

Evan Purkhiser

Full-Stack Systems Engineer

[in](#) — evanpurkhiser.com — evanpurkhiser@gmail.com

(...)-...-0474

Full-Stack Engineer with a 15-year career of building and scaling product experiences. Passionate about UX details, code quality, and developer experience. Known for end-to-end ownership and acting as an engineering force-multiplier.

LANGUAGES / LIBRARIES

TypeScript / React
Rust
Golang

Python / Django
Postgres SQL
Node.js

TOOLS / SOFTWARE

MacOS / Linux
Neovim
Datadog / Sentry

git
Nginx
Docker

Redis
GitHub Actions
Bash / Shell

EDUCATION

B.S. Computer Science
The University of Akron
2009 – 2013

EXPERIENCES

Sentry Senior Software Engineer

August 2017 – Present

Shipping Products

- Launched Uptime Monitoring from zero to GA, owning the entire architecture across a high-concurrency Rust checker, Kafka pipeline, Python processor, and React frontend; scaled to 100k+ customer monitors.
- Led multi-year end-to-end ownership of Cron Monitoring, evolving it from early PoC to a highly reliable system processing 60M+ daily check-ins, including timing-sensitive processors, internal anomaly detection, and frontend user experience.
- Built Slack Integration from the ground up atop the integration platform, becoming our most used integration, 120k+ orgs.
- Rebuilt onboarding experience with a heavy focus on clarity, polish, and micro-interactions, creating strong first impressions.

Platform Engineering

- Designed and implemented a generalized integrations platform (OAuth, permissions, feature gating, UI primitives), unlocking rapid development of Slack, GitHub, Jira, Vercel, etc integrations.
- Established a shared chart rendering pipeline across frontend and backend, enabling notification charts (email, Slack, etc) to reuse frontend code, significantly reducing maintenance overhead and ensuring consistent presentation.
- Reduced inconsistent frontend event search experiences by building a reusable, type-safe parsing grammar, enabling incremental parsing and consistent behavior across UI surfaces, while improving developer ergonomics and maintainability.

Systems Stewardship

- Led Python 2 → 3 migration across the monolith, unblocking dependency upgrades and future platform work.
- Proactively led multi-year frontend modernization efforts to preempt scaling and maintainability risks. Areas included upgrading routing infrastructure (React Router 3→6), component architecture (class → functional), and testing strategy (Enzyme → React Testing Library). Enabling faster, safer iteration and higher engineering velocity across the organization.
- Drove TypeScript adoption, converting 500+ JavaScript files to TypeScript across all aspects of Sentry's frontend application.
- Cut CI build times by 40% via a GitHub Actions Docker test image with dependency caching.

Organizational Level Leadership

- Led Frontend Technical Steering Committee, accomplishing complete migration to TypeScript, organization-wide code quality initiatives, React modernizations, and testing infrastructure improvements across 7+ product teams.
- Mentored engineers on system design, UX interaction, testing philosophy, and pragmatic tasteful decision-making.

Crunchyroll / VRV Senior Software Engineer

August 2014 – August 2017

- Designed and developed high-performance APIs using golang to serve video content metadata.
- Full-stack implementation of a "content partner" portal for video content management. Developed using React + Redux. Collaborated with designers to produce a polished, branded UI with an emphasis on usability.
- Promoted quality coding standards, git best practices, review best practices, automated style lint tooling, and improved developer environment tooling throughout the organization for a high-quality developer experience.

Acadaware Lead Web Applications Developer

March 2011 – July 2014

Developed a web application for managing Experiential Physician Education programs. Responsibilities included all stages of development, from initial requirement gathering to deployment and configuration of a production server environment via Chef.

PERSONAL HIGHLIGHTS

Prolink Tools

[prolink.tools](#) [prolink-tools](#) [prolink-connect](#)

Reverse-engineered and implemented a proprietary UDP/TCP protocol as a TypeScript library; built as an Electron + React app enabling DJs to stream real-time track metadata with broadcast-quality transitions.

Dots – Dotfile Manager

[dots](#) [dots-personal](#)

Cascading configuration management solution for sharing personal dot configuration files across systems and platforms.