ANEXO · TAREAS DEL ALUMNADO

EVENTOS Y ACCIONES: CADA CONTROL HACE ALGO

Nombre: _____ Fecha: _____

EL CONTRATO (6 MIN)

Evento: algo que pasa, normalmente algo que hace el usuario (un clic, una tecla, un envío). **Acción**: lo que hace la app como respuesta. **Handler**: la función que los conecta. Completa la tabla de NoteBox con la clase, antes de escribir código:

DÓNDE	EVENTO	ACCIÓN
Título de una fila	click	
Delete de una fila	click	
Botón Undo	click	
Ventana	keydown · Ctrl+Z	
Ventana	keydown · Escape	
(tiempo)	5 s después de borrar	

PROGRAMA CON EL PROFESOR (26 MIN)

En tu `notebox`, con el archivo de Figma 10112026 · Events and actions.

OK	HAZ	ESTÁ HECHO CUANDO...
☐	`NoteRow` recibe `note`, `selected`, `onselect` y `ondelete`, y tiene dos `<button>` reales. Estilos del frame 1.	La fila tiene el título y un botón Delete rojo.
☐	La página guarda `selectedId` con `$state` y pasa `onselect={select}`.	Al hacer clic en un título, la fila se pone gris.
☐	Funciones `remove(id)` y `undo()` con `splice`, y la barra Undo con `role="status"` (frame 2).	Borras una nota, pulsas Undo y vuelve a su sitio.
☐	`<svelte:window onkeydown={onKey} />`: Escape deselecciona, Ctrl+Z deshace.	Puedes deshacer con el teclado.
☐	Un `$effect` con `setTimeout` de 5 s y su limpieza `clearTimeout`.	La barra desaparece sola a los 5 segundos.

Recuerda: `onclick={() => onselect(note.id)}` llama **al hacer clic**. `onclick={onselect(note.id)}` es una **llamada**, no una función: la página carga con normalidad, pero al hacer clic la función se ejecuta, Svelte lanza `state_unsafe_mutation` y la página acaba en **500 Internal Error**. Si hace algo, es un `<button>`; si va a otro sitio, es un enlace. Nunca un `<div>` con `onclick`.

EJERCICIO · FIJAR + TECLA SUPR (11 MIN)

Individual, a partir del frame **4** de Figma. Dos líneas nuevas del contrato:

OK	DÓNDE	EVENTO	ACCIÓN
☐	Botón Pin de una fila	click	Cambia `pinned` de esa nota. Texto «Pin» / «Unpin»; una fila fijada muestra la insignia PINNED del viernes.
☐	Ventana	keydown · Delete	Borra la nota seleccionada: la misma acción que su botón Delete.

Si algo falla:

- **La fila cambia `note.pinned` ella misma.** Parece que funciona, pero las notas son de la página. La fila «toca el timbre» (`onpin`); la página cambia el valor.
- **Has copiado una función de borrar nueva para la tecla, sin Undo.** Una acción, una función: llama a `remove(selectedId)`.
- **La tecla Supr da error si no hay nada seleccionado.** Añade la condición `&& selectedId`.
- **Supr dentro del buscador borra la nota seleccionada.** La ventana oye todas las teclas, también mientras escribes. Comprueba además `!event.target.matches('input, textarea')`.
- **En un portátil pequeño no tienes tecla Supr.** Prueba Fn + Backspace.

Comprueba: usa tu app solo con el teclado (Tab): selecciona, fija, borra y deshaz.

ENTREGA

Commit **«Session 4: events»** y push. En Classroom: el enlace y, como comentario, la **tabla de eventos y acciones de tu propia app** (al menos cinco líneas). Prepárala aquí:

DÓNDE	EVENTO	ACCIÓN

PARA CASA

- Termina el ejercicio.
- En **tu propia app**: conecta al menos cinco líneas de tu tabla. Al menos una callback prop y un atajo de teclado. Haz commit.
- Tutorial: `svelte.dev/tutorial`, parte «Basic Svelte», sección Events.
- Pregunta de transferencia: un reproductor de música tiene Play/Pause, un control deslizante de volumen y la barra espaciadora como atajo de Play/Pause. Escribe su tabla de eventos y acciones. ¿Qué dos líneas llaman a la misma función?