**Desarrollo de un Componente de Grabación de Pantalla para Plataforma Educativa: Integración Backend-Frontend con NestJS y MongoDB**

**Bruno Aguirre Mayerlis Paola**

[mbrunoaguirre53@correo.unicordoba.edu.co](mailto:mbrunoaguirre53@correo.unicordoba.edu.co) **Moreno Zapata Yerson David** ymorenozapata@correo.unicordoba.edu.co

# Briceño Ojeda Cesar Luis

[cbricenoojeda42@correo.unicordoba.edu.co](mailto:cbricenoojeda42@correo.unicordoba.edu.co)

**Flórez Causil Arturo José**

# [aflorezcausil@correo.unicordoba.edu.co](mailto:aflorezcausil@correo.unicordoba.edu.co) Álvarez Ortega José Fernando

[jalvarezortega69@correo.unicordoba.edu.co](mailto:jalvarezortega69@correo.unicordoba.edu.co)

**TUTORES:**ALEXANDER ENRIQUE TOSCANO RICARDO.  
RAÚL EMIRO TOSCANO MIRANDA

# RESUMEN

Este artículo presenta el diseño e implementación de un componente de software que permite la grabación de pantalla con audio, desarrollado como parte de una plataforma educativa. El componente ofrece funcionalidades como grabación por zonas, control de audio. Se utilizó una arquitectura basada en NestJS para el backend, MongoDB Atlas para la persistencia de datos y una estructura modular que permite escalabilidad y mantenibilidad. Se describen las fases del desarrollo, las pruebas realizadas y los beneficios esperados en contextos educativos. La solución busca fomentar el aprendizaje activo mediante la creación y gestión autónoma de contenidos audiovisuales.

**Palabras clave**: API REST, backend, frontend, grabación de pantalla, MongoDB, NestJS, desarrollo modular, software educativo

# ABSTRACT

This article presents the design and implementation of a software component that enables screen recording with audio, developed as part of an educational platform. The component offers features such as zone-based recording, audio control. A NestJS-based architecture was used for the backend, MongoDB Atlas for data persistence, and a modular structure that allows for scalability and maintainability. The development phases, testing, and expected benefits in educational contexts are described. The solution seeks to foster active learning through the autonomous creation and management of audiovisual content.

**Key words**: REST API, backend, frontend, screen recording, MongoDB, NestJS, modular development, educational software.

# INTRODUCCIÓN

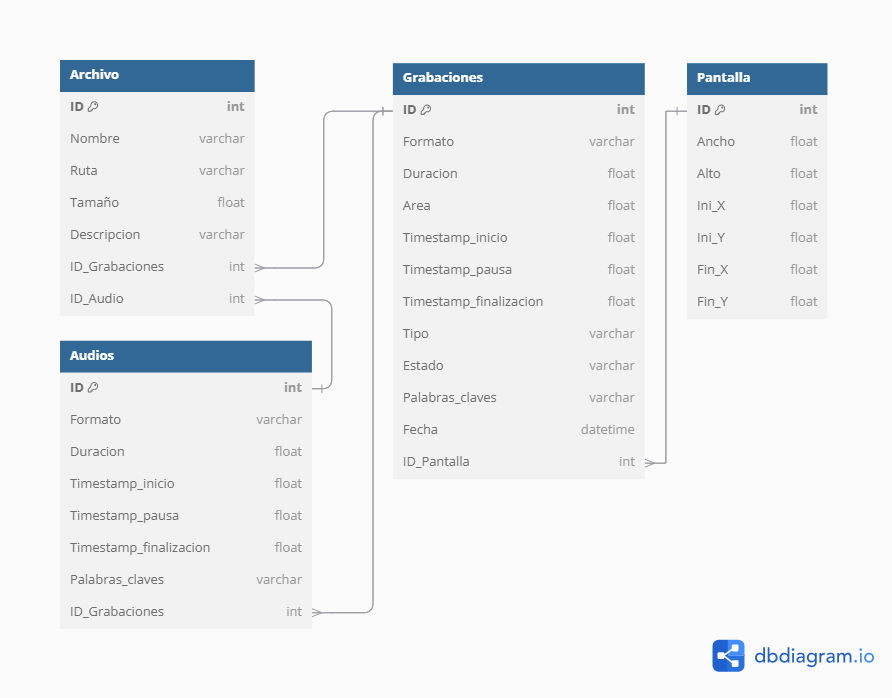
La grabación de pantalla es una herramienta clave en entornos educativos virtuales, permitiendo crear contenidos interactivos, realizar demostraciones y documentar procesos. Frente a esta necesidad, se desarrolló un componente de grabación integrado a una plataforma de administración de contenidos, apuntando a mejorar la comunicación, el aprendizaje y la documentación entre usuarios.

A diferencia de herramientas externas, esta solución fue diseñada para integrarse de forma nativa, con control sobre sus funcionalidades y una arquitectura flexible. Su construcción se dio en tres etapas: diseño y análisis, desarrollo del backend y desarrollo del frontend. Desde su concepción se contemplaron funcionalidades como la posibilidad de añadir títulos, créditos y marcas de agua a las grabaciones, así como opciones para ajustar brillo, contraste y volumen.

# METODOLOGIA.

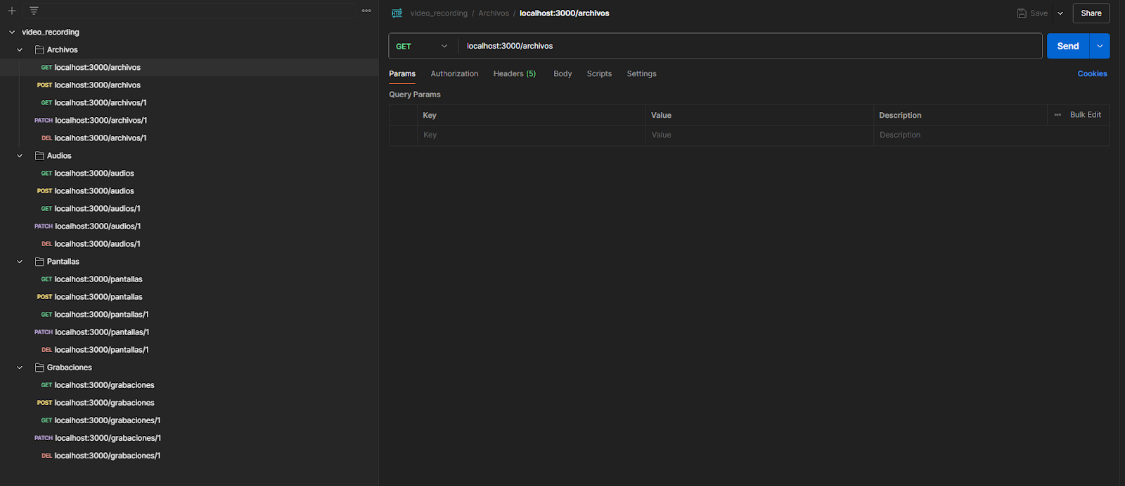
Se trabajó bajo un enfoque incremental, dividiendo el desarrollo en tres fases académicas. Se usaron metodologías inspiradas en MODESEC, combinadas con buenas prácticas de ingeniería de software.

* **Etapa 1 - Análisis y Diseño:** Se establecieron los requisitos funcionales y no funcionales. Se modelaron entidades como grabación, pantalla, archivo y audio. En esta fase inicial, se modelaron funcionalidades esenciales como grabar pantalla completa, grabar por zonas y capturar audio. También se documentaron casos de uso clave como pausar y guardar grabaciones *(ver figura 1)*



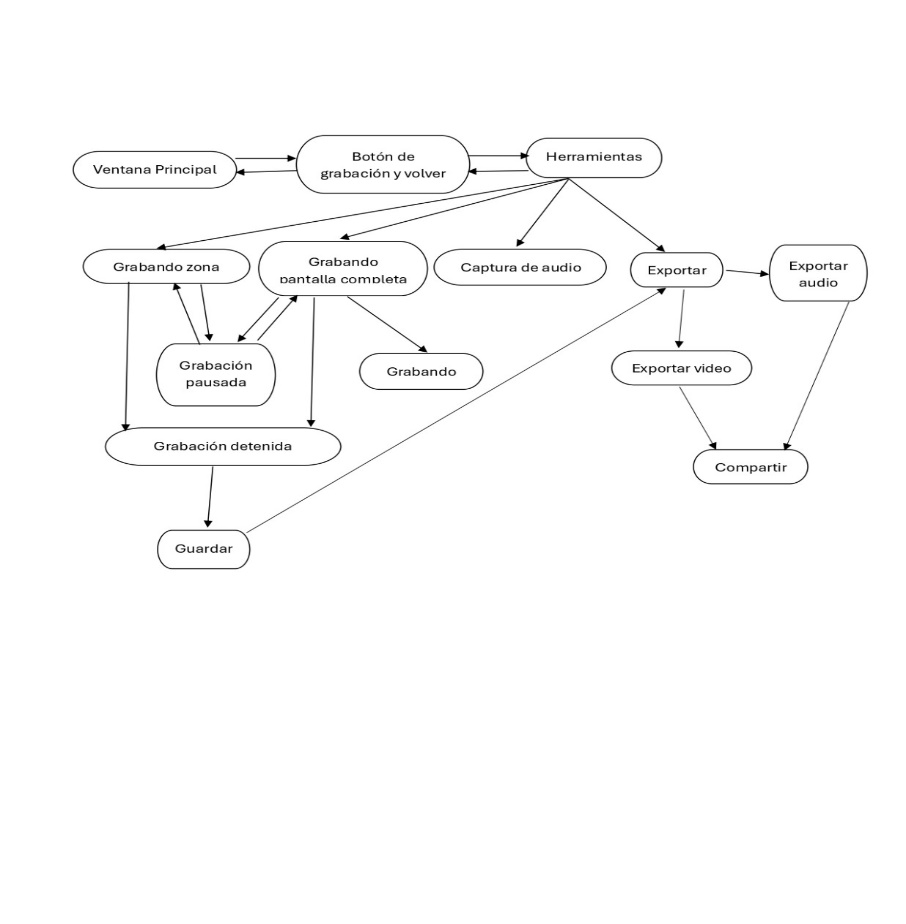
*Figura 1. Diagrama relacio*nal *con sus respectivas colecciones*

* **Etapa 2 - Backend:** Se implementó un backend REST con NestJS y MongoDB Atlas. Los módulos fueron organizados por responsabilidad: controladores, servicios y repositorios. Los datos se almacenaron en colecciones MongoDB estructuradas y se desarrollaron endpoints CRUD para cada entidad. Las pruebas se realizaron con Postman *(ver figura 2).*



Fgura 2. Creación y pruebas de la API con postman

* **Etapa 3 - Frontend:** Se diseñó una interfaz de usuario responsiva y clara, con pantallas para grabar, gestionar archivos y compartir videos. Se planificó la integración con la API para garantizar una experiencia fluida. *(ver figura 3)*



*Figura 3. Mapa de navegacin del sistema*

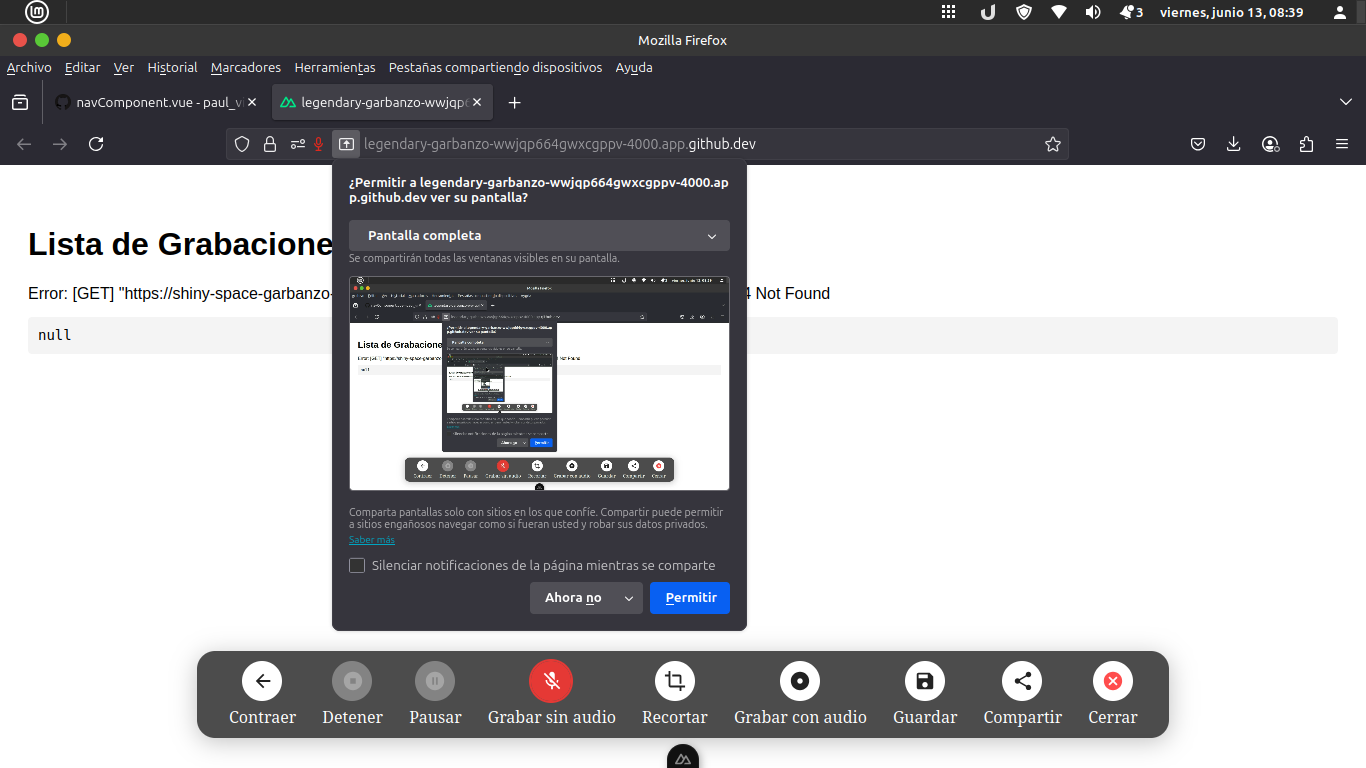
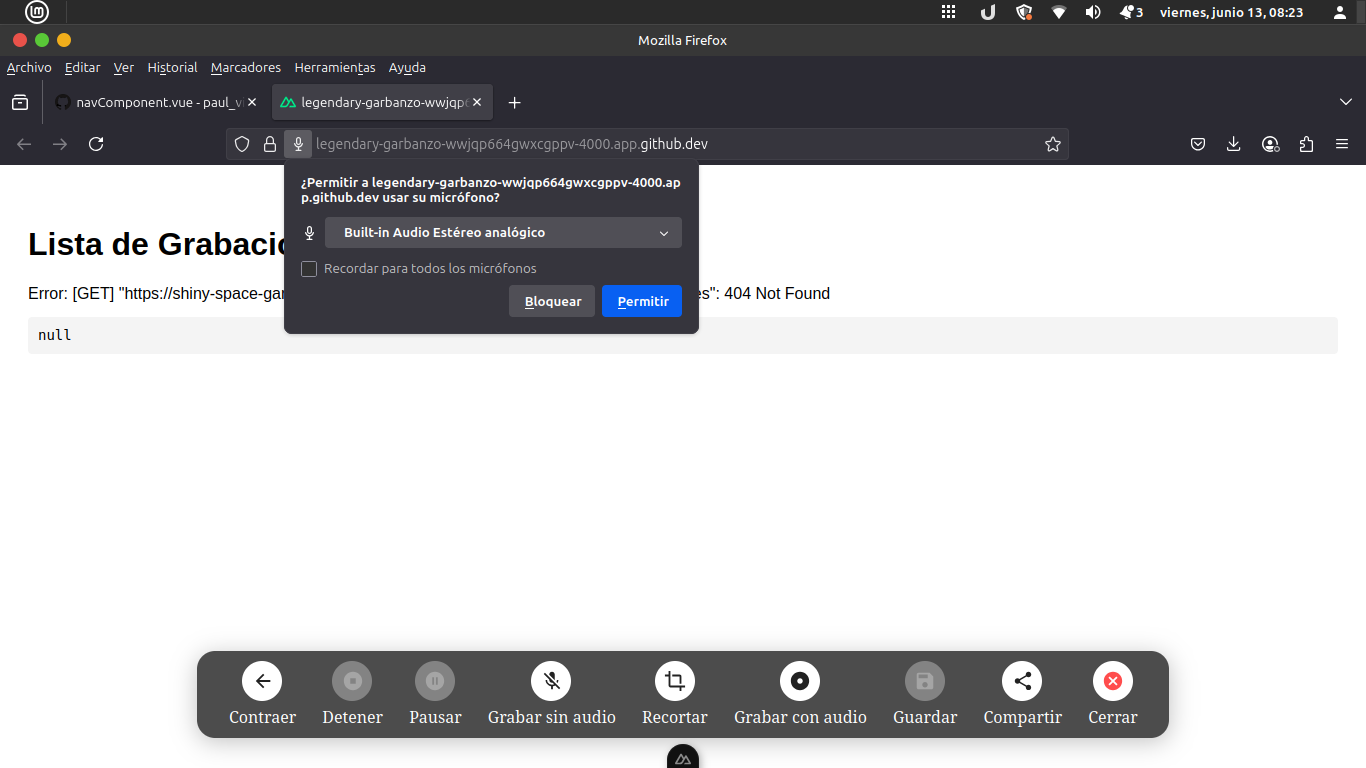
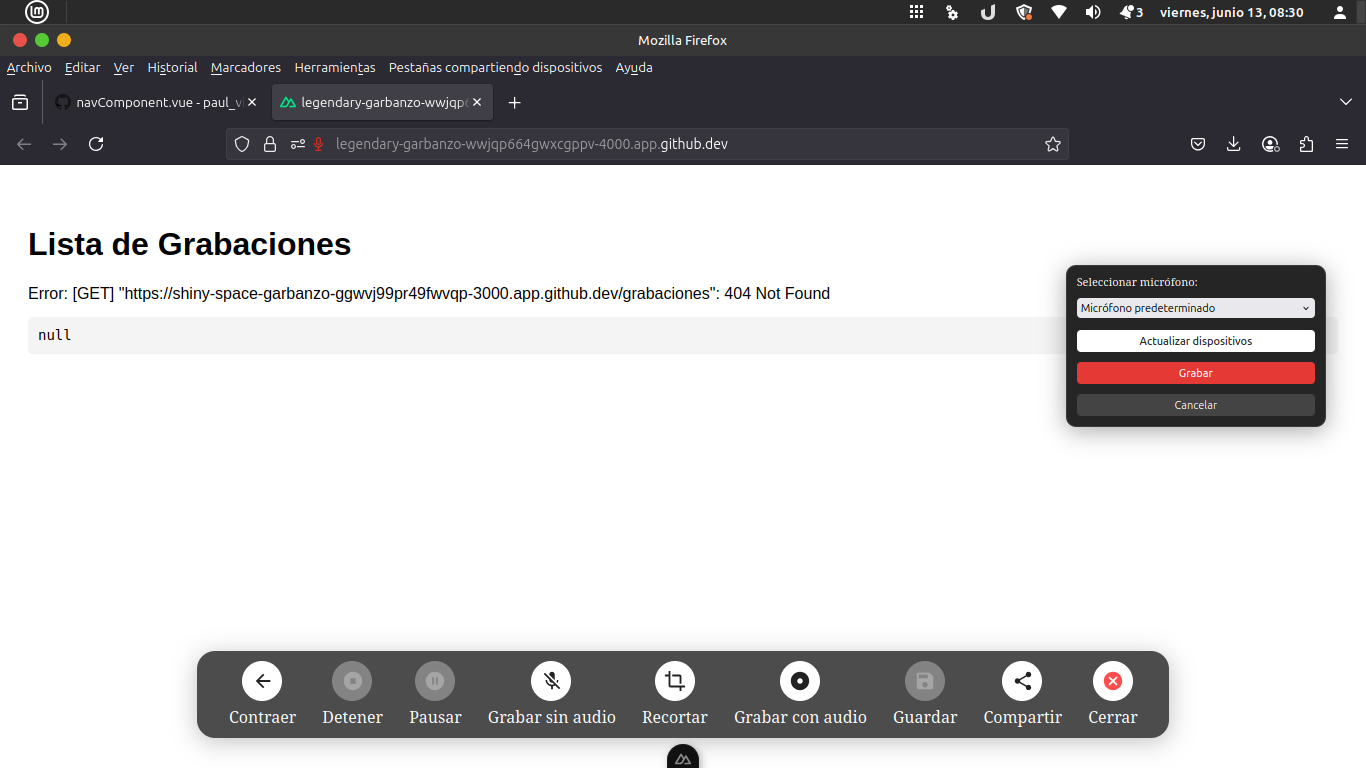
1. **RESULTADOS**

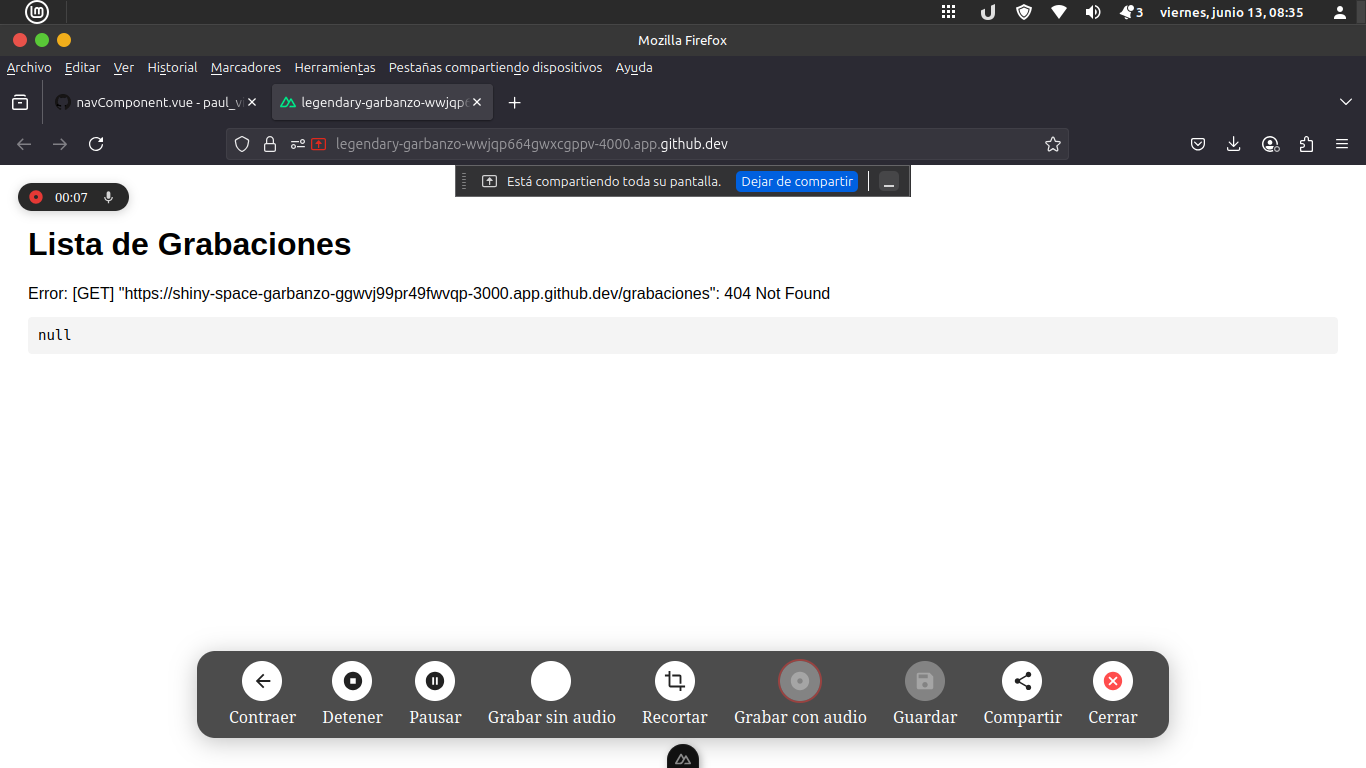
Durante las pruebas con Postman se verificaron correctamente los endpoints de cada módulo (grabaciones, audios, pantallas y archivos), evidenciando un comportamiento consistente en las operaciones CRUD. La base de datos, estructurada con MongoDB Atlas, permitió almacenar volúmenes de metadatos de manera flexible, favoreciendo la escalabilidad del sistema y la posibilidad de expansión futura. Por su parte, la interfaz gráfica permitió al usuario gestionar las grabaciones de forma intuitiva: iniciar, finalizar, guardar y descargar archivos, sin requerir conocimientos técnicos avanzados.

El componente desarrollado integra un conjunto de funcionalidades esenciales que lo convierten en una herramienta poderosa, práctica y educativa:

* **Grabación de pantalla completa con audio**

El usuario puede capturar todo su escritorio o escoger entre el navegador, alguna pestaña o programa que esté abierto, adaptando la grabación a sus necesidades pedagógicas o técnicas. *(Ver figura 4).* Posteriormente dependiendo de la necesidad del usuario, puede grabar con audio o sin audio. Permite grabar la voz del usuario o los sonidos que emite el equipo, garantizando flexibilidad para la narración de contenidos, explicaciones o demostraciones. Si graba con audio, el sistema le pedirá permiso para capturar audio *(ver figura 5)* y acto seguido, saldrá un selector de micrófonos disponibles *(ver figura 6).*  Finalmente, una vez seleccionado el dispositivo de entrada, el sistema comienza la grabación y se mostrará un contador en la parte superior indicando la duración de la grabación *(ver figura 7).*

  *Figura 4. Permiso para grabar Figura 5. Permiso para capturar audio. Figura 6. Selector de pantalla* *Micrófonos.*



*Figura 7. Componente grabando pantalla completa con audio*

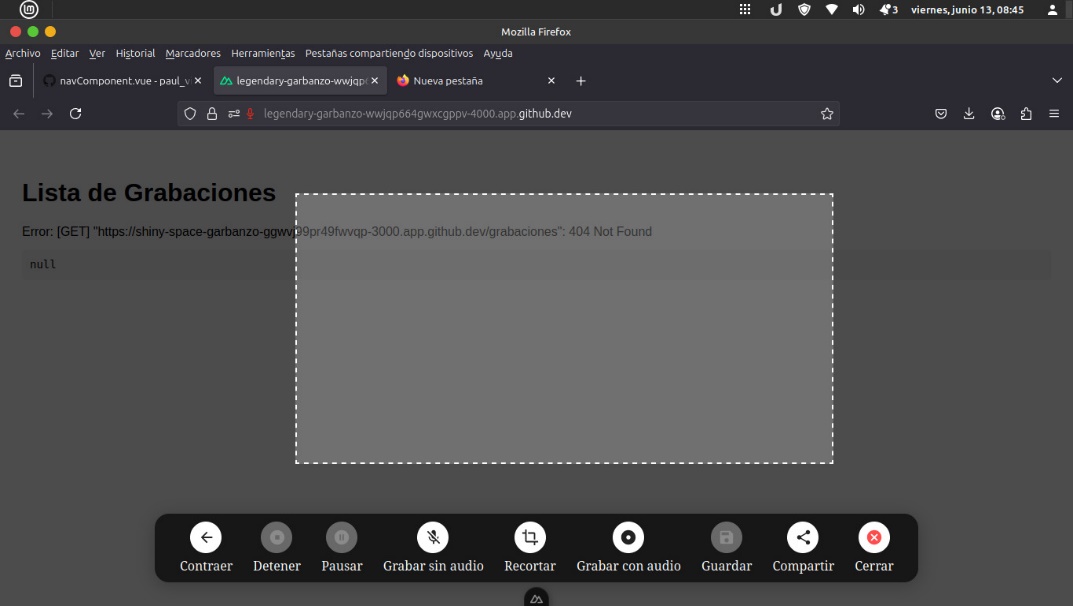
* **Grabación de pantalla completa sin audio**

Esta funcionalidad permite al usuario capturar visualmente todo lo que ocurre en su pantalla, ya sea de forma completa o en una zona específica, **excluyendo cualquier fuente de sonido**. Es especialmente útil en contextos donde el contenido visual es suficiente —como demostraciones técnicas, registros de navegación, tutoriales silenciosos o evidencias de procesos— sin necesidad de narración o sonido ambiente.El resultado es un archivo de video limpio, en alta resolución, optimizado para tareas donde **la imagen es el mensaje principal**.

Al igual que la opción de grabar con audio, el sistema le solicita el permiso de capturar pantalla *(ver figura 4)* pero no el de audio. Una vez que se selecciona el contenido a grabar empieza con la grabación *(ver figura 7)*, igual que antes pero el icono de micrófono en el contador cambia.

### **Grabación de pantalla por zona**

Esta funcionalidad permite al usuario definir y capturar únicamente una parte específica de la pantalla. Mediante un selector visual intuitivo, se delimita el área de interés antes de iniciar la grabación, enfocando exclusivamente lo que se desea mostrar. Esto es útil para grabar una ventana puntual, un fragmento de una presentación o una actividad concreta, evitando distracciones innecesarias del resto del entorno digital. Esta modalidad optimiza el contenido generado, haciéndolo más preciso y centrado.



*Figura 8. Pantalla de grabación por zonas*

### **Guardar grabación**

Una vez finalizada la sesión, el sistema ofrece la opción de guardar el archivo generado. Las grabaciones realizadas, incluyendo las que se hacen por zona o sin audio, se exportan en formato **.webm**, un estándar abierto y eficiente para archivos multimedia. Este formato permite una alta calidad visual con bajo peso de archivo, ideal para su distribución en plataformas web, sistemas de mensajería o almacenamiento en la nube. El sistema garantiza compatibilidad con navegadores modernos como Chrome, Firefox y Edge, permitiendo la reproducción inmediata sin necesidad de convertir el video a otros formatos. Además, este formato favorece la rápida descarga y economiza espacio en disco

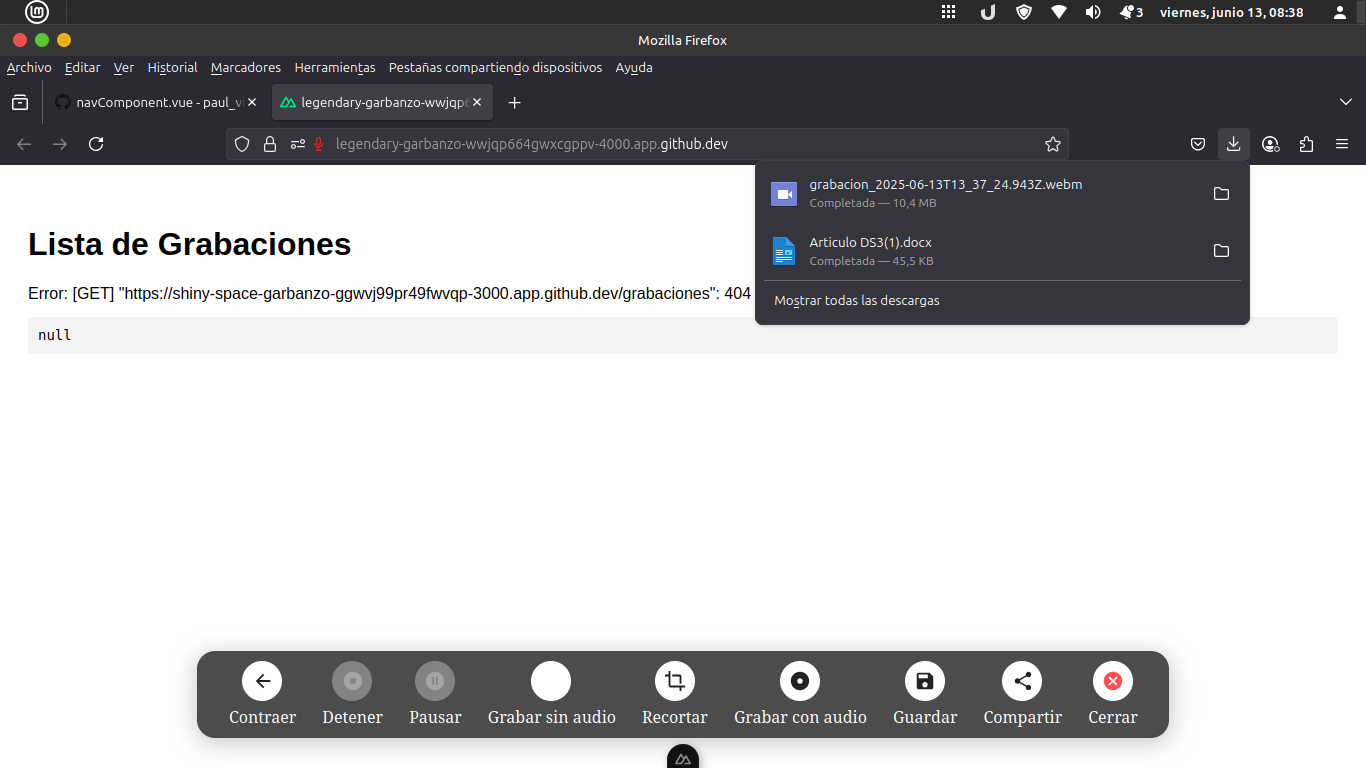


Figura 9. Archivo descargado en formato webm

1. **CONCLUSIONES**

Se establecieron las bases de un componente de grabación de pantalla funcional, integrable y diseñado para el contexto educativo. Esta solución facilita a los usuarios la creación, gestión y compartición de contenido visual, fomentando la autonomía y el aprendizaje activo. Las siguientes etapas se enfocarán en la implementación completa de las funcionalidades, correccion de errores e implementación de nuevas herramientas.

# REFERENCIAS

NestJS documentation. (s.f.). Recuperado de <https://docs.nestjs.com>

MongoDB Atlas documentation. (s.f.). Recuperado de <https://www.mongodb.com/cloud/atlas>

Pineres, M. C. (2020). MODESEC: Metodología de desarrollo de software educativo colaborativo. Montería, Colombia: Universidad de Córdoba.