Overall, we see Rust as a language fit for a larger project. Its robustness and capabilities with JSON parsing have impressed us, and although its GUI programming has not been trivial, it's workable as-is and has the potential to be much smoother with a newer version of Rust. The community has a focus on safety which shows in the robust error handling of many libraries. In addition, the compiler is incredibly developer-friendly, and often provides the exact fix needed to correct a given problem. The language, standard library, and third-party libraries had documentation available at a centralized website. Most of them provided not only method/type signatures but examples and guides for using them as well.

The main potential problem we see with using Rust is most modern GUI crates expect newer Rust compiler versions than the one currently used on the Khoury servers. In addition, the current cargo version on the Khoury servers seems to have a compilation bug when compiling many dependencies in parallel. We believe that the cause of this bug could be a result of the new concurrent compilation feature added in Rust v1.38 (Khoury servers run v1.39) and the way it interacts with the CPU allocation on the servers. As a temporary workaround, we limited the number of compilation threads to 4 in the Makefile for our programs. We plan to make a Piazza post about getting the Rust version updated, as well as to send an email to Khoury systems to see if they can help us out.

Despite these issues, we still believe Rust is fit for the project, and proved in assignment D that there is a GUI library setup that will work with the current resource set on the Khoury servers. We are excited to continue our exploration of this language and to build increasingly complex software.