

# Gaelon Stats Reporte



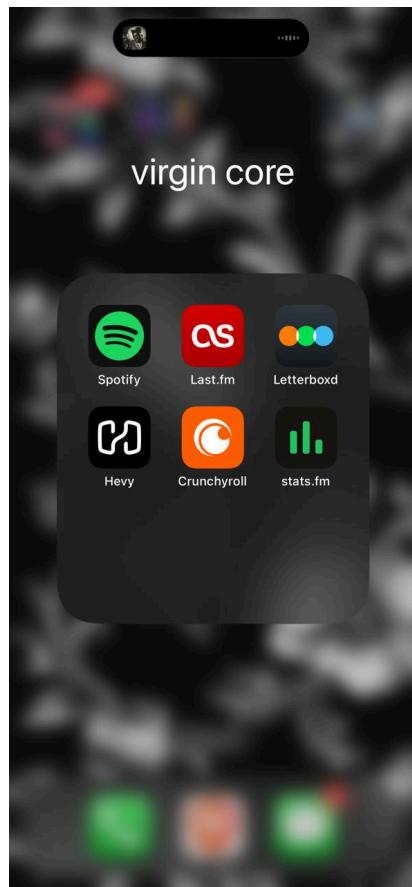
2024B - Programación para Internet

26/11/2024

222791214 | Navarrete Plancarte José Gael

## Definición del proyecto Gaelon Stats

Yo decidí ir por la otra versión del proyecto la cuál era, hacer una página web de lo que quisiéramos, pero que la usemos constantemente. Para esto me surgieron varias ideas en base a mis gustos, tenía planes de hacer algo ya sea sobre los videojuegos, el anime/manga o la música, que son las cosas que más me gustan. Pero sobre todas esas cosas la que más resalta es la música, para mi la música es espectacular y grandiosa. Como podemos ver, estas son algunas de las aplicaciones que tengo instaladas en mi celular sobre la música:



Tanto last.fm como stats.fm son aplicaciones las cuales nos permiten ver las estadísticas que tenemos en Spotify, tales como el artista favorito, la canción favorita, géneros y ese tipo de cosas. Pero lo malo que tienen, es que algunas cosas requieren pagos, y realmente la API de Spotify es totalmente gratis, solo tienen esas opciones para tener ganancia. Por lo que decidí directamente hacer mi propia versión de estas aplicaciones y hacerla de manera web. Este proyecto considero que tiene mucho alcance, ya que si logro acoplarlo, publicarlo y meterle unos anuncios no tan llamativos que afecten a la experiencia del usuario, sería un gran proyecto el cuál hasta me podría generar ingresos.

## Herramientas utilizadas:

Para este proyecto utilice React con TailwindCSS para el desarrollo del frontend, también está conectado directamente a mi API de Spotify, es bastante fácil conseguir una, el único requisito es tener una cuenta premium. Para esto solo me metí a Spotify For Developers y pedí crear una aplicación.

Uno de los problemas que tengo por ahora, es que la aplicación sigue en developer mode, por lo que hay una whitelist de las personas que pueden usar mi página web, que por cierto, esta hosteada en Netlify, agarra directamente mi repo de GitHub y lo hostea.

Cómo podemos ver, esta es mi whitelist de las personas que me han ayudado a probarla:

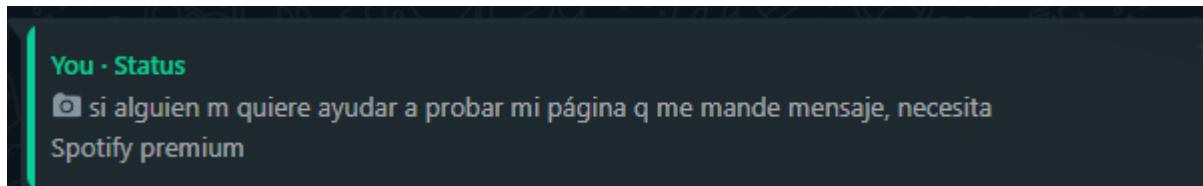
The screenshot shows the 'User Management' section of the Spotify Developer Platform. At the top, there's a header with 'Dashboard > gaelon stats > User Management'. Below the header, there are three tabs: 'Basic Information', 'User Management' (which is selected), and 'Extension Requests'. A sub-header reads: 'Manage the Spotify users you want to allow access to your app in development. The users you choose to configure for this app can authenticate with your application via extension request.' There is a text input field labeled 'Full Name' with the placeholder '17/25 added'. Below the input field is a table listing 17 users, each with a number and their name:

#	Name
1	Víctor Fausto
2	Oscar Castañeda
3	Daniel Alejandro Mendoza
4	Evelyn Castrojón
5	Sebastián Puga
6	Víctor Espinoza
7	José Alfonso Barbosa
8	Grecia Andrade
9	Saúl Pavón
10	Sonia Plancarte
11	David Ponce Santos
12	Leonardo Peletto
13	Maggie Lluna
14	Daniel Bello Mares
15	Diego Maldonado
16	Aldo Andrade
17	Lilicove

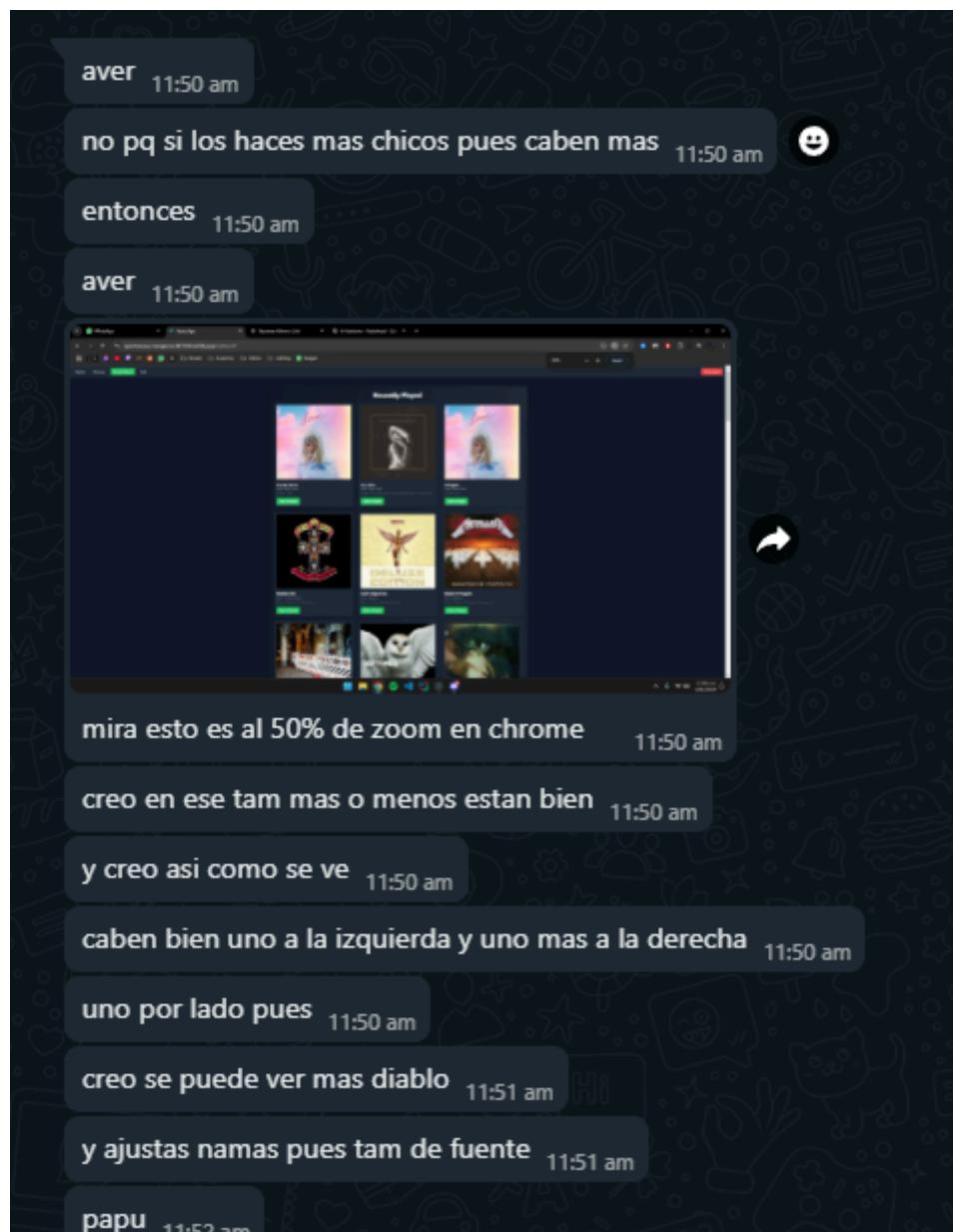
(Pongo la foto de esta manera ya que para agregarlos, necesito su correo de spotify y no quiero filtrar esa información)

Para la versión final, me gustaría poder publicar ya mi aplicación para que todos los usuarios de spotify premium puedan acceder a ella sin ningún problema.

Hice varias pruebas en su momento, con todos los usuarios a los que les pedí ayuda, en mis redes sociales hice una publicación para que el que quisiera ayudarme lo pudiera hacer, y de ahí recopile información. Subí el siguiente estado:



Ya de aquí personas me fueron ayudando y diciendo que cambios podría realizar dentro de la página, como este ejemplo:

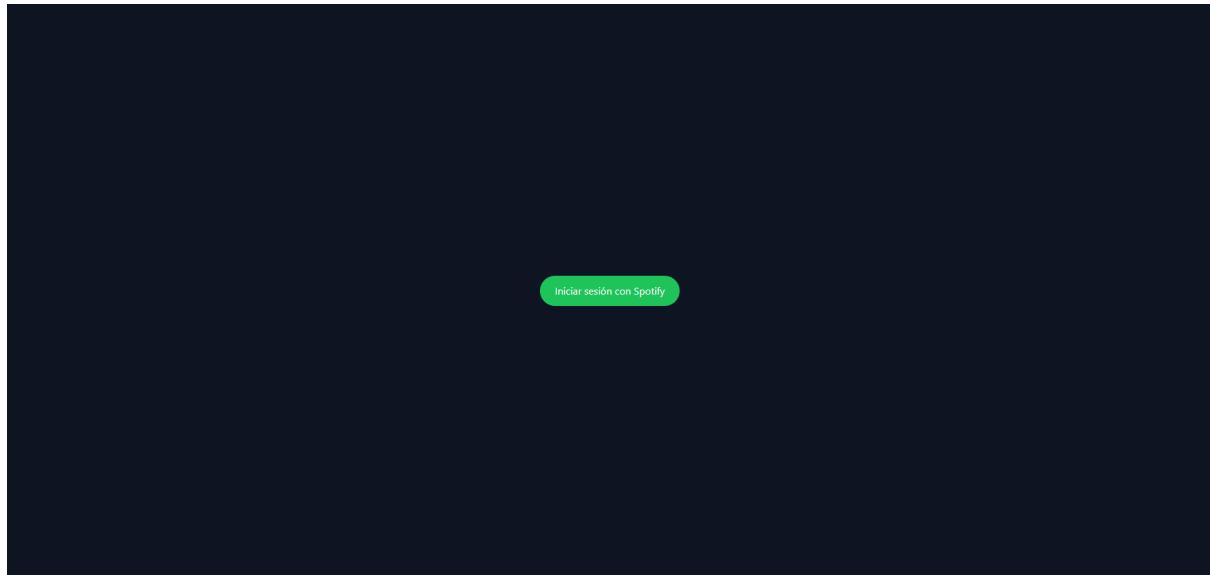


Y esto fue algo que cambió para la segunda versión, ahora están más pequeñas las

preview y se alcanzan a ver 5 por fila.

## Capturas de pantalla de la página web:

Versión computadora:



A screenshot of the Spotify Dashboard. The top navigation bar includes tabs for "Playlists" (which is selected), "Discovery", "Recently Played", and "Stats", along with a "Cerrar sesión" (Logout) button. On the left, a sidebar titled "Your Playlists" lists several playlists with their names and track counts: "gael" (456 tracks), "Gael is cool" (477 tracks), "Gael is cool 2.0" (452 tracks), "gael is cool 3.0" (73 tracks), "test api spoti" (0 tracks), "who tf is gael?" (1018 tracks), "deep in my heart, i hate you gaara" (165 tracks), "Ethereal ass songs" (214 tracks), "Kill Yourself (1-5)" (5 tracks), "i miss u" (20 tracks), "gym" (50 tracks), and "Defenestr-Mix" (50 tracks). On the right side, a grid of song cards is displayed, each featuring a small album thumbnail, the song title, the artist name, and a "Play Preview" button. The first few cards include "R U Mine?" by Arctic Monkeys, "Teddy Picker" by Arctic Monkeys, "Crying Lightning" by Arctic Monkeys, "Miss Murder" by AFI, "Bohemian Rhapsody" by Queen, "Before I Forget" by Slipknot, and "Can't Stop" by Red Hot Chili Peppers.

**Spotify Dashboard**

Playlists Discovery Recently Played Stats Cerrar sesión

## Descubre Nueva Música

Estado de ánimo: Feliz

**Flies**  
Artista: Aubrey Key  
Álbum: Cow

**Surfer Rosa**  
Artista: The Obsessives  
Álbum: The Obsessives

**Ritalin**  
Artista: HUNNY  
Álbum: Yes. Yes. Yes. Yes.

**Fashion Kill**  
Artista: A\$AP Rocky  
Álbum: LONG.LIVE.A\$AP (Delux...)

**Beef**  
Artista: Ethereal  
Álbum: Beef

**I Smoked Away My Brain ...**  
Artista: ASAP Rocky  
Álbum: I Smoked Away My Brai...

**Need 2**  
Artista: Pinegrove  
Álbum: Everything So Far

**It Almost Worked**  
Artista: TV Girl

**Borderline**  
Artista: Tame Impala

**New Divide**  
Artista: Linkin Park

**Dumb**  
Artista: Nirvana

**Plug in Baby**  
Artista: Muse

**Spotify Dashboard**

Playlists Discovery **Recently Played** Stats Cerrar sesión

## Recently Played

**Nangs**  
Artista: Tame Impala  
Álbum: Currents

**worldstar money (interlude)**  
Artista: Jorji  
Álbum: In Tongues (Deluxe)

**Like Him (feat. Lola Young)**  
Artista: Tyler, The Creator  
Álbum: CHROMAKOPIA

**I Smoked Away My Brain ...**  
Artista: ASAP Rocky  
Álbum: I Smoked Away My Brai...

**Need 2**  
Artista: Pinegrove  
Álbum: Everything So Far

**It Almost Worked**  
Artista: TV Girl

**Borderline**  
Artista: Tame Impala

**New Divide**  
Artista: Linkin Park

**Dumb**  
Artista: Nirvana

**Plug in Baby**  
Artista: Muse

**Spotify Dashboard**

Playlists Discovery Recently Played **Stats** Cerrar sesión

Últimas 4 semanas Últimos 6 meses Todo el año

**Top Canciones**

	Alternative Outro LUCKI	<b>Play</b>
	Like Him (feat. Lola Young) Tyler, The Creator	<b>Play</b>
	Sorry Alex G	<b>Play</b>
	The Shallows The Rare Occasions	<b>Play</b>
	Stone Born Without Bones	<b>Play</b>
	Thought I Was Dead (feat. Dead.) gael	<b>Play</b>

**Top Artistas**

	Tyler, The Creator
	Deftones
	Joji
	\$uicideboy\$
	Alex G
	Interpol

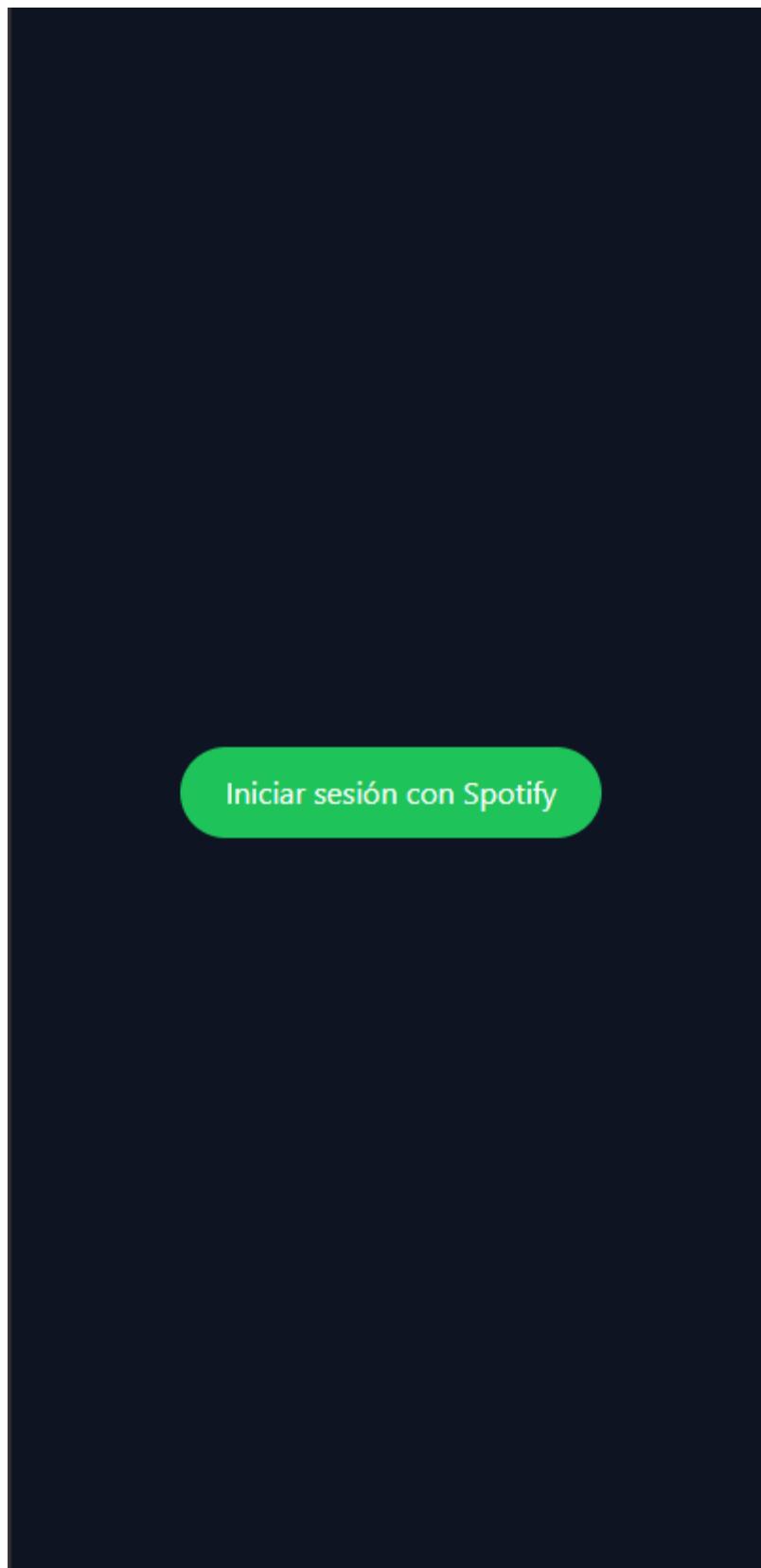
**Top Álbumes**

	CHROMAKOPIA Tyler, The Creator
	Alternative Trap LUCKI
	DSU Alex G
	Into The Shallows The Rare Occasions
	Baby Born Without Bones
	Bedside Kites Bedside Kites

**Top Playlists**

	gael
	Gael is cool
	Gael is cool 2.0
	gael 3.0
	test api spoti
	who tf is gael?

Versión celular:



The image shows a Spotify dashboard interface. At the top left is a line graph icon. In the center is the title "Spotify Dashboard". At the top right is a three-line menu icon. Below the header is a green button labeled "Playlists". Underneath the "Playlists" button are several other options: "Discovery", "Recently Played", "Stats", and "Cerrar sesión". The main content area is titled "Your Playlists" in large white text. It lists five playlists with their names and track counts:

- gael: 456 tracks
- Gael is cool: 477 tracks
- Gael is cool** 2.0: 452 tracks
- gael is cool 3.0: 73 tracks
- test api spoti: 0 tracks



**gael**

R U



**Mine?**

Arctic  
Monkeys

Play  
Preview

**Teddy**



**Picker**

Arctic  
Monkeys

Play  
Preview

**Crying**



**Lightning**

Arctic  
Monkeys

Play  
Preview

**Miss**



**Murder**

AFI

Play  
Preview

**Bohemian**



**Rhapsody**

Queen

Play  
Preview

Play

Preview



Spotify Dashboard



## Descubre Nueva Música

Estado de ánimo:

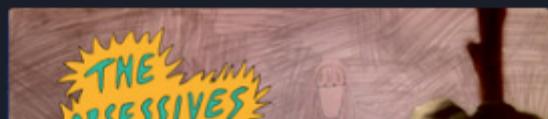
Feliz



**Something About You**

Artista: Eyedress

Álbum: Mulholland Drive





Spotify Dashboard



## Recently Played

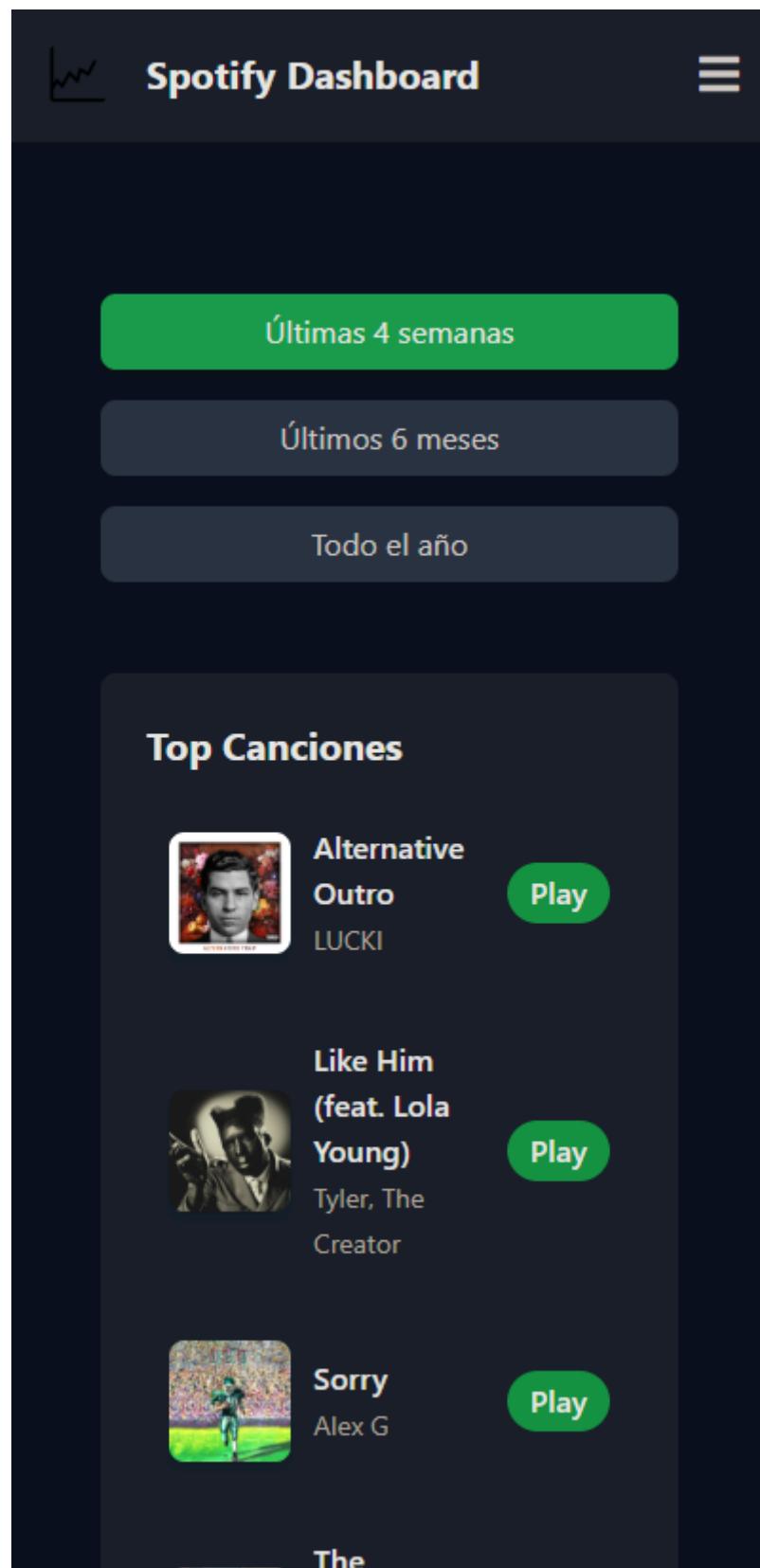


### Not Allowed

Artist: TV Girl

Album: Who Really Cares





Y para finalizar con este documento, me gustaría mostrar el archivo más importante del código, el cuál es el archivo que maneja las llamadas a la API de Spotify:

```
const CLIENT_ID = "YOUR_CLIENT";
const REDIRECT_URI = "https://spontaneous-kangaroo-b81304.netlify.app/callback";
const SCOPES = [
  "user-read-private",
  "user-read-recently-played",
  "user-read-email",
  "user-top-read",
  "playlist-read-private",
  "playlist-modify-public",
  "playlist-modify-private",
  "user-library-read",
  "user-library-modify",
];

```

(Borre el client id para no hacerlo público)

Los scopes (alcances) en el contexto de la autenticación de Spotify son permisos específicos que una aplicación solicita al usuario para poder acceder a ciertos datos o realizar ciertas acciones en su cuenta de Spotify. Al utilizar el flujo de autorización de Spotify, los scopes definen lo que la aplicación puede y no puede hacer en nombre del usuario.

Por ejemplo, los scopes en mi código son:

- **user-read-private:** Permite acceder a la información privada del usuario, como su nombre de usuario y otros detalles básicos.
- **user-read-recently-played:** Permite acceder a las canciones que el usuario ha reproducido recientemente.
- **user-read-email:** Permite acceder a la dirección de correo electrónico del usuario.
- **user-top-read:** Permite acceder a las canciones o artistas más escuchados del usuario.
- **playlist-read-private:** Permite acceder a las listas de reproducción privadas del usuario.
- **playlist-modify-public:** Permite modificar las listas de reproducción públicas del usuario.
- **playlist-modify-private:** Permite modificar las listas de reproducción privadas del usuario.
- **user-library-read:** Permite acceder a las canciones guardadas en la biblioteca del usuario.
- **user-library-modify:** Permite agregar o eliminar canciones de la biblioteca

del usuario.

```
export const getAuthUrl = () => {
  return (
    `https://accounts.spotify.com/authorize?` +
    `client_id=${CLIENT_ID}&` +
    `response_type=token&` + // Use 'token' to get access token directly
    `redirect_uri=${encodeURIComponent(REDIRECT_URI)}&` +
    `scope=${encodeURIComponent(SCOPES.join(" "))}` +
  );
};
```

Esta función genera una URL de autorización de Spotify. La URL contiene los parámetros necesarios para redirigir al usuario a la página de autorización de Spotify, donde se otorgará acceso a la aplicación. Utiliza los siguientes parámetros:

- **client\_id**: Identificador de la aplicación registrada en Spotify.
- **response\_type**: Tipo de respuesta, que en este caso es token para obtener directamente un token de acceso.
- **redirect\_uri**: URL a la que Spotify redirigirá después de la autorización.
- **scope**: Los permisos que la aplicación solicita al usuario, como leer la biblioteca, modificar listas de reproducción, etc.

```
export const fetchFromSpotify = async (endpoint) => {
  const token = localStorage.getItem("spotifyAccessToken");

  if (!token) {
    throw new Error("Access token is missing");
  }

  const response = await fetch(`https://api.spotify.com/v1${endpoint}`, {
    headers: {
      Authorization: `Bearer ${token}`,
    },
  });

  if (!response.ok) {
    throw new Error("Failed to fetch data from Spotify");
  }

  return response.json();
};
```

Esta función se encarga de hacer peticiones a la API de Spotify utilizando el token de acceso almacenado en localStorage. Si el token no está presente, lanza un error.

Si el token está disponible, realiza una solicitud HTTP con el token de autorización en el encabezado. Esta función permite consultar datos de la API de Spotify, como el perfil del usuario o su lista de reproducción, entre otros.

Y eso sería todo por ahora, así es como luce el proyecto en VSCode:

