

Zhaocheng Che

Pittsburgh, PA | chezhaocheng@outlook.com | 437-340-0470 | LinkedIn | GitHub

Education

Carnegie Mellon University, Master of Computational Data Science, Systems Track Expected Dec 2027

- **Fall 2026 Coursework:** Advanced and Distributed OS, Networking and the Internet, Deep Learning

University of Waterloo, Bachelor of Computer Science, GPA: 93.5/100 May 2023

Experience

Backend Software Developer, theScore – Toronto, ON May 2023 – Jul 2026

- Co-led the technical specification and GraphQL schema proposal for a new Rewards Hub, collaborating with product, frontend, and platform teams to define requirements and service integration contracts
- Implemented backend components for the first product feature to adopt a server-driven UI architecture using GraphQL Federation, reconciling new schema patterns with legacy types and existing service contracts
- Designed and implemented a GraphQL DataLoader using field aliases to batch downstream stitching requests, reducing service-to-service calls by 60%
- Improved GraphQL subscription reliability and observability by deduplicating updates and adding Kafka telemetry, reducing duplicate outbound WebSocket messages by 20%
- Built a k6 load-testing workflow that replayed production traffic from Datadog logs while preserving relative request timing, enabling more representative performance testing

Software Engineer Intern, Arista Networks – Vancouver, BC May 2022 – Aug 2022

- Enhanced the Arista EOS command-line interface to help users and automation inspect routing-table entries
- Profiled static-route insertion using perf and py-spy, then implemented a batch-insertion command to reduce per-route processing overhead

Undergraduate Research Assistant, University of Waterloo Jan 2022 – Apr 2022

- Implemented a new Linux `epoll` option to eliminate the overhead of the additional `epoll_ctl` system call in edge-triggered oneshot mode
- Analyzed synchronization between a callback registered after a socket returned `EAGAIN` and the existing readiness callback, addressing races between state observation and event notification

Projects

TrainsOS C, Rust, ARMv8, IPC

- Built a real-time kernel in C for ARMv8 with UART and timer interrupts, context switching, message passing, and MMU-based memory management
- Developed a real-time train-control application that estimated train positions and velocities, planned collision-free routes, and controlled train speeds and track switches
- Extended the kernel to load statically linked `no_std` Rust binaries and developed typed `MessageBox` IPC macros using scoped unsafe code to process variable-length messages in preallocated buffers
- Implemented a Bluetooth BLE GATT server in `no_std` Rust for wireless access to runtime telemetry and logs

Vision-Based Form Processing System Elixir, PyTorch, Qwen2.5-VL

- Built an end-to-end vision pipeline that processed photos of individuals holding paper forms, isolated portrait and document regions, and converted document content into structured records using a vision-language model
- Developed a Phoenix LiveView interface for batch uploads, human review and correction, image management, and document generation, backed by a Python inference service and S3-compatible object storage

Technical Skills

Languages: Elixir, C, Rust, C++, Python, SQL

Technologies: Linux, GraphQL Federation, Kafka, gRPC, Phoenix, PyTorch, S3