**Report**

* Features Implemented

Improved mirror tool: The mirror tool but as a button so that it can mirror all drawing tools rather than just the freehand.

A bubble tool: A new drawing tool which draws circles (bubbles) randomly around the cursor.

An eraser tool: A bog-standard eraser which allows the user to erase their previous drawings like freehand.

A brush size slider: A slider at the bottom enabling the user to change the size of their slider in a fun way.

Different colour borders for the colour palette: Depending on what colour chosen from the palette, it’ll have the border colour of the opposite colour on the palette (e.g. choosing the first colour to draw with will have the border colour of the last colour of the palette).

Looks improvement: Slight changes made to the CSS to enhance the look of the app but wasn’t changed too much as the target audience is set for a younger audience and if it was made too sleek or complicated then it would remove the fun of the app for the prioritized audience.

Drag & drop: this seemed complicated but with the help of referenced work provided, it was done quite fast and allows for the user to drag and drop an image file that they would wish to trace and it would turn the image opaque to help see their drawing on top of the image they used.

* Planning and coordinated development including progress log
* Code structure
* Coding techniques used
* Challenges faced

Mirroring tool: Converting the mirroring draw tool into just a button that works for all sounded simple at first but then we had to ensure that it still works for the other tools besides the freehand. It took a while to implement via functions and different methods but after time, we made a method that worked for all the tools and can be easily implemented into any new tools that were created.

Using the same principle the mirror already had plus some more code implementation into both the mirror and sketch file. It checks what tool is being used and whether the mirror button had been activated to then it’ll copy the drawing but as the mirrored result. The mirroring is all done without the use of load and update pixels too because this created issues at the time, so we worked around it. The only thing that uses load and update pixels is the drawLine() function that simply draws the line for the type of mirroring taking place.

* Self-evaluation
* Progress log (not word count)