Adhvik Kanagala

848-256-9450 | adhvik.kanagala@gmail.com | linkedin.com/in/adhvik-kanagala | github.com/addykan

EDUCATION

Carnegie Mellon University   Pittsburgh, PA
Bachelor of Science, Computer Science, Concentration in Computer Systems   *Aug. 2020 – May 2024*

EXPERIENCE

Vorticity, Inc   Jan 2024 – May 2024
Software Engineer #8   *Redwood City, CA*
- Designing and building a distributed job scheduling library for self-hosted datacenter
- Eliminate explicit scheduling of compute jobs, transforming the datacenter into a serverless interface for engineers

Duolingo   May 2023 – August 2023
Software Engineer Intern, Data Refinery   *Pittsburgh, PA*
- Built an automated monitoring tool to help data platform area track production datasets
- Eliminated active monitoring of dataset freshness by engineers
- Built tool to visualize dependency graph for datasets used by data science team
- Engineered new way to debug low-quality datasets by data science team
- Improved sustainability of data science area's 2023 growth objectives by upgrading tooling

Facebook (now Meta)   May 2022 – August 2022
Software Engineer Intern, WhatsApp Business Infra   *Menlo Park, CA*
- Migrated a critical billing service in WhatsApp Business backend to serverless infrastructure, reducing future development timelines on this service from weeks to hours
- Eliminated dependency on chat infra team for deployment/maintenance of business services
- Developed onboarding documentation and starter tasks for new engineers

School of Computer Science, Carnegie Mellon University   Feb 2021 – December 2023
Teaching Assistant   *Pittsburgh, PA*
- Distributed Systems 1 semester, Intro to Programming 6 semesters
- Updated and released two of the four programming projects for Distributed Systems, built in Go
- Taught 2 recitations per week, held office hours, and graded assignments

PROJECTS

c0c | *15-411 Compiler Design*   Spring 2024
- Building a compiler for the C0 language, a Turing-complete subset of C, targeting x86-64
- Implementing all phases of compilation, from lexing and parsing to explicit x86-64 instruction generation

Raft Consensus Algorithm | *15-440 Distributed Systems*   Fall 2022
- Referenced the original research paper to simulate the Raft consensus algorithm in Go
- Designed a DFA-style peer model that implements leader election and log replication

Distributed Bitcoin Miner | *15-440 Distributed Systems*   Fall 2022
- Implemented the Live Sequence Protocol, a bespoke network protocol that extends UDP
- LSP uses a server/client model to support automatic packet resending, exponential backoff, ordered message delivery
- Used LSP to build a mock distributed bitcoin miner with round-robin task scheduling

SLSports | *Developer*   Summer 2021
- Used the MERN stack to digitize player management for Sri Lankan national athletics
- Built a desktop website in React for sports associations and national sports council to track and approve activities, payments, and competitive standings
- Deployed a mobile app with Flutter for players and coaches to track activities and payments

TECHNICAL SKILLS

Languages: Python, Go, C++, Standard ML, C, OCaml, Erlang, Hack/PHP, Thrift, x86–64, Typescript, Dart, R
Developer Tools: Git, Docker, Mercurial, Thrift, Flask, SQL, BigQuery, Unix, React, MongoDB, Flutter
Coursework: Compiler Design, Distributed Systems, Database Systems, Parallel Computer Architecture and Programming, Algorithm Design and Analysis, Computer Security, Machine Learning, Writing for the Professions