**BOOTCAMP DE FULL STACK ENGINEERING**

**DESAFÍO SEMANAL F3-N17**

1. **OBJETIVOS**

* Persistir datos del inventario en un archivo utilizando el módulo fs.
* Implementar la subida de imágenes con multer.
* Asociar cada zapato con una imagen opcional.
* Mantener y extender las funcionalidades CRUD de la API REST.
* Profundizar en el uso de middlewares y gestión de archivos en Node.js.

1. **CONSIGNA**

A partir del desafío anterior (F3-N16), se debe mejorar la API REST de la zapatería incorporando persistencia de datos mediante el módulo fs de Node.js y la gestión de archivos con Multer.

Para ello, los datos del inventario deben guardarse en un archivo data/shoes.json, de modo que cada vez que se cree, modifique o elimine un zapato, este archivo se actualice, y al iniciar la aplicación se lean los datos desde allí.

Además, al crear un zapato con el endpoint POST /api/shoes, el usuario podrá adjuntar una imagen en formatos permitidos como image/png o image/jpeg, la cual se almacenará en la carpeta public/images/, guardando en la estructura del zapato el nombre de dicha imagen.

Finalmente, se deberá implementar un nuevo endpoint para subir las imágenes: POST /api/shoes/upload, que devuelva el nombre de la imagen subida.

1. **INSTRUCCIONES**
2. Crear la carpeta src/data con un archivo denominado shoes.json (archivo inicial con un array vacío []).
3. Configurar manejador de archivos
4. Configurar gestor de subidas
5. Almacenar los datos de los zapatos en el archivo json
6. **ESTRUCTURA DE DATOS**

Cada zapato debe tener la siguiente estructura dentro del archivo shoes.json:

{

id: number,

brand: string,

model: string,

size: number,

price: number,

thumbnail: string

}