

UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL

CARRERA DE SOFTWARE

Tema:

DESARROLLO DE UN SISTEMA PARA LA GENERACIÓN DE CONTENIDO AUDIOVISUAL UTILIZANDO RECURSOS BASADOS EN INTELIGENCIA ARTIFICIAL Y SU INCIDENCIA EN LOS TIEMPOS DE PRODUCCIÓN

Trabajo de titulación modalidad Proyecto de Investigación, presentado previo a la obtención del título de Ingeniero de Software

ÁREA: Software

LÍNEA DE INVESTIGACIÓN: Desarrollo de software

AUTOR: Daniel Gerardo Galarza Tamayo

TUTOR: Ing. Santiago David Jara Moya, Mg.

Ambato - Ecuador

febrero - 2025

APROBACIÓN DEL TUTOR

En calidad de tutor del trabajo de titulación con el tema: DESARROLLO DE UN SISTEMA PARA LA GENERACIÓN DE CONTENIDO AUDIOVISUAL UTILIZANDO RECURSOS BASADOS EN INTELIGENCIA ARTIFICIAL Y SU INCIDENCIA EN LOS TIEMPOS DE PRODUCCIÓN, desarrollado bajo la modalidad Proyecto de Investigación, por el señor Daniel Gerardo Galarza Tamayo, estudiante de la Carrera de Software, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 17 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.3 del instructivo del reglamento referido.

Ambato, febrero 2025.

SANTIAGO DAVID JARA MOYA

Firmado digitalmente por SANTIAGO DAVID JARA MOYA
Fecha: 2025.02.05 08:38:56 -05'00'

Ing. Santiago David Jara Moya, Mg.

TUTOR

AUTORÍA

El presente trabajo de titulación con el tema: DESARROLLO DE UN SISTEMA PARA LA GENERACIÓN DE CONTENIDO AUDIOVISUAL UTILIZANDO RECURSOS BASADOS EN INTELIGENCIA ARTIFICIAL Y SU INCIDENCIA EN LOS TIEMPOS DE PRODUCCIÓN es absolutamente original, auténtico y personal y ha observado los preceptos establecidos en la Disposición General Quinta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, febrero 2025.

Firmado electrónicamente por:
DANIEL GERARDO GALARZA TAMAYO

Daniel Gerardo Galarza Tamayo

C.C. 1851077451

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato para que reproduzca total o parcialmente este trabajo de titulación dentro de las regulaciones legales e institucionales correspondientes. Además, cedo todos mis derechos de autor a favor de la institución con el propósito de su difusión pública, por lo tanto, autorizo su publicación en el repositorio virtual institucional como un documento disponible para la lectura y uso con fines académicos e investigativos de acuerdo con la Disposición General Cuarta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato.

Ambato, febrero 2025.

Firmado electrónicamente por:
DANIEL GERARDO
GALARZA TAMAYO

Daniel Gerardo Galarza Tamayo

C.C. 1851077451

AUTOR

APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del informe final del trabajo de titulación presentado por el señor Daniel Gerardo Galarza Tamayo, estudiante de la Carrera de Software, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado DESARROLLO DE UN SISTEMA PARA LA GENERACIÓN DE CONTENIDO AUDIOVISUAL UTILIZANDO RECURSOS BASADOS EN INTELIGENCIA ARTIFICIAL Y SU INCIDENCIA EN LOS TIEMPOS DE PRODUCCIÓN nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 19 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.4 del instructivo del reglamento referido. Para cuya constancia suscribimos, conjuntamente con el señor Presidente del Tribunal.

Ambato, febrero 2025.

Firmado electrónicamente por:
FRANKLIN OSWALDO
MAYORGA MAYORGA

Ing. Franklin Oswaldo Mayorga Mayorga Mg.
PRESIDENTE DEL TRIBUNAL

Firmado electrónicamente por:
RUBEN EDUARDO
NOGALES PORTERO

Ing. Ruben Nogales Portero Mg.
PROFESOR CALIFICADOR

Firmado electrónicamente por:
LEONARDO DAVID
TORRES VALVERDE

Ing. Leonardo Torres Valverde Mg.
PROFESOR CALIFICADOR

DEDICATORIA

A mi madre, Margarita, gracias por ser mi confidente, mi guía y mi refugio en todo momento. Tu amor incondicional y tu sabiduría me han enseñado el verdadero significado de la entrega y el cariño. A mi padre Raúl, aunque no está físicamente, sé que desde el cielo me protege y se siente orgulloso de mí, siendo siempre una inspiración en mi vida.

A Gonzalo, mi padre de crianza, quiero agradecerle por educarme con valores firmes y enseñarme, con su ejemplo, el significado del trabajo, la responsabilidad y el amor incondicional. Su paciencia y dedicación han sido fundamentales para formar la persona que soy hoy.

A mis hermanos David, Fernando, Gabriela y Bolívar, gracias por ser mi fortaleza y mi mayor inspiración. David, gracias por siempre entenderme y engreírme con cariño, haciéndome sentir especial. Fernando, por impartir su sabiduría y afecto, siendo un ejemplo cercano de esfuerzo y dedicación. Gabriela, gracias por tu carácter fuerte, que me ayudó a distinguir lo correcto y lo incorrecto. Y Bolívar, con quien compartí mi niñez, le agradezco profundamente por su protección y cariño. Gracias a todos por actuar como mis segundos padres, llenándome de amor y guía.

A todos mis amigos, desde los más cercanos que han estado conmigo en cada paso importante, hasta aquellos que he conocido a lo largo de mi vida, les agradezco por los momentos compartidos, las risas y el apoyo incondicional. Su amistad ha sido un regalo invaluable que atesoro profundamente.

Daniel Gerardo Galarza Tamayo

AGRADECIMIENTOS

Agradezco profundamente a Dios por regalarme la vida, la salud y la sabiduría necesarias para enfrentar cada desafío. Su guía ha sido mi refugio en los momentos difíciles y mi fortaleza en los días de incertidumbre. Gracias por bendecirme con oportunidades y por rodearme de personas que han sido pilares en mi camino.

A todas las personas que han aportado a mi crecimiento personal A todas las personas que han dejado una huella en mi vida, gracias por su apoyo, consejos y enseñanzas. Desde mi familia y amigos hasta aquellos que, en algún momento, compartieron conmigo palabras de aliento o experiencias valiosas, su impacto ha sido fundamental para mi desarrollo personal. Cada uno de ustedes ha contribuido a que hoy sea quien soy, y por eso les estaré siempre agradecido.

Un agradecimiento especial a mi tutor, Santiago Jara, por su orientación, paciencia y enseñanza a lo largo de este proceso. Su apoyo incondicional y compromiso con mi formación académica han sido esenciales, no solo en mi investigación, sino también en mi desarrollo a lo largo de la carrera universitaria. Gracias por su confianza y por ser un ejemplo de dedicación y excelencia.

Daniel Gerardo Galarza Tamayo

ÍNDICE GENERAL DE CONTENIDOS

PORTADA i

APROBACIÓN DEL TUTOR ii

AUTORÍA iii

DERECHOS DE AUTOR iv

APROBACIÓN DEL TRIBUNAL DE GRADO v

DEDICATORIA vi

AGRADECIMIENTO vii

ÍNDICE GENERAL DE CONTENIDOS x

ÍNDICE DE TABLAS xi

ÍNDICE DE FIGURAS xiii

ÍNDICE DE ANEXOS xiv

ABSTRACT xvi

CAPÍTULO I. MARCO TEÓRICO 1

1.1 Tema de investigación 1

1.1.1 Planteamiento del problema 1

1.2 Antecedentes investigativos 2

1.3 Fundamentación teórica 10

1.3.1 Large Language Models (LLM) 10

1.3.2 Text To Speech (TTS) 10

1.3.3 Editores de video 11

1.3.4 Framework de Desarrollo de Escritorio 12

1.3.5 Lenguajes de Programación 13

1.3.6 Metodología Ágil 13

1.4 Objetivos 14

1.4.1 Objetivo general 14
1.4.2 Objetivos específicos 15

CAPÍTULO II. METODOLOGÍA **16**

2.1 Materiales 16
2.2 Métodos 16
2.2.1 Modalidad de la investigación 16
2.2.2 Población y muestra 18
2.2.3 Recolección de información 18
2.2.4 Procesamiento y análisis de datos 29

CAPÍTULO III. RESULTADOS Y DISCUSIÓN **32**

3.1 Análisis del proceso actual 32
3.1.1 Factores que afectan la calidad del contenido 34
3.2 Análisis de herramientas de desarrollo 35
3.2.1 Análisis y selección de tecnologías 35
3.3 Desarrollo de la propuesta 45
3.3.1 Planificación de requerimientos 46
3.3.2 Diseño de usuario 49
3.3.3 Construcción 64
3.3.4 Pruebas de aceptación 80
3.4 Comparativa de Tiempos: Manual vs. Automatizado 83
3.4.1 Etapas Consideradas 84
3.4.2 Metodología 84
3.4.3 Tiempos Promedios por Etapa 85
3.4.4 Cálculo de Indicadores 85
3.5 Análisis de la Productividad Antes y Después de la Implementación . . . 86
3.5.1 Cantidad de videos producidos 87
3.5.2 Total de contenido producido 88
3.5.3 Duración Promedio de los Videos 88

CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES **90**

4.1 Conclusiones 90
4.2 Recomendaciones 91

REFERENCIAS BIBLIOGRÁFICAS **96**

ANEXOS **97**

ÍNDICE DE TABLAS

1 Características generales de las tecnologías Text-to-Speech (TTS) 11
2 Buenas prácticas en metodologías ágiles 14
3 Escala de Likert para evaluación de preguntas 19
4 Promedios de las dimensiones evaluadas en la encuesta de Likert 19
5 Comparativa de LLMs 36
6 Comparativa de motores TTS 39
7 Comparativa de enfoques de desarrollo 41
8 Comparativa de frameworks de desarrollo cross-platform 42
9 Comparativa de Metodologiás Ágiles 45
10 Requerimientos del Sistema 46
11 Planificación de Iteraciones 48
12 Parámetros para los métodos de generación de presentaciones 72
13 Parámetros para la generación de nuevas diapositivas con IA 75
14 Descripción de IPCs 78
15 Pruebas de Aceptación de Requerimientos - Iteración 1 80
16 Pruebas de Aceptación de Requerimientos - Iteración 2 82
17 Pruebas de Aceptación de Requerimientos - Iteración 3 83
18 Tiempos promedios por etapa 85

ÍNDICE DE FIGURAS

1 Resultados de la entrevista en la pregunta uno. A) Estandarización del contenido, B) Mayor flexibilidad creativa, C) Reducción de tiempos . . . 20
2 Resultados de la entrevista en la pregunta dos. A) Asistencia en creación de diapositivas, B) Asistencia en la grabación de voz, C) Asistencia en escritura de guiones, D) Búsqueda de recursos visuales 21
3 Resultados de la entrevista en la pregunta tres. A) Planificación, B) Creación de diapositivas, C) Redacción del guión, D) Grabación , E) Edición, F) Revisión de calidad 22
4 Resultados de la encuesta en la pregunta cuatro. A) Errores/repeticiones en la grabación, B) Tiempo de edición, C) Tiempo de creación de diapositivas, D) Redacción de guiones 23
5 Resultados de la entrevista en la pregunta cinco. A) Diapositivas no claras, B) Mala calidad de audio, C) Errores de edicion, D) Imágenes de baja calidad 24
6 Resultados de la entrevista en la pregunta seis. A) Gestión de proyectos, B) LLMs, C) Síntesis de voz, D) Ninguna 25
7 Resultados de la entrevista en la pregunta siete. A) Herramientas de edición, B) Software para creación de presentaciones, C) Sistemas de gestión de proyectos 26
8 Resultados de la entrevista en la pregunta ocho. A) LLMs, B) Plataformas de diseño gráfico, C) Edición de video 27
9 Resultados de la entrevista en la pregunta nueve. A) Organización eficiente, B) Asistencia en la creación de contenido, C) Control sobre el contenido 28
10 Resultados de la entrevista en la pregunta diez A) Interfaz intuitiva, B) Control sobre el contenido, C) Automatización 29
11 Análisis del proceso de producción de contenido audiovisual 30
12 Diagrama de flujo del proceso propuesto 34
13 Diagrama de flujo del proceso propuesto 51
14 Arquitectura del Sistema 52
15 Prototipo de interfaz para la gestión de proyectos 53
16 Prototipo del formulario para la creación de proyectos 54

17 Prototipo de interfaz para la gestión de videos del proyecto 54
18 Prototipo del formulario para la creación de videos sin IA 55
19 Prototipo del formulario para la creación de videos con IA 56
20 Prototipo de interfaz de edición de diapositivas 56
21 Prototipo del formulario para añadir nuevas diapositivas con IA 57
22 Prototipo de formulario para la creación de nuevas diapositivas con IA . 58
23 Prototipo del formulario para la reescritura de los guiones con IA 58
24 Prototipo de formulario para el reemplazo de palabras en el guion 59
25 Prototipo de interfaz para la visualización del video resultante 59
26 Prototipo de interfaz para la gestión de la base de reemplazos 60
27 Prototipo del formulario para la creación de reemplazos 61
28 Prototipo de interfaz del Administrador 62
29 Prototipo de interfaz del Panel de Configuración de APIs y Servicios . . . 63
30 Prototipo de interfaz del Login 63
31 Diagrama de Base de Datos 66

ÍNDICE DE ANEXOS

A Guía de entrevista a los instructores de la empresa Ingedemy 97
B Prompts para la creación de diapositivas mediante Idea, Párrafo o Nueva Diapositiva 98
C Códigos Relevantes del Proyecto 104
D Mediciones de tiempos de los instructores 115
E Calculo t-Student para la producción de videos 117
F Manual de usuario para los instructores 119

RESUMEN EJECUTIVO

La producción de videos educativos enfrenta importantes retos en términos de tiempo y calidad. Este proyecto desarrolló un sistema automatizado para optimizar la generación de contenido audiovisual, utilizando herramientas basadas en inteligencia artificial. Entre las tecnologías empleadas destacan Microsoft Azure Text-to-Speech para la síntesis de voz, GPT-4 para la redacción de guiones, FFMPEG para la edición programática de video y Electron JS como framework para el desarrollo de una aplicación de escritorio intuitiva y escalable.

El sistema automatiza procesos clave como la creación de diapositivas, la generación de narraciones y la redacción de guiones, logrando una reducción del tiempo de producción en un 67.43%. Su diseño responde a las necesidades de los instructores de Ingedemy, quienes experimentaban cuellos de botella en etapas como la edición y la grabación. Los resultados del análisis validaron mejoras significativas en la eficiencia y calidad del contenido generado, destacando la capacidad del sistema para personalizar y estandarizar los resultados.

Se concluye que la integración de estas herramientas tecnológicas no solo reduce la carga de trabajo manual, sino que también facilita la adopción de soluciones innovadoras para satisfacer la creciente demanda de contenido educativo de alta calidad.

Palabras clave: Automatización, inteligencia artificial, síntesis de voz, redacción automatizada, videos educativos, desarrollo de software.

ABSTRACT

The production of educational videos poses significant challenges in terms of time and quality. This project developed an automated system to optimize audiovisual content creation using artificial intelligence tools. Key technologies include Microsoft Azure Text-to-Speech for voice synthesis, GPT-4 for scriptwriting, FFMPEG for programmatic video editing, and Electron JS for developing a scalable desktop application with a user-friendly interface.

The system automates critical processes such as slide creation, narration generation, and scriptwriting, achieving a 67.43% reduction in production time. It addresses the needs of Ingedemy instructors, who faced bottlenecks in stages like editing and recording. Results validated significant improvements in efficiency and quality, emphasizing the system's ability to personalize and standardize content.

This project demonstrates that integrating these technological tools not only reduces manual workloads but also fosters the adoption of innovative solutions to meet the growing demand for high-quality educational content.

Keywords: Automation, artificial intelligence, text-to-speech, automated scriptwriting, educational videos, software development.

CAPÍTULO I. MARCO TEÓRICO

1.1 Tema de investigación

DESARROLLO DE UN SISTEMA PARA LA GENERACIÓN DE CONTENIDO AUDIOVISUAL UTILIZANDO RECURSOS BASADOS EN INTELIGENCIA ARTIFICIAL Y SU INCIDENCIA EN LOS TIEMPOS DE PRODUCCIÓN

1.1.1 Planteamiento del problema

La educación es un pilar fundamental para el desarrollo de las sociedades en todo el mundo, siendo un factor determinante en la igualdad de oportunidades [1]. En este sentido, la demanda de contenido educativo de calidad ha ido en constante aumento, impulsada por la necesidad de adquirir conocimientos relevantes en un mundo cada vez más competitivo y dinámico [2]. Sin embargo, la generación de material educativo, especialmente en formato de video, representa un desafío considerable en términos de tiempo, esfuerzo y recursos necesarios para su elaboración [3].

A nivel mundial la producción de contenido educativo audiovisual requiere un tiempo promedio de entre 1 y 4 horas por cada video generado, con una duración promedio de 2 minutos [4]. Esta cifra refleja la complejidad y el costo asociado a la creación de material educativo de alta calidad, lo cual limita significativamente la disponibilidad de recursos educativos. Asimismo, la escasez de recursos humanos capacitados y especializados en la producción de contenido educativo representa un obstáculo adicional para satisfacer la creciente demanda de este tipo de material [5].

En Latinoamérica, la producción de contenido educativo es aún más crítica debido a diversos factores socioeconómicos y tecnológicos. En la region más del 62% de las instituciones educativas en la región carecen de los recursos necesarios para implementar tecnologías de aprendizaje digital [6]. Esto se traduce en una menor capacidad para generar y distribuir contenido educativo, limitando el acceso a una educación de calidad para la población. Además, la infraestructura tecnológica insuficiente y la escasa formación de los profesionales en el área de producción educativa son barreras significativas que enfrentan los países latinoamericanos para

avanzar en la digitalización de la educación [7].

En Ecuador, la capacitación insuficiente en producción de video educativo limita la creatividad y el potencial innovador de los docentes en la generación de material de alta calidad. Estudios indican que hasta el 66% de los docentes no han recibido formación en el uso de herramientas digitales para crear contenido educativo [8]. Frente a esta situación, la automatización en la producción de videos educativos mediante técnicas de Inteligencia Artificial (IA) representa una oportunidad significativa para superar estos desafíos y democratizar el acceso a una educación de calidad a nivel nacional.

1.2 Antecedentes investigativos

Para el desarrollo de esta investigación, es fundamental revisar trabajos previos relacionados con la automatización en la creación de contenido educativo en formato de video. Esto implica identificar los estudios que abordan enfoques similares, así como las diferencias relevantes en comparación con el presente proyecto. A partir de esta revisión, se establecieron los fundamentos teóricos y metodológicos que sirven de base para el análisis de las soluciones de automatización aplicadas en este ámbito.

En [9] aborda la generación automática de videos educativos, utilizando IA para clonar las voces y rostros de instructores a partir de contenido preexistente. El sistema se basa en diapositivas y guiones ya elaborados, los cuales se integran automáticamente en videos donde se simula a un instructor presentando el material. La voz del instructor se sintetiza mediante Tacotron 2, con una eficacia demostrada del 98.5% en precisión de síntesis vocal, y los movimientos faciales se sincronizan con esta voz a través de Wav2Lip, logrando una sincronización labial de 96.5% en precisión. La implementación ha sido evaluada positivamente en una encuesta realizada a 48 participantes, entre instructores y estudiantes, destacando la eficiencia en la producción de contenido educativo interactivo y la reducción de la carga de trabajo del instructor. Sin embargo, los resultados también subrayan la necesidad de políticas claras para abordar los desafíos éticos y de seguridad, particularmente los riesgos de desinformación asociados con la clonación de identidades en contextos educativos.

En [10] se examina la calidad de los recursos educativos generados por estudiantes y por IA. Según la percepción de los estudiantes, la calidad de los recursos generados

por IA es equivalente a la de los recursos generados por sus compañeros. Esto sugiere que los recursos generados por IA pueden servir como material suplementario viable en ciertos contextos educativos. En el análisis comparativo, tanto los recursos generados por estudiantes como por IA obtuvieron una mediana (Mdn) de 4 en una escala de Likert en tres aspectos clave: correcciónes, explicaciones y utilidad.

En [11] ofrece una comparativa profunda entre distintos modelos de lenguaje grande (LLMs), tanto de código abierto como cerrado, a través de un innovador benchmark llamado FOFO. El estudio demuestra una disparidad considerable en la precisión del seguimiento de formato, donde modelos cerrados como GPT-3.5 y Gemini Pro alcanzan una precisión en la tarea FOFO del 80.66% y 80.25% respectivamente, superando notablemente a modelos de código abierto como Openchat V3.2-super y WizardLM 13B V1.2, que logran 31.22% y 63.54% en la misma evaluación. A su vez, en benchmarks de seguimiento de contenido como AlpacaEval y MT-Bench, la distancia entre estos modelos se reduce, indicando que la habilidad para seguir formatos es una competencia distinta, no directamente correlacionada con la generación de contenido. Por ejemplo, Mistral 7B Instruct V0.1 exhibe un 92.78% en AlpacaEval, cercano al 93.42% de GPT-3.5, pero muestra una brecha significativa en FOFO con un 46.91%. Estas diferencias señalan que el dominio de los formatos por parte de los LLMs puede requerir ajustes finos especializados más allá de las afinaciones instruccionales regulares, lo cual es un punto de interés crítico para el desarrollo y la aplicación de LLMs en tareas especializadas.

En [12] se resalta la importancia de la automatización en la creación de videos mediante (IA),(LLM) y síntesis de voz (TTS), es esencial considerar las tecnologías disponibles para mejorar la calidad y eficiencia de este proceso. Google Cloud Text-to-Speech destaca por ofrecer el servicio adicional WaveNet, que utiliza un modelo avanzado para generar voces más naturales y precisas, acercándose al habla humana al procesar hasta 24,000 muestras por segundo, lo que produce sonidos más suaves y realistas en la síntesis de voz para videos automatizados. En la tabla comparativa de motores de (TTS), se observa que Nuance lidera con 53 idiomas y más de 199 voces, seguido por Microsoft Azure con 50 idiomas y más de 200 voces. Amazon Poly ofrece 29 idiomas y 60 voces, IBM Watson Text-to-Speech soporta 13 idiomas y 38 voces, Google Cloud Text-to-Speech cuenta con 40 idiomas y 220 voces, Readspeaker proporciona 30 idiomas y más de 90 voces, y Linguatec Voice Reader incluye 45 idiomas y 67

voces. Estos datos resaltan la diversidad y personalización que ofrecen los motores de TTS, así como la importancia de utilizar tecnologías avanzadas como WaveNet para mejorar la calidad y la naturalidad de la síntesis de voz en los videos generados de forma automatizada. Además, la Figura 3 muestra una comparación entre WaveNet de Google Cloud, otras voces sintéticas y la voz humana, basada en el Mean Opinion Score (MOS). En la figura, se observa que WaveNet obtiene un MOS de 4.21, superando a las técnicas concatenativas (3.86) y paramétricas (3.67), aunque aún por debajo de la calidad percibida de la voz humana (4.55). Estos hallazgos subrayan la relevancia de explorar y aprovechar las capacidades de los motores de TTS en la automatización de la creación de videos con IA y LLM para mejorar la experiencia del usuario y la efectividad del contenido audiovisual generado.

El estudio [13] investigó el efecto de presentaciones de vídeo educativo de alta velocidad utilizando discurso sintético en la comprensión y evaluación subjetiva de estudiantes universitarios. La investigación se llevó a cabo con 40 estudiantes de una universidad privada en Tokio. Los resultados mostraron que ni el factor del habla ni el factor de velocidad tuvieron un impacto significativo en el efecto de aprendizaje, lo que respalda hallazgos anteriores. Sin embargo, las evaluaciones subjetivas sobre la utilidad del vídeo educativo y la carga de información variaron según los factores del habla y la velocidad. Se concluyó que los estudiantes pueden aprender tanto con vídeos que utilizan habla sintética como con habla normal, y que el uso de habla sintética a doble velocidad puede mejorar la aceptabilidad y la inteligibilidad. No obstante, se resaltó la necesidad de investigaciones adicionales sobre la naturalidad del habla sintética y el impacto de la familiaridad del usuario en su percepción.

En [14] se introdujo ”Data Player”, una herramienta innovadora diseñada para la generación automática de videos de datos. Este sistema facilita la integración de visualizaciones estáticas y texto descriptivo con narraciones audiovisuales sincronizadas, utilizando (LLMs) para establecer enlaces semánticos entre texto y elementos visuales, y programación por restricciones para diseñar secuencias de animación que complementen la narración auditiva. El desarrollo de ”Data Player” implicó el uso de tecnologías avanzadas, incluyendo Python para la programación, la plataforma GSAP para las animaciones, y la tecnología Text-to-Speech de Microsoft Azure para la generación de narraciones. En la evaluación del sistema, que incluyó una galería de ejemplos, un estudio con usuarios y entrevistas con expertos, se demostró

que Data Player puede generar videos de datos que son comparables en calidad a los producidos manualmente por especialistas. Los resultados de la evaluación indicaron que los videos generados automáticamente lograron un alto grado de engagement y comprensión de datos, mostrando una mejora significativa en la eficiencia de producción de videos con reducción de tiempo de hasta el 70% en comparación con métodos tradicionales.

El artículo [15] presenta el desarrollo y los resultados del sistema URL2Video. Durante el proceso implementado en C++, se creó una herramienta de análisis de páginas web interna y se construyó un sistema de renderizado de vídeo utilizando MediaPipe, OpenCV y FFMpeg, junto con bibliotecas de código abierto como HarfBuzz y Skia. Se generaron vídeos para 50 páginas web, extrayendo en promedio 4,92 escenas por vídeo de 10 segundos, con un total de 15,78 tomas por vídeo. Se extrajeron 823 imágenes, 7 vídeos y 4874 elementos de texto. El proceso de corrección de errores tomó entre 3 y 5 minutos por página, mientras que el tiempo promedio para la puntuación, planificación y renderizado de activos fue de 54 segundos por página. Además, se excluyeron 10 páginas debido a la falta de respuesta del motor de renderizado web. En la evaluación con 65 respuestas únicas, se observaron diferencias estadísticamente significativas en el ritmo entre los vídeos de la línea de base y URL2Video, siendo el ritmo de la Línea Base estadísticamente más rápido. Sin embargo, ambos se consideraron más rápidos de lo ideal.

En [16] se presenta VideoEdit, una herramienta de edición de video web integrada en la plataforma Ambient Learning Spaces (ALS), diseñada para optimizar la producción de contenido educativo multimedia en las aulas. Esta solución tecnológica permite a los usuarios gestionar y editar videos de manera intuitiva, sin requerimientos de instalación software adicional o hardware especializado. VideoEdit se distingue por su arquitectura basada en la nube, que centraliza el almacenamiento y procesamiento de medios, asegurando accesibilidad desde diversos dispositivos a través de tecnologías web como Lenguaje de etiquetas de hipertexto (HTML), Hojas de estilo en cascada (CSS), JavaScript y ASP.NET, y utilizando la biblioteca FFmpeg para la conversión de formatos de video. Una evaluación de la herramienta mostró que más de 300 videos fueron editados y renderizados eficazmente, destacando una reducción significativa en los tiempos de procesamiento a aproximadamente 23 segundos para videos de un minuto en configuraciones de alta definición. Este sistema facilita la integración de

recursos didácticos multimedia en el aprendizaje, soportando la transformación digital en instituciones educativas y demostrando mejoras en la accesibilidad y eficiencia en la creación de materiales educativos visuales.

En [17] se desarrolló una aplicación de escritorio destinada a automatizar la extracción y generación de hojas de cálculo Excel de los resultados de los estudiantes universitarios, a partir de datos disponibles en el sitio web de una universidad. Esta solución se implementó utilizando tecnologías web y módulos de automatización en Python, destacándose el uso de Electron JS para el desarrollo del entorno de aplicación de escritorio. El sistema back-end integró Node JS y Python, gestionando los datos mediante MongoDB, lo que permitió la ejecución de procesos en lotes para la manipulación paralela de los datos extraídos. El sistema redujo significativamente el tiempo requerido para procesar los resultados de horas a aproximadamente 5-10 minutos para un lote de 60 estudiantes, dependiendo de la velocidad de respuesta del servidor y las condiciones de la red. La investigación demostró la viabilidad de automatizar procesos administrativos tediosos y propuso futuras incorporaciones de visualización de datos y análisis predictivo utilizando aprendizaje automático. Este enfoque resalta la utilidad de Electron JS no solo para crear interfaces de usuario robustas y eficientes sino también para integrar funciones de automatización en aplicaciones de escritorio sin depender de servicios externos, justificando su aplicación en contextos donde la interacción usuario-sistema y la eficiencia de procesamiento son críticas.

En el estudio [18] los investigadores examinaron la viabilidad de utilizar ChatGPT para desarrollar un currículo completo y planes de lección para un curso de posgrado, con el objetivo de evaluar si los estudiantes podían discernir que el contenido fue generado por IA. Este análisis incluyó tanto la generación de temas apropiados para las lecciones como la evaluación de la efectividad de la IA para organizar el contenido educativo de manera coherente. Utilizando un enfoque de estudio de caso, el curso fue diseñado exclusivamente a través de indicaciones a ChatGPT, y se desarrollaron múltiples iteraciones de contenido para afinar la precisión y relevancia de los temas tratados. El curso trató diversos temas relacionados con la IA en la educación, desde chatbots y asistentes virtuales hasta ética en IA. El estudio reveló limitaciones significativas en la capacidad de ChatGPT para integrar aprendizajes previos en nuevas situaciones de aprendizaje, así como para preparar estratégicamente a los estudiantes para futuros

resultados de aprendizaje. A pesar de estas limitaciones, la investigación demostró que ChatGPT podría ser una herramienta útil para diseñar rápidamente contenidos educativos, permitiendo a los educadores concentrarse más en adaptar las lecciones a las necesidades individuales de los estudiantes.

En [19] se aborda la creación de diapositivas de presentación desde cuadernos computacionales utilizando Slide4N, una herramienta que permite la colaboración humano-IA. Este sistema utiliza técnicas avanzadas de procesamiento de lenguaje natural para seleccionar y organizar automáticamente la información clave de las celdas del cuaderno en diapositivas, ofreciendo a los usuarios la capacidad de refinar y personalizar las diapositivas generadas. Integrado en JupyterLab, Slide4N proporciona una experiencia de usuario integrada y eficiente. Un estudio de usuario dividido en dos partes reveló que Slide4N puede reducir el tiempo de creación de diapositivas en un 50% en comparación con métodos totalmente manuales. Además, los resultados del estudio indican que Slide4N mejora significativamente la eficiencia de comunicación de análisis de datos complejos, siendo bien recibido por el 95% de los participantes que lo encontraron efectivo para audiencias tanto técnicas como no técnicas. Este enfoque destaca la importancia de la colaboración humano-IA en la automatización de la generación de contenidos educativos, posicionando a Slide4N como una herramienta clave para minimizar la brecha entre el análisis de datos y su presentación en formatos accesibles y comprensibles.

En [20] proponen un sistema automatizado para la creación y entrega de presentaciones educativas personalizadas, utilizando tecnologías de Procesamiento de Lenguaje Natural (NLP) y Aprendizaje Automático (ML). Este sistema, desarrollado en respuesta a la creciente demanda de contenido virtual durante la pandemia de COVID-19, automatiza la generación de presentaciones a partir de documentos estructurados como artículos de investigación. Utilizando técnicas de resumen basadas en BERT, el documento es condensado en puntos clave que se organizan en diapositivas. Además, se implementa un modelo de clonación de voz basado en una arquitectura inspirada en Tacotron y redes GAN (Generative Adversarial Networks) para narrar el contenido de las diapositivas en la voz personalizada del autor o cualquier otra voz deseada. Los resultados del sistema demostraron una notable eficiencia en la reducción del tiempo de creación y entrega de presentaciones, permitiendo presentaciones más atractivas y personalizadas. La implementación del sistema se llevó a cabo utilizando Python, y

los modelos preentrenados y los contenidos generados se almacenaron y gestionaron a través de Google Drive, asegurando una integración eficiente y segura del flujo de trabajo educativo.

En [21] se realiza un análisis comparativo detallado entre dos modelos de IA conversacional, Gemini y ChatGPT. Para compararlos, se emplearon pruebas empíricas y evaluaciones en escenarios del mundo real, midiendo métricas clave como coherencia de respuesta, precisión, latencia y escalabilidad. Se utilizó un conjunto de datos estandarizado para ambas plataformas y se aplicaron las mismas solicitudes de información para asegurar una evaluación justa. Gemini, integrando Google Search, logró una precisión factual del 78%, superior al 65% de ChatGPT. Por otro lado, ChatGPT alcanzó una mayor coherencia conversacional, con un 80% frente al 72% de Gemini. Este estudio no solo compara las diferencias arquitectónicas y las metodologías de entrenamiento de ambos modelos, sino que también destaca la importancia de entender sus capacidades para su aplicación estratégica en diversos contextos, discutiendo igualmente las consideraciones éticas y las futuras implicaciones de estas tecnologías.

En [22] se introduce el PowerPoint Task Completion benchmark, el cual se centra en evaluar la capacidad de los (LLMs) para completar tareas en archivos de PowerPoint basadas en instrucciones de usuario. Además de resaltar el rendimiento de GPT-4, se llevaron a cabo evaluaciones de otros LLMs tanto de código cerrado (ChatGPT, Davince-003) como de código abierto (LLaMa-2). Aunque GPT-4 demostró ser el LLM más sólido con una precisión del 75.1% en pruebas de diálogo de un solo turno, se encontró que enfrentaba dificultades al completar sesiones completas, logrando solo un 6% de precisión en este aspecto. Estos desafíos también se observaron en otros LLMs evaluados, lo que subraya la importancia de abordar los factores que limitan el rendimiento de estos modelos en tareas de finalización de instrucciones en entornos complejos.

En [23] se observaron diferencias significativas en la calidad de voz generada por diferentes Interfaces de programación de aplicaciones (APIs) de TTS. Microsoft Azure TTS obtuvo la puntuación más alta en Intelligibility y Naturalness, con un 5.6 en Intelligibility y un 5.6 en Naturalness para voces masculinas, y un 5.4 en Intelligibility y un 5.3 en Naturalness para voces femeninas. IBM Watson TTS siguió de cerca con un 5.4 en Intelligibility y un 5.3 en Naturalness para voces masculinas, y un 5.3 en

Intelligibility y un 5.2 en Naturalness para voces femeninas. Por otro lado, Google TTS y Play.ht mostraron las puntuaciones más bajas, con Google TTS obteniendo un 5.3 en Intelligibility y un 5.1 en Naturalness para voces masculinas, y un 3.3 en Intelligibility y un 2.6 en Naturalness para voces femeninas, mientras que Play.ht obtuvo un 5.0 en Intelligibility y un 4.4 en Naturalness para voces masculinas, y un 2.9 en Intelligibility y un 2.3 en Naturalness para voces femeninas. Estos resultados resaltan la importancia de considerar la calidad de voz al seleccionar un API de TTS para garantizar una experiencia de usuario satisfactoria en aplicaciones de generación de contenido auditivo.

En [24] se analiza el desarrollo de aplicaciones móviles híbridas con el framework Ionic, destacando su versatilidad para crear apps multiplataforma. Se proporciona una guía detallada sobre la implementación de Ionic con React, incluyendo herramientas y estilización de componentes. Se describe un proyecto y taller realizados con Ionic para Knowit Solutions Oy, demostrando su eficacia en la construcción de aplicaciones para iOS y Android con un único código base. Se recomienda el uso de Electron para distribuir aplicaciones de escritorio, con un aumento del 20% en eficiencia de desarrollo y distribución en comparación con métodos convencionales.

En [25] se aborda el desarrollo de un sistema automatizado para la generación de cursos educativos que hace uso de tecnologías de IA para facilitar el aprendizaje autodirigido. Dentro de las herramientas destacadas se encuentra Create-My-Course, la cual ofrece funciones para segmentar materiales grabados, generar resúmenes automáticos en varios idiomas y crear diferentes tipos de preguntas. Además, se menciona la integración de la biblioteca MoviePy, la cual es una herramienta de procesamiento de video en Python utilizada para la generación automática de transcripciones y resúmenes de conferencias grabadas. Este enfoque innovador, que también incluye la creación de preguntas de opción múltiple, verdadero o falso y de respuesta corta, junto con un asistente de IA disponible las 24 horas para atender consultas de los estudiantes, busca mejorar la participación y la experiencia de aprendizaje en entornos asincrónicos. Los resultados obtenidos muestran una precisión promedio del 93% en la generación de contenido educativo, lo que demuestra la eficacia de esta solución para optimizar el proceso educativo.

1.3 Fundamentación teórica

1.3.1 Large Language Models (LLM)

Los LLM son sistemas avanzados de inteligencia artificial entrenados para procesar y generar texto de manera precisa. A través del uso de redes neuronales profundas, estos modelos aprenden a identificar patrones lingüísticos complejos a partir de vastas cantidades de datos textuales. La capacidad de los LLM para adaptarse a distintos formatos y dominios de conocimiento les permite ser aplicados en una amplia gama de tareas, como la traducción, la redacción automática y el análisis de datos. Esta adaptabilidad ha sido clave para su implementación en diversas áreas, incluyendo la educación, el marketing y la investigación científica. Además, los LLM son escalables, lo que les permite manejar grandes volúmenes de información y generar contenido multilingüe, facilitando la creación de textos en diferentes idiomas [11].

Los beneficios más destacados de los LLM incluyen:

- **Alta precisión en la generación de contenido:** Los LLM pueden producir textos altamente coherentes y personalizados, que se ajustan a los requisitos específicos del usuario.
- **Escalabilidad:** Estos modelos pueden manejar grandes volúmenes de datos y son aplicables a proyectos de diferentes magnitudes.
- **Personalización y flexibilidad:** La adaptabilidad de los LLM a diferentes dominios y lenguajes los convierte en herramientas clave en diversos sectores.
- **Soporte para múltiples idiomas:** Pueden procesar y generar contenido en varios idiomas, lo que facilita la creación de contenido multilingüe y la traducción automática.

1.3.2 Text To Speech (TTS)

Las tecnologías Text-to-Speech (TTS) permiten la conversión de texto en habla mediante algoritmos avanzados de síntesis de voz. Estas tecnologías son fundamentales en la creación de interfaces conversacionales, donde la capacidad de generar voz que

imita la expresión humana es crucial. Los sistemas TTS utilizan diversos enfoques para producir una voz sintética que suena natural, desde el uso de grabaciones humanas preprocesadas hasta modelos generados por computadora que replican los patrones del habla. Estas tecnologías han sido adoptadas en áreas como la accesibilidad, la educación, y la automatización de servicios, permitiendo una interacción más fluida entre humanos y máquinas. Además, los motores TTS modernos soportan múltiples idiomas y estilos de voz, lo que facilita su aplicación en contextos multilingües y mejora la personalización de la experiencia del usuario [12].

Tabla 1. Características generales de las tecnologías Text-to-Speech (TTS)

Característica	**Detalles**
Cantidad de idiomas soportados	Soportan decenas de idiomas, facilitando la interacción multilingüe.
Variedad de voces disponibles	Ofrecen una amplia gama de voces personalizables en tono y velocidad.
Personalización de la voz	Ajustan características como tono, velocidad y pausas para emular la voz humana.
Uso en aplicaciones comerciales	Empleadas en asistentes virtuales, atención al cliente, y dispositivos de accesibilidad.
Accesibilidad mejorada	Fundamentales para personas con discapacidades visuales o dificultades de lectura.

1.3.3 Editores de video

Los editores de video son sistemas de software diseñados para permitir la manipulación y procesamiento de contenido audiovisual. Estos programas ofrecen herramientas para realizar diversas operaciones como cargar, cortar, unir, y ajustar videos y pistas de audio. La capacidad de editar videos incluye también la adición de efectos, transiciones y textos superpuestos, lo que facilita la creación de contenido personalizado y atractivo. Los editores de video modernos son esenciales en áreas como la educación, el entretenimiento, y la creación de contenido multimedia, proporcionando un entorno accesible y adaptable para usuarios de todos los niveles.

Además, estos editores soportan una amplia variedad de formatos de archivo, lo que permite su uso en diferentes dispositivos, desde computadoras hasta dispositivos

móviles. La flexibilidad que ofrecen es crucial para asegurar que los videos puedan ser exportados con las características y la calidad adecuadas para el propósito específico del usuario. Al integrar funciones avanzadas como la codificación y la decodificación de archivos, los editores de video facilitan la creación de proyectos audiovisuales de alta calidad y su implementación en plataformas digitales [16].

1.3.4 Framework de Desarrollo de Escritorio

Los frameworks proporcionan una base estructural predefinida para el desarrollo de software, facilitando la creación de aplicaciones mediante componentes reutilizables. A lo largo del tiempo, han demostrado ser esenciales para reducir los costos de desarrollo y mejorar la eficiencia en el proceso, ya que permiten a los desarrolladores trabajar sobre una estructura modular y flexible. Un framework no es solo un conjunto de herramientas, sino un sistema que soporta el desarrollo de aplicaciones dentro de un dominio específico, ofreciendo patrones que ayudan a la interacción entre los componentes. De esta manera, los frameworks permiten la extensibilidad de funcionalidades, asegurando que el software pueda crecer y adaptarse a las necesidades cambiantes de los proyectos [26].

Por otro lado, los frameworks de desarrollo de escritorio contribuyen significativamente a la mejora de la calidad del software al incorporar mecanismos estandarizados de prueba, integración y despliegue. Estas herramientas simplifican la interacción entre los componentes y fomentan la creación de aplicaciones seguras y eficientes. Además, los frameworks ofrecen una flexibilidad que permite adaptar el desarrollo a las necesidades del usuario sin perder las ventajas de un sistema organizado y probado. La correcta selección de un framework es crucial para el éxito de un proyecto, ya que una elección inadecuada puede aumentar la complejidad del desarrollo, alejando el enfoque del dominio específico para el cual fue concebido. En resumen, los frameworks de desarrollo de escritorio son una pieza clave en la creación de aplicaciones eficientes, escalables y mantenibles [26].

1.3.5 Lenguajes de Programación

Los lenguajes de programación son sistemas formales que permiten a los desarrolladores instruir a las computadoras para realizar tareas complejas. Estos lenguajes han sido diseñados con el objetivo de transformar las ideas humanas en un formato comprensible para las máquinas, facilitando la creación de software en diversos ámbitos, desde la educación hasta la investigación y el desarrollo industrial. A través de los lenguajes de programación, los desarrolladores pueden estructurar sus algoritmos y soluciones de manera eficiente, siguiendo una serie de reglas sintácticas y semánticas que optimizan la interacción entre el humano y la tecnología. Su evolución a lo largo del tiempo ha permitido que sean accesibles para distintos niveles de especialización, lo que los convierte en una herramienta clave para la innovación tecnológica en la sociedad actual.

El aprendizaje y el uso de lenguajes de programación no solo permite a los usuarios desarrollar competencias técnicas, sino también habilidades cognitivas como la resolución de problemas y el pensamiento lógico. Estos lenguajes proporcionan una estructura clara y definida que ayuda a organizar las ideas de forma sistemática, promoviendo un enfoque metódico en la resolución de desafíos tecnológicos. Al facilitar la creación de software, los lenguajes de programación han transformado la manera en que las personas interactúan con la tecnología, generando un impacto significativo en áreas como la automatización, la gestión de información y la innovación. Así, el uso de lenguajes de programación se ha consolidado como un pilar fundamental en el desarrollo de soluciones digitales que responden a las necesidades de una sociedad cada vez más conectada [27].

1.3.6 Metodología Ágil

Las metodologías ágiles son enfoques iterativos y flexibles que permiten realizar entregas continuas en el desarrollo de software, integrando de manera constante la retroalimentación del cliente en cada fase del proyecto. Este enfoque se adapta rápidamente a los cambios en los requisitos y necesidades del cliente, optimizando la gestión de recursos y asegurando que los resultados cumplan con las expectativas

planteadas desde el inicio. Además, estas metodologías promueven una colaboración frecuente entre los diferentes roles del equipo y el cliente, lo que facilita la toma de decisiones ágiles y minimiza los riesgos asociados a la incertidumbre del proyecto. Gracias a su corta duración y capacidad de ajuste, las metodologías ágiles son ideales para entornos dinámicos y proyectos donde los requisitos evolucionan rápidamente.

Dentro de este enfoque, se destacan una serie de buenas prácticas que ayudan a asegurar el éxito de los proyectos. La Tabla 7 del documento describe algunas de las más relevantes, enfocadas en mejorar tanto la eficiencia como la colaboración. Entre ellas, la entrega temprana, incremental y continua de productos permite obtener retroalimentación rápidamente y hacer ajustes necesarios. Además, la experimentación y trabajo iterativo fomenta una mejora constante, mientras que la colaboración frecuente entre los equipos y el cliente asegura que el proyecto se mantenga alineado con las expectativas y necesidades del usuario final [28].

Tabla 2. Buenas prácticas en metodologías ágiles [28]

Buenas prácticas
Colaboración frecuente entre roles dentro del equipo del proyecto y con el cliente.
Reflexión y ajuste a intervalos regulares.
Entrega temprana, incremental y continua de un producto de trabajo.
Retroalimentación frecuente.
Experimentación, trabajo iterativo.

1.4 Objetivos

1.4.1 Objetivo general

Implementar un sistema para la generación de videos educativos en la empresa “Ingedemy” utilizando recursos basados en IA, con el fin de reducir el tiempo de producción del material audiovisual.

1.4.2 Objetivos específicos

- Analizar los procesos actuales de producción de videos educativos para identificar las principales áreas de ineficiencia y los factores que afectan la calidad del contenido.
- Investigar tecnologías de software que permitan reducir el tiempo requerido para la generación de videos educativos mediante la automatización de procesos.
- Desarrollar un sistema que integre las tecnologías investigadas para optimizar la producción y calidad de los videos educativos.

CAPÍTULO II. METODOLOGÍA

2.1 Materiales

Para la recolección de información en el presente proyecto, se llevó a cabo una entrevista dirigida a los instructores de la empresa Ingedemy, con el propósito de comprender en detalle sus necesidades y expectativas en relación con la producción automatizada de videos educativos mediante inteligencia artificial. Este método cualitativo permitió explorar de manera profunda los desafíos operativos y las posibles mejoras que pueden implementarse en el proceso actual de producción audiovisual. La guía utilizada para la entrevista se encuentra en el Anexo A.

2.2 Métodos

2.2.1 Modalidad de la investigación

El desarrollo de este proyecto se llevará a cabo a través de una metodología de investigación que proporciona el marco estructurado necesario para abordar sistemáticamente los objetivos establecidos. Para el presente proyecto, se ha optado exclusivamente por las siguiente modalidades de investigación:

a. Investigación cualitativa

El análisis se realizará mediante entrevistas técnicas, adoptando un enfoque cualitativo que permite una comprensión profunda y detallada de las percepciones, experiencias y opiniones de los instructores involucrados en la producción de contenido educativo en Ingedemy. Este enfoque facilita la exploración de los aspectos subjetivos y contextuales de los procesos actuales, contribuyendo significativamente a la identificación de necesidades y oportunidades de mejora. Las conclusiones derivadas de este análisis cualitativo serán fundamentales para desarrollar soluciones personalizadas que optimicen los tiempos de producción y mejoren la calidad del contenido educativo mediante la automatización eficaz de los procesos.

b. Investigación cuantitativa

Se llevará a cabo una comparación de los tiempos de producción de videos educativos antes y después de implementar el software de automatización desarrollado en este proyecto. Se recogerán datos sobre la duración de las diferentes fases del proceso de producción utilizando los métodos tradicionales y, posteriormente, con el nuevo sistema automatizado. Esta evaluación se realizará midiendo cada etapa, desde la creación del guion hasta la generación del video final. Los resultados de esta comparativa permitirán cuantificar el impacto del software en la eficiencia de producción, proporcionando evidencia concreta sobre la posible reducción en los tiempos de entrega y optimización de los recursos.

c. Investigación de campo

La investigación de campo se llevará a cabo mediante la observación directa y la interacción con los instructores durante su proceso de trabajo. Esto permitirá recolectar datos en el entorno natural donde se desarrolla la producción de contenido educativo, proporcionando información contextual valiosa que no se puede obtener mediante otros métodos. La investigación de campo ayudará a identificar de manera precisa los desafíos operativos y las necesidades tecnológicas específicas que enfrenta el personal de Ingedemy.

d. Investigación bibliográfica

La investigación bibliográfica implicará la revisión exhaustiva de literatura existente, incluyendo artículos académicos, libros, estudios previos y documentación técnica relevante. Esta modalidad de investigación proporcionará un marco teórico sólido y ayudará a contextualizar los hallazgos empíricos del proyecto dentro del conocimiento existente en el campo de la automatización de la producción de videos educativos y el uso de inteligencia artificial en la educación.

2.2.2 Población y muestra

Población para la obtención de requerimientos

Para el desarrollo de este proyecto de titulación centrado en la automatización de la generación de videos educativos con IA, se identifica a la población de instructores de la empresa Ingedemy que participan en la creación de contenido audiovisual. Con el propósito de obtener los requerimientos específicos para la automatización eficaz, se ha decidido involucrar a todos los 20 instructores. Cada uno de ellos será consultado mediante entrevistas técnicas y cuestionarios detallados para capturar sus necesidades y expectativas precisas respecto a las herramientas y procesos de producción de videos educativos. Esta estrategia garantiza que el desarrollo del proyecto se alinee con las realidades prácticas y las exigencias específicas de los usuarios finales.

2.2.3 Recolección de información

La información se recolectó mediante la realización de entrevistas a los instructores de la empresa Ingedemy, quienes participan directamente en la producción de videos educativos. Estas entrevistas tuvieron como objetivo entender en profundidad los procesos actuales de producción, las dificultades enfrentadas, y las oportunidades de mejora que pueden implementarse con la automatización mediante inteligencia artificial. En el Anexo A, se presenta una tabla con las preguntas formuladas durante las entrevistas.

a. Validación del instrumento

Para la validación del instrumento, se aplicará una encuesta a los instructores utilizando una escala de Likert basada en las preguntas de la entrevista técnica previamente realizada. Esta encuesta tiene como objetivo evaluar tres criterios clave: claridad, coherencia y relevancia de las preguntas formuladas durante la entrevista, en relación con el proceso de producción de contenido audiovisual educativo. Al recopilar las percepciones de los instructores sobre dichas preguntas, se obtendrán datos cuantitativos que permitirán refinar y mejorar la calidad de las preguntas utilizadas en futuras evaluaciones.

Tabla 3. Escala de Likert para evaluación de preguntas

Dimensión	Muy baja	Baja	Mediana	Alta	Muy alta
Claridad	1	2	3	4	5
Coherencia	1	2	3	4	5
Relevancia	1	2	3	4	5

Posteriormente se llevó a cabo la entrevista técnica con los instructores, obteniendo información cualitativa sobre sus experiencias y necesidades en el proceso de producción de contenido audiovisual educativo. Luego, los resultados fueron evaluados mediante una encuesta en escala de Likert, en la cual se analizaron tres dimensiones clave: claridad, coherencia y relevancia de cada pregunta realizada durante la entrevista. A partir de las respuestas, se calcularon los promedios correspondientes para cada dimensión, los cuales se presentan en la siguiente tabla:

Tabla 4. Promedios de las dimensiones evaluadas en la encuesta de Likert

Pregunta	Claridad	Coherencia	Relevancia
P1	5.00	4.33	5.00
P2	4.00	4.33	5.00
P3	5.00	4.67	4.00
P4	4.33	4.33	4.33
P5	4.67	4.33	4.00
P6	4.33	4.67	5.00
P7	4.67	4.67	4.67
P8	4.00	4.33	4.67
P9	5.00	4.00	4.67
P10	4.67	5.00	4.67

En [29] se establece que los promedios obtenidos en escalas de Likert que oscilan entre 3.51 y 4.5 corresponden a una valoración alta, mientras que los promedios que superan el 4.51 se consideran muy altos. Este rango de calificaciones es utilizado para interpretar la claridad, coherencia y relevancia de las preguntas en la validación de instrumentos de investigación cualitativa, asegurando que las valoraciones reflejen un nivel satisfactorio y robusto de aceptación por parte de los entrevistados. En este sentido, los promedios obtenidos en la evaluación de las dimensiones de claridad, coherencia y relevancia en la Tabla 4, que en su mayoría se sitúan en rangos altos o muy altos, validan la eficacia del instrumento utilizado en la recolección de información. Dichos resultados confirman que las preguntas formuladas en la entrevista técnica no solo fueron percibidas como claras y coherentes, sino también como altamente relevantes para los objetivos del estudio, garantizando así una recolección de datos

adecuada y alineada con las mejores prácticas en metodologías cualitativas.

b. Resultados de las entrevistas aplicadas a los intructores de la empresa Ingedemy.

Para analizar los resultados de las entrevistas, se empleó la metodología de Análisis Cualitativo Multinivel (ACM), que permite una evaluación detallada de la información cualitativa recopilada. El ACM se desarrolla en tres niveles: primero, se realiza una categorización textual de las respuestas obtenidas; luego, se procede con un análisis estructural-relacional que mide las relaciones entre las categorías identificadas; finalmente, en el nivel social-hermenéutico, se interpreta y contextualiza el discurso de los instructores dentro de su entorno de producción audiovisual educativo. Este enfoque permite no solo identificar patrones en las respuestas, sino también profundizar en la comprensión de las percepciones y experiencias de los instructores sobre el proceso de creación de contenido educativo, asegurando una interpretación rica y contextualizada de los datos obtenidos [30].

Pregunta 1: ¿Cómo cree que la integración de nuevas tecnologías podría mejorar la calidad del contenido educativo?

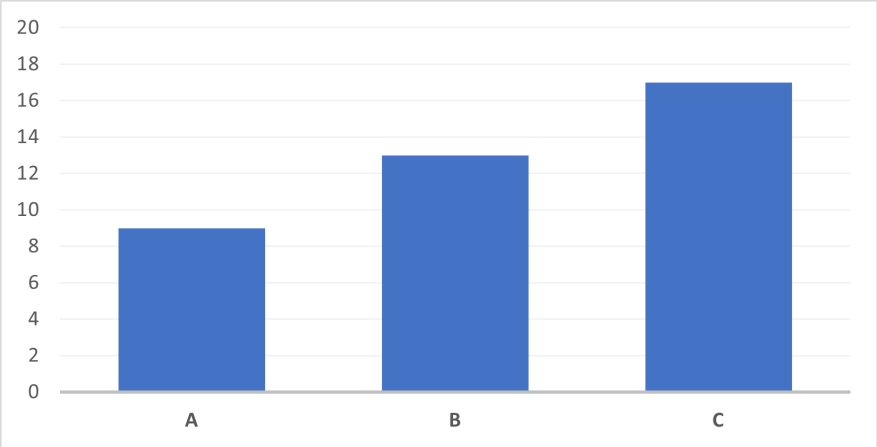


Figura 1. Resultados de la entrevista en la pregunta uno. A) Estandarización del contenido, B) Mayor flexibilidad creativa, C) Reducción de tiempos

Análisis e interpretación:

Los resultados muestran que los instructores consideran que integrar nuevas tecnologías mejoraría la calidad del contenido educativo principalmente por la reducción de tiempos, mencionada por 17 instructores. **Esto puede deberse a que los procesos actuales son manuales y repetitivos, y agilizar estas tareas permitiría enfocarse en otras actividades importantes**.

La flexibilidad creativa, destacada por 13 instructores, **podría explicarse porque facilitaría explorar nuevos formatos y estilos sin tantas limitaciones técnicas o de tiempo**. Por otro lado, la estandarización mencionada por 9 instructores **responde a la necesidad de mantener una calidad constante en los materiales, algo difícil de lograr cuando varias personas trabajan en un mismo proyecto**.

En general, los instructores valoran el potencial de estas herramientas para hacer su trabajo más eficiente y permitirles crear contenido más atractivo y uniforme.

Pregunta 2: ¿Qué funcionalidades de automatización cree que serían más beneficiosas para incorporar en su proceso de trabajo?

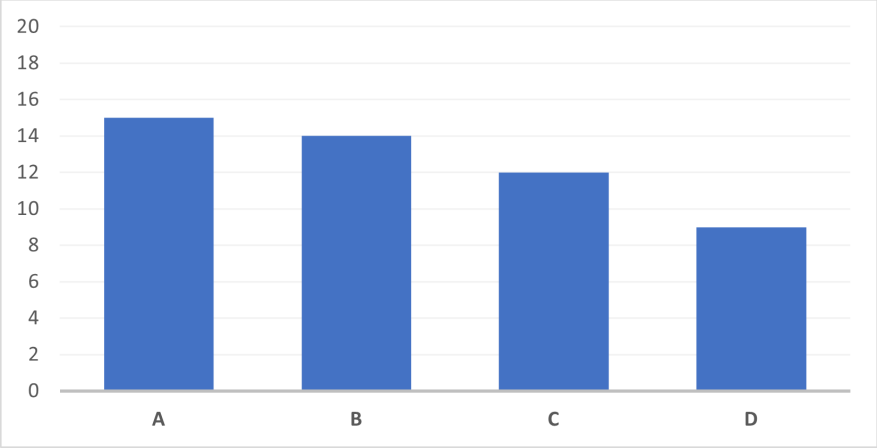


Figura 2. Resultados de la entrevista en la pregunta dos. A) Asistencia en creación de diapositivas, B) Asistencia en la grabación de voz, C) Asistencia en escritura de guiones, D) Búsqueda de recursos visuales

Análisis e interpretación:

Los resultados indican que la funcionalidad de automatización más valorada es la creación de diapositivas, mencionada por 15 instructores. **Esto puede deberse a la dificultad de estructurar ideas y organizar el contenido de forma pedagógica, asegurando que el material sea claro, lógico y efectivo para el aprendizaje**.

La grabación de voz, con 14 menciones, **podría ser prioritaria porque los instructores buscan simplificar una etapa que requiere precisión y repetir procesos para alcanzar una buena calidad**. Por otro lado, la escritura de guiones, mencionada por 12 instructores, **refleja la necesidad de apoyo para estructurar ideas de forma clara y organizada, algo que puede ser un desafío al desarrollar contenido educativo**.

Finalmente, la búsqueda automatizada de recursos visuales, con 9 menciones, **sugiere que, aunque es útil, no se percibe como tan esencial, posiblemente porque esta tarea puede realizarse con herramientas ya disponibles o porque tiene menos impacto directo en el flujo de trabajo**.

En general, las respuestas reflejan que los instructores priorizan funcionalidades que simplifiquen las etapas más complejas o repetitivas, permitiéndoles ahorrar tiempo y centrarse en el contenido educativo.

Pregunta 3: ¿Cuáles son los pasos clave en el proceso actual de producción de videos educativos en Ingedemy?

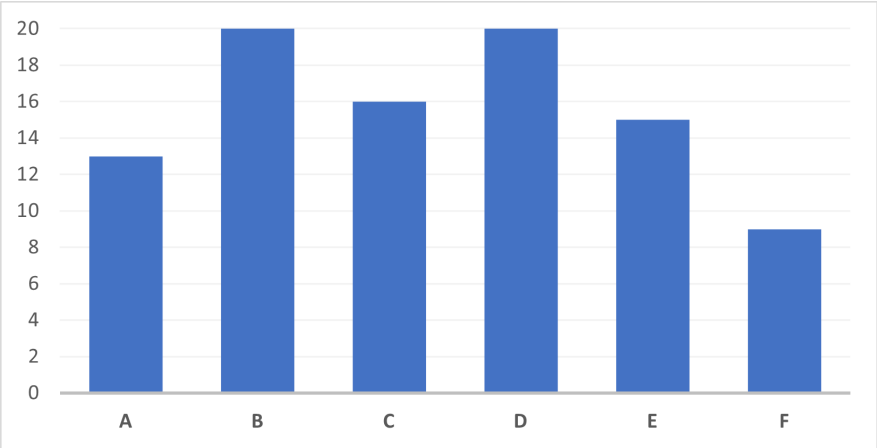


Figura 3. Resultados de la entrevista en la pregunta tres. A) Planificación, B) Creación de diapositivas, C) Redacción del guión, D) Grabación , E) Edición, F) Revisión de calidad

Análisis e interpretación:

El proceso de creación de videos incluye seis etapas: Planificación, Creación de Diapositivas, Redacción del Guion, Grabación, Edición y Revisión de Calidad. La Creación de Diapositivas y la Grabación, con 20 menciones cada una, **son prioritarias porque traducen conceptos pedagógicos en recursos claros y ejecutan el contenido de forma efectiva**. La Redacción del Guion, con 16 menciones, también es relevante, ya que estructura el contenido y guía la grabación.

La Edición, con 15 menciones, **es vista como importante, aunque menos crítica, posiblemente por un menor nivel de personalización requerido**. Finalmente, la Planificación (13 menciones) y la Revisión de Calidad (9 menciones) reciben menos atención, lo que sugiere que están más sistematizadas o son consideradas menos demandantes.

Pregunta 4: ¿Puede describir algún cuello de botella o desafío recurrente que enfrentan durante la producción de estos videos?

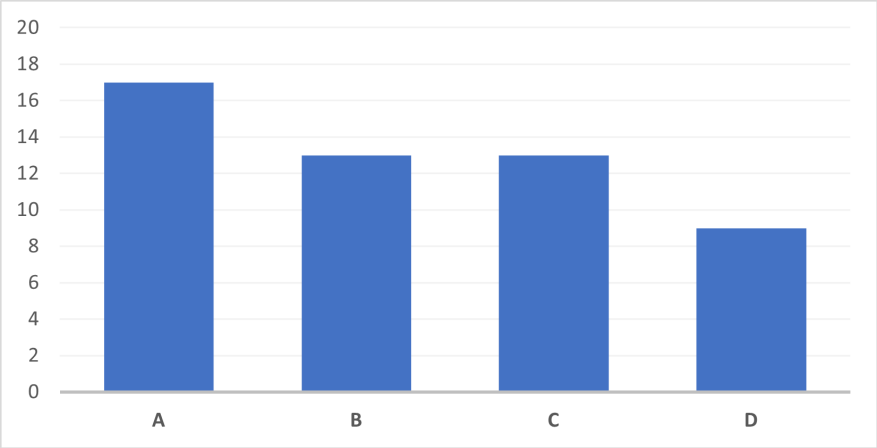


Figura 4. Resultados de la encuesta en la pregunta cuatro. A) Errores/repeticiones en la grabación, B) Tiempo de edición, C) Tiempo de creación de diapositivas, D) Redacción de guiones

Análisis e interpretación:

El principal cuello de botella identificado es la etapa de grabación, mencionada por 17 instructores. **Esto puede deberse a que es una fase susceptible a errores humanos, fallas técnicas o interrupciones externas, lo que obliga a repetir tomas y genera pérdidas de tiempo significativas**.

La edición y la creación de diapositivas, con 13 menciones cada una, también son desafiantes. **En la edición, el trabajo minucioso para corregir errores y mejorar**

la calidad puede ser percibido como tedioso, mientras que en la creación de diapositivas, la preparación de contenidos visuales efectivos requiere esfuerzo y tiempo.

Por último, la redacción de guiones, mencionada por 9 instructores, **refleja la complejidad de estructurar ideas claras y coherentes, lo que exige revisión y ajuste antes de grabar**. En general, los desafíos se concentran en las etapas que requieren mayor precisión y atención a los detalles.

Pregunta 5: ¿Hay factores específicos que, en su experiencia, afecten negativamente la calidad del contenido audiovisual educativo?

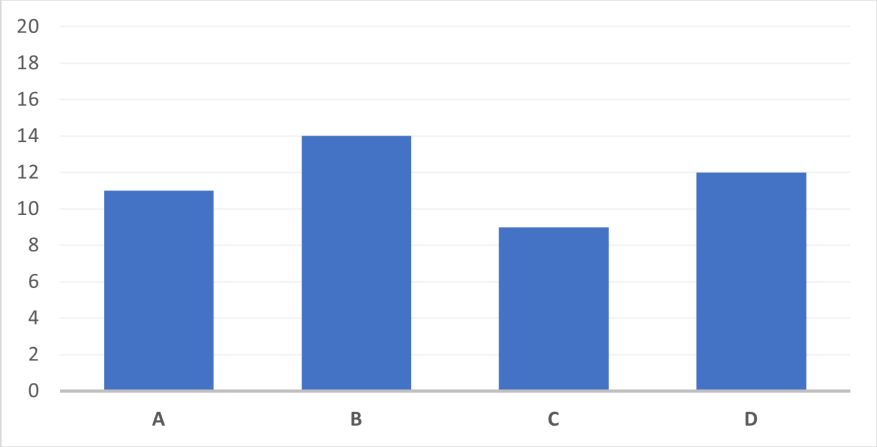


Figura 5. Resultados de la entrevista en la pregunta cinco. A) Diapositivas no claras, B) Mala calidad de audio, C) Errores de edicion, D) Imágenes de baja calidad

Análisis e interpretación:

Los factores que afectan negativamente la calidad del contenido audiovisual incluyen, en primer lugar, la mala calidad de audio, mencionada por 14 instructores. **Esto puede deberse a que el sonido deficiente dificulta la comprensión y genera una experiencia poco profesional para el espectador, lo que es crucial en materiales educativos**.

Las imágenes de baja calidad, señaladas por 12 instructores, **impactan la experiencia visual y reflejan una falta de claridad o profesionalismo en los recursos gráficos utilizados**. Asimismo, las diapositivas no claras, con 11 menciones, **indican que los instructores perciben como esencial que estas estén bien diseñadas y estructuradas, ya que complementan el contenido verbal y guían al espectador**.

Finalmente, los errores de edición, mencionados por 9 instructores, **sugieren que problemas como cortes abruptos o desincronización entre audio y video afectan la fluidez del contenido, reduciendo su impacto pedagógico**. En general, estos factores reflejan que tanto los aspectos técnicos como el diseño visual son claves para asegurar la calidad del material educativo.

Pregunta 6: ¿Está familiarizado con alguna tecnología de automatización actualmente utilizada en la producción de contenido audiovisual?

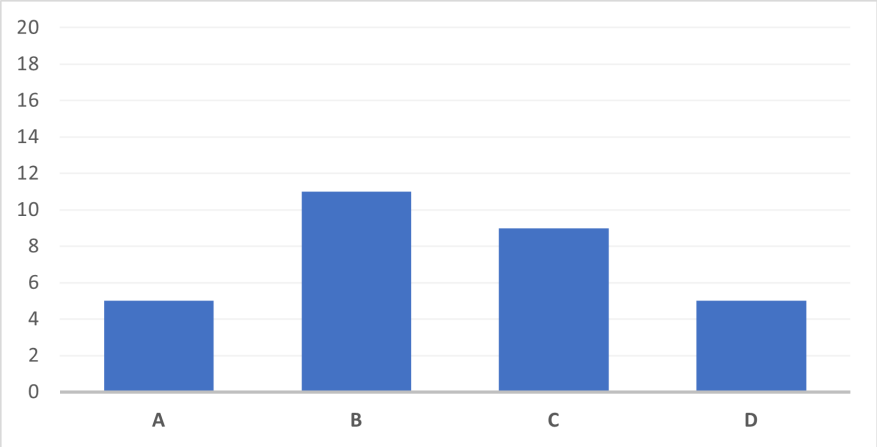


Figura 6. Resultados de la entrevista en la pregunta seis. A) Gestión de proyectos, B) LLMs, C) Síntesis de voz, D) Ninguna

Análisis e interpretación:

Los resultados muestran que varios instructores están familiarizados con tecnologías de automatización en la producción audiovisual. **La generación de guiones con herramientas como GPT o Gemini, mencionada por 11 instructores, destaca por su capacidad para facilitar el proceso creativo, eliminando la necesidad de empezar desde cero y permitiendo generar contenido de manera más rápida y estructurada**.

La síntesis de voz, señalada por 9 instructores, con ejemplos como Azure Speech y Google Cloud Text-to-Speech, **es valorada por generar narraciones automáticas de alta calidad, lo que reduce la dependencia de locutores humanos y optimiza tiempos en la producción**. Asimismo, 5 instructores mencionaron la importancia de herramientas de gestión de proyectos, que ayudan a coordinar y estructurar eficientemente cada etapa del proceso de creación audiovisual.

Sin embargo, 5 instructores indicaron no estar familiarizados con estas tecnologías, **lo que podría deberse a una falta de acceso, conocimiento o capacitación en estas**

herramientas, reflejando una brecha tecnológica que podría ser abordada con iniciativas formativas. En general, las respuestas evidencian una creciente adopción y valoración de tecnologías que simplifican y agilizan tareas clave en la producción de contenido educativo.

Pregunta 7: ¿Qué herramientas de software considera que son efectivas para la automatización de la producción de videos educativos?

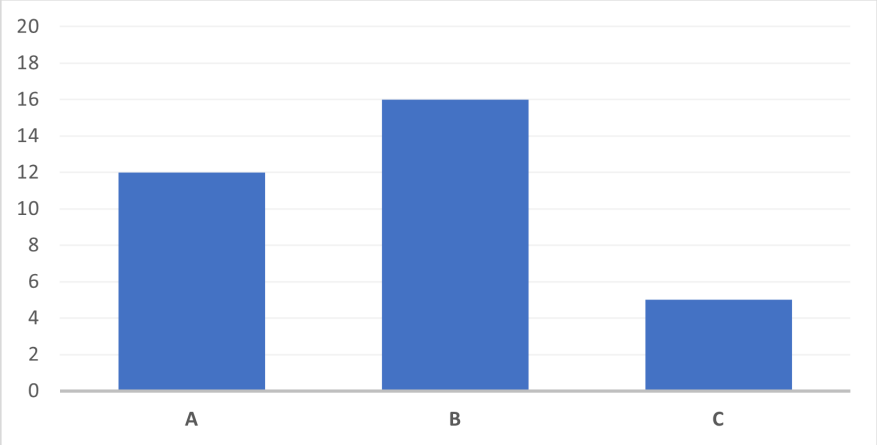


Figura 7. Resultados de la entrevista en la pregunta siete. A) Herramientas de edición, B) Software para creación de presentaciones, C) Sistemas de gestión de proyectos

Análisis e interpretación:

Los resultados muestran que el software para la creación de presentaciones, mencionado por 16 instructores, es considerado el más efectivo para automatizar la producción de videos educativos. **Esto puede deberse a que facilita la creación de recursos visuales de forma rápida y eficiente, lo que es esencial para complementar el contenido educativo**.

Las herramientas de edición, destacadas por 12 instructores, **son valoradas por agilizar la postproducción, permitiendo ajustes rápidos y mejorando la calidad final del material sin requerir habilidades avanzadas**. Por último, los sistemas de gestión de proyectos, mencionados por 5 instructores, **reflejan la importancia de mantener organizadas las distintas etapas del proceso, aunque su impacto directo en el producto final podría percibirse como menor**.

En general, las respuestas indican que los instructores priorizan herramientas que optimicen etapas clave y reduzcan el esfuerzo manual en la producción.

Pregunta 8: ¿Conoce algún ejemplo de software que haya ayudado a reducir los tiempos de producción en otros contextos similares?

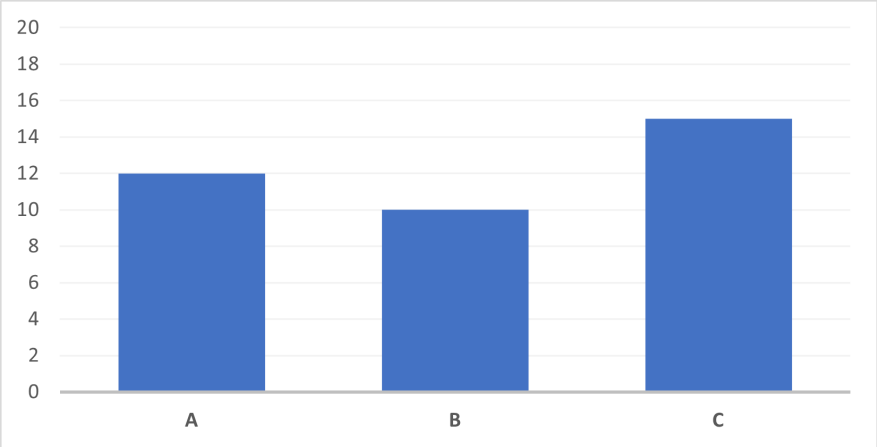


Figura 8. Resultados de la entrevista en la pregunta ocho. A) LLMs, B) Plataformas de diseño gráfico, C) Edición de video

Análisis e interpretación:

El software de edición de video, mencionado por 15 instructores, es considerado el más eficaz para reducir los tiempos de producción en contextos similares. **Esto se debe a su capacidad para agilizar la postproducción, permitiendo realizar ajustes y mejoras rápidamente, lo que reduce significativamente el esfuerzo manual en esta etapa**.

Los LLMs, mencionados por 12 instructores, **son valorados por su eficiencia en la generación de guiones e ideas, optimizando la fase inicial de creación y eliminando barreras relacionadas con la escritura desde cero**. Por último, las plataformas de diseño gráfico, con 10 menciones, **facilitan la creación rápida de recursos visuales, aunque su impacto se percibe como menos central en comparación con las herramientas de edición de video**.

En general, las respuestas reflejan que los instructores reconocen el valor de herramientas que optimizan tareas específicas y reducen tiempos en distintas etapas del proceso.

Pregunta 9: ¿Qué características considera esenciales en un sistema destinado a optimizar la producción de videos educativos?

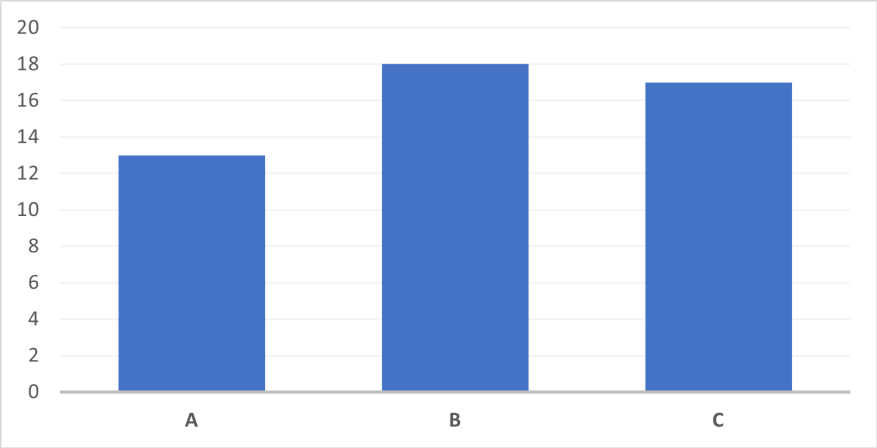


Figura 9. Resultados de la entrevista en la pregunta nueve. A) Organización eficiente, B) Asistencia en la creación de contenido, C) Control sobre el contenido

Análisis e interpretación:

Los resultados muestran que la característica más valorada por los instructores es la asistencia en la creación de contenido, mencionada por 18 instructores. **Esto refleja la necesidad de herramientas que simplifiquen el desarrollo de materiales como guiones, presentaciones y gráficos, permitiendo a los usuarios concentrarse en el contenido pedagógico sin partir desde cero**.

El control sobre el contenido, destacado por 17 instructores, **indica que los instructores priorizan la capacidad de personalizar y ajustar cada detalle del material, asegurando coherencia y calidad en los videos producidos**.

Finalmente, 13 instructores señalaron la importancia de una organización eficiente. **Esto sugiere que una gestión clara y estructurada de las fases de producción es esencial para optimizar tiempos y reducir la carga de trabajo, especialmente en proyectos complejos**. En conjunto, las respuestas destacan que los instructores buscan sistemas que combinen automatización, flexibilidad y organización para maximizar su productividad.

Pregunta 10: ¿Qué expectativas tiene sobre un sistema automatizado en términos de funcionalidad y facilidad de uso?

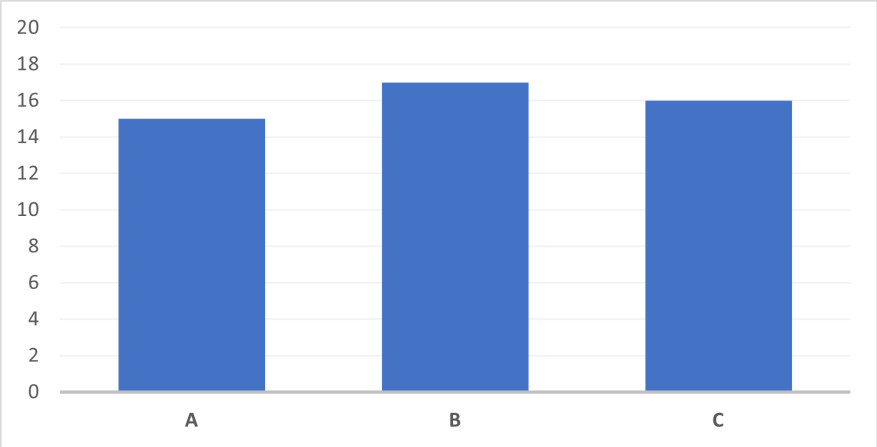


Figura 10. Resultados de la entrevista en la pregunta diez A) Interfaz intuitiva, B) Control sobre el contenido, C) Automatización

Análisis e interpretación:

Los resultados muestran que los instructores tienen expectativas claras sobre un sistema automatizado. La interfaz intuitiva, mencionada por 15 instructores, **refleja la necesidad de herramientas fáciles de usar, que no requieran una curva de aprendizaje elevada y permitan a los usuarios centrarse en la producción sin complicaciones técnicas**.

El control sobre el contenido, destacado por 17 instructores, **indica que, aunque se valora la automatización, los usuarios quieren supervisar y personalizar los resultados para garantizar que el material cumpla con sus estándares**.

Por último, la automatización, mencionada por 16 instructores, **es esencial para agilizar tareas repetitivas, como la edición o la creación de guiones, reduciendo tiempos y optimizando el flujo de trabajo**. En general, las respuestas reflejan que los instructores esperan un equilibrio entre automatización, facilidad de uso y control para maximizar la eficiencia y la calidad del sistema.

2.2.4 Procesamiento y análisis de datos

A través de las entrevistas realizadas a los instructores, se logró identificar varios puntos críticos en el proceso de producción de contenido audiovisual:

- El análisis del Grafico 11 permite identificar que las fases de creación de diapositivas, redacción de guion, grabación y edición presentan los mayores cuellos de botella y tienen un impacto directo en la calidad del contenido. Además, estas etapas ofrecen importantes oportunidades de automatización, donde el uso de inteligencia artificial puede optimizar los tiempos de producción y mejorar la precisión y coherencia del material educativo.

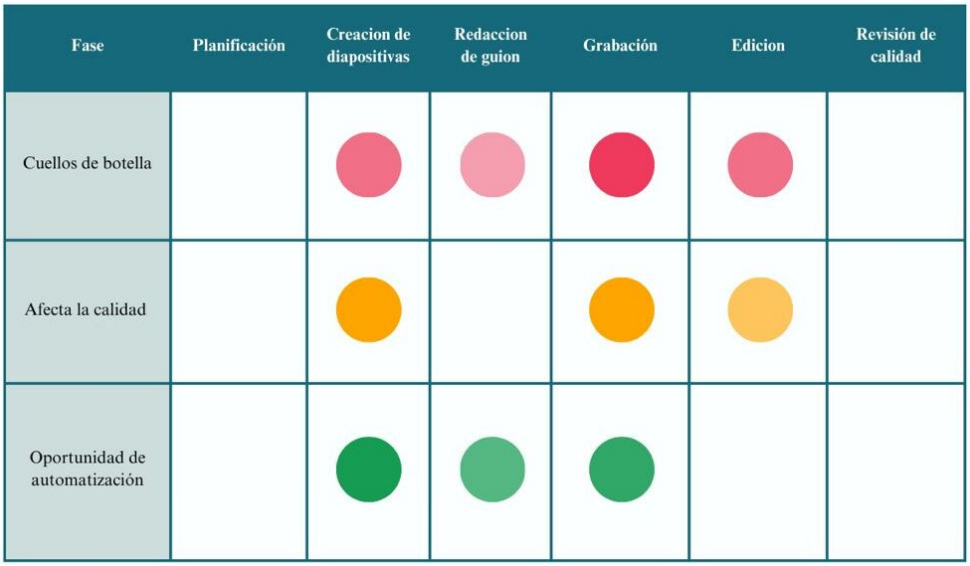


Figura 11. Análisis del proceso de producción de contenido audiovisual

- La mayoría de los instructores reportaron que enfrentan interrupciones frecuentes durante la grabación del contenido audiovisual, lo que genera la necesidad de múltiples tomas para obtener resultados satisfactorios. Este problema no solo afecta la eficiencia, sino que también incrementa la carga de trabajo en la etapa de edición.

- En cuanto a la edición de videos, se observó que el tiempo requerido para completar este proceso es considerablemente alto, lo que retrasa la entrega final del contenido. La falta de herramientas automatizadas y un sistema centralizado contribuye a esta ineficiencia.

- La creación de diapositivas para las presentaciones se realiza de manera manual, y los instructores manifestaron que este proceso es tedioso y consume mucho tiempo, lo que podría mejorarse con una herramienta que facilite la generación automática de diapositivas.

- En relación con la escritura de guiones, los instructores expresaron que a menudo les cuesta organizar sus ideas y estructurar el contenido de manera coherente. Esto ha llevado a una búsqueda de metodologías que simplifiquen esta tarea y les permitan enfocarse en la calidad del contenido.

- La encuesta realizada a los instructores indicó que, aunque están satisfechos con algunos aspectos de la producción audiovisual, existe una necesidad urgente de implementar un sistema que integre todas las etapas del proceso, desde la creación de las diapositivas hasta la generación del video final. Además, muchos instructores señalaron que la implementación de nuevas tecnologías como la IA permitiran reducir los tiempos de producción, tener mayor flexibilidad creativa y estandarizar el contenido.

CAPÍTULO III. RESULTADOS Y DISCUSIÓN

3.1 Análisis del proceso actual

Actualmente, la creación de contenido educativo en formato de video sigue un proceso mayoritariamente manual y segmentado, utilizando diferentes herramientas y software en cada etapa. Este proceso involucra varias etapas, que se describen a continuación:

1. **Planificación del contenido**:
   (a) El instructor define los temas, objetivos y estructura del curso.
   (b) Se elabora un esquema preliminar que sirve de base para el desarrollo del contenido.
2. **Creación de diapositivas**:
   (a) Se utiliza software de presentaciones (como PowerPoint, Google Slides o similares) para diseñar las diapositivas.
   (b) En esta etapa, el instructor agrega texto, imágenes, gráficos y animaciones manualmente, ajustando el diseño según el contenido.
3. **Escritura de guiones**:
   (a) Se redacta un guion para cada diapositiva o sección del curso.
   (b) Este proceso se lleva a cabo en un procesador de texto independiente (como Microsoft Word o Google Docs).
   (c) En caso de modificaciones, los cambios deben sincronizarse manualmente con las diapositivas.
4. **Grabación de audio y video**:
   (a) El instructor graba la narración utilizando un micrófono y software de grabación (como Audacity, Adobe Audition o similar).
   (b) Si se incluye grabación de video, se utiliza software de captura de pantalla o cámaras externas.
   (c) Este paso puede requerir múltiples intentos para obtener una grabación de calidad adecuada.

5. **Edición de audio y video**:

   (a) Los archivos de audio y video se editan en software especializado (como Adobe Premiere Pro, Final Cut Pro o DaVinci Resolve).

   (b) Se sincronizan manualmente las grabaciones con las diapositivas.

   (c) Se agregan elementos gráficos, transiciones y efectos adicionales para mejorar la presentación.

6. **Generación del video final**:

   (a) El video editado se exporta en formato MP4 o similar.

   (b) El archivo final se revisa para garantizar calidad y coherencia.

Limitaciones del proceso actual:

- **Alta dependencia de herramientas independientes**: Cada etapa requiere el uso de software específico, lo que genera una desconexión entre las fases del proceso.
- **Tiempo de producción elevado**: La falta de automatización y la necesidad de sincronización manual aumentan el tiempo necesario para completar un proyecto.
- **Mayor probabilidad de errores**: Los cambios realizados en una etapa no siempre se reflejan automáticamente en las demás, lo que puede causar inconsistencias.
- **Barreras técnicas**: El manejo de múltiples herramientas requiere habilidades específicas, lo que puede ser un desafío para usuarios menos experimentados.

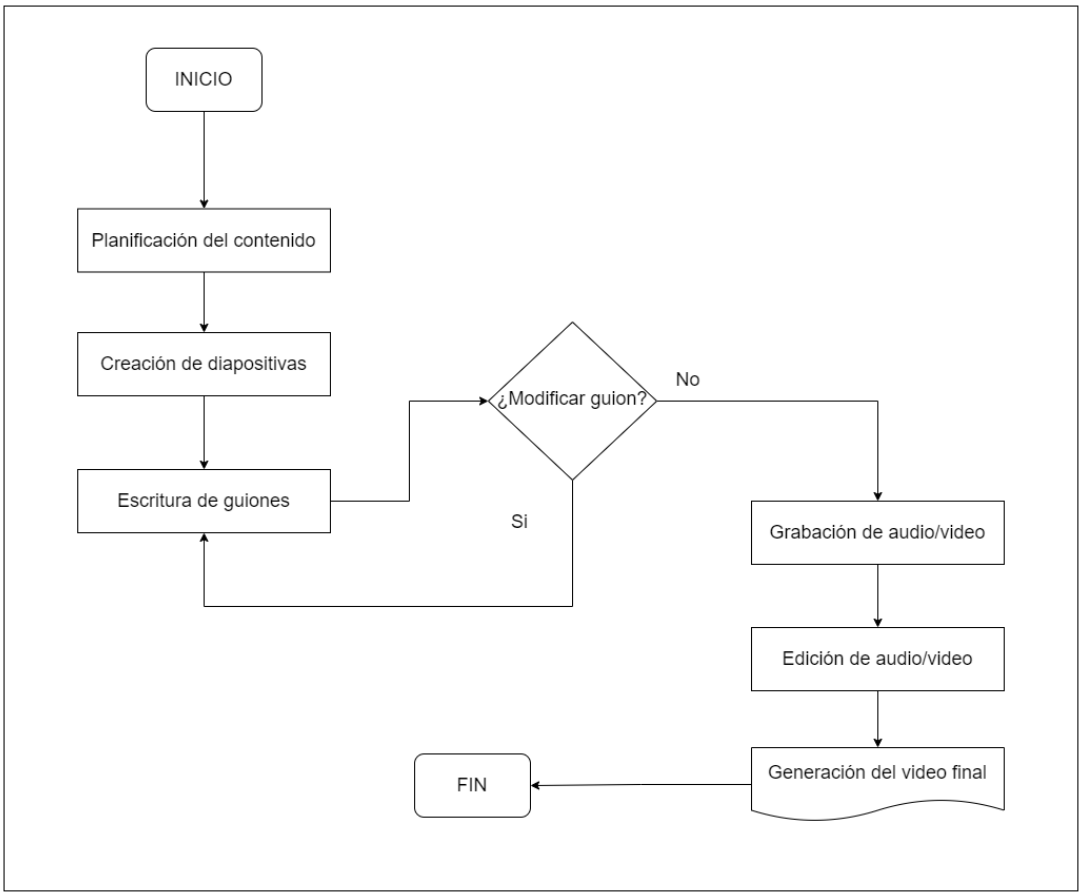


Figura 12. Diagrama de flujo del proceso propuesto

3.1.1 Factores que afectan la calidad del contenido

El análisis del proceso de producción de videos educativos ha identificado varios factores que impactan directamente en la calidad del contenido generado. Como se observa en la Figura 11, uno de los principales retos se encuentra en las etapas de *Creación de diapositivas*, *Redacción de guion*, y *Grabación*, donde se presentan cuellos de botella significativos. Estos cuellos de botella están relacionados con la elevada carga manual en la creación y edición del contenido, lo que no solo ralentiza el proceso, sino que también aumenta la probabilidad de errores y afecta la coherencia del material final.

Además, se evidencia que estas mismas etapas tienen un impacto negativo en la calidad del contenido debido a la falta de herramientas integradas que optimicen el flujo de trabajo. La etapa de *Edición*, aunque crucial para mejorar la presentación, también representa un punto crítico, dado que cualquier inconsistencia en las

fases previas se amplifica aquí. Finalmente, las oportunidades de automatización, representadas en verde en la Figura 11, destacan áreas donde el proceso puede optimizarse significativamente, especialmente en las fases de *Planificación* y *Creación de diapositivas*, reduciendo errores y tiempos de producción.

3.2 Análisis de herramientas de desarrollo

En la siguiente sección se presenta un análisis detallado de las herramientas de desarrollo que serán empleadas para la implementación del sistema de generación de contenido audiovisual. Se abordan tanto frameworks como tecnologías y metodologías que mejor se ajustan a los requerimientos del proyecto, considerando aspectos clave como la eficiencia, escalabilidad y flexibilidad en los procesos de producción de material audiovisual. El objetivo es seleccionar las herramientas que optimicen la calidad y el tiempo de desarrollo del sistema, alineándose con las necesidades específicas del entorno educativo digital.

3.2.1 Análisis y selección de tecnologías

Dado que el sistema de generación de contenido audiovisual será una aplicación de escritorio, es esencial seleccionar las tecnologías adecuadas para su implementación. Entre estas, se incluye un framework para el desarrollo de aplicaciones de escritorio, LLM para la generación de contenido y guiones, herramientas de TTS, un editor de video que permita la manipulación eficiente del contenido multimedia, y un lenguaje de programación que optimice el rendimiento y facilite la integración de los distintos componentes. Estas tecnologías serán evaluadas en función de su capacidad para satisfacer las necesidades del proyecto y garantizar su éxito.

a. LLM

En la revisión bibliográfica se mencionaron LLMs como herramientas clave para la generación de contenido. Específicamente, se destacan GPT, LLAMA2, Gemini y PALM2, que son capaces de crear guiones y textos educativos de manera efectiva. Estas tecnologías permitirán optimizar el proceso de producción de contenido audiovisual en la aplicación de escritorio.

- **ChatGPT:** El Generative Pre-trained Transformer (GPT) utiliza la arquitectura Transformer y es entrenado inicialmente con un gran corpus de texto, seguido de ajustes para tareas específicas. Estudios como [10], [14], [18], y [21] resaltan su eficacia en la generación de guiones, ejercicios y materiales educativos interactivos. Estos guiones pueden ser fácilmente convertidos en audio mediante software TTS, lo que hace a GPT una herramienta eficiente para la producción de contenido multimedia.

- **LLAMA2:** LLAMA2 es un modelo de lenguaje avanzado que mejora la eficiencia y generalización en tareas de procesamiento del lenguaje. Evaluado en el benchmark FOFO, ha demostrado una gran capacidad para seguir formatos complejos [11], [22]. Su precisión lo convierte en una herramienta prometedora para generar guiones específicos que luego pueden convertirse en archivos de audio mediante TTS, optimizando la producción de contenido multimedia.

- **Gemini y Gemini Pro:** Gemini y Gemini Pro son modelos de IA conversacional diseñados para integrar búsqueda en tiempo real y procesamiento de lenguaje complejo. Estudios como [21] y [11] destacan su precisión y coherencia en tareas de gestión de contenido y soporte al cliente. En el benchmark FOFO, Gemini Pro mostró habilidades superiores en el seguimiento de formatos detallados, ideal para aplicaciones que requieren alta precisión.

- **PALM2 for text:** PALM2 for text, desarrollado por la Universidad de Pekín y Microsoft Research Asia, aborda desafíos complejos en procesamiento de texto. Estudios como [11] y [22] subrayan su capacidad de comprensión contextual y generación de contenido relevante, haciéndolo útil en entornos educativos y profesionales, a pesar de los retos en tareas automatizadas complejas.

Tabla 5. Comparativa de LLMs [11], [31], [32]

LLM	FOFO SCORE	LLM ARENA SCORE	MMLU SCORE
GPT-4o	91.17	1337	0.842
Llama 2	45.44	1093	0.695
Gemini Pro	80.25	1299	0.827
PaLM for text	83.72	1004	0.692

En el contexto de los modelos de lenguaje, se emplean diversas métricas estandarizadas para evaluar su rendimiento en tareas de generación de contenido educativo y comprensión de información compleja. Entre las más relevantes se encuentran el FOFO Score [11], el LLM ARENA Score [31] y el MMLU Score [32].

El FOFO Score evalúa la capacidad de un modelo para seguir formatos estructurados específicos, midiendo su precisión y consistencia al generar texto con estas estructuras complejas. Un puntaje de 91.17 en este benchmark sitúa a GPT-4o como el modelo más preciso en este aspecto, superando significativamente a otros modelos como Llama 2, PALM y Gemini Pro. Esta alta puntuación refleja una habilidad sobresaliente para adherirse a los formatos especificados, lo cual es crucial para la creación de contenido educativo bien organizado y con una estructura coherente, facilitando así la comprensión y el aprendizaje.

El LLM ARENA Score se obtiene a través de una comparación a ciegas de las respuestas de dos modelos de lenguaje en una amplia variedad de tareas. En cada comparación, los usuarios votan por la respuesta que prefieren sin saber qué modelo la generó, basándose en factores como la precisión, coherencia y utilidad de la respuesta. Este puntaje no tiene un límite superior fijo, sino que sirve para comparar el desempeño relativo de los modelos entre sí. Un puntaje alto, como el 1337 de GPT-4, indica que este modelo fue elegido por los usuarios con mayor frecuencia en estas comparaciones, destacándose por encima de otros modelos como Gemini Pro (1299) y Llama 2 (1093). Esto sugiere que GPT-4 cumple consistentemente con las expectativas de los usuarios, ofreciendo respuestas más satisfactorias en comparación con los demás modelos evaluados.

El MMLU Score (Massive Multitask Language Understanding) es una métrica que evalúa la precisión de un modelo en un amplio conjunto de preguntas que abarcan más de 50 disciplinas, como matemáticas, historia, ciencias y tecnología. Este benchmark está diseñado para medir la capacidad del modelo de responder correctamente en áreas muy variadas, imitando el conocimiento que se esperaría de un humano bien informado en múltiples campos. El puntaje MMLU se calcula como el porcentaje de respuestas correctas en estas pruebas. Un MMLU Score de 0.842 para GPT-4, por ejemplo, indica que el modelo respondió correctamente el 84.2% de las preguntas, lo

cual es un indicador de alto rendimiento, ya que muestra una capacidad avanzada para manejar temas complejos y especializados, superando a todos los modelos.

Estos resultados, como se muestra en la Tabla 5, destacan a ChatGPT como una herramienta altamente eficaz para la generación de contenido educativo de calidad y la comprensión de tareas avanzadas en diversos dominios.

b. TTS

De acuerdo con la revisión bibliográfica, las tecnologías de TTS juegan un rol crucial en la generación de narraciones para proyectos educativos y multimedia. Entre las más destacadas se encuentran Microsoft Azure TTS, Google Cloud TTS, WaveNet, IBM Watson TTS, Tacotron 2, Nuance y Amazon Polly, todas ellas reconocidas por su capacidad de convertir texto en voz con alta calidad y naturalidad. Estas herramientas no solo facilitan la creación de contenido accesible, sino que también optimizan la producción de narraciones en múltiples idiomas, mejorando la interacción del usuario.

- **Microsoft Azure Text-to-Speech** Ofrece más de 200 estilos y 85 idiomas, siendo ideal para proyectos multilingües. Su alta calidad ha sido destacada en estudios como [12] y [23], con puntuaciones altas en Inteligibilidad (5.6) y Naturalidad (5.35). Es fundamental en la creación de contenido multimedia educativo, especialmente al convertir guiones en narraciones claras y comprensibles.

- **Google Cloud Text-to-Speech** Convierte texto en señales de audio de alta calidad mediante redes neuronales profundas, generando voces realistas. Estudios como [12] y [23] lo destacan por su personalización y claridad en múltiples idiomas, con puntuaciones de Inteligibilidad (5.2) y Naturalidad (3). Es útil en la automatización de videos educativos, optimizando la calidad del audio.

- **IBM Watson Text-to-Speech** Utiliza procesamiento de lenguaje natural y redes neuronales para convertir texto en habla de alta calidad. En [12] y [23], obtuvo puntuaciones de Inteligibilidad (5.3) y Naturalidad (5.3). Su aplicación mejora la accesibilidad y comprensión del contenido educativo, haciendo más efectivos los proyectos de automatización de videos.

- **Nuance** Soporta 53 idiomas y 199 voces, ofreciendo amplia personalización según las necesidades del proyecto. En [12] y [23], destacó por su adaptabilidad en la creación de videos educativos automatizados, asegurando contenido atractivo y accesible para audiencias diversas.

- **Amazon Polly** Ofrece 29 idiomas y 60 voces, enfocándose en la claridad y naturalidad del habla generada mediante aprendizaje profundo. En [12], se resaltó su capacidad para mejorar la experiencia en aplicaciones educativas y multimedia, facilitando la creación de contenido interactivo y accesible.

Tabla 6. Comparativa de motores TTS [12], [23]

Características	**Nuance**	**Amazon**	**IBM**	**Azure**	**Google**
# Idiomas	53	29	13	50	40
# Voces	199+	60	38	200	220
Costo por carácter ($)	-	0.000004	0.0000216	0.0000155	0.000014
Inteligibilidad	-	-	5.4	5.6	5.2
Naturalidad	-	-	5.3	5.4	3.0
Rendimiento (ms)	-	-	8965.7	252.45	564.3

Tras realizar una investigación preliminar, se identificaron diversas tecnologías de TTS relevantes para este proyecto, como Microsoft Azure TTS, Google Cloud TTS, IBM Watson TTS, Nuance y Amazon Polly. No obstante, debido a la falta de información completa sobre algunas de estas herramientas, se seleccionaron solo tres para un análisis más detallado: Microsoft Azure, Google Cloud y IBM Watson. Según estudios como [12], en los que se evaluó el rendimiento pasando un texto en inglés de 200 tokens 20 veces y calculando el promedio, Microsoft Azure TTS destacó por su rendimiento superior. Azure ofrece más de 200 estilos de voz en 85 idiomas, lo que lo convierte en una opción ideal para proyectos multilingües. En términos de calidad, obtuvo puntajes sólidos en Naturalidad (5.4) e Inteligibilidad (5.6) sobre una escala de 6, lo que demuestra su capacidad para generar voces claras y comprensibles, con una naturalidad cercana a la de una voz humana real. Estos resultados posicionan a Azure como una de las mejores opciones para generar voces fluidas y fáciles de entender, lo cual es clave en aplicaciones educativas y multilingües. Además, en cuanto a rendimiento, registró un tiempo promedio de respuesta de 252.45 ms. Un menor tiempo de respuesta significa que la plataforma procesa más rápido las solicitudes, lo que mejora la fluidez de la

experiencia del usuario, especialmente en aplicaciones interactivas donde la rapidez en la respuesta es crucial. Esto coloca a Azure muy por delante de sus competidores en esta métrica. Como se observa en la Tabla 6, estos resultados confirman que Microsoft Azure TTS es la mejor opción para la creación de narraciones educativas eficientes y de alta calidad.

c. Framework de Desarrollo

En el mundo del desarrollo de aplicaciones, existen tres enfoques principales que los frameworks utilizan para construir software: frameworks nativos, frameworks basados en la web, y frameworks cross-platform. Cada uno tiene ventajas y limitaciones específicas en términos de rendimiento, costo, y compatibilidad. Los frameworks nativos permiten un alto rendimiento pero requieren desarrollo independiente para cada plataforma. Los frameworks web brindan flexibilidad para aplicaciones web, pero carecen de acceso directo a características de sistema nativas. Finalmente, frameworks cross-platform ofrecen una solución intermedia, permitiendo la creación de una sola base de código para múltiples plataformas, lo que reduce el tiempo de desarrollo y los costos asociados [33]. A continuación, se presenta una tabla comparativa que examina varios aspectos de estos tres enfoques.

Tabla 7. Comparativa de enfoques de desarrollo [33]

Aspecto	Cross-Platform	Nativo	Web Apps
Código Reutilizable	+80% reutilizable en múltiples plataformas	Necesitas código específico para cada plataforma	Reutilizable, pero dependiente de navegadores
Costos	Menores (un solo equipo y código)	Altos (equipos separados y código duplicado)	Bajos, fácil encontrar desarrolladores web
Rendimiento	Bueno, cercano a nativo con optimización	Excelente, adaptado a la plataforma específica	Depende del navegador, con menor rendimiento
Acceso a Funcionalidades Nativas	Requiere puentes para módulos nativos específicos	Acceso completo a las funcionalidades nativas	Limitado, especialmente sin conexión a Internet
UI/UX	Consistente entre plataformas	UI/UX adaptada a cada sistema	Menos pulida, depende de la compatibilidad del navegador
Actualizaciones	Un solo código, fácil de mantener	Cada plataforma necesita sus propias actualizaciones	Actualizaciones automáticas vía navegador

La Tabla 7 destaca cómo los frameworks cross-platform son la opción ideal para proyectos que buscan un equilibrio entre reutilización de código, rendimiento y costos. Para un proyecto que involucra edición de video, es esencial tener acceso al hardware del sistema, como la GPU, para un procesamiento eficiente de imágen, audio y video. Esta capacidad de optimización garantiza un funcionamiento fluido y permite aprovechar al máximo los recursos disponibles. A continuación, se llevará a cabo una comparativa de diferentes frameworks cross-platform, analizando sus características y beneficios.

Tabla 8. Comparativa de frameworks de desarrollo cross-platform

Aspecto	Electron	Tauri	NW.JS	Flutter
Estrellas en Github	113.8k	82.4k	40.3k	165.2k
Última actualización	10/2024	10/2024	10/2024	10/2024
Lenguaje de uso (Backend)	JavaScript, C++	Rust	JavaScript, C++	Dart
Lenguaje de uso (Frontend)	HTML, CSS, JavaScript	HTML, CSS, JavaScript	HTML, CSS, JavaScript	Dart
Licencia	MIT	MIT	MIT	BSD 3-Clause
Soporte WebGL	Sí	Sí	Sí	Sí
Frameworks Frontend	Angular, React, Vue, Preact	Angular, React, Vue, Preact, Svelte	Angular, React, Vue, Preact	Ninguno
SO soportados	Windows, MacOS, Linux	Windows, MacOS, Linux	Windows, MacOS, Linux	Windows, MacOS, Linux
Tiempo de compilación	≈3002ms	≈174538ms	≈5080ms	≈32554ms
Uso de memoria	≈122MB	≈229MB	≈347MB	≈772MB

En la Tabla 8 al realizar una comparativa de frameworks de desarrollo cross-platform, se observa que Electron destaca como la mejor opción cuando se prioriza un ciclo de desarrollo rápido y una compatibilidad extensa con tecnologías web modernas como HTML, CSS y JavaScript. Con 113.8k estrellas en GitHub, Electron cuenta con la mayor popularidad y una comunidad sólida que respalda su continua actualización (10/2024). Su capacidad para integrar múltiples frameworks frontend como Angular, React, Vue y Preact, le proporciona gran flexibilidad a los desarrolladores. Además, su tiempo de compilación, aproximadamente 3002ms, es considerablemente más rápido que el de Tauri o NW.js, lo que facilita iteraciones rápidas durante el desarrollo. Electron también sobresale en eficiencia de recursos, con un consumo de memoria de 122MB, el más bajo entre los frameworks comparados. Estas características, junto

con su soporte para WebGL y compatibilidad con los principales sistemas operativos (Windows, MacOS, Linux), hacen de Electron la opción ideal para proyectos donde la rapidez y eficiencia son esenciales.

d. Metodologiá Ágil

De acuerdo con la revisión bibliográfica, las metodologías ágiles juegan un rol fundamental en la gestión eficiente de proyectos de software. Entre las más destacadas se encuentran Rapid Application Development (RAD), Extreme Programming (XP) y Kanban, cada una reconocida por su enfoque en mejorar la flexibilidad y la productividad en el desarrollo. RAD se enfoca en un desarrollo rápido mediante prototipos y retroalimentación continua, XP prioriza la adaptabilidad mediante entregas frecuentes y pruebas constantes, mientras que Kanban optimiza el flujo de trabajo visualizando tareas y limitando el trabajo en progreso. Estas metodologías no solo agilizan el ciclo de desarrollo, sino que también permiten una mejor alineación con las necesidades del cliente.

- **Rapid Application Development (RAD)** Es una metodología enfocada en el rápido desarrollo de software mediante la interacción continua con el cliente a través del prototipado. El enfoque principal de RAD es el desarrollo rápido y la validación de requerimientos a medida que se avanza, reduciendo los tiempos de entrega al cliente [34].

  Ciclo de Vida de RAD

  - **Recopilación de requisitos:** Interacción directa con los usuarios para definir los requisitos.
  - **Prototipado:** Desarrollo rápido de prototipos funcionales.
  - **Construcción:** Implementación del sistema basado en el prototipo aprobado.
  - **Pruebas:** Verificación continua de la funcionalidad.
  - **Implementación:** Entrega y despliegue del producto final.

- **Extreme Programming (XP)** Es una metodología ágil que pone énfasis en la satisfacción del cliente mediante la entrega frecuente de software funcional. Se

basa en ciclos cortos de desarrollo, con pruebas y retroalimentación constantes para adaptarse rápidamente a los cambios en los requisitos [35].

Ciclo de Vida de XP:

- **Planificación:** Definir historias de usuario y requisitos clave.
- **Diseño:** Elaborar un diseño básico y adaptable del sistema.
- **Desarrollo:** Implementar funcionalidades en ciclos cortos con pruebas continuas.
- **Pruebas:** Pruebas unitarias y de integración durante todo el ciclo.
- **Entrega:** Entregar versiones funcionales del software frecuentemente.

- **Kanban** Es un enfoque visual para gestionar tareas en un flujo de trabajo. Su principal objetivo es mejorar la eficiencia visualizando el progreso de las tareas, limitando el trabajo en progreso (WIP) y optimizando continuamente el flujo de trabajo [36].

Ciclo de Vida de Kanban:

- **Mostrar el proceso:** Visualizar las tareas en un tablero Kanban.
- **Limitar el trabajo en progreso (WIP):** Establecer un límite en las tareas simultáneas.
- **Optimizar el flujo de trabajo:** Mejorar continuamente el proceso.

Tabla 9. Comparativa de Metodologiás Ágiles [34], [35], [36]

Criterios	RAD	XP	Kanban
Velocidad de desarrollo	Alta	Alta	Media
Flexibilidad	Media	Alta	Alta
Involucramiento del cliente	Alto	Muy alto	Medio
Adecuado para	Proyectos pequeños y medianos	Proyectos pequeños y medianos	Todo tipo de proyectos
Riesgo de cuellos de botella	Bajo	Medio	Alto
Costo de implementación	Bajo a medio	Bajo a medio	Bajo
Mejor aplicación	Entregas rápidas	Cambios frecuentes	Optimización del flujo

Luego de analizar la Tabla 9, se puede concluir que RAD emerge como la mejor opción para proyectos que requieren un desarrollo ágil y una constante validación por parte del cliente. Esta metodología no solo acelera el proceso a través del uso de prototipos funcionales, sino que también permite iteraciones rápidas con retroalimentación continua. Al involucrar al usuario desde las primeras etapas, RAD reduce significativamente el riesgo de desviarse de los objetivos y asegura una mayor alineación entre el producto final y las expectativas del cliente. Esto hace que RAD sea especialmente adecuado para proyectos pequeños y medianos, donde el tiempo de entrega y la flexibilidad son factores críticos.

3.3 Desarrollo de la propuesta

En esta sección se describe el desarrollo de un sistema para automatizar la creación de contenido audiovisual en la empresa Ingedemy. La metodología seleccionada es RAD, previamente elegida como una opción adecuada para proyectos que requieren rapidez en el desarrollo sin sacrificar la calidad. Este enfoque permite una interacción constante con los usuarios y la entrega de prototipos funcionales de manera iterativa, lo que facilita ajustes y mejoras continuas. A continuación, se describen las principales fases que componen este ciclo de vida:

Primero, en la fase de Recopilación de requisitos, el usuario y el analista identifican

los objetivos y necesidades del sistema. Luego, en la fase de Prototipado, se crean versiones iniciales para obtener retroalimentación temprana. En la fase de Construcción, se implementa el sistema final, ajustando el prototipo según sea necesario. La fase de Pruebas garantiza el correcto funcionamiento del sistema, corrigiendo errores. Finalmente, en la fase de Implementación, el sistema es entregado y desplegado en su entorno operativo.

3.3.1 Planificación de requerimientos

En esta primera fase de Recopilación de requisitos, se realizó una especificación detallada a través de entrevistas con los usuarios clave. Estas entrevistas permitieron identificar los objetivos principales del sistema y los requisitos necesarios para su desarrollo. En la Tabla 10 se especifican los requerimientos establecidos para el desarrollo del proyecto, basados en la información recolectada durante las entrevistas.

Tabla 10. Requerimientos del Sistema

ID	Requerimiento	Descripción	Prioridad	Riesgo
R1	Crear proyecto	El usuario puede crear un nuevo proyecto para organizar videos.	Alta	Bajo
R2	Eliminar proyecto	El usuario puede eliminar un proyecto y todos sus contenidos asociados.	Media	Bajo
R3	Editar proyecto	El usuario puede editar el nombre y etiquetas de un proyecto existente.	Media	Bajo
R4	Búsqueda de proyectos	El usuario puede buscar proyectos por nombre.	Alta	Bajo
R5	Filtrar proyectos por etiquetas	El usuario puede filtrar los proyectos usando etiquetas definidas.	Media	Bajo
R6	Visualización de proyectos	Los proyectos se muestran en tarjetas con su logotipo, nombre y etiquetas.	Alta	Bajo
R7	Visualizar videos en proyecto	El usuario puede ver los videos asociados a un proyecto en formato de tabla.	Alta	Bajo

ID	Requerimiento	Descripción	Prioridad	Riesgo
R8	Crear video desde cero	El usuario puede crear un video seleccionando nombre, idioma y formato.	Alta	Medio
R9	Crear video con IA	El usuario puede generar un video utilizando IA.	Alta	Alto
R10	Edición de diapositivas	El usuario puede editar diapositivas generadas o crear nuevas diapositivas manualmente.	Alta	Alto
R11	Búsqueda de imágenes externas	El sistema permite buscar imágenes externas para agregarlas a las diapositivas.	Media	Medio
R12	Agregar IA a diapositivas	El usuario puede utilizar IA para generar diapositivas con contenido textual e imágenes.	Alta	Medio
R13	Crear y modificar guiones	El usuario puede crear y modificar guiones para vincular a las diapositivas.	Alta	Medio
R14	Guardar y editar proyectos en progreso	El usuario puede guardar un proyecto incompleto y retomarlo más tarde.	Alta	Bajo
R15	Selección de voz	El usuario puede seleccionar una voz IA para narrar el contenido de los guiones.	Alta	Medio
R16	Previsualización de guiones	El usuario puede previsualizar cómo sonarán los guiones antes de generar el video.	Alta	Medio
R17	Reescritura de guion con IA	El usuario puede solicitar que IA reescriba los guiones según directrices.	Alta	Alto
R18	Reemplazo de palabras en guiones	El usuario puede reemplazar palabras en los guiones para ajustar pronunciaciones.	Media	Medio
R19	Generación de video final	El sistema genera un video en formato MP4, combinando diapositivas y guiones.	Alta	Medio
R20	Descarga de video	El usuario puede descargar el video generado en formato MP4.	Alta	Bajo

Después de haber establecido los requerimientos necesarios, se definió el siguiente plan de iteraciones para estructurar y controlar el desarrollo del proyecto. Cada iteración busca implementar un conjunto específico de funcionalidades, garantizando una entrega gradual y controlada de las características clave del sistema.

Tabla 11. Planificación de Iteraciones

Iteración	**N.**	**ID**	**Requerimiento**	**Tiempo estimado**	
				Horas	**Días**
Iteración 1	1	R1	Crear proyecto	8	1
	2	R2	Eliminar proyecto	6	1
	3	R3	Editar proyecto	8	1
	4	R4	Búsqueda de proyectos	6	1
	5	R5	Filtrar proyectos por etiquetas	8	1
	6	R6	Visualización de proyectos	10	2
	7	R7	Visualizar videos en proyecto	12	2
Iteración 2	8	R8	Crear video desde cero	10	2
	9	R9	Crear video con IA	15	3
	10	R10	Edición de diapositivas	12	2
	11	R11	Búsqueda de imágenes externas	8	1
	12	R12	Agregar IA a diapositivas	10	2
	13	R13	Crear y modificar guiones	12	2
Iteración 3	14	R14	Guardar y editar proyectos en progreso	8	1
	15	R15	Selección de voz	10	2
	16	R16	Previsualización de guiones	8	1
	17	R17	Reescritura de guion con IA	10	2
	18	R18	Reemplazo de palabras en guiones	6	1
	19	R19	Generación de video final	15	3
	20	R20	Descarga de video	6	1

3.3.2 Diseño de usuario

a. Análisis del proceso propuesto

En el siguiente esquema se detalla el proceso de automatización para la generación de contenido educativo en formato de video mediante el uso de inteligencia artificial, donde:

1. Si el instructor no tiene un proyecto creado en la aplicación:
    (a) Procederá a crear un proyecto utilizando la funcionalidad "+ Crear proyecto" en la pantalla de inicio.
    (b) Asignará un nombre y etiquetas al proyecto para una mejor organización.
2. El instructor podrá crear un video dentro del proyecto seleccionado, donde:
    (a) Si decide crear un video desde cero:
        i. Se solicitará el nombre del video, idioma y resolución.
    (b) Si decide crear un video con IA:
        i. Se solicitarán parámetros como el número de diapositivas, cantidad de texto, tipo de entrada (idea o párrafo), y directrices específicas para generar contenido.
3. Una vez generado el video:
    (a) El instructor podrá acceder a la pantalla de edición de diapositivas, donde podrá:
        i. Crear nuevas diapositivas usando IA.
        ii. Editar el contenido generado por IA o añadir contenido manualmente.
        iii. Utilizar plantillas, imágenes, y texto para diseñar la presentación.
        iv. Buscar y añadir recursos gráficos externos.
4. Cuando las diapositivas estén listas, se procederá a la generación del guion, donde:
    (a) Se podrán escribir los guiones manualmente o generarlos con IA.

(b) Se podrá seleccionar una voz artificial para previsualizar o generar el video final.

(c) Se podrá preescuchar cada guion con la voz seleccionada antes de la generación final.

(d) Se incluirá la funcionalidad de reescritura de guiones utilizando IA, para mejorar el contenido según las directrices indicadas.

5. El instructor podrá utilizar la funcionalidad de reemplazo de palabras:

   (a) Al seleccionar una palabra en el guion, se abrirá un formulario para sustituir la palabra en todos los guiones.

   (b) Esto es útil para corregir errores de pronunciación en la voz sintetizada, ajustando la ortografía para que se lea correctamente.

6. Finalmente, se procederá a la generación del video:

   (a) Al hacer clic en “Generar Video”, se creará el archivo en formato MP4 a partir de las diapositivas y guiones generados.

   (b) Una vez terminado, el video estará disponible para descarga, optimizando el tiempo de producción.

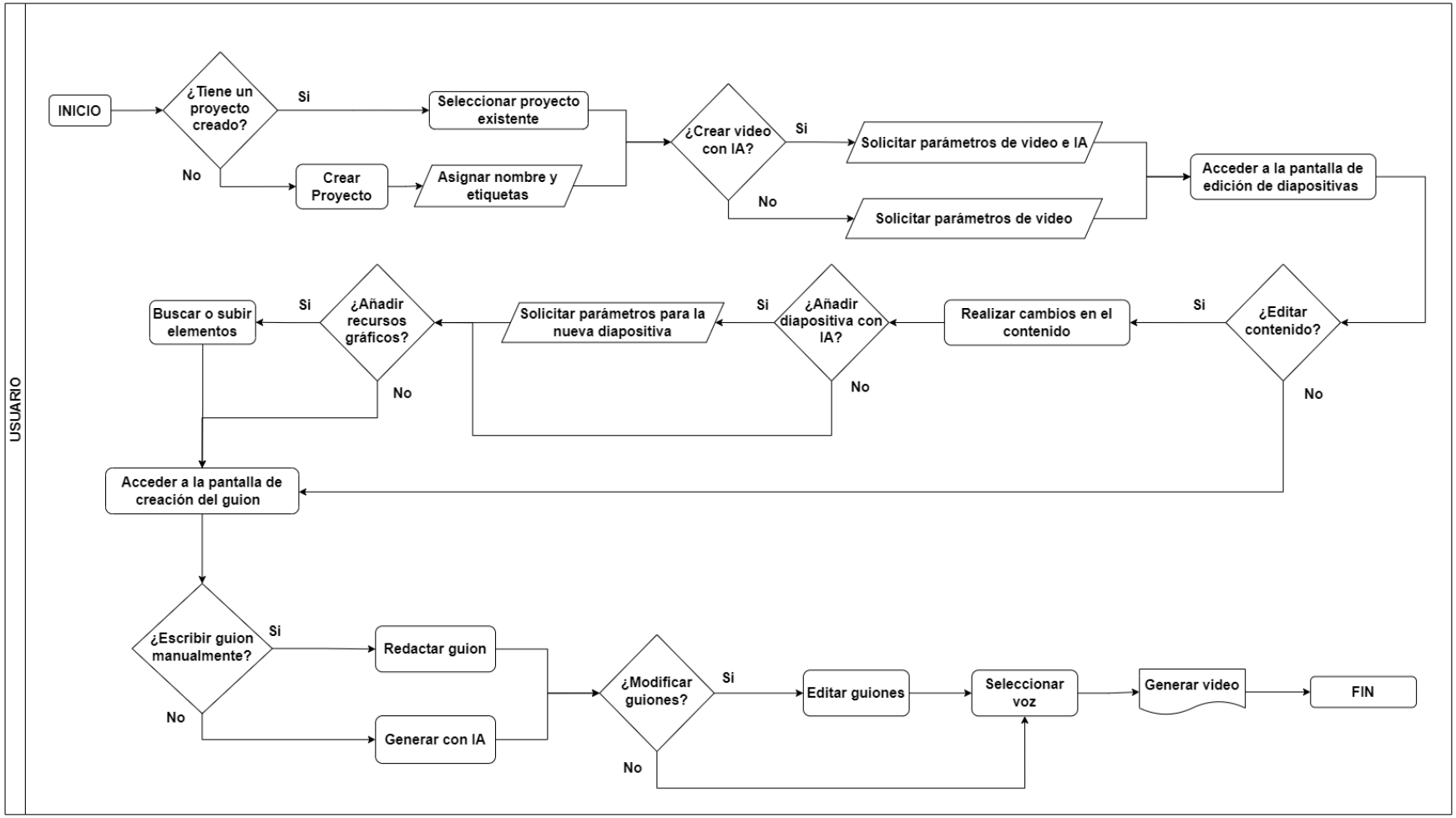


Figura 13. Diagrama de flujo del proceso propuesto

b. Arquitectura

La Figura 14 ilustra la arquitectura del sistema desarrollado para la automatización de la generación de contenido educativo en video mediante inteligencia artificial. La estructura está organizada en tres capas: **Capa de Presentación**, encargada de la interacción con el usuario a través de interfaces gráficas, y que utiliza APIs como ChatGPT para la generación de contenido textual y Azure Speech para síntesis de voz; **Capa de Negocio**, que gestiona la lógica del sistema mediante un servidor Node.js, orquesta las solicitudes entre componentes y realiza la generación multimedia mediante herramientas como FFmpeg, aprovechando comunicación entre procesos (IPCs); y **Capa de Datos**, que centraliza el almacenamiento y acceso a la información del sistema. Las tres capas están interconectadas mediante un flujo de peticiones y respuestas que garantizan la comunicación eficiente entre los módulos, permitiendo un proceso fluido desde la creación hasta la generación del contenido final.

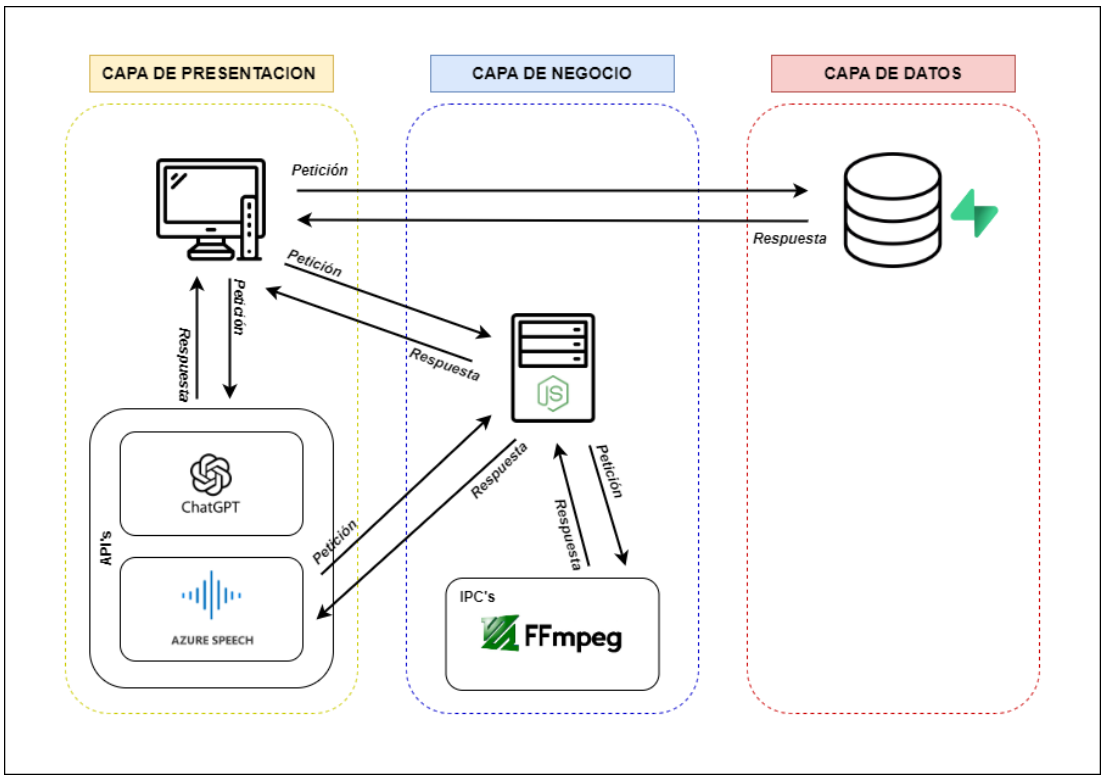


Figura 14. Arquitectura del Sistema

c. Prototipado

En esta fase de la metodología, se elaboraron prototipos con el propósito de mostrarlos a los instructores, facilitando así la validación previa antes de proceder con la etapa

de desarrollo. Esto permite recibir comentarios valiosos que ayudan a realizar ajustes, asegurando que las necesidades y expectativas de los usuarios estén adecuadamente cubiertas antes de avanzar.

En la Figura 15 se muestra una pantalla que permite gestionar y organizar los proyectos de contenido educativo. Los usuarios pueden crear nuevos proyectos a través del botón ”Crear proyecto” ubicado en la parte superior, y utilizar una barra de búsqueda para encontrar proyectos existentes por nombre o filtrar por categoría. Los proyectos se muestran en formato de tarjetas, cada una con su nombre, etiquetas y la opción de eliminarlos. Esta organización facilita la navegación y mantiene el contenido estructurado de manera eficiente.

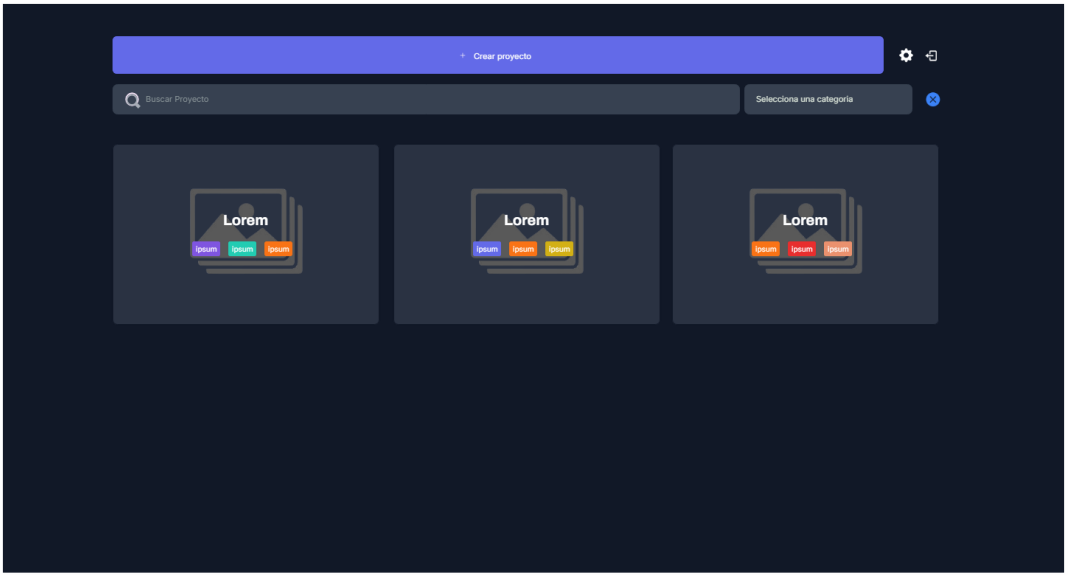


Figura 15. Prototipo de interfaz para la gestión de proyectos

El prototipo del formulario para la creación de proyectos, ilustrado en la Figura 16, está diseñado para que el usuario pueda ingresar el título del proyecto y asignar etiquetas que faciliten su organización. Este formulario incluye dos botones clave: uno para crear el proyecto con la información ingresada y otro para cancelar la acción. El diseño busca optimizar la experiencia de gestión y creación de proyectos dentro del sistema, haciéndolo más intuitivo y eficiente.

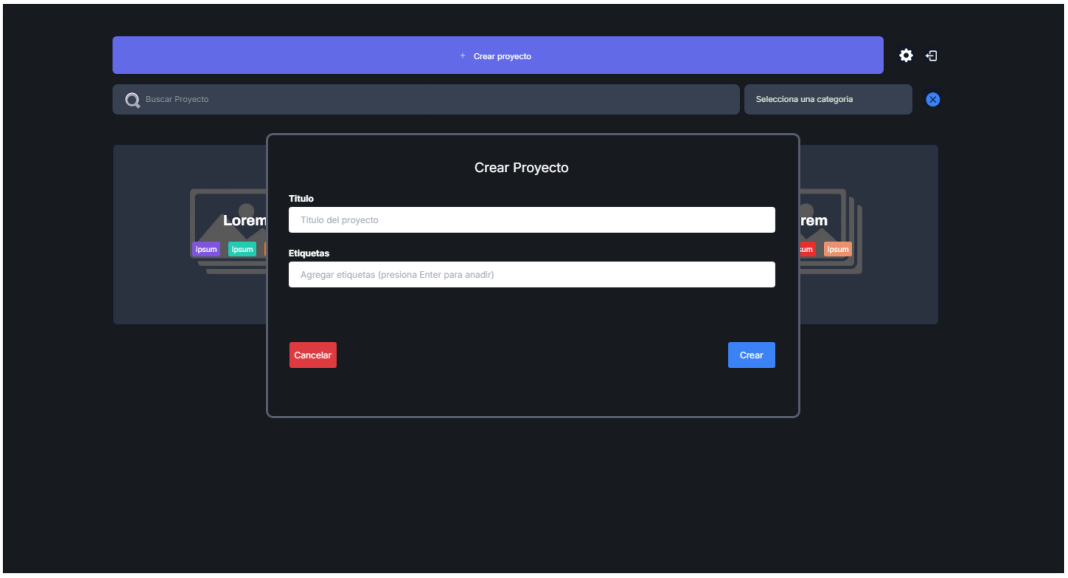


Figura 16. Prototipo del formulario para la creación de proyectos

En la Figura 17 se presenta el prototipo de la pantalla donde se listan los videos pertenecientes a un proyecto. En la parte superior, se muestra el nombre del proyecto junto con su costo total. Los videos están organizados en una tabla con columnas que indican el nombre, duración y fecha de creación de cada video. Además, se ofrecen opciones para editar o eliminar cada video. También incluye una barra de búsqueda para localizar videos específicos y un botón en la esquina superior derecha para crear nuevos videos. Este diseño permite una gestión sencilla y eficiente de los videos dentro de cada proyecto.

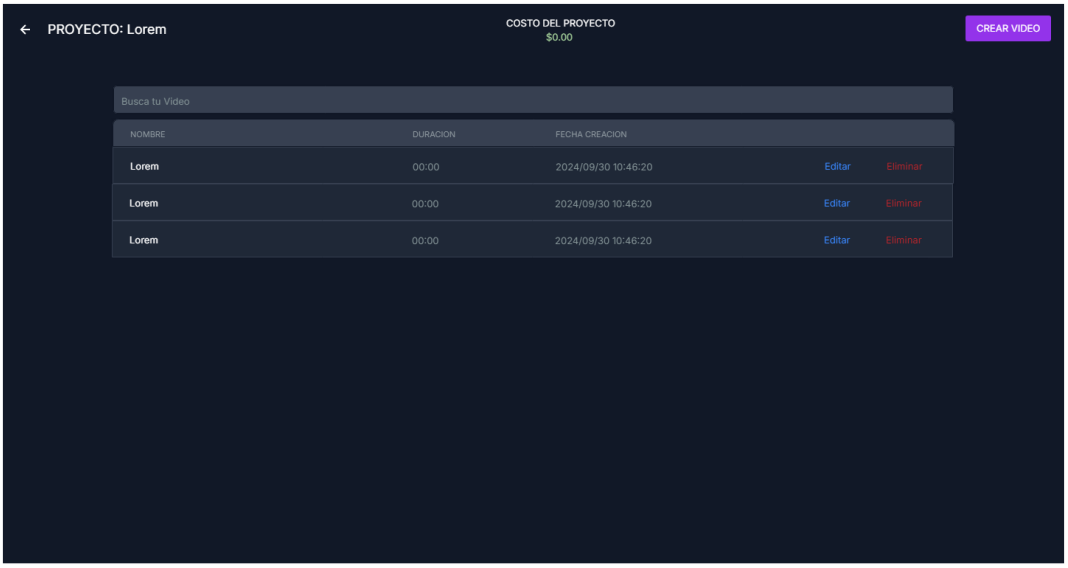


Figura 17. Prototipo de interfaz para la gestión de videos del proyecto

En la Figura 18 se muestra el formulario de creación de un video vacío, donde el usuario puede ingresar el nombre del video, seleccionar el tamaño de la presentación y el idioma, así como elegir entre crear el video desde cero o con la ayuda de IA. El formulario incluye opciones para cancelar o proceder con la creación mediante los botones correspondientes, facilitando así la configuración inicial del nuevo video dentro del proyecto.

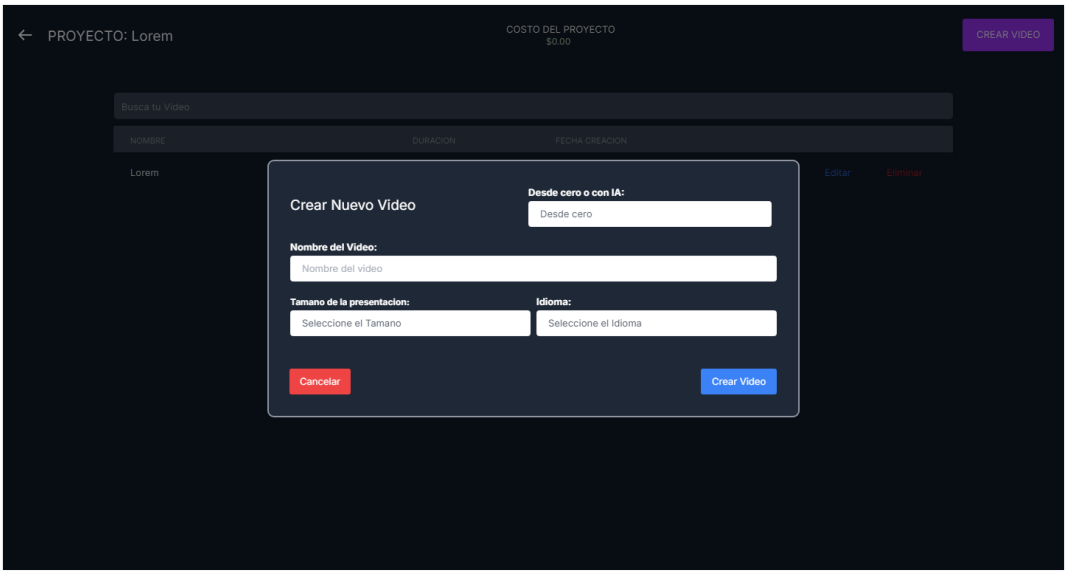


Figura 18. Prototipo del formulario para la creación de videos sin IA

En la Figura 19 se muestra el formulario para crear un video con IA, donde el usuario puede personalizar varios aspectos a través de menús desplegables. En "**Cantidad de Texto**", se puede elegir entre poca, media o alta, según la densidad de contenido textual deseada. En "**Tamaño de la Presentación**", se ofrecen tres opciones de formato: 1:1, 9:16 y 16:9, cada uno relacionado con diferentes tipos de visualización. En "**Tipo de Subida**", se selecciona si el video se generará a partir de un texto escrito o de una idea, que la IA desarrollará para crear el contenido. Además, se incluye un campo opcional para "**Directivas**", donde el usuario puede escribir instrucciones específicas que guiarán la creación del video, permitiendo una mayor personalización del resultado final.

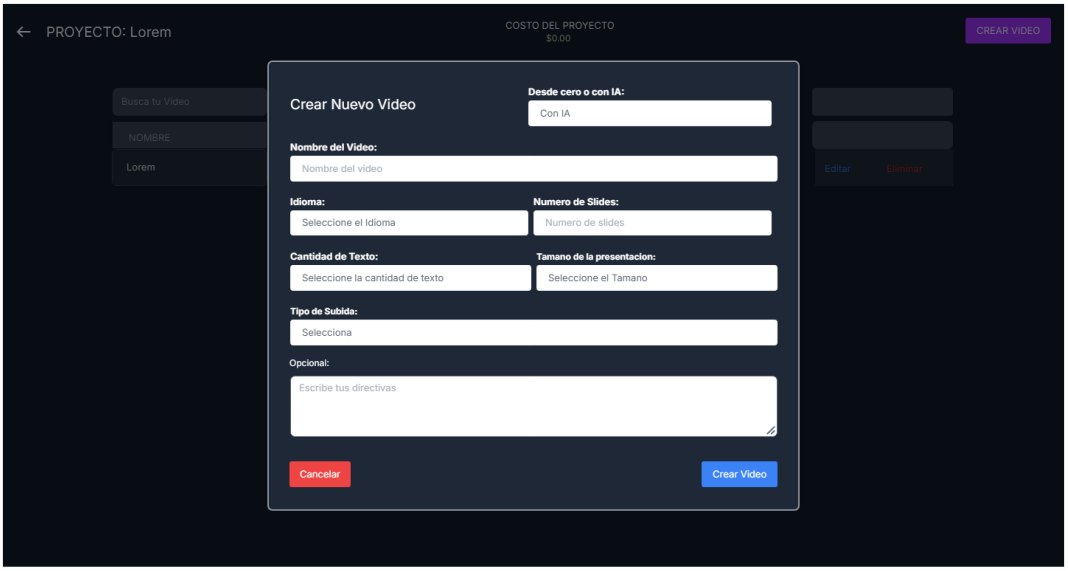


Figura 19. Prototipo del formulario para la creación de videos con IA

En la Figura 20 se presenta un editor de diapositivas donde el usuario puede crear o personalizar el contenido de un video. A la izquierda, se muestra un panel con diferentes categorías, como "**Plantillas**", "**Textos**", "**Fotos**", "**Elementos**"y "**Subir**", lo que permite al usuario añadir y modificar componentes visuales de la diapositiva. En el centro, se encuentra el área de edición principal, donde se arrastran los elementos seleccionados para diseñar el video. Además, en la parte superior derecha, el usuario tiene la opción de descargar las diapositivas editadas, facilitando el acceso y uso del material creado. El costo del video se actualiza en tiempo real según las nuevas diapositivas añadidas con IA.

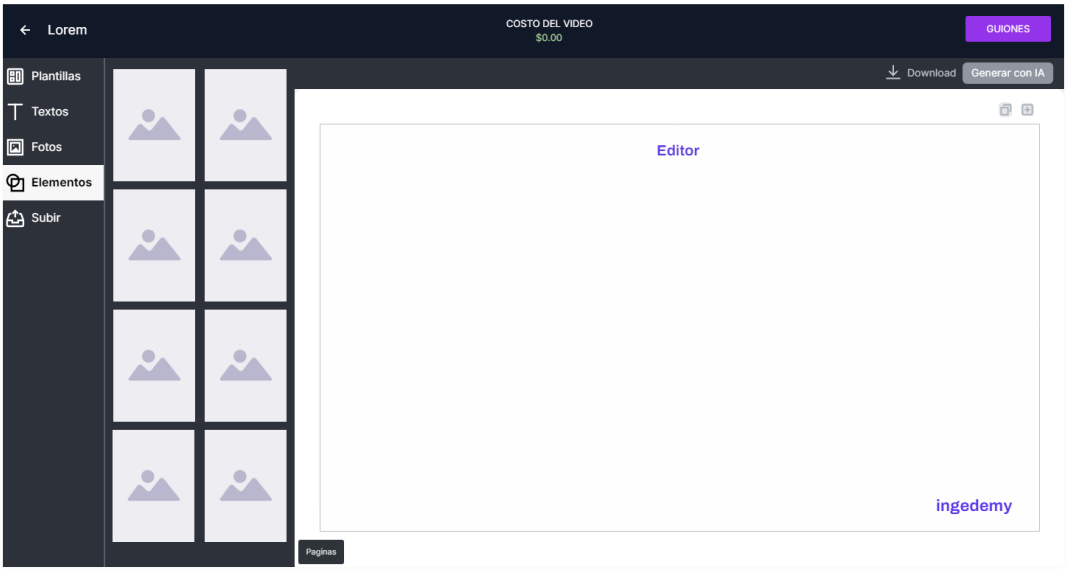


Figura 20. Prototipo de interfaz de edición de diapositivas

En la Figura 21 se presenta el formulario para añadir nuevas diapositivas con la ayuda de IA. Este formulario solicita tres parámetros principales. En primer lugar, el usuario debe seleccionar la cantidad de texto, que puede ser poca, media o alta, según la necesidad de contenido en la diapositiva. Luego, se debe ingresar el tema de la diapositiva, que será el concepto o idea principal que se desarrollará en el contenido. Finalmente, se ofrece un campo para directivas opcionales, donde el usuario puede añadir instrucciones específicas o detalles adicionales para guiar a la IA en la creación de la diapositiva.

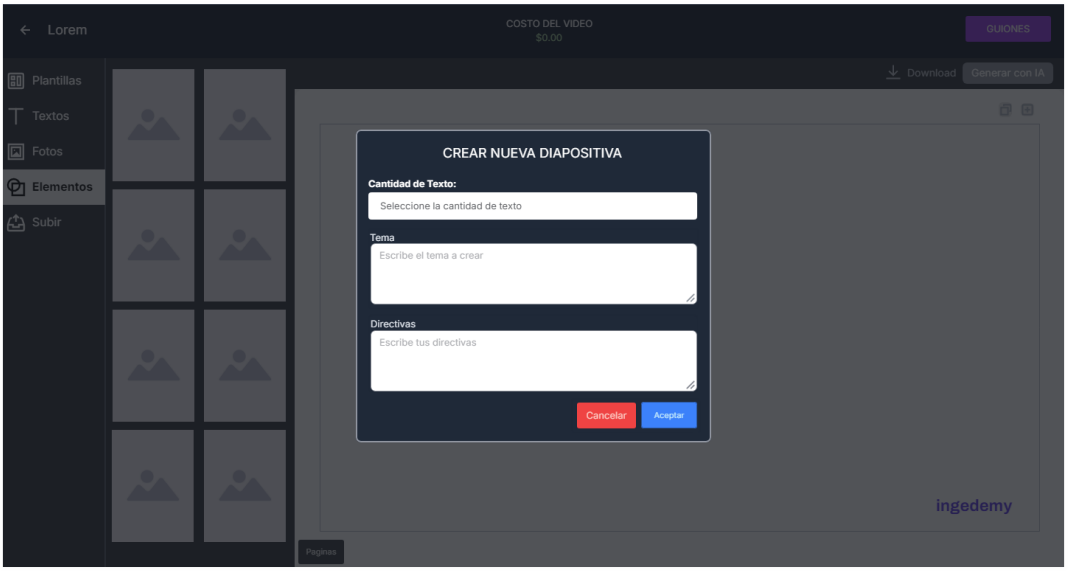


Figura 21. Prototipo del formulario para añadir nuevas diapositivas con IA

La Figura 22 presenta la pantalla donde el usuario tiene la posibilidad de generar guiones, ya sea manualmente o con la ayuda de la IA. A la izquierda se observa una vista previa de las diapositivas creadas, mientras que a la derecha se dispone del espacio para escribir o generar el guion correspondiente. El usuario puede reproducir el guion en tiempo real, seleccionar una voz para la narración y realizar ajustes antes de continuar con la producción. Además, se ofrece la opción de generar guiones automáticamente con IA, facilitando y agilizando el proceso creativo.

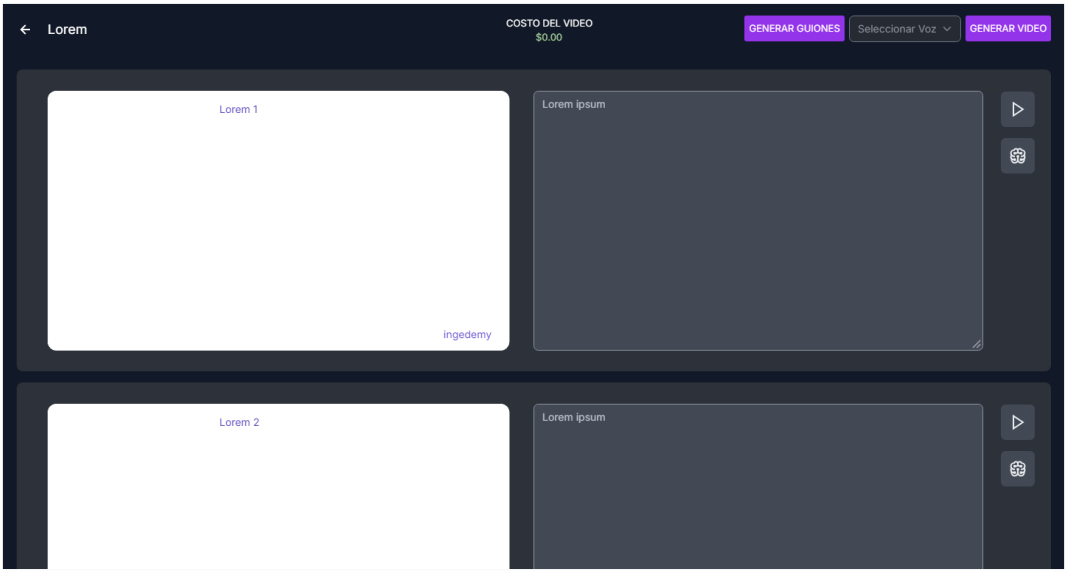


Figura 22. Prototipo de formulario para la creación de nuevas diapositivas con IA

La Figura 23 muestra el formulario que permite modificar un guion generado por IA. El usuario puede escribir directamente en el cuadro de texto el cambio o ajuste que desee realizar sobre el guion propuesto, lo que le ofrece flexibilidad para personalizar el contenido según sus necesidades. Una vez realizados los cambios, se puede confirmar la acción haciendo clic en ”Aceptar” o cancelar si se decide no realizar ninguna modificación.

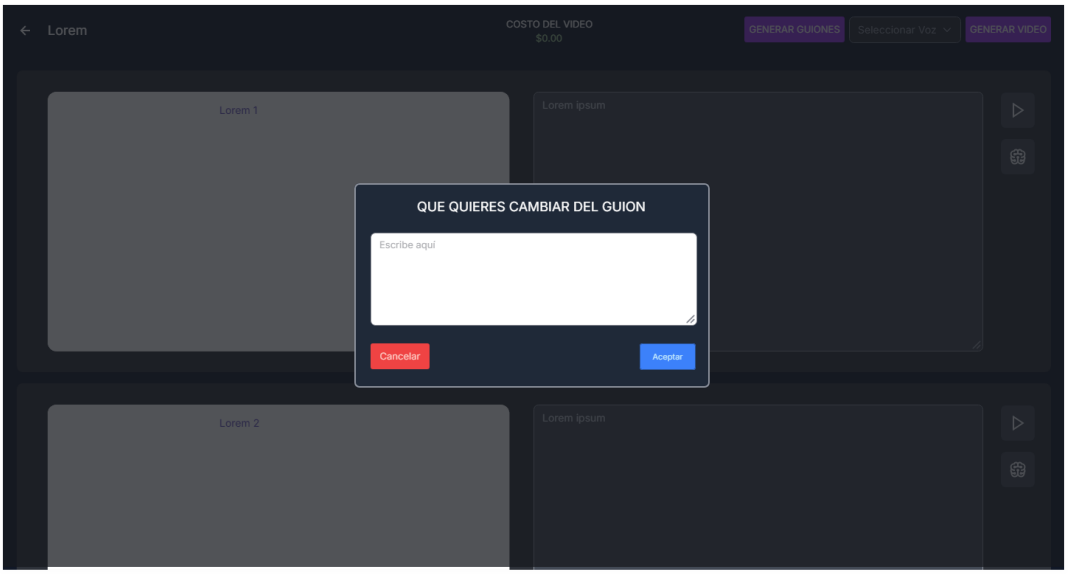


Figura 23. Prototipo del formulario para la reescritura de los guiones con IA

La Figura 24 muestra un formulario que permite al usuario reemplazar palabras o frases en el guion para mejorar cómo suena el texto cuando se narra. Esta funcionalidad está

diseñada para que el texto fluya de manera más natural, evitando que suene entrecortado o con pausas inadecuadas. Al hacer estos ajustes, el usuario puede asegurarse de que la narración sea más clara y fácil de entender.

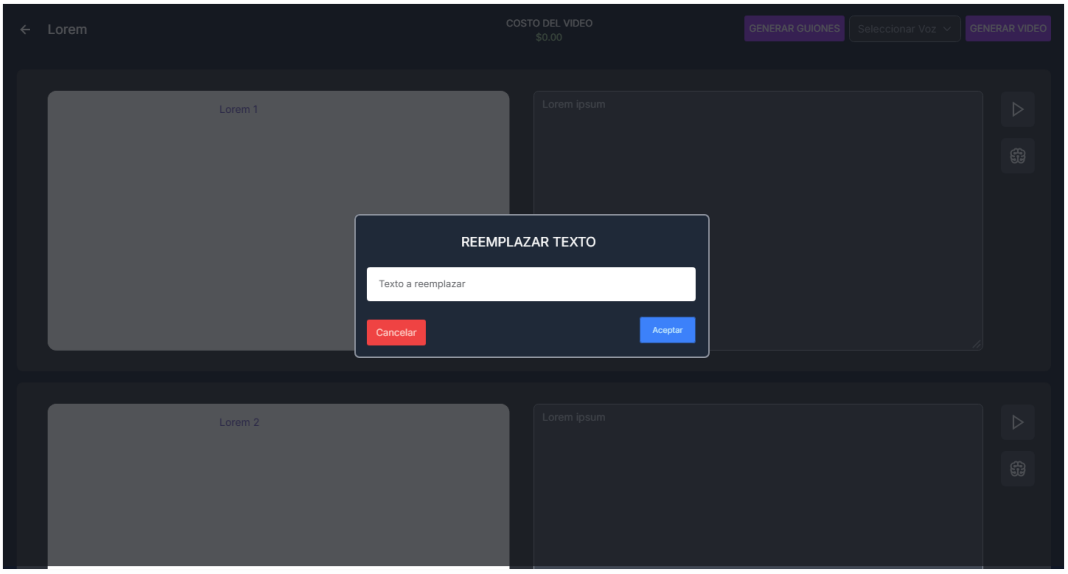


Figura 24. Prototipo de formulario para el reemplazo de palabras en el guion

La Figura 25 muestra la pantalla donde se puede visualizar el video final generado. Aquí, el usuario tiene la opción de reproducir el video y verificar el resultado antes de proceder a su descarga. Este paso asegura que el contenido cumple con las expectativas y es la versión definitiva lista para utilizarse o compartirse.



Figura 25. Prototipo de interfaz para la visualización del video resultante

La interfaz de la base de reemplazos permite gestionar las palabras que serán

modificadas automáticamente en los guiones generados. Como se observa en la Figura 26, se presentan dos columnas: una con el ”Texto Original” que el usuario desea cambiar, y otra con el ”Texto de Reemplazo” que lo sustituirá. Los campos de la columna ”Texto de Reemplazo” son editables, lo que permite ajustar las palabras de forma dinámica sin necesidad de eliminarlas. Además, cada fila cuenta con un botón para eliminar cualquier reemplazo previamente configurado, facilitando la gestión de cambios. En la parte superior derecha, el botón ”Añadir Reemplazo” permite agregar nuevas palabras a la lista.

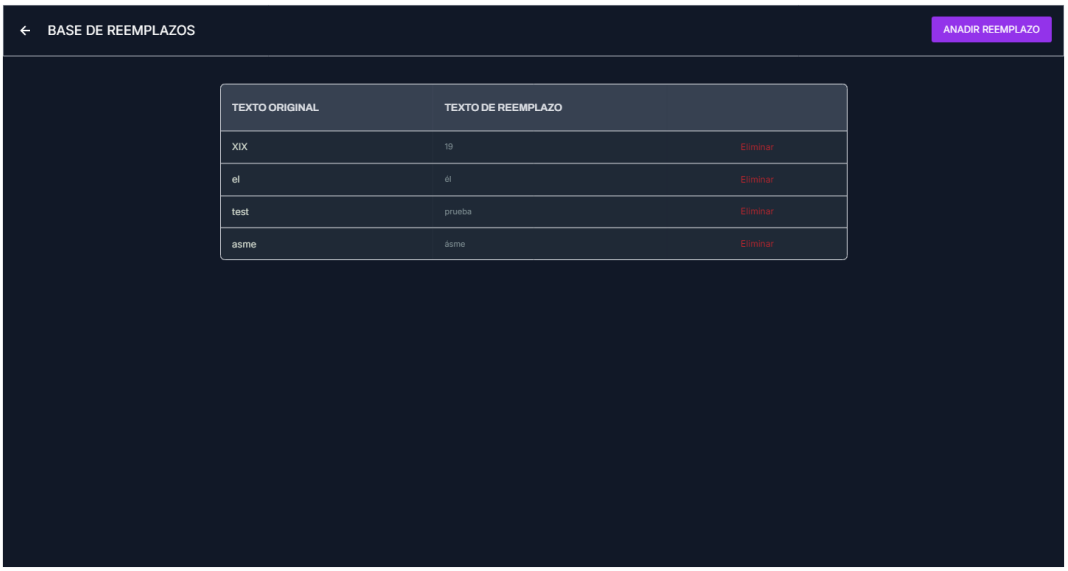


Figura 26. Prototipo de interfaz para la gestión de la base de reemplazos

El formulario para crear nuevos reemplazos, como se observa en la Figura 27, permite al usuario añadir palabras o frases que serán modificadas en el texto final. Cuenta con dos campos principales: ”Texto original”, donde se ingresa la palabra o frase que se desea reemplazar, y ”Texto a reemplazar”, donde se introduce el nuevo contenido que sustituirá al original. Al completar ambos campos, el usuario puede confirmar la operación presionando el botón ”Crear reemplazo”. En caso de no querer continuar, el botón ”Cancelar” permite descartar el proceso.

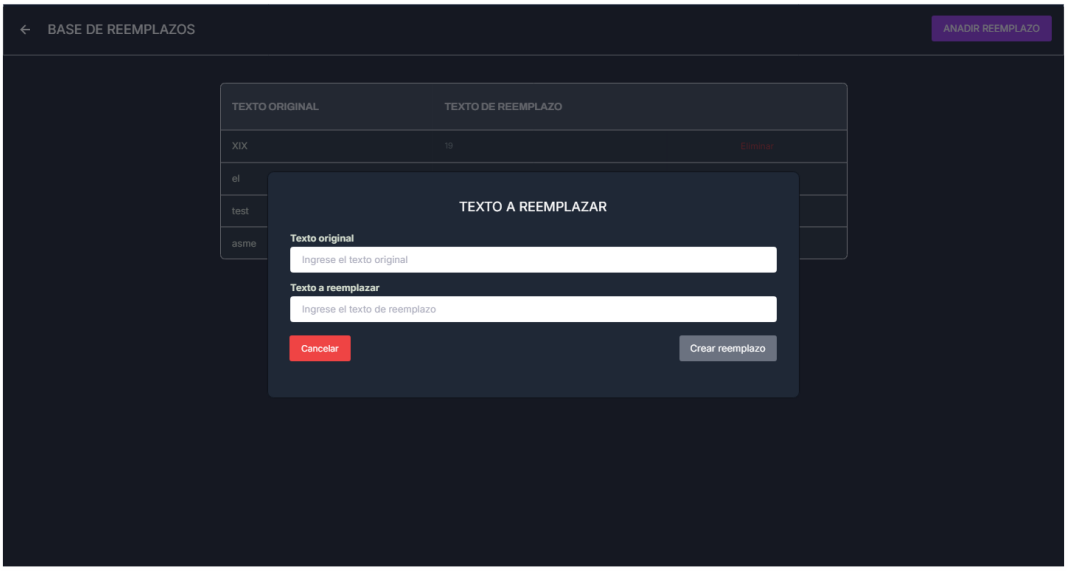


Figura 27. Prototipo del formulario para la creación de reemplazos

La interfaz de gestión de usuarios del módulo de administración, como se observa en la figura 17, permite la creación y eliminación de usuarios del sistema. En la sección de creación de usuarios, se deben ingresar los siguientes datos: correo electrónico, contraseña, y el rol del usuario. Estos campos son fundamentales para configurar el acceso de cada persona al sistema, asegurando que cuenten con los permisos correspondientes.

En la parte derecha de la pantalla, se encuentra la lista de usuarios ya registrados, donde se muestra su correo, el rol asignado y la opción para eliminarlos, permitiendo al administrador gestionar la seguridad y el acceso de manera eficiente.

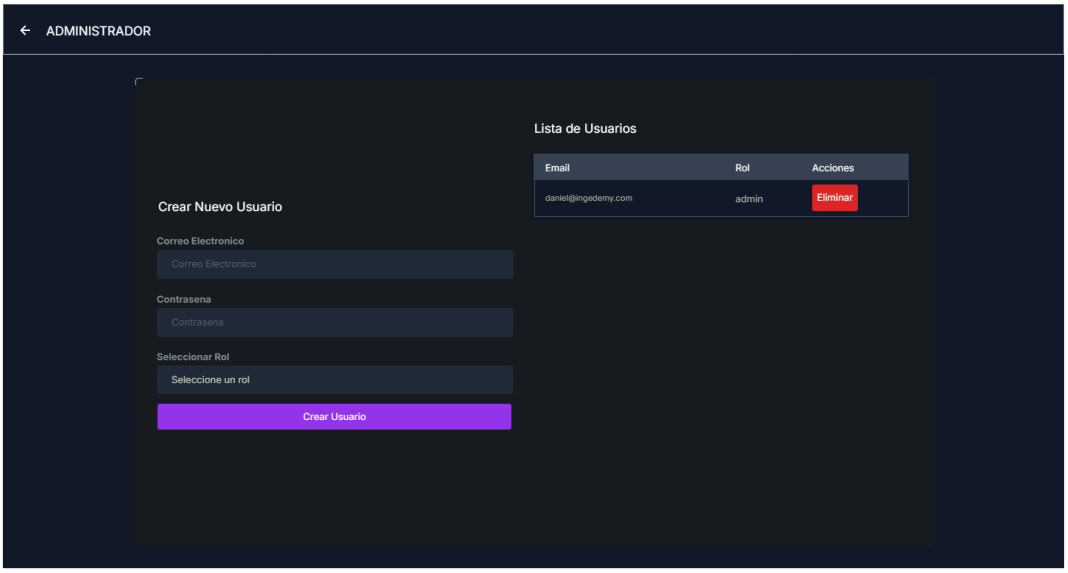


Figura 28. Prototipo de interfaz del Administrador

En la pantalla de configuración de APIs, como se muestra en la Figura 29, el usuario puede ingresar los datos necesarios para las integraciones con distintos servicios. El formulario contiene tres campos principales: el primero solicita el texto de entrada para la API de ChatGPT, el segundo campo está dedicado a la entrada de texto para la API Azure TTS (texto a voz), y el tercer campo permite al usuario seleccionar un modelo específico de ChatGPT mediante un menú desplegable. Al completar los campos, el botón ”Guardar” en la parte inferior permite confirmar la configuración realizada, mientras que un enlace en la parte superior de la pantalla ofrece la opción de regresar a la pantalla de inicio.

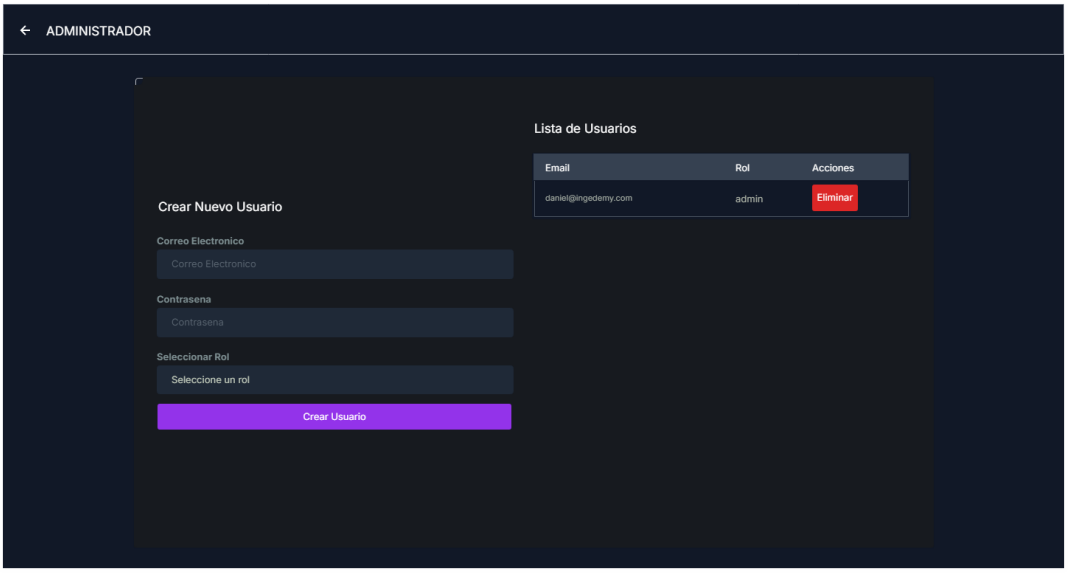


Figura 29. Prototipo de interfaz del Panel de Configuración de APIs y Servicios

En la Figura 30, se muestra la interfaz de inicio de sesión, la cual está dividida en dos secciones principales. A la izquierda, se presenta un espacio reservado para una imagen o logotipo, representado por un icono de imagen genérico. A la derecha, se encuentra el formulario de inicio de sesión que contiene dos campos: uno para ingresar la dirección de correo electrónico del usuario y otro para la contraseña. Debajo de los campos, hay un botón destacado en azul que permite al usuario proceder con el inicio de sesión tras ingresar las credenciales correctas. La disposición es minimalista y profesional, ofreciendo una experiencia clara y directa para el usuario.

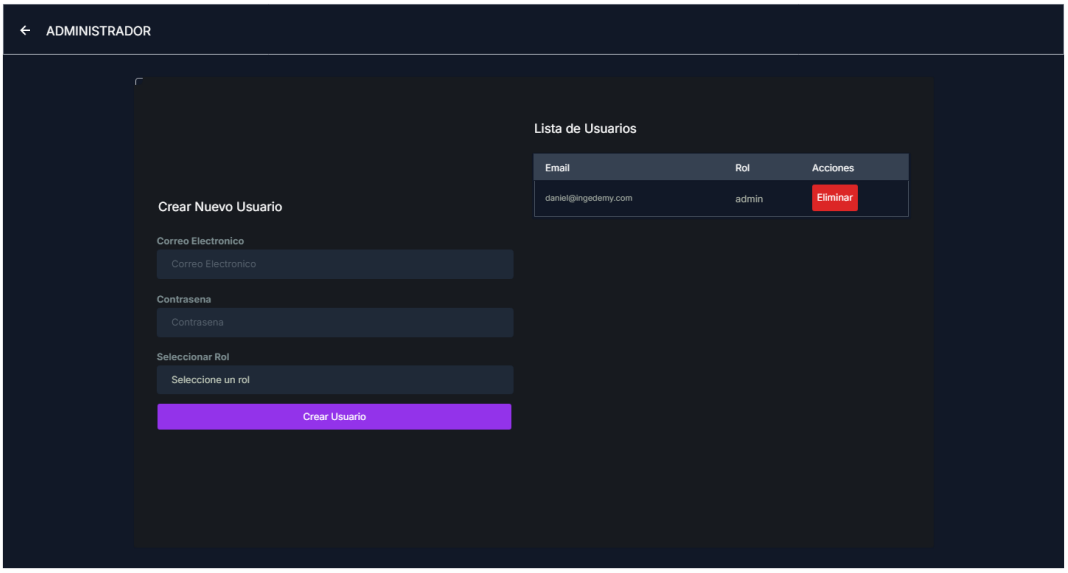


Figura 30. Prototipo de interfaz del Login

3.3.3 Construcción

a. Uso de Nextron

Nextron es un **boilerplate**, es decir, un proyecto base preconfigurado que facilita el desarrollo de aplicaciones de escritorio multiplataforma combinando *Electron* y *Next.js*. Los boilerplates son útiles para ahorrar tiempo y evitar la configuración inicial repetitiva, ofreciendo una estructura lista para comenzar a construir aplicaciones.
Su uso se justifica por las siguientes razones:

- **Desarrollo unificado**: Nextron facilita la creación de aplicaciones de escritorio y web con una sola base de código, aprovechando las capacidades de React y Next.js, lo que reduce el tiempo de desarrollo y mantenimiento.
- **Multiplataforma**: Gracias a Electron, las aplicaciones se pueden distribuir en Windows, macOS y Linux sin necesidad de desarrollar versiones específicas para cada sistema operativo.
- **Optimización**: El uso de Next.js permite optimizar el rendimiento a través del prerenderizado y la generación estática, mejorando la eficiencia de las aplicaciones.

- ***Instalación.***

1. **Instalación del CLI**:

```
npx create-nextron-app my-app
cd my-app
```

2. **Ejecución en desarrollo**:

```
npm run dev
```

3. **Empaquetado**:

```
npm run build
```

Nextron es una opción adecuada para proyectos que requieren aplicaciones de escritorio, dado que optimiza los tiempos de desarrollo y facilita el despliegue en múltiples plataformas con tecnologías modernas y escalables.

b. Estructura del Backend para Automatización de Contenidos Educativos

El modelo de base de datos está diseñado para optimizar y automatizar el flujo de trabajo en la creación de contenido educativo en formato video mediante IA. Cada tabla y relación están orientadas a mejorar la organización, accesibilidad y personalización de los recursos, minimizando la intervención manual y maximizando la eficiencia en la producción de videos educativos.

- **Tabla `projects`**: Gestiona los proyectos de contenido en curso, incluyendo información como nombre, ruta de almacenamiento y usuario asociado. La relación con `auth_users` facilita la identificación de los creadores y proporciona un historial claro de contribuciones, mejorando la trazabilidad en proyectos colaborativos.

- **Tablas `tags` y `projects_tags`**: La tabla `tags` clasifica los proyectos y agiliza la navegación en el sistema. La relación many-to-many entre `projects` y `tags` a través de `projects_tags` permite que los proyectos estén categorizados bajo múltiples etiquetas, facilitando la búsqueda y el filtrado eficientes para los instructores.

- **Tabla `videos`**: Almacena información detallada de cada video, como duración, idioma y estado de edición (campo `isReorganized`). Esto permite gestionar el contenido del video de manera estructurada y flexible. El campo `lang` habilita el soporte multilingüe, adaptando el sistema a audiencias internacionales.

- **Tabla `replacements`**: Permite personalizar y corregir la pronunciación de términos específicos en los guiones generados. Esta función es esencial para la precisión en los videos, especialmente para términos técnicos que requieren ajustes. La automatización mejora la coherencia en los guiones y ahorra tiempo a los instructores.

- **Tablas `roles` y `has_roles`**: Asignan permisos específicos a los usuarios dentro de la plataforma, garantizando control sobre el acceso y las acciones dentro del

sistema. Esto protege la integridad de los proyectos y facilita la administración de permisos sin necesidad de intervención manual frecuente.

- **Tabla `users`**: La tabla `users` actúa como un complemento paralelo al servicio de autenticación de Supabase, permitiendo referenciar y gestionar usuarios directamente desde el esquema de base de datos. Esto facilita la administración de roles y permisos sin modificar la estructura de autenticación predeterminada, optimizando la integración de datos adicionales y mejorando el control sobre los usuarios en el sistema.

En conclusión, el modelo de base de datos propuesto en la Figura 31 apoya la automatización en la creación de contenido educativo, permitiendo a los instructores organizar y producir videos de manera eficiente y con una alta personalización. La estructura relacional y el diseño enfocado en el control automatizado de términos, permisos y categorías mejoran la experiencia del usuario y maximizan la eficiencia en la producción de videos educativos de alta calidad.

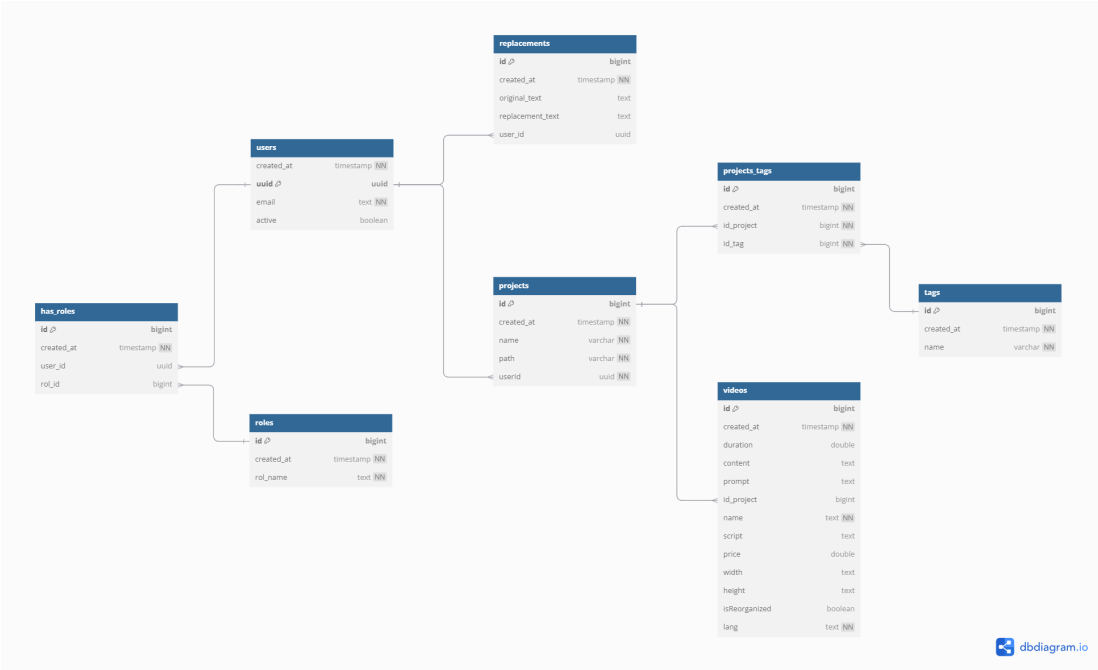


Figura 31. Diagrama de Base de Datos

c. Integración de Supabase en el Frontend

En el sistema, se configuró la biblioteca de cliente de Supabase para permitir la gestión de usuarios y el acceso seguro a datos desde el frontend. A continuación, se

describe el proceso de instalación y configuración realizado para integrar Supabase en la aplicación.

- **Instalación y Configuración del Cliente de Supabase**

  Primero, se instaló la biblioteca de cliente de Supabase en el proyecto utilizando el siguiente comando:

```
npm install @supabase/supabase-js
```

  Posteriormente, se importó la biblioteca y se configuró el cliente de Supabase con las credenciales de acceso proporcionadas por el proyecto, estableciendo la URL y la clave anónima:

```
import { createClient } from '@supabase/supabase-js'

const supabaseUrl = 'https://your-project-url.supabase.co'
const supabaseAnonKey = 'your-anon-key'

const supabase = createClient(supabaseUrl, supabaseAnonKey)
```

Con esta configuración, `supabase` quedó preparado para gestionar la autenticación de usuarios y permitir la interacción con la base de datos desde el frontend del sistema.

d. Gestión de Usuarios

En el sistema, la autenticación y la creación de usuarios se gestionan mediante el cliente de Supabase, que proporciona funciones para la administración de usuarios de forma segura y eficiente. A continuación, se explica el proceso de autenticación y de creación de usuarios, seguido de los fragmentos de código correspondientes.

- **Autenticación de Usuarios**

  Para autenticar usuarios, se emplea una función de inicio de sesión que utiliza el método `signInWithPassword` de Supabase. Este método verifica las

credenciales del usuario, y una vez autenticado, consulta la tabla `users` para asegurar que el usuario esté activo. Si el usuario no está activo, el sistema muestra un mensaje de error y no permite el acceso. El código de esta función puede encontrarse en el Anexo C1.

1. **Verificación de Credenciales**: Si las credenciales son incorrectas, se muestra un mensaje de error al usuario.
2. **Verificación de Estado Activo**: Después de autenticar, se consulta la tabla `users` para verificar si el usuario está activo.
3. **Redirección**: Si el usuario está activo, es redirigido a la página de inicio; de lo contrario, se le notifica que su cuenta está desactivada.

- **Creación de Usuarios**

  La creación de usuarios en el sistema se realiza a través del método `signUp` de Supabase, que registra al usuario y envía un correo de confirmación. Luego, el usuario es añadido a la tabla `users` y se le asigna un rol mediante la tabla intermedia `has_roles`. El fragmento de código relacionado con este proceso puede encontrarse en el Anexo C2.

  1. **Creación de Usuario**: Se crea un usuario en Supabase Auth, y automáticamente se envía un correo de confirmación.
  2. **Inserción en la Tabla `users`**: Una vez creado, el usuario es añadido a la tabla `users` con su estado inicial como activo.
  3. **Asignación de Rol**: Mediante la tabla `has_roles`, se asigna un rol específico al usuario.
  4. **Actualización de la Lista**: Finalmente, se actualiza la lista de usuarios y se resetea el formulario, mostrando un mensaje de éxito.

Esta implementación proporciona un sistema robusto de gestión de usuarios, permitiendo la autenticación y creación de usuarios con roles específicos, así como la verificación del estado activo de cada usuario.

e. Conexión con OpenAI

Para conectar el sistema con la API de OpenAI, se utiliza la biblioteca `OpenAI`, que facilita la comunicación directa con los modelos de inteligencia artificial provistos por

OpenAI. Esta biblioteca permite realizar peticiones de generación de contenido, entre otras funcionalidades.

La configuración inicial de la biblioteca se realiza importándola en el proyecto y estableciendo la clave de API, necesaria para autenticar y autorizar el acceso a los servicios de OpenAI. El siguiente código muestra la inicialización de la biblioteca:

```
import OpenAI from 'openai';

const openai = new OpenAI({
 apiKey: api_key,
});
```

En este fragmento, la variable `apiKey` recibe la clave de API proporcionada por OpenAI. Con esta configuración, el sistema queda listo para hacer solicitudes a los modelos de lenguaje de OpenAI a través de los métodos que expone la biblioteca.

f. Generación de Diapositivas

Para la generación de diapositivas en el sistema, se utilizó la biblioteca `Polotno`. Esta herramienta permite crear y manipular contenido gráfico de forma dinámica y personalizable, facilitando el diseño de diapositivas y otros elementos visuales en formato JSON. Polotno es especialmente útil para este propósito, ya que trabaja con una estructura de datos JSON que permite configurar diversos elementos visuales, como texto, líneas, e imágenes, manteniendo una alta flexibilidad y control sobre el contenido generado.

- **Estructura de Diapositivas**
  El formato JSON utilizado en Polotno organiza la diapositiva en un objeto con parámetros como el ancho (`width`), alto (`height`) y un conjunto de `pages`, cada una de las cuales contiene una lista de `children`. Los `children` son los elementos visuales individuales que se añaden a la diapositiva, tales como texto, líneas, e imágenes. Cada elemento posee una serie de atributos que determinan su posición, tamaño y propiedades estilísticas. A continuación, se describen los elementos principales:
  - **Text Elements** (tipo: `text`): Elementos de texto, que incluyen atributos como:

- `id`: Identificador único.
- `name`: Nombre del elemento.
- `x, y`: Coordenadas de posición.
- `width, height`: Dimensiones.
- `text`: Contenido textual.
- `fontSize`: Tamaño de fuente.
- `fontFamily`: Familia de fuente (por ejemplo, ”Arial”).
- `fontWeight`: Peso de fuente (por ejemplo, ”bold” o ”normal”).
- `fill`: Color del texto (por ejemplo, ”#000000”).
- `align`: Alineación del texto (por ejemplo, ”left”, ”center”, ”right”, ”justify”).

- **Line Elements** (tipo: `line`): Elementos de línea, utilizados para agregar líneas divisoras u otros detalles visuales, con los siguientes atributos:
  - `id`: Identificador único.
  - `name`: Nombre del elemento.
  - `x, y`: Coordenadas de posición.
  - `width, height`: Dimensiones.
  - `color`: Color de la línea (por ejemplo, ”#000000”).
  - `dash`: Patrón de línea discontinua (por ejemplo, `[4, 2]`).
- **Image Elements** (tipo: `image`): Elementos de imagen que permiten agregar contenido visual desde diversas fuentes, con los siguientes atributos:
  - `id`: Identificador único.
  - `name`: Nombre del elemento.
  - `x, y`: Coordenadas de posición.
  - `width, height`: Dimensiones.
  - `src`: Ruta o criterio de búsqueda de la imagen.
  - `cornerRadius`: Radio de las esquinas de la imagen.

- **Ejemplo de JSON para Diapositivas**

  A continuación, se muestra un ejemplo de la estructura JSON generada para

una diapositiva que contiene un elemento de texto, una línea y una imagen. Este JSON organiza cada elemento visual en el objeto `pages`, donde cada `child` representa un componente gráfico individual. El ejemplo completo puede encontrarse en el Anexo C3.

Este JSON es esencial para la organización y personalización del contenido visual dentro del sistema, permitiendo una representación dinámica de cada elemento que se despliega en las diapositivas generadas.

g. Generación de la estructura en JSON

La generación de JSON para las diapositivas se basa en el uso de prompts estructurados que permiten la configuración y creación de cada elemento en las diapositivas. A continuación, se describen los métodos principales para este proceso: `generateWithIdea` y `generateWithParagraph`.

- **generateWithIdea**

  El método `generateWithIdea` se utiliza para desarrollar una presentación a partir de una idea o tema general, proporcionando una estructura básica para cada diapositiva que contiene elementos visuales y de texto. El método puede encontrarse en el Anexo C4.

  Objetivo: Generar una presentación de diapositivas que ilustre y desarrolle el tema introducido, organizando el contenido visual y textual de manera coherente con el propósito del tema.

- **generateWithParagraph**

  El método `generateWithParagraph` se emplea para crear una presentación explicativa basada en un párrafo de texto dado. Este método organiza el contenido del párrafo en diapositivas, facilitando la exposición de cada punto relevante. El método puede encontrarse en el Anexo C5.

  Objetivo: Generar una presentación que estructure y desarrolle el contenido de un párrafo específico, permitiendo que la información sea presentada de manera visual y clara.

- **Parámetros Comunes**

  Ambos métodos comparten un conjunto de parámetros que permiten personalizar la salida de las presentaciones. A continuación, se detalla cada uno de ellos:

Tabla 12. Parámetros para los métodos de generación de presentaciones

Parámetro	**Descripción**	**Ejemplo**
`script/text`	Tema o texto para la presentación	*Inteligencia Artificial*
`numSlides`	Número de diapositivas	*5*
`amount`	Cantidad de texto por diapositiva	*bit, half, quite*
`language`	Idioma del contenido	*español*
`guidelines`	Directrices adicionales	*Usar tono formal*
`width`	Ancho del canvas	*1024*
`height`	Alto del canvas	*768*

h. Proceso de Generación Recursiva del JSON

La generación del JSON es un proceso recursivo que se basa en el uso del modelo GPT-4 para construir el contenido de cada diapositiva en fragmentos manejables. Esto permite que el JSON final sea válido y se adapte a los límites de tokens.

En cada iteración, el sistema verifica si el JSON está completo. Si no es así, se agrega el mensaje generado por el asistente al JSON acumulado y se solicita la continuación exacta en caso de que el asistente haya alcanzado el límite de longitud. Esta operación se repite hasta que el JSON esté completo, garantizando que el resultado final sea un JSON concatenado válido. El fragmento de código se encuentra en el Anexo C6.

i. Edición de Diapositivas

Para implementar la funcionalidad de edición de diapositivas, se utiliza la biblioteca Polotno, una herramienta de código abierto para la creación y manipulación de presentaciones y contenido visual en aplicaciones web. A continuación, se describe el proceso de instalación y configuración básica de Polotno para su uso en el sistema.

- **Instalación de Polotno**

Para instalar Polotno, se ejecuta el siguiente comando en la terminal del proyecto:

```
npm install polotno
```

- **Configuración de Imports**

Una vez instalada la biblioteca, se deben importar los componentes necesarios para estructurar la interfaz de usuario. Polotno ofrece contenedores específicos para el panel lateral y el espacio de trabajo donde se manipulan los elementos visuales. Los imports necesarios son los siguientes:

```
import { PolotnoContainer, SidePanelWrap, WorkspaceWrap }
from "polotno";
```

- **Función `createStore`**

La función `createStore` es esencial para inicializar y gestionar el estado de la aplicación de edición. Esta función crea un "almacén" o *store* que contiene el estado y configuración de la sesión de edición. La función requiere una clave API de Polotno, representada aquí por `"YOUR_POLTNO_API_KEY"`, la cual permite el uso de sus funcionalidades y opciones de personalización.

- **Estructura de la Interfaz de Usuario**

La configuración de la interfaz de usuario incluye la creación de un contenedor para Polotno (`PolotnoContainer`), un panel lateral (`SidePanelWrap`) y un área de trabajo principal (`WorkspaceWrap`).

En el código del Anexo C7, el contenedor `PolotnoContainer` abarca toda la interfaz de edición. El componente `SidePanelWrap` representa el panel lateral donde se encuentran las herramientas de edición, mientras que el componente `WorkspaceWrap` define el área de trabajo donde se visualizan y editan las diapositivas. Para la funcionalidad completa, es necesario reemplazar `"YOUR_POLTNO_API_KEY"` con una clave válida de API proporcionada por Polotno.

- **Uso del Canvas en Polotno**

  El uso del canvas en Polotno es fundamental para la funcionalidad de edición, ya que este actúa como el espacio de trabajo interactivo donde se representan y manipulan visualmente los elementos de la diapositiva. El canvas en Polotno permite añadir, posicionar, y personalizar componentes gráficos como texto, imágenes, y formas, proporcionando una interfaz en tiempo real para la edición de contenido. Polotno utiliza este canvas para renderizar cada elemento y aplicar ajustes de tamaño, color, fuente y disposición, asegurando que los cambios realizados sean inmediatamente visibles. Este enfoque facilita la creación de contenido visualmente coherente y profesional en un entorno controlado, aprovechando la flexibilidad y precisión que el canvas ofrece para gestionar cada detalle de diseño de las diapositivas.

- **Módulo de Imágenes Añadido a Polotno**

  Para mejorar la creación de contenido visual en el editor de diapositivas, se ha añadido un módulo personalizado que permite la búsqueda de imágenes en Google Imágenes mediante la API de Búsqueda Personalizada de Google. Este módulo facilita la selección e inserción de imágenes relevantes dentro de las diapositivas, enriqueciendo el contenido visual de las presentaciones.

  El código mostrado en el Anexo C8, realiza una llamada a la API de Google utilizando una clave de API (`apiKey`) y un identificador de motor de búsqueda personalizado (`cx`). Cuando el usuario ingresa una consulta, el sistema envía una solicitud para obtener imágenes de gran tamaño relacionadas con el tema especificado. Las URLs de las imágenes se almacenan en el array `imagesUrls`, el cual se integra con el estado de la aplicación. En caso de que no se encuentren imágenes, se imprime un mensaje de advertencia en la consola para facilitar la depuración. Tras un breve retardo, simulado con `setTimeout`, las URLs de las imágenes se asignan al estado del componente mediante `setImages`, mostrando las imágenes disponibles para su selección en el panel de edición.

  Para presentar las imágenes buscadas dentro de la interfaz de Polotno, se ha añadido una nueva sección en el panel lateral llamada `GPhotos`. Esta sección permite a los usuarios acceder y visualizar las imágenes obtenidas mediante una pestaña adicional en el panel lateral del editor.

- **Generación de nuevas diapositivas**

  Para implementar la generación de nuevas diapositivas utilizando inteligencia artificial, se ha diseñado una función denominada `generateNewSlide`. Esta función recibe varios parámetros clave y devuelve un objeto JSON que sigue una estructura específica, la cual facilita la creación de diapositivas personalizadas en el sistema. El JSON generado incluye elementos como títulos, líneas y secciones de texto, distribuidos y estilizados de acuerdo con las directrices del tema de la presentación.

  En el Anexo C9, se puede observar el código de la función `generateNewSlide`, que organiza y distribuye los elementos en el diseño de la diapositiva. Este código se ajusta a las directivas de posicionamiento y estilo, además de incorporar imágenes basadas en los criterios de búsqueda definidos al inicio del prompt. La siguiente tabla describe los parámetros utilizados en esta función y sus respectivas funciones.

  Tabla 13. Parámetros para la generación de nuevas diapositivas con IA

Parámetro	**Descripción**	**Ejemplo**
`script`	Tema de la presentación	*Inteligencia Artificial*
`amount`	Cantidad de texto por diapositiva	*bit, half, quite*
`language`	Idioma del contenido	*español*
`guidelines`	Directrices adicionales	*Usar tono formal*
`width`	Ancho del canvas	*1024*
`height`	Alto del canvas	*768*

  Esta función `generateNewSlide` estructura las diapositivas para que incluyan un título, un elemento separador en forma de línea, párrafos de texto justificado y, cuando sea necesario, imágenes relevantes. Además, se ajusta a diversos diseños predefinidos para optimizar la visualización de los elementos en el espacio del canvas.

j. Generación de Guiones

La generación de guiones se realiza a través de una serie de métodos implementados en JavaScript que utilizan el modelo ChatGPT para generar contenido en formato JSON. Este formato JSON devuelve un guion para cada diapositiva de una presentación educativa, asegurando que cada guion siga un tono adecuado y fluido para su uso en software de texto a voz. A continuación, se explica cada método en detalle:

- **Método `generate`**:

  En el Anexo C10 se observa el método principal que controla el flujo de generación de guiones. Si el guion inicial no existe, llama al método `generateInitialScripts` para generarlo. Si el guion ya existe, mapea las diapositivas y verifica si hay alguna que no tiene guion, en cuyo caso llama al método `generateMissingScripts` para generarlas.

- **Método `generateInitialScripts`**:

  En el Anexo C11 se muestra el método responsable de generar los guiones iniciales para cada diapositiva. Convierte las imágenes de cada diapositiva a formato base64 para enviarlas al modelo y solicita el guion para cada una. Luego, calcula el costo en tokens y actualiza el precio de generación.

- **Método `generateMissingScripts`**:

  En el Anexo C12 se muestra la función que se activara cuando existan posiciones vacías en el guion (diapositivas sin guion). Realiza el mismo proceso de conversión de imágenes y solicita los guiones faltantes al modelo. Una vez obtenidos, actualiza el precio y los textos correspondientes en el guion general.

- **Método `convertImageToDataURLSync`**:

  En el Anexo C13 se muestra el fragmento de codigó que convierte las imágenes de las diapositivas en datos base64 para que puedan ser enviadas al modelo de ChatGPT, facilitando el proceso de generación de guiones a partir de imágenes de las diapositivas.

- **Método `generateScriptForSlides`**:

  En el Anexo C14 se aprecia el encargado de realizar la solicitud al modelo ChatGPT, incluyendo las instrucciones específicas para que genere guiones

detallados para cada diapositiva en JSON. El modelo analiza cada imagen y devuelve un guion.

- **Método `generateMessageToChangeScript`**:

  En el Anexo C15 se muestra el método `generateMessageToChangeScript` el cual se encarga de crear un mensaje estructurado que se envía al modelo de ChatGPT. Este mensaje incluye el guion actual, una imagen en formato base64 de la diapositiva correspondiente, y el `userMessage`, que contiene la retroalimentación del usuario. El objetivo es que el modelo reescriba el guion con base en estas instrucciones, mejorando la claridad, los conectores y la información adicional que facilite una narración natural y educativa para el software de Texto a Voz (TTS). Este método garantiza que el nuevo guion esté alineado con las expectativas del usuario sin incluir texto adicional fuera del contenido actualizado.

k. Comunicación entre el Frontend y Backend mediante IPC

En aplicaciones de arquitectura cliente-servidor, es frecuente que el frontend necesite comunicarse con el backend para realizar operaciones como obtener datos, actualizar información o ejecutar tareas específicas en el sistema. Una forma eficiente y segura de realizar esta comunicación es a través de Inter-Process Communication (IPC), un mecanismo que permite el intercambio de datos entre procesos independientes. En el contexto de aplicaciones desarrolladas en entornos como Electron, IPC se usa comúnmente para establecer una comunicación directa entre el proceso principal (backend) y el proceso de renderizado (frontend).

El flujo de comunicación por IPC puede resumirse en dos paso:

- **Envió de solicitud**: el frontend envía una solicitud al backend a través de un canal de comunicación, especificando el tipo de operación requerida y los datos necesarios.
- **Respuesta del backend**: el backend recibe y procesa la solicitud, ejecutando las operaciones necesarias y retornando la respuesta al frontend a través del mismo canal.

En el Anexo C16, el frontend envía una solicitud de datos al backend a través del canal de comunicación `solicitud-datos`. Esta solicitud contiene parámetros específicos que el backend utilizará para procesar la operación requerida.

En el Anexo C17, el backend recibe y procesa la solicitud, realiza las operaciones necesarias, y luego envía los datos solicitados de vuelta al frontend a través del canal de respuesta `respuesta-datos`. Este flujo de comunicación es asíncrono y permite una interacción eficiente entre el frontend y el backend.

Tabla 14. Descripción de IPCs utilizados en el sistema

Nombre del IPC	**Descripción**
save-images	Guarda imágenes en formato base64 de cada diapositiva en la carpeta asignada para el proyecto (`idP`) y video (`idV`).
generate-video	Convierte scripts de texto a archivos de audio en formato MP3 para cada diapositiva, utilizando el servicio TTS. Luego, usa la función `createVideo`, para combinar los segmentos de audio y imagen de cada diapositiva en un único video MP4.
listen-preview	Crea un archivo de audio de vista previa para un script especificado. Borra archivos previos de la carpeta de vista previa y guarda el nuevo archivo de audio.

l. Generación de Audio con Azure TTS

La generación de archivos de audio se realiza mediante el servicio Text-to-Speech (TTS) de Azure, el cual convierte texto en voz de alta calidad a través de solicitudes HTTP POST. Primero, se configuran las credenciales mediante la clave de suscripción y la región de Azure, que se utilizan para formar el punto de enlace (endpoint) del servicio.

En el Anexo C18 se muestra la función `generateAudioFile` que lleva a cabo la solicitud de TTS, recibiendo como parámetros el tipo de voz, el texto a transformar y la ruta para guardar el archivo de audio generado. Utiliza SSML (Speech Synthesis Markup Language) para definir la voz y el texto en un formato XML estandarizado, enviando luego una solicitud POST con estos datos y las credenciales de Azure.

En la cabecera de la solicitud, se especifica el formato de salida de audio como `audio-24khz-48kbitrate-mono-mp3`, asegurando una calidad adecuada para la aplicación deseada.

La respuesta del servicio, que contiene el audio en formato de `arraybuffer`, se guarda en el sistema de archivos utilizando la ruta proporcionada. Esta implementación facilita la generación automática de archivos de audio para aplicaciones que requieren contenido auditivo, como videos educativos o asistentes de voz.

m. Edición de Video

En esta sección, se describe el proceso de creación de videos en formato MP4 mediante la combinación de archivos de imagen y audio, utilizando la biblioteca Fluent-FFMPEG. Fluent-FFMPEG es una herramienta de edición y procesamiento de video basada en Node.js, que proporciona una interfaz fácil de usar para trabajar con FFmpeg, una poderosa utilidad de línea de comandos para la manipulación de archivos multimedia.
El proceso de edición de video se implementa en varias etapas clave, descritas a continuación:

- **Creación de videos segmentados**: Para cada imagen y su correspondiente archivo de audio, se genera un segmento de video individual. La función recibe el identificador del proyecto (`idP`) y el identificador del video (`idV`), junto con el número de segmentos (`scriptsLength`), y las dimensiones de resolución (`width` y `height`) en píxeles.

- **Configuración de codificación**: Se usa FFmpeg para codificar cada segmento de video. Las imágenes se procesan en formato `libx264` para video y `aac` para audio, con una tasa de bits de audio configurada en 192 kbps. Cada imagen se repite para ajustarse a la duración del archivo de audio correspondiente mediante la opción de `loop`.

- **Mapeo de flujos de audio y video**: Para asegurar la sincronización correcta entre audio y video, se emplean las opciones `-map 0:v:0` y `-map 1:a:0`, que asignan el primer flujo de video y audio, respectivamente, para cada segmento de salida. Además, se utiliza la opción `-shortest` para truncar el video a la duración del audio.

- **Concatenación de videos**: Una vez generados todos los segmentos, se crea un archivo de lista temporal que contiene las rutas relativas de cada segmento. Luego, FFmpeg concatena estos segmentos usando el formato de entrada `concat` y el parámetro `-safe 0`, que permite rutas absolutas para garantizar la compatibilidad en sistemas como Windows. Finalmente, la salida se genera en un archivo único, que integra todos los segmentos en el video final en formato MP4.

- **Limpieza de archivos temporales**: Al finalizar la creación del video, se eliminan el archivo de lista y los archivos de segmentos temporales para liberar espacio y mantener el sistema ordenado.

Este flujo de trabajo que se ilustra en el Anexo C19, asegura la creación de un video final de alta calidad, con audio y video sincronizados, utilizando recursos y técnicas de procesamiento optimizados.

3.3.4 Pruebas de aceptación

Las pruebas de aceptación se llevaron a cabo con el propósito de validar que el sistema cumpla con los requisitos funcionales y no funcionales definidos a lo largo de las tres iteraciones de desarrollo. Estas pruebas abarcaron todas las funcionalidades esenciales del software, asegurando que cada una de las características implementadas responda a las necesidades y expectativas de los usuarios. Se evaluaron aspectos como la correcta gestión de proyectos, la generación automatizada de contenido y la optimización de recursos mediante inteligencia artificial, garantizando así un rendimiento robusto y una experiencia de usuario satisfactoria antes de su despliegue final.

Tabla 15. Pruebas de Aceptación de Requerimientos de la Primera Iteración

N	Criterio	Estado	Usuario
1	Al hacer clic en ”+ Crear proyecto”, se abre un formulario para ingresar nombre y etiquetas.	Aceptado	Instructor
2	Al guardar un nuevo proyecto, debe aparecer en la lista con el nombre y etiquetas ingresados.	Aceptado	Instructor

N	Criterio	Estado	Usuario
3	Al hacer clic en ”X” en una tarjeta, el proyecto y sus contenidos deben eliminarse.	Aceptado	Instructor
4	Al seleccionar la opción de editar, se muestra un formulario con nombre y etiquetas editables.	Aceptado	Instructor
5	Los cambios en la edición de un proyecto deben reflejarse correctamente en la lista.	Aceptado	Instructor
6	Al buscar un término en la barra, solo se muestran proyectos coincidentes.	Aceptado	Instructor
7	Al filtrar por etiquetas, deben mostrarse solo los proyectos con dichas etiquetas.	Aceptado	Instructor
8	La pantalla muestra proyectos en tarjetas con logotipo, nombre y etiquetas visibles.	Aceptado	Instructor
9	Al hacer clic en un proyecto, se muestra una tabla con los videos asociados.	Aceptado	Instructor

Tabla 16. Pruebas de Aceptación de Requerimientos de la Iteración 2

N	Criterio	Estado	Usuario
1	Al seleccionar ”Crear video desde cero”, el sistema muestra un formulario para ingresar nombre, idioma y formato del video.	Aceptado	Instructor
2	Al completar el formulario de ”Crear video desde cero” y confirmar, el video se agrega a la tabla de videos del proyecto.	Aceptado	Instructor
3	Al seleccionar ”Crear video con IA”, el sistema muestra opciones avanzadas como idioma, resolución, cantidad de diapositivas y tipo de entrada.	Aceptado	Instructor
4	Al generar un video con IA, el video aparece en la tabla de videos del proyecto con los parámetros seleccionados.	Aceptado	Instructor
5	Al editar una diapositiva, el sistema permite modificar texto e imágenes, reflejando los cambios en tiempo real.	Aceptado	Instructor
6	El sistema permite buscar y agregar imágenes externas dentro de la edición de diapositivas.	Aceptado	Instructor
7	Al seleccionar ”Agregar IA a diapositivas”, se puede generar una nueva diapositiva con contenido basado en las indicaciones proporcionadas.	Aceptado	Instructor
8	El instructor puede crear y modificar guiones asociados a cada diapositiva, utilizando opciones manuales o IA.	Aceptado	Instructor

Tabla 17. Pruebas de Aceptación de Requerimientos de la Iteración 3

N	Criterio	Estado	Usuario
1	El usuario puede guardar un proyecto incompleto y retomarlo más tarde.	Aceptado	Instructor
2	Al reabrir un proyecto guardado, el contenido previamente ingresado se muestra correctamente.	Aceptado	Instructor
3	El usuario puede seleccionar una voz IA para narrar el contenido de los guiones.	Aceptado	Instructor
4	El usuario puede escuchar una muestra de la voz seleccionada antes de aplicarla al guion.	Aceptado	Instructor
5	El usuario puede previsualizar el guion con la voz seleccionada para ajustar la entonación.	Aceptado	Instructor
6	Al solicitar reescritura de guion con IA, el sistema genera una versión alternativa coherente.	Aceptado	Instructor
7	El usuario puede reemplazar palabras en los guiones para ajustar pronunciaciones, y estos cambios se reflejan en la previsualización.	Aceptado	Instructor
8	El sistema genera un video en formato MP4 combinando las diapositivas y el guion final.	Aceptado	Instructor
9	El video generado es accesible para descarga en formato MP4.	Aceptado	Instructor
10	Al descargar el video, se guarda correctamente en el dispositivo del usuario en el formato esperado.	Aceptado	Instructor

3.4 Comparativa de Tiempos: Manual vs. Automatizado

El objetivo es evaluar los tiempos necesarios para completar cada una de las etapas del proceso de producción de un video educativo de 10 minutos, considerando los métodos

manual y automatizado. Además, se analiza la conformidad de los instructores con los resultados en cada etapa.

3.4.1 Etapas Consideradas

Las etapas del proceso son:

1. **Creación de diapositivas**: Diseño y estructuración de contenido visual.
2. **Redacción de guión**: Escritura de texto para narración o instrucción.
3. **Grabación**: Obtención del audio para el video.
4. **Edición**: Integración de audio, video y gráficos.

3.4.2 Metodología

a. Selección de Instructores

Para este análisis, se trabajó con 5 instructores. Este número fue seleccionado considerando que es suficiente para garantizar diversidad en los estilos de trabajo y experiencia práctica, permitiendo obtener un promedio representativo de tiempos sin comprometer la viabilidad del estudio. Además, la cantidad de instructores es adecuada para realizar mediciones repetibles y garantizar consistencia en los resultados.

b. Medición de Tiempos

- Cada instructor produjo un video de 10 minutos utilizando ambos métodos (manual y automatizado).
- Los tiempos de cada etapa se registraron utilizando cronómetros digitales.
- Para garantizar uniformidad, todos trabajaron en el mismo tema con requisitos estándar (contenido, gráficos y narración).

c. Evaluación de Conformidad

Al finalizar, cada instructor evaluó los resultados de cada etapa en una escala del 1 al 5:

- **1**: Muy insatisfecho.
- **2**: Insatisfecho.
- **3**: Neutral.
- **4**: Satisfecho.
- **5**: Muy satisfecho.

Los resultados de estas evaluaciones permitieron identificar patrones en la percepción de los instructores respecto a la calidad y eficiencia de los métodos manual y automatizado, así como su impacto en cada etapa del proceso.

3.4.3 Tiempos Promedios por Etapa

Tabla 18. Tiempos promedios por etapa

Etapa	**Producción Manual (min)**	**Producción Automatizada (min)**	**Diferencia (%)**	**Conformidad Manual**	**Conformidad Automatizada**
Creación de diapositivas	41.40	25.63	-38.09%	4.6	4.8
Redacción de guión	45.63	19.61	-57.02%	4.4	4.6
Grabación	39.51	0	-100%	4.0	-
Edición	29.60	5.34	-81.96%	4.2	4.8
Total	**156.14 (2.6 horas)**	**50.85 (0.84 horas)**	**-67.43%**	**4.3**	**4.73**

3.4.4 Cálculo de Indicadores

1. **Tiempo total ahorrado:**

$$\text{Tiempo ahorrado} = \text{Tiempo manual promedio} - \text{Tiempo automatizado promedio}$$

$$\text{Tiempo ahorrado} = 156.14 - 50.85 = 105.29 \,\text{minutos (1.75 horas)}$$

2. **Porcentaje de tiempo reducido:**

$$\text{Reducción (\%)} = \frac{\text{Tiempo ahorrado}}{\text{Tiempo manual promedio}} \times 100$$

$$\text{Reducción (\%)} = \frac{105.29}{156.14} \times 100 = 67.43\%$$

La automatización permitió una reducción significativa de los tiempos de producción en todas las etapas analizadas. La etapa de grabación se eliminó completamente debido al uso de herramientas de síntesis de voz, lo que representó un ahorro del 100% en este proceso. Las mayores reducciones de tiempo se observaron en la edición/exportación, con una disminución del 81.96%, seguida de la redacción de guion, que se redujo en un 57.02%, y la creación de diapositivas, que experimentó una reducción del 38.09%. En términos generales, el tiempo total necesario para producir un video educativo pasó de 2.6 horas utilizando el método manual a 0.84 horas con el método automatizado, lo que representa una disminución total del 67.43%.

En cuanto a la conformidad de los instructores, ambos métodos obtuvieron calificaciones altas, superiores a 4 en promedio, lo que refleja una buena aceptación de los resultados. Sin embargo, el método automatizado logró una conformidad ligeramente mayor, con un promedio de 4.63, en comparación con 4.48 para el método manual. Esto puede atribuirse a la percepción de mayor eficiencia y facilidad en el proceso automatizado. Por otro lado, la etapa de grabación, presente únicamente en el método manual, obtuvo la calificación más baja (4.1), lo que sugiere que el esfuerzo requerido en esta etapa influyó negativamente en la percepción de los instructores.

3.5 Análisis de la Productividad Antes y Después de la Implementación

Para evaluar el impacto del software en la productividad de los instructores, se realizó un análisis basado en el registro histórico de producción de videos educativos de la empresa. Este registro incluye información relevante como la fecha de creación, la duración de cada video y el instructor responsable, lo cual permitió realizar un estudio detallado y confiable.

El análisis se centró en dos períodos específicos: una semana antes y una semana después de la implementación del software. Durante la semana previa al uso del software, los instructores produjeron un total de 119 videos, mientras que en la semana posterior se produjeron 188 videos. Estas cifras no fueron seleccionadas arbitrariamente, sino que representan la cantidad total de videos creados en cada uno de los períodos analizados, lo que hace posible una comparación precisa del impacto

del software en la productividad.

3.5.1 Cantidad de videos producidos

El primer análisis se enfocó en comparar la cantidad de videos producidos por los instructores durante una semana antes y una semana después de la implementación del software. Este análisis busca determinar si el software tuvo un impacto significativo en la productividad de los instructores, medido en términos del número total de videos producidos en cada período.

a. Resultados

- **Cantidad total de videos.**
  - **Antes del software:** 119 videos.
  - **Después del software:** 188 videos.
  - **Incremento en la cantidad de videos producidos:** Un aumento del 57.98% en la producción total de videos.
- **Prueba t-Student.**

  Para evaluar si este aumento en la producción es estadísticamente significativo, se realizó una prueba t-Student somo se muestra en el Anexo E comparando la cantidad de videos producidos por cada instructor antes y después del software. Los resultados de la prueba fueron los siguientes:
  - **t-Statistic:** -9.864
  - **p-Value:** 1.51×10^{-8}

b. Interpretación

Con un valor de 1.51×10^{-8} (menor a 0.05), los resultados de la prueba t-Student indican que existe una diferencia estadísticamente significativa en la cantidad de videos producidos por los instructores antes y después de la implementación del software. Este resultado confirma que el incremento observado en la producción total de videos no es producto del azar, sino que puede atribuirse a la implementación del software.

Además, el aumento del 57.98% en la cantidad de videos producidos refleja un impacto positivo y sustancial en la productividad de los instructores. Este incremento sugiere

que el software facilita y agiliza el proceso de creación de videos, permitiendo a los instructores generar más contenido en el mismo período de tiempo.

3.5.2 Total de contenido producido

Se evaluó el impacto del software en el tiempo total de contenido generado, comparando las duraciones de los videos producidos antes y después de su implementación.

a. Resultados

- **Antes del software:**
  86,543 segundos (24 horas, 2 minutos y 23 segundos).
- **Después del software:**
  137,901 segundos (38 horas, 18 minutos y 21 segundos).
- **Incremento:**
  59.34%.

b. Interpretación

El software permitió generar un 59.34% más de contenido en la misma cantidad de tiempo, lo que demuestra una mejora significativa en la productividad de los instructores. Este aumento valida la efectividad del software para optimizar la creación de contenido educativo.

3.5.3 Duración Promedio de los Videos

Para evaluar si el incremento en la cantidad de videos y el tiempo total de contenido afectó la duración promedio de los videos producidos, se calculó este valor antes y después de la implementación del software.

a. Resultados

- **Antes del software:**
  12 minutos y 6 segundos.

- **Después del software:**

  12 minutos y 13 segundos.

b. Interpretación

La duración promedio de los videos se mantuvo prácticamente constante, con una variación mínima de 7 segundos. Los instructores continuaron produciendo videos de duración similar, lo que resalta la eficiencia del software al optimizar los procesos sin alterar las características del contenido.

CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

En el presente trabajo de tesis se desarrolló un sistema para la automatización de la generación de contenido educativo en formato de video utilizando herramientas basadas en inteligencia artificial. Este sistema tuvo como objetivo principal reducir significativamente los tiempos de producción y mejorar la calidad del contenido generado, en comparación con los métodos manuales tradicionales.

- Se identificaron múltiples áreas de ineficiencia que afectan directamente al tiempo de producción y la calidad del contenido audiovisual. Las entrevistas realizadas a los instructores de la empresa Ingedemy revelaron que los principales cuellos de botella incluyen la repetición de grabaciones debido a errores, la cantidad de tiempo requerido para la edición y la creación de diapositivas, y las dificultades en la redacción de guiones. Estos factores no solo incrementan los tiempos de producción, sino que también limitan la flexibilidad creativa y la estandarización del contenido educativo. Por tanto, es crucial abordar estas problemáticas para mejorar tanto la calidad como la eficiencia en la generación de material audiovisual. Sin embargo, gracias al proceso automatizado propuesto en este proyecto, se evidenció una mejora considerable en comparación con el proceso manual anterior, permitiendo optimizar las etapas críticas y facilitando la creación de material audiovisual de mayor calidad.

- Se exploraron diversas herramientas tecnológicas que permiten automatizar procesos clave en la producción de videos educativos. Tecnologías como motores de síntesis de voz (Text-to-Speech), modelos de lenguaje (LLM), editores de video programáticos demostraron ser eficaces para optimizar etapas como la creación de diapositivas, generación de guiones, generación de voz y la edición. En particular, herramientas como Microsoft Azure TTS, el LLM ChatGPT 4o, el editor FFMPEG y frameworks de desarrollo como Electron JS mostraron superioridad sobre otros en sus respectivas y un potencial significativo para integrar procesos avanzados que reduzcan la carga de trabajo manual. La selección adecuada de estas herramientas permitió una mejora sustancial en comparación con el proceso manual anterior, logrando una mayor eficiencia en

la ejecución de tareas y asegurando un estándar de calidad más consistente en el contenido final.

- El sistema desarrollado permitió una reducción del 67.43% en los tiempos de producción de videos educativos, optimizando recursos y automatizando etapas clave del proceso. Las pruebas realizadas evidencian un incremento del 57.98% en la cantidad de videos producidos y un aumento del 59.34% en el tiempo total de contenido generado, validado estadísticamente mediante una prueba t-Student ($p < 0.05$). Además, la duración promedio de los videos se mantuvo constante, lo que garantiza la calidad del contenido educativo producido. Este sistema no solo mejora los tiempos de entrega, sino que también permite a los instructores enfocarse en tareas de mayor valor creativo, satisfaciendo de manera eficiente la creciente demanda de contenido educativo de calidad.

- La implementación del sistema automatizado para la generación de videos educativos en la empresa Ingedemy ha permitido optimizar significativamente los tiempos y procesos de producción. Este resultado se debe al uso de recursos basados en inteligencia artificial, como la generación automática de guiones mediante modelos de lenguaje, la síntesis de voz para narraciones y la edición programática de video. La automatización no solo ha acelerado etapas clave como la creación de diapositivas y grabaciones, sino que también ha mejorado la eficiencia al reducir la carga de trabajo manual y minimizar errores. En consecuencia, el sistema ha contribuido a una producción más ágil y estandarizada, garantizando resultados de mayor calidad y acorde a las necesidades actuales del equipo de trabajo.

4.2 Recomendaciones

- Se recomienda realizar un análisis continuo de los procesos automatizados para identificar posibles áreas de mejora y optimización. Asimismo, es importante monitorear y documentar los resultados obtenidos con la implementación del sistema, comparándolos con el proceso manual anterior. Esto permitirá evidenciar los avances logrados, garantizar la sostenibilidad de las mejoras implementadas y facilitar la adaptación de nuevas tecnologías que puedan seguir optimizando la calidad y eficiencia en la producción del contenido audiovisual.

- Se sugiere realizar actualizaciones regulares en las tecnologías integradas, como motores de síntesis de voz y modelos de lenguaje, para incorporar avances tecnológicos que optimicen aún más los procesos. Adicionalmente, la revisión y ajuste de los algoritmos de automatización contribuirán a mantener la calidad y eficacia del sistema a largo plazo.

- Es recomendable establecer un sistema de monitoreo constante para evaluar el impacto del sistema desarrollado en los tiempos y la calidad de producción. Esta evaluación puede incluir encuestas periódicas a los usuarios y análisis de métricas de desempeño. Los datos recopilados permitirán identificar áreas de mejora y posibles ampliaciones en las funcionalidades del sistema.

- Se recomienda implementar programas de capacitación periódicos para los instructores de la empresa Ingedemy, enfocándose en el uso eficiente del sistema desarrollado y sus herramientas de automatización. Esto garantizará que los usuarios finales comprendan plenamente las funcionalidades del sistema, facilitando su adopción y maximizando los beneficios de la automatización en la producción de contenido educativo.

REFERENCIAS BIBLIOGRÁFICAS

[1] *Informe de seguimiento de la educación en el mundo. Informe sobre género: profundizar en el debate sobre quienes todavía están rezagados.* UNESCO, 1 2022.

[2] G. M. Insights, "Digital education content market size - by content type (e-books textbooks, multimedia content, assessments quizzes, interactive simulation), by delivery mode (web-based content, mobile applications, offline accessible content), end-user forecast 2023 – 2032," 6 2023.

[3] A. F. Cortina y J. P. Labra, "Desafíos de la producción multimedia en los mooc. estudio de caso interpretativo sobre las perspectivas docentes," *RIED. Revista Iberoamericana de Educación a Distancia*, vol. 25, 9 2021.

[4] T. Viik, "Educational videos at primary education during distance learning," 2021.

[5] S. Castillo, K. Calvitti, J. Shoup, M. Rice, H. Lubbock, y K. H. Oliver, "Production processes for creating educational videos," *CBE—Life Sciences Education*, vol. 20, p. es7, 6 2021.

[6] K. Valeria y Ángeles Soletic, "Políticas digitales en educación en américa latina: tendencias emergentes y perspectivas de futuro," 2022.

[7] "Uso de tic en educación en américa latina y el caribe: análisis regional de la integración de las tic en la educación y de la aptitud digital (e-readiness)," 2023.

[8] P. E. Muñoz-Castro, C. F. Guevara-Vizcaíno, y J. C. Erazo-Álvarez, "Competencias digitales pedagógicas de los docentes de administración de la universidad católica de cuenca, ecuador," *Revista Arbitrada Interdisciplinaria Koinonía*, vol. 7, p. 586, 8 2022.

[9] X. Q. Dao, N. B. Le, y T. M. T. Nguyen, "Ai-powered moocs: Video lecture generation," *ACM International Conference Proceeding Series*, pp. 95–102, 3 2021. [En línea]. Disponible en: https://dl.acm.org/doi/10.1145/3459212.3459227

[10] P. Denny, H. Khosravi, A. Hellas, J. Leinonen, y S. Sarsa, “Can we trust ai-generated educational content? comparative analysis of human and ai-generated learning resources,” 6 2023.

[11] C. Xia, C. Xing, J. Du, X. Yang, Y. Feng, R. Xu, W. Yin, y C. Xiong, “Fofo: A benchmark to evaluate llms’ format-following capability,” 2 2024. [En línea]. Disponible en: https://arxiv.org/abs/2402.18667v1

[12] A. Woollacott, “Benchmarking speech technologies,” *Benchmarking Speech Technologies*, 1 2021. [En línea]. Disponible en: https://www.academia.edu/45165394/Benchmarking_Speech_Technologies

[13] T. Nagahama, M. Makino, y Y. Morita, “Effect analysis of high-speed presentations of educational video utilizing synthetic speech,” vol. 13, pp. 66–74, 2019.

[14] L. Shen, Y. Zhang, H. Zhang, y Y. Wang, “Data player: Automatic generation of data videos with narration-animation interplay,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 30, pp. 109–119, 1 2024.

[15] P. Chi, Z. Sun, K. Panovich, y I. Essa, “Automatic video creation from a web page.” ACM, 10 2020, pp. 279–292.

[16] A. Ohlei, T. Schumacher, y M. Herczeg, “Videoedit: An easy-to-use web-based video creation tool for the classroom,” 3 2021, pp. 6076–6085.

[17] K. C. Pai, R. Nayak, S. Badal, S. U. Bhat, y M. Geethalaxmi, “University results extraction and analysis tool using batch processing implementation system,” *International Journal of Advances in Engineering and Management (IJAEM)*, vol. 5, p. 94, 2023. [En línea]. Disponible en: www.ijaem.net

[18] R. O. Davis y Y. J. Lee, “Prompt: Chatgpt, create my course, please!” *Education Sciences*, vol. 14, p. 24, 12 2023.

[19] F. Wang, X. Liu, O. Liu, A. Neshati, T. Ma, M. Zhu, y J. Zhao, “Slide4n: Creating presentation slides from computational notebooks with human-ai collaboration.” ACM, 4 2023, pp. 1–18.

[20] M. Mansoor, S. Chandar, y R. Srinath, “Ai based presentation creator with customized audio content delivery,” 6 2021. [En línea]. Disponible en: https://arxiv.org/abs/2106.14213v1

[21] N. Rane, S. Choudhary, y J. Rane, “Gemini versus chatgpt: applications, performance, architecture, capabilities, and implementation,” *Journal of Applied Artificial Intelligence*, vol. 5, pp. 69–93, 3 2024.

[22] Y. Guo, Z. Zhang, Y. Liang, D. Zhao, y N. Duan, “Pptc benchmark: Evaluating large language models for powerpoint task completion,” 11 2023. [En línea]. Disponible en: https://arxiv.org/abs/2311.01767v2

[23] N. Bumann, “Automated chatbot using speech-to-text and text-to-speech with mobile app integration,” 2023. [En línea]. Disponible en: https://folia.unifr.ch/global/documents/326838

[24] M. Tahvanainen, “Hybrid mobile development using ionic framework,” 2023. [En línea]. Disponible en: http://www.cs.helsinki.fi/

[25] A. Wijerathne y D. Oddugama, “Create-my-course: An automated course generator for self-paced programs,” *International Journal of Innovative Science and Research Technology*, vol. 7, 2022. [En línea]. Disponible en: www.ijisrt.com

[26] M. A. Rojas, J. Carlos, y M. García, “Componentes de software 1 introducciÓn y principios bÁsicos del desarrollo de software basado en componentes,” 2004. [En línea]. Disponible en: www.osmosislatina.com/diversos/open_source.ht

[27] F. Tejera-Martínez, D. Aguilera, y J. M. Vílchez-González, “Lenguajes de programación y desarrollo de competencias clave. revisión sistemática,” *Revista Electrónica de Investigación Educativa*, vol. 22, pp. 1–16, 12 2020. [En línea]. Disponible en: https://redie.uabc.mx/redie/article/view/2869/2053https://redie.uabc.mx/redie/article/view/2869

[28] E. K. Bravo-Huivin, S. E. Cieza-Mostacero, L. A. Flores-Rodriguez, y L. Uceda-Davila, “Revisión sistemática de la literatura sobre metodologías Ágiles en el desarrollo de software aplicadas a la gestión de proyectos,” *RISTI - Revista Iberica de Sistemas e Tecnologias de Informacao*, vol. 2022, pp. 314–326, 2022. [En línea]. Disponible en: https://research.upn.edu.pe/es/publi

cations/revisi%C3%B3n-sistem%C3%A1tica-de-la-literatura-sobre-metodolog%C3%ADas-%C3%A1giles-e

[29] “Vista de cómo validar una entrevista de preguntas abiertas: una propuesta para investigación filosófica empírica.” [En línea]. Disponible en: https://sabereseducativos.uchile.cl/index.php/RSED/article/view/71389/73709

[30] “El diseño del análisis cualitativo multinivel: una aplicación práctica para el análisis de entrevistas multi-level qualitative analysis design: a practical application for the analysis of interviews,” 2010.

[31] “Chatbot arena llm leaderboard: Community-driven evaluation for best llm and ai chatbots.” [En línea]. Disponible en: https://lmarena.ai

[32] “Holistic evaluation of language models (helm).” [En línea]. Disponible en: https://crfm.stanford.edu/helm/mmlu/latest/#/leaderboard

[33] I. Deremuk, “Native vs web vs cross-platform: What to choose for mobile app development,” Aug 2024. [En línea]. Disponible en: https://litslink.com/blog/mobile-applications-development-native-web-cross-platform

[34] F. Khan, S. Rasheed, M. Alsheshtawi, T. Mohamed, y S. Jan, “A comparative analysis of rad and agile technique for management of computing graduation projects,” *Cmc -Tech Science Press-*, vol. 64, pp. 777–796, 06 2020.

[35] S. Alsaqqa, S. Sawalha, y H. Abdel-Nabi, “Agile software development: Methodologies and trends,” *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 14, no. 11, p. pp. 246–270, Jul. 2020. [En línea]. Disponible en: https://online-journals.org/index.php/i-jim/article/view/13269

[36] J. A. Gaete, R. Villarroe, I. Figueroa, H. A. Cornide-Reyes, y R. MuÃ, “Enfoque de aplicación ágil con Scrum, Lean y Kanban,” *Ingeniare. Revista chilena de ingenierÃa*, vol. 29, pp. 141 – 157, 03 2021. [En línea]. Disponible en: http://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0718-33052021000100141&nrm=iso

ANEXOS

Anexo A. Guía de entrevista a los instructores de la empresa Ingedemy

Tabla A1. Guía de entrevista a los instructores de la empresa Ingedemy

N. Pregunta	Pregunta
1.	¿Cómo cree que la integración de nuevas tecnologías podría mejorar la calidad del contenido educativo?
2.	¿Qué funcionalidades de automatización cree que serían más beneficiosas para incorporar en su proceso de trabajo?
3.	¿Cuáles son los pasos clave en el proceso actual de producción de videos educativos en Ingedemy?
4.	¿Puede describir algún cuello de botella o desafío recurrente que enfrentan durante la producción de estos videos?
5.	¿Hay factores específicos que, en su experiencia, afecten negativamente la calidad del contenido audiovisual educativo?
6.	¿Está familiarizado con alguna tecnología de automatización actualmente utilizada en la producción de contenido audiovisual?
7.	¿Qué herramientas de software considera que son efectivas para la automatización de la producción de videos educativos?
8.	¿Conoce algún ejemplo de software que haya ayudado a reducir los tiempos de producción en otros contextos similares?
9.	¿Qué características considera esenciales en un sistema destinado a optimizar la producción de videos educativos?
10.	¿Qué expectativas tiene sobre un sistema automatizado en términos de funcionalidad y facilidad de uso?

Anexo B. Prompts para la creación de diapositivas mediante Idea, Párrafo o Nueva Diapositiva

```
GENERAR EN BASE A UNA IDEA

`You are tasked with generating a JSON object that represents a presentation on the theme:
${script}. The presentation will be displayed on a DOM canvas.
Follow the structure and attributes described below to create each slide, ensuring that all
elements are appropriately positioned and styled to fit the theme.

JSON Structure:
width: Canvas width in pixels ${width}.
height: Canvas height in pixels ${height}.
pages: An array of slide objects. Each slide contains:
id: Unique slide identifier.
children: An array of element objects. Each element can be of type text, line, or image,
with
specific attributes.

Element Attributes:
Text Elements (type: "text")

id: String Unique identifier.
name: Name of the element.
x, y: Position coordinates.
width, height: Dimensions.
text: The text content.
fontSize: Font size.
fontFamily: Font family (e.g., "Arial").
fontWeight: Font weight (e.g., "bold", "normal").
fill: Text color (e.g., "#000000").
align: Text alignment (e.g., "left", "center", "right", "justify").

Line Elements (type: "line")

id: Unique identifier.
name: Name of the element.
x, y: Position coordinates.
width, height: Dimensions.
color: Line color ("#000000").
dash: Dash pattern for the line (e.g., [4, 2]).

Image Elements (type: "image")

id: Unique identifier.
name: Name of the element.
x, y: Position coordinates.
width, height: Dimensions.
src: "search criteria of topic provided at start of prompt",
cornerRadius: Radius
```

```
Directives:
Content language: ${language}
Don't provide any placeholder text only the json object.
Adhere to the structure: Ensure your JSON object follows the provided structure, including
the width, height, pages, id, and children attributes.
Use specific attributes: Each element (text, line, image) must include all relevant
attributes as described.
Theme consistency: Align all content (text, images, lines) with the specified theme
${script}.
Clarity and readability: Ensure text elements are clear and readable, using appropriate
font sizes and colors.
Visual appeal: Make the presentation visually appealing by appropriately positioning and
styling elements.
Title color: Use the specified title color #4E2ADC.
Text alignment: always use justify in each paragraph.
Number of slides: Create the specified number of slides ${numSlides}.
Text amount: Adjust the text amount in each paragraph based on the specified value ${amount}
("bit: 100 words", "half: 160 words", "quite: 220 words").
Paragraphs per slide: Ensure each slide contains between 3 to 5 paragraphs based on what
is pertinent for the slide.
Use these Slide designs where you think necessary:
-  Slide n: Title centered at the top, followed by a horizontal line, paragraphs aligned
to the left occupying the left half of the slide width, and an image aligned to the right
occupying the right half.
-  Slide n: Title centered at the top, followed by a horizontal line, paragraphs aligned
to the right occupying the right half of the slide width, and an image aligned to the left
occupying the left half.
-  Slide n: Title centered at the top, followed by a horizontal line, paragraphs aligned
to the left occupying the left half of the slide width, and paragraphs aligned to the right
occupying the right half of the slide width.
-  Slide n: Title centered at the top, followed by a horizontal line, paragraphs aligned
to the center occupying all slide width.

Additionally, ensure the following:

Each slide should have a title element using the specified title color.
Each slide should include a line element as a separator.
Use bullet points where applicable to enhance readability.
Include images where you deem necessary.

${guidelines === '' ? '' : `Follow these guidelines: ${guidelines}`}`
```

```
  GENERAR EN BASE A UN PÁRRAFO

`You are tasked with generating a JSON object that represents a presentation to explain
this text: "${text}". The presentation will be displayed on a DOM canvas.
Follow the structure and attributes described below to create each slide, ensuring that all
elements are appropriately positioned and styled to fit the theme.

JSON Structure:
width: Canvas width in pixels ${width}.
height: Canvas height in pixels ${height}.
pages: An array of slide objects. Each slide contains:
id: Unique slide identifier.
children: An array of element objects. Each element can be of type text, line, or image, with
specific attributes.

Element Attributes:
Text Elements (type: "text")

id: String Unique identifier.
name: Name of the element.
x, y: Position coordinates.
width, height: Dimensions.
text: The text content.
fontSize: Font size.
fontFamily: Font family (e.g., "Arial").
fontWeight: Font weight (e.g., "bold", "normal").
fill: Text color (e.g., "#000000").
align: Text alignment (e.g., "left", "center", "right", "justify").


Line Elements (type: "line")

id: Unique identifier.
name: Name of the element.
x, y: Position coordinates.
width, height: Dimensions.
color: Line color ("#000000").
dash: Dash pattern for the line (e.g., [4, 2]).

Image Elements (type: "image")

id: Unique identifier.
name: Name of the element.
x, y: Position coordinates.
width, height: Dimensions.
src: "search criteria of topic provided at start of prompt",
cornerRadius: Radius

Directives:
Content language: ${language}
Adhere to the structure: Ensure your JSON object follows the provided structure, including
```

```
the width, height, pages, id, and children attributes.
Use specific attributes: Each element (text, line, image) must include all relevant
attributes as described.
Theme consistency: Align all content (text, images, lines).
Clarity and readability: Ensure text elements are clear and readable, using appropriate font
sizes and colors.
Visual appeal: Make the presentation visually appealing by appropriately positioning and
styling elements.
Title color: Use the specified title color #4E2ADC.
Text alignment: always use justify in each paragraph.
Number of slides: Create the specified number of slides ${numSlides}.
Text amount: Adjust the text amount in each paragraph based on the specified value ${amount}
("bit: 120 words", "half: 180 words", "quite: 250 words").
Paragraphs per slide: Ensure each slide contains between 3 to 5 paragraphs based on what is
pertinent for the slide.
Use these Slide designs where you think necessary:
-  Slide n: Title centered at the top, followed by a horizontal line, paragraphs aligned to
the left occupying the left half of the slide width, and an image aligned to the right
occupying the right half.
-  Slide n: Title centered at the top, followed by a horizontal line, paragraphs aligned to
the right occupying the right half of the slide width, and an image aligned to the left
occupying the left half.
-  Slide n: Title centered at the top, followed by a horizontal line, paragraphs aligned to
the left occupying the left half of the slide width, and paragraphs aligned to the right
occupying the right half of the slide width.
-  Slide n: Title centered at the top, followed by a horizontal line, paragraphs aligned to
the center occupying all slide width.

Additionally, ensure the following:

Each slide should have a title element using the specified title color.
Each slide should include a line element as a separator.
Use bullet points where applicable to enhance readability.
Include images where you deem necessary.

${guidelines === '' ? '' : `Follow these guidelines: ${guidelines}`}`
```

```
GENERAR UNA NUEVA DIAPOSITIVA

`You are tasked with generating a JSON object that represents an slide presentation with
the theme: ${script}. The presentation will be displayed on a DOM canvas.
Follow the structure and attributes described below to create each slide, ensuring that all
elements are appropriately positioned and styled to fit the theme.

JSON Structure:
id: Unique slide identifier "${Date.now()}".
children: An array of element objects. Each element can be of type text, line, or image,
with specific attributes.

Element Attributes:
Text Elements (type: "text")

id: String Unique identifier, created by concatenating the element's name with the current
timestamp (e.g., "Title${Date.now()}").
name: Name of the element.
x, y: Position coordinates as numbers (e.g., x: 960, y: 50).
width, height: Dimensions as numbers (e.g., width: 800, height: 59).
text: The text content.
fontSize: Font size as a number (e.g., fontSize: 24).
fontFamily: Font family (e.g., "Arial").
fontWeight: Font weight (e.g., "bold", "normal").
fill: Text color (e.g., "#000000").
align: Text alignment (e.g., "left", "center", "right", "justify").


Line Elements (type: "line")

id: Unique identifier, created by concatenating the element's name with the current
timestamp (e.g., "Line${Date.now()}").
name: Name of the element.
x, y: Position coordinates as numbers (e.g., x: 50, y: 100).
width, height: Dimensions as numbers (e.g., width: 800, height: 2).
color: Line color ("#000000").
dash: Dash pattern for the line (e.g., [4, 2]).

Image Elements (type: "image")

id: Unique identifier, created by concatenating the element's name with the current
timestamp (e.g., "Image${Date.now()}").
name: Name of the element.
x, y: Position coordinates as numbers (e.g., x: 500, y: 200).
width, height: Dimensions as numbers (e.g., width: 400, height: 300).
src: "search criteria of topic provided at start of prompt",
cornerRadius: Radius

Directives:
Content language: ${language}
Keep in mind to the size ad position should be according a slide size of width: ${width}
```

```
and height:${height}
Don't provide any placeholder text only the json object
Adhere to the structure: Ensure your JSON object follows the provided structure, including
the id, and children attributes.
Use specific attributes: Each element (text, line, image) must include all relevant
attributes as described.
Theme consistency: Align all content (text, images, lines).
Clarity and readability: Ensure text elements are clear and readable, using appropriate
font sizes and colors.
Visual appeal: Make the presentation visually appealing by appropriately positioning and
styling elements.
Title color: Use the specified title color #4E2ADC.
Title size: 48.
Text alignment: always use justify in each paragraph.
This is only one slide of a hole presentation
Text amount: Adjust the text amount in each paragraph based on the specified value ${amount}
("bit: 120 words", "half: 180 words", "quite: 250 words").
Paragraphs per slide: Ensure that the slide contains between 3 to 5 paragraphs based
on what is pertinent for the slide.
Use these Slide designs where you think necessary:
-  Slide n: Title centered at the top, followed by a horizontal line, paragraphs aligned
to the left occupying the left half of the slide width, and an image aligned to the right
occupying the right half.
-  Slide n: Title centered at the top, followed by a horizontal line, paragraphs aligned to
the right occupying the right half of the slide width, and an image aligned to the left
occupying the left half.
-  Slide n: Title centered at the top, followed by a horizontal line, paragraphs aligned to
the left occupying the left half of the slide width, and paragraphs aligned to the right
occupying the right half of the slide width.
-  Slide n: Title centered at the top, followed by a horizontal line, paragraphs aligned to
the center occupying all slide width.

Additionally, ensure the following:

Slide should have a title element using the specified title color.
Slide should include a line element as a separator.
Use bullet points where applicable to enhance readability.
Include images where you deem necessary.
Distributes the sizes and position of the elements on a canvas of width: ${width} and
height: ${height}.

Please format your response exactly as follows:
{
  "slide": {
    // Your slide JSON structure here
  }
}

${guidelines === '' ? '' : `Follow these guidelines: ${guidelines}`}`
```

```
const { data: user, error: loginError } = await supabase.auth.signInWithPassword({ email, password });

if (loginError) {
  setError('Invalid credentials');
  return;
}

// Verificar si el usuario está activo
const { data: userData, error: userError } = await supabase
  .from('users')
  .select('active')
  .eq('uuid', user.user.id)
  .single();

if (userError || !userData.active) {
  setError('Usuario desactivado. Por favor, contacta al administrador.');
  return;
}

// Redirigir al home si el login es exitoso
router.push('/home');
```

Figura C1. Función de autenticación de usuarios

```
try {
    // Crear el usuario en Supabase Auth y envia un correo de confirmación
    const { data: user, error: userError } = await supabase.auth.signUp({
        email: email,
        password: password,
    });

    if (userError) {
        setMessage("Error creando el usuario: " + userError.message);
        setLoading(false);
        return;
    }

    // Insertar en la tabla intermedia 'has_roles'
    const { error: userInsertError } = await supabase
        .from('users')
        .insert([{ uuid: user.user.id, email: email, active: true }]);

    if (userInsertError) {
        setMessage("Error insertando el usuario: " + userInsertError.message);
        setLoading(false);
        return;
    }

    // Asignar el rol
    const { error: roleError } = await supabase
        .from('has_roles')
        .insert([{ user_id: user.user.id, rol_id: selectedRole }]);

    if (roleError) {
        setMessage("Error asignando el rol: " + roleError.message);
        setLoading(false);
        return;
    }

    // Actualizar la lista de usuarios
    fetchUsers();
```

Figura C2. Función para la creación de usuarios y asignación de roles



```json
{
    "width": 1920,
    "height": 1080,
    "fonts": [],
    "pages": [
        {
            "id": "slide_1",
            "children": [
                {
                    "id": "title1",
                    "type": "text",
                    "name": "Title",
                    "x": 250,
                    "y": 50,
                    "width": 1032,
                    "height": 59,
                    "text": "Hola",
                    "fontSize": 48,
                    "fontFamily": "Arial",
                    "fontWeight": "bold",
                    "fill": "#4E2ADC",
                    "align": "center"
                },
                {
                    "id": "line1",
                    "type": "line",
                    "name": "Separator Line",
                    "x": 250,
                    "y": 120,
                    "width": 1032,
                    "height": 2,
                    "color": "#4E2ADC",
                    "dash": [
                        4,
                        2
                    ]
                },
                {
                    "id": "ingedemy_image",
                    "type": "image",
                    "name": "INGEDEMY",
                    "x": 1640,
                    "y": 990,
                    "width": 250,
                    "height": 60,
                    "src": "/images/INGEDEMY.jpeg",
                    "cornerRadius": 0
                }
            ],
            "background": "white"
        }
    ],
    "unit": "px",
    "dpi": 72
}
```

Figura C3. Ejemplo de JSON para la configuración de una diapositiva en Polotno

```
const generateWithIdea = (script, numSlides, amount, language, guidelines, width, height) => `
You are tasked with generating a JSON object that represents a presentation on the theme: ${script}. The presentation will be displayed on a DOM canvas.
Follow the structure and attributes described below to create each slide, ensuring that all elements are appropriately positioned and styled to fit the theme.

JSON Structure:
    width: Canvas width in pixels ${width}.
    height: Canvas height in pixels ${height}.
    pages: An array of slide objects. Each slide contains:
    id: Unique slide identifier.
    children: An array of element objects. Each element can be of type text, line, or image, with specific attributes.

    Element Attributes:
        Text Elements (type: "text")

            id: String Unique identifier.
            name: Name of the element.
            x, y: Position coordinates.
            width, height: Dimensions.
            text: The text content.
            fontSize: Font size.
            fontFamily: Font family (e.g., "Arial").
            fontWeight: Font weight (e.g., "bold", "normal").
            fill: Text color (e.g., "#000000").
            align: Text alignment (e.g., "left", "center", "right", "justify").


        Line Elements (type: "line")

            id: Unique identifier.
            name: Name of the element.
            x, y: Position coordinates.
            width, height: Dimensions.
            color: Line color ("#000000").
            dash: Dash pattern for the line (e.g., [4, 2]).
```

Figura C4. Prompt del método `generateWithIdea`

```
const generateWithParagraph = (text, numSlides, amount, language, guidelines, width, height) => `
You are tasked with generating a JSON object that represents a presentation to explain this text: "${text}". The presentation will be displayed on a DOM canvas.
Follow the structure and attributes described below to create each slide, ensuring that all elements are appropriately positioned and styled to fit the theme.

JSON Structure:
    width: Canvas width in pixels ${width}.
    height: Canvas height in pixels ${height}.
    pages: An array of slide objects. Each slide contains:
    id: Unique slide identifier.
    children: An array of element objects. Each element can be of type text, line, or image, with specific attributes.

    Element Attributes:
        Text Elements (type: "text")

            id: String Unique identifier.
            name: Name of the element.
            x, y: Position coordinates.
            width, height: Dimensions.
            text: The text content.
            fontSize: Font size.
            fontFamily: Font family (e.g., "Arial").
            fontWeight: Font weight (e.g., "bold", "normal").
            fill: Text color (e.g., "#000000").
            align: Text alignment (e.g., "left", "center", "right", "justify").


        Line Elements (type: "line")

            id: Unique identifier.
            name: Name of the element.
            x, y: Position coordinates.
            width, height: Dimensions.
            color: Line color ("#000000").
            dash: Dash pattern for the line (e.g., [4, 2]).
```

Figura C5. Prompt del método `generateWithParagraph`

```jsx
import React from "react";
import { PolotnoContainer, SidePanelWrap, WorkspaceWrap } from "polotno";
import { createStore } from "polotno/model/store";

const store = createStore({
  key: "YOUR_POLTNO_API_KEY", // Reemplazar con la clave API de Polotno
});

const SlideEditor = () => {
  return (
    <PolotnoContainer style={{ width: "100%", height: "100vh" }}>
      <SidePanelWrap store={store} />
      <WorkspaceWrap store={store} />
    </PolotnoContainer>
  );
};

export default SlideEditor;
```


Figura C7. Código de configuración de Polotno para la interfaz de edición de diapositivas.

```
const openai = new OpenAI({
    apiKey: savedData.input1,
    dangerouslyAllowBrowser: true
});

let finalPrompt = '';
if (promptName != '') {
    finalPrompt = generateWithIdea(prompt, slides, amount, lang, guidelines, w, h);
} else {
    finalPrompt = generateWithParagraph(prompt, slides, amount, lang, guidelines, w, h);
}

let conversation = [{ role: "system", content: finalPrompt }];
let fullContent = '';
let isComplete = false;
let totalPromptTokens = 0;
let totalCompletionTokens = 0;

while (!isComplete) {

    // console.log('llama');
    // console.log({conversation});

    const completion = await openai.chat.completions.create({
        model: "gpt-4o",
        messages: conversation as any,
        max_tokens: 4096,
    });

    const response = completion.choices[0].message.content;
    fullContent += response.replaceAll('```json', "").replaceAll("```", "");

    totalPromptTokens += completion.usage.prompt_tokens;
    totalCompletionTokens += completion.usage.completion_tokens;

    if (completion.choices[0].finish_reason === 'length') {
        conversation.push({
            role: 'assistant',
            content: response
        })
        conversation.push({
            role: 'user',
            content: 'Continue the json from where you left off (exact point of the last message). If the last message is unfinished JSON object dont start it again continue it...
    } else {
        isComplete = true;
    }

    // console.log({ fullContent })
}
```

Figura C6. Código de generación recursiva de JSON

```
const imagesUrls = [];

fetch(
  `https://www.googleapis.com/customsearch/v1?key=${apiKey}&cx=${cx}&searchType=image&q=${query}&imgSize=large`
)
  .then((response) => response.json())
  .then((data) => {

    const items = data.items;
    if (items) {
      items.forEach((item) => {
        imagesUrls.push({ url: item.link });
      });
    } else {
      console.log("No se encontraron imágenes.");
    }
  })
  .catch((error) => {
    console.error("Error al hacer la solicitud:", error);
  });

await new Promise((resolve) => setTimeout(resolve, 3000));

setImages(imagesUrls);
```

Figura C8. Código de integración de imágenes mediante la API de Google para Polotno.

```
const generateNewSlide = (script, amount, language, guidelines, width, height) => `
You are tasked with generating a JSON object that represents an slide presentation with the theme: ${script}. The presentation will be displayed on a DOM canvas.
Follow the structure and attributes described below to create each slide, ensuring that all elements are appropriately positioned and styled to fit the theme.

JSON Structure:
    id: Unique slide identifier "${Date.now()}".
    children: An array of element objects. Each element can be of type text, line, or image, with specific attributes.

    Element Attributes:
        Text Elements (type: "text")

            id: String Unique identifier, created by concatenating the element's name with the current timestamp (e.g., "Title${Date.now()}").
            name: Name of the element.
            x, y: Position coordinates as numbers (e.g., x: 960, y: 50).
            width, height: Dimensions as numbers (e.g., width: 800, height: 59).
            text: The text content.
            fontSize: Font size as a number (e.g., fontSize: 20).
            fontFamily: Font family (e.g., "Arial").
            fontWeight: Font weight (e.g., "bold", "normal").
            fill: Text color (e.g., "#000000").
            align: Text alignment (e.g., "left", "center", "right", "justify").


        Line Elements (type: "line")

            id: Unique identifier, created by concatenating the element's name with the current timestamp (e.g., "Line${Date.now()}").
            name: Name of the element.
            x, y: Position coordinates as numbers (e.g., x: 50, y: 100).
            width, height: Dimensions as numbers (e.g., width: 800, height: 2).
            color: Line color ("#000000").
            dash: Dash pattern for the line (e.g., [4, 2]).
```

Figura C9. Prompt del método generateNewSlide

```
const generate = async () => {

    setWorking(true)

    if (!script) {
        if (content.pages.length > 0 && !texts) {
            await generateInitialScripts(content);
            setWorking(false)
        }
    } else {
        const slideMap = new Map();
        script.slides.forEach((slide) => {
            slideMap.set(slide.id, slide.script);
        });
        const newTexts = content.pages.map((page) => {
            return slideMap.get(page.id) || "";
        });

        const emptyPositions = [];

        newTexts.forEach((text, index) => {
            if (text === "") {
                emptyPositions.push(index);
            }
        });

        if (emptyPositions.length > 0) {
            await generateMissingScripts(content, script, emptyPositions);
        }

        setWorking(false);
    }
}
```

Figura C10. Flujo principal de generación de guiones en el método `generate`

```
const generateInitialScripts = async (content) => {
    try {
        const slidesIds = content.pages.map(page => page.id);
        const base64Images = await Promise.all(
            content.pages.map((_, i) => convertImageToDataURLSync(`local://cursos/${idP}/${idV}/${i}.png`))
        );

        const generatioResult = await retryOrNotify(() => generateScriptForSlides(base64Images));

        const newScripts = { slides: [] };
        // console.log(generatioResult);

        if (generatioResult.scripts.length === slidesIds.length) {
            generatioResult.scripts.forEach((script, i) => {
                newScripts.slides.push({
                    script,
                    id: slidesIds[i]
                });
            });
        } else {
            console.error("La longitud de los arrays no coincide");
        }

        const cost = calculateCost(generatioResult.inputTokens, generatioResult.outputTokens);
        const newPrice = await saveScriptsAndUpdatePrice(newScripts, cost);

        setScript(newScripts);
        setPrice(newPrice);

        const newTexts = newScripts.slides.map((item) => item.script);
        setTexts(newTexts);

    } catch (error) {
        console.error("Error in generateInitialScripts:", error);
    }
};
```

Figura C11. Proceso de generación de guiones iniciales en `generateInitialScripts`

```
const generateMissingScripts = async (content, script, emptyPositions) => {
    try {
        const base64Images = await Promise.all(
            emptyPositions.map((i) => convertImageToDataURLSync(`local://cursos/${idP}/${idV}/${i}.png`))
        );

        const generatioResult = await retryOrNotify(() => generateScriptForSlides(base64Images));

        // console.log(generatioResult);

        const newScripts = { slides: [] };
        const slidesIds = emptyPositions.map((emptyPosition) => content.pages[emptyPosition].id);


        if (generatioResult.scripts.length === slidesIds.length) {
            generatioResult.scripts.forEach((script, i) => {
                newScripts.slides.push({
                    script,
                    id: slidesIds[i]
                });
            });
        } else {
            console.error("La longitud de los arrays no coincide");
        }
        const cost = calculateCost(generatioResult.inputTokens, generatioResult.outputTokens);
        const scriptsToUpdate = { slides: [] };

        content.pages.forEach((page) => {
            const allScripts = [...newScripts.slides, ...script.slides];
            const slideScript = allScripts.find(s => s.id == page.id);
            scriptsToUpdate.slides.push(slideScript);
        });

        const newPrice = await saveScriptsAndUpdatePrice(scriptsToUpdate, cost);

        setPrice(newPrice);
        const newTexts = scriptsToUpdate.slides.map((item) => item.script);
        setScript(scriptsToUpdate);
        setTexts(newTexts);

    } catch (error) {
        console.error("Error in generateMissingScripts:", error);
    }
};
```

Figura C12. Generación de guiones faltantes en el método `generateMissingScripts`

```
const convertImageToDataURLSync = (src) => {
    return new Promise((resolve, reject) => {
        const img = new Image();
        img.crossOrigin = 'Anonymous';
        img.src = src;
        img.onload = () => {
            const canvas = document.createElement('canvas');
            canvas.width = img.width;
            canvas.height = img.height;
            const ctx = canvas.getContext('2d');
            ctx.drawImage(img, 0, 0);
            const dataURL = canvas.toDataURL('image/png');
            resolve(dataURL);
        };
        img.onerror = (err) => {
            reject(err);
        };
    });
};
```

Figura C13. Conversión de imágenes a base64 con `convertImageToDataURLSync`



```
const generateScriptForSlides = async (base64Images) => {
    try {
        const promptMessage = {
            role: 'system',
            content: `You are a scriptwriter for educational videos. Your task is to create a script for...
        };

        const imageMessages = base64Images.map(base64 => ({
            role: 'user',
            content: [
                {
                    "type": "image_url",
                    "image_url": {
                        "url": `${base64}`,
                        "detail": "low"
                    },
                },
            ],
        }));

        const messages = [promptMessage, ...imageMessages];

        const savedData = JSON.parse(localStorage.getItem("settingsData"));

        const openai = new OpenAI({
            apiKey: savedData.input1,
            dangerouslyAllowBrowser: true
        });

        const completion = await openai.chat.completions.create({
            model: "gpt-4o",
            messages: messages,
            max_tokens: 4096,
        });

        const response = completion.choices[0].message.content;
        const content = response.replaceAll('```json', "").replaceAll("```", "");

        const responseJSON = JSON.parse(content);

        const inputTokens = completion.usage.prompt_tokens;
        const outputTokens = completion.usage.completion_tokens;

        return {
            scripts: responseJSON.slides.map(a => a.script),
            inputTokens, outputTokens
        };

    } catch (error) {
        console.error("Error generating scripts:", error);
        throw new Error("Script generation failed");
    }
}
```

Figura C14. Generación del guion para cada diapositiva en `generateScriptForSlides`

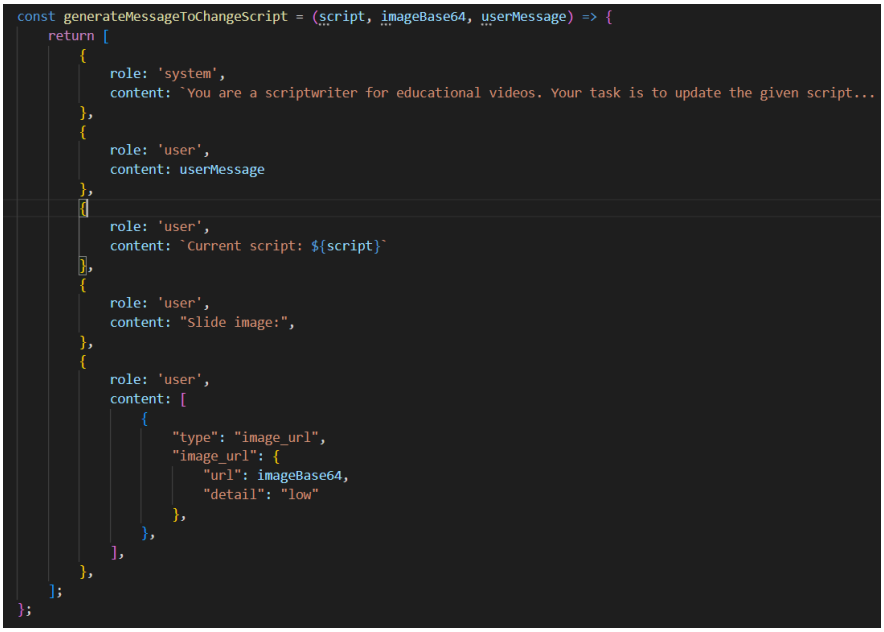


Figura C15. Método para reescribir el guion en base al feedback del usuario en `generateMessageToChangeScript`

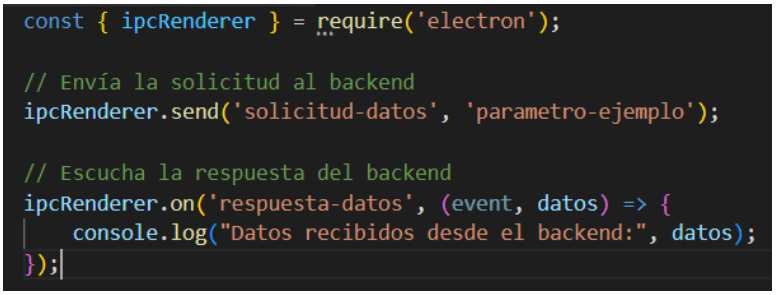


Figura C16. El frontend envía una solicitud de datos al backend a través de IPC

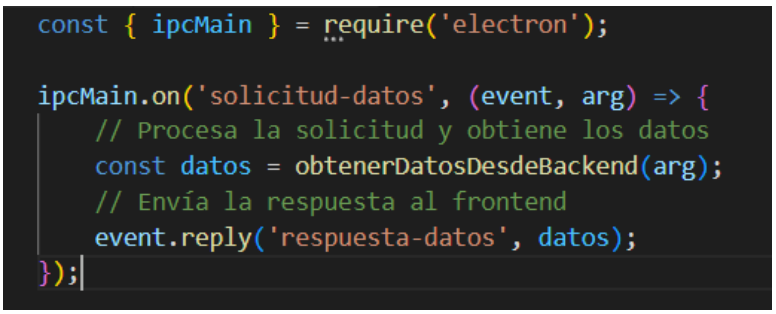


Figura C17. El backend procesa la solicitud y responde al frontend mediante IPC



```
// Configuración de Azure TTS
const apiKey = 'TU_CLAVE_AZURE';
const region = 'TU_REGION_AZURE'; // Ejemplo: eastus
const endpoint = `https://${region}.tts.speech.microsoft.com/cognitiveservices/v1`;

// Función para realizar la solicitud de TTS a Azure y guardar el audio
async function generateAudioFile(voice, text, filePath) {
  const ssml =
    <speak version="1.0" xmlns="http://www.w3.org/2001/10/synthesis" xml:lang="en-US">
      <voice name="${voice}">${text}</voice>
    </speak>;

  const response = await axios({
    method: 'post',
    url: endpoint,
    headers: {
      'Ocp-Apim-Subscription-Key': apiKey,
      'Content-Type': 'application/ssml+xml',
      'X-Microsoft-OutputFormat': 'audio-24khz-48kbitrate-mono-mp3',
    },
    data: ssml,
    responseType: 'arraybuffer',
  });

  // Guardar el archivo de audio en la ruta especificada
  fs.writeFileSync(filePath, response.data);
  console.log(`Archivo generado: ${filePath}`);
}
```

Figura C18. Código para la generación de audio con Azure TTS



```
const createVideo = async (idP, idV, scriptsLength, width, height) => {
  const baseDir = path.join(process.cwd(), "audios", idP, idV);
  const outputDir = path.join(process.cwd(), "videos", idP, idV);

  if (!fs.existsSync(outputDir)) {
    fs.mkdirSync(outputDir, { recursive: true });
  }

  const tempDir = path.join(outputDir, "temp");
  if (!fs.existsSync(tempDir)) {
    fs.mkdirSync(tempDir, { recursive: true });
  }

  // Crear videos individuales para cada imagen y audio
  const videoPromises = [];
  for (let i = 0; i < scriptsLength; i++) {
    const imagePath = path.join(
      process.cwd(),
      "cursos",
      idP.toString(),
      idV.toString(),
      `${i}.png`
    );
    const audioPath = path.join(baseDir, `${i}.mp3`);
    const videoPath = path.join(tempDir, `${i}.mp4`);

    const promise = new Promise((resolve, reject) => {
      ffmpeg()
        .input(imagePath)
        .loop() // Repite la imagen para que coincida con la duración del audio
        .input(audioPath)
        .audioCodec("aac")
        .videoCodec("libx264")
        .outputOptions("-b:a", "192k")
        .outputOptions("-map 0:v:0") // Map video stream
        .outputOptions("-map 1:a:0") // Map audio stream
        .outputOptions("-shortest") // Asegura que la salida tenga la misma duración que el audio
        .outputOptions("-vf", `scale=${width}:${height}`)
        .on("end", () => {
          console.log(`Video segment ${i} created successfully: ${videoPath}`);
          resolve(null);
        })
        .on("error", (err) => {
          console.error(`Error creating video segment ${i}:`, err);
          reject(err);
        })
        .save(videoPath);
    });

    videoPromises.push(promise);
  }
```

Figura C19. Código para la creación del video final usando FFMPEG

Anexo D. Mediciones de tiempos de los instructores

Tabla D1. Tiempos y Conformidad del Instructor 1

Etapa	**Producción Manual (min)**	**Producción Automatizada (min)**	**Conformidad Manual**	**Conformidad Automatizada**
Creación de diapositivas	40.50	25.10	4	5
Redacción de guión	46.20	20.00	5	4
Grabación	39.30	0.00	4	-
Edición	30.00	6.00	5	5

Tabla D2. Tiempos y Conformidad del Instructor 2

Etapa	**Producción Manual (min)**	**Producción Automatizada (min)**	**Conformidad Manual**	**Conformidad Automatizada**
Creación de diapositivas	42.00	26.00	5	4
Redacción de guión	45.50	19.70	4	5
Grabación	39.60	0.00	4	-
Edición	28.50	4.90	4	4

Tabla D3. Tiempos y Conformidad del Instructor 3

Etapa	**Producción Manual (min)**	**Producción Automatizada (min)**	**Conformidad Manual**	**Conformidad Automatizada**
Creación de diapositivas	41.00	24.80	5	5
Redacción de guión	45.00	19.50	3	4
Grabación	39.80	0.00	4	-
Edición	30.50	5.10	5	5

Tabla D4. Tiempos y Conformidad del Instructor 4

Etapa	**Producción Manual (min)**	**Producción Automatizada (min)**	**Conformidad Manual**	**Conformidad Automatizada**
Creación de diapositivas	41.30	25.90	4	5
Redacción de guión	46.10	19.90	5	5
Grabación	39.40	0.00	3	-
Edición	30.10	5.30	3	5

Tabla D5. Tiempos y Conformidad del Instructor 5

Etapa	**Producción Manual (min)**	**Producción Automatizada (min)**	**Conformidad Manual**	**Conformidad Automatizada**
Creación de diapositivas	42.10	26.20	5	5
Redacción de guión	46.00	20.10	5	5
Grabación	39.20	0.00	5	-
Edición	29.90	5.00	4	5

Anexo E. Calculo t-Student para la producción de videos

A continuación, se presentan los resultados del cálculo paso a paso para la prueba t-Student, utilizando los datos de cantidad de videos producidos antes y después del uso del software.

Cálculos Realizados

Media

$$\bar{X}_1 = 10.82 \quad \text{(Antes)}$$
$$\bar{X}_2 = 17.09 \quad \text{(Después)}$$

Varianza

$$s_1^2 = 1.56 \quad \text{(Antes)}$$
$$s_2^2 = 3.09 \quad \text{(Después)}$$

Tamaño de la muestra

$$n_1 = 11 \quad \text{(Antes)}$$
$$n_2 = 11 \quad \text{(Después)}$$

Grados de libertad

La fórmula para calcular los grados de libertad (df) es:

$$df = \frac{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}\right)^2}{\frac{\left(\frac{s_1^2}{n_1}\right)^2}{n_1 - 1} + \frac{\left(\frac{s_2^2}{n_2}\right)^2}{n_2 - 1}}.$$

Sustituyendo los valores:

$$\begin{aligned} df &= \frac{\left(\frac{1.56}{11} + \frac{3.09}{11}\right)^2}{\frac{\left(\frac{1.56}{11}\right)^2}{10} + \frac{\left(\frac{3.09}{11}\right)^2}{10}} \\ &= 18.06. \end{aligned}$$

Estadístico t

La fórmula para el cálculo del estadístico t es:

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}.$$

Sustituyendo los valores:

$$\begin{aligned} t &= \frac{10.82 - 17.09}{\sqrt{\frac{1.56}{11} + \frac{3.09}{11}}} \\ &= -9.64. \end{aligned}$$

Valor p

El valor p se obtiene utilizando el valor del estadístico t y los grados de libertad df. Para este caso:

$$p = 1.51 \times 10^{-8}$$

Anexo F. Manual de usuario para los instructores

Manual de usuario del sistema para los instructores de la empresa INGEDEMY

Daniel Gerardo Galarza Tamayo

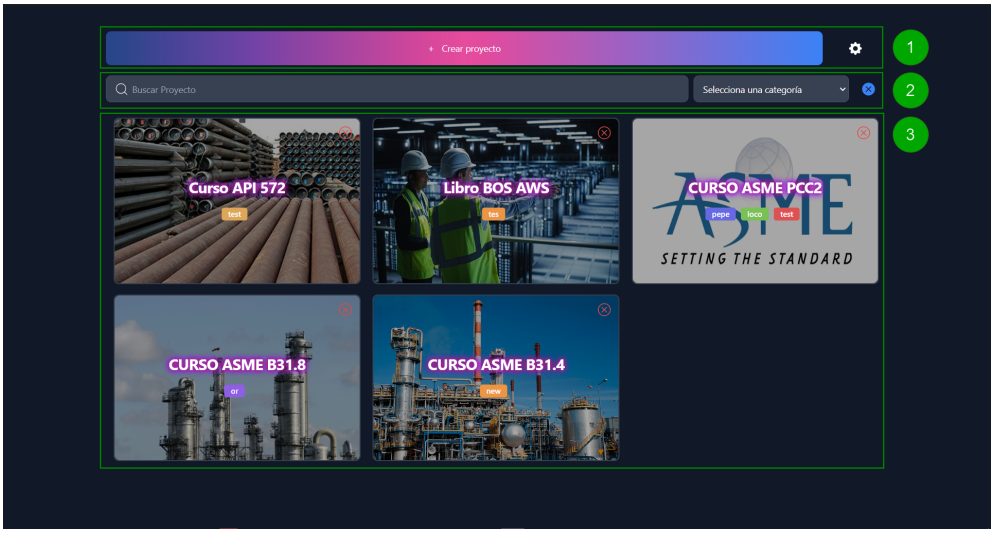
Noviembre, 2024

Introducción

Este manual de usuario proporciona una guía completa sobre el funcionamiento de la pantalla principal del sistema, destacando las principales secciones de la interfaz y sus funcionalidades.

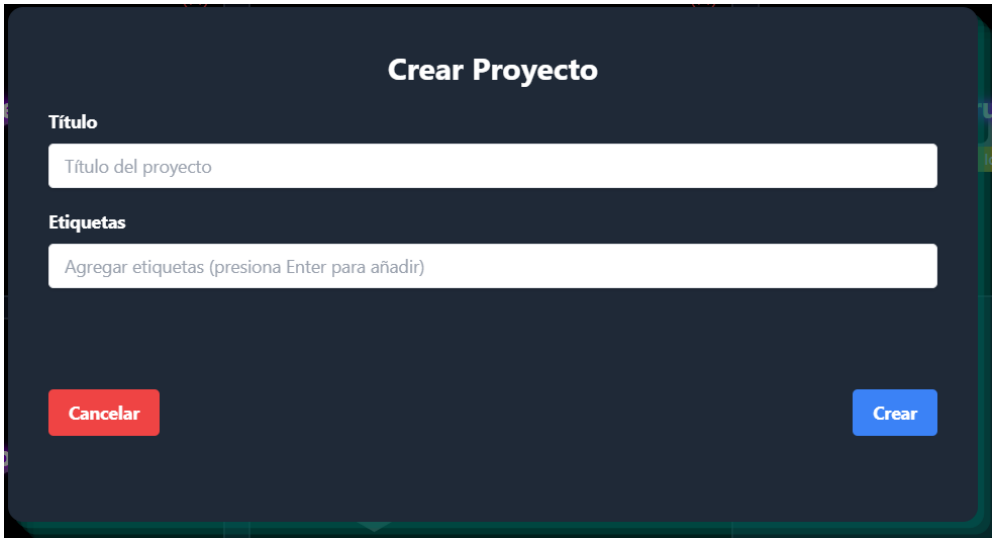
1 Pantalla Principal

1.1 Estructura de la Pantalla Principal

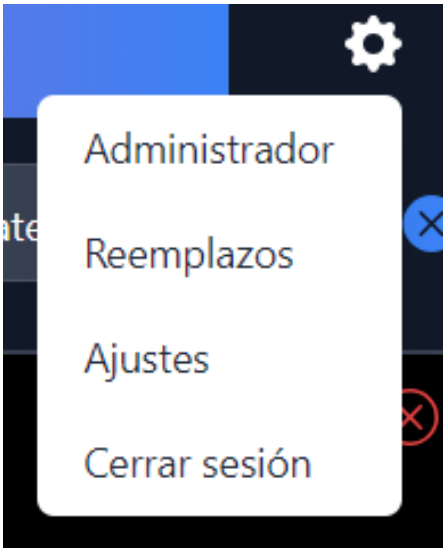


1. **Barra Superior**

    - **Botón "+ Crear proyecto"**: Permite al usuario crear un nuevo proyecto. Al hacer clic, se despliega un formulario para ingresar el nombre del proyecto y etiquetas opcionales.



- **Ícono de Configuración** (engranaje): Ubicado en la esquina superior derecha, permite al usuario acceder al menú de configuración del sistema.



2. **Barra de Búsqueda y Filtros**

   - **Buscar Proyecto**: Campo de texto para ingresar palabras clave y buscar proyectos específicos.
   - **Selector de Categoría**: Menú desplegable para seleccionar una categoría específica, mostrando solo los proyectos correspondientes.

3. **Lista de Proyectos**

   - **Logotipo del Proyecto**: Imagen de referencia del proyecto.
   - **Nombre del Proyecto**: Texto que indica el nombre del proyecto.

- **Etiquetas**: Etiquetas que proporcionan información adicional, como la categoría o estado del proyecto.
- **Botón Eliminar** (ícono de "X"): Ubicado en la esquina superior derecha de cada tarjeta, permite eliminar el proyecto.

1.2 Funcionalidades Principales

- **Crear un Proyecto Nuevo**
  1. Hacer clic en el botón **+ Crear proyecto**.
  2. Completar el formulario con el **nombre del proyecto** y agregar **etiquetas** (opcional).
  3. Guardar el proyecto para que aparezca en la lista de proyectos.
- **Buscar un Proyecto**
  1. Escribir el nombre o palabras clave en el campo **Buscar Proyecto**.
  2. Solo los proyectos coincidentes se mostrarán en la cuadrícula.
- **Filtrar por Categoría**
  1. Seleccionar una categoría en el menú **Selector de Categoría**.
  2. La pantalla mostrará únicamente los proyectos de la categoría seleccionada.
- **Visualizar Detalles de un Proyecto**
  1. Hacer clic en la tarjeta de un proyecto para ver una lista de los videos asociados a ese proyecto.
- **Eliminar un Proyecto**
  1. Hacer clic en el ícono de **X** en la esquina superior derecha de la tarjeta del proyecto.
  2. Esto eliminará el proyecto y su contenido de la lista.

2 Pantalla de Configuración

La pantalla de Configuración permite al usuario ingresar las credenciales de las APIs necesarias para el funcionamiento del sistema, así como seleccionar el modelo de ChatGPT a utilizar. Esta configuración es crucial para la operación de las funciones de generación de texto y voz en el sistema.

2.1 Estructura de la Pantalla de Configuración

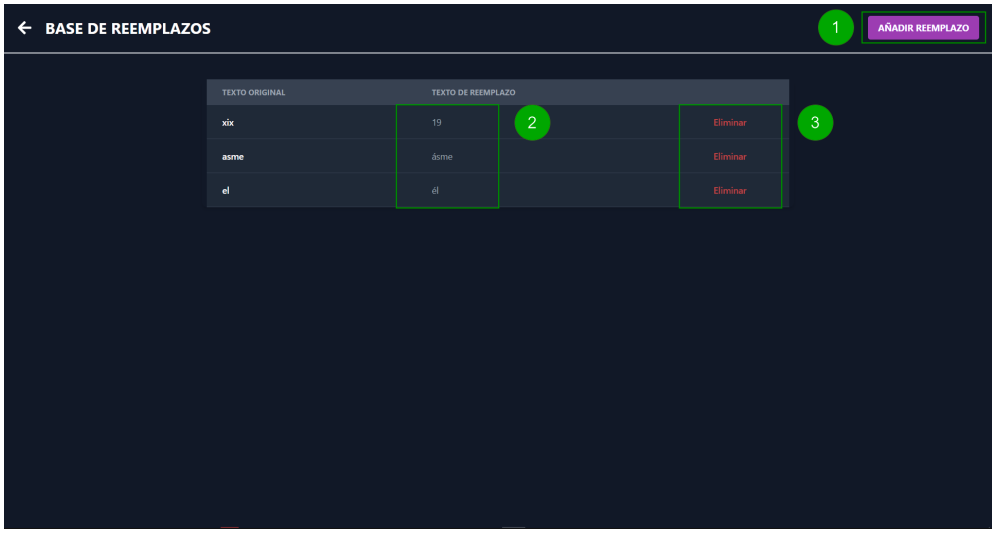


1. **Enlace Volver a Inicio**
    - Ubicado en la parte superior izquierda, este enlace permite al usuario regresar a la pantalla principal.
2. **API ChatGPT**
    - Campo de texto donde el usuario debe introducir la clave de la API de ChatGPT para habilitar las funcionalidades de generación de texto en el sistema.
3. **API Azure TTS**
    - Campo de texto para ingresar la clave de la API de Azure TTS (Text-to-Speech), la cual permite al sistema convertir texto en voz.
4. **Modelo ChatGPT**
    - Menú desplegable que permite al usuario seleccionar el modelo de ChatGPT a utilizar en el sistema.
5. **Botón Guardar**
    - Este botón permite guardar la configuración ingresada. Al hacer clic, los datos introducidos en los campos de API y la selección de modelo se almacenan para su uso en el sistema.

3 Pantalla de Reemplazos

La pantalla de Reemplazos permite al usuario definir reglas de sustitución de palabras o frases específicas con el fin de mejorar la pronunciación del sistema de Texto a Voz (TTS). De esta forma, el usuario puede asegurarse de que el TTS pronuncie correctamente términos que podrían resultar difíciles o que requieran una entonación especial.

3.1 Estructura de la Pantalla de Reemplazos



1. **Botón Añadir Reemplazo**

    - Ubicado en la parte superior derecha, permite al usuario agregar una nueva regla de reemplazo. Al hacer clic, se abre un formulario para ingresar el texto original y el texto de reemplazo deseado.



2. **Campo de Texto de Reemplazo (Editable)**
    - La columna **Texto de Reemplazo** contiene campos de entrada (input) que permiten al usuario editar directamente el texto de reemplazo.
    - Después de modificar el texto de reemplazo, el usuario debe presionar la tecla **Enter** para que el cambio se guarde correctamente en el sistema.
    - Este campo editable permite ajustar la forma en que el TTS interpretará y pronunciará el texto, logrando así una entonación o articulación más precisa.
3. **Botón Eliminar**
    - En cada fila de la lista, en la columna de **Texto de Reemplazo**, se encuentra el botón de **Eliminar** (en color rojo). Al hacer clic, se abrirá una advertencia de confirmación para asegurarse de que el usuario desea eliminar el reemplazo específico.



3.2 Funcionalidades Principales

- **Añadir un Nuevo Reemplazo**
    1. Hacer clic en el botón **Añadir Reemplazo**.
    2. Completar el formulario con el **Texto Original** y el **Texto de Reemplazo**, donde el Texto Original es la palabra o frase que se debe ajustar, y el Texto de Reemplazo es la forma en que el TTS debe pronunciarlo.
    3. Guardar la entrada para que aparezca en la lista de reemplazos.

- **Editar un Reemplazo**

    1. Localizar el campo editable en la columna **Texto de Reemplazo** correspondiente a la fila deseada.
    2. Hacer clic en el campo de texto y modificar el texto de reemplazo según sea necesario.
    3. Presionar la tecla **Enter** para que el cambio se guarde en el sistema, lo cual actualizará la forma en que el TTS pronuncia ese término o frase.

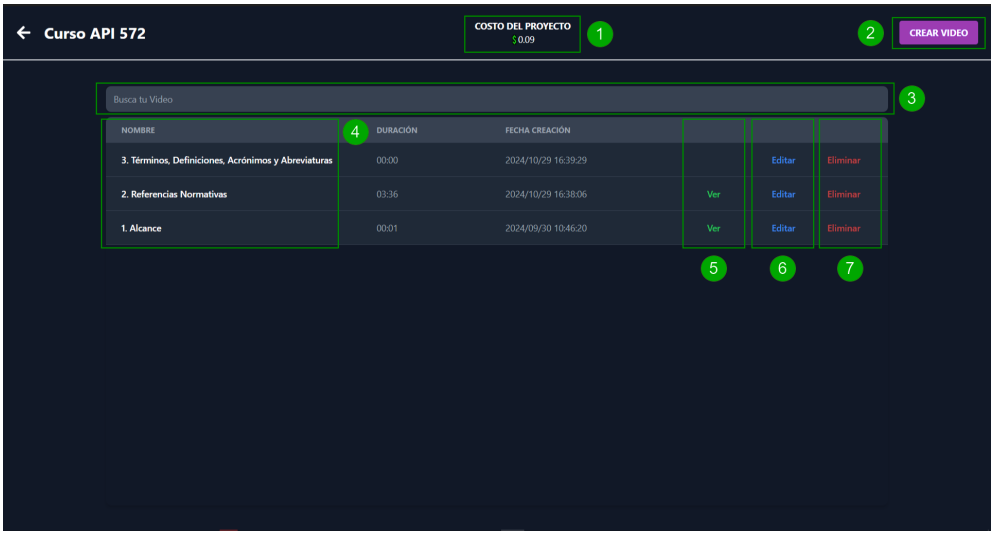
- **Eliminar un Reemplazo**

    1. Ubicar la fila correspondiente al reemplazo que se desea eliminar.
    2. Hacer clic en el botón **Eliminar** en la columna de **Texto de Reemplazo**.
    3. Aparecerá una advertencia de confirmación para verificar que el usuario realmente desea eliminar el reemplazo.
    4. Confirmar la acción para eliminar el reemplazo permanentemente.

4 Pantalla de Proyecto

La pantalla de Proyecto permite al usuario gestionar los videos asociados a un proyecto específico, incluyendo la creación, visualización, edición y eliminación de videos. Además, se muestra el costo total del proyecto, que representa el costo acumulado de todos los videos asociados.

4.1 Estructura de la Pantalla de Proyecto



1. **Costo del Proyecto**

   - Ubicado en el centro superior, muestra el costo total del proyecto, que corresponde a la suma de los costos de todos los videos asociados a este proyecto.

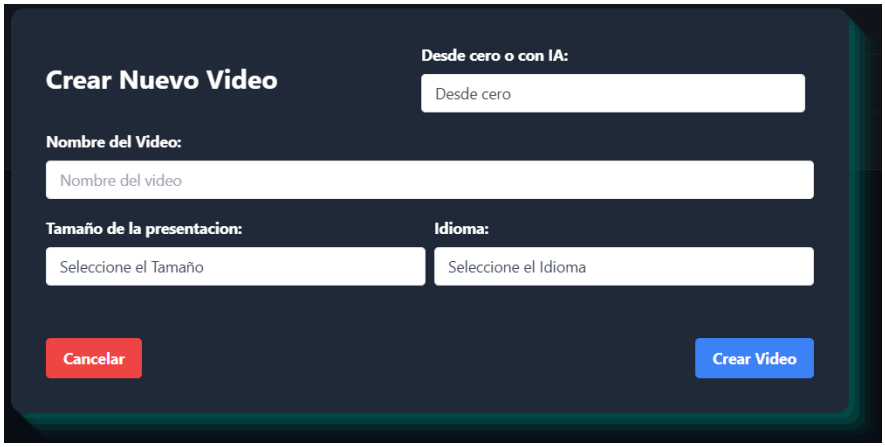
2. **Campo de Búsqueda**

   - Campo de entrada ubicado justo debajo del costo del proyecto, que permite al usuario buscar videos específicos por nombre dentro de la lista.

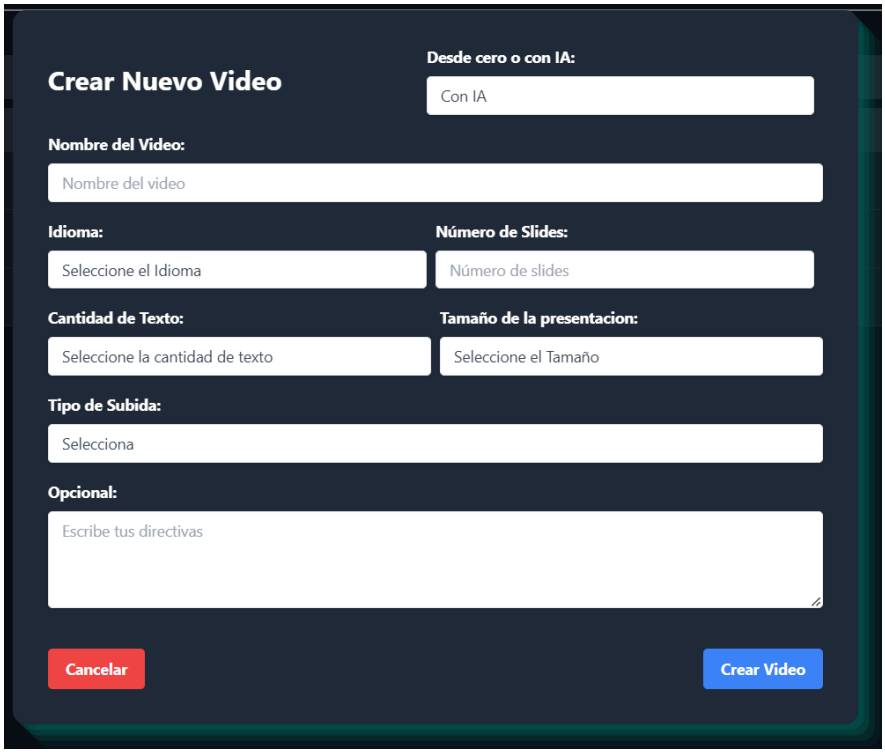
3. **Botón Crear Video**

   - Ubicado en la parte superior derecha, permite al usuario crear un nuevo video dentro del proyecto.
   - Al hacer clic en este botón, se abre un formulario que incluye un campo **Select** para escoger entre dos tipos de creación:

     (a) **Desde Cero**: Permite al usuario crear un video sin asistencia de inteligencia artificial.



(b) **Con IA**: Proporciona asistencia basada en inteligencia artificial para generar el video de acuerdo a las especificaciones dadas.

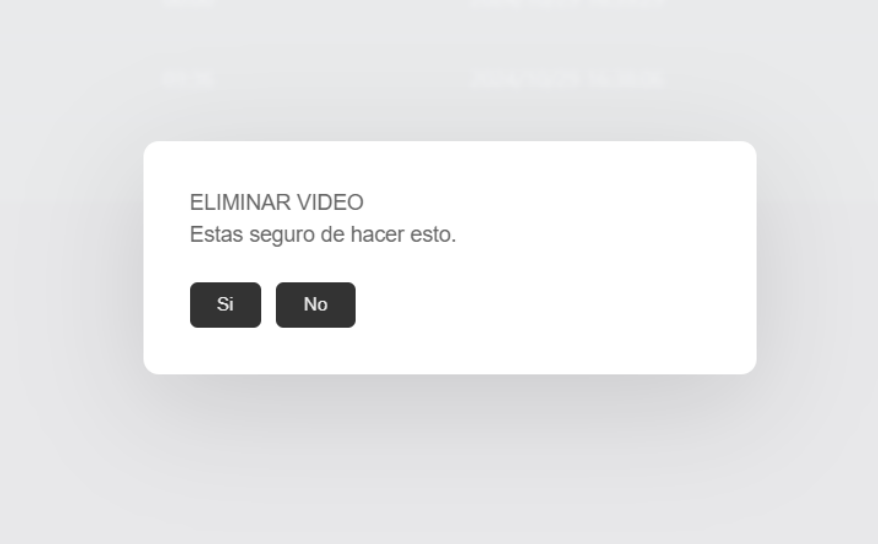


- Los campos específicos para cada tipo de creación son los siguientes:
  - **Desde Cero**:
    - **Nombre del Video**: Campo para introducir el nombre del video.
    - **Tamaño de Presentación**: Selector que permite escoger entre las dimensiones de presentación, con opciones:
      - **9:16** (vertical, ideal para redes sociales con videos verticales)
      - **16:9** (horizontal, adecuado para contenido de estilo YouTube)
      - **1:1** (cuadrado, ideal para posts de redes sociales)

        - **Idioma**: Selector que permite elegir el idioma del video.
    - **Con IA**:
        - **Nombre del Video**: Campo para introducir el nombre del video.
        - **Idioma**: Selector para elegir el idioma del video.
        - **Número de Slides**: Campo para especificar la cantidad de diapositivas que se generarán en el video.
        - **Cantidad de Texto**: Selector para elegir la cantidad de texto en el video, con opciones como *Poco*, *Medio* y *Alto*.
        - **Tamaño de Presentación**: Selector que permite escoger entre las dimensiones de presentación:
            - **9:16** (vertical, ideal para redes sociales con videos verticales)
            - **16:9** (horizontal, adecuado para contenido de estilo YouTube)
            - **1:1** (cuadrado, ideal para posts de redes sociales)
        - **Tipo de entrada**: Selector con opciones *Idea* y *Párrafo*, indicando el formato de entrada en base el cual se creará el contenido.
        - **Directivas**: Campo para incluir directivas o instrucciones específicas para la generación del video.

4. **Columna Nombre (Editable)**
    - La columna **Nombre** contiene campos de entrada (input) que permiten al usuario editar directamente el nombre de cada video.
    - Después de modificar el nombre, el usuario debe presionar la tecla **Enter** para que el cambio se guarde correctamente.
5. **Botón Ver**
    - En cada fila de la lista, el botón **Ver** (en color verde) permite al usuario ver los detalles o la reproducción del video seleccionado.
6. **Botón Editar**
    - El botón **Editar** (en color azul) permite realizar modificaciones adicionales en el video seleccionado.
7. **Botón Eliminar**
    - En cada fila, el botón **Eliminar** (en color rojo) permite al usuario eliminar el video correspondiente de la lista.

- Al hacer clic en el botón de **Eliminar**, se abrirá una advertencia de confirmación para asegurarse de que el usuario desea eliminar el video.



4.2 Funcionalidades Principales

- **Crear un Nuevo Video**

  1. Hacer clic en el botón **Crear Video**.
  2. Se abrirá un formulario que contiene un campo **Select** para seleccionar el tipo de creación:
     - **Desde Cero**: El formulario presentará los campos **Nombre del Video**, **Tamaño de Presentación** y **Idioma**.
     - **Con IA**: El formulario presentará los campos **Nombre del Video**, **Idioma**, **Número de Slides**, **Cantidad de Texto**, **Tamaño de Presentación**, **Tipo de Entrada** y **Directivas**.
  3. Completar el formulario con los detalles necesarios del video y guardar la información para añadir el video a la lista del proyecto.

- **Editar el Nombre de un Video**

  1. Localizar el campo editable en la columna **Nombre** correspondiente al video deseado.
  2. Hacer clic en el campo de texto y modificar el nombre según sea necesario.
  3. Presionar la tecla **Enter** para que el cambio se guarde en el sistema.

- **Ver un Video**

1. Hacer clic en el botón **Ver** en la fila correspondiente al video que se desea visualizar.

- **Editar un Video**

1. Hacer clic en el botón **Editar** en la fila correspondiente al video que se desea modificar.
2. Realizar las modificaciones necesarias en la ventana de edición.

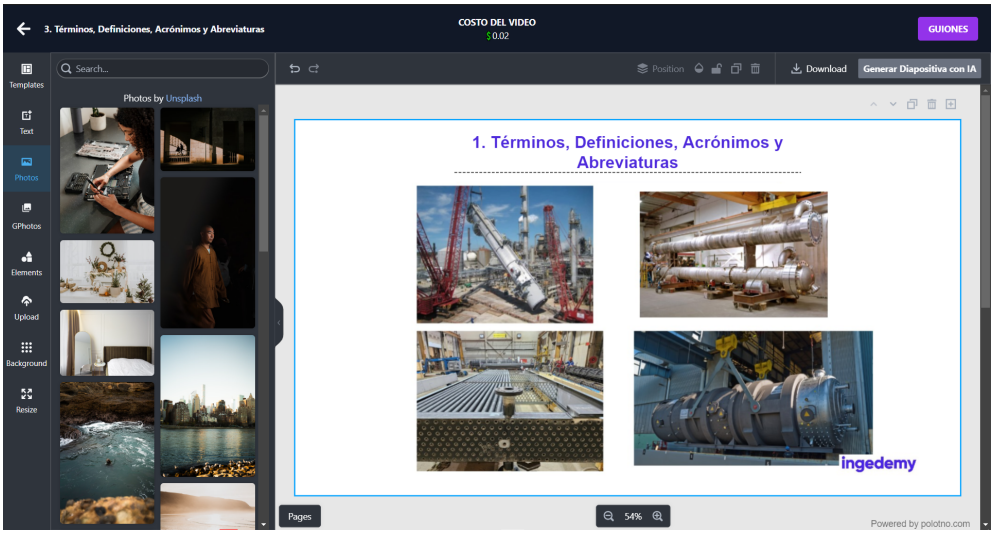
- **Eliminar un Video**

1. Ubicar la fila correspondiente al video que se desea eliminar.
2. Hacer clic en el botón **Eliminar**.
3. Aparecerá una advertencia de confirmación para verificar que el usuario realmente desea eliminar el video.
4. Confirmar la acción para eliminar el video permanentemente.

5 Pantalla de Diseño de Diapositivas

Esta pantalla permite al usuario crear y personalizar diapositivas para su proyecto de manera intuitiva. Ofrece herramientas de diseño, incluyendo opciones para agregar texto, imágenes, elementos gráficos, y personalizar el fondo y el tamaño de la diapositiva. Además, cuenta con una funcionalidad de inteligencia artificial para generar diapositivas automáticamente a partir de directrices específicas. El usuario puede descargar la diapositiva en formatos PDF o PNG, y organizar los elementos visuales a su gusto, facilitando la creación de presentaciones atractivas y profesionales.

5.1 Estructura de la Pantalla de Diseño de Diapositivas



1. **Barra Lateral (Sidebar)**

   - **Templates**: Permite seleccionar plantillas predefinidas que se aplicarán a la diapositiva actual.
   - **Text**: Opción para agregar cuadros de texto personalizables. Al seleccionar esta opción, se añade un cuadro de texto al área de trabajo.
   - **Photos**: Muestra una colección de imágenes proporcionadas por Unsplash, que se pueden arrastrar y soltar en la diapositiva.
   - **GPhotos**: Permite acceder a fotos desde Google Photos del usuario para incluir en la diapositiva.
   - **Elements**: Incluye formas, iconos y otros elementos gráficos personalizables en tamaño, color y posición.
   - **Upload**: Permite al usuario subir imágenes y otros archivos personalizados desde su dispositivo.
   - **Background**: Permite seleccionar colores, gradientes o imágenes de fondo para la diapositiva.
   - **Resize**: Opción para ajustar el tamaño de la diapositiva en diferentes proporciones, como 16:9, 9:16 y 1:1.

2. **Área de Trabajo (Diapositiva)**

   - **Espacio central donde se diseña y edita la diapositiva**: El usuario puede agregar y posicionar elementos desde la barra lateral.

- **Redimensionar y personalizar elementos**: Los elementos pueden ajustarse en tamaño, color y estilo mediante los controladores y opciones de la barra de herramientas superior.

3. **Barra de Herramientas Superior (Navbar)**

   - **Fuente y Tamaño de Texto**: Permite seleccionar la fuente y el tamaño del texto en la diapositiva.
   - **Opciones de Formato**: Incluye opciones como negrita, cursiva, subrayado y color de texto.
   - **Posición**: Herramienta para alinear y organizar la posición de los elementos en la diapositiva.
   - **Eliminar**: Borra el elemento seleccionado de la diapositiva.
   - **Descargar**: Ofrece la opción de descargar la diapositiva actual en formato **PDF** o **PNG**.

4. **Botón "Generar Diapositiva con IA"**

   - Ubicado en la esquina superior derecha, este botón permite acceder a la funcionalidad de generación automática de diapositivas mediante inteligencia artificial.
   - Al hacer clic, se abre un formulario para especificar el **Tema de la Diapositiva**, el **Cantidad de texto** deseado y **Directivas** para la generación automática.



5.2 Funcionalidades Principales

- **Añadir Elementos a la Diapositiva**
  1. Seleccionar un elemento en la **Barra Lateral** (por ejemplo, texto, imagen, o forma).
  2. Arrastrar y soltar el elemento en el área de trabajo para posicionarlo en la diapositiva.
- **Personalizar Estilos de los Elementos**
  1. Seleccionar un elemento en el área de trabajo.
  2. Usar la **Barra de Herramientas Superior** para cambiar el estilo (fuente, color, tamaño, etc.) del elemento seleccionado.
- **Descargar la Diapositiva**
  1. Hacer clic en la opción de **Descargar** en la Barra de Herramientas Superior.
  2. Seleccionar el formato deseado (**PDF** o **PNG**) para guardar la diapositiva en el dispositivo.

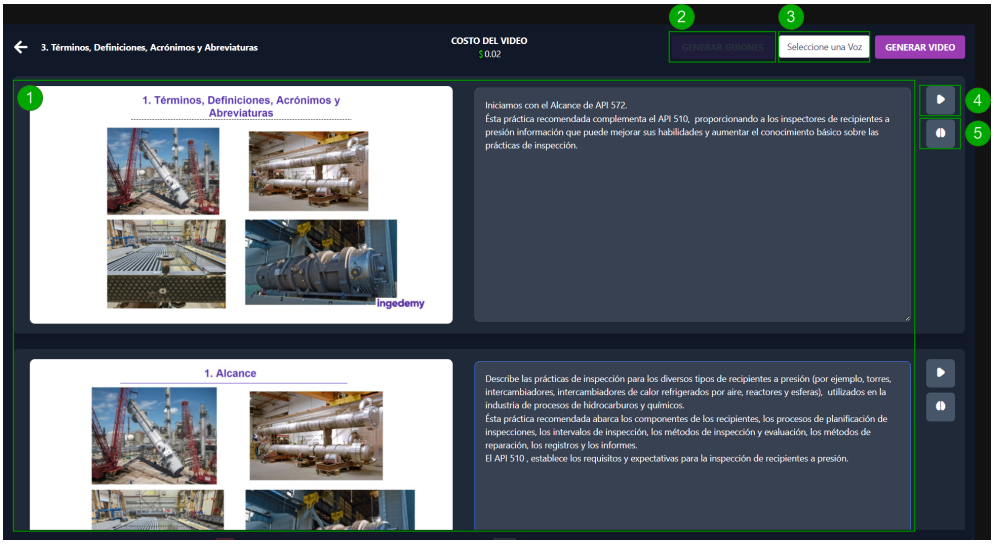
- **Generar Diapositiva Automáticamente con IA**

    1. Hacer clic en el botón **Generar Diapositiva con IA** en la esquina superior derecha.
    2. Completar el formulario con el **Tema de la Diapositiva**, **Cantidad de texto**, y **Directivas**.
    3. Confirmar para que la inteligencia artificial genere una diapositiva según las especificaciones dadas.

6 Pantalla de Guiones

La pantalla de Guiones permite al usuario gestionar el contenido narrativo asociado a cada diapositiva del proyecto. Aquí, el usuario puede escribir o editar los guiones que acompañan a las diapositivas, escuchar una vista previa del guion utilizando voces seleccionables, y realizar modificaciones automáticas o personalizadas en el texto. Esta sección facilita la creación y edición de guiones para videos de presentación de manera intuitiva y eficiente.

6.1 Estructura de la Pantalla de Guiones



1. **Área de Visualización de Diapositivas y Guiones**

    - Cada diapositiva está acompañada de un cuadro de texto donde el usuario puede ingresar o modificar el guion correspondiente.

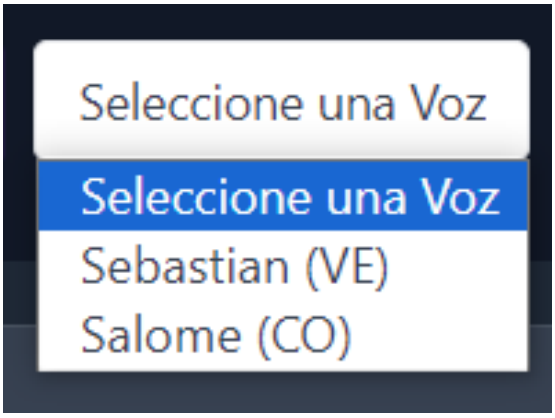
- Si el guion de una diapositiva está vacío, el botón **Generar Guiones** se habilita, permitiendo generar contenido automáticamente para los guiones vacíos o para todos los guiones, según se requiera.

2. **Botón "Generar Guiones"**

   - Este botón permite generar automáticamente todos los guiones o solo aquellos que estén vacíos en la pantalla actual.

3. **Selector de Voz**

   - Menú desplegable que permite seleccionar la voz que se usará para la vista previa del guion.

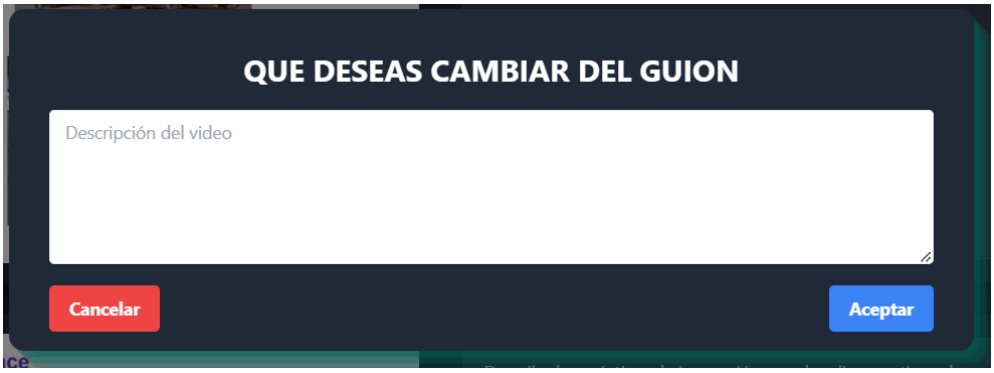


4. **Icono "Play" para Escuchar el Guion**

   - Icono de reproducción ("Play") ubicado junto al cuadro de texto de cada guion.
   - Permite escuchar una vista previa del guion utilizando la voz seleccionada.

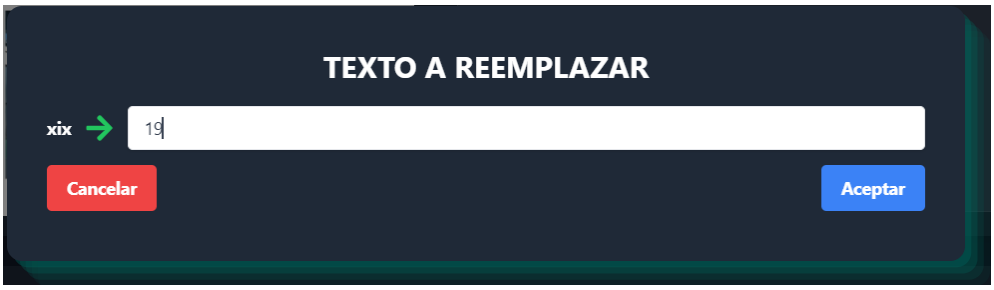
5. **Icono de Reescribir Guion**

   - Icono de reescritura ("cerebro") disponible junto al cuadro de texto del guion.
   - Al hacer clic, se abrira un formulario donde el usuario podra escribir un feedback para que se reescriba el guion utilizando IA, ofreciendo una nueva versión del texto.



6. **Formulario de Reemplazo de Palabras**

   - Al hacer doble clic en una palabra dentro del cuadro de texto del guion, se abre un formulario de reemplazo.



   - Este formulario permite seleccionar y reemplazar rápidamente la palabra elegida por sinónimos o términos alternativos, facilitando la edición precisa del guion.

6.2 Funcionalidades Principales

- **Generar Guiones**

  1. Hacer clic en el botón **Generar Guiones**.
  2. Esto generará automáticamente todos los guiones vacíos o, si así lo desea el usuario, reemplazará los guiones actuales.

- **Escuchar Vista Previa del Guion**

  1. Seleccionar una voz en el **Selector de Voz**.
  2. Hacer clic en el icono de **Play** junto al guion deseado para escuchar una vista previa del guion.

- **Reescribir el Guion**

1. Hacer clic en el icono de **Reescribir** (“cerebro”) para obtener una nueva versión del guion actual.

- **Reemplazar Palabras en el Guion**

1. Hacer doble clic en la palabra que desea reemplazar dentro del guion.
2. En el formulario de reemplazo, seleccionar la nueva palabra para actualizar el guion.

7 Pantalla de Visualización del Video Final

Esta pantalla permite al usuario reproducir y revisar el video final generado a partir de las diapositivas y los guiones creados previamente. Brinda un espacio limpio y enfocado para garantizar que el contenido visual y narrativo se ajuste a las expectativas antes de su descarga o distribución.

7.1 Estructura de la Pantalla de Visualización del Video Final



1. **Barra Superior**

- **Botón “Regresar”**: Permite al usuario volver a la pantalla de edición de guiones o diapositivas, en caso de necesitar realizar cambios.

2. **Área de Reproducción de Video**

- Una sección central que muestra el reproductor de video, donde el usuario puede visualizar el contenido final.

- Incluye controles estándar de reproducción: **play**, **pausa**, y barra de progreso para navegar por el video.

7.2 Funcionalidades Principales

- **Reproducir el Video**

  1. Hacer clic en el botón **Play** en el reproductor para comenzar a visualizar el video final.
  2. Utilizar la barra de progreso para avanzar o retroceder en el video.

- **Regresar a la Edición**

  1. Hacer clic en el botón **Regresar**, ubicado en la esquina superior izquierda, para volver a la pantalla de edición de guiones o diapositivas.