

# Andrew Liu

✉ andrew.liu2@uwaterloo.ca | 🌐 andrewliu.tech | 🔗 linkedin.com/in/andrewliu00 | 🐙 github.com/andrewliu00

## Education

---

**University of Waterloo**, BSc in Computer Engineering – Waterloo, ON, Canada Sept 2018 – May 2024

- **Coursework:** Systems Programming and Concurrency, Real-Time Operating Systems, Software Design and Architectures, Programming for Performance, Database Networks, Computer Networking

## Experience

---

**Software Engineer**, Stealth Startup – Toronto, ON, Canada Jan 2025 – May 2025

- Built a full-stack control platform with **Next.js**, **Node.js (Express)**, and **PostgreSQL**, enabling remote management of stereo video capture sessions with live telemetry and file organization features
- Developed a **Python-based** processing pipeline to convert **100TB+** of stored RAW10 footage into YUV420P10 and MP4 using **OpenCV**, **CUDA**, and **FFmpeg**, achieving **60 FPS** encode throughput
- Implemented a **PID-based** autoexposure system and integrated dynamic camera controls (gain, exposure, white balance) via **Python APIs** for robust capture under varying lighting
- Maintained video pipeline on **Ubuntu Linux**, handling low-level file I/O, systemd services, and hardware interfacing

**Software Engineer**, theScore – Toronto, ON, Canada Sept 2023 – Dec 2023

- Improved business-critical ingestion systems of **5** real-time sports data providers using **Elixir/Phoenix** and **Rails**, ensuring **99.9%** uptime and seamless data updates for **4.1 million** monthly active users on the premier sports media platform
- Streamlined data retrieval and improved developer debugging time by **15%** by designing and implementing a file-archiving pipeline for incoming payloads from data providers using **PostgreSQL** and **AWS S3**
- Integrated a sports data source via a **persistent SSE connection** in **Elixir**, enabling ongoing replay of historical NFL games to test the health of uninterrupted **real-time data ingestion** during core service updates
- Participated in an organization-wide migration of legacy ingestion to a parallelized cronjob system using **PostgreSQL** job queues, contributing to achieving **99%+** system availability and a **35%** increase in job throughput
- Deployed data monitoring solutions using **Datadog** and **Splunk**, improving analytical insights by creating custom dashboards to oversee key performance indicators, data ingestion metrics, and service-level agreements

**Backend Developer**, Drop – Toronto, ON, Canada May 2022 – Aug 2022

- Reduced fraud rates by **60%** by designing and implementing a back-end service in **Rails** and **Postgres**, featuring a purchase limit check algorithm to protect the integrity of the loyalty-point system
- Enhanced internal admin portal by developing tools with **ActiveAdmin** and **Rails**, streamlining user management for marketing and customer service teams, boosting efficiency and customer engagement
- Increased user engagement by **80%** by enhancing existing models with critical fields to support new redemption items
- Implemented test suites with **RSpec** and type checking with **Sorbet** to ensure code reliability

**Full Stack Developer**, XE.com – Newmarket, ON, Canada Sept 2021 – Dec 2021

- Enhanced performance and accessibility for **15+ million** users/month by optimizing asset sizes, implementing lazy loading, reducing page load times by over **30%** and increasing **Core Web Vitals** scores
- Revitalized IBAN pages by replacing legacy code, eliminating tech debt, and implementing **React** with **Next.js**, leveraging **Static Site Generation** to craft pre-rendered country pages for enhanced **SEO optimization**
- Developed reusable **React** components to unify and modernize layouts across 50+ XE marketing pages
- Reduced Cumulative Layout Shift (**CLS**) by **20%**, increasing visual stability across pages to improve viewing experience

**Full Stack Developer**, Remitbee – Mississauga, ON, Canada Jan 2021 – Apr 2021

- Built and optimized client portal in **React** and **Redux**, driving a **20%** boost in repeat transactions per user
- Optimized queries in a **MySQL** database to decrease core platform load times by **15%**
- Built dynamic page templates in **React** and **Strapi CMS** by designing new components and refactoring legacy code.
- Updated backend transaction flows using **Next.js** and **GraphQL** in preparation for new user group

## Skills

---

**Languages:** JavaScript, Ruby, Elixir, Python, Java, TypeScript, HTML/CSS, SQL

**Technologies:** React, Redux, Rails, Phoenix Liveview, Docker, AWS, Next.js, Express, Node.js, Linux, Git, Redis, Sidekiq, PostgreSQL GraphQL