

Trabajo Práctico Obligatorio Nro 3

Servidores Web

Desarrollar una API Rest para el dominio seleccionado en el TPO 2. La misma, debe cumplir con las siguientes características, aplicándola al dominio de cada grupo:

1. Proveer al menos 1 endpoint para crear o actualizar recursos. Ejemplo, para crear un usuario: **POST /api/usuarios**; para actualizar una casa **PUT /api/casas/:idCasa**
2. Proveer 1 endpoint para obtener 1 recurso (ejemplo: obtener 1 usuario por id). En caso de no encontrarse, el servidor debe devolver un **404**.
3. Proveer 1 endpoint para buscar varios recursos, permitiendo fijar la cantidad de elementos. Ejemplo: **GET /api/mascotas?cantidad=10&from=0** (trae las primeras 10 mascotas luego, si el usuario quiere las siguientes 10 mascotas, GET **/api/mascotas?cantidad=10&from=10**)
4. Cada **endpoint** debe validar los datos de entrada. Por ejemplo, si para crear una **mascota** es necesario el nombre de ella, el servidor debe validar que el campo "nombre" sea enviado y sea del tipo esperado (string). De lo contrario, deberá devolver un **400** explicando lo sucedido.
5. El servidor debe poder "servir" los archivos html/css/javascript desarrollados en el TPO2.
6. Actualizar los archivos del TPO 2 para poder consumir la API desarrollada en este TP.
7. De ser necesario, el **servidor** puede consumir datos de otro servidor REST (por ejemplo, consumir los datos del clima de una API abierta)
 - a. De hacer esto, deberá consumir y guardar en una variable global los datos y pedir nuevamente datos al servidor si pasaron más de N minutos, a fin de no cargar la API externa.
 - b. Si usa una api. considerar al leer los nuevos datos de no pisar todo, si no actualizar (Agregar o borrar) datos que tenga localmente.

Consideraciones:

1. Preferentemente, el servidor será desarrollado con Express.js. En caso negativo, deberá realizarse en Node directamente
2. Los endpoints deberán siempre empezar con **/api**
3. No es necesario tener una base de datos. Los recursos pueden estar almacenados en memoria (usando arreglos/listas, o la estructura de datos que consideren oportuna)