

Stanislav Shtelmakh

Full-Stack Developer · Next.js / TypeScript / Python

Stuttgart Region, Germany · CET / UTC+1 · Remote DE / EU & International

stanislav.shtelmakh@gmail.com · +49 176 740 45295

linkedin.com/in/shtelmakh · spectral.im (portfolio — full project documentation & architecture)

Work authorization: EU-wide right to work under §24 AufenthG (Ukrainian status) — no employer sponsorship required, immediate start possible.

PROFILE

Independent full-stack developer shipping production-grade SaaS and automation products end-to-end — from marketing landing and Stripe-based payment loops to license-gating backends, embeddable runtimes, and self-hosted Linux infrastructure. Comfortable across the stack: TypeScript (Next.js 16, React 19), Python (async, NLP, FSM), Node.js (Fastify), Docker, SQLite/Supabase. Three production deliverables live and verifiable. Engineering foundation from a Materials Science B.Sc. applied to software since 2025. C1 English (working language), A2 German.

EXPERIENCE

Independent Full-Stack Developer · Remote *Nov 2025 – Present*

Designed, built, and shipped three production-grade web products end-to-end: marketing sites, Stripe payment loops, backend services, embeddable runtimes, and ops. Full responsibility from product decisions to Linux VPS deployment, transactional email infrastructure, and post-launch monitoring. Three live products detailed in Selected Projects below.

Service Assistant · INTRAC Latvia, Saldus (LV) *May 2022 – Jul 2024*

- Structured fault diagnosis and ticket workflow — intake, analysis, prioritization, escalation, documentation — same methodology used in IT-support / Jira-style workflows.
- Technical communication between non-technical customers and engineering teams under time pressure.
- Independent documentation of troubleshooting procedures and resolution paths.

SELECTED PROJECTS

PingBoost · self-hosted SaaS micro-product 2026

pingboost.site · live · end-to-end SaaS shipped from one repository

A 5 KB (gzipped) embeddable Whatsapp / Viber / Telegram chat widget sold for \$39 lifetime, with the full purchase loop self-hosted on a single Hetzner VPS. Four services in one repo: a TypeScript widget mounted in Shadow DOM for full CSS isolation from the host site; a Fastify license server on SQLite (WAL mode, foreign keys, busy_timeout) with idempotent Stripe webhooks (UNIQUE constraint on session ID, handles checkout.session.completed, async_payment_succeeded, charge.refunded), and domain-bound auto-binding license keys; a Next.js 16 static-export landing with a /thanks page that polls the API and displays the key 1–3 seconds after payment; and a self-serve visual editor where buyers configure the snippet without contacting support. Resend handles transactional email (DKIM / SPF / DMARC verified).

Architecture supports 1,000+ active licenses on a single small VPS because validation runs from RAM, not disk. Verified end-to-end with live Stripe transactions.

Stack: TypeScript · Node.js · Fastify · Next.js 16 · React · SQLite (WAL, better-sqlite3) · Stripe API · Resend · esbuild · Zod · nginx · systemd · Let's Encrypt · Hetzner Cloud · Shadow DOM

PriceShift · browser-based pricing automation for SMB e-commerce 2026

priceshift-eight.vercel.app · live · production tool, real client

Production pricing tool built for a B2B e-commerce business (educational furniture, Ukraine) with a 500–5,000-product catalog. Parses messy XLSX / CSV exports entirely client-side; auto-detects price, category, SKU, and ID columns across EN / RU / UA naming conventions using a fuzzy-match scorer and multi-language alias dictionary; normalizes dirty data (currency symbols, thousand separators, decimal commas). Hierarchical markup engine — subcategory overrides category overrides global — with a real-time preview (color-coded change indicators) before any export. Original file structure preserved so re-import to the e-commerce platform is friction-free. No server upload, no analytics, no third-party scripts — pricing data never leaves the browser. Result: a typical bulk repricing session moved from 10–20

