We plan to create a realtime operating system (RTOS) using Rust. Rust is a low level programming language with some built in safety nets that prevent segfaults and thread safety. Real time operating systems are reliable in both security and performance in a given timeframe. The combination of the Rust and a RTOS leads to a robust and secure os. This os can be utilized by inherently vulnerable systems like IOT devices, vehicle systems, and other embedded systems. With the prevalence of these systems growing more secure solutions are needed.

The course most relevant to this project EECE4029 Operating Systems and Systems Programming. In this course we discussed the different layers of a computer system such as the kernel, the architecture privilege rings, and networking. Additionally we discussed concepts like paging and threading which are all relevant to creating a RTOS. A second course that be useful to this project is EECE2093C Software Engineering. This course introduced lifecycle models and requirement analysis which will be useful in the creation and execution of the project. Some of these concepts include user stories and black-box/smoke testing.

The only technical work related to this project comes from my co-op experience at the EPA. Rust shares a lot of similarities with C++ and while at the EPA I briefly worked on a C++ application. Some non technical skills built at my other co-ops which will be useful to this project is the software management system 'agile' and interpersonal skills. While I was at Kroger we used both kanban and scrum to plan our projects. Adapting these strategies to