








Proyecto: Upload de Ficheros con Node.js

Descripción del Proyecto

Este proyecto consiste en desarrollar una aplicación en **Node.js** que permita la subida de ficheros a una carpeta específica (**/uploads**) desde una interfaz web generada por el propio backend. Además, incluirá funcionalidades para visualizar, eliminar y monitorear el espacio ocupado en disco por los archivos subidos.

Funcionalidades Principales

-  **Subida de archivos** a través de una interfaz web realizada en JavaScript.
-  **Visualización de los archivos subidos** con detalles como nombre, tamaño y fecha de subida.
-  **Eliminación de archivos** desde la misma interfaz.
-  **Monitoreo del almacenamiento**, mostrando el espacio ocupado por cada archivo y el total de la carpeta **/uploads**.
-  **Sistema de notificaciones (opcional)**: Permite registrar un correo electrónico para recibir un resumen de los archivos almacenados cada cierto tiempo. Se recomienda el uso de **SendGrid** para el envío de correos.
-  **Dockerización completa** del proyecto para facilitar su despliegue.
-  **README.md** detallado con instrucciones para la puesta en producción.

Tecnologías a Utilizar

- **Node.js** (v20+)
- **Express.js** para la gestión del servidor
- **Multer** para la subida de archivos
- **FS (File System)** para manejar archivos en el servidor
- **SendGrid API** (para el sistema de notificaciones)
- **Docker** para la contenedorización
- Cualquier otro que se necesite y consideres oportuno.








Requisitos del Proyecto

1. Crear un servidor en Node.js con Express.
2. Implementar una API REST que permita:
 - **Subir archivos** (**POST /upload**)
 - **Listar archivos** (**GET /files**)
 - **Eliminar archivos** (**DELETE /files/:filename**)

- **Obtener el espacio total ocupado en disco** (GET /storage)
- 3. Implementar un frontend minimalista que consuma la API y permita:
 - Subir archivos desde un formulario.
 - Ver una tabla con la lista de archivos subidos.
 - Eliminar archivos mediante un botón.
 - Visualizar el espacio total ocupado. Si es posible gráficamente.
- 4. Implementar un sistema de **notificaciones vía correo electrónico** para enviar un resumen de los archivos almacenados periódicamente.
- 5. Dockerizar el backend y frontend para un despliegue eficiente.
- 6. Incluir un **README.md detallado** con instrucciones claras de instalación y uso.



Rúbrica de Evaluación

Criterio	Descripción	Puntuación
 Subida de archivos	Permite subir archivos correctamente en una interfaz web y los almacena en /uploads	2.5 pts
 Visualización de archivos	Lista los archivos en una interfaz amigable mostrando nombre y tamaño	2 pts
 Eliminación de archivos	Permite eliminar archivos correctamente desde la interfaz	1 pts
 Monitoreo del almacenamiento	Muestra el tamaño de cada archivo y el total ocupado en disco	2 pts
 Notificaciones vía correo	Implementa el envío de resumen de archivos a un correo cada cierto tiempo	1 pts
 Dockerización	El backend y frontend están correctamente dockerizados	1 pts
 Documentación	Instrucciones detalladas en README.md para instalación y uso	0.5 pts
TOTAL	10 pts	



Ejemplo de Interfaz

(Se recomienda incluir una captura de pantalla de la interfaz final para referencia)