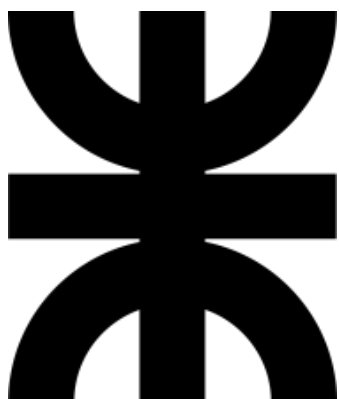


Universidad Tecnológica Nacional

Facultad Regional Córdoba

Ingeniería en Sistemas de Información



Cátedra de Proyecto Final 2026

SeBar: Sistema de Seguridad Barrial

Profesores:

Natalia Jaime
Victoria Ahumada
Felipe Adrián Steffolani
María Silvina Arenas
Francisco Alejandro Aquino

Curso: 5K4

Autores:

Tomás Barrionuevo	95012
Daniel Barrionuevo Siri	88618
Gabriel Emanuel Gonza	424123
Franco González Berges	98913
Franco Recalde	94661
Julián Torres Ramírez	94984



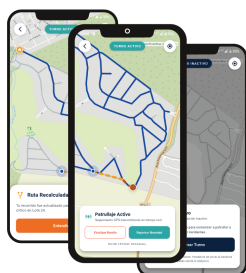
SeBar

Seguridad Barrial

Plataforma de patrullaje preventivo en respuesta a la baja trazabilidad de rondines tradicionales, el riesgo variable por sectores de vigilancia y la comunicación fragmentada ante emergencias.

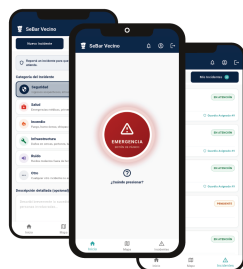


Para digitalizar y optimizar la seguridad barrial mediante itinerarios de patrullaje preventivo estructurados sobre grafos y respuesta a emergencias en tiempo real.



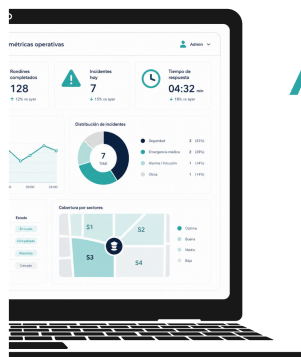
Guardia de Seguridad

- Cálculo optimizado de rondines
- Recálculo de ruta ante incidente
- Coordinación del cuerpo de seguridad



Vecino o Propietario

- Botón antipánico para emergencias
- Canal de reporte de incidentes menores
- Respuesta física inmediata



Administrador

- Visión operativa unificada en tiempo real
- Trazabilidad de incidentes y rondines
- Generación de métricas operativas



Garantiza máxima cobertura en sectores perimetrales críticos, reduce los tiempos de atención de incidentes y brinda trazabilidad auditable a la administración.



Tecnologías & Herramientas



Equipo

Tomás Barrionuevo
Daniel Barrionuevo Siri
Gabriel Emanuel Gonza
Franco González Bergés
Franco Recalde
Julián Torres Ramírez

Contacto

tbarrionuevo2210@gmail.com
danielbsiri@gmail.com
gonzagabriel428@gmail.com
franco20047@gmail.com
francorecalde15@gmail.com
jul14n.torres@gmail.com

Docentes

Natalia Jaime (Adjunta), Victoria Ahumada (JTP)

Cátedra: Proyecto Final 2026



FACULTAD
REGIONAL
CÓRDOBA
INGENIERÍA EN SISTEMAS DE INFORMACIÓN



SeBar: Plataforma Inteligente de Seguridad Barrial basada en Ruteo Optimizado sobre Grafos y Despacho en Tiempo Real

Recalde, Franco; Torres, Julián; González, Franco; Barrionuevo, Tomás; Gonza, Gabriel Emanuel; Barrionuevo, Daniel

Universidad Tecnológica Nacional, Facultad Regional Córdoba

Abstract — El presente trabajo abordó el diseño, desarrollo e implementación de SeBar, un sistema integral para digitalizar y optimizar la gestión de seguridad y patrullaje preventivo en barrios urbanos y suburbanos [1]. La problemática se fundamentó en la falta de itinerarios estructurados en las rondas de vigilancia y la fragmentación de canales de comunicación ante emergencias [1]. Se desarrolló una solución modular compuesta por dos aplicaciones móviles Android (propietarios y guardias), una plataforma web de administración y un motor de ruteo en el backend [1, 3]. El sistema modeló la red vial del predio como un grafo ponderado y aplicó la metaheurística GRASP (Greedy Randomized Adaptive Search Procedure) para optimizar los recorridos de patrullaje, priorizando tramos críticos [1]. Asimismo, se implementó un mecanismo de despacho de incidentes con comunicación en tiempo real vía WebSockets y recálculo dinámico de rutas cortas (Shortest Path) [3]. Los resultados permitieron validar la cobertura completa del perímetro, la trazabilidad de eventos y la reducción de tiempos de atención, obteniéndose una plataforma auditable y escalable [1, 2].

Palabras Clave — Seguridad Barrial, Patrullaje preventivo, Ruteo en Grafos, CARP, GRASP, WebSockets, PostGIS, FastAPI. Categoría: Producto.

Introducción

En la actualidad, la seguridad operativa en barrios cerrados de la provincia de Córdoba presenta desafíos vinculados a la falta de trazabilidad e ineficiencia en el uso de recursos de vigilancia [1]. La modalidad de patrullaje habitual se ejecuta sin un itinerario predefinido ni puntos de control cuantificables [1]. Esta práctica genera una cobertura desigual, provocando que ciertos sectores sean transitados repetidamente mientras áreas vulnerables, como los límites perimetrales y cruces de arroyos, permanecen sin supervisión por lapsos prolongados [1].

A esto se suma la fragmentación de los canales de comunicación ante emergencias

[1]. Los residentes deben optar entre llamadas telefónicas, servicios públicos o grupos informales de mensajería, sin garantía de transmisión automática de coordenadas ni notificación al guardia más cercano [1].

En este contexto surge SeBar, una plataforma integral de seguridad barrial cuyo objetivo es transformar el modelo de vigilancia tradicional en un esquema preventivo, optimizado e inteligente [1, 2]. El sistema toma como caso de estudio real al barrio *Carlos Paz Golf Country Club*, caracterizado por un perímetro con puntos críticos de acceso no controlado en los cruces del arroyo central y una marcada estacionalidad del riesgo [1]. SeBar aporta un modelo de ruteo adaptativo basado en investigación operativa que prioriza tramos según su nivel de riesgo y centraliza la atención de eventos en tiempo real, proporcionando herramientas de analítica y auditoría a la administración del barrio [1, 2].

Elementos del Trabajo y Metodología

Se aplicó la metodología ágil Scrum, organizada en sprints de dos semanas [1]. La arquitectura se diseñó bajo el Modelo C4 (niveles de Contexto, Contenedores y Despliegue) [3].

El sistema se estructuró en una arquitectura cliente-servidor desacoplada. El backend principal se desarrolló sobre FastAPI (Python) para el procesamiento asíncrono de datos GPS y conexiones concurrentes [3]. La persistencia se implementó en PostgreSQL con PostGIS, gestionado mediante el ORM SQLAlchemy y Alembic para el control de versiones del esquema [3].

Para la gestión en tiempo real se incorporó un servidor Redis [3]. Redis actuó como caché efímera almacenando las posiciones de guardias con un tiempo de vida (TTL) de 30 segundos y un *rate limiting* de 5 segundos por dispositivo para evitar la saturación de la red móvil [3]. Paralelamente, un hilo en segundo plano (*snapshot loop*) persistió las coordenadas desde Redis hacia PostGIS cada 30 segundos para garantizar el almacenamiento histórico de las trayectorias [3].

La red vial del barrio se importó desde OpenStreetMap y se modeló como un grafo ponderado $G = (V, E)$, sobre el cual se modelaron los tramos de patrullaje según su prioridad de vigilancia [1]. Para los rondines preventivos se implementó la metaheurística GRASP (Greedy Randomized Adaptive Search Procedure) [1]. La función de costo de cada arista se definió como:

$$patrol_cost = length_m / priority$$

Donde *priority* pondera la relevancia de vigilancia de cada tramo (con valores más altos en el perímetro y el arroyo) [1]. El algoritmo penalizó aristas recorridas recientemente y aplicó una búsqueda local Or-opt(1) para reducir tránsitos innecesarios [1]. En casos de múltiples guardias, se dividió el grafo en zonas disjuntas antes de calcular las trayectorias [1].

Para la comunicación bidireccional en tiempo real se dispuso un canal WebSocket en `/ws/tracking` [3]. Ante un incidente (creado mediante coordenadas explícitas o la vinculación automática de la propiedad del usuario), el backend calculó una ruta directa preliminar utilizando el algoritmo de camino mínimo (Shortest Path) sobre el grafo completo, enviando una alerta Push asíncrona hacia las aplicaciones móviles de los guardias candidatos [1, 3].

El despliegue en producción se orquestó mediante plataformas cloud PaaS: el backend y el motor de optimización se alojaron en Render, la base de datos espacial en Supabase, el panel web administrativo en Vercel, la distribución de notificaciones en Firebase FCM, y los mensajes SMS de emergencia vía la pasarela Twilio [3]. El pipeline de integración y despliegue continuo (CI/CD) se ejecutó mediante CircleCI [3].

Resultados

Se obtuvo como resultado una plataforma integrada compuesta por tres módulos operativos interconectados [1, 2].

El Módulo para Propietarios permitió la creación de reportes de incidentes clasificados según categoría y nivel de prioridad (LOW, MEDIUM, HIGH, CRITICAL), con geolocalización automática e inclusión de observaciones o evidencia [1, 2].

El Módulo para Guardias desplegó sobre el dispositivo móvil la representación gráfica del mapa del barrio con el itinerario de patrullaje optimizado mediante GRASP [1, 2]. Ante la notificación de un incidente por WebSocket, la aplicación mostró la alerta con la ruta de respuesta directa (*shortest path*) [1, 3]. La aceptación del evento por parte del guardia cambió el estado a IN_PROGRESS, registró la asignación y canceló automáticamente el rondín activo para preservar las métricas [1].

La Plataforma Web de Administración proporcionó a los supervisores la trazabilidad completa del ciclo de vida de los incidentes (PENDING, IN_PROGRESS, RESOLVED), visualización de observaciones de campo y *dashboards* con indicadores clave de rendimiento (KPIs), tales como tasas de resolución, frecuencias de patrullaje por sector y distribución temporal de eventos [1, 2].

Indicador / Métrica	Valor / Comportamiento Obtenido
Cobertura de Aristas Requeridas	100% de tramos perimetrales
Frecuencia en Zonas Críticas	2.5x respecto a zonas medias
Latencia en Alerta Push (WS)	< 800 ms
Tiempo de Despacho e Ingesta GPS	Ventana de 5s (Rate Limit)
Persistencia Histórica Snapshot	Intervalos de 30s a PostGIS

Tabla 1. Métricas de desempeño obtenidas en el entorno de pruebas integradas [4].

Discusión

Los resultados validan que la formalización del patrullaje mediante modelos de ruteo supera el esquema tradicional [1]. La aplicación de GRASP garantiza que las calles vulnerables reciban mayor frecuencia de paso [1].

La integración del recálculo dinámico de rutas ante emergencias resuelve la desconexión histórica entre el reporte del vecino y la acción del guardia [1]. Asimismo, la decisión de cancelar el rondín activo cuando un guardia acepta un incidente evita la distorsión de los indicadores de patrullaje, asegurando que las métricas reflejen con precisión el tiempo efectivo dedicado a la prevención versus la atención de eventos [1].

Entre las limitaciones se destaca la dependencia de la red celular para la transmisión continua por WebSocket [3]. Igualmente, la sincronización offline por lotes (/tracking/positions/sync) mitiga la pérdida de datos históricos en zonas de sombra [3].

Conclusión

El desarrollo de SeBar demuestra la viabilidad técnica y operativa de aplicar investigación operativa y tecnologías web en tiempo real a la seguridad privada

barrial [1, 3]. La plataforma logra sustituir procesos informales por un flujo de trabajo digitalizado, auditable y transparente para propietarios, guardias y administradores.

Como líneas de trabajo futuro, se proyecta la incorporación de módulos de análisis predictivo sobre los datos históricos de incidentes para ajustar automáticamente las prioridades de los tramos en el motor GRASP según la franja horaria y la época del año [1, 2]. Asimismo, se contempla el soporte nativo para dispositivos con sistema operativo IOS [1, 2].

Agradecimientos

Un especial agradecimiento a la administración del barrio *Carlos Paz Golf Country Club* por brindar el contexto operativo y los datos; asimismo, expresamos nuestra gratitud a la Cátedra de Proyecto Final y a las profesoras Natalia Jaime y Victoria Ahumada por su constante guía y soporte.

Referencias

- [1] Equipo SeBar, "Estudio Inicial y Alcance del Sistema SeBar," UTN FRC, 2026.
- [2] Equipo SeBar, "Definición del Alcance v1.0 - SeBar," UTN FRC, 2026.
- [3] Equipo SeBar, "Informe de Arquitectura C4 y Despliegue v2.0 - SeBar," UTN FRC, 2026.
- [4] Equipo SeBar, "Pruebas Integrales y End2End v1.0 - SeBar," UTN FRC, 2026.
- [5] T. A. Feo y M. G. Resende, "GRASP," *J. Glob. Optim.*, vol. 6, no. 2, pp. 109–133, 1995.
- [6] Ley de Protección de Datos Personales N° 25.326 Congreso de la Nación Argentina, 2000.
- [7] FastAPI Documentation, "Asynchronous Server Gateway Interface." [En línea]. Disponible: <https://fastapi.tiangolo.com/>. [Último acceso: agosto 2026].
- [8] PostGIS Developer Group, "PostGIS Spatial Database," 2026.

Declaración de Uso de IA

Se declara la utilización de modelos de IA para asistencia en la revisión sintáctica y estructuración. Las decisiones de diseño y desarrollo son exclusivas del equipo.

Datos de Contacto

Franco Recalde francorecalde15@gmail.com

Julián Torres jul14n.torres@gmail.com

Franco González franco20047@gmail.com

Tomás Barrionuevo tbarrionuevo2210@gmail.com

Gabriel Gonza gonzagabriel428@gmail.com

Daniel Barrionuevo danielbsiri@gmail.com

PLANILLA PARA CATALOGAR EL PROYECTO FINAL

AÑO	2026	CURSO Y NRO. DE GRUPO	5K4- G01
NOMBRE DEL SISTEMA / PROYECTO			
SeBar Sistema de Seguridad Barrial			
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, Android Studio, Docker Desktop, PgAdmin 4	
REPOSITORIOS Y VERSIONADO		GitHub, Git	
PROGRAMACIÓN		Python (FastAPI), Flutter (Dart), React (JavaScript), NetworkX	
BASE DE DATOS		PostgreSQL, PostGIS, Supabase, Redis 7	
COMUNICACIÓN INTERNA		Discord, WhatsApp, Google Meet	
CAPACITACIÓN		Manuales en PDF por tipo de Usuario	
PRUEBAS DE SISTEMA		Pytest, Postman, Ruff, Mypy	
GESTIÓN DEL PROYECTO		Jira, Metodología Ágil (Scrum), Marco PMI (PMBOK), CircleCI	
DOCUMENTACIÓN		Google Docs, Google Drive, Markdown, Confluence	
MODELOS		Modelo C4 (Contexto, Contenedores, Despliegue), Diagrams.net (Draw.io), BPMN, StarUML	