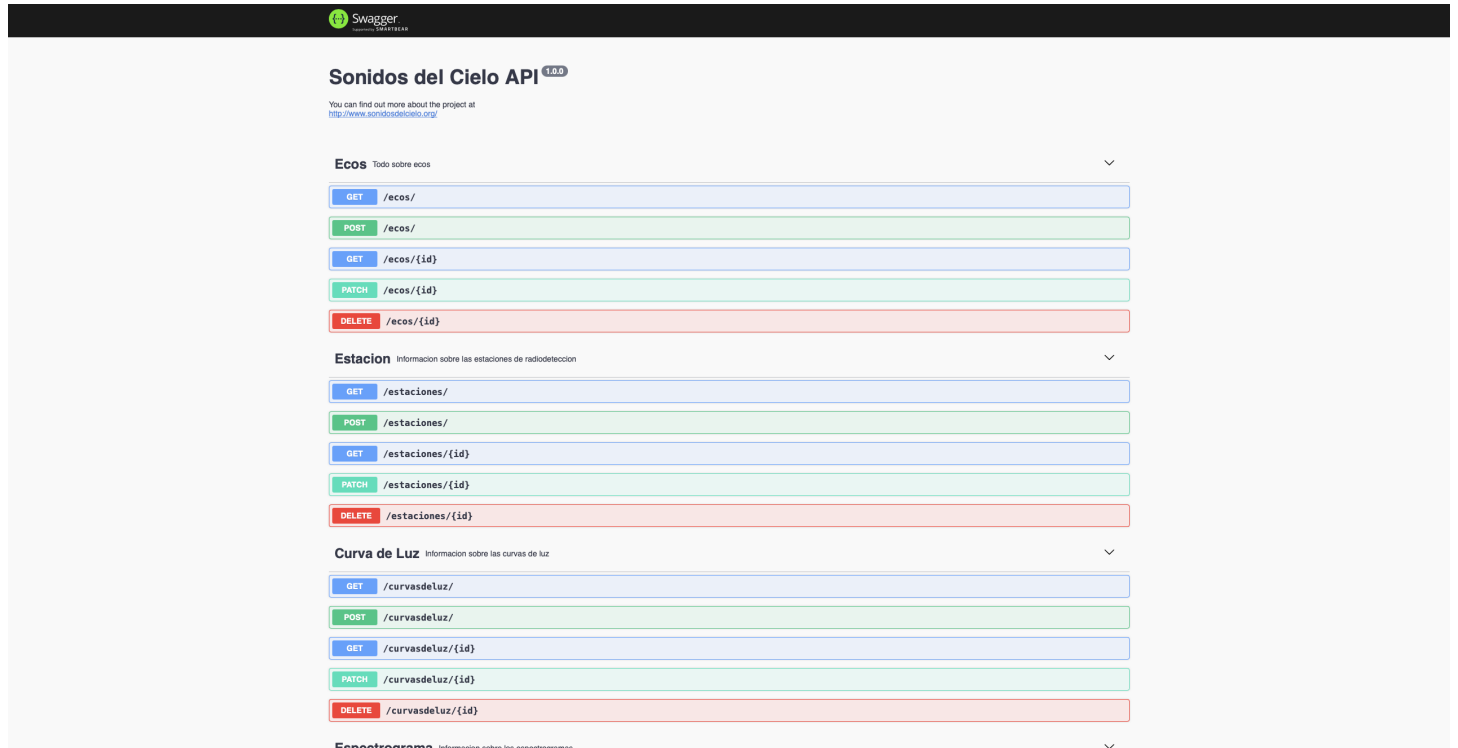


API REST para comunicar servicios con la base de datos del proyecto Sonidos del Cielo



Introducción

- El proyecto tiene como finalidad la de permitir establecer comunicaciones entre los distintos servicios del proyecto Sonidos del Cielo y la base de datos del proyecto.

About

- Este proyecto ha sido desarrollado por [Marcos Pino](#) como Trabajo de Fin de Grado en la [Universidad Politécnica de Madrid](#) para el grado de ingeniería informática de la [Escuela Técnica Superior de Ingenieros Informáticos](#) para el proyecto [Sonidos del Cielo](#) del [Laboratorio de Ciencia Ciudadana](#). Esta aplicación tiene como finalidad la de comunicar los microservicios emplazados dentro del proyecto Sonidos del Cielo.

Despliegue

- Antes que nada, tenemos que instalar [Docker](#)

Instalación en Windows

- Para instalar docker en windows, lo descargamos del siguiente [link](#).
- Cuando esté descargado, hacer doble-click en Docker for Windows Installer para correr el instalador.
- Al finalizar la instalación, se mostrará una indicación con un icono de ballena que señala que Docker está funcionando.

Instalación en Linux

- Primero, actualizamos la lista de paquetes existente con el comando `sudo apt update`
- A continuación, instalamos paquetes de requisitos previos para permitir a apt utilizar paquetes a través de HTTPS `sudo apt install apt-transport-https ca-certificates curl software-properties-common`

- Después, añadir la clave GPG para el repositorio oficial de Docker `curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -`
- Agregar el repositorio de Docker a las fuentes de APT `sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable"`
- A continuación, actualizar el paquete de base de datos con los paquetes de Docker del repositorio agregado `sudo apt update`
- Por último, instalar Docker: `sudo apt install docker-ce`

Instalación en Mac

- Para instalar docker en macOS, lo descargamos del siguiente [link](#).
- Cuando esté descargado, hacer doble-click en `Docker.img` para correr el instalador.
- Al finalizar la instalación, se mostrará una indicación con un icono de ballena que señala que Docker está funcionando.

Persistencia de la base de datos

- Para ello creamos una carpeta dentro del equipo dónde se vaya a ejecutar y añadimos la ruta completa dentro del fichero `docker-compose.yml`

Cadena de conexión para la base de datos

- Es necesario tener una cadena de conexión que conecte la API con la misma, para ello creamos un archivo `.env` que irá en la raíz del proyecto clonado. Dentro introducimos una línea de texto que siga esta configuración

`DB_CONNECTION=mongodb://usuario:contraseña@mongo:27017/baseDeDatos`

Despliegue de los contenedores

- Una vez tengamos Docker instalado y hayamos clonado el proyecto, ejecutamos el siguiente comando: `$ docker-compose up -d`
- Una vez completado el comando, tendremos una base de datos MongoDB funcionando en el puerto `27017`

Creación de las colecciones en MongoDB

- Para que exista un correcto funcionamiento, es necesario crear una serie de colecciones dentro de la base de datos. Para ello se puede utilizar el CLI que incluye mongo. Para entrar en el contenedor de mongo buscamos el id del contenedor mediante `docker ps -a`, este comando listará los distintos contenedores existentes en el equipo, tomamos el id del contenedor mongo (vale con los tres primeros caracteres) e introducimos en un terminal el comando `docker exec -it <id del contenedor> bash` esto cambiará el prompt. Una vez dentro introducimos el comando `mongo` para entrar en el CLI de mongo.
- Para crear las colecciones, simplemente tendremos que introducir el comando `db.createCollection(nombre)`. Necesitamos crear las colecciones Clasificacion, Curva Luz, Eco, Espectrograma, Estacion y Sonido.

Funcionamiento

Una vez hayamos desplegado los contenedores, tendremos acceso a una interfaz de pruebas desarrollada en [Swagger](#) a través de `http://localhost:3001/api/docs/`