**Ejercicios sesiones 7, 8 y 9**

En este ejercicio de React JS deberéis crear una lista, esta lista tendrá dentro distintos contactos y deberá cumplir con las siguientes funcionalidades:

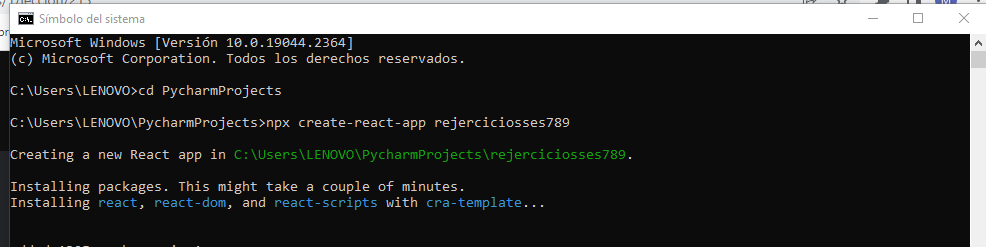
Mostrar contacto.

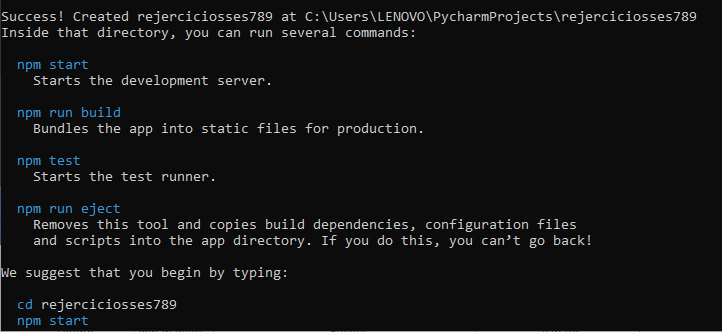
Crear contacto.

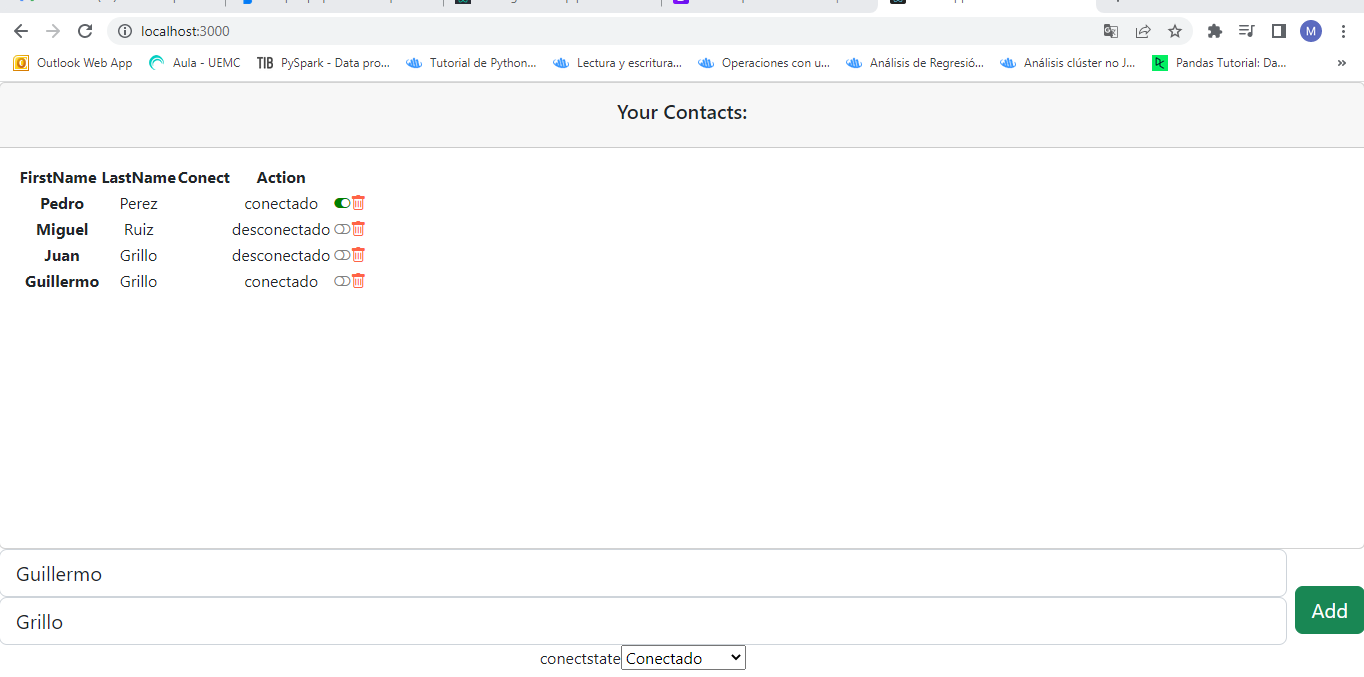
Eliminar contacto.

Cambiar el estado del contacto entre Conectado y Desconectado. Se ve el cambio de estado en la consola. NO pude renderizarlo

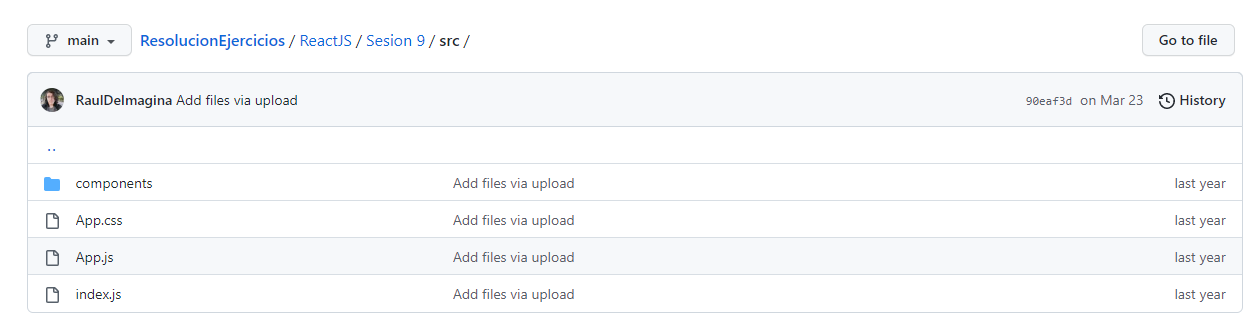
1. Generamos el proyecto en nuestra carpeta PycharmProjects

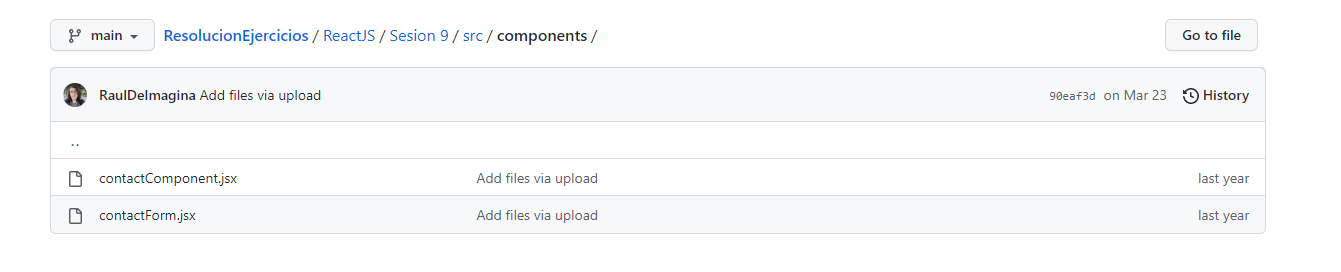






Solucion





**contactComponent.jsx**

|  |
| --- |
| import React from 'react'; |
|  |  |
|  | const ContactComponent = ({ contacto, changeState, remove }) => { |
|  | return ( |
|  | <div className="contact-component"> |
|  | <h2>{contacto.nombre}</h2> |
|  | <span>{contacto.email}</span> |
|  |  |
|  | <button |
|  | style={{ backgroundColor: contacto.estado ? 'green' : 'red' }} |
|  | className="state-button" |
|  | onClick={() => { |
|  | changeState(contacto); |
|  | }} |
|  | > |
|  | {contacto.estado ? 'Conectado' : 'Desconectado'} |
|  | </button> |
|  |  |
|  | <button className="remove-button" onClick={() => remove(contacto)}> |
|  | Borrar |
|  | </button> |
|  | </div> |
|  | ); |
|  | }; |
|  |  |
|  | export default ContactComponent;  **contactForm.jsx**   |  | | --- | | import React, { useRef } from 'react'; | |  |  | |  | const ContactForm = ({ onAddContact }) => { | |  | const name = useRef(''); | |  | const email = useRef(''); | |  |  | |  | function addContact(e) { | |  | e.preventDefault(); | |  |  | |  | const newContact = { nombre: name.current.value, email: email.current.value, conectado: true }; | |  |  | |  | onAddContact(newContact); | |  | name.current.value = ''; | |  | email.current.value = ''; | |  | } | |  |  | |  | return ( | |  | <form onSubmit={addContact} className="contact-component"> | |  | <h2>Añadir Contacto: </h2> | |  | <input ref={name} name="name" placeholder="Nombre contacto" className="form-control mb-2" /> | |  | <input ref={email} name="email" type="email" placeholder="Email contacto" /> | |  | <button onClick={addContact} type="submit" className="submit-button"> | |  | Añadir | |  | </button> | |  | </form> | |  | ); | |  | }; | |  |  | |  | export default ContactForm; |   **App.css**   |  | | --- | | \* { | |  | margin: 0; | |  | padding: 0; | |  | } | |  | .App { | |  | margin: 0; | |  | padding: 0; | |  | width: 100%; | |  | height: 100vh; | |  | display: flex; | |  | flex-direction: column; | |  | align-items: center; | |  | justify-content: center; | |  | background-color: darkgrey; | |  | row-gap: 10px; | |  | } | |  | .contact-component { | |  | background-color: white; | |  | margin: 5px; | |  | border-radius: 10px; | |  | padding: 10px; | |  | display: flex; | |  | flex-direction: column; | |  | row-gap: 10px; | |  | align-items: center; | |  | min-width: 200px; | |  | } | |  |  | |  | .card-container { | |  | display: flex; | |  | flex-wrap: wrap; | |  | } | |  |  | |  | .remove-button { | |  | width: 100%; | |  | border: none; | |  | background-color: tomato; | |  | padding: 5px; | |  | border-radius: 3px; | |  | color: white; | |  | font-size: 14px; | |  | cursor: pointer; | |  | } | |  |  | |  | .remove-button:hover { | |  | opacity: 0.8; | |  | } | |  |  | |  | .state-button { | |  | width: 100%; | |  | border: none; | |  | padding: 5px; | |  | border-radius: 3px; | |  | color: white; | |  | font-size: 14px; | |  | cursor: pointer; | |  | } | |  |  | |  | .state-button:hover { | |  | opacity: 0.8; | |  | } | |  |  | |  | input { | |  | background-color: darkgrey; | |  | border: none; | |  | outline: none; | |  | color: white; | |  | padding: 8px; | |  | font-size: 14px; | |  | border-radius: 5px; | |  | } | |  |  | |  | input::placeholder { | |  | color: white; | |  | } | |  |  | |  | .submit-button { | |  | width: 100%; | |  | border: none; | |  | padding: 5px; | |  | border-radius: 3px; | |  | color: white; | |  | font-size: 14px; | |  | cursor: pointer; | |  | background-color: black; | |  | } | |  |  | |  | .submit-button:hover { | |  | opacity: 0.8; | |  | } |   **App.js**   |  | | --- | | import React, { useState, useEffect } from 'react'; | |  | import ContactComponent from './components/contactComponent'; | |  | import ContactForm from './components/contactForm'; | |  | import './App.css'; | |  | const App = () => { | |  | const defaultContact = [ | |  | { nombre: 'Jose', email: 'Fernandez', conectado: true }, | |  | { nombre: 'Maria', email: 'Gómez', conectado: true }, | |  | ]; | |  |  | |  | const [nuevoContacto, setNuevoContacto] = useState(defaultContact); | |  |  | |  | function changeState(contacto) { | |  | const index = nuevoContacto.indexOf(contacto); | |  | const tempContact = [...nuevoContacto]; | |  |  | |  | tempContact[index].estado = !tempContact[index].estado; | |  | setNuevoContacto(tempContact); | |  | } | |  |  | |  | function remove(contacto) { | |  | const index = nuevoContacto.indexOf(contacto); | |  | const tempContact = [...nuevoContacto]; | |  | tempContact.splice(index, 1); | |  | setNuevoContacto(tempContact); | |  | } | |  |  | |  | function addContact(contacto) { | |  | const tempContact = [...nuevoContacto]; | |  | tempContact.push(contacto); | |  | setNuevoContacto(tempContact); | |  | } | |  |  | |  | return ( | |  | <div className="App"> | |  | <h1>Contactos</h1> | |  |  | |  | <div className="card-container"> | |  | {nuevoContacto.map((contacto, index) => { | |  | return ( | |  | <ContactComponent key={index} contacto={contacto} changeState={changeState} remove={remove} /> | |  | ); | |  | })} | |  | </div> | |  |  | |  | <ContactForm onAddContact={addContact}></ContactForm> | |  | </div> | |  | ); | |  | }; | |  |  | |  | export default App; |   **Index.js**   |  | | --- | | import React from 'react'; | |  | import ReactDOM from 'react-dom'; | |  | import App from './App'; | |  |  | |  | ReactDOM.render( | |  | <React.StrictMode> | |  | <App /> | |  | </React.StrictMode>, | |  | document.getElementById('root') | |  | ); | |