

Anton Zakharov

Senior Fullstack Product Engineer / Frontend-Strong Fullstack

Zaragoza, Spain | +34 652 287 286 | zakharov.toxa@gmail.com | toxuh.pro | linkedin.com/in/toxuh

SUMMARY

Senior Fullstack Product Engineer with 15+ years in product development, strongest in React/Next.js and hands-on backend with Node.js, Next.js APIs, Python/FastAPI, PostgreSQL/Prisma, Redis, workers, Docker, and AI integrations. Built production fintech, SaaS, AI, and creator tools solo and in startups: custom auth, billing, LLM pipelines, background jobs, analytics dashboards, financial flows, and data-heavy product interfaces. Open to remote startup roles across EU/UK; comfortable owning ambiguous product areas end-to-end.

CORE FULLSTACK STACK

Frontend: React, Next.js, TypeScript, App Router, SSR/ISR, Tailwind CSS, shadcn/ui, TanStack Query, Recharts/D3, WebSockets/SSE

Backend: Node.js, Next.js Route Handlers/API, PostgreSQL, Prisma, FastAPI, Python, Redis, ARQ/Celery, REST APIs, Zod/Pydantic, Django basics, NestJS basics

AI/Data/Infra: OpenAI API, Anthropic Claude, Gemini, Perplexity, RAG, Qdrant, structured outputs, LLM judging, Docker, GitLab CI/CD, DigitalOcean, Stripe

SELECTED FULLSTACK / AI PRODUCT PROJECTS

VOZ (voz.ad) | Live waitlist product | Solo Founder / Fullstack Product Engineer

GEO/AEO analytics SaaS for B2B SaaS companies targeting Spanish-speaking markets. Monitors brand visibility across ChatGPT, Claude, Perplexity, and Gemini; extracts mentions, sentiment, competitors, hallucination flags, citations, visibility rate, share of voice, and trend deltas.

- Built a production GEO/AEO analytics SaaS end-to-end: Next.js 16 + FastAPI + PostgreSQL, containerized with Docker Compose, deployed to a self-hosted DigitalOcean droplet via GitLab CI/CD from zero to production.
- Designed an LLM pipeline integrating OpenAI, Anthropic Claude, Perplexity, and Gemini via a common adapter pattern; used Claude as a judge model with tool-based structured output to extract brand mentions, competitor positions, sentiment, hallucination flags, citation URLs, and per-call cost tracking.
- Implemented backend-owned auth from scratch: JWT access tokens, rotating hashed refresh tokens in PostgreSQL, httpOnly/Secure/SameSite cookies, Argon2 password hashing, email verification, and password reset tokens.
- Built ARQ + Redis background job infrastructure with separate api / worker / scheduler services for LLM execution, judge analysis, daily metrics aggregation, weekly digest, alert checks, and prompt suggestions.
- Integrated Stripe Checkout, Customer Portal, subscription webhooks, and plan entitlement checks gating dashboard access and LLM run scheduling.
- Wrote production-grade handwritten SQL with asyncpg across 27 Alembic migrations, including JSONB operations, CROSS JOIN LATERAL, window functions, generate_series timeline gap filling, and daily_metrics UPSERTs for fast dashboard reads.
- Built a bilingual analytics dashboard with Next.js App Router, ES/EN next-intl routing, TanStack Query hydration, Zod response validation, Recharts visualizations, run-status polling, post-run auto-refetch, Sentry, and PostHog consent gating.

Stack: Next.js 16, React 19, TypeScript strict, FastAPI, Python 3.12, PostgreSQL, asyncpg, Alembic, ARQ, Redis, Stripe, OpenAI, Claude, Perplexity, Gemini, Docker, GitLab CI/CD

DGT Practice - Telegram Mini App | Live product | Solo Founder / Fullstack Product Engineer

Telegram Mini App + bot for Spanish DGT driving theory exam preparation, with official content ingestion, AI explanations, Russian helper translations, spaced repetition, adaptive practice, and readiness scoring.

- Architected and shipped a Next.js 16 + PostgreSQL + Prisma monorepo owning the full stack: Telegram Mini App UI, bot integration, Playwright-based content ingestion, AI enrichment, spaced repetition, readiness scoring, and admin console.
- Designed a 5-stage content pipeline: scraping official DGT sources, SHA256 image deduplication, content-level question deduplication with superset merging, keyword-based topic classification, and OpenAI structured-output enrichment with per-call audit records.
- Implemented Telegram Mini App authentication without third-party auth libraries: HMAC-SHA256 initData validation, custom HMAC-signed cookie session, and requireCurrentUserFromRequest guards in Route Handlers.
- Modelled a 20+ table PostgreSQL schema for questions, source attribution, structured explanations, vocabulary, spaced-review state, readiness snapshots, ingestion runs, and AI generation audit trails.
- Built probabilistic exam-readiness scoring using Beta-distribution mastery, topic shrinkage, a 10,000-run Monte Carlo pass-probability simulation, and immutable versioned snapshots to avoid request-time recomputation.
- Built a Leitner-style spaced repetition scheduler and topic-weighted roulette question selection with lapse reinforcement, response-time tracking, conditional translation UI, and Russian-language admin workflows.

Stack: Next.js 16, React 19, TypeScript strict, PostgreSQL, Prisma 7, Telegram Bot API, Telegram Mini App SDK, OpenAI structured outputs, Playwright, Docker

WriteAs.pro | Live product | Solo Founder / Fullstack Product Engineer

AI writing SaaS that learns a user's writing style from uploaded samples and generates on-brand content across LinkedIn posts, newsletters, threads, scripts, and other formats.

- Built the product end-to-end: product concept, UX, architecture, frontend, backend, database, AI pipeline, deployment, and monetization setup.
- Designed a style-conditioned generation pipeline with LLM-based style profile extraction, RAG via Qdrant vector search, OpenAI fine-tuning for large writing datasets, and real-time output streaming via SSE.
- Implemented a consent-first persona licensing architecture with encrypted training data, right-to-erasure including OpenAI model deletion flow, and Stripe-based credit monetization setup.

Stack: Next.js 14, TypeScript, FastAPI, Python, PostgreSQL, Redis, Qdrant, OpenAI API, embeddings, fine-tuning, Whisper, Celery, Docker, Stripe, AWS S3

DFM (driler.fm) | Live product with users | Solo Founder / Fullstack Product Engineer

Music tool for live streamers: real-time on-air display, shareable playlists, music management, and streaming/donation service integrations.

- Built and shipped the product solo with production users on the free plan, covering Next.js frontend, PostgreSQL schema, Prisma data layer, API routes, integrations, and operational workflows.
- Implemented Node.js workers for YouTube chat processing and donation event handling, plus YouTube/Rutube API integrations for playlist and media workflows.

Stack: Next.js, TypeScript, PostgreSQL, Prisma, Node.js workers, YouTube/Rutube APIs, OpenAI API, Stripe configured

PROFESSIONAL EXPERIENCE

Xaru.io | Remote | June 2021 - Present

Lead Frontend Engineer

Fintech startup operating B2B and B2C financial products: corporate account management with virtual card issuance, advanced spending analytics, FIAT/crypto exchange and wallet, and internal tooling.

- Led frontend architecture for 3 fintech products from scratch and contributed to product planning, architecture decisions, delivery planning, code review, mentoring, interviews, and engineering standards.
- Developed a B2B financial platform used by 150-200 companies, supporting ~10,000 monthly transactions and real-time analytics dashboards; overall products served 1,000+ monthly active users.
- Led frontend development of a B2C crypto/fiat platform with ~800 users, handling real-money transaction UX, wallet flows, conversions, analytics, and onboarding.
- Independently designed, built, and shipped a fullstack internal crypto portfolio tracker as sole engineer: Next.js frontend/API layer, Node.js workers, Prisma/PostgreSQL schema, crypto data integrations, admin/analytics UI, and production workflows; DevOps handled separately.
- Improved load time, bundle size, and heavy analytics rendering performance by ~2x through code splitting, lazy loading, SSR/ISR strategy, and targeted frontend optimization.

Stack: Next.js, React, TypeScript, WebSockets, Recharts/D3, Node.js, PostgreSQL, Prisma, Docker

Etc. Professional Translators (product: Marfa) | Remote | January 2019 - June 2021

Frontend Developer

B2B SaaS platform for professional document translators. Backend fragmented and anonymized source documents so translators worked only with isolated segments. The product was acquired by Sberbank.

- Owned key frontend features in a 2-engineer team and contributed to a B2B SaaS product later acquired by Sberbank.
- Led migration of ~90% of the frontend codebase from JavaScript to TypeScript, improving reliability and developer experience.
- Built complex UI flows for document management, secure task assignment, translator-PM collaboration, review states, and operational dashboards.

Stack: React, TypeScript, CRA

Stafory LTD (product: Vera Robot) | Remote | January 2017 - November 2018

Frontend Developer

HR tech startup building an automated cold-calling recruitment robot and CRM for candidate pipeline management.

- Built and maintained the client-facing CRM interface for calling campaigns, candidate profiles, screening results, and decision-tree recruitment flows.
- Developed a Chrome extension for automated candidate data extraction and built an internal UI component library.
- Introduced Scrum/Agile processes, sprint ceremonies, retrospectives, and iterative delivery practices.

Stack: JavaScript, React, Chrome Extensions API, HTML/CSS

Vsemayki.ru / Sky-IT / WOW Digital Agency | 2009 - 2017

Frontend Developer

- Frontend development across e-commerce and agency projects: site-wide redesigns, custom print designer, client storefronts, touch-screen kiosk product, HTML/CSS/jQuery client work, and early product delivery experience.

Stack: JavaScript, HTML, CSS, Bootstrap, jQuery

EDUCATION, CERTIFICATIONS & LANGUAGES

Education: Siberian State University of Geosystems and Technologies, Novosibirsk - Bachelor of Engineering, Information Technology (2012 - 2015)

Certification: Scrum / Agile Practitioner Certificate

Languages: Russian - Native/C2; English - B2 Professional; Spanish - A2 Basic; Polish - A1 Beginner