Project title:	Drawing App
Topic:	Week 14 Log

New Features/Improvements

- ★ Shapes can now be resized using the mouse
- ★ The slider is now used to determine the stroke weight of the shapes
- ★ Eraser tool has been implemented and works accordingly
- ★ A rudimentary version of the random string tool has been added

Planned Features

- ★ Have a look at trying to implement 3D drawings to the canvas
- ★ Add more options to the stamp tool
- ★ Start refactoring the code

Problems/Issues

- ★ I felt annoyed that I did not figure out how to resize shapes on the canvas before the midterm's deadline
- ★ The method erase() did not work upon initial implementation, so I had to update the p5min library in order to get it working
- ★ I wanted to add a general text tool; however, the amount of time it would have taken to integrate user input with existing draw functionality was not worth the risk, the time or effort. Therefore, I added a text file with strings into the assets folder where, with the help of the pickString function, random strings are displayed on the screen

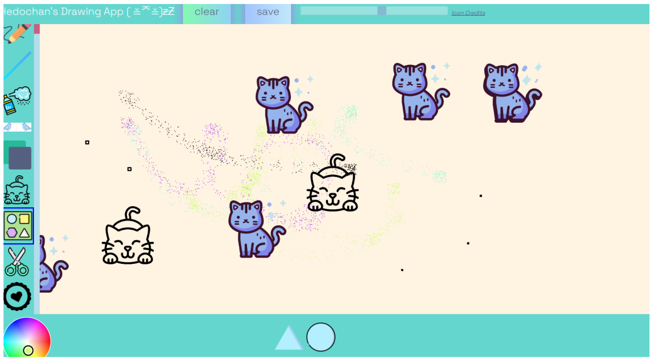


FIG 1: Shapes can now be resized.

On Target? Yes.

Project title:	Drawing App
Topic:	Week 15 Log

New Features/Improvements

- ★ Updated the random string tool to display random emoji fetched from an array containing these rather than relying on a text file stored in the assets folder
- ★ Updated the stamp tool to display more images and ensure that nothing is drawn on the screen until the user has clicked on one of the images as this was raised as an issue by a user
- ★ Started implementing the 3D feature
- ★ Refactored some of the jQueries in the helper function

Planned Features

- Implement a tool that draws a line and has some funny cat animation trailing after it
- Look into creating images from user interaction
- Have a look at implementing a cropping tool that uses some of the cutting tool logic as shown by the course instructors

Problems/Issues

- The cutting tool revealed some issues that I missed due to an update in the library. Tried out a couple of options; however, the template from the lecture also no longer works. Raised an issue in Github: https://github.com/processing/p5.js/issues/4999; however, the devs could not replicate the issue.
- In the end, I went with using https://github.com/Lartu/p5.clickable as a workaround; however, the implementation was faulty, and I've decided to scrap the cutting tool altogether as it added no value to the project in its current form.

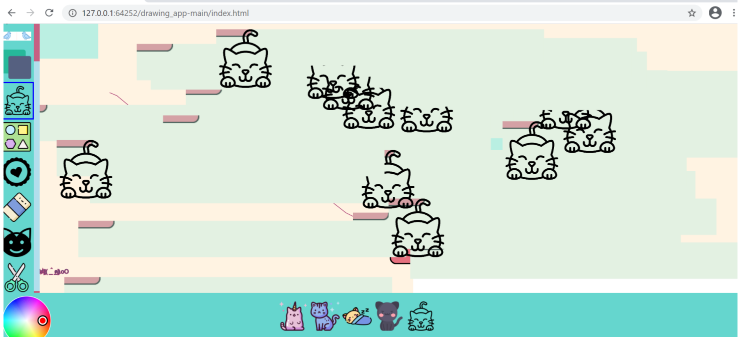


FIG 2: A faulty implementation of the cutting tool.

On Target? Yes.

Project title:	**Drawing App**
Topic:	**Week 16 Log**

The noteworthy thing about this week is that I had three to four people test the app. This resulted in some useful feedback.

New Features/Improvements

- Added more to the 3D tools - the 3D drawing is added via graphics and then stored as an image. If the frame count has hit a certain limit, a text appears indicating that the user has to reload in case they want a new 3D sketch
- Added a reload application button for giving the user the ability to reload.
- Refactored the sketch file -- the setup function has been moved to its own function
- More refactoring in the sense that the tools, the helper functions and the css stylesheet have been moved to their own directories
- Started implementing some basic animation(s) using sprites
- The change background tool has been renamed to AddTransparentLayer as it caused some confusion to the people testing the app
- A media query has been added to deal with large screens, although I am not certain that it will avoid the bug that one user pointed out
- The rubber tool now erases things using the background colour -- a user pointed out that having another colour was confusing

Planned Features

- Look into creating images from user interaction
- Refactor more
- Have a look at possibly re-adding the cutting tool

Problems/Issues

- I had to use createGraphics() in order to keep the 2D functionality working alongside the 3D one -- any 3D frames are stored as images
- Had a look at incorporating classes and use more of an OO approach; however, Javascript does not behave in a way I expected it to (compared to say Java, Python, etc), and it would have been too much of a learning curve at this rate to apply such changes

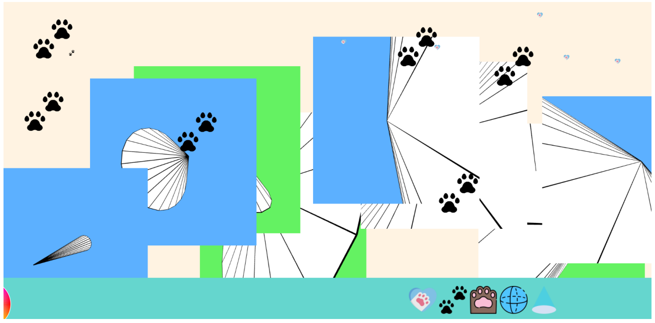


FIG 3: A first look at the 3D feature.

On Target? Yes.

Project title:	**Drawing App**
Topic:	**Week 17 Log**

New Features/Improvements

- Refactoring of the event handler that deals with assigning the cat id array entries with the specific image that is to be loaded
- Subtle change to the way the shapes are drawn, stroke() is now a random colour input, just to make things more interesting
- Added an option for the user to generate shapes using random colours via a new button
- Added more options to the animation, although it is a bit wonky
- Re-added the cutting tool

Planned Features

- Refactor more - especially the event helpers related to shapes

Problems/Issues

- The event handler dealing with the shapes is clunky
- The animations are a bit wonky and creepy looking

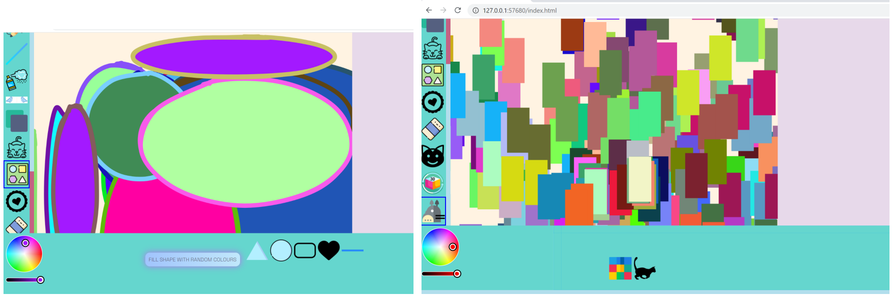


FIG 4 and 5: A button has been added to allow generated shapes in random colours and there is a rubik animation.

On Target? Yes.

Project title:	**Drawing App**
Topic:	**Week 18 Log**

As of this week, there are no more new planned features as this project is hitting a maturity level in terms of new features that is acceptable. I have also gathered more feedback from various colleagues.

Improvements

- ★ The cutting tool has been implemented / reimagined as a cropping tool with its logic being similar to that of the geometric shapes tool.
- ★ The animation tool is now finished and has more options and less wonky animation
- ★ A help html page has been added to provide information to an end user as some users raised issues to the fact that it was not easy to garner how certain features worked

Planned Improvements

- ★ Have a look at making the displaying of 3D images easier on the 3D tool as some users raised complaints
- ★ Looking at bits and bobs here and there

Problems/Issues

- ★ The cutting tool has been reimagined as a cropping tool after a lot of dead ends and quite a few external libraries being imported and then deleted. Attempts to rewrite the tutorial code resulted in the final decision to simply ditch the part that pastes an image and focus on the cutting of a selected area instead

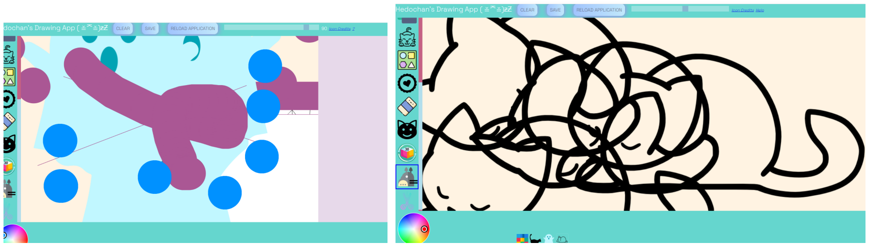


FIG 6 and 7: The cropping tool and some cat animation.

On Target? Yes.

Project title:	**Drawing App**
Topic:	**Week 19 Log**

The main challenge in this week and the coming weeks is juggling any improvements to the drawing app with the upcoming exams. I also had some users provide feedback this week.

Improvements

- The 3D tool has been refactored in the sense that the createGraphics only draws a canvas of 400 by 400. Moreover, the cone and sphere are no longer defined by mouseX and mouseY coordinates, which makes the display of these images smoother on the screen
- Also added a texture to the sphere, so that it looks a bit more interesting
- Refactored code as much as possible by removing multiple if statements from the tools and replacing them with ternary operators
- Animation tool with the cat has been rendered a bit more interesting; one more image has been added
- Report has been written and only needs some retooling
- Smoothed over some typos in the help html file.
- The cropping tool area being removed is now the same as the background colour rather than white after a user raised it as a bug.

Planned Improvements

- Have a look at adding/implementing improvements/suggestions from peer feedback.
- Resolve inconsistencies in the way the shape tools are sometimes filled and sometimes not as another user raised a bug

Problems/Issues

- Feedback on the 3D tool has been mixed, and I have contemplated its removal; however, I do hope that the improvements make it a little more interesting to users

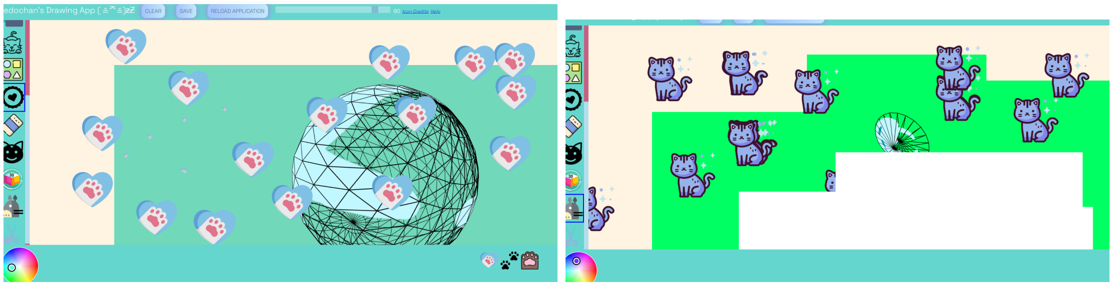


FIG 8 and 9: The 3D drawing now has a texture and an old version of the eraser tool with the selected area being white.

On Target? Yes.

Project title:	**Drawing App**
Topic:	**Week 20 Log**

This week things are coming to a head with finals on the horizon. The main focus of this week is testing and refactoring any sloppy code.

Improvements

- Reduced the amount of global variables used throughout the project - as far as possible without breaking the existing functionality.

- ★ Replaced the remaining global variables with singletons as recommended by the following article (https://robdodson.me/javascript-design-patterns-singleton/)
- ★ After the 3D tool's capacity has been exhausted, the canvas is restored to its original size rather than the modified 400x400 size the buffer canvas uses. This restores consistency in the project. The image is restored via a simple if check that displays nothing if the image is null or undefined

Planned Improvements

- ★ Have a look at fixing some of the performance issues (e.g. minification of library files)
- ★ Keep editing the report.

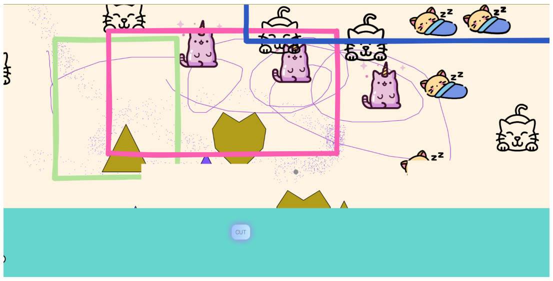


FIG 10: The eraser tool now uses the beige background in order to retain consistency.

Problems/Issues

- ★ I have received a lot of valuable feedback; however, some of it is more subjective than others. One of the difficult things I have noticed is that it is not always possible to please everyone. I feel bad for not addressing all of the criticism I have received, but sometimes -- as in the case of not drawing a default shape/image/etc on the canvas -- I got conflicting views. One user did not like being provided default options and preferred to choose via the selected images on the bottom of the canvas, while another felt that not having a default options made it difficult to gander what the purpose of the tool was
- ★ I noticed some performance issues while testing the application on Github; however, none of these issues were present while developing locally. The reload button should provide the end user with a way to deal with that particular issue, but further testing would be required to find a better fix
- ★ I received some feedback on the midterm, which pointed out that my reporting skills required improvement -- this means that my current report will have to be rewritten from scratch

On Target? Yes.

Project title:	**Drawing App**
Topic:	**Week 21 Log**

The last week is an opportunity to fix any final bugs and reflect on whether this project has been successful or not. Any feedback from the midterm will also be taken into consideration.

Improvements

- ★ The loop responsible for loading the cat and non sequence images has been rewritten into a nested loop, thus placing these pieces of code together.
- ★ A massive nested loop has been added to eventHelpers.js file, thus also putting all of that code in one place.
- ★ Converted the external libraries into minified versions in order to improve performance
- ★ Disabled some chrome options in the console log after discussing with colleagues on how performance could be improved

Problems/Issues

- I will keep testing the app for performance issues. I have one version where the external libraries are not minified and another where they are. Time will tell which one makes the final cut for the submission
- I have to rewrite my report the so that it adheres more to technical standards, which was not an issue I expected

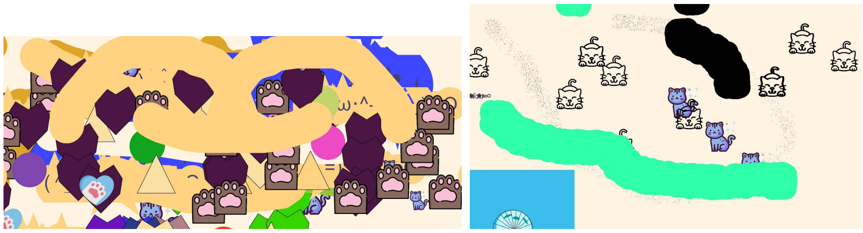


FIG 11 and 12: Screenshots produced by the save button.

On Target? Yes.

Project title: Drawing App
Subject: Log Post Final Extension

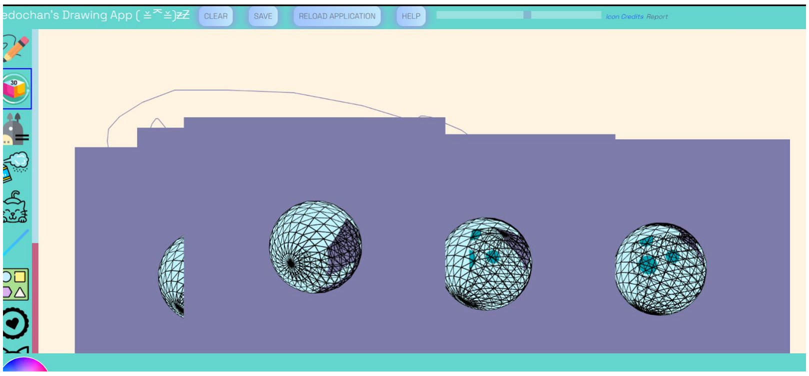
Just covering some changes done after the project's deadline was extended.

Improvements

- ★ Refactoring of the MirrorDrawTool in order to circumvent performance issues caused by the div being wiped after each time that the tool was unselected -- tool now uses buttons created via traditional HTML / CSS
- ★ PNG images have been replaced into WEBP ones in order to optimise performance
- ★ The cropping tool's button has been renamed into the crop button, just to make sure that this is what the tool's functionality is
- ★ Formatting has been improved, wherever possible
- ★ Fixing any bugs raised by a couple of users after some impromptu user testing
- ★ Fixed the broken images on the Help page
- ★ Made the Help page more obvious by turning the text link into a button
- ★ The report is also available in HTML format

Problems / Issues

- ★ Attempts to replace the existing code structure into ES6 fell foul as the time / effort required would have needed an extensive overhaul of the existing implementation
- ★ A first time user raised issues regarding the 3D tool and also said that the cut button did not work, which prompted me to change the Help link into a button; however, I am still worried whether I am making the user think too much.



On Track: Yes.