

Antonio Vila

Senior Full-Stack Engineer · Python · PHP/Symfony · Go

Python · Django · FastAPI · PHP 8.5 · Symfony 8.1 · PostgreSQL · RabbitMQ · Go ·
TypeScript · LangChain · Docker

vila.antoniojose@gmail.com

linkedin.com/in/antonio-vila

antoniovila.dev

github.com/antoniojosev

Venezuela · Remote (UTC-4)

PROFESSIONAL PROFILE

Full-stack engineer with 8 years shipping production systems, Python as my main stack and the same comfort in PHP/Symfony, Go and TypeScript. The last two years as technical owner of the claims core at Crabi, a Mexican auto insurtech: API layers for a mobile app, role-based authorization, invoicing, and LLM integrations that run in production. On my own time I design, build and operate ten products and open-source developer tools, each documented as a case study at antoniovila.dev.

TECHNICAL SKILLS

Python: Django, Django REST Framework, FastAPI, Celery, pytest, LangChain

AI engineering: LangChain, OpenRouter, LLM classification with deterministic guardrails, document extraction, vector search, MCP servers

Frontend & mobile: TypeScript, React, Next.js, Vue 3, Nuxt, Flutter, Tailwind

PHP & Go: PHP 8.5, Symfony 8.1, Doctrine ORM, PHPStan, Pest, PHPUnit; Go with gRPC and protobuf

Data & messaging: PostgreSQL, MySQL, SQLite, Redis, RabbitMQ, MongoDB

Infra & observability: Docker, Linux/systemd, GitHub Actions, Kubernetes (operation), AWS (S3, CloudFront, ECR, CloudWatch), KrakenD, Grafana, Loki, Prometheus, Jaeger, Sentry

PROFESSIONAL EXPERIENCE

Backend Developer · Crabi

Jul 2024 – Present | Mexico · Remote · Auto insurtech

Technical owner of Defender, the claims core service (PHP 8.5, Symfony 8.1, Doctrine, PostgreSQL 14, RabbitMQ). 762 pull requests merged across 10 services in two years, on a system that handles regulated personal data.

- ▶ Built the claims API layer for the policyholder mobile app across four product versions: service history, photos by repair stage, repair parts, roadside assistance with per-coverage consumption limits, and a new category of user-facing notices.
- ▶ Redesigned the authorization model after a QA audit surfaced 108 cases across 15 role families with generic errors, wrong codes or unauthorized actions. Remapped every exception to its HTTP status and wrote a 90-test runner that exercises each endpoint with a scoped-role user. Measured against production afterwards: 201 of 205 error codes and 51 of 66 policies correct.
- ▶ Point-in-time policy snapshot, materialized at claim filing with as-of semantics at the accident date, replacing live calls to the policy service that returned current state. Coverage calculations became reproducible and auditable.
- ▶ AI engineering on Mirai, the platform's LLM service (Python, FastAPI, LangChain, OpenRouter; 40 PRs): a claim classifier that routes each new claim to on-site or remote handling with a hybrid design, deterministic rules gating the model, geographic overrides and anti-fraud thresholds by policy age. 5,250 classifications since May 2026 at a 2-second median latency.
- ▶ Valuation scan: LLM extraction of damaged parts from uploaded files, matched against the parts catalog with vector search and confidence thresholds, integrated across three repositories and three languages.
- ▶ Delivered the internal role-manager backend across five services in PHP and Go: role, provider and partner catalogs, internal user creation with password email, and an enriched insured-party detail, including the JWT issuer, the API gateway and the public API docs.
- ▶ Refactored the payment-details model on a live table to a polymorphic pair, with migration, production backfill, controlled shutdown of automation triggers during the load, and a hotfix for the bulk invoice endpoint that broke on the change.
- ▶ Owned claims invoicing end to end: bulk issuance, coverage comparison, deductible payment types, zero-amount service closure, cancelled-invoice handling and reinsurance accounting constants.
- ▶ Identified and documented five security findings on my own initiative: an authorization fail-open on document download, a production route with its role check pending, plaintext-password links, PII leaking into monitoring, and a direct path to the production database that bypassed read-only and audit controls. Migrated notifications and surveys to signed links and API-key authentication.

- ▶ Built the tooling the team uses to sustain dozens of parallel branches: ephemeral environments that test a fix against real QA data before deploying, git hooks that log each commit to its task, and a FastAPI and React dashboard that syncs task state with the GitHub API and runs production data corrections behind a mandatory dry-run.
- ▶ Maintain a 141-note technical vault covering 17 platform services, used as an architecture reference outside my own team.

Stack: PHP 8.5, Symfony 8.1, Doctrine, PostgreSQL 14, RabbitMQ, Python, FastAPI, LangChain, Go, gRPC, KrakenD, Docker, GitHub Actions

Senior Backend Developer · [GESTOY \(SportYeah\)](#)

Aug 2022 – Dec 2023 · 1 yr 5 mos | Madrid, Spain · Remote · Sports social network

Python backend of a social network with live video, real-time chat and on-chain payments.

- ▶ Led the chat backend: direct and group messaging, 1:1 and group video calls, and real-time delivery over WebSockets.
- ▶ Led the content recommendation system for the main feed and live streams: ML similarity models over user behaviour, served from Django with asynchronous scoring in Celery.
- ▶ Built the live video streaming module and the stories module for ephemeral content; collaborated on the main publishing module.
- ▶ Led the monetization and payments microservice end to end, with USDT settlement and a custom token on-chain.
- ▶ Asynchronous processing with Celery and RabbitMQ, distributed caching with Redis for feeds and sessions under high concurrency; extensive unit tests and integration testing with the QA team.

Stack: Python, Django, Django REST Framework, Celery, RabbitMQ, Redis, PostgreSQL, MongoDB, WebSockets, GCP, AWS

Senior Backend Developer · [Fans And Me Social](#)

May 2022 – Aug 2022 · 4 mos | Spain · Remote · Crypto and NFT social commerce

- ▶ Built the backend of an Instagram-style network from scratch in Django: main feed, user profiles, statistics, and real-time chat with group, voice and video calls.
- ▶ Designed the relational model and query optimization for a high-traffic platform; automated media storage and processing, backups and deployment.
- ▶ NFT credits bank, trading of digital assets through platform credits, and the monetization and payments module with an on-chain integration and a custom token.

Stack: Python, Django REST Framework, Celery, Redis, PostgreSQL, WebSockets, Web3

Senior Backend Developer · [BlackHost](#)

Jun 2022 – Aug 2022 · 3 mos | Colombia · Remote · Software agency

- ▶ Django modules and microservices across client projects, from a medical platform to a remote-work tool; full monolithic builds for sports sites and e-commerce.
- ▶ Bulk data-loading pipelines from CSV, PDF and XLS sources; database adaptation and optimization; deployments.

Stack: Python, Django, Django REST Framework, PostgreSQL

Systems Lead & Full-Stack Developer · [Distribuidora Hipreca](#)

Mar 2019 – Nov 2021 · 2 yrs 9 mos | Venezuela · Auto-parts distributor

- ▶ Ran the systems department and its data: inventory, point-of-sale, web presence and social channels; architecture decisions, code review and mentoring of the internal team.
- ▶ Built the automation that tracks each product from import to sale, and a recommendation system in Python that suggests purchase investment, offers and promotions to customers.
- ▶ Automated timesheets and fee payments; full-stack development with Python/Django and Java (Hibernate, Maven) over PostgreSQL and MySQL.

Stack: Python, Django, Java, Hibernate, JavaScript, PostgreSQL, MySQL

Full-Stack Web Developer · [Repuestos Oriente](#)

Apr 2018 – Feb 2019 · 11 mos | Venezuela

- ▶ E-commerce site for auto parts, end to end: database design, Django backend, frontend and deployment. Product catalog, search and inventory management.

Stack: Python, Django, JavaScript, MySQL

OPEN-SOURCE DEVELOPER TOOLS

Self-hosted developer tools with tagged releases, ADRs and public case studies. Four in Go with a hexagonal architecture,

plus a Python CLI framework.

- ▶ **trapline** · Error tracking, tracing, uptime, crons and status page in a single 18 MB binary, compatible with the official Sentry SDKs. 19.7 MB of RAM at rest against the 16 GB minimum Sentry self-hosted publishes; 611 events/s in bench; 35 architecture decision records. MCP server so an agent can read, patch and close an error. v0.1.0, Sep 2026 · antoniovila.dev/#case/trapline
- ▶ **Traccia** · Web analytics on one Go binary and one Postgres. Plugins are JavaScript that runs inside the binary with a 100 ms budget per event, so extensions need no fork and no Node. v0.1.1 · antoniovila.dev/#case/traccia
- ▶ **kip** · CLI to share encrypted .env files through self-destructing links. AES-256-GCM in the client; the key travels in the URL fragment and never reaches the server. v0.1.0 · antoniovila.dev/#case/kip
- ▶ **pm** · Local project and port manager for WSL2: systemd units, port discovery, project.localhost via Caddy, and 13 MCP tools so agents can operate dev servers. 89% test coverage. v0.1.0 · antoniovila.dev/#case/pm
- ▶ **EagleKit** · Python CLI framework with a plugin system built on entry points: project registry, per-branch tasks and Git automation. Public plugins number commits per branch and mirror QA to Prod with cherry-picks. v0.2.1 · antoniovila.dev/#case/eaglekit
- ▶ **PyDollar-VZLA** · Python package that scrapes the Venezuelan dollar rate from several sources with a single interface. Precursor of Venekambio.

Stack: Go, SQLite, PostgreSQL, HTMX, Caddy, systemd, MCP, Python, Typer

PRODUCTS AND CLIENT WORK

milibrodeobra · Digital site logbook for civil engineers · client · in production

Jun 2026 – Present · antoniovila.dev/#case/milibrodeobra

- ▶ The daily report is entered once from the phone; the same state produces the WhatsApp message, a shareable page with photos, the task board and the logbook. In real use on job sites: 113 daily reports, 6 sites and 87 photos since June 2026; 14 tables and 17 production migrations with no data migration needed.

Stack: Next.js 16, React 19, Supabase, PostgreSQL, Zod, Tailwind, Vercel

PAX Elevators · Corporate site for an elevator manufacturer · client · live

2026 · paxelevators.com · antoniovila.dev/#case/paxelevators

- ▶ Static Nuxt 3 site with catalog, thermography diagnostics, an express quoter that ends on WhatsApp, one serverless function for the form, and four printed pieces from the same HTML. DNS migrated without downtime for mail. Closed its first client in its first month online.

Stack: Nuxt 3 SSG, TypeScript, Tailwind, Nitro, Resend, Vercel

Hotel booking tokenization · Web3 client engagement · live on mainnet

2026 · Fixed-fee contract

- ▶ P2P tokenized layer on top of hotel reservations: ERC-1155 contracts with lazy minting, Binance Pay integration and a Web2 bridge contract for the hotel's booking system.

Stack: Solidity, ERC-1155, Node.js, Binance Pay API

BioLinkStore · Multi-tenant SaaS · in production

2024 – Present · biolinkstore.com · antoniovila.dev/#case/biolinkstore

- ▶ Turns an Instagram bio link into a store for people who close sales on WhatsApp. Multi-tenant on a single schema, 0% commission, \$0/month infrastructure.

Stack: Next.js, NestJS, TypeScript, Prisma, PostgreSQL (Supabase), Vercel, Koyeb

Venekambio · Multi-rate exchange calculator · web, Android and Windows

2024 – Present · venekambio.com · antoniovila.dev/#case/venekambio

- ▶ One figure, six currencies at once, no backend: the client reads four public sources in parallel and runs at \$0/month.

Stack: Flutter, Riverpod, Nuxt 4 SSG, TypeScript, Vercel

LANGUAGES

Spanish (native)