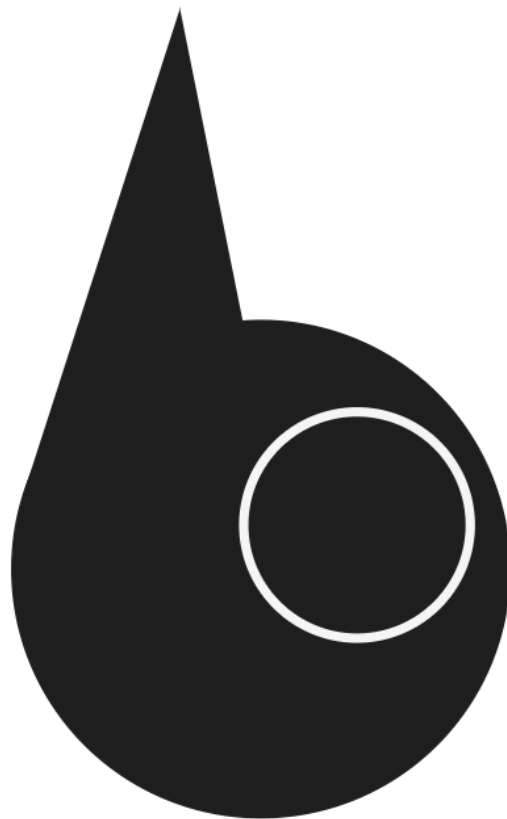


beat-on : Instrumentales online

Iván Naranjo Marín

DAW2 24/25



Tutor: Mario Alfonso Lasso Mesa

Índice

Idea y descripción del proyecto	3
Motivación del proyecto	4
Stack tecnológico	5
FRONT - END	5
BACK - END	5
TECNOLOGÍAS ADICIONALES	6
Funcionalidades y alcance de la aplicación	7
1 - RF relacionados con las cuentas y usuarios	7
2 - RF relacionados con los usuarios genéricos	7
3 - RF relacionados con usuarios gestores	8
4 - RF relacionados con usuarios administradores	8
5 - RF genéricos	8
Diseño cerrado de la BD:	9
AVATARS	10
USERS	10
ROLES	10
CATEGORIES	10
SOUNDS	11
INSTRUMENTALS	11
Enlaces a Github y repositorios	12
Repositorio de Front-end (Angular):	12
Repositorio de Back-end (Spring Boot Java):	12

Idea y descripción del proyecto

La idea del proyecto se basa en una aplicación Web diseñada para la creación de beats o ritmos musicales de forma gráfica y amigable para los usuarios mediante el almacenamiento de diferentes efectos e instrumentos en una base de datos.

El proceso de funcionamiento para la aplicación se basa en la implementación de un grid que funcione como mesa de mezclas, donde el usuario elegirá los diferentes efectos e instrumentos que elija entre las opciones que ofrece nuestro servicio, así como los momentos del ritmo donde estos sonarán, similar al funcionamiento de **controladores MIDI físicos**.

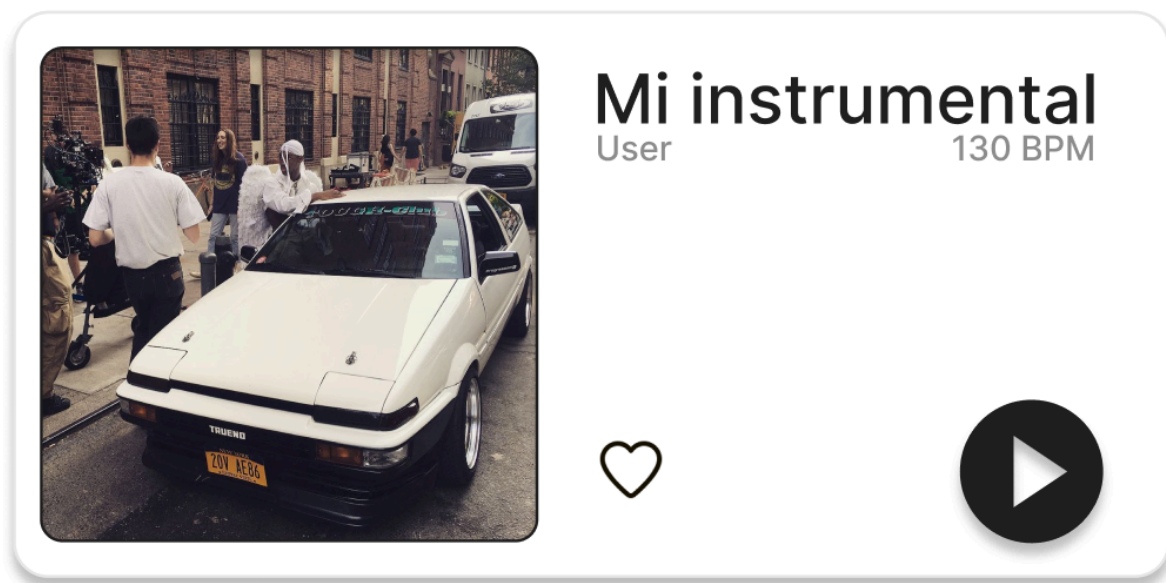
Ejemplo de controlador MIDI:



Esta es la idea principal de proyecto: poder ofrecer un servicio de creación de instrumentales; pero a la hora de diseñar la aplicación se ha decidido añadir funcionalidades similares a las de aplicaciones de redes sociales, como **Instagram** o **Tiktok**, así como funciones que tendrían aplicaciones de reproducción, como podrían ser **Spotify** o **Apple Music**.

Estas funciones pueden ser la acción de compartir la instrumental que el usuario la ha creado, para que otros usuarios puedan indicar si les ha gustado, en forma de likes o guardandola en una playlist.

Ejemplo de Mock-Up de la instrumental:



Como últimas funcionalidades que se tienen que tener en cuenta para describir este proyecto, también se tendrán en cuenta las funciones que se darán a los usuarios administradores de la aplicación, que será un equipo dedicado al control de usuarios y de cuentas; a su vez, el control de usuarios gestores, que se dedicarán al control y gestión del resto de utilidades que tiene el proyecto para su funcionamiento.

Para ello, se crearán diferentes dashboards de gestión, que ayuden al equipo de gestores y administradores a realizar de forma eficiente las tareas.

Además de esto, en esta aplicación se añadirán funciones estándar en el diseño web, como accesibilidad visual y auditiva, diseño adaptado a los diferentes tamaños de pantallas y personalización de los estilos de la aplicación (E.J, modo oscuro / Claro)

Motivación del proyecto

La idea se concibe debido a la curva de dificultad que pueden tener estos sistemas, así como los precios generales de las licencias de estos productos, en general muy poco económicos.

En resumen, la motivación principal del proyecto es **crear una interfaz gráfica simple y accesible** para la **creación de loops de instrumentales**, con la posibilidad de **compartir con otros usuarios las instrumentales**.

Stack tecnológico

Para la creación de esta aplicación web, pienso utilizar las siguientes tecnologías para el proyecto:

FRONT - END

Para el frontend del proyecto, la aplicación estará realizada con **Angular y Node**, debido a los conocimientos previos en este framework, así como la información a la que puedo acceder en caso de dudas con los tutores o con documentación en la moodle / documentación oficial.

Para obtener un diseño responsive en la aplicación web también se usará **tailwind css**, y se buscarán **librerías de componentes compatibles** o se crearán **componentes personalizados**.

En cuanto las librerías específicas para tratado de audio, investigaré las siguientes, usando las adecuadas para el proyecto: **audiobuffer-to-wav**, **fluent-ffmpeg**, **wav**.



BACK - END

La tecnología principal que voy a usar en mi proyecto para el entorno servidor va a ser **Spring Boot**, donde realizaré una API personalizada para la utilización de todas las funciones y procesos que tenga que realizar la aplicación web. A su vez, utilizaré **Docker** para la creación de contenedores donde se encontrará el servidor, así como **MariaDB** para la base de datos.

Para el tratado de sonidos por parte del back-end, investigaré las siguientes tecnologías: **Java.sound** y **JAAD**.



TECNOLOGÍAS ADICIONALES

Para el control de versiones, se utilizará **Git**, manteniendo el repositorio del proyecto en **GitHub** y utilizando extensiones u opciones de push integrado en los IDE que se van a utilizar.

A su vez, se buscarán tecnologías y / o librerías relacionadas a las previas para el manejo de las diferentes funcionalidades que tendrá la aplicación, si es necesario.



Funcionalidades y alcance de la aplicación

En este apartado se muestran los requisitos funcionales que busco cumplimentar en mi proyecto:

1 - RF relacionados con las cuentas y usuarios

RF01 - Sistema de login de los usuarios.

Los usuarios accederán a una vista de login, donde introducirán sus credenciales para acceder a su perfil.

RF02 - Sistema de logout de los usuarios.

Los usuarios tendrán la posibilidad de cerrar su perfil cuando lo deseen.

RF03 - Sistema de registro de usuarios.

Los usuarios podrán utilizar sus datos para crear una cuenta asociada a sus credenciales.

RF04 - Sistema de edición del perfil del usuario.

Si es necesario, el usuario podrá modificar sus datos personales y como se muestra su perfil en la aplicación.

RF05 - Sistema de eliminación de cuentas.

Los usuarios tendrán acceso a una sección para eliminar su cuenta del sistema si así lo desean, para ello se creará un menú de confirmación para que no exista la posibilidad de eliminar la cuenta accidentalmente.

2 - RF relacionados con los usuarios genéricos

RF06 - Creación de instrumentales personalizadas.

El usuario accederá a la mesa de trabajo, donde se le proporcionará las herramientas para la creación de la instrumental.

RF07 - Publicar de forma pública en la aplicación la instrumental creada.

Si así lo desea, el usuario podrá compartir su instrumental en la página, donde pasará a ser pública y cualquier usuario podrá visualizarla.

RF08 - Añadir a favoritos instrumentales propias o de otros usuarios.

El usuario podrá añadir a favoritos las diferentes instrumentales, que aparecerán en su pestaña de favoritos.

Extra - RFE01 - Exportación del instrumental propio o de otro usuario.

Los usuarios pueden descargar el loop que han creado en un archivo.

3 - RF relacionados con usuarios gestores

RF09 - Gestión de archivos de sonido (Creación, Edición y eliminación).

Los gestores accederán a un dashboard, donde podrán añadir, editar o eliminar los diferentes sonidos que aparecerán en la aplicación.

RF10 - Gestión de categorías (Creación, Edición y eliminación).

Los gestores accederán a un dashboard, donde podrán añadir, editar o eliminar las diferentes categorías de sonidos que aparecerán en la aplicación.

RF11 - Gestión de visibilidad de sonidos.

Los gestores tendrán la opción de ocultar o mostrar sonidos en los casos necesarios.

4 - RF relacionados con usuarios administradores

RF12 - Gestión de cuentas (Edición y eliminación).

Los administradores se dedicarán a editar y eliminar cuentas si es necesario, así como activar o desactivar el acceso a ellas.

5 - RF genéricos

RF13 - Modo Claro y Oscuro.

Se creará la posibilidad de seleccionar entre un modo claro o oscuro para reducir la fatiga visual de los colores.

RF14 - Menús de navegación dinámicos dependientes del tipo de usuario.

Esta funcionalidad está dedicada a mostrar los bloques de contenido que suario cada utiene acceso desde la sección de navegación, eliminando los contenidos no autorizados para el usuario.

RF15 - Sistema de notificaciones.

Los usuarios administradores podrán mandar notificaciones a los usuarios gestores y comunes.

RF16 - Soporte multilingüe (Español e inglés) .

La aplicación contará con traducción al inglés, para poder seleccionar entre español e inglés.

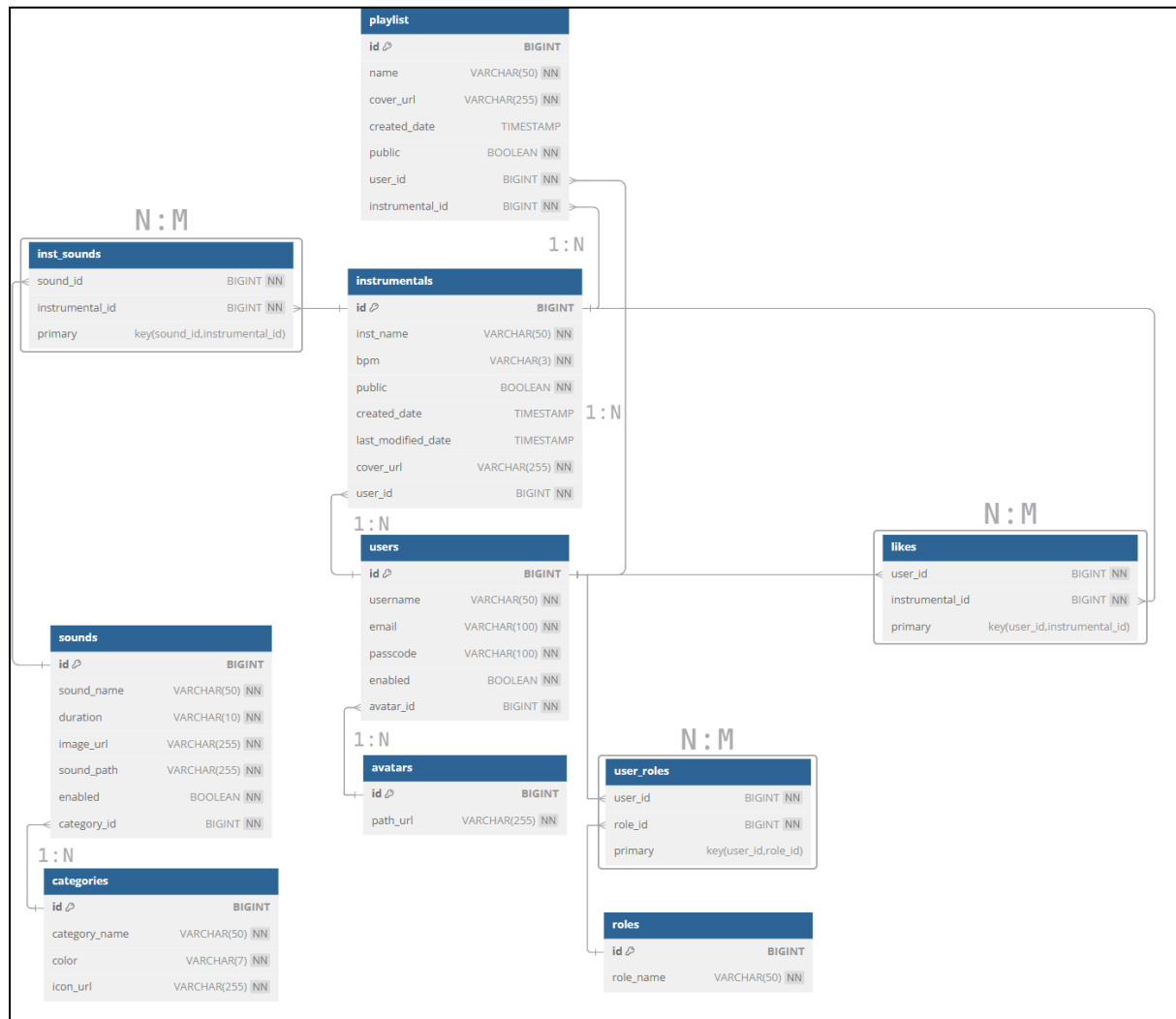
RF17 - Responsividad en todo tipo de dispositivos.

La aplicación tendrá diferente diseño dependiendo del tamaño de la pantalla en la que se haya accedido al front-end de la aplicación.

RF18 - Accesibilidad visual y auditiva.

La aplicación estará diseñada para disponer de ayudas para las personas con discapacidad, tanto visuales como auditivas, de acuerdo a los estándares.

Diseño cerrado de la BD:



En esta aplicación nos encontramos con 4 entidades principales: **Sonidos**, **Instrumentales**, **playlists** y **usuarios**; con sus respectivas entidades auxiliares.

A continuación, se muestra el diccionario de datos para todas las entidades de la BD:

AVATARS

DATO	TIPO DE DATO	DESCRIPCIÓN	RESTRICCIONES
id	BIGINT	Identificador único	Primary Key
path_url	VARCHAR(255)	Ruta de archivo del avatar	Not Null

USERS

DATO	TIPO DE DATO	DESCRIPCIÓN	RESTRICCIONES
id	BIGINT	Identificador único	Primary Key
username	VARCHAR(50)	Nombre de usuario	Not Null, Unique
email	VARCHAR(100)	Correo electrónico del usuario	Not Null, Unique
passcode	VARCHAR(100)	Contraseña del usuario	Not Null, Unique
enabled	BOOLEAN	Estado del usuario	Not Null
avatar_id	BIGINT	Referencia al avatar del usuario	Not Null, Foreign Key

ROLES

DATO	TIPO DE DATO	DESCRIPCIÓN	RESTRICCIONES
id	BIGINT	Identificador único	Primary Key
role_name	VARCHAR(50)	Nombre del rol	Not Null, Unique

CATEGORIES

DATO	TIPO DE DATO	DESCRIPCIÓN	RESTRICCIONES
id	BIGINT	Identificador único	Primary Key
category_name	VARCHAR(50)	Nombre de la categoría	Not Null
color	VARCHAR(7)	Color asociado a la categoría	Not Null
icon_url	VARCHAR(255)	Ruta de archivo del icono de la categoría	Not Null

SOUNDS

DATO	TIPO DE DATO	DESCRIPCIÓN	RESTRICCIONES
id	BIGINT	Identificador único	Primary Key
sound_name	VARCHAR(50)	Nombre del sonido	Not Null
duration	VARCHAR(10)	Duración del audio	Not Null
image_url	VARCHAR(255)	Imagen relacionada con el sonido	Not Null
sound_path	VARCHAR(255)	Ruta del sonido	Not Null
enabled	BOOLEAN	Estado del usuario	Not Null
category_id	BIGINT	Referencia a la categoría a la que pertenece el sonido	Not Null, Foreign Key

INSTRUMENTALS

DATO	TIPO DE DATO	DESCRIPCIÓN	RESTRICCIONES
id	BIGINT	Identificador único	Primary Key
inst_name	VARCHAR(50)	Nombre de la instrumental	Not Null, Unique
bpm	VARCHAR(3)	Beats por minuto	Not Null
public	BOOLEAN	Estado (Público / Privado)	Not Null
created_date	TIMESTAMP	Fecha de creación	Not Null
last modified date	TIMESTAMP	Fecha de última modificación	Not Null
cover_url	VARCHAR(255)	Portada de la instrumental	Not Null
user_id	BIGINT	Referencia al usuario	Not Null

Enlaces a Github y repositorios

Repositorio de Front-end (Angular):

Todo el contenido de angular dedicado al front-end se guarda en el repositorio llamado **beat-on-angular**, este repositorio está conectado a visual studio code para realizar desde el IDE todas las subidas del contenido con mayor comodidad, y además la configuración de angular ha sido cambiada para utilizar el puerto con el que se ha trabajado para el despliegue, facilitando aún más este último proceso y sin necesitar realizar cambios en la máquina servidor.

Repositorio de Back-end (Spring Boot Java):

Para el contenido de back-end, se ha creado otro repositorio llamado **beat-on-api**. Este repositorio también se encuentra vinculado al IDE que se está utilizando para la realización del proyecto, lo que ayuda a la hora de mantener el control de versiones limpio y cómodo a la hora de reconocer los archivos añadidos, actualizados o eliminados en los diferentes procesos de la creación de la API.

A su vez, los 2 repositorios cuentan con sus archivos .gitignore, que indican los archivos que no se suben a github. (Como pueden ser los .env, que se eliminan de estos por seguridad).