RESUMIDOR IA, SPEECH TO TEXT

Solanyi Mera - 2206256
Jose Alejandro Catacoli - 2210370
Jhon Hector Aguilar - 219143

INTRODUCCIÓN

En el contexto educativo actual, tanto en modalidades presenciales como virtuales, es común que los estudiantes enfrenten dificultades para registrar y organizar adecuadamente la información impartida durante las clases. Factores como la velocidad con la que se desarrolla el contenido, la complejidad de los temas tratados y la imposibilidad de tomar apuntes completos en tiempo real, afectan negativamente la comprensión y el repaso posterior del material académico.

OBJETIVOS.

Desarrollar una aplicación web que permite grabar clases, transcribir el audio en tiempo real, generar resúmenes automáticos, ofrecer soporte para fórmulas matemáticas y almacenar toda la información en una base de datos, con el fin de mejorar el acceso y organización del contenido académico para los estudiantes.

1. Implementar un sistema de grabación de audio en la web que permita al usuario capturar el contenido de clases presenciales o virtuales directamente desde el navegador.

2. Integrar una herramienta de reconocimiento de voz en tiempo real para transcribir automáticamente el audio grabado en formato de texto.

3. Utilizar la API de Gemini para generar resúmenes automáticos y respuestas conversacionales, facilitando la comprensión del contenido transcrito.

4. Incorporar un editor visual con soporte para MathJax, que permita visualizar y editar resúmenes con fórmulas matemáticas, ofreciendo una herramienta útil para materias técnicas y científicas.

LA APLICACIÓN

PROCESO

INTEGRACIÓN DE LA API WEB SPEECH

Se utilizó la Web Speech API provista por los navegadores modernos para el reconocimiento de voz en tiempo real. Esta API permite capturar el audio desde el micrófono del usuario y transcribirlo automáticamente a texto

INTEGRACIÓN DE LA API DE GEMINI

Para el procesamiento del resumen automático de texto, se utilizó la Gemini API (modelo de lenguaje desarrollado por Google). El texto transcrito se envía a esta API mediante una solicitud POST, y la respuesta incluye un resumen generado que es luego mostrado al usuario

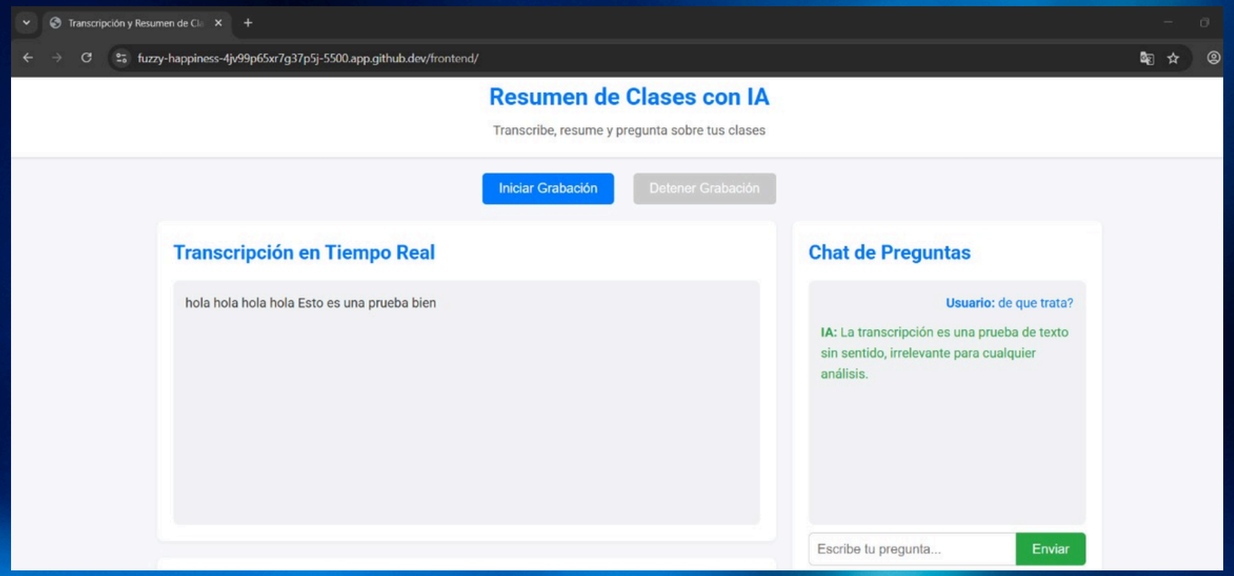
CREACIÓN DEL EDITOR VISUAL CON SOPORTE PARA MATHJAX

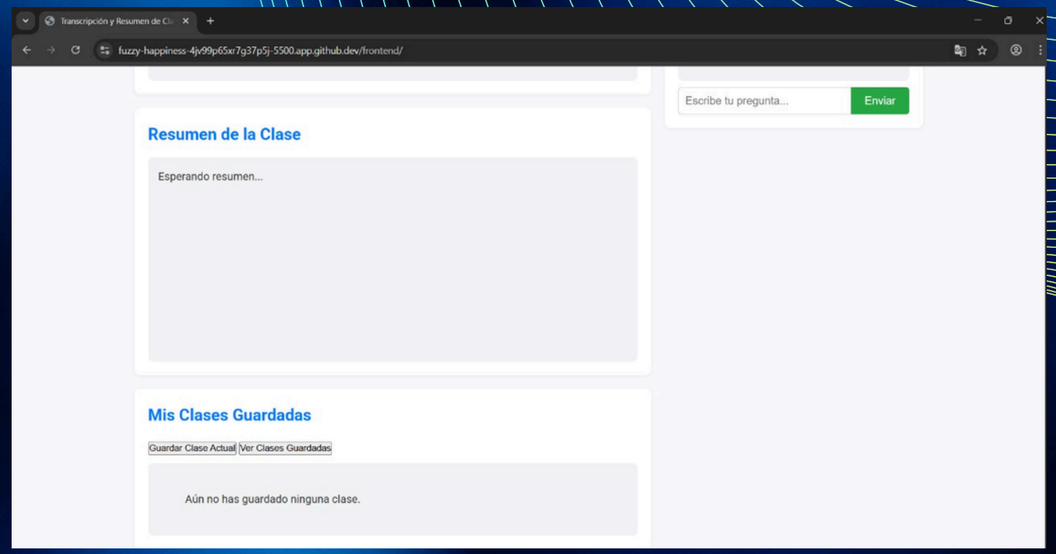
Se implementó un editor visual personalizado que permite al usuario editar y visualizar el resumen generado. Para brindar soporte a fórmulas matemáticas, se integró MathJax, una biblioteca JavaScript

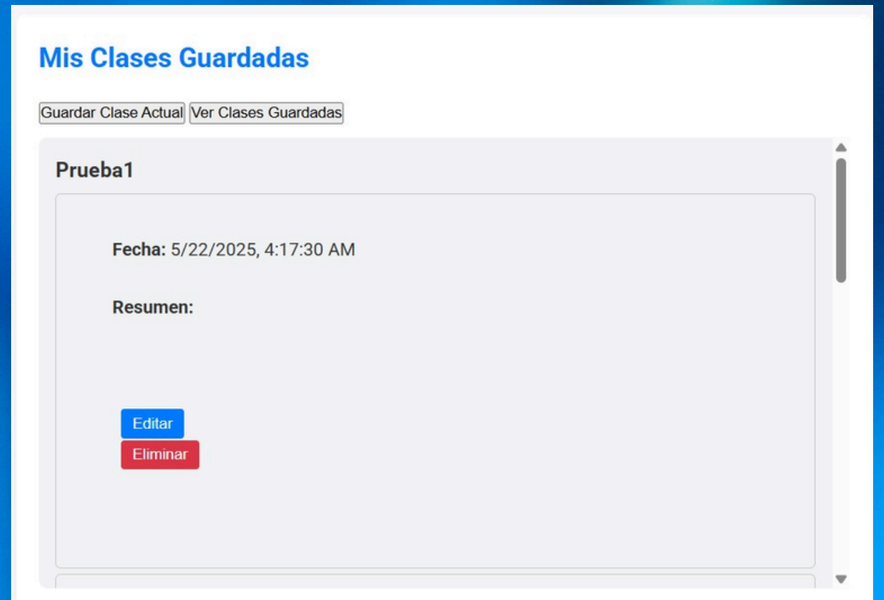
DESARROLLO DE LA INTERFAZ

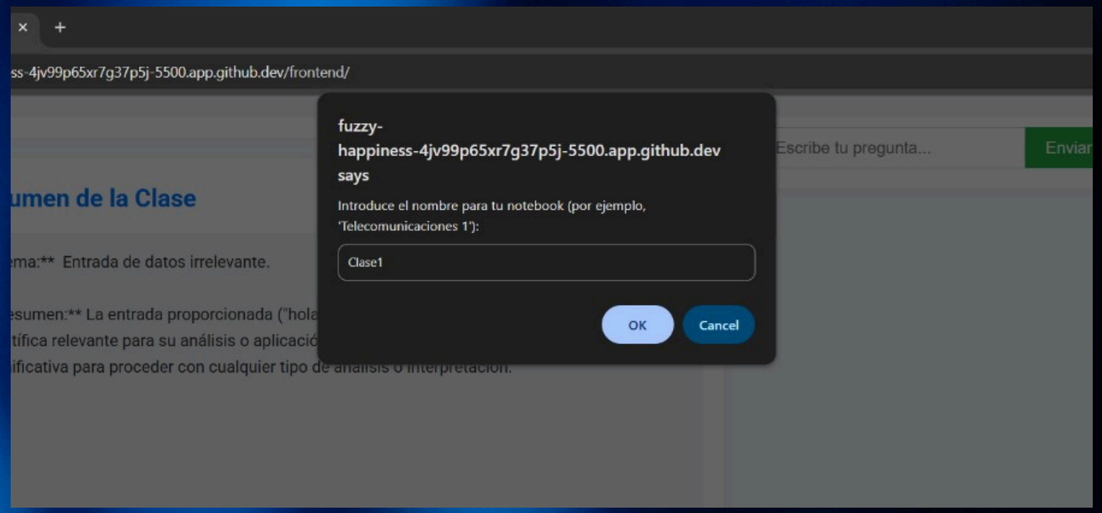
La interfaz de la aplicación fue desarrollada utilizando HTML, CSS y JavaScript puro, sin el uso de frameworks como React o Vue. Se implementaron componentes funcionales como botones de grabación, cuadros de texto, y áreas de visualización para las transcripciones y resúmenes.

SOLUCION PROPUESTA

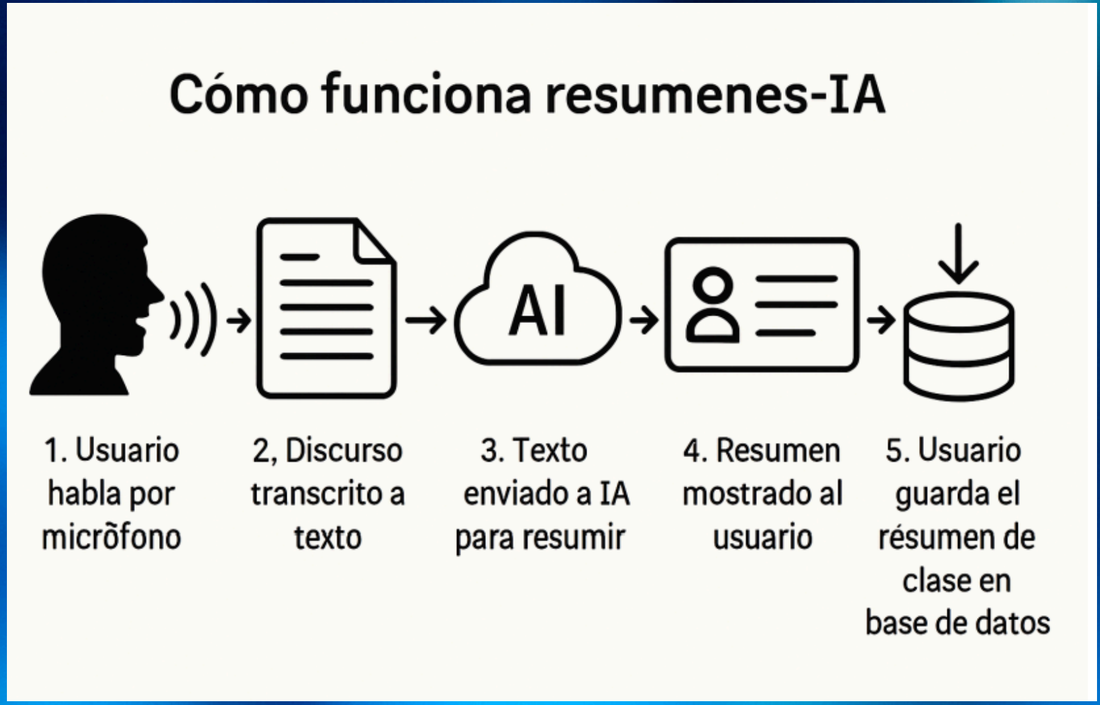


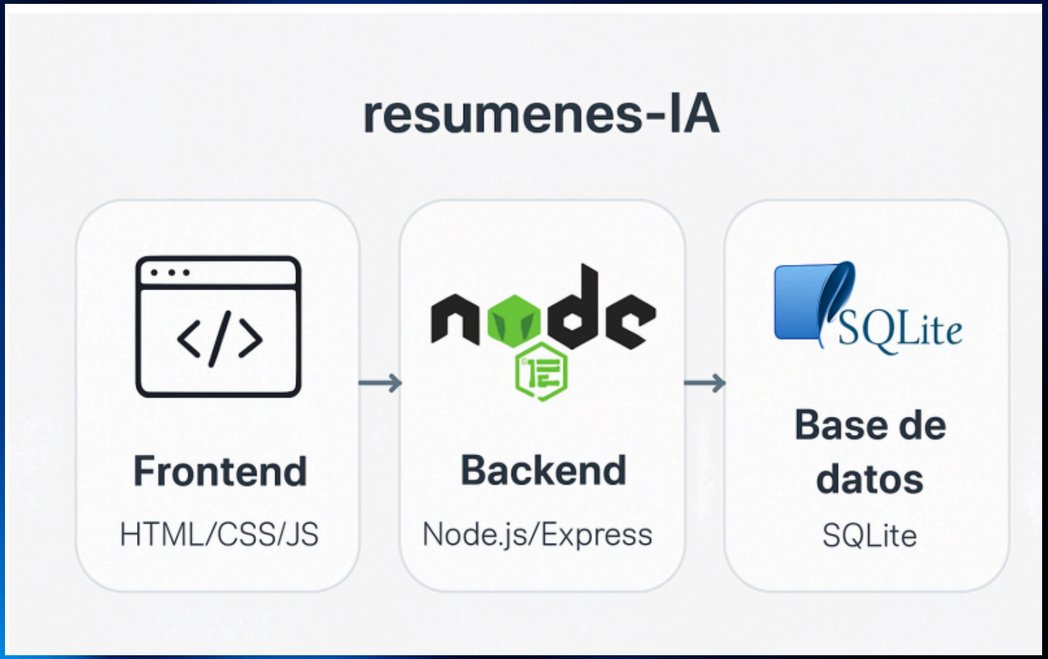






SOLUCION PROPUESTA





FRONTED

El frontend es la parte visual de una aplicación web, es decir, todo lo que el usuario ve e interactúa en la pantalla. Incluye la estructura de la página, los estilos visuales y la lógica del navegador que responde a las acciones del usuario.

1. index.html:
   Archivo principal HTML que muestra la página web.

2. style.css:
   Archivo con los estilos visuales (colores, tamaños, posiciones) del frontend.

3. script.js:
   Archivo con el código JavaScript del frontend. En este caso, está modificado para no usar localStorage sino hacer llamadas a la API REST del backend (usando fetch).

BACKEND

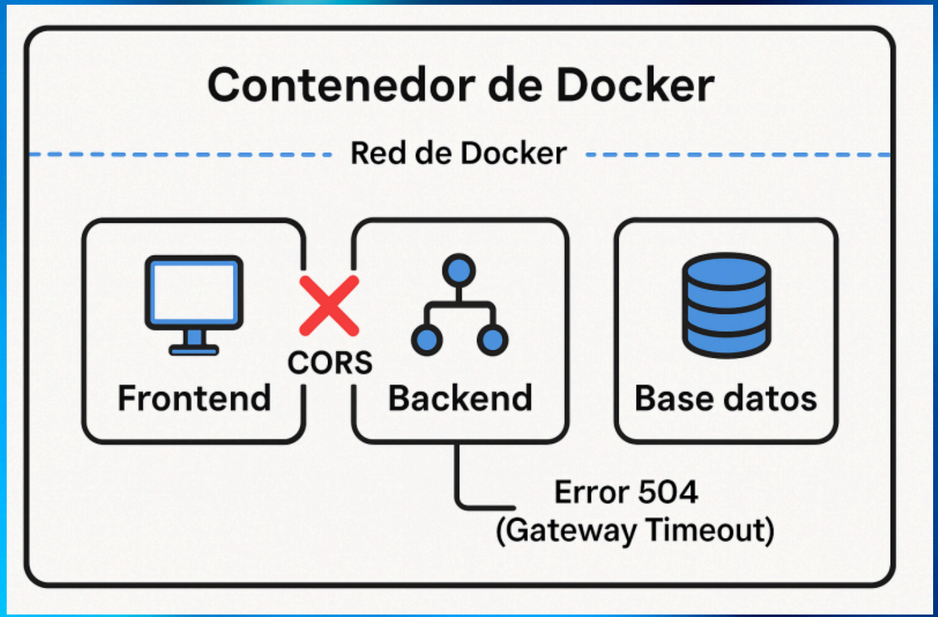
l backend se implementó usando Node.js con Express para crear una API RESTful que maneja la creación, consulta, actualización y eliminación de notebooks y clases, además de conectar con una base de datos SQLite para persistencia. Este diseño modular y contenerizado garantiza estabilidad, escalabilidad y facilita futuras integraciones con sistemas de inteligencia artificial para procesar y resumir contenido de clases grabadas.

1. Config/config.json
2. Models/database.js
3. Node_modules
4. package.json
5. Dockerfile

DOCKER COMPOSE

DOCKER ES UNA PLATAFORMA QUE PERMITE CREAR, EJECUTAR Y ADMINISTRAR APLICACIONES DENTRO DE CONTENEDORES.
UN CONTENEDOR ES UNA UNIDAD LIGERA Y PORTÁTIL QUE INCLUYE TODO LO NECESARIO PARA QUE UNA APLICACIÓN FUNCIONE: EL CÓDIGO, LAS DEPENDENCIAS, LIBRERÍAS Y EL SISTEMA DE ARCHIVOS.

DOCKER COMPOSE ES UNA HERRAMIENTA QUE PERMITE DEFINIR Y EJECUTAR MÚLTIPLES CONTENEDORES DOCKER A LA VEZ, USANDO UN ARCHIVO DE CONFIGURACIÓN LLAMADO DOCKER-COMPOSE.YML.
EN LUGAR DE EJECUTAR CADA CONTENEDOR MANUALMENTE, COMPOSE PERMITE ORQUESTARLOS TODOS JUNTOS



¡GRACIAS!