

PERFIL PROFESIONAL

Desarrollador de software con foco en arquitectura limpia, modelado de datos, integración de sistemas de terceros y automatización de procesos de negocio. Responsable del ciclo completo de más de quince plataformas web y móviles hoy en uso real — desde el relevamiento del problema hasta el despliegue y el mantenimiento — para clientes de doce rubros distintos. Trabajo tanto sobre sistemas propios como sobre bases de código existentes. Además de programar, formo y lidero equipos de trabajo y sostengo yo mismo la relación con cada cliente — una capacidad que reforcé con un curso de habilidades blandas en comunicación y oratoria. Estudiante avanzado de Ingeniería en Sistemas de Información en la Universidad Nacional del Sur.

EXPERIENCIA

Desarrollador de Software Independiente

Bahía Blanca, Argentina — en simultáneo con mis estudios

- Diseñé y desarrollé más de quince sistemas a medida para clientes de doce rubros distintos (gestión, RRHH, inmobiliario, industrial, comercio electrónico), aplicando arquitectura limpia y principios SOLID para reducir la deuda técnica y facilitar el mantenimiento.
- Formé y lideré equipos de trabajo para mis proyectos con clientes propios: delegué tareas de desarrollo, definí prioridades y revisé cada entrega, sosteniendo yo mismo la relación y la comunicación directa con el cliente de punta a punta.
- Modelé bases de datos relacionales con foco en integridad referencial, indexación y control de acceso por rol a nivel de base de datos (Row-Level Security), sosteniendo múltiples perfiles de usuario sobre una misma plataforma.
- Desarrollé APIs REST e integré servicios de terceros (pasarelas de pago, mensajería, almacenamiento en la nube) bajo esquemas de autenticación y autorización basados en roles.
- Apliqué estrategias de caché e invalidación, y optimicé consultas para sostener el rendimiento de catálogos con cientos de productos y accesos concurrentes.
- Construí un pipeline de automatización (scraping, normalización y deduplicación) que unificó el catálogo de cinco proveedores distintos en un motor de precios propio para un cliente distribuidor.
- Desarrollé paneles de administración a medida en los proyectos que lo requerían, para que el cliente publique productos, propiedades, búsquedas laborales y contenido sin intervención técnica ni dependencia del desarrollador en la operación diaria.
- Conduje el análisis previo al desarrollo con cada cliente: elicitación de requerimientos, glosario de dominio, definición de requerimientos no funcionales (rendimiento, escalabilidad, disponibilidad, seguridad) y modelo de datos, contemplando el cumplimiento de la normativa de protección de datos personales.
- Asumí el mantenimiento evolutivo del sistema de gestión interno de un cliente distribuidor, trabajando sobre una base de código existente en lugar de forzar un reemplazo.
- Optimicé el rendimiento en producción —compresión automática de imágenes en la subida, carga diferida y dimensionado responsivo— para sostener catálogos de cientos de productos sin degradar los tiempos de carga.
- Implementé posicionamiento técnico (SEO) en todos los proyectos —metadatos por página, datos estructurados (schema.org), sitemaps y arquitectura de información orientada a la intención de búsqueda— para mejorar la visibilidad en buscadores tradicionales y en motores de búsqueda con inteligencia artificial.
- Desarrollé formularios con validación robusta (React Hook Form + Zod) y mecanismos anti-spam sin CAPTCHA — campo trampa invisible y límite de envíos por IP — para sostener la calidad de los leads sin fricción para el usuario real.
- Construí sitios bilingües (español/inglés) con enrutamiento internacionalizado y mapas interactivos con agrupamiento de marcadores para catálogos geográficos de más de 50 ubicaciones.
- Diseñé un motor de evaluación con lógica de scoring propia para un test de personalidad interactivo, devolviendo un resultado personalizado por perfil sin depender de una herramienta de terceros.
- Integré pasarelas de pago (Mercado Pago) y automaticé el flujo con webhooks que disparan la entrega de contenido digital por email de forma inmediata, sin intervención manual.

- Escribí pruebas automatizadas (unitarias, de integración y end-to-end) para sostener la confiabilidad de flujos críticos como el checkout y la autenticación.

TRABAJO MÁS DESTACADO

Plenia — Plataforma de Gestión de Seguros

Bahía Blanca, Argentina — en desarrollo

Producto desarrollado en conjunto con un grupo de brokers de seguros, sobre necesidades reales del negocio

NUEVE ASEGURADORAS EN INTEGRACIÓN: Experta Seguros · Allianz Argentina · San Cristóbal Seguros · La Holando Sudamericana · Seguros Rivadavia · Zurich Argentina · Integrity Seguros · MAPFRE Argentina · Mercantil Andina

- Lidero la arquitectura general de una plataforma pensada para más de 2.500 usuarios (backend en FastAPI/PostgreSQL, frontend en Next.js, aplicación móvil en React Native) y coordino a un desarrollador colaborador: reparto tareas, reviso código y me encargo de detectar fallas de seguridad antes de que lleguen a producción.
- Trabajo en conjunto con los equipos de ingeniería y soporte técnico de cada aseguradora: relevamiento de sus APIs, homologación en sus entornos de prueba y resolución de incidencias, sosteniendo yo mismo la relación técnica con cada compañía a medida que se suman las integraciones.
- Diseñé la arquitectura de integración bajo un patrón adapter con capa anticorrupción y un registry por código de aseguradora: sumar una compañía es escribir su adapter y registrarlo, sin tocar el núcleo del sistema ni ninguna integración existente.
- Ninguna compañía cubre todas las operaciones, así que las capacidades de cada adapter —importar cartera, sincronizar novedades, cotizar, emitir, cobranzas, renovaciones, documentación, padrón— se resuelven por reflexión sobre la clase: el panel de administración habilita cada función sola, sin listas de compañías escritas a mano en el frontend.
- Integro REST/JSON, SOAP/XML y suites OpenAPI —una de ellas de 17 especificaciones—, con un esquema de autenticación distinto por compañía (OIDC, OAuth2, JWT, API key con subscription key) y credenciales cifradas por aseguradora y por ambiente. Cada adapter absorbe además los límites propios de su API, como ventanas de consulta acotadas que itera solo.
- Desarrollé el cotizador contra las APIs de las compañías, resolviendo el ciclo completo cotizar → emitir → consultar. Cada cotización se persiste con su número de presupuesto externo, sus planes y su vigencia, y avanza por estados (cotizada, emitida, descartada, expirada); una cotización vencida se bloquea sola en lugar de emitirse a un precio que ya no existe.
- Separé cotizar —que no compromete nada— de emitir, que crea una póliza real y compromete un cobro, y concentré los guardarrailes en la emisión: confirmación explícita, permisos de asesor dueño o administrador, y ambiente de pruebas por defecto, de modo que validar en TEST antes de tocar producción es el comportamiento por default y no un opcional. El payload de emisión toma el presupuesto y el plan de la cotización guardada y no de lo que envía el cliente, para que nadie pueda emitir sobre el presupuesto de otro.
- Definí qué sistema es fuente de verdad de cada dato —la relación comercial y la atribución las manda la plataforma, los datos y el estado de vida de la póliza los manda la aseguradora— y derivé de esa regla toda la sincronización. Sobre eso modelé el dominio (organizador, productor, asesor, cliente, póliza) resolviendo el caso real en que la compañía no distingue qué asesor vendió: la póliza guarda el asesor real, el administrador que emitió en su nombre y el código de productor usado, y ninguna novedad entrante puede pisar esa atribución.
- Integré el circuito de cobranzas de cada compañía —cuotas, forma de cobro, montos cobrados y adeudados, fechas de pago y de vencimiento—, que alimenta el estado de cuenta del cliente y la liquidación de la comisión real de cada asesor, junto con los avisos de las pólizas próximas a renovar.
- Diseñé el pipeline de importación de cartera: deduplicación de clientes por múltiples campos, enriquecimiento automático de datos faltantes consultando otros endpoints, y una bandeja de revisión para la primera carga de cada aseguradora.

PROYECTOS DESTACADOS

Sistema MBK

Plataforma de gestión financiera con dos roles conectados, permisos por rol y reglas de seguridad a nivel de base de datos.

Julián Tavano — JT Arquitectura

Estudio de arquitectura bilingüe (español/inglés) con catálogo de modelos prefabricados y mapa interactivo de más de 50 obras construidas.

BBJobs

Portal de empleo regional con arquitectura de tres roles (candidato, empresa, administrador) y matching por IA en desarrollo.

Neuhaus S.A.

Industria gráfica del sector farmacéutico y cosmético (ISO 9001, BPM y FSC), con formularios de cotización específicos por tipo de producto.

JLI Recruiters

Plataforma de RRHH con panel administrativo, permisos multiusuario y venta de contenido automatizada vía pasarela de pago.

Syrius Ingeniería SRL

Sitio institucional para el representante oficial de Emerson en Bahía Blanca, con catálogo técnico reorganizado por familia de producto.

NG Consultoría

Motor de evaluación psicométrica propio, con lógica de scoring desarrollada a medida.

Alport JF

Vidriera digital de siete rubros con carrito, ofertas con vencimiento automático y actualización de precios en lote desde el panel.

Portfolio completo, con capturas y el detalle de cada proyecto: gaelgonzalez.com.ar

HABILIDADES TÉCNICAS

Arquitectura y diseño	Clean Architecture · Principios SOLID · Patrones de diseño (Factory, Strategy, State Machine, Observer, Adapter) · Anti-Corruption Layer · Diseño de APIs REST · Modelado de dominio
Modelado y diagramación	Diagramas de entidad-relación (ER) · Diagramas de clases · Diagramas de casos de uso · Diagramas de estados · Diagramas de secuencia
Fundamentos de programación	Programación orientada a objetos · Programación funcional y lógica · Estructuras de datos y algoritmos · Análisis de complejidad · Programación concurrente y asíncrona
Bases de datos	PostgreSQL · MySQL · Supabase · Modelado relacional y no relacional · Indexación y optimización de consultas · Transacciones · Row-Level Security
Sistemas y rendimiento	Sistemas operativos (procesos, hilos, sincronización) · Arquitecturas distribuidas · Estrategias de caché · Colas y procesamiento en segundo plano
Backend e integraciones	Autenticación y autorización por roles (RBAC) · Integración de APIs de terceros (pasarelas de pago, mensajería, almacenamiento en la nube) · Webhooks · Automatización de procesos de negocio
Lenguajes y tecnologías	TypeScript / JavaScript · Python · Java · SQL · Prolog · React · Next.js · Node.js · FastAPI · React Native · Tailwind CSS
Ingeniería de software	Control de versiones (Git) · Testing · Elicitación de requerimientos · Verificación y validación · SEO técnico y datos estructurados (schema.org)
Desarrollo asistido por IA	Claude Code (principal) · Gemini · Antigravity — con criterio para elegir la herramienta según la tarea y revisión propia de todo lo que llega a producción

FORMACIÓN ACADÉMICA

Ingeniería en Sistemas de Información

Cursando — 4.º año, 2.º cuatrimestre

Universidad Nacional del Sur (UNS) — Bahía Blanca, Argentina

Materias aprobadas hasta la fecha: Algoritmos y Complejidad · Arquitectura y Diseño de Sistemas · Bases de Datos · Métodos Formales para Ingeniería de Software · Requerimientos de Sistemas · Lógica para Ciencias de la Computación · Arquitectura de Computadoras · Química IS · Modelos de Software · Modelos Estadísticos para Ciencias de la Computación · Tecnología de Programación · Organización de Computadoras · Teoría de la Computabilidad · Estructuras de Datos · Análisis Matemático II · Física I · Introducción a la Ingeniería de Software · Lenguajes Formales y Autómatas · Introducción a la Programación Orientada a Objetos · Análisis Matemático I · Elementos de Álgebra y de Geometría · Resolución de Problemas y Algoritmos.

IDIOMAS

Español

Nativo

Inglés

Nivel intermedio/técnico — examen de suficiencia UNS aprobado