

CHAPTER 5

IMPLEMENTATION AND TESTING

5.1 Implementation

5.1.1 Source Code sensorKaligarang.ino

The code contained in the arduino ide with the name sensorKaligarang.ino. Sensor will read data and will be sent to the database using connection.php for authentication and data.php to enter data into database after authentication.

5.1.2 Library and Declaring Variable

```

1. #include <SPI.h>
2. #include <Ethernet.h>
3. #include <DHT.h>
4. #include <TinyGPS++.h>
5. #include <SoftwareSerial.h>
6. #define DHTPIN 7
7. #define DHTTYPE DHT2

```

The line 1-7 above is for including library of Arduino Ethernet Shield, DHT22 sensor, NEO GPS 6M what will be need and declaring variable, pin that will be use.

5.1.3 Initializing

```

8. void setup() {
9. Serial.begin(9600);
10. serial_gps.begin(9600);
11. Ethernet.begin(mac, ip);
12. dht.begin();
13. pinMode(trigger, OUTPUT);
14. pinMode(echo, INPUT);
15. }

```

The code above is function for initializing variable and also starting libraries that has been called. Basicly those code above is used for setuping everything

we need to start the sketch. Line 9 is used to sets the data rate in bits per second (baud) for serial data transmission. It does same for line 10-12 except each of it has their own parameter, not only for setting bits per second. Like line 11 need to set up its mac and ip address to begin and line 12 no need to set up but still need to be initiate. While lane 13-14 is used for setuping the input and output system of ultrasonic sensor.

5.1.4 Execute and Looping

```

16. void loop() {
17.     while(serial_gps.available()) {
18.         gps.encode(serial_gps.read());
19.     }
20.     if(gps.location.isUpdated()) {
21.         latitude = gps.location.lat();
22.         slongitude = gps.location.lng();
23.     }
24.     float humidity = dht.readHumidity();
25.     float temperature = dht.readTemperature();
26.     float tegangan_turbidity = (float) (val*(5.00/1024.0));
27.     float NTU = (1000-(tegangan_turbidity/4.22)*1000);
28.     String lokasi = String(latitude)+"," + String(longitude);

```

The code above is function that used for execute commands that has been made before, and will continuously do those commands as long as it get power supply. This function will reading input or change the output, include connecting and sending values to db kaligarangIkom. For detailed information about the code above :

1. Line 17-19 is for command gps sensor to continuously reading data as long it is available
2. Line 20-23 is for set data of latitude and longitude of location into variable

3. Line 24 to get data of current humidity and put into variable, and also line 25 to get data of current temperature and put it into variable

4. Line 26 to calibrate and get the value of the voltage of turbidity sensor. It can be searched by analog value that has been read times 5 as maximum value of volt and divided by 1024 as the range of analog value.

5. Line 27 is to calibrate turbidity sensor. Getting voltage value of turbidity sensor that has been gotten before and convert it into one measurement (NTU).

5.1.5 Data Store

After sensorKaligarang.ino compiled and running. The read result of the sensor will be sent using PHP and the data will stored in the database kaligarangkom in the table data. Each data entered in the column temperature, humidity, tinggi, location, ntu in accordance with the commands that have been written on the file sensorKaligarang.ino.

id	event	temperatur	kelembaban	ph	volt	tinggi	lokasi	teg	ntu
2080	2018-07-27 00:32:26	29.10	58.70	3.43	3.14	0.53	0.00,0.00	3.13	260.87
2083	2018-07-27 00:33:14	29.10	58.20	3.46	3.14	0.77	0.00,0.00	3.15	255.10
2092	2018-07-27 00:56:41	29.70	57.70	1.12	3.55	0.77	0.00,0.00	3.57	155.83
2107	2018-07-27 01:58:04	30.10	58.20	6.14	2.67	0.77	0.00,0.00	2.64	377.46
2113	2018-07-27 01:58:41	30.00	58.30	6.32	2.64	0.77	0.00,0.00	2.69	365.92
2119	2018-07-27 01:59:18	29.80	57.10	6.22	2.65	0.77	0.00,0.00	2.65	374.00
2126	2018-07-27 02:00:01	29.70	57.20	6.29	2.64	0.78	0.00,0.00	2.70	363.61
2127	2018-07-27 02:00:07	29.70	57.50	6.26	2.65	0.77	0.00,0.00	2.63	378.62
2133	2018-07-27 02:00:44	29.50	57.10	6.37	2.63	0.77	0.00,0.00	2.62	380.02
2139	2018-07-27 02:01:21	29.40	57.20	6.39	2.62	0.77	0.00,0.00	2.66	371.69

Illustration 5.1: Picture Display Database kaligarangkom

The picture show the table, the name of table is data of database kaligarangkom that will use to store the data sensors.

5.1.6 PHP File

```
1 <?php
2 // Parameters for MySQL database
3 $host = "localhost"; // Name host
4 $user = "root";      // Username MySQL
5 $pass = "";         // Password MySQL
6 $namedb = "kaligarangIkon"; // Database name
7
8 // Create a connection to the MySQL database
9 $conn = mysqli_connect($host, $user, $pass);
10 $db = mysqli_select_db ($conn, $namedb );
11
12 ?>
```

Illustration 5.2: Picture Display connection.php File

This file use to connect the kaligarangIkom database. Host, user, pass to authentication and connected to kaligarangIkom database.

```

1  <?php
2  include ('connection.php');//include data to connection.php
3  $sql_insert = "INSERT INTO data (temperatur, kelembaban, ph, volt,
tinggi, lokasi, teg, ntu) VALUES ('".$_GET["temperatur"]."', '".$_GET
["kelembaban"]."', '".$_GET["ph"]."', '".$_GET["volt"]."', '".$_GET
["tinggi"]."', '".$_GET["lokasi"]."', '".$_GET["teg"]."', '".$_GET
["ntu"]."')";//Insert data to table
4  if(mysqli_query($conn,$sql_insert))
5  {
6  echo "Done";
7  mysqli_close($conn);
8  //if the data successfully in the input then printed "Done"
9  }
10 else
11 {
12 echo "error is ".mysqli_error($conn);
13 //if the data not connect in the input then printed "error is"
14 }
15 ?>

```

Illustration 5.3: Picture Display data.php File

This file use to send data from arduino sensors into table data of kaligarangIkom database.

5.2 Testing

5.2.1 Time Setting

In this section will display the output data that has been stored on the kaligarangIkom database in the table data. To display data, input manually Year, Month, Date, Hour, Minute, and second in the Waktu Mulai and the Waktu Selesai column. Click Tampilkan button to display data in chart form.

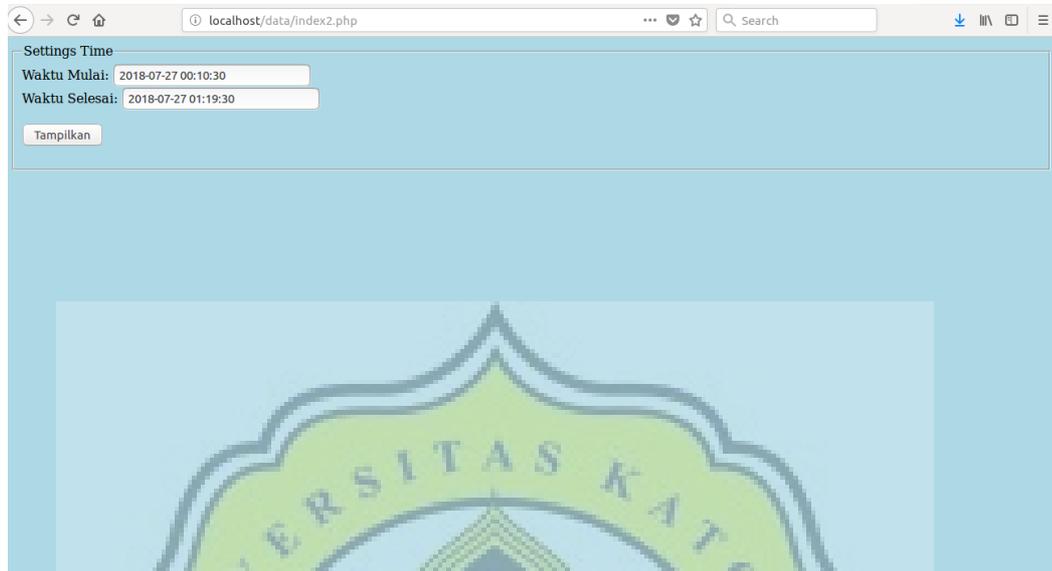
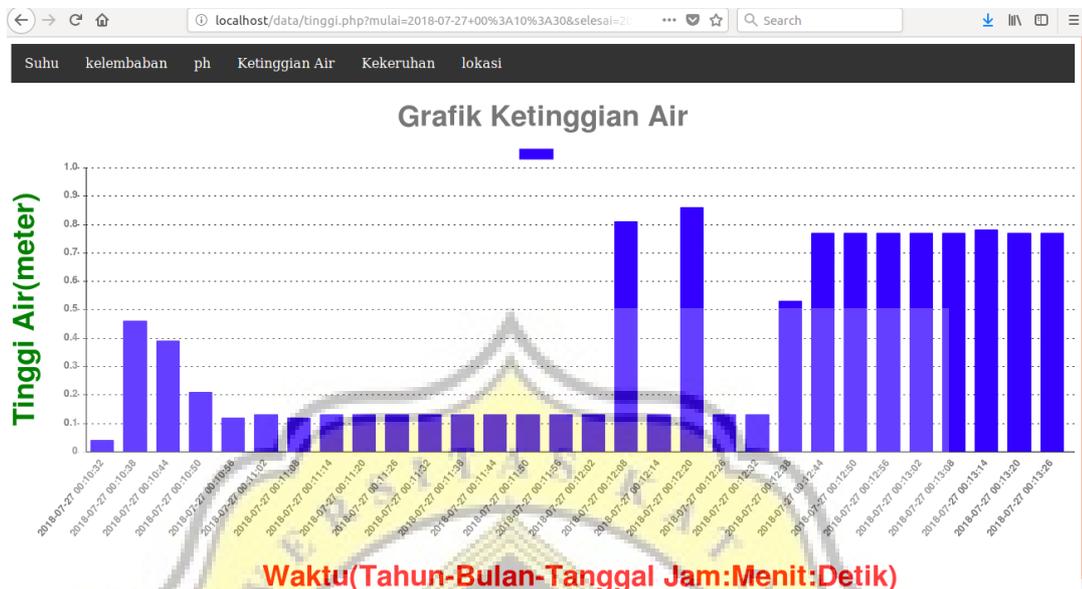


Illustration 5.4: Picture Display Time Setting

The picture above to set the time interval of the data to be displayed. Waktu Mulai is for beginning, and Waktu Selesai to determine the final limit of retrieving the data.

5.2.2 Chart

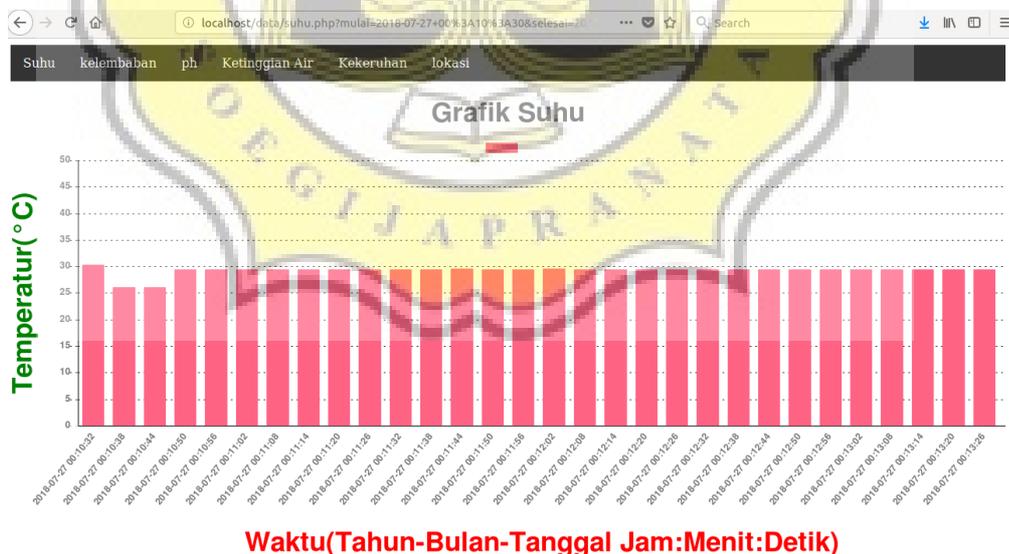
There are four charts showing the output of the table data in the temperature, humidity, tinggi, location, ntu columns from the kaligarangIkom database. The y line on the chart show the sensor value, the x line on the chart show the time of data that goes into database every five seconds. Color differences on the chart, to distinguish the sensors used. Except turbidity that use the same color as the temperature chart, this does not affect the output results.



Waktu(Tahun-Bulan-Tanggal Jam:Menit:Detik)

Illustration 5.5: Picture Display Water Level Chart

The picture above is a chart Ketinggian Air that displays data from kaligarangkom database, table data and column tinggi, to show the water level in meter.



Waktu(Tahun-Bulan-Tanggal Jam:Menit:Detik)

Illustration 5.6: Picture Display Temperature Chart

The picture above is a chart Suhu that displays data from kaligarangkom database, table data and column suhu, to show the temperature Celcius.

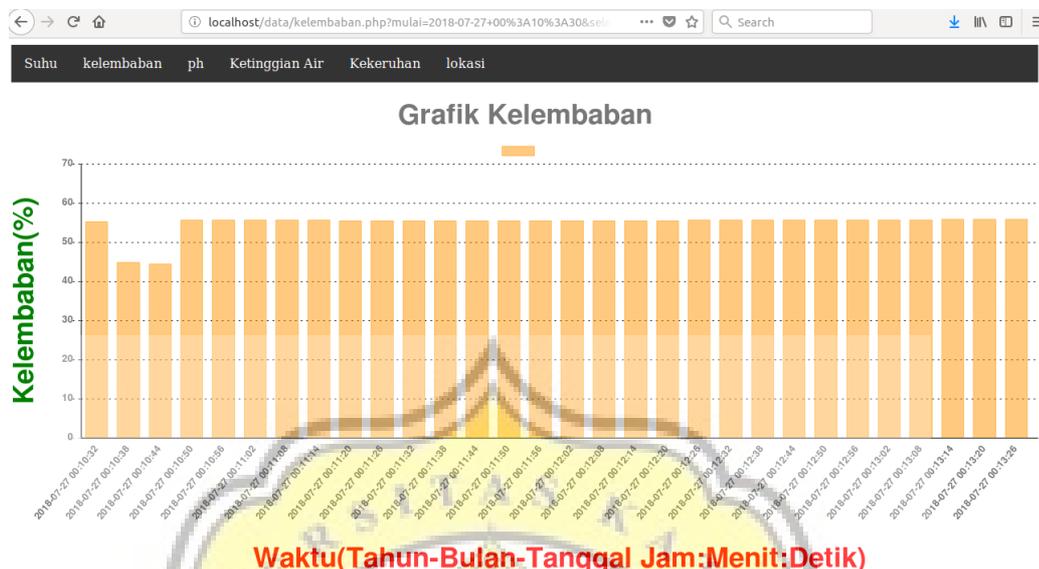


Illustration 5.7: Picture Display Humidity Chart

The picture above is chart that displays data from kaligarangIkom database, table data and column humidity, to show the humidity value.



Illustration 5.8: Picture Display Turbidity-Value Chart

The picture above is chart that displays data from kaligarangIkom database, table data and column keruh, to show the turbidity value in NTU.

5.2.3 Location

In the section, the location will be displayed basen on <https://www.google.com/maps/place/> input from PHP as well asadditional input from the GPS sensor. It will redirect to a new page where the map appear from Googlemaps with latitude, longitude coordinates.

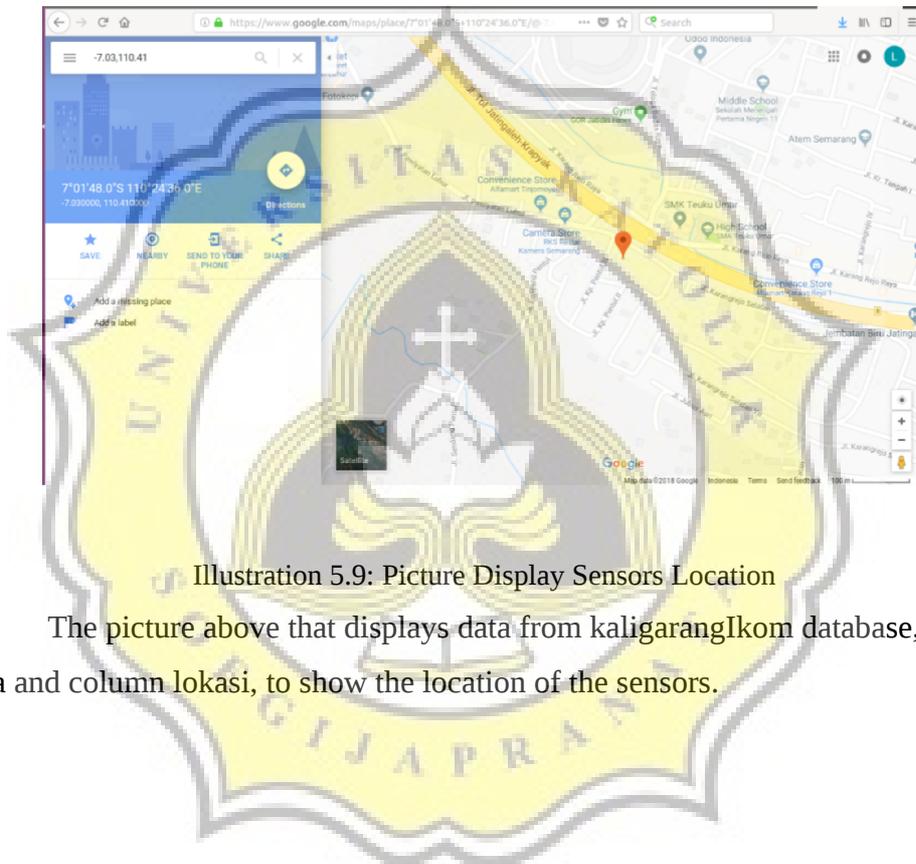


Illustration 5.9: Picture Display Sensors Location

The picture above that displays data from kaligarangIkom database, table data and column lokasi, to show the location of the sensors.