
  axios.get(url).then(res => {  
    this.setState({  
      cliente: res.data  
      ,status: false  
    });  
  });  
}  
  
componentDidMount = () => {  
  this.mostrarCliente();  
}
```

4, Con el render definimos con las variables de entrada que mostrar y que hacer

```
render() {
  if(this.state.status === true){
    return <Redirect to="/" />
  }
  return (
    <div>
      <h1>Modificar Cliente {this.props._id}</h1>
      <h2>{this.state.cliente.nombre}</h2>
      <NavLink to={'/detalles/' + this.props._id} className="btn btn-sm btn-dark">
        Detalles</NavLink>&nbsp;
      <NavLink to={'/'} className="btn btn-sm btn-dark">Lista</NavLink>
      <form onSubmit={this.modificarCliente} style={{width: "50%", margin: "auto"}}>
        <label>Cédula: </label>
        <input type="number" name="cajaced" className="form-control" ref={this.cajaCedRef}
          defaultValue={this.state.cliente.cedula}/>
        <label>Nombre: </label>
        <input type="text" name="cajanom" className="form-control" ref={this.cajaNomRef}
          defaultValue={this.state.cliente.nombre}/>
        <label>Dirección: </label>
        <input type="text" name="cajadir" className="form-control" ref={this.cajaDirRef}
          defaultValue={this.state.cliente.direccion}/>
        <label>Correo: </label>
        <input type="text" name="cajaema" className="form-control" ref={this.cajaEmaRef}
          defaultValue={this.state.cliente.email}/><br />
        <label>Teléfono: </label>
        <input type="text" name="cajatel" className="form-control" ref={this.cajaTelRef}
          defaultValue={this.state.cliente.telefono} /><br />
        <button className="btn btn-success">Modificar</button>
      </form>
    </div>
  )
}
```

Guardar en una variable los datos y escritos en cada cajón, creamos un objeto cliente y después se llama al put con axios para actualizar el cliente

```
modificarCliente = (e) => {  
  e.preventDefault();  
  var ced = this.cajaCedRef.current.value;  
  var nom = this.cajaNomRef.current.value;  
  var dir = this.cajaDirRef.current.value;  
  var ema = this.cajaEmaRef.current.value;  
  var tel = this.cajaTelRef.current.value;  
  var cliente = {  
    cedula: ced,  
    nombre: nom,  
    direccion: dir,  
    email: ema,  
    telefono: tel  
  };  
  var request = "/clientes/" + this.props._id;  
  var url = Global.urlclientes + request;  
  axios.put(url, cliente).then(res => {  
    this.setState({status: true});  
  });  
}
```

## Componente DeleteCliente

```
src > components > CrudClientes > JS DeleteCliente.js > DeleteCliente > render
```

```
1 import React, { Component } from 'react';  
2 import axios from 'axios';  
3 import Global from '../../Global';  
4 import { Redirect, NavLink } from 'react-router-dom';  
5  
6 export default class DeleteCliente extends Component {  
7  
8     state = { status: false };  
9  
10    eliminarCliente = () => {  
11        var request = "/clientes/" + this.props._id;  
12        var url = Global.urlclientes + request;  
13        axios.delete(url).then(res => {  
14            |   this.setState({ status: true });  
15            |   });  
16        }  
17  
18    render() {  
19        if(this.state.status === true){  
20            return <Redirect to="/" />  
21        }  
22        return (  
23            <div>  
24                <br />  
25                <h1 style={{color: "red"}}><Desea eliminar el cliente {this.props._id}></h1><br />  
26                <NavLink to="/" className="btn btn-light">Cancelar</NavLink>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~</NavLi~>  
27                <button onClick={this.eliminarCliente} className="btn btn-danger">Eliminar</button>  
28            </div>  
29        )  
30    }  
31 }
```