# Alex Zhang

778-886-8356 | alexjm.zhang@mail.utoronto.ca | linkedin.com/in/alexzjm | github.com/alexzjm | alexzjm.github.io

## Education

**Bachelor of Applied Science in Computer Engineering** Sep 2023 – May 2028
*University of Toronto* *Toronto, ON*

- Intended Minors: **AI Engineering & Engineering Business**
- 3rd Year Student: **3.88** CPGA, **88.0%** Cumulative Average, **Dean's Honour List** (2023 Fall - 2025 Winter)
- Relevant Courses: Applied Fundamentals of Deep Learning (**Python**, **PyTorch**), Software Design & Communcation (**C++, Git**), Computer Fundamentals (**C**), Linear Algebra (**MATLAB**)
- Upcoming Courses: Probability & Applications, Algorithms & Data Structures, Introduction to Databases (**SQL**), Probabilistic Reasoning

## Skills

**ML Libraries**: PyTorch, NumPy, Pandas, Scikit-learn, Jupyter Notebooks
**Languages**: C, C++, Python, Java, JavaScript, TypeScript, HTML, CSS
**Frameworks**: React, Next.js, Tailwind CSS, Bootstrap, Node.js, Express.js, Flask, FastAPI, Django
**Developer Tools**: Git, Docker, AWS, VS Code, Cursor, Vercel, Supabase, Figma
**APIs & Services**: OpenAI API, Google API, Supabase, Vercel
**Professional Skills**: Leadership, Communication, Initiative, Independent Self-Direction, Critical Thinking, Adaptability

## Experience

**AI/Automation Engineer Internship** May 2025 – Aug 2025
*JuniorKids* *Remote*

- Directed the development of a LangGraph-based **AI agent** powered by **Qwen3**, guiding implementation to automate lead sourcing and reduce manual prospecting time.
- Collaborated directly with the **CEO** to define requirements and led two interns, coordinating tasks and providing feedback to align technical execution with business goals.
- Built modular **Python scrapers** for ecommerce platforms, enabling the agent to autonomously source and qualify sales leads.
- Leveraged Gmail, Google Calendar, and Google Sheets through **Google APIs** to build an integrated automation pipeline, unifying outreach, scheduling, and **CRM** tracking into a single lightweight system.
- Converted the AI agent into a **full-stack web application** using **Next.js** and **FastAPI**, exposing backend workflows through **REST APIs** and building a simple interface to make automation features production-ready.

**CTO & Technical Co-Founder** Mar 2025 – Aug 2025
*TrainerIQ - DMZ Basecamp* *Toronto, ON*

- Directed the end-to-end technical launch of TrainerIQ's **MVP**, collaborating with an external development team to deliver the first production release.
- Rapidly prototyped new features using a lightweight stack (**React**, **Express.js**, **Supabase**), leveraging AI-powered coding environments to mock up functionality quickly while maintaining clean structure, clear commits, and reliable performance.
- Managed production infrastructure on **AWS**, deploying on **EC2** and handling **domain/DNS configuration** to ensure a stable, scalable foundation for growth.
- Designed TrainerIQ's **logo**, **color scheme**, and **marketing website** by applying **UI/UX design** and branding principles learned from DMZ workshops and EIRs, establishing the company's initial **brand identity** and digital presence.

**Software Engineer Internship** Jan 2025 – Apr 2025
*Savi Finance* *Toronto, ON*

- Engineered an **AI-powered PDF parsing pipeline** to extend upload support beyond CSV files, broadening customer options and enhancing overall user experience.
- Developed **middleware** for backend **API input validation** by enforcing file type, size, and format checks, safeguarding **data integrity** and blocking invalid or malicious uploads.
- Refined front-end UI with **TypeScript** by standardizing header capitalization and resolving layout inconsistencies, resulting in a polished, user-friendly interface.

- Validated pipeline functionality and edge cases by writing **unit and integration tests** in **Jest**, ensuring reliability and smooth feature rollouts.
- Ensured traceable progress during rapid development by following **Git workflows** with **feature branches**, **pull requests**, and clear commits.

## Projects

**Diabetic Retinopathy Detection** | *Python, PyTorch, Scikit-learn, NumPy, Matplotlib*
*APS360 Applied Fundamentals of Deep Learning*

- **Fine-tuned** a pretrained **VGG-16 CNN** on a competition dataset, achieving **F1**, **precision**, and **recall** scores that surpassed the benchmark by **10%**.
- Conducted **data preprocessing** and analysis using **NumPy**, applying **augmentation techniques** and **transfer learning** with fully connected layers to improve model generalization.
- **Led a team of 4**, dividing tasks across **data pipeline**, **model design**, and **evaluation**, while hosting **code reviews** and progress syncs.

**AI Spam Detector for text** | *Natural Language Processing (NLP), PyTorch*
*APS360 Applied Fundamentals of Deep Learning*

- Trained **character-level RNN models** (**vanilla**, **LSTM**, **GRU**) on the UCI SMS Spam dataset (5k+ texts, 747 spam), achieving **98% test accuracy**.
- Implemented custom **tokenization** with **unknown-character handling** and **length-batched sequences** to reduce padding and improve efficiency.
- Addressed **dataset imbalance** by **oversampling** spam texts, and validated model performance with separate spam/non-spam accuracy to ensure robustness across classes.

**SkinGuard AI** | *Python, Flask, PyTorch, NumPy, Matplotlib*
*Personal Project*

- Fine-tuned a pretrained **AlexNet CNN** on the HAM10000 skin lesion dataset using **transfer learning**, achieving **80% accuracy** on held-out test data.
- Applied **data augmentation**, tuned **hyperparameters**, and used **train/validation/test splits** along with **SGD optimizer + BCEWithLogitsLoss** to reduce overfitting.
- Designed a prototype **full-stack web app** using **Flask** for **image upload and inference**, integrating preprocessing (**resize**, **normalization**) with the trained model in a co-founded startup prototype.

**CityScope** | *C++, OpenStreetMap API, EZGL*
*ECE297 Software Design & Communication*

- Developed an interactive **GIS application** in **C++** with **OpenStreetMap data** and **EZGL rendering**, enabling users to **zoom**, **pan**, and **search points of interest**.
- Actualized **pathfinding** by implementing **A*** and **Dijkstra's algorithms** optimized for large map datasets, delivering the most efficient path between any two destinations in under a second.
- Optimized **multi-destination path planning** for tourism routes by applying multithreaded **metaheuristic approaches** (**2-opt**, **simulated annealing**, **hill climbing**) to the **Traveling Salesman Problem**, computing high-quality solutions in under 5 seconds.
- Enhanced the map interface with contextual insights by integrating a **libcurl-based HTTP module** to fetch and parse **real-time weather data**.

**React CSV Toolkit** | *React, Vite, JavaScript, Tailwind CSS, Git*
*Personal Project*

- Developed a **React**-based **CSV toolkit** with **file upload**, **cell editing**, and **table manipulation**, allowing users to interact with and modify data directly in the browser.
- Implemented **real-time state updates** with **React hooks** and **context**, along with **dynamic sorting** and **row/column operations**, delivering a fast, intuitive, and scalable frontend experience.
- Applied responsive styling using **Tailwind CSS**, ensuring consistent design and usability across different devices and screen sizes.
- Deployed the app on **Vercel** with GitHub integration, enabling automatic updates on every commit and providing a **live demo** for easy access.