

trackiD

Proyecto transversal asignaturas Programación,
Entornos de Desarrollo y Bases de Datos

FP Grado Superior Desarrollo Aplicaciones Web - Mult

Ismael Morillas
EEP iGroup
Madrid, 2021

ÍNDICE

INTRODUCCIÓN	2
MANUAL DE USUARIO	2
Inicio de sesión	2
Registro de un nuevo usuario.....	3
Biblioteca de usuario	4
Detalles de pista.....	5
Añadir canción por fichero de texto	6
Añadir canción desde la aplicación	7
Modificar datos de canción.....	8
DIAGRAMA DE CLASES.....	9
DIAGRAMA DE CASOS DE USO.....	10
DIAGRAMA DE SECUENCIA.....	12
DIAGRAMA E/R Y MODELO RELACIONAL (BBDD).....	13
PRUEBAS FUNCIONALES	14
➡ Proyecto en GitHub.....	14
BIBLIOGRAFÍA	15

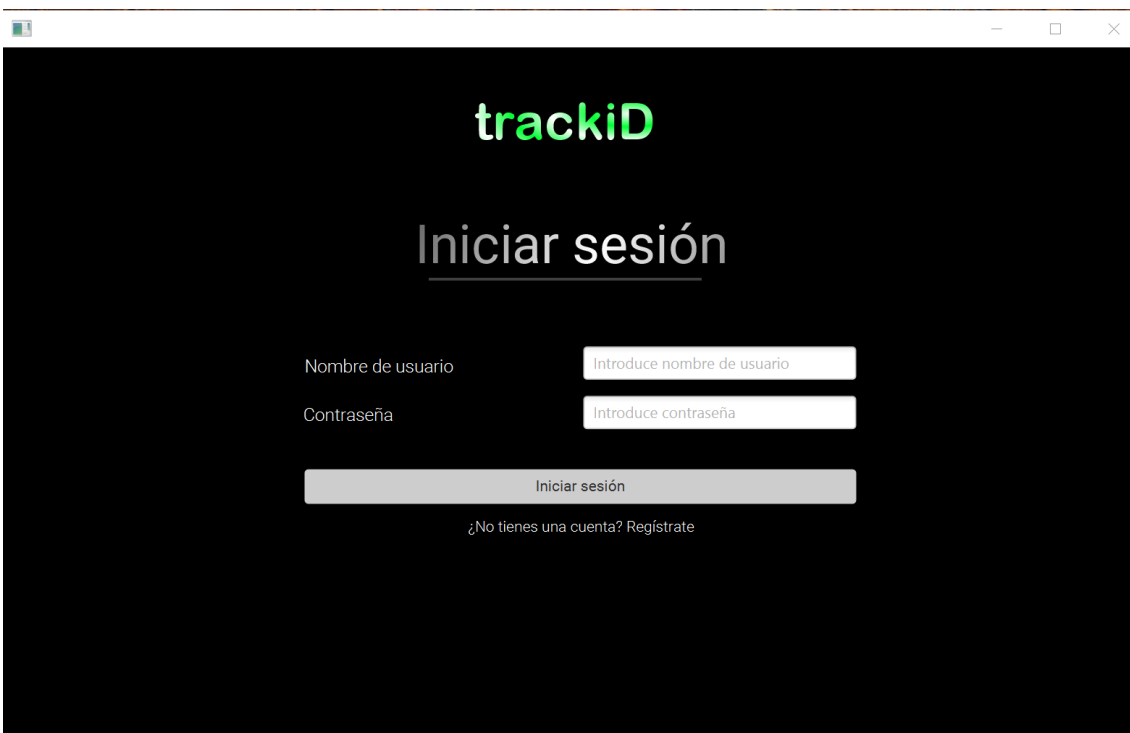
INTRODUCCIÓN

“*trackiD*” es una aplicación de escritorio que recrea el proceso de distribución de música en las diferentes plataformas digitales, ya que, para subir contenido a éstas, es necesaria una plataforma intermediaria.

“*trackiD*” permite al usuario añadir canciones a su perfil, así como modificarlas y borrarlas. También, podrá ver los detalles de sus canciones y reproducirlas.

MANUAL DE USUARIO

Inicio de sesión



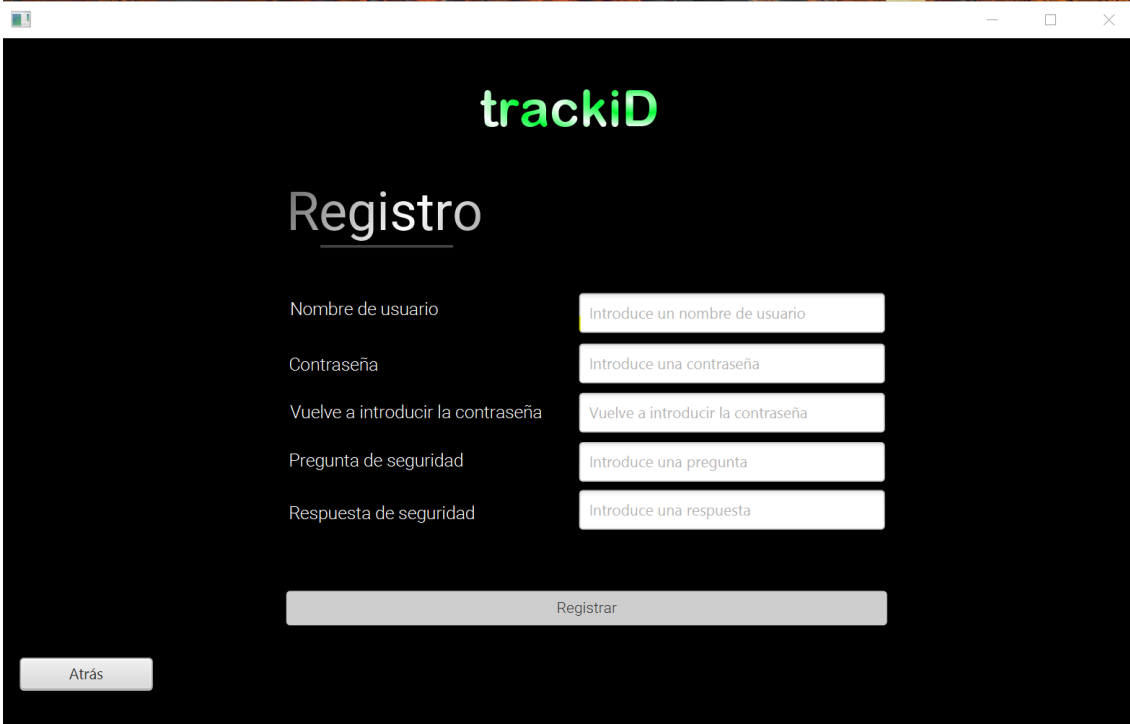
La ventana de inicio de sesión permite a un usuario registrado en la plataforma acceder al contenido de su perfil. En este caso, el usuario que haga inicio de sesión, verá canciones.

El botón con el texto “*Iniciar sesión*” será clave en este proceso, ya que, al pulsarlo, accederemos a nuestro perfil - siempre que los datos introducidos en los campos de texto sean correctos perteneciendo a un usuario ya registrado -.

Los campos de texto *“Introduce nombre de usuario”* e *“Introduce contraseña”* cumplirán la función de contener los datos del usuario que está intentando acceder a la plataforma.

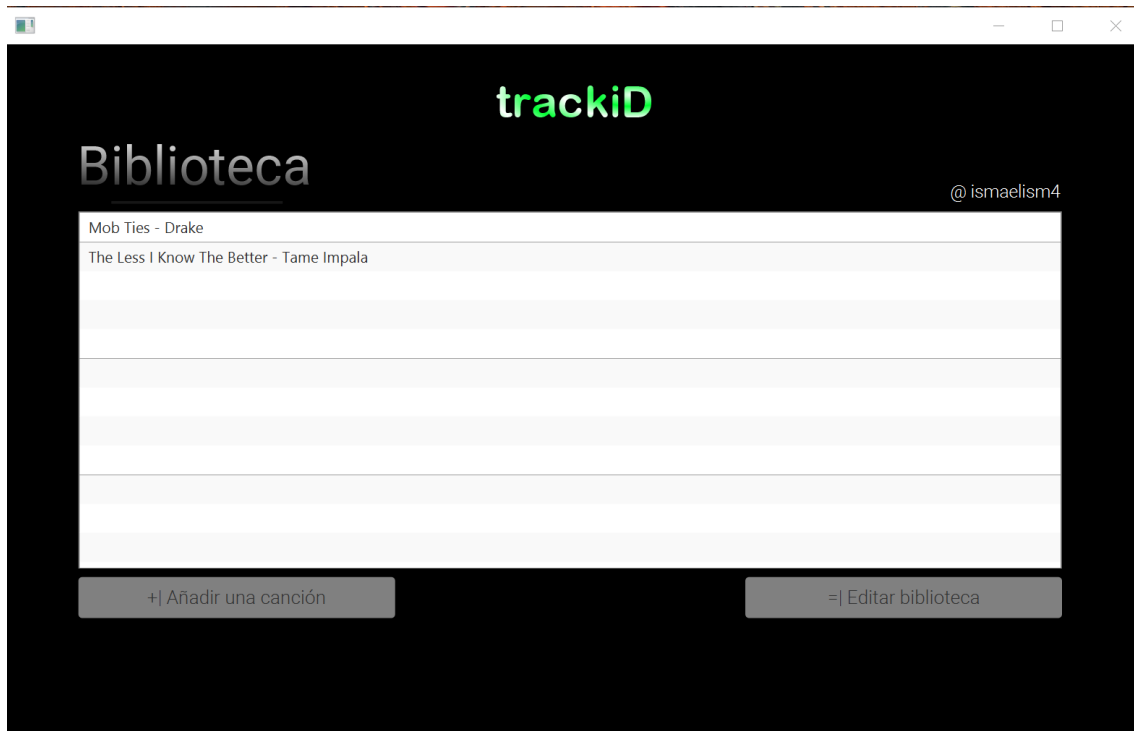
En esta ventana, el usuario tendrá opción de registrarse dándose de alta en la plataforma clicando sobre el texto *“¿No tienes una cuenta? Regístrate”*

Registro de un nuevo usuario

The image shows a web browser window displaying the 'trackiD' registration page. The page has a dark background with the 'trackiD' logo in green at the top. Below the logo, the word 'Registro' is written in large white letters. The registration form consists of five input fields with white text prompts: 'Introduce un nombre de usuario', 'Introduce una contraseña', 'Vuelve a introducir la contraseña', 'Introduce una pregunta', and 'Introduce una respuesta'. Each field is preceded by a label in white: 'Nombre de usuario', 'Contraseña', 'Vuelve a introducir la contraseña', 'Pregunta de seguridad', and 'Respuesta de seguridad'. At the bottom of the form is a wide 'Registrar' button. In the bottom left corner, there is an 'Atrás' button.

La ventana de Registro nos permitirá dar de alta a un nuevo usuario, es decir, una persona no dada de alta en la plataforma.

Ingresando un nombre de usuario en el campo de texto *“Introduce un nombre de usuario”*, una contraseña (introducida dos veces para validar que ambos campos de contraseña son similares) en los campos *“Introduce una contraseña”* y *“Vuelve a introducir la contraseña”*, una pregunta de seguridad en el campo de texto *“Introduce una pregunta”*, una respuesta de seguridad en el campo *“Introduce una respuesta”* y clicando en el botón *“Registrar”*, la acción de dar de alta al nuevo usuario será efectuada con éxito. En caso de que haya algún problema durante el proceso de registro, se lanzará un mensaje de error.

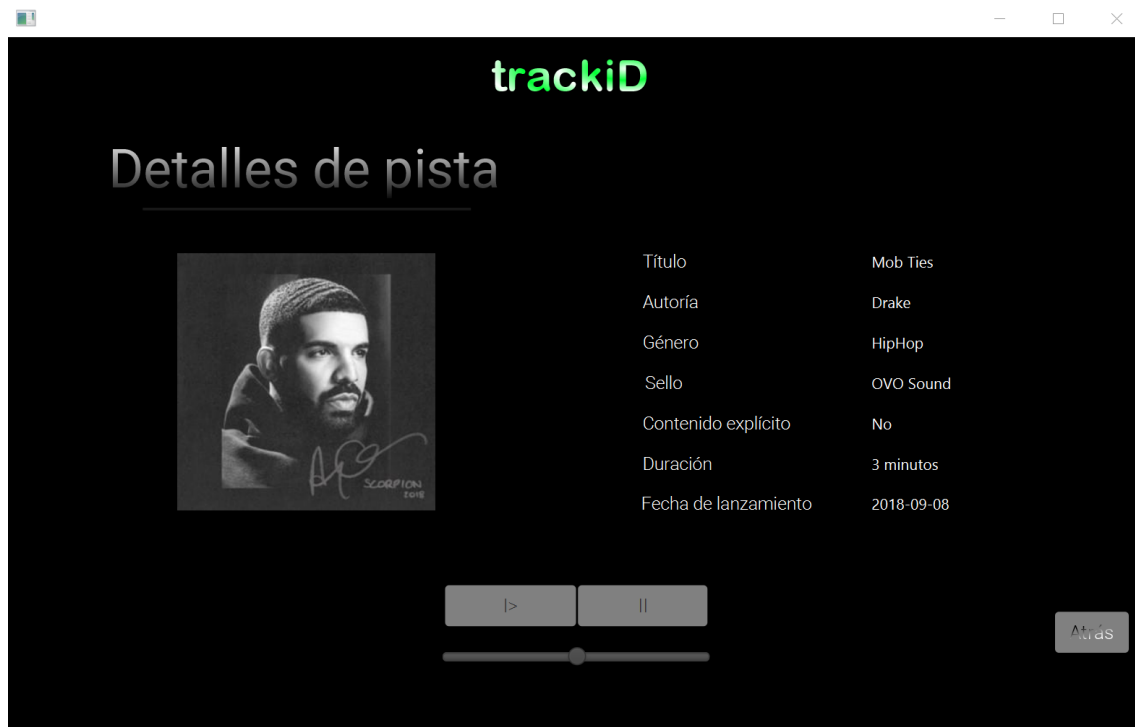
Biblioteca de usuario

La ventana de Biblioteca permitirá al usuario dado de alta en la plataforma ver todas las canciones que haya subido. En la imagen de muestra, podemos observar que el usuario tiene subidas dos canciones.

Clicando en alguna de las canciones de la biblioteca, podremos acceder a los detalles de la determinada pista, teniendo opción de reproducirla.

Clicando en el botón con el texto “Añadir una canción” se accederá a la ventana que permite añadir una canción con todas las características que deseemos para la misma.

Clicando en el botón “Editar biblioteca” se accederá a la ventana que permite la modificación de la canción que el usuario desee, así como su borrado.

Detalles de pista

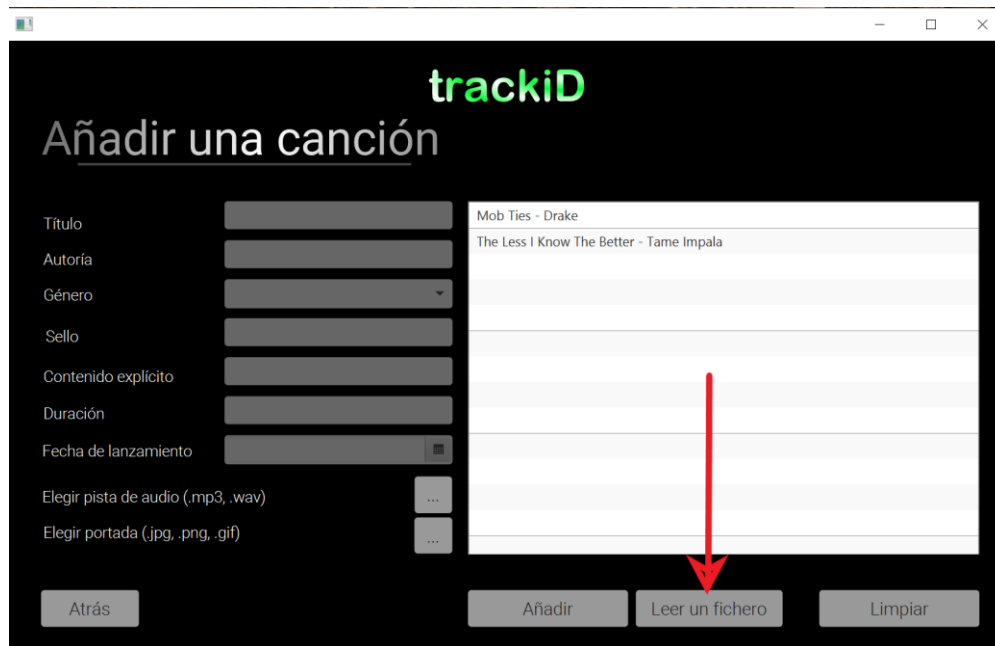
En la ventana Detalles de pista, podremos ver todos los detalles de la canción seleccionada. Pulsando en el botón con símbolo “Play” (|>), la canción empezará a reproducirse. Pulsando en el botón de su derecha “Pause” (||), la canción será pausada.

El usuario tendrá la posibilidad de volver a su Biblioteca clicando en el botón “Atrás”.

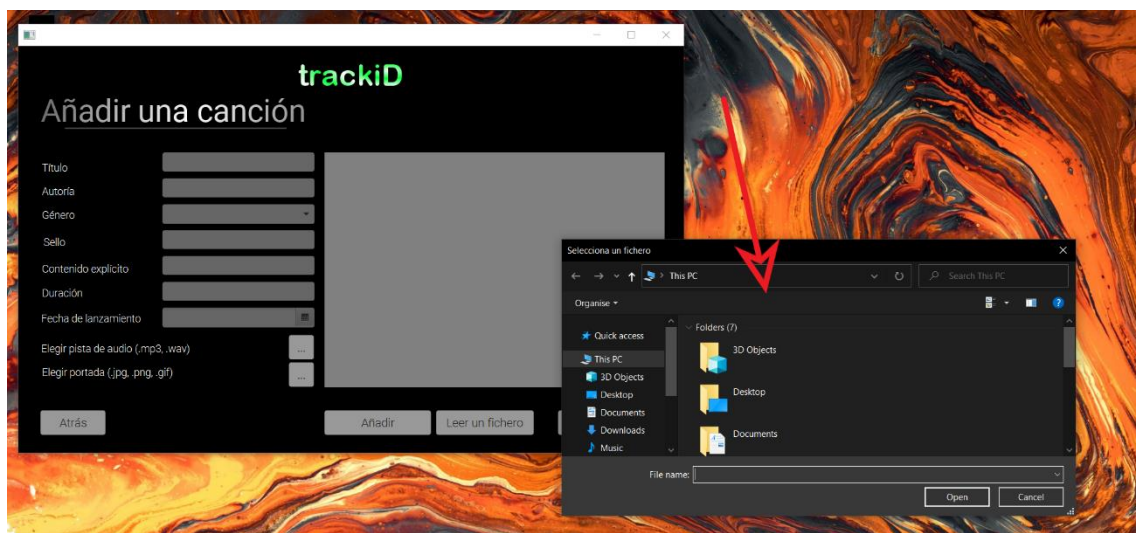
Añadir canción por fichero de texto

Para realizar este proceso, es necesario tener la pista de audio y la portada en una carpeta dentro del proyecto.

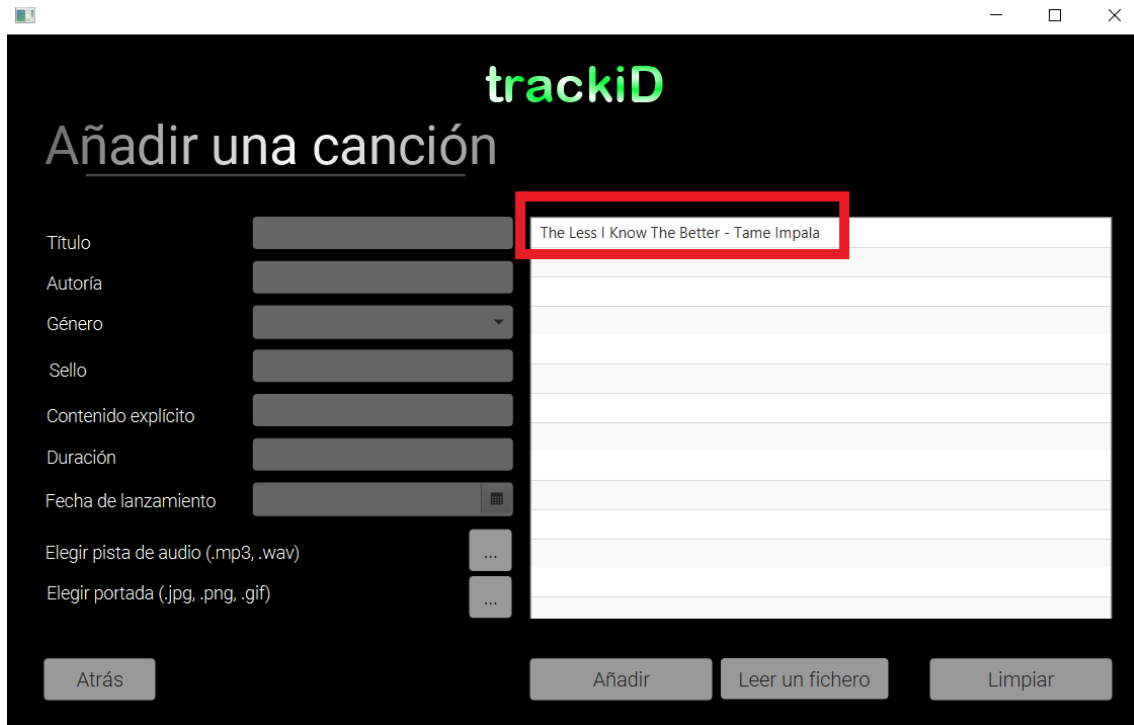
1. Clicar sobre botón “Leer un fichero”



2. Seleccionar fichero de texto con los datos de la canción desde nuestro equipo a través de la ventana que se abre.



3. Una vez seleccionado el fichero, si todo ha salido correctamente, la canción será añadida a la lista de canciones del usuario.



trackiD

Añadir una canción

Título:

Autoría:

Género:

Sello:

Contenido explícito:

Duración:

Fecha de lanzamiento:

Elegir pista de audio (.mp3, .wav):

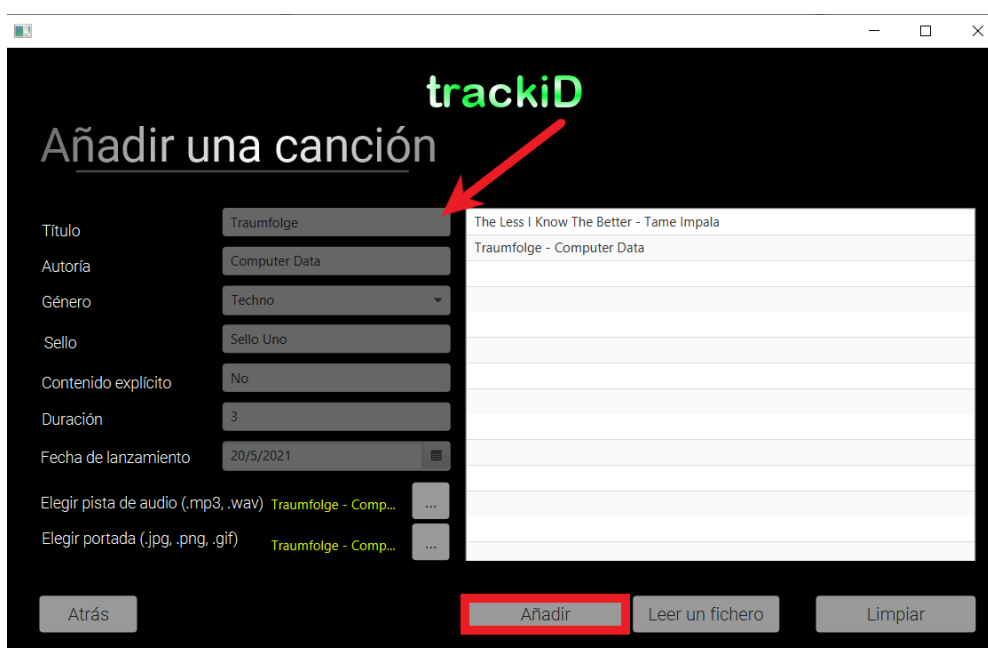
Elegir portada (.jpg, .png, .gif):

The Less I Know The Better - Tame Impala

Atrás Añadir Leer un fichero Limpiar

Añadir canción desde la aplicación

Rellenar los campos de texto, de fecha y de fichero con los datos de la canción y clicar en el botón “Añadir”.



trackiD

Añadir una canción

Título: Traumfolge

Autoría: Computer Data

Género: Techno

Sello: Sello Uno

Contenido explícito: No

Duración: 3

Fecha de lanzamiento: 20/5/2021

Elegir pista de audio (.mp3, .wav): Traumfolge - Comp...

Elegir portada (.jpg, .png, .gif): Traumfolge - Comp...

The Less I Know The Better - Tame Impala

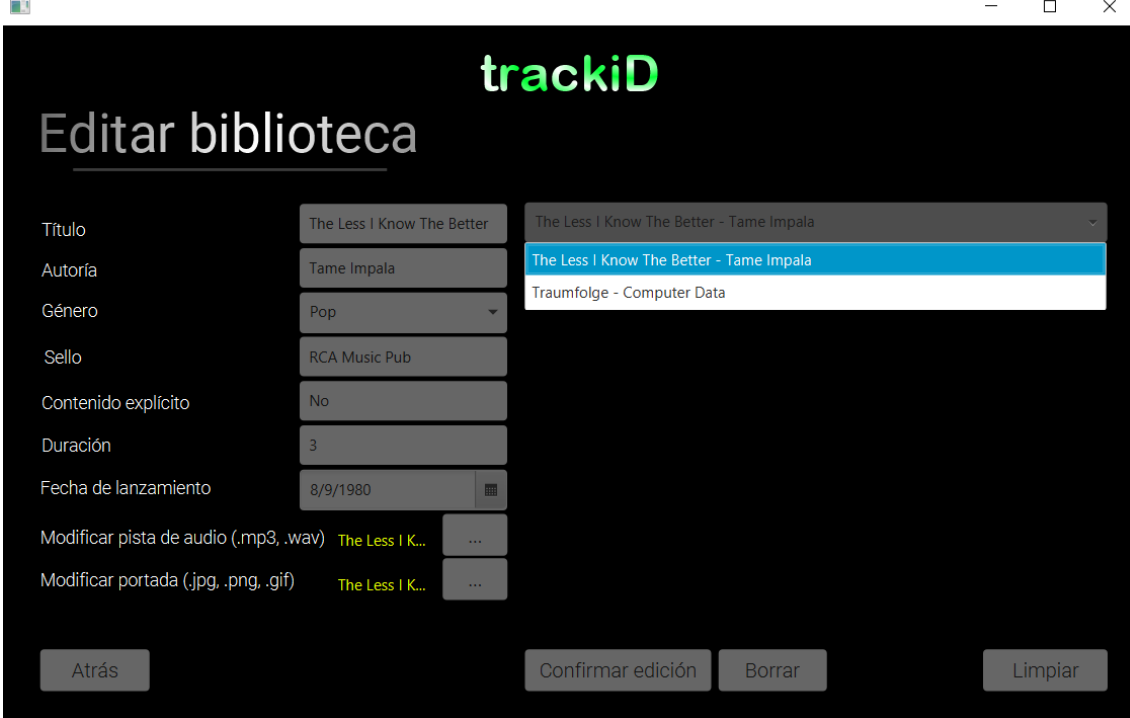
Traumfolge - Computer Data

Atrás Añadir Leer un fichero Limpiar

Modificar datos de canción

La modificación de canción permitirá al usuario editar las canciones de su biblioteca. Para realizar cambios, se deberá seleccionar la canción deseada del desplegable, y modificar alguno de los campos. Además, clicando en el botón “Borrar”, podremos eliminar la canción seleccionada.

Mediante el botón “Limpiar”, podremos desmarcar la canción seleccionada.

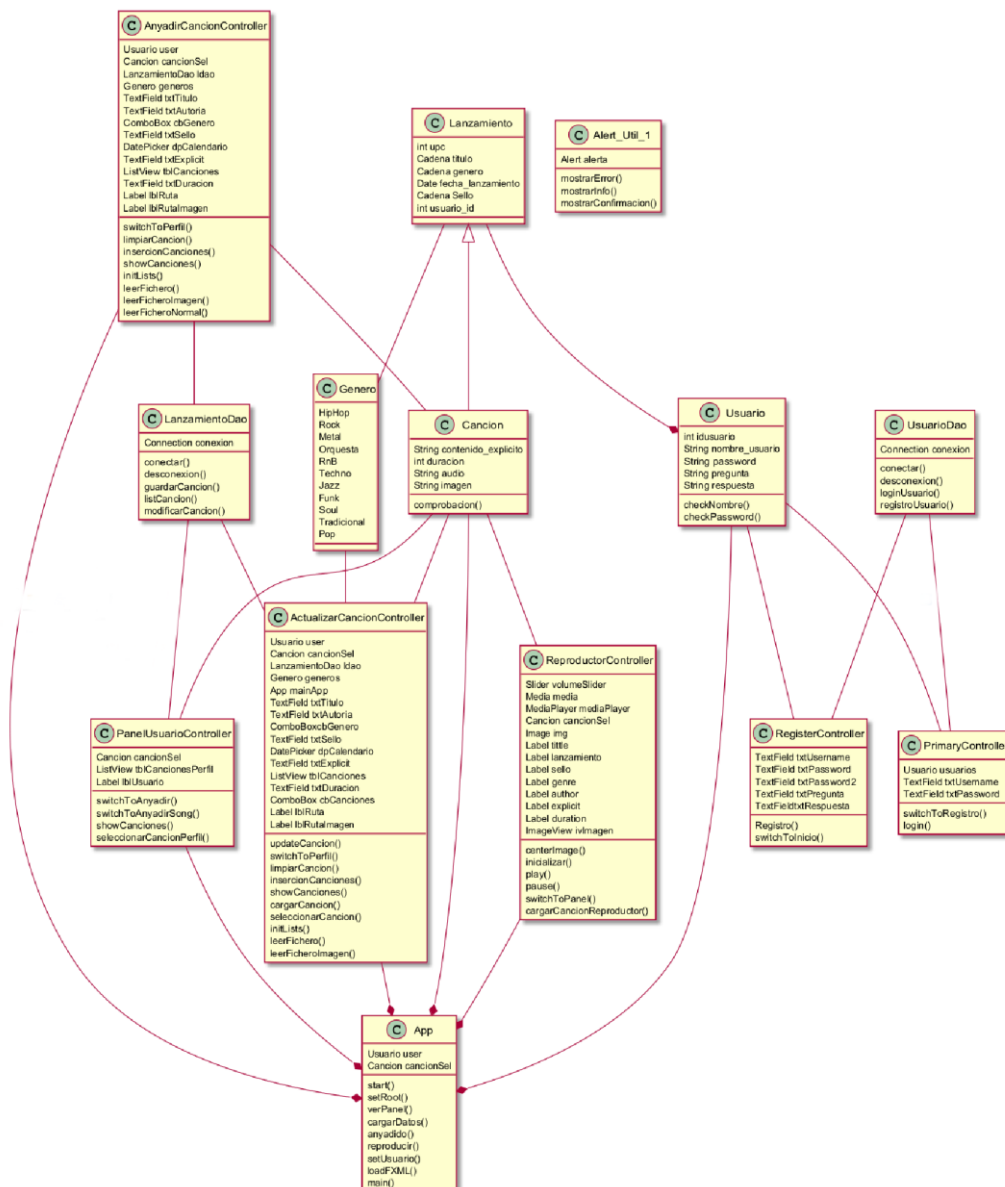


The screenshot shows a web application window titled "trackiD" with a dark theme. The main heading is "Editar biblioteca". The form contains the following fields and controls:

- Título:** A text input field containing "The Less I Know The Better".
- Autoría:** A text input field containing "Tame Impala".
- Género:** A dropdown menu with "Pop" selected.
- Sello:** A text input field containing "RCA Music Pub".
- Contenido explícito:** A text input field containing "No".
- Duración:** A text input field containing "3".
- Fecha de lanzamiento:** A date picker showing "8/9/1980".
- Modificar pista de audio (.mp3, .wav):** A text input field containing "The Less I K..." and a button with three dots.
- Modificar portada (jpg, .png, .gif):** A text input field containing "The Less I K..." and a button with three dots.
- Dropdown menu:** A dropdown menu on the right side of the form, currently showing "The Less I Know The Better - Tame Impala" as the selected item. The dropdown list is open, showing two options: "The Less I Know The Better - Tame Impala" (highlighted in blue) and "Traumfolge - Computer Data".

At the bottom of the form, there are four buttons: "Atrás", "Confirmar edición", "Borrar", and "Limpiar".

DIAGRAMA DE CLASES



En primer lugar, la clase *Cancion* hereda de la clase *Lanzamiento*, provocando así una relación de generalización, donde el elemento hijo (*Cancion*) comparte la misma estructura que el elemento padre (*Lanzamiento*). Además, *Lanzamiento* es una clase abstracta.

Por otro lado, la clase *Lanzamiento* guarda una relación de composición con la clase *Usuario*, ya que la primera no podría existir sin la segunda, es decir, no existiría canción sin usuario.

La clase enumerada *Genero* guarda una relación de asociación con la clase *Lanzamiento*, ya que ambas están relacionadas pero ambas clases son independientes la una de la otra.

Las clases *PrimaryController*, *ActualizarCancionController*, *AnyadirCancionController*, *PanelUsuarioController*, *ReproductorController*, *Cancion* y *Usuario* mantienen una relación de composición con la clase *App*, ya que aquellas que son *controller* requieren

de una clase principal (*App*) para poder funcionar, así como la utilización de objetos de las clases *Cancion* y *Usuario*.

Las clases *LanzamientoDao* y *UsuarioDao* se unirán con todas las clases de *controller* manteniendo una relación de asociación.

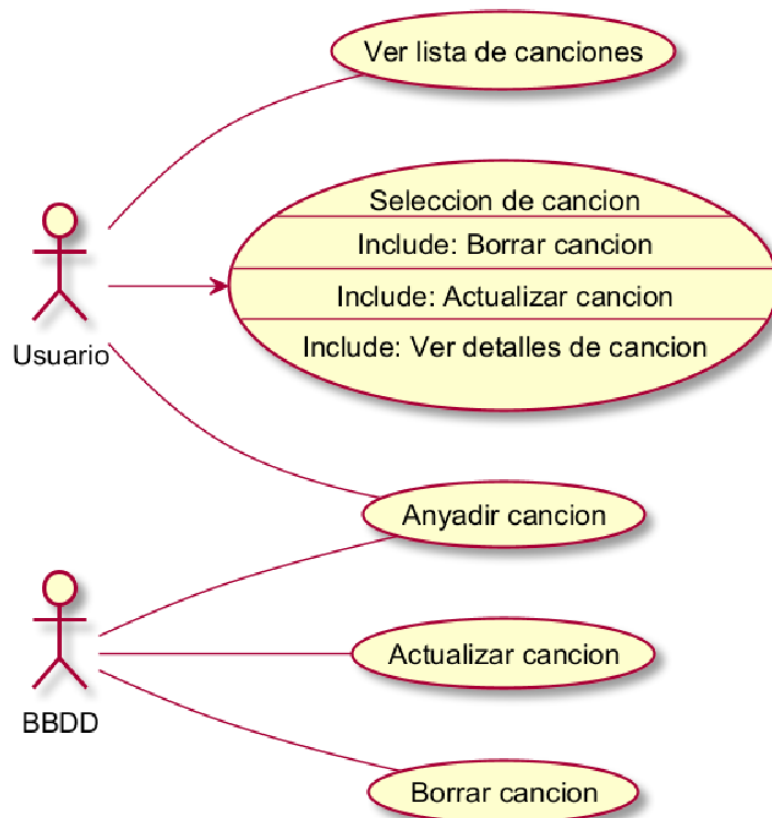
AnyadirCancionController guarda relación de asociación con *Cancion* ya que este controlador usará el objeto de la clase *Cancion* en uno o algunos de sus métodos.

ActualizarCancionController, *PanelUsuarioController*, *ReproductorController* guarda una relación de asociación con la clase *Cancion*, ya que este controlador usará objetos de la clase *Cancion* en uno o algunos de sus métodos.

RegisterController y *PrimaryController* guardan una relación de asociación con la clase *Usuario*, ya que estos controladores usarán objetos de la clase *Usuario* en uno o algunos de sus métodos.

La clase *Genero* guardará una relación de asociación con la clase *ActualizarCancionController*, ya que este controlador estará utilizando objetos de la clase enumerada *Genero* en alguno de sus métodos.

DIAGRAMA DE CASOS DE USO



Caso de Uso 1	
Nombre:	Ver lista de canciones
Actores:	Usuario
Función:	Mostrado de canciones
Descripción:	Mostrar todas las canciones del perfil
Caso de Uso 2	
Nombre:	Borrar canción
Actores:	Usuario y BBDD
Función:	Borrado de canción en BBDD
Descripción:	Borrado de elemento en BBDD
Caso de Uso 3	
Nombre:	Actualizar canción
Actores:	Usuario y BBDD
Función:	Actualizado de canción en BBDD
Descripción:	Actualizado de elemento en BBDD
Caso de Uso 4	
Nombre:	Añadir canción
Actores:	Usuario y BBDD
Función:	Añadido de canción
Descripción:	Añadido de un nuevo elemento a la BBDD

La aplicación estará compuesta por dos actores que se muestran como agentes externos al sistema: *Usuario* y *BBDD*.

De este modo, el actor *Usuario* (agente ya registrado en la plataforma con sesión iniciada) tendrá los siguientes casos de uso: ver lista de canciones; seleccionar una canción para poder borrarla, actualizarla o ver sus detalles; y añadir canción.

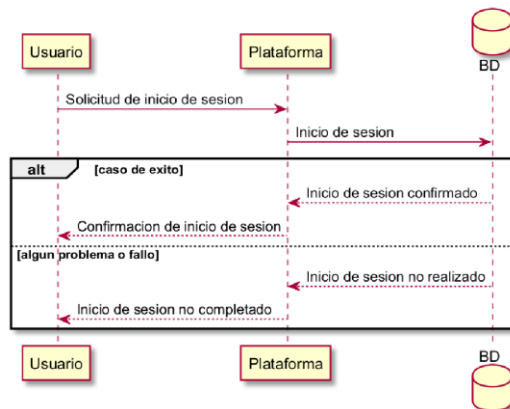
Por otro lado, será necesario contactar con el actor *BBDD* para realizar el añadido, actualización y borrado de canción, ya que estos usos modificarán la base de datos.

El borrado, actualización y vista de detalles de canción mantienen una relación de inclusión con *Usuario* mediante la selección de canción.

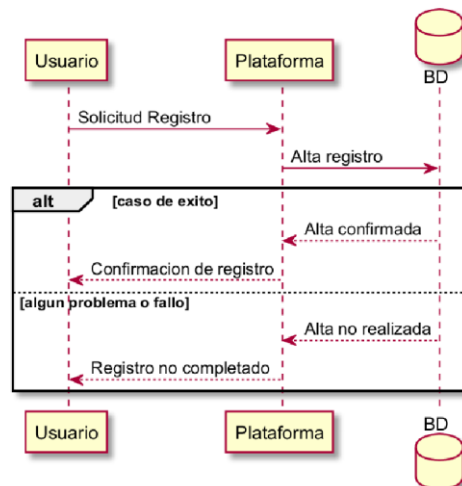
El resto de casos de uso mantendrán una relación de asociación o comunicación con sus respectivos actores.

DIAGRAMA DE SECUENCIA

Inicio de sesión



Registro

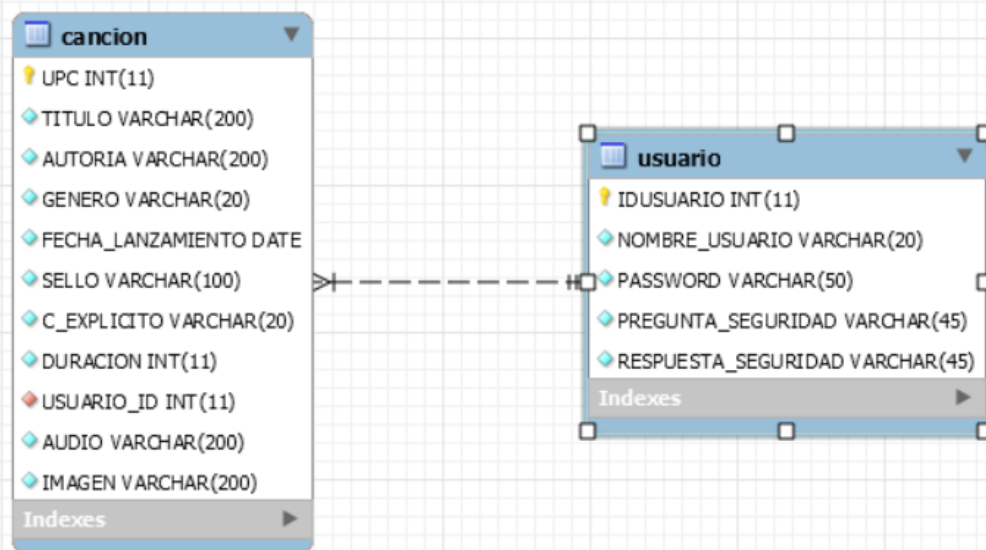
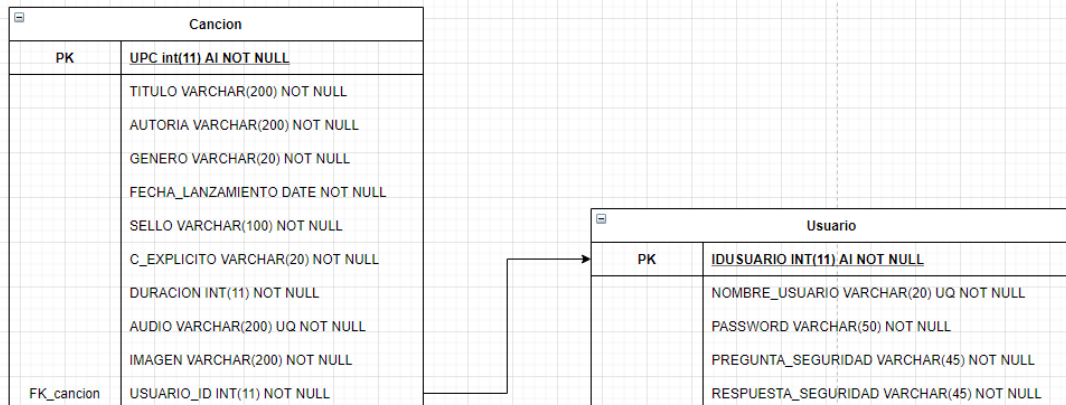
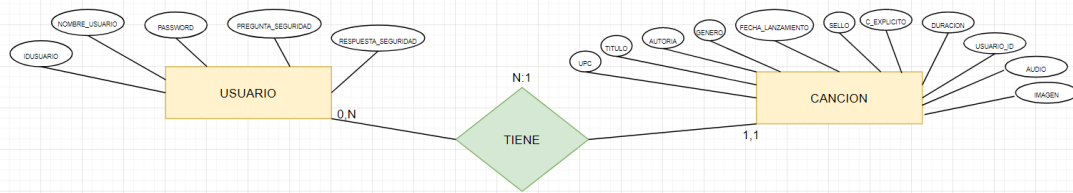


Formado por los participantes *Usuario*, *Plataforma* y *BD*. Se explicará el proceso de registro e inicio de sesión. Ambos se asemejan en cuanto al intercambio de información que se produce con la base de datos.

Un usuario se registrará o iniciará sesión enviando sus credenciales a la plataforma, la cual conectará con la base de datos haciendo el alta de registro o de inicio de sesión. Ante esto, pueden ocurrir dos situaciones:

- un caso de éxito, que se confirme el *login* o registro con la plataforma, y el usuario acceda o registre.
- un caso de fallo, que no se produzca de forma correcta el contacto con la base de datos y el registro o inicio de sesión no se complete.

DIAGRAMA E/R Y MODELO RELACIONAL (BBDD)



Podemos observar una relación de las tablas de la base de datos (*usuario* y *canción*) *trackiD* uno a muchos. Ya que es la conclusión obtenida del siguiente razonamiento:

- un usuario puede tener ninguna canción como mínimo y muchas como máximo (0, N).
- una canción solo podrá encontrarse en el perfil de un usuario al menos y como máximo (1, 1). Las canciones no se pueden repetir, tienen un código único (*UPC*) que hará que pertenezcan a un único usuario.

PRUEBAS FUNCIONALES

CASO 1: Usuario *ismaelism4* inicia sesión en la aplicación para añadir una canción que será distribuida en plataformas digitales en una fecha prevista. El usuario podrá subir su canción a la plataforma sin problema si se trata de un caso exitoso. De lo contrario, un mensaje de alerta aparecería en pantalla, advirtiéndole de que un error de conexión se ha producido. El caso erróneo obligaría a revisar la conexión con la base de datos.

CASO 2: Usuario *ismaelism4* inicia sesión en la aplicación y se dispone a modificar uno de sus lanzamientos. Se percata que se produce un fallo en el proceso de modificación de datos. Ante esta situación, sería de obligado cumplimiento revisar cómo se conecta la aplicación a la base de datos. El caso contrario y exitoso generaría una modificación de los parámetros introducidos para modificar.

	DESCRIPCIÓN	RESULTADO ESPERADO	RESULTADO OBTENIDO	OBSERVACIONES
CASO 1	Artista sube canción para lanzamiento	Conexión aplicación-base de datos. Guardado de información.	Fallo en la conexión. Reintroducción de datos.	Control de conexiones con base de datos.
CASO 2	Artista modifica canción	Conexión aplicación-base de datos. Modificación de información.	Fallo en la conexión. Reintroducción de datos.	Control de conexiones con base de datos.

- **Enlazar al proyecto en GitHub** 

BIBLIOGRAFÍA

- Bro Code. (2021, 13 marzo). *Reproductor de música con Java* [Vídeo]. YouTube. <https://www.youtube.com/watch?v=-D2OIekCKes&t=1501s>
- nicosiored. (2017, 7 diciembre). *Diagrama de Clases II - 7 - Tutorial UML en español* [Vídeo]. YouTube. <https://www.youtube.com/watch?v=NEmZZX8Xgns&t=578s>
- Deniz, N. (2021). *Código ejemplo*. GitHub.com. <https://github.com/NayraTeacher/Rutillas>
- *Centering an image in an ImageView*. (2021). StackOverFlow. <https://stackoverflow.com/questions/32781362/centering-an-image-in-an-imageview>