I've been working for the last 3.5 years as a C++ developer for embedded systems, and I have acquired very wide range of experience of writing robust and readable code. I started working with STM32F series MCUs, including "bare metal" code, as well as RTOS based systems. I then progressed into projects on linux embedded systems based on beagle bone or rapsberry pi, writing programs in pure C++ 17 and cmake as well as those made with Qt 5.

I must say, that I am very much enjoyed working with linux code, and see future of embedded systems moving into it even for the smallest devices, as the cost of compute will go down, therefore I see that more and more systems will move away from bare metal MCUs into linux based devices.

As I mentioned before, I've been writing both pure c++ software for linux as well as within Qt framework. I would say that even though I do enjoy the vas array of methods that Qt frameworks provide, and the ease of building GUI, my favourite project is made in pure c++ with cmake, as it is meant to run on a headless raspberry pi like system. Said project is the custom computer vision library that I've been developing for the last year. Written entirely from scratch, from the first principles, the goal was to create a lightweight codebase that can be run cross platform, that would