# Josh Blatt

**3A MECHATRONICS ENGINEERING**

## Personal Profile

I am a US/Canadian citizen studying Mechatronics Engineering at the University of Waterloo with strong interests in embedded systems, robotics and AI. I am extremely excited to apply my skills and continue to improve upon them.

## Skills

**Languages**

- C/C++
- Java
- Python
- MATLAB

**Development Tools**

- Git
- Vi / Vim
- CMake
- VS Code
- Make
- Visual Studio
- Linux
- CLion
- Eclipse
- PyCharm

**CAD Software**

- SolidWorks
- AutoCAD

**Team Skills / Methodologies**

- Agile
- Scrum
- JIRA
- TDD

## Contact Me

**Mobile**: +1 416-577-7559
**Website**: www.joshblatt.ca
**Email**: jablatt@uwaterloo.ca
**LinkedIn**: www.linkedin.com/in/josh-blatt/
**GitHub**: www.github.com/joshblatt

## Coursework

- Microprocessor Systems and Interfacing (C)
- Real-Time Operating Systems (C)
- Actuators and Power Electronics
- Sensors and Instrumentation
- Microprocessors & Digital Logic (Assembly)
- Data Structures and Algorithms (C++)
- Digital Computation (C++)

## Education

**University of Waterloo**
BASc in Honours Mechatronics Engineering
September 2018 - April 2023

## Interests

- Electronics
- Guitar
- Rock Climbing
- Chess
- Snowboarding
- Cutting Hair
- Rubik's Cubes
- Video Games
- Music
- TV Shows
- Book Clubs
- Hiking

## Work Experience

WINDOWS CORE SOFTWARE DEVELOPER

AMD | September 2020 - December 2020

- Refactored Radeon kernel-debug tool's code repository to make files easier to find and amended the CMake build architecture accordingly
- Implemented new commands and unit tests for the Radeon kernel-debug tool using C++ to make driver debugging easier for developers
- Added Radeon kernel-debug tool support to new areas of driver code using C++, allowing developers to easily extract new information from the live ASIC
- Wrote a C++ class to export Radeon telemetry data to MongoDB, rather than storing the parsed data in a CSV file as it was previously
- Programmed Python script to backup SQLite3 database files every 2 weeks to ensure telemetry data isn't lost when raw dumps are deleted

EMBEDDED DEVELOPER

ecobee | January 2020 - April 2020

- Improved firmware for thermostat temperature slider to allow for a greater degree of parameter customization, including slider sensitivity, scroll speed, and inactivity distance
- Designed and facilitated user tests with new thermostat firmware to determine the most preferred combination of temperature slider parameters
- Developed solution to customer issues with slider sensitivity by designing and prototyping buttons to be used with the thermostat slider to allow for more precision when adjusting the temperature
- Programmed new multi-speed fan QA PCB which was added to existing thermostat HVAC simulators, enabling automated testing of the hardware
- Developed numerous UI screens for the new thermostat multi-speed fan installation flow using C++ and created automated tests with Python to ensure its functionality

EMBEDDED TEST DEVELOPER

ecobee | May 2019 - August 2019

- Independently designed and programmed a Sensor Simulator using C++ to test the functionality of the SmartSensor and the SmartSensor for Doors and Windows, enabling the automation of 50+ test cases which previously took one week of manual testing
- Programmed Python script to parse and analyze sensor packet data to make product decisions, including how often and how far apart retry packets should be sent
- Automated 15+ thermostat integration tests using Python
- Designed integration test plan to verify functionality of production units of the SmartSensor, which was used for all customer-facing releases
- Assembled, soldered, and tested the hardware for HVAC simulator circuit boards used for QA

## Projects

Mome (WIP) | December 2020 - Present | Repository

- Mome is a new C-like low-level programming language with easier syntax being developed independently using C
- Similarly to C, it will have features such as memory management, pointers, structs, and unions, with some added features including cleaner typedef struct syntax, automatic variable type assignment if not specified, and no semi-colon requirement
- Wrote the compiler's lexer, which breaks down the code being compiled into recognizable tokens to be interpreted by the parser, and implemented unit tests
- Created stretchy-buffer data structure (essentially a vector for C) using preprocessor macros and wrote unit tests to ensure functionality
- Currently writing compiler's parser and unit tests associated with it