Focus group research was done to help prepare the questionnaire and acquire basic requirements in users’ aspects.

Year 1, year2 and year 3’s computer science students were invited to focus group meetings respectively. Each focus group had six students. The purpose of the focus groups was to receive feedback for questionnaire and get inspiration for some new ideas about this software.

Participants background and brief meeting process and discussion results are as follows. For full Focus Group Report which include focus group process introduction and more detailed information, please refer to appendix[NUM].

**Participants Background**

There are three focus groups divided by participants’ grades (from year 1 to year 3). To maintain the objectivity of the outcome of the discussion, it is carefully ensured that participants in the same group were unfamiliar with each other and team member. ~~Year 3 group attend to know about algorithms’ correctness.~~

**Discussions**

**Year 1 group**

Due to year 1 participants’ lack of relevant background knowledge, the discussion result of this focus group is less than expected but there are still some valuable advice.

Pseudo code blocks

It is mentioned by participants that writing pseudo-code is a hard part for beginner, game such as dragging pseudo code blocks may help them to learn the idea of writing sorting algorithm.

Time complexity

Participants suggested that the software can provide a time complexity comparison of different types of sorting algorithm. This idea is presented in the requirements.

**Year 2 group**

Year 2 students’ discussion was out of our expectation. Many innovative and creative ideas were raised in this focus group.

**Fun and game likely VS functionality**

Participants showed a strong preference for functional software compare to interesting game likely one. They would like the software to have a single purpose, without those distracting things. This result helped the team decide requirements.

**Method of selecting modules**

Two methods of selecting modules were introduced to participants. The first method was that users cannot access the next sorting algorithm’s learning module unless they finished the one before. The other method was free choosing. It seems that participants would like to freely choose an algorithm module instead of “unlocking” the next module by finishing the current one. This result overturned team members’ initial idea and first method is abandoned.

**Displaying code**

Participants thought pseudo-code is enough for learning since it gives a logical idea of sorting algorithms. Besides, pseudo-code is more friendly to beginners. Source code can also be presented, they claimed, and for language, C, Java and Python are preferred.

**Progress bar**

After presenting and explaining our preliminary prototype, participants showed great interest in the progress bar we designed. They thought that a display of where they were in the program would be helpful. Besides, they suggested us to design history and reset function.

**Platform preference**

Following options were provided to participates: PC, mobile app, WeChat mini program and website. The result is that the PC was more popular than others. Participants also said that PC would be a more suitable platform for notes’ quick export function.

Besides, participants also responded positively to following functions:

1.Multiple language support.

2.Quick export of brief notes, i.e., a file includes introduction and important points of a sorting algorithm.

3.Show efficiency and time complexity of algorithms.

4.Apply breakpoints.

5.A user guide to give users a quick look at how to use this software was highly recognized by participants.

6.Exercises.

或许三个年级全部合并？重要的详细写，不重要的就简单写下。Result随便写了几句总结，可以再看看。

Result

The outcome of this focus group is helpful. Many requirements related problems were solved. It also helped confirm the validity of questionnaire.