

**EngatiApp: Plataforma web para fortalecer la organización comunitaria y el liderazgo
ciudadano en Engativá**

Jhonatan Díaz Marín

Asesora

Maria Patricia Amortegui Vargas

Universidad Nacional Abierta y a Distancia – UNAD

Escuela de Ciencias Básicas, Tecnología e Ingeniería ECBTI

Tecnología en Desarrollo de Software

2025

Dedicatoria

A mi esposa, Vanessa Alejandra, quien me acompañó en cada madrugada, celebró cada pequeño avance y me sostuvo cuando dudé. Gracias por impulsarme a crecer siempre. Este trabajo es un reflejo de tu apoyo.

Agradecimientos

A mi esposa, Vanessa Alejandra, por su apoyo incondicional. Gracias por entender lo que este Proyecto Aplicado significa para mí, por sostenerme en los días de mayor presión y por recordarme, con tus palabras y tu calma, que valía la pena llegar hasta el final.

A mi tutora, María Patricia Amortegui Vargas, por su guía y confianza. Sus observaciones, oportunas y claras, me ayudaron a encaminar el trabajo y a mejorar cada entrega con criterio y rigor.

A las y los docentes de la UNAD del programa de Tecnología en Desarrollo de Software, por las herramientas y fundamentos que hicieron posible este proyecto, y por la exigencia académica que me impulsó a crecer.

A las y los vecinos y líderes de Engativá, por su disposición y comentarios durante el proceso. Sus aportes dieron sentido práctico a la propuesta y marcaron el rumbo de EngatiApp.

Resumen

Este trabajo presenta EngatiApp, un prototipo web orientado a fortalecer la articulación entre ciudadanía, líderes comunitarios y entidades locales en la localidad de Engativá. A partir del reconocimiento del problema de baja visibilidad y desarticulación de iniciativas comunitarias, se definieron requerimientos funcionales (registro e inicio de sesión, publicación de iniciativas, emisión/consulta de alertas y visualización de estadísticas básicas) y no funcionales (accesibilidad inicial y controles mínimos de protección de datos). Metodológicamente se integraron Investigación-Acción Participativa, Design Thinking y Scrum, lo que permitió iterar desde el entendimiento del contexto hasta un prototipo funcional con trazabilidad entre problema, objetivos y decisiones técnicas.

El desarrollo se implementó sobre tecnologías web y servicios de Firebase; se incorporaron ajustes razonables de accesibilidad (navegación por teclado, foco visible y enlace “Saltar al contenido”) y reglas de seguridad para controlar el acceso por rol. La validación incluyó pruebas funcionales y exploratorias, y se dejó definido un plan de usabilidad/accesibilidad y una métrica base para futuras evaluaciones con una muestra ampliada de usuarios.

Los resultados muestran la viabilidad técnica de EngatiApp y su potencial para centralizar información comunitaria, emitir alertas y apoyar la participación ciudadana. Como trabajo futuro se prioriza ampliar el módulo de administración, profundizar en pruebas piloto con líderes y ciudadanos, y fortalecer la accesibilidad y la protección de datos para avanzar hacia un despliegue productivo.

Palabras clave: participación ciudadana; aplicación web; accesibilidad; Firebase; gobierno local.

Abstract

This study introduces EngatiApp, a web prototype aimed at strengthening coordination among citizens, community leaders, and local institutions in Engativá. Addressing the low visibility and fragmentation of community processes, we defined functional requirements (sign-up/login, initiative publishing, alert issuing/consulting, and basic analytics) and non-functional requirements (initial accessibility measures and basic data-protection controls). The methodology combined Participatory Action Research, Design Thinking, and Scrum, enabling an iterative path from context understanding to a functional prototype with traceability from the problem and objectives to technical choices.

The implementation relied on web technologies and Firebase services; accessibility adjustments (keyboard navigation, visible focus, “Skip to content” link) and security rules for role-based access were incorporated. Validation comprised functional and exploratory tests, and we defined a usability/accessibility plan and baseline metrics for future evaluations with a larger user sample.

Results indicate the technical feasibility of EngatiApp and its potential to centralize community information, issue alerts, and support civic participation. Future work will expand the admin module, deepen pilot testing with leaders and citizens, and reinforce accessibility and data-protection controls towards production deployment.

Keywords: civic participation; web application; accessibility; Firebase; local government.

Tabla de Contenido

Introducción	14
Presentación del Proyecto	15
Objetivos	18
Objetivo General	18
Objetivos Específicos	18
Marco Teórico y Conceptual	19
Fundamentos Sociales y Comunitarios	19
Transformación Digital y TIC en la Gestión Social	21
Metodologías de Innovación y Participación	22
Ciclos de Vida del Software y Modelos de Desarrollo Seguro	23
Modelos Tradicionales: Cascada y Microsoft SDL	23
Modelos Ágiles y Seguridad en el Desarrollo	24
Seguridad en Aplicaciones Comunitarias	24
Protección de Datos Personales (Ley 1581/2012 y Decreto 1377/2013)	25
Diferencia entre Sitio Web y Aplicación Web	26
Arquitecturas Cliente/Servidor y Modelo MVC	26
Computación en la Nube y Firebase	27
Lenguajes y Tecnologías Utilizadas	27
HTML: Estructura Visual del Sistema	27
JavaScript: Interacción Dinámica	28
Lenguajes de Script: Lógica Cliente-Servidor	28

Firebase: Plataforma en la Nube Para Aplicaciones Comunitarias.....	28
Comparación y Justificación Tecnológica	28
Conceptos y Métodos de Usabilidad.....	31
Relevancia en Proyectos Comunitarios	31
Lenguaje Unificado de Modelado (UML)	31
Diseño Metodológico.....	33
Variable Independiente.....	33
Variable Dependiente	34
Dimensiones de Análisis.....	34
Técnicas e Instrumentos de Recolección	34
Investigación – Acción Participativa	36
Metodología Design Thinking	37
Metodología Scrum.....	38
Cronograma de Actividades.....	39
Requerimientos	40
Requerimientos Funcionales (RF)	40
Requerimientos No Funcionales (RNF).....	42
Estado de Cumplimiento.....	44
Matriz de Trazabilidad de Requerimientos.....	45
Arquitectura del Sistema.....	48
Patrones y Principios de Desarrollo.....	48
Principios Aplicados a EngatiApp	48

SOLID, DRY, KISS.	48
MVC como Principio de Separación	48
Patrones Relevantes	49
Antipatrones	49
Evidencias Prácticas en EngatiApp	49
Aplicación Práctica en EngatiApp	49
Plan de Pruebas Piloto con Líderes y Ciudadanos (Engativá)	50
Diseño y Tareas	50
Métricas de Salida	51
Diagrama de arquitectura	51
Diseño del Sistema	53
Justificación Técnica	54
Relación con la Documentación Técnica del Plan de Trabajo	54
Diagrama de Casos de Uso	55
Diagrama de Clases	56
Diagramas de Actividad	57
Actividad – Administrador	58
Actividad – Ciudadano	60
Actividad – Líder Comunitario	61
Diagrama de la base de datos	62
Las Colecciones Principales son las Siguietes	63
Aplicación EngatiApp	65

Propósito y Alcance	65
Mapa de Navegación de EngatiApp	65
Mockups.....	67
Alcance del Prototipo Entregado	79
El prototipo actualmente cubre:	79
Usabilidad y Pruebas del Prototipo EngatiApp	81
Enfoque Metodológico.....	81
Alcance Evaluado en EngatiApp	81
Instrumentos y Procedimiento	82
Alcance de la Validación en esta Fase	82
Evidencias mínimas	83
Métricas y Criterios de Aceptación.....	83
Criterios de Accesibilidad Por Validar	83
Registro y Mejora Continua.....	84
Despliegue del Sistema	95
Trabajos Futuros	96
Conclusiones	98
Referencias Bibliográficas	100

Lista de Tablas

Tabla 1 <i>Comparativo entre PHP + MySQL / AppServ y Firebase</i>	30
Tabla 2 <i>Matriz de Requerimientos (RF/RNF)</i>	45
Tabla 3 <i>Diagramas UML Seleccionados para EngatiApp</i>	53
Tabla 4 <i>Guion de Tareas</i>	85
Tabla 5 <i>Casos de Prueba Funcionales Ejecutados en el Prototipo de EngatiApp</i>	86
Tabla 6 <i>Plan de Métricas para Futuras Pruebas de Usabilidad</i>	89
Tabla 7 <i>Checklist Heurístico</i>	89
Tabla 8 <i>Registro de Hallazgos</i>	92
Tabla 9 <i>Plantilla de Evidencias</i>	93

Lista de Figuras

Figura 1 <i>Cronograma de Actividades</i>	39
Figura 2 <i>Diagrama de Componentes – Frontend / Firebase / Bd en la Nube</i>	52
Figura 3 <i>Diagrama de Casos de Uso de EngatiApp</i>	56
Figura 4 <i>Diagrama de Clases de EngatiApp</i>	57
Figura 5 <i>Diagrama General</i>	58
Figura 6 <i>diagrama de actividades del administrador</i>	59
Figura 7 <i>Diagrama de Actividades del Ciudadano</i>	60
Figura 8 <i>Diagrama de Actividades del Líder Comunitario</i>	61
Figura 9 <i>Esquema de la Base de Datos de EngatiApp en Cloud Firestore</i>	62
Figura 10 <i>Mapa de Navegación de EngatiApp</i>	66
Figura 11 <i>Página Principal de EngatiApp</i>	67
Figura 12 <i>Pantalla de Inicio de EngatiApp</i>	68
Figura 13 <i>Formularios de Registro e Inicio de Sesión</i>	69
Figura 14 <i>Pantalla para Publicación de Iniciativas por Parte del Líder Comunitario</i>	70
Figura 15 <i>Módulo de Visualización y Filtrado de Alertas</i>	71
Figura 16 <i>Módulo de Estadísticas de EngatiApp</i>	72
Figura 17 <i>Enlace Visible a La Política de Tratamiento de Datos Personales</i>	73
Figura 18 <i>Contenido de la Política de Tratamiento de Datos Personales Mostrada en la Aplicación</i>	73
Figura 19 <i>Navegación por Teclado con Foco Visible en los Componentes Interactivos</i>	74
Figura 20 <i>Definición de Texto Alternativo y Nombres Accesibles en Elementos Gráficos Del Dom</i>	75

Figura 21	<i>Dependencias del Proyecto en Package.Json, Evidenciando El Uso De Firebase.....</i>	75
Figura 22	<i>Dependencias del Proyecto en Package.Json, Evidenciando El Uso De Firebase.....</i>	76
Figura 23	<i>Archivo Firebase.Json con la Configuración de Hosting para EngatiApp.....</i>	77
Figura 24	<i>Reglas de Seguridad Definidas en Firestore para el Acceso a los Datos.....</i>	78

Lista de Apéndices

Apéndice A <i>Manual Técnico de EngatiApp</i>	105
Apéndice B <i>Manual de Usuario de EngatiApp</i>	120
Apéndice C <i>Evidencias de Trabajo Comunitario y Validación Exploratoria</i>	137

Introducción

La participación ciudadana y la coordinación entre actores comunitarios son condiciones clave para la gestión territorial. Sin embargo, en Engativá persisten brechas de articulación y visibilidad que dificultan difundir iniciativas, emitir alertas oportunas y acceder a información relevante. En este contexto se plantea EngatiApp, una aplicación web que centraliza la publicación de iniciativas, la gestión de alertas y la consulta de estadísticas básicas, con un enfoque de accesibilidad y protección de datos acorde con las capacidades de un prototipo académico.

El propósito del trabajo es diseñar e implementar un prototipo funcional que demuestre la viabilidad técnica de una plataforma comunitaria para Engativá. Para ello se integran metodologías de Investigación-Acción Participativa, Design Thinking y Scrum, que orientan la identificación de necesidades, la definición de requerimientos, el diseño de arquitectura y módulos, y la elaboración de un plan de pruebas de usabilidad y accesibilidad. La solución se apoya en tecnologías web y servicios de Firebase para autenticación, almacenamiento y despliegue.

El documento se organiza en cuatro bloques: (i) marco teórico y conceptual vinculado a los elementos efectivamente utilizados en el desarrollo; (ii) diseño metodológico, técnicas e instrumentos previstos; (iii) desarrollo y diseño del sistema (requerimientos, arquitectura y modelos UML); y (iv) usabilidad y pruebas, seguidas de trabajos futuros, conclusiones y referencias. Con ello se asegura la trazabilidad entre el problema, los objetivos y las decisiones técnicas que sustentan EngatiApp.

Presentación del Proyecto

Problema de Investigación

En Engativá persiste una baja articulación entre líderes comunitarios, ciudadanos y entidades públicas. A pesar de contar con múltiples organizaciones locales, no existen herramientas digitales efectivas que faciliten la gestión, publicación de iniciativas o acceso a alertas comunitarias. Esta falta de herramientas impide una participación ciudadana activa y oportuna, dificultando la organización local y el seguimiento a procesos colectivos. Se evidencia la necesidad de una plataforma tecnológica accesible, enfocada en promover la cohesión social, la transparencia en la gestión barrial y el liderazgo ciudadano.

Aunque la mayoría de los habitantes dispone de teléfonos inteligentes y conexión a Internet, la información comunitaria sigue fragmentada en canales informales y poco estructurados. Esto provoca que muchas personas no se enteren de convocatorias, actividades o decisiones relevantes para su territorio, mientras que los líderes sociales carecen de mecanismos centralizados para difundir sus iniciativas y recibir retroalimentación. Adicionalmente, no se cuenta con indicadores básicos que permitan conocer el nivel de participación o la respuesta de la comunidad frente a las acciones que se adelantan.

En este escenario, se plantea la necesidad de diseñar y construir una aplicación digital que organice la información comunitaria, facilite la publicación y consulta de iniciativas y alertas, y brinde herramientas mínimas para el seguimiento de la participación ciudadana. El reto consiste en que dicha solución sea técnicamente viable, accesible para diferentes perfiles de alfabetización digital y coherente con las dinámicas organizativas de la localidad.

Pregunta de Investigación

¿De qué manera una aplicación web comunitaria como EngatiApp puede facilitar la gestión, publicación y difusión oportuna de avisos e iniciativas en la localidad de Engativá, con el fin de fortalecer la participación ciudadana activa y la cohesión social en el territorio?

Justificación

La localidad de Engativá cuenta con una población estimada de 819.441 habitantes en 2024, con un crecimiento poblacional sostenido del 0,71% anual desde 2018 (SDP, 2024). A pesar de este crecimiento, enfrenta limitaciones en términos de participación, organización comunitaria y sostenibilidad ambiental.

De acuerdo con datos del IDPAC, solo el 30% de las JAC realizan actividades comunitarias constantes, y aunque el 82% recibió visitas institucionales en 2025, estas acciones no se traducen necesariamente en una estructura organizativa sólida (participacionbogota.gov.co). Esto refleja una débil articulación barrial y escasa incidencia ciudadana.

En cuanto al entorno laboral, la informalidad en Bogotá alcanza el 33%, cifra que tiende a ser mayor en localidades con características similares a Engativá. Esta situación afecta el desarrollo económico, limita el acceso a derechos laborales y reduce la cohesión social (DANE, 2023).

A nivel ambiental, los humedales Jaboque y Juan Amarillo, con una extensión combinada de más de 370 hectáreas, se han visto afectados por escombros, vertimientos y pérdida de biodiversidad, resultado de una baja conciencia ambiental comunitaria (OAB, 2024).

En contraste, el 85,9% de los habitantes urbanos en Bogotá usa Internet y el uso de smartphones es masivo, lo que representa una oportunidad para implementar soluciones digitales

que promuevan la participación, el liderazgo y la sostenibilidad. Experiencias previas como Bogotá Cuidadora y IDPAC App han mostrado que la tecnología puede dinamizar la acción colectiva y mejorar la gestión ciudadana.

Este proyecto propone una solución viable y escalable, orientada a fortalecer capacidades locales, empoderar a los líderes sociales y facilitar el acceso a información útil para la acción comunitaria, en concordancia con los ODS 11 y 16.

Objetivos

Objetivo General

Construir una aplicación de software como apoyo a la gestión comunitaria y el liderazgo ciudadano en Engativá.

Objetivos Específicos

Identificar necesidades comunitarias y funcionalidades requeridas para la gestión digital barrial.

Diseñar la arquitectura y los módulos funcionales de EngatiApp.

Implementar y realiza pruebas de un prototipo funcional con tecnologías web libres.

Diseñar la estrategia de validación de la utilidad de la plataforma.

Marco Teórico y Conceptual

Fundamentos Sociales y Comunitarios

La participación ciudadana en Colombia tiene sus bases en la Constitución Política de 1991, la cual estableció la democracia participativa como principio rector, reconociendo el derecho de los ciudadanos no solo a elegir a sus representantes, sino también a incidir en las decisiones que afectan el interés general (Constitución Política de Colombia, 1991). Este cambio marcó un giro en la relación entre Estado y sociedad, impulsando la creación de mecanismos que garantizan la deliberación pública y el control social.

En este marco, la Ley Estatutaria 1757 de 2015 (Ley de Participación Ciudadana) consolidó los avances normativos en torno a la participación social. La norma se estructura en tres pilares fundamentales: (i) los mecanismos de participación ciudadana, que buscan su uso frecuente y efectivo; (ii) la rendición pública de cuentas y el control social a lo público, que fortalecen el rol del ciudadano como garante de las decisiones administrativas; y (iii) la coordinación y promoción de la participación, a través del Consejo Nacional de Participación como parte constitutiva del Sistema Nacional de Participación (Ministerio del Interior, 2015). Estos pilares son esenciales para comprender cómo proyectos comunitarios con base tecnológica pueden alinearse a las políticas públicas vigentes.

Dentro de los instrumentos concretos de participación, las veedurías ciudadanas constituyen un mecanismo de control social que permite a la sociedad civil vigilar la gestión pública y contribuir al fortalecimiento de la transparencia estatal. Cubides (2021) sostiene que estas se convierten en expresiones prácticas de la democracia participativa, ya que posibilitan la intervención directa de la ciudadanía en los asuntos del Estado y en la protección de los derechos colectivos.

A su vez, el desarrollo comunitario se sustenta en el liderazgo social como un factor clave para dinamizar los procesos colectivos. Según de Izarra, Peña y Sáenz (2020), el liderazgo comunitario enfrenta retos derivados de los paradigmas de la gestión social, pues el líder debe responder a problemáticas sociales, económicas y políticas, representando las capacidades y demandas de la comunidad. Un liderazgo comprometido se convierte en un elemento central para la articulación de esfuerzos entre ciudadanos, organizaciones y Estado

Este liderazgo está estrechamente vinculado al capital social y al compromiso cívico. Putnam (2001) entiende el capital social como el conjunto de redes, normas y confianza que facilitan la cooperación para el beneficio común. A través del compromiso cívico, las personas participan en asociaciones, organizaciones y actividades colectivas que fortalecen la cohesión social y el ejercicio democrático (Revista Internacional de Sociología, 2001). En este sentido, una comunidad con alto capital social tiene mayores capacidades para generar confianza mutua, vigilar a sus instituciones y promover soluciones colectivas a los problemas locales.

En el caso de las comunidades urbanas y rurales de América Latina, la ausencia de una gestión organizacional eficiente ha limitado en muchos casos la consolidación de proyectos comunitarios sostenibles. Sin embargo, investigaciones recientes señalan que, cuando existe un liderazgo comunitario fuerte y un capital social consolidado, los procesos de participación logran mayor legitimidad y eficacia en la resolución de necesidades colectivas (Yagual-Rodríguez & León-Valle, 2023)

Por otra parte, la participación ciudadana debe entenderse hoy bajo el desafío de la inclusión digital. Si bien las leyes y políticas promueven la deliberación, la brecha tecnológica limita el ejercicio real de este derecho en comunidades vulnerables. Experiencias recientes como el uso de plataformas digitales en los presupuestos participativos de Bogotá muestran que las

tecnologías digitales amplían las oportunidades de participación, pero también evidencian la necesidad de garantizar accesibilidad y equidad en el acceso (Checa Pinzón, 2024)

Transformación Digital y TIC en la Gestión Social

La digitalización se ha convertido en un motor clave para la modernización de los procesos sociales y comunitarios, al permitir que las organizaciones ciudadanas y los actores sociales puedan optimizar sus dinámicas de interacción, toma de decisiones y control social. La incorporación de Tecnologías de la Información y la Comunicación (TIC) ha generado nuevas formas de participación y de gestión pública, donde la transparencia y la colaboración entre ciudadanos e instituciones son potenciadas a través de plataformas digitales (Castañeda, 2024).

En este sentido, la transformación digital no solo se limita al uso de herramientas tecnológicas, sino que implica un cambio cultural en la manera en que las comunidades interactúan con el Estado. Como señalan Cauce Montez y Romas Rodríguez (2025), la digitalización ha facilitado modelos más inclusivos de democracia deliberativa, permitiendo a los ciudadanos incidir en decisiones públicas mediante medios virtuales, aunque todavía persisten desafíos vinculados a la desigualdad en el acceso y la brecha digital

Asimismo, la digitalización también ha impactado en áreas sociales sensibles, como el trabajo comunitario. Durante la pandemia por COVID-19, el Trabajo Social debió migrar hacia esquemas digitales, validando el papel de las TIC como herramientas para mantener la cercanía con los ciudadanos y garantizar procesos de acompañamiento, aunque con el riesgo de que estas prácticas se reduzcan a automatismos sin la debida reflexividad crítica (Castañeda, 2024).

Por otro lado, las políticas públicas sobre gobierno digital y participación ciudadana han reconocido la necesidad de fortalecer la infraestructura tecnológica y las competencias digitales de la ciudadanía. En Colombia, la estrategia de Gobierno Digital busca integrar la innovación

tecnológica con la transparencia, la eficiencia y la participación activa de los ciudadanos en la gestión pública (Ministerio de Tecnologías de la Información y las Comunicaciones, 2023). Estas iniciativas son fundamentales para reducir la brecha digital y asegurar que la transformación digital contribuya al fortalecimiento de la democracia participativa.

Finalmente, la transformación digital en la gestión social debe considerar también la inclusión y accesibilidad digital. La implementación de estándares como las Pautas de Accesibilidad para el Contenido Web (WCAG 2.2) permite garantizar que personas con discapacidad o con limitaciones en el acceso puedan ejercer plenamente sus derechos digitales (Aristizábal Ángel, 2025).

Metodologías de Innovación y Participación

El desarrollo del proyecto se fundamenta en un conjunto de metodologías de innovación social y gestión ágil que permiten articular la participación comunitaria con procesos iterativos y centrados en el usuario. Estas metodologías facilitan tanto la identificación de necesidades reales como la construcción de soluciones viables y sostenibles en contextos sociales.

En primer lugar, Design Thinking se presenta como un enfoque centrado en el ser humano, basado en la empatía, la ideación y la experimentación rápida de prototipos. Tim Brown, CEO de IDEO, lo describe como una disciplina que emplea la sensibilidad y los métodos de los diseñadores para alinear las necesidades de las personas con lo tecnológicamente factible y lo económicamente viable (Brown, 2008, citado en Beltrán González, 2022). Este método resulta clave para poner a la comunidad en el centro del proceso de innovación, generando soluciones que surgen de la experiencia directa y de la co-creación.

Por otro lado, la Investigación-Acción Participativa (IAP) constituye una base metodológica orientada a la democratización del conocimiento y al empoderamiento

comunitario. Inspirada en la obra de Orlando Fals Borda, esta metodología busca integrar el compromiso social y científico, promoviendo la producción colectiva de conocimiento con y para las comunidades. Así, la IAP se entiende no solo como una estrategia investigativa, sino como una herramienta transformadora que fortalece la autonomía local y la justicia social (Velásquez et al., 2021).

Finalmente, las metodologías ágiles, como Scrum, se han consolidado como un marco flexible que facilita la organización del trabajo en equipos interdisciplinarios y promueve entregas incrementales de valor. En contextos de innovación social, la aplicación de Scrum fomenta la adaptabilidad, la transparencia y la colaboración, características que son esenciales en proyectos con alta incertidumbre y necesidad de participación activa (Arteche et al., 2025). En este sentido, la agilidad no solo mejora los procesos de gestión técnica, sino que también potencia la innovación al acelerar aprendizajes colectivos y fortalecer el capital relacional entre los actores involucrados.

Ciclos de Vida del Software y Modelos de Desarrollo Seguro

El desarrollo de software requiere de metodologías y marcos de trabajo que aseguren tanto la calidad técnica del producto como la protección de los usuarios que interactúan con él. En los entornos sociales y comunitarios, esta necesidad se hace aún más evidente, dado que los sistemas no solo gestionan datos sensibles, sino que también deben garantizar accesibilidad y confianza.

Modelos Tradicionales: Cascada y Microsoft SDL

Los modelos tradicionales se caracterizan por establecer fases secuenciales y altamente documentadas. El modelo en cascada, uno de los más conocidos, organiza el ciclo de vida en etapas de planificación, análisis, diseño, implementación, pruebas y mantenimiento. Su principal

ventaja es la claridad en las fases, lo que facilita la trazabilidad y la auditoría de los procesos. Sin embargo, su rigidez dificulta la adaptación a cambios, lo que representa un desafío en proyectos sociales donde las necesidades evolucionan rápidamente (Bejarano & Rey, 2020).

En este sentido, el Microsoft Security Development Lifecycle (SDL) propone un enfoque que introduce prácticas de seguridad desde el inicio del ciclo. Sus fases incluyen capacitación, establecimiento de requisitos de seguridad, diseño seguro, implementación con análisis estático, pruebas de seguridad y respuesta ante incidentes. Este modelo permite mitigar riesgos asociados al mal uso de la información y es un referente en la construcción de aplicaciones donde la protección de datos personales es prioritaria (Microsoft, 2023).

Modelos Ágiles y Seguridad en el Desarrollo

La aparición de las metodologías ágiles surge como respuesta a la necesidad de flexibilidad y entreg. Frameworks como Scrum permiten iterar en ciclos cortos, entregar valor incremental y adaptarse a nuevas demandas de los usuarios (Beltrán González, 2022).

En contextos comunitarios, esta combinación resulta fundamental, ya que garantiza un desarrollo ágil y participativo, al mismo tiempo que mantiene altos estándares de seguridad digital. De este modo, la comunidad puede confiar en que los datos recolectados (iniciativas ciudadanas, encuestas o alertas comunitarias) se gestionan de forma transparente y protegida.

Seguridad en Aplicaciones Comunitarias

En plataformas cívicas la seguridad es técnica, jurídica y operativa: protege datos personales, garantiza accesibilidad para la participación y sostiene la confianza mediante trazabilidad y transparencia. Este enfoque se alinea con el marco legal colombiano de protección de datos y con la accesibilidad como condición para el ejercicio de derechos.

Protección de Datos Personales (Ley 1581/2012 y Decreto 1377/2013)

En Colombia, el tratamiento de datos personales se rige por la Ley 1581 de 2012 y el Decreto 1377 de 2013, que establecen principios, derechos y obligaciones para responsables y encargados del tratamiento. Estas normas son especialmente relevantes en aplicaciones comunitarias que recolectan información de ciudadanos, ya que exigen salvaguardar la intimidad, la autodeterminación informativa y la seguridad de la información.

De manera general, pueden distinguirse dos grupos de obligaciones:

Obligaciones Núcleo

Observancia de los principios de tratamiento: legalidad, finalidad, libertad o consentimiento, veracidad o calidad, transparencia, acceso y circulación restringida, seguridad y confidencialidad.

Protección reforzada de datos sensibles y de niñas, niños y adolescentes, lo que implica obtener autorizaciones específicas y limitar el tratamiento a lo estrictamente necesario.

Obligaciones Operativas

Obtención de consentimiento previo, expreso e informado, idealmente por cada finalidad de tratamiento (registro, comunicaciones, estadísticas, etc.).

Disponibilidad de una política de tratamiento de datos personales y de un aviso de privacidad visibles para los titulares desde los primeros puntos de contacto.

Habilitación de mecanismos para que las personas puedan conocer, actualizar, rectificar y suprimir sus datos, así como revocar la autorización otorgada.

Conservación de prueba de las autorizaciones y definición de reglas claras para la transmisión de datos a proveedores tecnológicos (por ejemplo, servicios en la nube).

Aplicaciones Web y Arquitecturas

El desarrollo de aplicaciones web ha evolucionado significativamente desde los primeros sitios estáticos hasta las arquitecturas dinámicas y distribuidas que actualmente sustentan los sistemas digitales modernos. Comprender la diferencia entre un sitio web y una aplicación web es esencial para definir la estructura tecnológica más adecuada a los objetivos de un proyecto de software.

Diferencia entre Sitio Web y Aplicación Web

Un sitio web tradicional está orientado a la presentación de información estática y de navegación unidireccional, generalmente sin interacción avanzada entre usuario y sistema. En contraste, una aplicación web es un sistema interactivo que permite procesar datos, autenticar usuarios, generar contenido dinámico y ofrecer experiencias personalizadas desde el navegador (Franco & Reyes, 2024). Esta última integra componentes de software ejecutados tanto en el lado del cliente como del servidor, estableciendo una comunicación constante mediante peticiones HTTP o servicios RESTful.

Arquitecturas Cliente/Servidor y Modelo MVC

El paradigma cliente/servidor constituye la base de la mayoría de las aplicaciones web. En este modelo, el cliente (navegador) envía solicitudes al servidor, que procesa la información y devuelve una respuesta. Esta estructura permite distribuir responsabilidades y optimizar el rendimiento del sistema (Chancusig, 2025).

En el ámbito del diseño de software, el patrón Modelo-Vista-Controlador (MVC) organiza la aplicación en tres capas principales: el modelo, que gestiona la lógica de negocio y los datos; la vista, que representa la interfaz de usuario; y el controlador, que actúa como intermediario entre ambas. Este patrón mejora la escalabilidad, la reutilización del código y la

separación de responsabilidades, facilitando el mantenimiento del sistema y la incorporación de nuevas funcionalidades (Encinas et al., 2023).

Existen también variantes híbridas, como las arquitecturas de tres capas o las basadas en servicios, donde los datos y las lógicas del sistema se alojan en servidores independientes o en entornos en la nube. Estas aproximaciones fortalecen la seguridad y la disponibilidad del sistema, especialmente en aplicaciones comunitarias con múltiples usuarios simultáneos.

Computación en la Nube y Firebase

La aparición del Cloud Computing transformó la forma de desplegar aplicaciones, ofreciendo escalabilidad, resiliencia y bajo costo de infraestructura. Plataformas como Firebase, desarrollada por Google, integran servicios de autenticación, bases de datos en tiempo real (Firestore), almacenamiento y hosting en un entorno completamente gestionado. Según Encinas et al. (2023), las arquitecturas en la nube permiten desplegar infraestructuras bajo el modelo IaaS (Infraestructura como Servicio) y optimizar recursos mediante virtualización y comunicación entre nodos coordinados, lo que garantiza eficiencia y escalabilidad en los entornos distribuidos.

Lenguajes y Tecnologías Utilizadas

El desarrollo de aplicaciones web requiere la integración de distintos lenguajes de programación y plataformas que permitan definir la estructura visual, la lógica de interacción y la gestión de datos. En este proyecto, se seleccionaron tecnologías que aseguran compatibilidad, escalabilidad, accesibilidad y facilidad de mantenimiento, factores esenciales para aplicaciones de uso comunitario.

HTML: Estructura Visual del Sistema

El lenguaje HTML (HyperText Markup Language) constituye la base estructural de toda aplicación web. Permite definir los elementos que conforman la interfaz visual, como títulos,

formularios, enlaces, imágenes y botones, organizando la información mediante etiquetas jerárquicas. Según Luján (2002), HTML actúa como un lenguaje de marcado que describe la presentación y estructura del contenido, interpretado por el navegador para desplegarlo de forma accesible y coherente para el usuario final.

JavaScript: Interacción Dinámica

El lenguaje JavaScript es el responsable de dotar de dinamismo e interactividad a las páginas web. Sánchez (2001) lo define como un lenguaje embebido dentro del documento HTML que permite modificar elementos del DOM, responder a eventos del usuario y

Lenguajes de Script: Lógica Cliente-Servidor

Los lenguajes de script son aquellos que automatizan tareas y ejecutan instrucciones sin necesidad de compilación previa, facilitando la comunicación entre el cliente (navegador) y el servidor. Luján (2002) explica que un lenguaje de script opera dentro de un contexto determinado —por ejemplo, el navegador o el servidor— y su función es coordinar los procesos lógicos de la aplicación, como la validación de datos o la conexión con la base de datos.

Firebase: Plataforma en la Nube Para Aplicaciones Comunitarias

Firebase, plataforma de desarrollo propiedad de Google, combina herramientas de autenticación, base de datos, almacenamiento y alojamiento web, permitiendo la creación de aplicaciones escalables sin necesidad de infraestructura física. Según Ortiz Cárdenas (2022), los servicios en la nube simplifican el despliegue de sistemas y mejoran la disponibilidad y seguridad de la información, elementos críticos para proyectos sociales y colaborativos.

Comparación y Justificación Tecnológica

Tradicionalmente, muchas aplicaciones web se desarrollaban con PHP como lenguaje de servidor y MySQL como sistema de gestión de base de datos, utilizando entornos integradores

como AppServ, XAMPP o WAMP. Estos paquetes permiten instalar localmente Apache, PHP y MySQL para el desarrollo y prueba de aplicaciones. Si bien son soluciones completas, requieren configuraciones manuales, mantenimiento de servidores locales y actualizaciones de seguridad periódicas (Encinas et al., 2023).

Tabla 1*Comparativo entre PHP + MySQL / AppServ y Firebase*

Criterio	PHP + MySQL / AppServ	Firebase
Infraestructura	Requiere servidor local o hosting independiente.	Totalmente en la nube, administrado por Google.
Seguridad	Depende del mantenimiento manual del servidor.	HTTPS integrado y autenticación segura.
Escalabilidad	Limitada a la capacidad del servidor contratado.	Escalabilidad automática según demanda.
Tiempo de despliegue	Requiere instalación y configuración.	Inmediato mediante Firebase Hosting.
Costo y mantenimiento	Puede requerir costos de hosting y soporte técnico.	Plan gratuito inicial y bajo costo operativo.
Criterio	PHP + MySQL / AppServ	Firebase
Infraestructura	Requiere servidor local o hosting independiente.	Totalmente en la nube, administrado por Google.
Seguridad	Depende del mantenimiento manual del servidor.	HTTPS integrado y autenticación segura.

Tomado de. J. Díaz Marín (2025).

Usabilidad y Pruebas

Conceptos y Métodos de Usabilidad

La evaluación de usabilidad combina inspección, indagación y testing: (a) revisión experta de la interfaz, (b) entrevistas/encuestas y observación para captar necesidades reales y (c) ejecución de tareas representativas con medición de éxito, errores y tiempos (Granollers, 2022). En soluciones educativas y cívicas es pertinente complementar con instrumentos específicos y registros objetivos (bits de interacción), como muestran enfoques que incluyen métodos e instrumentos de usabilidad y la explotación de **logs** para analizar el uso real (Massa, 2017, caps. 5.4, 7 y apéndice D).

Relevancia en Proyectos Comunitarios

En plataformas comunitarias, la usabilidad es necesaria para la participación efectiva y la inclusión. Se priorizan ajustes razonables y eliminación de barreras (p. ej., modo lectura, texto **a voz**, navegación por teclado, semántica/ARIA, contraste), en línea con el mandato de accesibilidad de la Ley 1618 de 2013 (Congreso de la República, 2013). El uso de instrumentos estructurados (inspección heurística y cuestionarios de satisfacción) y evidencias objetivas (logs de uso) refuerza la confianza y orienta mejoras iterativas (Massa, 2017, caps. 7 y apéndice D).

Lenguaje Unificado de Modelado (UML)

El Lenguaje Unificado de Modelado (UML) es una notación de modelado visual que emplea distintos tipos de diagramas para representar la estructura, el comportamiento y las interacciones de un sistema de software. Su propósito es comunicar y documentar de forma clara y verificable las decisiones de análisis y diseño a lo largo del ciclo de vida del proyecto (desde requisitos hasta pruebas y mantenimiento) (Fontela, 2012). En EngatiApp, UML permite asegurar trazabilidad entre lo que se requiere, lo que se diseña (arquitectura web y patrón MVC)

y lo que se implementa (Bejarano & Rey, 2020; Pinzón Núñez, Rodríguez Guerrero, & Vanegas, 2019; Luján, 2002).

Diseño Metodológico

Tipo de Investigación

El proyecto se desarrollará bajo un enfoque metodológico mixto, combinando métodos cualitativos y cuantitativos para abordar tanto aspectos técnicos como sociales. Se integran tres enfoques principales:

Investigación Acción Participativa (IAP): permite la participación activa de la comunidad de Engativá en todas las fases del proyecto, desde la identificación de necesidades hasta la validación del sistema, promoviendo el empoderamiento ciudadano.

Design Thinking: guía el diseño de la solución digital mediante las fases de empatizar, definir, idear, prototipar, probar,

Scrum (Metodología Ágil): organiza el desarrollo del sistema en sprints semanales, priorizando entregables funcionales, retroalimentación continua y mejora progresiva del producto.

Variables

A partir de la pregunta de investigación, se definen las siguientes variables principales.

Variable Independiente

Accesibilidad y uso de la plataforma digital comunitaria (EngatiApp).

Definición conceptual: Grado en que los líderes y ciudadanos de Engativá pueden acceder a la aplicación, comprender su interfaz y utilizar sus funcionalidades básicas (registro, publicación de iniciativas, emisión de alertas y consulta de estadísticas).

Dimensiones de análisis:

Disponibilidad de dispositivo y conexión (smartphone/computador, datos móviles o Wi-Fi).

Facilidad de uso percibida (claridad de la interfaz, navegación, mensajes).

Frecuencia de uso de las funcionalidades principales.

Indicadores posibles: número de usuarios registrados, número de sesiones iniciadas, frecuencia de publicación de iniciativas y alertas por usuario, uso de opciones de accesibilidad.

Variable Dependiente

Participación ciudadana activa en procesos comunitarios.

Definición conceptual: Nivel de involucramiento de los habitantes de Engativá en actividades comunitarias, expresado en su respuesta a convocatorias, interacción con contenidos comunitarios y seguimiento a las iniciativas publicadas.

Dimensiones de Análisis

Respuesta a convocatorias (asistencia estimada a actividades, participación en encuestas o consultas).

Interacción con la información (visualización de iniciativas, alertas y estadísticas).

Permanencia y continuidad (usuarios activos mensuales, reincidencia en la participación).

Indicadores posibles: tasa de respuesta a invitaciones, número de comentarios o interacciones por publicación, número de usuarios activos en un periodo determinado.

Estas variables orientan tanto el diseño de la aplicación como la planeación de las pruebas piloto y los instrumentos de recolección de información, permitiendo analizar si la implementación de EngatiApp contribuye efectivamente a mejorar la gestión y la participación comunitaria en la localidad.

Técnicas e Instrumentos de Recolección

Para el desarrollo del proyecto se emplean técnicas mixtas de recolección de información –encuestas comunitarias, talleres participativos, pruebas de usabilidad exploratorias y análisis

documental– orientadas principalmente al diagnóstico de la comunidad y al diseño del prototipo. La validación ampliada del uso de la plataforma con una muestra más grande de habitantes se proyecta para fases posteriores.

Encuestas comunitarias: e aplicaron mediante formularios en línea (Microsoft Forms) a habitantes del sector, con preguntas cerradas y abiertas sobre participación, canales de comunicación y uso de herramientas digitales. Como evidencia se incluye en el Apéndice C un pantallazo del formulario aplicado.

Taller participativo / diario de sueños: se realizó una sesión de cartografía social con vecinos del barrio, donde se construyó colectivamente un “diario/mapa de sueños” sobre problemas y oportunidades del territorio. La fotografía del producto final se presenta en el Apéndice C

Pruebas de usabilidad exploratorias: se definieron tareas críticas (T1–T4: registro, publicación de iniciativas, gestión de alertas y consulta de estadísticas) y se realizaron recorridos críticos internos sobre el prototipo, identificando problemas de navegación, mensajes de error poco claros y ajustes de texto. Estos hallazgos alimentaron las mejoras descritas en Usabilidad y Pruebas.

Análisis documental: se revisan políticas públicas, diagnósticos locales y normativas relacionadas con participación ciudadana, gobierno digital, accesibilidad y protección de datos personales. Esta revisión sirve de base para definir los requerimientos no funcionales y alinear la aplicación con el marco legal vigente.

Las técnicas definidas (encuestas comunitarias, taller participativo/diario de sueños, pruebas de usabilidad exploratorias y análisis documental) orientaron el diagnóstico y el diseño del prototipo. Dado el alcance de esta versión, la aplicación extensa de encuestas y el

levantamiento sistemático de métricas quedan proyectados para la siguiente fase. Evidencias visuales de la instrumentación se integran en el Apéndice C (captura del formulario, fotografía del diario de sueños y pantallas del prototipo empleadas en las tareas).

Investigación – Acción Participativa

La Investigación Acción Participativa se adopta como enfoque central para vincular a la comunidad de Engativá en el proceso de construcción de EngatiApp. Este enfoque combina la producción de conocimiento con la acción transformadora, a través de ciclos sucesivos de diagnóstico, planificación, ejecución y reflexión colectiva.

En el proyecto se retoman las fases clásicas de la IAP y se adaptan de la siguiente manera.

Diagnóstico participativo: se identifican, junto con líderes y ciudadanos, los principales problemas de comunicación comunitaria, los canales existentes y las brechas de información. Esta fase se apoya en encuestas, talleres y análisis documental.

Planificación de la acción: a partir del diagnóstico se definen los objetivos del sistema, las funcionalidades prioritarias y los criterios de éxito, construyendo de forma conjunta un primer mapa de requerimientos de la aplicación.

Acción o intervención: se desarrolla el prototipo funcional de EngatiApp y se habilitan sus principales módulos (registro, iniciativas, alertas, estadísticas) para ser usados por un grupo piloto de participantes.

Observación y reflexión: se realizan pruebas piloto con líderes y ciudadanos, recogiendo evidencias de uso, percepciones de usabilidad y sugerencias de mejora. Los hallazgos se discuten con la comunidad y se incorporan como insumos para ajustar el sistema.

Metodología Design Thinking

El Design Thinking se emplea como metodología de innovación centrada en las personas para guiar el diseño de EngatiApp. Su aplicación se organiza en cinco fases principales:

Empatizar: se realiza un acercamiento a líderes comunitarios y ciudadanos de Engativá para comprender sus experiencias, problemas y expectativas frente a la gestión comunitaria. Se utilizan entrevistas breves, encuestas y observación de reuniones comunitarias para recoger narrativas y casos concretos.

Definir: con base en la información recopilada se formulan problemas de diseño claros, por ejemplo: “las iniciativas comunitarias no llegan de forma oportuna a todos los vecinos” o “no existe un canal único para centralizar avisos y alertas del barrio”. Esta fase permite priorizar necesidades y delimitar el alcance del prototipo.

Idear: se generan alternativas de solución mediante lluvias de ideas y bocetos de posibles pantallas y funcionalidades. Junto con los participantes se discuten opciones de organización de la información, tipos de alertas, categorías de iniciativas y modos de visualización de estadísticas.

Prototipar: se construye un prototipo de baja y luego de media fidelidad de EngatiApp, donde se representan la estructura de navegación, los roles de usuario (ciudadano, líder, administrador) y los flujos principales de interacción.

Probar: el prototipo se somete a pruebas con usuarios reales, quienes ejecutan tareas específicas y brindan retroalimentación sobre claridad, facilidad de uso y utilidad percibida. Los resultados de estas pruebas se documentan y sirven de base para iterar sobre el diseño y mejorar la experiencia de usuario.

Metodología Scrum

Para organizar el desarrollo técnico de EngatiApp se adopta el marco ágil Scrum, que permite trabajar en iteraciones cortas, priorizar entregables funcionales y ajustar el producto a partir de la retroalimentación obtenida.

El equipo de trabajo se estructura en los siguientes roles:

Product Owner: asume la representación de la comunidad de Engativá, priorizando funcionalidades según las necesidades identificadas con líderes y ciudadanos.

Scrum Master: facilita la aplicación del marco Scrum, remueve impedimentos y vela por la mejora continua del proceso.

Equipo de desarrollo: se encarga del diseño, implementación y pruebas del prototipo, abarcando tanto la parte de frontend como la integración con Firebase.

El proyecto se organiza en sprints de corta duración, al final de los cuales se espera contar con incrementos funcionales del sistema (por ejemplo, un sprint para el módulo de registro, otro para la publicación de iniciativas, otro para alertas y estadísticas).

Los principales artefactos de Scrum empleados son:

Product Backlog: lista priorizada de requerimientos funcionales y no funcionales de EngatiApp.

Sprint Backlog: conjunto de tareas seleccionadas para cada sprint, incluyendo actividades de diseño, codificación, pruebas y documentación.

Incremento: versión funcional del sistema al cierre de cada sprint, lista para ser presentada a los interesados.

Cronograma de Actividades

En la Figura 1 se presenta el cronograma de actividades del proyecto, organizado en fases que reflejan los objetivos específicos planteados: identificación de necesidades y definición de requerimientos, diseño de la arquitectura y del sistema, implementación del prototipo, realización de pruebas piloto y ajustes finales. Cada fase se distribuye en semanas, considerando las iteraciones de Scrum y los espacios de interacción con la comunidad de Engativá para la recolección de información y la validación de la solución.

Figura 1

Cronograma de Actividades

Cronograma de actividades				
Actividad	Mes de inicio	Mes final	Porcentaje del proyecto	{1-5} 10 v
Fase de análisis	1	1	15.00 %	
Fase de diseño	2	2	20.00 %	
Fase de desarrollo	2	3	35.00 %	
Fase de validación	3	4	15.00 %	
Fase de despliegue	4	5	15.00 %	
Total			100.00 %	

Tomado de. J. Díaz Marín (2025).

Requerimientos

Los requerimientos definen “lo que el sistema debe hacer” y “cómo debe ser” para satisfacer objetivos del usuario y restricciones del entorno. Tradicionalmente se clasifican en funcionales (capacidades observables del sistema) y no funcionales (propiedades de calidad y restricciones), y su correcta especificación sirve como acuerdo común entre las partes (Borrell, 2006).

Requerimientos Funcionales (RF)

A partir del problema identificado y de las técnicas de recolección de información aplicadas, se definieron los requerimientos funcionales de EngatiApp, entendidos como las capacidades observables que el sistema debe ofrecer a ciudadanos, líderes y administradores. Estos requerimientos se organizaron en cuatro grupos principales (registro y autenticación, gestión de iniciativas, gestión de alertas y visualización de estadísticas) y se documentan en la tabla correspondiente con su descripción, criterios de aceptación y sustento teórico-técnico. RF1.

Registro e inicio de sesión de usuarios

Descripción: El sistema deberá permitir registro y autenticación de ciudadanos, líderes y administradores.

Criterios de aceptación: Alta de cuenta con verificación básica; Inicio/cierre de sesión; Mensajes claros ante errores; Flujo de recuperación de acceso.

Sustento teórico/técnico: Arquitectura web cliente/servidor (Luján, 2002); separación de responsabilidades en la capa de control (MVC) (Pinzón Núñez, Rodríguez Guerrero, & Vanegas, 2019).

RF2. Publicación y gestión de iniciativas comunitarias

Descripción: El sistema deberá permitir crear, editar y listar iniciativas (título, descripción, categoría y metadatos mínimos).

Criterios de aceptación:

Formulario accesible y validado; 2) Persistencia de datos; 3) Listado con filtros básicos; 4) Confirmación y retroalimentación al usuario.

Sustento: Modelo de aplicación web con lógica de negocio separada (MVC) para mantenibilidad y escalabilidad (Pinzón Núñez et al., 2019; IZERTIS, s. f.).

RF3 Emisión y consulta de alertas comunitarias

Descripción: El sistema deberá permitir registrar alertas (p. ej., informativas, preventivas, urgentes) y consultarlas en tiempo real.

Criterios de aceptación:

Registro con tipificación; 2) Visualización cronológica/por tipo; 3) Indicadores básicos de actividad; 4) Notificación visual en interfaz.

Sustento: Arquitectura web y modularidad (Luján, 2002; IZERTIS, s. f.).

RF4 Visualización de estadísticas básicas

Descripción: El sistema deberá presentar indicadores de participación (ej., número de iniciativas por categoría y alertas por periodo).

Criterios de aceptación:

Panel con métricas mínimas; 2) Actualización al registrar nuevas acciones; 3) Explicación breve de cada indicador.

Sustento: Separación MVC para aislar lógica de agregación y presentación (Pinzón Núñez et al., 2019).

Requerimientos No Funcionales (RNF)

Además de las funcionalidades visibles para el usuario, EngatiApp debe cumplir con ciertas propiedades de calidad relativas a accesibilidad, protección de datos personales, usabilidad, desempeño, mantenibilidad y confiabilidad. Estos requerimientos no funcionales definen “cómo debe ser” el sistema para operar en un contexto comunitario real, alineado con la normativa vigente y con buenas prácticas de ingeniería de software. En la tabla correspondiente se presentan los RNF priorizados, junto con sus criterios de aceptación y las fuentes que los sustentan.

RNF1. Accesibilidad e inclusión

Descripción: La interfaz deberá incorporar ajustes razonables que garanticen el ejercicio de derechos de personas con discapacidad (Ley 1618/2013).

Criterios de aceptación:

a) Navegación por teclado con foco visible; b) Estructura semántica/etiquetado; c) Contraste adecuado; d) Pruebas con al menos un participante con necesidades de accesibilidad.

Sustento: Mandato legal de accesibilidad (Congreso de la República, 2013); método de inspección–indagación–testing para evaluar usabilidad y accesibilidad (Granollers, 2022).

RNF2. Protección de datos y privacidad

Descripción: El tratamiento de datos personales deberá cumplir Ley 1581/2012 y Decreto 1377/2013.

Criterios de aceptación:

a) Política de tratamiento y aviso de privacidad visibles; b) Consentimiento previo, expreso e informado (por finalidad) y prueba de autorización; c) Rutas para conocer/actualizar/rectificar/suprimir datos y revocar.

Sustento: Principios rectores y derechos del titular (Congreso de la República, 2012); operatividad, autorización, avisos y responsabilidad demostrada (Presidencia de la República, 2013).

RNF3. Usabilidad

Descripción: La interfaz deberá ser comprensible y eficiente para perfiles de alfabetización digital heterogéneos.

Criterios de aceptación:

a) se evalúa mediante tareas críticas, checklist, pruebas con usuarios; b) Cero errores bloqueantes; c) Reducción iterativa del tiempo por tarea; d) Instrumentos estructurados (heurística/encuesta).

Sustento: Marco de pruebas de usabilidad (Granollers, 2022) e instrumentación con guías/cuestionarios y explotación de registros de uso (Massa, 2017).

RNF4. Desempeño y disponibilidad

Descripción: La aplicación deberá responder con fluidez y mantener disponibilidad adecuada para picos de uso comunitario.

Criterios de aceptación:

a) Respuesta perceptible en operaciones básicas; b) Estrategias simples de caché/precarga en el cliente; c) Despliegue en infraestructura escalable (p. ej., nube).

Sustento: Arquitectura cliente/servidor y principios de simplicidad/KISS para minimizar latencias innecesarias (Luján, 2002; IZERTIS, s. f.).

RNF5. Mantenibilidad y escalabilidad

Descripción: El código deberá ser modular, legible y testeable.

Criterios de aceptación:

a) Estructura MVC (modelo, vista, controlador); b) Principios SOLID/DRY/KISS; c) Patrones DAO y Observer donde aplique; d) Cobertura mínima de pruebas por capa; e) Reutilización de componentes UI y utilidades.

Sustento: Patrones/principios de diseño profesional (Autentia & IZERTIS, s. f.; Pinzón Núñez et al., 2019).

RNF6. Confiabilidad y trazabilidad

Descripción: El sistema deberá registrar eventos clave y soportar auditorías.

Sustento: Metodología de pruebas apoyada en evidencias objetivas (Massa, 2017) y buenas prácticas de calidad y documentación técnica (IZERTIS, s. f.).

Estado de Cumplimiento

Accesibilidad. Cumplimiento parcial: se incorporó *Saltar al contenido*, foco visible y estructura semántica con <main>; la evaluación con usuarios con discapacidad y la verificación sistemática de contraste quedan proyectadas.

Protección de datos. Cumplimiento parcial: aviso y política visibles, autenticación con Firebase y reglas de Firestore por rol; vistas ARCO y registro formal de consentimientos por finalidad pasan a trabajo futuro.

Trazabilidad/observabilidad. Cumplimiento parcial: existe colección logs para eventos clave (autenticación, creación/edición), mientras que reportes periódicos y auditorías quedan pendientes para la siguiente versión. (Modelado y referencia en Manual Técnico / Arquitectura de datos).

Rendimiento/Disponibilidad. Sin pruebas de carga; al ser app estática con Hosting de Firebase, se asume disponibilidad básica de la plataforma y se documenta el plan para ensayos de estrés y latencia en fases posteriores.

Matriz de Trazabilidad de Requerimientos

Con el fin de garantizar coherencia entre lo que se requiere, lo que se diseña, lo que se implementa y lo que se prueba, se elaboró una matriz de trazabilidad que relaciona los requerimientos funcionales y no funcionales con sus criterios de aceptación, las evidencias de verificación y las fuentes de referencia. Esta matriz permite al jurado y a los actores del proyecto seguir el rastro de cada requerimiento a lo largo del ciclo de vida del sistema, verificando que exista soporte documental y técnico para su cumplimiento.

Tabla 2

Matriz de Requerimientos (RF/RNF) con Criterios de Aceptación, Evidencia y Fuentes

ID	Descripción breve	Criterios de aceptación clave	Evidencia / Verificación	Fuente
RF1	Registro e inicio de sesión de usuarios (ciudadano, líder, admin)	(1) Alta de cuenta; (2) Inicio/cierre de sesión; (3) Recuperación de acceso; (4) Mensajes de error claros y accesibles	Casos de prueba; capturas de flujo; logs de autenticación	Luján (2002); Pinzón Núñez et al. (2019)
RF2	Publicación y gestión de iniciativas	(1) Formulario validado y accesible; (2) Persistencia; (3) Listado con filtros; (4) Confirmación de registro	Casos de uso; evidencias de guardado; pruebas de filtros	Pinzón Núñez et al. (2019); Autentia & IZERTIS (s. f.)

ID	Descripción breve	Criterios de aceptación clave	Evidencia / Verificación	Fuente
RF3	Alertas comunitarias (registro y consulta)	(1) Tipificación (urgente, informativa, preventiva); (2) Orden cronológico/por tipo; (3) Indicadores mínimos; (4) Notificación visual	Registro de alertas; métricas; evidencias UI	Luján (2002); Autentia & IZERTIS (s. f.)
RF4	Estadísticas básicas de participación	(1) Panel de métricas; (2) Actualización tras eventos; (3) Glosario/ayuda del indicador	Capturas del panel; pruebas de actualización; notas de diseño	Pinzón Núñez et al. (2019)
RNF1	Accesibilidad e inclusión	(a) Navegación por teclado y foco visible; (b) Semántica/ARIA; (c) Contraste; (d) Pruebas con usuarios con discapacidad	Checklist de accesibilidad; resultados de testing; incidencias corregidas	Congreso de la República (2013); Granollers (2022)
RNF2	Protección de datos y privacidad	(a) Política y aviso visibles; (b) Consentimiento previo, expreso e informado por finalidad + prueba; (c) Rutas para conocer/actualizar/rectificar/suprimir y revocar; (d) Manejo reforzado de datos sensibles/NNA; (e) Contrato de transmisión con proveedor nube	Bitácora de consentimientos; versión de aviso/política; endpoints de derechos; contrato con proveedor	Congreso de la República (2012); Presidencia de la República (2013)

ID	Descripción breve	Criterios de aceptación clave	Evidencia / Verificación	Fuente
RNF3	Usabilidad	(a) $\geq 80\%$ de éxito en tareas críticas sin asistencia; (b) 0 errores bloqueantes; (c) Reducción del tiempo por tarea; (d) Instrumentos (heurística/encuesta) + logs	Guion de tareas; encuestas; reportes de testing; análisis de logs	Granollers (2022); Massa (2017)
RNF4	Desempeño y disponibilidad	(a) Respuesta fluida en operaciones básicas; (b) Precarga/caché ligera; (c) Despliegue en entorno escalable (nube)	Pruebas de rendimiento; evidencias de hosting; métricas de latencia	Luján (2002); Autentia & IZERTIS (s. f.)
RNF5	Mantenibilidad y escalabilidad	(a) Estructura MVC; (b) Principios SOLID/DRY/KISS; (c) Patrones DAO/Observer; (d) Cobertura mínima por capa; (e) Reutilización de componentes UI	Árbol de proyecto; reglas de lint; reportes de cobertura; PRs revisados	Autentia & IZERTIS (s. f.); Pinzón Núñez et al. (2019)
RNF6	Confiabilidad y trazabilidad	(a) Logs de autenticación/consentimiento/altas-bajas; (b) Trazabilidad de errores y recuperaciones; (c) Reportes periódicos de estado	Configuración de logging; ejemplos de auditoría; informe de incidentes	Massa (2017); Autentia & IZERTIS (s. f.)

Tomado de. J. Díaz Marín (2025).

Arquitectura del Sistema

A partir de los requerimientos definidos y del marco teórico de aplicaciones web y computación en la nube, se diseñó una arquitectura para EngatiApp basada en el modelo cliente/servidor y en el patrón Modelo–Vista–Controlador (MVC). Esta arquitectura distribuye la lógica de presentación en el navegador, la lógica de negocio en controladores y servicios, y la gestión de datos en una base de datos en la nube (Firebase/Firestore), lo que facilita la escalabilidad, la mantenibilidad y la integración de mecanismos de seguridad y accesibilidad. Sobre esta base se aplican distintos principios de diseño y patrones de arquitectura que se describen a continuación.

Patrones y Principios de Desarrollo

Principios Aplicados a EngatiApp

Cuatro reglas del diseño simple (*Four Rules of Simple Design*).

Orientan a mantener el código correcto, claro, sin duplicaciones y con la menor cantidad de elementos necesarios (Rainsberger, s. f.-a; Rainsberger, s. f.-b; Haines, s. f.). En EngatiApp se reflejan en funciones cortas, contratos simples y evitar complejidad innecesaria.

SOLID, DRY, KISS

La práctica profesional de *clean code* promueve clases con una sola responsabilidad, bajo acoplamiento, evitar duplicación (DRY) y soluciones simples (KISS) (Martin, s. f.; Autentia & IZERTIS, s. f.). En EngatiApp: servicios por dominio, utilidades compartidas y revisión de código.

MVC como Principio de Separación

Separar Modelo (datos y reglas), Vista (interfaz accesible) y Controlador (orquestración) mejora mantenibilidad y pruebas (Pinzón Núñez, Rodríguez Guerrero, & Vanegas, 2019).

Patrones Relevantes

MVC (Modelo–Vista–Controlador): estructura base para aislar interfaz, lógica y persistencia; en EngatiApp, controladores JS gestionan flujos (login, publicar iniciativas/alertas, estadísticas) y el modelo encapsula reglas/datos (Pinzón Núñez et al., 2019; Autentia & IZERTIS, s. f.).

DAO (Data Access Object): desacopla la lógica de negocio del detalle de almacenamiento; facilita pruebas con *mocks* y cambios de proveedor sin tocar la lógica (Fowler, s. f.; Alur, Crupi, & Malks, s. f.; Autentia & IZERTIS, s. f.).

Observer: publicación–suscripción para notificar cambios sin acoplar componentes (por ejemplo, alertas y métricas en tiempo real) (Gamma, Helm, Johnson, & Vlissides, s. f.; Freeman & Freeman, s. f.; Autentia & IZERTIS, s. f.).

Antipatrones

Spaghetti Code, The Blob, Golden Hammer, Cut-and-Paste Programming.

Se evitan con separación por capas (MVC), SRP de SOLID, utilidades reutilizables y decisiones justificadas (Autentia & IZERTIS, s. f.; Martin, s. f.).

Evidencias Prácticas en EngatiApp

Estructura por capas: Modelo (reglas/datos), Vista (HTML/CSS accesible), Controlador (JS y eventos) (Pinzón Núñez et al., 2019).

Aplicación Práctica en EngatiApp

La estructura de EngatiApp sigue un esquema MVC adaptado, donde:

Modelo: la base de datos Firestore almacena información de usuarios, alertas, iniciativas y estadísticas.

Vista: se implementa en HTML y CSS, con un diseño centrado en la usabilidad y la accesibilidad, especialmente para población diversa.

Controlador: el código JavaScript intermedia entre la interfaz y la base de datos, gestionando los eventos de usuario y la comunicación con Firebase.

Además, la utilización de Firebase Hosting y Authentication garantiza la seguridad y la integridad de los datos, mientras que el almacenamiento en la nube permite la escalabilidad del sistema sin necesidad de infraestructura física propia. Este modelo responde a las necesidades de una aplicación comunitaria orientada a la participación ciudadana, donde la fiabilidad y la protección de datos son esenciales para generar confianza en los usuarios (Chaves, 2025).

Plan de Pruebas Piloto con Líderes y Ciudadanos (Engativá)

Objetivo. Verificar que EngatiApp permite registrarse, publicar iniciativas, emitir alertas y consultar estadísticas de forma clara, rápida y accesible.

Participantes. ≥ 5 usuarios (líderes y ciudadanos).

Diseño y Tareas

Inspección (equipo): heurística con GEHOA (guía de evaluación heurística) adaptada a los módulos clave.

Testing: tareas representativas (registro, publicación, alerta, estadísticas, activar texto a voz).

Indagación: encuesta breve de satisfacción con ítems inspirados en CUSEOA (cuestionario para usuarios), y entrevista post-uso (percepción de barreras).

Evidencia objetiva: logs de interacción (flujo, errores, pasos, tiempo por tarea) para contrastar con los resultados del testing.

(Instrumentos y uso de logs según Massa, 2017, caps. 7.2.2 “GEHOA”, 7.3.2 “CUSEOA”, y apéndice D). 1Library

Métricas de Salida

Éxito de tareas $\geq 80\%$ sin asistencia; cero errores bloqueantes.

Reducción de tiempo por tarea e incremento de pantallas accesibles (texto a voz/navegación por teclado).

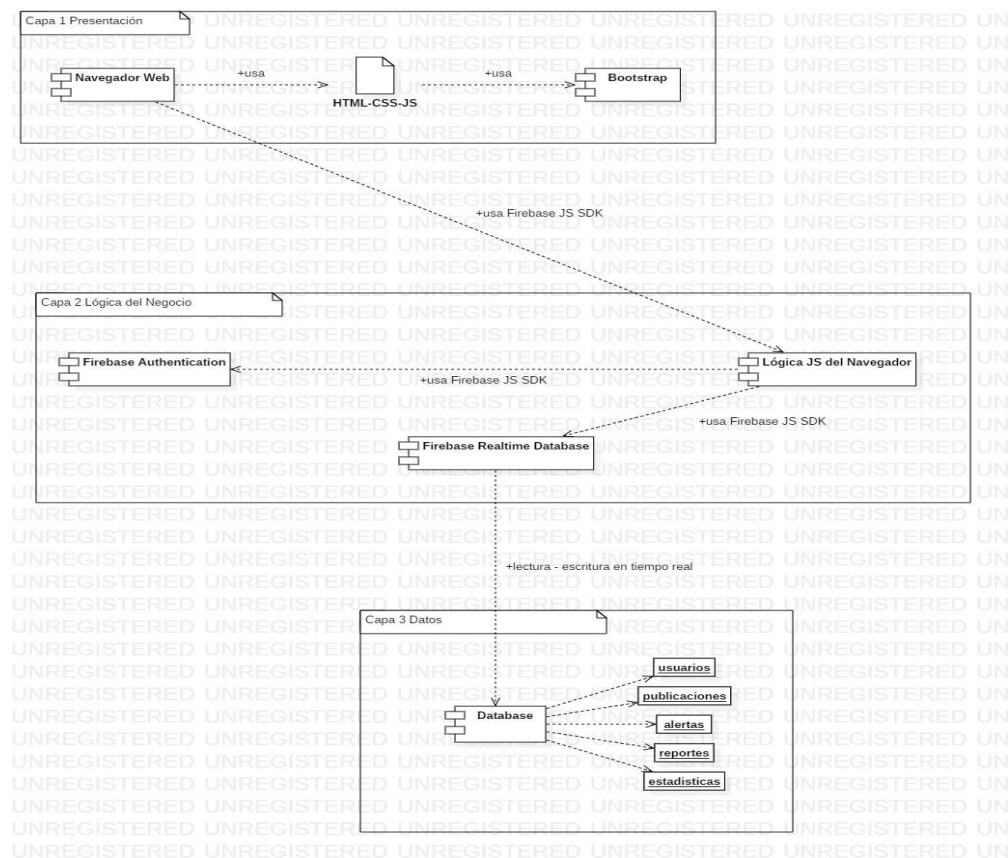
Hallazgos de heurística y satisfacción documentados → backlog de mejoras para el siguiente sprint.

Diagrama de arquitectura

En la Figura 2 se presenta el diagrama de arquitectura de EngatiApp, donde se observan los principales componentes del sistema: el cliente web (vista), los controladores y servicios que gestionan la lógica de negocio, y los servicios de Firebase (Authentication, Firestore y Hosting) que soportan la autenticación de usuarios, el almacenamiento de información y el despliegue en la nube. Este esquema refleja la adopción del patrón MVC y la arquitectura cliente/servidor descritos previamente.

Figura 2

Diagrama de Componentes – Frontend / Firebase / BD en la Nube



Tomado de J. Díaz Marín (2025).

Diseño del Sistema

Alineados con la arquitectura cliente/servidor y el patrón Modelo–Vista–Controlador (MVC) del proyecto, se utilizarán los siguientes diagramas:

Tabla 3

Diagramas UML Seleccionados para EngatiApp

Diagrama UML	Propósito en EngatiApp	Entregable Mínimo
Casos de uso	Capturar funcionalidades desde el punto de vista del usuario (registro, publicación de iniciativas, alertas, estadísticas) y vincularlas a RF/RNF.	1 diagrama general + fichas (actor, pre/postcondiciones, flujo principal y alternos) por caso de uso.
Clases	Modelar la estructura estática del dominio (Usuario, Iniciativa, Alerta, Métrica), atributos, métodos y relaciones, organizado por capas del MVC.	1 diagrama de clases del dominio con paquetes por capa (modelo/datos).
Secuencia	Describir la interacción temporal entre Vista–Controlador–Servicios–DAO–BD para los casos críticos (p. ej., publicar iniciativa, emitir alerta).	≥ 2 diagramas (registro/publicación/alerta) con lifelines y mensajes.
Arquitectura (componentes/despliegue)	Representar la arquitectura web y en la nube (Firebase: Auth, Firestore, Hosting),	1 diagrama de componentes + 1 de despliegue (navegador/servicios/BD nube).

componentes, dependencias y
nodos de ejecución.

Tomado de J. Díaz Marín (2025).

Justificación Técnica

MVC mejora mantenibilidad y escalabilidad al separar responsabilidades (Pinzón Núñez et al., 2019).

La arquitectura web cliente/servidor y sus variantes respaldan la distribución de funciones y la comunicación vía HTTP (Luján, 2002).

La documentación visual con UML facilita revisiones y acuerdos durante el ciclo de vida (Bejarano & Rey, 2020; Fontela, 2012).

Relación con la Documentación Técnica del Plan de Trabajo

El plan de trabajo del proyecto exige la elaboración de diagramas de clases, casos de uso, arquitectura y una memoria técnica. La relación y trazabilidad entre estos artefactos se establece de la siguiente manera:

Casos de uso → Requisitos (RF/RNF): cada caso de uso referencia de forma explícita los requerimientos funcionales y no funcionales que cubre, apoyándose en la matriz de trazabilidad presentada en el capítulo anterior.

Clases → Modelo de dominio/MVC: el diagrama de clases formaliza las entidades y relaciones del dominio (usuarios, iniciativas, alertas, métricas) y se organiza en paquetes por capa del MVC, facilitando la implementación y el mantenimiento.

Diagramas de secuencia → Lógica de orquestación: muestran cómo los controladores coordinan la interacción entre la vista, los servicios y la base de datos para cumplir cada caso de uso crítico, permitiendo identificar puntos de fallo y oportunidades de optimización.

Diagramas de componentes y despliegue → Vista de sistema: representan la topología cliente/servidor y los servicios en la nube usados por EngatiApp, alineados con la arquitectura web descrita en este capítulo.

Memoria técnica → Integración de evidencias: compila los diagramas anteriores, las decisiones de diseño y su alineación con el ciclo de vida y la metodología aplicada, sirviendo como referencia para mantenimiento, futuras extensiones y auditorías de calidad.

La trazabilidad integral de los requisitos se resume en la Tabla 2. En este capítulo se presentan los artefactos de diseño del sistema (diagramas de casos de uso, clases, actividades y componentes) y se describe su relación con los requerimientos funcionales y no funcionales priorizados. Cada figura incluye una etiqueta de cobertura que indica qué requisitos atiende y, cuando es necesario, se remite a la matriz del capítulo anterior para consultar el detalle de estas relaciones.

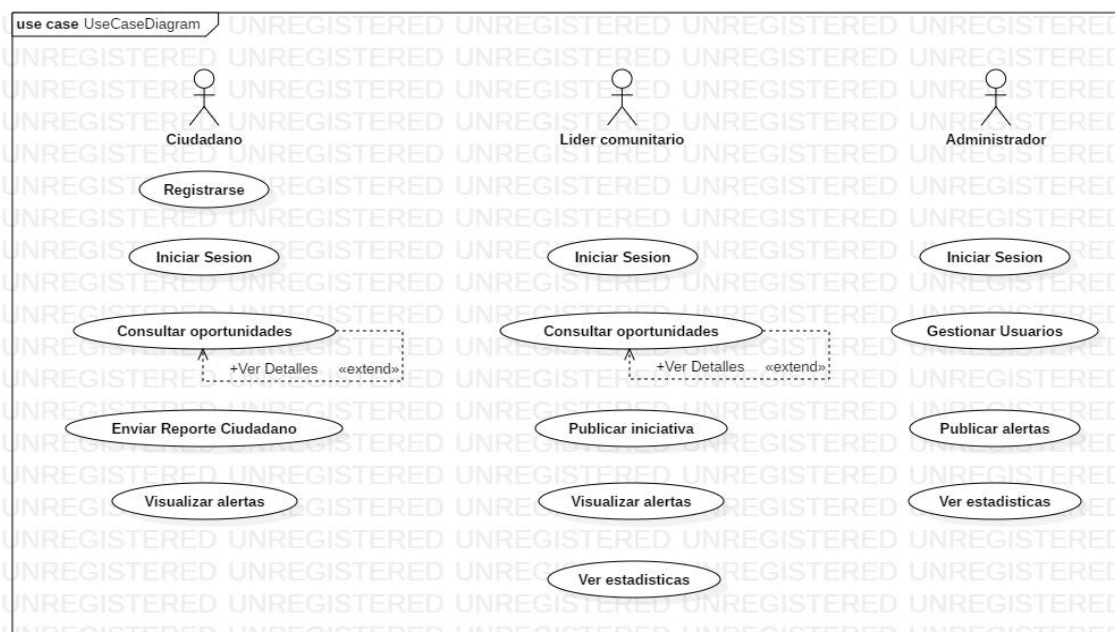
Diagrama de Casos de Uso

El diagrama de casos de uso de EngatiApp muestra la interacción de los tres actores principales del sistema: Ciudadano, Líder comunitario y Administrador. El ciudadano puede registrarse, iniciar sesión, consultar oportunidades disponibles en su sector, enviar reportes y visualizar alertas. El líder comunitario, además de autenticarse, puede consultar oportunidades, publicar iniciativas dirigidas a la comunidad y revisar estadísticas básicas de participación. Por su parte, el administrador gestiona las cuentas de usuario, publica alertas transversales al territorio y accede al módulo de estadísticas. En conjunto, estos casos de uso cubren el flujo

general de la aplicación y materializan los requerimientos funcionales RF1–RF4 definidos en el capítulo anterior.

Figura 3

Diagrama de Casos de Uso de EngatiApp. Actores: Ciudadano, Líder Comunitario, Administrador. COBERTURA: RF1–RF4; RNF–A (Accesibilidad) y RNF–P (Privacidad).



Tomado de. J. Díaz Marín (2025).

Diagrama de Clases

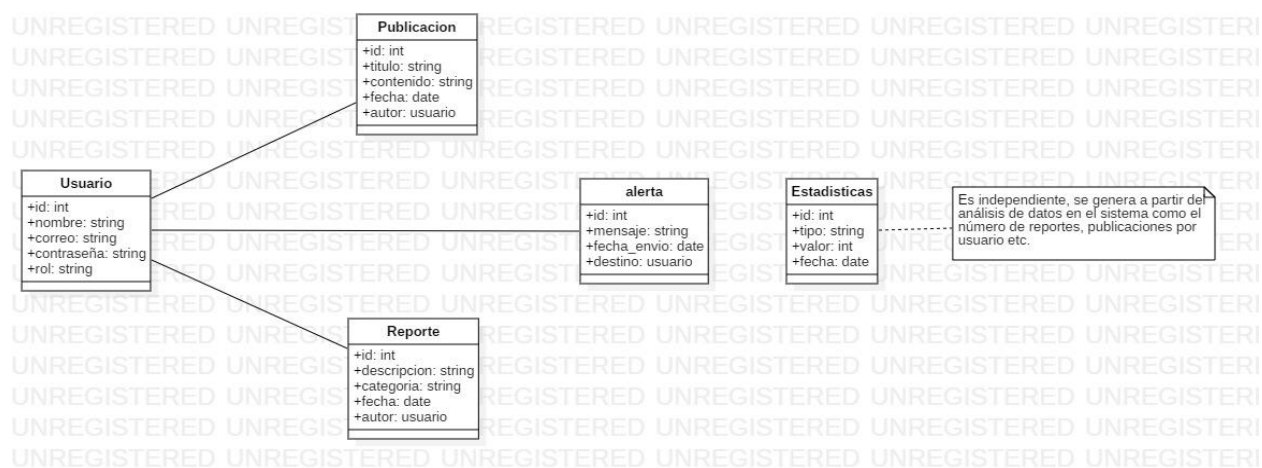
El diagrama de clases representa el modelo de dominio de EngatiApp y las entidades principales que intervienen en la gestión comunitaria. La clase Usuario almacena información básica de los participantes (identificador, nombre, correo, contraseña cifrada y rol), mientras que la clase Publicación agrupa las iniciativas creadas por los líderes comunitarios, incluyendo título, contenido, categoría, fecha y referencia al autor. La clase Reporte reúne los reportes ciudadanos asociados a una publicación o a una situación específica del barrio, y la clase Alerta registra

avisos relevantes emitidos por el administrador. Finalmente, la clase Estadística consolida métricas derivadas de las colecciones anteriores (número de usuarios, iniciativas, alertas, entre otras), que luego se visualizan en el módulo de estadísticas. Este modelo sirve de base tanto para la estructura de la base de datos como para la definición de las interfaces del sistema.

Figura 4

Diagrama de Clases de EngatiApp. Entidades: Usuario, Publicación, Reporte, Alerta y Estadística. Cobertura: RF2 (Publicar Iniciativas), RF3 (Alertas) y RF4 (Estadísticas). Véase

Tabla 2 para las Relaciones RF/RNF ↔ Entidades.



Tomado de J. Díaz Marín (2025).

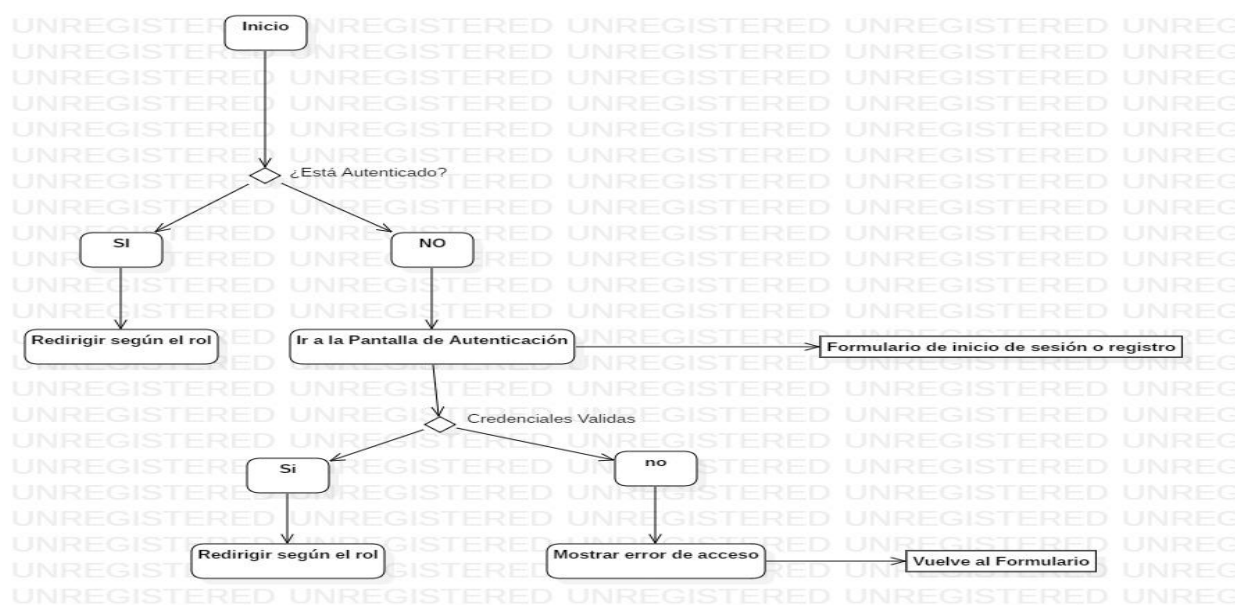
Diagramas de Actividad

El diagrama de actividad general representa el flujo de autenticación en EngatiApp desde la pantalla de inicio. Cuando el usuario accede al sistema, se verifica primero si ya se encuentra autenticado. Si la respuesta es afirmativa, la aplicación lo redirige directamente a la vista correspondiente a su rol (ciudadano, líder comunitario o administrador). Si no está autenticado, se le conduce a la pantalla de inicio de sesión o registro, donde diligencia sus credenciales. En caso de que estas sean válidas, el flujo continúa con la redirección según el rol; si son

incorrectas, se muestra un mensaje de error de acceso y se devuelve al formulario para permitir un nuevo intento. Este diagrama resume el comportamiento esperado del sistema frente al acceso de cualquier tipo de usuario y concretiza el requerimiento funcional RF1.

Figura 5

Diagrama General COBERTURA: RF1 (Registro/INICIO de Sesión); RNF-P (Consentimiento, Avisos y Trazabilidad Básica). Véase Tabla 2. para el Mapeo de RF1 y RNF-P a este Flujo.



Tomado de J. Díaz Marín (2025).

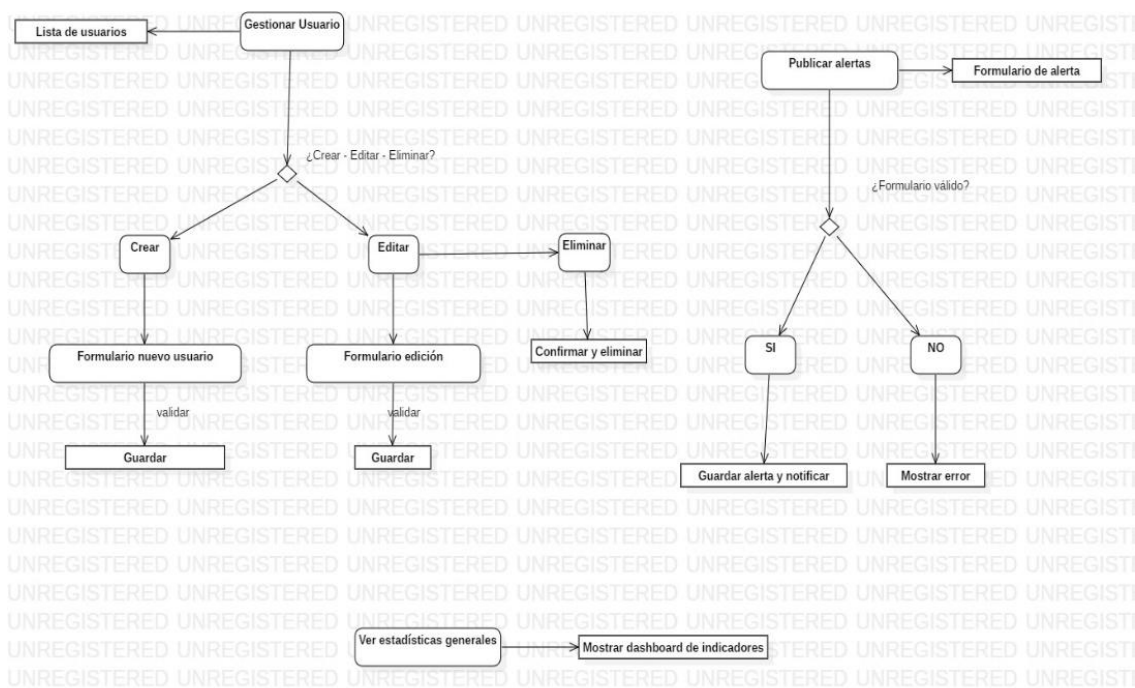
Actividad – Administrador

El diagrama de actividades del Administrador describe el flujo de trabajo asociado a la gestión de usuarios, la publicación de alertas y la consulta de indicadores. El proceso inicia con el acceso al listado de usuarios, desde donde el administrador puede crear nuevas cuentas, editar información existente o eliminar registros, con pasos intermedios de confirmación para evitar acciones accidentales. De forma complementaria, el administrador puede diligenciar el

formulario de alertas, validar los datos ingresados y publicar avisos que quedarán visibles para el resto de los usuarios autenticados. Finalmente, tiene la opción de visualizar las estadísticas generales del sistema mediante un tablero de indicadores.

Figura 6

Diagrama de Actividades del Administrador. Cobertura: RF1 (Gestión de Cuentas), RF3 (Publicar Alertas) y RF4 (ver Estadísticas); RNF-L (Registro Mínimo de Eventos) y RNF-P (Privacidad).



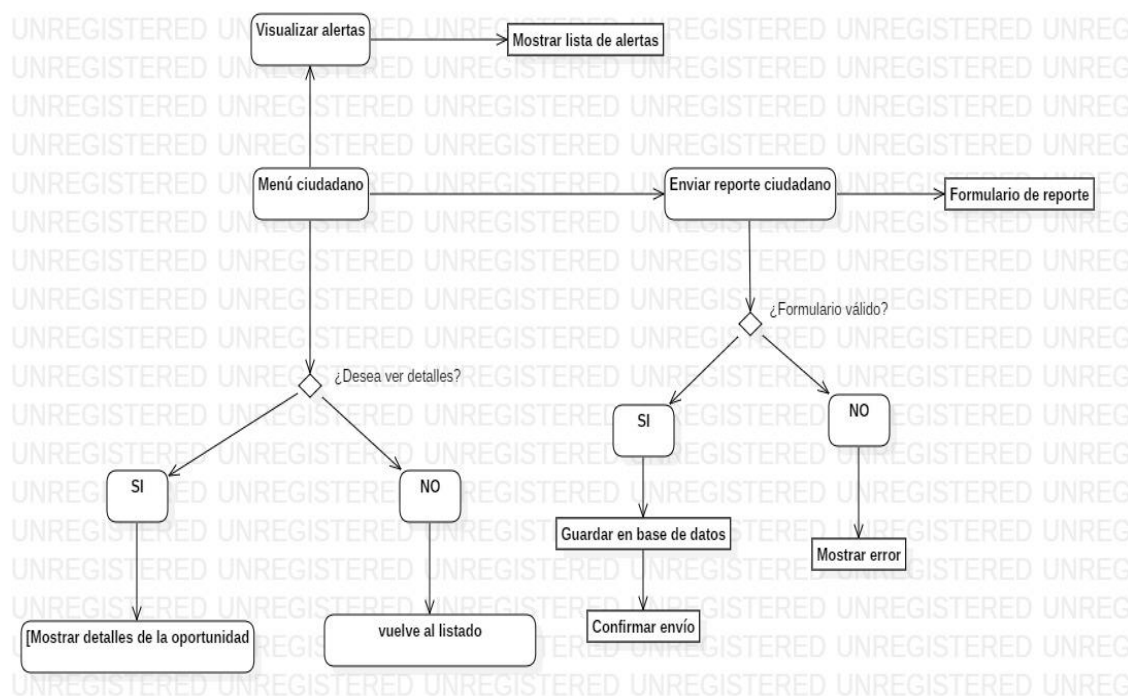
Tomado de J. Díaz Marín (2025).

Actividad – Ciudadano

El diagrama de actividades del Ciudadano describe cómo un usuario autenticado consulta información y participa en la plataforma. El flujo comienza con el inicio de sesión y continúa con la navegación hacia el módulo de oportunidades o alertas. Desde allí, el ciudadano puede filtrar la información por categoría o sector, visualizar el detalle de iniciativas, enviar reportes ciudadanos y revisar alertas activas e históricas. El diagrama contempla además la posibilidad de regresar a la pantalla principal y cerrar la sesión, asegurando un ciclo de uso sencillo y coherente con los escenarios definidos en los requerimientos funcionales.

Figura 7

Diagrama de Actividades del Ciudadano. COBERTURA: RF1 (Inicio de Sesión), RF2 (Consulta de Iniciativas), RF3 (Consulta de Alertas) y RF4 (Visualización de Estadísticas Básicas).



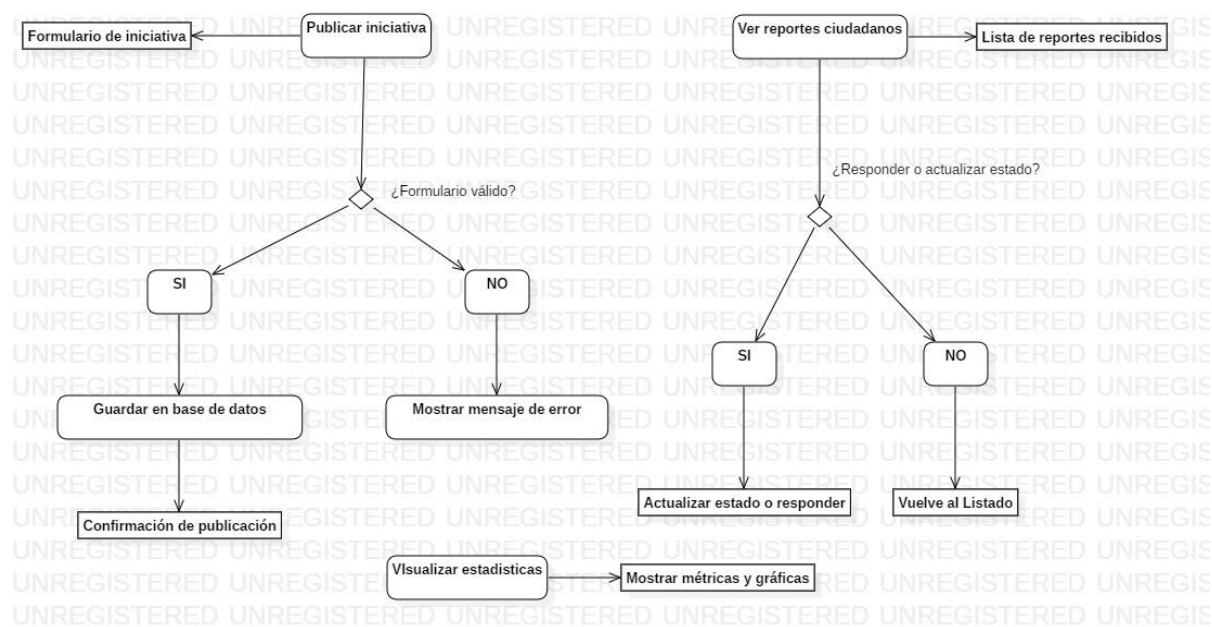
Tomado de J. Díaz Marín (2025).

Actividad – Líder Comunitario

El diagrama de actividades del Líder comunitario se centra en la creación y gestión de iniciativas. Tras iniciar sesión, el líder accede al módulo de iniciativas, donde puede diligenciar el formulario con título, descripción, categoría y sectores a los que va dirigida la propuesta. El flujo contempla la validación de campos obligatorios, la confirmación de la publicación y la actualización del listado de iniciativas visibles para la comunidad. Adicionalmente, el líder puede revisar las estadísticas relacionadas con sus publicaciones para conocer el nivel de actividad e interacción de los ciudadanos.

Figura 8

Diagrama de Actividades del líder Comunitario. Cobertura: RF2 (Publicar Iniciativas) y RF4 (ver Estadísticas por Iniciativas).



Nota. Véase Tabla 2 Tabla 2

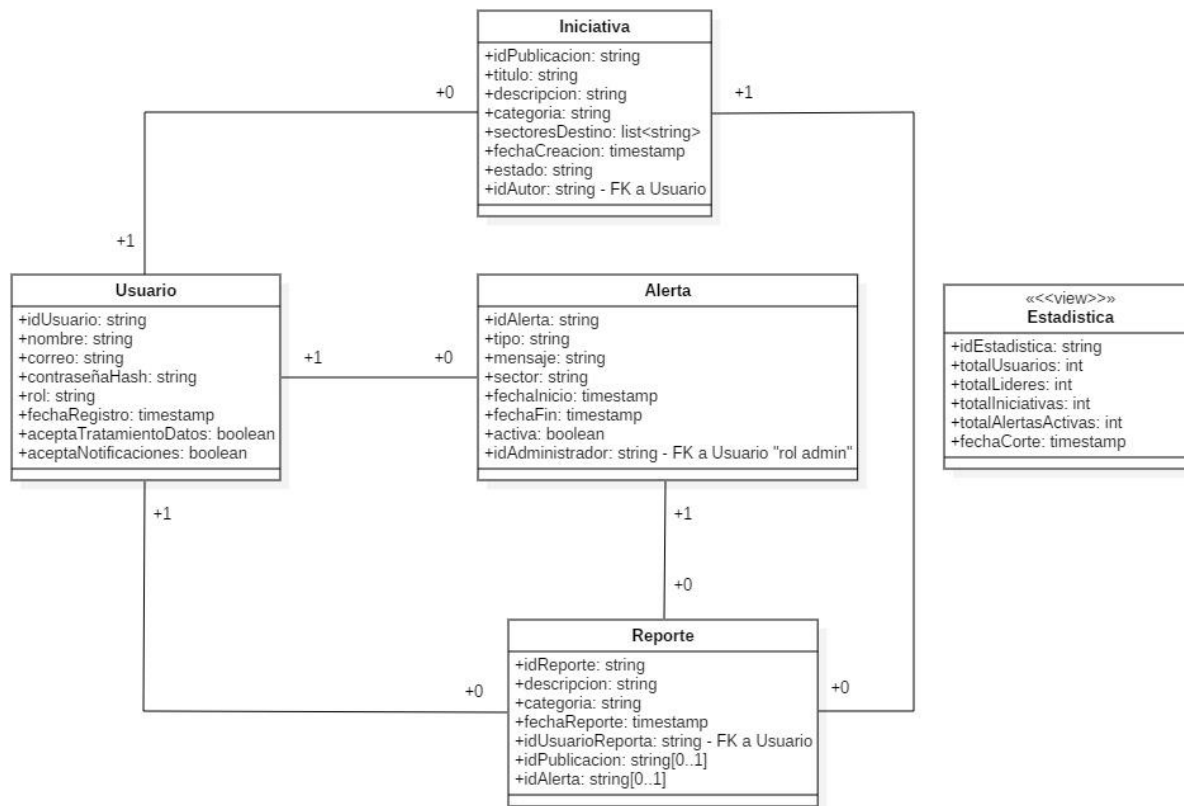
Matriz de Requerimientos (RF/RNF) con Criterios de Aceptación, Evidencia y Fuentes para Detalle de Mapeo RF/RNF ↔ Actividades. Tomado de J. Díaz Marín (2025).

Diagrama de la Base de Datos

EngatiApp utiliza una base de datos NoSQL basada en Cloud Firestore, organizada en colecciones y documentos en lugar de tablas relacionales tradicionales. La estructura se alinea con el modelo de clases del sistema y con los módulos funcionales de la aplicación.

Figura 9

Esquema de la Base de Datos de EngatiApp en Cloud Firestore: Colecciones Usuarios, Publicaciones, Alertas, Reportes y Estadísticas



Tomado de J. Díaz Marín (2025).

Las Colecciones Principales son las Siguietes

usuarios: almacena la información básica de cada cuenta, con campos como idUsuario, nombre, correo, rol (ciudadano, líder, administrador), sector, fechaRegistro y banderas de consentimiento (aceptaTratamientoDatos, aceptaNotificaciones).

publicaciones (iniciativas): registra las iniciativas creadas por los líderes comunitarios, incluyendo idPublicacion, titulo, descripcion, categoria, sectoresDestino (lista de sectores), fechaCreacion, estado y el identificador del autor (idAutor, que referencia a un documento de la colección usuarios).

alertas: contiene las alertas comunitarias gestionadas por el administrador, con campos como idAlerta, tipo, mensaje, sector, fechaInicio, fechaFin y activa.

reportes: agrupa los reportes ciudadanos asociados a una alerta o iniciativa, con campos idReporte, descripcion, categoria, fechaReporte, idUsuarioReporta y, opcionalmente, idAlerta o idPublicacion para establecer la relación.

estadísticas: consolida indicadores derivados de las colecciones anteriores, tales como totalUsuarios, totalLideres, totalIniciativas, totalAlertasActivas y fechaCorte, que luego se utilizan en el módulo de estadísticas y en la página principal de EngatiApp.

Las relaciones entre estas colecciones se implementan mediante identificadores de referencia: por ejemplo, cada iniciativa guarda el idAutor que apunta a un usuario líder, y los reportes almacenan el idUsuarioReporta y, cuando aplica, el idAlerta o el idPublicacion a la que están asociados. En el caso específico de la colección estadísticas, no se definen asociaciones directas con las demás entidades porque sus datos se generan de forma agregada a partir de las colecciones usuarios, publicaciones, alertas y reportes. Estadística funciona como una vista o resumen materializado que consolida indicadores (por ejemplo, total de usuarios, iniciativas

registradas y alertas activas); en el diagrama esta relación se representa mediante dependencias (flechas discontinuas) desde Estadística hacia las colecciones de origen, resaltando que su contenido es derivado y no almacena claves foráneas explícitas.

El diagrama de la Figura 9 resume esta estructura de Firestore, mostrando las colecciones, sus campos principales y las relaciones lógicas que permiten consultar y cruzar la información dentro de EngatiApp.

Aplicación EngatiApp

Propósito y Alcance

EngatiApp es una aplicación web orientada a fortalecer la organización comunitaria y el liderazgo ciudadano en Engativá. Esta sección describe los módulos implementados, los roles y permisos, y la navegación de la interfaz. Las capturas ilustran la versión funcional según el alcance definido en los RF/RNF.

Módulos

Autenticación (RF1): registro/inicio de sesión por rol.

Iniciativas (RF2): creación y consulta (líder).

Alertas (RF3): emisión (admin) y visualización (todos).

Estadísticas (RF4): indicadores para líder/admin.

Roles y Permisos

Ciudadano: visualizar alertas, enviar reporte ciudadano.

Líder comunitario: publicar iniciativas, ver estadísticas de sus iniciativas.

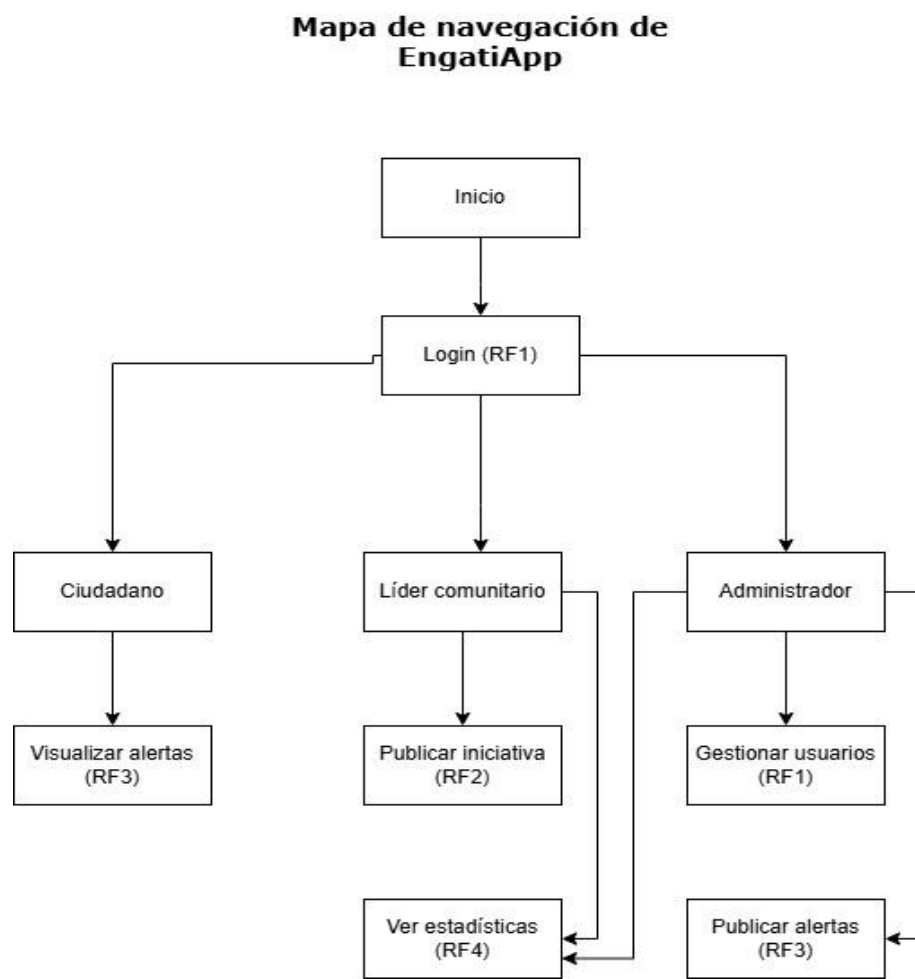
Administrador: gestionar usuarios, publicar alertas, ver estadísticas globales.

Mapa de Navegación de EngatiApp

En la Figura 10 se presenta el mapa de navegación de EngatiApp, donde se observa la relación entre la página de inicio, los módulos de registro/inicio de sesión, iniciativas, alertas y estadísticas. Este esquema resume los flujos principales de interacción para los roles de ciudadano, líder comunitario y administrador, y sirve como guía para la organización de las pantallas descritas en las secciones siguientes.

Figura 10

Mapa de Navegación de EngatiApp. Cobertura: RF1–RF4.



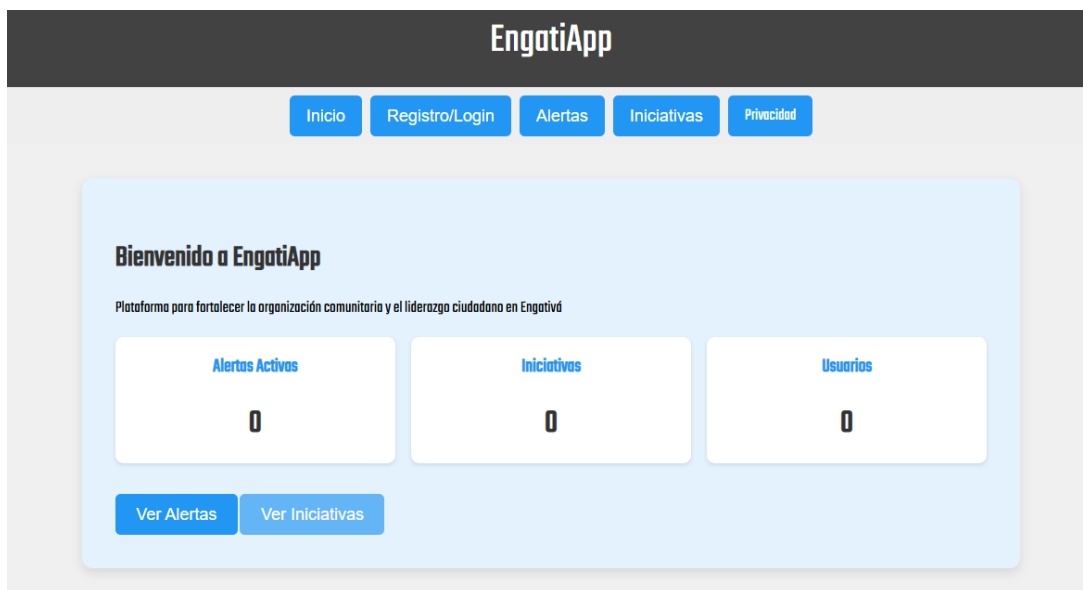
Tomado de. J. Díaz Marín (2025).

Mockups

La página principal de EngatiApp sirve como punto de entrada para todos los usuarios. En la parte superior se ubica una barra de navegación con acceso directo a los módulos de Inicio, Registro/Login, Alertas, Iniciativas y Privacidad. En el área central se presenta un mensaje de bienvenida que resume el propósito de la plataforma como herramienta para fortalecer la organización comunitaria y el liderazgo ciudadano en Engativá. Justo debajo se muestran tres tarjetas con indicadores resumidos de alertas activas, iniciativas y usuarios, que permiten tener un panorama rápido del estado del sistema. Finalmente, los botones “Ver Alertas” y “Ver Iniciativas” facilitan el acceso inmediato a los módulos más utilizados, guiando al usuario hacia las acciones principales de la aplicación.

Figura 11

Página Principal de EngatiApp con Barra de Navegación, Indicadores de Alertas, Iniciativas y Usuarios, y Accesos Directos a los Módulos de Alertas e Iniciativas.

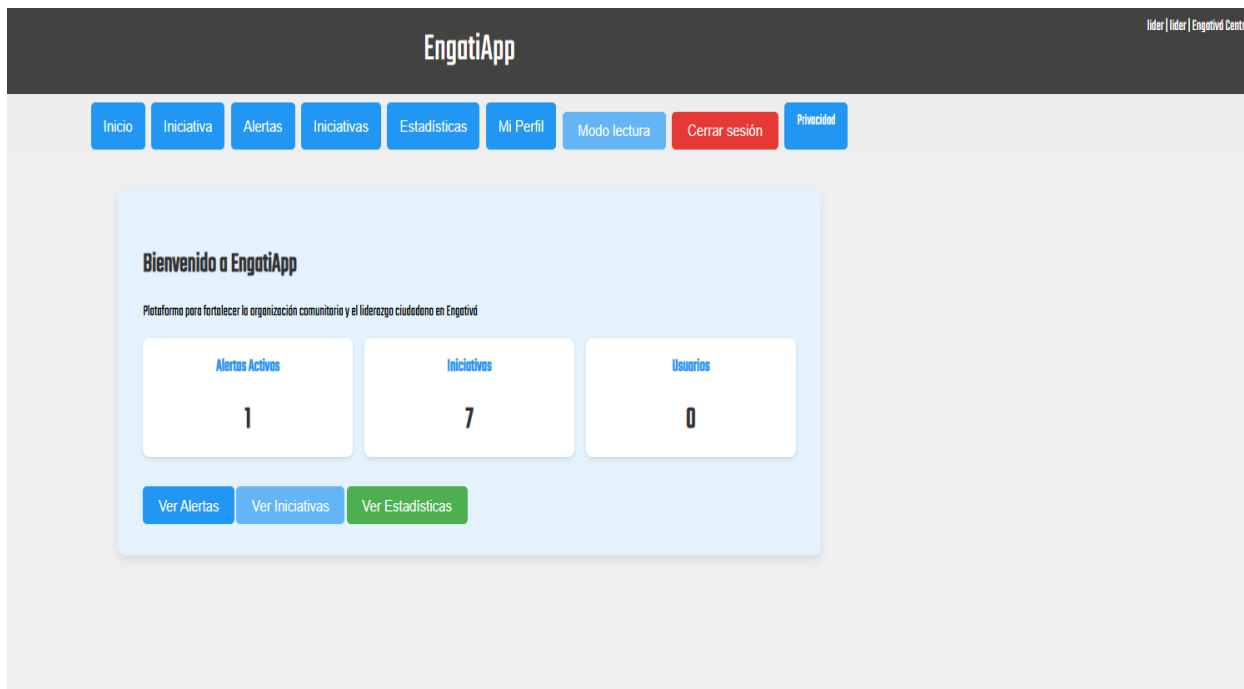


Tomado de J. Díaz Marín (2025).

La Figura 12 muestra la pantalla de inicio de EngatiApp, desde la cual se accede a los módulos principales de la aplicación y se presentan de forma resumida las opciones de navegación disponibles para cada tipo de usuario.

Figura 12

Pantalla de Inicio de EngatiApp. Cobertura: RF4..



Tomado de. J. Díaz Marín (2025).

En la Figura 13 se ilustra la pantalla de registro e inicio de sesión, que permite crear nuevas cuentas de usuario y autenticar a ciudadanos, líderes y administradores de acuerdo con el requerimiento funcional RF1.

Figura 13

Formularios de Registro e Inicio de Sesión. Cobertura: RFI (Registro/Login).

The image shows a web interface for 'EngotiApp'. At the top, there is a dark header with the app name 'EngotiApp' in white. Below the header, there are five blue navigation buttons: 'Inicio', 'Registro/Login', 'Alertas', 'Iniciativas', and 'Privacidad'. The main content area is a light green box titled 'Registro / Inicio de sesión'. It contains several input fields: 'Correo electrónico:' with the value 'lider@gmail.com', 'Contraseña:' with masked characters '*****', 'Nombre completo:', 'Sector/Barrio:' with a dropdown menu showing 'Engativá Centro', and 'Rol:' with a dropdown menu showing 'Ciudadano'. Below these fields is a 'Consentimiento de tratamiento de datos' section with a title and a sub-header 'Marca las finalidades permitidas. Puedes cambiarlos luego en "Mi Perfil"'. It lists five items, each with a checked checkbox: 'Publicación de iniciativas', 'Emisión/gestión de alertas (solo admin)', 'Envío de reportes ciudadanos', and 'Métricas y estadísticas de uso (anónimas)'. The 'Emisión/gestión de alertas' item has a corresponding input field. At the bottom of the form, there are three buttons: 'Iniciar sesión' (green), 'Registrarse' (orange), and '¿Olvidaste tu contraseña?' (blue).

Tomado de. J. Díaz Marín (2025).

La Figura 14 presenta el formulario de publicación de iniciativas para los líderes comunitarios, donde se diligencian campos como título, descripción, categoría y sectores a los que va dirigida la propuesta.

Figura 14

Pantalla para Publicación de Iniciativas por Parte del Líder Comunitario. Cobertura: RF2.

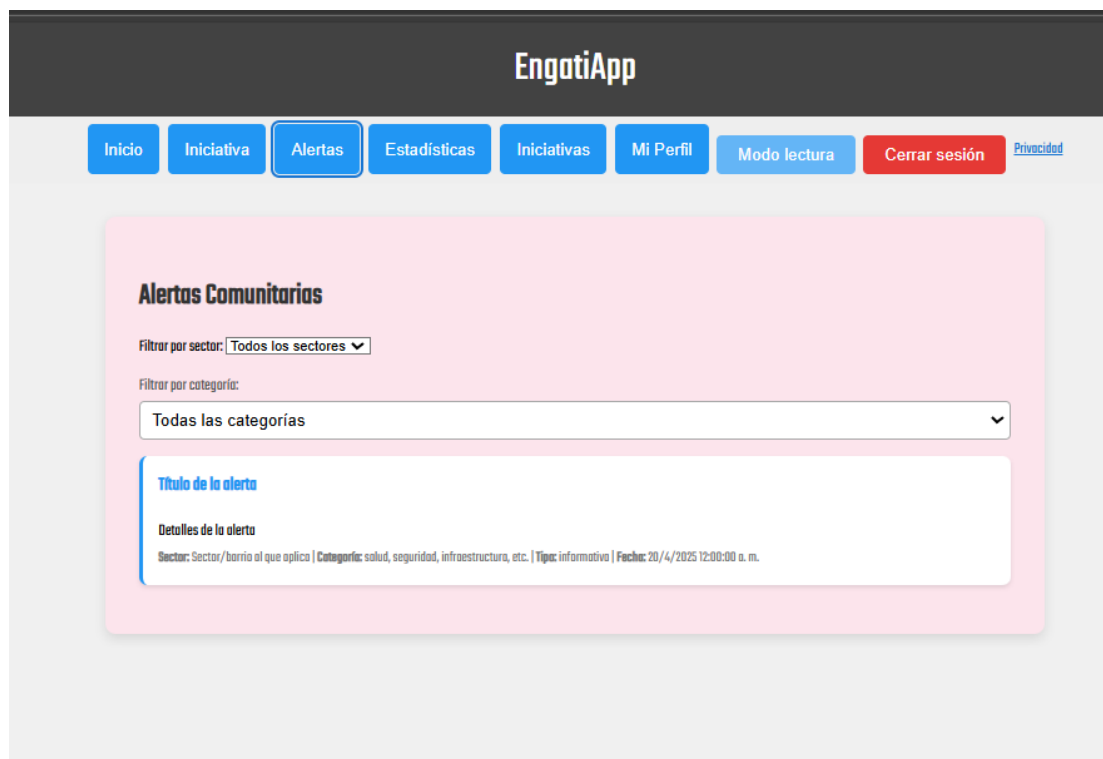
The screenshot shows the EngotiApp interface for publishing a community initiative. At the top, there is a dark header with the app name 'EngotiApp'. Below the header is a navigation bar with buttons for 'Inicio', 'Iniciativa', 'Alertas', 'Iniciativas', 'Estadísticas', 'Mi Perfil', 'Modo lectura', 'Cerrar sesión', and 'Privacidad'. The main content area is titled 'Publicar Iniciativa Comunitario' and features a green success message: 'Iniciativa publicada exitosamente'. Below this, there are several form fields: 'Título:' with the example text 'Ej: Jornada de reciclaje en el parque central'; 'Descripción:' with the placeholder text 'Describe la iniciativa...'; 'Categoría:' with a dropdown menu set to 'Educación'; 'Sector objetivo:' with a list of options including 'Sancti Spiritus', 'La Española', 'Bolivia', 'Villa Amalia', and 'Jardín Botánico'; and 'Fecha de realización:'.

Tomado de J. Díaz Marín (2025).

En la Figura 15 se observa el módulo de alertas, que permite consultar y filtrar los avisos activos según categoría y sector, así como acceder al historial de alertas comunitarias.

Figura 15

Módulo de visualización y Filtrado de Alertas. Cobertura: RF3.

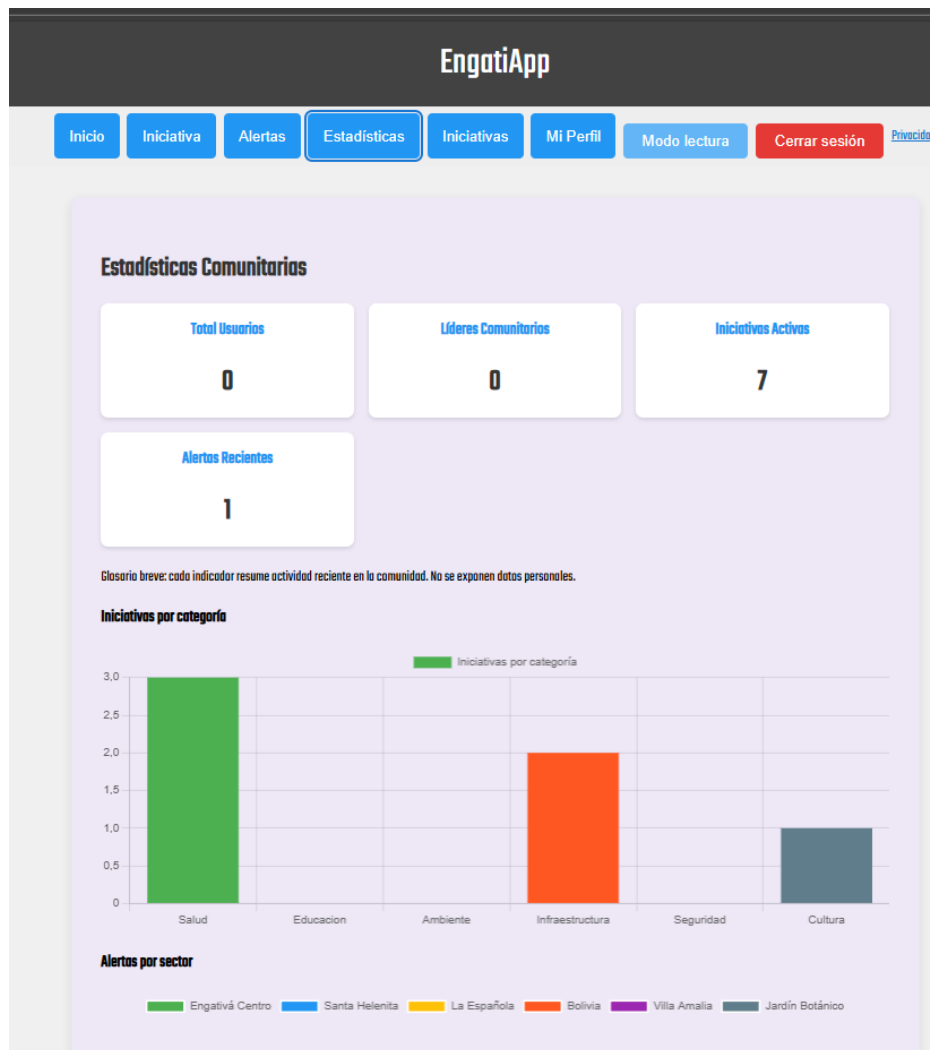


Tomado de. Jhonatan, D.

La Figura 16 muestra el módulo de estadísticas, donde se presentan indicadores básicos sobre usuarios, iniciativas y alertas mediante tarjetas numéricas y gráficos.

Figura 16

Módulo de Estadísticas de EngatiApp. Cobertura: RF4

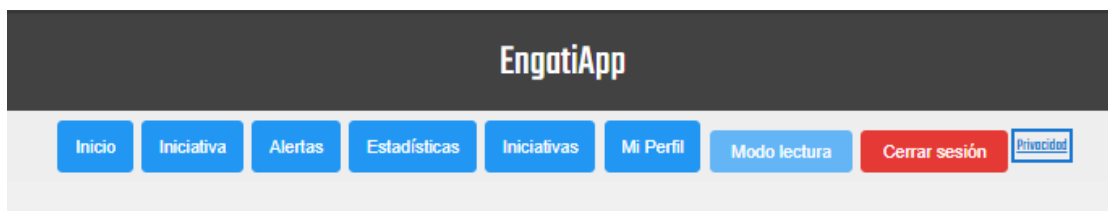


Tomado de J. Díaz Marín (2025).

La Figura 17 evidencia la presencia del aviso y enlace a la política de tratamiento de datos personales, accesible desde la interfaz principal para informar a los usuarios sobre el uso de su información.

Figura 17

Enlace Visible a la Política de Tratamiento de Datos Personales. Cobertura: RNF–Privacidad



Tomado de J. Díaz Marín (2025).

En la Figura 18 se aprecia un fragmento del contenido de la política de tratamiento de datos, donde se especifican finalidades, derechos de los titulares y canales de contacto.

Figura 18

Contenido de la Política de Tratamiento de Datos Personales Mostrada en la Aplicación.

Cobertura: RNF–Privacidad.

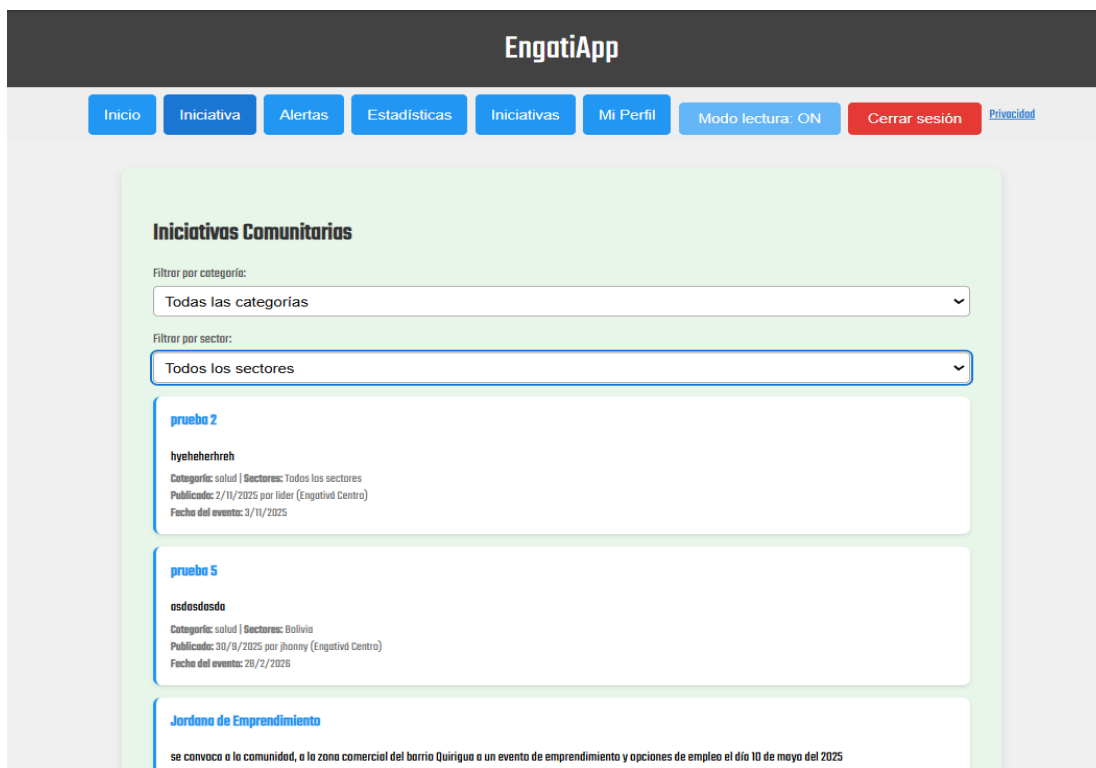


Tomado de J. Díaz Marín (2025).

La Figura 19 ilustra la navegación por teclado en EngatiApp, destacando el foco visible sobre los elementos interactivos como parte de los ajustes razonables de accesibilidad implementados.

Figura 19

Navegación por Teclado con Foco Visible en los Componentes interactivos. Cobertura: RNF–Accesibilidad.



Tomado de. J. Díaz Marín (2025).

En la Figura 20 se muestra, desde las herramientas de desarrollo del navegador, cómo se definen los nombres accesibles y atributos necesarios para que los elementos gráficos sean interpretables por tecnologías de apoyo.

Figura 20

Definición de Texto Alternativo y Nombres Accesibles en Elementos Gráficos del DOM.

Cobertura: RNF–Accesibilidad.

```

</div>
2
<p>Glosario breve: cada indicador resume actividad reciente en la comunidad. No se exponen datos personales.</p>
3
4
5 <h3>Iniciativas por categoría</h3>
6 <div class="chart-container"><canvas id="iniciativasChart" aria-label="Gráfico de iniciativas por categoría"></canvas></div>
7
8 <h3>Alertas por sector</h3>
9 <div class="chart-container"><canvas id="alertasChart" aria-label="Gráfico de alertas por sector"></canvas></div>
10
11 <h3>Usuarios por sector</h3>
12 <div class="chart-container"><canvas id="usuariosChart" aria-label="Gráfico de usuarios por sector"></canvas></div>
13 </div>

```

Tomado de J. Díaz Marín (2025).

La Figura 21 presenta el archivo package.json, donde se listan las dependencias principales del proyecto, incluyendo los paquetes de Firebase utilizados para la autenticación, la base de datos y el alojamiento.

Figura 21

Dependencias del Proyecto en Package.json, Evidenciando el uso de Firebase. Cobertura: RNF–Plataforma en la Nube

```

{} package.json > ...
1  {
2    "dependencies": {
3      "firebase": "^11.6.0"
4    }
5  }
6

```

Tomado de J. Díaz Marín (2025).

En la Figura 22 se observan los paquetes instalados registrados en package-lock.json, lo cual evidencia las versiones específicas de las librerías empleadas durante el desarrollo y despliegue del prototipo.

Figura 22

*Dependencias del Proyecto en Package.json, Evidenciando el uso de Firebase. Cobertura: RNF–
Plataforma en la Nube*

```

805 },
806 "node_modules/firebase": {
807   "version": "11.6.0",
808   "resolved": "https://registry.npmjs.org/firebase/-/firebase-11.6.0.tgz",
809   "integrity": "sha512-Xqm6j6zszIEmISnW1MPR8yTafORTSrW3mW69Lk9e1CjtQQ5iTEkKZiNtUm9y6XFOP18xoF1TNpxZe8HjgA00g==",
810   "license": "Apache-2.0",
811   "dependencies": [
812     "@firebase/analytics": "0.10.12",
813     "@firebase/analytics-compat": "0.2.18",
814     "@firebase/app": "0.11.4",
815     "@firebase/app-check": "0.8.13",
816     "@firebase/app-check-compat": "0.3.20",
817     "@firebase/app-compat": "0.2.53",
818     "@firebase/app-types": "0.9.3",
819     "@firebase/auth": "1.10.0",
820     "@firebase/auth-compat": "0.5.20",
821     "@firebase/data-connect": "0.3.3",
822     "@firebase/database": "1.0.14",
823     "@firebase/database-compat": "2.0.5",
824     "@firebase/firestore": "4.7.10",
825     "@firebase/firestore-compat": "0.3.45",
826     "@firebase/functions": "0.12.3",
827     "@firebase/functions-compat": "0.3.20",
828     "@firebase/installations": "0.6.13",
829     "@firebase/installations-compat": "0.2.13",
830     "@firebase/messaging": "0.12.17",
831     "@firebase/messaging-compat": "0.2.17",
832     "@firebase/performance": "0.7.2",
833     "@firebase/performance-compat": "0.2.15",
834     "@firebase/remote-config": "0.6.0",
835     "@firebase/remote-config-compat": "0.2.13",
836     "@firebase/storage": "0.13.7",
837     "@firebase/storage-compat": "0.3.17",
838     "@firebase/util": "1.11.0",
839     "@firebase/vertexai": "1.2.1"
840   ]
841 },

```

Tomado de J. Díaz Marín (2025).

La Figura 23 muestra la configuración de despliegue definida en el archivo firebase.json, donde se especifica la carpeta pública, las opciones de hosting y la relación con Firestore.

Figura 23

*Archivo Firebase.json con la Configuración de Hosting para EngatiApp. Cobertura: RNF–
Despliegue.*

```
firebase.json > ...
1  {
2    "hosting": {
3      "public": ".",
4      "ignore": [
5        "firebase.json",
6        "**/*.*",
7        "**/node_modules/**"
8      ],
9      "cleanUrls": true
10   },
11   "firestore": {
12     "rules": "firestore.rules",
13     "indexes": "firestore.indexes.json"
14   }
15 }
```

Tomado de. J. Díaz Marín (2025).

Finalmente, la Figura 24 presenta las reglas de seguridad configuradas en Firestore para restringir el acceso a las colecciones de la base de datos según el estado de autenticación del usuario y su rol en el sistema.

Figura 24

Reglas de Seguridad Definidas en Firestore para el Acceso a los Datos de EngatiApp.

Cobertura: RNF–Seguridad de Datos.

```

94  /* ===== reportes (Reporte ciudadano) ===== */
95  match /reportes/{docId} {
96    // Visibles para autor y admin; el resto NO.
97    allow read: if signedIn() && ( isAdmin() || resource.data.autorId == request.auth.uid );
98
99    allow create: if signedIn()
100     && request.resource.data.keys().hasAll(['titulo', 'descripcion', 'categoria', 'sector', 'fecha', 'estado', 'autorId', 'autorNombre'])
101     && request.resource.data.autorId == request.auth.uid
102     && isShortString(request.resource.data.titulo, 140)
103     && isShortString(request.resource.data.descripcion, 1000)
104     && request.resource.data.categoria in ['salud', 'educacion', 'ambiente', 'infraestructura', 'seguridad', 'convivencia']
105     && request.resource.data.estado in ['pendiente', 'revisado', 'convertido'];
106
107    // Autor puede editar mientras esté 'pendiente'; admin puede editar siempre
108    allow update: if signedIn() && (
109      isAdmin() ||
110      ( resource.data.autorId == request.auth.uid && resource.data.estado == 'pendiente' )
111    )
112     && ( !request.resource.data.keys().hasAny(['autorId']) || request.resource.data.autorId == resource.data.autorId )
113     && ( !request.resource.data.keys().hasAny(['titulo']) || isShortString(request.resource.data.titulo, 140) )
114     && ( !request.resource.data.keys().hasAny(['descripcion']) || isShortString(request.resource.data.descripcion, 1000) )
115     && ( !request.resource.data.keys().hasAny(['categoria']) || request.resource.data.categoria in ['salud', 'educacion', 'ambiente', 'infraestructura', 'seguridad', 'convivencia'] )
116     && ( !request.resource.data.keys().hasAny(['estado']) || request.resource.data.estado in ['pendiente', 'revisado', 'convertido'] );
117
118    allow delete: if signedIn() && ( isAdmin() || resource.data.autorId == request.auth.uid );
119  }
120
121  /* ===== usuarios ===== */
122  match /usuarios/{userId} {
123    allow create: if signedIn()
124     && request.auth.uid == userId
125     && request.resource.data.keys().hasAny(['rol'])
126     && request.resource.data.rol in ['ciudadano', 'lider']
127     && ( !request.resource.data.keys().hasAny(['nombre']) || isShortString(request.resource.data.nombre, 120) )
128     && ( !request.resource.data.keys().hasAny(['sector']) || isShortString(request.resource.data.sector, 80) );
129
130    allow read: if signedIn() && ( request.auth.uid == userId || isAdmin() );
131
132    allow update: if signedIn() && (
133      isAdmin() || (
134        request.auth.uid == userId
135        && ( !request.resource.data.keys().hasAny(['rol'])
136          || request.resource.data.rol == resource.data.rol )
137      )
138    )
139     && ( !request.resource.data.keys().hasAny(['nombre']) || isShortString(request.resource.data.nombre, 120) )
140     && ( !request.resource.data.keys().hasAny(['sector']) || isShortString(request.resource.data.sector, 80) );
141
142    allow delete: if signedIn() && request.auth.uid == userId;
143  }
144
145  /* ===== logs ===== */
146  match /logs/{docId} {
147    allow create: if signedIn();
148    allow read, update, delete: if isAdmin();
149  }
150
151  /* ===== consentimientos ===== */
152  match /consentimientos/{docId} {
153    allow create: if signedIn();
154    allow read, update, delete: if isAdmin();
155  }
156
157  /* ===== Denegar todo lo demás ===== */
158  match /{document=**} { allow read, write: if false; }

```

Tomado de J. Díaz Marín (2025).

Alcance del Prototipo Entregado

Este documento corresponde a la entrega final del Proyecto Aplicado para la Tecnología en Desarrollo de Software. En él se integran el planteamiento del problema, los requerimientos funcionales (RF1–RF4) y no funcionales priorizados, el diseño y la implementación del prototipo web (HTML/CSS/JavaScript con integración a Firebase) y las evidencias técnicas presentadas en las figuras y tablas (mapa de navegación, pantallas principales, controles de accesibilidad, configuración de hosting y reglas de seguridad de Firestore).

El prototipo actualmente cubre:

RF1: registro e inicio de sesión con almacenamiento del perfil básico y consentimientos mínimos.

RF2: publicación y visualización de iniciativas por parte de líderes comunitarios.

RF3: consulta y emisión administrada de alertas comunitarias.

RF4: visualización de estadísticas descriptivas a partir de los datos almacenados.

Adicionalmente, se incluyeron un aviso y enlace a la política de tratamiento de datos, un enlace “Saltar al contenido” y foco de teclado visible como ajustes razonables de accesibilidad, así como reglas de Firestore acordes al alcance académico del proyecto.

En cuanto a usabilidad y pruebas, se diseñó un plan de evaluación detallado (Tabla 4 Guion de Tareas (T1–T4) a Tabla 9 Plantilla de Evidencias (Figuras/Capturas)) que combina inspección heurística, ejecución de tareas representativas y registro de hallazgos. En esta entrega no se reportan métricas empíricas a gran escala; la aplicación de pruebas piloto con un grupo más amplio de ciudadanos y el análisis cuantitativo de participación se plantean como parte del trabajo futuro.

Finalmente, el prototipo se configuró para un entorno de demostración y no para un despliegue productivo. La evolución hacia un servicio en producción, con mayor robustez en seguridad, monitoreo y escalabilidad, se deja abierta para versiones posteriores de EngatiApp.

Usabilidad y Pruebas del Prototipo EngatiApp

Enfoque Metodológico

La evaluación de EngatiApp se plantea desde un enfoque mixto que combina inspección experta, testing con tareas e indagación con usuarios. El objetivo es comprobar, hasta donde lo permite el alcance del prototipo, el cumplimiento de los requerimientos funcionales RF1–RF4 y de los ajustes razonables de accesibilidad definidos en los requerimientos no funcionales.

En particular, se consideraron tres niveles de análisis:

Inspección heurística: revisión sistemática de la interfaz a partir de una lista de chequeo basada en heurísticas de usabilidad y accesibilidad web.

Testing con tareas: ejecución guiada de tareas críticas representativas del uso real de la aplicación.

Indagación con usuarios: recolección de percepciones mediante una breve encuesta de satisfacción y preguntas abiertas sobre barreras y oportunidades de mejora.

Alcance Evaluado en EngatiApp

Las tareas seleccionadas se alinean directamente con los requerimientos funcionales del sistema y corresponden a los flujos que se espera que utilicen con mayor frecuencia ciudadanos, líderes y administradores:

T1: Registro e inicio de sesión (RF1).

T2: Publicación de una iniciativa por parte de un líder comunitario (RF2).

T3: Emisión y consulta de alertas (RF3).

T4: Visualización de estadísticas básicas (RF4).

El guion de tareas, con sus objetivos, pasos sugeridos y criterios de éxito, se presenta en la Tabla 4.

Instrumentos y Procedimiento

Para cada tarea del guion se definieron los siguientes instrumentos:

Lista de chequeo heurístico: resume los criterios de usabilidad y accesibilidad que debían verificarse en los módulos clave (registro, iniciativas, alertas y estadísticas). Esta lista se detalla en la Tomado de. J. Díaz Marín (2025).

Tabla 7

Checklist Heurístico (Usabilidad y Accesibilidad).

Casos de prueba funcionales (CP01–CP07): describen los flujos críticos del prototipo (registro, inicio de sesión, publicación de iniciativas, gestión de alertas, visualización de estadísticas y cierre de sesión), indicando datos de entrada, resultado esperado, resultado obtenido, estado y tipo de prueba. Se consignan en la Tabla 5

Casos de Prueba Funcionales Ejecutados en el Prototipo de EngatiApp

Plantilla de métricas: destinada a registrar intentos, éxitos, tiempos por tarea y observaciones, tal como se muestra en la Tabla 6

Plan de Métricas para Futuras Pruebas de Usabilidad.

Plantilla de hallazgos: estructura los problemas encontrados, su severidad y las acciones propuestas Tabla 8

Registro de Hallazgos (se Completa con Datos de Usuarios Simulados)..

Plantilla de evidencias: permite asociar cada captura de pantalla o figura con el módulo evaluado, el requisito al que corresponde y la ubicación en el documento Tabla 9

Plantilla de Evidencias (Figuras/Capturas).

Alcance de la Validación en esta Fase

La validación se concentró en pruebas exploratorias y recorridos críticos sobre el prototipo, focalizadas en los flujos T1–T4 (registro/inicio de sesión, publicación de iniciativas, emisión/consulta de alertas y estadísticas). No se levantaron métricas cuantitativas con una muestra amplia, por lo que los porcentajes de éxito y tiempos de tarea definidos en el plan se mantienen como instrumentos para fases posteriores. Este enfoque es consistente con el objetivo

de verificar viabilidad técnica y detectar mejoras tempranas de navegación, mensajes y textos, dejando la prueba piloto ampliada para trabajo futuro con líderes y ciudadanía.

Evidencias Mínimas

Se adjuntan capturas del prototipo en los módulos clave (registro, iniciativas, alertas y estadísticas) y el plan de métricas (tabla de tareas con campos de intentos, éxitos y tiempos), como plantilla sin resultados para su aplicación en la siguiente iteración.

Métricas y Criterios de Aceptación

Aunque la muestra de usuarios es reducida y de carácter exploratorio, se definieron criterios de aceptación orientadores para futuras pruebas piloto:

Éxito por tarea: al menos el 80 % de las ejecuciones de T1–T4 deberían completarse sin asistencia y sin errores bloqueantes.

Eficiencia: el tiempo promedio por tarea debería disminuir entre iteraciones de ajuste.

Accesibilidad: todos los controles deben ser accesibles vía teclado, mantener un foco visible y respetar el orden lógico de navegación.

Las métricas propuestas (intentos, éxitos, tiempos y observaciones) se estructuran en la Tabla 6 como plan de medición. La recolección de datos con una muestra mayor de líderes y ciudadanos se plantea para una siguiente fase del proyecto.

Criterios de Accesibilidad Por Validar

Se priorizan ajustes razonables coherentes con el front-end HTML/CSS/JS implementado:

Navegación por teclado: todos los controles deben ser accesibles y activables vía teclado; evitar trampas de foco.

Orden de foco y atajos: secuencia lógica del foco; atajos (si existen) sin interferir con el navegador.

Contraste y estados visibles: suficiente contraste en elementos interactivos y foco visible.

“Saltar al contenido” y titulado de páginas: enlace de salto y títulos descriptivos.

Reajuste/fluidez: maquetación flexible y semántica correcta para distintos dispositivos.

Registro y Mejora Continua

Los hallazgos de usabilidad y accesibilidad se documentaron utilizando la plantilla de la Tabla 8, clasificándolos por módulo y severidad. Entre los ajustes ya implementados se encuentran la incorporación de un aviso al redirigir a usuarios no autenticados, la inclusión del landmark <main> para el contenido principal y la mejora del contraste en ciertos textos.

Cada hallazgo se incorporó al backlog de mejoras del proyecto y se asoció a evidencias concretas (capturas, fragmentos de código) registradas en la Tabla 9. Este esquema facilita la trazabilidad entre problemas detectados, acciones correctivas y versiones futuras de EngatiApp.

Tabla 4

Guion de Tareas (T1–T4)

Tarea	RF asociado	Objetivo del usuario	Pasos sugeridos	Criterio de éxito	Datos de prueba
-------	-------------	----------------------	-----------------	-------------------	-----------------

T1 Registro/Login	RF1	Crear cuenta e iniciar sesión	Abrir “Registro” → completar correo, contraseña, nombre, sector, rol → Registrarse → Iniciar sesión	Cuenta creada y sesión iniciada; navegación actualizada por rol	Correo válido; contraseña ≥ 6; rol (ciudadano/líder)
T2 Publicar iniciativa	RF2	Publicar una iniciativa (líder)	Login como líder → ir a Iniciativa → diligenciar título, descripción, categoría, sectores (múltiple) y fecha opcional → Publicar	Iniciativa aparece en listado; sectores correctos; contadores actualizados	Texto y selects válidos; fecha opcional
T3 Consultar alertas	RF3	Revisar alertas y filtrarlas	Ir a Alertas → aplicar filtros de categoría y sector	Listado refleja el filtro; aparece “No hay alertas activas” cuando no hay resultados	Categoría/sector con y sin coincidencias
T4 Visualizar estadísticas	RF4	Ver panorama de actividad	Ir a Estadísticas → revisar tarjetas y gráficas	Totales correctos y gráficas renderizadas (sin duplicarse)	Datos existentes en colecciones

Tomado de. J. Díaz Marín (2025).

En la Tabla 5 se resumen los principales casos de prueba funcionales ejecutados mediante recorridos críticos internos sobre el prototipo, cubriendo los flujos de registro, inicio de sesión, publicación de iniciativas, emisión de alertas, visualización de estadísticas y cierre de sesión.”

Tabla 5

Casos de Prueba Funcionales Ejecutados en el Prototipo de EngatiApp

ID	Módulo / flujo	Descripción del caso de prueba	Datos de entrada principales	Resultado esperado	Resultado obtenido en el prototipo	Estado	Tipo de prueba
CP01	Registro de usuario nuevo	Verificar que un ciudadano pueda crear una cuenta con datos válidos.	Nombre, correo válido, contraseña mínima, aceptación de términos.	El sistema crea el usuario en Firebase Auth, registra el perfil en Firestore y redirige a la pantalla inicial autenticada.	Se creó el usuario de prueba, se registró en Auth/Firestore y se mostró la pantalla inicial autenticada.	Aprobado	Recorrido crítico interno del desarrollador
CP02	Registro – manejo de errores	Verificar que el formulario no permita registro con campos vacíos	Campos obligatorios vacíos / correo ya registrado.	El sistema muestra mensajes de error claros y no crea cuentas	Se mostraron mensajes de error; no se creó usuario cuando faltaban datos	Aprobado (se ajustaron textos de error durante el desarrollo)	Recorrido crítico interno del desarrollador

		o correo ya usado.		duplicadas ni incompletas.	o el correo ya existía.		
CP03	Inicio de sesión	Verificar que un usuario registrado pueda iniciar sesión correctamente.	Correo y contraseña válidos de un usuario previamente registrado.	El sistema valida credenciales en Firebase Auth y redirige a la vista de inicio autenticada.	Inicio de sesión exitoso y acceso a las opciones de ciudadano (alertas, iniciativas, estadísticas).	Aprobado	Recorrido crítico interno del desarrollador
CP04	Publicación de iniciativa	Verificar que un usuario pueda registrar una nueva iniciativa comunitaria.	Título, categoría, descripción breve, sector/barrio.	La iniciativa se guarda en Firestore y aparece en el listado de iniciativas sin recargar la página completa.	La iniciativa de prueba se creó, quedó visible en el listado y se pudo verificar en la colección de Firestore.	Aprobado (se ajustó la validación de campos obligatorios)	Recorrido crítico interno del desarrollador
CP05	Creación y consulta de alerta	Verificar que se pueda crear una alerta y que esta se visualice en el módulo de alertas.	Tipo de alerta, título, mensaje, fecha/hora de publicación.	La alerta se registra en Firestore, aparece en el listado y puede filtrarse por tipo/fecha (si aplica).	La alerta de prueba se creó, quedó listada y se pudo consultar desde la vista de alertas.	Aprobado	Recorrido crítico interno del desarrollador

CP06	Visualización de estadísticas	Verificar que el módulo de estadísticas cargue datos y renderice los gráficos.	Datos existentes en Firestore (iniciativas/alertas de prueba).	El sistema consulta Firestore, agrupa la información y muestra los gráficos en Chart.js sin errores.	Los gráficos se mostraron correctamente con los datos de prueba; se comprobó el aria-label en los canvas.	Aprobado (se ajustó el texto de etiquetas en los gráficos)	Recorrido crítico interno del desarrollador
CP07	Cierre de sesión y restricción	Verificar que al cerrar sesión se bloquee el acceso a las pantallas privadas.	Usuario autenticado que selecciona “Cerrar sesión”.	El sistema cierra sesión en Firebase Auth y redirige a la pantalla pública; las rutas privadas ya no son accesibles.	Al cerrar sesión se redirigió a la página de inicio y, al intentar volver a una sección privada, se redirigió de nuevo a inicio.+	Aprobado	Recorrido crítico interno del desarrollador

Tomado de. J. Díaz Marín (2025).

En esta fase del proyecto no se realizó una medición cuantitativa formal con múltiples usuarios, por lo que no se reportan tiempos promedio ni porcentajes de éxito. Sin embargo, se definió un plan de métricas para futuras pruebas de usabilidad con habitantes de Engativá, en el que se registrarán, para cada tarea crítica, el número de intentos, éxitos, porcentaje de éxito y tiempos de ejecución. Dicho plan se resume en la Tabla 6.

Tabla 6

Plan de Métricas para Futuras Pruebas de Usabilidad

Tarea	Intentos	Éxitos	%Éxito	t1 (s)	t2 (s)	t3 (s)	t4 (s)	Promedio (s)
T1 Registro/Login								
T2 Publicar								
T3 Alertas								
T4 Estadísticas								

Nota. La tabla presenta el plan de métricas definido para fases posteriores de evaluación con usuarios; en la versión actual del prototipo no se han recolectado aún datos cuantitativos de tiempo ni porcentaje de éxito. *Tomado de.* J. Díaz Marín (2025).

Tabla 7*Checklist Heurístico (Usabilidad y Accesibilidad)*

ID	Criterio	Pregunta guía	Evidencia (Sí/No/NA)	Observaciones
H01	Teclado	¿Toda la navegación se puede hacer con Tab/Enter/Espacio?	Si	Botones <button> y links accesibles; skip-link presente.
H02	Foco visible	¿Existe estilo :focus claro en botones, links e inputs?	Si	:focus { outline: 3px ... } global.
H03	Orden de foco	¿El foco sigue un orden lógico y predecible?	Si	Header → Nav → Pantalla activa; skip-link ayuda.
H04	Contraste	¿Texto y controles tienen contraste suficiente?	Si	Botones y header OK; texto meta #777 es límitrofe (vigilar).
H05	Etiquetas	¿Inputs tienen <label> asociadas y texto claro?	Si	Formularios usan label y for.
H06	Semántica	¿HTML con encabezados y roles básicos?	Si	header, nav, aria-live y role="status"; sugerencia: añadir <main>.
H07	Responsividad	¿Se adapta a distintos tamaños de pantalla?	Si	Layout fluido, grids; responsive básico.
H08	Errores legibles	¿Los mensajes de error explican y orientan?	Si	Alerts de éxito/error en UI.
H09	Consistencia UI	¿Botones/colores/estados son consistentes?	Si	Esquema de tarjetas/colores consistente.
H10	Prevención de pérdida	¿No se pierden datos al navegar?	No	No hay guardado de borradores en formularios.

H11	Feedback	¿Hay confirmaciones claras de éxito/fracaso?	Si	Mensajes tras acciones clave (login, publicar, etc.).
H12	Atajos (opc.)	Si existen, ¿no chocan con el navegador?	NA	No hay atajos definidos.
H13	Saltar al contenido	¿Existe enlace para saltar al contenido principal?	Si	.skip-link implementado.
H14	Objetivo táctil	¿Botones con tamaño/separación adecuados?	Si	Botones con padding; revisar en móviles reales.

Tomado de. J. Díaz Marín (2025).

En la Tabla 8 se resumen los principales hallazgos identificados durante la revisión interna de accesibilidad y usabilidad del prototipo. Algunos ajustes, como la incorporación del landmark <main> para el contenido principal, ya fueron corregidos, mientras que otros (mensajes para usuarios no autenticados y mejora de contraste en textos secundarios) se mantienen abiertos y se proponen como mejoras para iteraciones futuras de EngatiApp.

Tabla 8

Registro de Hallazgos (se Completa con Datos de Usuarios Simulados).

ID	Módulo	Descripción	Severidad	Evidencia	Acción propuesta	Estado
HZ-01	Acceso	Usuario no autenticado: al solicitar una pantalla privada es redirigido a Inicio sin explicación.	Menor	Comportamiento observado al intentar acceder a una ruta privada sin autenticación en el prototipo.	Implementar un aviso breve (por ejemplo, un mensaje tipo “Inicia sesión para acceder a esta sección”) en la redirección.	Abierto
HZ-02	Semántica	Ausencia de landmark <main> para el contenido principal (ya corregido en el prototipo).	Menor	Inspector DOM mostrando <main id="app" role="main"> envolviendo el contenido principal.	Mantener el uso de <main> y estructura semántica consistente en nuevas pantallas o refactorizaciones.	Cerrado
HZ-03	Legibilidad	Texto meta en tarjetas usa color #777, con contraste bajo en algunas pantallas.	Menor	Captura de las tarjetas con texto usando color #777.	Ajustar el color de texto a #555 (o un valor equivalente con contraste adecuado) en la hoja de estilos.	Abierto

Tomado de. J. Díaz Marín (2025).

Tabla 9*Plantilla de Evidencias (Figuras/Capturas)*

Figura	Descripción	Módulo/RF	Ubicación en documento	Fecha	Autor
5	Mapa de navegación de EngatiApp. Cobertura: RF1–RF4.	RF1–RF4	Figura 5	04/11/2025	Jhonatan Díaz M.
6	Pantalla Inicio.	RF4	Figura 6	04/11/2025	Jhonatan Díaz M.
7	Login / Registro.	RF1	Figura 7	04/11/2025	Jhonatan Díaz M.
8	Publicar iniciativa.	RF2	Figura 8	04/11/2025	Jhonatan Díaz M.
9	Visualizar alertas.	RF3	Figura 9	04/11/2025	Jhonatan Díaz M.
10	Estadísticas.	RF4	Figura 10	04/11/2025	Jhonatan Díaz M.
11	Enlace a la política de tratamiento. Cobertura: RFN–Privacidad.	RFN–Privacidad	Figura 11	04/11/2025	Jhonatan Díaz M.
11.1	Contenido de la política de tratamiento. Cobertura: RFN–Privacidad.	RFN–Privacidad	Figura 11.1	04/11/2025	Jhonatan Díaz M.

12	Accesibilidad: navegación por teclado con foco visible.	RFN– Accesibilidad	Figura 12	04/11/2025	Jhonatan Díaz M.
13	Accesibilidad: texto alternativo en imágenes (DOM).	RFN– Accesibilidad	Figura 13	04/11/2025	Jhonatan Díaz M.
14	Dependencias del proyecto (package.json) – uso de Firebase.	RFN– Plataforma nube	Figura 14	04/11/2025	Jhonatan Díaz M.
15	Paquetes Firebase instalados (package-lock.json).	RFN– Plataforma nube	Figura 15	04/11/2025	Jhonatan Díaz M.
16	Configuración de despliegue (firebase.json) – Hosting.	RFN– Despliegue	Figura 16	04/11/2025	Jhonatan Díaz M.
17	Reglas de seguridad de Firestore (captura del editor).	RFN– Seguridad de datos	Figura 17	04/11/2025	Jhonatan Díaz M.

Tomado de. J. Díaz Marín (2025).

Despliegue del Sistema

El prototipo de EngatiApp se desplegó utilizando Firebase Hosting, aprovechando la integración nativa con los servicios de Authentication y Firestore. El código fuente del frontend (HTML, CSS y JavaScript) se preparó en una estructura estática y se configuró el archivo `firebase.json` para definir la carpeta pública, las reglas de limpieza de URLs y la relación con la base de datos.

Una vez inicializado el proyecto en la consola de Firebase, se ejecutó el comando de despliegue, generando una URL pública para fines académicos y de demostración. Este entorno permite mostrar el funcionamiento completo del prototipo sin requerir infraestructura propia, y facilita una futura migración a un entorno productivo con mayor monitoreo, respaldo y políticas de seguridad más estrictas.

Trabajos Futuros

A partir de la experiencia de diseño e implementación de EngatiApp, se identifican varias líneas de trabajo para una segunda versión de la aplicación:

Ampliar el módulo de administración: incorporar parametrización de categorías, sectores y tipos de alerta, así como un panel más completo de gestión de usuarios y roles (altas, bajas y cambio de perfiles).

Profundizar la validación de la participación ciudadana: realizar pruebas piloto con una muestra mayor de líderes y ciudadanos de Engativá, combinando encuestas y talleres comunitarios, y aplicando el plan de pruebas definido para recopilar métricas básicas de uso (tareas completadas, frecuencia de ingreso, módulos más consultados).

Completar los controles de protección de datos y trazabilidad: avanzar en la implementación de vistas para ejercer derechos ARCO (acceso, rectificación y supresión de datos) y en el registro de eventos relevantes de la aplicación (creación de alertas, iniciativas y cambios de rol), de acuerdo con los requerimientos no funcionales de privacidad y trazabilidad.

Fortalecer la accesibilidad: extender los ajustes actuales con pruebas específicas para personas con discapacidad visual o motora, revisar sistemáticamente contrastes y tamaños de fuente, y documentar pautas de diseño inclusivo que sirvan como guía para futuras versiones.

Integración con canales institucionales: explorar la conexión con canales oficiales de la alcaldía local u otras entidades, de modo que las alertas y convocatorias comunitarias puedan sincronizarse con información institucional y reforzar la articulación entre ciudadanía y Estado.

Explorar la integración con fuentes de datos abiertas del distrito: analizar la viabilidad de consumir datos públicos (por ejemplo, sobre seguridad, movilidad o equipamientos urbanos) para enriquecer las estadísticas de EngatiApp y apoyar mejor la toma de decisiones comunitarias.

Evolución tecnológica del prototipo: valorar la migración hacia una Progressive Web App (PWA) que permita notificaciones push, mejor soporte offline y una experiencia más cercana a una aplicación móvil nativa, así como estudiar, a mediano plazo, una arquitectura más modular que facilite la escalabilidad del sistema.

Conclusiones

El desarrollo de EngatiApp permitió abordar de manera estructurada el problema de la desarticulación y la baja visibilidad de los procesos comunitarios en la localidad de Engativá. A partir de un diagnóstico participativo sencillo —que incluyó actividades como el diario/mapa de sueños y encuestas en línea— y del análisis del contexto local, se identificó la necesidad de contar con una herramienta digital que centralizara la publicación de iniciativas, la emisión de alertas y la consulta de información relevante para los habitantes del territorio. La aplicación propuesta responde a esta necesidad mediante un diseño web sencillo, accesible desde distintos dispositivos y pensado para ser comprendido por personas con distintos niveles de alfabetización digital.

Metodológicamente, el proyecto integró la Investigación Acción Participativa, el Design Thinking y el marco ágil Scrum. Esta combinación permitió pasar del reconocimiento de las necesidades comunitarias a la construcción iterativa de un prototipo funcional, manteniendo como referencia la experiencia de ciudadanos y líderes involucrados en las acciones de diagnóstico y socialización. La definición de requerimientos funcionales y no funcionales, junto con la matriz de trazabilidad, ayudó a mantener coherencia entre el problema planteado, la pregunta de investigación, los objetivos específicos y las decisiones técnicas tomadas durante el diseño de la solución.

En términos de resultados, se logró implementar un prototipo web con módulos de registro e inicio de sesión, gestión de iniciativas, emisión y consulta de alertas y visualización de estadísticas básicas. Se incorporaron, además, ajustes iniciales de accesibilidad (enlace para “Saltar al contenido”, foco visible y estructura semántica con <main>) y controles básicos de protección de datos mediante avisos de privacidad y reglas de seguridad en Firestore orientadas a

limitar el acceso según el rol. Aunque la validación con usuarios reales se centró en pruebas exploratorias y recorridos críticos sobre el prototipo, se dejó definido un plan inicial de usabilidad y métricas de evaluación que puede aplicarse en fases posteriores con una muestra más amplia de participantes.

En suma, la versión entregada cumple el objetivo general y avanza en los específicos 1–3, mientras que el Objetivo 4 queda parcialmente atendido mediante validación exploratoria y totalmente encarrilado para su ampliación con el plan de evaluación definido.

Referencias Bibliográficas

- Alur, D., Crupi, J., & Malks, D. (2003). Core J2EE patterns: Best practices and design strategies.
- Aristizábal Ángel, V. M. (2025). Una estrategia para facilitar la implementación de la accesibilidad web en documentos HTML, conforme a las WCAG 2.2. Universidad EAFIT.
- Arteche, M., Santucci, M., & Welsh, S. (2025). Metodologías ágiles para la innovación en Latinoamérica. *Innovar*, *35*(96), e102267.
<https://doi.org/10.15446/innovar.v35n96.102267>
- Autentia & IZERTIS. (2024). Software design: Principios y patrones.
https://ahorasomos.izertis.com/autentia/wp-content/uploads/libros/SoftwareDesign_PrincipiosyPatrones-Autentia.pdf
- Barrena Herrán, M., Grijalba, O., & Modrego Monforte, I. (2024). Mapeando la ciudad a través del BigData: Análisis temporal de las dinámicas urbanas. En Actas del VIII Congreso ISUF-H (pp. 214–217). <https://doi.org/10.4995/ISUFh2024.2024.17568>
- Bejarano, M. H., & Rey, L. E. B. (2020). Ciclos de vida del software y metodologías ágiles. Editorial Universidad Nacional Abierta y a Distancia.
- Beltrán González, S. A. (2022). Design thinking y metodologías ágiles (Scrum) como efecto potenciador en la gestión de proyectos. Universidad Militar Nueva Granada.
- Castañeda Meneses, P. (2024). Del trabajo social presencial al trabajo social digital: Sistematización del proceso de desmaterialización vía digitalización de los desempeños presenciales. *Revista Perspectivas*, *43*(1), 11–41.
<https://doi.org/10.29344/07171714.43.3745>

- Cauce Montez, S., & Romas Rodríguez, M. J. (2025). Evolución de la participación ciudadana en la gestión pública: Tendencias y desafíos. *I+D Internacional – Revista Científica Académica*, *4*(1), 1–15.
- Chancusig López, E. J. (2025). Desarrollo de una aplicación web educativa interactiva de asistencia de tareas a niños y jóvenes del Centro de Estudios de la Corporación de Desarrollo Comunitario La Esperanza en el área de matemáticas. Universidad Nacional de Chimborazo.
- Chaves Parra, S. P. (2025). Construcción de comunidad digital: Un ejercicio de comunicación participativa en la Fundación Karisma de la ciudad de Bogotá D.C. Universidad Nacional Abierta y a Distancia.
- Checa Pinzón, D. M. (2024). Evaluación de la herramienta “Chatico” como mecanismo de votación para presupuestos participativos en Bogotá: Estudio de caso del programa durante su desarrollo en los años 2022 y 2023. Universidad Nacional Abierta y a Distancia.
- Colombia. Congreso de la República. (2012). Ley Estatutaria 1581 de 2012 (protección de datos personales). Gestor Normativo – Función Pública.
- Colombia. Congreso de la República. (2013). Ley 1618 de 2013 (disposiciones para garantizar el pleno ejercicio de los derechos de las personas con discapacidad). Gestor Normativo – Función Pública.
- Colombia. Presidencia de la República. (2013). Decreto 1377 de 2013 (reglamenta parcialmente la Ley 1581 de 2012). Gestor Normativo – Función Pública.
- Constitución Política de Colombia. (1991). Gaceta Constitucional No. 116.
<http://www.secretariassenado.gov.co/constitucion-politica>

- Cubides Cárdenas, J. A. (2021). Las veedurías ciudadanas como una expresión de la democracia participativa. *Revista Doctrina Distrital*, (41), 41–60.
<https://dialnet.unirioja.es/servlet/articulo?codigo=9086926>
- Encinas, D., Galarza, B., Bond, R., Zaccardi, G., & Mendes, N. (2023). *Arquitecturas en la nube: Diseño, simulación y despliegue*. Instituto de Ingeniería y Agronomía – UNAJ.
- Fowler, M. (2002). *Patterns of enterprise application architecture*.
- Franco, E., & Reyes, E. (2024). *Desarrollo de una plataforma web centralizada para la promoción y gestión de eventos deportivos en Bogotá*. Universidad Distrital Francisco José de Caldas.
- Freeman, E., & Freeman, E. (2022). *Head first design patterns*.
- Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). *Design patterns: Elements of reusable object-oriented software*.
- Granollers, T. (2022). *Pruebas de usabilidad: Inspección, indagación y testing*.
- Haines, C. (2014). *Understanding the four rules of simple design*.
- Kocot-Szymańska, A. (2020). *Innovación social y metodologías Lean: Retos en la práctica comunitaria*. [Artículo académico].
- Martin, R. C. (2023). *The Clean Code Blog*. <https://blog.cleancoder.com>
- Massa, S. M. (2017). *Objetos de aprendizaje: Metodología de desarrollo y evaluación de la calidad* (Tesis de doctorado, Universidad Nacional de La Plata, Facultad de Informática).
<https://1library.co/document/6zkkemez-objetos-aprendizaje-metodologia-desarrollo-evaluacion-calidad.html>
- Microsoft. (2023). *Security Development Lifecycle (SDL)*. <https://www.microsoft.com/en-us/sdl>

- Ministerio de Tecnologías de la Información y las Comunicaciones (MinTIC). (2023a). Política de Gobierno Digital en Colombia. <https://www.mintic.gov.co>
- Ministerio de Tecnologías de la Información y las Comunicaciones (MinTIC). (2023b). Lineamientos para intercambio de información vía API (acceso y reporte de información).
- Pérez Lara, J. E., & Covarrubias Moreno, O. M. (2025). Analítica de datos y gobierno abierto: Hacia una gestión pública basada en evidencia. *RIESED. Revista Internacional de Estudios sobre Sistemas Educativos*, *3*(16), 762–780.
<https://doi.org/10.5281/zenodo.15697620>
- Pinzón Núñez, S., Rodríguez Guerrero, R., & Vanegas, C. A. (2019). Patrón MVC y su aplicación en sistemas de información con interfaces de usuario.
- Putnam, R. D. (2001). Capital social y compromiso cívico. *Revista Internacional de Sociología*, *59*(29), 5–33.
<https://revintsociologia.revistas.csic.es/index.php/revintsociologia/article/view/1162/1651>
- Rainsberger, J. B. (s. f.-a). The four elements of simple design.
<https://blog.jbrains.ca/permalink/the-four-elements-of-simple-design>
- Rainsberger, J. B. (s. f.-b). Putting an age-old battle to rest.
<https://blog.thecodewhisperer.com/permalink/putting-an-age-old-battle-to-rest>
- Velásquez, L. A., Alvarado, S. Y., & Barroeta, V. D. V. (2021). Investigación-acción-participativa: Alternativa metodológica para el estudio de las comunidades. *Revista Scientific*, *6*(21), 314–335. <https://doi.org/10.29394/Scientific.issn.2542-2987.2021.6.21.17.314-335>

Yagual-Rodríguez, E. F., & León-Valle, B. W. (2023). El liderazgo comunitario y la gestión organizacional del barrio Los Sauces del cantón Santa Elena (2022–2023). *593 Digital Publisher CEIT, *8*(5)*, 253–264. <https://doi.org/10.33386/593dp.2023.5.2011>

Apéndices

Apéndice A

Manual Técnico de EngatiApp Manual Técnico de EngatiApp

Este manual técnico documenta los aspectos de instalación, configuración, arquitectura y mantenimiento del prototipo web EngatiApp, desarrollado como parte del Proyecto Aplicado de grado del programa Tecnología en Desarrollo de Software de la UNAD.

EngatiApp es una aplicación web que apoya la gestión comunitaria y el liderazgo ciudadano en la localidad de Engativá, centralizando funcionalidades de:

Publicación y consulta de iniciativas comunitarias.

Emisión y consulta de alertas.

Envío de reportes ciudadanos.

Visualización de estadísticas básicas sobre la actividad en la plataforma.

Requisitos previos:

Node.js y Firebase CLI instalados (`npm i -g firebase-tools`).

Acceso al proyecto Firebase (ID: engatiapp).

Navegador actualizado (Chrome/Edge).

Rol con permisos de Editor/Owner en Firebase Console.

Entorno tecnológico:

EngatiApp está construida como una aplicación web estática con lógica en el lado del cliente e integración con servicios en la nube:

HTML5, CSS3 y JavaScript para la interfaz de usuario y la lógica del cliente.

Firebase (versión 8.x) para:

Authentication (registro e inicio de sesión por correo y contraseña).

Cloud Firestore (almacenamiento de usuarios, iniciativas, alertas, reportes y logs).

Firebase Hosting (despliegue del sitio).

Chart.js para la generación de gráficos en el módulo de estadísticas.

APIs del navegador:

Local storage para recordar filtros y aceptación del aviso de privacidad.

SpeechSynthesis (modo lectura) como apoyo de accesibilidad.

Estructura del Proyecto:

El proyecto se organiza como un sitio hospedado en Firebase, con la siguiente estructura base.

Carpeta raíz del proyecto ENGATIAPP.

.firebase/ – Archivos internos de Firebase (hosting cache).

.vscode/ – Configuración del editor.

node_modules/ – Dependencias de Node instaladas (no se versionan normalmente).

public/

index.html – Página principal de EngatiApp usada por Firebase Hosting.

404.html – Página de error para rutas no encontradas.

privacidad.html – Página con la política de privacidad (enlazada desde la navegación).

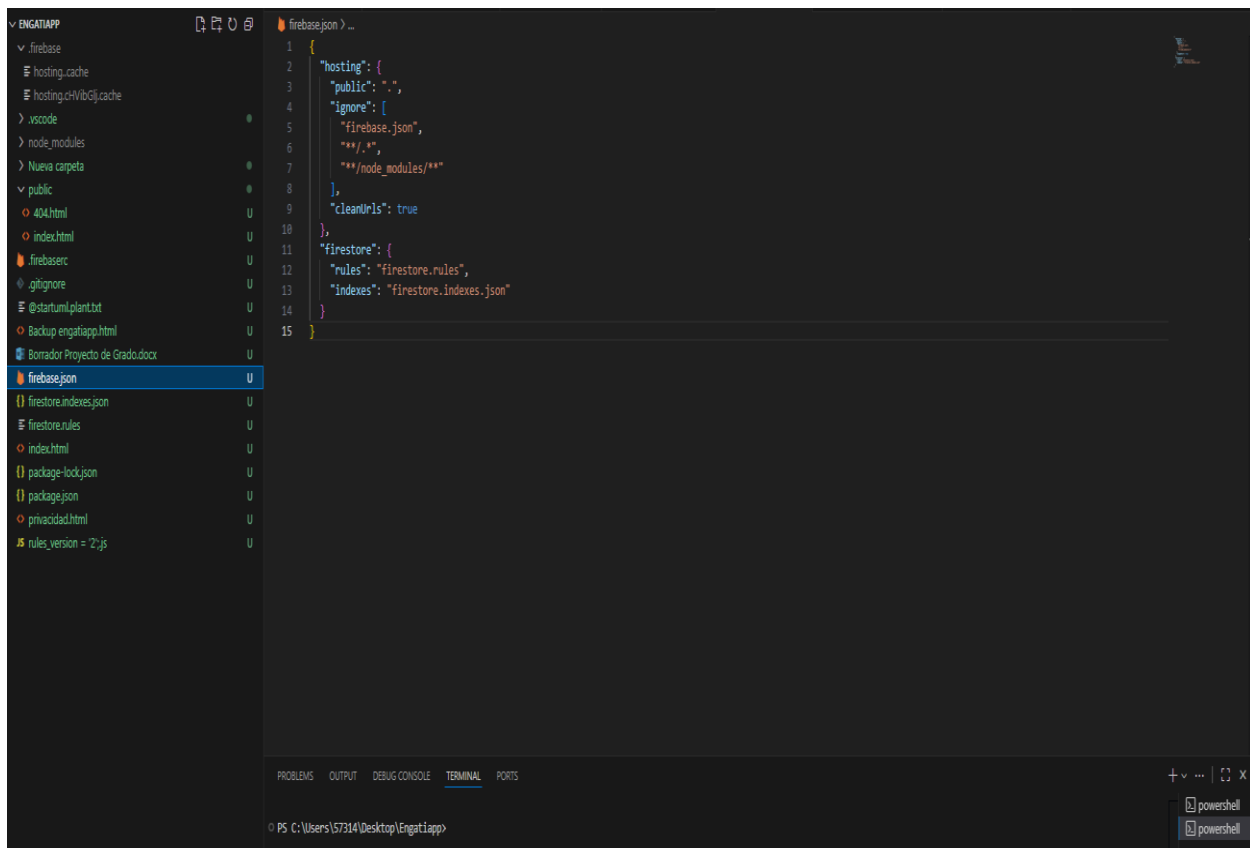
firebase.json – Configuración de Firebase Hosting (carpeta pública, rewrites, etc.).

firestore.rules – Reglas de seguridad de Firestore.

package.json y package-lock.json – Definición de dependencias y scripts de npm.

Otros archivos auxiliares (.firebaserc, @startuml.plant.txt, backups, etc.).

Estructura del Proyecto en VS Code



Nota. Se mantiene un index.html adicional en la raíz como copia de trabajo. El archivo que realmente sirve contenido en producción es aquel ubicado en la carpeta definida en firebase.json (por defecto, public/index.html).

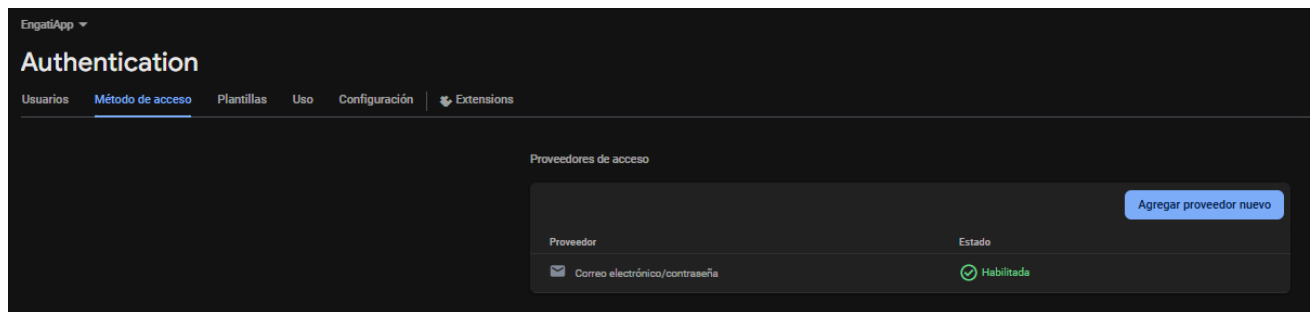
Tomado de J. Díaz Marín (2025).

Autenticación (Firebase Auth):

Proveedor habilitado: Email/Password.

Ruta: Firebase Console → Build → Authentication → Sign-in method.

Firebase Authentication: Email/Password Habilitado



Tomado de J. Díaz Marín (2025).

Asignación de roles:

Autenticar al usuario (registro o login).

En Firestore → Data → usuarios, abrir el documento cuyo ID = uid.

Establecer el campo rol a uno de: ciudadano | lider | admin.

(Opcional) Ajustar consentimientos.

Colecciones y Campos Principales (Firestore)

Colección	Clave del doc	Campos principales (ejemplos)
usuarios	uid	correo (string), nombre (string), sector (string), rol (string), fechaRegistro (ts), consentimientos. {auth,iniciativas,alertas,reportes,metricas,fecha
iniciativas	autogen	titulo, descripcion, categoria, sectores (array), fechaPublicacion (ts), fechaEvento, autorId, autorNombre, autorSector, estado
Alertas	autogen	titulo, descripcion, categoria, sector, tipo, fecha (ts), estado, autorId

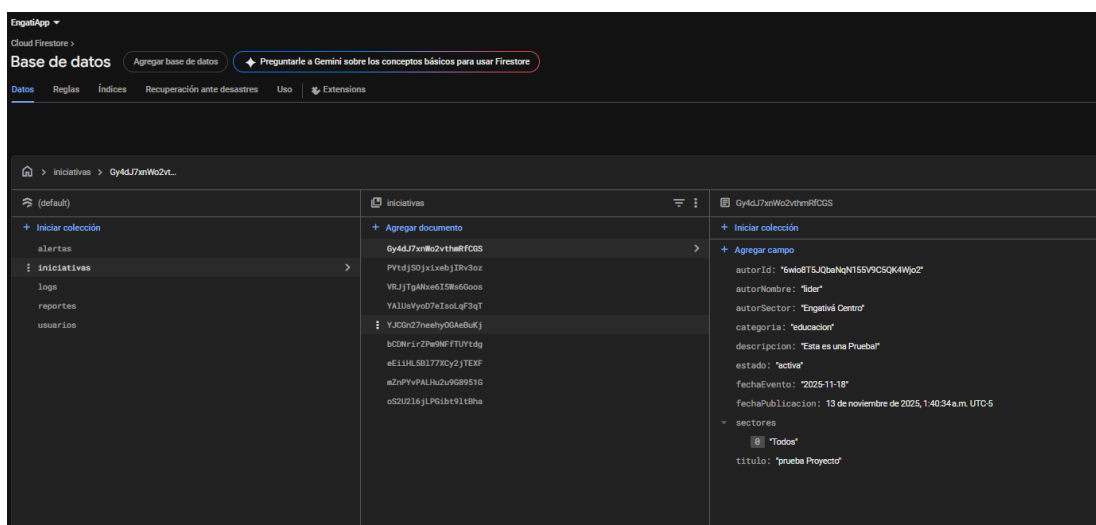
reportes	autogen	titulo, descripcion, categoria, sector, fecha (ts), estado, autorId, autorNombre
Logs	autogen	tipo (p.ej., auth.login, alerta.creada), uid, payload (objeto), fecha (ts)
consentimiento	autogen	tipo: 'banner', uid (si aplica), fecha (ts)

s

Nota. Resumen del esquema usado en el prototipo.

Tomado de J. Díaz Marín (2025).

Firestore: Colecciones del Prototipo



Nota. La aplicación usa las colecciones listadas en las Colecciones y Campos Principales (Firestore). A modo ilustrativo, se muestra la estructura de alertas.

Tomado de J. Díaz Marín (2025).

Arquitectura Lógica de la aplicación, módulos y pantallas:

El archivo index.html define toda la interfaz con múltiples pantallas (div.screen) que se muestran u ocultan dinámicamente:

Inicio (#inicio) Resumen general con contadores de alertas activas, iniciativas y usuarios, y accesos directos a alertas, iniciativas y estadísticas.

Registro/Login (#registro) Formulario para:

Registro de nuevo usuario (correo, contraseña, nombre, sector y rol).

Inicio de sesión.

Recuperación de contraseña. Incluye bloque de consentimiento por finalidad (iniciativas, alertas, reportes y métricas).

Publicar iniciativa (#iniciativa) Disponible solo para rol líder comunitario. Permite crear iniciativas con título, descripción, categoría, sectores objetivo y fecha de realización.

Alertas (#alertas)

Listado de alertas activas, con filtros por sector y categoría.

Formulario de emisión de alertas visible únicamente para rol admin.

Iniciativas (#iniciativas) Listado público de iniciativas con filtros por categoría y sector.

Reporte ciudadano (#reporte) Disponible para rol ciudadano. Permite enviar reportes con título, descripción, categoría y sector, y visualizar los “Mis reportes”.

Estadísticas (#estadisticas) Módulo para líderes y administradores, con:

Tarjetas de resumen (total usuarios, líderes, iniciativas activas, alertas recientes).

Gráficos Chart.js de iniciativas por categoría, alertas por sector y usuarios por sector.

Perfil (#perfil) Permite al usuario actualizar nombre, sector y consentimientos, exportar sus datos (.json) y solicitar eliminación de cuenta.

Administración (#admin)

Gestión básica de usuarios (cambio de rol).

Visualización y gestión de reportes ciudadanos (marcar revisado, convertir en alerta).

Además, se incluyen elementos globales:

Barra de navegación (<nav id="mainNav">) con botones contextuales según el rol.

Enlace “Saltar al contenido”, banner de privacidad y banner de estado offline.

Roles y permisos:

Los roles se almacenan en la colección usuarios y se usan en la lógica de navegación

(actualizar Navegación):

Ciudadano:

Puede ver alertas e iniciativas.

Puede enviar y consultar *reportes ciudadanos* (módulo #reporte).

Líder:

Puede publicar iniciativas (#iniciativa).

Puede acceder al módulo de estadísticas (#estadísticas).

Admin:

Puede gestionar usuarios y reportes (#admin).

Puede emitir alertas (form-alerta-admin en #alertas).

Modelo de datos (Firestore):

A continuación, se describen las colecciones principales de Firestore utilizadas por la aplicación:

Colección (Usuarios):

Documento por cada usuario autenticado (ID = uid de Firebase Auth).

Campos principales:

correo: string, correo electrónico.

nombre: string, nombre completo.

sector: string, barrio/sector (p. ej. “Engativá Centro”).

rol: string, uno de ciudadano, lider, admin.

fechaRegistro: timestamp.

consentimientos: objeto con:

auth: boolean (presencia del usuario en la base).

iniciativas: boolean (permite publicar iniciativas).

alertas: boolean (permite gestionar alertas).

reportes: boolean (permite enviar reportes ciudadanos).

metricas: boolean (permite el uso de datos anónimos de uso).

fecha: timestamp de la última actualización de consentimientos.

Colección (Iniciativas):

Cada documento representa una iniciativa comunitaria publicada por un líder.

Campos principales:

titulo: string.

descripcion: string (corto/medio).

categoria: string (salud, educación, ambiente, etc.).

sectores: array de strings (sectores objetivo o ['Todos']).

fechaPublicacion: timestamp.

fechaEvento: string/timestamp (fecha seleccionada en el formulario o null).

autorId: string (uid del usuario).

autorNombre: string.

autorSector: string.

estado: string (por defecto 'activa').

Colección (Alertas):

Documentos de alertas emitidas, generalmente por administradores.

título, descripción.

categoría: string.

sector: string.

tipo: string (informativa, preventiva, urgente).

fecha: timestamp.

estado: string (por defecto 'activa').

autorId: uid del administrador.

Colección (Reportes):

Registra los reportes enviados por ciudadanos:

título, descripción.

categoría, sector.

fecha: timestamp.

estado: string (pendiente, revisado, convertido).

autorId: uid.

autorNombre: nombre visible en el perfil.

Colección (Logs):

Registro interno de eventos relevantes: autenticar, crear iniciativas, crear alertas, convertir reportes, exportar datos, etc.

tipo: string (auth.login, iniciativa.creada, alerta.creada, etc.).

uid: identificador del usuario asociado (si aplica).

payload: objeto con detalles mínimos.

fecha: timestamp.

Colección (Consentimientos):

Se usa para registrar la aceptación del banner de privacidad:

tipo: 'banner'.

uid: uid del usuario (si estaba autenticado).

fecha: timestamp.

Permisos por módulo (roles)

Módulo / Acción	Ciudadano	Líder	Admin
Ver alertas	X	X	X
Emitir alerta			X
Ver iniciativas	X	X	X
Publicar iniciativa		X	X
Reportar / Mis reportes	X	X	X
Estadísticas		X	X
Administración (usuarios/reportes)			X

Nota. Visibilidad/acciones actuales del prototipo.

Tomado de J. Díaz Marín (2025).

Reglas de Seguridad:

Las reglas en firestore.rules deben alinearse con los roles descritos y el uso de las colecciones. A modo de lineamientos generales:

Solo usuarios autenticados (`request.auth != null`) pueden escribir en usuarios, iniciativas, alertas y reportes.

En usuarios, cada usuario solo debería poder leer y modificar su propio documento (excepto administradores).

En iniciativas y reportes, el autor debería poder crear y leer sus registros; la lectura agregada puede ser pública si así se define.

En alertas, la escritura se restringe a rol admin.

La colección logs debe ser de solo escritura desde el cliente (lectura restringida a administradores, o solo a backend si se amplía el sistema).

La política de acceso por rol y las restricciones de lectura/escritura se rigen por firestore.rules, que constituye la configuración vigente de seguridad.

Despliegue:

El prototipo se publica con Firebase Hosting; se mantiene el hosting estático y una URL pública como punto único de acceso. (CLI: `firebase deploy --only hosting`).

Fragmento Reglas:

```
(/* ===== usuarios ===== */
match /usuarios/{userId} {
  allow create: if signedIn()
    && request.auth.uid == userId
    && request.resource.data.keys().hasAny(['rol'])
```

```

        && request.resource.data.rol in ['ciudadano','lider']
        && ( !request.resource.data.keys().hasAny(['nombre']) ||
isShortString(request.resource.data.nombre, 120) )
        && ( !request.resource.data.keys().hasAny(['sector']) ||
isShortString(request.resource.data.sector, 80) );

allow read: if signedIn() && (request.auth.uid == userId || isAdmin());

allow update: if signedIn() && (
    isAdmin() || (
        request.auth.uid == userId
        && ( !request.resource.data.keys().hasAny(['rol'])
            || request.resource.data.rol == resource.data.rol )
    )
)
&& ( !request.resource.data.keys().hasAny(['nombre']) ||
isShortString(request.resource.data.nombre, 120) )
&& ( !request.resource.data.keys().hasAny(['sector']) ||
isShortString(request.resource.data.sector, 80) );

allow delete: if signedIn() && request.auth.uid == userId;
})

```

Fragmento de firestore.rules

```

1  rules_version = '2';
2  service cloud.firestore {
3    match /databases/{database}/documents {
4
5      /* ===== Helpers ===== */
6      function signedIn() { return request.auth != null; }
7
8      function userPath() { return /databases/{database}/documents/usuarios/{request.auth.uid}; }
9      function hasUserDoc() { return signedIn() && exists(userPath()); }
10     function userRole() { return hasUserDoc() ? get(userPath()).data.rol : null; }
11
12     function isAdmin() { return signedIn() && userRole() == 'admin'; }
13     function isLeader() { return signedIn() && userRole() == 'lider'; }
14
15     function hasConsent(key) {
16       return hasUserDoc() && get(userPath()).data.consentimientos[key] == true;
17     }
18
19     function isShortString(x, maxLen) { return x is string && x.size() > 0 && x.size() <= maxLen; }
20
21     /* ===== reportes (reporte ciudadano) ===== */
22     match /reportes/{docId} {
23       // Crear: cualquier usuario autenticado con consentimiento de "reportes"
24       allow create: if signedIn()
25         && hasConsent('reportes')
26         && request.resource.data.keys().hasAll(
27           ['titulo', 'descripcion', 'categoria', 'sector', 'fecha', 'estado', 'autorId', 'autorNombre']
28         )
29         && request.resource.data.autorId == request.auth.uid
30         && isShortString(request.resource.data.titulo, 140)
31         && isShortString(request.resource.data.descripcion, 880)
32         && request.resource.data.categoria in ['salud', 'educacion', 'ambiente', 'infraestructura', 'seguridad']
33         && request.resource.data.estado in ['pendiente', 'revisado', 'convertido'];
34
35       // Leer: autor o admin (no es público)
36       allow read: if signedIn() && ( isAdmin() || resource.data.autorId == request.auth.uid );
37
38       // Actualizar: solo admin; el autor no puede cambiar estado
39       allow update: if signedIn() && isAdmin()
40         && ( !request.resource.data.keys().hasAny(['autorId']) );

```

Tomado de J. Díaz Marín (2025).

Despliegue con Firebase Hosting:

Para desplegar EngatiApp en producción o en un entorno de pruebas, se utilizan las herramientas de Firebase CLI.

Instalación de Firebase CLI:

“npm install -g firebase-tools

firebase login”

Inicialización:

El proyecto ya cuenta con `.firebaserc` y `firebase.json`, lo que indica que se ha ejecutado `firebase init`. En un nuevo entorno solo sería necesario:

```
“firebase use <tu-proyecto-engatiapp>”
```

Compilación / Preparación:

URLs a `privacidad.html` y librerías externas funcionen correctamente.

Despliegue desde la carpeta raíz del proyecto:

```
(firebase deploy --only hosting)
```

Tras el despliegue, Firebase proporciona una URL pública (por ejemplo, `https://engatiapp.web.app`) desde la cual se accede a la aplicación.

Firestore Hosting: despliegue correcto del sitio



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS C:\Users\57314\Desktop\Engatiapp> firebase deploy --only "firestore:rules,hosting"
+ firestore: released rules firestore.rules to cloud.firestore
i hosting[engatiapp]: finalizing version...
+ hosting[engatiapp]: version finalized
i hosting[engatiapp]: releasing new version...
+ hosting[engatiapp]: release complete

+ Deploy complete!

Project Console: https://console.firebase.google.com/project/engatiapp/overview
Hosting URL: https://engatiapp.web.app
PS C:\Users\57314\Desktop\Engatiapp> |
```

Tomado de J. Díaz Marín (2025).

Accesibilidad Técnica:

Comprobación de atributo aria-label en gráficos de Chart.js

Ruta: abrir app → módulo Estadísticas → F12 (Elements) → seleccionar <canvas>.

Accesibilidad

```

▶ <div class="stat-card">...</div>
▶ <div class="stat-card">...</div>
</div>
▶ <p>...</p>
<h3>Iniciativas por categoría</h3>
▼ <div class="chart-container">
  <canvas id="iniciativasChart" aria-label="Gráfico de iniciativas por categoría" width="752" height="270" style="display: block; box-sizing: border-box; height: 300px; width: 836px;">
</div>
<h3>Alertas por sector</h3>
▼ <div class="chart-container">
  <canvas id="alertasChart" aria-label="Gráfico de alertas por sector" width="752" height="270" style="display: block; box-sizing: border-box; height: 300px; width: 836px;">
</div>
<h3>Usuarios por sector</h3>
▼ <div class="chart-container">
  <canvas id="usuariosChart" aria-label="Gráfico de usuarios por sector">
</div>
</div>
<!-- LISTA DE INICIATIVAS (público) -->
▶ <div class="screen" id="iniciativas" style="background-color:#e8f5e9">...</div> == $0 [?]
</div>

```

Tomado de J. Díaz Marín (2025).

Mantenimiento y Evolución:

Algunas tareas recomendadas de mantenimiento:

Revisar periódicamente las reglas de Firestore y los patrones de acceso (logs).

Hay que asegurar que las librerías externas (Firebase 8.x, Chart.js) se mantengan en versiones soportadas.

Respaldo eventual de colecciones críticas (usuarios, iniciativas, alertas, reportes).

Documentar cambios relevantes en roles, rutas o colecciones.

Apéndice B

Manual de usuario de EngatiApp

Este manual describe, de manera clara y paso a paso, el uso del prototipo web EngatiApp, una herramienta orientada a fortalecer la gestión comunitaria y el liderazgo ciudadano en Engativá. Su objetivo es guiar al usuario final en las tareas más frecuentes —registro e ingreso, publicación y consulta de iniciativas, emisión y lectura de alertas, reporte ciudadano, consulta de estadísticas, actualización de perfil y acciones básicas de administración— sin requerir conocimientos técnicos.

El objetivo es brindar instrucciones claras y paso a paso para que las y los usuarios de EngatiApp utilicen de forma correcta y segura las funcionalidades actualmente implementadas en el prototipo, según su rol (ciudadano, líder comunitario o administrador), respetando las pautas de accesibilidad y privacidad.

Acceso: orientar el registro, inicio de sesión y recuperación de contraseña.

Iniciativas (líder): guiar la publicación y consulta de iniciativas, incluyendo filtros por categoría y sector.

Alertas (admin): explicar la emisión de alertas por parte del administrador y la consulta por la ciudadanía.

Reporte ciudadano (ciudadano): describir el envío de reportes y su seguimiento; para el administrador, la revisión y cambio de estado.

Estadísticas (líder/admin): mostrar la lectura de tarjetas y gráficos del módulo de Estadísticas.

Perfil: indicar la actualización de datos personales y consentimientos, la exportación de datos y la solicitud de eliminación de cuenta.

Navegación y accesibilidad: señalar el uso del menú, la navegación por teclado, el foco visible y el modo lectura; así como la visualización del aviso de privacidad.

EngatiApp es una aplicación web pensada para apoyar la organización comunitaria y el liderazgo ciudadano en la localidad de Engativá. A través de la plataforma, los habitantes pueden:

- Consultar alertas e iniciativas comunitarias.

- Enviar reportes ciudadanos.

- Publicar iniciativas (líderes comunitarios).

- Visualizar estadísticas básicas sobre la participación (líderes y administradores).

Este manual explica el uso de la aplicación desde el punto de vista del usuario final.

Requisitos de uso:

- Dispositivo con navegador web moderno (computador, tableta o teléfono).

- Conexión a Internet estable.

Para usar funciones avanzadas (publicar iniciativas, reportes, estadísticas, administración) es necesario registrarse y iniciar sesión.

Roles y Permisos:

En EngatiApp existen tres tipos de usuario:

- Ciudadano

- Ver alertas e iniciativas.

- Enviar reportes ciudadanos y consultar “Mis reportes”.

- Líder comunitario

Todo lo anterior.

Publicar iniciativas comunitarias.

Acceder al módulo de estadísticas.

Administrador

Ver y gestionar usuarios.

Emitir alertas comunitarias.

Revisar y convertir reportes ciudadanos en alertas.

El rol se asigna al momento del registro o mediante el módulo de administración.

Primer Ingreso a la Plataforma:

Al abrir la URL de EngatiApp en el navegador, aparece la pantalla de inicio:

Título “EngatiApp” y barra superior con el nombre del usuario (cuando hay sesión iniciada).

Tarjetas con conteos de:

Alertas activas.

Iniciativas.

Usuarios registrados.

Botones para ver alertas, iniciativas y, si el rol lo permite, estadísticas.

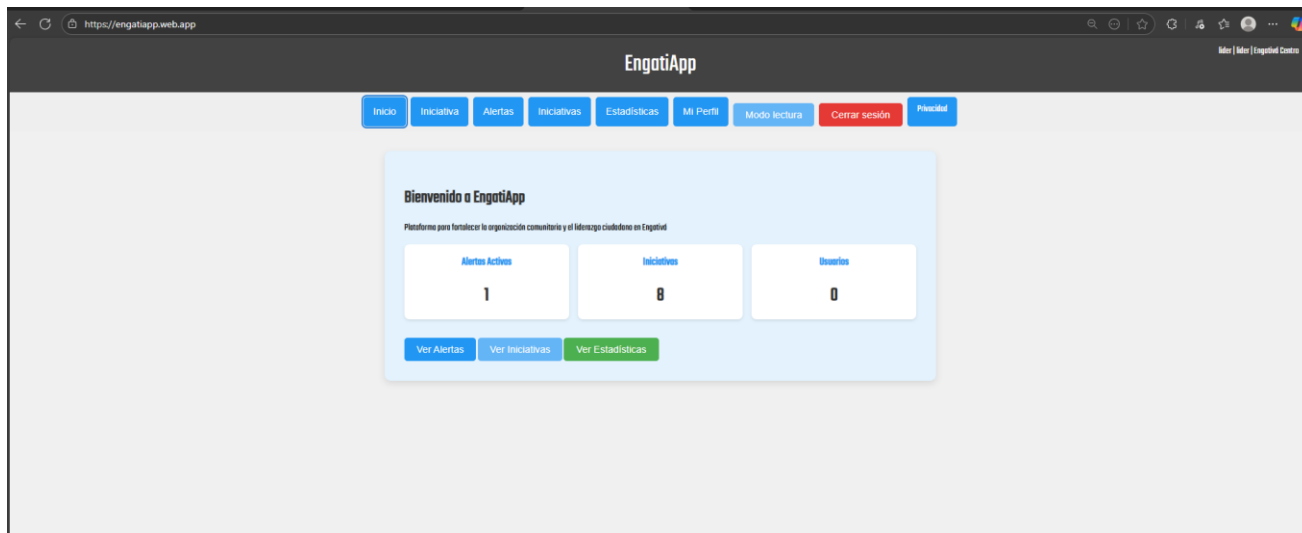
En la parte superior se encuentra el menú de navegación con botones como: Inicio, Registro/Login, Iniciativa, Alertas, Iniciativas, Reportar, Estadísticas, Mi Perfil, Admin y un enlace a Privacidad.

También se muestra, cuando aplica:

Un banner de aviso de privacidad, con un botón “Entendido” y enlace a la política.

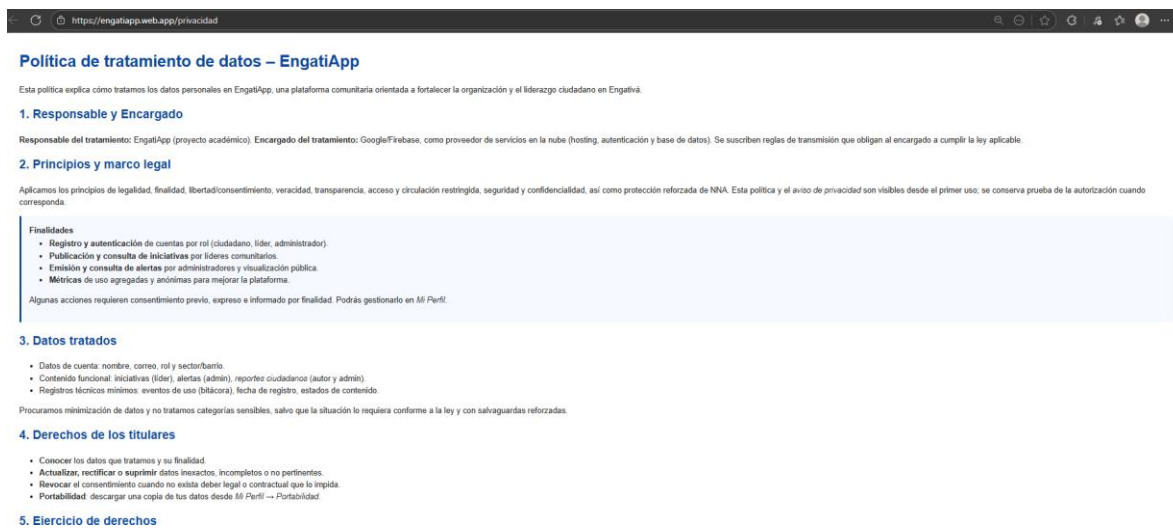
Un banner de estado offline cuando no hay conexión a Internet.

Pantalla de Inicio y Menú de Navegación



Tomado de J. Díaz Marín (2025).

Política de Tratamiento de Datos (Privacidad)



Tomado de J. Díaz Marín (2025).

Registro e Inicio de Sesión:

En la barra de navegación, hacer clic en “Registro/Login”.

Completar los campos:

Correo electrónico.

Contraseña.

Nombre completo.

Sector/Barrio (por ejemplo, Engativá Centro).

Rol (Ciudadano o Líder Comunitario).

En el bloque de Consentimiento de tratamiento de datos, marcar las finalidades permitidas:

Publicación de iniciativas.

Emisión/gestión de alertas (administradores).

Envío de reportes ciudadanos.

Métricas y estadísticas de uso.

Para registrar una cuenta nueva, pulsar “Registrarse”.

Para iniciar sesión con una cuenta existente, usar “Iniciar sesión”.

Si se olvida la contraseña, usar “¿Olvidaste tu contraseña?” y seguir las instrucciones recibidas por correo.

Una vez iniciada la sesión, en la barra superior se muestra el nombre, rol y sector del usuario, y se habilitan los botones correspondientes al rol.

Formulario de Registro/Login

EngotiApp

[Inicio](#) [Registro/Login](#) [Alertas](#) [Iniciativas](#) [Privacidad](#)

Registro / Inicio de sesión

Correo electrónico:

Contraseña:

Nombre completo:

Sector/Barría:

Rol:

Consentimiento de tratamiento de datos

Marca las finalidades permitidas. Puedes cambiarlos luego en "Mi Perfil".

Publicación de iniciativas	<input checked="" type="checkbox"/>
Emisión/gestión de alertas (solo admin)	<input checked="" type="checkbox"/>
Envío de reportes ciudadanos	<input checked="" type="checkbox"/>
Métricas y estadísticas de uso (anónimas)	<input checked="" type="checkbox"/>

[Iniciar sesión](#) [Registrarse](#) [¿Olvidaste tu contraseña?](#)

Tomado de J. Díaz Marín (2025).

Uso de los Módulos Principales:

Alertas comunitarias, en el menú seleccionar “Alertas”.

En la parte superior se encuentran filtros por sector y categoría (salud, educación, ambiente, etc.).

Debajo se lista cada alerta con:

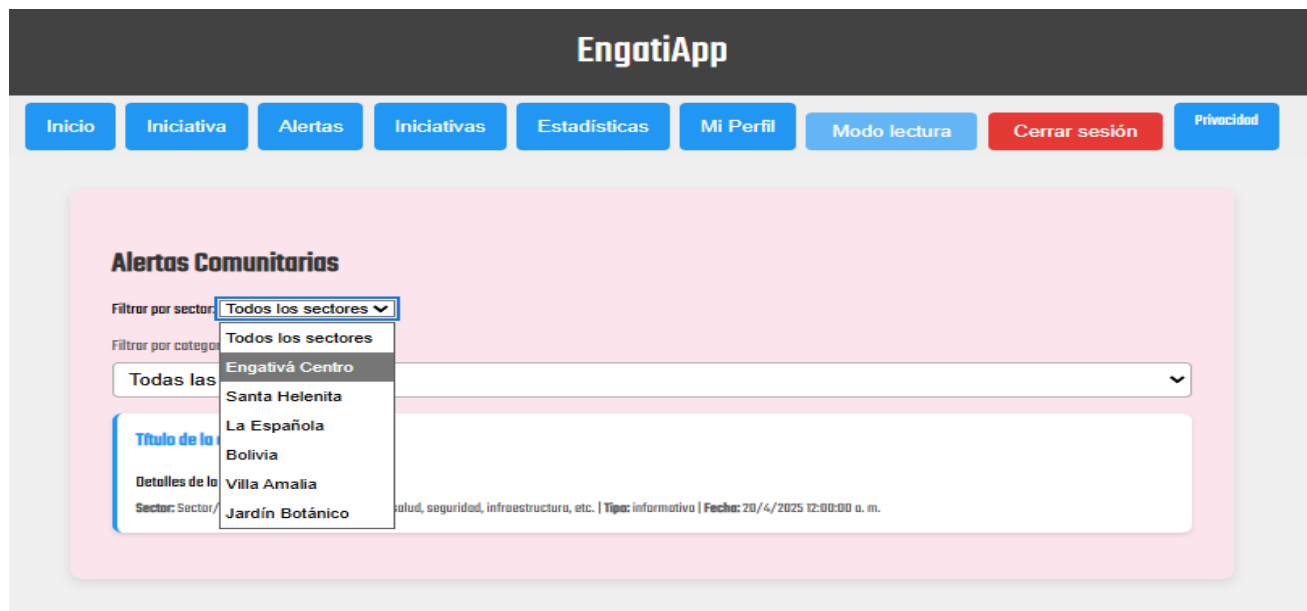
Título.

Descripción.

Sector, categoría y tipo.

Fecha de creación.

Listado y Filtros de Alertas



Tomado de J. Díaz Marín (2025).

Para administradores: Se muestra un formulario adicional “Emitir alerta (Administrador)” con campos de título, descripción, categoría, sector y tipo.

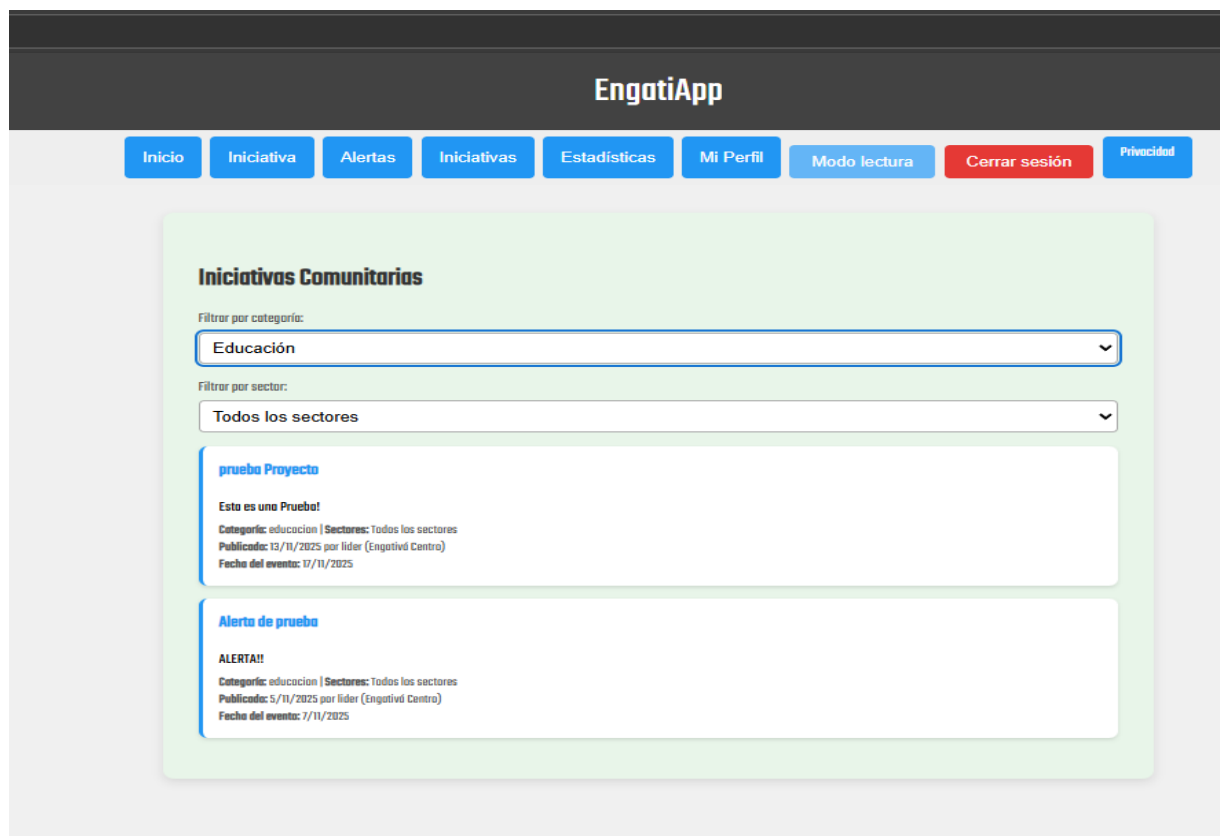
Completar los campos y pulsar “Publicar alerta” para que la alerta quede visible para todos los usuarios.

Iniciativas Comunitarias:

En el menú, seleccionar “Iniciativas”.

Aplicar filtros por categoría y sector si se desea.

Iniciativas: Filtros por Categoría y Sector



The screenshot displays the EngotiApp interface. At the top, there is a dark header with the 'EngotiApp' logo. Below the header is a navigation bar with buttons for 'Inicio', 'Iniciativa', 'Alertas', 'Iniciativas', 'Estadísticas', 'Mi Perfil', 'Modo lectura', 'Cerrar sesión', and 'Privacidad'. The main content area is titled 'Iniciativas Comunitarias' and features two filter dropdowns: 'Filtrar por categoría:' set to 'Educación' and 'Filtrar por sector:' set to 'Todos los sectores'. Below the filters, there are two event cards. The first card is titled 'prueba Proyecto' and contains the text: 'Esta es una Prueba!', 'Categoría: educación | Sectores: Todos los sectores', 'Publicado: 13/11/2025 por líder (Engotivó Centro)', and 'Fecha del evento: 17/11/2025'. The second card is titled 'Alerta de prueba' and contains the text: 'ALERTA!!', 'Categoría: educación | Sectores: Todos los sectores', 'Publicado: 5/11/2025 por líder (Engotivó Centro)', and 'Fecha del evento: 7/11/2025'.

Tomado de J. Díaz Marín (2025).

La lista muestra el título, la descripción, la categoría, los sectores objetivo, la fecha de publicación, el nombre del autor y la fecha del evento (si existe).

Publicar iniciativa (solo líderes comunitarios)

En el menú, seleccionar “Iniciativa”.

Completar:

Título de la iniciativa.

Descripción.

Categoría.

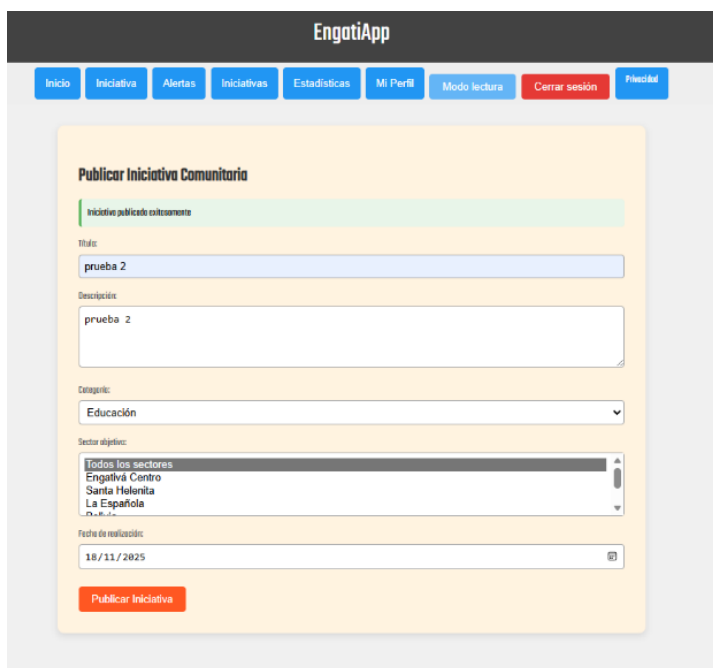
Sectores objetivo (uno o varios, o “Todos los sectores”).

Fecha de realización.

Pulsar “Publicar Iniciativa”.

La iniciativa aparecerá en el listado público de iniciativas y en las estadísticas.

Publicar Iniciativa



The screenshot shows the 'EngotiApp' interface for publishing a community initiative. At the top, there is a navigation bar with buttons for 'Inicio', 'Iniciativa', 'Alertas', 'Iniciativas', 'Estadísticas', 'Mi Perfil', 'Modo lectura', 'Cerrar sesión', and 'Ayuda'. Below this, the main content area is titled 'Publicar Iniciativa Comunitaria'. A green banner indicates 'Iniciativa publicada exitosamente'. The form includes the following fields:

- Título:** A text input field containing 'prueba 2'.
- Descripción:** A larger text input field containing 'prueba 2'.
- Categoría:** A dropdown menu with 'Educación' selected.
- Sector objetivo:** A multi-select dropdown menu with 'Todos los sectores', 'Engotivá Centro', 'Santa Helenita', and 'La Española' selected.
- Fecha de realización:** A date input field containing '18/11/2025'.

At the bottom of the form is a red button labeled 'Publicar Iniciativa'.

Tomado de J. Díaz Marín (2025).

Reporte Ciudadano:

En el menú, seleccionar “Reportar” Completar los campos:

Título.

Descripción del caso.

Categoría (salud, educación, ambiente, etc.).

Sector/Barrio.

Pulsar “Enviar reporte”.

Debajo se muestra la sección “Mis reportes”, donde se listan los reportes enviados, su categoría, sector, fecha y estado.

Reporte Ciudadano y “Mis Reportes”

The screenshot displays the EngotiApp interface. At the top, there is a navigation bar with the app name 'EngotiApp' and several menu items: 'Inicio', 'Alertas', 'Iniciativas', 'Reportar', 'Mi Perfil', 'Modo lectura', 'Cerrar sesión', and 'Privacidad'. Below the navigation bar, the 'Reporte ciudadano' form is visible. It includes a success message: 'Reporte enviado. Gracias por tu aporte.' The form fields are: 'Título' (with the example 'Ej: Luminaria dañada en parque del barrio'), 'Descripción' (with the placeholder 'Describe el caso, lugar y hora...'), 'Categoría' (set to 'Infraestructura'), and 'Sector/Barrio' (set to 'Engativá Centro'). A purple button labeled 'Enviar reporte' is located below the form. Below the form, the 'Mis reportes' section shows a list of reports. The first report is titled 'Luminaria dañada' and includes the description 'Luminaria dañada en la calle 85b con 86 esquina del parque.' and metadata: 'Categoría: Infraestructura | Sector: Engativá Centro | Estado: pendiente | Fecha: 13/11/2025 2:30:32 a. m.'

Tomado de J. Díaz Marín (2025).

Estadísticas (roles líder y admin):

En el menú, seleccionar “Estadísticas”.

Se muestran:

Tarjetas con totales de usuarios, líderes, iniciativas activas y alertas recientes.

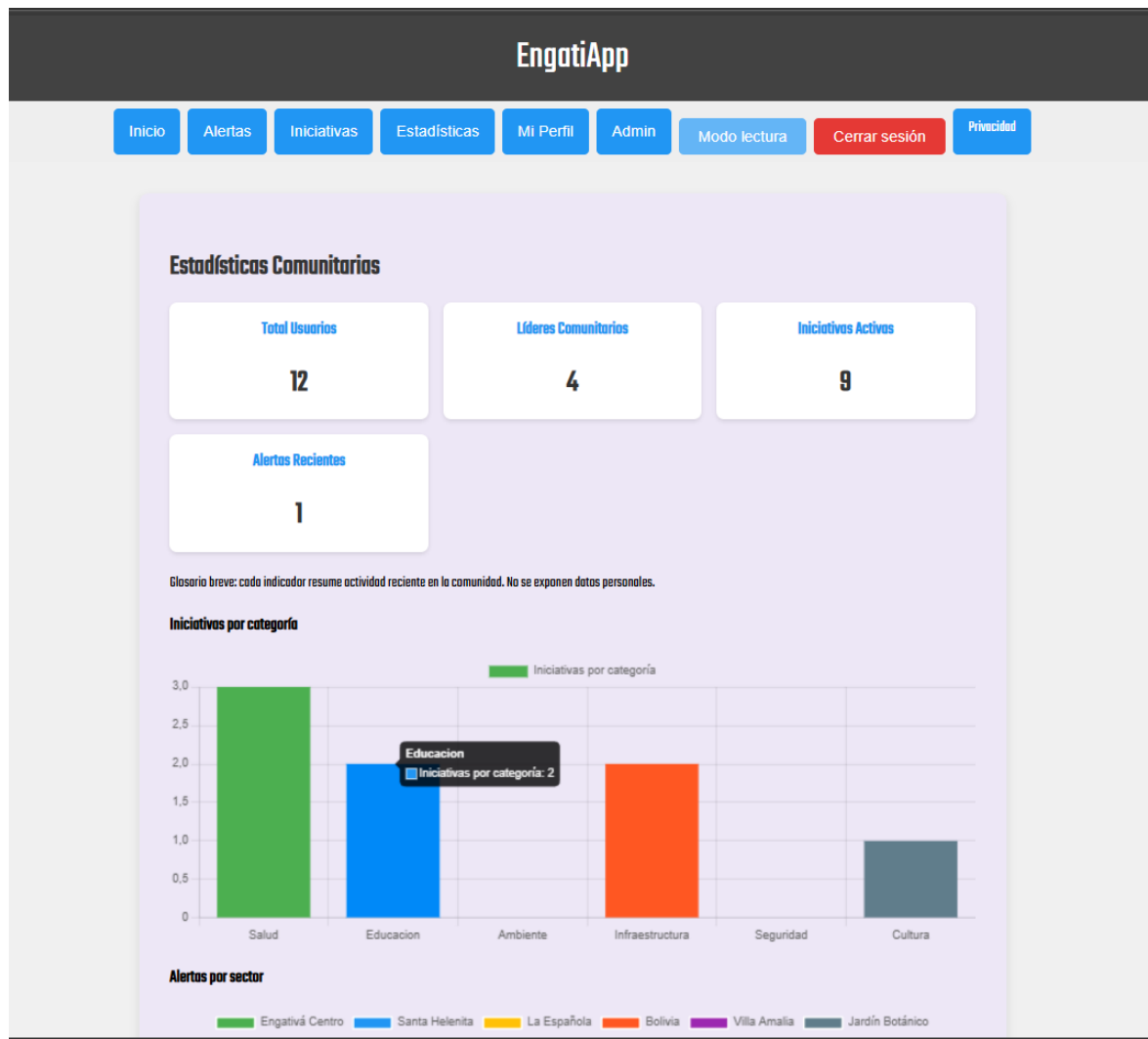
Gráfico de barras con iniciativas por categoría.

Gráfico de pastel con alertas por sector.

Gráfico de dona con usuarios por sector.

Estas estadísticas son anónimas y se basan en los datos almacenados en Firestore.

Estadísticas: Tarjetas y Gráficos



Tomado de J. Díaz Marín (2025).

Mi Perfil:

En el menú, seleccionar “Mi Perfil”.

El usuario puede:

Actualizar su nombre y sector.

Revisar su rol.

Marcar o desmarcar sus consentimientos (iniciativas, alertas, reportes, métricas).

Guardar cambios con el botón “Guardar cambios”.

En la misma pantalla se incluye un bloque de:

Portabilidad y supresión de datos, con botones para:

Descargar los datos en un archivo `mis_datos_engatiapp.json`.

Solicitar la eliminación de la cuenta.

Mi Perfil: Datos y Consentimientos

EngotiApp

[Inicio](#) [Alertas](#) [Iniciativas](#) [Estadísticas](#) [Mi Perfil](#) [Admin](#) [Modo lectura](#) [Cerrar sesión](#) [Privacidad](#)

Mi Perfil

Perfil actualizado.

Nombre completo

Sector/Barrio

Rol

Mis consentimientos

Publicación de iniciativas	<input checked="" type="checkbox"/>
Emisión/gestión de alertas	<input checked="" type="checkbox"/>
Envío de reportes ciudadanos	<input checked="" type="checkbox"/>
Métricas y estadísticas de uso	<input checked="" type="checkbox"/>

[Guardar cambios](#)

Portabilidad y supresión

[Descargar mis datos \(.json\)](#) [Eliminar mi cuenta](#)

Si eliminas tu cuenta, intentaremos borrar tu perfil de la base de datos. Es posible que necesites reautenticación.

Tomado de J. Díaz Marín (2025).

Administración (solo admin):

En el menú, seleccionar “Admin”.

Se muestran dos bloques principales:

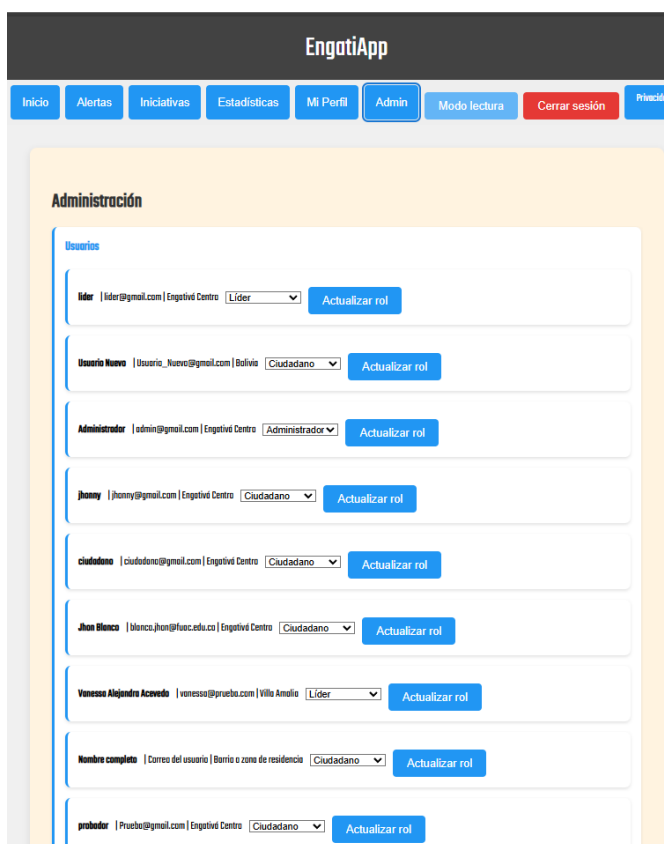
Usuarios:

Lista de usuarios con nombre, correo y sector.

Un selector para cambiar el rol (ciudadano, líder, administrador).

Un botón para “Actualizar rol” por cada usuario.

Administración: Gestión de Usuarios



Tomado de J. Díaz Marín (2025).

Reportes ciudadanos:

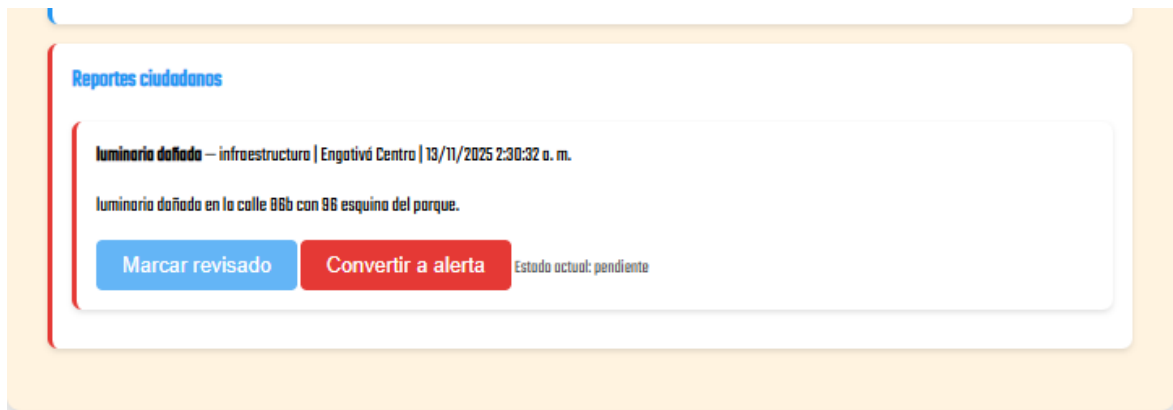
Listado de reportes enviados (título, categoría, sector, fecha, descripción).

Botones para:

“Marcar revisado” (actualiza el estado a revisado).

“Convertir a alerta” (crea una alerta a partir del reporte y marca el reporte como convertido).

Administración: Gestión de Reportes



Tomado de J. Díaz Marín (2025).

Accesibilidad y Ayuda:

EngatiApp incluye varias características de accesibilidad:

Enlace “Saltar al contenido” al inicio de la página, para saltar directamente al contenido principal usando teclado.

Navegación por teclado: los elementos interactivos pueden recorrerse con la tecla Tab.

Foco visible: al tabular, los elementos activos muestran un contorno resaltado.

Modo lectura (TTS): botón “Modo lectura” que activa la lectura en voz alta de los títulos de las pantallas.

Aviso de privacidad siempre disponible mediante el enlace “Privacidad”.

Mensaje de sin conexión cuando el navegador detecta que no hay Internet.

Si el usuario encuentra problemas en el uso de la plataforma (errores, textos poco claros, dificultades de lectura), se recomienda informar al equipo responsable del proyecto para su corrección en versiones posteriores.

Apéndice C

Evidencias de Trabajo Comunitario y Validación Exploratoria

Formulario de Encuesta Comunitaria (Microsoft Forms).

The image displays five sequential screenshots of a Microsoft Forms survey. Each screenshot shows a different section of the survey, with a consistent header and footer. The survey is titled "Encuesta comunitaria sobre participación y comunicación digital - Engativá (EngatiApp)".

- Section 0 - Consentimiento informado:** Includes a title, a paragraph of text, and a "Compartir" button.
- Section 1 - Perfil general:** Contains questions 2, 3, 4, and 5, each with a dropdown menu and radio button options.
- Section 2 - Conectividad:** Contains question 6 with radio button options and question 7 with radio button options.
- Section 3 - Tareas de uso:** Contains question 8 with a dropdown menu, question 9 with radio button options, question 10 with a Likert scale (1-5), question 11 with a text input field, question 12 with a dropdown menu, question 13 with radio button options, question 14 with a Likert scale (1-5), and question 15 with a text input field.
- Section 4 - Utilidad percibida por módulo:** Contains questions 20, 21, 22, and 23, each with a Likert scale (1-5).

Each screenshot also features a Microsoft 365 logo and a footer with a "Microsoft Forms" logo and a "Compartir" button.

Tomado de J. Díaz Marín (2025).

Mapa de Sueños Elaborado en Taller Participativo.

Tomado de J. Díaz Marín (2025).



Taller Participativo.


Nota. Se ocultan los rostros integrantes por privacidad.

Tomado de J. Díaz Marín (2025).

Tríptico Educativo para la Comunidad.

Estudiante: Jhonatan Díaz Marín
Programa: TDS
Centro: José Acevedo y Gómez
Zona: Bogotá
Acción: Conectando el barrio Bachué I
Lugar: Engativá, Bachué I
Participantes: 6 (2 jóvenes, 1 madre/comerciante, 3 vecinos)
Apoyo: 1 voluntario (foto/video)
Objetivo: Capacitar en EngatiApp para mejorar comunicación, publicar iniciativas, gestionar alertas y leer estadísticas; fortalecer liderazgo y participación digital.
Logros clave:

- 100% registrados y 1 iniciativa por persona
- 2 líderes identificados (administración y réplica)
- Alertas simuladas probadas
- Plan por cuadradas y uso responsable
- Alianzas: JAC + comercio local



Contáctanos

Contacto local: Jhonatan Díaz
3144104897

Administrador Engatiapp:
jdiazmari@unadvirtual.edu.co

Privacidad: datos mínimos, uso responsable, consentimiento.

Súmame: ciudadano · líder · aliado

engatiapp.web.app

ENGATIAPP


Participación ciudadana y alertas comunitarias

Bachué I · Engativá (Bogotá)
ODS: 9 · 11 · 16 · 17

¿QUÉ HACE?

- Iniciativas del barrio
- Alertas (seguridad/ambiente/convivencia Etc.)
- Estadísticas simples

¡Únete al Cambio!



Escanéame para entrar / registrarte

¿QUÉ ES ENGATIAPP?

Es tu agenda digital del barrio en el celular o el computador.

Un lugar único donde la comunidad puede ver avisos, visualizar iniciativas, compartir ideas y consultar datos sencillos sobre lo que pasa en Bachué I.

¿Para qué sirve?

Es un canal ordenado y común donde encontraras toda la información importante en un solo lugar.

Más participación y coordinación entre vecinos, líderes y aliados conectados.

Decisiones con datos básicos con números simples que ayudan a priorizar acciones en beneficio de la comunidad.

Queda registro de iniciativas, alertas y acuerdos del barrio.

COMO ENTRAR A LA APP

A. Primera vez (Registrarse)
Abrir la web de EngatiApp o escanear el QR. En el menú, ir a "Registro / Login".
Llenar los campos:
Correo electrónico, Contraseña, Nombre completo, Sector/Barrio (Engativá Centro, Santa Helenita, etc.)
Rol Ciudadano o Líder comunitario
Marcar las casillas de consentimiento según lo que quieres permitir:
Publicar iniciativas, Recibir/emitter alertas (solo Líderes)
Enviar reportes ciudadanos, Aceptar métricas anónimas
Pulsar "Registrarse". Verás un mensaje de usuario creado y luego podrás iniciar sesión.

B. Ya tengo cuenta (Ingresar)
Ir a "Registro / Login".
Escribir correo + contraseña.
Pulsar "Iniciar sesión".
Arriba verás tu nombre - rol - sector y el menú completo.
Para salir, usar el botón rojo "Cerrar sesión" en la barra superior.

Si olvidaste la contraseña
En "Registro / Login", escribir tu correo.
Pulsar "¿Olvidaste tu contraseña?".
Revisar tu correo y seguir el enlace para crear una nueva.

Apoyos de accesibilidad
Botón "Modo lectura": activa una voz que lee los títulos en voz alta (otl) para personas mayores).
Truco de letra grande:
PC: Ctrl + / Ctrl -
Celular: Ajustes → Accesibilidad → Tamaño de fuente

USO RÁPIDO POR MÓDULOS

1) INICIO (todos)
Muestra cuántas alertas, iniciativas y usuarios hay. Botones rápidos: Ver Alertas, Ver Iniciativas y (si eres líder/admin) Ver Estadísticas.

2) ALERTAS
Ver alertas: módulo "Alertas" → tarjetas con título, sector, categoría y fecha. Usar filtros por sector y categoría. Solo el admin puede crear nuevas alertas.

3) INICIATIVAS
Ver iniciativas: módulo "Iniciativas", filtra por categoría y sector. Publicar iniciativa: botón "Iniciativa" (solo líderes), completar datos y "Publicar".

4) REPORTE CIUDADANO
Módulo "Reportar" (rol ciudadano). Llenar título, descripción, categoría y sector. Pulsar "Enviar reporte" y revisar luego en "Mis reportes".

5) ESTADÍSTICAS
Solo para líder y admin (módulo "Estadísticas"). Ver tarjetas de totales y gráficos: iniciativas por categoría, alertas y usuarios por sector. Son datos agregados, sin información personal.

Normas básicas
Respeto - Nada de datos sensibles - Cierra sesión en equipos públicos.

Mini-glosario
Botón = acción - Menú = opciones - Filtro = mostrar solo lo que buscas.

Tomado de J. Díaz Marín (2025).

Ficha de validación exploratoria del prototipo:

El propósito es dejar constancia sintética de los recorridos críticos realizados sobre el prototipo, alineados con las tareas definidas en el documento principal.

Alcance probado:

- T1 Registro e inicio de sesión (RF1),
- T2 Publicación de iniciativa (RF2),
- T3 Emisión/consulta de alertas (RF3),
- T4 Visualización de estadísticas (RF4).

(Ver Tabla 4

Guion de Tareas (T1–T4) en el documento principal.)

Casos Funcionales Ejecutados:

Se siguieron los CP01–CP... para los flujos anteriores; cada caso registra datos de entrada, resultado esperado y resultado obtenido.

(Ver Tabla 5

Casos de Prueba Funcionales Ejecutados en el Prototipo de EngatiApp en el documento principal.)

Resultado General:

Los recorridos internos del desarrollador y pruebas exploratorias con usuarios cercanos evidenciaron funcionamiento correcto de los flujos principales y ajustes menores en textos de error/navegación.

(Ver Listado y Filtros de Alertas en el Apéndice b.)

Consideraciones Éticas y de Datos:

Las evidencias excluyen datos personales identificables; cualquier información sensible fue anonimizada.

Trazabilidad:

La definición de técnicas mixtas encuesta comunitaria y taller participativo/“diario de sueños” se presenta en Técnicas e Instrumentos de Recolección.

La evidencia visual correspondiente se incluye en Apéndice C: Formulario de encuesta en Microsoft Forms y Figura Diario/mapa de sueños de la comunidad.

El guion de tareas T1–T4 se documenta en Tabla 4

Guion de Tareas (T1–T4) y los casos de prueba CP en Tabla 5

Casos de Prueba Funcionales Ejecutados en el Prototipo de EngatiApp del documento principal.

Las capturas empleadas durante los recorridos críticos están en Apéndice B y se referencian así:

T1 Registro/Login →Formulario de Registro/Login (Formulario de Registro/Login).

T2 Publicar iniciativa →

Publicar Iniciativa (Publicar iniciativa).

T3 Alertas (consultar/filtrar) → Listado y Filtros de Alertas (Listado y filtros de alertas).

T4 Estadísticas → Estadísticas: Tarjetas y Gráficos (Estadísticas: tarjetas y gráficos).

(Si se incluyen casos adicionales en el cuerpo del documento):

CP05 Reporte ciudadano → Reporte Ciudadano y “Mis Reportes” (Reporte ciudadano y “Mis reportes”).

CP06 Administración: cambio de rol → Administración: Gestión de Usuarios (Administración: gestión de usuarios).

CP07 Administración: gestión de reportes → Administración: Gestión de Reportes (Administración: gestión de reportes).

Mejora Continua:

Las plantillas para métricas de usabilidad y registro de hallazgos se encuentran en Tabla 6
Plan de Métricas para Futuras Pruebas de Usabilidad a Tabla 9
Plantilla de Evidencias (Figuras/Capturas) del documento principal (p. ej., plan de métricas, checklist de accesibilidad, registro HZ y checklist operativo). Su aplicación cuantitativa se proyecta para la fase piloto ampliada; en esta entrega se reportan resultados cualitativos de recorridos críticos.