

Colocando el Gasto a Editar en el Modal

- Para ello necesito pasarle por props el objeto gastoEditar a Modal, y hacer una comprobación
- De esta forma va a saber que gasto tiene que editar, y que información actual tiene ese gasto
- Cuando hay tantos props la solución es context o redux
- Usaré un useEffect, porque este Modal se monta y se desmonta todo el tiempo
- Lo importo en Modal
- useEffect siempre se va a ejecutar una vez cuando tenga las dependencias vacías
- Si lo coloco en Modal, cuando le doy clic a nuevoGasto aparece en consola

```
useEffect(()=>{  
  console.log('componente listo')  
}, [])
```

- Ahora coloco la validación, que es la misma que anteriormente
- Si gastoEditar viene vacío es un nuevo gasto, si no estamos editando
- Si esta editando, si hay algo, hay que llenar los campos con los hooks
- Usaré la sintaxis de punto para que no choque

```
const Modal = ({setModal, animarModal, setAnimarModal, guardarGasto, gastoEditar}) => {  
  
  const [nombre, setNombre] = useState('')  
  const [cantidad, setCantidad] =useState('')  
  const [categoria, setCategoria] = useState('')  
  const [mensaje, setMensaje] = useState('')  
  
  useEffect(()=>{  
    if(Object.keys(gastoEditar).length > 0){  
      setNombre(gastoEditar.nombre)  
      setCantidad(gastoEditar.cantidad)  
      setCategoria(gastoEditar.categoria)  
    }  
  }, [])  
}
```

- Como el select tiene un value, va a seleccionar directamente esa opción
- Ahora si le doy a editar en el gasto, aparece el Modal relleno
- Pero si le doy al icono de gasto nuevo, VUELVE A APARECER EL GASTO EDITADO en el formulario.
- Como handleNuevoGasto es la función que abre cuando clicas el icono, se le puede añadir el setGastoEditar con un objeto vacío

```
const handleNuevoGasto = () =>{
  setModal(true)
  setGastoEditar({})
  setTimeout(()=>{
    setAnimarModal(true)
  },500)
}
```

Pero ahora no aparece nada cuando quiero editar, porque estoy llamando al handleNuevoGasto.

- Entonces, en lugar de llamarlo, puedo colocar la lógica en la verificación del useEffect del App.jsx y quito el setGastoEditar

```
const handleNuevoGasto = () =>{
  setModal(true)
  setGastoEditar({})

  setTimeout(()=>{
    setAnimarModal(true)
  },500)
}

useEffect(()=>{
  if(Object.keys(gastoEditar).length){
    setModal(true)

    setTimeout(()=>{
      setAnimarModal(true)
    },500)
  }
}, [gastoEditar])
```

- De esta forma va limpiando el estado, según ciertas acciones.

De eso trata React, ir reaccionado a ciertos eventos e ir cambiando al state

- Ahora si rellena los campos y si le doy a clic aparece vacío, pero ahora al editar me genera un nuevo objeto en el listado

- Además, me gustaría que en lugar de Nuevo Gasto, apareciera modificar gasto, que se renderice de forma condicional.
- En la próxima lección