Criterion A: Planning

Defining the problem:

My client is Mr. X, who is a family friend that teaches different grade levels at a school that doesn't have software for managing the school classes. Mr. X and his colleagues are trying to manage all of their classes digitally so they can keep themselves organized but have trouble finding a convenient application. Mr. X feels that he is sometimes forgetful and having this application would be very beneficial for his students and to be able to teach them. In our initial interview (refer to appendix A1), he mentioned that the only solution out there was 5 different applications and found it inconvenient to manage through all of them. He expressed his frustrations for him to input his class information multiple times on different applications and would rather have one application. He also mentioned that his computer doesn't have a lot of primary memory which made it difficult for him to run all the applications at once and really needed a small all in one application.

Rationale for the Proposed Solution:

After my discussion with my client (refer to appendix A2), I did research on different desktop application development tools that they might need. I narrowed my choices down to using java swing, native, C++, and Electron JS. Using native libraries would allow the highest possible performance but lacks cross platform without having to rewrite the entire code base. I then investigated using Java Swing but concluded that it was outdated to other tools and not as efficient for performance. Next, I looked into using OpenGL with C++ because it would be cross platform and have native performance but lacks memory management. Finally I decided to use Electron JS with Node JS because it allowed cross platform development, it had easy access to web API's and was a modern framework.

My client came up with an application that manages the classes all into one application (refer to appendix A2). One of the features for this app is an area where he has the ability to make report cards more easily with a database for ways to write down the report cards. Another important feature he requested was for a more simplified repository of information such as videos, pdfs, word documents, images, and more to help him set up the class. The last feature he really wanted was to have a more advanced schedule system that would allow him to see what classes he would have at the current time based on a drag and drop-based schedule.

Stating Success Criteria:

The program features:

1. The application detects if the user opened the application for the first time
2. An operating system layout with a header and dark / light mode
3. Ability to toggle the animation system
4. A shortcut to edit each class from the home page
5. Being able to create and edit different classes
6. Ability to drag and drop files into a repository of each individual class with the ability to preview them
7. A scheduling system in a drag and drop format that is seperated in multiple weeks
8. Having a text editor for writing report cards
9. A data based for the different sentences to add them
10. The report card section can learn from the user's input and write