

FORMULARIOS

- Usualmente cuando utilizamos formularios en HTML , la acción por defecto es tomar estos valores y enviarlos gracias al atributo action de los formularios.
- En estos casos, no tenemos la posibilidad por defecto de manipular los datos ingresados por el usuario.
- En el caso de querer manipularlos, deberemos generar funciones que nos permitan gracias al DOM acceder a estos valores y utilizarlos para nuestras necesidades, incluso guardarlos en alguna variable.
- En React nosotros tenemos la posibilidad tanto de no tener control o manejo de estos valores ingresado ,como la posibilidad de controlar los valores que el usuario ingresa en tiempo real .

Componentes Controlados

- En HTML, los elementos de formularios como los <input>, <textarea> y el <select> normalmente mantienen sus propios estados, valores y los actualizan de acuerdo a la interacción del usuario. En React, el estado mutable es mantenido normalmente en la propiedad estado de los componentes, y solo se actualiza con **setState()**.
- Podemos combinar ambos conceptos haciendo que el estado de React sea la “única fuente de la verdad”, utilizando el atributo **value** de los campos del formulario y dándoles como valor, estos estados.
- Para los casos de checkbox o radio, disponemos también del atributo **checked**, donde podemos manipularlo con los booleanos para marcar o desmarcarlo.
- En un componente controlado, los datos del formulario son manejados por un componente React.
- Un campo de un formulario cuyos valores son controlados por React de esta forma es denominado “componente controlado”.

Componentes No Controlados

- En lugar de escribir un controlador de eventos para cada actualización de estado, podemos utilizar otros métodos para referirnos a dichos campos o utilizar lo ya conocido por JS para poder manipularlo.
- Estos tipos de componentes nos permiten utilizar menos código ,lo cual hace una solución rápida a los formularios, pero no se da la desventaja de no poder controlar estos valores.
- De igual manera, en caso que querramos establecer algún valor predeterminado, disponemos de los atributos **defaultValue** (para elementos como input, textarea o select) o **defaultChecked** (para elementos como checkbox o radio),que nos permite agregarlo como prop a la etiqueta.
- A modo de información adicional, disponemos de algunos componentes de formulario, como puede ser el input de type file, que no puede ser un elemento controlado, ya que depende completamente del usuario, solo puede ser establecido por este. Por ello en ese tipo de casos podemos también hacer uso de una herramienta de React llamada referencia (ref) para lo que dejamos el siguiente link para profundizar el tema:
- <https://es.reactjs.org/docs/refs-and-the-dom.html>