Raffaele Cataldo

(707)409-8076 | raffaelecataldo.dev@gmail.com | raffaelecataldo.com | LinkedIn | GitHub

Frontend focused full-stack engineer who thrives on small, high-ownership remote teams. Experienced in React/Next.js and Django/HTMX, building test-driven applications in regulated and data-intensive domains.

Technical Skills

Languages: TypeScript, JavaScript, Python, Rust
Frontend: React, Next.js, Redux, Tailwind, HTMX, Web Components, Chart.js, Playwright, Jest, React Testing Library
Backend: Django, Celery, Redis, Node.js, Express, Axum Web
Data & Tools: Postgres, SQLite, MongoDB, Git, Docker, Linux, AWS, Superset, OpenTelemetry, Zod, Prisma

Work Experience

SnapLogic — Senior UI Engineer
Feb. 2025–Present

Remote (California) — contracted Feb.–Dec. 2025, converted to FTE

SnapLogic Intelligent Modernizer: greenfield platform that ingests legacy ETL workloads (Informatica, DataStage), analyzes them, and generates modern SnapLogic pipelines, reducing migration cost and time by up to 80%.

- Led frontend architecture on a 3-engineer team, choosing Django + HTMX + Web Components over a separate SPA/API stack to keep the codebase Python-centric for an ML product while delivering SPA-like interactivity through server-rendered views with lazy loading and partial updates
- Enabled daily deployments, major refactors, and AI-assisted development by building a parallelized Playwright suite with accessibility-first locators, keeping tests decoupled from markup and resilient to rapid UI changes
- Cut file-tree load time from ~325 ms to ~25 ms with a Redis caching layer and signal-driven invalidation, then to subsecond renders via lazy loading (DOM reduced from 1,000+ to 20–50 nodes)
- Eliminated cross-region UI sync bugs by designing a pub/sub event system with code-generated event definitions shared across server and client, decoupling UI regions from each other
- Replaced manual customer onboarding with a self-service upload pipeline backed by async task processing (Celery), cutting parse time from 10 minutes to under 1 minute
- Accelerated AI-assisted development by structuring feature work around versioned technical proposals, giving coding agents well-scoped problem definitions, documented trade-offs, and up-to-date decision context

Inabia — Next.js Developer
Jan. 2024–Sept. 2024

Remote (California) — Contractor at Consumer 51

- Led frontend development of a HIPAA-compliant clinical workflow system with real-time critical alerts and threshold-based validation, while mentoring 2 developers on React, Hooks, and advanced TypeScript
- Reduced context switching for medical staff by architecting a dockable multi-window system with custom React Context providers, unsaved-change guards, and confirmation-gated record deletion
- Cut QA cycle time by 60% by establishing the team's testing practice; authored Jest/RTL integration tests, created test examples, and led team training sessions
- Reduced API response times by 75% by parallelizing data fetches and eliminating over-fetched relations from patient page layouts

Encore — Full Stack Developer
May 2022–April 2023

Remote

- Built KYC and payment onboarding flows integrating Stripe Identity verification, Plaid bank account linking, and SSN validation with input masking, secured by 2FA for high-value transactions
- Reduced runtime errors across the platform by driving a JavaScript-to-TypeScript migration and configuring strict ESLint rules
- Developed a reusable multi-step funnel system (usePanelController hook, shared form components with react-hook-form) used across payment, invoice, team, and workspace onboarding flows

Open Source

Ratatui
Rust TUI framework

- Identified and fixed a rendering issue affecting layout behavior at small terminal widths and submitted documentation improvements

Volunteer

Radical Engineers   **March 2022–August 2023**
Software Engineer

- Built a Twilio phone-tree app (Python/Flask) enabling automated call routing for criminal justice nonprofit Straight Ahead; designed and developed a workplace survey app (React/TypeScript/NextAuth) for UK nonprofit Sic in the City

Personal Projects

International Clock — Full-stack time zone app: React/TypeScript client deployed to S3/CloudFront via GitHub Actions, backed by a Rust API on AWS Lambda using tzf-rs for coordinate-to-timezone resolution — sub-2ms response times, serverless architecture — clock.raffaelecataldo.com

ruhr — Rust CLI world clock with SQLite caching and location aliases, published on crates.io — github.com/farmeroy/ruhr

Education

The Open University   **Oct. 2021–2028** (*expected*)
BSc Honours Computer Science and Statistics

Career Foundry   **Sept. 2021–March 2022**
Fullstack Web Development Bootcamp

Languages

English (native) │ German (advanced) │ Spanish (advanced) │ Italian (intermediate) │ Turkish (intermediate)