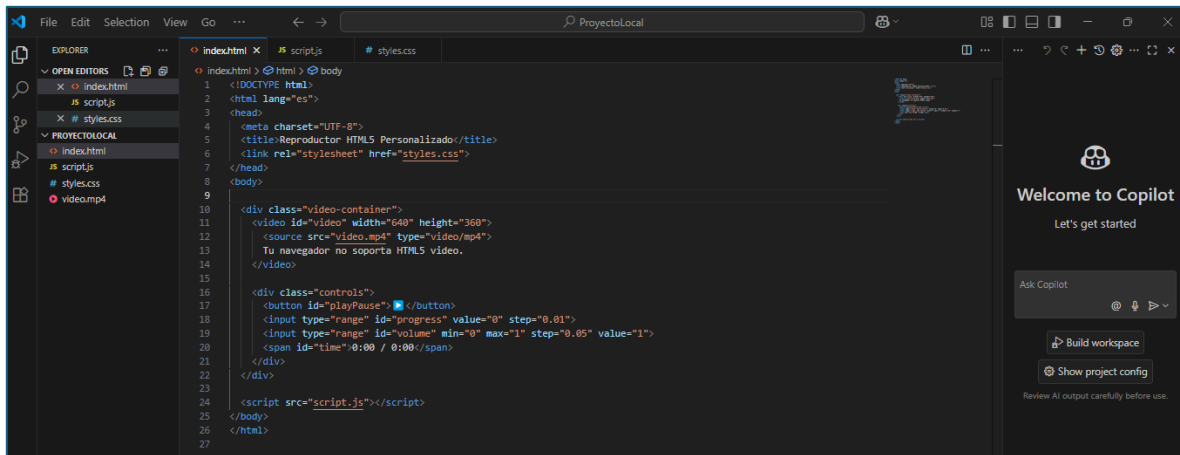


Nombre: Joaquín Cabrera Galarza

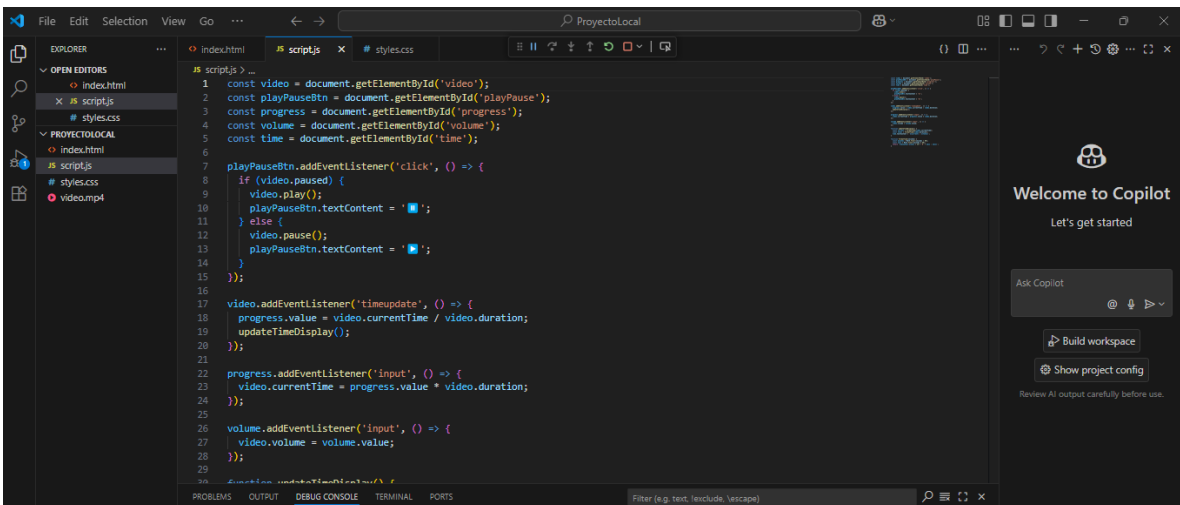
## TALLER 3.3. Integración de Contenido Multimedia en Aplicaciones Web

### Ejercicio 1: Reproductor HTML5 con Controles Personalizados

#### Código

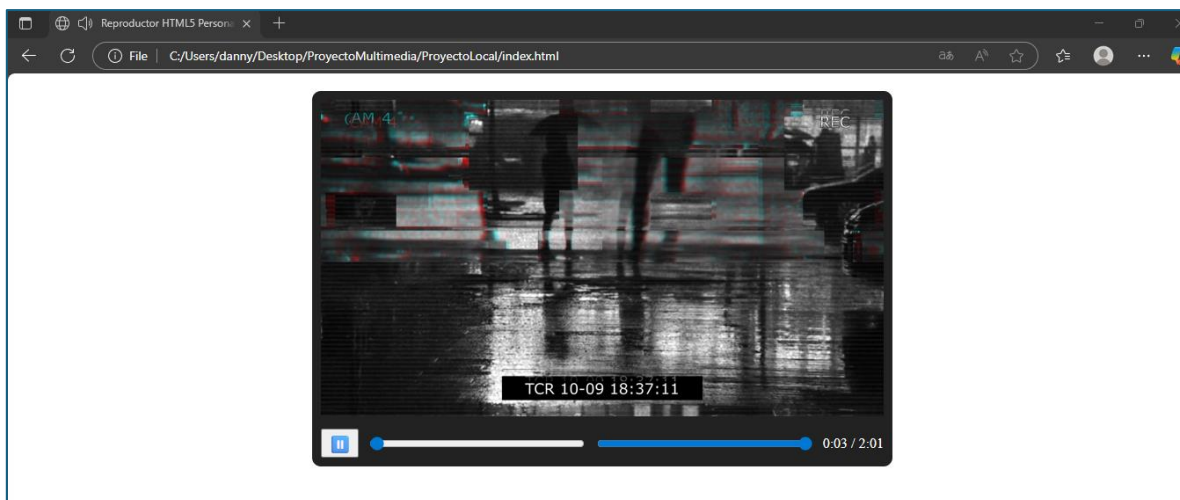


```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4 <meta charset="UTF-8">
5 <title>Reproductor HTML5 Personalizado</title>
6 <link rel="stylesheet" href="styles.css">
7 </head>
8 <body>
9
10 <div class="video-container">
11 <video id="video" width="640" height="360">
12 <source src="video.mp4" type="video/mp4">
13 Tu navegador no soporta HTML5 video.
14 </video>
15
16 <div class="controls">
17 <button id="playPause"></button>
18 <input type="range" id="progress" value="0" step="0.01">
19 <input type="range" id="volume" min="0" max="1" step="0.05" value="1">
20 <span id="time">0:00 / 0:00</span>
21 </div>
22 </div>
23
24 <script src="script.js"></script>
25 </body>
26 </html>
27
```



```
1 const video = document.getElementById('video');
2 const playPauseBtn = document.getElementById('playPause');
3 const progress = document.getElementById('progress');
4 const volume = document.getElementById('volume');
5 const time = document.getElementById('time');
6
7 playPauseBtn.addEventListener('click', () => {
8   if (video.paused) {
9     video.play();
10    playPauseBtn.textContent = '⏸';
11   } else {
12     video.pause();
13     playPauseBtn.textContent = '▶';
14   }
15 });
16
17 video.addEventListener('timeupdate', () => {
18   progress.value = video.currentTime / video.duration;
19   updateTimeDisplay();
20 });
21
22 progress.addEventListener('input', () => {
23   video.currentTime = progress.value * video.duration;
24 });
25
26 volume.addEventListener('input', () => {
27   video.volume = volume.value;
28 });
29
```

#### Ejecución

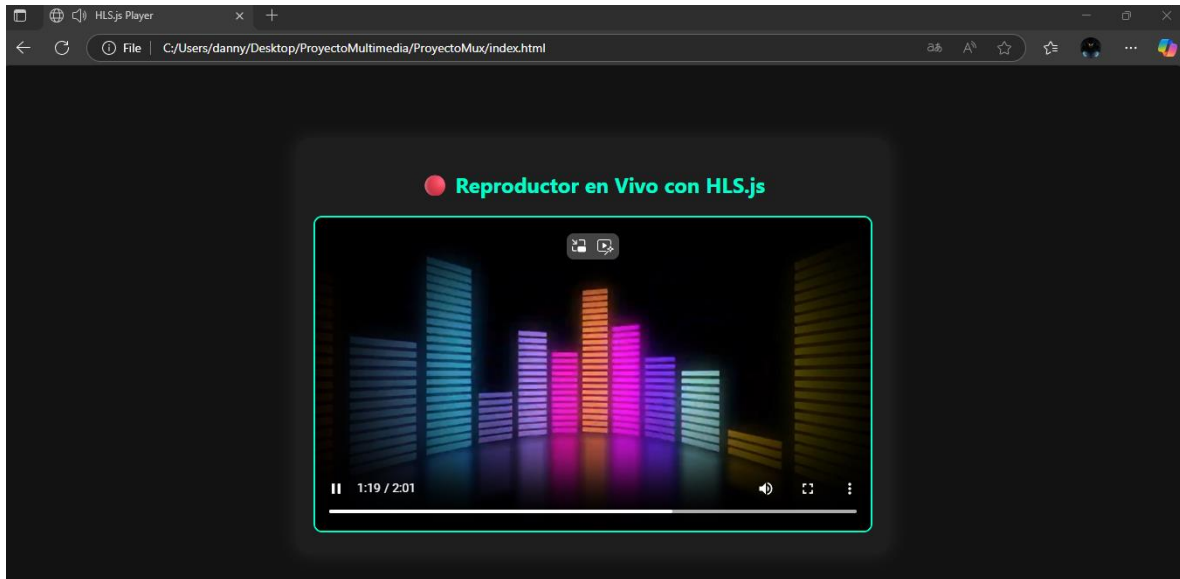


## Ejercicio 2: Integración de Contenido Multimedia en Aplicaciones Web (HTML5, Streaming y Plataformas SaaS)

### Código

```
index.html X
C:\Users\danny\Desktop\ProyectoMultimedia\ProyectoMux> index.html > ...
45 </head>
46 <body>
47
48 <div class="player-container">
49 <h1> Reproductor en Vivo con HLS.js</h1>
50 <video id="video" controls></video>
51 </div>
52
53 <script src="https://cdn.jsdelivr.net/npm/hls.js@latest"></script>
54
55 <script>
56 const video = document.getElementById('video');
57 const videoSrc = 'https://stream.mux.com/TPAu69701Qac9z1HeW805tH00zadJXR02Q64BykX01nTY.m3u8';
58
59 if (Hls.isSupported()) {
60   const hls = new Hls();
61   hls.loadSource(videoSrc);
62   hls.attachMedia(video);
63   hls.on(Hls.Events.MANIFEST_PARSED, () => {
64     video.play();
65   });
66 } else if (video.canPlayType('application/vnd.apple.mpegurl')) {
67   video.src = videoSrc;
68   video.addEventListener('loadedmetadata', () => {
69     video.play();
70   });
71 }
72 </script>
73 </body>
74 </html>
75
```

### Ejecución



### Links a Proyectos

[https://drive.google.com/file/d/13u5orj0AuCjJf\\_3ztlc\\_qOPUTdjQUk4K/view?usp=sharing](https://drive.google.com/file/d/13u5orj0AuCjJf_3ztlc_qOPUTdjQUk4K/view?usp=sharing)