Brian Antonio Jiménez Landa

Ingeniero Full-Stack · React / React Native · Node.js · PostgreSQL / MongoDB

Villahermosa, Tabasco, México · remoto o híbrido | +52 993 421 0130 | brian.landa.j@gmail.com
linkedin.com/in/brian-antonio-jimenez-landa | github.com/Blicx

PERFIL PROFESIONAL

Ingeniero en Desarrollo y Gestión de Software con más de 3 años entregando y operando sistemas en producción de inicio a fin. Desde agosto de 2025 construyo y opero buena parte de la plataforma de una empresa de telemetría de gas LP: portal web multi-inquilino (React + Node.js), apps React Native publicadas en App Store y Google Play, una API B2B en PostgreSQL y servicios en tiempo real para repartidores. En paralelo entrego sistemas a clientes y opero productos propios en producción. Inglés C1 como idioma de trabajo diario.

HABILIDADES TÉCNICAS

Lenguajes: JavaScript, TypeScript, Python, SQL
Frontend y móvil: React, React Native (Expo y bare), Tailwind CSS, TanStack Query, Zustand, Redux
Backend y APIs: Node.js (Express, Fastify), diseño de APIs REST, Socket.IO, Server-Sent Events, FastAPI
Bases de datos: PostgreSQL (PostGIS), MongoDB (change streams), Redis, MySQL
Pruebas: Jest, Vitest, node:test, Supertest
Cloud y DevOps: VPS Linux (nginx, pm2, Let's Encrypt), Docker Compose, Git, Firebase, Vercel
IA e integraciones: Tool calling con LLM (APIs compatibles con OpenAI), WhatsApp Cloud API, Twilio, n8n, APIs de Google Maps / Routes
También he usado: Kotlin y Swift (módulos nativos), Java (Spring Boot), PHP (Laravel), Django, MQTT, Drizzle, Mongoose

EXPERIENCIA PROFESIONAL

UrgeGas — Plataforma IoT de Monitoreo de Gas LP

Desarrollador Full-Stack, remoto · Ago 2025 – Actualidad

- **Portal de operación.** Reconstruí desde cero, como desarrollador principal, el portal web multi-inquilino de 2019 (React + Vite, Node.js/Express, MongoDB) sobre la base heredada en uso que comparten otros sistemas; en producción desde feb-2026. Diseñé la capa de acceso: JWT con renovación de sesión, filtrado de consultas por inquilino y 34 permisos nuevos traducidos a los códigos de rol heredados para que los roles existentes siguieran funcionando.
- **Operación en producción.** Despliego y opero staging y producción en un VPS Linux (nginx, pm2, Docker) con punto de retorno documentado en cada versión. Llevé a producción la prueba gratis por organización con índice único y prueba de concurrencia, monté revisiones cada 2 minutos con avisos por Telegram tras rastrear una caída hasta un cambio de permisos de directorio, y activé HTTP/2, que bajó la latencia con 48 peticiones simultáneas de 2.7 s a 161 ms.
- **Asistente de IA (staging).** Diseñé y construí un asistente con LLM para el portal con 15 consultas de solo lectura, cada una validada contra el rol del usuario: el modelo solo elige qué consultar y el servidor calcula cada cifra y arma las tarjetas de respuesta. Respuesta en streaming por SSE y más de 70 pruebas unitarias.
- **App móvil de administradores.** La construí desde cero como único desarrollador, en React Native + Expo con TypeScript estricto, y la publiqué en App Store y Google Play entre julio y agosto de 2026: mapa con clústeres, ficha del tanque, gráficas de nivel y consumo, clientes, avisos, recuperación de contraseña por OTP, baja de cuenta dentro de la app, español/inglés y temas claro y oscuro. Escribí un config plugin de Expo que genera un SceneDelegate en Swift para adoptar el ciclo de vida UIScene de iOS.
- **Cifras que coinciden con su rótulo.** Rehice el motor de consumo para que el total y las barras salgan del mismo bucle y reproduzcan la fórmula de la app de clientes sin error numérico sobre lecturas reales; diseñé las gráficas con interpolación monótona (Fritsch–Carlson) y detección de huecos por percentil, y eliminé seis clases de defectos en los que una cifra contradecía su rótulo.
- **Actualización de la app de clientes.** Llevé una base React Native de 2017 que ya no compilaba a RN 0.78 / React 19 por etapas (0.68 y 0.72), con el cambio de las librerías rotas de cámara, Wi-Fi, mapas y video, y escribí módulos nativos ESPTouch en Kotlin y Swift para dar de alta los sensores por Wi-Fi. Monté React Navigation 7 detrás de un adaptador para que ~145 llamadas de navegación heredadas siguieran funcionando, implementé el rediseño con tema claro/oscuro y llevé Android a API 36. En ambas tiendas desde finales de 2025.
- **App de clientes de nueva generación.** Construí como único desarrollador una app nueva para pedir gas (Expo SDK 56, React Native 0.85, TypeScript estricto): mapa de gaseras, pedido según el tanque, seguimiento del repartidor en vivo por Socket.IO y un asistente de soporte en dos capas (ayuda sin conexión y chat con LLM en streaming cuyas herramientas solo ven al usuario de la sesión, con su endpoint en Express). Primera versión de punta a punta en 6 días; la traspasé a un equipo externo con 7 documentos de arranque.
- **App de repartidores y flotilla en vivo.** Preparé la app de repartidores (Expo) y su backend Node.js/Socket.IO para operar con flota: saqué el GPS de MongoDB a Redis, agregué el adaptador de Socket.IO con Redis, moví la ruta multiparada al servidor (Google Routes API v2) y acabé con los repartidores que desaparecían del mapa con un latido de 30 s y respaldo por HTTP. Construí el módulo de flotilla del portal (6 endpoints, 41 pruebas unitarias).
- **API B2B de nivel de tanques.** Construí una API REST multi-inquilino (Node.js, Express, PostgreSQL 17) que entrega a clientes empresariales la última lectura de cada tanque sin exponer la calibración propietaria: ingesta de MongoDB a PostgreSQL con lista blanca por serial, deduplicación entre fuentes y outbox transaccional de escritura de vuelta. En producción desde jul-2026, con 0 errores 5xx en 8 días de auditoría con tráfico real y 479 pruebas automatizadas contra un PostgreSQL real.
- **Avisos por eventos (prototipo).** Diseñé un servicio de avisos sobre change streams de MongoDB en TypeScript con ingesta resistente a caídas (candado con turno, posición confirmada después de procesar, barrido cuando el oplog ya no la tiene): una ráfaga de 2,500 lecturas en staging dio 0 avisos perdidos y 0 duplicados, y tras un SIGKILL a la mitad cada lectura quedó registrada exactamente una vez.

Desarrollador Full-Stack Independiente — contratos concurrentes

Productos propios y clientes en México y LATAM, en parte vía GALABEX Technologies · Jul 2025 – Actualidad

- Diseñé y construí como único ingeniero una plataforma de almacén, herramienta y activos de renta para una empresa de servicios petroleros (React 18 + TypeScript, Express, PostgreSQL 17, Docker; 63 pantallas, más de 300 endpoints): existencias por ubicación, vales firmados con PIN, materia prima en barra por pulgadas, herramienta verificable por QR y conteos. Cargué ~1,800 productos del Excel maestro del cliente con cero errores; en uso diario desde julio de 2026.
- Con el mismo cliente hice el levantamiento de requerimientos y las revisiones mensuales de avance, escribí la guía de usuario y el manual técnico, y despliego con guiones que respaldan primero y dejan retorno de un comando (23 s de caída en una migración de esquema de septiembre de 2026).
- Construí solo una plataforma multiempresa de tickets por WhatsApp sobre la Cloud API de Meta (producto propio; Node.js 22, Fastify, PostgreSQL, React 19, Socket.IO), en producción desde sep-2026: un motor de reglas en memoria revisa precios y groserías en menos de 5 ms con 500 artículos, y el copiloto de IA lee las conversaciones de forma asíncrona y solo bloquea una respuesta si la empresa lo eligió. Más de 430 pruebas automatizadas.
- Construí solo un sistema multiempresa de cotización y costeo de fletes (Fastify, React 19, PostgreSQL + PostGIS), en producción desde sep-2026: cruza el trazo de Google Routes contra 1,376 puntos de casetas del catálogo nacional para hallar cada caseta de la ruta; en una cotización de demostración, costear casetas y regreso vacío convirtió un margen aparente de 15% en una pérdida de 59%. 149 pruebas del núcleo.
- Otros contratos (2025–2026): módulo de control documental con flujos de aprobación y exportación PDF/Excel para el sistema Laravel de una empresa; acceso con MetaMask y panel de administración para una universidad de Ecuador (Express + React); calculadora de finiquitos y liquidaciones conforme a la LFT (FastAPI + React); bot de citas por WhatsApp para un consultorio dental (n8n + Node.js + Cal.com).

Grupo Microanálisis — laboratorio de salud ocupacional, México

Desarrollador Full-Stack, promovido desde rol de datos · Feb 2024 – Abr 2025

- Promovido de validación de datos a desarrollador; desarrollador principal de un equipo de cuatro en el sistema de reportes de higiene industrial conforme a la STPS (NOM-010 agentes químicos, NOM-011 ruido): frontend React + Material UI sobre una API Django REST, con exportación a Excel por medición. Lo desplegué y lo presenté en sitio.

Insignia IT

Desarrollador Backend (estadía universitaria, plazo definido) · May 2023 – Nov 2023

- Desarrollé APIs RESTful con Java (Spring Boot) y PostgreSQL; colaboré en frontend con React usando patrones DTO/DAO.

EDUCACIÓN

Universidad Tecnológica de Tabasco — **Ingeniería en Desarrollo y Gestión de Software**, 2023–2025 (graduado con honores)
Universidad Tecnológica de Tabasco — TSU en Desarrollo de Software Multiplataforma, 2021–2023

CERTIFICACIONES

PCAP – Certified Associate in Python Programming · NDG Linux Essentials

IDIOMAS

Español (nativo) · **Inglés — C1**, idioma de trabajo diario: llamadas, especificaciones y documentación