

### Objetivo: Consumir datos de un API de Spotify

localhost:4200/artist/3nlpTZci9O5W8RsNoNH...

Artista: Luis R Conriquez

Artista: Luis Miguel

Artista: Juan Luis Guerra 4.40

Artista: Luis Fonsi

Artista: Luis Angel "El Flaco"

Artista: Luis Alfonso Partida El Yaki

Artista: Juan Luis Guerra 4.40

[Ir a la página del artista](#)

Foto	Album	Cancion	Vista Previa
	El Último Adios	El Último Adios - Varios Artistas Version	
	Bachata Rosa	Bachata Rosa	
	Bachata Rosa	Burbujas de Amor	
	Bachata Rosa	La Sirenita	
	Alondra	Bachata En Fukuoka	

Seguir indicaciones.

1. Crear proyecto con Routing

```
PS C:\Users\gsp_g\OneDrive\Documentos\OTecnologiasWebEneJun2022\angular2022\spoty> ng new spotify
? Would you like to add Angular routing? Yes
? Which stylesheet format would you like to use? CSS
```

2. Uso de Bootstrap versión 4.6 vía CDN

index.html

```
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist
</head>

.
.

<link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/css/bootstrap.min.css"
integrity="sha384-
zCbKRCUGaJDkqS1kPbPd7TveP5iyJE0EjAuZQTgFLD2ylzuqKfdKlfG/eSrtxUkn"
crossorigin="anonymous">

.
.

<script src="https://cdn.jsdelivr.net/npm/jquery@3.5.1/dist
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1
<script src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/
</body>

.
.

<script src="https://cdn.jsdelivr.net/npm/jquery@3.5.1/dist/jquery.slim.min.js"
integrity="sha384-
DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj"
crossorigin="anonymous"></script>
<script
src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js"
integrity="sha384-
9/reFTGAw83EW2RDu2S0VKaIzap3H66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN"
crossorigin="anonymous"></script>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/js/bootstrap.min.js"
integrity="sha384-
VHvPCCyXqtD5DqJeNxl2dtTyhF78xXNXdkwX1CZerUsQfRKp+tA7hAShOK/B/fQ2"
crossorigin="anonymous"></script>
```

En el **index.html** de su proyecto colocar CDN para uso de fontawesome. **Nota:** Este CDN es el de mi cuenta del sitio de fontawesome, cuando tengan oportunidad crear el propio. Para esta práctica usar el mío.

```
9 | <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.13.0/css/all.css" integrity="sha384-Bfad6CLCknfcloXF0yFnlgtENryhrpZCe29RTifKEixXQZ38WheV+i/6YWSzkz3V" crossorigin="anonymous">
10 | </head>
```

```
<link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.13.0/css/all.css"
integrity="sha384-
Bfad6CLCknfcloXF0yFnlgtENryhrpZCe29RTifKEixXQZ38WheV+i/6YWSzkz3V"
crossorigin="anonymous">
```

## Practica API spotify

En terminal de visual studio, \_\_

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
PS C:\Users\gsp_g\OneDrive\Documentos\0TecnologiasWebEneJun2022\angular2022\spoty\spotify> ng g s spotify
CREATE src/app/spotify.service.spec.ts (362 bytes)
CREATE src/app/spotify.service.ts (136 bytes)
PS C:\Users\gsp_g\OneDrive\Documentos\0TecnologiasWebEneJun2022\angular2022\spoty\spotify> █
```

API's de Spotify

<https://developer.spotify.com/console/>

En el método `getQuery` debes colocar tu token personal de SPOTIFY que tiene de vigencia solo 1 hora

app > TS spotify.service.ts > ...

```
import { Injectable } from '@angular/core';
import { HttpClient, HttpHeaders } from '@angular/common/http'; 1
import { map } from 'rxjs/operators'; 4
@Injectable({
  providedIn: 'root'
})
export class SpotifyService {

  constructor(private http: HttpClient) { 2
    console.log("Spotify Service listo");
  }

  getQuery(query:string){
    const url = `https://api.spotify.com/v1/${query}`;
    const headers = new HttpHeaders({'Authorization': 'Bearer BQA20sv 3
    return this.http.get(url, {headers});
  }

  getArtistas(termino: string){
    console.log("3");
    return this.getQuery(`search?q=${termino}&type=artist&limit=6`) 5
      .pipe( map ( (data:any) => (data.artists.items ) ));
  }
}
```

```
constructor(private http: HttpClient) {  
  console.log("Spotify Service listo");  
}  
  
getQuery(query:string){  
  const url = `https://api.spotify.com/v1/${query}`;  
  const headers = new HttpHeaders({'Authorization': 'Bearer  
BQA20svIYm9T5cZ8_BYj-  
OPx9m417KUApLdgsUd7fFZpTbkKPkVcDYkkjtJEr8v2Vh6FbLMxMcgnVP51cW8'})  
  return this.http.get(url, {headers});  
}  
  
getArtistas(termino: string){  
  
  return this.getQuery(`search?q=${termino}&type=artist&limit=6`)  
  .pipe( map ( (data:any) => (data.artists.items) ));  
}  
  
.  
.  
  
import { map } from 'rxjs';  
import { HttpClient, HttpHeaders } from '@angular/common/http';
```

En el **app.module.ts** configurar el HttpClientModule en el array de imports

```
imports: [  
  BrowserModule,  
  AppRoutingModule,  
  HttpClientModule  
],  
  
.  
.  
  
import { HttpClientModule } from '@angular/common/http';  
  
.  
.
```

```
import { HttpClientModule } from '@angular/common/http';  
  
  
, HttpClientModule
```

En terminal de visual studio, crear componente **search**

```
PS C:\Users\gsp_g\OneDrive\Documentos\0TecnologiasWebEneJun2022\angular2022\spoty\spotify> ng g c search  
CREATE src/app/search/search.component.html (21 bytes)  
CREATE src/app/search/search.component.spec.ts (626 bytes)  
CREATE src/app/search/search.component.ts (275 bytes)  
CREATE src/app/search/search.component.css (0 bytes)  
UPDATE src/app/app.module.ts (538 bytes)
```

En terminal de visual studio, crear componente **tarjetas**

```
PS C:\Users\gsp_g\OneDrive\Documentos\0TecnologiasWebEneJun2022\angular2022\spoty\spotify> ng g c tarjetas  
CREATE src/app/tarjetas/tarjetas.component.html (23 bytes)  
CREATE src/app/tarjetas/tarjetas.component.spec.ts (640 bytes)  
CREATE src/app/tarjetas/tarjetas.component.ts (283 bytes)  
CREATE src/app/tarjetas/tarjetas.component.css (0 bytes)  
UPDATE src/app/app.module.ts (714 bytes)
```

En terminal de visual studio, crear componente **artista**

```
PS C:\Users\gsp_g\OneDrive\Documentos\0TecnologiasWebEneJun2022\angular2022\spoty\spotify> ng g c artista  
CREATE src/app/artista/artista.component.html (22 bytes)  
CREATE src/app/artista/artista.component.spec.ts (633 bytes)  
CREATE src/app/artista/artista.component.ts (279 bytes)  
CREATE src/app/artista/artista.component.css (0 bytes)  
UPDATE src/app/app.module.ts (800 bytes)
```

En el **app-routing.module.ts**

```
app > TS app-routing.module.ts > ...
```

```
  ∨ import { NgModule } from '@angular/core';  
    import { RouterModule, Routes } from '@angular/router';  
    import { ArtistaComponent } from '../artista/artista.component';
```

```
  ∨ const routes: Routes = [  
    | {path: 'artist/:id', component: ArtistaComponent}  
    | ];
```

```
  ∨ @NgModule({  
    | imports: [RouterModule.forRoot(routes)],  
    | exports: [RouterModule]  
    | })  
    export class AppRoutingModule { }
```

```
    {path: 'artist/:id', component: ArtistaComponent}
```

```
    .  
    .  
    import { ArtistaComponent } from '../artista/artista.component';
```

En tarjetas.component.ts

```
import { Component, Input, OnInit } from '@angular/core';  
import { Router } from '@angular/router';  
  
@Component({  
  selector: 'app-tarjetas',  
  templateUrl: './tarjetas.component.html',  
  styleUrls: ['./tarjetas.component.css']  
})  
export class TarjetasComponent implements OnInit {  
  
  @Input() items: any[] = [];  
  constructor(private router: Router) { }  
  
  ngOnInit(): void {  
  
    verArtista( item: any ){  
      console.log("metodo verArtista de componente tarjetas");  
      let artistaId;  
  
      if(item.type === 'artist'){  
        artistaId = item.id;  
      }  
      else{  
        artistaId = item.artists[0].id;  
      }  
      this.router.navigate(['/artist',artistaId]);  
    }  
  }  
}
```



```
@Input() items: any[] = [];  
  constructor(private router: Router) { }  
  
.  
  
.  
  
verArtista( item: any ){  
  console.log("metodo verArtista de componente tarjetas");  
  let artistaId;  
  
  if(item.type === 'artist'){  
    artistaId = item.id;  
  }  
  else{  
    artistaId = item.artists[0].id;  
  }  
  this.router.navigate(['/artist',artistaId]);  
}  
  
.  
  
.
```

En search.component.ts

```
<> index.html    TS search.component.ts X
src > app > search > TS search.component.ts > SearchComponent
1  import { Component, OnInit } from '@angular/core';
2  import { SpotifyService } from '../spotify.service';
3
4  @Component({
5    selector: 'app-search',
6    templateUrl: './search.component.html',
7    styleUrls: ['./search.component.css']
8  })
9  export class SearchComponent implements OnInit {
10
11    bandera?: boolean;
12
13    constructor(private spotify: SpotifyService) { }
14    artistas: any[] = [];
15
16    buscar(termino: string){
17      console.log("buscar");
18      this.bandera = true;
19
20      this.spotify.getArtistas(termino)
21        .subscribe ( (data:any) => {
22          this.artistas = data;
23          this.bandera=false;
24        } );
25    }
26
27    ngOnInit(): void {
28    }
29
30  }
```

```
.  
.  
  
bandera?: boolean;  
  
constructor(private spotify: SpotifyService) { }  
artistas: any[] = [];  
  
buscar(termino: string){  
    console.log("buscar");  
    this.bandera = true;  
  
    this.spotify.getArtistas(termino)  
        .subscribe ( (data:any) => {  
            this.artistas = data;  
            this.bandera=false;  
        } );  
}  
  
.  
.  
import { SpotifyService } from '../spotify.service';
```


En terminal de visual studio, crear componente loading. Para el manejo de:



```
PS C:\Users\gsp_g\OneDrive\Documentos\0TecnologiasWebEneJun2022\angular2022\spoty\spotify> ng g c loading  
CREATE src/app/loading/loading.component.html (22 bytes)  
CREATE src/app/loading/loading.component.spec.ts (633 bytes)  
CREATE src/app/loading/loading.component.ts (279 bytes)  
CREATE src/app/loading/loading.component.css (0 bytes)  
UPDATE src/app/app.module.ts (624 bytes)
```

## loading.component.html

<> loading.component.html X

src > app > components > shared > loading > <> loading.component.html > 

```
1 <div class="row text-center animated fadeIn">
2   <div class="col">
3     <i class="fa fa-sync fa-spin fa-5x"></i>
4   </div>
5 </div>
```

```
<div class="row text-center animated fadeIn">
  <div class="col">
    <i class="fa fa-sync fa-spin fa-5x"></i>
  </div>
</div>
```

## style.css del proyecto

src > # styles.css > ...

```
1  /* You can add global styles to this file, :
2
3  /* Animaciones */
4
5  .animated {
6    -webkit-animation-duration: 1s;
7    animation-duration: 1s;
8    -webkit-animation-fill-mode: both;
9    animation-fill-mode: both;
10 }
11
12 .fadeIn {
13   animation-name: fadeIn;
14 }
```

```
.animated {  
  -webkit-animation-duration: 1s;  
  animation-duration: 1s;  
  -webkit-animation-fill-mode: both;  
  animation-fill-mode: both;  
}  
  
.fadeIn {  
  animation-name: fadeIn;  
}
```

En el `search.component.html`

```
<> search.component.html X  
src > app > components > search > <> search.component.html > ...  
1  <div class="row m-5" >  
2  |   <div class="col">  
3  |   |   <input #termino type="text" (keyup)="buscar(termino.value)" class="form-  
4  |   |   </div>  
5  |   </div>  
6  <app-loading *ngIf="this.bandera"></app-loading>  
7  <app-tarjetas *ngIf="!this.bandera" [items]="this.artistas"></app-tarjetas>  
8  |
```

```
<div class="row m-5" >  
  <div class="col">  
    <input #termino type="text" (keyup)="buscar(termino.value)"  
class="form-control" placeholder="Buscar Artista...">  
  </div>  
</div>  
<app-loading *ngIf="this.bandera"></app-loading>  
<app-tarjetas *ngIf="!this.bandera" [items]="this.artistas"></app-tarjetas>
```

En el **app.component.html**

```
<app-search></app-search>

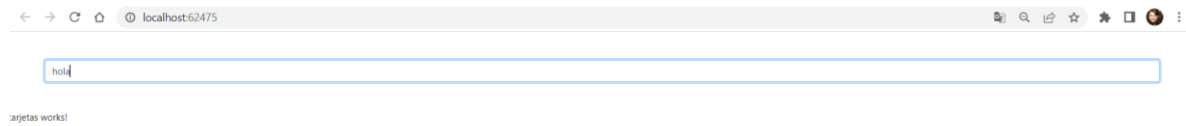
<div class="container m-5">
  <router-outlet></router-outlet>

</div>
```

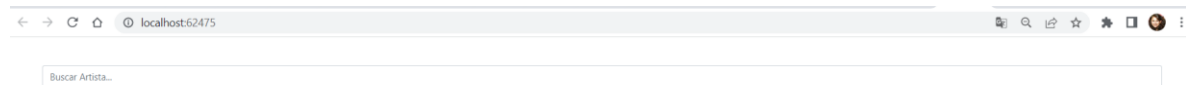
**Si expiro tu token, actualizarlo en el servicio**

Levantar el servidor

Así se ve



**Borrar el tarjetas Works**



Agregar dos métodos mas al *spotify.service.ts*

```
getArtista(id: string){  
  
    return this.getQuery(`artists/${id}`);  
  
}  
  
getTopTracks ( id:string){  
  
    return this.getQuery(`artists/${id}/top-tracks?country=us`)  
    .pipe( map ( (data:any) => data['tracks'] ));  
}
```

En **getArtistas** imprimir la variable termino y revisarla en consola.

```
getArtistas(termino: string){  
    console.log(termino);  
    return this.getQuery(`search?q=${termino}&type=artist&limit=6`)  
    .pipe( map ( (data:any) => (data.artists.items) ));  
}
```

Dentro de la clase de **artista.component.ts**, colocar el siguiente código

```
artista: any = {};  
loadingArtist: boolean;  
topTracks: any[] = [];  
  
constructor(private router: ActivatedRoute,  
             private spotify: SpotifyService) {  
    this.loadingArtist = true;  
  
    this.router.params.subscribe( params => {  
        this.getArtista( params['id'] );  
        this.getTopTracks( params['id'] );  
        console.log("llego" + params['id']);  
    });  
}  
  
ngOnInit(): void {  
}  
  
getArtista( id: string ){  
    this.loadingArtist = true;  
    this.spotify.getArtista(id).subscribe( artista => {  
        console.log(artista);  
        this.artista = artista;  
        this.loadingArtist = false;  
    });  
}  
  
getTopTracks( id: string ){  
  
    this.spotify.getTopTracks(id).subscribe( topTracks => {  
        console.log(topTracks);  
        this.topTracks = topTracks;  
    });  
}  
  
.  
.  
  
import { ActivatedRoute } from '@angular/router';  
import { SpotifyService } from '../spotify.service';
```



```
PS C:\Users\gsp_g\OneDrive\Documentos\0TecnologiasWebEneJun2022\angular2022\spoty\spotify> ng g c picture
CREATE src/app/picture/picture.component.html (22 bytes)
CREATE src/app/picture/picture.component.spec.ts (633 bytes)
CREATE src/app/picture/picture.component.ts (279 bytes)
CREATE src/app/picture/picture.component.css (0 bytes)
UPDATE src/app/app.module.ts (1098 bytes)
```

### ***Dentro de la clase del componente PictureComponent.ts***

```
@Input() width = '';
@Input() paddingTop = '';
@Input() maxWidth = '';
@Input() minWidth = '';
@Input() src = '';
@Input() alt = 'alt';

constructor() { }

ngOnInit(): void {
}

get imageConfig(){
  return {
    width:this.width,
    'padding-top':this.paddingTop,
    'max-width':this.maxWidth,
    'min-width':this.minWidth
  }
}

.
.

import { Component, Input, OnInit } from '@angular/core';
```

### ***En el picture.component.html***

```
<div [ngStyle]="imageConfig" style="position: relative;background-color:
lightgray;">
  <img style="position: absolute;width: 100%;height: 100%;top: 0;object-
fit: cover;" [src]="src" [alt]="alt">
</div>
```

### tarjetas.component.ts

Agregar el siguiente método ...

```
showCard(item:any){  
    return item && item.images && item.images.length > 0  
}
```

### tarjetas.component.html

```
<div class="card-columns m-5 animated fadeIn">  
    <ng-container *ngFor="let item of items">  
        <div (click)="verArtista( item )" class="card  
puntero" *ngIf="showCard(item)">  
  
            <app-picture width="100%" paddingTop="66%"  
[src]="item.images[0].url"></app-picture>  
  
            <div class="card-body">  
                <h5 class="card-title">Artista: {{item.name}} </h5>  
                <p class="card-text">  
                    <span *ngFor="let artista of item.artists" class="badge  
badge-primary"> {{artista.name}} </span>  
                </p>  
            </div>  
        </div>  
    </ng-container>  
  
</div>
```

Crear un pipe para seguridad del audio

```
PS C:\Users\gsp_g\OneDrive\Documentos\0TecnologiasWebEneJun2022\angular2022\spoty\spotify> ng g pipe  
domseguro  
CREATE src/app/domseguro.pipe.spec.ts (199 bytes)  
CREATE src/app/domseguro.pipe.ts (223 bytes)  
UPDATE src/app/app.module.ts (1012 bytes)
```

```
app > TS domseguro.pipe.ts > ...
```

```
import { Pipe, PipeTransform } from '@angular/core';
import { DomSanitizer } from '@angular/platform-browser';

@Pipe({
  name: 'domseguro'
})
export class DomseguroPipe implements PipeTransform {

  constructor(private domSanitizer: DomSanitizer){}

  transform(value: unknown, url: string): unknown {
    console.log('ENTRE AL PIPE');
    return this.domSanitizer.bypassSecurityTrustResourceUrl(url + value);
  }
}
```

```
.
.
import { DomSanitizer } from '@angular/platform-browser';
```

### *artista.component.html*

```
<app-loading *ngIf="loadingArtist"></app-loading>

<div class="row animated fadeIn" *ngIf="!loadingArtist">
  <div class="col-2">
    <img class="img-thumbnail img-circle" [src]="this.artista.images" alt="">
  </div>
  <div class="col">
    <h3> Artista: {{artista.name}} </h3>
    <p>
      <a [href]="artista.external_urls.spotify" class="text-warning" target="_blank"> Ir a la
pagina del artista </a>
    </p>
  </div>
</div>

<div class="row m-5">
  <div class="col">
    <table class="table">
      <thead>
        <tr class="text-black">
          <th>Foto</th>
          <th>Album</th>
          <th>Cancion</th>
          <th>Vista Previa</th>
        </tr>
      </thead>
      <tbody class="text-black">
        <tr *ngFor="let track of topTracks">
          <td>
            <app-picture maxWidth="100px" minWidth="100px" width="100%"
paddingTop="100%" [src]="track.album.images[0].url"></app-picture>
          </td>
          <td> {{ track.album.name }} </td>
          <td> {{ track.name }} </td>
          <td>
            <iframe [src]="track.id | domseguro:'https://open.spotify.com/embed/track/'"
width="300" height="80" frameborder="0" allowtransparency="true" allow="encrypted-media"></iframe>
          </td>
        </tr>
      </tbody>
    </table>
  </div>
</div>
```