

Trabajo Evaluación 1 - Servidor

QUÉ DEBE ENTREGAR

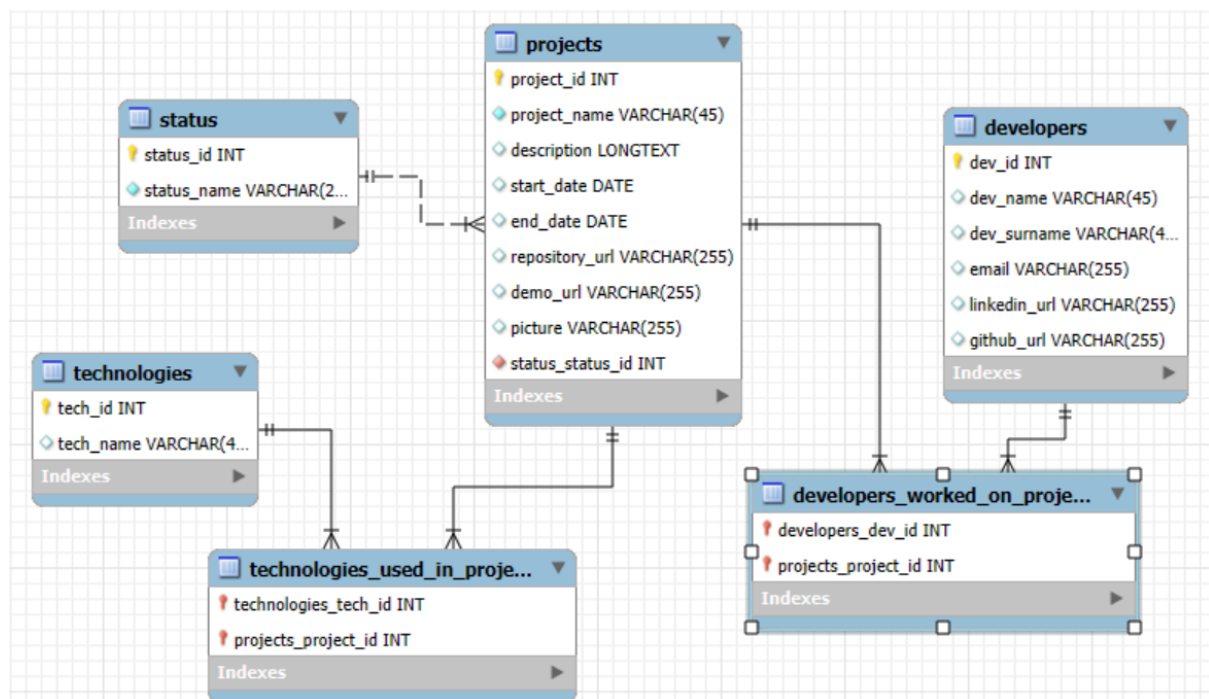
Este trabajo debe ser realizado usando el sistema de control de versiones GIT y debe exponer su resolución al profesor. Debe entregarse el enlace al repositorio de GIT del proyecto así como el trabajo comprimido en el classroom. La falta de cualquiera de estos dos entregables o de la exposición supone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

Enunciado:

El trabajo consiste en desarrollar una API REST que gestione la información de diferentes proyectos.

Dicha API REST deberá poder gestionar al completo las diferentes tablas de la base de datos para que el usuario pueda gestionar su información por medio de peticiones HTTP.

El esquema de la base de datos es el siguiente:



Para aprobar, el alumno deberá cumplir los siguientes puntos:

- Tener una estructura de proyecto apropiada, usando el patrón de diseño Modelo Vista Controlador (O para ser más correctos, Controlador Servicio Repositorio).
- Modelar las entidades con los tipos apropiados. Esto significa que la aplicación debe correr con la opción de “ddl-auto” con valor “validate”.
- Definir correctamente los métodos de cada repositorio. Deben estar definidos única y exclusivamente los métodos que se vayan a usar.
- Definir la lógica de tu aplicación mediante Servicios.
- El servidor nunca debe parar su ejecución. Interrumpir la ejecución del servidor es motivo de suspenso.
- La respuesta del controlador debe ser siempre un ResponseEntity con el código de respuesta adecuado.
- Hacer uso del sistema de control de versiones GIT.
- Definir los siguientes endpoints (base: /api/v1):
 - 1) GET /projects -> Obtener todos los proyectos paginados. En la respuesta, debe mostrarse en cada proyecto su estado, las tecnologías con las que se realizó y los desarrolladores que trabajaron en ella.
 - 2) GET /projects/{word} -> Obtener el proyecto que contenga la palabra word en su nombre. La estructura de la respuesta debe ser la misma que la de la respuesta del punto anterior, solo que apareciendo los resultados correspondientes.
 - 3) POST /projects -> Insertar proyecto.
 - 4) PUT /projects/{id} -> Editar el proyecto.
 - 5) Delete /projects/{id} -> Eliminar el proyecto.
 - 6) POST /developers -> Insertar un programador.
 - 7) DELETE /developers -> Borrar un programador.
 - 8) POST /technologies -> Insertar tecnología.
 - 9) DELETE /technologies -> Borrar tecnología.

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

- Buenas prácticas: uso de comentarios, tratamiento de errores, refactorización de código... (0.5 puntos)
- Fichero Readme con información COMPLETA de la aplicación (Tendréis una guía que os adjuntaré en la tarea). (0.5 puntos).
- Desarrollar un endpoint PATH /api/v1/projects/totesting y un endpoint PATH /api/v1/projects/toprod que pase el proyecto de Development a Testing y de Testing a Production. (0,5 puntos).
- Desarrollar un DTO de respuesta con tipo genérico. Esta clase contendrá como atributo un mensaje y el tipo genérico. Usar este DTO como respuesta a todos los métodos de manipulación de datos (POST, PUT, PATH, DELETE). (0,5 puntos).
- Desarrollar los endpoints POST /api/v1/technologies/used y POST /api/v1/developers/worked que establezca que una tecnología se ha usado en un proyecto y que establezca que trabajador a trabajado en un proyecto (1 punto).
- Añadir un swagger con la librería SpringDoc en el que describiremos la aplicación y detallaremos cada endpoint. (0.5 puntos).
- Desarrollar un RestControllerAdvice que capture las excepciones (0.5 puntos).
- Aplicar anotaciones de validaciones a los diferentes atributos, así como una custom validation para las url.
- Desarrollar un endpoint GET /api/v1/projects/tec/{tech} haciendo uso de las custom queries de JPA. Dicho endpoint debe devolver como respuesta los proyectos en los que se ha aplicado dicha tecnología. (0,5 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.

