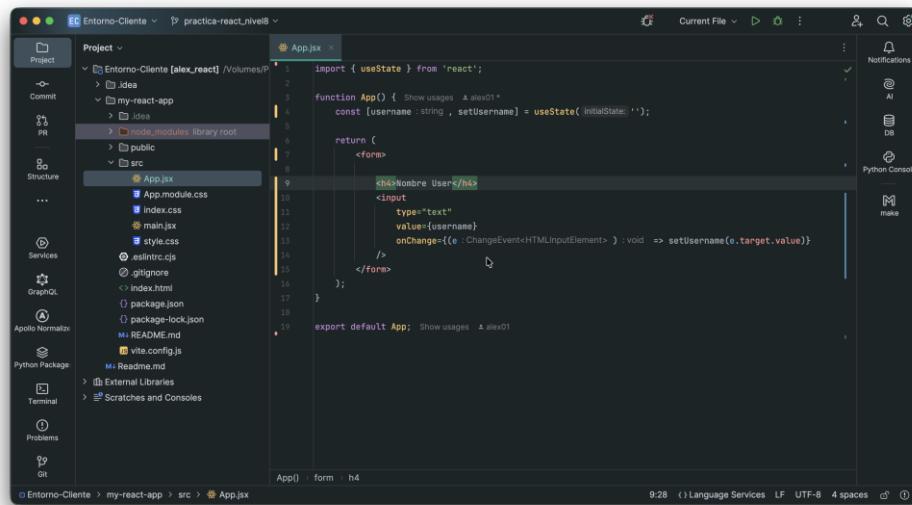


## 1- Parte A: Input controlado (Controlled Component)

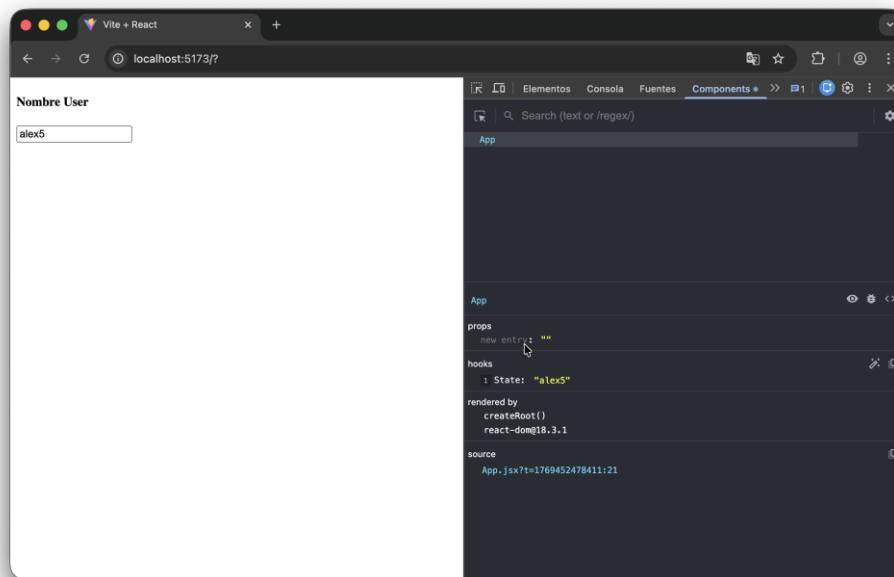


```
import { useState } from 'react';

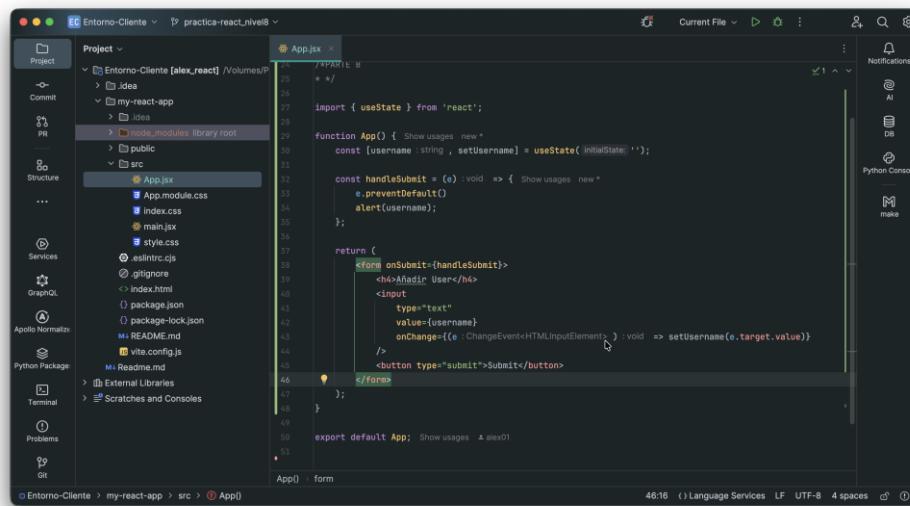
function App() {
  const [username, setUsername] = useState('alex01');

  return (
    <form>
      <h4>Nombre User</h4>
      <input
        type="text"
        value={username}
        onChange={(e: ChangeEvent<HTMLInputElement>) => setUsername(e.target.value)}
      />
    </form>
  );
}

export default App;
```



## 2- Parte B: Envío del formulario (onSubmit)



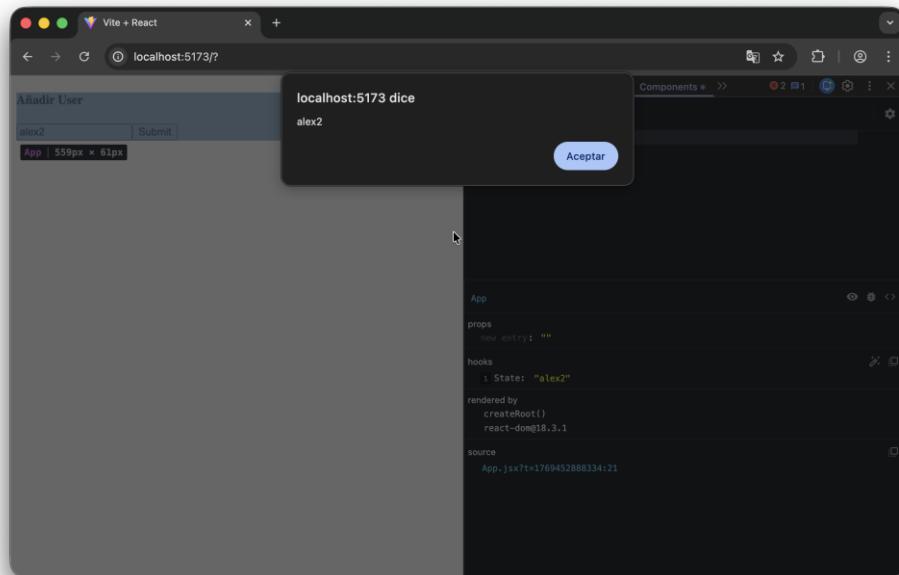
```
import { useState } from 'react';

function App() {
  const [username, setUsername] = useState(''); // Initial state

  const handleSubmit = (e) => { // Show usages new
    e.preventDefault();
    alert(username);
  };

  return (
    <form onSubmit={handleSubmit}>
      <h4>Añadir User</h4>
      <input
        type="text"
        value={username}
        onChange={(e: ChangeEvent<HTMLInputElement>) => setUsername(e.target.value)}
      />
      <button type="submit">Submit</button>
    </form>
  );
}

export default App; // Show usages a alex0
```



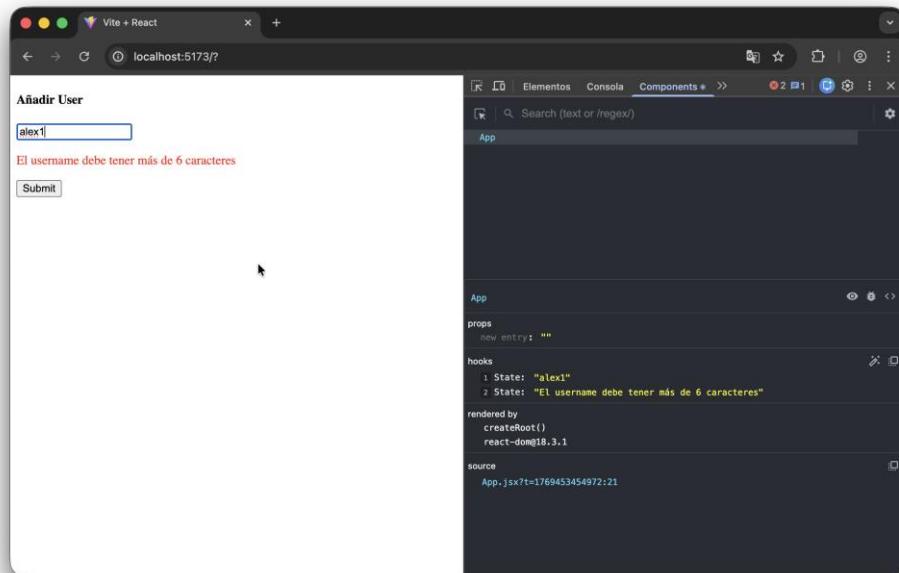
### 3- Parte C: Validación básica

```

function App() {
  const [usernameError, setUsernameError] = useState(initialState);
  const handleUsername = (e) => {
    const { value } = e.target;
    setUsername(value);
    if (value.length <= 6) {
      setUsernameError('El username debe tener más de 6 caracteres');
    } else {
      setUsernameError('');
    }
  };
  const handleSubmit = (e) => {
    e.preventDefault();
    if (usernameError) {
      alert('Error en el form');
    } else {
      alert(`Todo correcto: ${username}`);
    }
  };
  return (
    <form onSubmit={handleSubmit}>
      <h4>Añadir User</h4>
      <input type="text" value={username} onChange={handleUsername}>
      </input>
      <p style={{color: 'red'}}>{usernameError}</p>
      <button type="submit">Submit</button>
    </form>
  );
}

export default App;

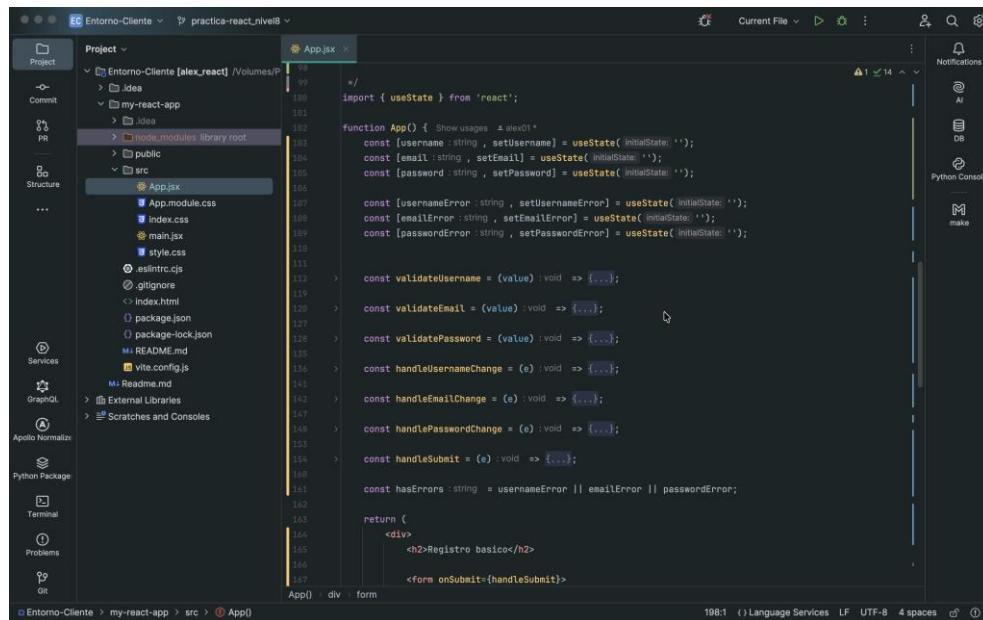
```



#### 4- Mini Reto

- Con el método del ejercicio anterior, he replicado lo mismo, pero con las validaciones necesarias y he añadido las variables de errores para cada error

- También he replicado el método que me muestra los errores mientras el usuario va rellenando los campos
- Para luego ser utilizados de forma independiente y validar que todo el formulario este correcto, en el caso de que no el botón de registrar se desactiva y no permite enviarlo



```

import { useState } from 'react';

function App() {
  const [username, setUsername] = useState(initialState);
  const [email, setEmail] = useState(initialState);
  const [password, setPassword] = useState(initialState);

  const [usernameError, setUsernameError] = useState('');
  const [emailError, setEmailError] = useState('');
  const [passwordError, setPasswordError] = useState('');

  const validateUsername = (value) => {
    // Validation logic for username
  };

  const validateEmail = (value) => {
    // Validation logic for email
  };

  const validatePassword = (value) => {
    // Validation logic for password
  };

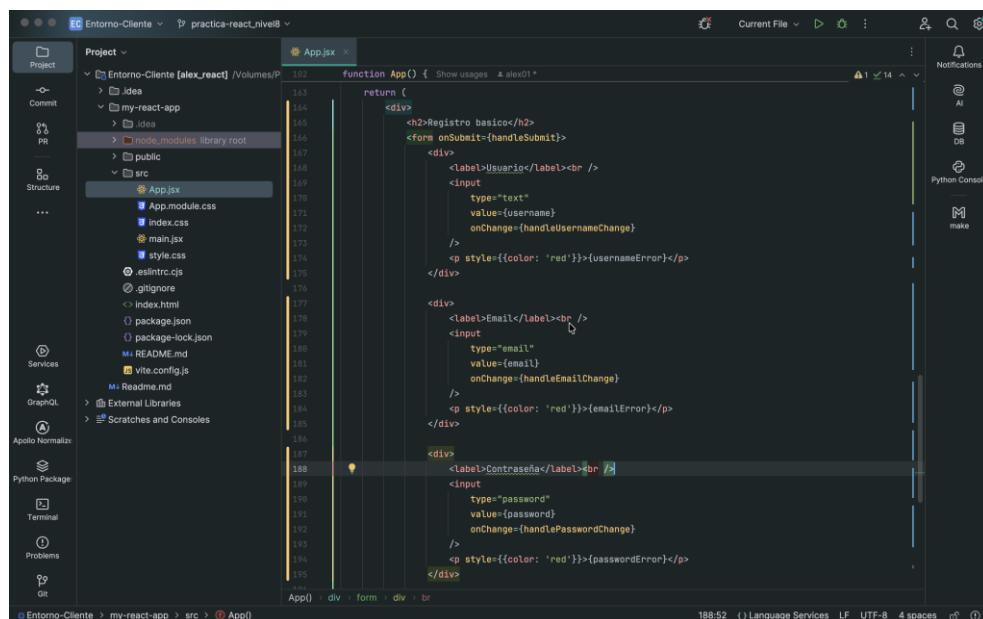
  const handleUsernameChange = (e) => {
    setUsername(e.target.value);
  };

  const handleEmailChange = (e) => {
    setEmail(e.target.value);
  };

  const handlePasswordChange = (e) => {
    setPassword(e.target.value);
  };

  const handleSubmit = (e) => {
    e.preventDefault();
    if (!validateUsername(username) || !validateEmail(email) || !validatePassword(password)) {
      return;
    }
    const hasErrors = usernameError || emailError || passwordError;
    if (hasErrors) {
      return;
    }
    const formData = {
      username,
      email,
      password
    };
    // Submit logic
  };
}

export default App;
  
```



```

function App() {
  return (
    <div>
      <h2>Registro basico</h2>
      <form onSubmit={handleSubmit}>
        <div>
          <label>Usuario</label><br />
          <input type="text" value={username} onChange={handleUsernameChange}>
        </div>
        <div>
          <label>Email</label><br />
          <input type="email" value={email} onChange={handleEmailChange}>
        </div>
        <div>
          <label>Contraseña</label><br />
          <input type="password" value={password} onChange={handlePasswordChange}>
        </div>
      </form>
    </div>
  );
}

const handleUsernameChange = (e) => {
  setUsername(e.target.value);
};

const handleEmailChange = (e) => {
  setEmail(e.target.value);
};

const handlePasswordChange = (e) => {
  setPassword(e.target.value);
};

const handleSubmit = (e) => {
  e.preventDefault();
  if (!validateUsername(username) || !validateEmail(email) || !validatePassword(password)) {
    return;
  }
  const hasErrors = usernameError || emailError || passwordError;
  if (hasErrors) {
    return;
  }
  const formData = {
    username,
    email,
    password
  };
  // Submit logic
};
  
```

## - Resultado

