

Documento técnico - Urbis

Servicios Financieros y Uso Eficiente de Recursos en Ciudades

Inteligentes | HackMTY 2025

César Morán Macías

Ana Paola Jiménez Sedano

Ebony Leticia Valadez Valverde

José Rafael Gutiérrez Suárez

26 de octubre del 2025

1. Resumen ejecutivo

El proyecto busca unir los servicios financieros digitales de Banorte con el uso eficiente de recursos urbanos tales como agua, energía, gasolina y transporte dentro de un ecosistema sostenible y accesible. La app propone un sistema de simulación y acompañamiento financiero-ambiental, el cual permite al usuario categorizar sus gastos mientras es consciente del impacto que tienen en su entorno sus prácticas financieras (emisión de CO_2 en Kg, consumo de energía por kWh y de agua en litros), así como progreso hacia metas planteadas de ahorro, recibiendo incentivos digitales por buenas prácticas. El objetivo es demostrar cómo una banca digital inteligente puede contribuir en el desarrollo de ciudades inteligentes y el bienestar económico de la población.

2. Contexto y Motivación

Las ciudades enfrentan problemas de sobreconsumo energético, escasez de agua y emisiones excesivas derivadas de la falta de optimización en la movilidad urbana. Actualmente, los bancos poseen datos financieros del usuario, pero no los cruzan con indicadores ambientales. Por tal motivo y sacándole provecho a esta área de oportunidad, Urbis propone convertir estos datos en una herramienta de cambio, buscando conectar hábitos de consumo con educación financiera y ambiental, así como el fomento del ahorro y la preservación de recursos.

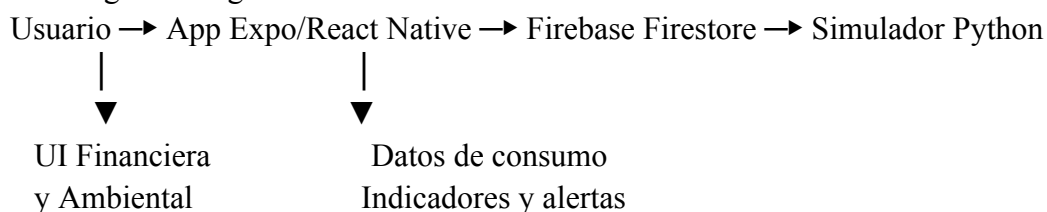
3. Arquitectura del Sistema

3.1 Descripción General

La arquitectura se basa en una integración de frontend móvil (Expo/React Native), registro de usuarios e integración de un módulo de analítica en Python y una base de datos simulada. El flujo de información es el siguiente:

- El usuario inicia sesión (se verifica la cuenta y contraseña con Firebase Auth). La aplicación consulta desde Firestore los datos de consumo asociados a cada usuario, incluyendo promedios acumulados de gasolina, luz y agua, así como indicadores de emisiones de CO_2 por kilómetro, rendimiento del vehículo y gastos reales mensuales en cada categoría
- Un módulo Python calcula indicadores (emisiones y consumo de recursos, así como flujo de ahorro por categorías de gasto).
- La app muestra visualizaciones, recomendaciones e incentivos.

3.2 Diagrama Lógico



4. Tecnologías Empleadas

Componente	Tecnología	Motivo de elección
Frontend	Expo / React Native	Permite desarrollo móvil rápido y multiplataforma
Backend simulado	Firebase Firestore	Integración sencilla y almacenamiento escalable
Analítica / IA	Python (Pandas, Numpy, Matplotlib)	Procesamiento y simulación de datos reales. Diseño de estructura de funciones de la app.
Visualización UI	Recharts / Tailwind / Shadcn UI	Estilo profesional inspirado en Banorte
Autenticación	Firebase Auth	Control de usuarios y datos personalizados
Geolocalización	Google Maps API	Visualización de zonas e impacto ambiental

5. Diseño del Sistema

5.1 Módulos Principales

- *Autenticación*: usuarios registrados con email, contraseña y nombre.
- *Consumo y presupuesto*: cada usuario tiene presupuesto mensual, gastos en agua, luz y gasolina y vehículo en caso de que aplique.
- *Simulación*: Python calcula indicadores como emisiones (tomando en cuenta bases de datos de características de consumo de energía y liberación de contaminantes de los vehículos, así como considerando gasto en luz y agua), ahorro y eficiencia.
- *Incentivos*: el usuario recibe puntos o beneficios al mejorar su consumo y probar un estado de prevalencia en dicho estado.
- *Dashboard*: muestra comparaciones con promedios de zona (con base en los resultados de usuarios cercanos y el consumo propio) y progreso hacia metas.

5.2 Estructura de Datos

Firebase se usa aquí como base de datos simulada en la nube (Firestore), para manejar usuarios y sus datos de consumo (agua, luz, gasolina, metas, etc.). Los datos se guardan como colecciones que contienen documentos, cada documento tiene campos con distintos tipos de datos (números, texto, booleanos).

Recolección: usuarios

Campos:

- id
- displayName
- email
- photoURL
- username
- PromedioAcumuladoGasolina
- PromedioAcumuladoLuz
- PromedioAcumuladoAgua
- auto
- zonaResidencia
- CO2porKm
- RendimientoKmL
- GastoRealGasolina
- GastoRealLuz
- GastoRealAgua

Al iniciar sesión, la app consulta el documento del usuario en los usuarios. Muestra su progreso de ahorro y su comparativa con el promedio de zona. Si el usuario mejora su consumo, se guarda un nuevo documento en incentivos. Las transacciones pueden alimentar el módulo de analítica (Python) para simular eficiencia.

6. Justificación de Diseño y Decisiones Técnicas

Experiencia de usuario: se buscó una interfaz clara, emocional y con feedback inmediato.

- Estructura modular: la división en módulos facilita la futura integración con APIs reales (Banorte, CFE, SIMAPA).
- Datos simulados: permiten una demostración funcional sin depender de infraestructura externa.
- Python como motor analítico: ofrece trazabilidad y posibilidad de incluir modelos predictivos.
- Firebase: simplifica autenticación y sincronización en tiempo real.
- Escalabilidad: el sistema puede crecer hacia una banca verde conectada a servicios públicos reales.

7. Impacto y Escalabilidad

- *Impacto ambiental*: reduce emisiones indirectas al crear conciencia de consumo.
- *Impacto financiero*: fomenta hábitos de ahorro ligados a recompensas.
- *Escalabilidad*: integración con APIs reales de Banorte, CFE, SIAPA o sistemas de transporte.

8. Funciones implementadas

- Nuestra propuesta no solo monitorea el consumo individual, sino que utiliza la información agregada por zonas para identificar patrones, comparar comportamientos y orientar acciones colectivas. Los datos de usuarios se agrupan por zona geográfica para comparar promedios de consumo de agua, luz y gasolina, detectando tendencias o desviaciones significativas entre comunidades.
- El sistema cuenta con un módulo de ajuste dinámico del ahorro, que con base en el promedio global del usuario propone metas personalizadas de reducción de gasto. A través de esta función, la aplicación sugiere cuánto podría ahorrar mensualmente al minimizar los gastos en gasolina, servicio de luz y agua, vinculando directamente el ahorro financiero con la reducción en el consumo de recursos naturales.
- La aplicación incluye un componente social que permite la interacción con otros usuarios mediante una categoría de amigos. Cada usuario puede enviar y aceptar invitaciones, comparando su progreso en términos de porcentaje de ahorro y reducción de consumo de recursos. Esta funcionalidad fomenta una competencia positiva y colaborativa, donde los usuarios se motivan entre sí para mejorar sus hábitos y contribuir colectivamente a una ciudad más inteligente, eficiente y consciente.

9. Áreas de oportunidad

Durante el desarrollo de Urbis, el enfoque principal fue construir un prototipo funcional que demostrara el potencial de integrar servicios financieros con el uso eficiente de recursos urbanos dentro del entorno limitado de un hackathon. Sin embargo, en un escenario de desarrollo más amplio, se identifican diversas áreas de oportunidad para fortalecer, escalar y consolidar la solución.

9.1 Integración con datos reales

Actualmente, la base de datos empleada es sintética, diseñada para simular patrones de consumo energético y financiero. En una versión futura, se podría conectar con fuentes reales de datos, como:

- Bases de datos financieras reales: conectar cuentas bancarias reales (con consentimiento del usuario) para leer gastos por categoría (gasolina, servicios, transporte).
- SENER (Secretaría de Energía): publica anualmente bases de datos sobre eficiencia energética y consumo promedio de combustibles por tipo de vehículo.
- CONUEE (Comisión Nacional para el Uso Eficiente de la Energía): cuenta con catálogos oficiales de rendimiento vehicular (km/litro), emisiones estimadas y tipo de combustible para automóviles comercializados en México.

9.2 Implementación de modelos predictivos avanzados

El prototipo actual usa un simulador analítico basado en funciones determinísticas en Python.

En una versión completa, podrían incorporarse modelos predictivos de Machine Learning para predecir el impacto ambiental a futuro con base en hábitos pasados o proponer metas de ahorro adaptativas y más precisas. Esto ampliaría la inteligencia del sistema y reforzaría la toma de decisiones personalizada.

9.3 Integración con sistemas de transporte y movilidad urbana

Actualmente, la aplicación calcula las emisiones y el gasto en gasolina a partir de los registros personales del usuario, pero no contempla la diversidad de medios de transporte ni los trayectos reales que realiza. En una versión ampliada, el sistema podría integrar módulos de movilidad inteligente que aprovechen APIs y datos abiertos para capturar y analizar patrones de desplazamiento. De esta manera, la aplicación no solo fomentaría la optimización del consumo energético en el hogar, sino también la eficiencia en el movimiento urbano.

9.4 Integración real de sistemas de pago

Durante el hackathon, el sistema de pagos fue simulado manualmente, sin una conexión activa a plataformas financieras externas. Si bien se consideró el uso de Stripe API para la gestión de transacciones, no fue posible implementarlo completamente debido a limitaciones de tiempo, configuración y verificación de cuentas. En una versión posterior, esta integración sería fundamental para lograr una experiencia bancaria real dentro del ecosistema de la app.

10. Conclusiones

Urbis demuestra cómo la integración de servicios financieros digitales con la gestión de recursos de un entorno puede convertirse en una herramienta estratégica para el desarrollo de ciudades más sostenibles e inteligentes. El sistema combina la analítica de datos, la educación financiera y la conciencia ambiental para transformar el comportamiento individual en acciones colectivas con impacto medible.

A través de su simulador en Python, la aplicación logra calcular indicadores de consumo de agua y energía, así como emisiones de CO₂, mientras que su arquitectura basada en Firebase permite visualizar, en tiempo real, el progreso de cada usuario respecto a su meta de ahorro y su comparación con el promedio de su zona. El módulo de ajuste dinámico de ahorro representa un punto clave de innovación, ya que permite al usuario comprender que sus decisiones financieras también tienen un peso positivo en el entorno, demostrando que la eficiencia económica y la sustentabilidad ambiental pueden coexistir en un mismo ecosistema digital.

De esta forma, Urbis no solo busca optimizar los recursos financieros y energéticos del usuario, sino también contribuir activamente al progreso sostenible, inteligente y humano de las ciudades del futuro.

Anexos

https://github.com/RafaGuSua19/SmartCitiesBANORTE_HackMTY2025_Hackaleros.git