

Viktors Afanasjevs

BACKEND & AI ENGINEER

Riga, Latvia · +371 28822665 · afanasjewww@gmail.com · [linkedin/afanasjewww](https://linkedin.com/in/afanasjewww) · [github/afanasjewww](https://github.com/afanasjewww) · afanasjewww.dev

PROFILE

Backend and AI engineer with 5+ years of experience building production systems across fintech, blockchain and B2B SaaS. I co-founded Borebase and built the product from architecture and domain logic to billing, deployment and customer onboarding. My core stack is async Python, PostgreSQL, TypeScript, C# and Redis, with hands-on experience in encrypted APIs, multi-chain infrastructure, real-time analytics and multi-agent RAG. I'm strongest when a complex system needs clear ownership from first principles to production.

EXPERIENCE

Co-Founder & CTO, Borebase · borebase.io

Mar 2025 - Present · Alongside full-time

- A live B2B SaaS that geotechnical firms pay for: EU-hosted, GDPR-compliant, and aligned with ISO 14688-1 and ISO 22476-1/-2. Built it end to end and still own the architecture.
- Designed it as a modular monolith with per-module hexagonal (ports & adapters) boundaries so each domain stays independently testable, then layered on multi-tenant isolation, RBAC, JWT/argon2id auth, and Stripe seat-based billing.
- Wrote the geotechnical compute engine from scratch: CPT/CPTu/SCPT, SPT, and dynamic-penetration analysis (Robertson soil-behavior index I_c , N60, shear strength), LAS geophysics import, and 2D cross-section generation.
- Built a configurable report engine that outputs multi-column PDF and CAD-editable DXF for AutoCAD/QGIS, and pushed heavy exports and LAS parsing onto async Redis/arq workers to keep the app responsive.

Python · FastAPI · PostgreSQL · Redis · SQLAlchemy · React · TypeScript · Stripe · Docker

Full-Stack Developer, DK Techno · brp-russia.ru

Feb 2026 - Present · Freelance

- Sole developer of one Next.js 16 codebase serving four surfaces from a single Docker image: a bilingual (ru/en) storefront, an admin CRM, an internal ERP, and a warehouse Telegram Mini App, with the public/internal split enforced in middleware.
- Built the admin core: JWT access/refresh with logout and tokenVersion invalidation, Zod-strict validation, CSRF/Origin checks, an audit log of every mutation, and a two-phase media pipeline (Sharp/WebP) with path-traversal guards and atomic writes.
- Built the ERP over 28 Prisma models: an invoice and bill-of-lading parser, shipments with generated PDF specifications and packing lists, inventory and a stock ledger, payments and EU customs MRN status tracking.
- Built the parts catalog with cross-brand duplicate detection, and a one-command importer that loads the full 250k-row supplier xlsx into Postgres.
- Built the notification layer: a Telegram bot routing per-entity alerts into forum topics, and an SMTP queue driving transactional mail and subscriber campaigns.
- Built in-house analytics tracker: unique visitors, views per day, top pages and most-clicked vehicles.
- Own deployment end to end: multi-stage Docker, hardened VPS (Nginx, UFW, fail2ban), GitHub Actions with migrations on container start, and ~130 Vitest suites.

Next.js 16 · React 19 · TypeScript · Prisma 7 · PostgreSQL · Tailwind v4 · Docker · Vitest

Full-Stack Developer / B2B Lead, PinToPay · pintopay.me

Sep 2025 - Apr 2026 · Full-Time

- Built end to end API encryption with replay-attack protection, fronted by a dual-layer encrypted API gateway.
- Built an admin dashboard with 3-layer client-side cryptography, RBAC across 5+ roles, and an API documentation portal with dynamic OpenAPI rendering.
- Integrated multiple blockchains with real-time indexing, hot-wallet deposit aggregation, and automatic TRX energy rental.
- Stood up a real-time ClickHouse analytics platform (DAU/WAU/MAU, funnels, retention, sessions) with batch event buffering, 10K req/min rate limiting, and multi-tenant isolation.
- Shipped a Telegram newsletter system reaching 100K+ users with error classification, plus MongoDB-aggregation reporting across 6+ metrics with geographic breakdown.
- Promoted to B2B Lead, managing 8 enterprise clients end to end - onboarding, technical demos, and integration support.

Python · Sanic · MongoDB · PostgreSQL · Redis · ClickHouse · Nuxt · Docker

Frontend Engineer, UNHOSTED · [PerpRoyale](https://perproyale.com)

Jun 2025 - Sep 2025 · Part-Time

- Built an offline-ready React PWA for PerpRoyale, a 1v1 / battle-royale trading game on Hyperliquid, with a reusable UI kit and iOS standalone optimizations.
- Built live trading views over WebSocket price feeds (Hyperliquid) and a thin API client with request tracing (X-Request-Id) and idempotent retries.
- Configured and hardened production VPS infrastructure: Nginx reverse proxy, SSL/TLS, and server hardening.

React (Vite/TS) · PWA · WebSockets · FastAPI · MongoDB · Docker

Software / AI Engineer, UNHOSTED · unhosted.ai

Feb 2024 - Jun 2025 · Full-Time

- Built a multi-agent system with orchestration, multi-step reasoning, and tool use that executes tasks autonomously.
- Gave the agents memory: a dual-layer setup with a Redis session cache plus pgvector RAG retrieval, wired to multiple LLMs through OpenRouter.
- Designed a standardized tool-integration layer over trusted data sources (Moralis, TAAPI), and authored an ERC-4337 trading-agent blueprint that separates "AI decides" from "wallet executes".

Python · FastAPI · pgvector · Redis · OpenRouter · Moralis · TAAPI · Docker

Software Engineer, ASML · asml.com

Sep 2023 - Feb 2024 · Part-Time

- Replaced a fragile CLI recipe workflow with a reliable PyQt5 GUI (FC-060), cutting delivery time per cycle by 65% and adding safeguards like duplicate checks and changeset preview before execution.
- Turned requirements into MoSCoW user stories and contributed to testing and configuration across 2-week sprints, which cut manual errors in recipe delivery.

Python · PyQt5 · ClearCase · Robot Framework

Software Engineer (C# / Digital Twins), Witteveen+Bos · witteveenbos.com

Sep 2022 - Jun 2023 · Full-Time

- Led a SysML-to-Unity translator for Digital Twins that turned formal architecture models into interactive 3D simulations.
- Built a scalable C# importer for SysML structure and behavior with project-ready presets, and supported real-time updates and stakeholder demos for large-scale infrastructure.

C# · .NET · Unity · SysML / UML

Programming Teacher, Hello IT · helloit.lv

Jan 2021 - Dec 2025 · Weekends

- Taught 500+ students (ages 7-18) C#, Java, and web fundamentals, designed the course programs myself, and ran workshops and coding camps.

Software Engineer (Industry Projects), Fontys

Sep 2019 - Jul 2024 · Alongside studies

- Delivered 10+ multidisciplinary projects with industry partners (AiROS, Winclove, Sioux, Intelly, Games For Health, Enversed) using full-cycle Agile.

C# · .NET 7/8 · WinForms · Unity

INDEPENDENT PROJECTS

5v5 Matchmaking Bot - a Discord bot running community 5v5 Call of Duty matches, written in C# on .NET. Any player can open a lobby; the bot holds the queue until 10 join, then splits the sides with a custom Elo model, deliberately varying the pairings so the same ten never get identical teams, and moves everyone into team voice channels. With no game API to read results from, players call the winner themselves: the team that collects more reactions takes the match, and the bot recomputes every player's rating and persists it in MongoDB.

C# · .NET · DSharpPlus · MongoDB

payp · github.com/artjoms-dev/payp-cli - an open-source AI CLI for data engineers & database architects. Turns natural-language requests into SQL, so you can query and explore databases without hand-writing queries.

Python

Spread Finder - a CEX/DEX crypto arbitrage engine. Monitors 12+ centralized exchanges (REST + live WebSocket order books) and DEX aggregators (Jupiter, 0x, 1inch, Raydium) across chains, detects cross-venue spreads, and validates each one for real executability - order-book depth, withdrawal/deposit network routes, cross-chain bridge feasibility (deBridge), and token safety (GoPlus) - before firing tiered Telegram alerts. Built on a pluggable per-exchange adapter registry with local Solana AMM quote validators (Raydium/Orca/Meteora), Redis dedup, and a paper-execution / PnL layer.

Python · FastAPI · PostgreSQL · Redis · SQLAlchemy · ccxt · WebSockets · Docker

netwatch - a personal prototype for layered outage attribution (Go). Probes router / ISP / DNS / VPN in parallel and uses consensus and hysteresis to infer the most likely failing layer, packaged as a single static binary with a live terminal dashboard.

Go

Telegram Commerce Bots - a USDT (TRON TRC-20) storefront on Telegram. A dual-bot system (admin + storefront) with full catalog/order CRUD, statistics, broadcasts, discounts, giveaways, and multi-wallet management. Detects bot bans and fails over to a backup token instantly, with on-chain USDT payment verification.

Python · Aiogram 3 · SQLAlchemy · PostgreSQL · Docker

SKILLS

Languages	Python, Go, TypeScript / JavaScript, C#, C / C++, Java, SQL
Backend	FastAPI, Sanic, .NET 7/8, Microservices, PostgreSQL / pgvector, MongoDB, Redis, ClickHouse, SQLAlchemy, REST, WebSockets, OpenAPI / Swagger, OAuth2 / JWT, Aiogram, APScheduler, Stripe
Blockchain	Multi-chain integration (TRON, TON, Solana, EVM), TronPy, Moralis, on-chain indexing, ERC-4337, TRC-20 / USDT, hot-wallet management
AI / Agents	Multi-agent orchestration (AGNO), RAG, dual-layer memory, semantic search, LLM orchestration (OpenRouter), tool-use integrations (Moralis, TAAPI)
Frontend	React, Next.js, Vue / Nuxt, Astro, Tailwind / MUI, Mapbox GL, desktop UI (WinForms, PyQt5)
DevOps	Docker, GitLab & GitHub CI/CD, Nginx, MinIO (S3), Google Cloud Run, VPS provisioning, SSL/TLS
Security	End to end / API encryption, replay-attack protection, OWASP-minded validation, system hardening

EDUCATION

BSc, ICT - Fontys Hogeschool

Eindhoven, Netherlands · 2019 - 2024

LANGUAGES

Russian	Native
English	Fluent
Latvian	Fluent