



Universidad Juárez del Estado de Durango

Facultad de Ingeniería Ciencias y Arquitectura

Ingeniería en Tecnologías Computacionales

Programación Web

Reporte de Práctica 20

Diego Rea Morales

Ing. Fabian Gallegos Gutiérrez

10 de Octubre del 2025

Índice

Contenido_____3

- Explicación del código JavaScript_____5

Conclusiones_____6

Contenido

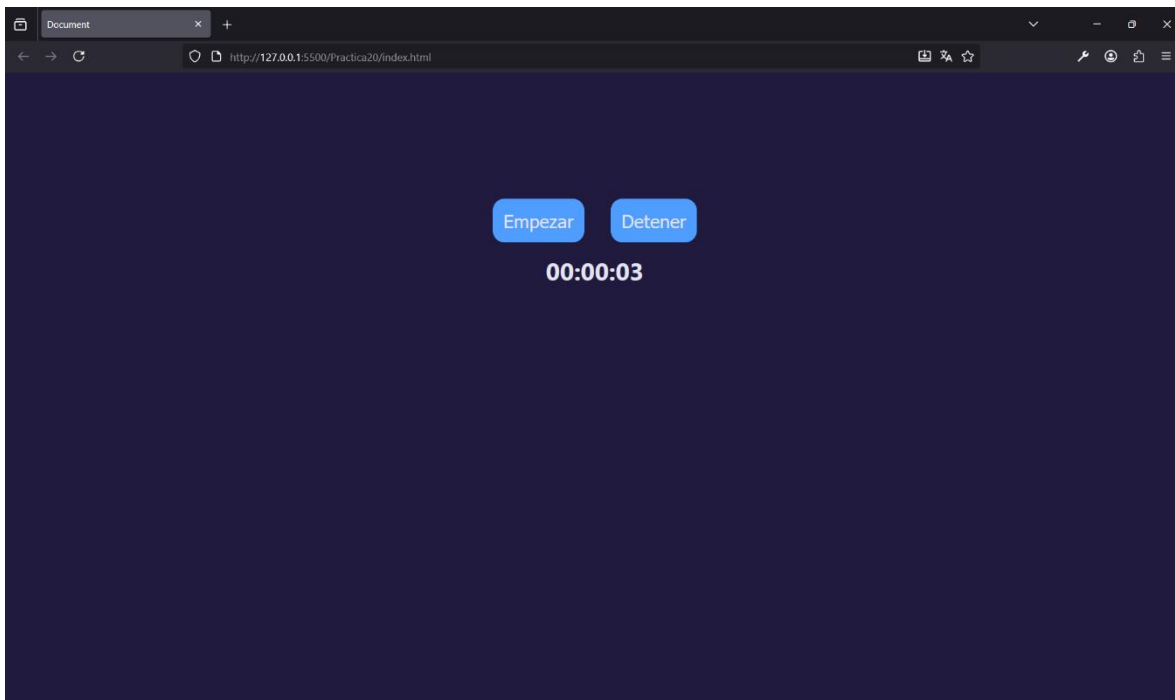
La práctica número 20 tiene como objetivo implementar en una pagina web, un cronometro, el cual, como objetivo, es mostrar el tiempo transcurrido desde que se inició hasta que se termine o se detenga.

En este caso, se utilizó JavaScript para poder realizar el cronometro funcional, aunado con el HTML y los estilos CSS.

```
index.html script.js styles.c
Practica20 > styles.css > *
1 {
2   margin: 0;
3   padding: 0;
4   box-sizing: border-box;
5 }
6 html {
7   font-size: 16px; /* 1rem = 16px */
8 }
9 body {
10  background-color: #211a3f;
11  display: flex;
12  justify-content: center;
13  align-items: center;
14  font-family: Segoe UI;
15 }
16 .crono {
17  margin: 150px;
18  display: flex;
19  flex-direction: column;
20  justify-content: center;
21  align-items: center;
22 }
23 button {
24  font-size: 1.5rem;
25  padding: 10px;
26  margin: 15px;
27  background-color: #4F9DFF;
28  color: #E4E4F2;
29  border: 4px solid transparent;
30  border-radius: 15px;
31  transition: all 0.7s ease;
32  cursor: pointer;
33 }
34 button:hover{
35  color: #4F9DFF;
36  background-color: #E4E4F2;
37  animation: coloresBorder 4s linear infinite;
38 }
39 @keyframes coloresBorder {
40  0% {
41    border-color: #3E2F78;
42  }
43  25% {
44    border-color: #00C8FF;
45  }
46  50% {
47    border-color: #8E44FF;
48  }
49  75% {
50    border-color: #00FFAA;
51  }
52  100% {
53    border-color: #FF3B6A;
54  }
55 }
56 h1{
57  color: #E4E4F2;
58 }
59
```

```
index.html X script.js styles.css Settings
Practica20 > index.html > ...
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <link rel="stylesheet" href="styles.css">
7   <title>Document</title>
8 </head>
9 <body>
10  <div class = "crono">
11    <div class="botones">
12      <button id="start" onclick="timer()">Empezar</button>
13      <button id="stop" onclick="stopTimer()">Detener</button>
14    </div>
15    <div class="text">
16      <h1 id="crono-text">00:00:00</h1>
17    </div>
18  </div>
19  <script src="script.js"></script>
20 </body>
21 </html>
```

```
index.html script.js styles.css Settings
Practica20 > script.js > ...
1  let idInterval
2  let segundos = 0;
3
4  function timer(){
5      idInterval = setInterval(starTimer, 1000);
6  }
7  function starTimer() {
8      const reloj = document.getElementById("crono-text");
9      segundos++;
10     let horas = Math.floor(segundos / 3600)
11     let minutos = Math.floor((segundos / 3600) / 60);
12     let segs= segundos % 60;
13
14     horas = horas < 10 ? '0' + horas: horas;
15     minutos = minutos < 10 ? '0' + minutos: minutos;
16     segs = segs < 10 ? '0' + segs: segs;
17
18     const tiempo = horas + ":" + minutos + ":" + segs;
19     reloj.innerHTML = tiempo;
20 }
21 function stopTimer(){
22     clearInterval(idInterval);
23     idInterval = null;
24 }
```



Explicación del código JavaScript

El código inicia creando e inicializando dos variables “*idInterval*” y “*segundos*”. La segunda se inicializa con un valor igual a cero, esta variable nos va a servir para llevar la cuenta del tiempo, en este caso cada segundo.

La primera será la que nos permita realizar el conteo, o mejor dicho crear un intervalo de tiempo en el cual se esté ejecutando nuestra lógica.

Después seguirían tres funciones, una de ellas seria para iniciar el cronometro, una se encarga de la lógica y la otra se encarga de parar el cronometro, dentro de los siguientes párrafos explicare el funcionamiento de cada una de estas funciones.

La primera función de nombre *timer*, esta función se realizó para poder iniciar el intervalo. Aquí es donde se tendrá que iniciar el cronometro. La forma en que funciona es a partir es un método el cual como parámetro principal o el que lo distingue es que acepta un valor en milisegundos, aparte requiere de otro parámetro, el cual en mi caso es otra función, es decir, este método toma como parámetros el tiempo o intervalo y lo que quiere que se ejecute cada que se intervalo ocurre.

La segunda función es la que se ejecuta cada segundo. El funcionamiento de esta función es que, cada que se ejecute, es decir cada segundo, el contador de los *segundos* se actualiza y se le suma uno, esto para que cada segundo que se ejecute esta función, el contador sume uno. Esto para que después, una variable de nombre *reloj*, esta variable toma el valor de un elemento HTML de id *crono-text*, que es un `<h1>` que se encuentra en el body del HTML y tiene de contenido “00:00:00”, para después, tres variables los cuales, en base al valor del contador de los segundos, estas toman el valor correspondiente en horas, minutos y segundos.

Después existen tres líneas de código, que lo que hacen cada una de ellas es simple y se repite en cada una de ellas, por ende, solo explicaré uno.

Primero, un valor (horas, minutos o segundos) se le asigna o agrega un 0 enfrente del número, siendo este número una unidad, es decir, si solo es un dígito, este le agrega un 0 enfrente, para que quede algo así: 01, 02, 03, 04, etc.

Cuando todo este listo, ahora el `<h1>` que le muestra el tiempo al usuario se actualiza y se muestra el tiempo transcurrido.

Por último, la ultima función simplemente detiene el cronometro.

Conclusiones

La práctica número 20 permitió comprender y aplicar de manera práctica el uso de JavaScript para controlar la lógica del tiempo en una página web, trabajando de forma conjunta con HTML y CSS. A través de esta actividad se logró implementar un cronómetro capaz de iniciar, actualizar y detener la cuenta del tiempo, demostrando la utilidad del lenguaje JavaScript para manipular elementos del DOM y gestionar eventos en tiempo real.

Durante el desarrollo, se reforzaron conceptos como la utilización de funciones, el manejo de intervalos de tiempo mediante `setInterval`, y la actualización dinámica del contenido en la interfaz. Asimismo, se evidenció la importancia de la interacción entre las distintas tecnologías web: HTML para la estructura, CSS para la presentación visual y JavaScript para la funcionalidad.