Post Planning

ANJALI ROY:

* **# of lines written:** I have written between 100-150 lines of code this sprint. My focus was implementing the keyboard function through the view text as well as creating the generate highlight button. When implementing my previous written code, we had implemented it with a text field. Later, we realized that we needed a text view for our syntax highlighter. This caused the overlay to override the keyboard functionality. So, I had to go back in and figure out how to make the keyboard function appear again. I also fixed the tab button for out text view function. When implementing the tab feature there were a lot of errors. There was no object that was assigned to the indentation, so I had to create a new definition and implement properly. Morgan and I also spent a lot of time fixing errors in the code. As we continued to merge all the code, more and more errors appeared. Most of my time was spent fixing the errors. This took many hours and was quite tedious. As I fixed some errors, new ones would appear as a result. So, I eventually had to start commenting out some of the code to build it successfully.
* **Is code commented?:** The code I have implemented includes comments. I have finally gotten around to writing comments. This is an important part of coding that I have not been sticking too. I should be better moving forward about holding myself accountable. The readability will be much easier once I start following this rule.
* **Testing results**: When it came to testing, there were a lot of components that came into play. This is where a lot of our time was spent during this sprint. I created a lot of test cases to ensure we hit all the possibilities in our code. This showcased a lot of insecurities. Moving from this step, I went back in and refactored the current code. We learned a lot about this in Software Engineering I and it was nice to recall that class and get to work.
* **Issues:** A lot of issues we had was getting the highlighter to implement correctly. This was the main portion or Morgan’s job this sprint. Her and I spent a lot of time discovering different ways to implement this main feature. Morgan and I talked through a lot of different cases before picking our current design. The issue with this was implementing this feature when we already had the smaller features. What we should have done is implement the highlight feature first and then build off that. This would have saved us a lot of time with code implementation and reduced debugging time.
* **To-Dos for next Sprint:** For the next sprint, we will continue to make our app better. This include making the design more eye appealing. Right now, we have a lot of blocks and default fonts. We will go in and really dive into the design. This will make our app look nicer and more cohesive.
* **Helpful links found during Sprint:** <https://stackoverflow.com/questions/73675220/how-can-i-add-code-syntax-highlighting-in-a-nstextview-using-highlighter-js>

<https://developer.apple.com/documentation/uikit/keyboards_and_input/creating_a_custom_keyboard>