

Thesis Project Portfolio

A Grading Tool to Simplify the Process of Grading Papers

(Technical Report)

**Title of STS Research Paper Notetaking Applications in Computer Science and their
Connection with Different Notetaking Strategies and Theories of Learning in Computer
Science Education**

(STS Research Paper)

An Undergraduate Thesis

Presented to the Faculty of the School of Engineering and Applied Science
University of Virginia • Charlottesville, Virginia

In Fulfillment of the Requirements for the Degree
Bachelor of Science, School of Engineering

Andrew Song

Spring, 2023

Department of Computer Science

Contents of Portfolio

Executive Summary

A Grading Tool to Simplify the Process of Grading Papers

(Technical Report)

Notetaking Applications in Computer Science and their Connection with Different Notetaking Strategies and Theories of Learning in Computer Science Education

(STS Research Paper)

Prospectus

Executive Summary

In my research, I aim to examine the problem of how notetaking applications can be created to cater towards computer science education and learning. With the rise of digital notetaking in the classroom, ideas about how to maximize learning with digital notetaking becomes a question of greater concern. While digital notetaking offers convenience and speed in notetaking, it can come with a number of drawbacks especially when it comes to learning computer science. Digital notetaking apps, however, have great potential to be even more useful than they are now to students, and even more so for computer science subjects. Thus, in my research, I explore the question of how this can be achieved.

To address the stated research problem, I conducted two projects in my technical research project and my STS research project. My technical work involved creating an annotation application to grade papers, aiming to reduce the amount of time spent on grading. Grading papers can often involve writing and giving the same, or similar, feedback to different students which can result in an unnecessarily time-consuming process. Thus, our research team developed a web application designed with a React frontend and Django backend to allow users to annotate online submissions with pre-set comments submitted through the assignment's rubric. After about a year of work, the results consisted of one prototype application and one incomplete application. The prototype application was tested briefly by the professor who led our research team and several teaching assistants for his class. Ultimately, the prototype, while it met the requirements of the application the professor had in mind and led to a faster grading process, was too limited in its overall functionality, and thus, our team shifted to building on the ideas on the initial app to create a new application which was not completed. The ideas and designs behind our annotation application are also applicable to notetaking applications. I believe that, in

creating the annotation application, we were able to gain insights about what a notetaking application catered towards computer science learning would look like, such as implementation of quick annotating/notetaking.

In my STS work, I researched the question of: How do computer science notetaking apps built today make use of computer science notetaking strategies to fit various theories of learning in computer science education? During the process, I examined three popular notetaking apps today that are used frequently in computer science: Obsidian, Notion, and Evernote. These notetaking apps made use of many different strategies that helped users to create more effective notes, fitting under theories of learning such as constructivism and cognitivism. For example, Obsidian makes use of graph views which are an ideal visualization how ideas and topics in programming can relate to each other. Examining sources such as interviews with the developers of the apps themselves, or users of the apps, for example, it was evident that the apps' designs were influenced by different social groups, and that these contributed to the strategies of notetaking in the applications. For instance, Obsidian was created keeping in mind the user experience of a typical software developer. Ultimately, by examining these notetaking applications, it was clear that they all had certain features that appeared to reinforce specific theories of learning while being influenced by various social groups.

Altogether, even though I did not achieve all that I set out to do in my projects, I believe that the research projects that I conducted this year were fruitful. I was able to make steps to answer my general research question by creating an annotation application with similar concepts to notetaking apps, and, in addition, I was able to analyze popular notetaking applications to see how they aided learning. I did, however, fall short in creating a more advanced annotation application in my technical project. Future research in these topics would involve gaining a

deeper understanding of how notetaking strategies fit under different learning theories. Further, it would involve increasing the breadth of current research, which has only examined a couple applications and theories of learning. Finally, further work in the technical aspect of the research would require implementation of a more advanced software application that could serve as a stronger foundation for an ideal notetaking application for computer science students.