

Actividad Proyecto. Myspotify

CASO DE USO Reproducir Canciones

Nuestro proyecto necesita como funcionalidad principal el que el usuario pueda reproducir canciones. En esta actividad vamos a abordar como realizar esto en el proyecto.

Hay una serie de pasos que debes contemplar para hacer esto

1 Crear almacén de archivos de audio

Cada canción se almacenará de en archivo music de la carpeta songs del proyecto. Esto más adelante se sustituirá por una ruta al servidor de streaming (ICECAST)

Ruta proyecto/songs/

2.- Copia el archivo de audio (c1.mp3) del aula virtual a este directorio de tú proyecto

Vamos a trabajar con una canción de prueba. Baja la canción c1.mp3 del aula virtual y guárdala en tú proyecto

3.-Cambia el controller Canción para que devuelva el archivo de audio de una canción que se le solicite

Fíjate en la nomenclatura de la ruta de esta acción

entidad/{id}/accion

```
# incluye esto en tú controller de canciones
use Symfony\Component\HttpFoundation\BinaryFileResponse;

#[Route('/cancion/{songName}/play', name: 'play_music', methods: ['GET'])]
public function playMusic(string $songName): Response
{
    $musicDirectory = $this->getParameter('kernel.project_dir') . '/songs/';
    $filePath = $musicDirectory . $songName;

    if (!file_exists($filePath)) {
        return new Response('Archivo no encontrado', 404);
    }

    return new BinaryFileResponse($filePath);
}
```

```
}
```

Ahora añade un método a CancionController para que puedas atender el inicio de la interfaz de búsqueda y reproducción de una canción

```
#[Route('/music', name: 'app_music')]

public function index(): Response
{
    return $this->render('play/play.html.twig', [
        'controller_name' => 'MusicController',
    ]);
}
```

La interfaz HTML es muy sencilla. Fíjate en el código javascript porque es donde se enlaza el nombre de la canción que poner con la solicitud de reproducción

```
<!DOCTYPE html>

<html lang="es">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Reproductor de Música</title>

</head>

{% block body %}

<body>

    <h1>Reproductor de Música</h1>

    <input type="text" id="songName" placeholder="Nombre de la canción (example.mp3)" />

    <button onclick="playSong()">Reproducir</button>

    <audio id="audioPlayer" controls style="display: none; margin-top: 20px;">

        Tu navegador no soporta el elemento de audio.

    </audio>
```

```

<script>

function playSong() {

    const songName = document.getElementById('songName').value;

    if (!songName) {

        alert('Por favor, ingresa el nombre de la canción.');
```

```
        return;

    }

    const audioPlayer = document.getElementById('audioPlayer');

    audioPlayer.src = `/cancion/${songName}/play`

    console.error('Error al reproducir la canción:', err);

    alert('No se pudo reproducir la canción. Asegúrate de que el archivo existe.');
```

```
    });

}

</script>
</body>
{% endblock %}
</html>

```

Prueba todos estos componentes en tú proyecto para que puedas reproducir la canción desde el navegador

CAMBIOS NECESARIOS EN EL PROYECTO

- 1.- Debes **añadir un nuevo campo a la clase Canción** denominado archivo. Este nuevo campo guardará el nombre del archivo físico de la canción (c1.mp3"
- 2.- La interfaz gráfica debe mostrar una forma de mostrar las canciones de forma que cuando pulses sobre ellas se reproduzca (el código de java script actue y realice la llamada al controller)

¿Dónde se realiza la solicitud al controlador?

¿ Cómo hacemos para que nos aparezcan canciones que podamos reproducir?

Idea: por cada canción que devuelva el Repositoty canciones puedes hacer que se devuelva un código javascript como este.

```
const img = document.createElement('img');  
  
img.src = song.image; // Ruta de la imagen asociada a la canción  
  
img.alt = song.name; // Este name será el valor del atributo archivo de cada Cancion  
  
img.onclick = () => playSong(song.name);
```

Este contenido crearía dinámicamente cada canción en el bloque de central de la plantilla twig