**Computing Project:**

**Drawing Application Report**

# **Introduction:**

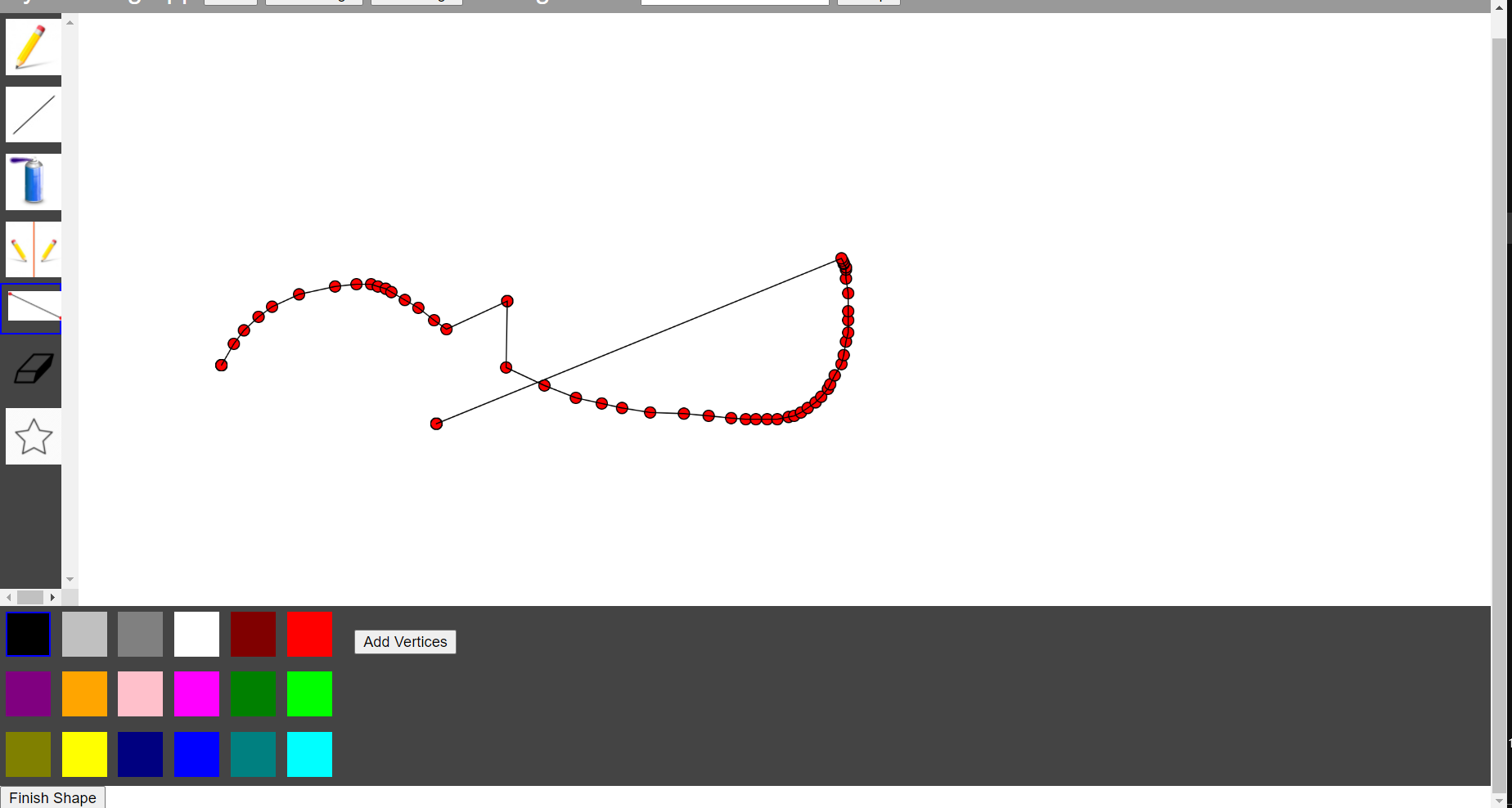
This report will discuss the features added to the drawing app. The information will include an edit shape tool, eraser tool, start trail tool, load image and changing the width of an instrument.

## **Implementation:**

**Edit Shape Tool**

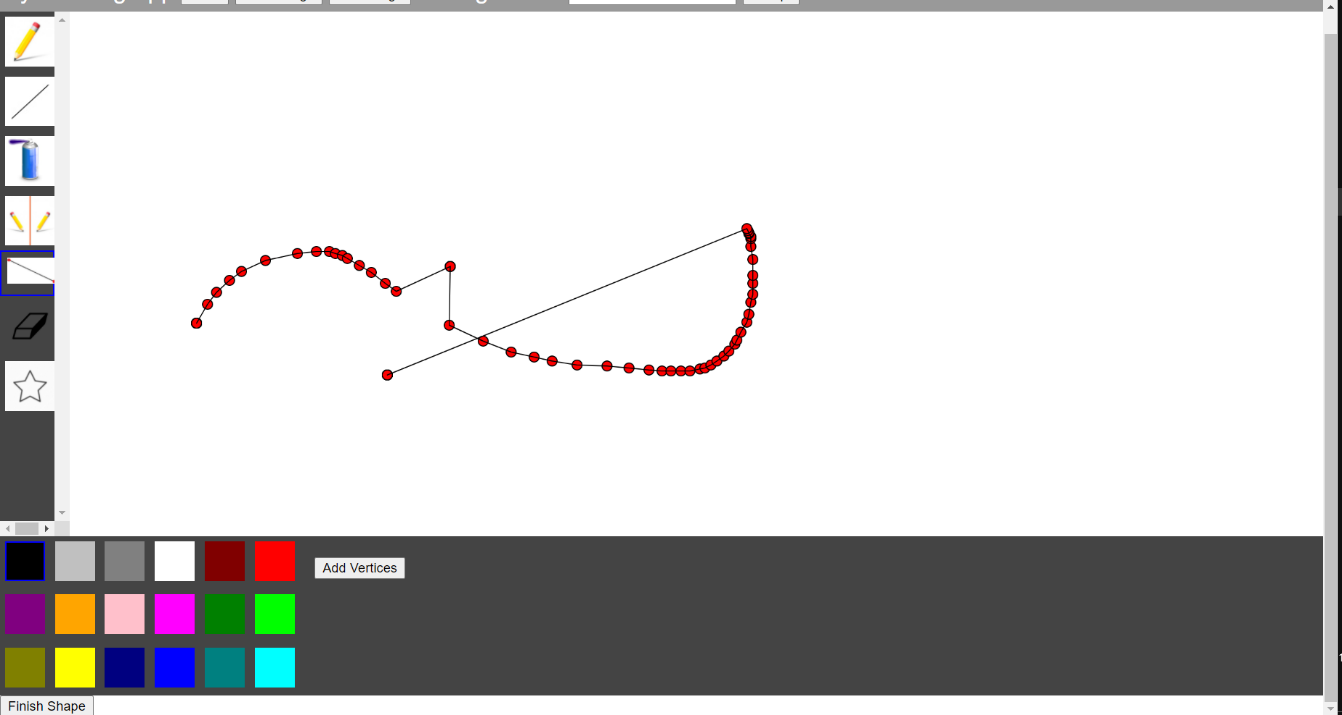
The edit shape tool allows users to create unique shapes and then manipulate that shape depending on when the mouse pressed on the canvas. As Thouficul did the project independently, each feature and tools made—making a schedule depending on the tool and what might have been involved and decided as this was difficult to implement to take three weeks. This plan was to include any problems during testing and when coding the tool.

However, some problems did arise when the tool is integrating into the application and the buttons. The first was integrating the instrument, where it was not finishing the shape pressed the "Finish Shape" button. To rectify this issue, reset the array that each mouse pressed location was at and then draw the shape again when the edit mode is false.



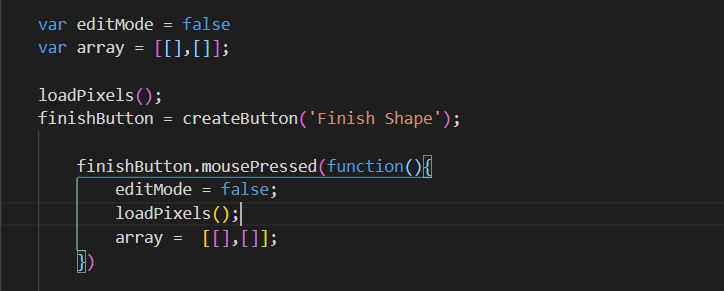
Reloads the pixels and resets the array and edit mode to false.

Also, changing the tool itself to be a constructor function is when the draw function is calling to draw the shape with the new array. The second was the tool's buttons to be next to the colours, so it would be easily accessible and keep a unified layout. The problem was adding two buttons, as it is a CSS layout, adding one of them, and that being the "Edit Shape" button and leaving the finish shape button at the bottom of the page.



Pressing edit shape changes the buttons name to Add Vertices and draws little red circles at each point of when the mouse is pressed.

The code constructed using constructor functions to make the shape into an object, and the array reset when the finish button pressed to use the array again and create a new shape. A coding technique that helped with this tool is using load pixels and update pixels function to clear and reset the canvas with the unique shape.

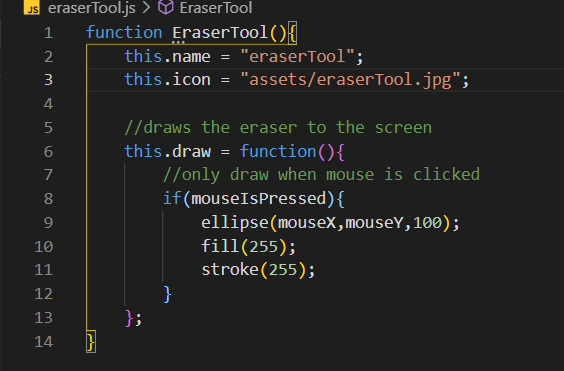


Whenever the button is pressed to reset the array and then draw again

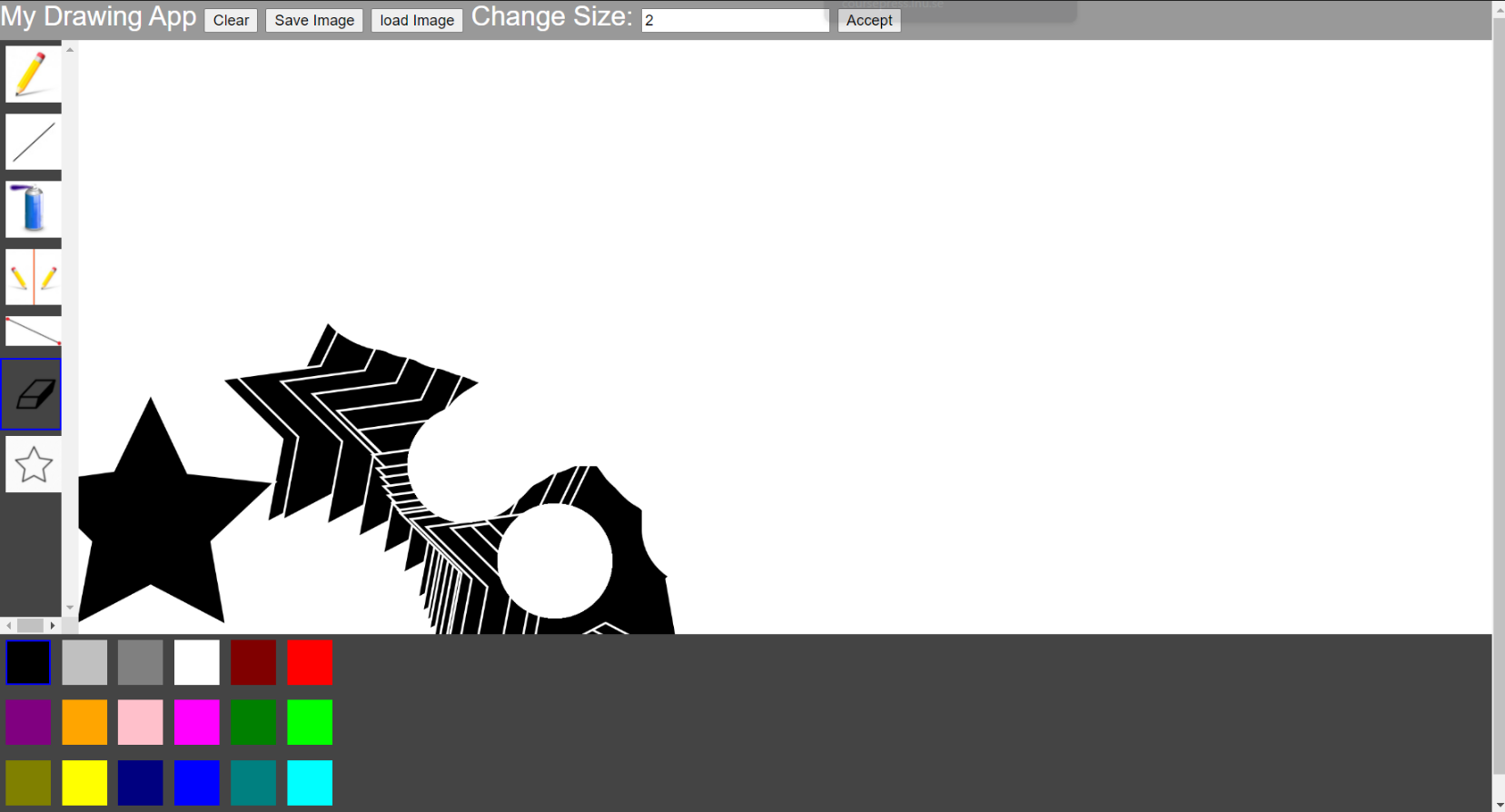
Calling “loadPixels()” calls a new set of pixels on the screen and allows those pixels to be manipulate again and again.

**Eraser tool**

The eraser tool uses a white fill for the ellipse and only continues to erase when the mouse pressed on canvas. This tool does not erase itself merely draws a circle on top of it to give the appearance of erasing a tool. The only problem that occurred was when the fill colour changed whenever pressed would change; to rectify this having a permanent fill colour meant that it could not be changed. As for the schedule was decided it would be completed in the last week, set aside for the edit shape tool and ready for the peer-review.



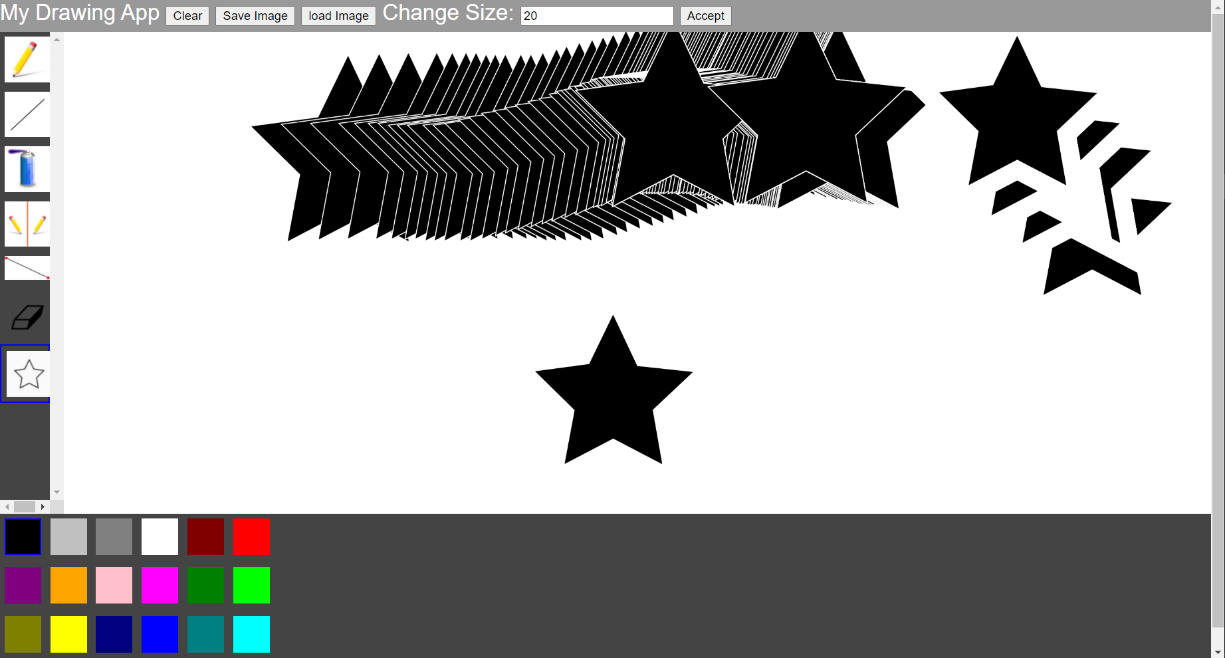
Stroke and fill are set so that the colour palette cannot affect the eraser tool otherwise it becomes a circle tool.



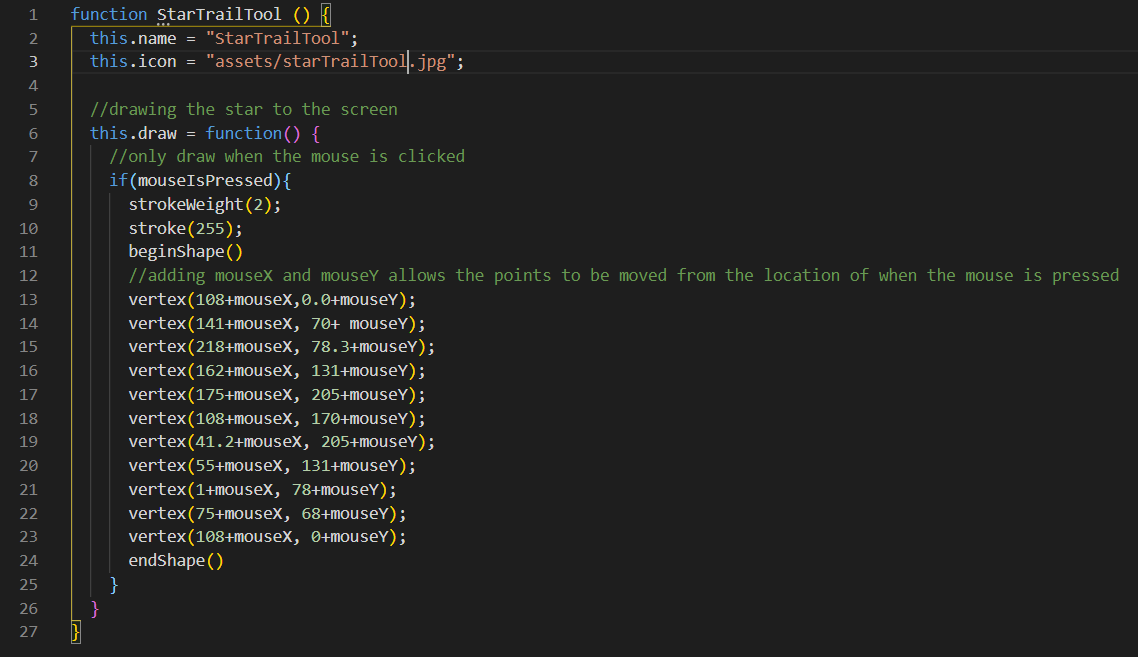
The eraser tool draws a white circle on top of what is currently on screen

**Star trail tool**

The star trail tool creates a trail of stars from the mouse pressed location and continues to draw them until pressed no longer by adding the mouseX and mouseY coordinates to each vertex of the star and manipulating each point about the mouse. However, the only problem was 200px that the star was away from the mouse, many methods later but no changing the distance from both mouseX and mouseY. The timeline allocated for this was one week to include the implementation of the toll and make time to test and adjust as progressed.



Tool when the stroke weight width is changed and adds a larger border to the star.

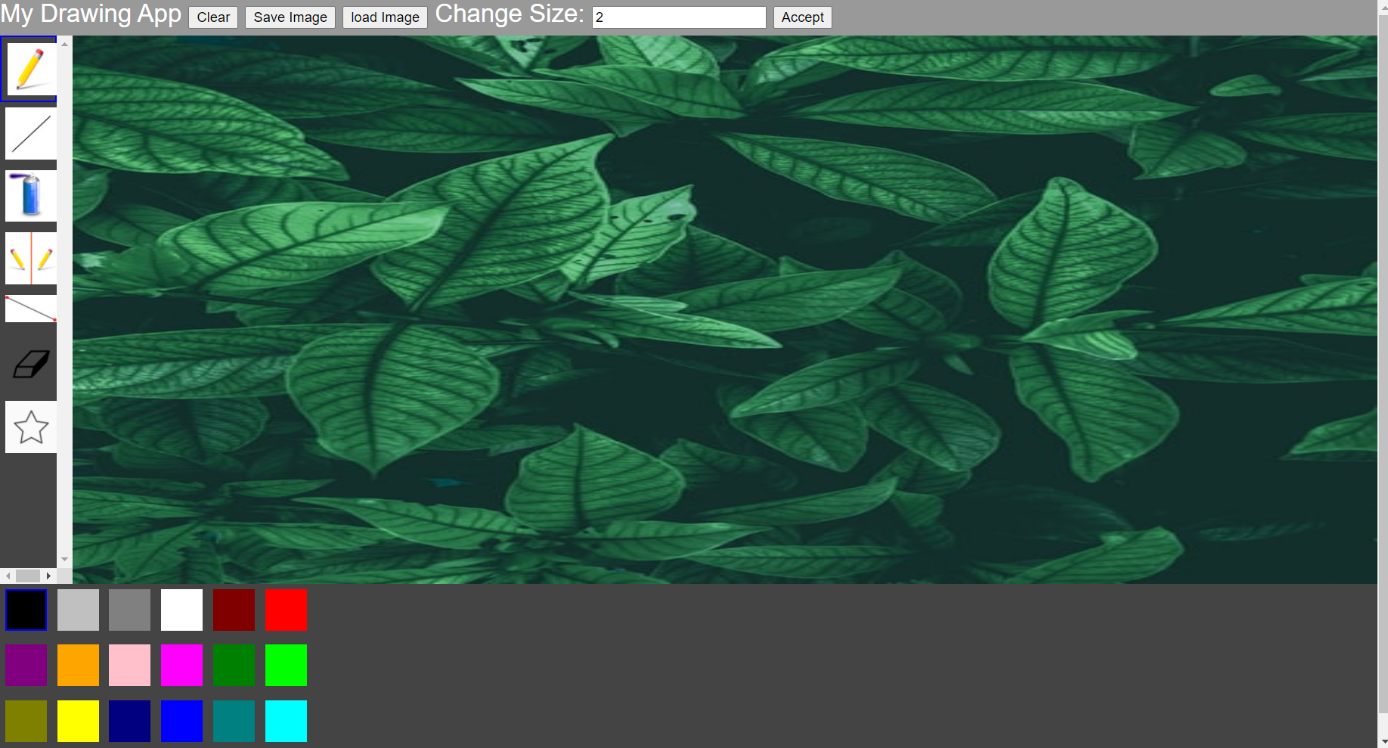


Stroke weight set to 2.

Allows the shape to move along with mouse.

**Load Image Button**

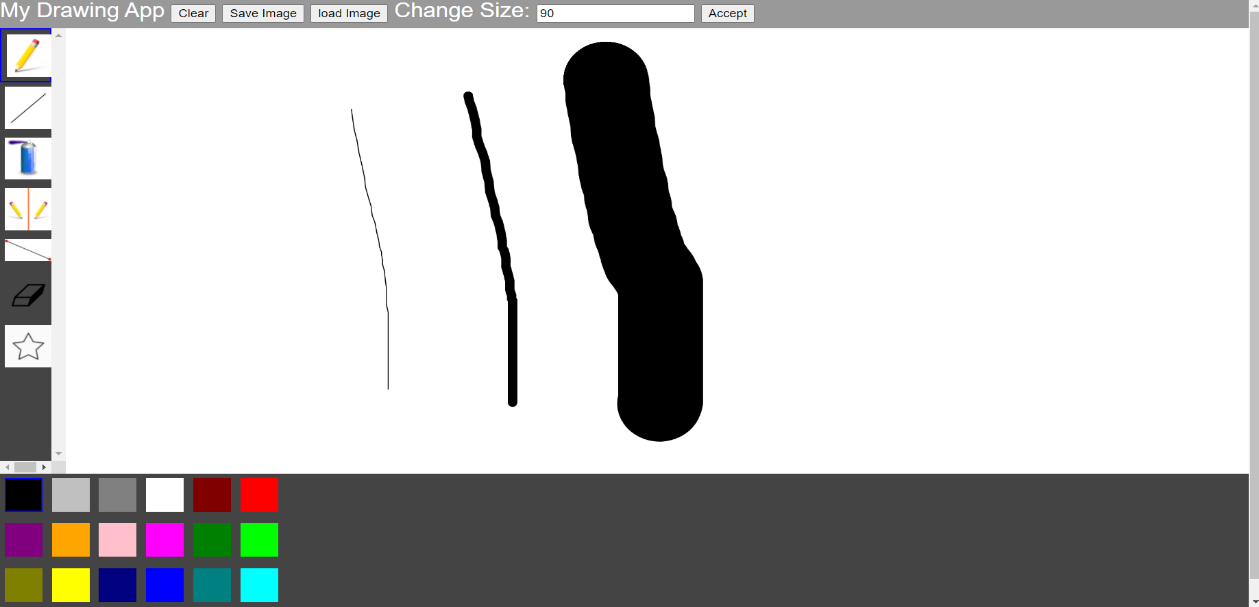
The load image button will load an image to the size of the canvas. However, the first time implemented using the computers folder system to upload and choose a file was the first. However, reverting to using the load image function and making sure the image to upload is called "image.jpg". The timeline I gave for this was one week to ensure the image would load and save when the button is pressed and allow other application tools to function on top of the image.



When load image is pressed to load an image from the assets file and load a file that is called “image.jpg”

**Pencil width adjustment**

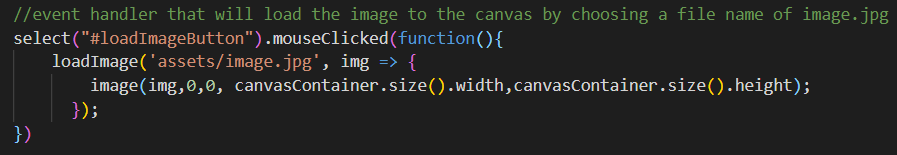
The pencil tool only allows users to adjust the width of the pencil by any range of pixels as desired. Nevertheless, the default for either tool is two pixels; otherwise, there would be nothing drawn to the screen for either tool. However, there was no way to put the pencil width adjustment textbox at the bottom during the tool's implementation. CSS option allows one item in the grid area, so instead, the textbox for the pencil width and its buttons near the top of the app. The changes made to a tools stroke weight affect both the tool itself and the line tool. The width created a problem as the width will not reset when the tool is changed, but it affected the star trail tool, increasing the applied stroke colour's width during testing. Then fixed the stroke weight to 2 so it could not be changed. So, for this tool, two weeks were allocated to help make it.



Whenever the input number is changed changes the stroke weight of the tool and

**Evaluation**

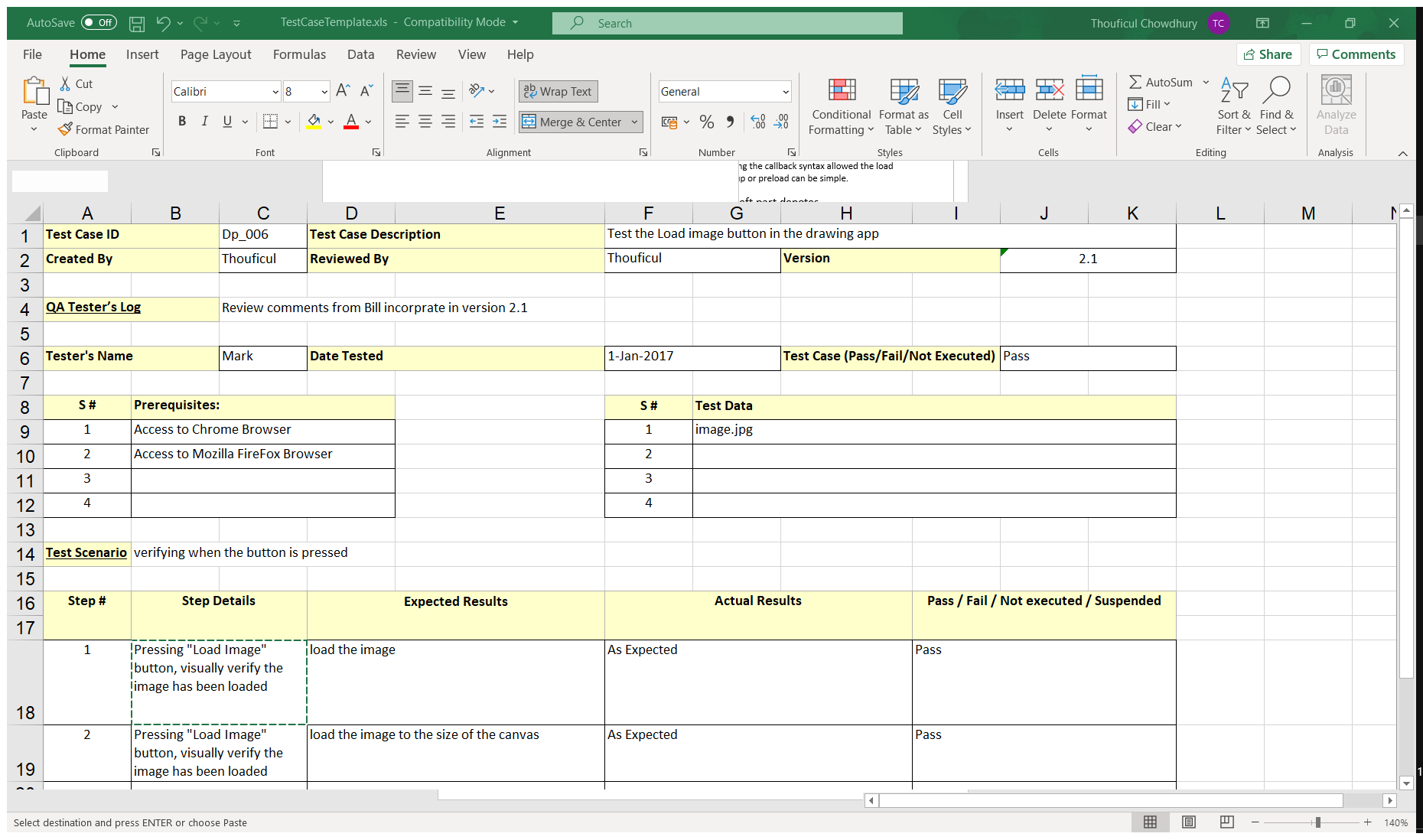
Overall, the implementation of tools went smoothly, and one of the tools that went well is the load image button because using the load image callback syntax to load an image to the canvas and still be drawn on when other tools were selected. However, using the callback syntax allowed the load image to work as it does not require many functions as setup or preload can be simple.



An ES6 function called arrow function, where the left part denotes the input and the right part is the output of the function.

The button that this function will be linked to.

I would change the one thing, having a free choice of an image by having the windows folder appear, but not have one that has to be called "image.jpg".



To sum up, the development experience was very interesting and exciting because many different ideas and techniques had to be used for each tool and manipulated differently as new methods had to be thought of.

**Appendix**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Progress Log** | | | |  |
| Date | Description | Date Completed | Comments | Pass/Fail/  Suspended |
| 26/02/2021 | Ideas for the drawing app discussed | 26/02/2021 | Edit Shape tool, eraser tool, load image button, pencil/line width tool | Pass |
| 26/02/2021 | Started the development of edit shape tool | 19/03/2021 | However, the finished tool has a few bugs and does not have an options button to a certain extent. | Suspended |
| 12/03/2021 | Started the development of eraser tool | 19/03/2021 | The eraser tool was finished, but it worked when the mouse pressed | Fail |
| 12/03/2021 | README File for the application | 19/03/2021 | Added the file to the app as a text file. Explains what the app is and the developments that have been added | Suspended |
| 19/03/2021 | submitting for peer-review | 19/03/2021 | Submitted my files before 4 pm | Pass |
| 19/03/2021 | Peer-review feedback | 26/03/2021 | looked at all the applications given to me and made notable feedback | Pass |
| 26/03/2021 | star trail tool | 09/04/2021 | Star trail implemented and tested | Pass |
| 09/04/2021 | pencil/line width adjustment | 23/04/2021 | Pencil/line tool adjusted to make sure that it does not affect other tools like star trail or edit shape tool | Pass |
| 19/04/2021 | Final Adjustments and sorting any new bugs | 25/04/2021 | Sorted any issues, fixed the indentation and made a note of any changes in the testing logs. | Pass |