Requirement and Specification

This section describes the requirement and specifications of the software. We hope specifying the requirements can be better to get the coal. The first part is an overall description of the user requirements of both functional and non-functional. The second part outlines functional system specifications and related non-functional specifications of the software.

User Requirements

Functional Requirements

Users want to learn sorting algorithms through the software.

Users want to watch animations of the process of sorting algorithms.

Users want to understand and prove the correctness of sorting algorithms.

Non-functional Requirements

The Majority but not all sorting algorithms will be animated.

System Specifications

Functional Specification

The lists give the main feature of what a stakeholder shall be able to do. We prioritized the requirements in which the highest priority is 2, followed by 1.5, 1, 0.5.

Priority is 2:

A user shall be able to access a starting module according to individual mastery level.

A user shall be able to select a module of a type of sorting algorithm.

A user shall be able to see the animation, which shows the sorting algorithm process.

A user shall be able to read explanations of each step of sorting algorithms.

A user shall be able to control the progress bar of the animation.

A user shall be able to input the array data manually

A user shall be able to ask the software to randomly generate input.

A user shall be able to test the algorithm through several sets of legal inputs to see if it can work correctly.

A user shall be able to see explanations of sorting algorithms’ correctness.

A user shall be able to choose several sets of illegal inputs.

A user shall be able to test the algorithm cannot work out the sequence and see explanations.

Priority is 1.5:

A user shall be able to follow the guide of the basic algorithms.

A user shall be able to export learning notes.

A user shall be able to see how to use the software.

Priority is 1:

A user shall be able to see history of the latest learned module.

A user shall be able to compare different sorting algorithms’ time complexity.

A user shall be able to see individual learning progress.

A user shall be able to do an after-class exercise.

Priority is 0.5:

A user shall be able to sort an array of numbers manually.

A user shall be able to create breakpoints to see the output for each step.

A user shall be able to share the software.

A user shall be able to see the pseudo-code of sorting algorithms.

A user shall be able to see the programming code of sorting algorithms.

A user shall be able to write feedback to development.

Non-functional Specifications

1. Interoperability

The software should be able to run on Windows system.

The software should be open source.

1. Usability:

The software should be Multilingual. The version of English, Chinese and other languages should be available.

The software should be able to provide the code using C, Python, and Java language.

1. Capability

The software should be able to open in 1 second by users.