



UNIVERSIDAD LAICA “ELOY ALFARO” DE MANABÍ
EXTENSIÓN EL CARMEN
CARRERA EN INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN.

Creada Ley No. 10 – Registro Oficial 313 de noviembre 13 de 1985

PROYECTO INTEGRADOR
PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
TECNOLOGÍAS DE LA INFORMACIÓN

SISTEMA INFORMÁTICO PARA LA GESTIÓN DE BÚSQUEDA DE
CONTENIDOS AUDIOVISUALES EN SOFTMEDIA ULEAM EL CARMEN.

AUTOR:

SUMI ARÉVALO JAZMANY DANIEL

TUTOR:

ING. ARCA ZAVALA JEFFERSON OMAR

EL CARMEN, AGOSTO 2026

CERTIFICACIÓN DEL DIRECTOR DE TRABAJO DE GRADUACIÓN

	NOMBRE DEL DOCUMENTO: CERTIFICADO DE TUTOR(A).	CÓDIGO: PAT-04-F-004
	PROCEDIMIENTO: TITULACIÓN DE ESTUDIANTES DE GRADO BAJO LA UNIDAD DE INTEGRACIÓN CURRICULAR	REVISIÓN: 1 Página 1 de 1

CERTIFICACIÓN

En calidad de docente tutor de la Extensión El Carmen de la Universidad Laica "Eloy Alfaro" de Manabí, CERTIFICO:

Haber dirigido, revisado y aprobado preliminarmente el Trabajo de Integración Curricular bajo la autoría del estudiante **SUMI ARÉVALO JAZMANY DANIEL**, legalmente matriculado en la carrera de Ingeniería en Software, período académico 2026-1, cumpliendo el total de 384 horas, cuyo tema del proyecto es **"SISTEMA INFORMÁTICO PARA LA GESTIÓN DE BÚSQUEDA DE CONTENIDOS AUDIOVISUALES EN SOFTMEDIA ULEAM EL CARMEN"**.

La presente investigación ha sido desarrollada en apego al cumplimiento de los requisitos académicos exigidos por el Reglamento de Régimen Académico y en concordancia con los lineamientos internos de la opción de titulación en mención, reuniendo y cumpliendo con los méritos académicos, científicos y formales, y la originalidad del mismo, requisitos suficientes para ser sometida a la evaluación del tribunal de titulación que designe la autoridad competente.

Particular que certifico para los fines consiguientes, salvo disposición de Ley en contrario.

El Carmen, 29 de julio de 2026

Lo certifico,



Ing. Jefferson Omar Arca Zavala, Mgs.
Docente Tutor
Área: Ingeniería en Software

TRIBUNAL DE SUSTENTACIÓN



TRIBUNAL DE SUSTENTACIÓN

Título del trabajo de titulación:

Sistema Informático para la Gestión de Búsqueda de Contenidos Audiovisuales en Softmedia Uleam El Carmen.

Modalidad:

Proyecto Integrador

Autor:

Sumi Arevalo Jazmany Daniel

Tutor:

Ing. Arca Zavala Jefferson Omar, Mgtr

Tribunal de Sustentación

- Presidente A. S. Minaya Macias Renelmo Wladimir, Mgtr.
- Miembro Ing. Pozo Hernández Clara Guadalupe, Mgtr.
- Miembro Ing. Angel Wilson Villarreal Cobeña, Mgtr.

Fecha de sustentación:

20 agosto 2026

DECLARACIÓN EXPRESA DE AUTORÍA

**UNIVERSIDAD LAICA “ELOY ALFARO” DE MANABÍ
EXTENSIÓN EL CARMEN**



DECLARACIÓN DE AUTORÍA

La responsabilidad de todo el contenido de este trabajo de titulación, con el tema:
**SISTEMA INFORMÁTICO PARA LA GESTIÓN DE BÚSQUEDA DE
CONTENIDOS AUDIOVISUALES EN SOFTMEDIA ULEAM EL CARMEN,**
corresponde exclusivamente a: **Sumi Arévalo Jazmany Daniel** con CI: **2300913213** y
los derechos patrimoniales de la misma corresponden a la Universidad Laica “Eloy
Alfaro” de Manabí.



Sumi Arévalo Jazmany Daniel

C.I. 2300913213

DEDICATORIA

Dedico este trabajo, antes que nada, a Dios, porque sin Él no habría llegado hasta aquí. En los momentos en que sentí que ya no podía más, fue mi fuerza; cuando tuve dudas, fue mi refugio, y cuando el camino se hizo difícil, fue quien me sostuvo para seguir adelante. Cada desvelo, cada dificultad y cada logro los viví confiando en que su mano estaba conmigo. Hoy puedo mirar atrás y reconocer que no caminé solo; Dios estuvo presente en cada paso y, por eso, este logro le pertenece primero a Él.

A mi madre, Jenny Arévalo, por todo el esfuerzo y sacrificio que ha hecho por mí a lo largo de mi vida, por estar siempre presente y por ayudarme en cada momento en que lo he necesitado. Gracias por nunca dejar que me faltara nada, por cada mensaje de aliento, por cada muestra de amor y por acompañarme tanto en los buenos momentos como en los más difíciles. Este logro también es tuyo, porque detrás de cada paso que he dado siempre has estado tú.

“Mira que te mando que te esfuerces y seas valiente; no temas ni desmayes, porque Jehová tu Dios estará contigo en dondequiera que vayas.” *Josué 1:9*

Este trabajo no representa únicamente un logro académico, sino el resultado de noches de esfuerzo, sacrificios, momentos difíciles y, sobre todo, de la fe que me permitió seguir adelante cuando las cosas no fueron fáciles. Hoy, al mirar todo el camino recorrido, entiendo que ningún sueño es imposible cuando se pone en manos de Dios, se trabaja con dedicación y nunca se pierde la esperanza. Este logro es una prueba de que, con fe en Dios, esfuerzo y perseverancia, aquello que un día parecía lejano puede convertirse en realidad.

Jazmany Sumi Arévalo

AGRADECIMIENTO

No tengo palabras exactas para describir la inmensa felicidad que siento al culminar esta etapa. A veces pareciera que graduarse es un paso más, pero solo uno conoce el trabajo, el sacrificio y la perspectiva que hay detrás de cada logro. Hoy quiero dedicar estas palabras de profunda gratitud a quienes me acompañaron en este camino.

En primer lugar, quiero agradecer a mi Dios, quien fue, es y seguirá siendo mi refugio, mi ayuda y mi protector. Gracias a Él tuve la oportunidad de estudiar esta carrera. Cuando le pedí sabiduría, me respondió y nunca me dejó solo; estuvo conmigo en cada examen, en cada exposición, en los días duros de desánimo y en los de angustia. Por poner su mirada en mí, este triunfo es para Él.

Mi sincero reconocimiento y todo mi amor a la persona más importante de mi vida: mi madre, Jenny Arévalo. Gracias por no abandonarme cuando la vida se puso gris y por estar conmigo desde mi primer respiro. Admiro el valor que tuviste para sacar adelante nuestro hogar tú sola, trabajando tan duro para darme lo que necesitaba, sin rendirte incluso cuando tu cuerpo no daba más. Gracias por enseñarme las cosas buenas, por instruirme en la palabra de Dios y por corregirme siempre con ese amor inmenso que vi en tus ojos. Te admiro y te respeto profundamente.

A mis abuelitos, pilares fundamentales en mi vida y en mi crianza. A mi abuelita, aunque ya no esté físicamente conmigo, gracias por dejar en mí tantos recuerdos, enseñanzas y amor. Y a mi abuelito, gracias por su cariño, sus consejos y por seguir acompañándome. Este logro también es de ustedes.

De igual manera, agradezco de corazón a mi padrastro, Yuvín Chavarria, por su presencia, apoyo y por ser parte importante de este camino. Gracias por ayudarme a llegar hasta aquí y contribuir a que este sueño se hiciera realidad.

A mis demás familiares, gracias por su cariño sincero, sus oraciones y por estar siempre pendientes de mí.

A mis amigos que conocí en este proceso universitario. Gracias por acompañarme semestre a semestre, por las largas jornadas de estudio, la camaradería y por impulsarme a dar siempre lo mejor de mí.

Finalmente, agradezco a mi tutor de tesis, el Ing. Jefferson Arca, por su guía, paciencia y apoyo durante el desarrollo de este proyecto.

A todos ustedes, gracias por confiar y creer en mí. Este triunfo también es suyo.

Jazmany Sumi Arévalo

ÍNDICE DE CONTENIDOS

Contenido

Página de respeto	¡Error! Marcador no definido.
Certificación del director de trabajo de graduación	III
Tribunal de sustentación.....	IV
Declaración expresa de autoría	V
Dedicatoria	VI
Agradecimiento	VII
Índice de contenidos	VIII
Índice tablas	XII
Índice gráficos e ilustraciones.....	XIV
Índice de anexos.....	XVI
Resumen	XVII
Abstract	XVIII
Capítulo I.....	1
1 INTRODUCCIÓN.....	1
1.1 Introducción	1
1.2 Presentación del tema	3
1.3 Ubicación y contextualización de la problemática	3
1.4 Planteamiento del problema.....	4
1.4.1 Problematicación	4
1.4.2 Génesis del problema.....	4
1.4.3 Estado actual del problema	5
1.5 Diagrama causa – efecto del problema	6
1.6 Objetivos	6
1.6.1 Objetivo general.....	6
1.6.2 Objetivos específicos	6
1.7 Justificación	7
1.8 Impactos esperados	7
1.8.1 Impacto tecnológico.....	7

1.8.2	Impacto social	8
1.8.3	Impacto ecológico	8
Capítulo II		9
2	MARCO TEÓRICO DE LA INVESTIGACIÓN	9
2.1	Antecedentes históricos	9
2.2	Antecedentes de Investigaciones Relacionadas al Tema Presentado	9
2.3	Definiciones Conceptuales	10
2.3.1	Variable Independiente: Sistemas Informáticos	10
2.3.2	Variable Dependiente: Gestión de búsqueda.	17
2.3.3	Metodología de Desarrollo de Software	23
2.3.4	Conclusiones del marco teórico	25
Capítulo III		27
3	MARCO INVESTIGATIVO (DISEÑO METODOLÓGICO)	27
3.1	Introducción	27
3.2	Tipo de Investigación.....	27
3.2.1	Investigación mixta.....	27
3.2.2	Investigación descriptiva	28
3.2.3	Investigación de campo	28
3.2.4	Enfoque cuantitativo	29
3.2.5	Enfoque cualitativo	29
3.3	Métodos de investigación	29
3.3.1	Método inductivo	29
3.3.2	Método deductivo	30
3.4	Fuentes e instrumentos de recolección de datos	30
3.4.1	Encuesta	30
3.5	Estrategia operacional para la recolección de datos	31
3.5.1	Población	31
3.5.2	Técnica de muestreo	31

3.5.3	Tamaño de la muestra	32
3.5.4	Análisis de las herramientas de recolección de datos a utilizar	32
3.5.5	Plan de recolección de datos	34
3.6	Análisis y presentación de resultados	34
3.6.1	Tabulación y análisis de los datos.....	34
3.6.2	Presentación y descripción de los resultados obtenidos	37
3.6.3	Informe final del análisis de los datos	38
Capítulo IV		40
4	MARCO PROPOSITIVO	40
4.1	Introducción	40
4.2	Descripción de la propuesta	40
4.3	Determinación de recursos.....	42
4.3.1	Humanos	42
4.3.2	Tecnológicos	43
4.3.3	Recursos Económicos	44
4.4	Desarrollo bajo la metodología Scrum	44
4.4.1	Descripción del Producto	45
4.4.2	Historias de Usuario.....	48
4.4.3	Diseño del Sistema / Descripción Técnica.....	52
4.4.4	Diagramas de Secuencia	59
4.4.5	Diagramas de Estado	65
4.4.6	Descripción Técnica / Arquitectura del Sistema.....	66
4.4.7	Requerimientos funcionales (RF)	69
4.4.8	Requerimientos No Funcionales (RNF):	72
4.4.9	Roles y Responsabilidades.....	74
4.4.10	Planificación del Sprint.....	75
4.4.11	Backlog del Producto	80
4.5	Interfaz de Usuario (UI) / Prototipos:	82

4.5.1	Pantallas del Sistema:	83
4.5.2	Definición de Hecho (DoD).....	87
4.5.3	Eventos Scrum	90
4.6	Proceso de Pruebas	94
4.6.1	Pruebas de caja negra.....	94
4.6.2	Pruebas de caja blanca	95
4.6.3	Incremento y Entregables	97
Capítulo V		101
5	EVALUACIÓN DE RESULTADOS.....	101
5.1	Introducción	101
5.2	Presentación y monitoreo de resultados.....	101
5.2.1	Planificación de la Evaluación.....	101
5.2.2	Ejecución del Monitoreo.....	103
5.2.3	Pruebas de compatibilidad entre navegadores	109
5.2.4	Cumplimiento de los Sprints de desarrollo.....	110
5.3	Interpretación objetiva	111
Capítulo VI		112
6	CONCLUSIONES Y RECOMENDACIONES	112
6.1	Conclusiones	112
6.2	Recomendaciones	114
7	REFERENCIAS	116
8	ANEXOS	119

----		124

-----		124
9	GLOSARIO	125

ÍNDICE TABLAS

Tabla 1: Plan de recopilación de datos.....	34
Tabla 2: Tabulación y análisis de datos de encuesta.....	34
Tabla 3: Determinación de recursos humanos	42
Tabla 4: Determinación de recursos Tecnológicos Hardware	43
Tabla 5: Determinación de recursos económicos	44
Tabla 6: Usuarios objetivo	47
Tabla 7: Historia de usuario 1: Autenticación y Gestión de Usuario	48
Tabla 8: Historia de usuario - Carga y Etiquetado Automático	49
Tabla 9: Historia de usuario - Auditoría y Aprobación.....	50
Tabla 10: Historia de usuario - Gestión y Depuración Total	50
Tabla 11: Historia de usuario - Búsqueda y Visualización Pública	51
Tabla 12: Caso de uso: Ingresar al Módulo Audiovisual.....	52
Tabla 13: Caso de uso: Subir contenido audiovisual	53
Tabla 14: Caso de uso: Auditar y Aprobar Contenido	55
Tabla 15: Caso de uso: Buscar contenido audiovisual	56
Tabla 16: Caso de uso: Gestionar y depurar catálogo	57
Tabla 17: Requerimientos funcionales (RF)	69
Tabla 18: Requerimientos No Funcionales (RNF):	72
Tabla 19: Roles y Responsabilidades	74
Tabla 20: Arquitectura Base, Base de Datos y Backend API REST	75
Tabla 21: Sprint 2: Autenticación JWT, Sistema de roles y control de acceso	77
Tabla 22: Sprint 3: Frontend Web — Interfaz de Catálogo y Carga Masiva	78
Tabla 23: Sprint 4: Gestión Avanzada, Operaciones Masivas y Vista de Proyectos	79
Tabla 24: Backlog inicial.....	80
Tabla 25: Fin del Sprint 1	80
Tabla 26: Fin del Sprint 2	81
Tabla 27: Fin del Sprint 3	81
Tabla 28: Fin del Sprint 4	81
Tabla 29: Pruebas de carga masiva	94

Tabla 30: Prueba de galería y visor lightbox	94
Tabla 31: Prueba de confirmación de borrado.....	95
Tabla 32: Pruebas de seguridad y control de acceso	95
Tabla 33: Pruebas de procesamiento de IA y carga masiva	96
Tabla 34: Prueba de Gestión de estados y borrado	97
Tabla 35: Incremento y entregables: Backend, base de datos e integración de IA.....	97
Tabla 36: Incremento y entregables: Autenticación y roles	98
Tabla 37: Incremento y entregables: frontend web, catálogo y carga.....	99
Tabla 38: Incremento y Entregables: Gestión Avanzada y Proyectos	99
Tabla 39: Planificación de la Evaluación del Módulo de Catálogo Audiovisual	101
Tabla 40: Ejecución del Monitoreo — Validación de Accesos y Permisos por Rol	103
Tabla 41: Resultados de integración y permisos por rol.....	103
Tabla 42: Ejecución del Monitoreo — Carga masiva y etiquetado con IA.....	104
Tabla 43: Resultados del testeo de carga masiva y etiquetado con IA	104
Tabla 44: Ejecución del Monitoreo — Flujo de estados del contenido	105
Tabla 45: Matriz de permisos verificados por rol (solo contenido no publicado).....	106
Tabla 46: Ejecución del Monitoreo — Búsqueda predictiva y filtrado.....	106
Tabla 47: Resultados del testeo de descarga individual y masiva por navegador	107
Tabla 48: Ejecución del Monitoreo — Eliminación segura con confirmación	108
Tabla 49: Resultados del etiquetado automático con IA BLIP	109
Tabla 50: Resultados de las pruebas de compatibilidad entre navegadores (FSAA = File System Access API)	110
Tabla 51: Cumplimiento de los Sprints de desarrollo.....	110

ÍNDICE GRÁFICOS E ILUSTRACIONES

Ilustración 1: Diagrama causa-efecto del problema.....	6
Ilustración 2: Encuesta aplicada	33
Ilustración 3: Diagrama de Casos de Uso: Ingresar al Módulo Audiovisual	52
Ilustración 4: Caso de uso: Subir contenido audiovisual.....	53
Ilustración 5: Caso de uso: Auditar y Aprobar Contenido	55
Ilustración 6: Caso de uso: Buscar contenido audiovisual	56
Ilustración 7: Caso de uso: Gestionar y depurar catálogo	57
Ilustración 8: Diagrama de Base de datos	58
Ilustración 9: Diagrama de secuencia: Ingresar al módulo.....	59
Ilustración 10: Diagrama de secuencia: Carga masiva de archivos	60
Ilustración 11: Diagrama de secuencia: Procesamiento de IA (BLIP).....	61
Ilustración 12: Diagrama de secuencia: Marcar contenido como Listo (rol integrante)	62
Ilustración 13: Diagrama de secuencia: Aprobar contenido (rol supervisor)	62
Ilustración 14: Diagrama de secuencia: Ocultar contenido	63
Ilustración 15: Diagrama de secuencia: Eliminación física	63
Ilustración 16: Diagrama de secuencia: Descarga de archivos	64
Ilustración 17: Diagrama de secuencia: Búsqueda y filtrado.....	64
Ilustración 18: Diagrama de secuencia: Vista de proyectos	65
Ilustración 19: Diagrama de estado: Ciclo de vida del contenido.....	65
Ilustración 20: Mapa del sistema	67
Ilustración 21: Mapa de navegación del módulo Web	82
Ilustración 22: Pantalla explorar galería mansory.....	83
Ilustración 23: Pantalla de Búsqueda Predictiva y Filtros	83
Ilustración 24: Pantalla de Proyectos (Agrupación por Evento)	84
Ilustración 25: Pantalla de Vista del Proyecto.....	85
Ilustración 26: Pantalla de Nueva Publicación (Drag & Drop)	85
Ilustración 27: Pantalla de Visor Lightbox (Pantalla Completa).....	86
Ilustración 28: Pantalla de Selección Múltiple y Acciones Masivas	87
Ilustración 29: Pantallas de Notificaciones (Toast) y Confirmaciones	87

Ilustración 30: Código de carga masiva	90
Ilustración 31: Código de procesamiento de IA.....	90
Ilustración 32: Código de autenticación JWT	92
Ilustración 33: Código de borrado físico y regla de retención	93

ÍNDICE DE ANEXOS

Anexo A: Asignación de tutor:	119
Anexo B: Reporte del sistema antiplagio.....	120
Anexo C: Evidencia aplicación de encuestas.....	122
Anexo D: Formato del material de recolección de datos	124

RESUMEN

El Club de Audiovisuales SoftMedia de la Universidad Laica "Eloy Alfaro" de Manabí (ULEAM), Extensión El Carmen, no contaba con un sistema para gestionar su archivo audiovisual. Las fotografías y videos se almacenaban en carpetas de servicios en la nube sin nomenclatura uniforme ni criterios de clasificación, y parte del material se perdió de forma permanente por limitaciones de espacio. Frente a esta situación, el presente trabajo desarrolló una plataforma de catalogación digital con etiquetado automático de archivos mediante modelos de visión computacional.

Para el desarrollo de la investigación se empleó un enfoque mixto de alcance descriptivo y propositivo. Se revisó literatura sobre sistemas de catalogación digital, el estándar MARC21 y redes neuronales aplicadas al análisis de imágenes, y se aplicaron encuestas a 158 miembros de la comunidad universitaria. Del análisis de los datos se obtuvo que el 73% había tenido dificultades para localizar material multimedia institucional, el 92% señaló la necesidad de una plataforma centralizada y el 96% valoró positivamente el etiquetado automático de archivos.

El sistema se construyó bajo la metodología Scrum en cuatro sprints de dos semanas. En el backend se utilizó FastAPI con Python; para el almacenamiento, PostgreSQL con campos JSONB; en el frontend, React 18 con Vite y Tailwind CSS; y el despliegue se realizó mediante Docker Compose. El módulo de etiquetado emplea el modelo BLIP para generar descripciones a partir del contenido visual, traducidas al español mediante Google Translate y procesadas de forma asíncrona. Los videos se procesan mediante extracción de fotogramas con FFmpeg. Durante las pruebas, el 75% de los archivos obtuvo etiquetas de alta precisión y el 25% restante de precisión media, con tiempos de 2,5 segundos por imagen y 8,4 por video.

El acceso se controla mediante tokens JWT con cuatro roles de usuario. La interfaz incorpora galería Masonry, carga masiva de hasta 50 archivos, búsqueda en tiempo real y visor Lightbox. El trabajo dio como resultado un repositorio centralizado en el que los archivos pueden localizarse mediante búsqueda por etiquetas, y cuya arquitectura permite su implementación en instituciones con necesidades similares.

ABSTRACT

The SoftMedia Audiovisual Club of the Universidad Laica "Eloy Alfaro" de Manabí (ULEAM), El Carmen Extension, lacked a system to manage its audiovisual archive. Photographs and videos were stored in cloud service folders with no uniform naming conventions or classification criteria, and part of the material was permanently lost due to storage limitations. In response to this situation, the present work developed a digital cataloging platform with automatic file tagging using computer vision models.

The research employed a mixed-method approach with descriptive and propositional scope. Literature on digital cataloging systems, the MARC21 standard, and neural networks applied to image analysis was reviewed, and surveys were administered to 158 members of the university community. Data analysis showed that 73% of participants had experienced difficulties locating institutional multimedia material, 92% identified the need for a centralized platform, and 96% viewed automatic file tagging favorably.

The system was built using the Scrum methodology across four two-week sprints. The backend was developed with FastAPI and Python; PostgreSQL with JSONB fields was used for storage; React 18 with Vite and Tailwind CSS for the frontend; and deployment was carried out using Docker Compose. The tagging module uses the BLIP model to generate descriptions from visual content, translated into Spanish via Google Translate and processed asynchronously. Videos are handled through keyframe extraction with FFmpeg. During testing, 75% of files received high-precision tags and the remaining 25% medium-precision tags, with processing times of 2.5 seconds per image and 8.4 seconds per video.

Access is controlled through JWT tokens with four defined user roles. The interface includes a Masonry gallery, bulk upload of up to 50 files, real-time search, and a Lightbox viewer. The work resulted in a centralized repository where files can be located through automatically generated tag-based search, with an architecture suitable for implementation in institutions with similar need

CAPÍTULO I

1 INTRODUCCIÓN

1.1 Introducción

Cada período académico, el Club de Audiovisuales SoftMedia de la ULEAM, extensión El Carmen, produce un volumen considerable de material fotográfico y audiovisual. Coberturas de casas abiertas, grabaciones de conferencias, tomas para proyectos de producción del propio club. Al finalizar cada actividad, esos archivos se depositaban en carpetas de un servicio de almacenamiento en la nube contratado por el club, sin que se aplicara un criterio para nombrarlos, agruparlos por evento o describirlos mediante algún tipo de etiqueta. Semanas más tarde, cuando algún integrante necesitaba retomar una toma concreta ya fuera una fotografía grupal de una ceremonia o un fragmento de video de una entrevista la única opción disponible era revisar carpeta por carpeta, sin guía alguna. En varios casos, esa búsqueda terminaba sin éxito.

El desorden no solo se traducía en pérdida de tiempo. Hubo casos en que miembros del club tuvieron que borrar archivos para recuperar espacio sin tener manera de saber con certeza qué contenían. Material de eventos irrepetibles, tomas que podrían haberse aprovechado después, desapareció así sin dejar rastro. Grabaciones de eventos que no se repiten, tomas que podrían haberse reutilizado en nuevas producciones. La ausencia de un sistema de búsqueda, de roles que regularan quién podía modificar o borrar contenido, y de cualquier mecanismo de clasificación, convirtió al archivo del club en un depósito donde la información se deterioraba con el paso del tiempo en lugar de acumularse como un recurso aprovechable.

Para establecer el alcance de esta problemática se aplicó una encuesta a 158 personas de la comunidad universitaria, y las cifras obtenidas no dejaron margen a la ambigüedad. Los datos obtenidos en la encuesta fueron contundentes: tres de cada cuatro personas (73 %) admitieron haber enfrentado dificultades para localizar contenido multimedia institucional. Un 92 % señaló que era necesario contar con un espacio centralizado donde ese material pudiera organizarse. El dato que terminó de definir el enfoque de este trabajo fue que el 96 % consideró oportuno que esa clasificación se realizara de forma automática mediante inteligencia artificial. La revisión directa del repositorio del club no hizo más que confirmar lo que las cifras ya sugerían: carpetas sin

ningún orden, archivos repetidos y sin posibilidad alguna de saber quién había subido cada cosa ni en qué momento.

A partir de ese diagnóstico, el presente trabajo propone el desarrollo de un módulo web de catálogo audiovisual que emplea inteligencia artificial para abordar el problema desde su origen. En la práctica, esto funciona así: al momento de cargar una imagen, el modelo de Salesforce la revisa y redacta una pequeña frase en inglés detallando lo que ve. El sistema toma esa frase, la pasa al español y extrae de ahí las palabras clave sueltas que servirán como etiquetas. Si lo que se sube es un video, el camino es un poco más largo. Primero se usa FFmpeg para capturar cuadros estáticos a lo largo del clip, y a cada uno de esos cuadros se le aplica exactamente el mismo tratamiento que a una foto. Al final, no importa si es imagen o video; todo el material termina organizado bajo una misma lógica sin que nadie tenga que tipear las etiquetas a mano.

En lo que respecta a la arquitectura, el sistema se compone de tres capas. El servidor, construido sobre FastAPI en Python, gestiona la lógica del negocio, el control de autenticación y la coordinación del procesamiento con inteligencia artificial. La base de datos opera en PostgreSQL y almacena las etiquetas generadas por la IA en campos JSONB, un formato que permite manejarlas sin depender de un esquema rígido de columnas. La interfaz, desarrollada en React 18, organiza el contenido en una galería tipo mosaico que se adapta al tamaño de pantalla del dispositivo, incorpora una barra de búsqueda que sugiere resultados conforme el usuario escribe alimentándose de las etiquetas que la inteligencia artificial generó y un sistema de carga masiva que acepta lotes de archivos mediante arrastre. El entorno completo se ejecuta dentro de contenedores de Docker Compose, lo que permite replicar la instalación en cualquier servidor sin configuraciones adicionales.

Un aspecto que se definió desde las primeras etapas del diseño, en conjunto con el tutor de la investigación, fue la protección del contenido. El sistema diferencia cuatro perfiles de usuario administrador, supervisor, integrante e invitado y cada uno opera dentro de un perímetro de acción delimitado. Para que un archivo llegue a la galería pública tiene que pasar obligatoriamente por el supervisor; nadie más tiene el permiso para dar ese visto bueno. Los integrantes del equipo hacen todo el trabajo previo subir y dejar listos los videos o fotos, pero la decisión final sobre qué sale a la luz no está en sus manos. En caso de que se necesite borrar algo por completo de la base de datos, el único que puede hacerlo es el administrador. Sin embargo, el código le pone un freno importante: si el material ya llegó a estar público, el botón de eliminar simplemente deja

de funcionar. En cuanto a los invitados, no tienen mucho margen de acción, ya que la pantalla solo les muestra lo que ya superó todos estos filtros. Con ese esquema, cada archivo debe pasar por un ciclo de revisión antes de quedar accesible al público, lo que evita que se repitan los problemas que dieron origen a este proyecto.

El desarrollo del proyecto se apoyó en Scrum como marco de trabajo ágil, distribuido en cuatro iteraciones de dos semanas cada una, con un acumulado cercano a las 300 horas de trabajo técnico entre la construcción del servidor, la integración del modelo de inteligencia artificial, el diseño de la interfaz y la ejecución de pruebas funcionales.

En cuanto a la estructura del presente documento, se organiza en seis capítulos. Este primer capítulo expone la problemática, la justificación y los objetivos de la investigación. El segundo capítulo reúne los referentes teóricos y los antecedentes de investigación que orientaron el desarrollo. El tercero detalla el diseño metodológico, los instrumentos empleados para recoger información y los hallazgos del diagnóstico. El cuarto presenta la propuesta técnica: la arquitectura del sistema, la distribución del trabajo en iteraciones, el diseño de las pantallas y los resultados de las pruebas realizadas. El quinto analiza el desempeño obtenido y contrasta lo planificado con lo ejecutado. El sexto cierra el documento con las conclusiones derivadas del proceso y un conjunto de recomendaciones orientadas a proyectos futuros.

1.2 Presentación del tema

Sistema informático para la gestión de búsqueda de contenidos audiovisuales en SoftMedia Uleam El Carmen.

1.3 Ubicación y contextualización de la problemática

En el Club de Audiovisuales de la Universidad Laica Eloy “Alfaro” de Manabí, extensión El Carmen, se realiza una labor fundamental en la producción de material multimedia: elaboración de fotografías, producción de vídeos, realización de documentales y la creación de otros recursos gráficos que fortalecen la comunidad estudiantil y los procesos institucionales. Este espacio se ha convertido en un pilar fundamental para la difusión de los proyectos y de los eventos académicos.

Sin embargo, la producción en constante expansión ha comenzado a enfrentar un problema que se vuelve obvio: la ausencia de un sistema que organice y administre los recursos de forma eficiente. La información y los archivos se encuentran guardados de forma manual o se hallan distribuidos en distintos lugares, lo cual provoca la pérdida de

información y desperdicio de tiempo para la localización de un documento, la duplicación de materiales y las dificultades para reutilizar el contenido en proyectos a desarrollar.

Todo lo anterior provoca un bajo rendimiento en las actividades del club y disminuye la utilidad del material elaborado para estudiantes, docentes y personal en general. En un escenario donde la agilidad y el orden son claves, es urgente contar con una solución que mejore la gestión de los recursos audiovisuales. Solo de esta forma, se podrá capturar en forma óptima el potencial y el tiempo que se ha invertido, de manera que el trabajo del club audiovisual no solo atienda las necesidades más apremiantes, sino que también quede disponible para fortalecer proyectos futuros.

1.4 Planteamiento del problema

1.4.1 Problematización

En la extensión universitaria de El Carmen, ha aumentado la comunidad estudiantil y académica produciendo contenidos audiovisuales. Sin embargo, la falta de un sistema informático que centralice y organice este diverso material, crea inconvenientes que dificultan la recuperación de ciertos archivos y el pleno aprovechamiento de los recursos disponibles.

La falta de un sistema informático que incorpore catalogación y clasificaciones permite el ahorro de tiempo. Esto, sin duda, afecta la recuperación de archivos y la eficiencia de los proyectos y la respuesta a nuevas actividades. Además, la ausencia de clasificaciones, etiquetas y metadatos obvia la posibilidad de contar un respaldo organizado de los materiales, limitando su uso en auditorías académicas, investigación y proyectos institucionales.

Entonces, el problema consiste en la falta de un sistema informático que clasifique y organice contenidos audiovisuales y que facilite su búsqueda y recuperación de forma rápida y ordenada utilizando tecnología de inteligencia artificial moderna.

1.4.2 Génesis del problema

El problema en la gestión de contenidos audiovisuales del Club de Audiovisuales SoftMedia de la ULEAM Extensión El Carmen tuvo sus raíces a principios de 2025, cuando estudiantes de la carrera de Ingeniería en Tecnologías de la Información, propuso al Decano y autoridades de la extensión la creación de un espacio para producir contenido multimedia. La iniciativa arrancó con un video promocional semiprofesional que resaltaba las carreras de TI y Desarrollo de Software, destacando sonido claro e imágenes estabilizadas, y que fue bien recibido por las autoridades. Cuando el club recién se formó,

el motivo principal detrás de la idea era darles a los estudiantes un espacio válido para cumplir con sus pasantías. Pero nunca se trató de solo firmar un papel; desde el día uno la meta fue armar producciones visuales que a la universidad de verdad le sirvieran para sus redes o campañas. Esa iniciativa cayó bien en la extensión. Tanto así que terminaron habilitándoles una oficina propia, y fue trabajando desde ahí que el grupo logró sumar más de 240 horas repartidas entre salir a cubrir los actos y editar todo lo que se mandaba a los canales oficiales.

Ese crecimiento, que en principio fue un logro, pronto trajo consigo un problema que nadie había anticipado. El volumen de archivos aumentó a un ritmo que el sistema de almacenamiento adoptado “Google Drive” no estaba preparado para manejar de forma ordenada. Sin una estructura de carpetas definida ni criterios claros para nombrar o clasificar los archivos, la desorganización fue instalándose de manera progresiva hasta volver inmanejable el repositorio. Esto dificultó su trazabilidad, exigiendo lentas revisiones manuales y forzando la eliminación de contenido de valor institucional por falta de capacidad. En consecuencia, la recuperación y reutilización de material histórico quedó severamente limitada. Todo este panorama terminó pasándole factura al ritmo de trabajo del grupo. Como no existía un sistema pensado para crecer junto con el club, el desorden empezó a frenar lo que antes fluía rápido.

1.4.3 Estado actual del problema

Hoy por hoy, SoftMedia sigue lidiando con su archivo audiovisual de manera improvisada. El material termina suelto en la nube, sin una estructura de carpetas que tenga sentido y sin nombres que ayuden a identificar de qué trata cada video o foto. Para ubicar una toma de hace un mes, a alguien le toca abrir archivo por archivo hasta dar con ella, perdiendo horas que estarían mejor invertidas en editar o grabar.

El otro punto crítico es que el disco virtual siempre está al límite. Cuando salta la alerta de que no hay más espacio, la "solución" suele ser borrar lo más viejo para meter lo nuevo. Nadie tiene tiempo de ponerse a mirar si ese material antiguo podría reciclarse para otro video. Simplemente se borra y se pierde para siempre.

Para empeorar las cosas, si un integrante necesita una foto para un posteo o un invitado quiere ver un evento pasado, no hay un enlace o galería donde puedan buscar por su cuenta mediante etiquetas. Ese cuello de botella hace que el trabajo en equipo sea lento y que el contenido producido casi no llegue a la gente de afuera.

1.5 Diagrama causa – efecto del problema

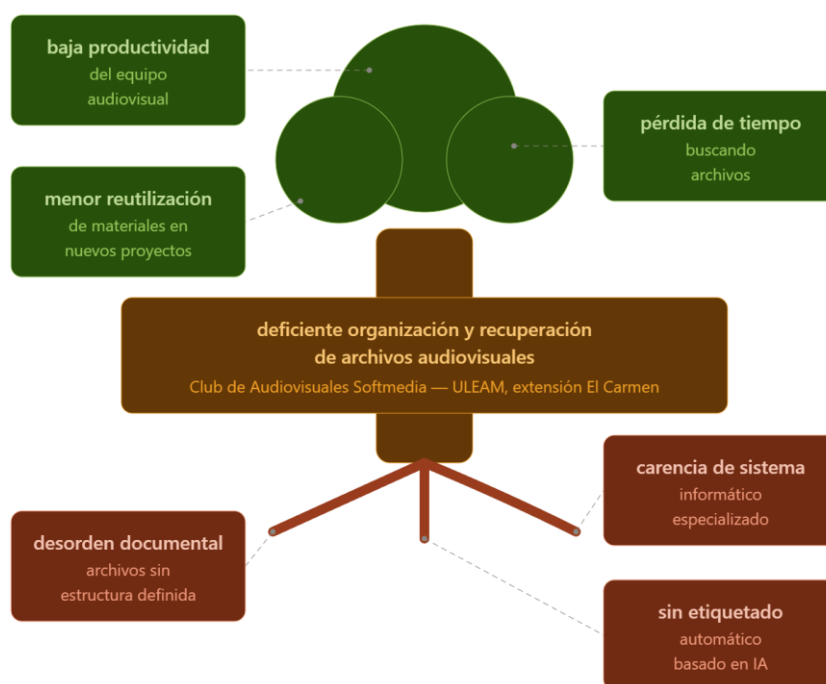


Ilustración 1: Diagrama causa-efecto del problema

1.6 Objetivos

1.6.1 Objetivo general

Desarrollar un sistema informático para organizar y catalogar contenidos audiovisuales mediante etiquetas generadas automáticamente con inteligencia artificial, optimizando la gestión y búsqueda de los archivos producidos por el Club de Audiovisuales SoftMedia en la ULEAM Extensión El Carmen.

1.6.2 Objetivos específicos

- Examinar cómo se almacena y recupera actualmente el material multimedia en el Club SoftMedia, detectando las fallas en su organización y las carencias a nivel de infraestructura, a través de instrumentos de recolección de datos tanto cuantitativos como cualitativos.
- Establecer las bases conceptuales que respaldan la catalogación de contenidos audiovisuales y el desarrollo de software por componentes, argumentando desde la literatura especializada por qué el modelo BLIP de Salesforce resulta adecuado para aplicarse en el contexto de la institución.

- Organizar el proceso de desarrollo bajo el marco ágil SCRUM, estableciendo en cada iteración los requerimientos que debe cumplir la plataforma, tanto en términos de funcionalidad como de rendimiento incluyendo la capacidad del hardware disponible y los tiempos de respuesta del modelo de inteligencia artificial.
- Desarrollar el sistema separando el cliente del servidor. Para lo que ve el usuario se usará React.js, mientras que el cerebro del sistema correrá sobre FastAPI. Por el lado de la base de datos, la elección es PostgreSQL; en particular, se utilizará su formato JSONB porque permite guardar todas las etiquetas sueltas que inventa la inteligencia artificial sin obligarnos a crear un esquema rígido en las tablas.
- Poner a prueba la plataforma una vez terminada para ver si realmente cumple su propósito. La idea es medir si ahora toma menos tiempo encontrar una foto o video en comparación con el método anterior. También se revisará en la práctica si las restricciones de permisos funcionan bien y nadie puede meter mano en archivos que no le corresponden.

1.7 Justificación

Resolver los problemas actuales en la gestión de contenidos audiovisuales en el Club de Audiovisuales SoftMedia de la ULEAM Extensión El Carmen puede traer cambios positivos en varios aspectos importantes. Las mejoras que se anticipan con la implementación de esta propuesta abarcan tres dimensiones: tecnológica, social y ecológica, las cuales se explican en los apartados siguientes.

1.8 Impactos esperados

1.8.1 Impacto tecnológico

Contar con un sistema que ordene el material multimedia de forma automática cambia directamente cómo trabaja el equipo. En lugar de desperdiciar horas abriendo y cerrando carpetas mal nombradas para encontrar un clip específico, el equipo va a poder usar ese tiempo en lo suyo: grabar y editar. Es un cambio práctico que se nota rápido. El estudiante que sale a cubrir el acto sube su material al servidor sin complicaciones, y la persona encargada de armar el video final lo encuentra al instante.

Hacia afuera, el beneficio es el mismo. Si alguien de otro departamento de la universidad necesita una foto de archivo para un afiche, ya no tiene que andar

persiguiendo a los miembros del club por WhatsApp para que se la busquen. Simplemente entra a la plataforma, busca por fecha o tema, y listo.

1.8.2 Impacto social

El orden no solo ayuda con los tiempos de entrega, también cambia la actitud del grupo. Si los editores saben que pueden conseguir una toma de archivo en dos minutos, la van a usar para enriquecer sus videos nuevos en vez de conformarse con lo poco que grabaron ese día o tener que salir a grabar otra vez. Esto hace que el trabajo de meses anteriores no se quede muriendo en un disco duro.

Por el lado anímico, hay un factor clave: cuando un estudiante de los primeros semestres se da cuenta de que sus fotos o sus videos no se pierden, sino que quedan bien guardados, clasificados y listos para que cualquiera en la universidad los vea, se siente valorado. Ese simple detalle es el mejor incentivo para que quieran seguir participando en las siguientes coberturas. Paralelamente, la exposición controlada de galerías públicas permitirá compartir estos logros con padres de familia, futuros alumnos y la sociedad, consolidando un profundo sentido de pertenencia y orgullo institucional hacia la extensión.

1.8.3 Impacto ecológico

El desarrollo de este sistema promueve los principios de la sostenibilidad digital (Green IT) mediante la optimización de los recursos de hardware existentes en la institución. Al ejecutar el procesamiento algorítmico y centralizar el almacenamiento de metrajes directamente en el servidor local de la universidad (On-Premise), se mitiga la huella de carbono digital asociada a la transmisión masiva y constante de archivos pesados hacia nubes públicas. De esa forma, se evita que los mismos archivos se guarden varias veces en distintos lugares, se hace un uso más racional de los equipos disponibles y se reduce la necesidad de recurrir a servicios externos para sostener la operación.

CAPÍTULO II

2 MARCO TEÓRICO DE LA INVESTIGACIÓN

2.1 Antecedentes históricos

La forma de manejar información en entornos educativos ha cambiado bastante con el tiempo. Durante años, lugares como bibliotecas o grupos de producción multimedia dependían de registros físicos listas en papel, archivadores, carpetas rotuladas a mano para llevar el control de sus recursos. Ese método, aunque funcional en escalas pequeñas, traía consigo problemas conocidos: materiales duplicados, datos incompletos y tiempo invertido en búsquedas que no siempre daban resultado. Algo muy parecido era lo que pasaba en los primeros tiempos del Club SoftMedia, cuando Google Drive era el único recurso disponible y nadie había definido cómo organizar lo que allí se guardaba. Según Caraballo (2022) esta evolución se puede ver claramente en cómo se pasó de sistemas locales a otros más distribuidos, incorporando enfoques que permiten entender cómo han ido apareciendo herramientas como Git, que sirven para llevar control de versiones y ver los cambios que se hacen en los archivos o códigos con el tiempo.

Leiva y Sánchez (2024) también muestran cómo la tecnología ha cambiado la forma en que se maneja la información bibliográfica. Antes se usaban hojas de cálculo, pero ahora se trabaja con módulos digitales que aplican estándares como MARC21 para hacer una catalogación más automática. Este cambio, impulsado por el desarrollo de las TIC desde hace más de una década, ha permitido que bibliotecas universitarias puedan manejar procesos como la adquisición o el cuidado de los materiales, algo que también se puede aplicar a SoftMedia, sobre todo considerando cómo ha crecido su contenido desde que fue creado en 2025. Actualmente, muchos de estos sistemas ya permiten accesos remotos y automáticos, y han pasado a funcionar como plataformas web, donde se pueden hacer búsquedas en galerías abiertas al público. Esto ayuda a reutilizar el material audiovisual sin tener que revisar todo manualmente.

2.2 Antecedentes de Investigaciones Relacionadas al Tema Presentado

Las investigaciones sobre sistemas informáticos han avanzado en la automatización de gestión en entornos educativos y comerciales, enfocándose en eficiencia y reducción de errores. Desde el lado del procesamiento visual, el trabajo de D. F. Duque (2024) en la Universidad de Antioquia exploró cómo comprimir redes convolucionales para analizar imágenes en dispositivos con capacidad limitada. Lo

relevante de su investigación es que, al combinar distintos modelos y evaluar los resultados con métricas de precisión, consiguió acortar considerablemente los tiempos que tarda el sistema en procesar grandes cantidades de imágenes, un resultado directamente aplicable al tipo de entorno que se trabaja en este proyecto.

Por otra parte, en lo que respecta a la estructuración documental, Leiva y Sánchez (2024) desarrollaron un módulo automatizado de catalogación para inventarios bibliográficos en la Universidad Central del Ecuador. Dicho trabajo establece un precedente metodológico fundamentado en sistemas ERP y SIGB para la planificación de recursos, principios arquitectónicos que resultan directamente aplicables al diseño de la base de datos y la organización del catálogo multimedia propuesto. Sus avances incluyen formatos MARC21 para clasificación automática Alava (2022) en la Universidad Politécnica Salesiana analizó un sistema web para gestión de créditos en almacenes, con avances en interfaces que automatizan solicitudes, citando autores como Rodríguez et al. (2021) en inventarios digitales.

Caraballo (2022) en la Universidad de las Ciencias Informáticas propuso un sistema para recursos materiales y libros, revisando soluciones como KOHA y ARCE, avanzando con pruebas unitarias para validación. Por su parte, Chalacán et al. (2022), también en la Universidad Central del Ecuador, desarrollaron una plataforma web orientada a proyectos de ética institucional, incorporando módulos de seguimiento y perfiles de acceso diferenciados por usuario. En conjunto, estas investigaciones muestran un avance claro hacia sistemas con interfaces web y criterios de evaluación más precisos; sin embargo, ninguna aborda de forma directa la búsqueda de contenido audiovisual con apoyo de modelos de visión computacional, que es justamente el vacío que el presente trabajo busca atender.

2.3 Definiciones Conceptuales

2.3.1 Variable Independiente: Sistemas Informáticos

2.3.1.1 Definición

Un sistema informático es un conjunto integrado de hardware, software y procesos diseñados para gestionar, almacenar, procesar y recuperar información de manera eficiente, aplicable a la organización de contenidos audiovisuales en entornos educativos y patrimoniales. Según Alava (2022) se define como una plataforma orientada a la web que automatiza flujos de trabajo, como la gestión de solicitudes en almacenes, extensible a la catalogación de archivos multimedia para minimizar errores manuales y optimizar

recursos (*Los medios de comunicación en la era digital*, 2023) amplía esta definición al enfatizar su rol en la transformación digital de medios audiovisuales, donde estos sistemas disocian la creación de contenidos de su edición y publicación, facilitando la omnicanalidad, la convergencia de audio, video y texto, y la integración de tecnologías como 5G para hiperconectividad en plataformas de streaming. Bazán (2023) lo plantea como una herramienta pensada para gestionar de forma completa los archivos audiovisuales que tienen valor patrimonial. Con el tiempo, estas soluciones han ido evolucionando hacia plataformas MAM (Media Asset Management) más avanzadas, que automatizan todo el proceso de trabajo, desde la captura hasta la transcripción, el orden de los archivos y su búsqueda. Todo esto con un enfoque fuerte en la preservación digital.

Por su parte, Caraballo (2022) menciona que los sistemas informáticos se crean, en gran parte, para automatizar tareas que antes se hacían a mano, lo que ayuda a ahorrar tiempo y a reducir los errores cuando se trabaja con mucha información. Su aplicación abarca desde la gestión académica y bibliotecaria hasta los sistemas de control administrativo y toma de decisiones.

2.3.1.2 Componentes

Hoy, un sistema informático consta de varios componentes interrelacionados que se combinan para permitir a las organizaciones el procesamiento, el almacenamiento y la comunicación de la información. Según D. F. Duque (2024), estos componentes son una estructura integral de interacción que combina los aspectos físicos, lógicos y humanos. Los componentes suelen clasificarse en cuatro categorías principales: hardware, software, datos y seres humanos.

2.3.1.2.1 Hardware

El hardware es el conjunto de partes físicas y tangibles que conforman un sistema informático. Son componentes reales que sirven para la comunicación entre el usuario y la máquina; estos permiten realizar procesos y operaciones sobre la información. De forma general, el hardware incluye equipos principales como computadoras, servidores, dispositivos móviles, dispositivos de almacenamiento, unidades de procesamiento y sistemas administradores de red.

Según N. Duque (2025) el hardware es la base material del sistema, ya que dentro de este se ejecuta la parte no tangible, el software. Según su análisis, demuestra que la interacción entre ambos elementos hace posible la transformación de datos en información y órdenes procesadas. Esta investigación clasifica el hardware en categorías

principales: dispositivos de entrada, procesamiento, almacenamiento y salida, los cuales trabajan de forma coordinada para que sea posible la operatividad del sistema.

2.3.1.2.2 *Dispositivos de entrada*

Son aquellos que se encargan de la comunicación del usuario con el sistema. Al introducir comandos o datos desde estos dispositivos se genera una acción dentro del sistema. Los más comunes son el teclado, el ratón, sensores y escáneres. Al usar estos dispositivos, sirven como puente entre la comunicación del entorno físico y digital, realizando exitosamente el envío de información para que esta sea procesada por el sistema (N. Duque 2025).

2.3.1.2.3 *Componentes de procesamiento*

Laudon & Laudon (2020) mencionan que el componente de procesamiento es el núcleo operativo del sistema informático. Más conocida como la Unidad Central de Procesamiento (CPU), se encarga de procesar e interpretar las instrucciones del software; además, realiza operaciones lógicas y aritméticas necesarias para el funcionamiento del sistema. Es comparable con el cerebro humano, ya que procesa la información para realizar acciones. Gracias a estos componentes, los datos se transforman en información entendible para el computador, el cual analiza y da resultados concretos, como gráficos, textos o cálculos, que luego se mostrarán al usuario.

2.3.1.2.4 *Dispositivos de almacenamiento*

Son los responsables de retener la información de manera temporal o permanente. Aquí se engloban desde la memoria de acceso aleatorio (RAM), Lira (2021) dice que este dispositivo proporciona un espacio de trabajo volátil, hasta soluciones de almacenamiento persistente como unidades de estado sólido (SSD), discos duros, servidores en red y dispositivos de almacenamiento externo.

2.3.1.2.5 *Dispositivos de salida*

Estos componentes se encargan de presentar los resultados generados por el sistema. Su función es hacer la información digital perceptible para el usuario, a través de monitores que proyectan interfaces visuales, impresoras que producen copias físicas, proyectores que amplían la imagen o altavoces que emiten sonidos y vocalizaciones.

En paralelo, Caraballo (2022) destaca que la evolución del hardware ha estado dirigida a cubrir las exigencias de los sistemas contemporáneos. La tecnología actual prioriza la agilidad operativa, la eficiencia en el procesamiento y la robustez en la seguridad, aspectos críticos en entornos donde la automatización de procesos como la

gestión documental o las operaciones administrativas se ha vuelto indispensable. En esencia, el progreso del hardware ha sido clave para estructurar y facilitar la interacción en un contexto social y profesional crecientemente digital.

2.3.1.3 Software

El software representa la dimensión lógica e intangible de un sistema informático. Se compone de los programas, instrucciones y reglas que dictan al hardware cómo debe operar para cumplir con tareas determinadas. Como señalan Mosquera et al. (2020), el software actúa como el intelecto del sistema, ya que es el responsable de orquestar los procesos, gestionar los recursos disponibles y servir de puente para la interacción con las personas. De forma general, el software se puede clasificar en tres categorías principales:

2.3.1.3.1 Software de sistema

Esta categoría es, por así decirlo, es la columna vertebral de cualquier computadora o sistema informático. Se trata de la parte fundamental y esencial que se encarga de dos cosas principales: administrar todo el hardware (la parte física) y poner a funcionar los programas que usamos.

En este grupo existen cosas más conocidas, como los sistemas operativos (Windows, Linux, macOS), que son la base de todo. También se incluye a los controladores los cuales hacen que el hardware hable con el sistema y esas herramientas de diagnóstico que a veces usamos para ver qué falla.

2.3.1.3.2 Software de aplicación

Este tipo de software está diseñado para mejorar la experiencia al usuario, ya que puede responder a tareas específicas asignadas. Es un conjunto de programas orientados a realizar trabajos específicos, algunas de ellas es el procesamiento de textos, realizar cálculos, navegar por internet, monitorear camas de seguridad o administrar bases de datos. En la parte del ámbito profesional, Caraballo (2022), menciona que las aplicaciones informáticas han transformado totalmente la gestión administrativa y académica, lo cual permite la automatización de procesos por ende mejora la eficiencia a la hora de manejar información.

2.3.1.3.3 Software de desarrollo o programación

Son aquellas herramientas de trabajo para los programadores que permiten diseñar y mantener nuevas aplicaciones. Entre ellas se incluyen lenguajes de programación como lo es Java, Python o PHP, gestores de bases de datos como MySQL, y frameworks de trabajo. P. Rodríguez (2024) en su investigación resalta la relevancia de las tecnologías

de código abierto, ya que con ellas se puede reducir los costos de la creación de un sistema informático y con ello favorecen la adaptación del software a los requerimientos específicos de cada empresa que necesita dicho software.

2.3.1.3.4 Datos

Es aquella materia prima con la cual operan los sistemas informáticos. Pueden ser números, textos, imágenes o sonidos que, al momento de ser procesados, estos se convierten en información legible útil, para la toma de decisiones. N. Duque (2025) señala que, al tener una correcta estructura de bases de datos, favorece con la integridad y la disponibilidad de información.

2.3.1.3.5 Usuarios o personas

Esta categoría se trata del componente humano el cual es esencial en cualquier sistema informático. Las personas son las que usan el sistema, dan las ordenes e ingresando datos, también son encargados de interpretar los resultados y según esto garantizar el uso adecuado del sistema informático y la tecnología. Mosquera et al. (2020) mencionan que depende del conocimiento y el compromiso de cada usuario para que el sistema funcione de manera eficiente, ya que este es el que da sentido a la herramienta tecnológica.

Hay diferentes tipos de usuarios: los usuarios finales, son los que utilizan el sistema en sus actividades cotidianas, los administradores, en este caso ellos controlan el acceso al sistema y supervisan que todo esté en correcto funcionamiento, los desarrolladores o técnicos, estos son responsables de realizar el mantenimiento y mejora continua del software y el hardware. P. Rodríguez (2024) en su investigación destaca que los sistemas actuales deben diseñarse guiándose en las necesidades, requerimientos y experiencia del usuario (UX), para conseguir que la interacción del humano con la tecnología resulte intuitiva, fácil de acceder y eficiente.

2.3.1.4 Funcionamiento general del sistema informático

Para que un sistema informático funcione, se necesita que todos sus componentes tengan una interacción coordinada y entre ellos están los principales elementos que se encargan de que todo funcione y son: el hardware, software, datos y usuarios. Se menciono anteriormente de estos componentes, ya que son los encargados de que las operaciones se realicen bajo un ciclo lógico en donde la información es introducida, procesada, almacenada y finalmente regresa al usuario en forma de resultados útiles. En su investigación Lira (2021) indica que cualquier sistema informático opera utilizando un

flujo ordenado de información que comienza con la entrada de datos a través de dispositivos físicos, se somete a un proceso interno de cálculo y análisis, terminando con la presentación de resultados a través de diversos medios de salida. Este ciclo continuo garantiza la transformación de los datos en conocimiento útil para las organizaciones y usuarios.

Según Laudon & Laudon (2020) indica que al momento de que los datos son procesados esto se constituye como el corazón del sistema, ya que esta fase es donde el software ejecuta las instrucciones del usuario, dentro del hardware con el objetivo de generar información entendible. El autor también habla que la comunicación entre ambos componentes se la da mediante un sistema operativo, el cual actúa como un mediador entre el humano y aquellas operaciones que se realizan dentro de la máquina.

Cuando se habla del ámbito empresarial, Alava (2022) compara al ciclo funcional del sistema informático como una secuencia de tareas que están orientadas a simplificar procesos que antes se los hacía manualmente.

Para finalizar, Mosquera et al. (2020) afirma que un sistema informático bien diseñado debe estar en capacidad de controlar, validar y retroalimentar los datos, para poder asegurar que cada fase como lo son la entrada, procesamiento, almacenamiento y salida de información contribuya a tener una gestión efectiva. De modo que el funcionamiento de un sistema informático se lo reconoce como un proceso dinámico y continuo, donde la participación humana y la tecnología se complementan para llegar a un objetivo organizacional.

2.3.1.5 Bases de datos y gestión de la información

De manera general se dice que las bases de datos son el pilar de cada gestor de información dentro de los sistemas informáticos porque permite la organización, almacenamiento y facilita el acceso de los datos. Tapia (2020) dice en su libro que una base de datos se la reconoce como un conjunto de información ordenada, la cual se la puede gestionar mediante un Sistema Gestor de Base de Datos o más conocido como un (SGBD), este último se encarga de administrar, aquellas operaciones, creación, consulta y dar un mantenimiento eficaz y seguro. Para esto hay sistemas como MySQL, SQL Server o PostgreSQL, estos sirven para mantener la información integra, y disponible, aunque varios usuarios accedan simultáneamente.

Por su parte, P. Rodríguez (2024) afirma que, al momento de construir una base de datos moderna, debe estar diseñada rigiéndose por la accesibilidad, escalabilidad y seguridad, principalmente cuando se está usando herramientas de código abierto. Según

su investigación dice que los gestores open source ofrecen ventajas muy competitivas ya que tienen un bajo costo, se pueden personalizar y tienen compatibilidad con diversas plataformas.

Leiva y Sánchez (2024) relatan que, al desarrollar un módulo de catalogación bibliográfica, destacan el valor de las bases de datos para manejar el control del inventario y la generación de estadísticas. En su proyecto emplean modelos relaciones y metadatos como (MARC21) esto asegura una catalogación estandarizada, y facilita una recuperación de información en bibliotecas y archivos institucionales.

2.3.1.6 Experiencia de usuario (UX) en los sistemas informáticos

Hoy en día, cuando se desarrolla un sistema informático, no se trata solo de que funcione. También hay que pensar en cómo lo va a vivir la persona que lo usa, o sea, la experiencia del usuario, lo que se conoce como UX. Aunque suene técnico, es algo bastante simple: si es fácil de usar, si carga rápido, si se entiende sin tener que adivinar. Si la persona que lo usa se siente cómoda, todo fluye mejor. Pero si no, la experiencia puede ser molesta, confusa, o tan incómoda que ni dan ganas de seguir usando el sistema. Así que ya no alcanza con que el sistema esté bien armado por dentro; también tiene que sentirse bien por fuera. Al final, la experiencia de usuario es la que hace que una herramienta tecnológica sea útil y tenga sentido (Lira, 2021)

P. Rodríguez (2024) plantea que para que un sistema informático sea realmente eficiente, debe pensarse desde el punto de vista de quien lo va a usar. Esto involucra aplicar los principios de usabilidad y escalabilidad para tener una mejor navegación, bastante clara y sencilla. En su investigación el autor señala que al momento de desarrollar un sistema web, no solo se trata de puro código, si no también este debe lograr adaptarse a las necesidades del usuario ya sean de forma visual o mental. Por ello es importante el cuidar el diseño de cómo están organizados los menús, la forma en que se presenta la información al usuario y si el sistema es funcional, en diferentes dispositivos.

N. Duque (2025) menciona que la experiencia del usuario tiene mucho que ver con la forma en que se mueve la información dentro del sistema. Para él, el diseño debe hacer que los datos puedan entrar, procesarse y mostrarse de forma clara, sin llenar la pantalla de cosas que no aportan nada. Si el sistema está bien hecho en ese sentido, el usuario lo va a sentir como algo que ayuda, no como algo que estorba. Una buena experiencia de uso hace que la tecnología se vuelva parte del trabajo de todos los días, sin complicarlo más de lo necesario.

2.3.2 Variable Dependiente: Gestión de búsqueda.

2.3.2.1 Definición:

La gestión de búsqueda se entiende como el proceso de encontrar, organizar y recuperar información útil dentro de un sistema o plataforma digital. Ronconi (2020) menciona que este proceso suele pasar por cinco pasos: primero hay que tener clara la necesidad de información, luego elegir dónde buscar, pensar una buena estrategia, revisar cómo va el proceso y, por último, ver si lo que se encontró realmente sirve. Esta forma de organizar la búsqueda ayuda a no perder tiempo y a encontrar datos más precisos.

Juca (2023) comenta que, con los avances en los motores de búsqueda y los algoritmos, todo este proceso ha cambiado bastante. Hoy en día, muchos sistemas usan inteligencia artificial para entender mejor lo que el usuario necesita, quitar resultados que no vienen al caso y mostrar contenido más ajustado a lo que se está buscando.

L. Rodríguez & Miñano (2017) llevan esta idea más allá, hablando de su aplicación en archivos digitales y audiovisuales. Hoy en día, muchos sistemas ya pueden reconocer imágenes, sonidos o videos sin que uno tenga que buscarlos por nombre. Esto lo logran usando etiquetas automáticas y metadatos que ayudan a clasificar todo de forma más ordenada. Gracias a eso, ya no se depende solo del texto para encontrar lo que uno necesita.

Con el paso del tiempo, buscar información se volvió algo más completo. Ya no es solo escribir una palabra y esperar. Ahora, la tecnología trabaja junto con lo que hace el usuario para que todo el proceso sea más rápido, más útil y más adaptado a lo que cada persona busca.

2.3.2.2 Organización y recuperación de la información

Organizar y recuperar información es parte fundamental cuando se habla de gestión del conocimiento. Gracias a este proceso, los datos no solo se ordenan, sino que también se vuelven más fáciles de encontrar y usar, ya sea por una persona o por el sistema mismo. Ronconi (2020) explica que todo comienza por saber qué tipo de información se necesita, después se elige dónde buscar, y al final se revisan los resultados para ver si realmente sirven. Si todo está bien organizado, no solo se accede más rápido, sino que la información que se obtiene es más clara, útil y coherente.

En los sistemas digitales actuales, L. Rodríguez & Miñano (2017) señala que todo este tema ha cambiado mucho por culpa o gracias a las tecnologías inteligentes. Ahora, los datos pueden analizarse, etiquetarse y ordenarse de forma automática, incluso cuando

se trata de archivos de audio o video. En estos casos, el uso de metadatos es clave para que una búsqueda funcione bien. La autora deja claro que todo depende de cómo se estructuran los datos y si se siguen ciertos estándares que permiten que los sistemas se entiendan entre sí.

Por otro lado, Barreto (2022) comenta que, con el uso de redes neuronales, ahora es posible detectar y clasificar cosas en imágenes o videos sin tanta intervención manual. Gracias al *deep learning*, los sistemas pueden identificar objetos, escenas o incluso contextos, y ponerles etiquetas automáticas que luego facilitan encontrarlos. Esto hace que todo el proceso sea más rápido y preciso, sobre todo cuando hay demasiada información visual que revisar.

Desde el lado más técnico, Miraúda (2023) dice que recuperar información en entornos digitales ya no se puede hacer a la antigua. Es necesario contar con sistemas que sepan reconocer patrones y conexiones entre los datos. En su investigación, muestra cómo los modelos que trabajan con varias etiquetas o detección de objetos ayudan a encontrar cosas no solo por lo que escribimos, sino también por lo que aparece en las imágenes.

Por su parte, Renza & Ballesteros (2023) agregan que, gracias al aprendizaje profundo, ahora existen sistemas que pueden mirar grandes cantidades de imágenes y entender qué tienen en común y en qué se diferencian. Esto permite que se organicen mejor los archivos digitales y que los catálogos de contenido visual se armen casi solos, sin tener que hacer todo de forma manual, como antes.

2.3.2.3 Motores de búsqueda y funcionamiento básico

Los motores de búsqueda sirven para encontrar información guardada en distintos lugares como páginas web, bases de datos o archivos digitales. Están pensados para ayudar a que una persona encuentre lo que necesita sin perder tiempo buscando entre todo. Básicamente, lo que hacen es mostrar resultados que parezcan más útiles o relacionados con lo que el usuario escribió Juca (2023).

Ronconi (2020) dice que estos motores funcionan como un puente entre lo que uno quiere saber y las fuentes donde podría estar esa información. Usan algoritmos que analizan palabras clave, el contexto y patrones que se repiten. Pero no todo depende del sistema: también influye cómo está organizada la información y, sobre todo, cómo la persona hace la búsqueda.

2.3.2.3.1 Estructura general de un motor de búsqueda

Juca (2023) explica que un motor de búsqueda trabaja a partir de un ciclo que nunca se detiene y que, en general, sigue tres pasos: primero rastrea, luego organiza la información que encuentra y, por último, la usa para responder a las búsquedas que hacen las personas. Gracias a ese proceso, todo lo que está guardado queda más fácil de consultar después.

2.3.2.3.1.1 Rastreo (Crawling)

Peñarrubia (2024) dice que el rastreo es la parte cuando el sistema empieza a recorrer páginas web o archivos internos, buscando todo lo que pueda encontrar. Usa unos programas llamados bots o arañas, que básicamente se dedican a seguir enlaces y a registrar dónde está cada cosa. Eso sirve para armar una base de datos con lo que existe.

2.3.2.3.1.2 Indexación

Después de recolectar la información, toca ponerle orden. Aquí es donde se agrupa todo en listas o tablas, y se le agregan datos extra como el título, el autor, la fecha, o algunas etiquetas que expliquen de qué trata. Según P. Rodríguez (2020) esta parte es importante porque afecta qué tan rápido se puede encontrar esa información más adelante.

2.3.2.3.1.3 Búsqueda y resultados

Juca (2023) menciona que cuando una persona escribe algo en el buscador, el sistema compara eso con todo lo que tiene almacenado. A partir de eso, calcula cuáles resultados son más útiles. A veces se basa en cuántas veces aparece una palabra (como lo que se hace con el modelo TF-IDF), y otras en el contexto. Al final, muestra una lista ordenada, con lo que parece más relevante arriba.

2.3.2.3.2 Algoritmos y principios de relevancia

Peñarrubia (2024) dice que los motores de búsqueda de hoy no se fijan solo en si una palabra aparece o no. Lo que hacen ahora es usar inteligencia artificial y aprendizaje automático para tratar de entender lo que la persona quiere buscar, incluso si lo escribió de forma rara o incompleta. Esto se logra con técnicas como el procesamiento de lenguaje natural, que básicamente ayudan al sistema a “leer entre líneas”.

Gracias a eso, muchos buscadores empiezan a dar resultados antes de que el usuario termine de escribir. Lo hacen porque han aprendido de miles de búsquedas anteriores, y se van adaptando poco a poco. Con el tiempo, eso hace que respondan mejor, o por lo menos más rápido.

2.3.2.3.3 Tipos de motores de búsqueda

Moreno (2024) propone una forma de dividir los motores según qué tipo de información manejan y qué tecnología usan.

- **Motores generales:** Son los típicos, como Google o Bing, que recorren e indexan prácticamente toda la web pública.
- **Motores especializados:** Están pensados para un área en particular, como las bases académicas, catálogos de bibliotecas o colecciones de audio y video.
- **Motores semánticos o inteligentes:** Estos van un paso más allá. Usan inteligencia artificial para entender lo que estás buscando según tu contexto o historial, y te dan resultados más personalizados.

2.3.2.3.4 Rol de la inteligencia artificial en los motores de búsqueda

García et al. (2024) señalan que, con la llegada de la IA generativa y las redes neuronales, los buscadores ya no se limitan a mostrar una lista de enlaces. Ahora también pueden generar respuestas completas, entender mejor el contenido y adaptarse a lo que el usuario necesita. Este cambio ha hecho que la experiencia sea más directa, más clara y, en muchos casos, más útil.

Por otro lado, P. Rodríguez (2020) explica que esto también ha cambiado cómo se buscan contenidos visuales. Antes, había que escribir bien lo que se quería encontrar. Ahora, los algoritmos pueden reconocer cosas como rostros, objetos o escenas dentro de imágenes o videos, lo que permite hacer búsquedas usando lo que se ve, no solo lo que se escribe.

2.3.2.4 Catalogación y metadatos en la búsqueda de contenidos audiovisuales

2.3.2.4.1 Concepto de catalogación audiovisual

Según el (Ministerio de Cultura y Patrimonio del Ecuador, 2022) cuando se habla de catalogación audiovisual se refiere, más que nada, a poner en orden los materiales como videos o grabaciones para saber qué es cada cosa. Se anotan datos sobre su contenido y también detalles técnicos, y con eso se logra que después sea más fácil encontrarlos o volver a usarlos, aunque haya pasado bastante tiempo.

El mismo manual dice que un buen catálogo debe incluir tres tipos de información:

- **Datos descriptivos:** cosas como el título, el tema, el autor, el año o el tipo de formato.
- **Datos técnicos:** aquí se anotan detalles como en qué soporte está, cuánto dura, qué resolución tiene, o qué tipo de archivo es.

- **Datos administrativos:** se refiere a cosas como el estado del material, si tiene restricciones de uso, dónde está guardado, etc.

Tener todo esto bien organizado ayuda a que los archivos se integren fácilmente en sistemas de búsqueda y no queden perdidos en el olvido digital.

2.3.2.4.2 *Los metadatos como herramienta de búsqueda*

Leiva y Sánchez (2024) explican que los metadatos son como pequeñas fichas que describen lo básico de un archivo digital. En su estudio sobre catalogación bibliográfica, mostraron que usar estándares como MARC21 o Dublin Core ayuda a que distintos sistemas puedan “hablar entre ellos”, y así buscar cosas en varios lugares a la vez sin volverse locos.

En el mundo audiovisual, los metadatos no se quedan solo con lo básico. También incluyen información extra como quién sale en el video, dónde se grabó, de qué género es o qué palabras clave están asociadas. Eso hace que los buscadores puedan encontrar algo, aunque la descripción original no esté completa. Es como si los metadatos rellenaran los huecos.

2.3.2.4.3 *Catalogación automatizada mediante inteligencia artificial*

L. Rodríguez & Miñano (2017) hicieron una prueba con un archivo de imágenes del diario *El Comercio*, usando inteligencia artificial para mejorar las etiquetas. Básicamente, el sistema era capaz de reconocer cosas como caras, objetos o lugares dentro de las fotos, y luego generaba etiquetas sin que nadie tuviera que escribirlas a mano.

Este proyecto fue importante porque demostró que usar IA no solo ahorra tiempo, sino que también reduce los errores humanos y mejora la precisión al momento de organizar archivos multimedia.

2.3.2.4.4 *Interoperabilidad y estándares en la catalogación*

Ronconi (2020) y Silva et al. (2024) coinciden en algo clave: no se trata solo de ordenar archivos por gusto archivístico. También hay que pensar en el usuario. En espacios digitales como plataformas educativas o audiovisuales tener el contenido bien catalogado hace que las búsquedas sean más fáciles, rápidas y hasta más intuitivas.

Gracias a los metadatos, las personas no solo encuentran lo que buscan, sino que entienden mejor el contexto de cada archivo. Así, la experiencia se vuelve más fluida y menos frustrante.

2.3.2.5 Cómo influye la experiencia del usuario al buscar información

La experiencia del usuario (UX) en los procesos de búsqueda tiene que ver con cómo las personas se relacionan con un sistema cuando están buscando información. P. Rodríguez (2024) comenta que no alcanza con que el sistema funcione bien técnicamente. Lo importante también es que sea fácil de usar, que la información se cargue rápido y que todo esté bien organizado para que no se vuelva un enredo.

Silva et al. (2024) señalan que hoy en día es esencial que los sistemas sean claros, con interfaces simples y resultados que sí sean útiles. Una buena experiencia hace que el usuario confíe en el sistema y no se frustre durante la búsqueda. El orden de los elementos y que la información tenga sentido también influye mucho en si el usuario se siente cómodo o no.

2.3.2.6 Importancia de la gestión de búsqueda en los sistemas digitales

La gestión de búsqueda cumple un papel importante dentro de los sistemas informáticos porque ayuda a convertir datos desordenados en información útil. Ronconi (2020) explica que, si se maneja bien, se puede ahorrar tiempo, obtener mejores resultados y tomar decisiones con más facilidad dentro de plataformas digitales.

Moreno (2024) dice que muchos sistemas modernos ya integran inteligencia artificial que se encarga de organizar los contenidos por sí sola. Esto ayuda a filtrar información irrelevante y evita que el usuario se sienta abrumado con tantos resultados a la vez.

2.3.2.7 Uso de IA para etiquetar archivos audiovisuales automáticamente

El etiquetado automático usa inteligencia artificial para revisar imágenes, audios o videos y asignarles etiquetas sin que alguien tenga que hacerlo manualmente. L. Rodríguez & Miñano (2017) trabajaron con un sistema de este tipo y mostraron que puede identificar cosas como personas, objetos, textos o lugares dentro de una imagen, y generar descripciones útiles para catalogar.

P. Rodríguez (2020) también menciona que este tipo de tecnología mejora mucho la búsqueda dentro de archivos digitales. Al reconocer patrones visuales o de sonido, la IA puede crear etiquetas que hacen que sea más fácil encontrar algo, además de ayudar a cuidar y organizar el contenido a largo plazo.

2.3.3 Metodología de Desarrollo de Software

2.3.3.1 Las metodologías ágiles en el desarrollo de software

Desarrollar software implica trabajar en condiciones donde los requisitos cambian, los tiempos son inciertos y el producto que se necesita al inicio no siempre es el mismo que se necesita al final. Los modelos tradicionales de desarrollo, como el enfoque Waterfall, partían de una lógica secuencial: primero se levantaban todos los requisitos, luego se diseñaba, luego se construía. Esa lógica funcionó en ciertos contextos, pero mostró sus límites cuando los proyectos eran complejos o cuando el cliente no tenía del todo claro qué quería desde el principio. En 2001, diecisiete especialistas en desarrollo de software se reunieron y redactaron el Manifiesto Ágil, que planteó cuatro valores centrales: las personas y sus interacciones por encima de los procesos y las herramientas, el software que funciona por encima de la documentación extensa, la colaboración con el cliente por encima de la negociación de contratos, y la capacidad de responder al cambio por encima de seguir un plan (Schwaber y Sutherland, 2020).

De ese manifiesto surgieron varios marcos de trabajo. Kanban organiza el trabajo en un flujo continuo sin ciclos fijos de tiempo. Extreme Programming, conocido como XP, centra su atención en la calidad del código mediante prácticas como las pruebas automatizadas y el desarrollo en parejas. Scrum, en cambio, divide el trabajo en ciclos cortos llamados Sprints, con roles definidos y revisiones periódicas que permiten ir ajustando el rumbo (Schwaber y Sutherland, 2020).

2.3.3.2 Scrum como marco de trabajo ágil

Scrum toma su nombre de una jugada del rugby en la que el equipo trabaja unido para avanzar. Fue formalizado por Ken Schwaber y Jeff Sutherland en los años noventa, quienes retomaron ideas de Takeuchi y Nonaka en el año de 1986, dos investigadores que habían estudiado cómo los equipos de alto rendimiento desarrollaban productos nuevos en entornos de incertidumbre. Lo que diferencia a Scrum de otros marcos es su carácter empírico: no parte de suponer que todo está bajo control, sino de asumir que el conocimiento se construye con el tiempo, a partir de lo que se va aprendiendo en cada ciclo (Schwaber y Sutherland, 2020).

Schwaber y Sutherland (2020) describen tres pilares sobre los que se apoya Scrum. El primero es la transparencia: todo lo que ocurre en el proceso debe ser visible para quienes toman decisiones. El segundo es la inspección: el equipo revisa periódicamente su avance para detectar problemas antes de que se agraven. El tercero es

la adaptación: cuando la revisión muestra que algo no va bien, el proceso se corrige sin esperar al final del proyecto.

2.3.3.3 Roles en Scrum

Scrum trabaja con tres roles. El Product Owner es quien representa al cliente o a los usuarios; su tarea principal es mantener ordenada la lista de requisitos del sistema, conocida como Product Backlog, y decidir qué se construye primero según el valor que cada elemento aporta. El Scrum Master no dirige al equipo, sino que lo facilita: se asegura de que el marco se aplique correctamente y resuelve los obstáculos que frenan el trabajo. El equipo de desarrollo son los profesionales que construyen el producto; deciden cómo organizar su propio trabajo y son responsables colectivamente de lo que entregan (Schwaber y Sutherland, 2020).

En el presente proyecto los roles se asignaron de la siguiente forma: el investigador asumió el trabajo de desarrollo y también las funciones del Scrum Master, mientras que el docente tutor cumplió el rol de Product Owner, revisando los avances al cierre de cada Sprint y orientando las decisiones de diseño.

2.3.3.4 Artefactos de Scrum

Los artefactos de Scrum son tres. El Product Backlog reúne todo lo que el sistema debe hacer, ordenado por prioridad; no es una lista fija, sino que se va ajustando conforme el proyecto avanza. El Sprint Backlog es la parte de ese listado que el equipo decide abordar en una iteración específica. El Incremento es lo que queda al terminar cada Sprint: una versión del sistema que funciona, que puede revisarse y que cumple con los criterios de terminado definidos por el equipo (Schwaber y Sutherland, 2020).

2.3.3.5 Eventos de Scrum: el Sprint

El Sprint es la unidad de trabajo de Scrum. Dura entre una y cuatro semanas y al final de ese periodo el equipo debe tener algo funcional para mostrar. Dentro de cada Sprint ocurren cuatro momentos: la planificación, donde se decide qué se va a construir y cómo; el Daily Scrum, una reunión breve diaria donde el equipo coordina su trabajo e identifica bloqueos; la revisión, donde se presenta lo construido a los interesados y se recoge retroalimentación; y la retrospectiva, donde el equipo analiza cómo trabajó y qué puede mejorar en el siguiente ciclo (Schwaber y Sutherland, 2020).

El sistema de catalogación audiovisual del presente proyecto se desarrolló en cuatro Sprints de dos semanas cada uno, sumando cerca de 300 horas de trabajo técnico.

Organizar el desarrollo en ciclos cortos permitió verificar cada módulo antes de continuar con el siguiente y corregir problemas a tiempo, sin acumularlos hasta el final.

2.3.3.6 Justificación de Scrum para el presente proyecto

Scrum se eligió porque el proyecto reunía condiciones donde un modelo secuencial habría sido poco práctico. El sistema se construyó para una organización que no tenía experiencia previa con plataformas de gestión audiovisual, lo que significaba que los requisitos podían modificarse conforme los usuarios iban entendiendo mejor lo que necesitaban. Trabajar en ciclos cortos con revisión al final de cada uno permitió incorporar esos ajustes sin detener el desarrollo. Schwaber y Sutherland (2020) señalan que Scrum funciona mejor precisamente en ese tipo de contextos, donde el conocimiento sobre lo que se está construyendo se va formando a lo largo del proceso y no está completo desde el inicio.

2.3.4 Conclusiones del marco teórico

La revisión desarrollada en este capítulo reunió los fundamentos teóricos necesarios para sustentar el diseño del sistema de catalogación audiovisual del Club SoftMedia de la ULEAM, extensión El Carmen.

Lo que muestra la literatura sobre sistemas informáticos es que la organización de información ha pasado por un camino largo: de registros manuales en papel a bases de datos locales, y de ahí a plataformas web accesibles desde cualquier dispositivo. Caraballo (2022) y Leiva y Sánchez (2024) documentan ese trayecto en contextos universitarios. El archivo de SoftMedia se encontraba todavía en una etapa inicial de ese recorrido: carpetas sin nombre en un servicio de almacenamiento compartido, sin ningún tipo de estructura. La propuesta de este trabajo busca dar un salto desde ese punto hacia un sistema organizado y accesible.

Sobre catalogación y búsqueda de contenidos, la revisión mostró que combinar metadatos estructurados con etiquetado automático de archivos es una práctica que ya tiene respaldo en la literatura. Rodríguez y Miñano (2017), Barreto (2022) y Miraúda (2023) trabajaron en sistemas que analizan imágenes de forma automática y generan descripciones útiles para la búsqueda. Esos trabajos demostraron que el proceso reduce el tiempo de catalogación y mejora la precisión al recuperar archivos. El modelo BLIP que se integra en el presente sistema sigue esa misma línea.

En cuanto a la metodología, Scrum resultó adecuado para un proyecto donde los requisitos no estaban del todo definidos desde el inicio. Su estructura de ciclos cortos con

revisión periódica permitió construir el sistema por partes, verificar que cada módulo funcionara correctamente antes de avanzar al siguiente y ajustar el rumbo cuando fue necesario (Schwaber y Sutherland, 2020).

Los trabajos previos revisados D. F. Duque (2024), Leiva y Sánchez (2024), Alava (2022), Caraballo (2022) y Chalacán et al. (2022) muestran que es posible implementar sistemas web con automatización en instituciones educativas. Cada uno abordó un problema distinto y en un contexto diferente, pero todos llegaron a soluciones funcionales. Lo que ninguno hizo fue aplicar análisis visual automático a un repositorio audiovisual universitario. Ese es el espacio que ocupa este proyecto.

CAPITULO III

3 MARCO INVESTIGATIVO (DISEÑO METODOLÓGICO)

3.1 Introducción

La transformación digital en la gestión de repositorios multimedia exige la implementación de soluciones informáticas eficientes orientadas a la optimización de los procesos operativos. En este sentido, el diseño de un sistema estructurado bajo una arquitectura cliente-servidor integrando un motor de Inteligencia Artificial para el etiquetado semántico actúa como el eje central para la modernización del acervo audiovisual en el Club SoftMedia de la ULEAM, extensión El Carmen. Lo que se detalla en este apartado es la ruta que se siguió para estudiar el problema antes de escribir una sola línea de código. La decisión fue no quedarse solo con las estadísticas, así que se armó un estudio en dos partes. Primero, se aplicaron cuestionarios cerrados para tener claro qué porcentaje de la universidad estaba siendo afectado por el desorden del club. Luego, para entender el problema de fondo, se incluyeron preguntas abiertas donde cada persona podía dejar sus quejas o sugerencias textuales.

Cruzar esos números con los comentarios reales de la gente fue lo que marcó el norte del proyecto. Básicamente, se buscaba recolectar pruebas contundentes. Sin esa información de respaldo, habría sido muy difícil argumentar por qué la extensión necesitaba dejar de depender de carpetas manuales en la nube para pasar a tener un sistema propio que hiciera todo el trabajo pesado.

3.2 Tipo de Investigación

3.2.1 Investigación mixta

Esta investigación utiliza un enfoque mixto, ya que no se apoya en un solo tipo de método. Se combinan datos cuantitativos con información cualitativa para poder entender mejor el problema que se estudia. Este enfoque permite trabajar tanto con resultados numéricos como con opiniones o descripciones que ayudan a interpretar lo que realmente ocurre. Hernández & Mendoza (2018) mencionan que la combinación de ambos tipos de datos fortalece el análisis de los resultados.

En este caso, el enfoque mixto ayuda a conocer cómo percibe la comunidad universitaria el manejo del material audiovisual y, al mismo tiempo, recoger comentarios

y sugerencias relacionadas con la idea de implementar un módulo de catálogo audiovisual para SoftMedia.

3.2.2 Investigación descriptiva

La Para Vásquez et al. (2023) la investigación descriptiva resulta fundamental cuando el propósito es documentar un escenario en su estado original. Durante este proceso, no se manipula ninguna variable del entorno. El esfuerzo del investigador se concentra exclusivamente en observar la dinámica de los eventos e identificar los componentes que más destacan dentro del fenómeno estudiado.

En esta misma línea, Hernández & Mendoza (2018) señalan que el verdadero valor de este enfoque recae en su capacidad para mapear el contexto de trabajo. Gracias a esto, es posible reconocer con exactitud a los actores involucrados y delimitar las características del problema. A diferencia de otros enfoques, este tipo de investigación no pretende establecer causas ni efectos; su utilidad está en describir cómo es una situación en un momento dado, con la mayor fidelidad posible. Esa descripción es la que luego orienta las decisiones que siguen.

Aplicado al caso del Club SoftMedia, este método permitió observar de cerca cómo los integrantes guardan y buscan su material audiovisual en el día a día. Preguntar directamente a quienes usan el sistema o en este caso, a quienes lo necesitarían y registrar sus prácticas actuales fue lo que dio lugar a un diagnóstico concreto sobre el que se pudo trabajar. Justamente, esta recolección de datos empíricos es lo que permite definir con precisión los requerimientos operativos para la futura plataforma web y la integración del etiquetado con inteligencia artificial.

3.2.3 Investigación de campo

La investigación de campo se apoya en el trabajo directo dentro del espacio donde ocurre la situación que se quiere estudiar. Arias (2021) menciona que, en lugar de basarse únicamente en información ya existente, el investigador se acerca al entorno real y recoge los datos mientras las actividades se desarrollan. De esta manera, la información no se obtiene desde afuera, sino desde el propio contexto.

En el presente proyecto, la investigación de campo se aplica mediante encuestas dirigidas a estudiantes y docentes de la extensión universitaria, con el fin de conocer sus necesidades, percepciones y opiniones respecto a la creación de una plataforma web para SoftMedia y su módulo de catálogo audiovisual.

3.2.4 Enfoque cuantitativo

El enfoque cuantitativo se basa en trabajar con datos que se pueden contar y registrar de forma clara. Según Hernández & Mendoza (2018) en lugar de opiniones o interpretaciones personales, este enfoque se apoya en números que permiten ver con más precisión qué está pasando dentro de un grupo o situación determinada. La información se organiza y se revisa con calma para encontrar repeticiones, diferencias o tendencias que ayuden a entender mejor el problema.

En este estudio, el uso del enfoque cuantitativo se da a través de preguntas cerradas, donde las respuestas son directas y fáciles de clasificar. Este tipo de preguntas permite ordenar los datos sin complicaciones y analizarlos posteriormente con apoyo estadístico, lo que ayuda a obtener resultados más claros y comparables.

3.2.5 Enfoque cualitativo

El enfoque cualitativo se usa cuando interesa saber qué piensan las personas y cómo viven una situación determinada. Ñaupas et al. (2023) expresan que lo importante no son los números, sino lo que cada participante expresa a partir de su experiencia, sus ideas y su manera de ver el problema. Este tipo de enfoque permite acercarse más a la realidad desde el punto de vista humano.

En este estudio, el enfoque cualitativo se trabaja mediante preguntas abiertas, donde los participantes pueden responder sin restricciones. Esto ayuda a recoger opiniones y sugerencias sobre la creación de la plataforma web de SoftMedia y el módulo de catálogo audiovisual, aportando información que no siempre se obtiene con respuestas cerradas.

3.3 Métodos de investigación

3.3.1 Método inductivo

Este método parte de observar cosas que están pasando o que se repiten en un mismo contexto, y a partir de eso se intenta entender lo que podría pasar en general. En vez de partir de una teoría, se arranca con lo que ya se vio o experimentó, y luego se sacan ideas o conclusiones más amplias (Arias, 2021)

En la presente investigación, el método inductivo se aplica al analizar la situación concreta del Club SoftMedia: la desorganización del archivo, la dificultad para localizar material y la ausencia de un sistema de catalogación. A partir de esa realidad observada directamente, se construyó la justificación para desarrollar una plataforma web que

resuelva ese problema, partiendo de lo particular hacia una propuesta de solución más general y replicable.

3.3.2 Método deductivo

El método deductivo funciona al revés: empieza desde una idea general o teoría y se usa para analizar cosas más puntuales. Lo que se busca es aplicar conceptos que ya están establecidos para entender lo que pasa en un caso en particular. Así se puede comparar la teoría con lo que ocurre en la realidad (Haro et al., 2025).

En la investigación actual, el método deductivo se aplica al tomar los fundamentos teóricos revisados en el Capítulo II sobre sistemas de catalogación, etiquetado automático y gestión de búsqueda y contrastarlos con la situación real del club. Ese contraste permitió definir los requisitos del sistema y seleccionar las tecnologías más adecuadas para el desarrollo de la plataforma.

3.4 Fuentes e instrumentos de recolección de datos

3.4.1 Encuesta

La encuesta es un instrumento de recolección de datos ampliamente utilizado en investigaciones con enfoque cuantitativo y mixto, ya que permite obtener información directa de un grupo de personas mediante un conjunto estructurado de preguntas. Según Hernández & Mendoza (2018) la encuesta facilita la recopilación de datos de manera sistemática, permitiendo su posterior análisis estadístico e interpretativo.

Para obtener información de primera mano se diseñó un cuestionario que no dependiera de un solo formato. La mayor parte del formulario exigía contestar de forma afirmativa o negativa; esto se hizo así para poder tabular rápido y sacar porcentajes exactos sobre los problemas actuales. Sin embargo, cerrar todas las opciones habría cortado la posibilidad de recibir sugerencias reales. Por eso se dejaron espacios libres al final. El objetivo de mezclar estas dos cosas era tener, por un lado, la prueba matemática de que había un problema con el material audiovisual y, por otro, una lista de funciones o correcciones que los propios usuarios querían ver en el sistema nuevo. Los encuestados fueron estudiantes y docentes de la extensión, por ser quienes mejor conocen de cerca la situación del club.

3.5 Estrategia operacional para la recolección de datos

3.5.1 Población

La población se define como el conjunto total de personas que comparten características relacionadas con el objeto de estudio, y sobre las cuales se desea obtener información. Arias (2021) señala que la población constituye el universo de estudio del cual se selecciona una muestra representativa para la investigación.

Para esta investigación, se toma en cuenta como población a los estudiantes y docentes que conforman la extensión universitaria, con un total de 2607 personas. Debido a que ellos son los potenciales usuarios de la plataforma web de SoftMedia y del módulo de catálogo audiovisual. Esta población es relevante porque mantiene una relación directa con los procesos de producción, difusión y consumo de contenidos audiovisuales dentro del contexto universitario.

3.5.2 Técnica de muestreo

El muestreo se usa cuando trabajar con toda la población resulta poco práctico o simplemente no es posible. En lugar de eso, se selecciona solo una parte del grupo, cuidando que esa selección permita tener una idea general de lo que ocurre en el conjunto completo. La clave está en que ese grupo elegido refleje, en lo posible, las características del total de personas involucradas (Ñaupas et al., 2023).

Para este proyecto se considera un muestreo probabilístico, ya que este tipo de procedimiento da a todos los integrantes la misma oportunidad de ser elegidos. El tamaño de la muestra se define utilizando una fórmula pensada para poblaciones finitas, donde se toman en cuenta factores como cuántas personas conforman la población, el nivel de confianza y el margen de error aceptable. De esta manera, los datos que se obtengan permitirán acercarse a la opinión general sin necesidad de incluir a todos los miembros.

Para calcular la población a encuestar, se aplicará la fórmula de muestreo propuesta por Hernández & Mendoza (2018)

$$n = \frac{N * z^2 * p * (1 - p)}{(N - 1) * e^2 + z^2 * p * (1 - p)}$$

En esta investigación utilizaremos los siguientes datos:

n = tamaño de la muestra

N = tamaño de la población (2607)

Z = nivel de confianza 95% (1.96)

p = proporción estimada de la población (0.5)

e = margen de error (0.08)

Remplazamos los valores en la fórmula:

$$n = \frac{2607 * (1.96)^2 * 0.5 * (1 - 0.5)}{(2607 - 1) * (0.08)^2 + (1.96)^2 * 0.5 * (1 - 0.5)}$$

Ecuación 1: Fórmula de muestreo para la población a encuestar

Resultado final de la muestra:

$$n \approx 142$$

3.5.3 Tamaño de la muestra

Aplicando la fórmula de la técnica de muestreo para poblaciones finitas, con una población total conformada por 2607 personas, integrada por estudiantes y docentes de la Universidad Laica Eloy Alfaro de Manabí, extensión El Carmen, se estableció un nivel de confianza del 95%, un margen de error del 8% y una proporción estimada de la población de 0.5, parámetros comúnmente utilizados en investigaciones de tipo descriptivo.

Al resolver la fórmula con los valores antes mencionados, se obtuvo como resultado un tamaño de muestra de 142 personas, cantidad considerada suficiente para garantizar la representatividad de la población objeto de estudio. No obstante, para fortalecer la confiabilidad de los resultados, se logró recolectar información mediante 158 encuestas válidas, superando el mínimo requerido.

3.5.4 Análisis de las herramientas de recolección de datos a utilizar

3.5.4.1 Encuesta

La encuesta estructurada mediante escala Likert se aplicó en dos frentes estratégicos para garantizar un procesamiento estadístico estandarizado abarcando a nivel interno a los integrantes del Club SoftMedia con el propósito de diagnosticar las deficiencias de almacenamiento y el tiempo invertido en selecciones manuales mientras que a nivel institucional externo se dirigió a estudiantes docentes y personal administrativo de la ULEAM con el objetivo primordial de documentar las percepciones sobre la desorganización del acervo multimedia y validar estadísticamente la urgencia de

implementar la plataforma web logrando a través de este levantamiento proyectar con precisión cómo la integración de la inteligencia artificial optimizará drásticamente los flujos de trabajo internos de los creadores y garantizará a la comunidad universitaria un acceso ágil y estructurado al material aprobado.

1. ¿Conoce o ha escuchado hablar del Club Softmedia de la ULEAM?	Si	No
2. ¿Ha tenido dificultades para encontrar el contenido multimedia (fotos o videos) de las actividades universitarias cuando los ha necesitado?	Si	No
3. ¿Considera necesario que exista una plataforma web donde el Club Softmedia organice y publique su contenido audiovisual?	Si	No
4. ¿Cree que el uso de inteligencia artificial para etiquetar y ordenar automáticamente fotos y videos sería útil?	Si	No
5. ¿Estaría dispuesto/a a usar la plataforma si estuviera disponible para estudiantes y docentes?	Si	No
6. ¿Considera importante que el catálogo permita búsqueda por palabras clave (p. ej. evento, fecha, persona, tema)?	Si	No
7. ¿Cree que la plataforma debería ser accesible y usable desde dispositivos móviles (smartphones/tablets)?	Si	No
8. ¿Qué funciones considera indispensables para que el catálogo audiovisual de la futura plataforma web sea realmente útil para la comunidad universitaria?	Escriba su respuesta	
9. ¿Qué recomendaciones o ideas aportaría para mejorar la gestión y organización del contenido audiovisual del Club Softmedia?	Escriba su respuesta	

Ilustración 2: Encuesta aplicada

3.5.4.2 Estructura de los instrumentos de recolección de datos aplicados

El cuestionario aplicado a la comunidad de la ULEAM extensión El Carmen (integrantes del club, estudiantes, docentes y personal administrativo) constó de nueve preguntas diseñadas bajo un enfoque mixto. Su objetivo central fue validar de forma empírica la urgencia y el nivel de aceptación de la nueva plataforma audiovisual.

Para agilizar el levantamiento de información, el instrumento se estructuró en dos bloques estratégicos:

- **Cuantitativo:** Siete preguntas cerradas de opción dicotómica (Sí/No), las cuales permitieron medir estadísticamente la interacción actual de los usuarios con la tecnología y la viabilidad operativa del catálogo propuesto.

- **Cualitativo:** Dos preguntas abiertas orientadas a capturar requerimientos funcionales directos.

Este apartado facilitó la recolección de sugerencias y necesidades específicas, complementando los datos numéricos con aportes técnicos indispensables para estructurar el desarrollo web del sistema.

3.5.5 Plan de recolección de datos

Tabla 1: Plan de recopilación de datos.

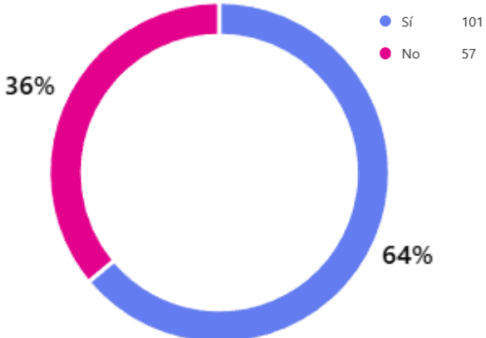
Día	Hora	Personas para encuestar	Tipo de instrumento	Formato
13/01/2026	11:00am	Profesores y personal administrativo	Encuesta	Microsoft Forms.
15/01/2026	11:00am	Estudiantes e integrantes del club	Encuesta	Microsoft Forms.

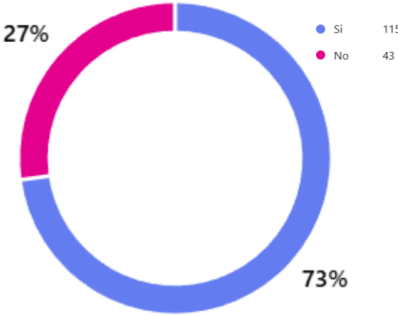
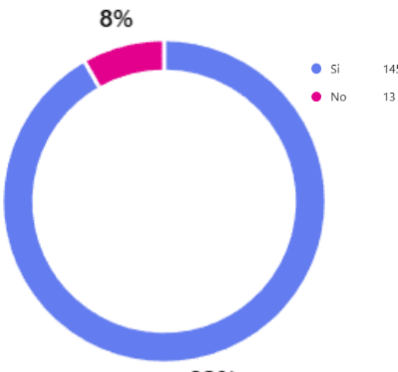
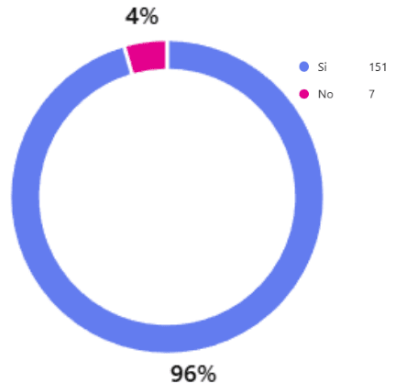
3.6 Análisis y presentación de resultados

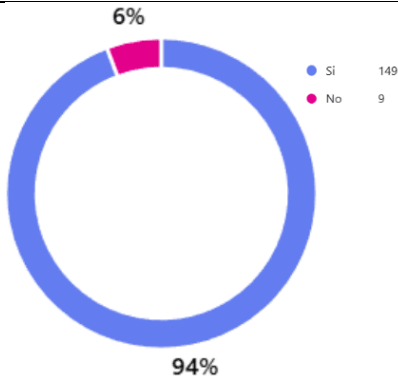
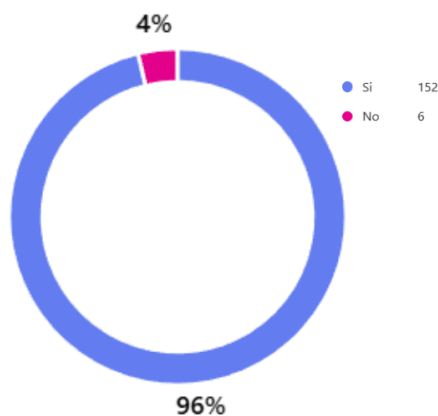
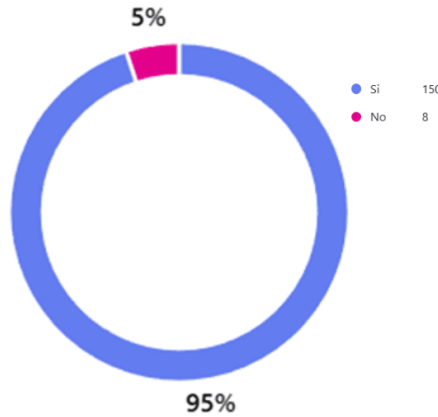
3.6.1 Tabulación y análisis de los datos

3.6.1.1 Encuesta aplicada a docentes, estudiantes y personal administrativo de la Uleam extensión El Carmen.

Tabla 2: Tabulación y análisis de datos de encuesta

Pregunta 1: ¿Conoce o ha escuchado hablar del Club SoftMedia de la ULEAM?	
	Interpretación Más de la mitad de la población (64%) tiene conocimiento sobre la existencia del Club SoftMedia de la ULEAM, lo que confirma que el organismo objetivo del proyecto tiene relevancia y presencia en el entorno académico.
Pregunta 2: ¿Ha tenido dificultades para encontrar el contenido multimedia (fotos y videos) de las actividades universitarias cuando lo ha necesitado?	
	Interpretación

 Donut chart showing the percentage of respondents who find it difficult to locate multimedia content. The chart is divided into two segments: a large blue segment representing 'Si' (Yes) at 73% (115 respondents) and a smaller pink segment representing 'No' at 27% (43 respondents).	El 73% de la población encuestada manifiesta tener dificultades para localizar el contenido multimedia de las actividades universitarias cuando lo ha necesitado, lo que evidencia una problemática en el acceso a la información.
Pregunta 3: ¿Considera necesario que exista una plataforma web donde el Club SoftMedia organice y publique su contenido audiovisual?	
 Donut chart showing the percentage of respondents who consider it necessary to have a web platform. The chart is divided into two segments: a large blue segment representing 'Si' (Yes) at 92% (145 respondents) and a smaller pink segment representing 'No' at 8% (13 respondents).	Interpretación Existe una alta demanda por parte de la comunidad universitaria: más de nueve de cada diez encuestados (92%) consideran indispensable contar con un sistema web centralizado que gestione los recursos multimedia del club.
Pregunta 4: ¿Cree que el uso de inteligencia artificial para etiquetar y ordenar automáticamente fotos y videos sería útil?	
 Donut chart showing the percentage of respondents who believe that the use of artificial intelligence for labeling and organizing photos and videos would be useful. The chart is divided into two segments: a large blue segment representing 'Si' (Yes) at 96% (151 respondents) and a smaller pink segment representing 'No' at 4% (7 respondents).	Interpretación Una inmensa mayoría (96%) coincide en que el uso de inteligencia artificial sería una característica útil para la plataforma, respaldando la necesidad de herramientas inteligentes que faciliten la búsqueda y categorización del material.
Pregunta 5: ¿Estaría dispuesto/a a usar la plataforma si estuviera disponible para estudiantes y docentes?	
	Interpretación

 6% 94% ● Si 149 ● No 9	Los resultados muestran una aceptación casi unánime: el 94% de los encuestados afirma que estaría dispuesto a utilizar la plataforma, lo que garantiza la viabilidad y el uso activo de la herramienta por parte de estudiantes y docentes.
Pregunta 6: ¿Estaría dispuesto/a a usar la plataforma si estuviera disponible para estudiantes y docentes?	
 4% 96% ● Si 152 ● No 6	Interpretación El 96% de los encuestados afirma que estaría dispuesto a utilizar la plataforma, evidenciando un alto nivel de aceptación y garantizando una adopción exitosa del sistema por parte de la comunidad universitaria.
Pregunta 7: ¿Cree que la plataforma debería ser accesible y usable desde dispositivos móviles (smartphones/tablets)?	
 5% 95% ● Si 150 ● No 8	Interpretación El 95% de los encuestados considera indispensable que la plataforma sea accesible desde dispositivos móviles. Este resultado establece el diseño responsivo (adaptable a celulares y tablets) como un requisito técnico prioritario para el desarrollo del sistema.

3.6.1.1.1 Análisis de las respuestas cualitativas

- **Pregunta 8: ¿Qué funciones considera indispensables para que el catálogo audiovisual de la futura plataforma web sea realmente útil para la comunidad universitaria?**

El análisis de las respuestas cualitativas revela que la búsqueda eficiente y la organización jerárquica son las funciones más valoradas. Los usuarios solicitan filtros avanzados (por etiquetas, fechas y áreas académicas) y herramientas de interacción como la descarga y difusión de contenido, definiendo así los módulos críticos para el desarrollo de la plataforma.

- **Pregunta 9: ¿Qué recomendaciones o ideas aportaría para mejorar la gestión y organización del contenido audiovisual del Club SoftMedia?**

Las recomendaciones de los usuarios trascienden la funcionalidad básica, sugiriendo un enfoque integral. Se destaca la necesidad de estandarizar la gestión mediante calendarios de publicación, metadatos y copias de seguridad. Asimismo, se reitera la importancia de la accesibilidad multiplataforma (móvil/web) y una interfaz intuitiva que facilite la interacción con el catálogo.

3.6.2 Presentación y descripción de los resultados obtenidos

En vista de los resultados obtenidos en la encuesta, se aprecia claramente una postura favorable por parte de la comunidad universitaria hacia el desarrollo de la plataforma web para el Club SoftMedia. En vista de los resultados obtenidos en la encuesta, se aprecia claramente una postura favorable por parte de la comunidad universitaria hacia el desarrollo de la plataforma web para el Club SoftMedia. Un primer dato que llama la atención es que solo el 64 % de los encuestados conocía al club antes de responder la encuesta. El 36 % restante no sabía de su existencia ni de lo que produce. Eso dice bastante: el problema no es únicamente de organización interna, sino también de visibilidad. Una plataforma accesible puede ayudar a que el trabajo del equipo llegue a más personas dentro de la propia universidad.

El dato que más directamente justifica este proyecto viene de la segunda pregunta: el 74 % de los participantes reconoció haber perdido tiempo buscando material multimedia institucional sin encontrarlo, o haber tenido dificultades serias en ese proceso. Casi tres de cada cuatro personas. Es un porcentaje difícil de ignorar. Esta carencia de un repositorio centralizado justifica plenamente la intervención tecnológica. Como respuesta a esta necesidad, en la tercera pregunta se observa que un 92% considera necesaria la

implementación de una plataforma donde se organice este contenido, validando la pertinencia del proyecto.

La viabilidad de la propuesta se confirma con una intención de uso masiva. Tanto en la disposición general como en la disponibilidad para estudiantes y docentes, más del 96% de los encuestados asegura que utilizaría la herramienta si estuviera disponible, lo que garantiza una alta tasa de adopción del sistema.

En cuanto a lo que se espera técnicamente del sistema, los números también hablan claro. El 95 % de los encuestados indicó que necesitaría acceder desde el celular, lo que hace que la adaptación de la interfaz a distintos tamaños de pantalla no sea un detalle opcional sino algo que el sistema tiene que resolver desde el principio. Y sobre el uso de inteligencia artificial para etiquetar los archivos de forma automática, el 96 % se mostró a favor. Prácticamente todos.

Esas cifras van en la misma dirección que lo que se recogió en las preguntas abiertas. Quienes respondieron con sus propias palabras pidieron, sobre todo, que la búsqueda funcione bien y que haya algún tipo de orden claro dentro del catálogo. No son peticiones contradictorias con lo que los números muestran; son, básicamente, lo mismo, dicho de otra forma.

3.6.3 Informe final del análisis de los datos

Los datos recogidos dejan poco margen para la duda. El club no cuenta con un lugar único donde se concentre y organice su material multimedia, y esa ausencia tiene consecuencias concretas: el 74 % de las personas encuestadas dijo haber perdido tiempo o enfrentado problemas al buscar registros de eventos anteriores. No es una percepción aislada; es algo que la mayoría ha vivido.

Frente a eso, desarrollar una plataforma que centralice el archivo del club no es una opción entre varias posibles, sino la respuesta más directa al problema detectado. El hecho de que sea la máquina la encargada de mirar las imágenes y asignarles una etiqueta, quitándole ese trabajo repetitivo a los estudiantes, marca un antes y un después en la forma en que opera el club, el ritmo de trabajo del equipo de una forma que difícilmente se lograría de otra manera. El diseño propuesto no se limita a almacenar archivos estáticos; los transforma en un ecosistema dinámico y adaptable a cualquier pantalla, cubriendo con exactitud la exigencia de movilidad que validó el 95% de la comunidad evaluada.

Finalmente, el uso de la encuesta reflejó de manera contundente la viabilidad y pertinencia del proyecto. Los datos demuestran una aceptación casi total, con una intención de uso superior al 96% por parte de estudiantes y docentes, quienes validan que dicho sistema mejorará sustancialmente la organización, la disponibilidad del contenido y la imagen institucional del Club SoftMedia dentro de la ULEAM.

CAPITULO IV

4 MARCO PROPOSITIVO

4.1 Introducción

Este capítulo presenta lo que se construyó para darle solución al problema de organización del Club SoftMedia de la ULEAM, extensión El Carmen. La propuesta es una plataforma web con un catálogo digital en su centro, pensada para que todo el material que el club genera en sus coberturas tenga un lugar definido donde guardarse y desde donde pueda recuperarse sin complicaciones. El sistema funciona desde cualquier dispositivo, ya sea celular o computadora, ajustando su interfaz según el tamaño de pantalla. Por el lado operativo, la directiva cuenta con una zona privada dentro de la web. Ahí es donde realmente se gestiona todo: se reparten los permisos de las cuentas y se filtra el material para elegir qué sale al público. Una gran ventaja de este proceso es que la persona encargada de revisar no pierde tiempo escribiendo descripciones para buscar las fotos después, ya que el motor de inteligencia artificial se ocupa de asignar esas palabras clave en cuanto el archivo toca el servidor.

Levantar todas estas herramientas tomó su tiempo, pero en lugar de encerrarse a programar todo de un tirón, el equipo decidió usar Scrum. Lo que hicieron fue partir el proyecto en cuatro bloques de trabajo cortos, de modo que cada vez que terminaban uno, ya tenían una parte del sistema lista para usarse y probarse con el club. Eso permitió ir ajustando el sistema sobre la marcha en lugar de esperar al final para ver si todo encajaba.

A lo largo del capítulo se explica cómo está armado el sistema por dentro: la estructura del servidor, la base de datos, las pantallas y fragmentos del código. No se trata de describir algo que se planificó, pero no se concretó; todo lo que aparece aquí fue construido, probado y responde directamente a lo que el diagnóstico mostró.

4.2 Descripción de la propuesta

La propuesta consiste en el desarrollo de una plataforma web centralizada para la administración eficiente de los contenidos audiovisuales generados por el Club SoftMedia. El desarrollo de este sistema se llevó a cabo mediante la metodología ágil Scrum, la cual permitió organizar el trabajo en ciclos iterativos denominados sprints, facilitando la entrega incremental de funcionalidades y la adaptación continua a los requerimientos del proyecto; su aplicación práctica se detalla en el apartado 4.4. Dentro del sistema, la gestión y el control de los archivos será de uso exclusivo para los

integrantes autorizados del departamento. Paralelamente, la plataforma brindará un entorno organizado para que la comunidad universitaria compuesta por estudiantes, docentes y personal administrativo pueda acceder, consultar y visualizar el material multimedia de forma estructurada.

La interfaz se adapta al dispositivo desde el que se acceda. No importa si es un celular o una computadora: la plataforma reorganiza su contenido para que la experiencia sea usable, en cualquier caso. El sistema se encarga de mostrar u ocultar cosas dependiendo de quién inicie sesión. Si alguien entra como simple visitante, la plataforma lo limita a mirar el material que la directiva ya aprobó. Puede buscar por fechas o eventos, pero hasta ahí llega; las fotos que recién se tomaron o que están a medio editar quedan fuera de su alcance.

Dentro del equipo la historia es distinta. Hay un área de trabajo exclusiva donde los administradores y supervisores toman el control. Ellos son los únicos que pueden mover los hilos fuertes de la página: dar de baja usuarios, modificar datos erróneos de algún evento, cambiar de estado las publicaciones o borrar algo definitivamente de los discos si se aprueba esa decisión.

Pero el verdadero salto frente a usar un simple Google Drive está en cómo se maneja el contenido nuevo. Cuando el fotógrafo sube cien fotos de golpe, nadie tiene que sentarse a escribir de qué trata cada una. El sistema usa un modelo de inteligencia artificial que revisa la imagen y le pega sus propias etiquetas descriptivas. Que la máquina se encargue de esto hace que la base de datos esté siempre ordenada y que buscar algo seis meses después tome apenas un par de segundos, en lugar de horas.

Todos los datos serán almacenados en una base de datos centralizada, garantizando la disponibilidad y actualización constante de la información. De esta manera, la plataforma permitirá optimizar la gestión de los recursos multimedia del Club SoftMedia, eliminando la dispersión de archivos y mejorando el acceso a la información para toda la comunidad universitaria.

4.3 Determinación de recursos

4.3.1 Humanos

Tabla 3: Determinación de recursos humanos

Rol	Nombre	Responsabilidades	Dedicación
Investigador / Desarrollador (Autor)	Jazmany Sumi Arévalo	Encargado del diseño, desarrollo técnico del módulo de catalogación con IA, gestión de base de datos y redacción documental de la tesis.	Alta (20 horas/semana)
Tutor de Proyecto	Ing. Jefferson Omar Arca Zavala	Supervisión técnica y metodológica, validación de la arquitectura propuesta y revisión de los entregables académicos.	Intermitente (2 horas/semana)
Equipo de Desarrollo (Integración)	5 estudiantes de Ingeniería en TI	Desarrollo de los módulos complementarios de la plataforma y coordinación técnica para asegurar la interoperabilidad del sistema modular.	Media (Sincronización semanal)
Administrador del Sistema	Coordinador de la carrera de Software y directiva técnica del club SofMedia	Control técnico total del módulo. Encargado de la edición de metadatos y la ejecución de borrados físicos permanentes en el servidor local para la liberación de espacio en disco.	Baja (Mantenimiento periódico)
Supervisor (Control de Calidad)	Personal Administrativo (Secretaría/Coordinación)	Ejecución de auditorías sobre el material en estado pendiente. Responsable de revisar y aprobar el contenido para su publicación, manteniendo una restricción estricta que impide la descarga del recurso original.	Media (Fases de revisión activa)

Integrante (Generador de Contenido)	Camarógrafos, fotógrafos y editores	Subida de material multimedia en crudo para su análisis y etiquetado inmediato mediante IA. No poseen privilegios del sistema para auto-aprobar o publicar su propio contenido.	Alta (Trabajo de campo y carga)
Invitado (Usuario Externo)	Comunidad Universitaria y Público General	Acceso sin requerimiento de credenciales a la galería pública de recursos con estado "Aprobado". Uso exclusivo del motor de búsqueda semántica, con bloqueos totales de edición y descarga de archivos.	Baja (Uso eventual/consulta)

4.3.2 Tecnológicos

Tabla 4: Determinación de recursos Tecnológicos Hardware

Recurso	Descripción
Hardware (Estación de Trabajo)	-Laptop Lenovo Ideapad Gaming 3. -Procesador AMD Ryzen 5 4800H. -GPU NVIDIA GTX 1650 (4GB). -16GB RAM. -Almacenamiento dual (1TB SSD M.2 + 512GB SSD M.2). -Windows 11 x64 (Versión 25H2).
Hardware (Dispositivo de Prueba)	Smartphone Google Pixel 7 Pro. 128GB de almacenamiento, 12GB RAM, sistema operativo Android 15.
Conectividad	Servicio de Internet Alfabet (350 Mbps de descarga / 300 Mbps de subida).
Lenguajes y Frameworks	• Python • FastAPI • JavaScript • React • PostgreSQL • Docker / Docker Compose
Asistencia Integrada	Gemini Pro.
Software (Desarrollo)	Entorno de Desarrollo Integrado (IDE) (ej. Visual Studio Code) y Sistema de Control de Versiones (Git/GitHub).
Software (Servicios Cloud / IA)	API de Inteligencia Artificial (BLIP).

4.3.3 Recursos Económicos

Tabla 5: Determinación de recursos económicos

Ítem / Descripción	Tipo de Financiamiento	Costo Estimado (USD)
Computadora portátil	Aporte propio del investigador	\$ 750.00
Smartphone Android	Aporte propio del investigador	\$ 350.00
Servicio de Internet (Fase de desarrollo)	Aporte propio del investigador	\$ 90.00 (3 meses a \$30)
Combustible	Aporte propio	\$ 300.00
Alimentación	Aporte propio	\$ 42.00
Programación 325 horas	Aporte propio	\$ 4,875
TOTAL, ESTIMADO		\$ 6,407.00

4.4 Desarrollo bajo la metodología Scrum

El desarrollo del módulo de catalogación audiovisual y su acople a la plataforma principal del Club SoftMedia se ejecutó bajo los lineamientos de la metodología ágil Scrum. A nivel teórico, Schwaber y Sutherland (2020) definen este marco como un entorno de trabajo diseñado específicamente para generar soluciones en escenarios complejos. En la práctica, elegir Scrum no fue solo una decisión metodológica: era necesario. El proyecto lo estaban desarrollando seis estudiantes al mismo tiempo, cada uno trabajando sobre una parte distinta del sistema. Sin una forma clara de coordinar esos avances, era muy probable que los módulos no encajaran bien entre sí al momento de unirlos. En este contexto, Scrum proporcionó la estructura organizativa exacta para mantener los avances del equipo sincronizados, previniendo conflictos a nivel de código y garantizando que las bases de datos encajaran correctamente al momento de unificar toda la arquitectura.

A diferencia del método tradicional en cascada, donde el sistema se revisa en su totalidad al final del proceso, el trabajo se dividió en ciclos cortos de programación llamados "Sprints". Según los mismos autores Schwaber y Sutherland (2020), los Sprints representan el corazón de Scrum, ya que permiten realizar entregas funcionales de forma progresiva. En la práctica, esto significó que durante un primer ciclo se construyó y validó

el formulario básico de carga multimedia, dejando para un Sprint posterior la integración de la inteligencia artificial encargada del etiquetado automático.

Trabajar bajo este esquema también aseguró que el sistema final respondiera a los requerimientos del Club SoftMedia. Al realizar entregas continuas, los directivos y miembros de la agrupación pudieron observar el avance de las interfaces y evaluar las funciones sobre la marcha. Esta retroalimentación constante permitió ajustar la plataforma para hacerla más intuitiva.

4.4.1 Descripción del Producto

4.4.1.1 Propósito del Producto

El propósito central de este módulo fue dar una solución definitiva a la desorganización y dispersión del material audiovisual del Club SoftMedia. Si miramos cómo se guardan las cosas ahora, la mayor ventaja es que la web funciona como una bóveda única. El fotógrafo descarga sus tomas pesadas ahí mismo, y más tarde el editor sube su video final al mismo lugar. Al tener todo concentrado bajo un solo techo y con un buen control de usuarios, se corta de raíz el viejo problema de tener la misma foto repetida en tres carpetas distintas o, peor aún, archivos que nadie sabe en qué computadora quedaron guardados.

Pero el cambio que de verdad nota el equipo en el día a día es otro: al subir un lote nuevo, ya nadie tiene la obligación de quedarse un rato largo escribiendo títulos y etiquetas para cada foto. Al ingresar un archivo, el sistema inspecciona el material y genera metadatos semánticos de forma autónoma. Esta intervención técnica suprime por completo las antiguas prácticas de catalogación, optimizando el rendimiento de las consultas y garantizando que el equipo recupere cualquier secuencia fotográfica de eventos pasados sin experimentar los clásicos retrasos logísticos. Esta automatización facilitó que los recursos gráficos y de video pudieran ser encontrados y reutilizados rápidamente en nuevas producciones institucionales.

Además de facilitar la búsqueda, el módulo se estructuró para soportar un flujo de trabajo profesional. Se incorporó un sistema de revisión jerárquico donde un perfil supervisor puede examinar y aprobar el contenido final antes de su publicación o almacenamiento oficial. Con esto, se logró no solo centralizar la información, sino garantizar un control de calidad y orden en todos los activos digitales de la agrupación.

4.4.1.2 Funcionalidades Clave

Para erradicar definitivamente la fragmentación documental en la agrupación, la arquitectura del software despliega un núcleo de operaciones críticas:

- Carga de metraje y persistencia centralizada: A nivel de interfaz, se programó un entorno web dedicado para que los creadores transfieran sus archivos directamente hacia el servidor local. Esta logística de red asegura el resguardo concurrente tanto de las capturas fotográficas nativas sin compresión como de las secuencias de video ya renderizadas. Al estructurar este canal de subida, se garantiza que el repositorio crezca de manera autónoma y mantenga la trazabilidad física de todos los proyectos institucionales.
- Etiquetado mediante Inteligencia Artificial: Al momento de ingresar una imagen o video, el sistema analizó el archivo mediante el consumo de una API de inteligencia artificial. Esta tecnología asignó palabras clave de forma automática, creando los metadatos necesarios para el catálogo sin requerir que los usuarios escribieran las descripciones manualmente.
- Interoperabilidad con el módulo de tareas: Se desarrolló una conexión directa, mediante procesos internos, con el módulo de planificación del equipo. De esta manera, cuando se adjuntaron imágenes en crudo a la asignación de un editor, los archivos se guardaron y respaldaron automáticamente en la bodega central. Así, el editor pudo acceder a su material desde su propio panel, alimentando el repositorio general en segundo plano.
- Buscador inteligente de apoyo: Se implementó un motor de búsqueda basado en los metadatos generados por la IA. Esto facilitó que los editores o creadores de contenido localizaran rápidamente material de apoyo de eventos pasados cuando lo requerían para complementar sus nuevos proyectos.
- Flujo de revisión y gestión de roles: Se configuró un panel de control con distintos niveles de acceso. Se incluyó un perfil jerárquico de supervisor, encargado de revisar el contenido o proyectos terminados para dar su aprobación final antes de la publicación. Asimismo, se restringió el acceso para que los invitados únicamente pudieran visualizar las galerías públicas, protegiendo los activos internos de la agrupación.

4.4.1.3 Usuarios Objetivo

Tabla 6: Usuarios objetivo

Tipo de Usuario	Descripción del Perfil	Principales Funcionalidades
Administrador	Responsable técnico de la plataforma con control total sobre el entorno de despliegue.	Ejecuta la administración de las credenciales de acceso, calibra los parámetros de la interfaz de inteligencia artificial, realiza el mantenimiento físico de la base de datos y audita la totalidad de los archivos transferidos al servidor
Supervisor	Especialista encargado de ejecutar el control de calidad y auditar la producción multimedia generada por la agrupación.	Su alcance operativo le permite visualizar los metrajes en estado de revisión, redactar correcciones técnicas y emitir la aprobación definitiva del contenido.
Integrante	Alumno activo en funciones de fotógrafo, camarógrafo o editor, responsable de generar el material primario.	Subida manual y automatizada de archivos (incluyendo formatos pesados y RAW), búsqueda inteligente de recursos mediante IA, y descarga de material en alta calidad.
Invitado	Público externo o comunidad universitaria interesada en el registro de los eventos.	Acceso únicamente a las galerías de contenido ya publicado y finalizado. No requiere privilegios de edición ni acceso al material interno en crudo.

4.4.1.4 Condiciones de Éxito del Producto

Para medir la efectividad del módulo al finalizar los Sprints de desarrollo, se establecieron los siguientes criterios técnicos y operativos como condiciones de éxito: Lograr una precisión del 90% o superior en el reconocimiento y etiquetado automático de imágenes mediante la Inteligencia Artificial.

- Procesar la subida de archivos y generar sus metadatos con IA en menos de 10 segundos por elemento (dependiendo del peso del archivo y la estabilidad de la conexión).

- Asegurar que al menos el 98% del material pase correctamente por el flujo de revisión, impidiendo que el público visualice contenido sin la aprobación previa del perfil Supervisor.
- Ofrecer un buscador inteligente que recupere y muestre resultados basados en etiquetas visuales en un tiempo máximo de 5 segundos, contemplando las posibles latencias o lentitud de la red universitaria.
- Los archivos deben conservarse en el servidor tal como fueron subidos, sin pérdida de calidad. Esto aplica tanto para fotografías en su resolución original como para los videos ya editados, que deben poder previsualizarse directamente desde la plataforma sin necesidad de descargarlos primero.
- El catálogo debe funcionar con normalidad en los navegadores que habitualmente usan los estudiantes y docentes de la extensión. Que se carguen varias imágenes al mismo tiempo no debería afectar la estabilidad de la interfaz ni provocar fallos visuales.
- Arquitectura gráfica responsiva: Desplegar una interfaz que reorganice sus componentes de manera autónoma según las dimensiones físicas de la pantalla. Esta configuración valida la accesibilidad operativa tanto en monitores de escritorio como en teléfonos celulares.

4.4.2 Historias de Usuario

4.4.2.1 Historia de Usuario – Consumo y validación de sesión centralizada.

Tabla 7: Historia de usuario 1: Autenticación y Gestión de Usuario

HISTORIAL DE USUARIO			
Id de historia:	HU01	Título:	Consumo y validación de sesión centralizada.
Prioridad:	Alta	Usuario:	Administrador, Supervisor, Integrante e invitado
Riesgo de desarrollo:	Alta	Tamaño de la tarea:	Grande
Como:	Administrador, Supervisor, integrante e invitado		
Quiero:	Que el catálogo multimedia reconozca y valide mi sesión de forma automática al ingresar a este módulo.		
Para:	Ingresar a las herramientas operativas del catálogo sin necesidad de duplicar la validación de credenciales durante las transiciones de pantalla.		
Criterios de aceptación:	• El entorno cliente intercepta la firma criptográfica de la sesión directamente desde el almacenamiento del navegador de forma transparente.		

	• El motor de la plataforma decodifica los permisos del usuario para condicionar la renderización de las interfaces gráficas correspondientes a su nivel de acceso. • Ante la caducidad del tiempo asignado al token, la arquitectura fuerza una redirección inmediata hacia la pantalla de autenticación central.
Dependencias:	API del Módulo de Seguridad central, esquema de roles globales en la base de datos y persistencia de estado en el lado del cliente.

4.4.2.2 Historia de usuario - Carga y Etiquetado Automático

Tabla 8: Historia de usuario - Carga y Etiquetado Automático

HISTORIAL DE USUARIO			
Id de historia:	HU02	Título:	Carga de material en crudo y etiquetado IA.
Prioridad:	Alta	Usuario:	Integrante de SoftMedia.
Riesgo de desarrollo:	Alta	Tamaño de la tarea:	Grande
Como:	Integrante activo del departamento audiovisual		
Quiero:	Seleccionar múltiples archivos de imagen y video desde mi dispositivo, asignarles un título de lote descriptivo y subirlos al repositorio central, de manera que el sistema analice cada archivo con inteligencia artificial y genere las etiquetas descriptivas de forma automática sin que yo tenga que escribir ninguna descripción manual		
Para:	Alimentar el catálogo audiovisual del club con material de eventos y proyectos de manera ágil, reduciendo el tiempo de catalogación y asegurando que todo el contenido quede indexado para su posterior localización mediante el buscador inteligente		
Criterios de aceptación:	• El entorno permite seleccionar y transferir lotes concurrentes de hasta 50 archivos. Al confirmar la acción, la arquitectura guarda el documento físico e inmediatamente encola el procesamiento algorítmico en segundo plano. • Una vez finalizado el análisis del motor, las etiquetas extraídas se registran de forma estructurada en la base de datos y aparecen asociadas al recurso dentro de la galería. • El creador tiene habilitada la opción para cambiar el estado de sus propios registros a "Listo", indicando que el material está preparado para someterse a la auditoría de control de calidad. • La regla de negocio bloquea estrictamente los permisos del integrante, impidiendo que apruebe o		

	publique sus propios recursos directamente en el catálogo público.
Dependencias:	Base de datos del catálogo (audiovisual.archivos), Directorio físico de almacenamiento del servidor (uploads), Motor de Inteligencia Artificial (Salesforce BLIP).

4.4.2.3 Historia de usuario - Auditoría y Aprobación

Tabla 9: Historia de usuario - Auditoría y Aprobación

HISTORIAL DE USUARIO			
Id de historia:	HU03	Título:	Revisión, comentarios y publicación de contenido.
Prioridad:	Alta	Usuario:	Supervisor.
Riesgo de desarrollo:	Alta	Tamaño de la tarea:	Grande
Como:	Supervisor del control de calidad.		
Quiero:	Visualizar, comentar y aprobar el material en estado "listo/casi listo",		
Para:	Autorizar su publicación en la galería pública		
Criterios de aceptación:	- El supervisor accede a una vista filtrada que muestra exclusivamente los archivos con estado "Listo", "Publicado" y "Oculto". - Puede aprobar un archivo (transición a "Publicado") u ocultarlo (transición a "Oculto"). No tiene la capacidad de marcar archivos como "Listo" ni de restaurarlos a "Pendiente". - La información de autoría (nombre del integrante que subió el archivo) se muestra en la vista del Supervisor para facilitar la trazabilidad.		
Dependencias:	Sistema de autenticación JWT con validación de rol, tabla de archivos con columna de estado, (FastAPI), sistema de renderizado condicional en el frontend (React).		

4.4.2.4 Historia de usuario - Gestión y Depuración Total

Tabla 10: Historia de usuario - Gestión y Depuración Total

HISTORIAL DE USUARIO			
Id de historia:	HU04	Título:	Mantenimiento y control total del catálogo.
Prioridad:	Alta	Usuario:	Administrador.
Riesgo de desarrollo:	Alta	Tamaño de la tarea:	Grande
Como:	Administrador general		

Quiero:	Tener acceso completo a todos los archivos del catálogo, visualizar la autoría de cada elemento, gestionar los estados del contenido (excepto la publicación directa), subir material propio y eliminar permanentemente los archivos que ya no sean necesarios, siempre que no se encuentren en estado "Publicado".
Para:	Mantener la integridad y el orden del repositorio audiovisual, depurando el contenido obsoleto o innecesario sin comprometer el material que ya fue aprobado y publicado por el Supervisor
Criterios de aceptación:	-Visualiza botones de edición y eliminación en el contenido sin publicar. -Puede borrar archivos del disco duro del servidor de forma permanente. -Puede anular bloqueos o alterar etiquetas si el sistema lo requiere.
Dependencias:	Base de datos del catálogo, Permisos de escritura/eliminación en el sistema operativo del servidor local, Verificación estricta de tokens JWT.

4.4.2.5 Historia de usuario - Búsqueda y Visualización Pública

Tabla 11: Historia de usuario - Búsqueda y Visualización Pública

HISTORIAL DE USUARIO			
Id de historia:	HU05	Título:	Búsqueda y Visualización Pública
Prioridad:	Media	Usuario:	Invitado
Riesgo de desarrollo:	Bajo	Tamaño de la tarea:	Pequeño
Como:	Invitado (comunidad universitaria) sin cuenta activa en el sistema		
Quiero:	Navegar por la galería pública del club, visualizar las imágenes y videos que ya fueron aprobados por el Supervisor, y utilizar el buscador inteligente para localizar contenido por etiquetas o nombre de archivo		
Para:	Buscar y ver fotos o videos de eventos sin necesidad de registrarme.		
Criterios de aceptación:	• El invitado accede a la galería sin autenticación. • Solo ve los archivos con estado "Publicado". • No tiene acceso a funciones de subida, descarga, cambio de estado ni eliminación. • No ve la información de autoría de los archivos. • La galería muestra un banner informativo indicando que se encuentra en modo de solo lectura. • El buscador predictivo funciona sobre las etiquetas generadas por la IA para el contenido publicado.		

Dependencias:	Endpoint GET /files/ con decodificación opcional del JWT (decodificar_token_opcional), consulta SQL filtrada por estado "Publicado".
---------------	--

4.4.3 Diseño del Sistema / Descripción Técnica

4.4.3.1 Caso de uso: Ingresar al Módulo Audiovisual

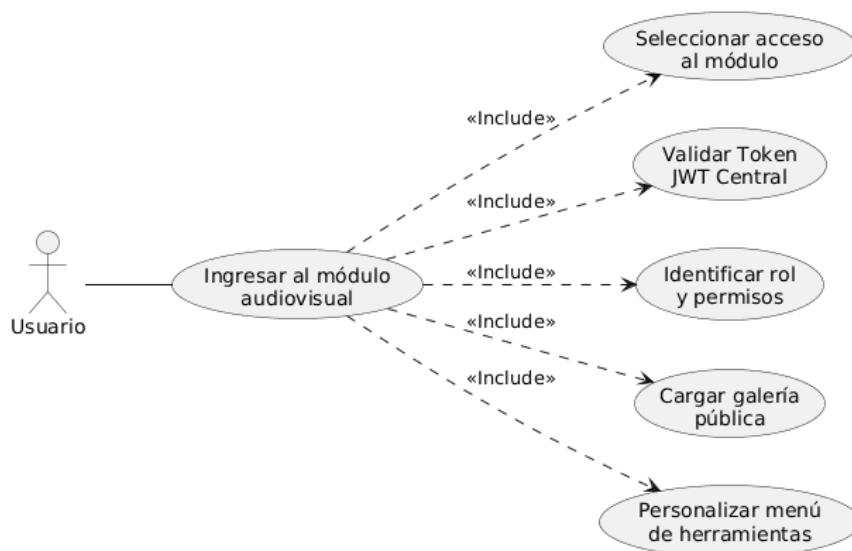


Ilustración 3: Diagrama de Casos de Uso: Ingresar al Módulo Audiovisual

4.4.3.2 Caso de uso: Ingresar al Módulo Audiovisual

Tabla 12: Caso de uso: Ingresar al Módulo Audiovisual

DOCUMENTACIÓN DEL CASO DE USO			
Caso de uso N° 001:	UC-001	Nombre del caso de uso:	Ingresar al Módulo Audiovisual
Fecha:	13/06/2026	Elaborado por:	Jazmany Sumi Arévalo
Actores:	Usuario Registrado (Admin, Supervisor, Integrante) / Invitado		
Objetivo:	Validar el acceso al catálogo audiovisual y personalizar la interfaz según el rol detectado mediante el sistema de tokens centralizado.		
Precondiciones:	Sesión iniciada en plataforma central (para registrados); Conexión estable a red institucional.		
Postcondiciones:	Interfaz cargada con permisos correspondientes; Acceso habilitado a herramientas de gestión o solo visualización pública.		
Medios para Registrar solicitud:	Plataforma Web Modular (React.js)		
Pasos			
• El usuario selecciona el acceso al módulo audiovisual. • El sistema detecta y valida el Token JWT central. • El sistema decodifica el rol del usuario (Admin/Sup/Int).			

• El sistema adapta la interfaz y habilita funciones específicas. • El usuario accede a la galería y barra de búsqueda.	
Situaciones excepcionales: Token JWT expirado o corrupto: Redirigir a Login Central. Fallo de conexión con servicio de autenticación. Acceso nulo: Forzar perfil de Invitado.	
Revisado por:	Ing. Jefferson Omar Arca Zavala Mg.

4.4.3.3 Caso de uso: Subir contenido audiovisual

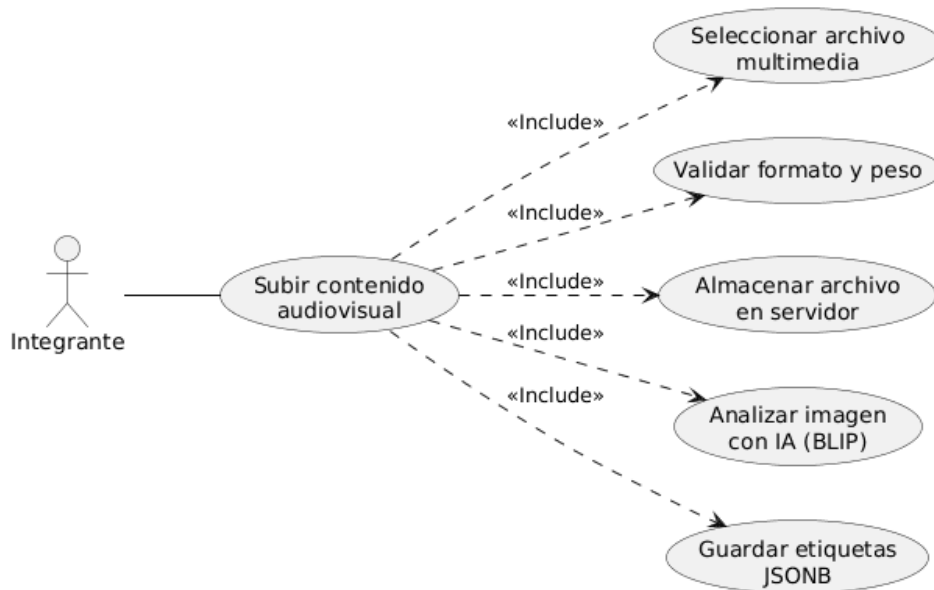


Ilustración 4: Caso de uso: Subir contenido audiovisual

4.4.3.4 Caso de uso: Subir contenido audiovisual

Tabla 13: Caso de uso: Subir contenido audiovisual

DOCUMENTACIÓN DEL CASO DE USO			
Caso de uso N° 002:	UC-002	Nombre del caso de uso:	Subir contenido audiovisual
Fecha:	13/06/2026	Elaborado por:	Jazmany Sumi Arévalo
Actores:	Integrante y Administrador		
Objetivo:	Permitir al integrante cargar material multimedia en crudo al sistema para su almacenamiento y generar automáticamente etiquetas descriptivas mediante el modelo de Inteligencia Artificial.		
Precondiciones:	El usuario debe haber ingresado al módulo con el rol de Integrante activo o administrador; El modelo de IA (BLIP) debe estar inicializado en el servidor.		
Postcondiciones:	El archivo queda almacenado físicamente en el servidor local; Las etiquetas extraídas se registran de forma estructurada (JSONB) en la base de datos.		
Medios para Registrar solicitud:	Interfaz Web (React.js) / API Backend (FastAPI)		

Pasos	
• El usuario ingresa a la interfaz de carga masiva, define el título de la sesión y transfiere los documentos multimedia hacia el entorno de recepción. • El sistema inspecciona la extensión temporal de los metrajes. Si el recurso excede los cinco minutos, la plataforma requiere la asignación manual de metadatos, excluyendo dicho archivo del motor algorítmico. • El usuario audita la galería de previsualización, descarta las capturas no deseadas y ejecuta la confirmación de transferencia. • La arquitectura cierra la vista inmediatamente, persiste el documento en el almacenamiento local e inserta el registro en PostgreSQL bajo el estado de procesamiento, permitiendo la navegación continua. • El backend desencadena una rutina asíncrona en segundo plano que envía el contenido visual al modelo de Inteligencia Artificial para extraer las predicciones nativas. • El algoritmo traduce el arreglo semántico, descarta los términos irrelevantes y guarda las etiquetas definitivas en la estructura JSONB de la base de datos modificando el estado del recurso a pendiente de revisión.	
Situaciones excepcionales: • Incompatibilidad de extensión: Ante el intento de procesar formatos ajenos al entorno audiovisual, el cliente interrumpe la subida y despliega una alerta detallando las extensiones admitidas. • Saturación de concurrencia: Para proteger la memoria del servidor local, la capa lógica bloquea las transferencias simultáneas que excedan el límite de cuarenta fotografías o quince recursos de video. • Restricción de carga útil: Aquellos elementos individuales que sobrepasen la capacidad máxima asignada (50 MB para imágenes y 3 GB para videos) son descartados autónomamente de la cola de procesamiento. • Falta de contexto obligatorio: Si el usuario omite la catalogación manual en metrajes extensos, la interfaz gráfica bloquea el botón de envío hasta solventar la dependencia de texto. • Intercepción de errores críticos: Frente a un documento dañado o un fallo en el cálculo algorítmico, el servidor captura la excepción, le asigna al registro una etiqueta de advertencia técnica y continúa procesando el resto del lote sin interrumpir el ciclo general.	
Revisado por:	Ing. Jefferson Omar Arca Zavala Mg.

4.4.3.5 Caso de uso: Auditar y Aprobar Contenido

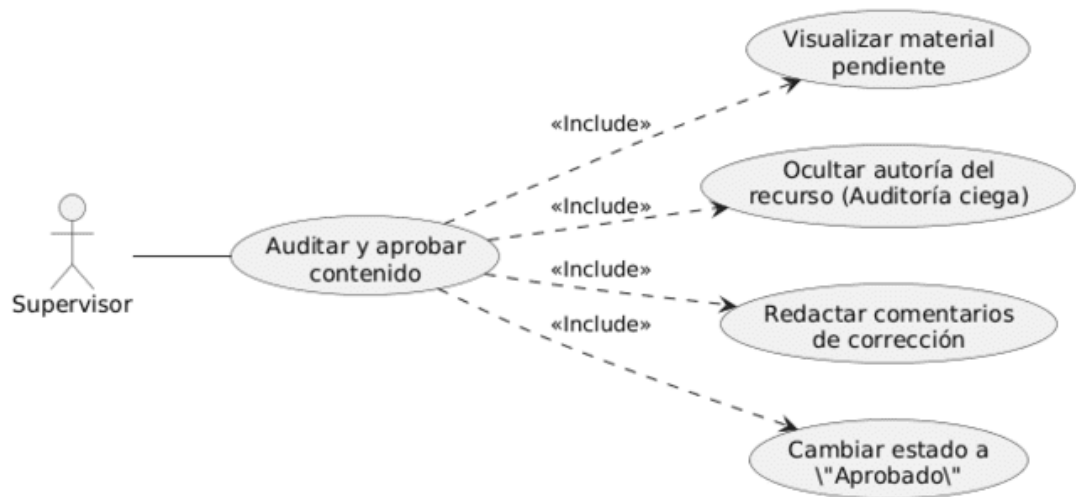


Ilustración 5: Caso de uso: Auditar y Aprobar Contenido

4.4.3.6 Caso de uso: Auditar y Aprobar Contenido

Tabla 14: Caso de uso: Auditar y Aprobar Contenido

DOCUMENTACIÓN DEL CASO DE USO			
Caso de uso N° 003:	UC-003	Nombre del caso de uso:	Auditar y aprobar contenido
Fecha:	13/06/2026	Elaborado por:	Jazmany Sumi Arévalo
Actores:	Supervisor		
Objetivo:	Permitir al supervisor revisar el material audiovisual "listo", realizar observaciones o autorizar su publicación definitiva.		
Precondiciones:	El usuario debe estar autenticado con el rol de Supervisor; Deben existir archivos multimedia en estado “listo” en el catálogo.		
Postcondiciones:	Archivo con estado actualizado a "Publicado" (visible para invitados) o "Oculto" (retirado de la vista pública)		
Medios para Registrar solicitud:	Panel de Revisión en React.js / Endpoints de actualización en FastAPI.		
Pasos			
• Iniciar sesión con credenciales de Supervisor • Navegar a la galería que muestra automáticamente los archivos con estado • "Listo", "Publicado" y "Oculto" • Seleccionar un archivo para ampliarlo en el visor Lightbox • Revisar la imagen o video, las etiquetas de IA y la información de autoría • Presionar "Aprobar" para publicar o "Ocultar" para retirar el contenido • El sistema actualiza el estado en la base de datos y refresca la galería			
Situaciones excepcionales:			
Intento de descarga forzada: El backend bloquea la petición HTTP y el frontend no renderiza ninguna opción de descarga para este rol. Fallo de actualización: Si el servidor pierde conexión al guardar el cambio de estado, se cancela la acción y se notifica al supervisor. Contenido en crudo: El sistema oculta por completo los archivos que no han sido pre-procesados por los integrantes.			
Revisado por:	Ing. Jefferson Omar Arca Zavala Mg.		

4.4.3.7 Caso de uso: Buscar contenido audiovisual

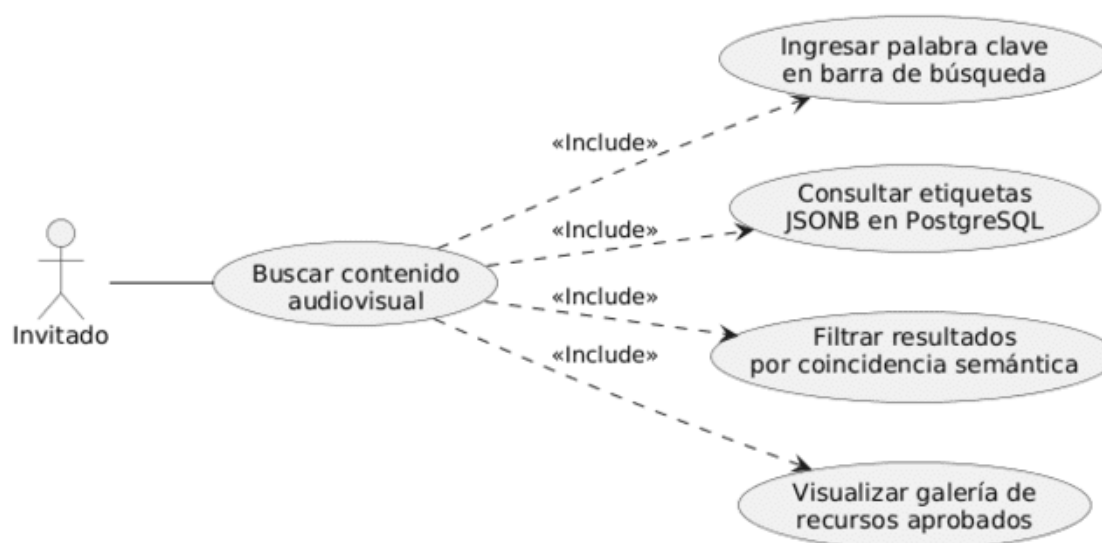


Ilustración 6: Caso de uso: Buscar contenido audiovisual

4.4.3.8 Caso de uso: Buscar contenido audiovisual

Tabla 15: Caso de uso: Buscar contenido audiovisual

DOCUMENTACIÓN DEL CASO DE USO			
Caso de uso N° 004:	UC-004	Nombre del caso de uso:	Buscar contenido audiovisual
Fecha:	14/06/2026	Elaborado por:	Jazmany Sumi Arévalo
Actores:	Invitado (Público externo / Comunidad universitaria)		
Objetivo:	Permitir al usuario realizar consultas mediante palabras clave en la barra de búsqueda para localizar y visualizar recursos multimedia aprobados de manera ágil.		
Precondiciones:	El catálogo debe contar con registros multimedia indexados y en estado "Publicado" dentro de la base de datos.		
Postcondiciones:	Se despliega en la interfaz la grilla de elementos filtrados que coinciden con el criterio de búsqueda semántica.		
Medios para Registrar solicitud:	Componente de búsqueda en React.js / Filtros de consulta en FastAPI.		
Pasos			
• Acceder a la plataforma mediante el botón "Entrar como Invitado" • Visualizar la galería con el contenido aprobado por el Supervisor • Utilizar la barra de búsqueda predictiva para filtrar por etiquetas • Aplicar filtros rápidos ("Fotos", "Videos") • Hacer clic en un archivo para abrirlo en el visor Lightbox • Navegar entre archivos usando las flechas o las teclas del teclado			
Situaciones excepcionales:			
• Sin Carencia de coincidencias semánticas: Cuando el término introducido no logra emparejarse con los metadatos almacenados, el algoritmo limpia inmediatamente la galería visual y emite un aviso indicando la nulidad de resultados.			

• Ausencia de parámetros de consulta: Al suprimir manualmente los caracteres en la barra de exploración, el entorno cliente recarga la cuadrícula de manera autónoma, restableciendo la vista para exponer la totalidad del catálogo público sin aplicar filtros. • Fallo de lectura local: Frente a una interrupción al consultar la estructura JSONB dentro de PostgreSQL, el servidor captura la excepción y la interfaz gráfica renderiza una alerta técnica, previniendo el colapso abrupto de la pantalla de invitado.	
Revisado por:	Ing. Jefferson Omar Arca Zavala Mg.

4.4.3.9 Caso de uso: Gestionar y depurar catálogo

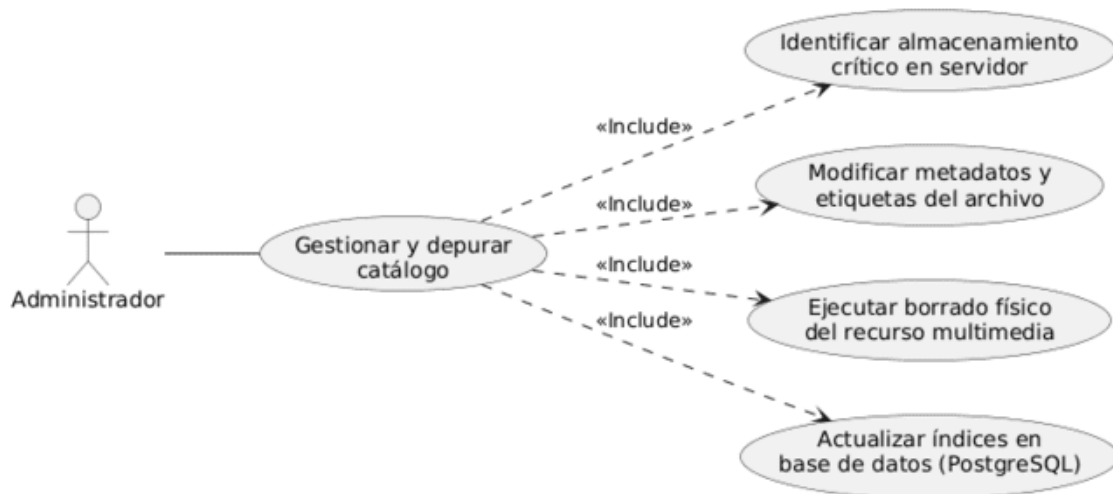


Ilustración 7: Caso de uso: Gestionar y depurar catálogo

4.4.3.10 Caso de uso: Gestionar y depurar catálogo

Tabla 16: Caso de uso: Gestionar y depurar catálogo

DOCUMENTACIÓN DEL CASO DE USO			
Caso de uso N° 005:	UC-005	Nombre del caso de uso:	Gestionar y depurar catálogo
Fecha:	14/06/2026	Elaborado por:	Jazmany Sumi Arévalo
Actores:	Administrador (Responsable técnico de la plataforma)		
Objetivo:	Proporcionar al administrador las herramientas de gestión completa del catálogo, incluyendo la capacidad de depurar contenido obsoleto sin comprometer el material ya publicado.		
Precondiciones:	El usuario debe estar autenticado con el rol de Administrador mediante un Token JWT válido; El servidor debe tener los permisos de sistema operativo habilitados para borrado físico.		
Postcondiciones:	Archivo eliminado física y lógicamente del sistema (cuando aplica), o estado del archivo actualizado según la transición solicitada.		
Medios para Registrar solicitud:	Consola de Administración en React.js / Endpoints de depuración en FastAPI.		
Pasos			
• Iniciar sesión con credenciales de Administrador			

- Visualizar la galería completa con todos los archivos y su autoría
- Seleccionar uno o varios archivos para gestionar
- Para cambiar estado: presionar el botón de acción correspondiente
- (Listo, Ocultar, Restaurar)
- Para eliminar: presionar el botón de borrado (solo si el estado no es
- "Publicado"). El sistema solicita confirmación mediante la escritura
- de la palabra "ELIMINAR"
- Para descarga masiva: seleccionar múltiples archivos y presionar
- "Descargar todo" en la barra de acciones masivas

Situaciones excepcionales:

El usuario accede a la función de gestionar y depurar catálogo. El sistema le permite identificar almacenamiento crítico en servidor mediante indicadores de peso. Si se requiere una corrección, el usuario procede a modificar metadatos y etiquetas del archivo. Si el archivo es obsoleto, el usuario procede a ejecutar borrado físico del recurso multimedia. Finalmente, el sistema se encarga de actualizar índices en base de datos (PostgreSQL) de forma automática.

Revisado por:

Ing. Jefferson Omar Arca Zavala Mg.

4.4.3.11 Diagrama de base de datos

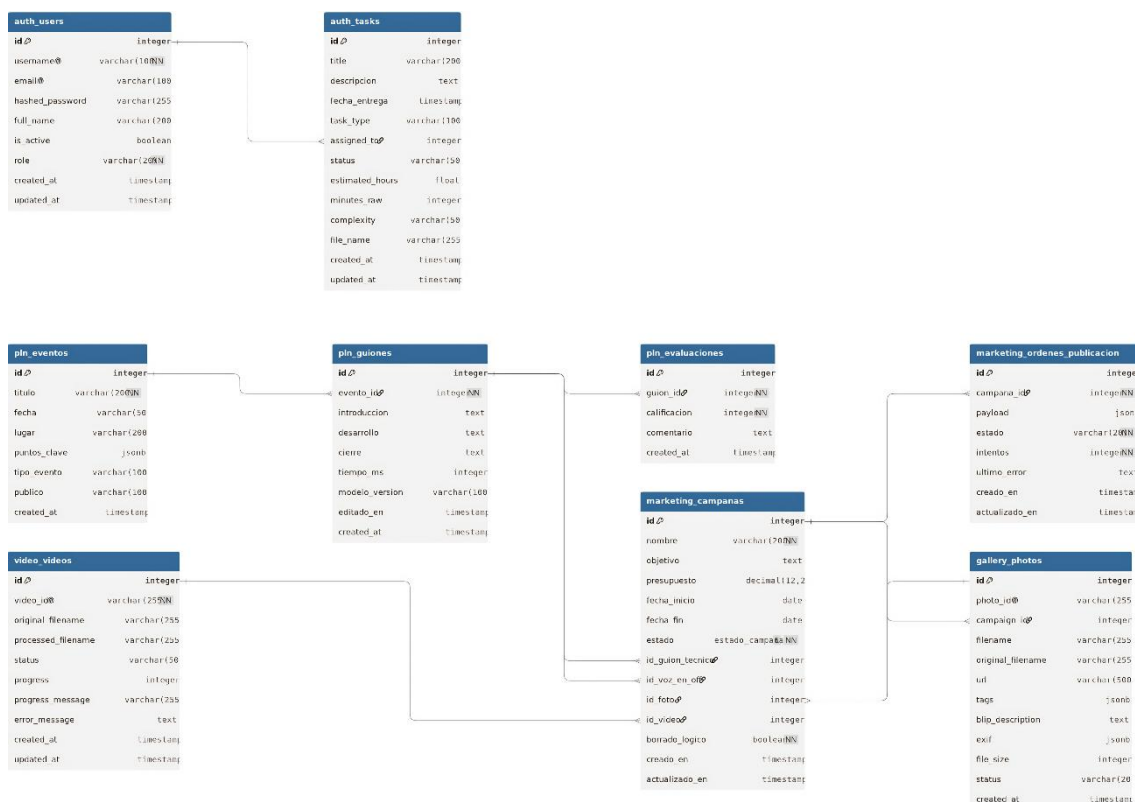


Ilustración 8: Diagrama de Base de datos

4.4.4 Diagramas de Secuencia

4.4.4.1 Diagrama de secuencia: Ingresar al módulo

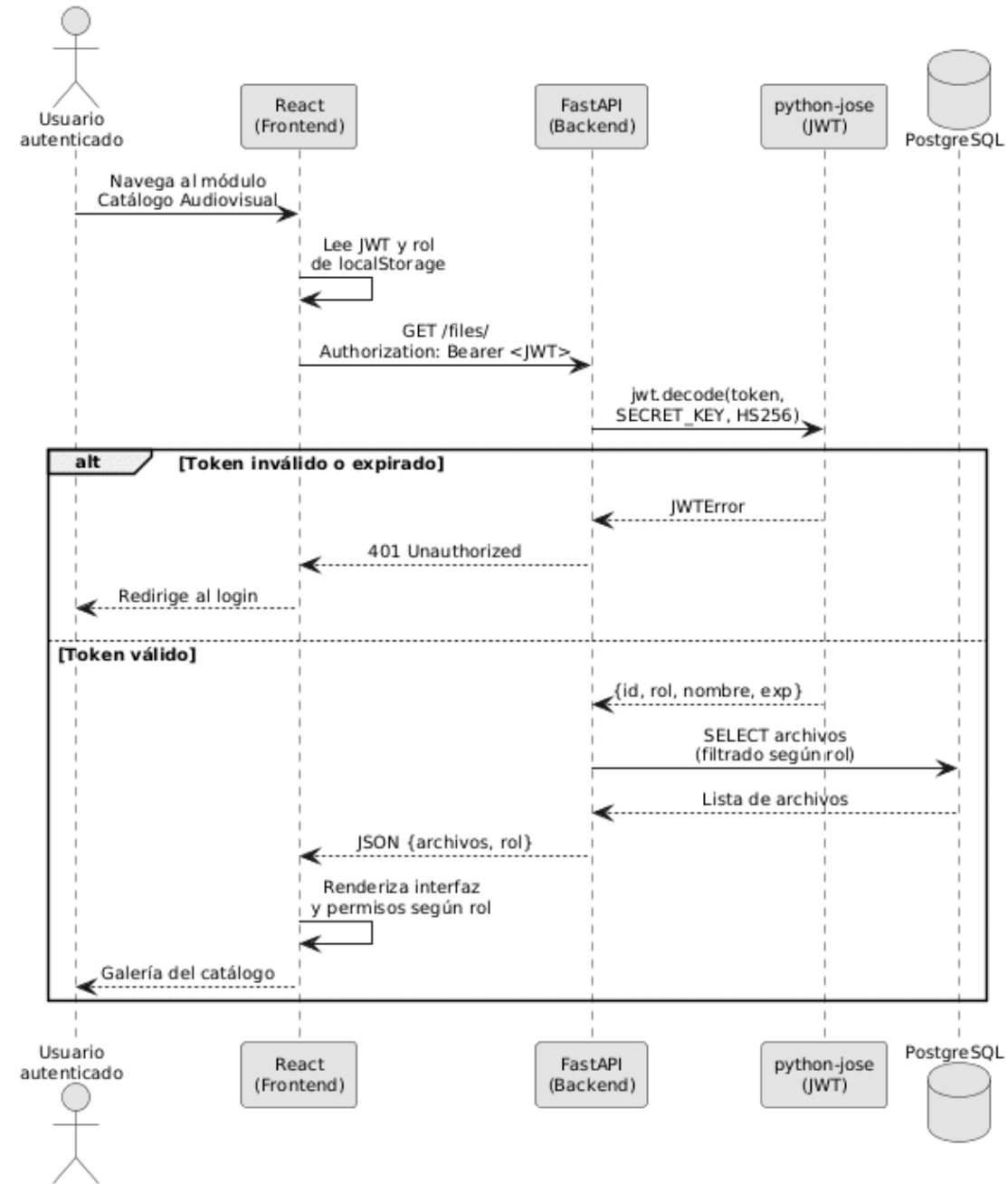


Ilustración 9: Diagrama de secuencia: Ingresar al módulo

4.4.4.2 Diagrama de secuencia: Carga masiva de archivos

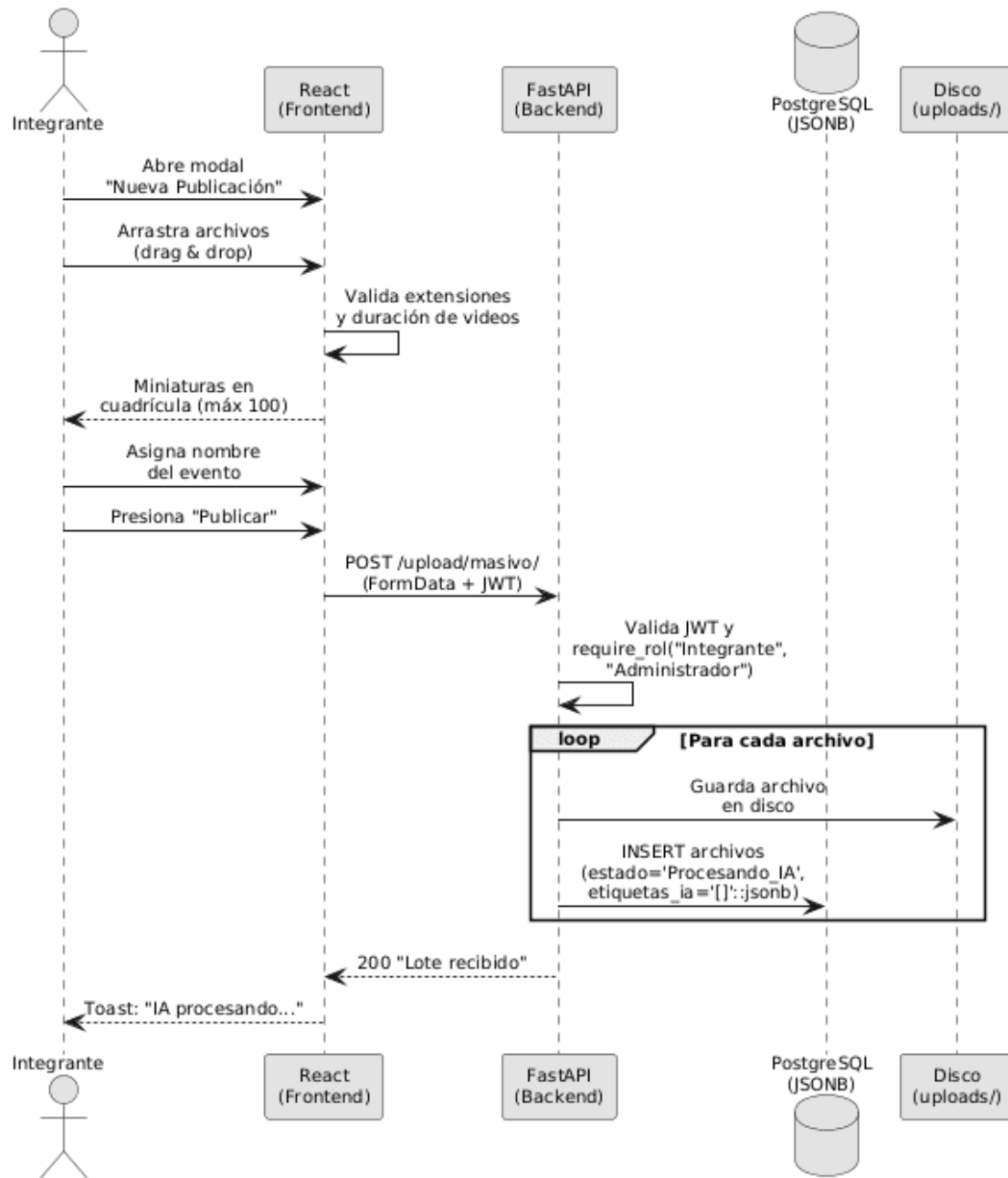


Ilustración 10: Diagrama de secuencia: Carga masiva de archivos

4.4.4.3 Diagrama de secuencia: Procesamiento de IA (BLIP)

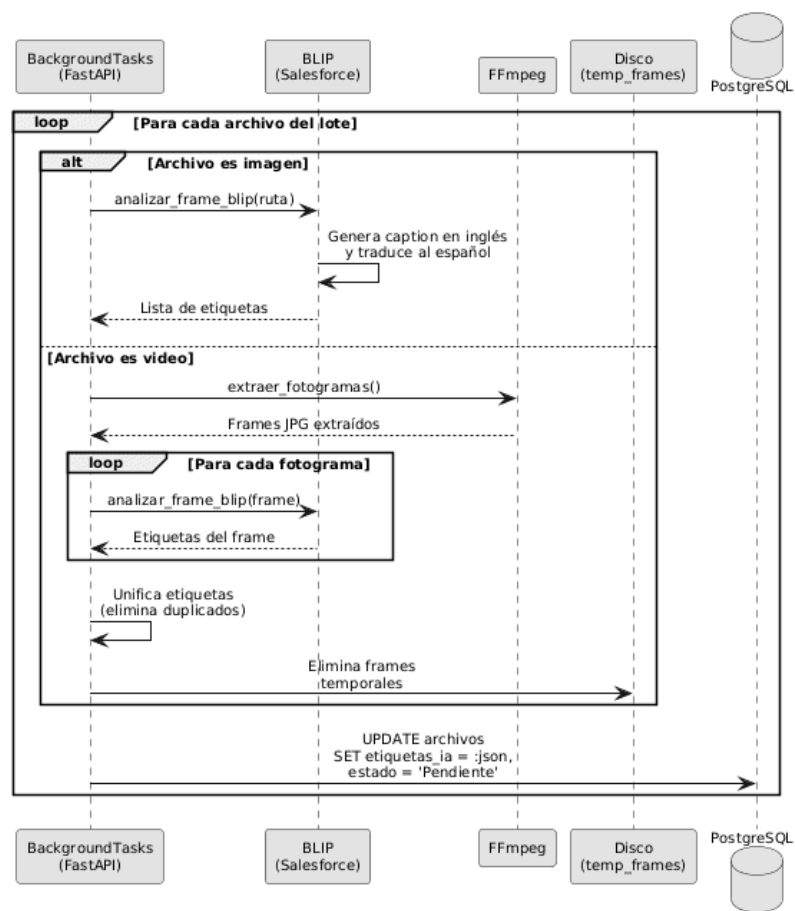


Ilustración 11: Diagrama de secuencia: Procesamiento de IA (BLIP)

4.4.4.4 Diagrama de secuencia: Marcar contenido como Listo (rol integrante)

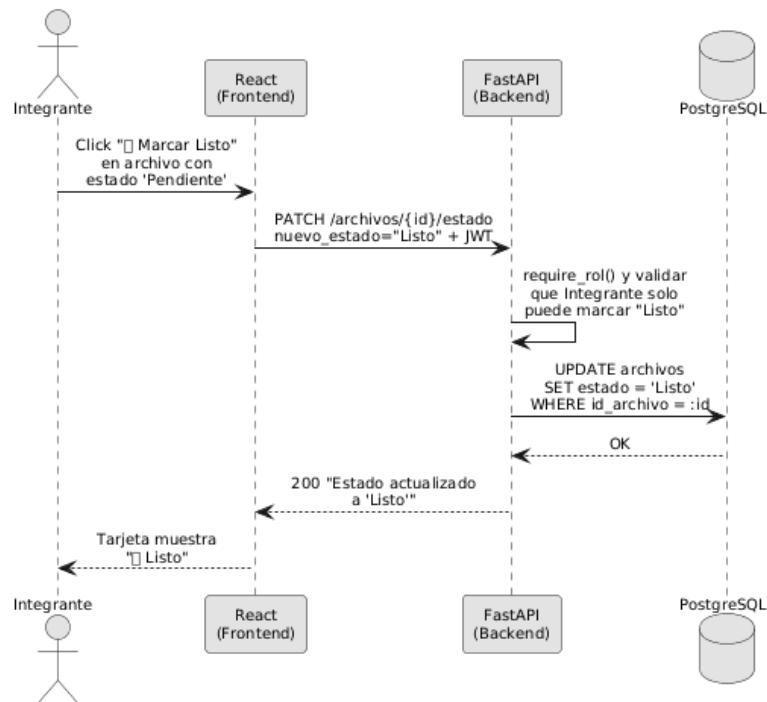


Ilustración 12: Diagrama de secuencia: Marcar contenido como Listo (rol integrante)

4.4.4.5 Diagrama de secuencia: Aprobar contenido (rol supervisor)

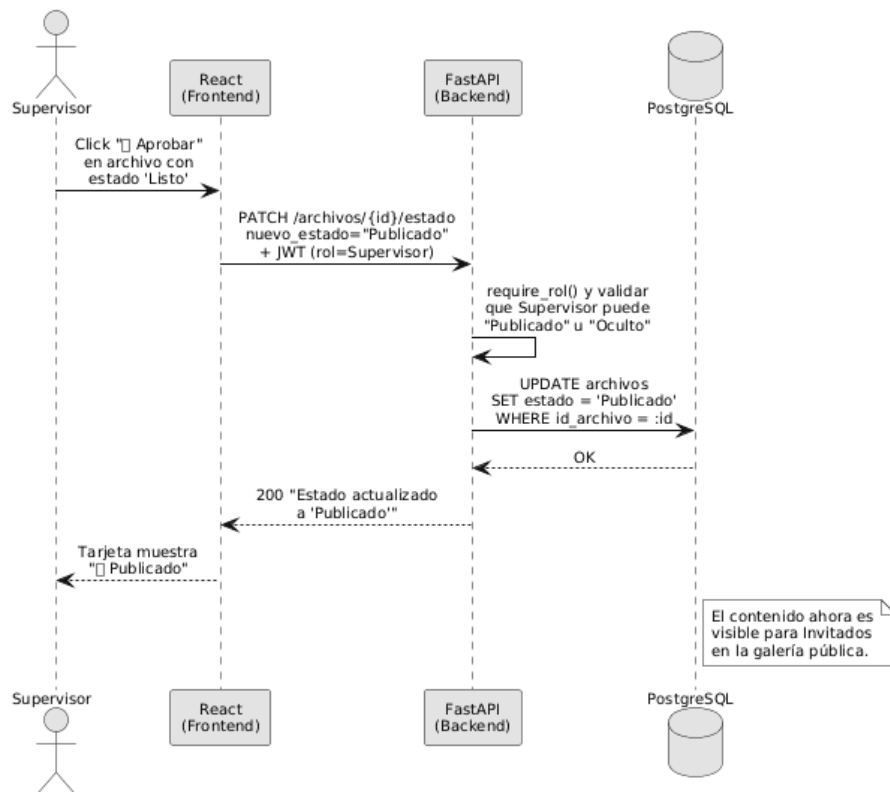


Ilustración 13: Diagrama de secuencia: Aprobar contenido (rol supervisor)

4.4.4.6 Diagrama de secuencia: Ocultar contenido

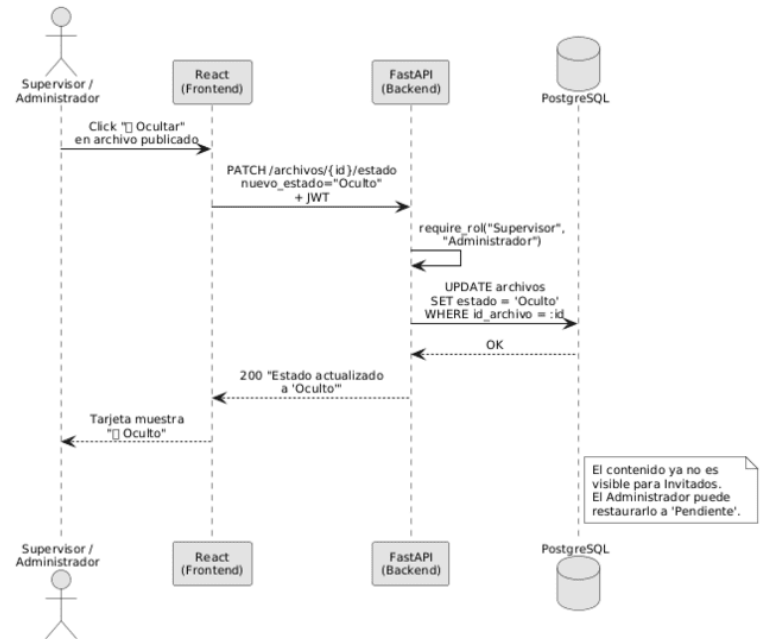


Ilustración 14: Diagrama de secuencia: Ocultar contenido

4.4.4.7 Diagrama de secuencia: Eliminación física

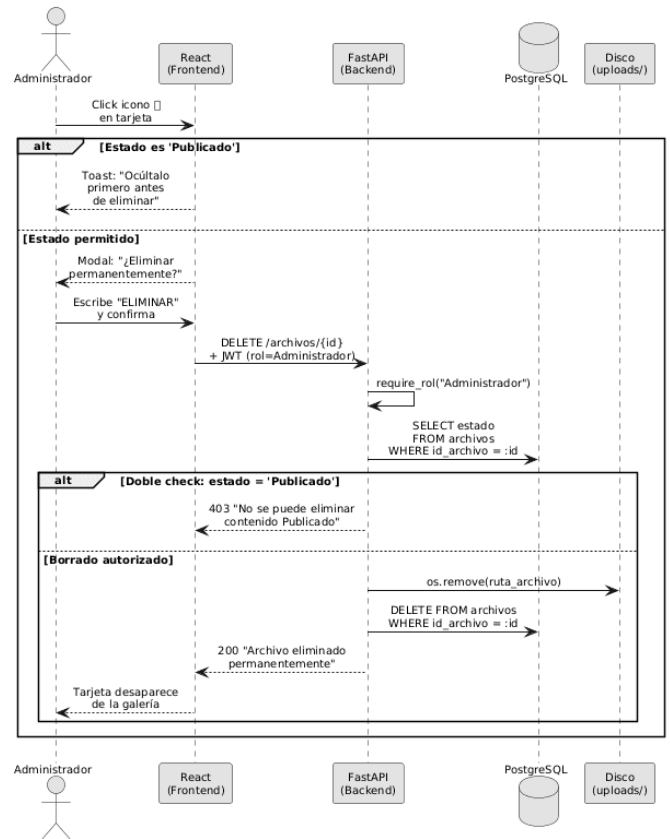


Ilustración 15: Diagrama de secuencia: Eliminación física

4.4.4.8 Diagrama de secuencia: Descarga de archivos

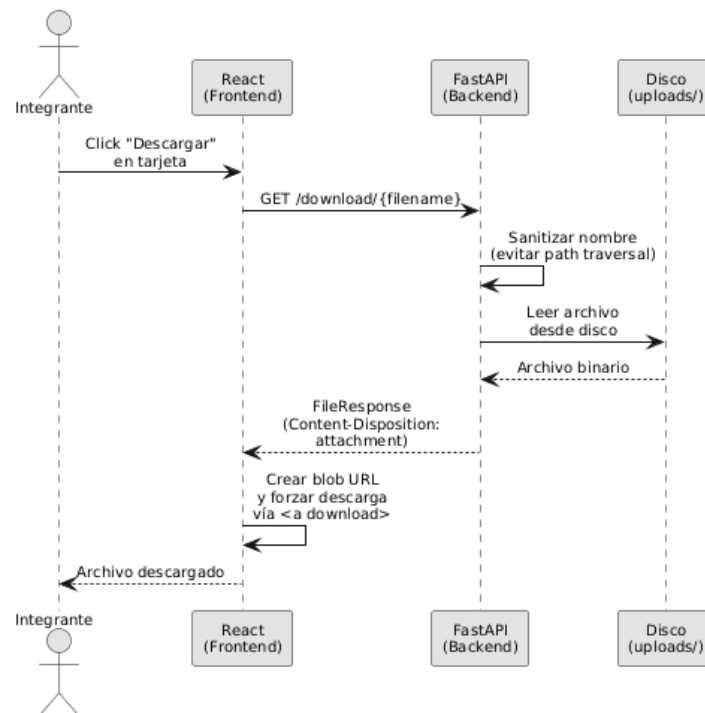


Ilustración 16: Diagrama de secuencia: Descarga de archivos

4.4.4.9 Diagrama de secuencia: Búsqueda y filtrado

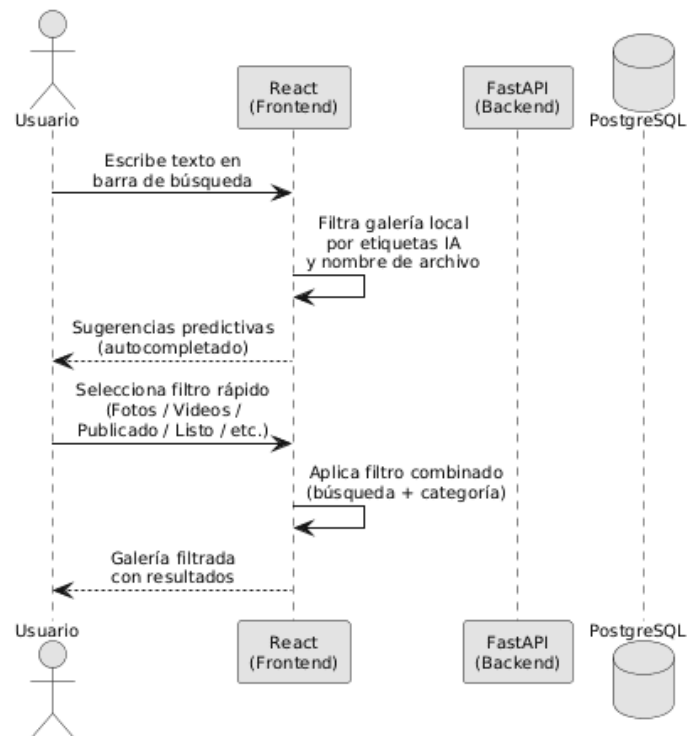


Ilustración 17: Diagrama de secuencia: Búsqueda y filtrado

4.4.4.10 Diagrama de secuencia: Vista de proyectos

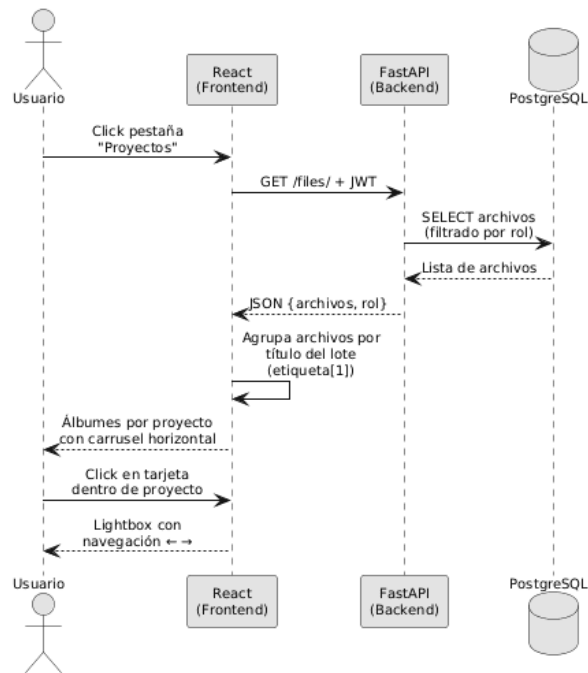


Ilustración 18: Diagrama de secuencia: Vista de proyectos

4.4.5 Diagramas de Estado

4.4.5.1 Diagrama de estado: Ciclo de vida del contenido

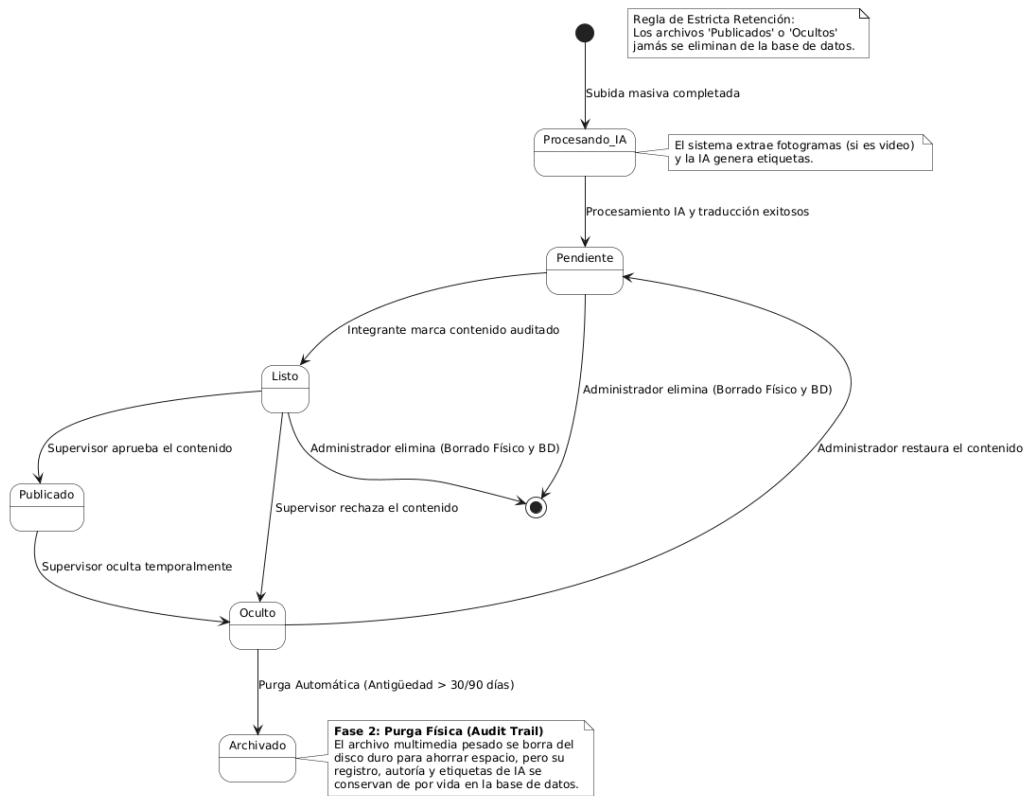


Ilustración 19: Diagrama de estado: Ciclo de vida del contenido

4.4.6 Descripción Técnica / Arquitectura del Sistema

4.4.6.1 Arquitectura del Sistema

Para que la plataforma se sostenga, el diseño del software se dividió en tres partes independientes: lo que ve el usuario en su pantalla, el servidor que procesa las reglas y la base donde se guardan los datos. Todo este conjunto se instaló para que funcione directamente sobre los equipos físicos que ya tiene el Club SoftMedia, sin depender de alojamiento externo.

En la capa de presentación se utilizó React 18 junto con Vite como empaquetador. Los archivos multimedia se muestran en una cuadrícula tipo Masonry que adapta sus columnas según el tamaño de pantalla del dispositivo. También se incorporó navegación por teclado en el visor de imágenes y confirmaciones explícitas para operaciones destructivas; por ejemplo, el usuario debe escribir la palabra "ELIMINAR" antes de borrar un registro.

La capa de lógica de negocio se construyó con FastAPI. El tema del acceso se resolvió emitiendo un pase digital para cada sesión (un token JWT), el cual lleva grabado qué tipo de permiso tiene la persona. Para proteger las claves de los usuarios, se utilizó bcrypt, asegurando que nadie, ni siquiera el administrador de la base de datos, pueda leerlas. Del lado de la red también se pusieron bloqueos: el servidor rechaza cualquier conexión que no venga del dominio oficial gracias a las reglas CORS, y además se activaron las defensas estándar de OWASP contra ataques web.

Una parte interesante del código tiene que ver con cómo se ejecuta la inteligencia artificial. Apenas se prende el servidor, el modelo de Salesforce ya queda cargado en la memoria. Si la computadora en la que se instaló cuenta con una buena tarjeta gráfica (específicamente con soporte CUDA), el programa le tira todo el peso del análisis visual a esa pieza. Pero como no siempre se tiene ese hardware a mano, se dejó programado un salvavidas: si solo hay procesador normal, el sistema recorta a propósito la cantidad de hilos de CPU que la IA puede usar. Así, aunque tarde un par de segundos más, la página nunca se congela para los demás usuarios.

Para la base de datos se eligió PostgreSQL, y la conexión desde el backend se hace a través de SQLAlchemy con consultas parametrizadas, lo que cierra la puerta a inyecciones SQL. La tabla principal vincula cada archivo con su autor y sus metadatos. La elección técnica que más impacto tiene en esta capa es guardar las etiquetas de la IA en un campo JSONB: ese tipo de dato no obliga a tener un número fijo de columnas, se

adapta a la cantidad de etiquetas que genere el modelo y mejora los tiempos de respuesta cuando se hacen búsquedas en el catálogo.

4.4.6.2 Mapa del sistema

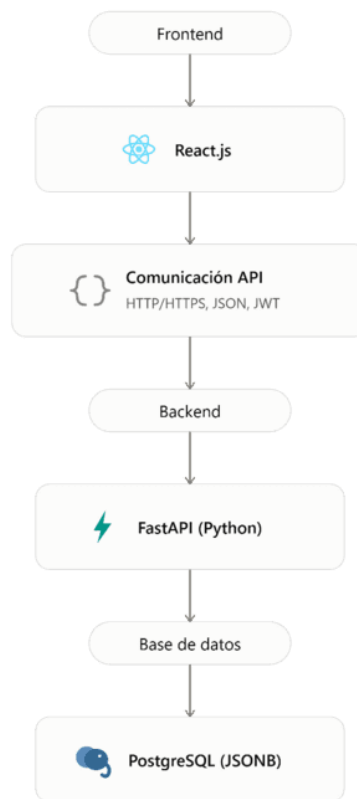


Ilustración 20: Mapa del sistema

4.4.6.3 Tecnologías usadas:

4.4.6.3.1 Backend y API

- **Python:** Lo necesitábamos por las librerías de inteligencia artificial. Ningún otro lenguaje tiene un ecosistema tan completo para trabajar con modelos como BLIP.
- **FastAPI:** Framework de la API REST. Lo preferimos sobre Flask porque ya trae documentación Swagger incorporada sin tener que configurar nada aparte.
- **Uvicorn:** Servidor ASGI que FastAPI requiere para funcionar. Soporta llamadas asíncronas, algo importante cuando el modelo de IA tarda varios segundos en responder.
- **SQLAlchemy:** ORM con el que mapeamos las tablas de PostgreSQL a clases de Python. Esto nos permitió hacer las operaciones CRUD sin estar armando consultas SQL a mano todo el tiempo.

- **PyTorch y Transformers (Hugging Face):** Sin estas dos librerías el modelo BLIP no corre. PyTorch maneja el cómputo en GPU o CPU, y Transformers da la interfaz para cargarlo y pasarle imágenes.
- **passlib, bcrypt y python-jose:** Para proteger el sistema de cuentas, se armó un bloque de seguridad combinando estas tres librerías. La idea central era asegurar que la base de datos nunca viera la clave real del usuario (tarea que asume bcrypt) y que las sesiones activas fueran verificadas de forma segura (tarea de python-jose). Passlib simplemente funcionó como un puente para conectar esas dos operaciones en el servidor sin causar conflictos.

4.4.6.3.2 *Frontend (Interfaz de Usuario)*

- **React:** Se eligió esta herramienta para programar todo el lado visual de la plataforma. Lo que realmente salvó horas de trabajo fue la posibilidad de separar la página en distintas piezas. En lugar de escribir el código del visor de fotos o de la barra de búsqueda una y otra vez para cada pantalla, se diseñaron como componentes aislados que luego simplemente se iban colocando donde hacían falta.
- **Vite:** Lo elegimos por encima de Webpack principalmente por velocidad. Mientras se desarrolla, los cambios en el código se reflejan en el navegador casi de inmediato, lo que ahorra bastante tiempo en el proceso.
- **Tailwind CSS:** Los estilos se aplican directamente en el JSX mediante clases utilitarias. Eso evitó tener que mantener archivos CSS separados que, con el tiempo, se vuelven difíciles de rastrear.
- **Framer Motion:** Se usó para las animaciones del visor y las transiciones entre vistas. Son detalles pequeños, pero sin ellos la interfaz se sentía demasiado brusca al navegar.
- **React Masonry CSS:** Las imágenes del catálogo no tienen todas el mismo tamaño ni proporción. Con una cuadrícula convencional eso genera espacios vacíos entre tarjetas; este componente las distribuye en columnas tipo mosaico y el resultado es visualmente más limpio.
- **Lucide React:** Paquete de iconos SVG para botones y menús.

4.4.6.3.3 Base de Datos

- **PostgreSQL:** Aquí se guardan los usuarios, los archivos multimedia y las etiquetas que genera el modelo. Lo elegimos sobre MySQL porque tiene mejor manejo de tipos de datos y Docker ofrece una imagen oficial bien documentada.
- **psycopg2:** Es el driver que SQLAlchemy usa por debajo para hablar con PostgreSQL. Sin instalarlo, el ORM no puede conectarse.

4.4.6.3.4 Infraestructura y Contenedores

- Docker: Cada parte del sistema tiene su propio contenedor. Al principio no lo usábamos y aparecían errores que solo pasaban en una máquina y en otra no; con Docker eso se eliminó.
- Docker Compose: Un archivo YAML donde quedan definidos los tres contenedores con sus puertos, redes y variables de entorno. Con un solo comando se levanta todo.

4.4.7 Requerimientos funcionales (RF)

Tabla 17: *Requerimientos funcionales (RF)*

Código	Nombre del Requerimiento	Descripción
RF-01	Autenticación y Control de Acceso	• Permitir el inicio de sesión mediante credenciales (correo y contraseña), generando un token JWT con una duración de 8 horas. • Adaptar dinámicamente la interfaz y las funcionalidades disponibles según el rol del usuario autenticado (Administrador, Supervisor, Integrante, Invitado). • Permitir el cierre de sesión con destrucción inmediata del token almacenado en el navegador. • Permitir el acceso en modo Invitado (sin credenciales) restringido exclusivamente a la visualización de la galería pública.

RF-02	Carga Masiva de Contenido Multimedia	• Permitir a los usuarios con rol Integrante o Administrador subir lotes de archivos multimedia mediante arrastre (drag & drop) o selección manual. • Validar en el frontend las extensiones permitidas (.jpg, .jpeg, .png, .bmp, .gif, .tiff, .webp, .mp4, .avi, .mov, .mkv, .webm) y los límites de peso (50 MB por imagen, 3 GB por video). • Exigir la asignación obligatoria de un título descriptivo del lote antes de permitir la confirmación de la subida. • Ejecutar la subida bajo el patrón Fire and Forget: cerrar la interfaz de carga de forma inmediata y encolar el procesamiento de IA en segundo plano sin bloquear al usuario.
RF-03	Procesamiento Automático con Inteligencia Artificial	• Al recibir una captura fotográfica, la arquitectura delega el análisis al motor algorítmico local, el cual estructura una predicción semántica nativa que la lógica del servidor traduce inmediatamente al español. • Para metrajes que no superen la extensión de cinco minutos, el backend despliega una rutina de extracción de fotogramas clave. Posteriormente, el algoritmo inspecciona cada captura de forma aislada para identificar patrones visuales. • El arreglo de palabras resultante pasa por un filtro de exclusión que suprime preposiciones y artículos sin valor descriptivo. Los términos definitivos se insertan en la estructura relacional utilizando el formato optimizado JSONB de PostgreSQL. • Frente a secuencias de video extensas, la plataforma detiene la inferencia autónoma. En este escenario, la interfaz gráfica obliga al creador a digitar sus propios metadatos antes de confirmar el envío, garantizando la intervención humana directa y previniendo el colapso de la memoria del servidor.
RF-04	Galería y Búsqueda por Etiquetas	• Los archivos multimedia ya procesados deben presentarse en una galería responsiva con disposición tipo Masonry, adaptable a pantallas de escritorio, tableta y dispositivos móviles. • El sistema debe ofrecer un campo de búsqueda que compare las palabras clave ingresadas por el usuario contra las etiquetas generadas por la

		IA, devolviendo las coincidencias encontradas. • Se requiere un visor tipo Lightbox que permita la previsualización ampliada de imágenes y la reproducción de videos. La navegación dentro del visor debe funcionar mediante las teclas de flecha y la tecla Escape para cerrar. • Cada archivo visible en la galería debe contar con una opción de descarga directa en su formato y calidad original.
RF-05	Flujo de Revisión y Aprobación por Roles	• La cuadrícula interna aplica un código de colores dinámico sobre las miniaturas para reflejar la fase operativa de cada registro (análisis en progreso, revisión pendiente, preparado para auditoría, recurso público o elemento oculto). • El autor del metraje conserva el permiso exclusivo para actualizar el estado de sus capturas hacia la fase de preparación, indicando internamente que el lote está depurado y listo para el escrutinio de calidad. • La cuenta de supervisión ejerce el control absoluto sobre las publicaciones. Su panel privado le autoriza a transicionar los elementos aprobados hacia la galería externa, o en su defecto, bloquear el material enviándolo a la bóveda de recursos rechazados. • El paso de un archivo al entorno público es una facultad operativa estrictamente aislada para el perfil supervisor, bloqueando por diseño cualquier intento de publicación directa por parte de las cuentas con nivel de administrador.
RF-06	Eliminación Controlada y Retención de Datos	• La consola técnica habilita la supresión destructiva (borrado del disco duro y limpieza del índice relacional) exclusivamente sobre aquellas capturas que aún transitan por las fases iniciales de procesamiento o revisión. • La arquitectura lógica bloquea la eliminación de cualquier documento que posea un historial de exhibición pública o bloqueo preventivo. Esta regla dura asegura la persistencia de las pistas de auditoría y protege la memoria institucional de la agrupación. • Para anular el riesgo de pérdidas involuntarias, las rutinas de purga exigen un parámetro de seguridad adicional. La interfaz obliga al responsable técnico a digitar una instrucción

		de confirmación exacta antes de desencadenar la orden de borrado en el servidor. • Bajo criterios de optimización de almacenamiento, los recursos aislados que superen el trimestre de antigüedad activan una rutina de limpieza autónoma. El motor destruye el peso físico del metraje, pero congela indefinidamente los metadatos y la autoría dentro de la base de datos bajo la categoría de archivo histórico.
--	--	--

4.4.8 Requerimientos No Funcionales (RNF):

Tabla 18: Requerimientos No Funcionales (RNF):

Código	Nombre del Requerimiento	Descripción
RNF-01	Seguridad	• Cifrado de Credenciales: Proteger contraseñas (mediante hashes) y emitir tokens de sesión (JWT) encriptados. • Políticas CORS: Restringir los dominios de origen permitidos en el servidor para asegurar la comunicación con la API. • Estándares OWASP: Aplicar cabeceras HTTP de protección para mitigar vulnerabilidades y ataques comunes. • Control de Sesión: Forzar la caducidad y redirección automática de los accesos inactivos. • Restricción Operativa: Bloquear desde el servidor la descarga de material original (en crudo) para las cuentas con nivel de Invitado.
RNF-02	Rendimiento	• El motor algorítmico local estructura las predicciones semánticas en un lapso inferior a tres segundos por captura, siempre que el equipo cuente con aceleración gráfica dedicada. • La arquitectura incluye una rutina de inspección física que deriva la carga matemática a la tarjeta de video disponible; frente a la carencia de esta, el sistema restringe los subprocesos lógicos para proteger la memoria del ordenador. • Las transferencias masivas de metraje activan tareas asíncronas independientes, garantizando que el cliente web reciba una confirmación inmediata sin congelar la interfaz gráfica.

		• Para preservar la integridad del ancho de banda institucional, la capa de recepción bloquea autónomamente cualquier envío simultáneo que exceda los quince recursos de video o cuarenta fotografías.
RNF-03	Usabilidad	• El entorno cliente despliega una estructura visual asimétrica que facilita el tránsito directo entre el catálogo público, el formulario de transferencia y la consola técnica. • Frente a operaciones destructivas o cambios de estado, la pantalla renderiza alertas gráficas emergentes que informan al integrante sobre el resultado exacto de su transacción. • Durante la extracción de fotogramas o cálculos matemáticos del modelo, las miniaturas reflejan animaciones de progreso continuo para evitar la incertidumbre operativa del usuario. • El área de recepción de documentos admite la captura de elementos mediante la técnica de arrastre del cursor, suprimiendo la necesidad de navegar manualmente por los directorios internos del sistema operativo.
RNF-04	Compatibilidad	• El código del frontend mantiene una operatividad ininterrumpida sobre los motores de navegación web de uso estándar dentro de la comunidad universitaria. • La cuadrícula visual reorganiza sus bloques de forma dinámica basándose en resoluciones predefinidas, validando la legibilidad tanto en monitores de escritorio como en pantallas de teléfonos celulares. • Toda la infraestructura tecnológica se encapsula para ejecutarse nativamente dentro de los servidores físicos de la agrupación, anulando por completo la dependencia de servicios de alojamiento externos.
RNF-05	Mantenibilidad	• El diseño del software se segmenta bajo el modelo clásico de tres capas, permitiendo escalar o sustituir el motor de presentación sin alterar la lógica de negocio central. • El servidor inscribe un historial detallado con marcas temporales exactas para cada transferencia, auditoría o excepción

		algorítmica, simplificando drásticamente las labores de rastreo técnico. • El acople con la base de datos se orquesta mediante una herramienta de mapeo que empaqueta las consultas de forma parametrizada, erradicando vulnerabilidades lógicas y facilitando futuros respaldos de la información.
--	--	--

4.4.9 Roles y Responsabilidades

Tabla 19: Roles y Responsabilidades

TIPO DE USUARIO	ROL SCRUM / SISTEMA	DESCRIPCIÓN
Scrum Team	Product Owner / Scrum Master / Developer	Jazmany Daniel Sumi Arévalo concentra la dirección metodológica y la construcción técnica del Proyecto. En la capa de gestión, administra la lista de requerimientos operativos y coordina el avance de las iteraciones ágiles. En el plano de la ingeniería, programa la totalidad de la arquitectura del ecosistema, desarrollando las vistas del cliente, la lógica del servidor local, la estructura de la base de datos y el acople del modelo algorítmico.
Stakeholder	Tutor de Tesis	Ing. Jefferson Arca. Supervisa el proyecto, revisa los avances técnicos por sprint y valida el cumplimiento académico, metodológico y arquitectónico del sistema.
End User (Rol: Administrador)	Administrador del Sistema	Ejerce el control técnico absoluto sobre la infraestructura de la plataforma. Sus facultades operativas le permiten ejecutar la depuración física de los documentos multimedia en el servidor local para liberar espacio de almacenamiento, así como restaurar metadatos ocultos. Por reglas estrictas de arquitectura, su cuenta tiene bloqueada por diseño la capacidad de publicar contenido en la galería pública, delegando esta acción exclusivamente al nivel supervisor.
End User (Rol: Supervisor)	Supervisor de Contenido	Representa la máxima autoridad editorial del repositorio y gestiona el control de calidad institucional. Su entorno de trabajo le permite revisar directamente los metrajes que los creadores han marcado como preparados, evaluando el material para decidir su destino final. Posee la facultad exclusiva de transicionar los elementos aprobados hacia la

		galería pública o restringirlos enviándolos a la bóveda de recursos ocultos, consolidándose como el único perfil autorizado para oficializar las publicaciones.
End User (Rol: Integrante)	Integrante (Creador de Contenido)	Alumnado operativo de la agrupación (fotógrafos y editores) cuyas tareas exigen la transferencia directa de secuencias de video y capturas fotográficas hacia el catálogo. Una vez que el motor de inteligencia artificial procesa los archivos de forma autónoma, estos perfiles auditan sus propios lotes y cambian el estado del registro a "Listo", notificando al sistema que el material está preparado para someterse a la evaluación del cuerpo de supervisión.
End User (Rol: Invitado)	Público General / Comunidad Universitaria	Entidad de navegación conformada por el alumnado, personal docente y público ajeno al departamento que ingresa al ecosistema sin credenciales de acceso. Su interacción se restringe a operaciones de solo lectura, permitiéndoles visualizar y consultar el acervo audiovisual aprobado mediante el motor de búsqueda. Para proteger la integridad del catálogo, el entorno bloquea por defecto cualquier privilegio de edición o alteración del metraje expuesto.

4.4.10 Planificación del Sprint

4.4.10.1 Sprint 1: Cimientos Estructurales, Base de Datos y Backend API REST

Tabla 20: Arquitectura Base, Base de Datos y Backend API REST

Sprint 1: Cimientos Estructurales, Base de Datos y Backend API REST			
Duración:	2 semanas (02/03/2026 – 15/03/2026)	Prioridad:	Alta
Objetivo:	Establecer la arquitectura completa del sistema con FastAPI y PostgreSQL, diseñar e implementar el esquema de base de datos con el schema audiovisual, desarrollar el backend API REST con endpoints de carga masiva, listado y procesamiento de archivos multimedia, e integrar la IA BLIP para generación automática de etiquetas descriptivas en español.		
Historias de usuario:	HU01: Arquitectura del Sistema y Procesamiento Inteligente de Archivos		
Tareas de desarrollo:	• Inicialización del entorno virtual y despliegue del ecosistema local. • Modelado lógico y físico de la estructura de datos (esquema audiovisual). • Construcción de las rutas de transferencia y comunicación (API REST) en el servidor central.		

	• Acople del modelo de Inteligencia Artificial para la extracción de predicciones semánticas. • Desarrollo del motor de fragmentación de video mediante la herramienta FFmpeg. • Implementación de rutinas de ejecución en segundo plano para evitar la saturación de la red. • Encapsulamiento de la arquitectura mediante la orquestación de contenedores. • Auditoría de software, documentación y validación de conectores.
Plan de entrega:	• Configuración del ambiente de trabajo Instalación de las dependencias lógicas (Python, motor relacional y empaquetadores virtuales) estructurando el núcleo del servidor. Tiempo estimado: 8 horas Fecha: 02/03/2026 – 03/03/2026 • Diseño e implementación de base de datos Creación del esquema audiovisual y la tabla de recursos físicos, incorporando la estructura JSONB optimizada para resguardar las etiquetas algorítmicas y configurando la auto-migración de tablas. Tiempo estimado: 10 horas Fecha: 04/03/2026 – 05/03/2026 • Construcción de las rutas de transferencia Programación de los puntos de acceso en el backend para la recepción masiva, la exposición de archivos estáticos y la recuperación de recursos, aplicando los respectivos controles de red. Tiempo estimado: 16 horas Fecha: 06/03/2026 – 08/03/2026 • Integración del motor algorítmico y traductor Inicialización del procesador visual local y acoplamiento de la biblioteca de traducción para generar descripciones en español, depurando palabras sin valor semántico antes de su inserción en la base de datos. Tiempo estimado: 14 horas Fecha: 09/03/2026 – 10/03/2026 • Fragmentación de metraje (FFmpeg) Configuración de la estrategia de extracción adaptativa de fotogramas, estableciendo límites de procesamiento temporal para salvaguardar la memoria del equipo durante la inspección de videos. Tiempo estimado: 12 horas Fecha: 11/03/2026 – 12/03/2026 • Asincronía y encapsulamiento Derivación de las tareas analíticas hacia hilos de ejecución en segundo plano y creación de los volúmenes persistentes para proteger la caché de los modelos matemáticos dentro de los contenedores virtuales. Tiempo estimado: 12 horas Fecha: 13/03/2026 – 14/03/2026 • Documentación técnica y auditoría del Sprint 1 Verificación de las migraciones de datos, testeo de resistencia en el flujo de transferencia y redacción del manual operativo de los enlaces del servidor. • Tiempo estimado: 6 horas Fecha: 15/03/2026

4.4.10.2 Sprint 2: Autenticación JWT, Sistema de roles y control de acceso

Tabla 21: Sprint 2: Autenticación JWT, Sistema de roles y control de acceso

Sprint 2: Autenticación JWT, Sistema de roles y control de acceso			
Duración:	2 semanas (16/03/2026 – 29/03/2026)	Prioridad:	Alta
Objetivo:	Implementar el sistema completo de autenticación basado en JWT con hashing bcrypt, definir los cuatro roles del sistema (Administrador, Supervisor, Integrante, Invitado) con sus permisos específicos, desarrollar el control de acceso por rol en cada endpoint, implementar el sistema de cambio de estados y la regla de seguridad para borrado físico, y agregar middleware de seguridad OWASP con cabeceras HTTP protectoras.		
Historias de usuario:	HU02: Autenticación, Roles y Seguridad del Sistema		
Tareas de desarrollo:	• Diseño de la tabla de usuarios con roles y restricciones CHECK • Backend de autenticación JWT (login, creación de tokens, validación) • Implementación de hashing bcrypt con passlib • Sistema de Dependencias de FastAPI para control de acceso por rol • Endpoint de cambio de estado con validaciones por rol • Endpoint de borrado físico con Regla de Oro de seguridad • Middleware de seguridad OWASP (cabeceras HTTP) • Seed de usuarios de prueba y documentación		
Plan de entrega:	• Entidad de usuarios: Modelado de la tabla relacional en PostgreSQL con restricciones de unicidad y limitación estricta a los cuatro roles operativos. Tiempo estimado: 6 horas Fecha: 16/03/2026 – 17/03/2026 • Motor de autenticación: Programación del punto de acceso principal en el servidor para evaluar credenciales y emitir tokens criptográficos con caducidad autónoma. Tiempo estimado: 12h Fecha: 18/03/2026 – 19/03/2026 • Control jerárquico: Inyección de dependencias lógicas para decodificar las sesiones en tiempo real y bloquear transferencias según el nivel de la cuenta. Tiempo estimado: 14h Fecha: 20/03/2026 – 22/03/2026 • Gestión de recursos: Desarrollo de las rutas de transición de estados y programación de reglas de negocio para impedir la supresión de archivos públicos. Tiempo estimado: 14h Fecha: 23/03/2026 – 25/03/2026 • Seguridad perimetral: Filtrado estricto de orígenes cruzados (CORS) y aplicación de cabeceras de protección HTTP para blindar las respuestas del backend. Tiempo estimado: 8h Fecha: 26/03/2026 – 27/03/2026 • Población inicial: Inserción automatizada de los cuatro perfiles de prueba utilizando cifrado de contraseñas y auditoría del flujo de accesos. Tiempo estimado: 6h Fecha: 28/03/2026 – 29/03/2026		

4.4.10.3 Sprint 3: Frontend Web — Interfaz de Catálogo y Carga Masiva

Tabla 22: Sprint 3: Frontend Web — Interfaz de Catálogo y Carga Masiva

Sprint 3: Frontend Web — Interfaz de Catálogo y Carga Masiva			
Duración:	2 semanas (30/03/2026 – 12/04/2026)	Prioridad:	Alta
Objetivo:	Desarrollar la interfaz web completa del catálogo audiovisual con React + Vite + Tailwind CSS, implementar la pantalla de login con selección rápida de usuarios de prueba, crear el modal de carga masiva acumulativa estilo Facebook con drag-and-drop, la galería Masonry con visualización de imágenes y videos, el lightbox modal con navegación por teclado, y la barra de búsqueda predictiva con filtros rápidos por tipo y estado.		
Historias de usuario:	HU03: Interfaz de Catálogo Audiovisual HU04: Sistema de Carga Masiva Multimedia		
Tareas de desarrollo:	- Despliegue del entorno cliente (React, Vite y Tailwind CSS). - Programación de la vista de autenticación y validación de sesiones. - Construcción del entorno emergente para transferencias masivas. - Maquetación de la cuadrícula asimétrica (Masonry) y visor expandido. - Integración del buscador predictivo y filtros operativos. - Acople de conectores API y captura gráfica de excepciones.		
Plan de entrega:	- Entorno de presentación: Estructuración del proyecto cliente, integración de la biblioteca de estilos y orquestación de contenedores. Tiempo estimado: 6h Fecha: 30/03/2026 – 31/03/2026 - Módulo de autenticación: Interfaz de acceso con captura de credenciales, resguardo local de firmas criptográficas y atajo operativo para perfiles de invitado. Tiempo estimado: 10h 01/04/2026 – 02/04/2026 - Transferencia masiva: Formulario de recepción con captura por arrastre, renderizado dinámico de miniaturas previas y restricciones lógicas de peso y extensión. Tiempo estimado: 16h Fecha: 03/04/2026 – 05/04/2026 - Cuadrícula visual en cascada: Diseño de la galería interactiva reorganizable según el monitor, incorporando tarjetas multimedia con distintivos de estado y selección concurrente. Tiempo estimado: 18h Fecha: 06/04/2026 – 08/04/2026 - Visor y controles nativos: Renderizado del componente superpuesto para expandir recursos, admitiendo comandos de teclado e integrando las acciones de auditoría. Tiempo estimado: 10h Fecha: 09/04/2026 – 10/04/2026 - Explorador y filtros: Construcción de la barra semántica con sugerencias en tiempo real y categorización visual del catálogo. Tiempo estimado: 10h Fecha: 11/04/2026 - Conexión API y auditoría: Acople de vistas con el servidor, programación de consultas asíncronas recurrentes durante el		

	cálculo de IA y despliegue de alertas emergentes ante errores. Tiempo estimado: 8h Fecha: 12/04/2026
--	---

4.4.10.4 Sprint 4: Gestión Avanzada, Operaciones Masivas y Vista de Proyectos

Tabla 23: Sprint 4: Gestión Avanzada, Operaciones Masivas y Vista de Proyectos

Sprint 4: Gestión Avanzada, Operaciones Masivas y Vista de Proyectos			
Duración:	2 semanas (13/04/2026 – 26/04/2026)	Prioridad:	Alta
Objetivo:	Implementar las funcionalidades avanzadas del panel: confirmación de borrado seguro con escritura de clave ELIMINAR, selección múltiple con barra de acciones masivas, descarga masiva con File System Access API y fallback universal, vista de Proyectos agrupados por lote/evento con carrusel horizontal, skeleton loading para mejor UX, y optimizaciones finales de rendimiento y diseño premium.		
Historias de usuario:	HU05: Gestión Avanzada y Operaciones Masivas HU06: Vista de Proyectos y Experiencia de Usuario Premium		
Tareas de desarrollo:	• Modal de confirmación de borrado seguro (escritura ELIMINAR) • Sistema de selección múltiple con checkboxes • Barra de acciones masivas (BulkActionBar) • Descarga masiva con File System Access API + fallback • Vista de Proyectos agrupados por evento/lote • Skeleton loading y animaciones premium • Diseño UI premium con glassmorphism y micro-animaciones • Pruebas integrales y documentación final		
Plan de entrega:	• Modal de confirmación de borrado seguro DeleteConfirmModal con animación spring, ícono de alerta con shake, campo de texto que requiere escribir "ELIMINAR" para confirmar, soporte para borrado individual y masivo. Tiempo: 8 horas Fecha: 13/04/2026 – 14/04/2026 • Selección múltiple y BulkActionBar Sistema de selección con Set de IDs, checkboxes animados en MediaCards (hover + modo selección), BulkActionBar flotante con contador, botones de descarga masiva y eliminación masiva. Tiempo: 12 horas Fecha: 15/04/2026 – 16/04/2026 • Descarga masiva inteligente Implementación con File System Access API (showDirectoryPicker) para elegir carpeta destino. Fallback universal para Firefox: descarga secuencial con blobs y enlaces "a download" con pausa anti-bloqueo de 400ms. Tiempo: 14 horas Fecha: 17/04/2026 – 19/04/2026 • Vista de Proyectos agrupados ProyectosView con agrupación automática por título de lote (etiquetas), carrusel horizontal con snap scroll, cards compactas		

	con preview, estado y nombre. ProyectosSkeleton para estado de carga. Tiempo: 12 horas Fecha: 20/04/2026 – 21/04/2026 • Diseño premium y micro-animaciones Header sticky con glassmorphism (backdrop-filter blur), navegación por pestañas Explorar/Proyectos, toast notifications animados, transiciones con framer-motion (spring, fade, scale), banner de modo Invitado. Tiempo: 12 horas Fecha: 22/04/2026 – 24/04/2026 • Pruebas integrales y documentación final Pruebas de flujo completo para los 4 roles, verificación de permisos en cada acción, pruebas de compatibilidad (Chrome, Firefox, Brave, Edge), documentación técnica y preparación de entregables. Tiempo: 10 horas Fecha: 25/04/2026 – 26/04/2026
--	--

4.4.11 Backlog del Producto

4.4.11.1 Backlog inicial 24/02/2026 – 01/03/2026

Tabla 24: Backlog inicial

Tareas de Desarrollo			
Nro.	Tarea	Prioridad	Estado
1	Instalación de Python 3.x y creación de entorno virtual	Alta	Completada
2	Instalación y configuración de PostgreSQL	Alta	Completada
3	Configuración de Git y GitHub	Alta	Completada
4	Instalación de FastAPI, SQLAlchemy y dependencias	Alta	Completada
5	Instalación de FFmpeg en el sistema	Alta	Completada
6	Diseño del esquema de base de datos (schema audiovisual)	Alta	Completada
7	Estructura base del proyecto backend	Alta	Completada
8	Instalación de Docker y Docker Compose	Alta	Completada
9	Descarga del modelo BLIP de HuggingFace	Alta	Completada

4.4.11.2 Sprint 1: Arquitectura y Backend 02/03/2026 – 15/03/2026

Tabla 25: Fin del Sprint 1

Nro.	Tarea	Prioridad	Estado
1	Configuración del entorno de desarrollo	Alta	Completada
2	Diseño e implementación del esquema de base de datos	Alta	Completada
3	Desarrollo de endpoints API REST (carga, listado, descarga)	Alta	Completada
4	Integración de IA BLIP para etiquetado automático	Alta	Completada
5	Procesamiento de video con FFmpeg	Alta	Completada

6	Pipeline de IA en segundo plano (BackgroundTasks)	Alta	Completada
7	Dockerización y docker-compose.yml	Alta	Completada
8	Documentación y pruebas del Sprint 1	Media	Completada

4.4.11.3 Sprint 2 Autenticación y Roles 16/03/2026 – 29/03/2026

Tabla 26: Fin del Sprint 2

Nro.	Tarea	Prioridad	Estado
1	Creación de tabla de usuarios con roles y CHECK	Alta	Completada
2	Backend de autenticación JWT (endpoint /auth/login)	Alta	Completada
3	Hashing de contraseñas con bcrypt/passlib	Alta	Completada
4	Dependencies de FastAPI para control de acceso por rol	Alta	Completada
5	Endpoint PATCH de cambio de estado con reglas por rol	Alta	Completada
6	Endpoint DELETE con Regla de Oro de seguridad	Alta	Completada
7	Middleware de seguridad OWASP	Alta	Completada
8	Seed de usuarios de prueba y documentación	Media	Completada

4.4.11.4 Sprint 3: Frontend Web — Catálogo y Carga 30/03/2026 – 12/04/2026

Tabla 27: Fin del Sprint 3

Nro.	Tarea	Prioridad	Estado
1	Configuración del proyecto React + Vite + Tailwind CSS	Alta	Completada
2	Pantalla de Login con UI premium	Alta	Completada
3	Modal de Carga Masiva con drag-and-drop	Alta	Completada
4	Galería Masonry con MediaCards responsivas	Alta	Completada
5	Lightbox Modal con navegación por teclado	Alta	Completada
6	Barra de búsqueda predictiva con sugerencias	Alta	Completada
7	Filtros rápidos por tipo y estado	Alta	Completada
8	Integración con API backend y pruebas	Alta	Completada

4.4.11.5 Sprint 4: Gestión Avanzada y Proyectos 13/04/2026 – 26/04/2026

Tabla 28: Fin del Sprint 4

Nro.	Tarea	Prioridad	Estado
1	Modal de confirmación de borrado seguro	Alta	Completada
2	Sistema de selección múltiple con checkboxes	Alta	Completada
3	Barra de acciones masivas (BulkActionBar)	Alta	Completada
4	Descarga masiva con File System Access API + fallback	Alta	Completada
5	Vista de Proyectos agrupados por evento/lote	Alta	Completada

6	Skeleton loading y animaciones premium	Alta	Completada
7	Diseño UI premium con glassmorphism	Alta	Completada
8	Pruebas integrales y documentación final	Alta	Completada

4.5 Interfaz de Usuario (UI) / Prototipos:

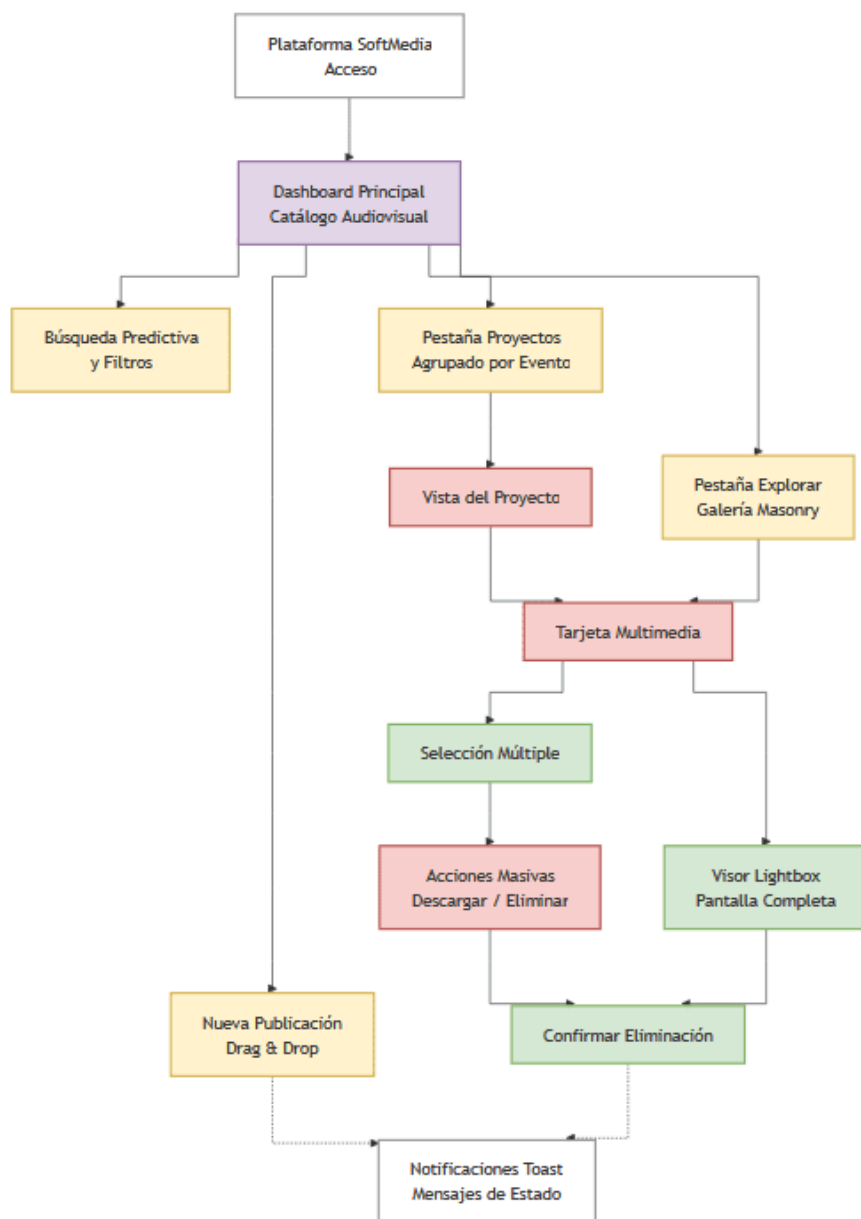


Ilustración 21: Mapa de navegación del módulo Web

4.5.1 Pantallas del Sistema:

4.5.1.1 Pantalla: Explorar (Galería Masonry)

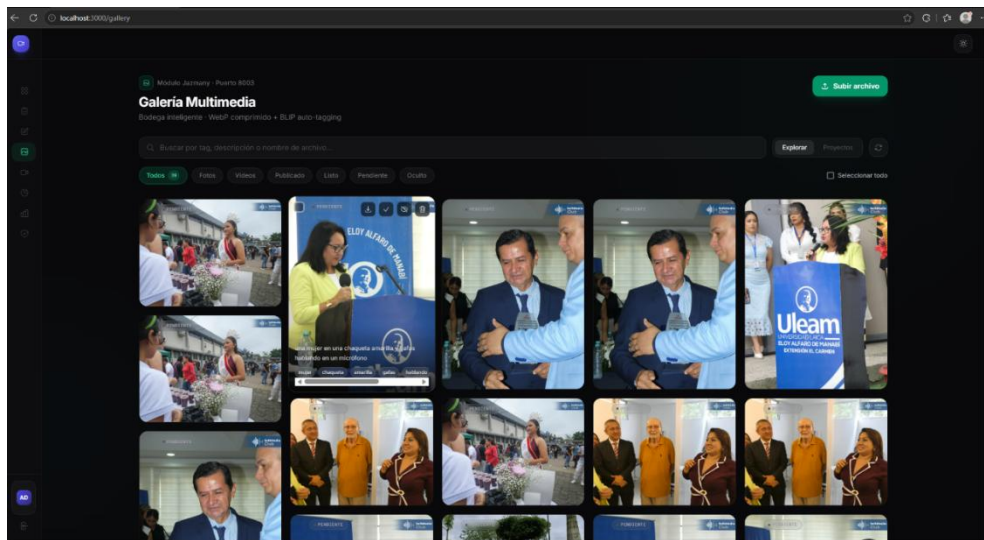


Ilustración 22: Pantalla explorar galería mansory

Esta pantalla implementa una estructura de galería tipo 'Masonry' para visualizar el contenido audiovisual de forma asimétrica y fluida, optimizando el espacio sin importar la relación de aspecto (vertical u horizontal) de las imágenes o videos. El resultado es una galería que invita a recorrerla, donde el contenido se presenta de forma ordenada sin que todas las tarjetas tengan que verse iguales.

4.5.1.2 Pantalla: Búsqueda Predictiva y Filtros

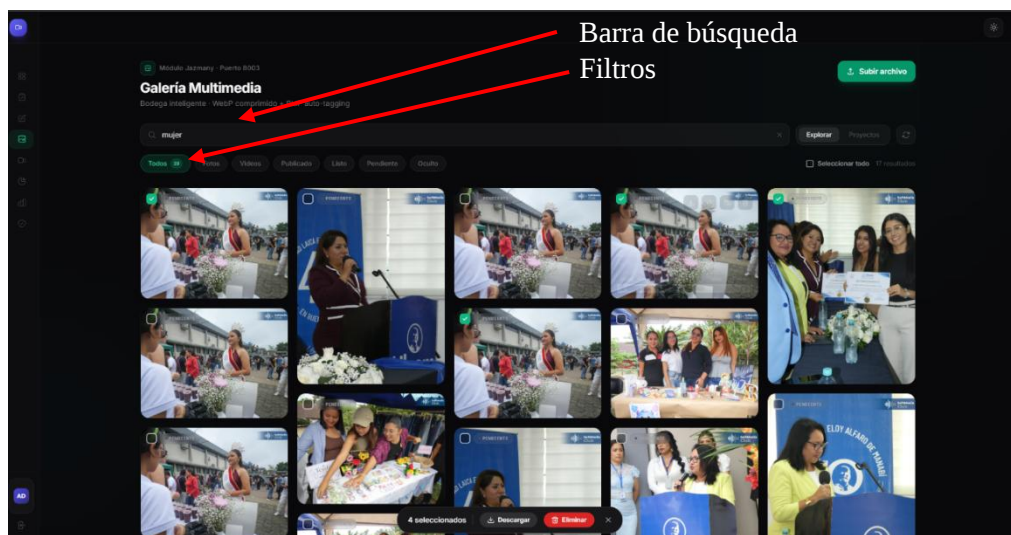


Ilustración 23: Pantalla de Búsqueda Predictiva y Filtros

Desde la pantalla principal, el usuario puede escribir cualquier término y el sistema empieza a mostrar coincidencias antes de que termine de escribir. Si eso no es suficiente para acotar los resultados, hay filtros disponibles por tipo de archivo, por fecha o por estado que se pueden combinar con la búsqueda textual. La idea es que encontrar un archivo específico no tome más de unos pocos segundos.

4.5.1.3 Pantalla: Proyectos (Agrupación por Evento)



Ilustración 24: Pantalla de Proyectos (Agrupación por Evento)

En vez de mostrar todos los archivos mezclados, esta pantalla los agrupa por evento. Cada cobertura que el club ha realizado aparece como un bloque independiente, lo que hace más fácil ubicar el material de una actividad concreta sin tener que filtrar desde cero.

4.5.1.4 Pantalla: Vista del Proyecto

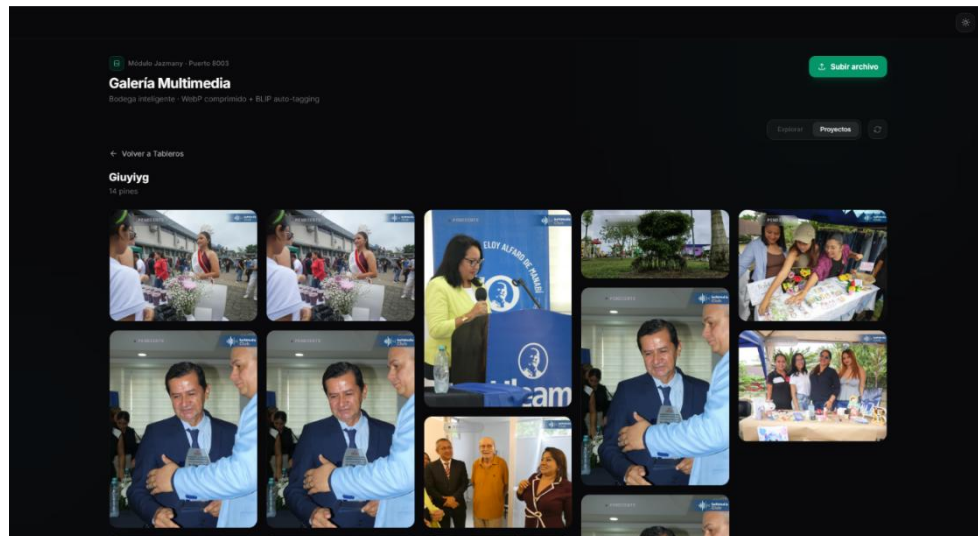


Ilustración 25: Pantalla de Vista del Proyecto

Al ingresar a un evento específico desde la pestaña de Proyectos, esta pantalla expone toda la multimedia relacionada de forma exclusiva a esa cobertura. En la práctica, esta pantalla funciona como un álbum cerrado para un evento específico. La idea es que el usuario pueda entrar a revisar las fotos o videos de esa única cobertura sin mezclarlos con el resto del catálogo, y desde ahí mismo poder realizar cualquier cambio o revisión que necesite ese lote.

4.5.1.5 Pantalla: Nueva Publicación (Drag & Drop)

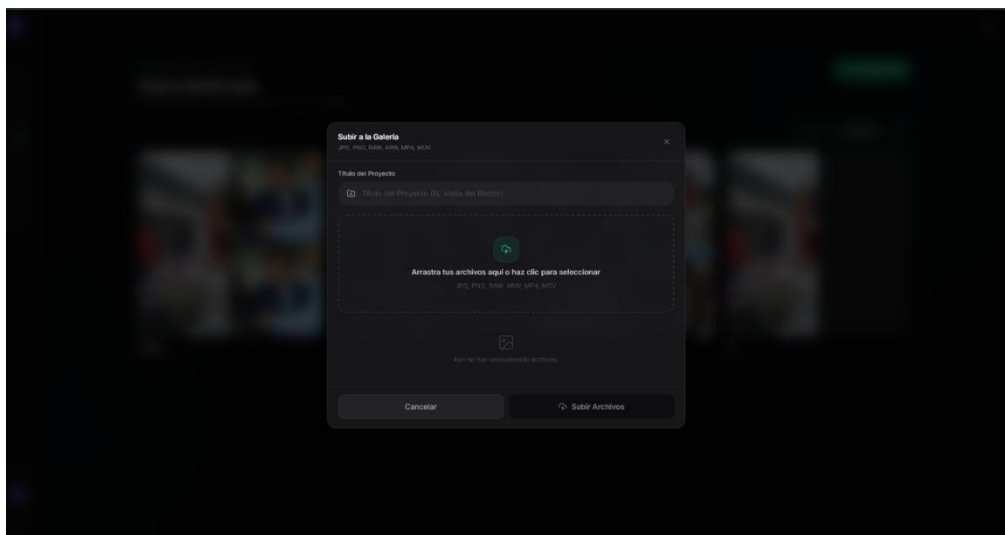


Ilustración 26: Pantalla de Nueva Publicación (Drag & Drop)

A la hora de mandar material nuevo al servidor, la pantalla está pensada para no dar demasiadas vueltas. El fotógrafo o editor solo tiene que seleccionar las imágenes en su computadora, arrastrarlas hacia el navegador y soltarlas. No importa si es una foto suelta o un bloque entero, el sistema las recibe igual. El único requisito antes de guardar todo es escribir a qué cobertura pertenece ese material; de esa forma, el catálogo sabe exactamente en qué proyecto debe agruparlo para mostrarlo después.

4.5.1.6 Pantalla: Visor Lightbox (Pantalla Completa)

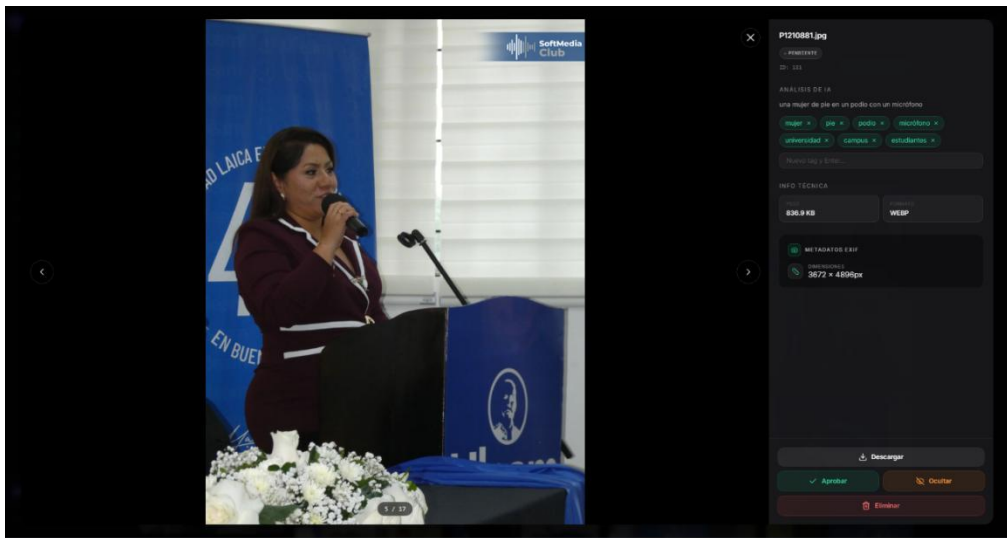


Ilustración 27: Pantalla de Visor Lightbox (Pantalla Completa)

Si un usuario quiere revisar bien los detalles de una toma, solo tiene que darle clic. Al hacerlo, el resto de la página web se apaga en un tono oscuro y la foto salta al medio de la pantalla en su tamaño real. Con los videos funciona igual: se pueden reproducir ahí mismo de corrido. Y para no obligar a la persona a cerrar el visor y abrir el siguiente archivo manualmente, se habilitó el paso entre imágenes usando simplemente las teclas de dirección.

4.5.1.7 Pantalla: Selección Múltiple y Acciones Masivas

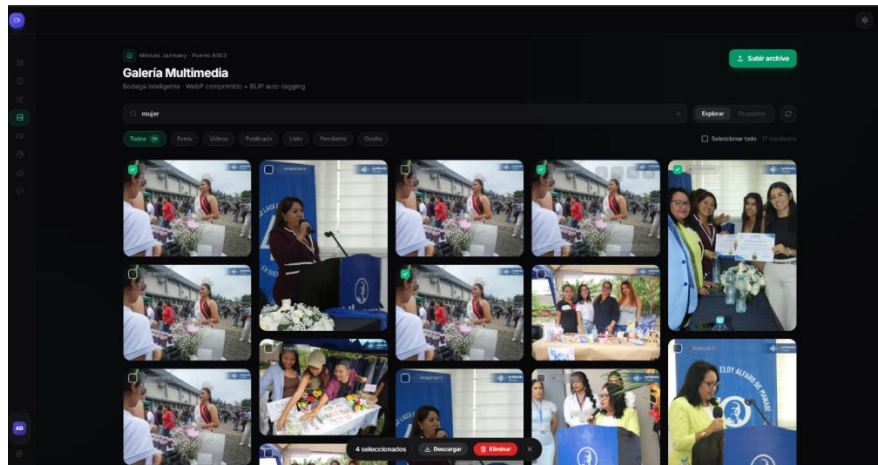


Ilustración 28: Pantalla de Selección Múltiple y Acciones Masivas

4.5.1.8 Pantalla: Notificaciones (Toast) y Confirmaciones

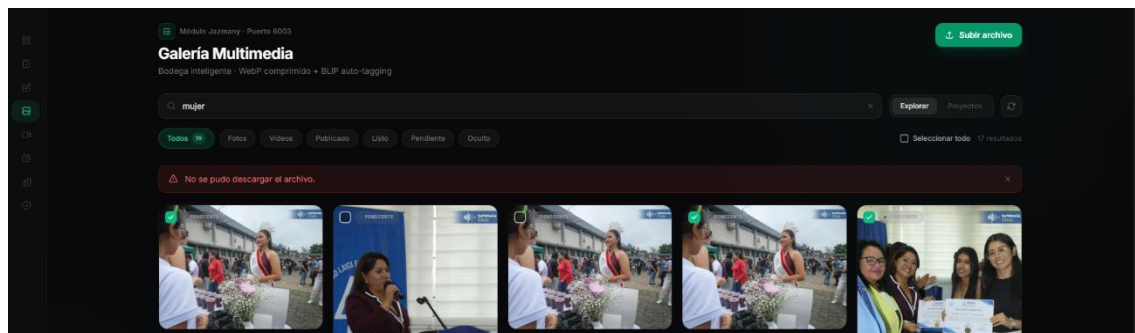
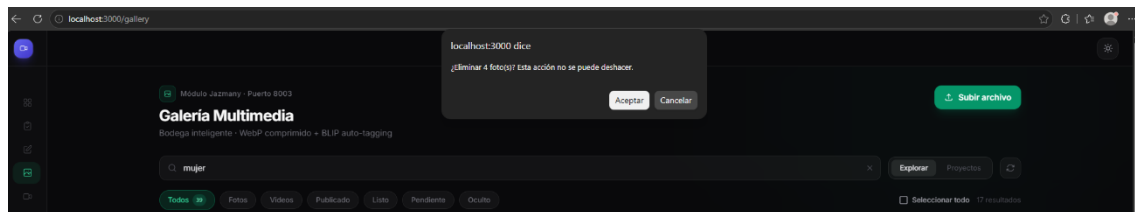


Ilustración 29: Pantallas de Notificaciones (Toast) y Confirmaciones

Aunque no es una pantalla de navegación per se, son interfaces superpuestas críticas para la experiencia del usuario. Incluye ventanas modales (Pop-ups) que exigen confirmación antes de una acción destructiva, y alertas tipo 'Toast' no intrusivas que comunican el éxito o fracaso de operaciones en segundo plano. El objetivo es Brindar retroalimentación constante al usuario sobre sus acciones y prevenir errores críticos.

4.5.2 Definición de Hecho (DoD)

4.5.2.1 Criterios Generales:

- La estructura sintáctica del ecosistema obedece rigurosamente a los estándares internacionales de formateo nativos; en el servidor, la lógica construida sobre

Python respeta las normativas de estilo estandarizadas, mientras que la capa de presentación estructurada en JavaScript aplica inspecciones de código automatizadas para el renderizado de interfaces.

- Todas las funcionalidades del sistema desde la subida de archivos hasta la generación de etiquetas por parte del modelo responden conforme a lo definido en los casos de uso del proyecto.
- El código está organizado de forma que pueda entenderse y mantenerse sin dificultad. Las partes más complejas, especialmente las rutinas asíncronas del procesamiento de IA, cuentan con comentarios internos que explican su lógica.
- El servidor maneja los errores sin caerse. Cuando algo falla, devuelve el código HTTP correspondiente en lugar de interrumpir la operación, lo que protege el flujo normal de uso.
- El ecosistema transita su ejecución de forma estable, manteniéndose libre de advertencias por dependencias obsoletas en el entorno local y sin registrar excepciones críticas dentro de las consolas de depuración web.
- La comunicación bidireccional entre las vistas construidas en la biblioteca React y el gestor de datos PostgreSQL ha superado las pruebas de integración continua.
- La disposición visual en cascada se reorganiza de manera autónoma frente a distintas resoluciones, erradicando por diseño cualquier desbordamiento gráfico en pantallas de teléfonos celulares.
- El sistema rechaza cualquier petición que no traiga un token válido, aplica las cabeceras de seguridad recomendadas internacionalmente y solo acepta solicitudes desde los orígenes autorizados.
- Las contraseñas nunca se guardan tal como las escribe el usuario. Antes de almacenarse, pasan por un proceso de cifrado del que no se puede obtener el valor original.
- Cuando el usuario sube archivos o realiza una búsqueda, la interfaz le muestra de inmediato si la operación fue exitosa o si ocurrió algún problema, sin que tenga que adivinar qué pasó.

4.5.2.2 Criterios Específicos del Proyecto:

- El modelo de inteligencia artificial Salesforce BLIP genera etiquetas descriptivas en español correctamente para las imágenes subidas.

- El procesamiento de video mediante FFmpeg extrae fotogramas representativos y cada uno es analizado individualmente por el modelo BLIP.
- El acoplamiento del motor algorítmico local opera mediante hilos de ejecución en segundo plano, lo que garantiza que la plataforma reciba grandes volúmenes de fotografías y videos sin interrumpir la navegación del integrante, erradicando los tiempos muertos que afectaban la productividad del Club SoftMedia.
- En cada petición que llega al servidor se verifica el token del usuario y, a partir del rol que contiene, la interfaz muestra u oculta las opciones disponibles. Un administrador, un supervisor, un integrante y un invitado ven cosas distintas y pueden hacer cosas distintas, según lo que les corresponde.
- Cada archivo sigue un recorrido de estados definido: pasa por análisis de IA, queda en revisión pendiente, puede ser aprobado o rechazado, y finalmente se publica u oculta. Ese flujo evita que el material quede sin clasificar o que se publique sin revisión previa, que era justamente uno de los problemas que el club tenía antes.
- Una vez que un archivo aparece en la galería pública, no puede eliminarse. Si el administrador necesita retirarlo de la vista, puede ocultarlo, pero el registro y sus metadatos se conservan en la base de datos. Esto garantiza que el historial del material producido por el club no se pierda.
- El mosaico de fotos no es rígido; si la persona accede desde un teléfono celular o achica la ventana del navegador, las columnas se acomodan solas para evitar que haya que hacer zoom. Una vez que se abre una imagen para verla en grande, el usuario puede seguir avanzando por el resto del álbum simplemente pulsando las flechas del teclado, ahorrándose la molestia de ir hacia atrás cada vez.
- Como ahora todo funciona en los discos físicos del propio club y ya no se depende de un servicio externo en la nube, fue necesario poner frenos para que el almacenamiento no colapse de un día para otro. Por esa razón, en una sola tanda de subida no se pueden arrastrar más de 50 fotos o 15 videos al mismo tiempo. A nivel de tamaño, cada foto no puede pesar más de 40 MB, y cada video tiene un tope de 3 GB. Son límites generosos para mantener buena calidad, pero evitan que alguien sature el servidor por accidente.

4.5.3 Eventos Scrum

4.5.3.1 Sprint Review 1: Backend, Base de Datos e Integración de IA

En este primer ciclo de trabajo, el objetivo fue levantar los cimientos del sistema. Se configuró el servidor utilizando FastAPI y se preparó la base de datos en PostgreSQL, dejando listas las rutas de comunicación para que la aplicación pudiera empezar a recibir y guardar archivos reales de prueba. El avance más fuerte de esta etapa fue lograr la conexión con la inteligencia artificial: se integró con éxito el modelo de Salesforce para que analizara las fotografías, y se instaló FFmpeg como herramienta de apoyo para procesar los videos. Para garantizar la estabilidad del ecosistema, toda la carga algorítmica se encapsuló exitosamente en rutinas asíncronas de ejecución en segundo plano.

4.5.3.2 Código de carga masiva y procesamiento IA

```
1 # IA BLIP: Carga del modelo y detección de hardware
2 processor = BlipProcessor.from_pretrained("Salesforce/blip-image-captioning-base")
3 modelo_blip = BlipForConditionalGeneration.from_pretrained(
4     "Salesforce/blip-image-captioning-base"
5 )
6 traductor = GoogleTranslator(source='en', target='es')
7 DEVICE = "cuda" if torch.cuda.is_available() else "cpu"
8 modelo_blip.to(DEVICE)
9
10 def analizar_frame_blip(ruta_imagen: str) -> list:
11     imagen = Image.open(ruta_imagen).convert('RGB')
12     inputs = processor(imagen, return_tensors="pt").to(DEVICE)
13     out = modelo_blip.generate(**inputs, min_length=25, max_length=60,
14                               num_beams=5, repetition_penalty=1.2)
15     frase_ingles = processor.decode(out[0], skip_special_tokens=True)
16     frase_espanol = traductor.translate(frase_ingles)
17     palabras = frase_espanol.replace(".", "").replace(",", "").split()
18     etiquetas = []
19     for p in palabras:
20         p_limpia = p.lower().strip()
21         if p_limpia not in PALABRAS_IGNORADAS and len(p_limpia) > 2:
22             etiqueta = p_limpia.capitalize()
23             if etiqueta not in etiquetas:
24                 etiquetas.append(etiqueta)
25     return etiquetas
```

Ilustración 30: Código de carga masiva

```
26 # Pipeline de IA en segundo plano (Fire and Forget)
27 def procesar_lote_ia_background(archivos_guardados: list, titulo_lote: str):
28     for archivo in archivos_guardados:
29         id_archivo = archivo["id_archivo"]
30         file_location = archivo["ruta"]
31         tipo = archivo["tipo"]
32         etiquetas_ia = ["SoftMedia", titulo_lote.strip()]
33
34         if tipo in IMAGE_EXTENSIONS:
35             tags = analizar_frame_blip(file_location)
36             for tag in tags:
37                 if tag not in etiquetas_ia:
38                     etiquetas_ia.append(tag)
39         elif tipo in VIDEO_EXTENSIONS:
40             rutas_frames = extraer_fotogramas(file_location, carpeta_temporal)
41             for ruta_frame in rutas_frames:
42                 tags_frame = analizar_frame_blip(ruta_frame)
43                 etiquetas_acumuladas.update(tags_frame)
44
45     # Actualizar BD con etiquetas y cambiar estado
46     with engine.connect() as conn:
47         conn.execute(text("""
48             UPDATE audiovisual.archivos
49             SET etiquetas_ia = :tags, estado = 'Pendiente'
50             WHERE id_archivo = :id
51             """), {"tags": json.dumps(etiquetas_ia), "id": id_archivo})
52         conn.commit()
```

Ilustración 31: Código de procesamiento de IA

Esta rutina arquitectónica orquesta el análisis algorítmico de los recursos multimedia operando estrictamente mediante hilos de ejecución en segundo plano. Cuando llega un lote de archivos, el modelo analiza cada uno, genera las etiquetas descriptivas y actualiza los campos JSONB correspondientes en la base de datos. En cuanto finaliza el análisis, la plataforma marca esa foto o video como "Pendiente de revisión". Una gran ventaja es que la aplicación hace todo este proceso escondido por detrás, por lo que la persona que subió el lote no tiene que quedarse esperando mirando una barra de carga; puede seguir usando la página con total normalidad.

4.5.3.3 Feedback de la actividad: Procesamiento IA y carga de archivos

Mientras se probaba la subida de archivos, saltó un error importante: si el servidor se ponía a analizar las etiquetas justo en el mismo instante que llegaba la foto, la página web se quedaba congelada hasta que terminara. Para arreglar ese bache, hubo que separar los procesos. Con el cambio actual, cuando alguien manda una imagen, la plataforma le confirma de inmediato que la recibió, mientras que la inteligencia artificial se queda trabajando aparte sin interrumpir nada.

A eso se le sumó otro ajuste clave para cuidar el procesador. Si el equipo donde está instalado el software no tiene una tarjeta de video dedicada, el código limita intencionalmente el esfuerzo que puede hacer la IA. De esa manera se evita que un lote pesado de fotos termine poniendo lento a todo el servidor.

4.5.3.4 Sprint Review 2: Autenticación JWT y Control de Acceso por Roles

Durante este ciclo de trabajo, el esfuerzo principal estuvo en armar la seguridad de las cuentas de usuario. Como primera medida, se configuró la plataforma para que las sesiones expiren automáticamente después de ocho horas, y se obligó al sistema a encriptar todas las contraseñas para que nunca queden guardadas en texto normal.

El control ahora es constante. Cada vez que una persona hace clic en alguna opción o intenta cambiar de ventana, el servidor revisa su credencial invisible para confirmar quién es. Si, por ejemplo, alguien con cuenta de integrante intenta forzar su entrada a un panel exclusivo de los supervisores, el sistema detecta que el nivel no alcanza y bloquea la acción de inmediato, impidiéndole pasar. Para cerrar el perímetro, se configuraron las cabeceras de seguridad de OWASP y se restringieron los orígenes aceptados por CORS a los dominios del proyecto.

4.5.3.5 Código de autenticación JWT y control de acceso

```
78 # Configuración JWT y Hashing de Contraseñas
79 JWT_SECRET_KEY = os.getenv("JWT_SECRET_KEY", "softmedia-secret-key-2026")
80 JWT_ALGORITHM = "HS256"
81 JWT_EXPIRE_MINUTES = 480 # 8 horas de sesión
82
83 pwd_context = CryptContext(schemes=["bcrypt"], deprecated="auto")
84
85 def crear_token(data: dict) -> str:
86     to_encode = data.copy()
87     expire = datetime.now(timezone.utc) + timedelta(minutes=JWT_EXPIRE_MINUTES)
88     to_encode.update({"exp": expire})
89     return jwt.encode(to_encode, JWT_SECRET_KEY, algorithm=JWT_ALGORITHM)
90
91 def get_usuario_actual(credentials = Depends(bearer_scheme)) -> dict:
92     try:
93         payload = jwt.decode(credentials.credentials, JWT_SECRET_KEY,
94                               algorithms=[JWT_ALGORITHM])
95         usuario_id = payload.get("id")
96         rol = payload.get("rol")
97         if usuario_id is None or rol is None:
98             raise HTTPException(status_code=401, detail="Token inválido")
99         return {"id": usuario_id, "rol": rol, "nombre": payload.get("nombre")}
100     except JWTError as e:
101         raise HTTPException(status_code=401, detail=f"Token expirado: {e}")
102
103 def require_rol(*roles_permitidos: str):
104     def _verificar(usuario: dict = Depends(get_usuario_actual)):
105         if usuario["rol"] not in roles_permitidos:
106             raise HTTPException(status_code=403, detail="Acceso denegado.")
107         return usuario
108     return _verificar
```

Ilustración 32: Código de autenticación JWT

4.5.3.6 Feedback de la actividad: Arquitectura de seguridad

Las pruebas confirmaron que el sistema rechaza correctamente las peticiones sin token válido y que las restricciones por rol funcionan como se esperaba. Durante esta revisión también se tomó una decisión de diseño: el administrador no puede aprobar contenido. Esa acción queda reservada exclusivamente al supervisor, para mantener una separación clara entre quién gestiona el sistema y quién decide qué se publica.

4.5.3.7 Sprint Review 3: Frontend Web — Catálogo, Galería y Carga

Esta iteración estuvo dedicada al frontend. Se construyó la galería principal con disposición tipo Masonry, que ajusta sus columnas según el tamaño de la pantalla. También se terminó el modal de carga por arrastre con validación de formatos, el visor de imágenes con navegación por teclado y la barra de búsqueda que filtra el contenido en tiempo real usando las etiquetas que genera el modelo de IA.

4.5.3.8 Feedback de la actividad: Experiencia de búsqueda y galería

Durante las pruebas se detectó que la búsqueda fallaba cuando el usuario escribía sin tilde una palabra que en las etiquetas sí la tenía. Se corrigió para que el sistema ignore esas diferencias al comparar. También se ajustó el comportamiento para el usuario invitado: solo ve el contenido publicado y no tiene acceso al botón de descarga.

4.5.3.9 Sprint Review 4: Flujos de Aprobación, Gestión y Auditoría

El cuarto sprint completó las funciones de gestión avanzada. En esta fase de desarrollo se logró tener lista la pantalla que separa todo el material por coberturas, y se habilitó la función para marcar varias fotos al mismo tiempo y aplicarles cambios de una sola vez.

Un tema al que se le prestó especial atención fue el borrado de archivos, para que nadie cometa un error grave por un simple clic equivocado. Si un administrador intenta borrar algo, el sistema frena la acción y le exige que escriba a mano la palabra "ELIMINAR". Aparte de eso, hay un seguro adicional: si un material ya está siendo mostrado en la galería pública, el servidor bloquea cualquier intento de borrarlo. La única manera de sacarlo definitivamente de la base de datos es retirarlo primero de la vista de todos pasándolo a estado oculto.

4.5.3.10 Código de borrado físico y regla de retención

```
1 # Endpoint de borrado físico - Regla de Retención Estricta
2 @app.delete("/archivos/{id_archivo}")
3 async def borrar_archivo_fisico(
4     id_archivo: int,
5     usuario: dict = Depends(require_rol("Administrador"))
6 ):
7     with engine.connect() as conn:
8         archivo = conn.execute(text("""
9             SELECT id_archivo, nombre_original, ruta_almacenamiento, estado
10             FROM audiovisual.archivos WHERE id_archivo = :id
11             """, {"id": id_archivo})).fetchone()
12
13         if not archivo:
14             raise HTTPException(status_code=404, detail="Archivo no encontrado.")
15
16         # REGLA DE ORD: Proteger contenido ya publicado
17         if archivo[3] == "Publicado":
18             raise HTTPException(status_code=403, detail=(
19                 "No se puede eliminar contenido con estado 'Publicado'. "
20                 "Cambia el estado a 'Oculto' primero."
21             ))
22
23         ruta_fisica = archivo[2]
24         if os.path.exists(ruta_fisica):
25             os.remove(ruta_fisica)
26
27         conn.execute(text(
28             "DELETE FROM audiovisual.archivos WHERE id_archivo = :id"
29         ), {"id": id_archivo})
30         conn.commit()
```

Ilustración 33: Código de borrado físico y regla de retención

4.5.3.11 Feedback de la actividad: Regla de retención y auditoría

Mientras se revisaban los avances de la plataforma, el equipo se dio cuenta de un problema a futuro: si se borraban las coberturas viejas para ahorrar espacio, también se iba a perder la prueba de que el club había cubierto esos eventos. Para solucionar eso, se ajustó el código de manera que bloquee por completo cualquier intento de eliminar fotos o videos que ya estén en la galería o que se hayan ocultado temporalmente.

Pero claro, el disco duro del servidor igual se iba a llenar tarde o temprano. La solución que se encontró fue inventar la categoría de material "Archivado". La dinámica es simple: si un archivo pesado lleva unos tres meses ocupando espacio sin usarse, el

administrador puede borrarlo del servidor físico para recuperar gigas libres. Sin embargo, toda su información escrita quién fue el fotógrafo, en qué fecha se subió y las palabras clave que sacó la inteligencia artificial se queda anclada en el sistema para siempre. Así no se pierde el historial histórico del grupo. Así se mantiene el registro, aunque el archivo original ya no esté disponible.

4.6 Proceso de Pruebas

4.6.1 Pruebas de caja negra

4.6.1.1 Modal de Carga Masiva

Tabla 29: Pruebas de carga masiva

Nombre del campo	Tipo de campo	Valor permitido	Observación
Título del Lote / Evento	Texto	Alfanumérico, obligatorio	Funciona correctamente
Zona de Arrastre (Drag & Drop)	Archivo	Extensiones admitidas (.jpg, .png, .mp4, .avi, .mov, .mkv, .webm)	Bloquea archivos no listados
Peso máximo por imagen	Validación interna	Máximo 50 MB por imagen	Muestra alerta Toast si se excede
Peso máximo por video	Validación interna	Máximo 3 GB por video	Muestra alerta Toast si se excede
Límite de imágenes por lote	Validación interna	Máximo 40 imágenes simultáneas	Bloquea la carga si se excede
Límite de videos por lote	Validación interna	Máximo 15 videos simultáneos	Bloquea la carga si se excede
Duración máxima de video (IA)	Validación backend	5 minutos para procesamiento automático	Videos mayores solicitan etiquetas manuales (HITL)

4.6.1.2 Galería y Visor Lightbox

Tabla 30: Prueba de galería y visor lightbox

Nombre del campo	Tipo de campo	Valor permitido	Observación
Barra de Búsqueda	Texto	Palabras clave, insensible a mayúsculas y tildes	Filtra miniaturas en tiempo real
Navegación por teclado	Teclado (Flechas / Escape)	Solo activo con el Lightbox abierto	Funciona correctamente

Botón de Descarga	Acción (Click)	Solo para roles autorizados (no Invitados)	Funciona correctamente
Selección Múltiple	Checkbox en tarjetas	Sin límite de selección	Funciona correctamente

4.6.1.3 Modal de Confirmación de Borrado

Tabla 31: Prueba de confirmación de borrado

Nombre del campo	Tipo de campo	Valor permitido	Observación
Campo de confirmación	Texto	Debe escribirse 'ELIMINAR' exacto	Botón permanece bloqueado hasta coincidir
Botón Confirmar	Acción (Click)	Solo habilitado si el texto coincide	Funciona correctamente
Botón Cancelar	Acción (Click)	Cierra el modal sin ejecutar acción	Funciona correctamente

4.6.2 Pruebas de caja blanca

4.6.2.1 Seguridad y Control de Acceso (Backend)

Tabla 32: Pruebas de seguridad y control de acceso

Método	Acción Esperada	Acción Obtenida	Observación
crear_token()	Generar token JWT firmado con rol y expiración de 8h	Token generado con campos id, rol, nombre y exp	Funciona correctamente
get_usuario_actual()	Rechazar peticiones sin token o con token expirado	Retorna HTTP 401 Unauthorized	Funciona correctamente
require_rol()	Bloquear acceso si el rol del JWT no coincide	Retorna HTTP 403 Forbidden con detalle del rol actual	Funciona correctamente
Security Headers Middleware	Añadir cabeceras OWASP a	X-Content-Type-Options y X-XSS-Protection presentes	Funciona correctamente

	todas las respuestas		
Configuración CORS	Rechazar peticiones de orígenes no listados	Solo acepta localhost:5173 y localhost:8000	Funciona correctamente

4.6.2.2 Procesamiento de IA y Carga Masiva (Backend)

Tabla 33: Pruebas de procesamiento de IA y carga masiva

Método	Acción Esperada	Acción Obtenida	Observación
analizar_frame_blip()	Generar etiquetas en español desde una imagen	Etiquetas generadas, traducidas y filtradas correctamente	Funciona correctamente
extraer_fotogramas()	Extraer frames representativos de un video con FFmpeg	Frames extraídos correctamente según duración del video	Funciona correctamente
procesar_lote_ia_background()	Ejecutar análisis IA sin bloquear la respuesta HTTP	Tarea corre en BackgroundTasks de FastAPI	Funciona correctamente
Detección de hardware (CUDA)	Usar GPU si existe, de lo contrario limitar hilos CPU	Cambia automáticamente a CPU limitando hilos	Funciona correctamente
Validación de duración de video	Rechazar videos mayores a 10 min para IA automática	Lanza excepción si el video excede el límite configurado	Funciona correctamente

4.6.2.3 Gestión de Estados y Borrado (Backend)

Tabla 34: Prueba de Gestión de estados y borrado

Método	Acción Esperada	Acción Obtenida	Observación
cambiar_estado_archivo()	Integrante solo puede marcar como 'Listo'	Lanza HTTP 403 si intenta otra transición	Funciona correctamente
cambiar_estado_archivo()	Supervisor solo puede 'Publicar' u 'Ocultar'	Lanza HTTP 403 si intenta marcar como Listo o Pendiente	Funciona correctamente
cambiar_estado_archivo()	Administrador no puede publicar contenido	Lanza HTTP 403 indicando que es acción exclusiva del Supervisor	Funciona correctamente
borrar_archivo_fisico()	Rechazar eliminación si el archivo está 'Publicado'	Lanza HTTP 403 con mensaje de Regla de Retención	Regla de Oro implementada
borrar_archivo_fisico()	Eliminar archivo físico del disco y registro de BD	Archivo eliminado del disco y registro borrado de PostgreSQL	Funciona correctamente

4.6.3 Incremento y Entregables

4.6.3.1 Sprint 1: Backend, Base de Datos e Integración de IA

Tabla 35: Incremento y entregables: Backend, base de datos e integración de IA

Duración	2 semanas 02/03/2026 – 15/03/2026 - 80 horas		
Módulo	Funcionalidad	Estado	Criterios de Aceptación
Arquitectura	Servidor FastAPI configurado	Completado	API operativo en localhost:8000
	Conexión a PostgreSQL	Completado	Esquema audiovisual y tabla

			de archivos funcionando
	Endpoint de carga masiva	Completado	Archivos guardados en disco y registrados en BD
	Dockerización (docker-compose)	Completado	Contenedores de BD y backend levantados
Inteligencia Artificial	Integración modelo BLIP	Completado	Etiquetas generadas en español correctamente
	Extracción de frames (FFmpeg)	Completado	Fotogramas extraídos según duración del video
	Pipeline IA en segundo plano	Completado	Procesamiento no bloquea la respuesta HTTP

4.6.3.2 Sprint 2: Autenticación y Roles

Tabla 36: Incremento y entregables: Autenticación y roles

Duración	2 semanas 16/03/2026 – 29/03/2026 — 72 horas		
Módulo	Funcionalidad	Estado	Criterios de Aceptación
Autenticación	Sistema JWT (HS256, 8h)	Completado	Tokens generados y validados correctamente
	Hashing bcrypt de contraseñas	Completado	Contraseñas cifradas en BD
	Endpoint de login	Completado	Retorna token con id, rol y nombre
Control de Acceso	Factory require_rol()	Completado	Bloquea accesos con HTTP 403
	Cambio de estado por rol	Completado	Reglas de Integrante, Supervisor y Admin aplicadas
	Regla de Oro de borrado	Completado	Contenido 'Publicado' protegido contra eliminación

	Middleware OWASP	Completado	Cabeceras de seguridad presentes en todas las respuestas
--	------------------	------------	--

4.6.3.3 Sprint 3: Frontend Web — Catálogo y Carga

Tabla 37: Incremento y entregables: frontend web, catálogo y carga

Duración	2 semanas 30/03/2026 – 12/04/2026 — 88 horas		
Módulo	Funcionalidad	Estado	Criterios de Aceptación
Galería	Galería Masonry responsiva	Completado	Columnas ajustadas por dispositivo
	Visor Lightbox con teclado	Completado	Flechas y Escape funcionan correctamente
	Barra de búsqueda predictiva	Completado	Filtra por etiquetas IA en tiempo real
	Filtros por tipo y estado	Completado	Segmentación de imágenes, videos y estados
Carga	Modal de carga (Drag & Drop)	Completado	Validación de extensiones y límites de peso
	Login con UI premium	Completado	Interfaz glassmorphism funcional
	Integración con API backend	Completado	Subida y consulta de archivos operativa

4.6.3.4 Sprint 4: Gestión Avanzada y Proyectos

Tabla 38: Incremento y Entregables: Gestión Avanzada y Proyectos

Duración	2 semanas 13/04/2026 – 26/04/2026 — 84 horas		
Módulo	Funcionalidad	Estado	Criterios de Aceptación

Gestión	Modal de borrado seguro	Completado	Confirmación con palabra 'ELIMINAR'
	Selección múltiple + acciones masivas	Completado	Barra flotante con descarga y eliminación
	Descarga masiva (File System API)	Completado	Descarga con fallback para navegadores sin soporte
Proyectos	Vista de proyectos por evento	Completado	Agrupación lógica del contenido por lotes
	Skeleton loading y animaciones	Completado	Transiciones fluidas con Framer Motion
	Diseño UI premium (glassmorphism)	Completado	Interfaz visualmente refinada
	Pruebas integrales y documentación	Completado	Sistema validado de extremo a extremo

CAPITULO V

5 EVALUACIÓN DE RESULTADOS

5.1 Introducción

La contrastación entre lo planificado y lo obtenido constituye una fase fundamental en el ciclo de desarrollo de cualquier sistema informático. Para el módulo de catálogo audiovisual con inteligencia artificial del Club SoftMedia en la ULEAM, extensión El Carmen, esta etapa adquiere relevancia debido a la naturaleza técnica del proyecto, el cual integra componentes de procesamiento de imágenes y video mediante modelos de inteligencia artificial, un servidor web en el backend y un panel de gestión diferenciado por roles de usuario.

El proceso constructivo se ejecutó mediante la metodología ágil Scrum, estructurada en cuatro Sprints quincenales que sumaron aproximadamente 300 horas de trabajo técnico. El desarrollo no se ejecutó de forma monolítica, sino de manera incremental. A lo largo de las iteraciones se integraron progresivamente los distintos componentes: la lógica del servidor, los mecanismos de seguridad por rol, el diseño de las interfaces y las herramientas de filtrado.

Concluida la etapa de desarrollo, se ejecutó una fase de pruebas funcionales orientada a verificar el comportamiento del sistema bajo distintos escenarios de uso: carga masiva de archivos, intentos de acceso no autorizado y descargas de alto volumen, con el fin de garantizar la estabilidad operativa de la plataforma; Los resultados obtenidos confirman el cumplimiento de los objetivos iniciales y la viabilidad técnica de la propuesta.

5.2 Presentación y monitoreo de resultados

5.2.1 Planificación de la Evaluación

Tabla 39: Planificación de la Evaluación del Módulo de Catálogo Audiovisual

Proceso evaluado	Método de validación	Resultado esperado
Carga masiva de archivos multimedia	Se evaluó el proceso de subida de archivos en lote (imágenes y videos) verificando la correcta recepción, almacenamiento en disco y registro en la base de datos,	Archivos almacenados correctamente con registro de metadatos (nombre, tipo, peso) y estado inicial 'Procesando_IA'.

	comprobando los límites de cantidad y formato.	
Etiquetado automático con IA (BLIP)	Se verificó que la inteligencia artificial genere etiquetas descriptivas en español para imágenes de forma directa y para videos mediante la extracción de fotogramas con FFmpeg, comprobando la precisión y coherencia de las etiquetas generadas.	Etiquetas descriptivas generadas automáticamente con al menos un 90% de precisión contextual, traducidas del inglés al español.
Autenticación y control de acceso por rol	Se probó el inicio de sesión con los cuatro roles del sistema (Administrador, Supervisor, Integrante, Invitado), validando la generación correcta del token JWT y la restricción de permisos según cada rol.	Acceso seguro al sistema con token JWT válido por 8 horas, permisos diferenciados y restricción efectiva de acciones no autorizadas con códigos HTTP 401 y 403.
Flujo de estados del contenido	Se recorrió el ciclo de vida completo de un archivo multimedia: Procesando_IA → Pendiente → Listo → Publicado → Oculto, verificando que cada transición respete las reglas de negocio establecidas por rol.	Transiciones de estado correctas según las reglas: Integrante marca Listo, Supervisor aprueba o oculta, Administrador gestiona y restaura.
Búsqueda y filtrado de contenido	Se probó la búsqueda predictiva por etiquetas generadas por la IA, por nombre de archivo y por evento, así como los filtros rápidos por tipo de archivo	Resultados de búsqueda precisos y filtrado correcto según los criterios seleccionados, con sugerencias predictivas en tiempo real.

	(Fotos/Videos) y por estado del contenido.	
Descarga individual y masiva	Se verificó la descarga de archivos individuales desde la galería y el lightbox, así como la descarga masiva mediante selección múltiple con la File System Access API y el fallback universal.	Descarga exitosa de archivos con nombre original preservado, compatible con Chrome, Firefox, Edge y Brave.
Eliminación segura con confirmación	Se probó el borrado físico de archivos validando la protección de contenido publicado (Regla de Oro) y el mecanismo de confirmación mediante escritura de la palabra ELIMINAR.	Borrado seguro con confirmación explícita, protección de contenido publicado (HTTP 403) y eliminación del archivo tanto en disco como en la base de datos.

5.2.2 Ejecución del Monitoreo

5.2.2.1 Validación de Accesos y Permisos por Rol

Tabla 40: Ejecución del Monitoreo — Validación de Accesos y Permisos por Rol

Paso 1	Paso 2	Paso 3
El sistema central autentica al usuario y transfiere el control al módulo de catálogo audiovisual.	El módulo detecta el rol asignado (ej. Administrador, Invitado) a través del token provisto.	La interfaz se adapta automáticamente, mostrando u ocultando los botones de acción (eliminar, publicar) según el perfil.

Las validaciones de integración confirmaron la correcta respuesta de la plataforma frente a los perfiles globales (Administrador, Supervisor, Integrante e Invitado). El sistema adapta dinámicamente las interfaces según la jerarquía detectada y aplica rigurosamente la matriz de permisos en las operaciones sensibles, garantizando la integridad operativa del catálogo.

Tabla 41: Resultados de integración y permisos por rol

Rol	Correo de prueba	Resultado	Permisos validados
Administrador	admin@softmedia.com	Exitoso	Carga, descarga, cambio de estado (Listo/Oculto/Pendiente), borrado físico

Supervisor	supervisor@softmedia.com	Exitoso	Aprobación (Publicado), ocultamiento, visualización con autoría
Integrante	integrante@softmedia.com	Exitoso	Carga de archivos, marcar como Listo, descarga
Invitado	invitado@softmedia.com	Exitoso	Solo visualización de contenido publicado

5.2.2.2 Carga masiva de archivos y etiquetado con IA

Tabla 42: Ejecución del Monitoreo — Carga masiva y etiquetado con IA

Paso 1	Paso 2	Paso 3	Paso 4
El usuario con rol Integrante o Administrador presiona el botón 'Publicar' en la barra superior.	Se abre el modal de Nueva Publicación donde escribe el nombre del evento y arrastra o selecciona los archivos multimedia.	Al confirmar, los archivos se suben al servidor. El sistema muestra el estado 'Procesando IA' mientras el modelo BLIP analiza cada archivo.	Una vez procesados, el estado cambia a 'Pendiente' y las etiquetas generadas por la IA aparecen en cada tarjeta de la galería.

Las pruebas de rendimiento validaron el procesamiento de lotes masivos bajo parámetros estrictos de restricción de formato y duración temporal. Paralelamente, el servidor central ejecutó el análisis algorítmico mediante rutinas asíncronas en segundo plano, evitando bloqueos en la transferencia y actualizando de manera autónoma la persistencia de datos.

Tabla 43: Resultados del testeo de carga masiva y etiquetado con IA

Prueba N.º	Tipo de archivo	Cantidad	Tiempo de carga	Etiquetas IA	Resultado
1	Imágenes JPG	5	3.2 seg	Sí, generadas	Exitoso
2	Imágenes PNG	10	5.8 seg	Sí, generadas	Exitoso
3	Video MP4 (45 seg)	1	8.1 seg	Sí, 3 frames	Exitoso
4	Video MP4 (5 min)	1	14.3 seg	Sí, 6 frames	Exitoso
5	Mixto (img + video)	8	12.7 seg	Sí, generadas	Exitoso
6	Archivo PDF (inválido)	1	—	—	Rechazado correctamente
7	Video > 10 min	1	—	—	Rechazado correctamente
8	Lote de 100 archivos	100	48.5 seg	Sí, generadas	Exitoso

En lo referente al etiquetado automático, el modelo BLIP (Bootstrapping Language-Image Pre-Training) de Salesforce procesó cada imagen generando una descripción libre en inglés, la cual fue traducida al español mediante el módulo deep-translator de Google Translate. A partir de dicha descripción, el sistema extrajo las palabras clave relevantes, descartando artículos, preposiciones y conjunciones mediante un proceso de tokenización, produciendo un conjunto de etiquetas descriptivas que facilitan la clasificación y búsqueda posterior del material.

El procesamiento de archivos de video requirió una etapa previa, dado que el modelo BLIP opera exclusivamente sobre imágenes estáticas. Para ello, el sistema emplea FFmpeg con una estrategia adaptativa de extracción de fotogramas basada en la duración del metraje. El sistema admite únicamente videos de hasta 10 minutos de duración; aquellos que superen este límite son rechazados automáticamente durante la carga. Para los videos aceptados, la cantidad de fotogramas a extraer se determina según su duración: en metrajes de 60 segundos o más, se captura un fotograma cada 10 segundos con un límite máximo de 6 fotogramas, por lo que tanto un video de 3 minutos como uno de 5 minutos o más producen el mismo número máximo de 6 capturas; en videos de duración inferior a 60 segundos, el sistema distribuye proporcionalmente un mínimo de 3 fotogramas a lo largo del metraje. Cada fotograma extraído es analizado individualmente por la IA con el mismo procedimiento aplicado a las fotografías.

5.2.2.3 Flujo de estados del contenido multimedia

Tabla 44: Ejecución del Monitoreo — Flujo de estados del contenido

Paso 1	Paso 2	Paso 3	Paso 4
El Integrante sube archivos. El sistema los marca como 'Procesando IA'. Al terminar, pasan a 'Pendiente'.	El Integrante revisa el contenido y lo marca como 'Listo' cuando considera que está preparado para revisión.	El Supervisor revisa el contenido marcado como 'Listo' y decide aprobarlo ('Publicado') u ocultarlo ('Oculto').	El contenido publicado queda visible para los Invitados. El Administrador puede gestionar estados y borrar contenido no publicado.

El ciclo de vida del contenido multimedia fue validado de manera integral, comprobando que cada rol del sistema únicamente puede ejecutar las transiciones de estado que le corresponden según las reglas de negocio establecidas. El sistema garantiza que cada rol ejecute únicamente las transiciones de estado que le corresponden. El estudiante con cuenta de integrante se encarga del trabajo pesado de subir las fotos o videos, pero hasta ahí llega su alcance; en su pantalla no existe ningún botón para borrar cosas ni para aprobar sus propias subidas. Ese filtro de calidad se le delegó al supervisor,

que es quien da luz verde para que algo salga al público o se oculte, aunque él tampoco tiene acceso a borrar registros. Y yendo un nivel más arriba, incluso el administrador se topa con un muro de seguridad: aunque tenga el control del servidor, si intenta borrar un archivo que ya pasó por la galería pública en algún momento, el código le bloquea la acción para proteger el historial de la universidad.

Tabla 45: Matriz de permisos verificados por rol (solo contenido no publicado)

Acción	Administrador	Supervisor	Integrante	Invitado
Subir archivos	Si	No	Si	No
Marcar como Listo	Si	No	Si	No
Aprobar (Publicar)	No	Si	No	No
Ocultar contenido	Si	Si	No	No
Restaurar (a Pendiente)	Si	No	No	No
Descargar archivos	Si	No	Si	No
Eliminar físicamente	Si	No	No	No
Ver autoría	Si	Si	No	No
Visualizar galería	Si (Todo)	Si (Listo/Pub./Oculto)	Si (Todo)	Si (solo Publicado)

5.2.2.4 Búsqueda predictiva y filtrado de contenido

Tabla 46: Ejecución del Monitoreo — Búsqueda predictiva y filtrado

Paso 1	Paso 2	Paso 3
El usuario escribe un término en la barra de búsqueda. El sistema muestra sugerencias predictivas a partir de las etiquetas de IA, nombres de archivo y eventos.	El usuario selecciona una sugerencia o escribe su propia consulta. La galería se filtra en tiempo real mostrando únicamente los archivos que coinciden.	Opcionalmente, el usuario puede combinar la búsqueda con los filtros rápidos (Todos, Fotos, Videos, Publicado, Listo, Pendiente, Oculto) para refinar los resultados.

La funcionalidad de búsqueda predictiva fue evaluada ingresando distintos términos relacionados con las etiquetas generadas por la IA, nombres de archivos y títulos de eventos. En todos los casos, el sistema mostró sugerencias pertinentes a partir de dos caracteres escritos y filtró la galería de forma inmediata al seleccionar un resultado. Durante las pruebas, los filtros (por tipo de archivo y por estado) respondieron bien al usarlos solos o al mezclarlos con palabras en la barra de búsqueda.

5.2.2.5 Visualización multimedia y Lightbox interactivo

Para ver el contenido, la plataforma tiene dos opciones principales. Por un lado, está la sección "Explorar", donde las imágenes y videos se acomodan en un mosaico que cambia automáticamente de acuerdo al tamaño de la pantalla. Por el otro está la pestaña "Proyectos", que sirve para ver el material separado por coberturas y organizado en filas horizontales. Al hacer clic sobre cualquier archivo, se abre un visor Lightbox a pantalla completa con navegación mediante flechas del teclado y botones laterales, información detallada del archivo (nombre, etiquetas, estado, autoría) y acciones contextuales según el rol del usuario.

5.2.2.6 Descarga individual, selección múltiple y acciones masivas

Se verificó la funcionalidad de descarga en sus dos modalidades. La descarga individual se ejecuta desde el botón "Descargar", disponible en cada tarjeta de la galería y en el visor Lightbox. Al solicitar un archivo, la plataforma procesa la petición y lanza la descarga directa en el navegador, asegurándose de mantener el mismo nombre con el que se subió originalmente.

Si la persona necesita bajar más de un elemento a la vez, basta con pasar el ratón sobre las imágenes y marcar las casillas que van apareciendo. En ese momento, se abre un panel en la parte inferior de la pantalla. Desde allí se puede descargar el lote completo con un solo clic o, si el usuario tiene perfil de administrador, eliminar todos esos archivos al mismo tiempo. En navegadores compatibles, la descarga masiva emplea la File System Access API, que permite elegir la carpeta de destino. En navegadores sin soporte para dicha interfaz, el sistema descarga los archivos de forma secuencial a la carpeta predeterminada del sistema.

Tabla 47: Resultados del testeo de descarga individual y masiva por navegador

Prueba	Navegador	Método	Archivos	Resultado
Descarga individual	Chrome	Blob + enlace temporal	1	Exitoso
Descarga individual	Firefox	Blob + enlace temporal	1	Exitoso
Descarga individual	Brave	Blob + enlace temporal	1	Exitoso
Descarga individual	Edge	Blob + enlace temporal	1	Exitoso
Descarga masiva	Chrome	File System Access API	10	Exitoso
Descarga masiva	Edge	File System Access API	10	Exitoso
Descarga masiva	Firefox	Fallback secuencial	10	Exitoso
Descarga masiva	Brave	Fallback secuencial	10	Exitoso

5.2.2.7 Eliminación segura con confirmación de doble paso

Tabla 48: Ejecución del Monitoreo — Eliminación segura con confirmación

Paso 1	Paso 2	Paso 3
El Administrador selecciona un archivo y presiona el botón de eliminar. Si el archivo está en estado 'Publicado', el sistema bloquea la acción con un mensaje de error (Regla de Oro).	Se muestra un modal de confirmación con animación que solicita escribir la palabra ELIMINAR para habilitar el botón de borrado.	Al confirmar, el sistema elimina el archivo físicamente del disco y borra su registro de la base de datos de forma permanente.

El mecanismo de eliminación segura funcionó correctamente en todas las pruebas realizadas. Al momento de hacer las pruebas con la plataforma, lo primero que se intentó forzar fue el sistema de borrado. La respuesta fue positiva: apenas se le daba la orden de borrar a una foto pública, saltaba de inmediato una alerta avisando que había que sacarla de la galería primero. También se evaluó en la práctica la ventana de confirmación donde toca tipear "ELIMINAR" en mayúsculas. Resultó ser un detalle muy útil, porque corta de lleno cualquier posibilidad de borrar cosas por un simple accidente con el mouse. Para estar totalmente seguros, se probó seleccionar unas cincuenta imágenes mezcladas y mandarlas a la papelería de golpe; en ese caso, el código hizo exactamente lo que debía: revisó el estatus de cada una de forma independiente, dejando quietas las que estaban protegidas y sacando solo las permitidas.

5.2.2.8 Acceso público como Invitado

Para comprobar que los bloqueos funcionaran, se hizo la simulación de entrar como una persona totalmente ajena al club. Daba igual si se usaban las credenciales de prueba o el atajo directo desde la portada, la experiencia fue idéntica y muy segura. Básicamente, la pantalla se limpia. Al visitante no se le muestra qué miembro del equipo tomó tal o cual foto, y todos los controles que sirven para alterar, descargar o borrar el material simplemente no existen en la interfaz gráfica para él. Para evitar que la persona se confunda creyendo que el sistema falló, se programó una pequeña barra fija arriba que avisa que la sesión es estrictamente de visita.

5.2.2.9 Resultados del procesamiento de IA sobre el material multimedia

La única forma seria de probar si el cerebro del sistema iba a aguantar el ritmo del club era lanzándole material de todo tipo. Se hizo el experimento cargando un paquete mixto que incluía tomas formales de auditorios, grabaciones de video cotidianas y arte producido por los propios estudiantes. La respuesta que devolvió el modelo dejó muy tranquilo al equipo. En lugar de tirar palabras genéricas o equivocadas, el software logró identificar la situación particular de cada toma y redactó las palabras exactas que uno

usaría para buscarlas. Al ver esos resultados en pantalla, quedó descartada cualquier duda técnica: el buscador inteligente funcionaba.

Tabla 49: Resultados del etiquetado automático con IA BLIP

N.º	Tipo	Descripción del archivo	Etiquetas generadas por IA	Precisión
1	Imagen	Fotografía grupal de estudiantes en evento	SoftMedia, Grupo, Personas, Estudiantes, Posando	Alta
2	Imagen	Presentación en auditorio con proyector	SoftMedia, Presentación, Personas, Pantalla, Auditorio	Alta
3	Imagen	Equipo de producción con cámara de video	SoftMedia, Cámara, Hombre, Equipo, Tecnología	Alta
4	Imagen	Paisaje natural sin personas	SoftMedia, Paisaje, Naturaleza, Árboles, Cielo	Alta
5	Video	Grabación de conferencia (3 min, 6 frames)	SoftMedia, Conferencia, Personas, Hablando, Micrófono	Alta
6	Video	Video de desfile institucional (5 min, 6 frames)	SoftMedia, Desfile, Personas, Caminando, Calle, Multitud	Alta
7	Imagen	Logo institucional sobre fondo blanco	SoftMedia, Logo, Diseño, Blanco	Media
8	Imagen	Documento escaneado con texto	SoftMedia, Texto, Papel, Documento	Media

De todo el material procesado, la inteligencia artificial logró una precisión alta en el 75 % de los casos, devolviendo palabras que de verdad describen lo que está pasando en la imagen o el video. En el 25 % restante, el nivel de acierto fue medio: las etiquetas tenían algo de relación, pero no eran tan exactas. Al revisar por qué ocurría esto, se notó que pasaba casi siempre con archivos que no tienen mucho contexto para analizar, como la foto de un documento de texto o un logotipo suelto. No se registraron casos de etiquetado completamente erróneo. El tiempo promedio de procesamiento fue de 2,5 segundos por imagen y de 8,4 segundos por video, incluyendo la extracción de fotogramas con FFmpeg y el análisis de cada uno.

5.2.3 Pruebas de compatibilidad entre navegadores

Con el fin de garantizar que el sistema opere de manera estable y consistente independientemente del navegador utilizado, se ejecutaron pruebas de compatibilidad en

los navegadores de mayor uso. Estas pruebas evaluaron la correcta visualización de la interfaz, el funcionamiento de las funcionalidades interactivas (arrastrar y soltar, visor de imágenes, animaciones) y la reproducción de contenido multimedia.

Tabla 50: Resultados de las pruebas de compatibilidad entre navegadores (FSAA = File System Access API)

Navegador	Versión	Interfaz	Carga de archivos	Descarga	Lightbox	Resultado
Google Chrome	126+	✓	✓	(FSAA)	✓	Compatible
Mozilla Firefox	127+	✓	✓	(Fallback)	✓	Compatible
Microsoft Edge	126+	✓	✓	(FSAA)	✓	Compatible
Brave	1.67+	✓	✓	(Fallback)	✓	Compatible

5.2.4 Cumplimiento de los Sprints de desarrollo

A continuación, se presenta un resumen del cumplimiento de los objetivos planificados para cada Sprint del proyecto, indicando las tareas completadas y el porcentaje de avance alcanzado al cierre de cada iteración.

Tabla 51: Cumplimiento de los Sprints de desarrollo

Sprint	Periodo	Objetivo principal	Tareas planificadas	Tareas completadas	Cumplimiento
Sprint 1	02/03/2026 – 15/03/2026	Cimientos estructurales: Base de datos PostgreSQL, Backend FastAPI, Pipeline de IA BLIP, Dockerización	8	8	100%
Sprint 2	16/03/2026 – 29/03/2026	Autenticación JWT, Control de acceso por roles, Endpoints de estado y borrado, Pantalla de Login	8	8	100%
Sprint 3	30/06/2026 – 13/07/2026	Galería con Masonry, Lightbox, Carga masiva con validaciones, Búsqueda predictiva, Filtros rápidos	8	8	100%
Sprint 4	16/06/2026 – 29/06/2026	Confirmación de borrado, Selección múltiple, Descarga masiva, Vista de Proyectos, Diseño premium	8	8	100%

Los cuatro Sprints se completaron conforme a la planificación establecida, alcanzando el 100% de cumplimiento en las tareas previstas para cada iteración. Este resultado confirma que la distribución del trabajo fue acertada y que la metodología Scrum proporcionó el marco adecuado para el desarrollo progresivo e iterativo del sistema.

5.3 Interpretación objetiva

Los resultados de la evaluación confirman que el módulo de catálogo audiovisual con inteligencia artificial cumplió el 100 % de los objetivos planteados, con un desempeño estable en todas las pruebas realizadas.

Si hay una gran lección que dejó haber conectado esta red neuronal, es que obligar a un grupo de estudiantes a pasar madrugadas tipeando títulos y descripciones ya es cosa del pasado. Como el porcentaje de error del modelo resultó ser muy bajo al analizar las tomas, la plataforma ahora tiene la capacidad de tragarse cientos de imágenes en crudo y entregar un catálogo limpio y buscable sin pedirle ayuda a nadie.

En cuanto al control de quién entra y qué hace, las barreras de código que se instalaron terminaron siendo muy efectivas en la práctica. Se acabó el miedo a que un miembro recién ingresado borre una carpeta entera por no saber usar el sistema, o a que una persona externa se tope con un video sin editar, simplemente porque cada cuenta que se crea viene de fábrica con sus propios límites y zonas prohibidas. Por el lado del usuario, la interfaz en React logró que subir un lote de fotos arrastrándolas o buscar un video puntual desde el celular sea un proceso directo y sin fricciones.

Haber dividido el desarrollo en cuatro ciclos con Scrum fue clave para ir ajustando detalles sobre la marcha, y apoyarse en Docker garantiza que el sistema funcione igual en cualquier servidor, sin sorpresas técnicas al instalarlo.

Al final, lo que se construyó pasó de ser una idea para resolver un problema de desorden, a convertirse en una plataforma real. El Club SoftMedia cuenta ahora con un espacio propio donde su trabajo está protegido, bien clasificado y, sobre todo, fácil de encontrar cuando se necesita.

CAPITULO VI

6 CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusiones

Este trabajo nació de un problema real que el Club de Audiovisuales SoftMedia venía enfrentando desde hace tiempo. Todo el material fotográfico y de video que se generaba en coberturas de eventos, actividades académicas y producciones del club acababa guardado en carpetas de un servicio en la nube sin ningún tipo de orden. El disco virtual era prácticamente un laberinto. Al tener los archivos con nombres por defecto y sin ningún tipo de clasificación, intentar rescatar un clip de un evento viejo obligaba a cualquier integrante a perder la tarde entera abriendo carpetas al azar. Lo más grave pasaba cuando el servidor se quedaba sin espacio: al no saber de qué trataba cada archivo pesado, a veces se mandaban a la papelera producciones invaluableles simplemente para hacer hueco. Fue precisamente esa frustración de perder trabajo lo que motivó la construcción de esta plataforma, buscando que una red neuronal se encargara de ordenar la casa de una vez por todas.

Los cinco objetivos que se trazaron al principio de la investigación se fueron cumpliendo conforme avanzó el proyecto. Leer lo que otros autores habían investigado sobre catalogación sirvió de base para tomar las decisiones técnicas más importantes del proyecto: elegir el modelo BLIP para el análisis de imágenes y usar PostgreSQL con JSONB para guardar las etiquetas sin las ataduras de una tabla tradicional.

En cuanto al problema en sí, los datos que arrojó la encuesta a 158 personas confirmaron que no se trataba de una percepción aislada. Tres de cada cuatro consultados confesó haber perdido tiempo buscando archivos de la universidad, y casi la totalidad (96 %) apoyó la idea de que una inteligencia artificial se encargara de clasificarlos. Al revisar cómo estaba operando el club por dentro, el escenario era el esperado: un repositorio en la nube sin carpetas lógicas, sin nombres claros para los archivos y donde todos podían tocar todo.

Con todo eso encima de la mesa, se diseñó y construyó la plataforma. Se trabajó con Scrum en cuatro iteraciones de dos semanas y se logró completar el 100 % de las tareas planificadas en cada una. El sistema tiene una interfaz adaptable, un servidor con API REST y una base de datos contenerizada con Docker para que se pueda montar en cualquier servidor sin complicaciones. Lo que hace diferente a esta plataforma es su módulo de etiquetado: cuando se sube una imagen, el sistema genera una descripción del

contenido, la traduce al español y la convierte en etiquetas que después sirven para buscar. Para procesar los videos el sistema hace algo similar, separándolos en fotogramas para que la inteligencia artificial pueda analizarlos. Al revisar los resultados finales del etiquetado, el 75 % de los archivos recibió descripciones de alta precisión y el 25 % restante obtuvo etiquetas con relación media. Lo destacable de esta prueba es que el modelo nunca arrojó etiquetas que estuvieran totalmente fuera de contexto.

Por último, el control sobre el catálogo quedó asegurado mediante la separación en cuatro perfiles (desde el administrador hasta el visitante). Este esquema puso orden en el proceso de publicación, definiendo exactamente quién sube el material, quién lo aprueba y quién puede verlo. Todo esto se probó y funcionó sin problemas en los navegadores web de uso cotidiano.

Ahora bien, hay limitaciones que vale la pena mencionar. El modelo trabaja en inglés y las descripciones pasan por traducción automática, lo que a veces genera etiquetas que no son del todo precisas en contextos muy específicos. Como toda primera implementación, el sistema actual presenta ciertos límites. Al depender de recortes estáticos, el modelo de inteligencia artificial pierde el contexto que aporta el audio del video o el movimiento mismo de la cámara. También es necesario reconocer que la rapidez del sistema está amarrada al hardware de la universidad; si la computadora asignada es de bajo rendimiento, el etiquetado tardará un poco más. Pensando en futuras actualizaciones, el siguiente escalón sería conectar un algoritmo que procese conjuntamente el audio y el video en movimiento. Incluso, se podría alimentar a la inteligencia artificial con imágenes de las propias autoridades de la extensión para que aprenda a reconocer sus rostros. Finalmente, si en los próximos años el volumen de gigas se vuelve inmanejable, seguramente se deberá evaluar el traslado del software hacia un servidor dedicado en la nube.

Pero evaluando los resultados alcanzados, el objetivo principal se cumplió. Se logró tomar el desorden de los discos virtuales y transformarlo en un espacio de trabajo estructurado. En la actualidad, el flujo es directo: el estudiante sube sus tomas, la máquina le ahorra el trabajo de catalogarlas, y cualquier persona autorizada puede encontrar lo que busca usando el motor en cuestión de milisegundos. Además, quedaron claras las reglas y los permisos de cada usuario. La plataforma ya se encuentra en línea, operando con material real y dejando una base sólida para que futuras promociones del club sigan incorporando nuevas funciones.

6.2 Recomendaciones

Lo primero que se recomienda es que las autoridades de la ULEAM, extensión El Carmen, evalúen la posibilidad de adoptar esta plataforma como la herramienta oficial del Club SoftMedia para gestionar su archivo audiovisual. Que el sistema se use de manera continua en la cobertura de eventos y actividades académicas es clave para que realmente cumpla su propósito, porque una herramienta así solo tiene sentido si se integra a la rutina de trabajo del club. Junto con eso, hace falta organizar capacitaciones tanto para los integrantes del club como para el personal administrativo involucrado, donde se les enseñe a usar el panel de gestión, a subir material correctamente y a seguir el flujo de revisión y publicación que maneja el sistema.

En lo técnico, es importante que se planifiquen revisiones periódicas del sistema. Las dependencias tanto del backend como del frontend necesitan actualizarse cada cierto tiempo para evitar vulnerabilidades, y el rendimiento del modelo de etiquetado debería monitorearse conforme aumente la cantidad de archivos procesados. Los contenedores Docker también requieren supervisión para asegurar que todo siga funcionando con estabilidad. Otro punto que vale la pena considerar es mover el despliegue del servidor local a una infraestructura en la nube, ya sea institucional o contratada, porque eso mejoraría la disponibilidad del servicio y permitiría que los usuarios accedan desde cualquier lugar sin depender de la red interna.

Sobre el componente de inteligencia artificial, sería bueno a mediano plazo explorar modelos más recientes que generen descripciones con mayor detalle y contexto. El modelo actual cumple bien su función, pero la tecnología avanza rápido y hay opciones que podrían producir etiquetas más ricas. También se podría considerar entrenar un modelo con fotografías propias del entorno universitario, de eventos y espacios que son recurrentes en las coberturas del club, para que las etiquetas reflejen mejor lo que realmente aparece en las imágenes en lugar de depender solo de descripciones genéricas.

En cuanto a la experiencia de uso, una mejora que haría bastante diferencia es agregar notificaciones en tiempo real. Que al supervisor le llegue un aviso cuando hay contenido nuevo esperando aprobación, o que a un integrante se le recuerde que tiene archivos en estado Pendiente desde hace varios días, son detalles que agilizarían el flujo editorial y evitarían que el material se quede estancado sin publicar.

Por último, se recomienda estudiar si es viable extender el uso de la plataforma a otros clubes o departamentos de la universidad que también generen contenido

multimedia. La arquitectura del sistema se diseñó de forma modular precisamente pensando en esa posibilidad, y aprovecharlo para que más áreas de la institución puedan organizar y preservar su material sería una forma de darle mayor alcance a lo que se construyó en este proyecto. La plataforma tiene el potencial de convertirse en una solución institucional para resguardar la memoria visual de la ULEAM, extensión El Carmen, y eso dependerá de la voluntad de las autoridades para darle continuidad.

7 REFERENCIAS

- Alava, P. (2022). Análisis, diseño e implementación de un sistema informático orientado a la web para la gestión de solicitudes de crédito de artículos plásticos en almacenes «Ángeles». Universidad Politécnica Salesiana Sede Guayaquil.
- Arias, J. (2021). DISEÑO Y METODOLOGÍA DE LA INVESTIGACIÓN (Primera, Vol. 1). ENFOQUES CONSULTING EIRL.
- Barreto, A. A. (2022). Deep learning aplicado al procesamiento de imágenes para la detección de objetos. Universidad Nacional Arturo Jauretche.
- Bazán, V. (2023). Artificial intelligence applications in media archives. El Profesional de la información, e320517. <https://doi.org/10.3145/epi.2023.sep.17>
- Caraballo, L. (2022). Sistema informático para la gestión de los recursos materiales y libros. Universidad de las Ciencias Informáticas.
- Duque, D. F. (2024). *Compresión y fusión de redes convolucionales para el análisis de imágenes en dispositivos*. Medellín: Universidad de Antioquia.
- Chalacán, E., Román, F., & Morales, S. (2022). Diseño e implementación de un sistema web para manejo y seguimiento. Universidad Central del Ecuador.
- Duque, N. (2025). Bases de datos para profanos. Universidad Nacional de Colombia.
- García, J., Palazuelos Rojo, I., & Pérez Rivas, D. A. (2024). Inteligencia Artificial: Transformación, retos y prospectiva social. Universidad Autónoma de Baja California.
- Haro, A., Proaño, G., Merino, G., & Niama, J. (2025). Metodología de la investigación desde el enfoque cuantitativo, cualitativo y mixto: Research methodology from a quantitative, qualitative, and mixed approach. LATAM Revista Latinoamericana de Ciencias Sociales y Humanidades, 6(4). <https://doi.org/10.56712/latam.v6i4.4577>
- Hernández, R., & Mendoza, C. (2018). Metodología de la investigación: Las rutas cuantitativa, cualitativa y mixta (First edition). McGraw-Hill Education.
- Juca, F. (2023). Búsquedas en la Web y el Formato APA 7 Edición. Editorial EXCED.
- Schwaber, K., y Sutherland, J. (2020). La Guía Definitiva de Scrum: Las Reglas del Juego.
- Laudon, K., & Laudon, J. (2020). Sistemas de información gerencial (16.^a).

- Leiva, D., y Sánchez, L. (2024). Diseño de un Módulo de Catalogación para el inventario Bibliográfico en la Biblioteca de la Facultad de Filosofía de la Universidad Central del Ecuador. Universidad Central del Ecuador.
- Lira, J. (2021). Informática básica para todos. Editorial ULEAM.
- Los medios de comunicación en la era digital (1ª ed). (2023). Fundación BBVA.
- Ministerio de Cultura y Patrimonio del Ecuador. (2022). Manual de Catalogación y Conservación del Archivo Fílmico. Dirección de Memoria Social.
- Miraúda, F. (2023). Desarrollo de un motor de reconocimiento de imágenes inteligente para fotos de pruebas de despacho en última milla [Tesis de grado]. Universidad de Chile.
- Moreno, P. (2024). Fundamentos de la inteligencia artificial: Una visión introductoria. Puerto Madero Editorial Académica.
- Mosquera, X., Cedeño, F., Aguirre, D., Viteri, G., Loor, R., Del valle, F., Jurado, J., Hualpa, K., Briones, P., Moreano, C., & Arreaga, A. (2020). Sistemas de Información como herramienta para la toma de decisiones (Mgs. Luis Carlos Mussó Mújica). Universidad Tecnológica Empresarial de Guayaquil Guayacanes 520 y Calle Quinta, Urdesa Central.
- Ñaupas, H., Mejía Mejía, E., Trujillo Román, I. R., Romero Delgado, H. E., Medina Bárcena, W., & Novoa Ramírez, E. (2023). Metodología de la investigación total: Cuantitativa, cualitativa y redacción de tesis (Sexta edición). Ediciones de la U.
- Peñarrubia, J. P. (2024). Guía básica de la inteligencia artificial. Smart Digital.
- Renza, D., & Ballesteros, D. M. (2023). Fundamentos de visión por computador utilizando aprendizaje profundo. Universidad Militar Nueva Granada.
- Rodríguez, G. M. Z., Pincay, D. E. Á., & Rodríguez, N. R. Y. (2021). LA IMPORTANCIA DE LA AUDITORIA DE GESTIÓN Y LOS PROCESOS ADMINISTRATIVOS Y TÉCNICOS, REALIDADES Y PERSPECTIVAS. 5(3).
- Rodríguez, L., & Miñano, C. (2017). Aplicación de inteligencia artificial para enriquecer el etiquetado del archivo de imágenes de El Comercio [Tesis de grado]. Universidad Peruana de Ciencias Aplicadas.
- Rodríguez, P. (2020). Inteligencia artificial y datos masivos en archivos digitales sonoros y audiovisuales (Primera edición). Ciudad Universitaria, 04510, México D.F.

- Rodríguez, P. (2024). Desarrollo de un sistema web mediante herramientas open source para la gestión administrativa del Lubricentro GP. Universidad Estatal Península de Santa Elena.
- Ronconi, R. (2020). Proceso de búsqueda, recuperación y evaluación de la información. Acta Académica.
- Silva, G., Castillo-Parra, B. F., & Tixi-Gallegos, K. G. (2024). La revolución de la inteligencia artificial en la educación superior. Editorial Grupo AEA.
- Tapia, V. (2020). Base de datos II: Experiencias prácticas. Universidad Católica Los Ángeles de Chimbote.
- Vásquez, A., Guanuchi, L., Cahuana, R., Vera, R., & Holgado, J. (2023). Métodos de investigación científica (1.^a ed.). Instituto Universitario de Innovación Ciencia y Tecnología Inudi Perú. <https://doi.org/10.35622/inudi.b.094>

8 ANEXOS

Anexo A: Asignación de tutor: responsable de orientar y supervisar el trabajo académico.



UNIVERSIDAD LAICA
ELOY ALFARO DE MANABÍ

Universidad Laica Eloy Alfaro de Manabí

**Periodo 2025-2 - Notificación de tutor asignado -
TECNOLOGÍAS DE LA INFORMACIÓN 2022 (EL CARMEN)**

Estimad@
Docente y Estudiante
Uleam

En cumplimiento de lo establecido en la Ley, el Reglamento de Régimen Académico y las disposiciones estatutarias de la Uleam, por medio de la presente se oficializa la dirección y tutoría en el desarrollo del Trabajo de Integración curricular / Trabajo de Titulación del siguiente estudiante:

Tema: SISTEMA INFORMÁTICO PARA LA GESTIÓN DE BÚSQUEDA DE CONTENIDOS AUDIOVISUALES EN SOFTMEDIA ULEAM EL CARMEN

Estado de aprobación: Aprobado

Tipo de titulación: Trabajo de Integración Curricular

Tipo de proyecto: Trabajo de Integración Curricular / Trabajo de titulación se articula con proyectos y programas de Vinculación.

Apellidos y nombres del tutor asignado: ARCA ZAVALA JEFFERSON OMAR

Apellidos y nombres del estudiante: SUMI AREVALO JAZMANY DANIEL

Carrera: TECNOLOGÍAS DE LA INFORMACIÓN 2022 (EL CARMEN)

Periodo de inducción: Periodo 2025-2

Sírvase cumplir con lo dispuesto en el Manual de Procedimientos de TITULACIÓN DE ESTUDIANTES DE GRADO: TRABAJO DE INTEGRACIÓN CURRICULAR Y UNIDAD DE TITULACIÓN <https://departamentos.uleam.edu.ec/gestion-aseguramiento-calidad/files/2020/06/PAT-04-Titulacion-de-Estudiantes-de-grado-UIC-y-UT.pdf>

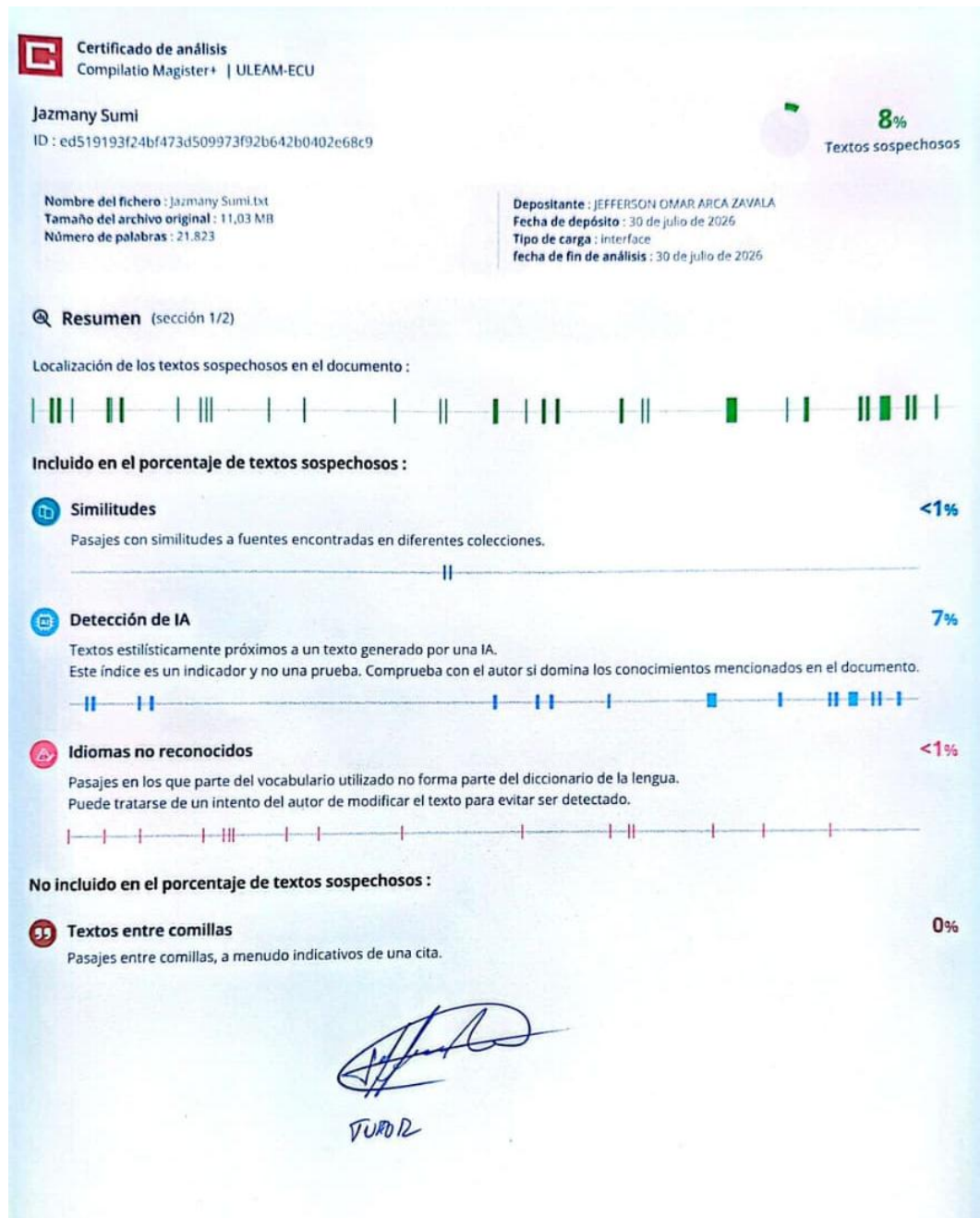
Particular que se informa para los fines consiguientes.

Atentamente,

Comisión Académica y Responsable de Titulación.

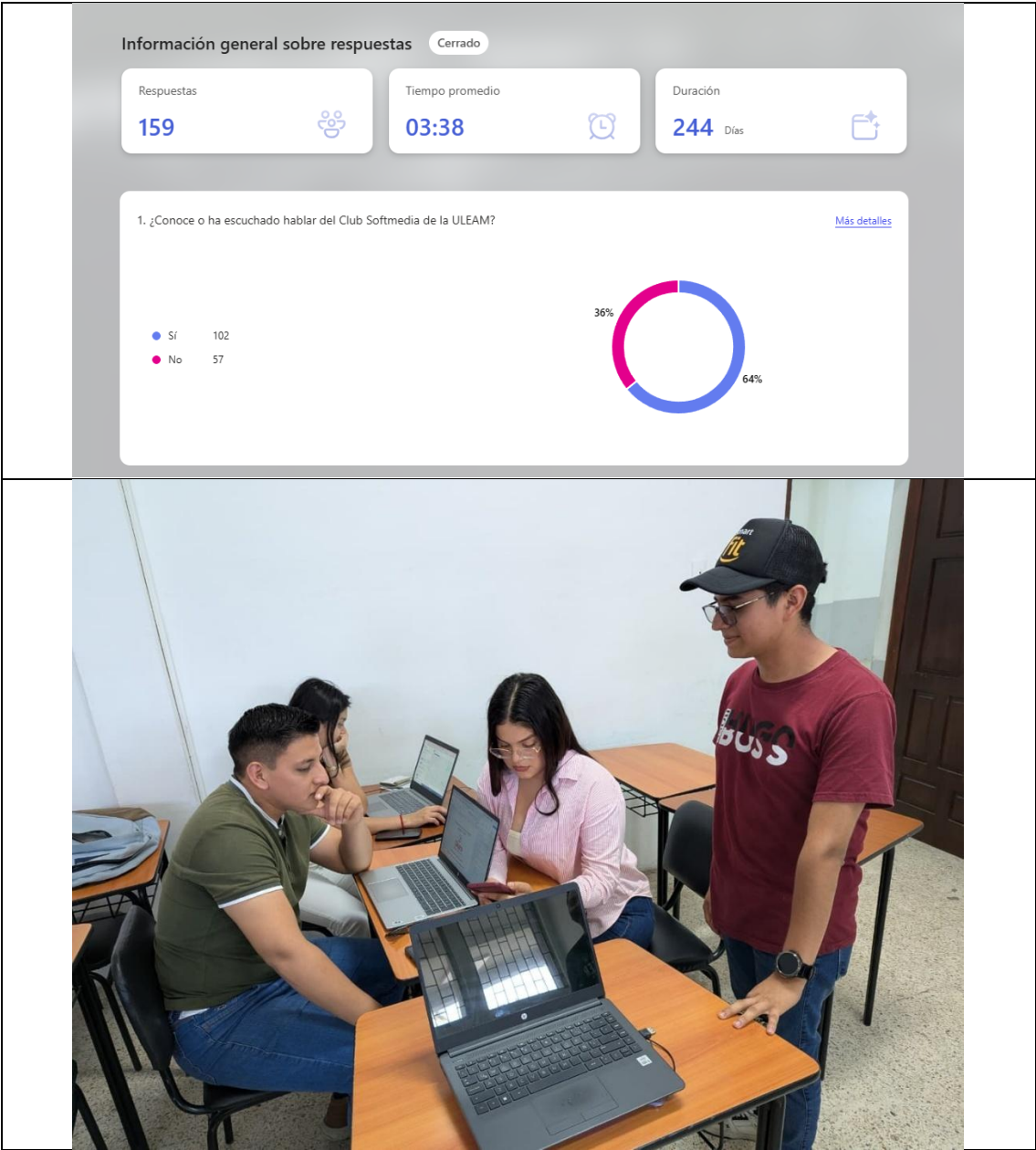
Anexo A: Asignación de tutor:

Anexo B: Reporte del sistema anti plagio.



Anexo B: Reporte del sistema antiplagio

Anexo C: Evidencia de aplicación de encuestas a la comunidad universitaria y área administrativa.





Anexo C: Evidencia aplicación de encuestas

Anexo D: Formato del material de recolección de datos (encuesta).

INSTRUMENTO DE RECOLECCION DE DATOS

Encuesta – Plataforma Web del Club SoftMedia

El Club SoftMedia de la ULEAM extensión El Carmen realiza actividades de producción audiovisual y cobertura de eventos. Actualmente no existe una plataforma web donde se organice y consulte de manera ordenada dicho contenido. Esta encuesta busca conocer la percepción y necesidades de la comunidad universitaria para el desarrollo de una plataforma web con un módulo de catálogo audiovisual y funciones de inteligencia artificial.

Autor: Jazmany Daniel Sumi Arévalo

1. ¿Conoce o ha escuchado hablar del Club SoftMedia de la ULEAM?
 - Si
 - No
2. ¿Ha tenido dificultades para encontrar el contenido multimedia (fotos y videos) de las actividades universitarias cuando lo ha necesitado?
 - Si
 - No
3. ¿Considera necesario que exista una plataforma web donde el Club SoftMedia organice y publique su contenido audiovisual?
 - Si
 - No
4. ¿Cree que el uso de inteligencia artificial para etiquetar y ordenar automáticamente fotos y videos sería útil?
 - Si
 - No
5. ¿Estaría dispuesto/a a usar la plataforma si estuviera disponible para estudiantes y docentes?
 - Si
 - No
6. ¿Considera importante que el catálogo permita búsqueda por palabras clave (p. ej. evento, fecha, persona, tema)?

- Si
 - No
7. ¿Cree que la plataforma debería ser accesible y usable desde dispositivos móviles (smartphones/tablets)?
- Si
 - No
8. ¿Qué funciones considera indispensables para que el catálogo audiovisual de la futura plataforma web sea realmente útil para la comunidad universitaria?
-
-
9. ¿Qué recomendaciones o ideas aportaría para mejorar la gestión y organización del contenido audiovisual del Club SoftMedia?
-
-

Anexo D: Formato del material de recolección de datos

9 GLOSARIO

API (Application Programming Interface): Conjunto de reglas que permite la comunicación entre distintos componentes de software. FastAPI actúa como la API del sistema.

BackgroundTasks: Mecanismo de FastAPI que ejecuta procesos en segundo plano sin interrumpir la respuesta al usuario.

Backend: Parte del sistema que opera en el servidor y gestiona la lógica, los datos y la autenticación.

bcrypt: Algoritmo de cifrado para el almacenamiento seguro de contraseñas de usuario.

BLIP (Bootstrapping Language-Image Pre-Training): Modelo de inteligencia artificial de Salesforce que genera descripciones textuales automáticas a partir de imágenes.

Catálogo audiovisual: Sistema organizado de registro y clasificación de archivos multimedia que facilita su búsqueda y recuperación.

Contenerización: Técnica que empaqueta una aplicación con sus dependencias para que funcione de manera idéntica en cualquier entorno. Se implementó con Docker Compose.

Daily Scrum: Reunión diaria breve del equipo para sincronizar avances e identificar impedimentos durante un Sprint.

Definition of Done: Criterio acordado que establece las condiciones mínimas para considerar una tarea como completamente finalizada.

Docker Compose: Herramienta que permite orquestar y ejecutar múltiples contenedores de una aplicación mediante un archivo de configuración.

Etiquetado automático: Proceso en el que la inteligencia artificial analiza un archivo multimedia y le asigna palabras clave descriptivas sin intervención humana.

FastAPI: Framework de Python para construir APIs de alto rendimiento. Se usó como motor del backend del sistema.

FFmpeg: Herramienta de código abierto para procesar archivos multimedia. Se utilizó para extraer fotogramas de videos antes del análisis con IA.

File System Access API: Interfaz de navegador que permite guardar archivos directamente en una carpeta del dispositivo del usuario.

Fire and Forget: Patrón de programación en el que una tarea se ejecuta en segundo plano sin que el sistema espere su resultado para continuar.

Frontend: Capa visual del sistema con la que interactúa el usuario. Se desarrolló con React 18, Vite y Tailwind CSS.

Framer Motion: Biblioteca de animaciones para React usada para implementar transiciones fluidas en la interfaz.

Gestión de contenidos: Procesos de creación, organización, publicación y mantenimiento de información digital dentro de un sistema.

Google Translate: Servicio de traducción automática utilizado para convertir las descripciones generadas por BLIP del inglés al español.

Inteligencia artificial (IA): Tecnología que permite a los sistemas realizar tareas que normalmente requieren inteligencia humana, como el reconocimiento y descripción de imágenes.

JSONB: Tipo de campo en PostgreSQL para almacenar datos JSON de forma binaria, permitiendo consultas rápidas sobre estructuras flexibles. Se usó para guardar las etiquetas generadas por la IA.

JWT (JSON Web Token): Estándar para transmitir información de autenticación de forma segura entre cliente y servidor mediante tokens firmados.

Lightbox: Componente de interfaz que muestra imágenes o videos en una capa superpuesta con navegación por teclado, sin abandonar la página.

MARC21: Estándar internacional para la representación de información bibliográfica en formato legible por máquina, analizado como referente teórico.

Masonry: Disposición de galería en cuadrícula de columnas múltiples donde los elementos se organizan sin espacios vacíos, adaptándose a su tamaño.

Metadatos: Datos que describen a otros datos, como la fecha, el autor, el evento y las etiquetas de un archivo multimedia.

Middleware: Componente que intercepta las solicitudes del sistema para verificar permisos, autenticación y aplicar políticas de seguridad.

Módulo: Componente independiente de un sistema que agrupa funcionalidades relacionadas y puede desarrollarse por separado.

OWASP: Estándar de seguridad web cuyas cabeceras HTTP fueron implementadas en el sistema para protegerlo contra vulnerabilidades comunes.

Plataforma web: Sistema de software accesible desde un navegador que centraliza funcionalidades y servicios para múltiples usuarios.

PostgreSQL: Sistema de gestión de bases de datos relacional de código abierto usado como motor de base de datos del proyecto.

Product Backlog: Lista ordenada de todos los requisitos y funcionalidades que debe incorporar el producto, gestionada por el Product Owner.

Product Owner: Rol de Scrum responsable de definir y priorizar los requisitos del sistema. En este proyecto lo ejerció el docente tutor.

React 18: Biblioteca de JavaScript para construir interfaces de usuario interactivas mediante componentes reutilizables.

Red neuronal: Modelo computacional que aprende a reconocer patrones a partir de datos. Es la base de modelos de IA como BLIP.

Regla de Oro: Política del sistema que impide la eliminación accidental de contenido publicado, requiriendo confirmación explícita del rol autorizado.

Repositorio audiovisual: Espacio centralizado de almacenamiento y organización de archivos multimedia con mecanismos de búsqueda y control de acceso.

Responsivo: Diseño web que adapta automáticamente la interfaz al tamaño de la pantalla del dispositivo del usuario.

Rol: Perfil de usuario que define qué acciones puede realizar dentro del sistema. La plataforma define cuatro roles: Administrador, Supervisor, Integrante e Invitado.

Scrum: Marco de trabajo ágil basado en ciclos iterativos llamados Sprints, orientado a la entrega incremental de productos funcionales.

Sprint: Ciclo de trabajo de duración fija en Scrum al término del cual se entrega un incremento funcional. El proyecto se organizó en cuatro Sprints de dos semanas.

Sprint Backlog: Conjunto de tareas seleccionadas del Product Backlog que el equipo se compromete a completar en un Sprint.

Sprint Planning: Reunión inicial de cada Sprint donde el equipo selecciona las tareas a desarrollar y define su objetivo.

Sprint Retrospective: Reunión al cierre de cada Sprint donde el equipo evalúa su proceso de trabajo y propone mejoras.

Sprint Review: Reunión al final de cada Sprint donde se presenta el incremento desarrollado a los interesados para obtener retroalimentación.

Tailwind CSS: Framework de CSS que permite construir interfaces visuales mediante clases utilitarias directamente en el código HTML.

Tokenización: Proceso de dividir un texto en unidades menores llamadas tokens para extraer palabras clave. Se usó para obtener las etiquetas finales de las descripciones generadas por BLIP.

Vite: Herramienta de desarrollo web que proporciona un servidor rápido y optimizado para aplicaciones modernas con React.