Part A. Node Canvas

1.Where in this code does the drawing occur?

In client.js, the lines in function socket.on() is where the drawing occurs.

2.What are the inputs to the drawing function?

The input of the drawing function is the position of where the drawing will occur. The position is passing from the Arduino.

3.How can the screen be cleared?

When the arduino sends the server.js a “rst” signal, the server.js will pass this command to client.js, and invoke the ‘reset’ function.

Part B. Etch a Sketch

1.Describe which sensors you are using as inputs and why.

Potentiometers and FSR.

2.In what format does the server expect the X & Y values from the Arduino?

A String that has the format: “String(x),String(y)”.

3.What else does the Arduino need to do? Send the analogue numbers to the server by serial print.

The sensorCode folder has an almost working example.

4.What range of output X & Y values do you need?

The width and height of the screen. Is it better to map the values from the inputs to the outputs in the Arduino code, the Node code or Client code ? Why?

It’s better to map the values from the inputs to the outputs in the Client code, because the client knows the real time screen size.

5.How often do you need to be sending data from the Arduino?

When x or y input value changes.

6.Include a copy of the Arduino code, and a copy of your Server and client side code.

Arduino: https://github.com/yz1339/etch-a-sketch/blob/master/sensorCode/sensorCode.ino

Server: https://github.com/yz1339/etch-a-sketch/blob/master/server.js

Client: https://github.com/yz1339/etch-a-sketch/blob/master/public/client.js

Part C. Make it yours

1.Upload video of your Etch-a-Sketch in action, being used by someone else in the class!

https://youtu.be/xL8kopkvZaw