Chapter 5 Implementation

This chapter claims implementations done by the team in this academic year. It includes key implementation decisions and how the team implement major system components. The team has decided the software’s platform, programming language, software framework and Integrated Development Environment (IDE) according to technical research and stakeholders’ preferences. Results and analysis of decisions are presented in Section 5.1. Section 5.2 shows the implementations of major system components.

5.1 Key Implementation Decisions

After technical research, Team 10 made decisions in four aspects. The platform is decided to be Windows system. JavaScript (JS) will be the programming language. For IDE, Visual Studio Code is chosen.

5.1.1 Platform

Among four frequently used platforms, PC (macOS/Windows), mobile devices (iOS or Android), web and WeChat Mini Program, team 10 has made a choice. After analysing each platform, WeChat Mini Program and mobile terminal were excluded from the list due to the small size of their adaptive devices. The web is not considered because using cookies may lead to serious privacy problems. However, according to Watts et al., PC is one of the most popular platforms \cite{watts2013cbt}. Since the software we build visualises sorting processes and may have codes, a larger space is needed to display more information. In this case, PC would be suitable to run the software. Besides, the questionnaire results also show that PC is more welcomed among our stakeholders. After considering both platforms’ features and stakeholders’ opinions, PC platform, particularly Windows and MacOS systems are chosen.

5.1.2 Programming Language

The project focuses on animation, which requires a robust front-end tool but does not need any backend service. Therefore, the team decided to only focus on the front-end and implement all the functionalities with a JavaScript front-end programming language. Although AWT, Swing and JavaFX can work with Java and provide a user interface, the restriction on User Interface (UI) design and functionalities make them less attractive. In comparison, JavaScript works well with HTML5 and Cascading Style Sheet (CSS), and HTML5 and CSS are helpful tools in UI design. Besides, existing learning resources are sufficient on the Internet, which would make the process easier.

5.1.3 Software framework

After analysing similar works in the market, we chose React to be the JavaScript library to help the team focus on feature implementation rather than fundamental things. React is a component-based JavaScript library that enables building software based on reusable components. As our software may have several repeated components and have a high level of interaction, React \cite{React} is a suitable library to enable us to create interactive and reusable UIs. With React, the software could only be a web app dependent on a web browser. Electron \cite{Electron} is the framework to allow the team to build desktop applications based on the web. In other words, Electron is a wrapper that makes it possible to install and run the React app on a PC. Therefore, the team chose React as the JavaScript library and Electron to be the framework.

5.1.4 Integrated Development Environment

The team has considered IntelliJ IDEA or Visual Studio Code as the Integrated Development Environment. After technical research, Visual Studio Code (VSC) is selected as the development tool. VSC has plentiful features on front-end programming, such as real-time preview and CSS preloading. As an IDE with a community of millions of developers, VSC also has an active extension market that provides better support to React than it in IDEA. More importantly, most of the tutorials online teaching React and JavaScript use VSC as well. We chose to use VSC as a mature platform to ensure stability during the whole development phase.

References

@article{watts2013cbt,

title={CBT for depression: a pilot RCT comparing mobile phone vs. computer},

author={Watts, Sarah and Mackenzie, Anna and Thomas, Cherian and Griskaitis, Al and Mewton, Louise and Williams, Alishia and Andrews, Gavin},

journal={BMC psychiatry},

volume={13},

number={1},

pages={1--9},

year={2013},

publisher={BioMed Central}

}

@misc{Electron,

howpublished = "\url{ https://www.electronjs.org/docs/tutorial/quick-start }",

title = { Electron Documentation },

year = {2021},

author = { OpenJS Foundation and Electron contributors },

note = "Accessed: 2021-3-25"

}