Desarrollo en Servidor

Estudio de devolver una respuesta en JSON desde un Contralador del Servidor

La siguiente estructura muestra el formato JSON que tiene que generar el controlador PlayList (PlayListController) al ser accedido por la RUTA http

/playlists/json.

Esta ruta permite obtener la estructura JSON de las PlayList de un usuario del sistema para que el componente Angular de cliente muestre esta información en la interfaz

NOTA : El ejemplo siguiente asume que en la base de datos (Servidor) hay dos playlist y se asume que son las del sistema, pero en el proyecto serán las del usuario que ha hecho LOGIN

PlayList creadas->"Playlist Favorita" con canción A;B y C y "Playlist Relajante" con canciones D y E


```
[
  {
    "id": 1,
    "nombre": "Playlist Favorita",
    "visibilidad": "Privada",
    "likes": 150,
    "canciones": [
      {
        "id": 1,
        "titulo": "Canción A",
        "duracion": 210,
        "album": "Álbum A",
        "autor": "Autor A",
        "genero": {
          "id": 1,
          "nombre": "Electrónica",
          "descripcion": "Música electrónica para todos los gustos."
        },
        "likes": 1000
      },
```

```
    {
      "id": 2,
      "titulo": "Canción B",
      "duracion": 180,
      "album": "Álbum B",
      "autor": "Autor B",
      "genero": {
        "id": 2,
        "nombre": "Pop",
        "descripcion": "Pop para todos los momentos."
      },
      "likes": 500
    },
    {
      "id": 3,
      "titulo": "Canción C",
      "duracion": 240,
      "album": "Álbum C",
      "autor": "Autor C",
      "genero": {
        "id": 1,
        "nombre": "Electrónica",
        "descripcion": "Música electrónica para todos los gustos."
      },
      "likes": 750
    }
  ]
},
{
  "id": 2,
  "nombre": "Playlist Relajante",
```

```
  "visibilidad": "Pública",
  "likes": 250,
  "canciones": [
    {
      "id": 4,
      "titulo": "Canción D",
      "duracion": 300,
      "album": "Álbum D",
      "autor": "Autor D",
      "genero": {
        "id": 3,
        "nombre": "Clásica",
        "descripcion": "Música clásica para momentos de tranquilidad."
      },
      "likes": 1200
    },
    {
      "id": 5,
      "titulo": "Canción E",
      "duracion": 200,
      "album": "Álbum E",
      "autor": "Autor E",
      "genero": {
        "id": 4,
        "nombre": "Jazz",
        "descripcion": "Jazz relajante para disfrutar la tarde."
      },
      "likes": 650
    }
  ]
}
```

```
 ]
```

Hay varias opciones para generar este JSON

1 Usar json_encode() directamente

Si las clases están bien definidas con métodos getters, puedes simplemente convertir las instancias de tus objetos con:

```
$json = json_encode($miPlaylist, JSON_PRETTY_PRINT | JSON_UNESCAPED_UNICODE);

echo $json;
```

2: Implementar JsonSerializable

Si tus objetos tienen propiedades privadas, lo mejor es implementar la **interfaz JsonSerializable** para controlar qué se serializa:

```
class Cancion implements JsonSerializable {

// resto de clase…

//Implementar este método de la interfaz

public function jsonSerialize(): array

 { return [

 'id' => $this->id,

'titulo' => $this->titulo,

 'duracion' => $this->duracion,

'album' => $this->album,

'autor' => $this->autor,

'genero' => $this->genero,

'likes' => $this->likes

 ];

 }
```

```
class Playlist implements JsonSerializable {

// resto de clase…


//Implementar este método de la interfaz
```

```
public function jsonSerialize(): array {
return [ 'id' => $this->id,
'nombre' => $this->nombre,
'visibilidad' => $this->visibilidad,
'likes' => $this->likes,
'canciones' => $this->canciones ];
 }
```

Acciones en el Proyecto

Prueba una de las acciones para conseguir devolver la playlist y canciones al componente angular de la parte cliente

Para ello añade un nuevo método al PlayListController denominado getJSONPlayList()

La estructura de este método será la siguiente

```
Ruta /playlist/json
public function getJSONPlayLists():response{
//1.- obtener usurio de la sesión para obtener sus playlist
//2.-Implementar el acceso a las playlist del usuario
 //3.- serialización en JSON del objeto u objetos Playlists (PlayListRespository)
//4.- La respuesta será la estructura JSON comentada
}
```