

Trabajo Evaluación 2 - Acceso a Datos

QUÉ DEBE ENTREGAR

Este trabajo debe ser realizado usando el sistema de control de versiones GIT y debe exponer su resolución al profesor. Debe entregarse el enlace al repositorio de GIT del proyecto así como el trabajo comprimido en el classroom. La falta de cualquiera de estos dos entregables o de la exposición supone un suspenso directo.

Enunciado:

El trabajo consiste en crear una red social. Para ello debemos tener un usuario que tenga un username, un email, una contraseña, una descripción, una fecha de creación. Tanto el username como el email deben ser únicos y no pueden repetirse. Un usuario podrá publicar muchas publicaciones. Las publicaciones tendrán un autor, un texto, una fecha de creación y una fecha de edición. Como es una red social, un usuario puede seguir a muchos usuarios y un usuario puede ser seguido por otros muchos usuarios.

Para aprobar, el alumno deberá cumplir los siguientes puntos:

- Tener el proyecto estructurado de manera apropiada.
- Crear una base de datos MySQL consistente que resuelva el problema planteado. La contraseña del usuario debe estar cifrada con Bcrypt.
- Desarrollar entidades para todas las tablas con las anotaciones adecuadas, así como definir sus repositorios con los métodos necesarios.
- Desarrollar los DTO que sean necesarios para evitar enviar información sensible. La exposición de la contraseña implica suspenso directo.
- Definir los servicios para cada entidad, desarrollando los métodos que implementan la lógica de tu aplicación.
- Definir el/los controlador/es necesarios para tu API REST, siguiendo una estructura de endpoints adecuada.

- Implementar Spring Security con JWT para evitar que usen la API si no están logueados.

Y tener como mínimo los siguientes endpoints en su API:

- Registro (público)
- Login (público)
- Editar descripción (privado)
- Buscar usuario por username (público)
- Obtener todos los usuarios que sigue un usuario (privado)
- Obtener todos los usuarios que siguen a un usuario (privado)
- Obtener todas las publicaciones (privado)
- Obtener todas las publicaciones de un usuario (público)
- Obtener todas las publicaciones de los usuarios que sigues (privado)
- Insertar publicación (privado)
- Editar publicación (privado)
- Borrar publicación (privado)

Para el resto de la nota, se deben cumplir los siguientes puntos:

- Fichero Readme con información COMPLETA de la aplicación (No se aceptarán ficheros muy básicos). (0.5 puntos)
- Implementar un Swagger para documentar los endpoints. (0.5 puntos)
- Definir al menos un ControllerAdvice que nos devuelva una respuesta con mensaje al saltar una excepción. (0.5 puntos)
- Implementar la publicación de comentarios en las publicaciones así como la inserción de imágenes en las publicaciones. (1.5 puntos)
- Crear una interfaz gráfica con Flutter que consuma la API. (2 puntos)

El alumno deberá demostrar en la exposición el conocimiento de la implementación y funcionamiento del programa, así como de todos los puntos de manera detallada. El desconocimiento de la manera de realizar alguno de los puntos implica un 0 en ese apartado en concreto (en caso de ser de uno de los apartados mínimos para aprobar, se suspenderá el trabajo completo). Por tanto, la

obtención completa de cada punto está ligada a su correcta exposición. La mala ejecución del programa (Cualquier tipo de fallo no controlado) provocará el suspenso directo del alumno.