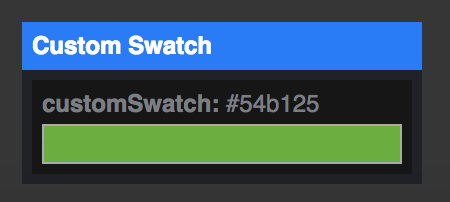
**Big Project Report – Painting App**

We had decided to build upon the drawing app and add our own features. The features we planned on implementing were; eraser tool, text tool, shapes tool, importing images, colour picker, copy / paste and brush sizes.

**Libraries**

In total there were only 2 JavaScript libraries used, p5.gui and tippy.js (apart from jQuery and p5.js). Both are very simple and don’t need too much studying. It took us around 1-2 weeks to fully understand the syntax and functions of both libraries. P5.gui is another library provided by p5. It takes global variables and turns them into good looking ‘Graphical User Interface’ (GUI) windows that allow the user to input i.e. sliders, dropdown menus, checkboxes and most importantly a colour picker. The tippy.js library makes the popovers more pleasant and visible. We didn’t like the fact that the GUI windows were draggable however we looked into the library files and saw that there was a setting where we could turn off the draggable setting, this way the GUI window is always fixed. We also didn’t like the default colour scheme of the GUI windows therefore we had to change the CSS files of the p5.gui library to change the colour that matches our theme.

After implementing p5.gui

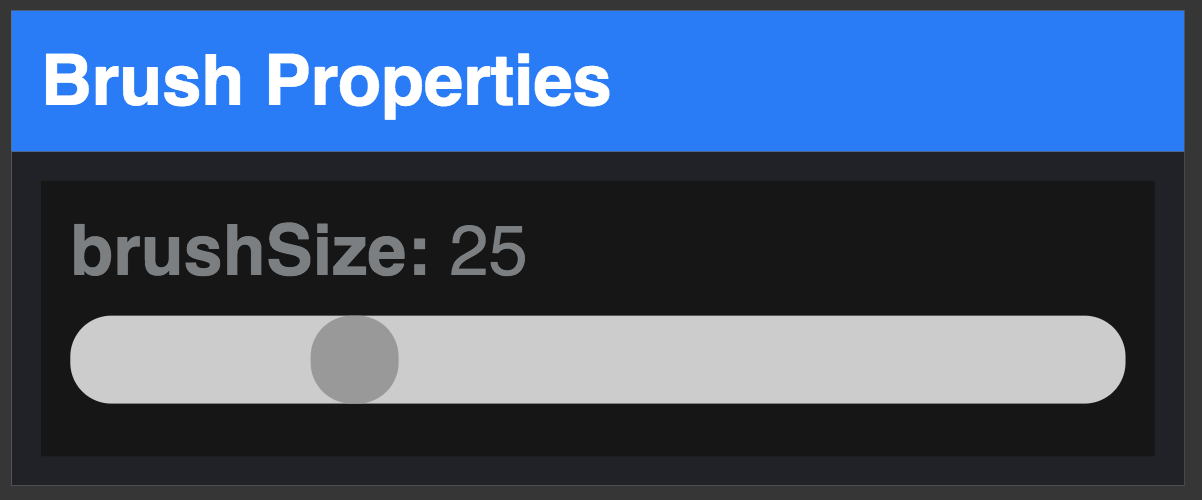


First Itteration

**Features Implemented**

We implemented almost every feature we were planning to implement. We were not able to implement 2 planned features; copy / paste and importing images. Every other planned feature we managed to implement and some unplanned ones too. With all the tools we had a bare bone functionality at first, then all the tools where polished off and at the final iterations had the user inputs to change brush sizes for example. Here is a list of all the features in more detail (in the order it was implemented);

1. **Brush tool**

The brush tool was the first tool we implemented, it was also the easiest as it didn’t require much lines of code. It is meant to replicate a paint brush which is much thicker than the already implemented pencil tool. The code for the tool was borrowed from the pencil tool and slightly modified to accommodate for p5.gui so a slider for brush size can be added.

1. **Eraser Tool**

The eraser tool is meant to function so users can erase drawings. This was also very easy to implement as the code again was borrowed from the pencil but with a slider, so users can change the eraser size. However, whereas both the pencil and brush tool can change the colours the eraser is essentially only white to give the effect that the canvas is being erased.

1. **Text Tool**

The text tool essentially allows the user to add a line of text onto the canvas via a textbox input. The first iteration of the tool had a bug where when the text had been placed but the mouse was still clicked, and the mouse was moved the text would act like a brush and place a lot of the text but in the final iteration it was fixed. In the final iteration we also managed to add extra features such as different fonts and option to make text bold.

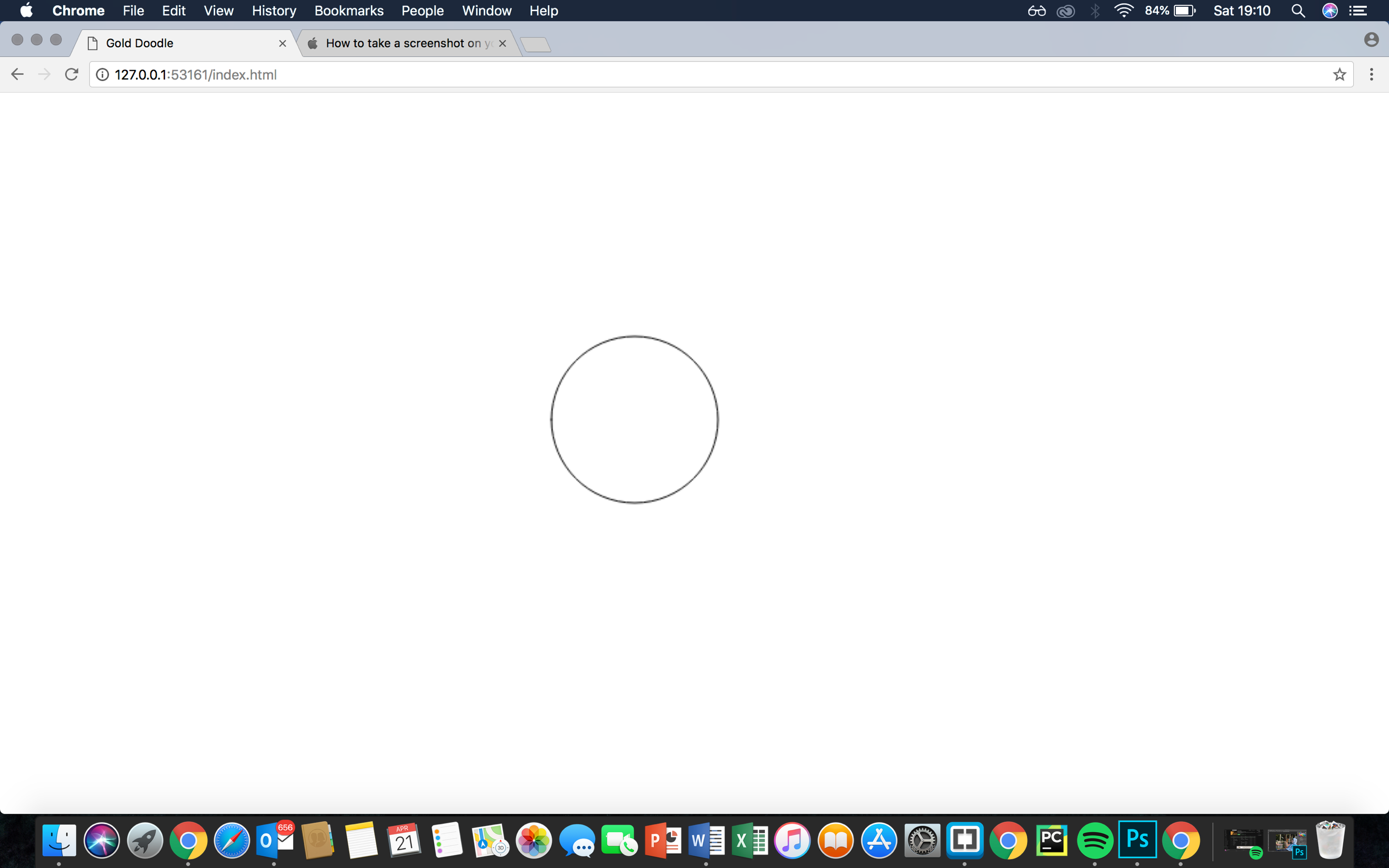
1. **Colour Picker**

The first feature we thought of was this one, to allow the user to draw with any colour they wish. This was added very late into the development stage as the p5.gui library was discovered by us at a very late stage in development. Using the p5.gui adding a colour picker was made much easier. We had to add a colour swatch that was always the colour that the user chose from the colour picker. The only known bug is that once the colour has been chosen in the colour picker the user has to click on the colour swatch again for the tool to change colour.

1. **Shapes Tool**

The shapes tool allows the user to draw 5 different kinds of shapes onto the canvas. This was probably most difficult tool to implement as a moderate amount of calculations were required. Some code was borrowed from the line tool, so the shape can be any size the user desires. We were able to use the p5.gui dropdown menu so the user can choose what shape they want to draw and also the check box to allow for the fill of the shape to be filled.

1. **Other Tools**

Almost every other existing tool was also upgraded in one way or another. For example, the line tool now has an option to change the thickness of the line. The spray can tool can change the spread of the spray so. This is all to allow the user freedom and not be limited. Another feature we added was the marker feature. This feature is essentially a function that can be reused on different tools. The marker allows the user to see the outline or a tool, so the user knows how big the size of the tool is.

**Planned and Coordinated Development**

Most of our planning was done within the first few days of development and some planning was done later on in the stage when we discovered p5.gui library. All of our programming was done via pair programming, we took turns to be “drivers” at the workstation whilst the other was reviewing the code being typed. This way we didn’t really need a service where to sync files.

**Code Structure**

We made sure the codes structure is neat, consistent and readable so marking can be easier. All tools are separated to their own .js files and functions. We also made sure that the code is also properly indented. We tried to avoid global variables however that proved difficult due to p5.gui requiring global variables to function properly.

**Challenges Faced**

Some features we planned on implementing were very difficult to implement. We both tried thinking of ways we could implement them, but unfortunately we struggled. The two features we weren’t able to implement where the copy / paste feature and image import feature. Another challenge we faced was to test the site on windows. At the start we weren’t bothered about testing on Windows OS but in the later stages once we did test on Windows OS we found that it had an unforeseen bug, there where scroll bars on places that we didn’t intend to have. After looking online we found a way to get rid of scrollbars completely using CSS by setting the display to none for the scrollbars.

**What We Would Do Different Next Time**

If we had the chance to do this project again there are several things we would do different. Firstly, we would consider libraries in our planning to reduce the iterations the site goes through.

Another thing we would do different is not use pair programming to be able to simultaneously work on the project, this way we could add more features in less time and also focus on other things such as UI. For the most part we didn’t touch the look of the programme, next time we would’ve spent more time on the user interface such as the icons and colour theme.

**Wiki Appendix**

**Final Project: Drawing App (Case Study 1)**

Features to add:

* **Importing Images (**To be able to import images to the canvas)
* **Text**(To be able to write text on the canvas)
* **Brush Size** (To be able to change brush size for spray-can, line)
* **Shapes** (Circle, Square)
* **More colours** (Colour picker)
* **Eraser**
* **Copy / Paste** (copy and paste images)

**Development Period**

**Week 1 - 2**

Adding the feature Importing images & re-skinning some aspects of project - Usama

Adding text feature - Arif

**Week 3 - 4**

Adding different brush sizes - Usama

Adding shapes feature - Arif

**Week 5 - 6**

Adding more colours - Usama

Adding Eraser functions - Arif

**Week 7 - 8**

Adding copy / paste feature - Usama + Arif

Starting report - Usama + Arif

**Progress**

**Reading Week**

* Text tool full functional**-**added by Usama
* Eraser Added but contains Bug - Added by Usama
* Line tool can have different weights now - Added by Arif
* All intended features Logo added to the toolbox - Arif

**Progress by 16 / 04 / 18**

* All drawing tools have a slider to change the size of (via p5.gui library)
* Eraser Bug fixed -  by Arif
* p5.gui library studied and implemented (makes the UI look much cleaner and professional) - by both Usama and Arif
* Most lines of code was not explained via comments, so both Usama and Arif commented their code
* Most intended features have now been added e.g colour picker, multiple shapes. However both copy / paste feature and import image will most likely not be implemented due to difficulty, no library has been found as of yet that will simplify things. However we are still trying.
* Tweaked the p5.gui library so the slider "windows" aren't draggable, only stay stationary - by Arif
* Both Usama and Arif are coding and testing on Mac OS, we tested if the programme is working the same on Windows OS. The testing showed there are major visual bugs in the programme on Windows OS. Will have to fix this

**Progress by 21 / 04 / 18**

* Visual bug fixed on windows
* another simple library used (tippy.js) that makes the popover much better looking
* Implementing the copy / paste feature and import image is looking difficult to implement so both of us have agreed not to add them anymore
* fonts and bold option added for the text tool
* Some code structure has been polished to finish everything off