

## Resumen

Crearemos una aplicación que utilice la API de Spotify y la utilizaremos para crear playlists, añadirles canciones etc. (el uso básico de Spotify). También utilizaremos el API de Genius para manejar información e imágenes de los grupos o cantantes que tengan dichas redes sociales.

## Listado exhaustivo de las funcionalidades a implementar

- Iniciar sesión.
- Cerrar sesión.
- Crear una playlist.
- Eliminar una playlist.
- Eliminar una canción de una playlist.
- Editar una playlist.
- Añadir una playlist creada por otro usuario.
- Ver el contenido de las playlists.
- Mostrar la canción con mas visitas de la playlist.
- Mostrar imagen del perfil del artista ó grupo.
- Mostrar gráficas de las estadísticas de la playlist ó artista/grupo (popularidad).

## Bocetos de las pantallas de la aplicación

En esta primera figura podemos ver la pantalla de inicio de sesión, donde se nos pedirá el nombre de usuario o correo electrónico y la contraseña para poder acceder a nuestra cuenta.



The image shows a web browser window with the title 'SpotifyList'. The address bar contains 'https://www.spotifylist.es'. The main content area has the heading 'Inicio de sesión'. Below the heading, there are two input fields: the first is labeled 'USUARIO o CORREO' and the second is labeled 'CONTRASEÑA'. Below these fields is a button labeled 'Acceder'. The browser window has a standard toolbar with back, forward, and search icons.

Figura 1: Inicio de sesión.

En la figura 2 tenemos la pantalla principal de la aplicación, pantalla a la que accederíamos una vez completado el login. En esta pantalla podemos seleccionar las playlists que queramos y eliminarlas o compartirlas pulsando los respectivos botones. También podemos crear una nueva playlist en el botón de crear playlist.

A mayores podemos buscar otras playlists en la barra de búsqueda. Por último podemos salir de la aplicación en el botón de la X donde ponde cerrar sesión.



Figura 2: Pantalla de inicio.

La figura número 3 nos muestra la pantalla que salta una vez que hacemos click en el botón de cerrar sesión. Aparecerá un cuadro donde confirmaremos si queremos o no cerrar sesión.

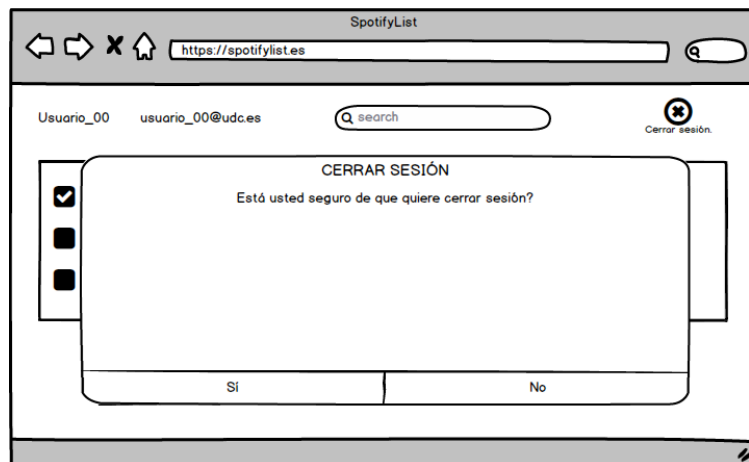


Figura 3: Cerrar sesión

En esta figura número 4 vemos la pantalla a la que accedemos al pulsar el botón de crear playlist, debemos ingresar el título de la playlist y la privacidad de esta.

Crear Playlist

Título de tu playlist

☐ Pública ☐ Privada

Figura 4: Cerrar sesión

Si hacemos click encima de una de las playlist accederemos a la pantalla de la figura 5 donde podremos modificar la propia playlist y ver a mayores información acerca de la playlist. Podemos seleccionar un artista para ver sus datos y compartir dicha playlist. Por defecto aparecerán los temas por orden descendente en cuanto a número de reproducciones.

PlayList\_00  [Pantalla principal](#)

CANCION	VISITAS	ARTISTA	SELECCIONAR
Primavera	40345	ANTONIO	<input checked="" type="checkbox"/>
Verano	38545	VELAZQUEZ	<input checked="" type="checkbox"/>
Invierno	43421	JUAN	<input type="checkbox"/>
Otoño	12324	ANA	<input checked="" type="checkbox"/>
Solsticio	33445	SORIA	<input type="checkbox"/>

Figura 5: Vista canciones playlist

Si hacemos click en mostrar estadísticas de la playlist obtendremos una nueva pantalla con una gráfica donde se mostrará el número de visitas por canción, figura 6.



Figura 6: Estadísticas playlist

Búsqueda Playlist

Q Título de la playlist

Regg.  
Reggaeton 2020  
Clásicos reggaeton  
Novedades reggaeton

Cancelar Añadir

Figura 7: Búsqueda de Playlist

Cuando hacemos click en el search de la pantalla de inicio, nos aparecerá una pantalla en la cual podremos buscar una lista existente en Spotify para añadirla a las nuestras. Figura 7.



Figura 8: Búsqueda de Canción

Cuando pulsamos el botón de añadir canción en la vista de una playlist, aparecerá una pantalla en la que buscaremos una canción existente en Spotify para añadirla a nuestra playlist. Figura 8.

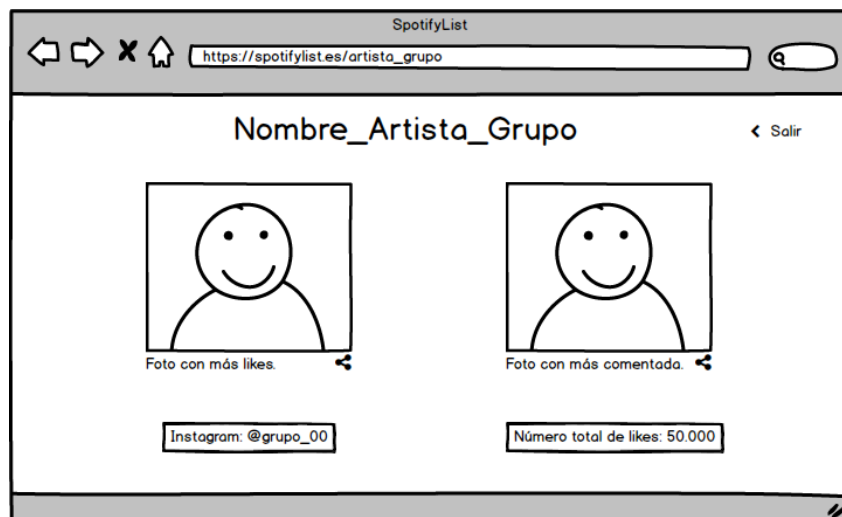


Figura 9: Instagram del artista y estadísticas

Si en la figura 5 hacemos click en el nombre de un artista, accederemos a la figura 9 donde podremos consultar cierta información del artista. Dicha información es: su nombre de perfil en instagram, su foto y también podremos ver los diferentes nombres del artista ó grupo en Facebook, Instagram y twitter.

## Flujo de datos de aplicación

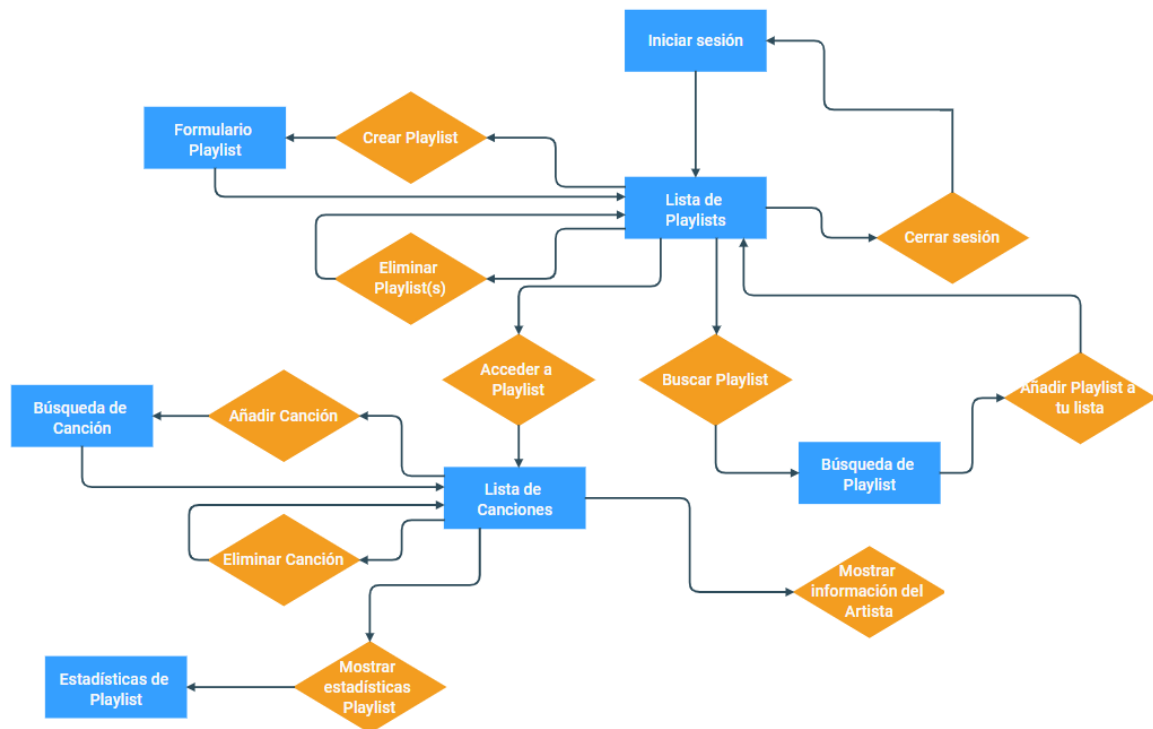


Figura 10: Flujo de datos de la aplicación

## APIs utilizadas

- API SPOTIFY.
- API GENIUS.

### Información a extraer del API SPOTIFY

Obtendremos información de los artistas, las playlists, el número de reproducciones, las canciones que hay en cada playlist etc... En resumen, sacaremos información acerca de las playlists de los usuarios de Spotify.

### Información a extraer del API GENIUS

Obtendremos los perfiles de Instagram de los artistas, también su nombre en twitter y facebook en caso de tener dichas redes sociales. También incluiremos una foto del artista/grupo.

## Uso de Pandas dentro de la práctica

Utilizaremos Pandas para tratar la información obtenida de las APIs, para mostrar valores que calcularemos posteriormente (máximos, mínimos etc.), también usaremos Pandas para

mostrar toda la información correspondiente a las playlists y los artistas (nombre, número de reproducciones, álbumes, número de seguidores etc.)

Se destaca su uso para la creación de un conjunto de datos con los artistas más populares de cada playlist, donde creamos un dataframe en base a la popularidad de los 5 artistas más escuchados.

## **Funcionalidades a implementar en la primera iteración de la práctica**

Para esta primera iteración iniciaremos y cerraremos sesión, crearemos una playlist y obtendremos los datos. También obtendremos datos de Instagram como fotos, likes etc. Por último, plotearemos algunas gráficas comparativas con el número de likes o el número de seguidores.

## **Uso de librerías externas (opcional)**

Para crear las gráficas usaremos Matplotlib que nos imprimirá las canciones más populares de cada playlist recolectadas en un dataframe con pandas.

## **Contenedor Docker**

Tal y como se indica hemos creado un contenedor Docker para nuestro proyecto y se ha subido la correspondiente imagen al Docker Hub.

## **Errores conocidos**

### **0.1. Eliminar una canción de una playlist o vaciarla:**

- Solo funcionarán en caso de que la playlist se haya creado por el propio usuario, si es de otro usuario no funcionará ya que al no ser los propietarios de dicha playlist obtendremos un error de autorización por parte de Spotify.

### **0.2. Editar una playlist:**

- Sucede lo mismo que en el caso anterior, tanto como para borrar como para añadir nuevos items a una playlist.

### **0.3. Ver el contenido de las playlists:**

- Podremos ver todas las playlist creadas por nosotros (tanto públicas como privadas), pero no podremos ver las playlists privadas creadas por otros usuarios (únicamente podremos verlas si son públicas).

## **Errores conocidos en los TEST.**

### **0.4. Peticiones API Spotify, Genius:**

- Se recibirá un código de error HTTP 401 si las credenciales de autorización son incorrectas, pero en la salida posterior a la ejecución de los tests obtendremos un mensaje que nos indica que faltan headers aunque ese no sea el motivo de error.

## **Horas de trabajo:**

- En reuniones grupales a través de teams hemos realizado aproximadamente 40 horas de trabajo en grupo. A mayores cada uno de nosotros ha trabajado de forma autónoma unas 6 horas extra semanales.