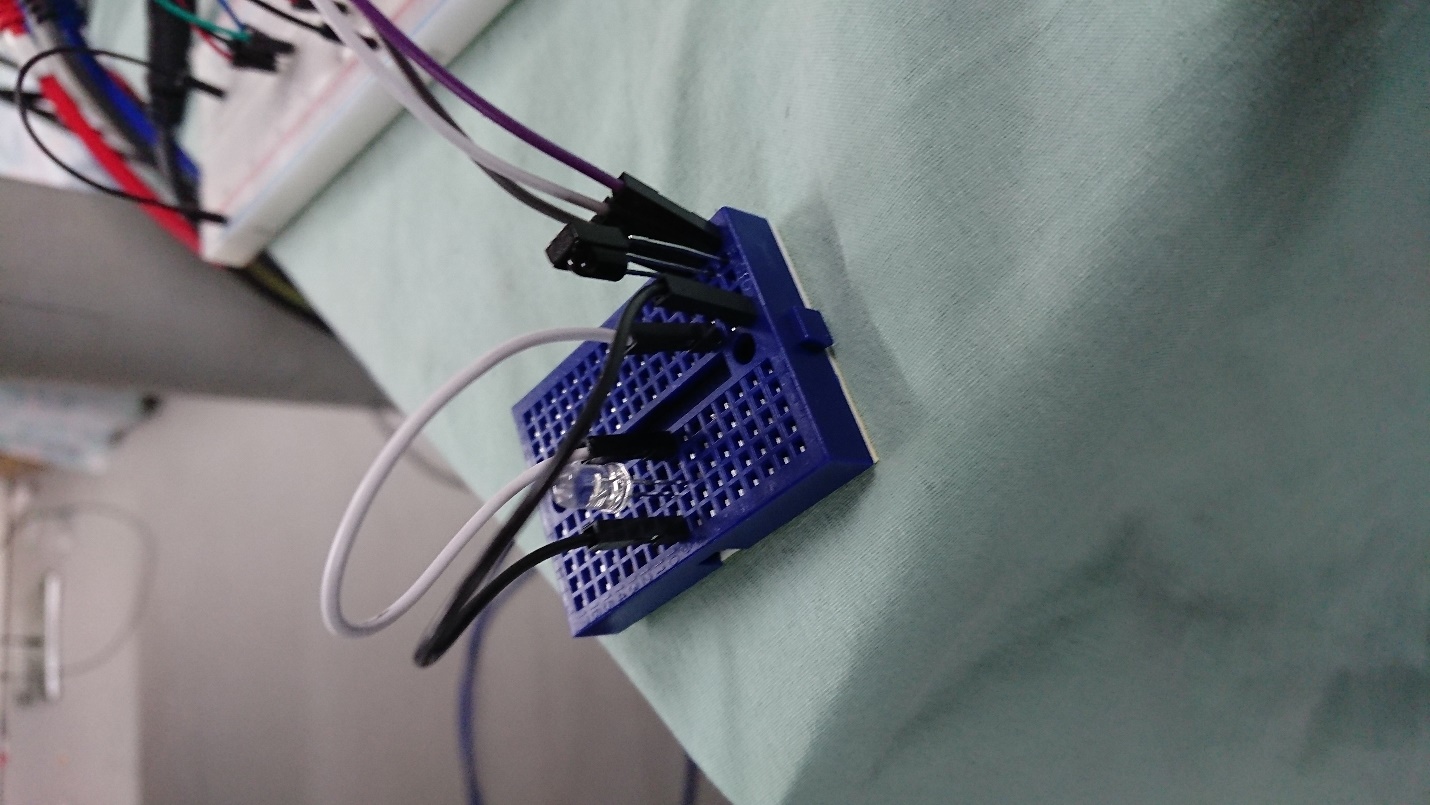
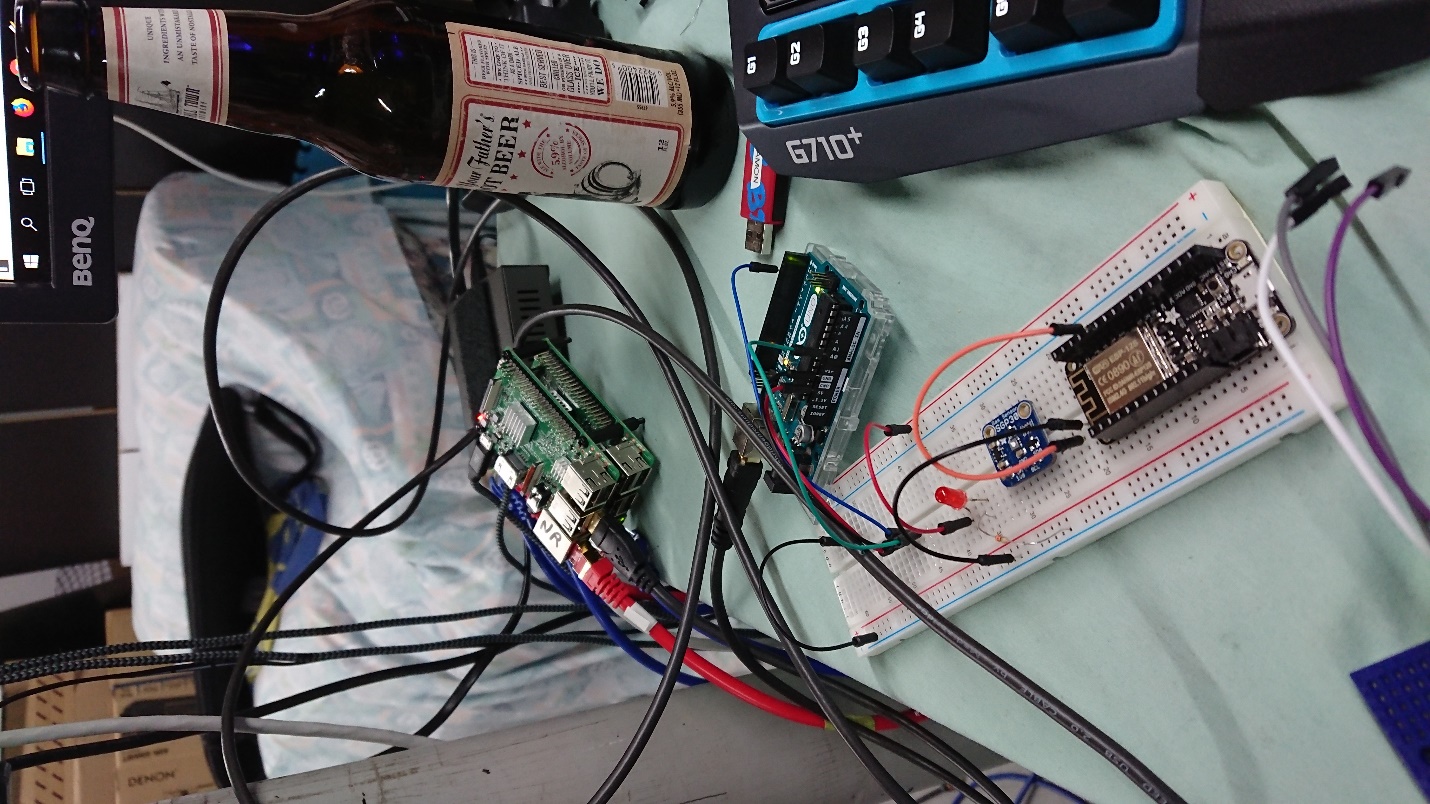
Mocked up an IR receiver with activity LED for Arduino light control



Serial communication successful serial communication between raspberry pi and Arduino to implement air quality sensor information gathering and manipulation.



\*\*picture is of an Arduino connected to the Node Red (NR) PI VIA USB Serial. The Arduino is connected to an air quality sensor.

Installation of serial com node

npm install node-red-node-serialport

reboot

….. Infinite reboot loop...

After much research it turns out that a file needs to be created at every boot instance for the serial to write to.

I had to write a bit of code and add it to the /.bashrc file so that it makes a file on boot to prevent node red from going into an infinite reboot loop on reboot. The small line is below….

sudo touch /dev/input/event0

Now that node red is working properly again I can find the USB serial communication address the below linux command was used to find appropriate serial connection device name to add to the serial input node.

for sysdevpath in $(find /sys/bus/usb/devices/usb\*/ -name dev); do ( syspath="${sysdevpath%/dev}"; devname="$(udevadm info -q name -p $syspath)"; [[ "$devname" == "bus/"\* ]] && continue; eval "$(udevadm info -q property --export -p $syspath)"; [[ -z "$ID\_SERIAL" ]] && continue; echo "/dev/$devname - $ID\_SERIAL"; ); done

This ended up being “/dev/ttyACM0” while the USB cable was plugged into the lower left USB port. The tty device seems to change with each usb port so we need to make sure that if we are using multiple Arduinos we keep track of which ports we are using and which devices plug into each port.

\*\*Note\*\* We are using a 9600 Baud Rate with 8 Data Bits for communication.

For This sprint I also assisted in the creation / assembly and painting of the model. We also started wiring and soldering sensors / LEDs.

