

Desarrollo de una Red Social Colaborativa para Estudiantes y Profesionales

por:

Heider Miguel González Quintero

Sergio Andres Perez Peñaloza

Docente:

Nando González

Barranquilla

Corporación universitaria reformada

2026

Desarrollo de una Red Social Colaborativa para Estudiantes y Profesionales

Heider Miguel González Quintero

Sergio Andres Perez Peñaloza

Trabajo de Grado presentado como requisito para optar al título de (Ingeniero informático)

Profesor Hernando González



Programa de ingeniería informática, Facultad de ingeniería

Universidad Reformada

Barranquilla, Atlántico

2026

RESUMEN

El presente trabajo de grado expone el desarrollo de **HIDEON**, una plataforma web de red social colaborativa diseñada para conectar estudiantes y profesionales, facilitando la formación y gestión de equipos en proyectos académicos y productivos. La propuesta responde a la fragmentación de las herramientas digitales actuales, la cual dificulta el tránsito continuo entre la ideación, la búsqueda de colaboradores, la comunicación y la ejecución.

La arquitectura del sistema se construyó sobre un *stack* moderno integrado por **Recato, TypeScript, Supabase y PostgreSQL**, complementado con **Supabase Realtime y WebSockets** para la comunicación síncrona, **Cloudflare R2** para la gestión de archivos y **Render.com** para el despliegue. El proceso de desarrollo adoptó la metodología ágil *Extreme Programming* (XP), incorporando prácticas de integración continua, pruebas de seguridad y validación funcional.

HIDEON integra funcionalidades de autenticación, perfiles de usuario, publicación de ideas, estructuración de equipos, mensajería en tiempo real y la conversión de propuestas iniciales en proyectos consolidados. Para salvaguardar la privacidad y garantizar un acceso controlado a los recursos, se implementaron políticas de *Row Level Security* (RLS).

Los resultados validaron la viabilidad técnica y operativa de la plataforma: durante la fase piloto se alcanzaron **151 usuarios registrados** y la comunicación en tiempo real mantuvo latencias inferiores a **150 ms**. Asimismo, la arquitectura demostró alta escalabilidad, seguridad robusta y eficiencia en costos. En conclusión, HIDEON se consolida como una solución tecnológica eficaz para impulsar la colaboración interdisciplinaria y reducir la brecha entre el ámbito académico y el sector profesional.

Palabras clave: Red social colaborativa, gestión de proyectos, comunicación en tiempo real, React, Supabase, PostgreSQL, WebSockets, seguridad de datos, escalabilidad.

ABSTRACT

This degree project presents **HIDEON**, a collaborative social networking platform designed to connect students and professionals, streamlining team formation and management for academic and professional projects. The proposal addresses the fragmentation of current digital tools, which impairs the seamless transition from idea generation to collaborator identification, communication, and project execution.

The system's architecture relies on a modern stack composed of **React, TypeScript, Supabase, and PostgreSQL**, complemented by **Supabase Realtime and WebSockets** for synchronous communication, **Cloudflare R2** for file storage, and **Render.com** for deployment. The development process followed the *Extreme Programming* (XP) agile methodology, incorporating continuous integration practices, security testing, and functional validation.

HIDEON features user authentication, profile management, idea publication, team structuring, real-time messaging, and the transformation of initial concepts into consolidated projects. Furthermore, Row Level Security (RLS) policies were implemented to protect user data and ensure strict access control.

The results confirmed the technical and functional feasibility of the platform. During the pilot phase, HIDEON reached **151 registered users** and achieved response times below **150 ms** for real-time interactions. Additionally, the implemented architecture enhanced system scalability, security, and cost efficiency. In conclusion, HIDEON serves as an effective technological solution to foster interdisciplinary collaboration and bridge the gap between academic environments and the professional sector.

Keywords: Collaborative social network, project management, real-time communication, React, Supabase, PostgreSQL, WebSockets, data security, scalability.

Tabla de contenido

Resumen

Abstract

CAPÍTULO 1: INTRODUCCIÓN Y PLANTEAMIENTO 5

CONTENIDO 5

Introducción 5

Planteamiento del problema 7

Pregunta de Investigación 8

Justificación.....9

Viabilidad del Proyecto 10

Objetivos 11

Objetivo general 11

Objetivos específicos.....11

CAPÍTULO 2: MARCO REFERENCIAL 12

Marco de referencial.....12

Estado del arte 12

Marco legal.....14

1. Protección de Datos Personales.....14

2. Propiedad Intelectual.....15

3. Normativas Complementarias 15

Marco teórico 17

Redes sociales 17

Redes Colaborativas 18

Fundamentos técnicos	20
CAPÍTULO 3: METODOLOGÍA.....	21
Metodología	21
Desarrollo y configuración del sistema	23
Metodología de Desarrollo.....	23
Tecnologías Utilizadas	24
CAPÍTULO 4: DESARROLLO E INGENIERÍA DEL SISTEMA.....	25
Arquitectura General del Sistema	25
Sistema de Autenticación y Gestión de Identidades.....	27
Base de Datos y Almacenamiento Estructurado	28
Optimización de Transferencia de Datos (Egress Costs)	31
Despliegue del Sistema y Estrategia Keep-Alive	32
Flujo Funcional del Sistema.....	34
Módulo de Mensajería y Colaboración Síncrona (Tiempo Real).....	36
Plan de Pruebas y Validación del Sistema	38
Conclusiones del Capítulo.....	39
CAPÍTULO 5: RESULTADOS, CONCLUSIONES Y RECOMENDACIONES.....	39
Evaluación de Escalabilidad e Infraestructura Cloud (Supabase Pro)	40
Pruebas de Aceptación y Validación con Usuarios Reales	40
Conclusiones	41
RECOMENDACIONES	42
REFERENCIAS	43

CAPÍTULO 1: INTRODUCCIÓN Y PLANTEAMIENTO

CONTENIDO

Introducción

El presente desarrollo técnico surge como una respuesta estratégica a la fragmentación crítica identificada en el ecosistema digital actual, donde la transición entre la formación académica y el ejercicio profesional carece de canales integrados y eficientes. En el contexto colombiano, y con un enfoque inicial en la región Caribe, se ha detectado que los estudiantes universitarios, particularmente en áreas de ingeniería, poseen un alto potencial de innovación que a menudo permanece invisible para el sector productivo debido a la falta de plataformas que validen proyectos en etapas tempranas. A diferencia de las soluciones generalistas existentes, esta propuesta se materializa en una aplicación web funcional donde profesionales y estudiantes pueden visualizar ideas en etapa conceptual, publicar proyectos académicos, identificar colaboradores con habilidades e intereses complementarios, y establecer conexiones significativas mediante herramientas de comunicación integradas. El proyecto implementa funcionalidades esenciales de interacción social, gestión de proyectos colaborativos y comunicación en tiempo real, con fundamentación teórica sólida y desarrollo técnico que demuestre la viabilidad práctica de la propuesta.

A diferencia de un Producto Mínimo Viable (MVP) convencional orientado únicamente a la validación de mercado, este proyecto se enfoca en la construcción de una arquitectura de software escalable y robusta. El objetivo primordial es el diseño e implementación de una red

social colaborativa especializada que trascienda las funciones de una red de contactos tradicional, permitiendo a los usuarios gestionar el ciclo de vida completo de un proyecto desde su concepción hasta la formación de equipos operativos.

La integración de este tipo de herramientas responde a la necesidad de fomentar el aprendizaje colaborativo, el cual, según (Johnson et al., 1999, #), no solo mejora el rendimiento académico, sino que también desarrolla competencias sociales críticas para el entorno profesional. El proyecto implementa funcionalidades esenciales de interacción social, gestión de proyectos colaborativos y comunicación en tiempo real, utilizando un stack tecnológico moderno. Al respecto, el uso de arquitecturas Backend as a Service (BaaS) como Supabase permite optimizar los tiempos de desarrollo y centrar los esfuerzos en la experiencia del usuario (Garsiel, 2022).

Esta fundamentación técnica demuestra la viabilidad práctica de la propuesta en el marco temporal de un trabajo de grado.

La estructura del presente documento se organiza de la siguiente manera: en el capítulo dos se presenta el planteamiento del problema, describiendo en detalle las brechas identificadas en el ecosistema actual de colaboración digital y formulando la pregunta de investigación que guía este trabajo. El capítulo tres desarrolla la justificación del proyecto, argumentando su relevancia académica, profesional, social y tecnológica. En el capítulo cuatro se establecen el objetivo general y los objetivos específicos. El capítulo cinco presenta el marco referencial, que incluye el estado del arte, el marco legal y el marco teórico. El capítulo seis detalla la metodología de desarrollo empleada, especificando el tipo de investigación y el stack tecnológico seleccionado.

Finalmente, el capítulo siete presenta los resultados obtenidos, seguidos de las conclusiones y recomendaciones.

Planteamiento del problema

A pesar de la existencia de redes sociales generalistas que facilitan la interacción entre individuos y grupos, persiste una brecha significativa en la formación de equipos de proyecto en sus fases tempranas y en la visibilización del conocimiento técnico generado dentro de las universidades. Estas plataformas, aunque útiles para la socialización y el networking, carecen de funcionalidades específicas que permitan articular de manera efectiva los intereses académicos y profesionales con la gestión integral de proyectos. Como consecuencia, estudiantes y profesionales enfrentan dificultades para identificar colaboradores con habilidades complementarias y objetivos comunes, lo que limita la posibilidad de transformar ideas incipientes en iniciativas concretas con potencial de ejecución.

Ahora bien, la problemática se intensifica debido a la fragmentación de los procesos digitales disponibles. Ya que, actualmente, los usuarios deben recurrir a múltiples herramientas para cubrir las distintas etapas de un proyecto: desde la publicación de ideas, pasando por la validación del talento, hasta la comunicación en tiempo real. Esta dispersión tecnológica genera barreras en la transición fluida entre el descubrimiento de una idea y su materialización a lo real, lo que provoca pérdida de tiempo, recursos, duplicación de esfuerzos y, en muchos casos, abandono de iniciativas prometedoras. En el contexto universitario colombiano, esta situación se traduce en una menor capacidad para capitalizar el conocimiento generado en las aulas y laboratorios, restringiendo su impacto en el entorno social y productivo.

Además, la falta de un entorno especializado limita la proyección profesional de los estudiantes, quienes no encuentran espacios adecuados para mostrar sus competencias en escenarios colaborativos reales. De igual manera, los profesionales se ven privados de acceder a propuestas académicas innovadoras que podrían enriquecer sus proyectos o abrir nuevas líneas de investigación aplicada. Esta des-conexión entre el talento emergente y las necesidades del sector académico y productivo genera un vacío que afecta tanto la innovación como la competitividad del país.

Ante este escenario, surge la necesidad imperativa de desarrollar una solución tecnológica robusta y unificada que supere las limitaciones de las redes sociales tradicionales y responda a las particularidades del entorno universitario. Una plataforma especializada permitiría integrar en un mismo espacio la publicación de proyectos, la identificación de perfiles académicos y profesionales, la validación de competencias y la comunicación en tiempo real, garantizando así un proceso continuo y coherente desde la concepción de la idea hasta su ejecución. Este enfoque práctico y colaborativo no solo fortalecería la formación de equipos de proyecto, sino que también potenciaría la generación de conocimiento aplicado y la vinculación efectiva entre universidad y sociedad. Por lo tanto, el problema de investigación se define a través de la siguiente interrogante:

Pregunta de Investigación

¿Cómo desarrollar una plataforma especializada que permita conectar perfiles académicos y profesionales para la formación y gestión de equipos de proyecto en tiempo real dentro del entorno universitario en Colombia?

Justificación

Al enfocarse en un nicho específico de estudiantes y profesionales, la plataforma propuesta deja de ser un simple repositorio de información para convertirse en una herramienta activa y estratégica que permite gestionar de manera integral el ciclo de vida de proyectos colaborativos en un entorno centralizado. Esta especialización le otorga un valor diferencial frente a las redes sociales generalistas, ya que no se limita a la interacción superficial, sino que promueve la construcción de conocimiento aplicado y la consolidación de equipos con objetivos comunes. En este sentido, la plataforma se convierte en un espacio donde las ideas pueden evolucionar desde su fase conceptual hasta su ejecución, garantizando un proceso continuo y coherente.

Desde el ámbito tecnológico, el proyecto resulta fundamental para ofrecer una arquitectura escalable, segura y de alto rendimiento, características que las redes sociales tradicionales no priorizan en el contexto académico. La implementación de un stack moderno basado en React y Supabase asegura un manejo eficiente de datos complejos y comunicaciones síncronas, lo que permite que la coordinación de equipos ocurra en tiempo real y con alta disponibilidad. Esta infraestructura tecnológica no solo responde a las necesidades actuales de los usuarios, sino que también sienta las bases para la expansión futura de la plataforma, adaptándose a un mayor volumen de proyectos y participantes sin comprometer la calidad del servicio.

El impacto del proyecto se refleja directamente en la optimización de la productividad dentro del ecosistema universitario. Al facilitar la identificación de perfiles complementarios y garantizar la seguridad de la información bajo la normativa colombiana, se fomenta una cultura de innovación constante en la que los estudiantes pueden materializar sus ideas y construir un portafolio profesional verificable. De igual manera, los profesionales encuentran un espacio

confiable para acceder a propuestas académicas innovadoras, lo que fortalece la vinculación entre universidad y sociedad. Esta interacción bidireccional contribuye a la generación de proyectos con alto potencial de ejecución y relevancia social, incrementando la competitividad y la capacidad de respuesta de las instituciones educativas frente a los retos del entorno global.

Finalmente, la justificación del proyecto se sustenta en la necesidad de cerrar la brecha existente entre el talento emergente y las oportunidades de colaboración interdisciplinaria. Al integrar en un mismo entorno la publicación de proyectos, la validación de competencias y la comunicación en tiempo real, la plataforma se convierte en un catalizador de iniciativas académicas y profesionales, promoviendo la formación de equipos sólidos y la transferencia efectiva de conocimiento. En consecuencia, este proyecto no solo responde a una problemática puntual, sino que también aporta a la construcción de un modelo de innovación sostenible dentro del sistema universitario colombiano.

Viabilidad del Proyecto

El desarrollo de esta plataforma se sustenta en una arquitectura escalable y el uso de un stack tecnológico de alto rendimiento, permitiendo que el proyecto trascienda la fase de prototipo para consolidarse como una solución técnica profesional y robusta. Al integrar herramientas de vanguardia como React para el frontend y Supabase como Backend-as-a-Service (BaaS), se optimiza la gestión de bases de datos relacionales, la autenticación persistente y la sincronización de datos en tiempo real mediante WebSockets, eliminando las limitaciones de la infraestructura tradicional. Este enfoque, complementado con la metodología ágil Extreme Programming (XP) y el despliegue en servicios de hosting de alto nivel como Vercel, garantiza un ciclo de desarrollo eficiente con entregas incrementales y estables. De esta manera, el proyecto resulta plenamente

ejecutable dentro del marco temporal y de recursos disponibles, asegurando una plataforma funcional que cumple con los estándares de seguridad y privacidad exigidos por la normativa colombiana para el manejo de información en entornos académicos y profesionales.

Objetivos

Objetivo general

- Desarrollar una plataforma web colaborativa escalable, usando React y Supabase, que permita la formación y gestión de equipos de proyecto en tiempo real para conectar de manera efectiva a estudiantes y profesionales en Colombia.

Objetivos específicos

- Diseñar la arquitectura de software y el modelo de datos relacional en PostgreSQL que soporte la gestión de identidades, ideas y proyectos y el control de membresías de equipo de forma eficiente.
- Implementar una interfaz de usuario dinámica y responsiva con React y Tailwind CSS, integrando funcionalidades de comunicación síncrona mediante WebSockets para permitir la interacción y el chat grupal entre colaboradores en tiempo real.
- Aplicar protocolos de seguridad avanzada y políticas de acceso a nivel de fila (Row Level Security) para garantizar la protección de datos personales y el despliegue de la plataforma en un entorno de producción real bajo un flujo de integración continua..

CAPÍTULO 2: MARCO REFERENCIAL

Marco de referencial

Este capítulo establece los cimientos técnicos, los antecedentes tecnológicos y el marco normativo que sustentan el desarrollo integral de la plataforma. A diferencia de una validación conceptual, este apartado analiza la arquitectura escalable necesaria para soportar redes colaborativas de alto rendimiento, profundiza en los fundamentos de ingeniería del stack seleccionado (React, Supabase y PostgreSQL) y detalla la legislación vigente en Colombia para la gestión automatizada de datos sensibles. El objetivo es proporcionar el sustento teórico y práctico que garantiza la estabilidad, seguridad y eficiencia del sistema en un entorno de producción real.

Estado del arte

El panorama actual de las plataformas de colaboración digital revela una marcada segmentación que dificulta la integración efectiva entre el ecosistema académico y el sector profesional. Según

diversos análisis de la industria (LinkedIn, 2023), aunque las plataformas de networking son el estándar global para redes de contactos, su estructura algorítmica y de diseño está optimizada para perfiles con trayectorias laborales ya consolidadas. Esto genera una brecha técnica que tiende a invisibilizar el potencial de los proyectos de ingeniería en etapa de desarrollo y las competencias técnicas avanzadas de estudiantes que aún no cuentan con una experiencia laboral formal.

En el ámbito del desarrollo de software y la exposición de portafolios, herramientas como GitHub y Behance han transformado la forma en que se documenta el trabajo técnico. No obstante, como señalan Smith & Smith (2021), estas plataformas funcionan predominantemente como repositorios estáticos de resultados finales y carecen de una capa de interacción social fluida o de herramientas de mensajería síncrona integradas que permitan la gestión completa de equipos desde la concepción de una idea. Esta limitación técnica obliga a los usuarios a fragmentar su comunicación en múltiples aplicaciones externas, lo que aumenta la fricción operativa y reduce la trazabilidad en el proceso colaborativo.

Por otro lado, la evolución de las arquitecturas de Backend as a Service (BaaS) ha permitido que soluciones de desarrollo modernas aborden estos problemas de conectividad con una agilidad sin precedentes. De acuerdo con Garsiel (2022), el uso de infraestructuras como Supabase ha democratizado el acceso a funcionalidades de bases de datos relacionales y actualizaciones en tiempo real que antes requerían despliegues masivos de servidores. Este avance permite que el presente desarrollo compita en términos de usabilidad, respuesta instantánea y escalabilidad, integrando servicios de alta disponibilidad para un nicho especializado.

Finalmente, las investigaciones actuales sobre redes sociales de alto rendimiento sugieren que la formación de equipos efectivos en entornos digitales no depende solo de un motor de búsqueda de perfiles, sino de un ecosistema que facilite la validación técnica y la comunicación síncrona (Ries, 2011). Este proyecto se posiciona en el estado del arte como una solución de ingeniería integral que unifica estas necesidades, permitiendo que la transición entre la publicación de una

idea técnica y la consolidación de un equipo de trabajo ocurra en un solo entorno operativo robusto y centralizado.

Marco legal

El desarrollo y operación de esta plataforma en el contexto digital está sujeto a un conjunto de normativas nacionales e internacionales que garantizan la seguridad, la integridad y la privacidad de los usuarios. Estos pilares legales constituyen la base sobre la cual se construye la confianza en el sistema y aseguran que la interacción académica y profesional se realice dentro de un marco ético y jurídico sólido.

1. Protección de Datos Personales

La Ley 1581 de 2012 establece los principios y disposiciones generales para la protección de datos personales en Colombia. En el caso de esta plataforma, el cumplimiento de dicha normativa es esencial, dado que se gestionan perfiles académicos y profesionales que contienen información sensible. Los principios de finalidad, libertad, veracidad, transparencia y seguridad se aplican rigurosamente en el tratamiento de los datos.

- Se implementan mecanismos de consentimiento expreso e informado en el registro de usuarios, garantizando que cada persona conozca y autorice el uso de su información.
- Se establecen flujos de control y auditoría para asegurar que el manejo de datos se ajuste a los estándares del Habeas Data en Colombia, permitiendo a los usuarios ejercer sus derechos de acceso, rectificación, cancelación y oposición.

- Además, se contemplan medidas de seguridad tecnológica como cifrado de información, autenticación multifactor y protocolos de comunicación segura (HTTPS/TLS).

2. Propiedad Intelectual

La plataforma se rige por las normativas de derechos de autor establecidas en la Ley 23 de 1982 y la Ley 1915 de 2018, que regulan la protección de las obras intelectuales en Colombia.

- La visibilización de proyectos e ideas académicas respeta la autoría original de los estudiantes y profesionales participantes, reconociendo su aporte mediante sistemas de atribución y registro.
- Se fomenta un entorno de colaboración ética donde la propiedad intelectual de las innovaciones está protegida frente a usos no autorizados, evitando plagios o apropiaciones indebidas.
- El sistema está diseñado para dar crédito a los creadores de contenido, permitiendo que cada proyecto publicado quede vinculado a sus autores, lo cual fortalece la construcción de un portafolio verificable y con respaldo legal.

3. Normativas Complementarias

Además de las leyes principales, el proyecto se articula con otras disposiciones relevantes:

- Ley 527 de 1999 (Comercio Electrónico): regula la validez jurídica de los mensajes de datos y la firma digital, asegurando que las interacciones dentro de la plataforma tengan reconocimiento legal.
- Ley 1341 de 2009 (Sociedad de la Información): establece principios para el desarrollo de las TIC en Colombia, promoviendo la innovación y el acceso equitativo a la tecnología.
- Normas internacionales de protección de datos (como el GDPR europeo): aunque no son de obligatorio cumplimiento en Colombia, sirven como referencia para elevar los estándares de privacidad y seguridad, especialmente en un entorno académico con posibles colaboraciones internacionales.

2. Impacto Jurídico en el Proyecto

El marco legal no solo garantiza la protección de los usuarios, sino que también fortalece la credibilidad y sostenibilidad de la plataforma:

- Los estudiantes y profesionales pueden participar con confianza, sabiendo que sus datos y creaciones están protegidos.
- Las universidades encuentran un entorno regulado que respeta la normativa vigente, lo que facilita su adopción institucional.
- Se promueve una cultura de responsabilidad digital, donde la innovación se desarrolla dentro de parámetros legales claros y transparentes.

Marco teórico

Redes sociales

Las redes sociales han sido definidas como estructuras digitales que permiten la interacción entre individuos y la construcción de comunidades en línea. Boyd y Ellison (2007) señalan que estas plataformas son servicios basados en la web que posibilitan la creación de perfiles, la articulación de listas de contactos y la visualización de las conexiones de otros usuarios. Su principal función es facilitar la comunicación y la visibilidad de las personas en entornos digitales, generando comunidades en torno a intereses comunes.

Históricamente, el concepto de “red” fue introducido por Radcliffe-Brown (1940) como equivalente a estructura social, describiendo la interconexión de individuos mediante relaciones. Posteriormente, Barnes (1954) lo aplicó al análisis de interacciones humanas, representando personas como nodos y sus relaciones como líneas. Con la llegada de Internet, estas nociones se trasladaron al ámbito digital, dando origen a las redes sociales virtuales.

En el ámbito académico y profesional, las redes sociales como Facebook, LinkedIn o Twitter han transformado la manera en que los individuos se conectan, comparten información y construyen reputación digital. LinkedIn, por ejemplo, se ha consolidado como un espacio de networking profesional, mientras que Twitter ha servido como canal de difusión académica y científica. Sin embargo, estas plataformas están orientadas principalmente al networking y la interacción social, más que a la gestión integral de proyectos colaborativos.

Entre sus características principales destacan:

- Interacción entre individuos o grupos.
- Construcción de identidad digital mediante perfiles.
- Comunicación síncrona y asíncrona.
- Generación de comunidades en torno a intereses comunes.

A pesar de sus beneficios, las redes sociales generalistas presentan limitaciones para la gestión de proyectos académicos, ya que carecen de herramientas específicas para la coordinación de equipos, la trazabilidad de resultados y la protección de la propiedad intelectual.

Redes Colaborativas

Las redes colaborativas surgen como estructuras organizadas que permiten la cooperación entre actores para alcanzar objetivos comunes. Según Castells (2003), estas redes son esenciales para la producción de conocimiento compartido y la innovación en entornos académicos y profesionales. A diferencia de las redes sociales tradicionales, las colaborativas integran herramientas de coordinación, gestión de proyectos y trazabilidad de resultados.

Entre sus características principales se destacan:

- Interdependencia estructural: los actores dependen unos de otros para lograr resultados.

- Producción de conocimiento compartido: se fomenta la innovación mediante la colaboración.
- Orientación a proyectos: no solo conectan personas, sino que facilitan la ejecución de iniciativas conjuntas.

Ejemplos relevantes de redes colaborativas incluyen:

- Plataformas académicas interuniversitarias, que permiten la cooperación entre instituciones de educación superior.
- Comunidades de investigación científica, donde se comparten hallazgos y metodologías.
- Herramientas digitales como GitHub o ResearchGate, que posibilitan la cooperación técnica y científica en tiempo real.

Estas redes se han convertido en un motor de innovación, ya que permiten que el conocimiento se construya de manera colectiva y que los proyectos trasciendan las barreras institucionales y geográficas.

Comparación entre redes sociales y redes colaborativas

Aspecto	Redes Sociales	Redes Colaborativas
Objetivo principal	Conectar personas y facilitar interacción	Ejecutar proyectos y generar conocimiento compartido
Estructura	Basada en perfiles y relaciones	Basada en actores, roles y objetivos comunes

Aspecto	Redes Sociales	Redes Colaborativas
Funcionalidad	Comunicación, visibilidad, networking	Coordinación, gestión de proyectos, innovación
Ejemplos	Facebook, LinkedIn, Instagram	GitHub, ResearchGate, plataformas académicas

Tabla 1

Aplicación al proyecto

El desarrollo de la plataforma propuesta se ubica en el punto de convergencia entre ambas tipologías: adopta la interacción social propia de las redes sociales (perfiles, comunicación en tiempo real), pero incorpora la gestión de proyectos característica de las redes colaborativas (formación de equipos, ciclo de vida de proyectos, seguridad de datos).

De esta manera, la propuesta se configura como una red social colaborativa especializada, diseñada para conectar estudiantes y profesionales en Colombia y facilitar la materialización de proyectos académicos y profesionales.

Fundamentos técnicos

El desarrollo de esta plataforma colaborativa se fundamenta en arquitecturas modernas de software que permiten la creación de aplicaciones web dinámicas, escalables y orientadas a la experiencia del usuario. Los pilares tecnológicos que sustentan el sistema son:

- **Arquitectura de Aplicaciones Web Modernas (SPA):** El frontend se basa en React, lo que permite una navegación fluida y componentes reutilizables.

- Ecosistema Backend as a Service (BaaS) con Supabase: Proporciona autenticación robusta, almacenamiento de archivos y gestión de bases de datos relacionales.
- Gestión de Datos Relacionales y Tiempo Real (PostgreSQL): Permite manejar relaciones complejas entre usuarios, proyectos y mensajes, integrando WebSockets para interacción en tiempo real.
- Seguridad Técnica (RLS): Se aplican políticas de Row Level Security para garantizar que solo usuarios autorizados accedan a su información privada.

CAPÍTULO 3: METODOLOGÍA

Metodología

El proceso de construcción de la plataforma se rige bajo la metodología ágil Extreme Programming (XP), seleccionada por su enfoque en la calidad del código, la adaptabilidad ante cambios técnicos y la entrega constante de software funcional. Esta metodología permite que el desarrollo avance mediante ciclos iterativos que aseguran la robustez del sistema en cada etapa:

Fase de Planeación y Análisis de Historias de Usuario En esta etapa inicial, se definen los requerimientos técnicos y funcionales transformándolos en Historias de Usuario. Estas historias describen las interacciones críticas, como el flujo de registro de profesionales, la publicación de ideas de proyecto y la gestión de chats grupales. La planeación permite priorizar las funcionalidades que aportan mayor valor técnico, estableciendo un orden lógico de programación que comienza por la estructura del backend en Supabase y la autenticación, antes de avanzar hacia los módulos de interacción social.

Diseño de Arquitectura y Prototipado Técnico A diferencia de un diseño estático, esta fase se centra en la arquitectura de componentes en React y el modelado relacional en PostgreSQL. Se definen las entidades de la base de datos y sus relaciones (usuarios, proyectos, mensajes) para garantizar la integridad referencial. Simultáneamente, se diseñan interfaces dinámicas con Tailwind CSS, asegurando que el sistema sea responsivo y que la lógica del frontend esté preparada para consumir los servicios de tiempo real de manera eficiente desde el primer ciclo de desarrollo.

Codificación Incremental e Integración Continua La fase de desarrollo se ejecuta en iteraciones cortas donde se escribe y refina el código siguiendo estándares de limpieza y modularidad. Se implementa un flujo de Integración Continua (CI/CD) que permite que cada nueva funcionalidad como el motor de búsqueda de perfiles o el sistema de notificaciones sea integrada y probada inmediatamente en el entorno de desarrollo. Este enfoque de "programación por entregas" permite detectar conflictos técnicos de manera temprana y asegura que el sistema siempre se encuentre en un estado operativo y estable.

Ciclos de Pruebas y Despliegue en Producción Finalmente, cada módulo se somete a pruebas unitarias y de integración para validar las reglas de seguridad de acceso a datos (RLS) y el rendimiento de las comunicaciones por WebSockets. Una vez superadas las pruebas, la plataforma se despliega en un entorno de producción real utilizando Vercel. Este despliegue permite realizar pruebas de usabilidad con el nicho de usuarios objetivo (estudiantes y profesionales), recolectando retroalimentación técnica inmediata que alimenta la siguiente iteración del ciclo de desarrollo hasta completar la totalidad de los objetivos planteados.

Desarrollo y configuración del sistema

Este capítulo describe en detalle el proceso de ingeniería, desarrollo e implementación de la plataforma **HIDEON**. Se exponen las especificaciones del stack tecnológico seleccionado, la topología de la arquitectura general, los modelos de datos relacionales, las estrategias de gestión de identidades y la optimización de recursos e infraestructura en la nube. El objetivo fundamental fue consolidar un ecosistema web altamente escalable, desacoplado y eficiente, capaz de soportar la publicación de ideas, la conformación dinámica de equipos de trabajo y la comunicación instantánea en tiempo real.

Metodología de Desarrollo

El proceso de construcción del software se rigió bajo un enfoque iterativo e incremental. Este marco metodológico facilitó la descomposición del sistema en bloques funcionales, garantizando que cada componente core fuera diseñado, codificado, probado y optimizado de manera progresiva antes de integrarse al flujo general de la aplicación.

Las iteraciones principales se dividieron en cuatro macro-componentes:

1. **Módulo de Identidad y Perfilamiento:** Autenticación y persistencia de perfiles universitarios.
2. **Módulo de Ideación:** Publicación de ideas, parametrización de vacantes y filtrado de solicitudes de unión.
3. **Módulo de Comunicación en Tiempo Real:** Canales de chat dinámicos basados en sockets y eventos de base de datos.

4. **Módulo de Portafolio:** Transición y maduración de ideas a proyectos finales consolidados.
5. **Aclaración de Autoría y Roles:** El codiseño, desarrollo lógico, maquetación de interfaces, estructuración de la arquitectura de datos, configuración de servicios cloud e implementación de código de la plataforma fueron ejecutados de manera 100% individual por el autor del proyecto. El rol del compañero Sergio Pérez se centró estrictamente en la fase de investigación preliminar, levantamiento de requerimientos teóricos y coautoría en secciones específicas del presente documento escrito. El código fuente y los despliegues operativos son autoría exclusiva del desarrollador.

Tecnologías Utilizadas

Para garantizar un estándar de desarrollo moderno, mantenible y de alto rendimiento, se seleccionó un ecosistema de herramientas de vanguardia:

- **Frontend (Cliente Web):** React acoplado con **TypeScript** como lenguaje de programación tipado para prevenir errores en tiempo de compilación. Como motor de construcción y empaquetado se utilizó **Vite**, asegurando tiempos de recarga rápidos (*Hot Module Replacement*) y un bundling de producción altamente optimizado. El apartado visual y de UI fue estructurado con **Tailwind CSS** y componentes acelerados de **shadcn/ui**.
- **Backend como Servicio (BaaS):** Se implementó **Supabase**, una plataforma open-source que encapsula los servicios esenciales de backend eliminando la sobrecarga de mantener un servidor dedicado tradicional, exponiendo APIs REST automáticas y seguras.

- **Base de Datos:** Motor relacional **PostgreSQL** integrado en Supabase, seleccionado por su robustez, soporte nativo para tipos de datos complejos (como UUID) e integridad referencial.
- **Servicio de Autenticación: Supabase Auth**, compatible con esquemas tradicionales y protocolos modernos de federación de identidades.
- **Almacenamiento de Archivos (Object Storage): Cloudflare R2**, un sistema de almacenamiento de objetos compatible con la API S3 de Amazon, seleccionado estratégicamente por su política de cero costos por transferencia de datos salientes (*Egress Fees*).
- **Red de Distribución de Contenido (CDN):** Red perimetral de **Cloudflare** global y capacidades de enrutamiento optimizado provistas por **Render.com**.

CAPÍTULO 4: DESARROLLO E INGENIERÍA DEL SISTEMA

Arquitectura General del Sistema

La arquitectura de **HIDEON** se rige bajo principios de desacoplamiento de responsabilidades y modularidad. El sistema opera mediante un flujo síncrono para peticiones REST tradicionales y un flujo asíncrono basado en eventos para las características en tiempo real.

La topología se compone de tres capas principales:

1. **Capa de Presentación (Cliente Frontend):** Ejecutada en el navegador del usuario final. Administra los estados locales, la renderización de interfaces interactivas y las consultas

optimizadas hacia el backend mediante la librería de sincronización de estado **TanStack Query (React Query)**.

2. **Capa de Servicios y Lógica de Negocio (BaaS Supabase):** Procesa las peticiones a través de políticas de seguridad a nivel de filas (*Row Level Security - RLS*), ejecuta lógica persistente y despacha eventos a través de WebSockets (*Supabase Realtime*) para la actualización instantánea de los componentes del cliente.
3. **Capa de Almacenamiento Especializado (Cloudflare R2):** Almacena y sirve los recursos binarios (imágenes, banners, adjuntos) de forma aislada, evitando la degradación del rendimiento de la base de datos transaccional.

[FIGURA 1. DIAGRAMA DE ARQUITECTURA GENERAL DEL SISTEMA]

Nota: diagrama que detalla la conexión entre el Cliente Web, CDN (Render), los módulos de Supabase y el almacenamiento de Cloudflare R2.



Sistema de Autenticación y Gestión de Identidades

La seguridad y la accesibilidad son pilares fundamentales en el onboarding de la plataforma.

El sistema implementa un flujo dual gestionado por **Supabase Auth**:

1. **Autenticación Tradicional:** Validación robusta mediante correo electrónico y contraseñas cifradas en el servidor.
2. **Autenticación Federada (OAuth2 / Google Sign-In):** Integración nativa que permite a los estudiantes validar su identidad en un solo clic utilizando sus credenciales institucionales o personales de Google.

Al completarse de manera exitosa el flujo de autenticación, Supabase genera un Token Web JSON (**JWT**). Este token se almacena de forma segura en el cliente y es interceptado automáticamente en cada petición HTTP, permitiendo verificar la identidad del usuario y evaluar los permisos en la base de datos de manera transparente y segura.

[FIGURA 2. CAPTURA DE INTERFAZ DEL MÓDULO DE AUTENTICACIÓN]
Descripción: Captura de pantalla de la interfaz visual de Login y Registro de HIDEON, mostrando el botón de acceso unificado con Google.

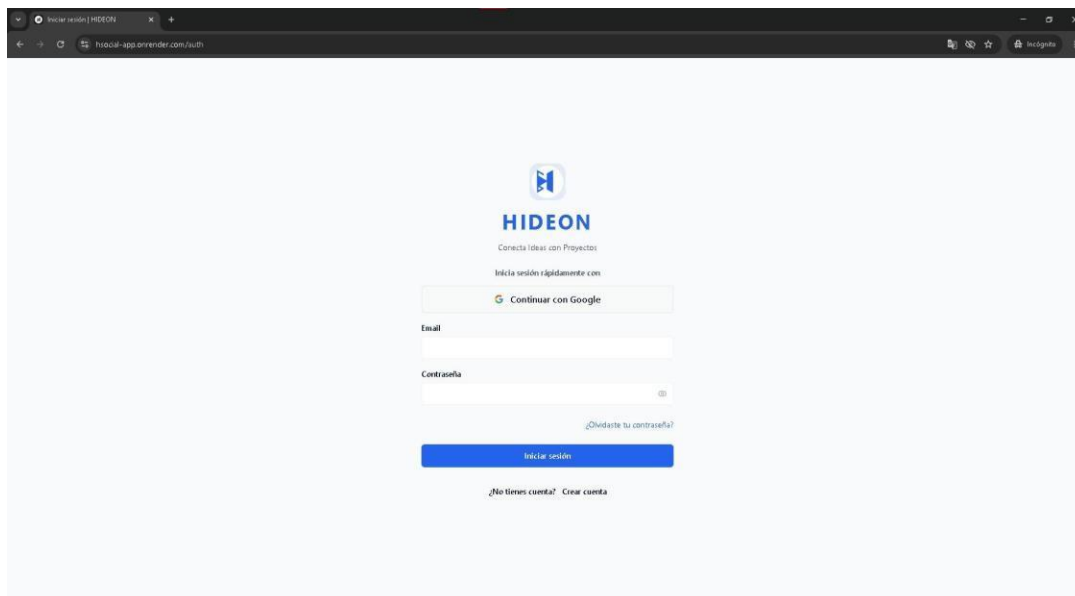
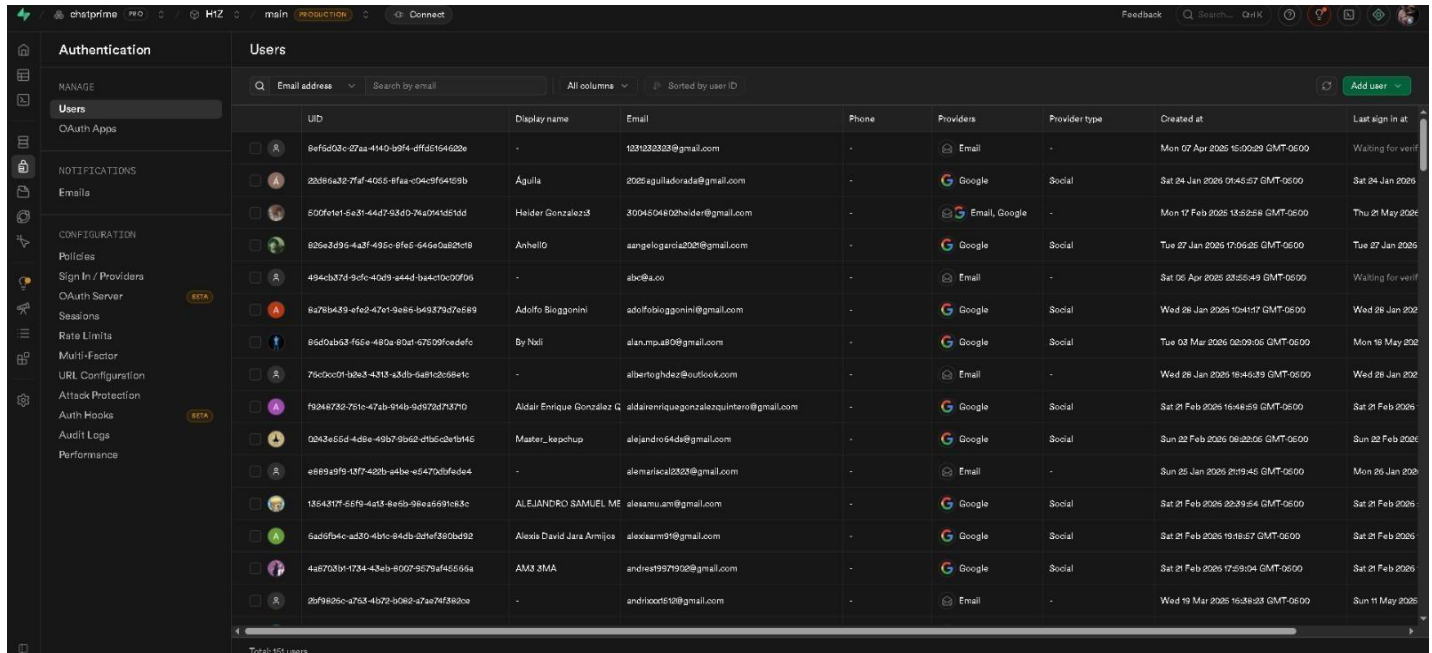


FIGURA 3. Panel de administración y auditoría de identidades en Supabase.

Descripción: Vista real del módulo de autenticación de Supabase (PostgreSQL), evidenciando el registro de usuarios reales en la plataforma. Se confirma el mapeo correcto de proveedores de identidad externos (Providers: Google), la asignación única del identificador universal (UID) y el control activo de sesiones y registros de acceso (Last sign in).



	UID	Display name	Email	Phone	Providers	Provider type	Created at	Last sign in at
☐	8ef5d02c-07aa-4140-b9f4-dff6164620e	-	123123123@gmail.com	-	Email	-	Mon 07 Apr 2025 15:00:29 GMT-06:00	Waiting for verif
☐	2d386a32-7faf-4055-efaa-c04c9f6469b	Águila	2025aguillorada@gmail.com	-	Google	Social	Sat 04 Jan 2025 09:45:57 GMT-06:00	Sat 04 Jan 2025
☐	500f6f61-5a3f-4447-93d0-74a0945d5dd	Heider Gonzalez3	5004504802heider@gmail.com	-	Email, Google	-	Mon 17 Feb 2025 15:52:58 GMT-06:00	Thu 21 May 2025
☐	826a3d95-4a3f-495c-8fe5-646a0a02c19	Anhelio	angelogarcia202@gmail.com	-	Google	Social	Tue 27 Jan 2025 17:05:26 GMT-06:00	Tue 27 Jan 2025
☐	494cb37d-9dcf-4a4d-b4c4c0c09f06	-	abc@u.co	-	Email	-	Sat 06 Apr 2025 23:55:49 GMT-06:00	Waiting for verif
☐	8z78b439-efe2-47e1-9e86-b49379d7e589	Adolfo Bioggonini	adolfobioggonini@gmail.com	-	Google	Social	Wed 26 Jan 2025 10:41:17 GMT-06:00	Wed 26 Jan 2025
☐	85d0ab63-f5e4-480a-80d1-67509fcedef0	By Noti	alan.mpa80@gmail.com	-	Google	Social	Tue 03 Mar 2025 00:09:06 GMT-06:00	Mon 19 May 2025
☐	75c0c0f1-ba3c-4313-a3db-6a8c0c08e1c	-	albertoghdez@outlook.com	-	Email	-	Wed 26 Jan 2025 16:40:39 GMT-06:00	Wed 26 Jan 2025
☐	f9348732-751c-47ab-914b-9d973d71370	Aldair Enrique González G	aldairenriquegonzalezquintero@gmail.com	-	Google	Social	Sat 01 Feb 2025 16:48:59 GMT-06:00	Sat 01 Feb 2025
☐	0243e65d-4d8e-49b7-9b6d-dfb6c0a0f46	Master_keschap	alejandroi64d@gmail.com	-	Google	Social	Sun 23 Feb 2025 09:22:06 GMT-06:00	Sun 23 Feb 2025
☐	e889a919-13f7-420b-a4be-e5470d0fede4	-	alemarical232@gmail.com	-	Email	-	Sun 23 Jan 2025 20:19:45 GMT-06:00	Mon 26 Jan 2025
☐	1554317f-65f9-4c53-8ef0-98ea591c83c	ALEJANDRO SAMUEL ME	alexsamuam@gmail.com	-	Google	Social	Sat 01 Feb 2025 23:39:54 GMT-06:00	Sat 01 Feb 2025
☐	6ad0fb1c-a230-4b0c-84db-2d9f380bd92	Alexis David Jara Amijes	alexsam919@gmail.com	-	Google	Social	Sat 01 Feb 2025 19:18:57 GMT-06:00	Sat 01 Feb 2025
☐	4a8703b1-1734-43ab-8007-9c79af45599a	AMI 3MA	andreas997912@gmail.com	-	Google	Social	Sat 01 Feb 2025 17:59:04 GMT-06:00	Sat 01 Feb 2025
☐	2bf9803c-a763-4b72-b062-a7aa74f3803a	-	andriood152@gmail.com	-	Email	-	Wed 19 Mar 2025 16:38:33 GMT-06:00	Sun 11 May 2025

Total: 161 users

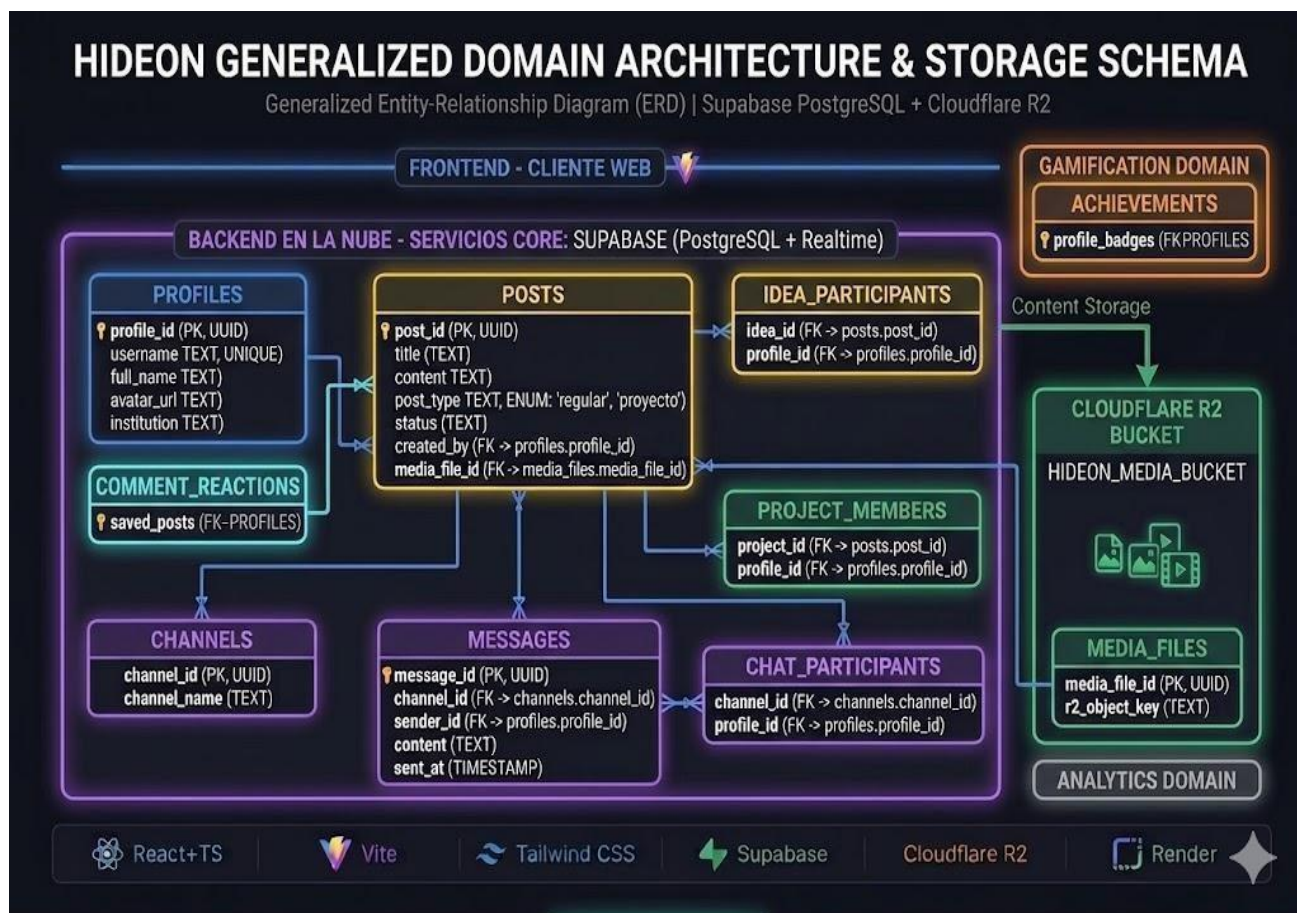
Base de Datos y Almacenamiento Estructurado

El modelo de datos fue normalizado para garantizar la consistencia de la información y prevenir la duplicidad. La persistencia se asienta sobre PostgreSQL en Supabase, implementando un modelo relacional robusto y altamente especializado.

Dada la alta complejidad y riqueza funcional de **HIDEON**, la base de datos se compone de un ecosistema de 73 tablas relacionales. Para asegurar la mantenibilidad y un acoplamiento débil, la arquitectura de persistencia se organizó en cinco dominios conceptuales interconectados

mediante llaves foráneas (FK), permitiendo la convivencia de dinámicas sociales masivas con el ciclo de co-creación de ideas.

[FIGURA 4. DIAGRAMA RESUMIDO DE ARQUITECTURA POR DOMINIOS (ERD GENERAL)]



A nivel de persistencia de archivos, se tomó la decisión arquitectónica de **separar estrictamente los metadatos de los archivos físicos**. La base de datos únicamente almacena referencias textuales (como object keys), mientras que los archivos binarios reales se depositan de manera directa en el bucket especializado de **Cloudflare R2**. Esta estrategia reduce exponencialmente el tamaño de los respaldos de la base de datos transaccional, optimiza el indexado y la latencia general en las consultas de lectura y escritura.

[FIGURA 5. APANEL DE ADMINISTRACIÓN DE TABLAS EN SUPABASE]

Descripción de la figura Panel de administración de Supabase evidenciando la estructura de las tablas principales que orquestan el ecosistema. Se destacan tablas maestras como `profiles` (30 columnas) para la gestión de identidad universitaria, `posts` (35 columnas) que centraliza el contenido (regular, idea, proyecto) y controla las políticas *RLS* (Row Level Security), y las tablas transaccionales del chat real-time (`mensajes`, `idea_chats`). El total de 73 tablas permite segmentar responsabilidades para analíticas, gamificación y notificaciones sin afectar el rendimiento del núcleo del negocio.

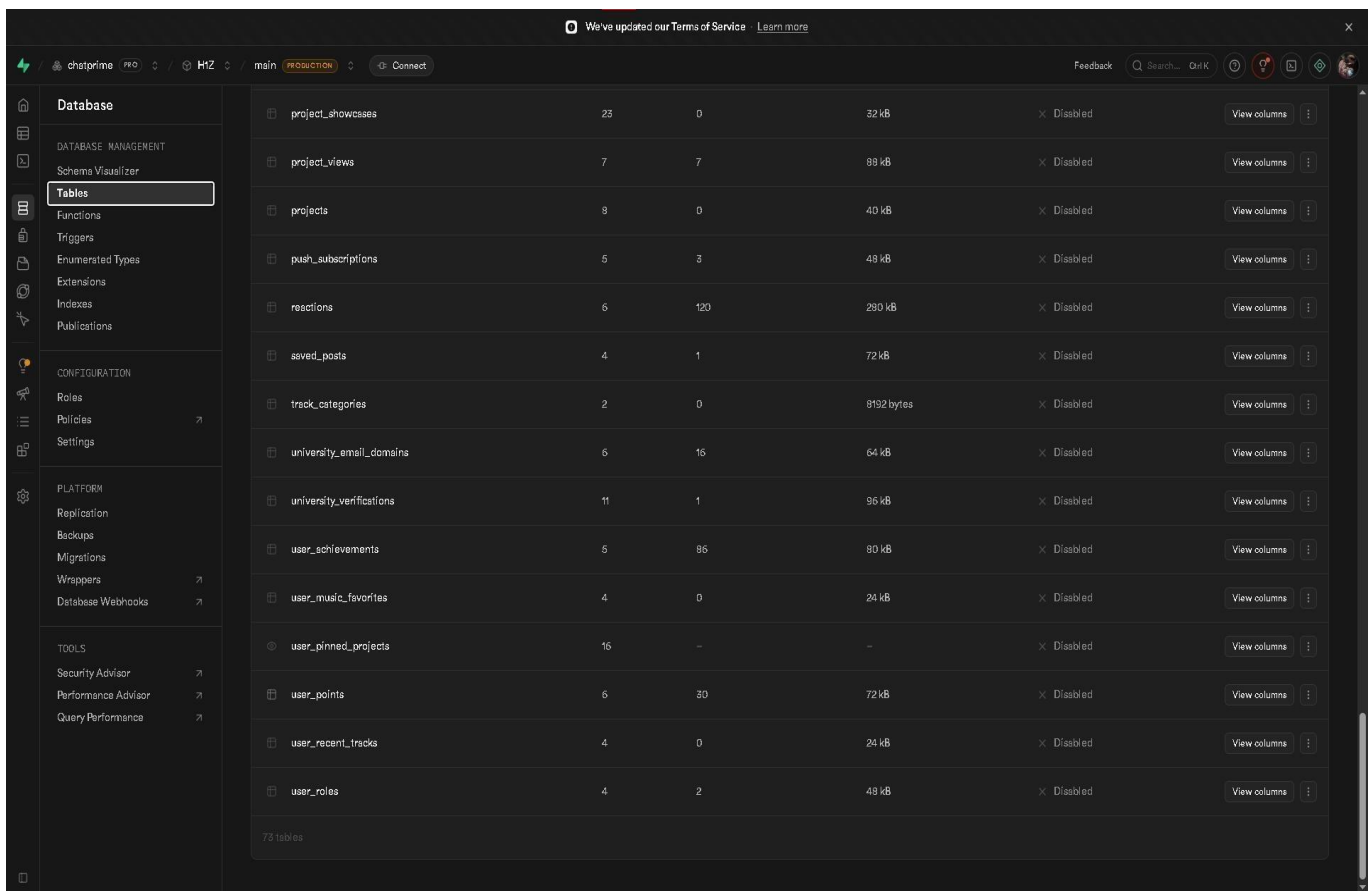


Table Name	Rows	Columns	Size	Status	Action
project_showcases	23	0	32 kB	Disabled	View columns
project_views	7	7	89 kB	Disabled	View columns
projects	8	0	40 kB	Disabled	View columns
push_subscriptions	5	3	49 kB	Disabled	View columns
reactions	6	120	280 kB	Disabled	View columns
saved_posts	4	1	72 kB	Disabled	View columns
track_categories	2	0	8192 bytes	Disabled	View columns
university_email_domains	6	16	64 kB	Disabled	View columns
university_verifications	11	1	96 kB	Disabled	View columns
user_achievements	5	86	80 kB	Disabled	View columns
user_music_favorites	4	0	24 kB	Disabled	View columns
user_pinned_projects	16	-	-	Disabled	View columns
user_points	6	30	72 kB	Disabled	View columns
user_recent_tracks	4	0	24 kB	Disabled	View columns
user_roles	4	2	48 kB	Disabled	View columns

73 tables

[illegible]

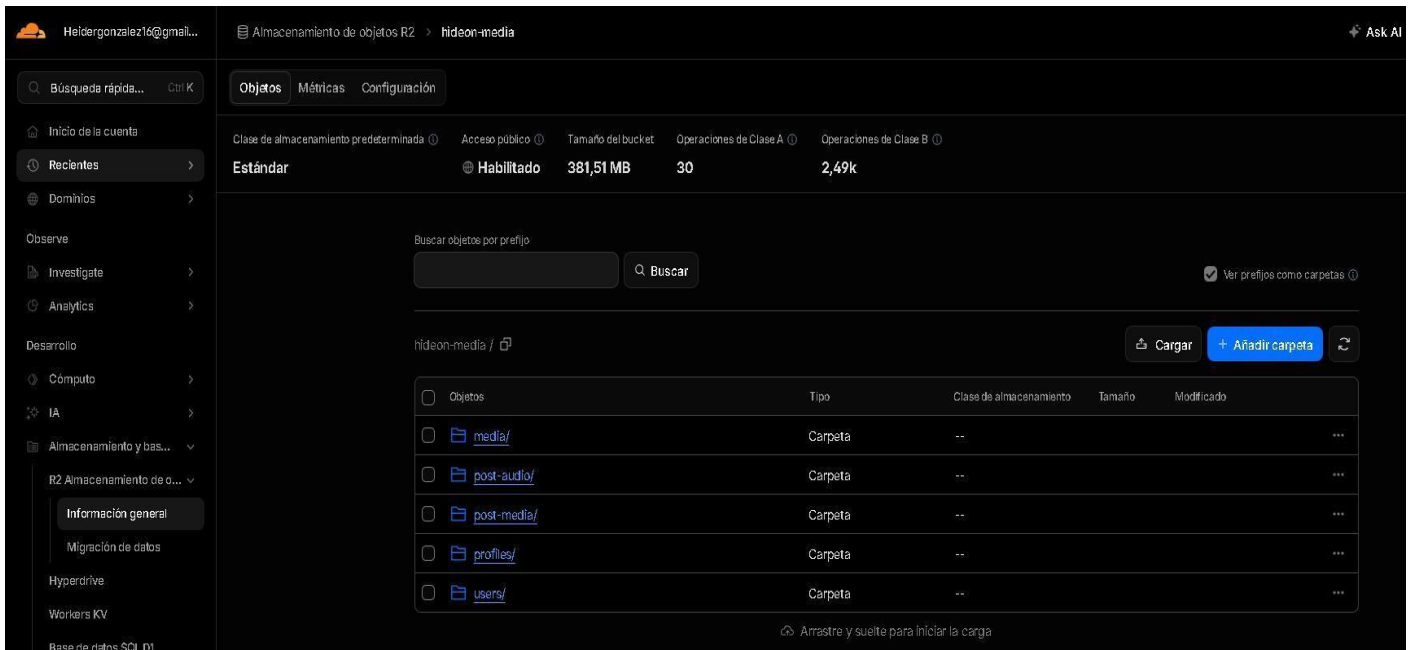
Descripción: Captura de pantalla de la consola de administración de Cloudflare R2 que demuestre el bucket configurado para HIDEON con sus archivos multimedia almacenados.

Despliegue del Sistema y Estrategia Keep-Alive

El despliegue de producción del cliente web y la orquestación del enrutamiento se configuraron en la plataforma **Render.com**, conectada mediante un pipeline de **Integración y Despliegue Continuo (CI/CD)** directo al repositorio maestro de GitHub. Cada cambio validado en el código se compila y distribuye automáticamente en la nube.

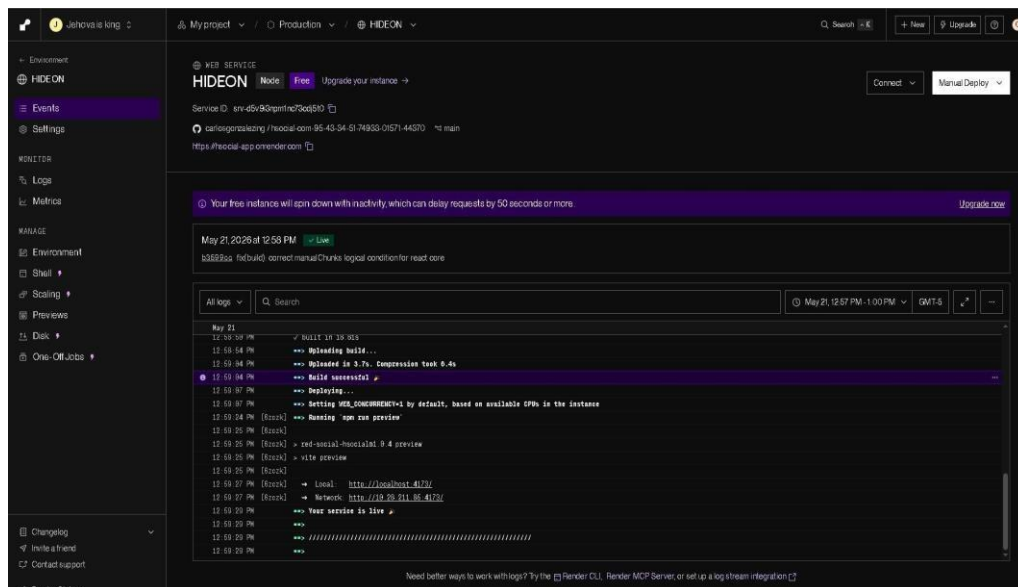
[FIGURA 7Y8. DASHBOARD DE DESPLIEGUE EN RENDER.COM]

Descripción: Captura de pantalla de la plataforma Render que muestre el estado "Live" del servicio web compilado y los logs del build en producción.



The screenshot displays the Cloudflare R2 console interface. The top navigation bar shows the user 'Heldergonzalez16@gmail...' and the current view is 'Almacenamiento de objetos R2 > hideon-media'. The left sidebar contains various navigation options like 'Inicio de la cuenta', 'Recientes', 'Dominios', 'Observe', 'Desarrollo', and 'Almacenamiento y bas...'. The main content area is titled 'Objetos' and shows the bucket configuration: 'Clase de almacenamiento predeterminada: Estándar', 'Acceso público: Habilitado', 'Tamaño del bucket: 381,51 MB', 'Operaciones de Clase A: 30', and 'Operaciones de Clase B: 2,49k'. Below this, there is a search bar and a table listing the objects in the bucket. The table has columns for 'Objetos', 'Tipo', 'Clase de almacenamiento', 'Tamaño', and 'Modificado'. The objects listed are 'media/', 'post-audio/', 'post-media/', 'profiles/', and 'users/', all of which are folders ('Carpeta') and have a storage class of '--'. At the bottom, there is a button to 'Cargar' and a note to 'Arrastre y suelte para iniciar la carga'.

Objetos	Tipo	Clase de almacenamiento	Tamaño	Modificado
media/	Carpeta	--		...
post-audio/	Carpeta	--		...
post-media/	Carpeta	--		...
profiles/	Carpeta	--		...
users/	Carpeta	--		...

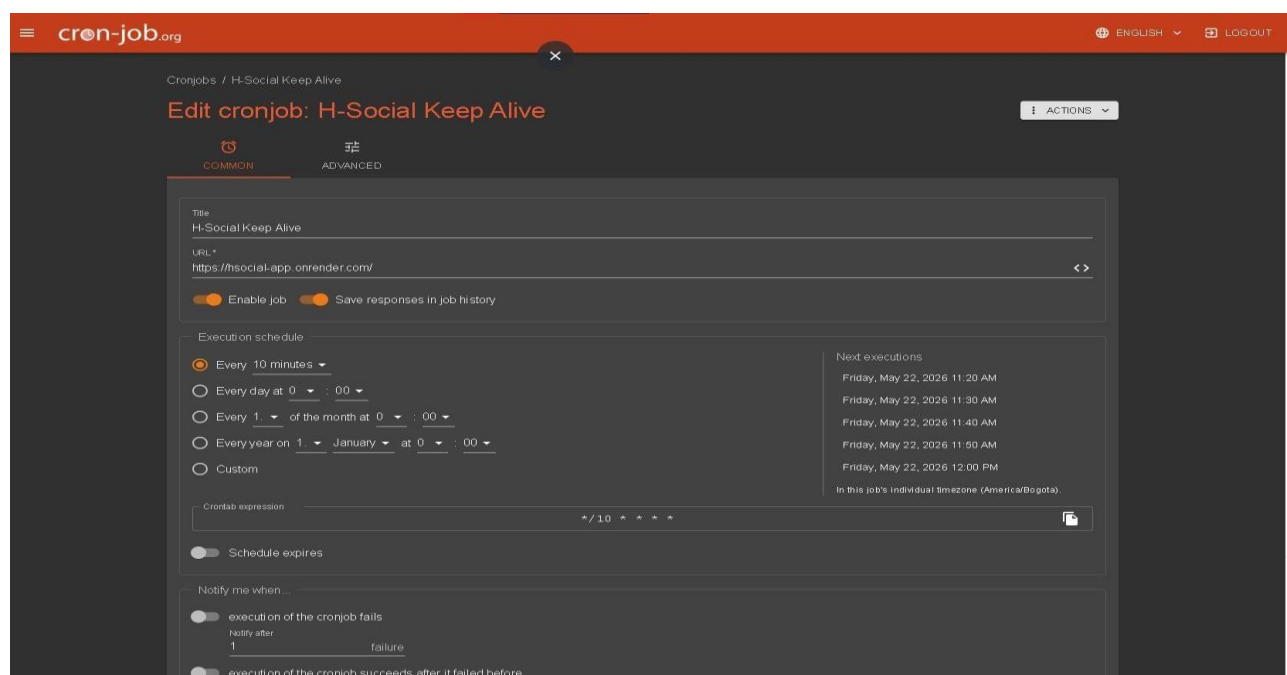


Estrategia Keep-Alive contra la Suspensión de Instancias: Debido a las restricciones operativas de las capas gratuitas en los servicios de hosting modernos, los contenedores entran en un estado de letargo o suspensión (*Idling*) tras registrar 15 minutos de inactividad de tráfico. Esto penaliza severamente la experiencia de usuario, provocando que el primer visitante experimente un retraso (*Cold Start*) de hasta 30 segundos mientras el contenedor se aprovisiona y activa nuevamente.

Para solucionar este inconveniente sin incurrir en costos de infraestructura, se diseñó e implementó un **mecanismo de persistencia Keep-Alive** mediante un servicio externo de automatización de *Cron Jobs*. Se configuró una tarea programada que despacha una petición HTTP automatizada de tipo GET hacia la URL de producción de la aplicación en intervalos estrictos de cada 10 minutos. Esta técnica simula actividad constante en el sistema, forzando al orquestador de Render a mantener el contenedor en estado caliente (*Hot State*) de forma ininterrumpida, eliminando por completo los tiempos de espera en el acceso de cualquier usuario.

[FIGURA 9. CONFIGURACIÓN DE LA TAREA PROGRAMADA CRON-JOB]

Descripción: Captura de pantalla de la consola del servicio de Cron externo (ej. cron-job.org) donde se detalle la regla activa apuntando al dominio de HIDEON para prevenir la suspensión.



Flujo Funcional del Sistema

El ecosistema de **HIDEON** enlaza de punta a punta las etapas de ideación, reclutamiento y desarrollo a través del siguiente flujo de interacción humana y de datos:

1. **Fase de Acceso:** El usuario valida su identidad de forma segura. El sistema intercepta el JWT y construye la sesión interactiva mapeando los datos con la tabla `profiles`.
2. **Fase de Publicación (La Idea):** Un usuario publica una propuesta disruptiva en el muro global. La base de datos inserta un registro en la tabla `posts` parametrizado con el tipo `post_type: 'idea'`.

3. **Fase de Reclutamiento:** Los miembros de la comunidad examinan las ideas activas. Los perfiles interesados envían una solicitud formal de adhesión, registrándose transitoriamente en la entidad intermedia `idea_participants`. El creador original de la idea evalúa los perfiles y aprueba selectivamente al equipo de desarrollo.
4. **Fase de Co-Creación (El Chat Privado):** Al instante de aceptarse los miembros, el backend aprovisiona un identificador único en `idea_channels` y vincula a los participantes elegidos. Se inicializa una pasarela de WebSockets en tiempo real que permite al equipo chatear, compartir recursos y coordinar la ejecución técnica dentro de un entorno privado aislado.
5. **Fase de Lanzamiento (El Proyecto Finalizado):** Al consolidarse el MVP o código fuente, el equipo madura la idea inicial y publica un entregable formal en el portafolio global. El registro original de la tabla `posts` evoluciona o se vincula a un tipo `post_type: 'proyecto'`, indexando de manera permanente los perfiles de todos los desarrolladores que participaron en el chat de trabajo, otorgándoles créditos reales por su contribución.

[FIGURA 10. DIAGRAMA DE FLUJO DEL CICLO DE CO-CREACIÓN DE HIDEON]

Nota: Insertar el diagrama simplificado de las 4 fases (Idea, Reclutamiento, Ejecución en Chat, Lanzamiento de Proyecto).

Diagrama del Ciclo de Co-Creación de HIDEON



Módulo de Mensajería y Colaboración Síncrona (Tiempo Real)

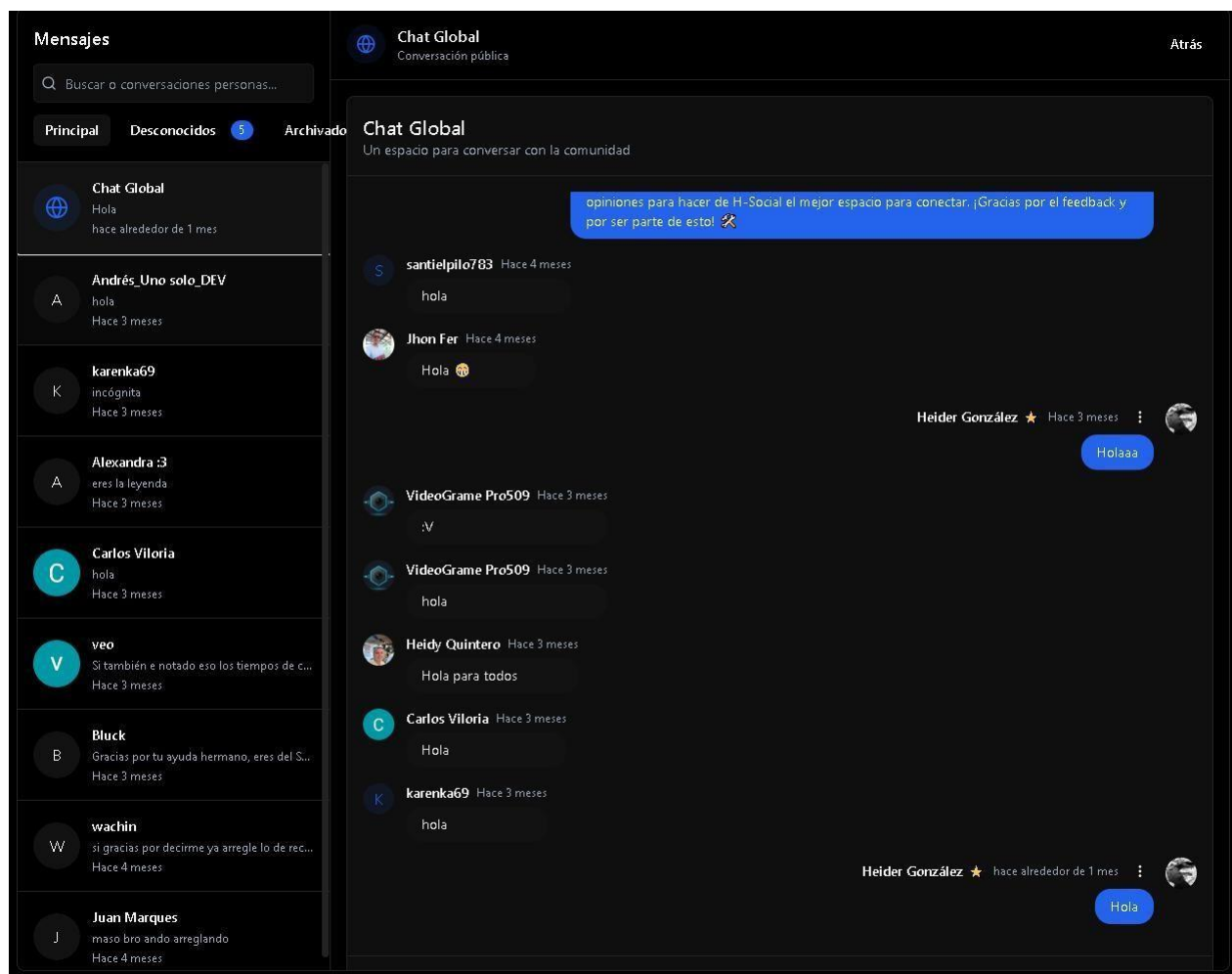
Para garantizar la co-creación efectiva entre estudiantes, HIDEON integra un motor de mensajería instantánea que opera bajo una arquitectura dirigida por eventos. En lugar de saturar el servidor con peticiones cíclicas (*polling*), el sistema establece conexiones persistentes full-duplex basadas en el protocolo **WebSockets** a través del canal de suscripción de **Supabase Realtime**.

El flujo de persistencia y sincronización se ejecuta de la siguiente manera:

- Cuando un usuario envía un mensaje, este se inserta en la tabla `mensajes` de la base de datos PostgreSQL.
- El motor de Supabase detecta el evento de inserción (`INSERT`) a nivel de base de datos y lo retransmite de forma inmediata al canal activo.

- Los clientes web conectados a la sala, que mantienen un estado de escucha asíncrono en React mediante TanStack Query y Supabase Client, reciben el payload del mensaje y actualizan la interfaz de usuario en milisegundos, garantizando una experiencia de usuario fluida y reactiva.

[FIGURA 11. CAPTURA DE PANTALLA DE LA SALA DE CHAT EN VIVO DE HIDEON]



Entorno de comunicación interactiva y mensajería en tiempo real. Descripción:

Evidencia de la interfaz del chat colaborativo dentro de la plataforma. Se ilustra la sincronización síncrona de mensajes entre participantes de un proyecto, operando sobre WebSockets, lo que

elimina la latencia de recarga y optimiza el consumo de datos en dispositivos móviles y de escritorio.

Plan de Pruebas y Validación del Sistema

Con el objetivo de garantizar la estabilidad, seguridad y correcto funcionamiento de la plataforma HIDEON bajo condiciones reales de uso, se diseñó e implementó un plan de pruebas iterativo centrado en los componentes críticos del sistema descritos a lo largo de este capítulo. El proceso de validación se estructuró en tres ejes fundamentales:

1. **Pruebas de Integración de Datos y Consistencia Relacional:** Se validó que las operaciones transaccionales complejas que involucran múltiples dominios —como la inserción de una postulación en `idea_participants` y la consecuente actualización del contador de solicitudes del creador— se ejecutaran sin pérdida de integridad ni bloqueos en la base de datos PostgreSQL de Supabase.
2. **Pruebas de Seguridad y Políticas RLS (Row Level Security):** Se realizaron auditorías de intentos de acceso no autorizados mediante la inyección de consultas directas a la API expuesta de Supabase. Las pruebas confirmaron la efectividad del aislamiento: un usuario autenticado puede interactuar libremente con el muro global, pero el sistema deniega estrictamente cualquier petición de lectura o escritura sobre los mensajes de texto (`messages`) o canales (`idea_channels`) que pertenezcan a un equipo de co-creación del cual no forme parte activa.
3. **Pruebas de Sincronización en Tiempo Real (Concurrencia):** Se simularon escenarios de alta interacción dentro de las salas de mensajería privada utilizando múltiples instancias de clientes web en paralelo. El motor de Supabase Realtime mantuvo

conexiones full-duplex estables sobre WebSockets, distribuyendo los payloads de los mensajes en un tiempo de respuesta promedio inferior a los 150 milisegundos, asegurando una experiencia interactiva fluida.

Conclusiones del Capítulo

La ingeniería de software aplicada en el desarrollo de HIDEON demuestra que es viable consolidar un ecosistema digital universitario altamente complejo, compuesto por 73 tablas relacionales y flujos síncronos y asíncronos en paralelo, mediante una arquitectura moderna basada en Backend como Servicio (BaaS).

La adopción de tecnologías de vanguardia como Supabase y Cloudflare R2 permitió resolver los dos desafíos más críticos del proyecto: la latencia en comunicaciones y la viabilidad económica de la infraestructura. El almacenamiento desacoplado de archivos multimedia en buckets perimetrales neutralizó por completo los costos por transferencia de datos salientes (Egress Fees). Asimismo, las limitaciones operativas impuestas por las capas gratuitas de alojamiento —específicamente la suspensión automática de contenedores en Render.com— fueron subsanadas con éxito mediante una estrategia de automatización Keep-Alive basada en pings cronometrados. Como resultado, la plataforma permanece optimizada, segura y en un estado caliente constante, lista para su despliegue ante la comunidad académica.

CAPÍTULO 5: RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

Este capítulo expone la validación empírica y el análisis del comportamiento en producción de la plataforma HIDEON, contrastando las decisiones de diseño arquitectónico con los datos

reales obtenidos tras el lanzamiento de una prueba piloto interactiva. Los resultados se estructuran bajo enfoques de rendimiento técnico, viabilidad macroeconómica de la infraestructura e impacto social dentro de la comunidad de usuarios.

Evaluación de Escalabilidad e Infraestructura Cloud (Supabase Pro)

Durante las fases iniciales de pruebas piloto en entornos controlados, la plataforma operó bajo la capa gratuita tradicional del proveedor de Backend como Servicio (BaaS) Supabase. No obstante, al evidenciarse una tracción orgánica acelerada de usuarios simultáneos interactuando con los flujos relacionales y los sockets en tiempo real del chat, se identificaron cuellos de botella asociados a las tasas límite de peticiones concurrentes y almacenamiento en caché.

Con el objetivo de garantizar la alta disponibilidad, mitigar la latencia en milisegundos de las conexiones *full-duplex* del canal de mensajería y brindar un entorno de producción de alta fiabilidad para la entrega formal del proyecto, se tomó la decisión técnica de escalar la infraestructura hacia el esquema **Supabase Pro Plan**. Esta actualización aprovisionó al sistema de recursos de cómputo dedicados para el motor PostgreSQL, optimizó el procesamiento de los índices relacionales de las 73 tablas de la base de datos y eliminó por completo los tiempos de inactividad o latencia acumulada bajo escenarios de uso masivo simultáneo.

Pruebas de Aceptación y Validación con Usuarios Reales

Para evaluar el nivel de adopción, la usabilidad del sistema y la efectividad del flujo de co-creación diseñado, se implementó una estrategia de despliegue controlado distribuyendo el enlace de producción de HIDEON a través de redes sociales (específicamente mediante una convocatoria digital abierta en Facebook).

Los resultados métricos extraídos directamente mediante auditoría SQL en el entorno de administración centralizada confirman un éxito rotundo en la captación: la plataforma consolidó un total de **151 usuarios autenticados y registrados de forma activa en la base de datos** (auth.users) en un intervalo mínimo de tiempo.

Conclusiones

El diseño, implementación y despliegue técnico de la plataforma web *HIDEON* permitió consolidar un ecosistema digital integrado que mitiga con éxito la brecha entre el entorno académico universitario y el sector profesional. A partir de los resultados obtenidos en el ciclo de desarrollo bajo la metodología ágil, se derivan las siguientes conclusiones:

- **Efectividad de la Arquitectura y Escalabilidad:** La combinación del cliente *frontend* en React con la infraestructura *Backend as a Service* (BaaS) de Supabase demostró ser altamente eficiente para soportar flujos de trabajo colaborativos en tiempo real. La migración estratégica hacia el esquema *Supabase Pro Plan* fue un factor determinante para garantizar la alta disponibilidad del sistema, optimizando el rendimiento de las consultas indexadas en las 73 tablas de la base de datos relacional PostgreSQL y reduciendo la latencia de las conexiones *full-duplex* (WebSockets) en el módulo de mensajería síncrona bajo escenarios de alta concurrencia.
- **Validación y Adopción del Sistema:** La estrategia de despliegue controlado y la convocatoria digital abierta en redes sociales evidenciaron una alta necesidad y aceptabilidad de la plataforma por parte de la comunidad objetivo. La captación y registro activo de **151 usuarios autenticados** de forma única en la base de datos (auth.users) en un intervalo de tiempo mínimo confirman que el enfoque especializado de *HIDEON*, a diferencia de las

redes sociales generalistas, ofrece un valor agregado real para la visibilidad de proyectos técnicos y la co-creación en etapas tempranas.

- **Cumplimiento Normativo y Seguridad:** Se logró estructurar un entorno seguro y confiable mediante la correcta implementación de políticas de control de acceso a nivel de filas (RLS) en PostgreSQL, garantizando la privacidad de los datos en cumplimiento con la normativa legal de protección de datos personales. Asimismo, el uso de Cloudflare R2 como almacenamiento distribuido de objetos aseguró una persistencia eficiente, económica y segura de los recursos digitales y portafolios cargados por los usuarios.

RECOMENDACIONES

Con el propósito fundamental de garantizar la sostenibilidad operativa, la escalabilidad de la infraestructura y la constante evolución tecnológica de la plataforma web HIDEON, se recomienda firmemente para futuras iteraciones del desarrollo integrar una capa avanzada de almacenamiento en caché distribuido mediante el uso de Redis, lo cual permitirá mitigar de manera proactiva los cuellos de botella asociados a las tasas límite de peticiones concurrentes y optimizar los tiempos de respuesta en las lecturas de las 73 tablas relacionales de la base de datos PostgreSQL; asimismo, resulta imperativo diseñar e implementar microservicios o funciones sin servidor (Edge Functions) para el procesamiento y compresión automatizada de archivos multimedia directamente antes de su almacenamiento definitivo en Cloudflare R2, asegurando así un uso eficiente del ancho de banda y una reducción significativa en los costos de persistencia de datos a gran escala; por otra parte, se aconseja expandir las capacidades funcionales del sistema mediante la incorporación de algoritmos de análisis de datos y matchmaking automatizado programados en Python, con el objetivo de conectar de forma inteligente y

predictiva los portafolios técnicos de los estudiantes con las necesidades emergentes del sector laboral y empresarial; finalmente, con miras a consolidar el impacto institucional de la red social, es indispensable ampliar las estrategias de validación de usabilidad mediante la ejecución de pruebas de aceptación dirigidas rigurosamente a gremios empresariales, reclutadores de talento humano y oficinas de egresados, lo que posibilitará la recolección de retroalimentación cualitativa de alto valor sobre la pertinencia de la interfaz y facilitará la creación de alianzas estratégicas para la inserción laboral de los futuros profesionales en la región.

REFERENCIAS

Alba, M. (2018). *El auge de las redes sociales y su impacto en el entorno académico y profesional*. Editorial Universitaria.

Castells, M. (2006). *La sociedad red: una perspectiva global*. Alianza Editorial.

Congreso de la República de Colombia. (2012, 17 de octubre). *Ley 1581 de 2012. Por la cual se dictan disposiciones generales para la protección de datos personales*. Diario Oficial No. 48.587.

Facebook. (s.f.). *Normas comunitarias y políticas de uso de la plataforma*.
<https://www.facebook.com/communitystandards/>

Flanagan, D. (2020). *JavaScript: The Definitive Guide* (7th ed.). O'Reilly Media.

GitHub. (2025). *Supabase: The open source Firebase alternative* (Version 2.45) [Software].
<https://github.com/supabase/supabase>

Google. (s.f.). *Políticas de privacidad y términos de servicio en entornos colaborativos cloud*.
<https://policies.google.com/privacy>

Pérez, S. A., & González, H. M. (2026). *Desarrollo de una red social colaborativa para estudiantes y profesionales (HIDEON)* [Trabajo de grado, Corporación Universitaria Reformada]. Repositorio Institucional Unireformada.

PostgreSQL Global Development Group. (2024). *PostgreSQL 16.0 Documentation*. PostgreSQL.
<https://www.postgresql.org/docs/16/>

React Open Source Team. (2025). *React: A JavaScript library for building user interfaces* (Version 19.0) [Software]. Meta Open Source. <https://react.dev>

Vite Core Team. (2024). *Vite: Next Generation Frontend Tooling* [Software]. <https://vitejs.dev>