

Trabajo práctico final TP2

El objetivo principal de este trabajo práctico es que los estudiantes tengan la oportunidad de explorar las posibilidades de Node.js, Express y Sequelize, y que desarrollen habilidades para crear soluciones más robustas y complejas en términos de arquitectura de software y calidad de código. Esto implica que los estudiantes deberán aplicar buenas prácticas de desarrollo, como la modularidad, la escalabilidad y la seguridad, y que tendrán que tener en cuenta aspectos como el rendimiento, la usabilidad y la mantenibilidad del código.

Para desarrollar el trabajo práctico, los estudiantes deberán seguir los siguientes pasos:

1. Crear un proyecto de Node.js utilizando el gestor de paquetes npm.
2. Instalar y configurar Express como framework para el servidor web.
3. Configurar Sequelize para conectarse a una base de datos.
4. Crear los modelos necesarios para la base de datos.
5. Crear las rutas necesarias para el API RESTful.
6. Implementar las funciones CRUD para cada modelo utilizando Sequelize.
7. Organizar el código utilizando el patrón Modelo-Vista Controlador (MVC), repository o DAO.

Es importante tener en cuenta que el trabajo práctico debe cumplir con las siguientes especificaciones y se anima a investigar y hacer implementaciones adicionales:

- La base de datos debe contar con al menos 2 tablas relacionadas entre sí.
- Las rutas del API RESTful deben permitir realizar operaciones CRUD en la base de datos.
- El código debe estar organizado utilizando el patrón MVC,

repository o DAO .

- Se debe utilizar manejo de errores adecuado para garantizar la robustez del servidor.
- El trabajo práctico debe entregarse en un repositorio Git, junto con un archivo README que contenga las instrucciones necesarias para ejecutar el servidor y utilizar el API RESTful.
- Los estudiantes deberán realizar una presentación oral del trabajo práctico, en la que expliquen su implementación y respondan preguntas del profesor.

La evaluación del trabajo práctico se realizará teniendo en cuenta los siguientes aspectos:

- Correcta implementación de las rutas del API RESTful.
- Correcta implementación de las funciones CRUD utilizando Sequelize.
- Correcta organización del código utilizando el patrón MVC, repository o DAO.
- Correcta gestión de errores y excepciones en el servidor.
- Correcta documentación del código y del repositorio.
- Calidad y claridad de la presentación oral y capacidad para responder preguntas.

Se recomienda a los estudiantes que utilicen las herramientas y recursos proporcionados durante el curso para la realización del trabajo práctico, y que consulten la documentación oficial de Node.js, Express y Sequelize en caso de ser necesario. Además, se sugiere que investiguen y hagan implementaciones adicionales para mejorar la solución básica y que practiquen la presentación oral con anterioridad para asegurarse de que se comprende y se transmite claramente el trabajo realizado.