

Primera entrega parcial

- Al menos dos eventos de ratón empleando propiedades relacionadas con coordenadas y tamaño de elementos.
- Al menos una de las siguientes herramientas:
 - Uso de delegación de eventos o handleEvent.
 - Evento personalizado con CustomEvent.
- Al menos una vez se manejan acciones asociadas a fase de burbuja o propagación (detener o capturar eventos comunes a varias etiquetas anidadas).
- Dos validaciones de teclado:
 - Caracteres alfanuméricos.
 - Teclas especiales (CTRL, SHIFT, ENTER, etc.).

Entrega la última versión del código, además de un documento en el que indiques lo siguiente:

Los archivos y líneas de código donde se implementa cada uno de los apartados requeridos en la entrega parcial.

Todos los cambios están en el script de la sala 2

Eventos de ratón:

```
21 //unidad 5 - evento de raton usando elemento
22 function tamaño(evento) {
23     const ancho = evento.target.offsetWidth;
24     const alto = evento.target.offsetHeight;
25
26     console.log(`Este libro mide ${ancho}px de ancho y ${alto}px de alto`);
27 }
28

67 // Unidad 5- eventp de raton con coordenadas
68 const bibliotecaCoordenada = document.getElementById("biblioteca");
69
70 bibliotecaCoordenada.addEventListener("mousemove", (evento) => {
71     const x = evento.clientX;
72     const y = evento.clientY;
73
74     console.log(`Estas en la coordenda X:${x}, Y:${y}`)
75
76
77 })
78
79
```

Uso de HandleEvent y stopPropagation:

```
59  //Tema 5 handleEvent + control burbuja
60  const manejadorEvento = {
61    handleEvent(evento) {
62      evento.stopPropagation();
63      const idLibro = evento.target.id;
64      revisar(idLibro);
65    }
66  };
67
85  libro1.addEventListener("click", manejadorEvento);
86  libro2.addEventListener("click", manejadorEvento);
87  libro3.addEventListener("click", manejadorEvento);
88  libro4.addEventListener("click", manejadorEvento);
```

Uso de fase de burbuja:

```
80  //Unidad 5 - burbuja
81  bibliotecaCoordenada.addEventListener("click", () => {
82    console.log("click detectado");
83  });
84
```

Validaciones de teclado:

AlfaNumerico:

```
<div>
  <label for=""> Detector</label>
  <input type="text" id="input" placeholder="Escribe aqui">
</div>
90 //Unidad 5 -validacion alfanumerica
91
92 const codigoInput = document.getElementById("input");
93
94 codigoInput.addEventListener("keypress", (evento) => {
95   const caracter = evento.key;
96
97   //Letras y numeros
98   const esAlfanumerico = /^[a-zA-Z0-9]$/.test(caracter);
99
100 if (!esAlfanumerico) {
101   evento.preventDefault();
102   console.log("caracter no permitido: ", caracter)
103 }
104
105});
```

Detector de teclas especiales:

```
107 //Unidad 5- detección de teclas especiales
108
109 document.addEventListener("keydown", (evento) => {
110     if (evento.ctrlKey) {
111         console.log("Tecla CTRL pulsada");
112     }
113     if (evento.shiftKey) {
114         console.log("Tecla SHIFT pulsada");
115     }
116     if (evento.key === "Enter") {
117         console.log("Tecla ENTER pulsada");
118     }
119 });
120
```

Página funcionando:

The screenshot shows a game interface titled "Sala 2 — Biblioteca". On the left, there's a "Instrucciones" box stating: "Estás en una biblioteca, la puerta por la que pasaste se cierra detrás de ti, hay otra puerta al otro lado de la habitación, a lo mejor ahí está la salida. Hay un letrero arriba que dice, "Encuentra los 3 libros que conectan con la puerta." Hay varios libros, pero 4 te llaman la atención, revisalos a ver si te pueden ayudar." Below it is an "Estantería" section with four buttons: "Libro de economía", "Libro de mecánica", "Libro de juegos", and "Libro de fantasía". A "Intentar abrir puerta" button is also present. On the right, the browser's DevTools console is open, showing the following log entries:

Message	File
Este libro mide 129px de ancho y 35px de alto	sala2.js:27
Estas en la coordenada X:1942, Y:328	sala2.js:151:25
Estas en la coordenada X:1985, Y:331	sala2.js:151:25
Estas en la coordenada X:996, Y:347	sala2.js:151:25
Estas en la coordenada X:987, Y:359	sala2.js:151:25
Tecla CTRL pulsada	sala2.js:111
@ carácter no permitido:	sala2.js:110:2
@ carácter no permitido:	sala2.js:110:2
@ Tecla CTRL pulsada	sala2.js:111
@ Tecla SHIFT pulsada	sala2.js:110:2
carácter no permitido: %	sala2.js:110:2
Tecla SHIFT pulsada	sala2.js:111
carácter no permitido: %	sala2.js:110:2
Estas en la coordenada X:365, Y:359	sala2.js:151:25
Estas en la coordenada X:366, Y:354	sala2.js:151:25
Este libro mide 140px de ancho y 35px de alto	sala2.js:27
Estas en la coordenada X:366, Y:350	sala2.js:151:25
Estas en la coordenada X:367, Y:345	sala2.js:151:25
Estas en la coordenada X:370, Y:348	sala2.js:151:25
Estas en la coordenada X:370, Y:351	sala2.js:151:25
Estas en la coordenada X:372, Y:358	sala2.js:151:25

Explica al menos los cambios de dos commits de GitHub:

Añadido eventos de teclado:

En este commit se añaden eventos de teclado que pide la entrega. Se añaden tanto la comprobación alfanumérica y de teclas especiales en este caso de ctrl, shift y enter. También se añadió un input en el html para poder probarlo, si intentas escribir solo deja poner letras tanto minúsculas o mayúsculas de la a-z y numero del 0-9 en caso de escribir algo fuera de esos parámetros, además de no aparecer en consola te indicara que ese carácter no es válido y te mostrara cual usaste. Lo mismo si usas algún carácter especial de los 3 seleccionados aparecerá un mensaje pero no se escribirán.

The screenshot shows a GitHub commit interface. At the top, the commit message is "Añadido eventos de teclado". Below it are two code snippets: "index.html" and "validar.js".

index.html:

```
<html>
<head>
    <title>Validación de Entrada</title>
    <script type="text/javascript" src="validar.js"></script>
</head>
<body>
    <table border="1">
        <tr>
            <td>Nombre:</td>
            <td><input type="text" id="input" placeholder="Tu nombre aquí"></td>
        </tr>
    </table>
</body>
</html>
```

validar.js:

```
function validarNombre(nombre) {
    let resultado = true;
    let caracteresValidos = /^[a-zA-ZñÑ\s]+$/;
    if (!caracteresValidos.test(nombre)) {
        resultado = false;
        console.log(`Carácter no permitido: ${nombre}`);
    }
    return resultado;
}

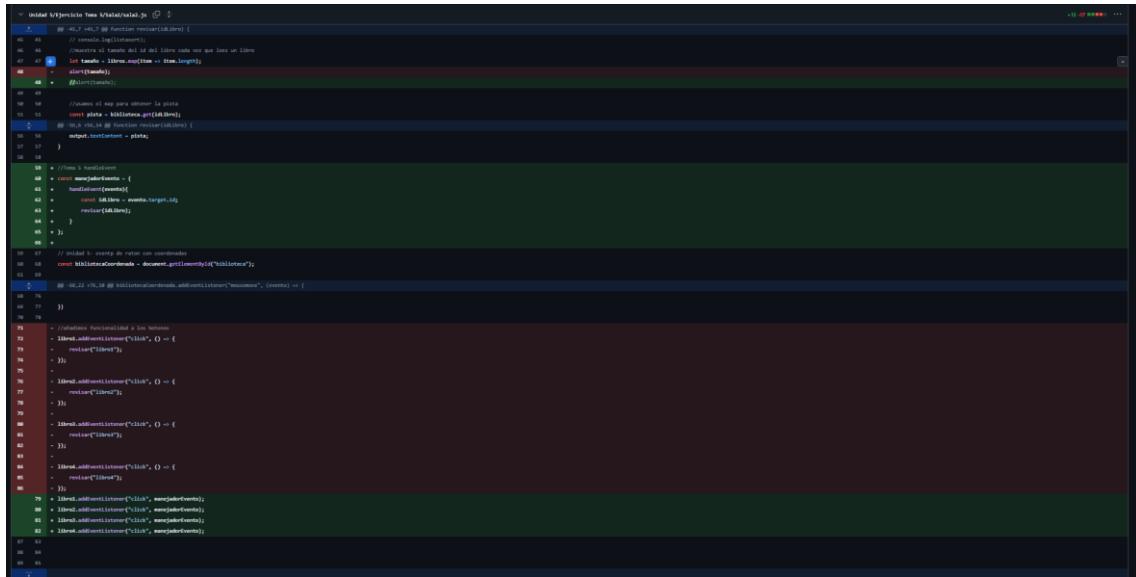
document.addEventListener("DOMContentLoaded", () => {
    const input = document.getElementById("input");
    const nombre = input.value;
    const resultado = validarNombre(nombre);
    if (!resultado) {
        console.log(`Nombre incorrecto`);
    } else {
        console.log(`Nombre correcto`);
    }
});

document.addEventListener("keyup", (evento) => {
    const nombre = document.getElementById("input").value;
    const resultado = validarNombre(nombre);
    if (!resultado) {
        console.log(`Carácter no permitido: ${evento.key}`);
    }
});

document.addEventListener("keydown", (evento) => {
    const nombre = document.getElementById("input").value;
    const resultado = validarNombre(nombre);
    if (!resultado) {
        console.log(`Carácter no permitido: ${evento.key}`);
    }
});

document.addEventListener("enter", (evento) => {
    const nombre = document.getElementById("input").value;
    const resultado = validarNombre(nombre);
    if (!resultado) {
        console.log(`Carácter no permitido: ${evento.key}`);
    }
});
```

Otro commit importante es el de handleEvent(). Este cambio la forma en que se usan los botones de los libros. Añadiendo stopPropagation() para detener la propagación al pulsar el botón. Además se cambiaron los EventListeners para que funcionen con la nueva función implementada que usa handleEvent() que a su vez llama la función revisar que se implementó en entregas anteriores.



```
1 // Unidad A/Ejercicio Tarea Actualizar.js
2
3 43 // console.log('libro');
4 44 // muestra el nombre del libro cada vez que loes un libro
5 45 const libro = document.querySelector('#libro');
6 46 libro.addEventListener('click', revisarLibro);
7
8 47 // start(book);
9
10 48 // #libro(book);
11
12 49 // /!> click para obtener la lista
13 50 const plata = biblioteca.getLibro();
14 51 // libro.click(book) => Function revisar(book);
15 52 // libro.click(book) => Function revisar(book);
16 53
17 54 // objeto.textContent = plata;
18
19
20 55 // /!> click(book);
21 56 const revisarBook = {
22 57   handleEvent(event) {
23 58     const libroTarget = event.target;
24 59     const libro = libroTarget.closest('.libro');
25 60     revisar(libro);
26 61   }
27 62 };
28
29
30 63 // /!> click(book);
31 64 const bibliotecaCreada = document.getElementById('biblioteca');
32
33 65 // libro.addEventListener('click', revisar);
34 66 // libro.addEventListener('click', revisar);
35 67 // libro.addEventListener('click', revisar);
36 68 // libro.addEventListener('click', revisar);
37 69 // libro.addEventListener('click', revisar);
38 70 // libro.addEventListener('click', revisar);
39 71 // libro.addEventListener('click', revisar);
40 72 // libro.addEventListener('click', revisar);
41 73 // libro.addEventListener('click', revisar);
42 74 // libro.addEventListener('click', revisar);
43 75 // libro.addEventListener('click', revisar);
44 76 // libro.addEventListener('click', revisar);
45 77 // libro.addEventListener('click', revisar);
46 78 // libro.addEventListener('click', revisar);
47 79 // libro.addEventListener('click', revisar);
48 80 // libro.addEventListener('click', revisar);
49 81 // libro.addEventListener('click', revisar);
50 82 // libro.addEventListener('click', revisar);
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
```