Justin Bax

438.763.6066 | jbax@uwaterloo.ca | github.com/justinbax | linkedin.com/in/justin-bax

Skills

Programming Languages: C, C++, Python, TypeScript, Java, HTML/CSS, 6502 Assembly
Tech/Tools: AWS, React, Next.js, PostgreSQL, MongoDB, Linux, Docker, GitHub Actions, CI/CD

Education

University of Waterloo September 2024 — May 2029
Bachelor of Software Engineering (expected) *Waterloo, ON*

- **Academics**: 4.0 GPA, 96% academic average

Professional Experience

Software Engineer Intern May 2025 — Present
NationGraph *Toronto, ON*

- Built an autonomous scraping pipeline for **880,000** unique websites, resulting in **3.29TB/day** of scraped public sector procurement data using AI agents and Python
- Created an automatic maintenance system to fix fauly agentic scrapers, **doubling** the amount of client-facing data using **PostgreSQL triggers** and defensive programming
- Improved the speed of a video clip search from **150s to 16s** by using vector embeddings and a term-frequency filter
- Created a containerization framework for web scrapers, fixing **OOM** errors from **8** sources in production with Python
- Improved a data mining system's accuracy from **40% to 81%** by building AI-controlled web crawling logic

Software Engineer Intern June 2024 — September 2024
Tail'ed *Montreal, QC*

- Built and deployed a job/candidate ranking AI to AWS, leading to **30%** lower infrastructure costs than the previous iteration, using Pinecone **vector databases**, **Python**, and Cohere
- Initiated the automation of the **CI/CD** workflow (auto-build, test & deploy) with **AWS** tools and GitHub Actions
- Optimized the résumé leaderboard **cache** system to reduce database fetches by 60% using **Next.js** and **TypeScript**

Projects

NESRev — *C, Assembly, OpenGL* justinbax/nesrev

- Solo development of a Nintendo emulator achieving industry-level cycle accuracy. Features step-by-step execution, debugging tools, ROM file creation from assembly source code and correct graphics and audio pipeline
- Built using a custom pixel rendering engine in **OpenGL** and **C**

MollyDB — *Rust* fletcherdares/mollydb

- Built a relational **database engine** and SQL interpreter optimized to compete with in-memory cache stores in **Rust**
- Working towards full compatibility with SQLite using differential testing and random workload generation

Research Experience

AI Research Intern September 2024 — May 2025
Julie Plante Computer Science Laboratory *Waterloo, ON*

- Built and benchmarked 9 modern implementations of priority queues used in AI algorithms using **C** and **Python**
- Implemented **Test-Driven Development** methods to ensure consistent operation between all versions using **gtest**
- Selected out of 2000 students to receive a 5000$ government award for college-level research

Co-Researcher, Lead Organizer September 2023 — June 2024
FLOSS (Open-Source) Club *Montreal, Quebec*

- Created software to automate 63 statistical tests, leading to 9 suggestions to the Debian team, using **Python/NumPy**
- Co-researcher in a statistical study on Debian usability, resulting in a talk at a worldwide open-source conference

Additional Information

Activities: 4x hackathon winner, jazz ensemble leader, annual math contests, jiu-jitsu, chess, game theory