Adi Zimmerman

adizim.me | github.com/adizim | linkedin.com/in/adizim | adizimmerman1@gmail.com | (408) 510-8702

Education

University of California, Berkeley — Aug 2016 — Dec 2019

- B.A. Computer Science, Dec 2019. GPA: 3.6/4.0.
- Coursework: Data Structures, Algorithms, Computer Architecture, Operating Systems, Machine Learning, Artificial Intelligence, Security, Databases, Discrete Math, Linear Algebra, Probability Theory.

Employment

Lyft, Dispatch Infrastructure — Software Engineer — Feb 2020 — Present

- Technical lead for matching pipeline, redesigned from Python worker script into Golang microservices. Key metrics include Improved latency by 2x, reduced compute resources by 10x, and immeasurably improved reliability, observability, and developer confidence.
- Designed and implemented data models for demand and supply lifecycles. Migrated various life cycles to event driven architecture to support streaming as opposed to naive batch retrieval.
- Implemented end to end regression testing infrastructure for the matching pipeline to isolate degradations pre-code commit.
- Organizational cross-functional leader for peak event readiness (Halloween, New Years Eve).
- Mentored intern to full time hire with strongly exceeding expectations.

Lyft, Dispatch Infrastructure — Software Engineering Intern — May 2019 — Aug 2019

- Built a scalable, reliable, low latency per-driver observability system from database to API level in Go that captured and alarmed on up to ~30k supply failures per 2-5 second matching cycle.
- Isolated 30 supply failure reasons in complex legacy Python codebase as input to observability system.
- Designed an algorithm for editable party size on shared rides that decreased unmatched cancel rate by 1.6%.

Cisco, Kinetic IoT — Software Engineering Intern — May 2018 — Aug 2018

- Worked on IoT messaging, focused on distributed systems with high horizontal scalability. Java.

Marvell, Embedded Platform — Software Engineering Intern — June 2017 — Aug 2017

- Worked on IoT networking for SBCs and external devices. C and Python.

Research

RISElab (Anyscale) — Ray, Tune: Scalable Hyperparameter Search — Jan 2019 – May 2019

- Ported ask and tell search algorithms for usage on Tune (scikit-optimize, Facebook's Nevergrad and Ax).
- Led design and implementation of Tune's RESTful web server and post experimentation analysis tools.
- Iterated and improved on user-facing documentation and tutorial.

Languages and Technologies

- Programming: Python, Go, Java, C, HTML/CSS/Javascript, RISC-V Assembly.
- Technologies: Unix, Git, SQL, AWS (DynamoDB, EC2 on EKS, S3), Kubernetes, Microservices, Docker, Jenkins, Spark. Grafana, Prometheus, ElasticSearch, Pagerduty
- Languages: English, Hebrew.