Philip Zhu

Pittsburgh, PA | 510-206-1672 | philipzhu@cmu.edu | github.com/philipzhux

EDUCATION

Carnegie Mellon University Pittsburgh, PA
Master of Science, **Computer Science** - Information Networking; GPA: 4.0/4.0 Expected Dec 2024
Core Courses: **Cloud Computing**, **Distributed Systems**, Computer Systems, Software Engineering, Information Security, Parallel Programming

University of California, Berkeley Berkeley, CA
Exchange Undergraduate Student in **Computer Science**; GPA: 4.0/4.0 Jan 2022 - Aug 2022
Core Courses: **Operating System** and System Programming, Computer Security, Intro to Artificial Intelligence, Data Structure and Algorithms

SKILLS SUMMARY

Programming: C, C++, Python, Java, Go, Javascript, SQL
Web Dev & Database: MySQL, MongoDB, Redis | Flask, Django, Express.js, Spring Boot | HTML, CSS, JQuery, React, Redux
Cloud & DevOps: Aws S3, RDS, Google Kubernetes Engine, Docker, Kubernetes, Terraform, Jenkins Pipeline, Jira, Bitbucket
Miscellaneous: Git, Agile Software Development, Quality Assurance, gRPC, Linux Kernel, Unit Testing, Performance Analysis

WORK EXPERIENCE

Boxifly Inc. Guangzhou, China
Software Engineer Aug 2022 – Aug 2023
Tech Stack: ***React, Flask, MySQL, Redis*** | Spearheaded the development and scaling of a door-to-door storage delivery functionality

- Implemented **RESTful API**s and deployed Kubernetes-orchestrated **microservices** for ordering and delivery task dispatching, reducing average response time by 70% through **Redis caching** and optimized database indexing strategies
- Rearchitect the application backend for the transition to a serverless model and optimized cloud architectures in a team of three, increasing cost efficiency by 35%
- Built an **performance metrics dashboard** using Grafana, integrated with Prometheus for real-time data collection, enabling continuous monitoring of system health, user activity, and service uptime
- Elevated service uptime from 97% to 99.99% amid a 250% traffic surge in promo peak season, boosting customer retention by 20%

UC Berkeley Sky Lab Berkeley, CA
Undergraduate Research Assistant April 2022 – Aug 2022
Proposed and implemented a system to advise cost-effective state-of-the-art LLM memory optimizations schemes on cloud

- Developed PAPAYA, a system to predict the space-time tradeoff on LLM memory optimizations schemes in distributed training settings and advise the optimal schemes before the workload
- Profiled memory footprint on training and referencing workloads with C++ ***CUDA runtime*** library to provided data points to verify the proposed performance model
- Established ***Github Actions*** to streamline fast iterations of experiments on different LLM models and configurations including BERT and GPT on AWS ***EC2*** and leveraged ***Terraform*** to automate the provisioning of a matrix of GPU configurations

Shenzhen Research Institute of Big Data Shenzhen, China
Backend Software Engineer Intern Jun 2021 – Aug 2021
Tech Stack: ***Express.js, MySQL*** | Developed analytics and authentication API endpoints for a campus big data dashboard

- Developed academic performance and well-being analytics API and established Jenkins pipelines to streamline build, testing, and deployment workflows, decreasing the integration cycle time for code changes by 90%
- Improved the efficiency of the user login process by integrating the authentication API with the university OAuth, leading to a 45% reduction in user login time

PROJECTS

Simplified C - a C language Compiler implemented in C++

- Built a lexer, parser, syntax analyzer, and code generator to compile C code into MIPS assembly
- Implemented a NFA regular expression engine and a Bison-like LR(1) table driven parser generator that supports programmed customized productions rules and semantic routine
- Developed data structures for efficient symbol table management and AST traversal and derived LR(1) production rules and semantic routines for C language

TINYKV - Fault-tolerant Raft-based Horizontally Scalable Distributed Key-value Storage System in Golang

- Integrated ***Raft*** algorithm features including membership and leadership changes, ensuring system fault-tolerance
- Implemented ***multi-version concurrency control,*** optimizing transaction management and atomic operations
- Developed TinyScheduler for centralized node management and timestamp generation, utilizing Protocol Buffers over ***gRPC*** for efficient inter-node communication

PINTOS - Operating System Kernel Development Project at UC Berkeley in C

- Led a team of four to design and implement core components of a unix-like ***operating system kernel***
- Implemented semaphore system calls and user-space interfaces of ***synchronization primitives*** (mutex and condvar)
- Optimized the ***filesystem*** to an extent-based allocation and implemented unix-like inode structures
- Achieved **top 2%** overall performance among the CS162 class at UC Berkeley