

Plataforma para la difusión de información sobre arte, cultura y tradición en UNIMINUTO Villavicencio.



Plataforma para la difusión de información sobre arte, cultura y tradición en UNIMINUTO

Villavicencio.

Nicolas Alfonso Manrique Martínez

Santiago Andrés Suescun Beltrán

Corporación Universitaria Minuto de Dios

Rectoría Oriente

Centro Universitario Villavicencio (Meta)

Programa Tecnología en Desarrollo de Software

mayo de 2025

Plataforma para la difusión de información sobre arte, cultura y tradición en UNIMINUTO Villavicencio.

Plataforma para la difusión de información sobre arte, cultura y tradición en  
UNIMINUTO Villavicencio.

Nicolas Alfonso Manrique Martínez

Santiago Andrés Suescun Beltrán

Trabajo de Grado presentado como requisito para optar al título de Tecnólogo en Desarrollo de  
Software

Asesor 1

Shirley Paola Giraldo Gaitán

Ingeniera de Sistemas

Asesor 2

Wilmer Hernández Álvarez

Ingeniero de Sistemas

Corporación Universitaria Minuto de Dios

Rectoría Oriente

Centro Universitario Villavicencio (Meta)

Programa Tecnología en Desarrollo de Software

mayo de 2025

## Tabla de contenido

Lista de tablas .....	6
Lista de figuras .....	7
Resumen.....	9
Abstract .....	10
Introducción .....	11
CAPÍTULO I .....	12
1.1    Objetivo general.....	12
1.1.1    Objetivos específicos.....	12
1.2    Planteamiento del problema .....	13
1.2.1    Formulación del problema .....	14
1.3    Justificación .....	14
2    CAPITULO II.....	16
2.1    Marco teórico .....	16
2.1.1    Patrimonio cultural y su preservación digital.....	16
2.1.2    Proyección social universitaria y difusión cultural .....	17
2.1.3    Sistemas de Gestión de Contenidos (CMS) .....	17
2.1.4    Tecnologías web para la difusión cultural .....	18
2.1.5    Diseño Centrado en el Usuario para Contenidos Culturales.....	18
2.2    Antecedentes Teóricos .....	19
2.2.1    Evolución de las plataformas digitales para la difusión cultural .....	19
2.2.1.1    Cultura UNAM.....	19

Plataforma para la difusión de información sobre arte, cultura y tradición en UNIMINUTO Villavicencio. 4

2.3	Marco Conceptual.....	23
2.4	Marco Legal .....	25
3	CAPITULO III .....	27
3.1	Tipo de investigación .....	27
3.2	Población .....	28
3.3	Muestra.....	28
4	CAPITULO IV .....	30
4.1	Metodología de desarrollo de software .....	30
4.1.1	Fase de análisis.....	32
4.1.2	Repositorio.....	39
4.2	Análisis de requerimientos .....	40
4.2.1	Requerimientos funcionales del sitio web .....	40
4.2.2	Requerimientos no funcionales del sitio web .....	40
4.2.3	Requerimientos funcionales del CMS.....	41
4.2.4	Requerimientos no funcionales del CMS .....	41
4.3	Historias de usuario .....	41
4.4	Diseño de la aplicación .....	45
4.4.1	Casos de uso.....	50
4.4.2	Diagrama de secuencia .....	52
4.4.3	Mockups .....	57
4.4.4	Diagrama de clases .....	62
4.5	Desarrollo del aplicativo .....	63
4.5.1	Prisma como ORM .....	64

4.5.1	CockroachDB.....	65
4.5.2	Auth.js.....	65
4.5.1	Tailwind CSS .....	67
4.6	Diccionario de datos .....	68
4.6.1	Relaciones.....	79
4.7	Plan de pruebas.....	80
5	CAPITULO VI .....	84
5.1	Conclusiones .....	84
5.2	Resumen Analítico Especializado – RAE .....	86
6	Referencias.....	89
7	Anexos.....	92
7.1	Manual de Usuario.....	92
7.1.1	Iniciar Sesión en el CMS.....	92
7.1.2	Navegación en el Panel de Control .....	93
7.1.3	Crear un Cartel Publicitario.....	95
7.1.4	Crear un Evento.....	96

## Lista de tablas

Tabla 1 Cuadro comparativo de Scrum.....	31
Tabla 2. Cuadro Comparativo de Trello.....	36
Tabla 3. Historia de usuario Loguin .....	41
Tabla 4. Historia de usuario Gestión de Eventos.....	42
Tabla 5. Historia de usuario Gestión de contenido .....	43
Tabla 6. Historia de usuario Visualización de Testimonios .....	43
Tabla 7. Historia de usuario Visualización de Eventos y Contenido.....	44
Tabla 8. Modelo User.....	69
Tabla 9. Modelo Account .....	70
Tabla 10. Modelo Billboard .....	71
Tabla 11. Modelo AboutUs.....	72
Tabla 12. Modelo Gallery .....	73
Tabla 13. Modelo News.....	74
Tabla 14. Modelo Article.....	74
Tabla 15. Modelo SpecialEvent .....	75
Tabla 16. Modelo SpecialEventComment.....	76
Tabla 17 Modelo Event .....	77
Tabla 18. Modelo EventComment .....	78
Tabla 19. Modelo Testimonials .....	78
Tabla 20. Plan de pruebas .....	81

## Lista de figuras

Figura 1. Cultura UNAM.....	20
Figura 2 MNAC .....	21
Figura 3 Sounds of India.....	22
Figura 4. Tablero de Trello .....	32
Figura 5. Repositorio de github .....	39
Figura 6. Interfaz del Loguin.....	45
Figura 7. Interfaz de Gestión de Eventos .....	46
Figura 8. Interfaz de Gestión de contenido.....	47
Figura 9. Interfaz de Testimoniales .....	48
Figura 10. Interfaz Visualización de Eventos y contenido.....	49
Figura 11. Interfaz Visualización de Eventos Futuros o Actuales .....	50
Figura 12. Caso de uso de sistema CMS .....	50
Figura 13. Caso de uso página web .....	52
Figura 14. Diagrama de secuencia del CMS.....	54
Figura 15 Diagrama de secuencia de la Página web .....	56
Figura 16. Mockup del Login .....	57
Figura 17 Mockup de Gestión de Eventos.....	58
Figura 18. Mockup de Gestión de contenido .....	59
Figura 19. Mockup de Testimoniales .....	60
Figura 20. Mockups de Visualización de Eventos y contenido .....	61
Figura 21. Mockups Visualización de Eventos Futuros o Actuales .....	62

Figura 22. Diagrama de clases.....	62
Figura 23. Estructura de la Base de Datos con Prisma ORM.....	65
Figura 24. Configuración de Proveedor de Credenciales en Auth.js.....	66
Figura 25. Configuración personalizada de Tailwind CSS.....	68
Figura 26. inicio de sesión.....	92
Figura 27. Panel de control.....	94
Figura 28 .Cartel publicitario.....	96
Figura 29. Creación de eventos.....	98

## Resumen

El proyecto, impulsado por la Dirección de Proyección Social de UNIMINUTO Villavicencio, responde a la necesidad de centralizar y difundir recursos culturales de la región Oriente, como eventos, podcasts, videos e imágenes sobre arte, cultura y tradiciones, que actualmente están dispersos y tienen acceso limitado. Esta fragmentación restringe la apropiación del conocimiento cultural por parte de la comunidad universitaria y el público general, justificando la creación de una plataforma digital unificada para fortalecer la proyección social y preservar el patrimonio de la Oriente.

El objetivo principal es desarrollar una plataforma web con un Sistema de Gestión de Contenidos (CMS) para gestionar y promover estos recursos digitales. La investigación se diseñó como un estudio de caso, centrado en el análisis profundo del Centro de Arte, Cultura y Tradición de la Orinoquía (CACTO), una unidad específica de UNIMINUTO Rectoría Oriente. Este enfoque resulta idóneo, ya que permite explorar detalladamente las dinámicas, objetivos y necesidades del CACTO en su contexto particular, generando conocimiento aplicado a partir de una experiencia concreta. Se adoptó un enfoque cualitativo, priorizando la comprensión de significados, percepciones y experiencias relacionadas con la gestión y difusión del arte, la cultura y las tradiciones. La plataforma busca mejorar el acceso al contenido cultural, aumentar la visibilidad de UNIMINUTO y fomentar la interacción cultural regional, con potencial para incorporar mejoras como soporte multilingüe o analíticas avanzadas, consolidando un aporte significativo al ámbito educativo y cultural de Oriente.

**Palabras clave:** Gestión de contenidos digitales, difusión cultural, desarrollo web, proyección social, metodología ágil

### **Abstract**

The project, driven by the Social Outreach Department of UNIMINUTO Villavicencio, addresses the need to centralize and disseminate cultural resources from the Oriente region, such as events, podcasts, videos, and images related to art, culture, and traditions, which are currently scattered and have limited access. This fragmentation restricts the appropriation of cultural knowledge by the university community and the public, justifying the creation of a unified digital platform to strengthen social outreach and preserve Oriente's heritage.

The main objective is to develop a web platform with a Content Management System (CMS) to manage and promote these digital resources. The research was designed as a case study, focusing on an in-depth analysis of the Orinoquía Center for Art, Culture, and Tradition (CACTO), a specific unit of UNIMINUTO Oriente Rectory. This approach is ideal, as it allows for a detailed exploration of CACTO's dynamics, objectives, and needs within its specific context, generating applied knowledge from a concrete experience. A qualitative approach was adopted, prioritizing the understanding of meanings, perceptions, and experiences related to the management and dissemination of art, culture, and traditions. The platform aims to improve access to cultural content, increase UNIMINUTO's visibility, and foster regional cultural interaction, with potential for enhancements such as multilingual support or advanced analytics, consolidating a significant contribution to the educational and cultural landscape of Oriente.

**Keywords:** Digital content management, cultural dissemination, web development, social outreach, agile methodology.

## **Introducción**

Este proyecto se basó en el desarrollo de una plataforma web que integra un Sistema de Gestión de Contenidos (CMS) diseñado específicamente para gestionar y difundir información sobre arte, cultura y tradiciones de la región de Oriente, generada por la Corporación Universitaria Minuto de Dios (UNIMINUTO) sede Villavicencio, como parte de su labor de proyección social. Esta solución tecnológica organiza recursos digitales como podcasts, videos, imágenes y textos, empleando tecnologías como HTML5, CSS3, JavaScript, React y bases de datos SQL. Además, se capacitará a un único administrador designado por UNIMINUTO para garantizar un control eficiente y riguroso del contenido, asegurando su calidad y coherencia.

Más allá de ser una herramienta funcional, el proyecto busca promover la apropiación cultural y académica de los conocimientos generados, por la comunidad universitaria. La decisión de desarrollar un CMS a medida responde a la necesidad de un sistema especializado que se adapte a las particularidades del contenido cultural regional, ofreciendo una categorización clara y un acceso intuitivo para los usuarios.

## **CAPÍTULO I**

### **1.1 Objetivo general**

Desarrollar un sitio web y un Sistema de Gestión de Contenidos (CMS) para la difusión de recursos digitales sobre arte, cultura y tradición, dirigido a la comunidad universitaria y al público en general.

#### **1.1.1 Objetivos específicos**

Identificar los requerimientos del software mediante historias de usuario, mesas de trabajo y entrevistas con el cliente.

Diseñar una interfaz intuitiva y accesible para el CMS y la página web utilizando Figma, asegurando que cumpla con los estándares de usabilidad y accesibilidad requeridos.

Implementar la arquitectura backend con bases de datos SQL y Prisma ORM para una gestión eficiente y segura del contenido.

Aplicar metodologías y marcos de trabajo ágiles en el proceso de desarrollo del software, fomentando la planificación iterativa, la colaboración continua y la mejora constante.

Realizar pruebas funcionales y de seguridad en los módulos del CMS y la página web para garantizar su correcto funcionamiento, accesibilidad y protección de datos.

## **1.2 Planteamiento del problema**

La Dirección de Proyección Social de la Corporación Universitaria Minuto de Dios (UNIMINUTO) sede Villavicencio ha impulsado diversas estrategias como publicaciones en redes sociales para difundir información sobre arte, cultura y tradiciones del Oriente, como parte de su compromiso con el fortalecimiento de la identidad cultural y el desarrollo comunitario. Actualmente, se utilizan canales como bibliotecas digitales y físicas para almacenar investigaciones académicas, documentos y recursos relacionados con estas áreas. La interacción entre docencia e investigación genera constantemente nuevo conocimiento que se comparte con la comunidad universitaria y el público general mediante eventos, festivales culturales y publicaciones en sitios web y redes sociales. Sin embargo, estos esfuerzos carecen de una centralización efectiva, lo que limita su alcance y accesibilidad.

La distribución de la información cultural se realiza de manera dispersa a través de múltiples plataformas digitales y físicas, dificultando la integración de los recursos y su apropiación por parte de la comunidad. Aunque existen alianzas con entidades culturales locales y el respaldo del Ministerio de Cultura, persiste la necesidad de una plataforma digital robusta que consolide el contenido generado por UNIMINUTO. Esta carencia impide que el valioso conocimiento sobre el patrimonio del Oriente como tradiciones llaneras, expresiones artísticas y estudios académicos llegue de forma eficiente a la sociedad. El objetivo es lograr una difusión más accesible y estructurada, promoviendo una mayor interacción y comprensión cultural entre la comunidad universitaria.

### **1.2.1 Formulación del problema**

¿Cómo desarrollar una plataforma web basada en un CMS que permita a UNIMINUTO difundir y facilitar el acceso al conocimiento sobre arte, cultura y tradición, tanto para la comunidad universitaria como para el público en general?

### **1.3 Justificación**

La importancia de esta iniciativa radica en su capacidad para democratizar el acceso al conocimiento y promover la difusión de la cultura y las tradiciones del Oriente, en el área de influencia de la Corporación Universitaria Minuto de Dios (UNIMINUTO) sede Villavicencio. Este proyecto, liderado por el Centro de Arte, Cultura y Tradición de la Dirección de Proyección Social, responde a la necesidad de centralizar recursos culturales en un entorno digital, ante la disminución de visitas a espacios físicos en la actualidad. Al aprovechar tecnologías modernas, se busca garantizar que la producción académica y cultural de UNIMINUTO llegue de manera efectiva a la comunidad universitaria y al público general interesado en el patrimonio regional.

El desarrollo de esta plataforma web incrementa la accesibilidad a contenidos generados por el Centro de Arte, Cultura y Tradición, como investigaciones, eventos, podcasts, videos e imágenes, fortaleciendo la proyección social de la institución. Al consolidar estos recursos en un sistema eficiente e intuitivo, se fomenta la apropiación del conocimiento y el enriquecimiento cultural en la sociedad del Oriente. Este proyecto aborda directamente la dispersión de la información cultural identificada por la Dirección de Proyección Social, ofreciendo una solución tecnológica que agiliza el acceso y visibiliza el trabajo de UNIMINUTO. Así, la institución se

Plataforma para la difusión de información sobre arte, cultura y tradición en UNIMINUTO 15  
Villavicencio.

posiciona como referente en la gestión digital del patrimonio cultural, asegurando un impacto significativo en su comunidad objetivo.

## 2 CAPITULO II

### 2.1 Marco teórico

El desarrollo de una plataforma web para difundir información sobre arte, cultura y tradición en UNIMINUTO Villavicencio requiere un sustento teórico robusto que aborde el patrimonio cultural, la proyección social, las tecnologías digitales y el diseño centrado en el usuario. Este capítulo explora estos fundamentos, complementados con antecedentes extensos de proyectos similares y un marco conceptual detallado, para justificar la relevancia y viabilidad de la iniciativa.

#### 2.1.1 Patrimonio cultural y su preservación digital

El patrimonio cultural engloba las expresiones materiales e inmateriales que definen la identidad de una comunidad, incluyendo tradiciones, arte y costumbres (CABRERA PEÑA, 2024). En la región del Oriente colombiano, este patrimonio se manifiesta en prácticas como el joropo, la artesanía llanera y las festividades locales, las cuales enfrentan riesgos de desaparición debido a la modernización y la falta de registro sistemático (MINISTERIO, 2023). La digitalización emerge como una solución clave para preservar y difundir estos elementos, permitiendo su acceso global y su integración en la educación y la proyección social.

Un ejemplo emblemático es "Memoria Chilena", gestionada por la Biblioteca Nacional de Chile. Desde su lanzamiento en 2001, esta plataforma ha digitalizado más de 800,000 documentos, incluyendo textos, imágenes y audios, con un tráfico anual de 2 millones de visitas

(chile, s.f.). Este caso demuestra cómo las plataformas digitales pueden revitalizar el patrimonio cultural, un objetivo alineado con la propuesta para UNIMINUTO.

### **2.1.2 Proyección social universitaria y difusión cultural**

La proyección social en las universidades busca conectar el conocimiento académico con las demandas comunitarias, fomentando el desarrollo social y cultural (Pico Macías, 2023). Para UNIMINUTO Villavicencio, esta misión se traduce en iniciativas que promueven el patrimonio de la Orinoquia, como eventos culturales y producción académica, siguiendo su modelo educativo orientado a la transformación social (UNIMINUTO, 2021).

Un caso relevante es el programa "Cultura Viva" de la Universidad Nacional de Colombia. Desde 2015, esta iniciativa ha digitalizado contenidos culturales y organizado actividades que alcanzan a más de 50,000 personas al año, fortaleciendo la identidad regional (Viva, 2023). Este modelo destaca el potencial de las universidades como agentes de difusión cultural, un enfoque que la plataforma de UNIMINUTO busca replicar y escalar mediante tecnología digital.

### **2.1.3 Sistemas de Gestión de Contenidos (CMS)**

Los CMS son herramientas tecnológicas que facilitan la creación, gestión y publicación de contenidos digitales sin exigir habilidades técnicas avanzadas (Serrano-Cobos, 2021). En contextos culturales, ofrecen funciones como categorización temática, gestión de metadatos,

soporte multimedia y flujos de curaduría, esenciales para organizar y difundir recursos de manera efectiva (Ann Rockley, 2020).

El Museo del Prado en España ilustra este concepto. Su sitio web, basado en un CMS personalizado desde 2010, gestiona más de 30,000 obras digitalizadas y recibe 3 millones de visitas anuales (Domínguez, 2023). Este ejemplo evidencia cómo un CMS puede centralizar contenidos culturales, un principio aplicable al sistema propuesto para UNIMINUTO, que priorizará la administración intuitiva de eventos, podcasts y galerías.

#### **2.1.4 Tecnologías web para la difusión cultural**

Las tecnologías web modernas, como Single Page Applications (SPA) y Progressive Web Apps (PWA), optimizan la experiencia del usuario mediante interfaces reactivas (React.js), renderizado del lado del servidor (Next.js) y diseños adaptativos (CSS3, Bootstrap). APIs como Prisma y GraphQL facilitan la gestión eficiente de datos (Osmani, 2022)

Un caso destacado es "Europeana", el repositorio cultural europeo. Desde 2008, esta plataforma utiliza tecnologías como React y APIs para ofrecer acceso a más de 50 millones de ítems culturales, soportando millones de usuarios simultáneos (Foundation, 2021). Este éxito respalda la elección de herramientas similares para la plataforma de UNIMINUTO, asegurando escalabilidad y accesibilidad.

#### **2.1.5 Diseño Centrado en el Usuario para Contenidos Culturales**

El diseño centrado en el usuario (DCU) prioriza las necesidades de los usuarios mediante investigación etnográfica, prototipos iterativos y pruebas de usabilidad, garantizando plataformas accesibles y atractivas (Norman, 2022). En contextos culturales, este enfoque equilibra rigor informativo y facilidad de uso (Janne Berg jaberg, 2021)

El portal "Smithsonian Open Access" (2020) es un referente. Con más de 4 millones de objetos digitalizados y una interfaz intuitiva, el 87% de sus usuarios reporta una experiencia satisfactoria (Institution, 2023). Este caso subraya la importancia del DCU en la propuesta de UNIMINUTO, que busca satisfacer a estudiantes, docentes y el público general.

## **2.2 Antecedentes Teóricos**

### **2.2.1 Evolución de las plataformas digitales para la difusión cultural**

La digitalización cultural ha evolucionado desde iniciativas básicas hasta plataformas interactivas. El "Proyecto Gutenberg" (1971), pionero en libros electrónicos, marcó un hito inicial con más de 70,000 títulos disponibles en 2024 (Gutenberg., 2024).

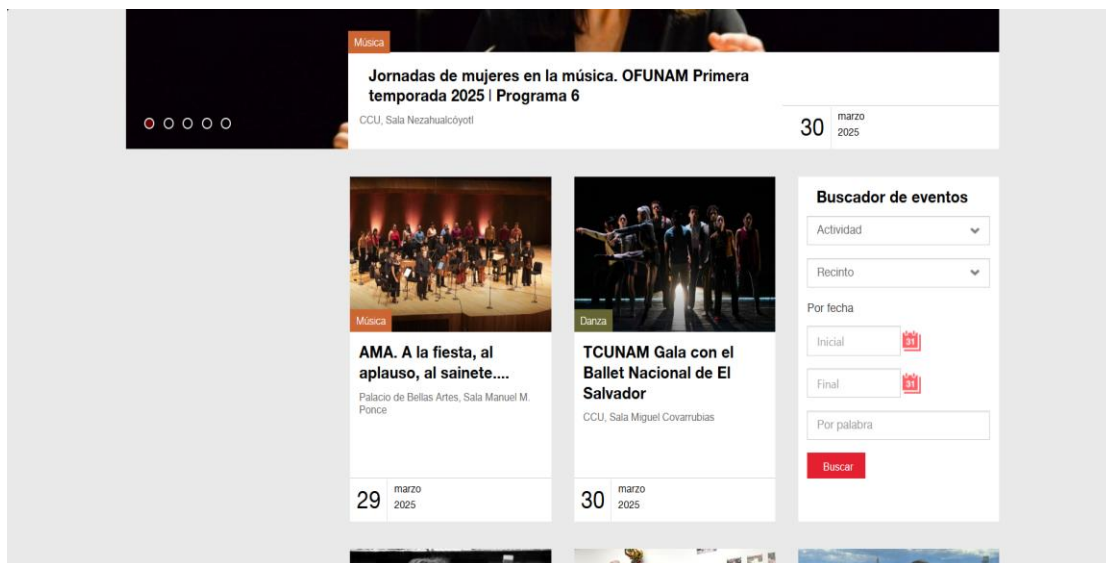
Posteriormente, Google Arts & Culture (2011) integró tecnologías avanzadas como realidad virtual, alcanzando más de 40 millones de usuarios mensuales (Google, 2023). Estos proyectos reflejan la tendencia hacia la multimedialidad y accesibilidad que inspira la plataforma de UNIMINUTO

#### **2.2.1.1 Cultura UNAM**

La Universidad Nacional Autónoma de México lanzó en 2015 una plataforma para difundir eventos y contenidos culturales. Con más de 1,200 recursos digitalizados (obras de arte, conciertos, conferencias y archivos históricos) y recibe más de 600,000 visitas anuales, según su informe de 2023. Su éxito en la promoción de actividades culturales como el festival “Cultura UNAM en Línea” demuestra el potencial de las plataformas universitarias para ampliar el acceso a la cultura (UNAM, 2025). Este modelo es replicable en UNIMINUTO, adaptándolo a su contexto educativo.

La Figura 1 muestra un ejemplo de la plataforma 'Cultura UNAM', destacando su diseño para la difusión de eventos culturales, como conciertos, balé y obras teatrales. La interfaz incluye un buscador avanzado por fecha, recinto o palabra clave, facilitando la accesibilidad para el usuario.

**Figura 1. Cultura UNAM**



**Fuente: UNAM (2025)**

### 2.2.1.2 Píldoras Románicas (Museu Nacional d'Art de Catalunya)

Desde 2011, este proyecto del Museu Nacional d'Art de Catalunya (MNAC) utilizó Facebook para difundir cápsulas culturales sobre arte románico, logrando una interacción colaborativa con más de 15,000 usuarios en su primer año (MNAC, 2023). Aunque basado en redes sociales, el caso evidencia cómo estrategias digitales breves y visuales pueden democratizar el acceso al patrimonio. Este principio es aplicable a la plataforma de UNIMINUTO, adaptándolo a un formato web con mayor profundidad académica.

Como se observa en la Figura 2, el MNAC utiliza su plataforma digital para integrar exposiciones temporales (ej. 'Zurbarán'), actividades educativas y venta de entradas, demostrando un modelo de difusión cultural integral

Figura 2 MNAC



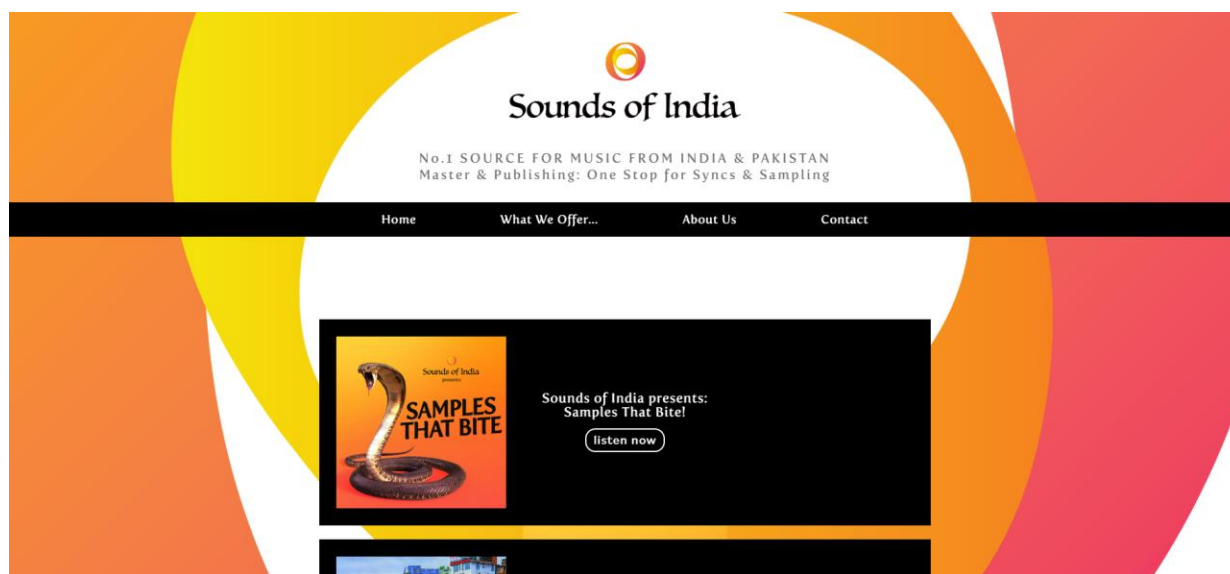
Fuente: MNAC (2023)

### 2.2.1.3 Sounds of India

Lanzado en 2018, este proyecto digital se ha convertido en un archivo de referencia para la música tradicional india, con más de 1,000 horas de grabaciones de géneros como Carnatic e Hindustani, y 300,000 descargas registradas hasta 2023 (India, 2024). Su plataforma web, facilita su uso tanto para académicos como para el público general. Este modelo que prioriza la preservación sin sacrificar la usabilidad es replicable en UNIMINUTO para difundir expresiones culturales colombianas como el *joropo*, la *poesía llanera* o las *músicas andinas*, adaptándolo a contextos locales.

El caso de 'Sounds of India' (2023) demuestra que plataformas minimalistas pero rigurosas pueden lograr alto impacto en preservación cultural, especialmente cuando integran metadata detallada (ej.: intérpretes, contextos históricos) junto a herramientas de búsqueda intuitiva.

**Figura 3 Sounds of India**



**Fuente: Sounds of India (2024)**

Estos antecedentes confirman que las plataformas digitales son efectivas para centralizar y promover contenidos culturales, ofreciendo lecciones sobre usabilidad, interacción y escalabilidad para el proyecto actual.

### 2.3 Marco Conceptual

**Sistema de Gestión de Contenidos (CMS):** Un CMS es una herramienta tecnológica que permite crear, gestionar y publicar contenidos digitales de forma sencilla, sin necesidad de conocimientos técnicos avanzados. Esto facilita la organización y difusión de información en entornos digitales (Serrano-Cobos, 2021). En proyectos como el Centro Digital de Arte, Cultura y Tradición de UNIMINUTO, el CMS debe incluir medidas de seguridad y privacidad para proteger datos sensibles y garantizar un acceso seguro (García-Holgado et al., 2022).

**Difusión del Conocimiento:** La difusión del conocimiento en plataformas digitales implica compartir información académica y cultural de manera sistemática, utilizando herramientas como CMS para preservar y promover el patrimonio cultural (Serrano-Cobos, 2021). En el contexto de UNIMINUTO, este proceso es esencial para hacer accesible el arte y las tradiciones de Villavicencio a un público amplio (Ministerio de Cultura de Colombia, 2022).

**Frontend:** Es la parte de una aplicación o sitio web con la que interactúa el usuario. Incluye elementos visuales y de usuario, como botones y formularios (Richards, 2017).

**Cliente:** Es un programa o aplicación informática que se ejecuta en un dispositivo (como una computadora o un smartphone) y que solicita servicios o recursos a un servidor (Richards, 2017).

**Servidor:** Ordenador empleado por muchos usuarios para realizar una tarea específica, como ejecutar aplicaciones de red o de Internet (Stair, 2024).

**Backend:** Es la parte de una aplicación o sitio web que se encuentra en el servidor y se encarga del procesamiento de datos, la lógica de negocio y la gestión de la base de datos (Richards, 2017).

**Arquitectura MVC:** Es un patrón de arquitectura de software que se utiliza para desarrollar interfaces de usuario dividiendo una aplicación en tres componentes interconectados: modelo, vista y controlador. Su separación permite a los desarrolladores escribir aplicaciones más limpias, mantenibles y escalables (Freeman, 2023).

**Base de Datos Relacional:** Es una forma sencilla pero muy útil de organizar datos en colecciones de tablas bidimensionales denominadas relaciones. Cada fila de la tabla representa una entidad y cada columna representa un atributo de esa entidad. Cada fila de una tabla se identifica unívocamente mediante una clave primaria y el tipo de datos que puede contener una columna de la tabla puede especificarse como número entero, número decimal, fecha, texto, entre otros (Stair, 2024).

**Sistema de gestión de bases de datos (DBMS):** Conjunto de programas utilizados para acceder a una base de datos y gestionarla, así como para proporcionar una interfaz entre la base de datos y sus usuarios y otros programas de aplicación (Stair, 2024).

**Consulta:** Desde la perspectiva de una base de datos, una consulta es una petición específica hecha al DBMS para la manipulación de datos, por ejemplo, leer o actualizar los datos (Stair, 2024).

**SQL (Structured Query Language):** Es un lenguaje de programación específico para acceder a los datos almacenados en una base de datos relacional y manipularlos (Stair, 2024).

## 2.4 Marco Legal

**Ley de Habeas Data de Colombia. Ley 1581 de 2012. Diario Oficial de la República de Colombia, Número 48.425, 17 de octubre de 2012:** “La presente ley tiene por objeto desarrollar el derecho constitucional que tienen todas las personas a conocer, actualizar y rectificar las informaciones que se hayan recogido sobre ellas en bancos de datos , y los demás derechos, libertades y garantías constitucionales relacionadas con la recolección, tratamiento y circulación de datos personales a que se refiere el artículo 15 de la Constitución Política, así como el derecho a la información establecido en el artículo 20 de la Constitución Política, particularmente en relación con la información financiera y crediticia, comercial, de servicios y la proveniente de terceros países.” (Colombia, 2012)

**Ley Estatutaria 1581 del 17 de octubre del 2012. Ley para la protección de datos personales (Art. 1º, Objeto):** “La presente ley tiene por objeto desarrollar el derecho constitucional que tienen todas las personas a conocer, actualizar y rectificar las informaciones que se hayan recogido sobre ellas en bases de datos o archivos, y los demás derechos, libertades y garantías constitucionales a que se refiere el artículo 15 de la Constitución Política; así como el derecho a la información consagrado en el artículo 20 de la misma.” (Colombia, 2012)

### 3 CAPITULO III

#### 3.1 Tipo de investigación

Este proyecto se fundamentó en las líneas de investigación establecidas por la Corporación Universitaria Minuto de Dios (UNIMINUTO), específicamente en la línea de Innovaciones Sociales y Productivas, que promueve el desarrollo de soluciones con impacto social y cultural. Se apoyo en el semillero de investigación DataScience, un espacio dedicado a la formación en análisis de datos y tecnologías emergentes, perteneciente al grupo de investigación GITSAI, reconocido por su trabajo en innovación tecnológica y social.

El proyecto adopta un tipo de investigación de estudio de caso, ya que se centra en analizar en profundidad el Centro de Arte, Cultura y Tradición de la Orinoquía (CACTO) como una unidad específica dentro de UNIMINUTO Rectoría Oriente. Este tipo de investigación es pertinente porque permite explorar de manera detallada las dinámicas, objetivos y necesidades de este centro cultural en su contexto particular, generando conocimiento aplicado a partir de una experiencia concreta. Se optó por un enfoque cualitativo, priorizando la comprensión de significados, percepciones y experiencias relacionadas con la gestión y divulgación del arte, la cultura y la tradición en la región.

Para la recolección de datos, se realizaron cinco entrevistas no estructuradas a María Isabel Rodríguez Leyva, quien fue la directora del proyecto en su momento. Estas entrevistas proporcionaron información sobre los objetivos, la estructura y las estrategias del CACTO, incluyendo el documento maestro que sirve como base del presente estudio. La naturaleza no estructurada de las entrevistas permitió flexibilidad para profundizar en los aspectos más

relevantes según las respuestas de la entrevistada, ofreciendo una visión integral del proyecto desde su liderazgo.

Este proyecto se define como investigación aplicada, ya que no busca generar nuevo conocimiento teórico, sino utilizar y divulgar la información ya existente sobre el CACTO, contenida en el documento maestro suministrado. El propósito es desarrollar una solución práctica, como un sistema de gestión de contenidos (CMS), que facilite la difusión de los contenidos relacionados con el arte, la cultura y la tradición de la Orinoquía, alineándose con los objetivos de proyección social de UNIMINUTO.

### **3.2 Población**

La población de este estudio abarca a la comunidad vinculada a UNIMINUTO Rectoría Oriente, que incluye las sedes de Bucaramanga, Barrancabermeja, Puerto Carreño, Cúcuta, Ocaña, Tibú, Puerto Inírida, Yopal, Mitú y Villavicencio, todas ubicadas en la región de la Orinoquía y áreas circundantes del oriente colombiano. Esta población comprende estudiantes, docentes, personal administrativo y actores externos interesados en las iniciativas culturales del CACTO, quienes constituyen el ámbito de influencia directa del centro y sus actividades de proyección social.

### **3.3 Muestra**

La muestra se seleccionó mediante un muestreo no probabilístico por conveniencia, priorizando la accesibilidad y disponibilidad de los participantes desde la perspectiva de los

investigadores. En este caso, se eligió la sede de Villavicencio como foco de la muestra, dado que los investigadores son estudiantes de esta sede, lo que facilita la interacción directa con la comunidad académica y el acceso a información relevante. Además, Villavicencio destaca como un centro neurálgico de la Rectoría Oriente, tanto por su relevancia administrativa como por su posición estratégica en la región, lo que la convierte en un punto representativo para evaluar el impacto potencial del CACTO.

La muestra se concreta en las cinco entrevistas realizadas a María Isabel Rodríguez Leyva, cuya experiencia directiva aporta una perspectiva privilegiada sobre el proyecto. Aunque limitada en tamaño, esta selección responde a la lógica del estudio de caso, que busca profundidad y detalle en lugar de representatividad estadística, enfocándose en la calidad de los datos obtenidos para orientar el desarrollo del CMS y la difusión del patrimonio cultural del CACTO.

## 4 CAPITULO IV

### 4.1 Metodología de desarrollo de software

El marco de trabajo usado para el proyecto de la plataforma web de difusión cultural de UNIMINUTO Oriente es SCRUM. Esta elección se fundamenta en una serie de ventajas clave que ofrece:

- La gestión ágil del proyecto mediante sprints que permiten entregas incrementales de funcionalidad
- La adaptabilidad a cambios en los requerimientos durante el desarrollo
- La transparencia en el proceso mediante reuniones y revisiones de sprint
- La participación de los stakeholders de UNIMINUTO en el desarrollo
- La mejora continua a través de retrospectivas regulares
- La priorización efectiva de funcionalidades según las necesidades institucionales

Durante el desarrollo de la plataforma se emplearon los distintos artefactos y eventos de SCRUM que permitieron la construcción organizada y eficiente de la aplicación. Las fases y elementos principales del desarrollo son los siguientes

A continuación, se presenta un cuadro comparativo que ilustra cómo Scrum se adaptó a las necesidades del proyecto de la plataforma web de difusión cultural de UNIMINUTO Oriente, en contraste con Cascada y Kanban, destacando ejemplos concretos del desarrollo

**Tabla 1 Cuadro comparativo de Scrum**

<b>Aspecto</b>	<b>Scrum</b>	<b>Cascada (Waterfall)</b>	<b>Kanban</b>
<b>Flexibilidad</b>	Alta: Ajustes iterativos en sprints basados en retroalimentación (ej., HU005 ajustada tras entrevistas).	Baja: Fases rígidas, cambios costosos tras la planificación inicial.	Media: Flexible para tareas continuas, pero sin ciclos estructurados.
<b>Colaboración</b>	Fuerte: Mesas de trabajo y revisiones con stakeholders (ej., entrevistas con María Isabel Rodríguez Leyva).	Limitada: Interacción solo al inicio y final, sin iteraciones.	Moderada: Enfocada en flujo, menos interacción con stakeholders.
<b>Entregas</b>	Incrementales: Funcionalidades cada 2 semanas (ej., carrusel de eventos en Figura 11).	Única: Producto completo al final, sin entregas parciales.	Continuas: Tareas completadas según flujo, sin ciclos definidos.
<b>Gestión de Requerimientos</b>	Dinámica: Historias de usuario priorizadas en Trello, adaptadas por retroalimentación.	Estática: Requerimientos fijos, difícil de modificar.	Flexible: Sin priorización estructurada, depende del flujo visual.

Aspecto	Scrum	Cascada (Waterfall)	Kanban
<b>Adecuación al Proyecto</b>	Ideal: Alineado con necesidades cambiantes del CACTO, entregas rápidas y colaborativas.	Inadecuada: No soporta ajustes dinámicos en proyectos culturales.	Parcial: Útil para mantenimiento, menos efectiva para desarrollo inicial.

#### 4.1.1 Fase de análisis

Para la gestión del proyecto bajo el marco de trabajo SCRUM, se utilizó Trello como la herramienta principal. Trello, una plataforma basada en tableros Kanban, permitió organizar las tareas, asignar responsables y realizar un seguimiento visual y colaborativo del progreso del proyecto. En este caso, Trello fue clave para gestionar el Product Backlog, planificar los sprints y monitorear el avance de las tareas en tiempo real.

**Figura 4. Tablero de Trello**



**Fuente: Autores**

El uso de Trello en el proyecto proporcionó las siguientes ventajas:

- Visualización clara del estado de las tareas en cada etapa del desarrollo.
- Asignación eficiente de responsabilidades a los miembros del equipo.
- Priorización de historias de usuario según las necesidades definidas por los stakeholders.
- Facilitación de la comunicación y colaboración entre el equipo de desarrollo y las partes interesadas.
- El Product Backlog se construyó durante la fase inicial del proyecto, basada en un análisis exhaustivo de las necesidades identificadas por los stakeholders. Este proceso se llevó a cabo de la siguiente manera:
  - Entrevistas y reuniones con stakeholders: Se realizaron sesiones para recopilar información sobre los problemas y expectativas relacionados con la difusión de contenido cultural en UNIMINUTO Orinoquia.
  - Priorización de requerimientos: Se clasificaron las funcionalidades según el valor que aportaban a los usuarios y a la institución, utilizando criterios como urgencia y viabilidad técnica.
  - Definición de historias de usuario: Cada requerimiento se transformó en una historia de usuario con criterios de aceptación claros, facilitando su implementación.
  - Ejemplos de historias de usuario que formaron parte del backlog fueron:
    - "Como administrador cultural, quise poder subir y categorizar contenido multimedia para que los usuarios lo encontraran fácilmente."
    - "Como estudiante, quise acceder a eventos culturales pasados y futuros desde mi dispositivo móvil para estar informado."

- En Trello, el backlog se organizó mediante tarjetas individuales para cada historia de usuario, las cuales se priorizaron y ajustaron según retroalimentación recibida a lo largo del proyecto.
- Los sprints se planificaron en iteraciones de dos semanas, cada una con objetivos específicos alineados con las prioridades del Product Backlog. El equipo siguió las ceremonias estándar de SCRUM para estructurar el trabajo:
- Sprint Planning: Al inicio de cada sprint, se seleccionaron las historias de usuario del backlog que el equipo se comprometió a completar, trasladándolas a la columna en Trello.
- Sprint Review: Al finalizar cada sprint, se presentaron las funcionalidades completadas a los stakeholders para obtener su retroalimentación.
- Sprint Retrospective: El equipo analizó internamente el proceso, identificó áreas de mejora y ajustó la metodología para los siguientes sprints.
- Trello permitió un seguimiento dinámico, ya que las tarjetas se movieron entre las columnas "En Progreso", "Revisión" y "Hecho" a medida que se avanzaba en las tareas.

Para complementar el análisis de la metodología Scrum utilizada en el desarrollo de la plataforma web de difusión cultural de UNIMINUTO Oriente, es fundamental abordar la herramienta que facilitó la gestión ágil del proyecto: Trello. Esta plataforma de gestión de tareas fue seleccionada por su capacidad para adaptarse a las necesidades dinámicas de un equipo pequeño, trabajando en un entorno colaborativo y con requerimientos en constante evolución, como fue el caso del proyecto CACTO (Centro de Arte, Cultura y Tecnología Oriental). A lo

largo del desarrollo, Trello proporcionó una serie de ventajas clave que aseguraron la eficiencia, transparencia y flexibilidad necesarias para cumplir con los objetivos institucionales.

El uso de Trello en este proyecto ofreció beneficios significativos, como la visualización clara del estado de las tareas, permitiendo al equipo seguir el progreso en tiempo real mediante tableros Kanban intuitivos. Esto fue especialmente útil para un equipo compacto, donde la simplicidad y la claridad eran prioritarias. Además, la asignación eficiente de responsabilidades a través de tarjetas asignadas a miembros específicos facilitó la distribución del trabajo, mientras que la priorización de historias de usuario permitió al equipo enfocarse en las funcionalidades más valiosas para los stakeholders, como la integración de contenido multimedia cultural. La herramienta también fue clave para facilitar la comunicación y colaboración, ya que los comentarios en las tarjetas y las menciones directas promovieron una interacción fluida tanto dentro del equipo como con las partes interesadas, incluyendo las mesas de trabajo con figuras como María Isabel Rodríguez Leyva. Finalmente, Trello permitió un seguimiento dinámico del Product Backlog, con historias de usuario organizadas en tableros que se ajustaban fácilmente a las iteraciones de dos semanas, alineándose perfectamente con el ritmo de los sprints.

Para comprender mejor por qué Trello fue la elección ideal, es útil contrastarlo con otras herramientas populares de gestión de proyectos, como Jira y Asana. A continuación, se presenta un cuadro comparativo que evalúa cada herramienta en función de aspectos críticos para el éxito del proyecto CACTO, destacando ejemplos concretos del desarrollo y explicando cómo Trello respondió de manera óptima a las necesidades del equipo y los stakeholders.

**Tabla 2. Cuadro Comparativo de Trello**

<b>Aspecto</b>	<b>Trello</b>	<b>Jira</b>	<b>Asana</b>
<b>Visualización clara del estado de las tareas</b>	Excelente: Tableros Kanban intuitivos con columnas personalizables (ej., "En Progreso", "Hecho") para un seguimiento visual sencillo, ideal para equipos pequeños.	Compleja: Tableros Kanban disponibles, pero más orientada a flujos de trabajo técnicos con configuraciones detalladas, lo que puede ser abrumador para equipos no técnicos.	Buena: Ofrece vistas de tablero, lista y calendario, pero la interfaz es menos minimalista que Trello, con más opciones que pueden distraer en proyectos simples.
<b>Asignación eficiente de responsabilidades</b>	Fácil: Asignación directa de tarjetas a miembros con notificaciones integradas, adecuada para equipos compactos como el del proyecto CACTO.	Robusta: Asignaciones detalladas con roles y permisos, pero requiere configuración previa, más adecuada para equipos grandes o técnicos.	Versátil: Permite asignar tareas con claridad, pero su enfoque en múltiples vistas puede complicar la gestión en proyectos con pocos miembros.

<b>Aspecto</b>	<b>Trello</b>	<b>Jira</b>	<b>Asana</b>
<b>Priorización de historias de usuario</b>	Sencilla: Historias de usuario organizadas en tarjetas priorizadas en Trello, ajustadas fácilmente por retroalimentación (ej., HU005).	Avanzada: Potente para priorizar historias con puntos de historia y epics, pero su curva de aprendizaje es alta para usuarios no familiarizados con Scrum.	Flexible: Soporta priorización con listas y etiquetas, pero menos intuitiva para historias de usuario en comparación con el enfoque visual de Trello.
<b>Facilitación de comunicación y colaboración</b>	Alta: Entorno colaborativo con comentarios en tarjetas y menciones, ideal para mesas de trabajo con stakeholders (ej., entrevistas con María Isabel Rodríguez Leyva).	Moderada: Ofrece colaboración, pero está más enfocada en desarrolladores, con menos énfasis en interacción directa con stakeholders no técnicos.	Alta: Fuerte en colaboración con comentarios y actualizaciones, pero su interfaz más cargada puede ralentizar la interacción en equipos pequeños.
<b>Seguimiento dinámico del Product Backlog</b>	Dinámico: Backlog gestionado en tableros, con tarjetas	Completo: Excelente para gestionar backlogs complejos,	Eficiente: Gestiona backlogs con listas, pero su enfoque

Aspecto	Trello	Jira	Asana
	movibles para sprints de 2 semanas, perfecto para iteraciones rápidas en el proyecto CACTO.	pero requiere experiencia en configuración de flujos Scrum, lo que puede ser excesivo para este proyecto.	multipropósito no está tan optimizado para Scrum como Trello o Jira.
<b>Adecuación al Proyecto</b>	Ideal: Simplicidad y visualización Kanban perfectas para un equipo pequeño, necesidades cambiantes del CACTO y colaboración con stakeholders no técnicos.	Parcial: Potente para proyectos técnicos grandes, pero su complejidad es innecesaria para un equipo pequeño y un proyecto cultural.	Buena: Versátil, pero su enfoque generalista no se alinea tan bien con Scrum ni con la simplicidad requerida por el proyecto CACTO.

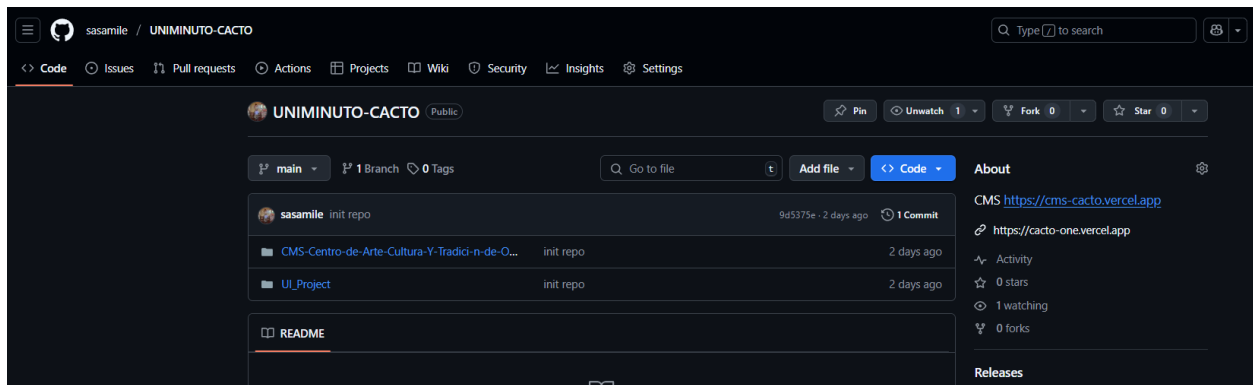
Este cuadro comparativo ilustra cómo Trello se destacó en los aspectos más relevantes para el proyecto de la plataforma web de difusión cultural. Su enfoque visual y colaborativo, combinado con la facilidad de uso, lo convirtió en la herramienta óptima para gestionar un desarrollo ágil en un entorno académico y cultural como el de UNIMINUTO Oriente. En

contraste, Jira, aunque potente, habría introducido una complejidad innecesaria para un equipo pequeño y no técnico, mientras que **Asana**, con su versatilidad, no ofrecía la misma claridad y simplicidad que Trello proporcionó para las iteraciones rápidas y la colaboración directa con los stakeholders.

#### 4.1.2 Repositorio

El proyecto fue subido a GitHub para centralizar el código y facilitar su acceso, con el fin de documentar el desarrollo del CMS. Esta plataforma permite un control de versiones eficiente y asegura la trazabilidad de los cambios realizados. La elección de GitHub responde a la intención de mantener el trabajo accesible y alineado con buenas prácticas, permitiendo futuras iteraciones o contribuciones de manera práctica y abierta.

**Figura 5. Repositorio de github**



**Fuente: Autores**

Enlace: <https://github.com/sasamile/UNIMINUTO-CACTO>

## **4.2 Análisis de requerimientos**

Los requerimientos funcionales especifican las funcionalidades esenciales que la plataforma web de difusión cultural de UNIMINUTO Orinoquia debe ofrecer para satisfacer las necesidades de los usuarios y guiar el proceso de desarrollo. Los requerimientos no funcionales definen las características de calidad, rendimiento y restricciones técnicas del sistema.

### **4.2.1 Requerimientos funcionales del sitio web**

- Debe mostrar eventos destacados en la página principal, accesibles públicamente para que todos los usuarios puedan visualizar el contenido.
- Debe permitir a los usuarios dejar comentarios en los eventos
- Debe permitir ver todas las noticias importantes

### **4.2.2 Requerimientos no funcionales del sitio web**

- El tiempo de carga de las páginas no debe exceder los 3 segundos
- El sistema debe soportar múltiples usuarios concurrentes
- Las operaciones de carga de contenido deben ser optimizadas
- La interfaz debe ser intuitiva y fácil de navegar
- El diseño debe ser responsive para adaptarse a todos los dispositivos
- Los elementos de la interfaz deben ser consistentes en todo el sitio

#### 4.2.3 Requerimientos funcionales del CMS

- La plataforma web debe permitir crear, editar y eliminar eventos culturales
- El sistema debe permitir subir y gestionar contenido sobre arte, cultura y tradición
- Cada evento debe poder incluir texto descriptivo, imágenes, videos y podcasts.
- Debe mantener un historial de contenido publicado
- Debe proporcionar un panel de administración para gestores de contenido
- Debe implementar un sistema seguro de autenticación para administradores

#### 4.2.4 Requerimientos no funcionales del CMS

- Debe funcionar en los principales navegadores (Chrome, Firefox, Safari, Edge)
- Debe ser accesible desde dispositivos móviles, tablets y computadores
- Debe mantener la consistencia visual en diferentes plataformas
- Debe implementar autenticación segura para el acceso administrativo

### 4.3 Historias de usuario

**Tabla 3. Historia de usuario Loguin**

<b>ID Historia</b>	HU001
<b>Nombre Historia</b>	Loguin
<b>Descripción</b>	Como administrador de contenido, quiero poder iniciar sesión en la plataforma para gestionar el contenido de la página del usuario.

<p><b>Validación o Aceptación</b></p>	<ol style="list-style-type: none"> <li>1. Se espera que el administrador pueda acceder de forma segura al sistema.</li> <li>2. Se espera que solo el usuario con el rol "admin" tenga acceso al panel de administración.</li> <li>3. Se espera que los intentos de acceso no autorizados sean bloqueados y muestren un mensaje de error adecuado.</li> <li>4. Se espera que el administrador pueda gestionar el contenido de la página del usuario una vez dentro del sistema.</li> </ol>
---------------------------------------	---

Fuente: Autores

**Tabla 4. Historia de usuario Gestión de Eventos**

<p><b>ID Historia</b></p>	<p>HU002</p>
<p><b>Nombre Historia</b></p>	<p>Gestión de Eventos</p>
<p><b>Descripción</b></p>	<p>Como administrador, quiero crear, editar y eliminar eventos en la plataforma para que los usuarios puedan ver información actualizada sobre próximos eventos.</p>
<p><b>Validación o Aceptación</b></p>	<ol style="list-style-type: none"> <li>1. Se espera que el administrador pueda gestionar los eventos de manera eficiente.</li> <li>2. Se espera que los eventos creados se muestren correctamente en la página pública.</li> <li>3. Se espera que los usuarios puedan ver eventos actualizados sin retrasos.</li> </ol>

Fuente: Autores

**Tabla 5. Historia de usuario Gestión de contenido**

<b>ID Historia</b>	HU003
<b>Nombre Historia</b>	Gestión de Contenido
<b>Descripción</b>	Como administrador, quiero crear y personalizar secciones de la página para mostrar contenido relevante como carteles, publicidad, imágenes y podcasts.
<b>Validación o Aceptación</b>	<ol style="list-style-type: none"><li>1. Se espera que el administrador pueda gestionar las secciones de la página sin dificultad.</li><li>2. Se espera que los usuarios vean el contenido actualizado en la página pública.</li><li>3. Se espera que el CMS permita una fácil administración sin conocimientos técnicos avanzados.</li></ol>

Fuente: Autores

**Tabla 6. Historia de usuario Visualización de Testimonios**

<b>ID Historia</b>	HU004
<b>Nombre Historia</b>	Visualización de Testimonios
<b>Descripción</b>	Como visitante de la página, quiero ver los testimonios de otros usuarios para conocer sus experiencias con los eventos y el contenido de la página.

<p><b>Validación o Aceptación</b></p>	<ol style="list-style-type: none"> <li>1. Se espera que los visitantes puedan ver testimonios de otros usuarios.</li> <li>2. Se espera que los testimonios se muestren de manera clara y ordenada.</li> <li>3. Se espera que los testimonios reflejen opiniones reales y mejoren la credibilidad de la página.</li> </ol>
---------------------------------------	---

**Fuente: Autores**

**Tabla 7. Historia de usuario Visualización de Eventos y Contenido**

<p><b>ID Historia</b></p>	<p>HU005</p>
<p><b>Nombre Historia</b></p>	<p>Visualización de Eventos y Contenido</p>
<p><b>Descripción</b></p>	<p>Como visitante de la página, quiero poder visualizar los eventos culturales próximos y ver la información de eventos, imágenes, videos y podcasts para estar informado sobre actividades y novedades.</p>
<p><b>Validación o Aceptación</b></p>	<ol style="list-style-type: none"> <li>1. Se espera que los visitantes puedan acceder fácilmente a la información de eventos y contenido.</li> <li>2. Se espera que el contenido se muestre correctamente en todos los dispositivos.</li> <li>3. Se espera que la página pública tenga una interfaz intuitiva para mejorar la experiencia del usuario.</li> </ol>

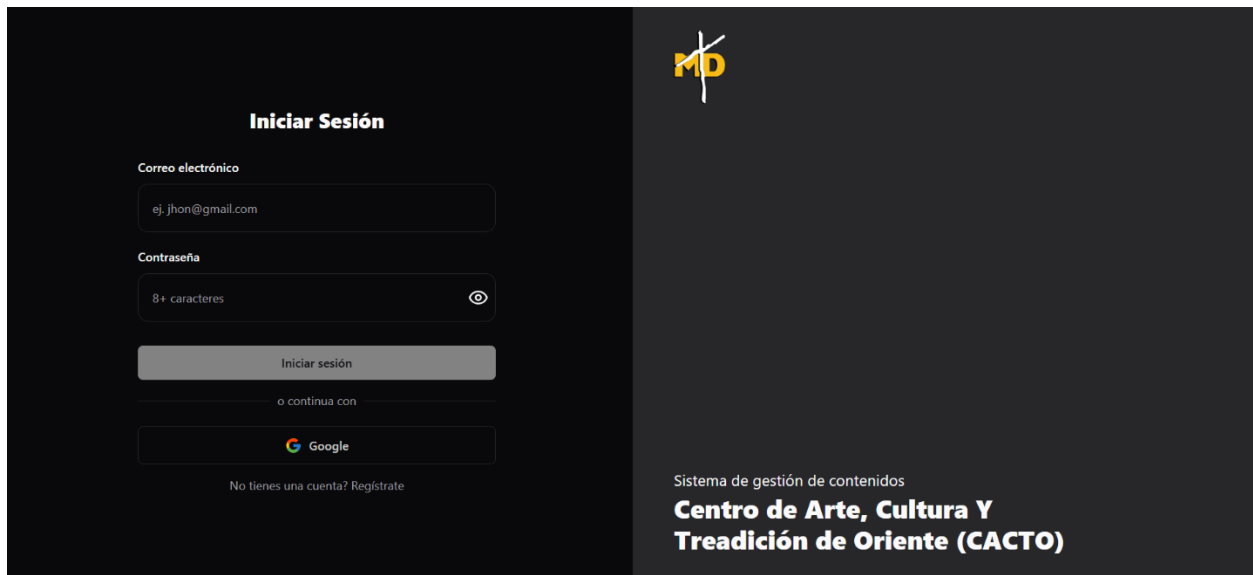
**Fuente: Autores**

#### 4.4 Diseño de la aplicación

El CMS implementa un sistema de gestión de contenidos que permitirá a los administradores mantener actualizada la información del centro cultural. A continuación, se presentan las ilustraciones correspondientes a las historias de usuario que definen la funcionalidad del sistema.

La figura muestra la interfaz de inicio de sesión del sistema de gestión de contenidos para CACTO. Esta interfaz corresponde a la Historia de Usuario HU001 (Loguin), donde el administrador de contenido puede ingresar sus credenciales para acceder al panel de administración. La pantalla presenta campos para correo electrónico y contraseña, así como la opción de iniciar sesión con Google. Esta funcionalidad garantiza que solo los usuarios autorizados puedan gestionar el contenido del sitio web.

**Figura 6. Interfaz del Loguin**



**Fuente: Autores**

La figura muestra la interfaz de administración para la gestión de eventos del sistema UNIMINUTO. Esta pantalla corresponde a la Historia de Usuario HU002 (Gestión de Eventos), donde el administrador puede visualizar, crear, editar y eliminar eventos culturales de la institución. La interfaz presenta una tabla con cuatro eventos existentes, mostrando título, descripción, ubicación y fechas de inicio y fin para cada uno. En la parte superior se observa un botón "Agregar" que permite la creación de nuevos eventos, así como un campo de filtrado por título para facilitar la búsqueda. Esta funcionalidad permite mantener actualizada la información de eventos culturales como conciertos, exposiciones, festivales y ferias artesanales que se realizarán en Villavicencio, Meta.

**Figura 7. Interfaz de Gestión de Eventos**

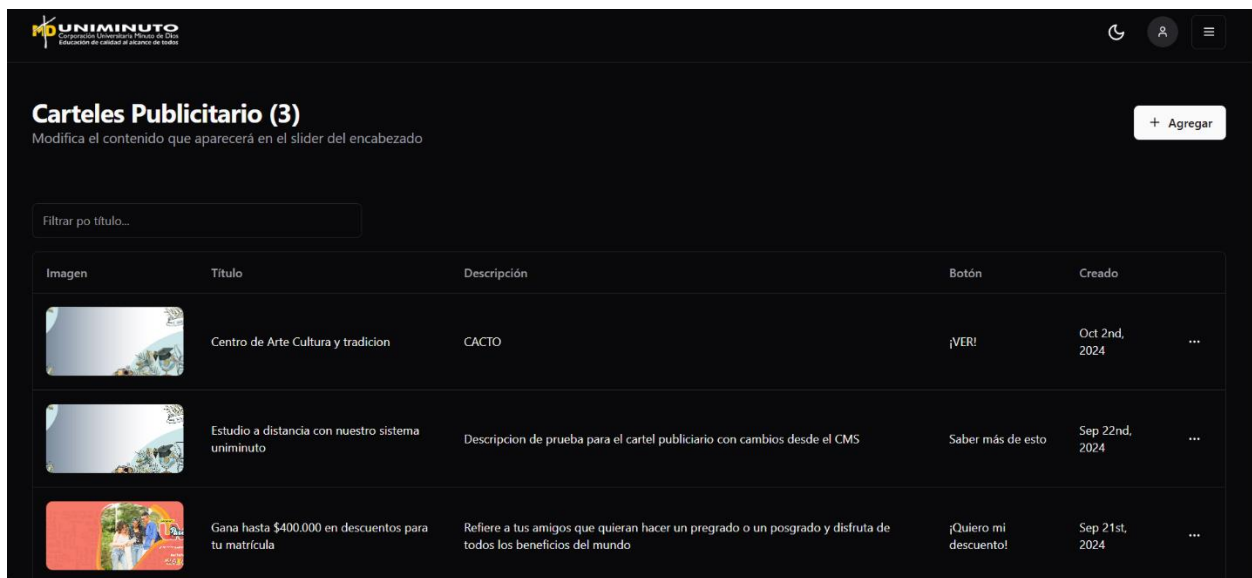
Titulo	Descripción	Ubicación	Fecha de inicio	Fecha de fin	
Concierto de Música Andina	Presentación en vivo de música andina a cargo de la agrupación del Centro de Cultura de Uniminuto.	Villavicencio, Meta, Parque las Malocas	30 sept 2024	1 oct 2024	...
Exposición de Arte Contemporáneo	Muestra de las obras de artistas emergentes del Centro de Arte de Uniminuto.	Centro de Convenciones, Villavicencio, Meta	11 oct 2024	11 oct 2024	...
Festival de Música Folclórica	Festival anual que celebra la tradición musical de las regiones colombianas.	Villavicencio, Meta	1 oct 2024	6 oct 2024	...
Feria de Artesanías Locales	Exposición y venta de artesanías tradicionales hechas por comunidades locales.	Villavicencio, Meta	8 nov 2024	13 nov 2024	...

**Fuente: Autores**

La figura muestra la interfaz de administración para la gestión de carteles publicitarios dentro del sistema UNIMINUTO. Esta pantalla corresponde a la Historia de Usuario HU003

(Gestión de Contenido), donde el administrador puede crear y personalizar secciones de la página web. En este caso específico. La interfaz presenta una tabla con información de cada cartel, incluyendo la imagen promocional, título, descripción, texto del botón de acción y fecha de creación. El diseño permite filtrar por título y cuenta con un botón "Agregar" para incluir nuevos carteles. Los contenidos actuales promocionan el Centro de Arte Cultura y tradición, programas de estudio a distancia y descuentos para matrículas, facilitando así la gestión del contenido promocional de manera intuitiva sin necesidad de conocimientos técnicos avanzados.

**Figura 8. Interfaz de Gestión de contenido**

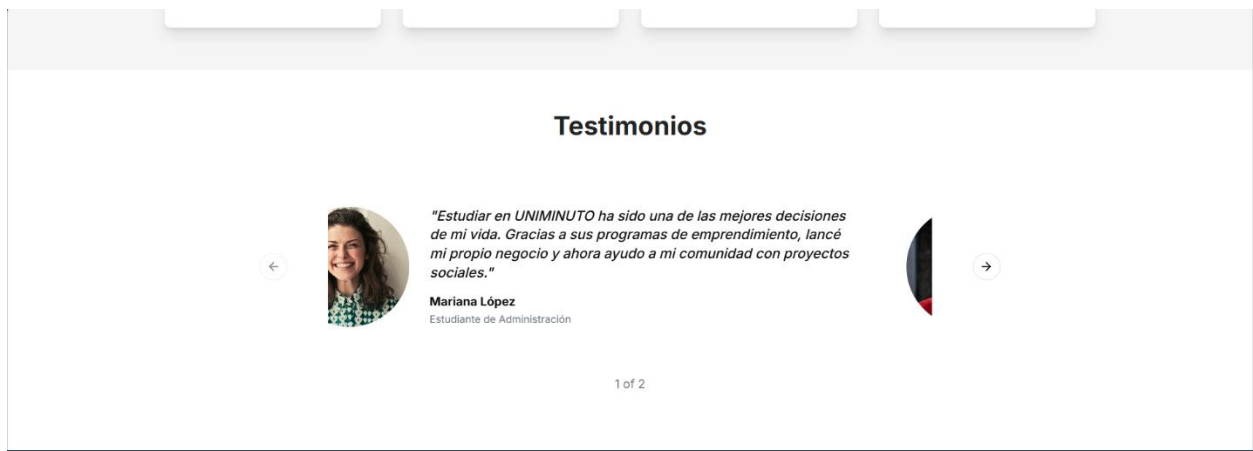


**Fuente: Autores**

La figura muestra la sección de testimonios en la página pública de UNIMINUTO, correspondiente a la Historia de Usuario HU004. Esta sección está diseñada para que los visitantes puedan conocer las experiencias de otros usuarios con la institución. La interfaz presenta un diseño limpio y ordenado, con un testimonio destacado de Mariana López, estudiante

de Administración, quien comparte su experiencia positiva sobre cómo estudiar en UNIMINUTO impactó su vida profesional y su capacidad para contribuir a su comunidad. La sección incluye una fotografía de la estudiante, su testimonio completo, nombre y carrera. También cuenta con controles de navegación (flechas izquierda y derecha) que permiten a los visitantes explorar más testimonios, indicando que se muestra 1 de 2 testimonios disponibles. Esta funcionalidad ayuda a aumentar la credibilidad de la institución al mostrar experiencias reales de sus estudiantes.

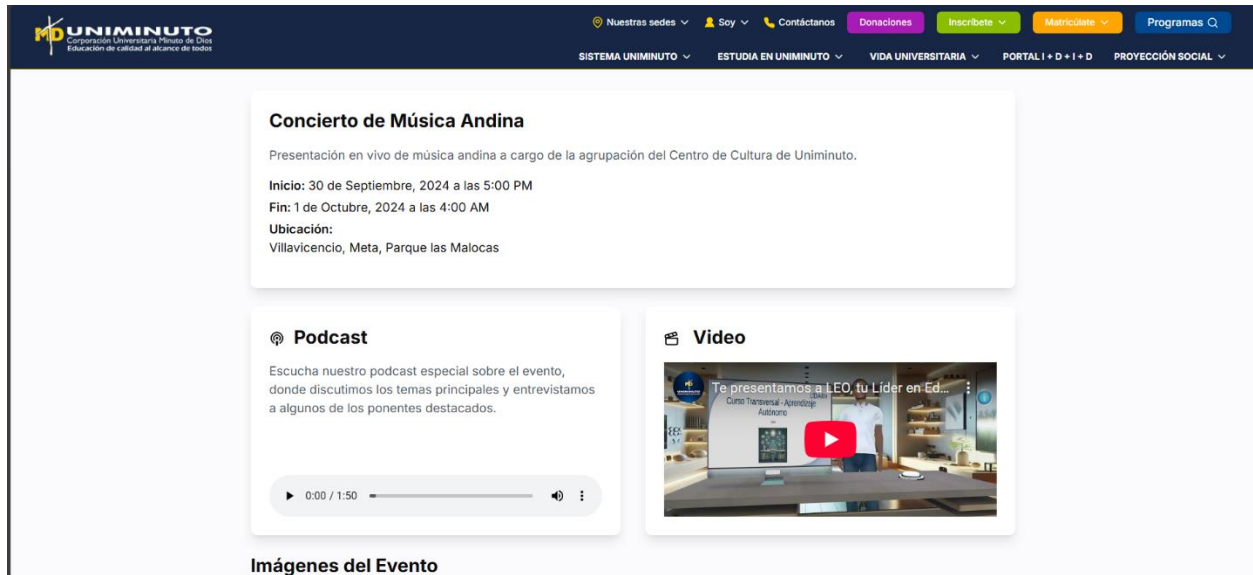
**Figura 9. Interfaz de Testimoniales**



**Fuente: Autores**

La figura muestra la página de correspondiente a la Historia de Usuario HU005 (Visualización de Eventos y Contenido). La interfaz presenta información estructurada del evento con descripción, fechas y ubicación, complementada con secciones de podcast y video relacionados. El diseño es intuitivo, con controles de reproducción multimedia y navegación principal visible, permitiendo a los visitantes acceder fácilmente a toda la información relevante del evento cultural organizado por UNIMINUTO.

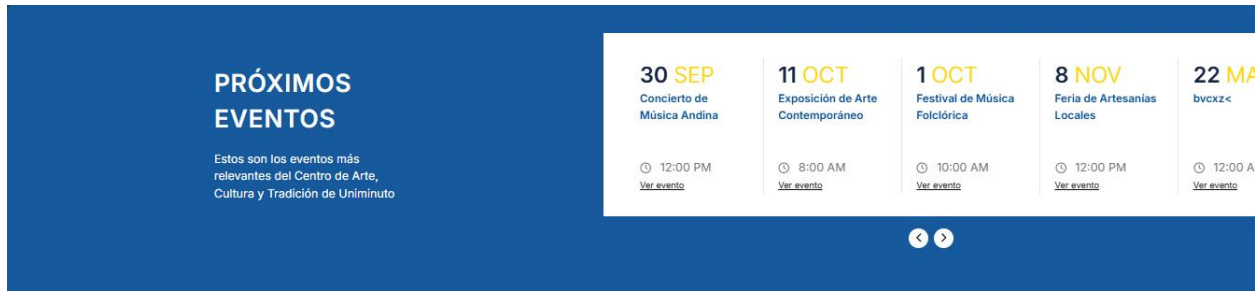
**Figura 10. Interfaz Visualización de Eventos y contenido**



**Fuente: Autores**

La figura corresponde a la Historia de Usuario HU005 (Visualización de Eventos y Contenidos) y muestra una sección de la plataforma CACTO. La interfaz destaca un carrusel de "Próximos Eventos" con fechas, nombres y horarios, como el "Concierto de Música Andina" del 30 de septiembre, y una descripción del Centro bajo "¿Quiénes somos?". El diseño intuitivo incluye navegación mediante flechas y un enlace "Ver evento", facilitando a los usuarios de UNIMINUTO el acceso a información cultural relevante.

**Figura 11. Interfaz Visualización de Eventos Futuros o Actuales**



### ¿Quiénes somos?

Somos el Centro de Arte, Cultura y Tradición de Oriente, una iniciativa de UNIMINUTO Rectoría Oriente dedicada a fomentar el desarrollo sociocultural en la región. En nuestro espacio se integran diversas expresiones artísticas y culturales, promoviendo la inclusión, la diversidad y la preservación del patrimonio cultural. Nuestro principal objetivo es fortalecer la identidad cultural de la comunidad a través de la investigación, formación y promoción de actividades artísticas y culturales.



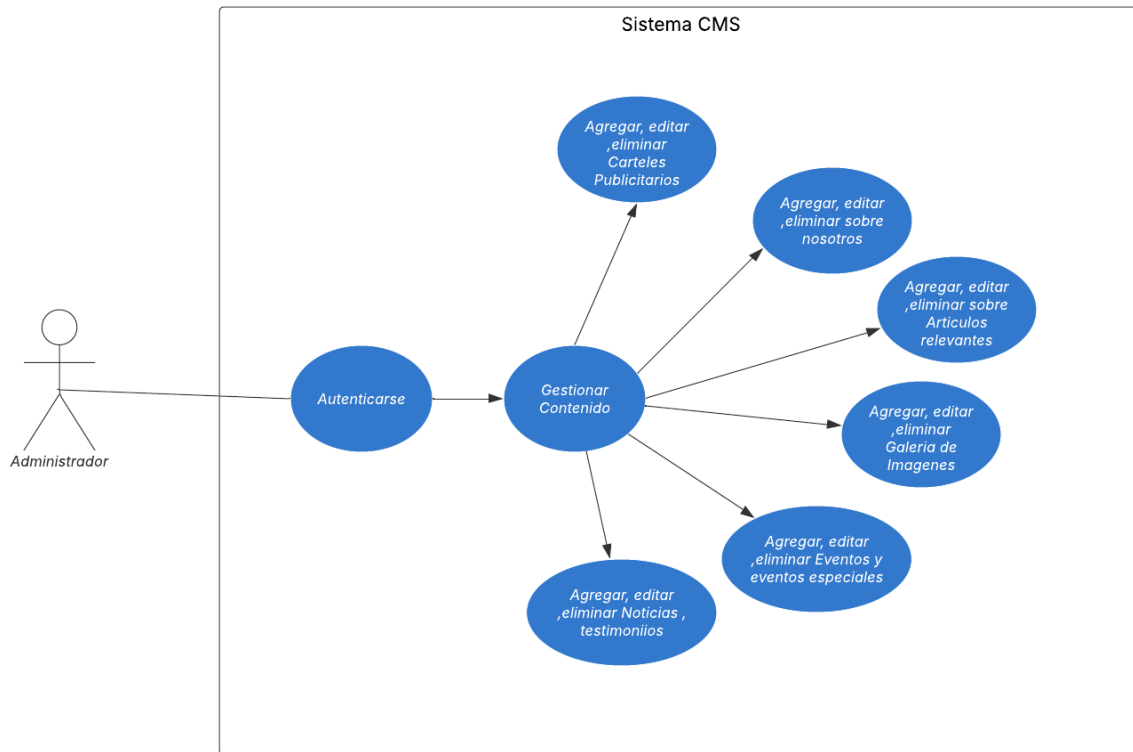
**Fuente: Autores**

## 4.4.1 Casos de uso

### 4.4.1.1 Caso de uso: Administrador

El diagrama resalta el rol central del administrador como gestor del contenido, con acceso exclusivo tras la autenticación. Cada subcaso corresponde a un modelo de la base de datos y las acciones (agregar, editar, eliminar) cubren las operaciones CRUD. Esto asegura que el administrador pueda mantener el sitio actualizado, alineado con los requisitos de un CMS.

**Figura 12. Caso de uso de sistema CMS**

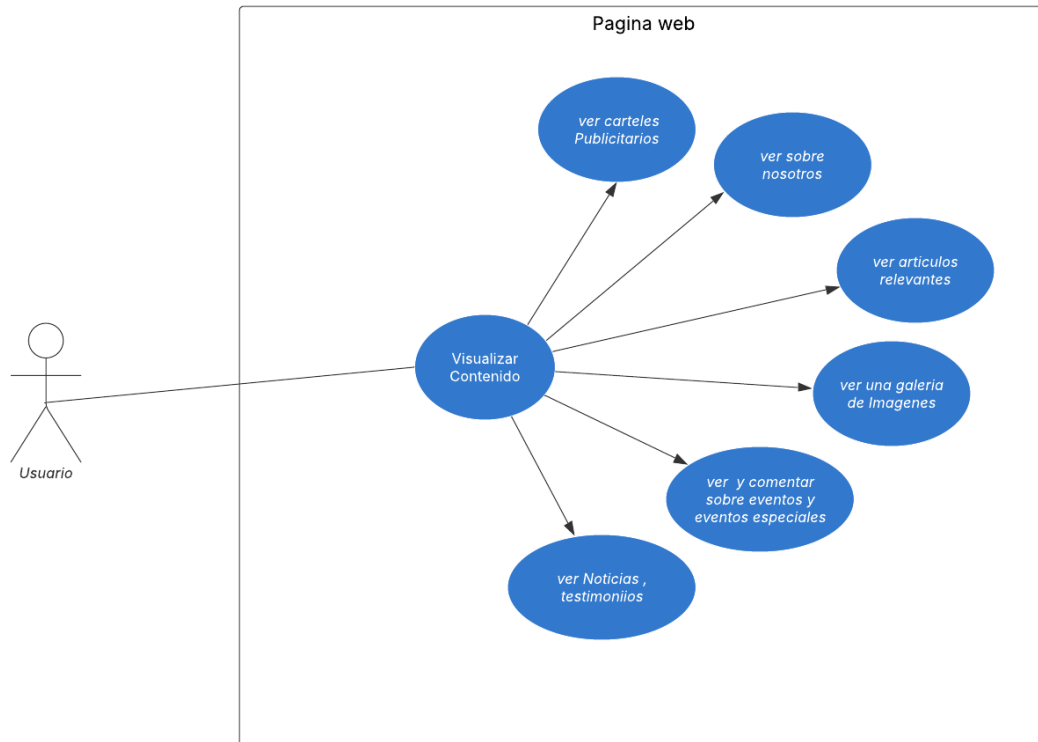


**Fuente: Autores**

#### 4.4.1.2 Caso de uso: usuario

El diagrama muestra un acceso público sin necesidad de autenticación, enfocado en la visualización pasiva del contenido gestionado por el administrador. La opción de comentar en eventos agrega una interacción activa, permitiendo a los usuarios participar. Los subcasos corresponden al menú de navegación que compartiste, reflejando una experiencia de usuario estructurada y completa.

**Figura 13. Caso de uso página web**



**Fuente: Autores**

## 4.4.2 Diagrama de secuencia

### 4.4.2.1 Diagrama: Administradores

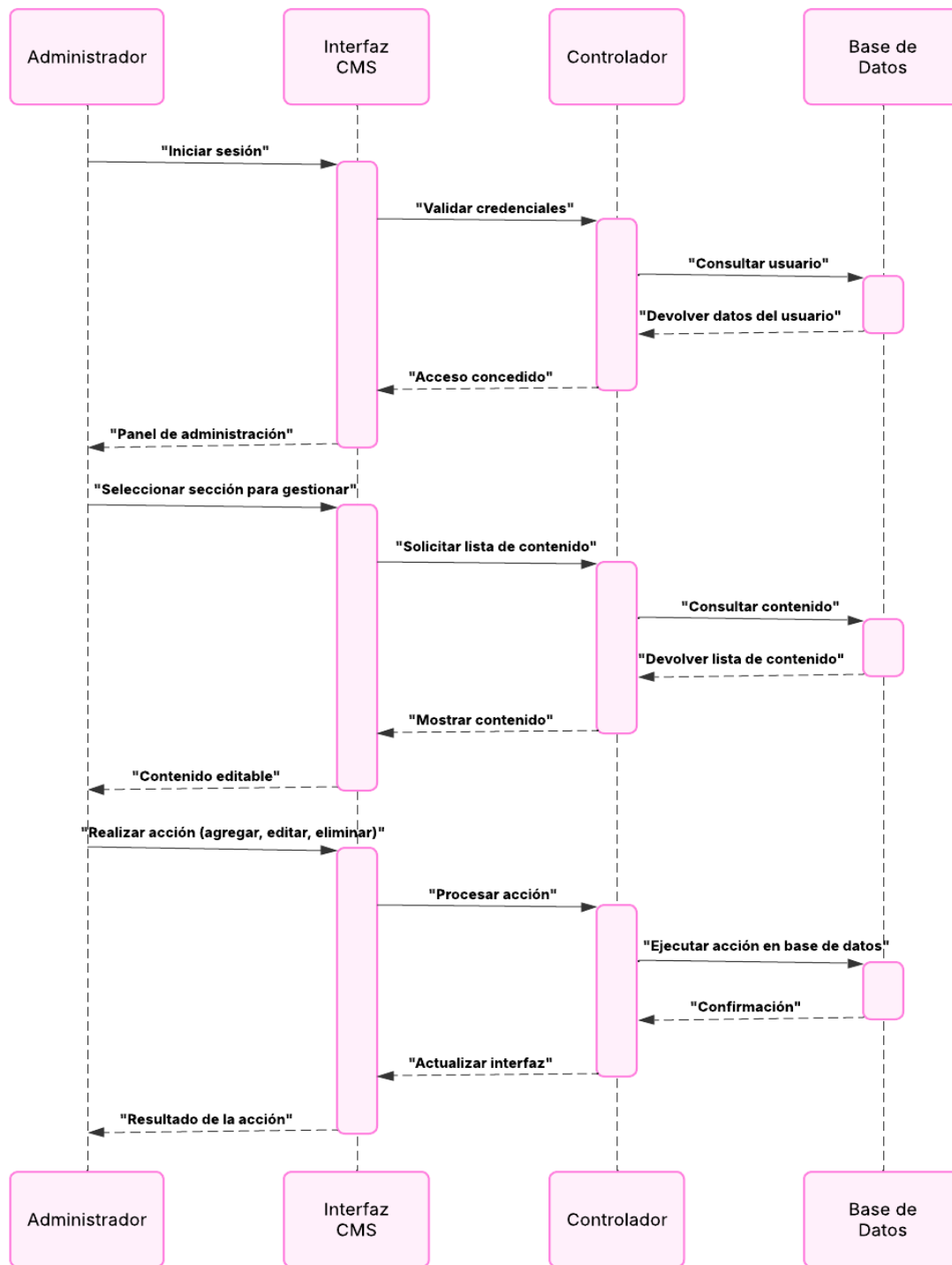
Este diagrama representa de manera general el flujo de trabajo del administrador en el CMS. Comienza con la autenticación, un paso esencial para garantizar que solo usuarios autorizados accedan al sistema. Una vez autenticado, el administrador puede seleccionar cualquier sección (como carteles, noticias, eventos, etc.) y realizar acciones CRUD (Crear, Leer, Actualizar, Eliminar) sobre el contenido. El flujo muestra cómo las solicitudes pasan del interfaz

Plataforma para la difusión de información sobre arte, cultura y tradición en UNIMINUTO 53  
Villavicencio.

al controlador, que interactúa con la base de datos, y cómo los cambios se reflejan en la interfaz.

Su diseño general permite aplicarlo a todas las secciones (carteles publicitarios, artículos, galería, etc.).

Figura 14. Diagrama de secuencia del CMS

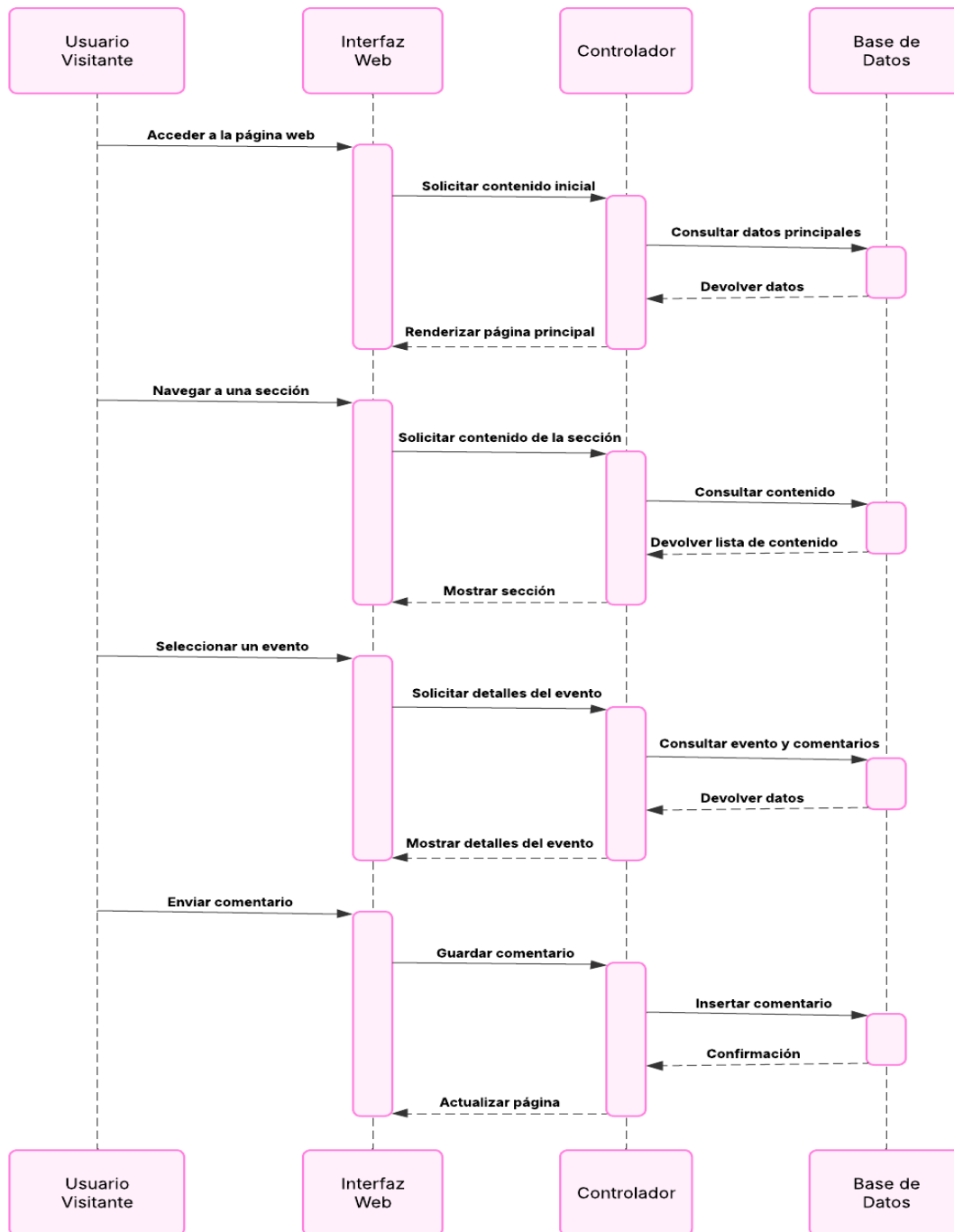


Fuente: Autores

#### **4.4.2.2 Diagrama: Usuario**

Este diagrama ilustra el flujo de interacción del usuario visitante con la página web. Comienza con el acceso a la página principal, donde se muestra un resumen de contenido. Luego, el usuario puede navegar a cualquier sección del menú (como noticias, eventos, galería, etc.), y el sistema consulta y muestra la información correspondiente. Para los eventos, se incluye la funcionalidad de ver detalles y agregar comentarios, reflejando una interacción más activa. El flujo muestra cómo las solicitudes se procesan a través de la interfaz web, el controlador y la base de datos, asegurando que el contenido sea accesible y actualizable en tiempo real.

Figura 15 Diagrama de secuencia de la Página web

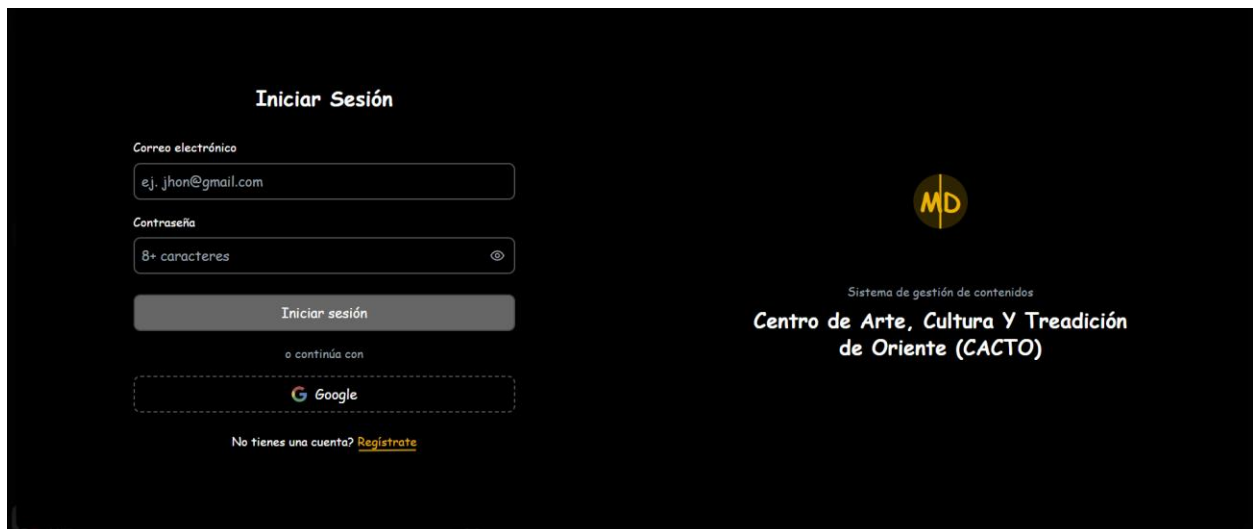


Fuente: Autores

#### 4.4.3 Mockups

Este mockup presenta una pantalla de inicio de sesión del sistema de gestión de contenidos del Centro de Arte, Cultura y Tradición de Oriente (CACTO). La interfaz muestra un formulario para ingresar con un correo electrónico y una contraseña, junto con una opción para iniciar sesión mediante Google. También incluye un enlace para registrarse en caso de no tener una cuenta. El diseño es minimalista, con un fondo oscuro y el logotipo de "MD" en la parte superior derecha, reflejando una propuesta funcional y accesible para la autenticación de usuarios.

**Figura 16. Mockup del Login**



**Fuente: Autores**

Este mockup presenta una pantalla de gestión de eventos del sistema del Centro de Cultura de Uniminuto, diseñada para organizar y visualizar actividades culturales y artísticas. La interfaz muestra una tabla con información detallada de los eventos, incluyendo título,

descripción, ubicación, fechas de inicio y fin, junto con un botón para agregar nuevos eventos. El diseño es minimalista, con un fondo oscuro y el logotipo de "MD Uniminuto" en la parte superior izquierda, ofreciendo una experiencia funcional y accesible para la administración de eventos. Además, incluye una opción de filtrado por título, facilitando la búsqueda y gestión de actividades para los usuarios del sistema.

**Figura 17 Mockup de Gestión de Eventos**

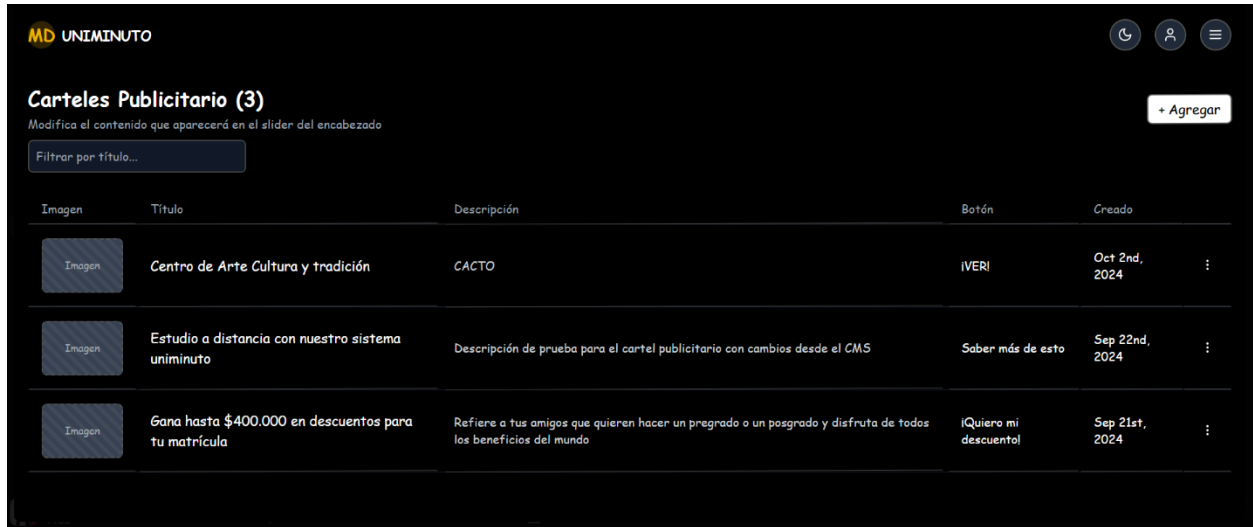
Título	Descripción	Ubicación	Fecha de inicio	Fecha de fin	
Concierto de Música Andina	Presentación en vivo de música andina a cargo de la agrupación del Centro de Cultura de Uniminuto.	Villavicencio, Meta, Parque las Malacas	30 sept 2024	1 oct 2024	⋮
Exposición de Arte Contemporáneo	Muestra de las obras de artistas emergentes del Centro de Arte de Uniminuto.	Centro de Convenciones, Villavicencio, Meta	11 oct 2024	11 oct 2024	⋮
Festival de Música Folclórica	Festival anual que celebra la tradición musical de las regiones colombianas.	Villavicencio, Meta	1 oct 2024	6 oct 2024	⋮
Feria de Artesanías Locales	Exposición y venta de artesanías tradicionales hechas por comunidades locales.	Villavicencio, Meta	8 nov 2024	13 nov 2024	⋮

**Fuente: Autores**

Este mockup presenta una interfaz de gestión de carteles publicados en el sistema de Uniminuto, específicamente en la sección "Carteles Publicitario". La pantalla muestra una tabla con tres entradas, cada una con una imagen, título, descripción, botón de acción, creador y fecha de creación. Los carteles están relacionados con el Centro de Arte, Cultura y Tradición de Oriente (CACTO), un estudio a distancia, y una promoción de descuentos. La interfaz tiene un

diseño oscuro y funcional, con opciones para filtrar por título y un botón para agregar nuevos carteles, lo que sugiere un sistema organizado para la administración de contenido publicitario.

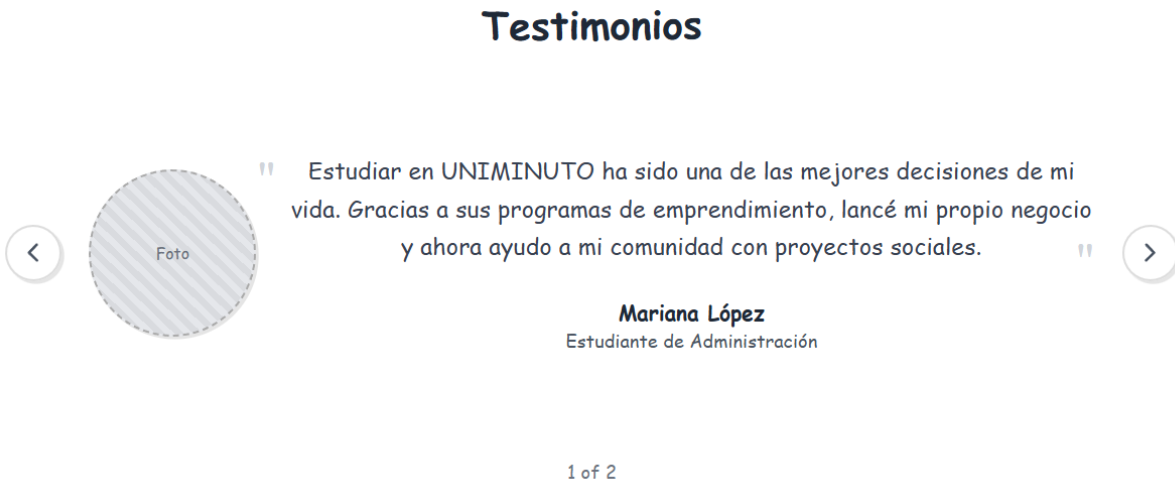
**Figura 18. Mockup de Gestión de contenido**



**Fuente: Autores**

Este mockup presenta una sección de testimonios diseñada para el sistema de Uniminuto. La interfaz muestra un carrusel con un testimonio destacado de Mariana López, estudiante de Administración, quien comparte su experiencia positiva sobre cómo estudiar en Uniminuto ha impactado su vida, gracias al programa de emprendimiento y el apoyo a proyectos sociales. El diseño es limpio y minimalista, con un fondo claro, una imagen circular para la foto del usuario (aún sin cargar), y controles de navegación para desplazarse entre los testimonios, indicando que hay al menos dos entradas disponibles.

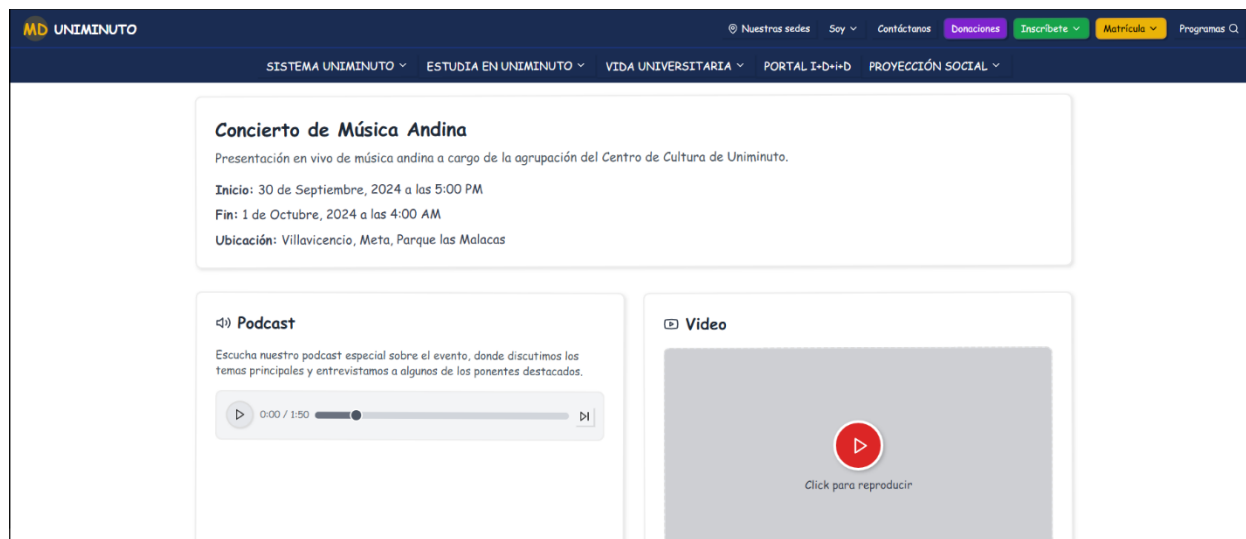
**Figura 19. Mockup de Testimoniales**



**Fuente: Autores**

Este mockup presenta una página de eventos del sistema de Uniminuto, específicamente para el "Concierto de Música Andina" organizado por el Centro de Cultura de Uniminuto. La interfaz detalla la información del evento, incluyendo la fecha de inicio y fecha de fin y la ubicación (Además, incluye una sección para un podcast con una breve descripción y controles de reproducción, junto a un espacio para un video, representado por un botón de reproducción. El diseño es claro y estructurado, con una barra de navegación superior que ofrece acceso a otras secciones del sistema, como "Sistema Uniminuto", "Estudia en Uniminuto", "Vida Universitaria", "Portal T y D", y "Proyección Social".

**Figura 20. Mockups de Visualización de Eventos y contenido**



**Fuente: Autores**

Este mockup presenta una pantalla de "Próximos Eventos" del sistema del Centro de Cultura de Uniminuto, diseñada para destacar las actividades culturales y artísticas programadas. La interfaz muestra un calendario visual con fechas clave, como el Concierto de Música Andina, la Exposición de Arte Contemporáneo, el Festival de Música Folclórica y la Feria Artesanal Locales, junto con una sección informativa que resalta la misión del centro. Con un diseño limpio y moderno, el fondo claro y los elementos visuales organizados facilitan la navegación, ofreciendo a los usuarios una vista rápida y atractiva de los eventos por venir, mientras se promueve la riqueza cultural y artística de la comunidad.

**Figura 21. Mockups Visualización de Eventos Futuros o Actuales**

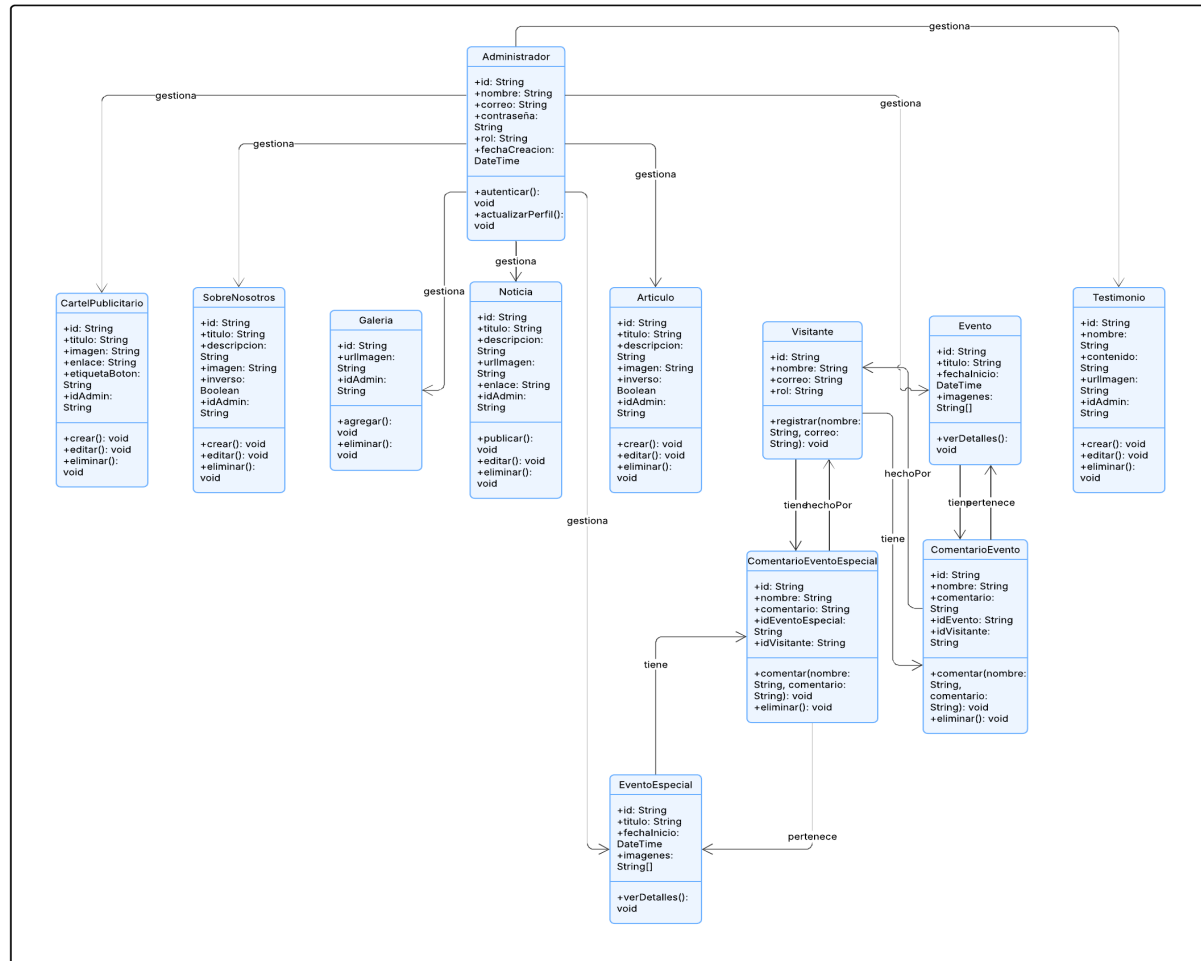


**Fuente: Autores**

#### 4.4.4 Diagrama de clases

Se realizó un diagrama de clases para visualizar la estructura de objetos y cómo se relacionan entre sí. Esto ayuda a definir las relaciones y propiedades de los objetos, y también facilita la programación al proporcionar una estructura clara y organizada.

**Figura 22. Diagrama de clases**



Fuente: Autores

## 4.5 Desarrollo del aplicativo

El desarrollo del aplicativo se llevó a cabo utilizando lenguajes y tecnologías modernas que permiten una implementación eficiente y escalable. La aplicación fue desarrollada principalmente con **Next.js**, un framework basado en JavaScript que facilita la creación de aplicaciones web del lado del cliente y del servidor, optimizando el rendimiento y la experiencia de usuario mediante renderizado del lado del servidor (SSR) y generación estática de sitios (SSG). Para la gestión de la base de datos, se empleó **Prisma ORM**, una herramienta que

simplifica las operaciones de acceso a datos y proporciona un modelo de esquema claro y tipado para interactuar con la base de datos CockroachDB configurada en el proyecto. Como back-end y almacenamiento, se integraron servicios en la nube, destacando **Uploadthing**, una solución para la subida y gestión de archivos (como imágenes) de manera segura y eficiente.

Además, se utilizó **Node.js** como entorno de ejecución para el servidor, permitiendo la ejecución del código JavaScript en el lado del servidor y la integración de las dependencias necesarias para el proyecto. Esta combinación de tecnologías asegura un desarrollo robusto, con un enfoque en la escalabilidad y la facilidad de mantenimiento.

A continuación, se presenta un fragmento de código que ilustra la configuración básica de Prisma en el proyecto, específicamente la definición de un modelo y una consulta simple para recuperar datos.

#### 4.5.1 Prisma como ORM

Prisma se utilizó para interactuar eficientemente con la base de datos, simplificando la gestión de modelos de datos, consultas y migraciones. Esta herramienta permitió una integración fluida con CockroachDB, optimizando las operaciones de lectura y escritura.

Figura 23. Estructura de la Base de Datos con Prisma ORM

```
1 // Try Prisma Accelerator: https://prisma.lydci5/accelerator-test
2
3 generator client {
4   provider = "prisma-client-js"
5 }
6
7 datasource db {
8   provider = "cockroachdb"
9   url      = env("DATABASE_URL")
10 }
11
12 model User {
13   id          String @id @default(cuid())
14   name        String!
15   email       String! @unique
16   emailVerified DateTime?
17   password    String!
18   image       String?
19
20   accounts Account[]
21
22   createdAt DateTime @default(now())
23   updatedAt DateTime @updatedAt
24 }
25
26 model Account {
27   userId      String
28   type        String
29   provider    String
30   providerAccountId String
31   refresh_token String?
32   access_token String?
33   expires_at  Int?
34   token_type  String!
35   scope       String?
36   id_token    String?
37   session_state String?
38
39   createdAt DateTime @default(now())
40   updatedAt DateTime @updatedAt
41
42   user User @relation(fields: [userId], references: [id], onDelete: Cascade)
43
44   @@id([provider, providerAccountId])
45 }
46
47 model Billboard {
48   id          String @id @default(cuid())
49   title       String
50   description String
51   image       String
52   imageUrl    String
53 }
54
55 model SpecialEventComment {
56   id          String @id @default(cuid())
57   name        String
58   comment     String
59
60   eventId String
61   event   SpecialEvent @relation(fields: [eventId], references: [id], onDelete: Cascade)
62
63   createdAt DateTime @default(now())
64   updatedAt DateTime @updatedAt
65 }
66
67 model Event {
68   id          String @id @default(cuid())
69   title       String
70   description String
71   address     String
72   startDate  DateTime
73   endDate    DateTime
74   billboard  String // imagen del evento
75   images     String[]
76   position1  String
77   video1     String
78
79   comments EventComment[]
80
81   createdAt DateTime @default(now())
82   updatedAt DateTime @updatedAt
83 }
84
85 model EventComment {
86   id          String @id @default(cuid())
87   name        String
88   comment     String
89
90   eventId String
91   event   Event @relation(fields: [eventId], references: [id], onDelete: Cascade)
92
93   createdAt DateTime @default(now())
94   updatedAt DateTime @updatedAt
95 }
96
97 model Testimonial {
98   id          String @id @default(cuid())
99   name        String
100  position   String
101  content    String
102  imageUrl   String
103
104  createdAt DateTime @default(now())
105  updatedAt DateTime @updatedAt
106 }
```

Fuente: Autores

### 4.5.1 CockroachDB

Como servicio de base de datos en la nube, CockroachDB fue elegido por su alta disponibilidad, escalabilidad y tolerancia a fallos. Se diseñó un esquema de datos robusto que soporta todas las funcionalidades del sistema, desde la gestión eventos hasta la visualización del contenido

### 4.5.2 Auth.js

Para garantizar un proceso de autenticación seguro, se empleó Auth.js, que facilitó la implementación de funcionalidades como el registro, inicio y cierre de sesión. Asimismo, se

integraron medidas de protección de rutas basadas en roles, asegurando que cada usuario acceda únicamente a las funcionalidades autorizadas

**Figura 24. Configuración de Proveedor de Credenciales en Auth.js**

```
1 import Credentials from "next-auth/providers/credentials"
2 import Google from "next-auth/providers/google"
3 import { NextAuthConfig } from "next-auth"
4 import bcrypt from "bcryptjs"
5
6 import { LoginFormSchema } from "@/schemas/auth"
7 import { getUserByEmail } from "@/actions/user"
8
9 export default {
10   providers: [
11     Google({
12       clientId: process.env.AUTH_GOOGLE_ID,
13       clientSecret: process.env.AUTH_GOOGLE_SECRET,
14       authorization: {
15         params: {
16           prompt: "consent",
17           access_type: "offline",
18           response_type: "code",
19         },
20       },
21     }),
22     Credentials({
23       authorize: async (credentials) => {
24         const result = LoginFormSchema.safeParse(credentials)
25
26         if (result.success) {
27           const { email, password } = result.data
28
29           const user = await getUserByEmail(email)
30
31           if (!user || !user.password) {
32             return null
33           }
34
35           const passwordMatch = await bcrypt.compare(password, user.password)
36
37           if (passwordMatch) return user
38         }
39
40         return null
41       },
42     }),
43   ],
44 }
```

**Fuente: Autores**

#### **4.5.1 Tailwind CSS**

Finalmente, Tailwind CSS se utilizó para estilizar la aplicación, logrando una interfaz visualmente atractiva y completamente responsive. Su sistema de clases utilitarias facilitó la creación de diseños adaptables para dispositivos móviles, tabletas y pantallas de escritorio.

Figura 25. Configuración personalizada de Tailwind CSS

```
1 import type { Config } from "tailwindcss"
2 import defaultTheme from "tailwindcss/defaultTheme"
3
4 const config: Config = {
5   darkMode: ["class"],
6   content: [
7     "./pages/**/*.{js,ts,jsx,tsx,mdx}",
8     "./components/**/*.{js,ts,jsx,tsx,mdx}",
9     "./app/**/*.{js,ts,jsx,tsx,mdx}",
10  ],
11  theme: {
12    extend: {
13      screens: {
14        xs: "480px",
15        ms: "580px",
16        mx: "852px",
17        ...defaultTheme.screens,
18      },
19      colors: {
20        background: "hsl(var(--background))",
21        foreground: "hsl(var(--foreground))",
22        card: {
23          DEFAULT: "hsl(var(--card))",
24          foreground: "hsl(var(--card-foreground))",
25        },
26        popover: {
27          DEFAULT: "hsl(var(--popover))",
28          foreground: "hsl(var(--popover-foreground))",
29        },
30        primary: {
31          DEFAULT: "hsl(var(--primary))",
32          foreground: "hsl(var(--primary-foreground))",
33        },
34        secondary: {
35          DEFAULT: "hsl(var(--secondary))",
36          foreground: "hsl(var(--secondary-foreground))",
37        },
38        muted: {
39          DEFAULT: "hsl(var(--muted))",
40          foreground: "hsl(var(--muted-foreground))",
41        },
42        accent: {
43          DEFAULT: "hsl(var(--accent))",
```

Fuente: Autores

#### 4.6 Diccionario de datos

El diccionario de datos presentado a continuación describe la estructura de la base de datos utilizada en el desarrollo de la plataforma web para la difusión de información sobre arte, cultura y tradición en UNIMINUTO Villavicencio. Este diccionario detalla las entidades (modelos) que componen el sistema, sus atributos, tipos de datos, restricciones y relaciones, proporcionando una visión clara y organizada de cómo se almacenan y gestionan los datos en la base de datos CockroachDB mediante el uso de Prisma como ORM (Object-Relational Mapping). Esta documentación es esencial para comprender la lógica de almacenamiento del contenido cultural y las interacciones del sistema, facilitando tanto el mantenimiento como futuras ampliaciones de la plataforma.

A continuación, se presenta el diccionario de datos con las tablas correspondientes a los modelos definidos en el esquema Prisma, incluyendo sus campos, tipos de datos, y una breve descripción de su propósito dentro del sistema.

**Tabla 8. Modelo User**

<b>Campo</b>	<b>Tipo de Dato</b>	<b>Restricciones</b>	<b>Descripción</b>
id	String	@id @default(cuid())	Identificador único del usuario, generado automáticamente mediante CUID.
name	String?	Opcional	Nombre del usuario, puede ser nulo.
email	String?	@unique, Opcional	Correo electrónico único del usuario, usado para autenticación.
emailVerified	DateTime?	Opcional	Fecha y hora de verificación del correo electrónico.

<b>Campo</b>	<b>Tipo de Dato</b>	<b>Restricciones</b>	<b>Descripción</b>
password	String?	Opcional	Contraseña del usuario para autenticación local.
image	String?	Opcional	URL de la imagen de perfil del usuario.
createdAt	DateTime	@default(now())	Fecha y hora de creación del registro.
updatedAt	DateTime	@updatedAt	Fecha y hora de la última actualización del registro.
accounts	Account[]	Relación uno a muchos	Lista de cuentas asociadas al usuario (proveedores externos como Google).

**Fuente:** Autores

**Tabla 9. Modelo Account**

<b>Campo</b>	<b>Tipo de Dato</b>	<b>Restricciones</b>	<b>Descripción</b>
userId	String	Requerido	Identificador del usuario asociado a esta cuenta.
type	String	Requerido	Tipo de cuenta (ej. OAuth).
provider	String	Requerido	Proveedor de autenticación (ej. Google).
providerAccountId	String	Requerido	Identificador único de la cuenta en el proveedor externo.

<b>Campo</b>	<b>Tipo de Dato</b>	<b>Restricciones</b>	<b>Descripción</b>
refresh_token	String?	Opcional	Token de actualización para la autenticación.
access_token	String?	Opcional	Token de acceso para la autenticación.
expires_at	Int?	Opcional	Tiempo de expiración del token de acceso (en segundos).
token_type	String?	Opcional	Tipo de token (ej. Bearer).
scope	String?	Opcional	Ámbito de permisos otorgados por el proveedor. ERA
id_token	String?	Opcional	Token de identificación del proveedor.
session_state	String?	Opcional	Estado de la sesión del proveedor.
createdAt	DateTime	@default(now())	Fecha y hora de creación del registro.
updatedAt	DateTime	@updatedAt	Fecha y hora de la última actualización del registro.

**Fuente: Autores**

**Tabla 10. Modelo Billboard**

<b>Campo</b>	<b>Tipo de Dato</b>	<b>Restricciones</b>	<b>Descripción</b>
id	String	@id @default(cuid())	Identificador único del cartel publicitario.

<b>Campo</b>	<b>Tipo de Dato</b>	<b>Restricciones</b>	<b>Descripción</b>
title	String	Requerido	Título del cartel publicitario.
description	String?	Opcional	Descripción del cartel.
image	String	Requerido	URL de la imagen asociada al cartel.
buttonLabel	String	Requerido	Texto del botón de acción del cartel.
href	String	Requerido	Enlace al que dirige el botón del cartel.
createdAt	DateTime	@default(now())	Fecha y hora de creación del registro.
updatedAt	DateTime	@updatedAt	Fecha y hora de la última actualización del registro.

**Fuente:** Autores

**Tabla 11. Modelo AboutUs**

<b>Campo</b>	<b>Tipo de Dato</b>	<b>Restricciones</b>	<b>Descripción</b>
id	String	@id @default(cuid())	Identificador único de la sección "Sobre Nosotros".
title	String	Requerido	Título de la sección.
description	String	Requerido	Descripción de la sección.
image	String	Requerido	URL de la imagen asociada a la sección.

<b>Campo</b>	<b>Tipo de Dato</b>	<b>Restricciones</b>	<b>Descripción</b>
reverse	Boolean	Requerido	Indica si el diseño de la sección debe invertirse.
createdAt	DateTime	@default(now())	Fecha y hora de creación del registro.
updatedAt	DateTime	@updatedAt	Fecha y hora de la última actualización del registro.

**Fuente: Autores**

**Tabla 12. Modelo Gallery**

<b>Campo</b>	<b>Tipo de Dato</b>	<b>Restricciones</b>	<b>Descripción</b>
id	String	@id @default(cuid())	Identificador único de la imagen en la galería.
imageUrl	String	Requerido	URL de la imagen almacenada en la galería.
createdAt	DateTime	@default(now())	Fecha y hora de creación del registro.
updatedAt	DateTime	@updatedAt	Fecha y hora de la última actualización del registro.

**Fuente: Autores**

**Tabla 13. Modelo News**

<b>Campo</b>	<b>Tipo de Dato</b>	<b>Restricciones</b>	<b>Descripción</b>
id	String	@id @default(cuid())	Identificador único de la noticia.
title	String	Requerido	Título de la noticia.
description	String	Requerido	Descripción de la noticia.
imageUrl	String	Requerido	URL de la imagen asociada a la noticia.
link	String	Requerido	Enlace externo relacionado con la noticia.
createdAt	DateTime	@default(now())	Fecha y hora de creación del registro.
updatedAt	DateTime	@updatedAt	Fecha y hora de la última actualización del registro.

**Fuente:** Autores

**Tabla 14. Modelo Article**

<b>Campo</b>	<b>Tipo de Dato</b>	<b>Restricciones</b>	<b>Descripción</b>
id	String	@id @default(cuid())	Identificador único del artículo.
title	String	Requerido	Título del artículo.
description	String	Requerido	Contenido o descripción del artículo.
image	String	Requerido	URL de la imagen asociada al artículo.

reverse	Boolean	Requerido	Indica si el diseño del artículo debe invertirse (izquierda/derecha).
createdAt	DateTime	@default(now())	Fecha y hora de creación del registro.
updatedAt	DateTime	@updatedAt	Fecha y hora de la última actualización del registro.

**Fuente: Autores**

**Tabla 15. Modelo SpecialEvent**

<b>Campo</b>	<b>Tipo de Dato</b>	<b>Restricciones</b>	<b>Descripción</b>
id	String	@id @default(cuid())	Identificador único del evento especial.
title	String	Requerido	Título del evento especial.
description	String	Requerido	Descripción del evento especial.
address	String	Requerido	Dirección o ubicación del evento.
startDate	DateTime	Requerido	Fecha y hora de inicio del evento.
endDate	DateTime	Requerido	Fecha y hora de finalización del evento.
billboard	String	Requerido	URL de la imagen principal del evento.
images	String[]	Requerido	Lista de URLs de imágenes adicionales del evento.

podcastUrl	String	Requerido	URL del podcast asociado al evento.
videoUrl	String	Requerido	URL del video asociado al evento.
createdAt	DateTime	@default(now())	Fecha y hora de creación del registro.
updatedAt	DateTime	@updatedAt	Fecha y hora de la última actualización del registro.
comments	SpecialEventComment[]	Relación uno a muchos	Lista de comentarios asociados al evento especial.

**Fuente:** Autores

**Tabla 16. Modelo SpecialEventComment**

<b>Campo</b>	<b>Tipo de Dato</b>	<b>Restricciones</b>	<b>Descripción</b>
id	String	@id @default(cuid())	Identificador único del comentario.
name	String	Requerido	Nombre del usuario que dejó el comentario.
comment	String	Requerido	Contenido del comentario.
eventId	String	Requerido	Identificador del evento especial asociado.
createdAt	DateTime	@default(now())	Fecha y hora de creación del registro.

updatedAt	DateTime	@updatedAt	Fecha y hora de la última actualización del registro.
-----------	----------	------------	---

**Fuente: Autores**

**Tabla 17 Modelo Event**

<b>Campo</b>	<b>Tipo de Dato</b>	<b>Restricciones</b>	<b>Descripción</b>
id	String	@id @default(cuid())	Identificador único del evento.
title	String	Requerido	Título del evento.
description	String	Requerido	Descripción del evento.
address	String	Requerido	Dirección o ubicación del evento.
startDate	DateTime	Requerido	Fecha y hora de inicio del evento.
endDate	DateTime	Requerido	Fecha y hora de finalización del evento.
billboard	String	Requerido	URL de la imagen principal del evento.
images	String[]	Requerido	Lista de URLs de imágenes adicionales del evento.
podcastUrl	String	Requerido	URL del podcast asociado al evento.
videoUrl	String	Requerido	URL del video asociado al evento.
createdAt	DateTime	@default(now())	Fecha y hora de creación del registro.
updatedAt	DateTime	@updatedAt	Fecha y hora de la última actualización del registro.

comments	EventComment[]	Relación uno a muchos	Lista de comentarios asociados al evento.
----------	----------------	-----------------------	---

Fuente: Autores

**Tabla 18. Modelo EventComment**

Campo	Tipo de Dato	Restricciones	Descripción
id	String	@id @default(cuid())	Identificador único del comentario.
name	String	Requerido	Nombre del usuario que dejó el comentario.
comment	String	Requerido	Contenido del comentario.
eventId	String	Requerido	Identificador del evento asociado.
createdAt	DateTime	@default(now())	Fecha y hora de creación del registro.
updatedAt	DateTime	@updatedAt	Fecha y hora de la última actualización del registro.

Fuente: Autores

**Tabla 19. Modelo Testimonials**

Campo	Tipo de Dato	Restricciones	Descripción
id	String	@id @default(cuid())	Identificador único del testimonio.
name	String	Requerido	Nombre de la persona que da el testimonio.

position	String	Requerido	Cargo o posición de la persona (ej. estudiante, docente).
content	String	Requerido	Contenido del testimonio.
imageUrl	String	Requerido	URL de la imagen asociada al testimonio.
createdAt	DateTime	@default(now())	Fecha y hora de creación del registro.
updatedAt	DateTime	@updatedAt	Fecha y hora de la última actualización del registro.

**Fuente: Autores**

#### 4.6.1 Relaciones

Las relaciones en el esquema Prisma definen la estructura relacional de la base de datos de la plataforma de manera eficiente.

El modelo User establece una relación de uno a muchos con Account, permitiendo que un usuario tenga múltiples cuentas asociadas (por ejemplo, autenticaciones con Google), las cuales se vinculan mediante el campo userId de Account que referencia el id de User. A su vez, Account tiene una relación de muchos a uno con User, indicando que cada cuenta pertenece a un único usuario.

Por otra parte, el modelo SpecialEvent mantiene una relación de uno a muchos con SpecialEventComment, lo que significa que un evento especial puede tener varios comentarios asociados, conectados a través del campo eventId de SpecialEventComment que apunta al id de SpecialEvent.

De forma similar, Event presenta una relación de uno a muchos con EventComment, permitiendo que un evento albergue múltiples comentarios, ligados por el campo eventId de EventComment que referencia el id de Event.

Los modelos restantes, como Billboard, AboutUs, Gallery, News, Article y Testimonials, no tienen relaciones directas con otros modelos, funcionando como entidades independientes diseñadas para gestionar contenido específico de la plataforma, como carteles publicitarios, secciones informativas, imágenes, noticias, artículos y testimonios.

Estas relaciones garantizan la integridad de los datos y soportan funcionalidades clave del sistema, como la autenticación de usuarios, la gestión de eventos culturales y la interacción mediante comentarios.

#### **4.7 Plan de pruebas**

El plan de pruebas es una herramienta fundamental para garantizar que el aplicativo web funcione correctamente y cumpla con los objetivos planteados, y así garantizar la mejor atención y así dar una mejor confianza a las personas.

**Tabla 20. Plan de pruebas**

<b>Módulo de prueba</b>	<b>Objetivo de la prueba</b>	<b>Responsable de la prueba</b>	<b>Resultado de la prueba</b>	<b>Resultados esperados</b>
Registro de Usuarios	Comprobar el correcto funcionamiento del módulo	Nicolás Alfonso Manrique Martinez	- Intento 1: Validó correctamente que todos los campos fueran llenados por el usuario  - Intento 2: Se realizó el registro exitosamente y redirigió a la ruta correspondiente	Que se pudiera registrar un usuario por primera vez correctamente y su rol por defecto fuera "admin"
Autenticación de Administrador	Verificar la seguridad del acceso al CMS	Santiago Andrés Suescun Beltran	- Intento 1: Acceso exitoso con credenciales válidas  - Intento 2: Bloqueó intentos con credenciales inválidas y mostró mensaje de error	Que solo usuarios con rol "admin" accedieran al CMS y se bloquearan intentos no autorizados

<b>Módulo de prueba</b>	<b>Objetivo de la prueba</b>	<b>Responsable de la prueba</b>	<b>Resultado de la prueba</b>	<b>Resultados esperados</b>
Gestión de Eventos	Validar la creación y visualización de eventos	Nicolás Alfonso Manrique Martinez	- I Intento 1: Creó un evento con texto, imagen y video, visible en la página pública  - Intento 2: Editó y eliminó el evento sin errores	Que el administrador pudiera crear eventos y estos se mostraran actualizados en la página pública
Carga de Contenido Multimedia	Evaluar la subida y reproducción de archivos	Santiago Andrés Suescun Beltran	- Intento 1: Subió imagen y podcast exitosamente, reproducibles en la interfaz  - Intento 2: Detectó error al subir archivo mayor a 50 MB y mostró advertencia	Que se pudieran subir y reproducir diversos formatos multimedia sin errores
Visualización Pública	Confirmar accesibilidad del contenido sin login	Nicolás Alfonso Manrique Martínez	- Intento 1: Mostró eventos y noticias en dispositivos móviles y de escritorio sin retrasos	Que los visitantes accedieran al contenido

<b>Módulo de prueba</b>	<b>Objetivo de la prueba</b>	<b>Responsable de la prueba</b>	<b>Resultado de la prueba</b>	<b>Resultados esperados</b>
			- Intento 2: Los comentarios se mostraron ordenados y visibles tras ser enviados	fácilmente desde cualquier dispositivo

**Fuente: Autores**

## 5 CAPITULO VI

### 5.1 Conclusiones

El proyecto desarrolló con éxito una plataforma web con un Sistema de Gestión de Contenidos (CMS) que centraliza y difunde recursos culturales, artísticos y tradicionales de UNIMINUTO Villavicencio, cumpliendo los objetivos establecidos. Mediante cinco entrevistas no estructuradas con María Isabel Rodríguez Leyva, mesas de trabajo y la definición de historias de usuario, se identificaron y priorizaron requerimientos funcionales y no funcionales en Trello, garantizando una solución alineada con las necesidades del Centro de Arte, Cultura y Tradición de la Orinoquía (CACTO).

La optimización fue un pilar clave del proyecto en múltiples niveles. En el diseño de la interfaz, creado en Figma, se priorizó la usabilidad y accesibilidad con mockups como la página de inicio de sesión y el carrusel de eventos, logrando tiempos de carga inferiores a 3 segundos, compatibilidad multiplataforma (móviles, tabletas y escritorios) y una experiencia de usuario intuitiva, validada en pruebas de visualización pública. Técnicamente, la implementación con React, Next.js, Prisma ORM y CockroachDB optimizó la gestión de datos y contenidos multimedia, soportando operaciones CRUD eficientes y acceso concurrente de múltiples usuarios. La arquitectura backend, basada en Prisma y CockroachDB, aseguró escalabilidad y tolerancia a fallos, mientras que el uso de Uploadthing optimizó la carga y almacenamiento de archivos multimedia, reduciendo latencias y mejorando el rendimiento.

La metodología ágil Scrum, aplicada en sprints de dos semanas, optimizó el proceso de desarrollo al fomentar la colaboración con stakeholders, la priorización dinámica de tareas en

Trello y la mejora continua mediante revisiones y retrospectivas. Las pruebas funcionales y de seguridad, incluyendo autenticación y gestión de eventos, garantizaron un sistema robusto, seguro y confiable, con bloqueo de accesos no autorizados y manejo eficiente de errores. Esta solución fortaleció la proyección social de UNIMINUTO, promoviendo la preservación del patrimonio cultural de la Orinoquía, y estableció una base sólida para futuras mejoras, como la incorporación de soporte multilingüe, analíticas avanzadas y optimizaciones adicionales en la experiencia de usuario.

## 5.2 Resumen Analítico Especializado – RAE

Título	Plataforma Web Para la Difusión de Información Sobre Arte, Cultura y Tradición en UNIMINUTO
Autores	Nicolás Alfonso Manrique Martínez Santiago Andrés Suescun Beltrán
Fecha	12 mayo 2025
Palabras clave	Plataforma web, difusión cultural, UNIMINUTO Orinoquia, Sistema de Gestión de Contenidos (CMS), Scrum, React, Prisma, accesibilidad, contenido multimedia, proyección social
Descripción	Este proyecto desarrolló una plataforma web basada en un Sistema de Gestión de Contenidos (CMS) para centralizar y difundir recursos culturales, artísticos y tradicionales de UNIMINUTO Orinoquia. Se emplearon tecnologías modernas como HTML5, CSS3, JavaScript, React, Next.js, Prisma y CockroachDB, junto con la metodología ágil Scrum, para garantizar un desarrollo eficiente y alineado con las necesidades de la institución. El proceso incluyó la identificación de requerimientos mediante entrevistas, mesas de trabajo y historias de usuario, el diseño de interfaces en Figma, y la implementación de una solución accesible que organiza contenido multimedia como eventos, podcasts, videos e imágenes. La plataforma fortalece la proyección social de UNIMINUTO al mejorar el acceso y la apropiación del conocimiento cultural en la región.

<b>Problema</b>	UNIMINUTO Oriente enfrentaba la dispersión de recursos culturales a través de múltiples canales no integrados, como bibliotecas digitales, sitios web y redes sociales, lo que limitaba su accesibilidad y alcance. La falta de una plataforma centralizada dificultaba la difusión eficiente de contenidos artísticos y tradicionales, restringiendo su apropiación por parte de la comunidad universitaria y el público general, y afectando la proyección social de la institución en el ámbito digital.
<b>Objetivo</b>	Desarrollar un sitio web y un Sistema de Gestión de Contenidos (CMS) para administrar, difundir y promover recursos digitales sobre arte, cultura y tradición, dirigido a la comunidad universitaria y al público en general, mediante una interfaz intuitiva, un backend robusto y la metodología Scrum, para optimizar la accesibilidad y centralización del contenido cultural de UNIMINUTO Oriente.
<b>Conclusiones</b>	El proyecto culminó con éxito en la entrega de una plataforma funcional que centralizó los recursos culturales de UNIMINUTO Orinoquia, mejorando su difusión y accesibilidad. La identificación precisa de requerimientos mediante historias de usuario y el uso de Scrum aseguraron un desarrollo alineado con las necesidades institucionales. Las tecnologías implementadas (React, Prisma, Next.js) proporcionaron una solución escalable y eficiente. Se recomienda realizar pruebas de usabilidad continuas, integrar herramientas analíticas y establecer un plan de mantenimiento para garantizar la sostenibilidad y el impacto a largo plazo de la plataforma.

Autor RAE	Nicolás Alfonso Manrique Martínez Santiago Andrés Suescun Beltrán
Fecha creación de RAE	12 de mayo de 2025

## 6 Referencias

- Ann Rockley, C. C. (february de 2020). *Managing Enterprise Content: A Unified Content Strategy, Second Edition*. Obtenido de <https://www.oreilly.com/library/view/managing-enterprise-content/9780132931656/>
- CABRERA PEÑA, K. (12 de february de 2024). *Cultura y derechos de propiedad intelectual en la era digital*. *Revista de Derecho*. Obtenido de <https://doi.org/10.14482/dere.60.190.252>
- chile, B. n. (s.f.). *BIBLIOTECA NACIONAL DIGITAL, MEMORIA CHILENA, CHILE PARA NIÑOS*. Obtenido de <https://www.bibliotecanacional.gob.cl/sites/www.bibliotecanacional.gob.cl/files/2024-10/Memoria%20BND.pdf>
- Colombia, E. C. (01 de 2012). Acto Legislativo. Artículo 1°.
- Domínguez, M. H. (2023). *El Museo del Prado como ecosistema cultural*. Junta de Andalucía: Published by Consejería de Turismo, Cultura y Deporte.
- Foundation, E. (2021). *Europeana: We transform the world with culture*. Obtenido de <https://pro.europeana.eu/>
- Freeman, A. (2023). *Pro ASP.NET Core 7*. Manning Publications.
- Google. (2023). *Google Arts & Culture Impact Report*. Obtenido de <https://www.casaculturameta.gov.co/clone/wp-admin/customize.php?url=https%3A%2F%2Fwww.casaculturameta.gov.co%2Fclone%2F>
- F
- Gutenberg., P. (2024). *About Project Gutenberg*. Obtenido de <https://www.gutenberg.org/about/>
- India, S. o. (15 de junio de 2024). *Sounds of India*. Obtenido de <https://www.soundsofindia.org/>

Institution, S. (2023). *Smithsonian Open Access Usability Study*. Washington, DC: Smithsonian.

Obtenido de <https://www.si.edu/openaccess>

Janne Berg jaberg, J. L. (2021). *How do we know that it works? Designing a digital democratic innovation with the help of user-centered design*. Obtenido de

<https://journals.sagepub.com/doi/full/10.3233/IP-200282>

MINISTERIO. (2023). *Informe de gestión 2023 del ministerio de las culturas, las artes y los saberes*. Obtenido de [https://www.mincultura.gov.co/transparencia/Documents/4-planeacion-presupuesto-e-informes/informe-de-](https://www.mincultura.gov.co/transparencia/Documents/4-planeacion-presupuesto-e-informes/informe-de-gestion/INFORME%20DE%20GESTI%C3%93N%202023%20MINISTERIO%20DE%20LAS%20CULTURAS%20LAS%20ARTES%20Y%20LOS%20SABERES.pdf)

[gestion/INFORME%20DE%20GESTI%C3%93N%202023%20MINISTERIO%20DE%20LAS%20CULTURAS%20LAS%20ARTES%20Y%20LOS%20SABERES.pdf](https://www.mincultura.gov.co/transparencia/Documents/4-planeacion-presupuesto-e-informes/informe-de-gestion/INFORME%20DE%20GESTI%C3%93N%202023%20MINISTERIO%20DE%20LAS%20CULTURAS%20LAS%20ARTES%20Y%20LOS%20SABERES.pdf)

MNAC. (2023). *Museu Nacional d'Art de Catalunya (MNAC)*. Obtenido de

<https://www.museunacional.cat>

Norman, D. (2022). *The Design of Everyday Things (2nd ed.)*. Basic Books. editora del libro.

Obtenido de <https://mitpress.mit.edu/9780262525671/>

Osmani, A. (2022). *Learning JavaScript Design Patterns*. Obtenido de

<https://books.google.com.uy/books?hl=es&lr=&id=L46fX62D5qYC&oi=fnd&pg=PR3&dq=.+Learning+JavaScript+Frameworks.+Sebastopol:+O%E2%80%99Reilly+Media&ots=Tqu04jXSab&sig=vwFgOkb4XPbVIBgpfyqBid4nie8#v=onepage&q&f=false>

Pico Macías, M. E. (2023). *La proyección social de las instituciones de educación superior y el fortalecimiento del liderazgo comunitario*. Obtenido de Revista EDUCARE - UPEL-IPB

- Segunda Nueva Etapa 2.0: <https://doi.org/10.46498/reduipb.v26iExtraordinario.1675>

Richards, M. (2017). *Software Architecture Patterns*. O'Reilly Media.

Serrano-Cobos, J. (2021). *Evolución de los sistemas de gestión de contenidos (CMS). Del mainframe al open source. Profesional De La información*. Obtenido de <https://doi.org/10.3145/epi.2007.may.05>

Stair, R. P. (2024). *Principles of Information Systems (Mindtap Course List)*.

UNAM, U. N. (2025). *Cultura UNAM*. Obtenido de <https://cultura.unam.mx/>

UNIMINUTO. (2021). *Corporación Universitaria Minuto de Dios*. Obtenido de Modelo Educativo UNIMINUTO.

Viva, U. N. (2023). *UNAL*. Obtenido de

[http://planeacion.bogota.unal.edu.co/boletin/Noticias/Agosto/Museo\\_de\\_Arte\\_Cocrea/](http://planeacion.bogota.unal.edu.co/boletin/Noticias/Agosto/Museo_de_Arte_Cocrea/)

## 7 Anexos

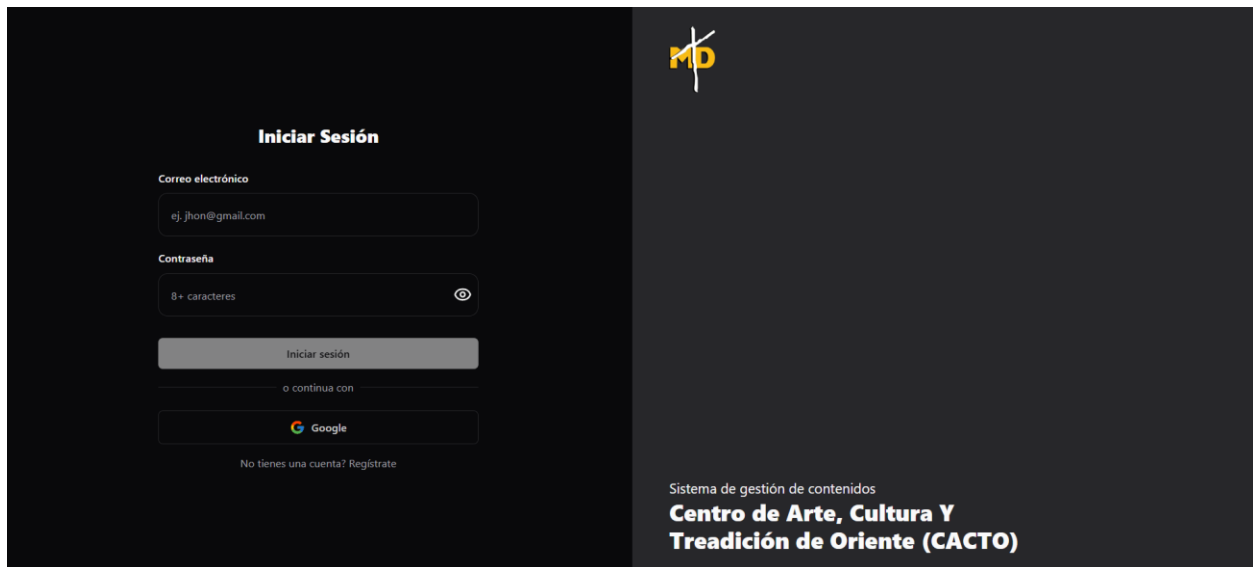
### 7.1 Manual de Usuario

#### 7.1.1 Iniciar Sesión en el CMS

El primer paso para acceder al CMS y gestionar el contenido es iniciar sesión. Sigue estos pasos:

Accede a la página de inicio de sesión: Abre tu navegador y dirígete a la URL del CMS del CACTO. Verás una pantalla como la siguiente:

**Figura 26. inicio de sesión**



**Fuente: Autores**

**Ingresa tus credenciales:**

- En el campo Correo electrónico, escribe: **test@gmail.com**.
- En el campo Contraseña, escribe: **Test123**.

Asegúrate de que la contraseña tenga al menos 8 caracteres y respete las mayúsculas y minúsculas.

### **Inicia sesión**

Haz clic en el botón Iniciar sesión. Si las credenciales son correctas, serás redirigido al panel principal del CMS, donde podrás gestionar el contenido del Centro de Arte, Cultura y Tradición de Oriente (CACTO).

### **Alternativa**

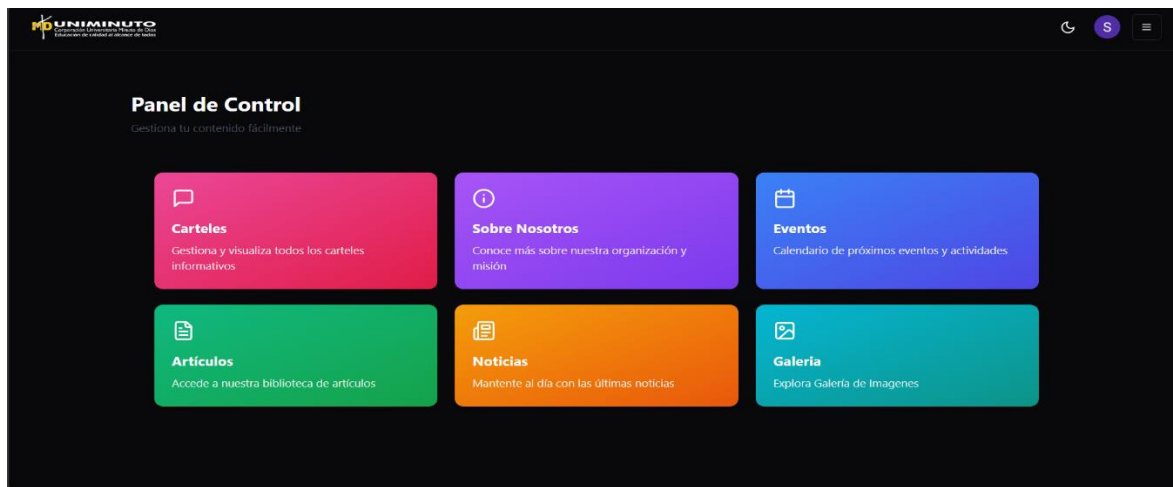
Si prefieres, puedes usar el botón Continuar con Google para iniciar sesión con tu cuenta de Google, siempre que esté vinculada al sistema.

#### **7.1.2 Navegación en el Panel de Control**

Una vez que hayas iniciado sesión en el CMS del Centro de Arte, Cultura y Tradición de Oriente (CACTO), serás redirigido al **Panel de Control**, la interfaz principal desde la cual podrás gestionar todo el contenido. Este panel está diseñado para ser intuitivo y facilitar el acceso a las diferentes secciones de la aplicación. A continuación, se detalla cómo navegar y utilizar las opciones disponibles:

1. **Vista General del Panel de Control:**
2. Al ingresar, verás una pantalla como la siguiente:

**Figura 27. Panel de control**



**Fuente: Autores**

### **Opciones de Navegación:**

El Panel de Control cuenta con varios botones que te permiten acceder a secciones específicas de la aplicación. Cada botón está identificado con un nombre y una breve descripción para guiarte en la gestión del contenido. Las opciones disponibles son:

- **Carteles:** Haz clic en este botón para gestionar y visualizar todos los carteles informativos. Aquí podrás crear, editar o eliminar carteles según las necesidades del CACTO.
- **Sobre Nosotros:** Accede a esta sección para modificar la información relacionada con la organización y su misión.
- **Eventos:** Administra el calendario de próximos eventos y actividades, permitiéndote agregar, editar o eliminar eventos.
- **Artículos:** Dirígete a esta sección para acceder a la biblioteca de artículos y gestionar su contenido.
- **Noticias:** Mantén actualizada la sección de noticias con las últimas novedades del CACTO.
- **Galería:** Explora y administra la galería de imágenes, donde puedes subir, organizar o eliminar fotos.

#### **Acceso a las secciones:**

Para navegar, simplemente haz clic en el botón correspondiente a la sección que deseas gestionar. Por ejemplo, si deseas crear un nuevo cartel, selecciona el botón **Carteles**, y serás redirigido a una página donde podrás agregar un nuevo cartel con la información necesaria.

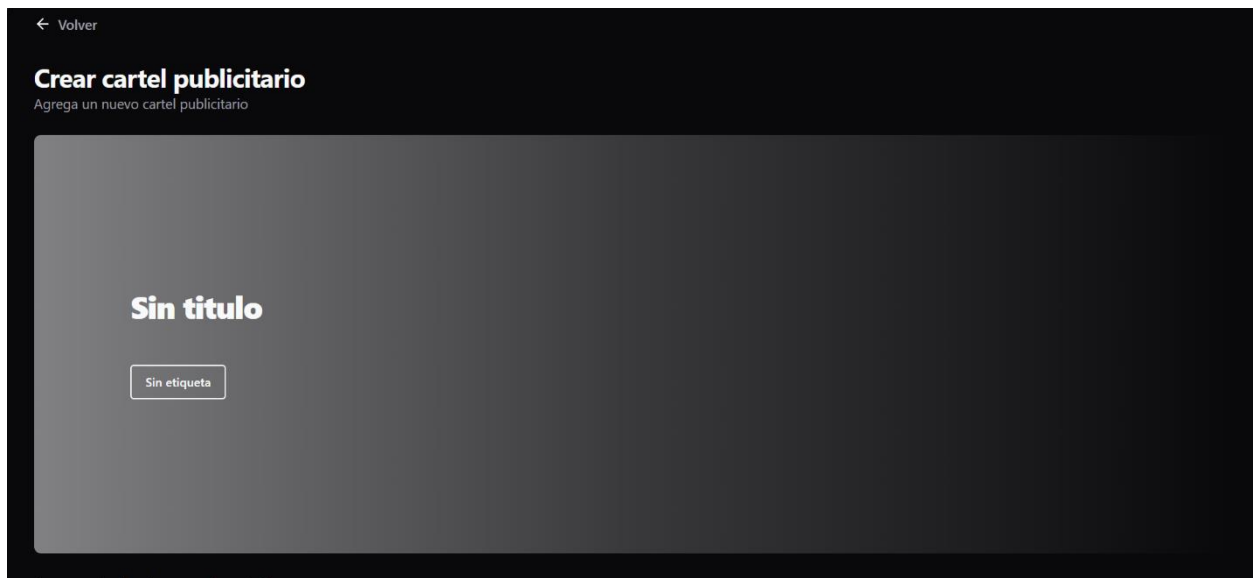
#### **7.1.3 Crear un Cartel Publicitario**

El CMS del Centro de Arte, Cultura y Tradición de Oriente (CACTO) permite crear carteles publicitarios para promocionar eventos o actividades. En esta sección, podrás subir una imagen y agregar un enlace que redirija a los usuarios a más información sobre el evento. A continuación, se detalla el proceso para crear un nuevo cartel:

### **Accede a la sección de carteles:**

Desde el Panel de Control, haz clic en el botón Carteles. Esto te llevará a la sección donde puedes gestionar los carteles publicitarios.

### **Figura 28 .Cartel publicitario**



**Fuente: Autores**

## **7.1.4 Crear un Evento**

El CMS del Centro de Arte, Cultura y Tradición de Oriente (CACTO) permite gestionar eventos de manera detallada, incluyendo la posibilidad de agregar información general, imágenes, enlaces a videos promocionales y podcasts. A continuación, se detalla el proceso para crear un nuevo evento

### **Accede a la Sección de Eventos:**

Desde el **Panel de Control**, haz clic en el botón **Eventos**. Esto te llevará a la sección donde puedes gestionar el calendario de eventos y actividades.

### **Inicia la Creación de un Nuevo Evento:**

Una vez en la sección de eventos, selecciona la opción para crear un nuevo evento. Serás redirigido a una pantalla como la siguiente:

**Figura 29. Creación de eventos**

El formulario de creación de eventos está dividido en varias secciones:

- Información General:** Incluye tres campos de texto: "Título \*" (con el placeholder "Título del evento"), "Descripción \*" (con el placeholder "Descripción del evento") y "Dirección del evento \*" (con el placeholder "Dirección del evento").
- Fecha de inicio \*:** Un campo de fecha con el formato "MM/dd/yyyy hh:mm aa" y un ícono de calendario.
- Fecha de finalización \*:** Un campo de fecha con el formato "MM/dd/yyyy hh:mm aa" y un ícono de calendario.
- Recursos Multimedia:** Incluye una sección "Imágenes del evento" con un ícono de una cámara y un signo más (+) para agregar imágenes.
- URL del video:** Un campo de texto con el placeholder "Enlace del video promocional".
- Podcast:** Un botón con un ícono de archivo y el texto "Selecciona un archivo".

**Fuente: Autores**