

# Andrew Zheng

[andrew.zheng1@uwaterloo.ca](mailto:andrew.zheng1@uwaterloo.ca) | [in/andrewzheng2007](https://in.andrewzheng2007) | [github.com/awzheng](https://github.com/awzheng) | [awzheng.me](https://awzheng.me) | [Portfolio](#)

## EDUCATION

**University of Waterloo** | *Bachelor of Applied Science in Computer Engineering*

- GPA: 3.9

Sep 2025 – Apr 2030

## SKILLS

**Languages:** C, C++, Python, Golang, Verilog, Java, HTML, CSS, JavaScript, TypeScript, Bash

**Libraries/Frameworks:** FastAPI, React, MongoDB, Streamlit, NoSQL, REST API, Tailwind CSS, Next.js

**DevOps/Tools:** AWS (Lambda, S3, IAM), Docker, Git, Jupyter, QEMU, Virtualization, VMWare Fusion, Linux

**Hardware Design:** Altium Designer, KiCad, AutoCAD, Autodesk Fusion, SolidWorks, Bambu Studio

## EXPERIENCE

**Embedded Systems Designer** | Waterloo Rocketry | Waterloo, ON

Sep 2025 – Present

- **RocketCAN Remote Arming System** ([GitHub](#)): Engineered a safety-critical **KiCad** PCB featuring dual-redundant power inputs and a discrete P-MOS high-side switching topology, implementing fail-safe logic to guarantee recovery system arming during critical MCU power-loss scenarios
- **Remote Arming Firmware** ([GitHub](#)): Implemented and debugged low-level **C** firmware for **PIC18** to aggregate UART altimeter telemetry onto the **CAN** bus, integrating ADC health monitoring via INA180 current sensing and pyro continuity validation to ensure launch readiness

**Hardware Designer** | UW Biomechatronics Design Team | Waterloo, ON

Sep 2025 – Present

- **EMG Fabric Bionic Arm PCB** ([GitHub](#)): Spearheaded the migration of the Bionic Arm PCB from **KiCad** to **Altium Designer**, engineering an **ESP32**-based control system with **USB-C** connectivity
- Validated electro-mechanical integration by collaborating on design constraints in **SolidWorks** for seamless fit

**Case Competition Coach** | Self-employed

Jun 2024 – Present

- Scaled self-founded coaching business from contract work to fully independent, achieving **5-figure revenue**
- Formulated training materials for 150+ secondary school students across 10+ schools, leading to 50+ international qualifiers, 30+ finalists, and 33% (4/12) of Team Ontario's **1st-place** finishes at ICDC 2024

## PROJECTS

**SageWall** | ML Cloud Security System ([GitHub](#)) | *AWS, Python, Streamlit, Machine Learning*

Dec 2025

- Architected a serverless Intrusion Detection System (IDS) deployed as a **SageMaker** endpoint using an **XGBoost** classifier, analyzing network traffic with **99.9%** classification accuracy and **<100ms** inference latency
- Engineered and automated the ETL pipeline using Lambda to transform raw network logs in S3, processing **125,000+** records from the NSL-KDD dataset and provisioning infrastructure via AWS SDK (Boto3)
- Deployed a Streamlit web dashboard for real-time system observability, secured with AWS IAM policies, using CloudWatch to monitor system health and triggering SNS security threat alerts via email

**CrawlStars** | Concurrent Search Engine ([GitHub](#)) | *Golang, MongoDB, REST API, Concurrency*

Dec 2025

- Engineered concurrent web crawler with custom HTTP client using scalable producer-consumer architecture with parallel goroutines and buffered channels, processing **2000+** pages/minute with **<15MB** memory usage
- Optimized high-concurrency pure **Golang** backend with thread-safe deduplication using sync.Map and atomic operations, enabling O(1) (constant-time) lookup for **50,000+** URLs and filtering **70%** of redundant crawls
- Developed a relevance ranking algorithm using MongoDB Atlas Search integration with aggregation pipelines and fuzzy query matching to map SEO relevance metrics to a 5-star rating system

**Mangaroo** | PDF-to-Manga AI Illustrator ([GitHub](#)) | *FastAPI, Python, REST API, Generative AI*

Nov 2025

- Architected an asynchronous REST API orchestrating an agentic pipeline (**Gemini 1.5 Pro + Imagen 3**) using **FastAPI**, handling **1000+** concurrent sessions to process **250+** page novels and **50MB** input
- Engineered Story Bible context system with structured JSON prompting and session-scoped caching, reducing token usage by **92%** (1.4M to 100K per 100 pages) while maintaining O(1) memory complexity vs. O(n) alternatives

## AWARDS

**1st Place World Champion (Marketing – Product Management)** | *DECA ICDC @ Los Angeles*

Apr 2024

**7x National Honour Roll** | *University of Waterloo CEMC Mathematics Contests*

2020 – 2024