

JUSTIN KWINECKI

Software Engineering Undergrad

(905) 341-6497 | kwineckj@mcmaster.ca | **Linkedin:** [Justin Kwinecki](#)
GitHub: <https://github.com/jjustin-k> | **Portfolio:** <https://jjustin-k.github.io/>

EDUCATION

McMaster University, Hamilton, ON

Sept. 2022 - April 2026

Major: Bachelor of Software Engineering (in-major GPA: 12/12 or 4.0)

Relevant Coursework: Object Oriented Programming, Software Engineering Practice and Experience, Data Structures and Algorithms, Software Development 1, Discrete Mathematics 1 and 2.

Awards and Achievements: Faculty of Engineering Entrance Scholarship, Dean's List, Ontario Scholar.

EXPERIENCE/EXTRACURRICULARS

DeltaHacks 10, Hackathon Competitor - Lead Developer

Jan. 2024 - Jan. 2024

- Created a gesture recognition software program that translated ASL gestures into characters.
- Utilized libraries such as **Tensorflow**, **NumPy**, and **MediaPipe** to train an **AI model**, and **OpenCv** for computer vision.
- Led a team of three to complete the project within a 36 hour time limit. Trained the AI model using **Ubuntu** due to conflicting library dependencies.

Mac AI Society - Member

Sept. 2023 - Present

- Attended a workshop in which I created a program that predicted the odds of having kyphosis at different ages.
- Utilized **Python** libraries such as **Matplotlib**, **NumPy**, **Scikit-Learn** and **Pandas**.

Rockway Vineyards - Back-Shop Worker

May 2021 - Aug. 2021

- Managed and maintained the back shop and was responsible for distributing, receiving, and maintaining golf carts.
- Adapted to changing responsibilities due to COVID-19 protocols and mandates.

TECHNICAL SKILLS

- **Languages:** Java, Python, C, BASH, HTML, CSS, JavaScript, UML, XML.
- **Technologies:** Git, GitHub, Ubuntu, Maven, TensorFlow, Pandas, Scikit-Learn, OpenCV, Bootstrap.
- **Operating Systems:** Windows and Linux.

PROJECTS

Maze Runner

- Created a program that can solve any maze by implementing the Right-Hand Algorithm.
- Designed a program following SOLID principles as well as practicing encapsulation.
- With shielding, new algorithms can be implemented to solve mazes without changing the base code.
- Program was written in **Java** and compiled with **Maven**, **Git** was used for version control, **XML** was used to add dependencies such as Apache CLI.

Facial-Recognition Software

- Built a program that can identify known people using **Python**.
- Known faces are kept in a folder, with the ability to add new faces.
- Implemented **OpenCV** for real-time images and computer vision.

Password-Resilience Analyzer

- Created a program that measures the strength of a given password and can remember the strength of the password previously entered.
- Program was built using **C**, and was built as an introductory to cyber-security project.