JIMMY MILLER

jimmyhmiller@gmail.com | jimmyhmiller.github.io | @jimmyhmiller

WORK EXPERIENCE

SHOPIFY (AUG. 2022-MAY 2023)

Staff Compiler Engineer

- Worked on cutting edge JIT Compiler for Ruby (YJIT)

CISCO - THREAT RESPONSE (MAY 2021-AUG. 2022)

Technical Leader

- Increased performance of security investigation engine 10-100x
- Lead and architected efforts for analyzing large scale security events
- Worked with partners across teams to ensure org-wide availability of data

KEVEL (JAN. 2019-MAY 2021)

Senior Software Engineer

- Revitalized reporting system including infrastructure, db, deployment, backend and frontend.
- Enabled processing of 30+ billion requests per month.

HEALTHFINCH (JAN. 2018-JAN. 2019)

Senior Software Engineer

- Used rules engine to encode medical protocols for patient care.
- Developed custom visualization for medical recommendation templating.

NEXTGEAR CAPITAL (JULY 2017-DEC. 2017)

Senior Software Engineer - Contractor

- Built neural network based document processing system.

TRABIAN TECHNOLOGY (APRIL 2016-JULY 2017)

Lead Software Engineer

- Lead all development efforts from prototype to production
- Performed large-scale code migrations utilizing ast rewriting to modernize client codebases.

NEXTGEAR CAPITAL (OCT. 2014-APRIL 2016)

Senior Software Engineer

- Developed microservice architecture to support monolithic application.
- Led workshops, gave presentations, and created internal discussion group.
- Integrated application into corporation-wide distributed system.

Additional experience available upon request

PODCAST

- Future of Coding

TECHNOLOGIES

- AWS, Clojure, ClojureScript, Docker, GraphQL, Haskell, Idris, Java, JavaScript, Python, React, Redshift Rust, Snowflake, SQL

INTERESTS

- Analytic Philosophy
- Distributed Systems
- Programming Language Theory

TALKS

- Meander: Declarative Explorations at the Limits of FP
- Paradigms Without Progress: Kuhnian Reflections on Programming Practice

READINGS I ENJOY

- "Pygmalion: A Creative Programming Environment" by David Canfield Smith
- "On Understanding Data Abstraction, Revisited" by William R. Cook
- "Simple and Effective Type Check Removal through Lazy Basic Block Versioning" by Maxime Chevalier-Boisvert
- "Programming as Theory Building" by Peter Naur
- "Lazy Evaluation of Transactions in Database Systems" -Jose M Faleiro, et al.

"We make not just to have, but to know" - Alan Kay