

Universidad Tecnológica Nacional
Facultad Regional Córdoba
Ingeniería en Sistemas de información

Cátedra: Proyecto Final



Curso: 5K4 - Grupo N°: 15 - Año: 2026

Integrantes:

- | | |
|---------|-----------------------------|
| - 82770 | Andrade Patricio Valentin |
| - 87661 | Ayarzabal Fernandez Amilcar |
| - 81165 | Belbruno Franco Manuel |
| - 85308 | Camargo Mano Juan Ignacio |
| - 85206 | Paredes Genaro |
| - 60377 | Virgolini Rodrigo |

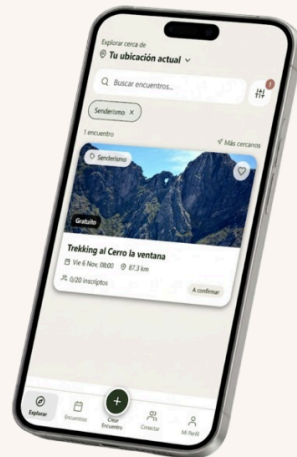
Docentes:

- Ing. Jaime, María Natalia
- Ing. Arenas, Silvina



Trama

Una red social para salir al mundo real.



Nuestro propósito



Facilitar que las personas descubran, organicen y vivan encuentros presenciales a partir de intereses compartidos, priorizando la experiencia en el mundo real por sobre la permanencia dentro de la pantalla.

¿Qué ofrecemos?



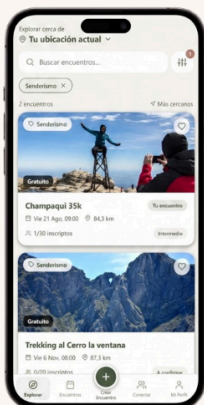
Construí comunidad

Conectate con personas, intereses y espacios que promueven vínculos fuera de la pantalla.



Encontrá qué hacer

Descubrí planes, propuestas y actividades cerca tuyo según tus intereses.



Organizá encuentros fácil

Creá actividades, definí lugar, fecha, cupos y condiciones de participación.



Reservá y participá

Sumate a experiencias reales con información clara, asistentes y detalles relevantes.



Tecnologías y herramientas



React Native



Expo



TypeScript



Node.js



Express



Supabase



PostgreSQL



Jira



GitHub

Metodología Scrum • Story Mapping • ADR

UTN - Facultad Regional Córdoba

Cátedra Proyecto Final - Año 2026
Ingeniería en Sistemas de Información

Docentes

- Natalia Jaime
- Silvina Arenas

Integrantes del equipo

- Andrade Patricio Valentín
- Ayarzabal Fernández Amilcar
- Belbruno Franco Manuel
- Camargo Mano Juan Ignacio
- Paredes Genaro
- Virgolini Rodrigo



Trama: de los intereses compartidos a los encuentros presenciales

Andrade Patricio Valentin · Ayarzal Fernandez Amilcar · Belbruno Franco Manuel
Camargo Mano Juan Ignacio · Paredes Genaro · Virgolini Rodrigo

Universidad Tecnológica Nacional, Facultad Regional Córdoba

Abstract

Trama se desarrolló como una aplicación móvil orientada a centralizar el descubrimiento, la organización y la participación en encuentros presenciales. La propuesta se fundamentó en un relevamiento de potenciales usuarios, que permitió identificar barreras sociales y logísticas vinculadas con la falta de compañía, la dificultad para encontrar actividades y la información dispersa. El proyecto se organizó mediante Scrum, User Story Mapping y un backlog priorizado, mientras que las decisiones técnicas se registraron mediante Architectural Decision Records. La solución utilizó React Native con Expo y TypeScript para la aplicación móvil, Node.js y Express para la API, y Supabase PostgreSQL para persistencia, autenticación y almacenamiento. Como resultado se obtuvo un producto funcional que integra registro y perfil, intereses y nivel, exploración y detalle de encuentros, ubicación en mapa, creación y administración de actividades, inscripción, gestión de cupos, recurrencia, borradores, quórum, listado de asistentes y un panel estadístico para organizadores. La solución consolidó un recorrido digital destinado a facilitar la concreción de actividades reales.

Palabras clave— encuentros presenciales, comunidad, aplicación móvil, Scrum, React Native, producto.

Introducción

La difusión y coordinación de actividades presenciales se resuelve habitualmente mediante redes sociales, grupos de mensajería, formularios y recomendaciones informales. Esta dispersión obliga a consultar distintos canales para conocer una propuesta, verificar cupos, identificar requisitos y confirmar asistencia. La problemática también comprende la incertidumbre de incorporarse a grupos desconocidos, la falta de acompañantes y la ausencia de referencias suficientes sobre organizadores y participantes [1].

El relevamiento inicial de Trama, realizado mediante encuesta a potenciales usuarios, mostró que el 61,6 % correspondió a personas de 18 a 24 años y el 17,8 % a personas de 25 a 34 años. Entre las barreras más mencionadas se encontraron “no tengo con quién ir” (63 menciones), “no encuentro actividades fácilmente” (42), mala organización

(30) y falta de información clara (28). Ante la propuesta, el 56,2 % indicó que la utilizaría y el 34,2 % que tal vez lo haría. Buscar actividades cercanas y filtrar por tipo fueron las capacidades más valoradas, con 117 y 100 menciones respectivamente [1].

El análisis de Luma, Fever, Strava y Meetup mostró soluciones especializadas en difusión, ticketing, actividad física o comunidades de interés. Trama propone integrar descubrimiento, gestión y confianza dentro de una experiencia móvil cuyo objetivo no es prolongar el tiempo de pantalla, sino reducir la fricción entre la intención de participar y la concreción de una actividad presencial [1], [2].

Herramientas y metodología

El proyecto se abordó como un producto de software con un alcance funcional definido para el ciclo académico. Se aplicó Scrum en iteraciones de dos semanas, con instancias de planificación, revisión y retrospectiva; el seguimiento cotidiano del trabajo se realizó de forma asincrónica.

El User Story Mapping organizó el recorrido en acceso, configuración de perfil, descubrimiento, reserva y asistencia, gestión de encuentros, confianza, comunicación y evolución futura. Esta representación permitió priorizar el circuito principal de valor y mantener trazabilidad sobre funcionalidades complementarias, como pagos integrados o recomendaciones avanzadas [2], [3]. Jira se utilizó para gestionar el Product Backlog; GitHub, para el versionado y las revisiones mediante Pull Requests; y una estrategia multirepositorio separó aplicación móvil, backend, base de datos, documentación y pruebas automatizadas. La trazabilidad vinculó historias, ramas, cambios, evidencias y documentación [4], [5].

La arquitectura se compuso de una aplicación React Native y Expo en TypeScript, una API REST en Node.js, Express y TypeScript, y persistencia relacional mediante Supabase PostgreSQL. Supabase Auth y Storage cubrieron identidad y recursos multimedia. Las decisiones

principales se documentaron mediante ADR, incluyendo la adopción de una arquitectura en capas, el uso de una API REST y la separación de repositorios [4].

La estrategia de calidad combinó pruebas unitarias, de integración, exploratorias, de regresión y de flujo completo. Se definió un repositorio externo de testing con TypeScript y Playwright para pruebas de API, contratos, smoke tests y recorridos end-to-end sobre Expo Web. Los cambios relevantes se asociaron con criterios de aceptación, evidencia y revisión cruzada [5].

La redacción del artículo contó con asistencia de herramientas de inteligencia artificial. Los datos, decisiones y conclusiones fueron revisados por el equipo contra la documentación del proyecto, conforme al acuerdo de uso responsable de inteligencia artificial del equipo [6].

Resultados

La versión desarrollada permitió crear una cuenta mediante Google o correo electrónico y completar un perfil con intereses y nivel. También se implementaron un muro y un feed vinculados con su actividad, que funcionaron como punto de entrada para la exploración de propuestas y la consulta de información personal.

El módulo de descubrimiento centralizó el listado de encuentros y permitió consultar su información, ubicación y condiciones de participación. La vista de detalle presentó los datos necesarios para decidir la asistencia, mientras que la sección ‘Mis Encuentros’ reunió las actividades asociadas al usuario. La inscripción registró la participación y permitió al organizador consultar la lista de asistentes.

Para la creación de encuentros se implementó un asistente dividido en etapas, con reglas de validación para fecha, ubicación, cupos, nivel y descripción. También se implementaron la edición y cancelación de actividades, la creación de encuentros recurrentes, el guardado de borradores, la definición de cupo mínimo o quórum y la consulta de un dashboard estadístico. El panel de gestión concentró la administración de la propuesta y de las personas inscritas.

La integración de estos módulos conformó dos recorridos complementarios. El recorrido del participante comprendió el registro, la configuración de identidad, el descubrimiento de una actividad, la consulta de sus condiciones y la inscripción. El recorrido del organizador incluyó la publicación de propuestas, la definición de condiciones, la gestión de asistentes y la consulta de información de seguimiento. Ambos recorridos se ejecutaron desde un único perfil,

evitando la separación entre cuentas de participante y organizador.

El producto fue acompañado por documentación funcional y técnica, convenciones de ramas, revisiones mediante Pull Requests, líneas base y registros de prueba. Esta estructura permitió mantener correspondencia entre necesidades, historias de usuario, contratos de API y componentes implementados. La estrategia de pruebas aportó validaciones funcionales e integradas sobre los principales recorridos del sistema [3]–[7]. El modelo de información contempló relaciones entre usuarios, encuentros, participaciones, seguimiento y reseñas, lo que permitió representar tanto el rol de participante como el de organizador sin duplicar identidades. También se definieron reglas para proteger secretos, separar configuraciones por ambiente, registrar errores relevantes y evitar el versionado de credenciales. Estas decisiones establecieron una base de operación coherente con los requerimientos de seguridad, mantenibilidad y crecimiento previstos para el producto.

Discusión

Las capacidades desarrolladas mantienen correspondencia con las barreras detectadas en el relevamiento. La exploración centralizada atende la dificultad para localizar propuestas; el detalle, el mapa, los cupos y el quórum reducen la falta de información; y la inscripción junto con la visualización de asistentes disminuye el esfuerzo de coordinación. El perfil y el historial constituyen la base para fortalecer progresivamente la confianza entre usuarios. La posibilidad de alternar entre participante y organizador también responde a la naturaleza comunitaria del dominio, donde una misma persona puede asistir a una actividad y crear otra sin requerir cuentas o circuitos separados.

A diferencia de plataformas centradas exclusivamente en difusión, ticketing o seguimiento deportivo, Trama integra el recorrido desde el descubrimiento hasta la participación dentro de un mismo producto. Su propuesta de valor se encuentra en facilitar actividades comunitarias o de escala acotada y en permitir que una misma persona alterne entre los roles de participante y organizador sin abandonar la aplicación [1]. La experiencia se diseña alrededor de información accionable —fecha, ubicación, nivel, cupos, asistentes y condiciones— para evitar que el usuario deba completar el proceso mediante publicaciones, formularios y conversaciones dispersas. De este modo, el componente social se utiliza como soporte de la actividad presencial y no como un fin basado en la permanencia dentro de la plataforma.

La arquitectura multiplataforma y la separación de responsabilidades favorecen la mantenibilidad y la evolución de la solución. El uso de TypeScript en frontend y backend reduce inconsistencias de contrato, mientras que PostgreSQL permite modelar relaciones entre usuarios, encuentros, participaciones y futuras reseñas. Supabase concentra servicios administrados de autenticación, persistencia y almacenamiento, disminuyendo la infraestructura que el equipo debe mantener. Como contrapartida, la estrategia multirepositorio exige coordinación entre contratos, ambientes y releases [4], [5]. La documentación de decisiones mediante ADR resulta relevante para conservar el razonamiento técnico y evitar que futuras modificaciones se realicen sin considerar los trade-offs originalmente evaluados.

El alcance presentado se concentra en el circuito principal de identidad, descubrimiento, gestión e inscripción. Los pagos integrados y las recomendaciones avanzadas se mantienen como evolución posterior. Esta delimitación evita incorporar complejidad transaccional o algorítmica antes de consolidar el proceso que genera valor principal. Asimismo, la encuesta valida el interés previo, pero no mide todavía adopción sostenida ni impacto social después del uso. Una evaluación futura deberá incorporar pruebas de campo y métricas de conversión entre visualización e inscripción, porcentaje de asistencia efectiva, recurrencia de participación, cancelaciones y percepción de confianza. También será necesario observar si la incorporación de funciones sociales mejora la concreción de encuentros o, por el contrario, aumenta el tiempo de uso sin aportar actividad presencial.

Conclusión

Trama constituye una solución móvil que centraliza información y herramientas necesarias para descubrir, organizar y participar en encuentros presenciales. El producto transforma una necesidad relevada en un recorrido funcional que integra identidad, perfil, exploración, detalle, creación, administración e inscripción. La metodología incremental y la arquitectura seleccionada permitieron sostener trazabilidad entre el problema, las decisiones de diseño y la solución construida. La integración de ambos roles dentro de una misma identidad simplifica la experiencia y refuerza la idea de una comunidad en la que las personas pueden proponer y compartir actividades, además de asistir a ellas.

La evolución del producto contempla mecanismos adicionales de reputación, validación de asistencia, comunicación y recomendación, manteniendo como criterio central que la

tecnología facilite la acción presencial en lugar de reemplazarla. Las futuras validaciones con usuarios permitirán determinar en qué medida Trama reduce la fricción social y logística y favorece la concreción de actividades compartidas. El análisis de uso deberá combinar métricas cuantitativas con observaciones cualitativas sobre claridad de la información, facilidad de organización y confianza al incorporarse a grupos desconocidos. Estos resultados permitirán priorizar nuevas capacidades sin perder la orientación original del producto.

Referencias

- [1] Equipo Trama, “Estudio Inicial”, versión 0.4, Universidad Tecnológica Nacional, Facultad Regional Córdoba, 2026.
- [2] Equipo Trama, “Plan de Proyecto”, versión 0.4, Universidad Tecnológica Nacional, Facultad Regional Córdoba, 2026.
- [3] Equipo Trama, “User Story Mapping y Backlog Inicial”, versión 0.3, Universidad Tecnológica Nacional, Facultad Regional Córdoba, 2026.
- [4] Equipo Trama, “Arquitectura y Stack Tecnológico”, versión 0.4, Universidad Tecnológica Nacional, Facultad Regional Córdoba, 2026.
- [5] Equipo Trama, “Plan de Gestión de la Configuración” y “Plan de Testing”, versiones 0.4, Universidad Tecnológica Nacional, Facultad Regional Córdoba, 2026.
- [6] Equipo Trama, “Working Agreement”, versión 0.4, Universidad Tecnológica Nacional, Facultad Regional Córdoba, 2026.
- [7] Equipo Trama, “Planificación y Seguimiento de Sprints”, Universidad Tecnológica Nacional, Facultad Regional Córdoba, 2026.
- [8] K. Schwaber y J. Sutherland, “The Scrum Guide”, noviembre de 2020. Disponible: <https://scrumguides.org>. Último acceso: julio de 2026.
- [9] Meta Platforms, Inc., “React Native Documentation”. Disponible: <https://reactnative.dev/docs/getting-started>. Último acceso: julio de 2026.
- [10] Expo, “Expo Documentation”. Disponible: <https://docs.expo.dev>. Último acceso: julio de 2026.
- [11] Supabase, “Supabase Documentation”. Disponible: <https://supabase.com/docs>. Último acceso: julio de 2026.
- [12] Microsoft, “Playwright Documentation”. Disponible: <https://playwright.dev/docs/intro>. Último acceso: julio de 2026.

Agradecimientos

A la cátedra de Proyecto Final de la UTN Facultad Regional Córdoba por el acompañamiento, la guía y las instancias de revisión brindadas durante el desarrollo del proyecto.

Datos de contacto

Equipo Trama · Universidad Tecnológica Nacional, Facultad Regional Córdoba · Correo de contacto: tramaapp26@gmail.com

AÑO	2026	CURSO Y NRO. DE GRUPO	5K4 - G15
NOMBRE DEL SISTEMA / PROYECTO			
Trama			
CATEGORÍA (Solución De Negocio / Producto / Proyecto De Impacto Social)			
Producto			
HERRAMIENTAS Y TECNOLOGÍAS UTILIZADAS			
ÁMBITO DE APLICACIÓN		NOMBRE Y VERSIÓN	
ENTORNO DE DESARROLLO		Visual Studio Code	
REPOSITORIOS Y VERSIONADO		GitHub - Google Drive	
PROGRAMACIÓN		TypeScript, React Native, Expo, Node.js, Express	
BASE DE DATOS		PostgreSQL (Supabase)	
COMUNICACIÓN INTERNA		Whatsapp	
CAPACITACIÓN		YouTube, Documentación de tecnologías	
PRUEBAS DE SISTEMA		Casos de Prueba - Playwright	
GESTIÓN DEL PROYECTO		Jira	
DOCUMENTACIÓN		Google Docs - Word	
MODELOS		Miro - Mermaid - Draw.io	