Ke Li

(780)-885-0852 | kegrad2023@gmail.com | Toronto, ON, Canada
damianli.com | Linkedin

Education

University of Toronto Toronto, ON
Master of Engineering in Computer Engineering & Identity, Privacy and Security (IPS) *Jan. 2024 – Dec 2025*

- Coursework: Computer Security, Cloud Computing, Deep Learning & Neural Network, Parallel Programming, Performant System with Rust
- GPA: 4.0/4.0

University of Alberta Edmonton, AB
Bachelor of Science in Computer Science *Sep. 2019 – Aug 2023*

- Coursework: Operating System, Computer Networks, Computer Architecture, Web Development, Mobile App Development, Database Management, Machine Learning, Agile Methodology
- Awards: *Dean's Honor Roll (22-23)*

Experience

Student Developer Jan. 2023 – Apr. 2023
University of Alberta ALT Lab *Edmonton, AB*

- Developed a website that generates interactive word graphs based on Cree words and their domains using React, D3.js and Docker.
- Optimize graph load times from 10 to 0.5 seconds by implementing the Lazy loading time Design Pattern.
- Reviewed 50+ PR and contributed 4K+ lines of code to the codebase via Git.
- Maintained web application documentation using MkDocs.
- Utilized Jest for unit testing and Cypress for E2E testing.

Back-End Engineer Intern May 2021 – Aug 2021
Nandou Six Star System integration Co., LTD *Wuhan, China*

- Wrote backend code that handled external HTTP requests from third party endpoints.
- Commended for clear communication and strong willingness to learn.
- Gained experience in Linux, Gunicorn, Nginx, SQLite and Django REST framework.

Projects

KV Store Database Github Jul. 2024 – Sep. 2024

- Build a Key-Value storage database from scratch, supporting fundamental C++ data types key-value storage.
- Developed the Memtable module by using Red-Black tree, and multi-level Index for range Scan and random Search in SST file.
- Use protobuf to serialize KV data and store it in flash / disk in compact binary format.
- Use Google Test to write unit tests for all modules.
- **Utilized**: C++, CMake, GTest, protobuf

Distributed Linux Performance Analysis and Monitoring System May. 2024 – Jul. 2024

- Developed a Docker-based setup to build project environments with dependencies, facilitating easy deployment across multiple servers.
- Implemented the monitor module using the Factory Design Pattern to create an abstract monitoring interface, including CPU status, system load, software interrupts, memory, and network monitoring.
- Built a Distributed System using gRPC; Deploying server on target machines and client library used by monitor and display modules, ensuring low coupling and high modularity.
- Utilized Protobuf for serialization to define comprehensive data structures for the project.
- **Utilized**: C++, CMake, Docker, gRPC, protobuf, Qt

Technical Skills

Languages: C++, Rust, Python, C, JavaScript, Java
Frameworks: React, Django, Cypress, Jest, GraphQL, Material-UI, Ant-Design
Developer Tools: CMake, gRPC, protobuf, Qt, Git, Docker, Kubernetes, AWS
Libraries: Scikit-learn, Pandas, NumPy, Matplotlib, Scapy, D3.js, ReactFlow, Mininet