Google

Summer of Code

# Proposal

## aris - Add a web user interface to GNU Aris

By
Saksham Attri

# Table Of Contents

# About Me

| | |
|---|---|
| Name | Saksham Attri |
| University | Birla Institute of Technology and Science Pilani, Hyderabad Campus |
| Major | Computer Science |
| Github | @sucksome |
| Email | sakshamattri3@gmail.com |
| Discord | sak#3446 |
| Phone | (+91) 7291998949 |
| Timezone | Indian Standard Time (UTC + 05:30) |

# Project Details

## Add a web user interface to GNU Aris

### Summary

The project aims at extending GNU Aris by adding a web user interface. It involves some design changes in the GUI and migrating the current GTK based UI to Qt. Consequently, a WebApp based on HTML, Javascript and WebAssembly will be developed.

### Benefits

Currently the GNU Aris user interface is based on GTK and does not support every platform and is buggy on a few others. Since most of the source code was written around ten years ago, there also exists some amount of deprecated code, most of which deals with GTK.

The current UI also warrants a redesign as a number of issues already exist:

1. #5 Virtual keyboard is not sending data to the goal window
2. #4 Crash if a virtual keyboard is used for a closed main window
3. #7 "New" windows generate new symbol menu

Redesigning a part of the UI and migrating to Qt can resolve issues above and other GTK related problems. It will also ease development of the WebApp as Qt has great Wasm support.
Currently, the GUI is built into the main application logic and debugging can be a challenge, thus, to improve debugging experience, readability and modularity, efforts will be made to separate the GUI skeletal from business logic as much as possible.

Secondly, installing GNU Aris can be a daunting task for many as all distros don't have a binary release with compiling and building from source being the only option for some.
GNU Aris is aimed at Academia, and many end users (students and teachers) would like to avoid the hassle, this can be achieved with a web-based interface. This web-based application would be able to run locally on the client side, thus it can also be distributed as an offline package.

## Deliverables

The project can be broadly divided into two parts; migrating from GTK to Qt and building a WebApp from the Qt GUI using Emscripten.

A more comprehensive list of deliverables follows:

1. Design changes
    a) Merge the rules window, virtual keyboard and main window into a single window.
    b) Add a panel in the main window to display the goals window.

   Finer aspects will be decided after more discussion with my mentor(s).
2. Develop a Qt based GUI
    a) Separate the GUI from application logic.
    b) Business logic and UI will interact through an API.
3. Develop a WebApp

a) Compile the C logic layer to Wasm and the UI to HTML and JavaScript with Emscripten.

4. Detailed Documentation and example proofs

## Plan

| Date | Task |
|---|---|
| Community Bonding Period | |
| May 4 - May 16 | Discuss finer design aspects and project outcomes with mentor(s) |
| May 17 - May 28 | Explore implementation procedures |
| Coding Begins | |
| May 29 - June 17 | Work on developing the Qt GUI skeletal |
| June 18 - July 10 | Work on interfacing the GUI with Application Logic |
| July 10 - July 14 | Midterm Evaluation |
| July 15 - Aug 6 | Testing the GUI and start work towards WebApp development from Qt GUI with Emscripten |
| Aug 7 - Aug 13 | Complete the WebApp development and start testing |
| Aug 13 - Aug 20 | Update older documentation and write some test proofs |
| Aug 21 - Aug 28 | Submit final code and work product |
| Aug 28 - Sep 4 | Final Phase Evaluation |

**Note :** I have final exams from May 6 to May 22 so I'll not be able to direct much attention towards the project. I will still be available on discord and email. After May 22 however, the project will be my main priority and I'm willing to work at least 40 hours a week.

### Communication

I will primarily contact my mentor(s) through discord and email. To keep my mentor(s) up to date with my progress, I plan on providing weekly reports. I will also try to keep in touch on a daily basis for a timely resolution of doubts and/or other issues that may arise.

# Qualification

> *Why did this project appeal to you?*

Last semester, I took a logic course which I really liked. As a part of the course, I had a two-part mini assignment involving:

1. A simple tool for generating parse trees of propositional formulas and their truth evaluation among other things. ( link to repo )
2. A simple tool for verifying whether a certain proof of a given sequent is valid or not. ( link to repo )

Since the assignment was quite limited in terms of the rules supported and the UI, I was planning on extending its functionalities.
Thus, while browsing through GNU's GSoC projects, I came across Aris and it resonated with me.

> *What will you do once the project is "finished"?*

I will try to extend the rules library as some important ones like disjunction elimination are missing. I will also try to add a Dark Mode to Aris.

> *Have you worked on any Free Software before?*

I did get a PR merged into Aris for fixing a bug. Apart from that, I have not.
Also, this is my first time applying to GSoC and this is the only project I'm applying for.

> *Of the skills that you will need to complete the project, which do you already have?*

I’m well versed with C programming and HTML. I’m learning more about Qt and WebAssembly.