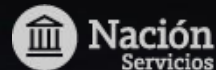


# NODE JS

## Clase 50



## ... Manejo controlado

Como vemos, en el manejo controlado, se usa un estado para guardar la información del formulario y se le asigna ese valor al input. Por tanto, actualizarlo actualiza tanto lo que se ve como la información guardada.

## ... Manejo controlado

React muestra errores en consola si un componente controlado tiene un valor *null* o *undefined*. Si bien el error no afecta el funcionamiento, la forma recomendada es usar un string vacío "" para un input vacío.

## ... Inputs y tipos de datos

Los elementos HTML tienen un atributo `type`, por el cual el navegador aplica cierto control sobre lo que puede introducirse. Algunos ejemplos:

- `text`
- `number` (admite solo números, incluyendo decimales).
- `date` (admite fechas, con el formato `yyyy-mm-dd`, pero lo muestra según el lenguaje y país del navegador, por ejemplo, `dd-mm-yyyy` en Argentina).
- `email`

## ... Problemas con el tipo de datos

Los cambios en un input por defecto se guardan como string. Dependiendo de cómo esté configurado el Back End, esto puede lanzar error en caso de que esté esperando otro tipo de datos. Esto puede ajustarse convirtiendo los datos en el Front End, o ajustando el Back End para que espere la información del formulario de otra manera.

## ... Problemas con el tipo de datos

En algunos casos el manejo controlado de React puede ser causa de error. Por ejemplo, si el campo es numérico, enviar un string vacío puede generar error. Para evitar esto, habrá que convertirlo a *null*, sea en el Front o en el Back.

## ... Etiqueta form

La etiqueta form de HTML está configurada por defecto para recargar la página completa. Esto puede evitarse en React tomando el evento que lo provoca y usar el método `preventDefault`.

## ... Etiqueta form

Otra alternativa es no usar la etiqueta form, y usar divs u otras etiquetas para armar el formulario.

```
const handleSubmit = (e) => {  
  e.preventDefault();  
  postData();  
};
```



## ... Validaciones avanzadas y librerías

Pueden introducirse validaciones más avanzadas en el input de un formulario, por ejemplo, aceptar valores con un largo máximo o que cumplan con una cierta expresión regular.

## ... Validaciones avanzadas y librerías

El objetivo de esto es no permitir que se envíe desde el Front End información inadecuada, que pueda generar un error en el servidor o la base de datos, así como controlar qué tipo de información aceptar.

Si la información es inadecuada, se le puede mostrar un mensaje o Toast al usuario para que la adecúe.

## ... Librerías

Algunos desarrolladores más avanzados en React prefieren usar librerías para manejar formularios, que tienen ciertas funciones de validación o manejo de errores ya incorporadas, en lugar de usar `useState` para manejar la información del formulario.

No obstante, es importante manejar las bases de React antes de utilizar estas soluciones más avanzadas.

# ¡Muchas gracias!

