

Computing Project 1 - Drawing Application

Report by Shaquille Muhammad Uddin



Introduction

I have chosen to work on the drawing application. I created seven new tools and one new feature. The new tools are: draw ellipse, draw rectangle, square pen, rectangle pen, ellipse pen, eraser and fill canvas. The new feature made to complement the tools is a stroke weight tool to thicken the pens.

Tools & Features

Stroke Weight Tool

In the sketch.js file's setup function, I created a variable "sw" and a select option, with the option's parent set to the header class. Using a for loop in the range of 1 to 50 with an

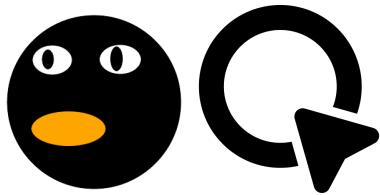
increment of 1, I implemented a feature where the stroke weight increases by 1 per selection. This is finalized in the draw function with a line of code setting the stroke weight to the select option's value, found in the application header.

Stroke Weight + Shape Size 1 ▾

Draw Shapes

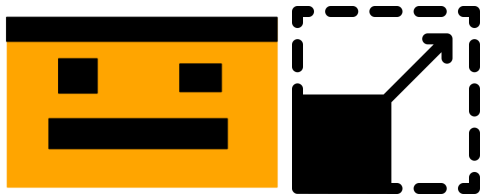
Draw Ellipse

This tool allows the user to draw an ellipse on the canvas. When the user holds their left click, a circle can be drawn and dragged into a desired place. When the left click is released, the circle is drawn on the canvas and set in place.



Draw Rectangle

The draw rectangle tool is similar to the ellipse tool, but it draws rectangular shapes instead of circles.



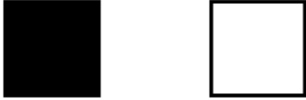
Shape Pens

Shape pens consist of three pens for clicking and dragging squares, rectangles, or circles.

Square Pen

The square pen is a tool that allows the user to draw a square on the canvas by clicking on the canvas. When the user left-clicked on the canvas, a filled square of 100px width and

100px height will be drawn on the canvas. When the user right-clicked on the canvas, a square of 100px width and 100px height will be drawn on the canvas with no fill, making it outline only.



Rectangle Pen

The rectangle pen is a tool that allows the user to draw a rectangle on the canvas by clicking on the canvas. When the user left-clicked on the canvas, a filled rectangle of 100px width and 50px height will be drawn on the canvas. When the user right-clicked on the canvas, a rectangle of 100px width and 50px height will be drawn on the canvas with no fill, making it outline only.



Ellipse Pen

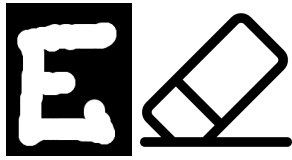
The ellipse pen has a width and height of 25 units. Similar to the other two draw-shape tools, the left click is a filled circle and the right click is a circle outline. Both stroke weight and shape size can also be controlled by the stroke weight tool.



Eraser

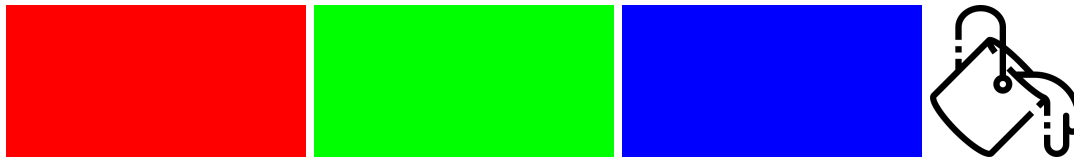
The eraser tool is a tool that allows the user to erase the canvas by clicking on the canvas. When the user left-clicked on the canvas, a filled white ellipse of 20px width and 20px

height will be drawn on the canvas.



Fill Canvas

The user must hold down on the F key and then click on the canvas to fill the canvas with colour to avoid accidental filling of the canvas. When the user left-clicked on the canvas, a filled rectangle of the size of the canvas will be drawn on the canvas.

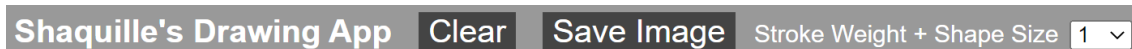


Usability Changes

I made small adjustments to the code in the HTML and CSS to better the application's presentation.

Header

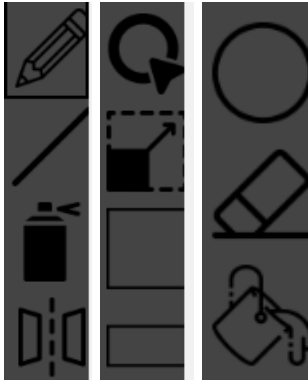
The app title now has bold text and a cleaner position. The clear and save image buttons were modified with CSS to match the app's colour scheme, complete with 0.5s transitions that include colour and opacity changes during hover and selection.



Sidebar

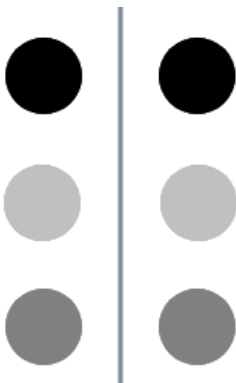
The drawing app's tools are located in the sidebar, and I used simple black .png icons to maintain consistency and clearness. To align with the app's colour scheme, I changed the

selected tool outline from blue to black.



Mirror Tool

The mirror tool line colour changed to light slate grey for better visibility on red canvas.



```
push();  
strokeWeight(3);  
stroke("LightSlateGrey");
```

Progress Update

-
- Wk 1. Stroke weight feature

```
// adds a select option to select strokeWeight
let sw = select('.header'); // selects the header class
sel = createSelect(); // creates a select option
sel.parent(sw); // sets the parent of the select option to the header class
for(var i = 1; i <= 50; i += 1) // for loop to set the value of the select option
{
  sel.option(i); // adds the value to the select option
}

}
```

// var i = 1 is the starting value of the select option
// i <= 50 is the maximum value of the select option
// i += 1 is the increment value of the select option
// sel.option(i) adds the value to the select option

```
function draw() {
  //call the draw function from the selected tool.
  //hasOwnProperty is a javascript function that tests
  //if an object contains a particular method or property
  //if there isn't a draw method the app will alert the user
  if (toolbox.selectedTool.hasOwnProperty("draw")) {
    toolbox.selectedTool.draw();
  } else {
    alert("it doesn't look like your tool has a draw method!");
  }

  strokeWeight(sel.value()); // sets the strokeWeight to the value of the select option
}
```

- Wk 2. Square Pen, Rectangle Pen

```
function squarePenTool() { // Constructor function name
  this.icon = "assets/SquarePenIcon.png"; // Image for the button
  this.name = "squarePenTool"; // Name of the tool

  this.draw = function () { // Draw the tool
    let x = mouseX; // X is the mouse position
    let y = mouseY; // Y is the mouse position
    let w = 25; // Width of the rectangle
    let h = 25; // Height of the rectangle

    if (mouseIsPressed && mouseButton == LEFT) { // If the mouse is pressed
      // Draw the square with filled background
      rect(x-12.5, y-12.5, w, h); // Draw the square with the numbers above
    } else if (mouseIsPressed && mouseButton == RIGHT) { // If the mouse is pressed with right click
      noFill(); // No fill, making it outline only
      rect(x-12.5, y-12.5, w, h); // Draw the rectangle with outline only
      fill(0); // Fill the rectangle
    }
  };
}
```

```
function rectanglePenTool() { // Constructor function name
  this.icon = "assets/RectanglePenIcon.png"; // Image for the button
  this.name = "rectanglePenTool"; // Name of the tool

  this.draw = function () { // Draw the tool
    let x = mouseX; // X is the mouse position
    let y = mouseY; // Y is the mouse position
    let w = 25; // Width of the rectangle
    let h = 15; // Height of the rectangle

    if (mouseIsPressed && mouseButton == LEFT) { // If the mouse is pressed
      // Draw the rectangle with filled background
      rect(x-12.5, y-7.5, w, h); // Draw the rectangle with the numbers above
    } else if (mouseIsPressed && mouseButton == RIGHT) { // If the mouse is pressed with right click
      noFill(); // No fill, making it outline only
      rect(x-12.5, y-7.5, w, h); // Draw the rectangle with outline only
      fill(0); // Fill the rectangle
    }
  };
}
```

- Wk 3. Ellipse Pen, Eraser

```
function ellipsePenTool() { // Constructor Function name
  this.icon = "assets/EllipsePenIcon.png"; // Image for the button
  this.name = "ellipsePenTool"; // Name of the tool

  this.draw = function () { // Draw the tool
    let x = mouseX; // X is the mouse position
    let y = mouseY; // Y is the mouse position
    let w = 25; // Width of the ellipse
    let h = 25; // Height of the ellipse

    if (mouseIsPressed && mouseButton == LEFT) { // If the mouse is pressed
      ellipse(x, y, w, h); // Draw the ellipse with the numbers above
    } else if (mouseIsPressed && mouseButton == RIGHT) { // If the mouse is pressed with right click
      noFill(); // No fill, making it outline only
      ellipse(x, y, w, h); // Draw the ellipse with outline only
      fill(0); // Fill the ellipse
    }
  }
};
}
```

```
function eraserTool() { // Constructor Function name
  this.icon = "assets/EraserToolIcon.png"; // Image for the button
  this.name = "eraserTool"; // Name of the tool

  this.draw = function () { // Draw the tool
    let x = mouseX; // X is the mouse position
    let y = mouseY; // Y is the mouse position
    let w = 15; // Width of the ellipse
    let h = 15; // Height of the ellipse

    if (mouseIsPressed && mouseButton == LEFT) { // If the mouse is pressed
      noStroke(); // No stroke for the circle
      fill(255, 255, 255) // Set the fill to white
      ellipse(x, y, w, h); // Draw the ellipse with the numbers above
    }
  }
}
```

- Wk 4. Draw Ellipse

```
function drawEllipseTool() {
  this.icon = "assets/DrawEllipseIcon.png"; // The image that will load for the tool.
  this.name = "drawEllipseTool"; // The name of the tool.

  var startMousePosX = -1; // The starting mouse position on the X axis.
  var startMousePosY = -1; // The starting mouse position on the Y axis.
  var drawing = false; // If the user is drawing or not.

  this.draw = function () {
    if (mouseIsPressed) { // If the mouse is pressed
      if (startMousePosX == -1) { // If the startMousePosX is -1
        startMousePosX = mouseX; // StartMousePosX is the mouse position
        startMousePosY = mouseY; // StartMousePosY is the mouse position
        drawing = true; // Drawing is true
        loadPixels(); // Load the pixels
      } else {
        updatePixels(); // Update the pixels
        ellipse(startMousePosX, startMousePosY, mouseX, mouseY); // Draw the ellipse
      }
    } else if (drawing) { // If the user is drawing
      drawing = false; // Drawing is false
      startMousePosX = -1; // StartMousePosX is -1
      startMousePosY = -1; // StartMousePosY is -1
    }
  };
}
```

- Wk 5. Draw Rectangle

```
function drawRectangleTool() {
  this.icon = "assets/DrawRectangleIcon.png"; // The image that will load for the tool.
  this.name = "drawRectangleTool"; // The name of the tool.

  var startMousePosX = -1; // The starting mouse position on the X axis.
  var startMousePosY = -1; // The starting mouse position on the Y axis.
  var drawing = false; // If the user is drawing or not.

  this.draw = function () {
    if (mouseIsPressed) { // If the mouse is pressed
      if (startMousePosX == -1) { // If the startMousePosX is -1
        startMousePosX = mouseX; // StartMousePosX is the mouse position
        startMousePosY = mouseY; // StartMousePosY is the mouse position
        drawing = true; // Drawing is true
        loadPixels(); // Load the pixels
      } else {
        updatePixels(); // Update the pixels
        rect(startMousePosX, startMousePosY, mouseX, mouseY); // Draw the rectangle
      }
    } else if (drawing) { // If the user is drawing
      drawing = false; // Drawing is false
      startMousePosX = -1; // StartMousePosX is -1
      startMousePosY = -1; // StartMousePosY is -1
    }
  };
}
```

- Wk 6. Fill Canvas

```
function fillCanvas() { // Constructor Function Name
  this.icon = "assets/FillCanvasIcon.png"; // Image for the button
  this.name = "fillTool"; // Name of the tool

  canvasContainer = select("#content");
  var w = canvasContainer.size().width * 10; // Get the size of the canvas
  var h = canvasContainer.size().height * 10; // Get the size of the canvas

  this.draw = function () { // Draw the tool
    if (keyIsPressed && (key === "f" || key === "F") && mouseIsPressed && mouseButton === LEFT) // Hold F and then click mouse to fills creen with colour
    {
      rect(-6, 0, w, h); // Draw over the size of canvas
    }
  };
}
```

- Wk 7. Report finalising and application evaluation

Application Limitations

- Some of the tool icons may be confusing for first-time users e.g. the shape pens.
- The spray paint tool produces increasingly bubbly strokes as the stroke weight rises.



This is stroke weights: 1, 3, 5, 7.

- The draw-shape tools start off unnecessarily big as you go further down the x and y-axis of the canvas
- With the shape pen tools, after a right-click you must choose a new colour. If not, your next left click will fill black with an outline in the originally selected colour.



- Increasing stroke weight causes right-clicked no-fill shape-pens to fill.
- The rectangle pen tool becomes more square-shaped with increasing stroke weight.
- The eraser tool sets the white colour and doesn't revert back to the previously selected colour. The user needs to manually select a colour after using the eraser to avoid drawing in white.
- The eraser tool size remains the same regardless of stroke weight.
- The fill tool can be improved by limiting the fill to a selected area rather than the entire canvas.
- Heavy brush stroke weight can cause accidental drawing on the canvas when switching to a new tool.



This is when I had the square pen and was selecting the free draw tool.

- The stroke weight selection doesn't match the header buttons.