CHAPTER 5
IMPLEMENTATION AND TESTING

5.1 Implementation

```
1. int status_val = analogRead(val);
2.   float tegangan = status_val / 1023.0 *  5.0;
3.       float dB = tegangan * 50.0;
4.       float vdB = 36.0 + dB;
5.       String keluaran = suara + satuandB;
6.       Serial.print("tingkat kebisingan :");
7.       Serial.println(vdB);
8. delay(100);
```

Coding on line 2 serves to change the analog value obtained from the sensor into a voltage, on the line to 3 voltage obtained multiplied by 50.0 is to convert voltage to decibel value and then the final result is added 36 as the level of sensitivity of the sound sensor Source code https://wiki.dfrobot.com/Gravity__Analog_Sound_Level_Meter_SKU_SEN0232

```
9. int chk = DHT.read22(DHT22_PIN);
10.             hum = DHT.humidity;
11.             temp= DHT.temperature;
12.             Serial.print("Humidity: ");
13.             Serial.print(hum);
14.             Serial.print(" Temp: ");
15.             Serial.print(temp);
16.             Serial.println(" Celsius");
17.       delay(100);
```

This Code is used to measure laboratory temperature and humidity. Code on line 10 to get the room humidity and on line 11 to get the room temperature in Celsius units Source http://www.labelektronika.com/2016/09/dht22-sensor-suhu-dan-kelembaban-arduino.html

```
18.       hitung = constrain(hitung, 0, 60);
19.             keluar = constrain(keluar, 0, 60);
20.             jumlah = hitung - keluar;
21.             status1 = digitalRead(pinSensor);
22.             status2 = digitalRead(pinSensor2);
23.       if (status1 == HIGH && status2 == HIGH)
24.             {
25.                 keluar = keluar;
26.                 hitung = hitung;
27.                 kondisi1 = 0;
28.             }
29.       else if (status1 == LOW && kondisi1 == 0
          && status2 == HIGH)
30.             {
31.                 hitung += 1;
32.                 kondisi1 = 1;
33.             }
34.       else if (status1 == HIGH && kondisi1 == 0
          && status2 == LOW)
35.             {
36.                 keluar += 1;
```

```
37.                 kondisi1 = 1;
38.             }
39.       else if (status1 == LOW && kondisi1 == 1)
40.             {
41.                 hitung = hitung;
42.                 kondisi1 = 1;
43.             }
44.             if (jumlah < 0)
45.             {
46.                 hitung = 0;
47.                 keluar = 0;
48.             }
49.             else
50.             {
51.                 hitung = hitung;
52.             }
53.             Serial.print("jumlah : ");
54.             Serial.println(jumlah);
55.       delay(100);
```

Coding on line 20 serves to count the number of people in the lab. On line 29 serves to give the condition when proximity 1 die and proximity 2 turns on then the number of people entering will increase (hitung). On line 34 is condition when proximity 1 turn on and proximity 2 die then the number of people coming out will increase (keluar). At line 40 use to reset calculation to 0 if the result from line 20 is less than 0 Sourcecode https://nofgipiston.wordpress.com/2017/03/05/membuat-alat-penghitung-barang-otomatis-menggunakan-arduino-dan-sensor-jarak-infra-merah/.

5.2 Testing

Testing done on this project is to change the value of the voltage obtained by the sound sensor to sound level with the Decibell unit. The first step changes the analog value from the sound sensor to a voltage using the program

```
Tingkat Kebisingan :356
Tingkat Kebisingan :376
Tingkat Kebisingan :420
Tingkat Kebisingan :472
Tingkat Kebisingan :533
Tingkat Kebisingan :576
Tingkat Kebisingan :614
Tingkat Kebisingan :654
Tingkat Kebisingan :680
Tingkat Kebisingan :722
Tingkat Kebisingan :758
Tingkat Kebisingan :797
Tingkat Kebisingan :835
Tingkat Kebisingan :867
Tingkat Kebisingan :898
Tingkat Kebisingan :900
Tingkat Kebisingan :912
Tingkat Kebisingan :911
Tingkat Kebisingan :905
Tingkat Kebisingan :901
Tingkat Kebisingan :864
```

Illustration 5.1: Analog value of the sound sensor

```
Tingkat Kebisingan :0.39
Tingkat Kebisingan :0.01
Tingkat Kebisingan :0.85
Tingkat Kebisingan :0.40
Tingkat Kebisingan :0.31
Tingkat Kebisingan :0.00
Tingkat Kebisingan :0.36
Tingkat Kebisingan :0.48
Tingkat Kebisingan :0.31
Tingkat Kebisingan :1.17
Tingkat Kebisingan :0.40
Tingkat Kebisingan :0.00
Tingkat Kebisingan :0.53
Tingkat Kebisingan :0.00
Tingkat Kebisingan :0.39
Tingkat Kebisingan :0.05
Tingkat Kebisingan :1.16
Tingkat Kebisingan :1.21
```

Illustration 5.2: Analog to voltage conversion

Data on the left shows the analog value of the sound sensor, data to the right shows the value of the voltage obtained from the analog value. This voltage value is used to determine the level of noise that can be obtained from the sensor.

Second step after converting to voltage, insert algorithm to change voltage to Decibell,

```
Tingkat Kebisingan :36.00
Tingkat Kebisingan :36.00
Tingkat Kebisingan :52.37
Tingkat Kebisingan :68.99
Tingkat Kebisingan :50.17
Tingkat Kebisingan :36.98
Tingkat Kebisingan :92.21
Tingkat Kebisingan :88.54
Tingkat Kebisingan :36.00
Tingkat Kebisingan :55.55
Tingkat Kebisingan :70.70
Tingkat Kebisingan :74.86
Tingkat Kebisingan :48.46
```

Illustration 5.3: Noise level

Data from illustrations 5.3 is a value of noise level gained from the algorithm in decibels. The algorithm used in this project is

```
1. float tegangan = status_val / 1023.0 *   5.0;
2.      float dB = tegangan * 50.0;
3.      float vdB = 36.0 + dB;
```

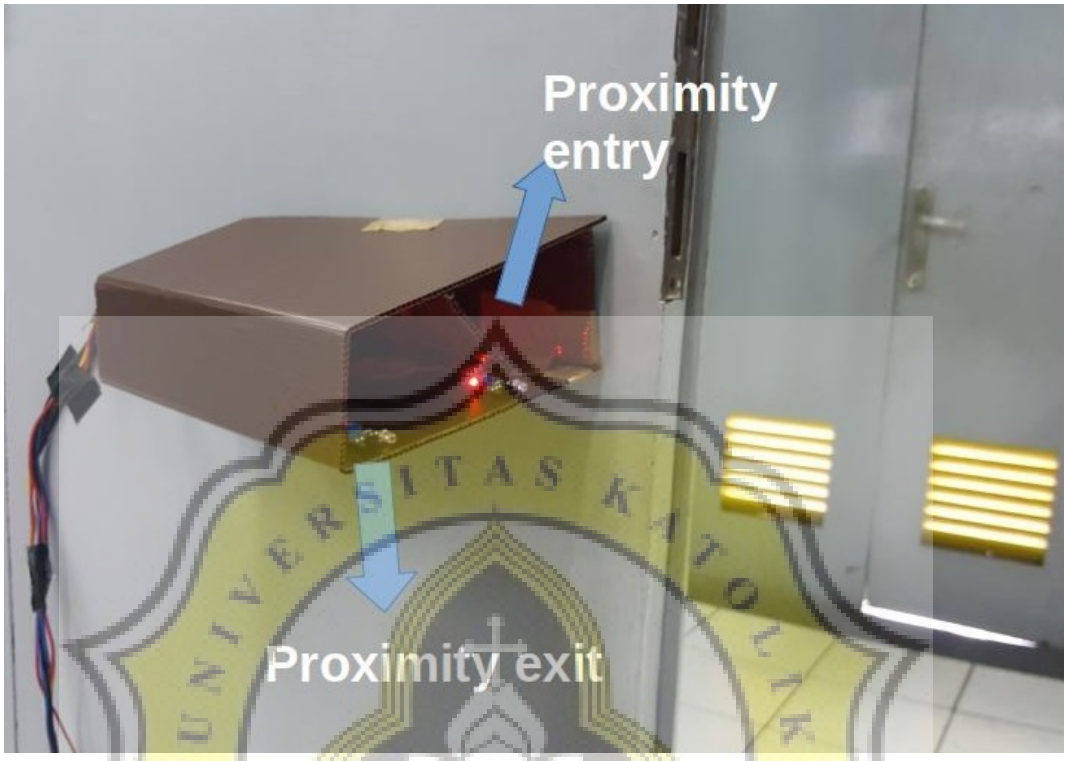

Illustration 5.4: proximiy entry

Proximity sensor entry detects people who walk in computer lab Proximity sensor entry lit up and followed by the proximity sensor exit lit, in this condition the number of people entering will increase. When people walk out the lab proximity sensor exit will illuminate first and followed proximity entry sensors, in this condition the number of people who come out increases. The total number of people in the laboratory is the total number of people who came out minus the total number of people who entered.

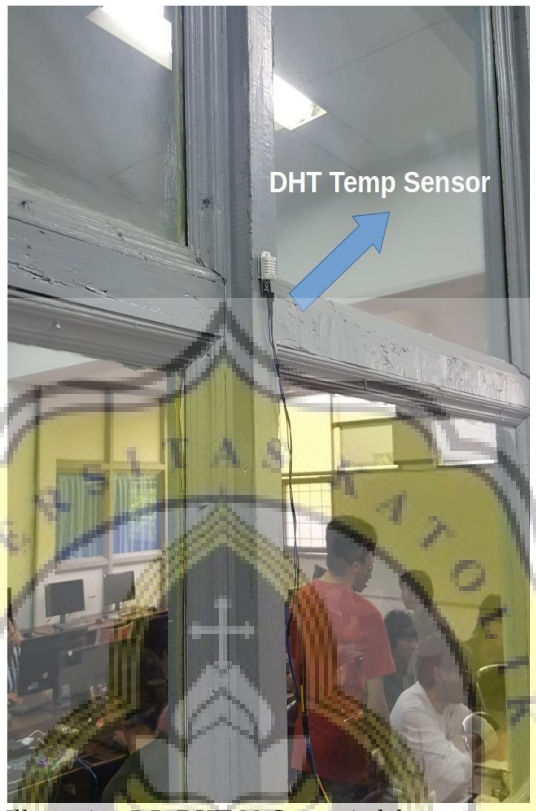

Illustration 5.5: DHT 22 Sensor in lab

The DHT sensor is on the left side of the center of the computer lab room. Temperature sensors get the value of room temperature and humidity level in the computer lab. The temperature in the computer is very influential with the number of people in the room, the more people then the temperature is getting hot.

Table 5.1: Table of temperature and the number of students

Suhu	Kelembaban	Jumlah Orang
26.5	54.5	1
27.7	51.2	4
28.3	45.8	12
28.8	51.6	20

Rata-Rata Tingkat Kebisingan

46.03 dB

Rata-Rata Suhu

28.10

Show 10 entries Search:

No	jam	tanggal	suara	suhu	kelembapan	jumlah
31	11:03:18	2019/06/24	40.15	27.3	52.7	15
32	11:04:13	2019/06/24	39.42	27.5	52.1	17
33	11:04:53	2019/06/24	39.42	27.6	51.6	17
34	11:04:58	2019/06/24	39.91	27.6	51.6	18
35	11:05:03	2019/06/24	39.42	27.6	51.6	18

Illustration 5.6: web view

The picture above is a view on the website displaying the sensor value already installed. In that view can see at what time students use labs, temperature in the laboratory and the level of noise caused. In the Web view it also displays the average noise and temperature levels in the laboratory of all collected data

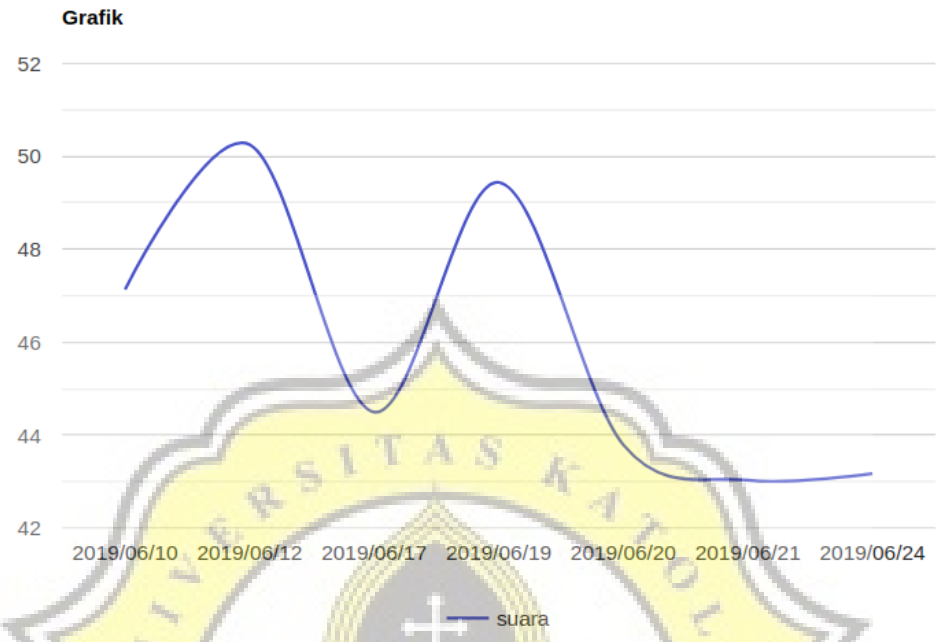

Illustration 5.7: graph data collect

The graph above represents a seven-day average noise level from the data collection process. If the chart shows the value above 46 then the lab conditions can be in a crowded category. When the graph shows below 46 then the lab condition is dicataloged normally. The graph displays the average noise level based on the day when the lab is used

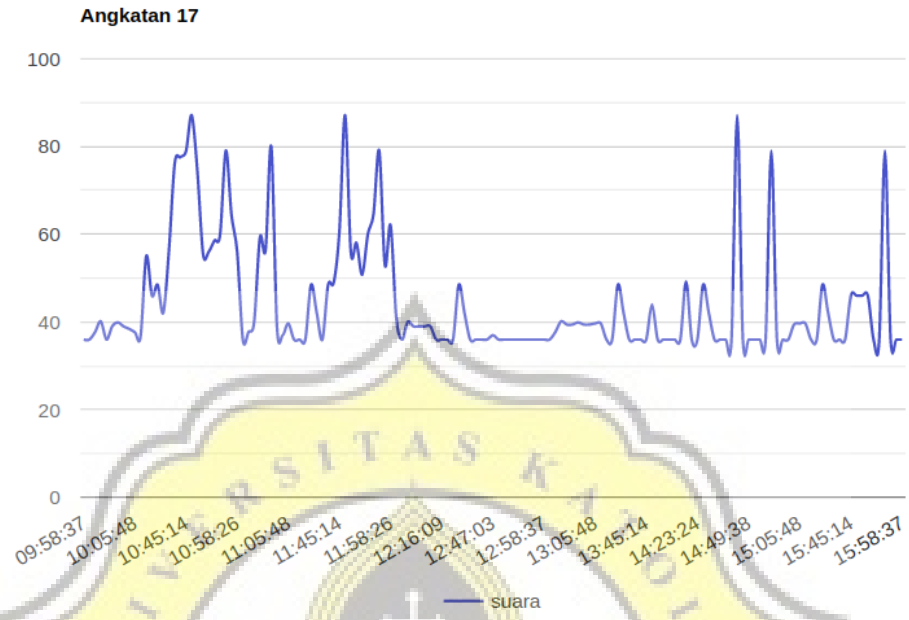

Illustration 5.8: graph noise level

The chart above shows the level of noise against the use of computer labs on a particular day where increased noise levels are influenced by student activities in the lab computer. The graph displays noise levels based on the hour when the lab is used