Assignment Implementation Summary

Item Number	Item Description	Implementation Level
1	Git setup and usage	Fully implemented
2	Additional functionality	Fully implemented
3	PLOT tool	Fully implemented
4	RECTANGLE tool	Fully implemented
5	ELLIPSE tool	Fully implemented
6	POLYGON tool	Fully implemented
7	Undo button	Fully implemented
8	Loading Vec images	Fully implemented
9	Saving Vec images	Fully implemented
10	GUI design	Fully implemented

Statement of Contribution

Team Members - Lena Stockwell and Joshua Rowley
Contributions:

Lena Stockwell, n10063137	Joshua Rowley n10233121
● Report ● GUI design and implementation ● Plot tool ● Line tool ● Rectangle tool ● Circle tool ● Polygon tool	● Report ● Back end implementation- vector commands, file saving and loading. ● Save/load functionality ● Drawn Shape object ● Drawing command objects ● Polygon tool

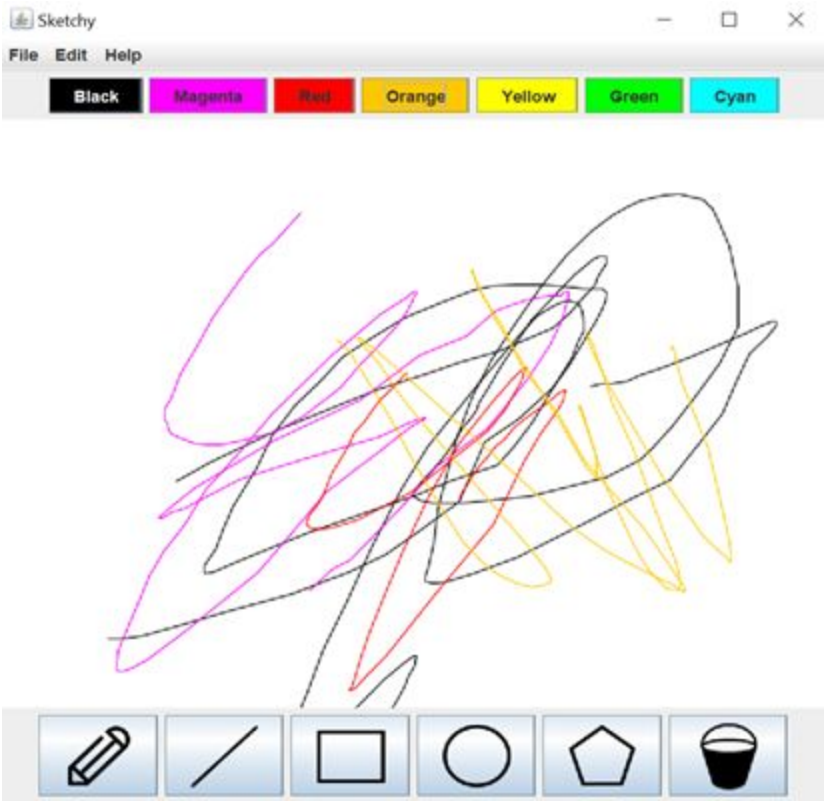
Agile software development processes documentation

An example of how Agile software development principles were used for Sketchy is given through screenshots showcasing the project's history of Git commits over time involving major GUI updates, and the features that were added with each additional update. Initially, the first commit only offers a form with the a very basic implementation of mouse listeners and 'drawing', but eventually evolves to include buttons, colours, and much more functionality, while remaining a working program throughout each version.

Commit 1- Gui Test

Updates:

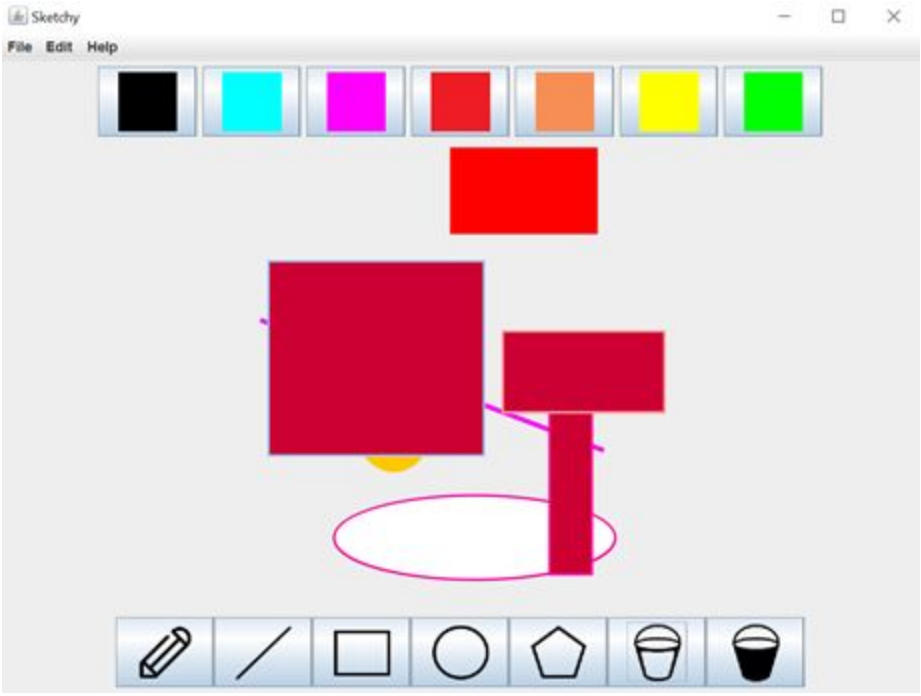
Runnable window
Able to close when prompted
Window is titled
Early placeholder ‘painting’ implementation - just a mouse listener within a JPanel that tracks mouse clicks/drags and paints red squares accordingly.
Incredibly simplistic, made just to get the hang of using basic Swing functions.



Commit 19- Big GUI overhaul

Updates:

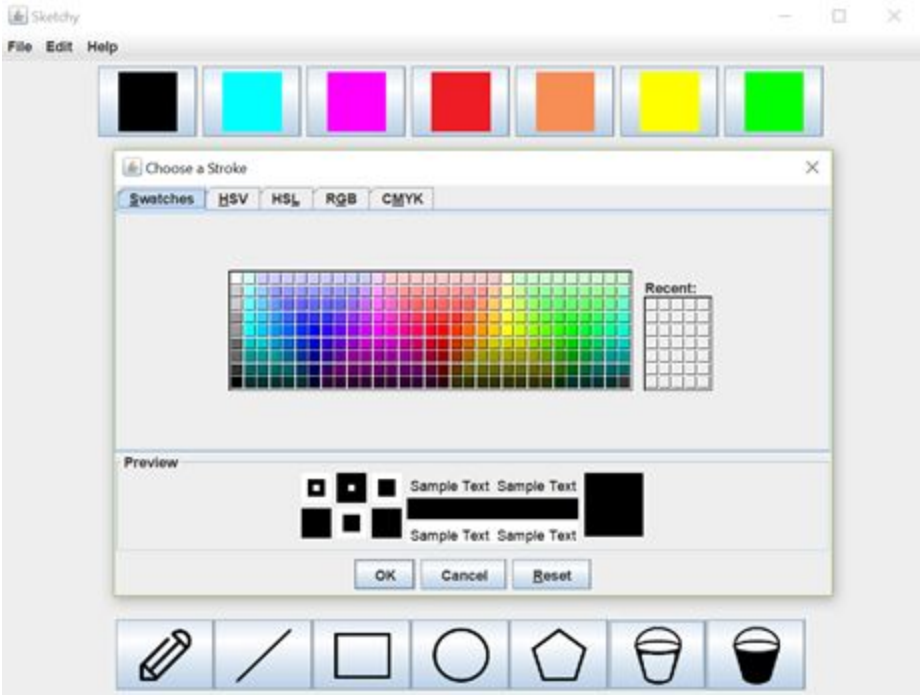
Basic placeholder buttons and icons, most of which were not implemented yet. The only functioning buttons were the colour buttons.
A menu with file, edit and help buttons, and smaller save/load and edit buttons.
Functionality to switch between basic colours
Cursor now draw lines rather than squares.



Commit 30- Lots of functionality

Updates:

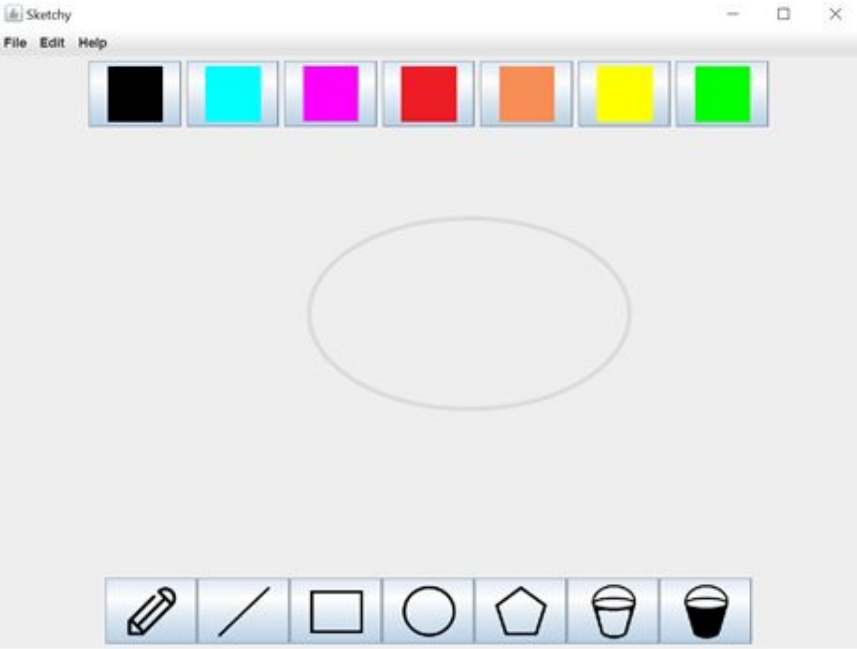
Plot, line, rectangle, and circle tools are now fully implemented
Tools can be switched between
You can now directly edit fill/stroke colours of shapes. However, there is no option for no fill on the rectangle/circle shape



Commit 30- Colour picker window support

Updates:

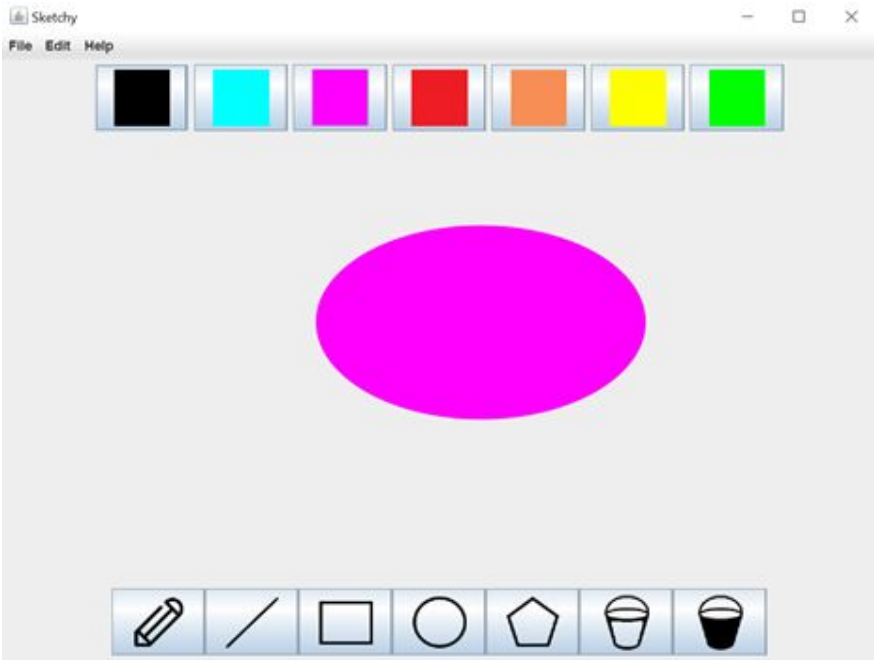
Colour picker now allows user to pick any colour value, not just the seven default colours shown at the top of the frame.



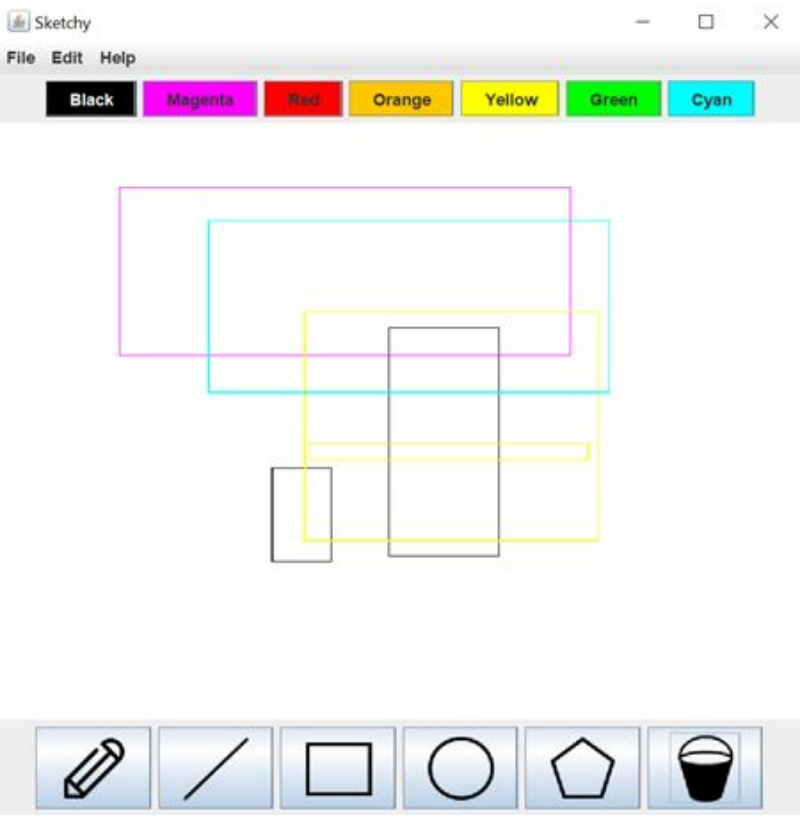
Commit 39- Shape/Line preview

Updates:

Shape preview functionality enables the user to view plots, lines, rectangles and circles before they are drawn with a translucent grey outline, automatically updating as the user drags their mouse.



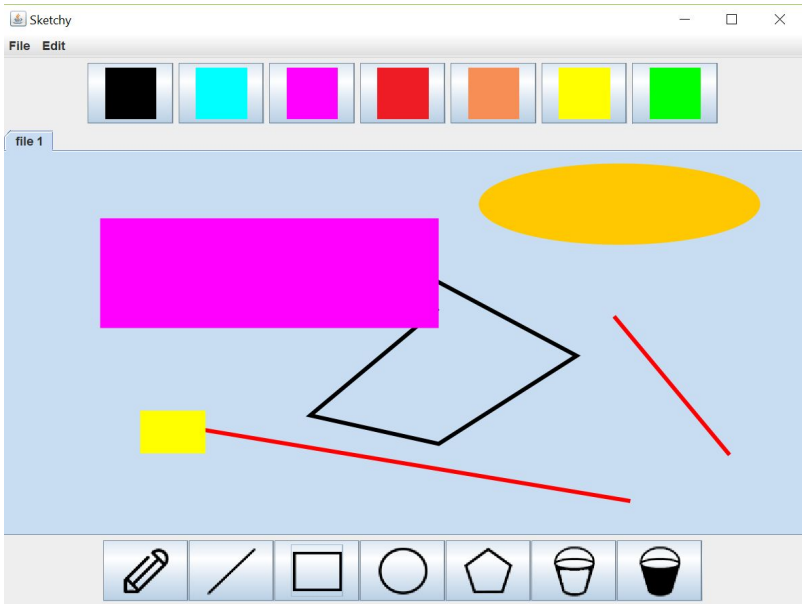
And when the user releases the mouse, the shape is drawn, and the grey outline completely disappears.



Commit 40- Fill can now be turned off

Updates:

Fill is no longer enabled by default, and can only be turned on by selecting the fill icon or the colour buttons.
However, once turned on, fill cannot be disabled until the user exits and reruns the program.



Commit 49- File Tabs, Polygon tool

Updates:

Canvas can now be opened, closed, saved and edited as individual tabs
Polygon tool added

Software Architecture

Our project consists of five class files with two classes used for objects, one is used for methods to connect the two objects, one function is used for saving and loading files and the last file consists of the gui and canvas object. The file loading function is directly connected to the drawing command class as the file is being loaded a list of drawing commands is created. This list is then turned into drawn shapes through the Object exchanger class which bridges the two objects. This bridge is needed as the canvas uses the drawn shape object to draw images. The canvas uses a wide range of buttons to change between different drawing functions. This connection is the reversed when saving the file, the drawn shape objects are the turned into drawing commands as they can easily be parsed by the file loading function that is used to save a new file.

Abstraction

Abstraction was implemented in the interface class. This was in order to allow for the creation command and property command objects to implement their own functions.

Encapsulation

Encapsulation is used extensively to avoid objects being overwritten, every shape drawn and drawing command feature obfuscated data that cannot be accessed from outside of the object.

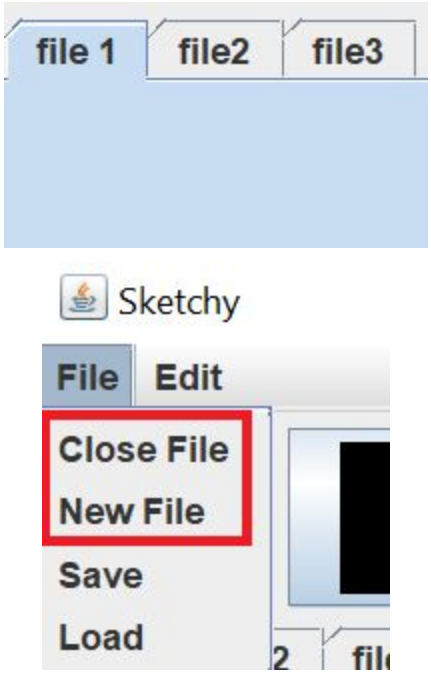
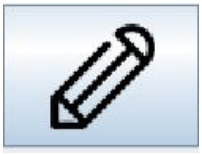
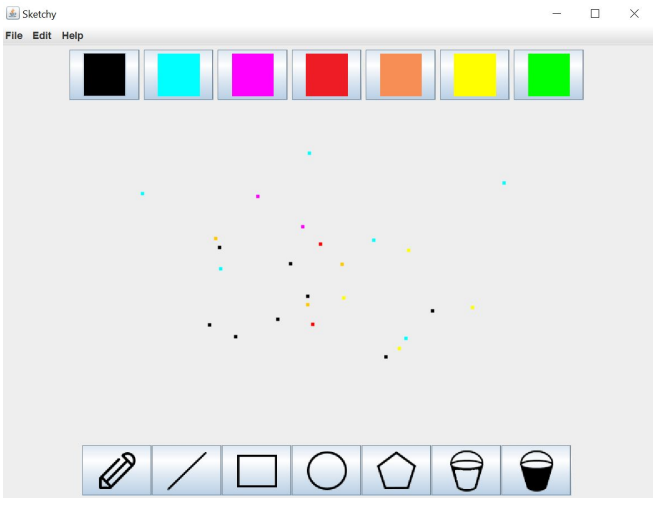
Inheritance

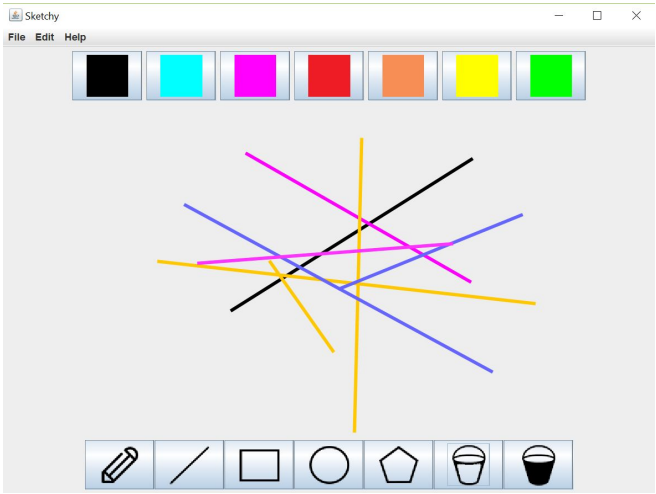
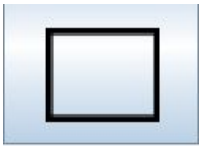
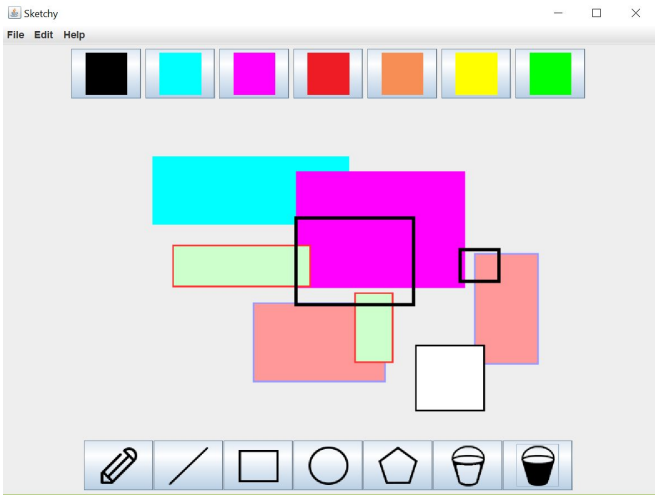
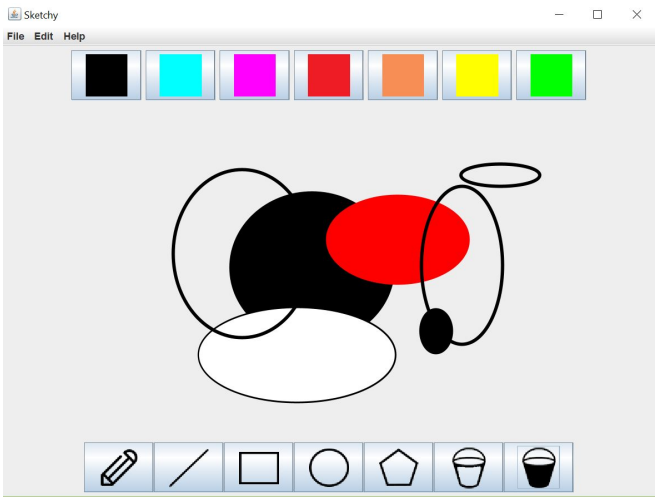
As an extension of the drawing command class, the drawnshape class inherits properties from the drawing command class. Unfortunately, due to group miscommunication this was difficult to implement.

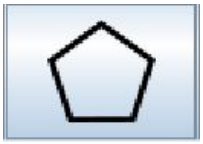
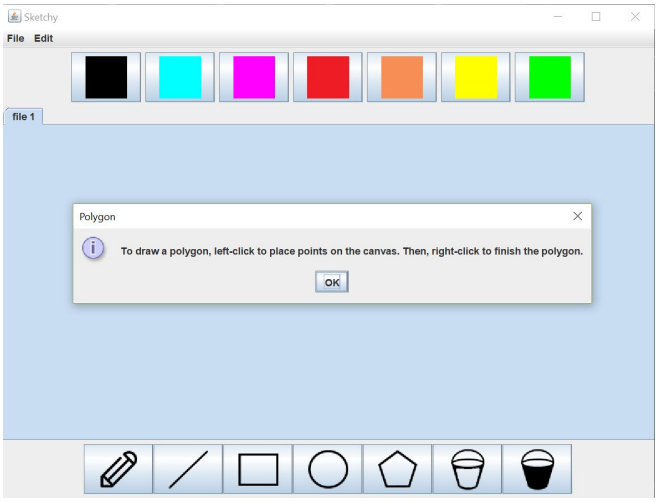
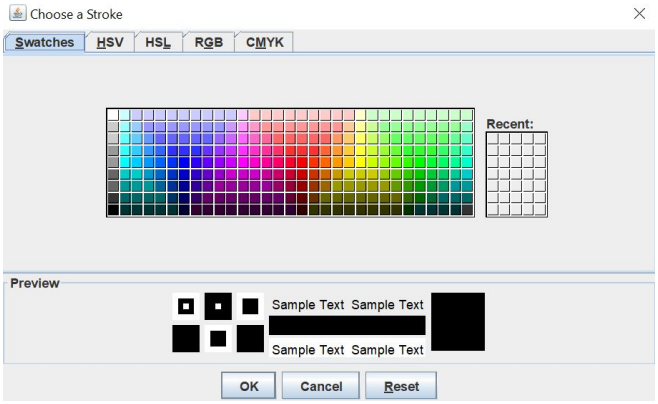
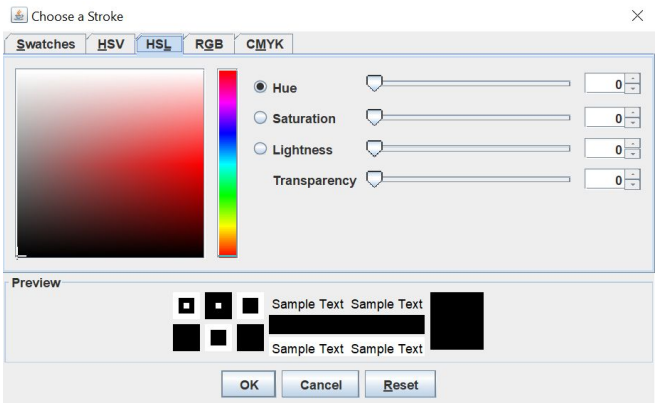
Polymorphism

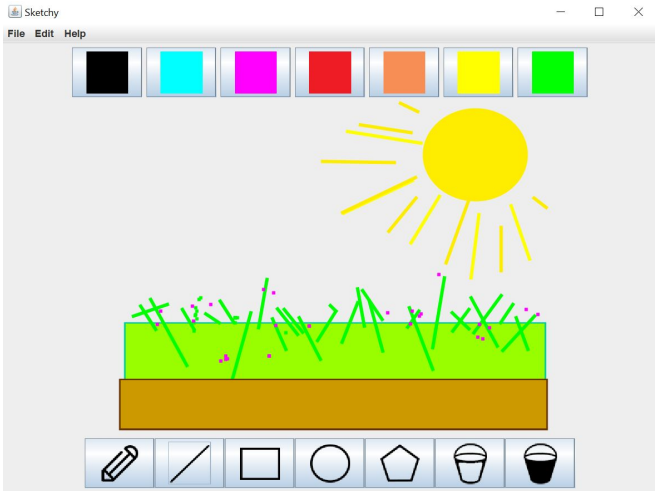
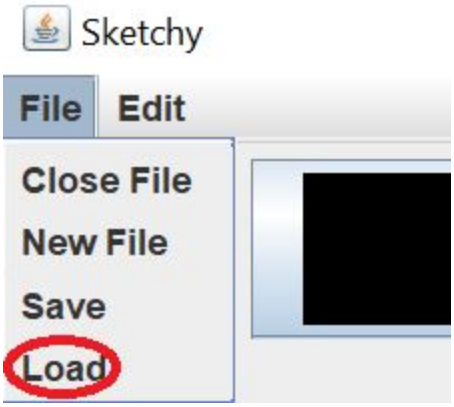
Polymorphism could have been implemented with the creation command and property command class to simplify the classes down to a single object. This could have been used to remove the interface class as well. Polymorphism wasn't implemented to allow for simplified viewing of the command structure.

Software Usage Documentation

Functionality Item	Usage guide	Example
Additional functionality	Tab functionality allows the user to draw on multiple canvases in one session, allowing them to open a new file from the menu and switch between existing tabs by clicking.	
PLOT tool 	This tool is enabled by default. Upon running the program,click anywhere on the canvas to draw a plot.To switch back to using the plot tool, click on the associated pencil icon.	

LINE tool 	This tool is enabled by clicking on the line icon. TO draw a line, click on the canvas, draw the mouse, and release. While the mouse is being dragged, a grey preview line will be shown. To update the line's colour, the user can select the colour icons at the top of the frame, or the empty bucket icon to open a colour picker window.	
RECTANGLE tool 	This tool is enabled by clicking on the rectangle icon. To draw a rectangle, click on the canvas, draw the mouse, and release. While the mouse is being dragged, a grey preview rectangle will be shown. To update the rectangle's line colour and fill colour, the user can select the colour icons at the top of the frame, or either/both bucket icons to open a colour picker window.	
ELLIPSE tool 	This tool is enabled by clicking on the circle icon. To draw an ellipse, click on the canvas, draw the mouse, and release. While the mouse is being dragged, a grey preview ellipse will be shown. To update the ellipse's line colour and fill colour, the user can select the colour icons at the top of the frame, or either/both bucket icons to open a colour picker window.	

POLYGON tool 		
LINE Color Changer Tool 	This tool is enabled by clicking on the empty bucket icon. Upon clicking, a colour picker window appears enabling the user to navigate between different menu screen in order to select any possible colour for their plot, line, or shape outline.	
FILL Color Changer Tool 	This tool works very similarly to the above, but is used to set the fill of a shape rather than the outline. This tool cannot be selected when the plot and line tools are in use, as they cannot be filled in.	

GUI Design	The GUI is designed similarly to a microsoft Paint application mockup, with big colour swatches at the top, tools at the bottom, and a large blank canvas in the centre. The user has the option to choose basic colours from the top, or choose a more specific line/fill colour by clicking on the empty/filled bucket icons. At the top left of the frame, the user can save and load VEC files, as well as undoing strokes from the edit menu.	
Undo button	From the menu, the undo button can be selected. It removes the last drawn shape, both from the screen and from the associated file.	
Loading Vec images	The functionality can be accessed via the menu at the top, by pressing the load button and navigating to the relevant .vec file.	
Saving Vec images	The functionality can be accessed via the menu at the top, by pressing the save button and navigating to the relevant .vec file.