

Toni Zeng

tyzeng@uwaterloo.ca | /in/tonizeng | github.com/tonizeng | tonizeng.ca

Education

University of Waterloo

September 2022 – April 2027

Bachelor of Applied Science in Mechatronics Engineering (Coop)

Relevant Courses: Data Structures & Algorithms, Real-Time Operating Systems, Digital Computation

Skills

Languages: Java, Python, C++, C#, SQL, JavaScript, HTML/CSS

Tools/Technologies: Git, Jenkins, Docker, Datadog, Logstash, Artifactory, Azure DevOps, Power BI

Frameworks/Libraries: .NET, React, scikit-learn, PyTorch, OpenCV, Pandas, NumPy

Databases/OS: MySQL, SQLite, ChromaDB, UNIX, Mainframe, z/OS

Work Experience

DevOps Developer Intern | Kinaxis

September 2024 – December 2024

- Developed enterprise-scale **C#** applications using **.NET**, applying **OOP** principles to design robust solutions.
- Optimized logging processes for service logs transmitted to **Datadog** by implementing structured logging and refining **Logstash** parsing, reducing log processing time by **25%**.
- Managed build artifacts in **Artifactory** to ensure reliable cross-environment testing of infrastructure service tools.
- Created **Confluence** documentation on code infrastructure, reducing onboarding time from **2 weeks to 3 days**.

Machine Learning Developer | Social and Intelligent Robotics Research Lab

May 2024 – August 2024

- Conducted user studies with **60** participants to evaluate computer vision models for an autonomous robotic arm.
- Developed and trained a **Mask R-CNN model** in **Python** using **OpenCV** and **PyTorch** on **50,000+** images, achieving **90%** accuracy in anonymizing volunteer participant images.
- Optimized image processing with **Mask R-CNN** for accurate object detection and segmentation, extracting critical features like boundaries and labels, enhancing image clarity and segmentation accuracy by **30%**.

DevOps Engineering Intern | Royal Bank of Canada

January 2024 – April 2024

- Developed production code validation pipelines using **Java**, **Jenkins**, and **Docker**, reducing code errors by **91%**.
- Led the identification and resolution of bottlenecks in deployment pipelines, decreasing deployment time by **35%**.
- Deployed **DevOps infrastructure** with **UNIX** on **30+** Mac Minis, enabling department-wide mobile deployments.
- Implemented developer portal using **React** and **JavaScript** with **API** functionality, providing **90+** employees with a centralized hub for resources, streamlining work efficiency and accessibility.
- Created **GitHub** tools using **API** integration with **Jenkins** to enforce branch protection and enhance security.

Data Automation Engineering Intern | Metrolinx

May 2023 – August 2023

- Led the implementation of **Python** scripts for automated file validation, enhancing **data ETL** efficiency by **98%**.
- Preprocessed **10,000** transit records with **Python**, **Pandas**, **NumPy** for trend analysis and performance metrics.
- Developed optimized **SQL queries** to extract and transform data for **10+** customer-facing **Power BI** dashboards.
- Implemented **Azure DevOps** integration for seamless **GitHub** code deployment with **CI/CD pipelines**.
- Created document management apps utilizing **MySQL** for data storage and **Power Apps** for frontend interfaces.

Projects

spotSpot | Ctrl+Hack+Del 2024

Python, PyTorch, Streamlit, scikit-learn

- Developed an acne classification model using **PyTorch's ShuffleNet CNN**, achieving **76% identification accuracy** across various acne types by optimizing model weights for high performance on a small dataset.
- Labelled and augmented **850+** photos to improve model robustness and variability for real-world image inputs.
- Built an interactive **Streamlit** frontend with live camera functionality for real-time acne diagnosis and guidance.
- Awarded **3rd place out of 62 teams** and **235 participants**, recognized for innovative design and technical execution.

JiraBot | Kinaxis Hackathon 2024

OpenAI, Python, JavaScript, Node.js, ChromaDB, Docker

- Developed a **Teams** chatbot to assist developers in real-time ticket resolution by searching **Jira's** ticket database and suggesting relevant tickets, using the **Jira REST API** to fetch data into **ChromaDB** for efficient semantic search.
- Fine-tuned an **OpenAI LLM** model to continuously learn from developer preferences, enhancing user experience.
- Built the frontend in **Node.js/JavaScript**, and the backend in **Python**, using **Docker** for environment consistency.