**A Curious ‘Glance’ to Open Source**

**Project Description**

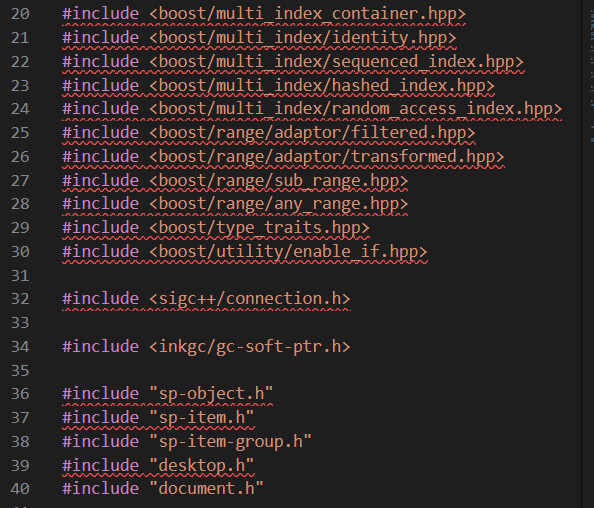
The project which attracts me most is the Refactoring. After viewing the idealist, there rise some thoughts about the idea 'Remove dependence on object-set. h in files in src/livarot directory' in little projects. And here are the details!

As .h files always include the declaration and definition of the functions, removing their dependence is a good way to move the functions right to the relative files in the src/livarot directory. To avoid probable errors, it is necessary to track where are they exactly.

**Implementation**

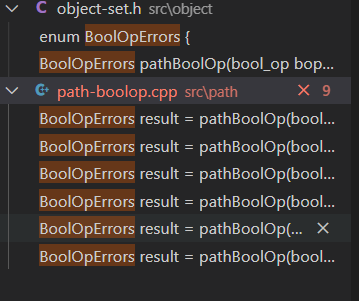
Firstly I’d like to talk about the refactoring work. Having built the Inkscape project on the computer and looked through, there are some ideas. And I look through the object-set. h and have some discoveries.

In this .h file there are plenty of errors. In my viewpoint these are rising because of the removed files. To make it clearer, it is necessary to find out if there are any dependencies and then remove them.



Then here comes the usable code. To sum up, There are some enum, class, and struct types defined in this .h file, which should be removed. In addition, some member functions are also defined in it. They need to be moved into the corresponding files which have called them.

As for this ‘BoolOpErrors’ enum, I find it called in these files:



What I should do is to define it in the specific .cpp file, which should solve the problem.

Besides, about my idea, if considering transplanting it to another platform, we may just keep the main drawing and saving functions, leaving the higher-leveled ones on the computer. That’s a lot of work! Though I think I can just handle few of it, I still wish Inkscape grow better!

**Development**

**5/4 Bonding**

Before the coding begins, I would use this period to contact a mentor, getting to know more about the tasks ahead. Also, I'd like to consult some questions about Inkscape, as knowing the application better is always helpful for doing the work!

**5/29 Coding Beginning**

By the time of the next stage(midterm), I plan to finish 70-80% of the total work.

In detail, there are about 20 functions and variable defined in this .h file and up to 70 types clarified in it. To reach the goal before midterm, that means 2~4 weeks of early stage working, so I need to make sure solving 7 problems each day. I belive it isnot too hard and I can make it go well.

**6/10-7/14 Midterm**

At this stage, the mentor has to submit the midterm evaluations according to the present work---hope I've achieved the goals!

**7/15-8/20 Coding**

Next 26 days I need to finish the remaining problems. If finally I haven’t reach the last goal by midterm, I suppose this stage is a good chance to solve it.

**8/21-8/28 Final Week**

After all the development, this week is the time to submit the final work product and the mentor evaluation. I will use this stage to perfect my coding work, maybe with the documents, and also submit my evaluation on time.

**About Me**

* Name: Du Jiali
* GitHub: <https://github.com/Yousone>

(In the repositories are the projects that I created with my friends.)

* Education:
  + 2021-Present: Southwest Jiaotong University, Automation Specialty (master)
* Email: [atress.du@gmail.com](mailto:atress.du@gmail.com)(frequently used)

[3011959096@qq.com](mailto:3011959096@qq.com)

* Tel: +86 19949800193

I wasn’t exposed to large open-source projects before, except for some small projects on GitHub, which I developed with my teammates. That means I'm new to open source with basic knowledge of C, C++, and python. In the past I’ve also got to know little about CSS and JS. And lately, I'm trying to learn about Linux(I guess I’m just too curious about these things, of which I regard as an important ability to apply GSOC).

I've learned that many excellent projects are open source, which means they are developed with many people's effort. The idea of team-based work attracts me hard: it is a marvelous way to use the power of the internet and team. In this situation, a large-scale project can be finished conveniently, at the same time getting the users and developers together. These all help bring out a nearly perfect product.

I'm interested in this year's GSOC and would like to know more about the project about Inkscape. Having known that it is a highly commended open-source design tool(in fact I've seen someone compare it to ai), I'd like to learn more about it. That's when I started to consider its GSOC project.

I have pulled the Inkscape project and looked through some of its code---the project looks so huge and cool, and the name list of contributors had somehow touched me! I realize that is the spirit of open source.

Thank you for your time in reading this. Hope it will be a funny summer!