

TUDAI | Final Web 2: 05 de Septiembre de 2023

Alumno	DNI	Firma	#hojas

La Facultad quiere realizar un relevamiento de la calidad de sus docentes. Para ello, nos piden que armemos un sistema donde los **alumnos** puedan registrar la opinion que perciben de los profesores de una materia.

Aunque la opinión es anónima, se almacena información del alumno para evitar que opine mas de una sobre el mismo docente. Por lo tanto cada opinión debe almacenar **dni, fecha, profesor, e imagen** (positiva - negativa - ns/nc). Además, para cada profesor el sistema debe almacenar su **nombre, edad y materia**.

Las tablas y relaciones de la BBDD para representar el problema son:

- Materia (**id**, nombre)
- Profesor (**id**, nombre, edad, id materia)
- Opinion (**id**, dni, fecha, imagen, id profesor)

NOTA: no es necesario implementar las vistas ni TPL's.

1. Implemente siguiendo el **patrón MVC** la funcionalidad para:

Insertar un registro de una opinión en el sistema.

Tenga en cuenta que:

- El sistema debe controlar posibles errores.
- El sistema no debe permitir que un mismo alumno voté dos veces al mismo profesor.

2. Implemente siguiendo el **patrón MVC** la funcionalidad para:

Obtener la imagen de un profesor determinado (dar % de imagen positiva y negativa).

Tenga en cuenta que:

- El sistema debe controlar posibles errores.
- Asuma que existen dos funciones llamadas **calcularImagenPositiva(\$arreglo)** y **calcularImagenNegativa(\$arreglo)** que reciben el arreglo de opiniones por profesor y retornan el valor correspondiente (%). **Se debe pasar solo las opiniones del profesor deseado.**
- Se deben mostrar todos los datos asociados al profesor (nombre, edad, materia, etc)

3. **La Facultad** está pensando en diseñar un servicio REST para exponer alguna de las funcionalidades del sistema.

a. Obtener la lista de profesores

- b. Obtener un profesor en particular
 - c. Eliminar un profesor
 - d. Obtener la lista de profesores ordenada por imagen positiva ascendente o descendente
-
- i) Defina la tabla de ruteo para cada requerimiento. (ENDPOINT + METODO HTTP + CONTROLADOR+MÉTODO). No es necesario implementar.
 - ii) Implemente los puntos B y D
-
4. Explique el **patrón de diseño MVC**. Describa cada una de sus partes y la responsabilidades de las mismas. Ventajas y desventajas del uso de este patrón.
-
5. Complete las siguientes frases:
- a. PDO es más seguro que conectarse a la base de datos porque...
 - b. Si hay una sesión abierta no necesariamente hay un usuario logueado ya que...
 - c. Al tener roles de usuario, no alcanza con ocultar los links en la vista porque...
 - d. Un password debe hashearse y no encriptarse porque...