Edward Song

613-668-1312 | ey2song@uwaterloo.ca | https://github.com/edwarddsongg | LinkedIn

Education

University of Waterloo - Bachelor of Computer Science (Co-op Program) *2021-2025*

Skills

Languages	C++, C, Python, Java, JavaScript, Typescript, HTML, CSS, LaTeX, C#, Racket, R
Technologies	AWS, React, Redux, MongoDB, PostgreSQL, MySQL, Express, Node.js, Flask, LLMs, Git, Linux, NumPy, Jira, Pandas

Work Experience

ConnectX
May 2024 – Present

BACKEND DEVELOPER / AI RESEARCHER *Python, TypeScript*

- Developed an automation tool using **Python**, **TypeScript** and **Node.js** to generate and maintain Swagger API documentation for controllers and middleware, using **GitHub Actions** to optimize the development workflow
- Engineered an **AdonisJS** based authentication middleware with customizable guards to verify **JSON Web Tokens**, preventing unauthorized API access, and maintaining data integrity
- Optimized ETL workflow using **AWS Glue** to extract and transform large datasets in **AWS S3**, and load structured data into a **MySQL database**, enabling data integrity and seamless data flow for over **100GB** of data

Trend Micro
Sept. 2023 – Dec. 2023

BACKEND DEVELOPER *Java, TypeScript*

- Developed a **Java** based **REST** API microservice, responsible for parsing, managing, and serializing sensitive user database server settings in **PostgreSQL**, while enforcing API security with OAuth tokens and rate limiting
- Designed an **AWS** service to streamline real-time monitoring of API log requests using **Amazon CloudWatch** and **API Gateway**, achieving a response time under 700ms, and significantly improving API troubleshooting and debuggability
- Implemented load balancing across multiple **EC2 instances** and utilized asynchronous task queues via **AWS Lambda** to achieve a **100ms reduction in notification delivery latency**
- Increased test coverage by **57%** by building a comprehensive end-to-end test suite for a **DynamoDB** notification service pipeline using **Typescript** to automate test scenarios and **Postman** to test API response times

Cisco Systems
Jan. 2023 - April. 2023

SOFTWARE ENGINEER *C++, Python*

- Developed internal tools in **C++** and **Python**, optimizing hardware efficiency for IOS-XR software on Cisco 8000 routers by implementing traffic shaping, traffic classification, and route caching for packet data
- Engineered a **packet monitoring algorithm** in **C++** to track incoming ingress and egress packets to Netflow, improving **network monitoring throughput and latency**, saving developers over 10 hours of log debugging
- Built automated testing infrastructure in **Python** to write unit tests for ERSPAN and SPAN, adding support for regression testing

Ford Motor Company
May. 2022 - Aug. 2022

SOFTWARE DEVELOPER *C++, C*

- Built an IT automation tool on **GitHub** to label pull requests depending on passing and failing unit tests, providing developers a quick way to assess the status of their pull requests
- Designed a code coverage monitor for a vehicle codebase with historical telemetry and coverage visualization, improving code reliability and robustness

Personal Projects

SnapChef – Full-stack application to generate personalized recipes based on real-time video *github/SnapChef*

- Built a web application with **React, Redux, Node.js**, and **Express.js** to create recipes using the **ChatGPT** API from detected food
- Trained a deep learning object detection model using **YOLOv8** and **PyTorch** to identify 30+ food items with an 85% test accuracy
- Implemented robust user authentication using **Bcrypt**, enabling users to create, save, and delete recipes stored within a **MongoDB** database using a **Node** and **Flask** backend

Global Power Rankings – REST API that analyzes League of Legends team performances *github/LeagueHack*

- Deployed a **REST API** using **AWS Gateway** and **Lambda** to display international and regional team rankings
- Performed data cleaning, and transformation using **Pandas** to train and test an **XGBoost** model on multiple features
- Implemented custom ELO ranking algorithms to analyze team match history performance using **Python**