

# Ryan Saperstein

rmsaperstein@gmail.com ◊ 908-797-6374 ◊ Flemington, NJ  
github.com/rmsap ◊ linkedin.com/in/ryansaperstein ◊ rmsap.github.io

*Software engineer building developer tools, currently building Phoenix Gen, a typed API codegen tool that eliminates type drift across TypeScript, Python, Go, and OpenAPI schemas*

## EDUCATION

**Northeastern University, Boston, MA**

September 2021 - May 2025

*B.S., Computer Science (AI Concentration), GPA: 3.93 / 4.0 · Summa Cum Laude*

## TECHNICAL SKILLS

**Languages:** Rust, TypeScript/JavaScript, Python, Java, SQL, Scala

**Frontend:** React, React Native, Redux

**Backend & Infrastructure:** PostgreSQL, Django, Node, Kubernetes, AWS (S3), GCP, Azure, Docker, MongoDB

## SELECTED PROJECT

**Phoenix** | *Rust, Cranelift* | [github.com/rmsap/phoenixlang](https://github.com/rmsap/phoenixlang)

March 2026 - Present

*Built in two months by pairing with Claude Code and Codex across the full stack, lexer to LSP.*

- **Phoenix Gen, one schema, four languages:** typed API codegen tool that emits TypeScript, Python, Go, and OpenAPI 3.1 from a single `.phx` schema; **where** constraints on struct fields compile to runtime validators in each target language, eliminating type drift between frontend and backend
- **Compiler pipeline in Rust:** 15-crate Cargo workspace covering lexer, recursive-descent parser, two-pass semantic analyzer with name resolution and type inference, SSA-based IR with a standalone round-trip interpreter for verification, and Cranelift-backed native codegen
- **Tracing GC with portability in mind:** precise root-finding via per-thread shadow stack and a GcHeap trait abstraction that isolates the collector from codegen so a WASM-GC implementation can drop in without touching the runtime
- **Full developer tooling:** VS Code LSP with diagnostics, hover, autocomplete, go-to-definition, find-references, and rename; CLI with separate `run`, `run-ir`, `build`, `check`, and `gen` commands; 3,000+ tests at ~90% coverage

## EXPERIENCE

**Playbook Sports**

Hoboken, NJ

*Software Developer*

November 2025 - Present

- Migrated real-time chat from REST polling to WebSockets, cutting message latency and unblocking real-time features that polling couldn't support
- Discovered a client-side token exposure in the magic-link auth flow; redesigned around server-side cryptographically signed tokens with 15-minute expiration, eliminating indefinite access from intercepted links
- Replaced a public S3 bucket with per-user scoped access control, closing a data exposure risk that scaled with every new league onboarded
- Moved deep-link routing from client to a Django backend service, removing 200 lines of app-side logic and enabling link structure changes to ship without app store releases

**Generate Product Development**

Boston, MA

*Tech Lead (Frontend)*

August 2024 - December 2024

- Co-led team of six shipping an MVP mobile app for an early-stage founder; owned all frontend architecture across a React Native/TypeScript codebase while co-lead owned Go/SQL backend
- Designed the component, typography, navigation, and data-fetching patterns the team built every screen against: an auth-aware root navigator with nested stack and tab navigators, plus a typed React Query hook layer that centralized auth, caching, and loading state
- Patterns held up through six engineers and full MVP delivery; no major refactors required as feature scope expanded

**Broad Institute of MIT and Harvard**

Cambridge, MA

*Software Engineer Co-op*

January 2024 - June 2024

- Added support for a new revision of the WDL (Workflow Descriptor Language) spec in our Scala execution engine
- After a bug I caused reached production undetected, traced it to a years-old TODO in the integration testing framework limiting type coverage to strings and ints; rewrote the framework to support complex types and backfilled coverage

**VMware**

Boston, MA

*Co-op Member of Technical Staff*

January 2023 - December 2023

- Built CLI tool that parses customer log bundles (hundreds of thousands of lines) into a queryable, filterable interface, reducing debugging time on support tickets from days to hours

**Saperstein Software Solutions**

Remote

*Owner and Software Developer*

March 2022 - Present

- Shipped two React Native apps for a NJ non-profit; 2,000+ downloads on iOS and Android across 6 counties

*Writing: One Schema, Four Languages: Solving API Type Drift · The Anatomy of Push Notifications · April 2026*