



UNIVERSIDAD NACIONAL DEL ALTIPLANO

**FACULTAD DE  
INGENIERÍA MECÁNICA ELÉCTRICA, ELECTRÓNICA Y  
SISTEMAS  
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**

---



TRABAJO ENCARGADO

CURSO:

ESTRUCTURA DE DATOS AVANZADAS

DOCENTE:

ING. COLLANQUI MARTINEZ FREDY

PRESENTADO POR:

BELTRAN EDWIN MAMANI MAMANI

PUNO- PERU

2024

---

## REPRODUCTOR DE VÍDEO CON RE COMENDADOR

El estilo CSS define el diseño visual de la página, mientras que el JavaScript gestiona la funcionalidad, incluyendo la búsqueda de videos similares y la actualización dinámica del reproductor y la tabla de resultados.

### Estructura del Documento HTML

```
1  <html lang= "es" >
2  <head>
3  <meta charset= "UTF-8" >
4  <meta name= "viewport" content= "width=device-width, initial-scale=1.0" >
5  <title> Reproductor de vídeo con recomendador </title>
6  <style>
7  /* CSS para el estilo de la página */
8  </style>
9  </head>
10 <body>
11 <div class= "container" >
12 <h1> Reproductor de vídeo con recomendador </h1>
13 <!-- Formulario de entrada de parámetros -->
14 <div class= "form-group" >
15 <label for= "duration" > Duración (s): </label>
dieciséis <input type= "number" id= "duration" placeholder= "Ej. 180" >
17 </div>
18 <div class= "form-group" >
19 <label for= "resolution" > Resolución (p): </label>
20 <input type= "number" id= "resolution" placeholder= "Ej. 1080" >
21 </div>
22 <div class= "form-group" >
23 <label for= "framerate" > FPS: </label>
24 <input type= "number" id= "framerate" placeholder= "Ej. 30" >
25 </div>
26 <button id= "searchButton" > Buscar videos similares </button>
27 <!-- Reproductor de video -->
28 <div class= "player" >
29 <img id= "coverImage" src= "https://via.placeholder.com/400" alt= "Imagen de portada" >
30 <video id= "videoPlayer" controles >
31 <source src= "" type= "video/mp4" >
32 Tu navegador no soporta el elemento de vídeo.
33 </video>
34 </div>
35 <!-- Tabla para mostrar resultados -->
36 <table id= "resultsTable" >
37 <thead>
38 <tr>
39 <th> ID del vídeo </th>
40 <th> Duración </th>
41 <th> Resolución </th>
42 <th> FPS </th>
43 <th> Distancia </th>
44 </tr>
45 </thead>
46 <tbody></tbody>
47 </table>
```

```

48     </div>
49     <script >
50         // JavaScript para la funcionalidad
51     </script>
52 </body>
53 </html>

```

## Estilos CSS

Los estilos CSS proporcionan el diseño y formato visual de la página. Aquí están los detalles:

- ❖ **body**: Se establece la fuente, el color de fondo, y se eliminan los márgenes y el relleno.
- ❖ **.container**: Se define el ancho, el margen, el relleno, el color de fondo, la sombra, el radio del borde y el alineamiento del texto.
- ❖ **h1**: Se define el color del texto del título.
- ❖ **.form-group**: Se define el margen inferior para los grupos de formulario.
- ❖ **label**: Se define el color del texto y el margen inferior para las etiquetas.
- ❖ **input** y **button**: Se definen estilos para los campos de entrada y el botón, incluyendo el tamaño, el color de fondo, el color del texto, y el borde.
- ❖ **table**: Se definen estilos para la tabla, incluyendo el borde, el colapso del borde y el margen superior.
- ❖ **.player**: Se define el estilo del contenedor del reproductor de video, incluyendo el ancho, el color de fondo y la posición.
- ❖ **.player img** y **.player video**: Se definen los estilos para la imagen de portada y el video, incluyendo la posición, el tamaño y la opacidad.

## Estructura HTML

- ✓ **Formulario de Entrada**: Permite al usuario ingresar parámetros como duración, resolución y FPS para buscar videos similares.
- ✓ **Reproductor de Video**: Muestra una imagen de portada y un video con controles.
- ✓ **Tabla de Resultados**: Muestra los resultados de búsqueda de videos similares con sus características y la distancia calculada.

## JavaScript

El script JavaScript se encarga de la funcionalidad dinámica de la página:

### Datos de Video:

```

1  constantes videos = [
2  { id : 1 , duración : 180 , resolución : 1080 , velocidad de fotogramas : 30 , imagen :
3  https://via.placeholder.com/400' , video : ' video1.mp4' } ,
4

```

```

5 { id : 2 , duración : 200 , resolución : 720 , velocidad de fotogramas : 60 , imagen : '
6 https://via.placeholder.com/400' , vídeo : 'video2.mp4' } ,
  { id : 3 , duración : 190 , resolución : 1080 , velocidad de fotogramas : 24 , imagen :
  'https://via.placeholder.com/400' , vídeo : ' video3.mp4' } ,
  { id : 4 , duración : 175 , resolución : 480 , velocidad de cuadros : 30 , imagen : 'https://via.placeholder.com/400'
  , vídeo : ' video4.mp4 ' }
];

```

Define un array de objetos que representan los videos disponibles con sus propiedades.

#### ▮ Evento del Botón:

```

1 documento.getElementById( 'searchButton' ).addEventListener( 'click' , function () {
2   const duración = documento.getElementById( 'duración' ).valor;
3   const resolución = documento.getElementById( 'resolución' ).valor;
4   const velocidad de cuadros = documento.getElementById( 'velocidad de cuadros' ).valor;
5   if ( ! duración || ! resolución || ! velocidad de cuadros ) {
6     alert( 'Por favor, completa todos los campos.' );
7     devolver ;
8   }
9   constante entradaVideo = {
10    duración : parseFloat (duración),
11    resolución : parseFloat (resolución),
12    velocidad de fotogramas : parseFloat (velocidad de fotogramas)
13  };

```

Se añade un evento al botón de búsqueda que recoge los valores de los campos de entrada, valida que no estén vacíos y calcula la distancia.

#### ▮ Cálculo de Distancia:

```

1 const resultados = videos.map(video => {
2   const distancia = Math.sqrt(
3     Math.pow(video.duración - inputVideo.duration , 2) + Math.pow (video.resoluci ó n - inputVideo.resolución,
4     2) + Math .pow(video.velocidad de fotogramas - inputVideo.framerate, 2 )
5
6   );
7   return { ...video, distancia : distancia.toFixed( 2 ) };
8 }).sort((a, b) => a.distancia - b.distancia);

```

Se calcula la distancia entre el video de entrada y cada video en el array usando la fórmula de la distancia euclidiana. Los resultados se ordenan por distancia.

#### ▮ Actualización de la Tabla:

```

1 const resultsTable = documento.getElementById( 'resultsTable' ).getElementsByTagName( 'tbody' )[ 0 ];
2 resultadosTable.innerHTML = " ;
3 resultados.forEach(video => {
4   const fila = resultsTable.insertRow();
5   fila.insertCell( 0 ).textContent = video.id;row.insertCell
6   ( 1 ).textContent = video.duración ;row.insertCell
7   ( 2 ).textContent = video.resolución ;

```

```
8 row.insertCell( 3 ).textContent = video.velocidad de fotografamas;
9 fila.insertCell( 4 ).textContent = video.distancia;
10 fila.addEventListener( 'click' , () => {
11     documento.getElementById( 'coverImage' ).src = video.imagen;
12     documento.getElementById( 'videoPlayer' ).src = video.video;
13 });
14 });
```

Se actualiza la tabla con los resultados de la búsqueda. Cada fila en la tabla muestra los detalles del video y permite que al hacer clic en una fila se actualicen la imagen de portada y el video del reproductor.

## Reproductor de vídeo con recomendador

Duración (s):

Ejemplo 180

Resolución (p):

Ejemplo 1080

FPS:

Ejemplo 30

Buscar videos similares



Vídeo de identificación	Duración	Resolución	FPS	Distancia
-------------------------	----------	------------	-----	-----------