**Construye un reloj 25 + 5**

**Nota:** **React 18 tiene incompatibilidades conocidas con las pruebas de este proyecto (ver** [**issue**](https://github.com/freeCodeCamp/freeCodeCamp/issues/45922)**)**

**Objetivo:** Construye una aplicación que sea funcionalmente similar a esta: <https://25--5-clock.freecodecamp.rocks>.

Completa las siguientes historias de usuario y consigue que todas las pruebas pasen. Utiliza cualquier librería o API que necesites. Dale tu propio estilo personal.

Puedes utilizar cualquier combinación de HTML, JavaScript, CSS, Bootstrap, SASS, React, Redux y jQuery para completar este proyecto. Debes usar un framework frontend (como por ejemplo React), ya que esta sección trata sobre el aprendizaje de frameworks frontend. No se recomiendan otras tecnologías que no estén en la lista anterior y su uso queda bajo tu propia responsabilidad. Estamos considerando dar soporte a otros frameworks frontend como Angular y Vue, pero actualmente no están soportados. Aceptaremos e intentaremos solucionar todos los informes de problemas que utilicen la stack tecnológica sugerida para este proyecto. ¡Feliz día programando!

**Historia de usuario #1:** Puedo ver un elemento con id="break-label" que contiene una cadena (por ejemplo: "Break Length").

**Historia de usuario #2:** Puedo ver un elemento con id="session-label" que contiene una cadena (por ejemplo, "Session Length").

**Historia de usuario #3:** Puedo ver dos elementos que se pueden hacer clic con sus correspondientes IDs: id="break-decrement" y id="session-decrement".

**Historia de usuario #4:** Puedo ver dos elementos que se pueden hacer clic con sus correspondientes IDs: id="break-increment" y id="session-increment".

**Historia de usuario #5:** Puede ver un elemento con un id="break-length", que por defecto (en la carga) muestra un valor de 5.

**Historia de usuario #6:** Puedo ver un elemento con un correspondiente id="session-length", que por defecto muestra un valor de 25.

**Historia de usuario #7:** Puedo ver un elemento con un correspondiente id="timer-label", que contiene una cadena que indica que se inicializa una sesión (por ejemplo, "Session").

**Historia de usuario #8:** Puedo ver un elemento con el correspondiente id="time-left". NOTA: En pausa o en ejecución, el valor de este campo debe mostrarse siempre en formato mm:ss (es decir, 25:00).

**Historia de usuario #9:** Puedo ver un elemento que se puede hacer clic con el correspondiente id="start\_stop".

**Historia de usuario #10:** Puedo ver un elemento que se puede hacer clic id="reset".

**Historia de usuario #11:** Cuando hago clic en elemento con el id de reset, cualquier temporizador en marcha debe detenerse, el valor dentro de id="break-length" debe volver a 5, el valor dentro de id="session-length" debe volver a id="time-left" debe restablecerse a su estado por defecto.

**Historia de usuario #12:** Cuando hago clic en elemento con el id de break-decrement, el valor dentro de id="break-length" disminuye en un valor de 1, y puedo ver el valor actualizado.

**Historia de usuario #13:** Cuando hago clic en elemento con el id de break-increment, el valor dentro de id="break-length" se incrementa en un valor de 1, y puedo ver el valor actualizado.

**Historia de usuario #14:** Cuando hago clic en el elemento con el id de session-decrement, el valor dentro de id="session-length" decrementa por un valor de 1, y puedo ver el valor actualizado.

**Historia de usuario #15:** Cuando hago clic en el elemento con el id de session-increment, el valor dentro de id="session-length" se incrementa en un valor de 1, y puedo ver el valor actualizado.

**Historia de usuario #16:** No debe poder establecer una duración de sesión o pausa a <= 0.

**Historia de usuario #17:** No debe poder establecer una duración de sesión o pausa de > 60.

**Historia de usuario #18:** Cuando hago clic por primera vez en el elemento con id="start\_stop", el temporizador debe empezar a correr desde el valor actualmente mostrado en id="session-length", incluso si el valor ha sido incrementado o decrementado desde el valor original de 25.

**Historia de usuario #19:** Si el temporizador está en marcha, el elemento con el id time-left debe mostrar el tiempo restante en formato mm:ss (decrementando en un valor de 1 y actualizando la pantalla cada 1000ms).

**Historia de usuario #20:** Si el temporizador está en marcha y hago clic en el elemento con id="start\_stop", la cuenta atrás debería pausarse.

**Historia de usuario #21:** Si el temporizador está pausado y hago clic en el elemento id="start\_stop", la cuenta atrás debe reanudarse desde el punto en el que fue pausada.

**Historia de usuario #22:** Cuando la cuenta atrás de la sesión llega a cero (NOTA: el temporizador DEBE llegar a 00:00) y una nueva cuenta atrás comienza, el elemento cuyo id es timer-label debe mostrar una cadena en la que se indica que el descanso (break) ha comenzado.

**User Story #23:** When a session countdown reaches zero (NOTE: timer MUST reach 00:00), a new break countdown should begin, counting down from the value currently displayed in the id="break-length" element.

**User Story #24:** When a break countdown reaches zero (NOTE: timer MUST reach 00:00), and a new countdown begins, the element with the id of timer-label should display a string indicating a session has begun.

**User Story #25:** When a break countdown reaches zero (NOTE: timer MUST reach 00:00), a new session countdown should begin, counting down from the value currently displayed in the id="session-length" element.

**User Story #26:** When a countdown reaches zero (NOTE: timer MUST reach 00:00), a sound indicating that time is up should play. This should utilize an HTML5 audio tag and have a corresponding id="beep".

**User Story #27:** The audio element with id="beep" must be 1 second or longer.

**User Story #28:** The audio element with id of beep must stop playing and be rewound to the beginning when the element with the id of reset is clicked.

You can build your project by [using this CodePen template](https://codepen.io/pen?template=MJjpwO) and clicking Save to create your own pen. Or you can use this CDN link to run the tests in any environment you like: https://cdn.freecodecamp.org/testable-projects-fcc/v1/bundle.js

Once you're done, submit the URL to your working project with all its tests passing.