John Szaniszlo

Canyon Country, CA, USA
661-606-5974 • johnszaniszlo@gmail.com
linkedin.com/in/john-szaniszlo-02685a2ba github.com/jjszaniszlo

Full Stack Software Engineer with a year of production TypeScript and React experience. Lead engineer on an AI marketing platform (about 81% of commits): designed its contract-first API and most of its PostgreSQL schema, built React UI, and set up per-pull-request preview environments with end-to-end tests. On a second product, contributed to the Python FastAPI backend and the document ingestion UI, and owned the AWS infrastructure as Terraform. Uses Claude Code daily from design through testing and release. B.S. Computer Science, CSU Northridge (December 2025). US citizen, available immediately, open to remote work or on-site in the San Fernando Valley / Santa Clarita area.

Core Competencies

- **Front End**: TypeScript, JavaScript, React 19, TanStack Start/Router/Query, Tailwind CSS v4, shadcn-style components on Base UI and Radix UI, accessible markup (ARIA), Astro
- **Back End & APIs**: Node.js, contract-first RPC and REST APIs (oRPC, Zod, OpenAPI), Python (FastAPI, Pydantic, SQLAlchemy, Alembic, Redis/RQ workers), authentication and organizations (WorkOS, Clerk), durable workflows (Trigger.dev, Temporal)
- **Databases**: PostgreSQL (Neon, Amazon RDS), schema design and migrations, Drizzle ORM, row-level security for multi-tenant data, pgvector
- **AI Features in Products**: Vercel AI SDK, OpenRouter, LLM agents with tool calling and structured outputs, retrieval-augmented generation, web scraping with Firecrawl, human-in-the-loop approvals; Claude Code for day-to-day engineering
- **Testing & Delivery**: Vitest, Playwright end-to-end tests, MSW, database-backed integration tests, GitHub Actions CI/CD, ephemeral preview environments, Vercel, Docker, Biome, Renovate; Agile (Kanban) team
- **Cloud & Infrastructure**: AWS (VPC, EC2, RDS, S3, CloudFront, Route 53, IAM), Terraform, Kubernetes (k3s), FluxCD GitOps, Linux, NixOS, OpenTelemetry, Sentry
- **Languages**: TypeScript, JavaScript, SQL, Python, C++, Bash

Work Experience

- **Full Stack Web Engineer** — **September 2025 - September 2026**
  CAS BioSciences, LLC — *Los Angeles, CA*

  Empyris – AI marketing content and strategy platform for the company's supplement brands (lead engineer, ~81% of commits, June - September 2026)
  - Architected a TypeScript monorepo (pnpm, Turborepo): TanStack Start and React 19 on Vercel, a contract-first oRPC API across 11 routers, Drizzle ORM on Neon PostgreSQL, Trigger.dev durable workflows, WorkOS authentication and organizations
  - Argued for, and won, one typed API contract shared by the UI and the AI agents instead of separate clients per surface; the typed client and the agents' tools are generated from it, and every action is testable with MSW-mocked RPC across 51 Vitest suites
  - Built UI in React 19 with TanStack Router and Query, Tailwind CSS and a shadcn-style component library on Base UI, with ARIA state attributes across 17 components
  - Wrote most of the PostgreSQL schema (12 of 17 tables, with foreign keys, unique constraints and indexes) and 15 of its 20 migrations; implemented 80 row-level security policies for multi-tenant isolation, verified by a two-role integration test in CI
  - Built the product's AI features: an agent kernel running nine declarative agents (brand voice, Amazon A+ Content, PPC and market strategy, image generation) with tool calling, Zod-validated structured outputs and idempotent retries, plus a Firecrawl web scraper that turns a product page into structured marketing inputs

- Built human-in-the-loop approvals as one layer wrapping the whole API, so AI-generated content ships only after a person approves it
- Built ephemeral per-pull-request preview environments (Neon database branch, Vercel deploy, Trigger.dev branch) with Playwright end-to-end tests against the live preview, automatic teardown and a nightly cleanup job; added Pino and OpenTelemetry logging, Renovate and the team's docs site

CASGen – AI website generator with domain purchase and provisioning (application contributor; owned infrastructure)

- Contributed about 45% of commits to the original Python backend (FastAPI, Pydantic, SQLAlchemy, Alembic, Redis/RQ workers): role-gated endpoints, domain registration with retry and throttling, and a provisioning worker with idempotent AWS operations; the team later migrated the backend to TypeScript
- Built the document upload and ingestion flow for the RAG knowledge base: presigned-S3 uploads, ingestion UI and integration of the GPU document extraction service; about 2,000 documents and 2 million pages ingested, cutting manual data entry time by about 80%
- Redesigned the domain-purchase UI with per-domain error reporting and built LLM reranker settings in the admin dashboard
- Owned all AWS infrastructure as Terraform (nine modules: VPC, RDS PostgreSQL, EC2 Auto Scaling, S3/CloudFront, Route 53, ACM, ECR, IAM), with plan-on-pull-request reviews through Digger and GitHub OIDC, so no static AWS keys exist in CI
- Migrated Kubernetes from EKS to self-managed k3s on a custom NixOS image, cutting AWS costs 57% (~$350 to ~$150 per month); shipped to production daily through FluxCD GitOps (159 automated deploys) with database migrations gating each rollout

Math and Computer Science Tutor — **September 2022 - December 2025**
College of the Canyons — *Santa Clarita, CA*

- Tutored undergraduates one-on-one and in small groups in mathematics, physics, and computer science while completing both degrees
- Improved student exam scores by personalizing explanations; tracked progress and adapted teaching methods weekly

Independent Projects (Open Source)

mfd – cockpit multifunction display — **August 2026**
C++26, Qt 6, Gtk4, OpenGL 3.3, MIT licensed — *github.com/jjszaniszlo/mfd*

- Moving map with HTTPS map tiles and a disk cache, flight plan, systems and traffic pages; a UI-toolkit-independent core library (84 unit tests) drives Qt and Gtk4 front ends from one shared scene model, with clang-tidy and sanitizers in CI

pfd – primary flight display — **January 2026**
C++23, OpenGL 3.3, MIT licensed — *github.com/jjszaniszlo/pfd*

- Flight display fed over UDP by an included 6-DOF flight simulation or FlightGear; zero heap allocations in the render loop (verified) and published frame-time jitter measurements; 117 automated checks run in CI

Education

Bachelor of Science in Computer Science — **2024 - 2025**
California State University, Northridge — *Northridge, CA*
Degree completed December 2025. Topics: full stack software engineering, DevOps

Associate's Degree in Computer Science — **2021 - 2023**
College of the Canyons — *Santa Clarita, CA*
Degree completed 2023. Topics: full stack software engineering, DevOps

References

Available upon request.