

//formData: valores del formulario setFormData: actualiza el estado e formData useState: inicializa es estado con los siguientes campos (valor inicial vacío)

// handleChange: Esta función maneja el evento onChange en cada campo de entrada. Actualiza el estado (formData) con el valor actual del campo que se ha modificado. Esta función se asegura de que cada vez que se modifique un campo en el formulario, el estado se actualice para reflejar ese cambio.

//e: Es el evento que se dispara cuando un campo del formulario cambia.

//name y value: Se extraen del evento (e.target) el nombre del campo (name) y su valor actual (value).

//setFormData({...}): Actualiza el estado del formulario. Se utiliza la e.preventDefault(): Previene el comportamiento predeterminado del formulario de recargar la página cuando se envía. sintaxis de "spread operator" (...formData) para copiar el estado actual de formData, y luego se actualiza solo el campo que ha cambiado ([name]: value).

// Input: Cada campo del formulario (input) está vinculado a su correspondiente valor en el estado mediante la propiedad value, y se actualiza con el onChange para reflejar cualquier cambio en tiempo real.

//handleSubmit: Esta función se ejecuta cuando se envía el formulario. Se evita el comportamiento predeterminado del formulario (que recarga la página) con e.preventDefault() y se maneja el envío de los datos

// e.preventDefault(): Previene el comportamiento predeterminado del formulario de recargar la página cuando se envía.

// console.log(formData): Muestra el contenido de formData en la consola cuando se envía el formulario. Si tuviéramos los datos de un servidor se podría enviar

// Renderización del formulario = return

: El formulario tiene un manejador onSubmit que llama a handleSubmit cuando se envía.

: Cada campo de entrada (input) está conectado al estado del formulario (formData). La propiedad value se liga al valor correspondiente en formData, y onChange se liga a handleChange para actualizar el estado cuando el usuario escribe en los campos. name: Es importante porque handleChange usa el valor de name para identificar qué campo está cambiando y actualizar el estado correspondiente. Botón de Enviar (  ): Permite enviar el formulario. Al hacer clic en él, se dispara la función handleSubmit.

// Exportación del componente:

```
export default NewUserForm;
```

Permite que el componente NewUserForm se pueda importar y utilizar en otros archivos dentro de la aplicación React