

Bases del ejercicio

El objetivo es desarrollar una API (No se debe realizar un FrontEnd) para consumir un servicio externo y utilizar su información, además se tiene que ampliar el scope del desarrollo agregando una funcionalidad más que permita persistir información nueva en el sistema, relacionada con los datos de la API externa.

El único requisito obligatorio es usar Java con Spring Boot como tecnología de desarrollo y alojar en un repositorio en Github o Gitlab, como así también ir commiteando a medida que se realizan avances, inclusive si estos contemplan correcciones/cambios que se van realizando a lo largo de la resolución, y no todo en un solo commit. Como regla general, 10-20 commits suele ser razonable, siendo 1-5 muy poco y más de 30 demasiado.

Problema de negocio

Se tiene que consumir información del siguiente servicio externo que cuenta con los datos de usuarios, sus álbumes y sus fotos; además de sus posts y los comentarios de otros usuarios sobre ellos: <https://jsonplaceholder.typicode.com/>

De la información del servicio se tiene que poder acceder a través de nuestra API a:

1. Los usuarios.
2. Las fotos.
3. Los álbumes del sistema y de cada usuario.
4. **Plus:** Las fotos de un usuario.

De esta forma la API va a poder concentrar esta información para luego poder extenderla a una nueva funcionalidad, la cual consiste en implementar una gestión de permisos básica a cada álbum de fotos (lectura y escritura) para compartir álbumes entre los usuarios de la plataforma. De esta manera se debe registrar qué usuarios tienen acceso a álbumes que no son propios y los permisos de dicho usuario para ese álbum. Para persistir y consumir esta información nueva el sistema debe permitir:

1. Registrar un álbum compartido con un usuario y sus permisos.
2. Cambiar los permisos de un usuario para un álbum determinado.
3. Traer todos los usuarios que tienen un permiso determinado respecto a un álbum específico.

En cuanto a los comentarios, se espera que la aplicación pueda traerlos del servicio externo brindando la posibilidad de filtrar por el campo "name" o por el usuario que realizó dicho comentario.