

LAMPIRAN

LAMPIRAN A DOKUMENTASI PENGUJIAN



Gambar 1 Tampilan LCD Kalibrasi Flowsensor



Gambar 2 Tampilan botol air 500ml Kalibrasi Flowsensor



Gambar 3 Tampilan Kondisi Awal Meteran Konvensional Saat Kalibrasi



Gambar 4 Tampilan Kondisi Akhir Meteran Konvensional Saat Kalibrasi



Gambar 5 Tampilan Botol Air 500mL Kalibrasi Meteran Konvensional

11:39:31.560 -> Flow Rate: 0.17 L/min

Gambar 6 Kecepatan Terbaca Paling Rendah

```
COM3
16:15:34.223 -> 0.0000 L
16:15:34.223 -> Changed connections
16:15:34.223 -> New Connection, nodeId = 1293581979
16:15:34.307 -> Received from 1293581979 msg={"node":3,"vol":0.0008453046903014183}
16:15:34.307 -> Node: 3
16:15:34.307 -> Volume: 0.00
16:15:35.260 -> {"Node":"nodeNumber","Volume":TOTAL}Node: 1
16:15:35.260 -> Volume: 0.00
16:15:37.249 -> VOL. :0.00
16:15:37.249 -> TOTAL:
16:15:37.249 -> 0.0000 L
```

Gambar 7 Pengujian Jaringan Kondisi 1

```
COM3
17:08:08.082 -> Changed connections
17:08:08.082 -> New Connection, NodeId = 1298103367
17:08:11.128 -> Message received from 1298103367
17:08:11.128 -> Node: 2 Volume: 0.000000
17:08:14.127 -> Adjusted time 163184434. Offset = 5184043
17:08:17.173 -> Message received from 1298103367
17:08:17.173 -> Node: 2 Volume: 0.000000
17:08:20.173 -> Message received from 1293581979
17:08:20.220 -> Node: 3 Volume: 0.000000
17:08:26.265 -> Adjusted time 175315887. Offset = 9134
17:08:29.264 -> Message received from 2229147199
17:08:29.311 -> Node: 4 Volume: 0.000000
17:08:32.310 -> Message received from 1298103367
17:08:32.310 -> Node: 2 Volume: 0.000000
17:08:35.357 -> Message received from 2229147199
17:08:35.357 -> Node: 4 Volume: 0.000000
 Autoscroll  Show timestamp
```

Gambar 8 Pengujian Jaringan Kondisi 2

LAMPIRAN B TABEL HASIL PENGUJIAN

Lampiran B.1 Tabel 1 *Monitoring* Penggunaan Air Hari Ke-1

No.	Waktu	Node ID							
		Node 2		Node 3		Node 4		Node 5	
		Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)
1	00:00	0	0	0	0	0	0	0	0
2	01:00	0	0	0	0	0	0	0	0
3	02:00	0	0	0	0	0	0	0	0
4	03:00	0	0	0	0	0	0	0	0
5	04:00	0	0	0	0	0	0	0	0
6	05:00	0	0	0	0	0	0	0	0
7	06:00	0	0	0	0	0	0	0	0
8	07:00	0	0	0	0	0.0281	64.66	0	0
9	08:00	0.0432	99.45	0.0274	62.91	0.0760	174.71	0	0
10	09:00	0.0432	99.45	0.0274	62.91	0.0760	174.71	0	0
11	10:00	0.0693	159.31	0.0511	117.63	0.0760	174.71	0	0
12	11:00	0.0693	159.31	0.0720	165.57	0.0760	174.71	0	0
13	12:00	0.0693	159.31	0.0720	165.57	0.0760	174.71	0.0265	60.91
14	13:00	0.0693	159.31	0.0720	165.57	0.0760	174.71	0.0265	60.91
15	14:00	0.0693	159.31	0.0720	165.57	0.0760	174.71	0.0265	60.91
16	15:00	0.0693	159.31	0.0720	165.57	0.0760	174.71	0.0265	60.91
17	16:00	0.0693	159.31	0.0720	165.57	0.0760	174.71	0.0265	60.91
18	17:00	0.0984	226.36	0.0720	165.57	0.0760	174.71	0.0265	60.91
19	18:00	0.0984	226.36	0.0720	165.57	0.0760	174.71	0.0265	60.91
20	19:00	0.0984	226.36	0.0720	165.57	0.0760	174.71	0.0265	60.91
21	20:00	0.0984	226.36	0.0720	165.57	0.0760	174.71	0.0265	60.91
22	21:00	0.0984	226.36	0.0720	165.57	0.0760	174.71	0.0265	60.91
23	22:00	0.0984	226.36	0.0720	165.57	0.0760	174.71	0.0265	60.91
24	23:00	0.0984	226.36	0.0720	165.57	0.0760	174.71	0.0265	60.91

Lampiran B.2 Tabel 2 *Monitoring* Penggunaan Air Hari Ke-2

No.	Waktu	Node ID							
		Node 2		Node 3		Node 4		Node 5	
		Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)
1	00:00	0	0	0	0	0	0	0	0
2	01:00	0	0	0	0	0	0	0	0
3	02:00	0	0	0	0	0	0	0	0
4	03:00	0	0	0	0	0	0	0	0
5	04:00	0	0	0	0	0	0	0	0
6	05:00	0	0	0	0	0	0	0	0
7	06:00	0	0	0	0	0	0	0	0
8	07:00	0.0099	22.77	0	0	0.0489	112.51	0	0
9	08:00	0.0503	115.71	0	0	0.0489	112.51	0.0685	157.55
10	09:00	0.0503	115.71	0.0805	185.15	0.0489	112.51	0.1768	406.68
11	10:00	0.0503	115.71	0.0805	185.15	0.0489	112.51	0.1768	406.68
12	11:00	0.0503	115.71	0.0805	185.15	0.0489	112.51	0.1768	406.68
13	12:00	0.0503	115.71	0.0805	185.15	0.0489	112.51	0.1768	406.68
14	13:00	0.0503	115.71	0.0805	185.15	0.0489	112.51	0.1768	406.68
15	14:00	0.0503	115.71	0.0805	185.15	0.0489	112.51	0.1768	406.68

16	15:00	0.0503	115.71	0.0805	185.15	0.0489	112.51	0.1768	406.68
17	16:00	0.0847	194.81	0.0805	185.15	0.0489	112.51	0.1768	406.68
18	17:00	0.1009	232.14	0.0805	185.15	0.0847	194.81	0.1890	434.70
19	18:00	0.1009	232.14	0.0805	185.15	0.0847	194.81	0.1890	434.70
20	19:00	0.1009	232.14	0.0805	185.15	0.0847	194.81	0.1890	434.70
21	20:00	0.1009	232.14	0.0805	185.15	0.0847	194.81	0.1890	434.70
22	21:00	0.1009	232.14	0.0805	185.15	0.0847	194.81	0.1890	434.70
23	22:00	0.1009	232.14	0.0805	185.15	0.0847	194.81	0.1890	434.70
24	23:00	0.1009	232.14	0.0805	185.15	0.0847	194.81	0.1890	434.70

Lampiran B.3 Tabel 3 *Monitoring* Penggunaan Air Hari Ke-3

No.	Waktu	Node ID							
		Node 2		Node 3		Node 4		Node 5	
		Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)
1	00:00	0	0	0	0	0	0	0	0
2	01:00	0	0	0	0	0	0	0	0
3	02:00	0	0	0	0	0	0	0	0
4	03:00	0	0	0	0	0	0	0	0
5	04:00	0	0	0	0	0	0	0	0
6	05:00	0	0	0	0	0	0	0	0
7	06:00	0	0	0	0	0.031	71.3	0	0
8	07:00	0	0	0	0	0.031	71.3	0	0
9	08:00	0.0235	54.15	0.0301	69.23	0.031	71.3	0.0161	37.03
10	09:00	0.0612	140.76	0.0301	69.23	0.031	71.3	0.0161	37.03
11	10:00	0.0612	140.76	0.0301	69.23	0.031	71.3	0.0161	37.03
12	11:00	0.0612	140.76	0.0301	69.23	0.031	71.3	0.0161	37.03
13	12:00	0.0612	140.76	0.0301	69.23	0.031	71.3	0.0161	37.03
14	13:00	0.0612	140.76	0.0301	69.23	0.031	71.3	0.0161	37.03
15	14:00	0.0612	140.76	0.0301	69.23	0.031	71.3	0.0161	37.03
16	15:00	0.0612	140.76	0.0301	69.23	0.031	71.3	0.0161	37.03
17	16:00	0.0740	170.26	0.0493	113.39	0.031	71.3	0.0161	37.03
18	17:00	0.0988	227.28	0.0838	192.74	0.0752	173.2	0.0275	63.25
19	18:00	0.0988	227.28	0.0838	192.74	0.0752	173.2	0.0275	63.25
20	19:00	0.0988	227.28	0.0838	192.74	0.0752	173.2	0.0275	63.25
21	20:00	0.0988	227.28	0.0838	192.74	0.0752	173.2	0.0275	63.25
22	21:00	0.0988	227.28	0.0838	192.74	0.0752	173.2	0.0275	63.25
23	22:00	0.0988	227.28	0.0838	192.74	0.0752	173.2	0.0275	63.25
24	23:00	0.0988	227.28	0.0838	192.74	0.0752	173.2	0.0275	63.25

Lampiran B.4 Tabel 4 *Monitoring* Penggunaan Air Hari Ke-4

No.	Waktu	Node ID							
		Node 2		Node 3		Node 4		Node 5	
		Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)
1	00:00	0	0	0	0	0	0	0	0
2	01:00	0	0	0	0	0	0	0	0
3	02:00	0	0	0	0	0	0	0	0
4	03:00	0	0	0	0	0	0	0	0
5	04:00	0	0	0	0	0	0	0	0
6	05:00	0	0	0	0	0	0	0	0
7	06:00	0	0	0	0	0	0	0	0
8	07:00	0.0144	33.12	0	0	0.0164	37.72	0.009	20.7

9	08:00	0.03	69.09	0.0281	64.63	0.03	69.09	0.0145	33.35
10	09:00	0.0422	97.1	0.0281	64.63	0.03	69.09	0.0145	33.35
11	10:00	0.0544	127.46	0.0281	64.63	0.03	69.09	0.0145	33.35
12	11:00	0.0544	127.46	0.0281	64.63	0.03	69.09	0.0145	33.35
13	12:00	0.0544	127.46	0.0281	64.63	0.03	69.09	0.0145	33.35
14	13:00	0.0544	127.46	0.0281	64.63	0.03	69.09	0.0145	33.35
15	14:00	0.0544	127.46	0.0281	64.63	0.03	69.09	0.0145	33.35
16	15:00	0.0544	127.46	0.0281	64.63	0.03	69.09	0.0145	33.35
17	16:00	0.0544	127.46	0.0281	64.63	0.0711	163.53	0.0245	56.35
18	17:00	0.0817	187.91	0.0658	151.4	0.0711	163.53	0.0245	56.35
19	18:00	0.0925	212.81	0.0658	151.4	0.0711	163.53	0.0245	56.35
20	19:00	0.0925	212.81	0.0658	151.4	0.0711	163.53	0.0245	56.35
21	20:00	0.0925	212.81	0.0658	151.4	0.0711	163.53	0.0245	56.35
22	21:00	0.0925	212.81	0.0658	151.4	0.0711	163.53	0.0245	56.35
23	22:00	0.0925	212.81	0.0658	151.4	0.0711	163.53	0.0245	56.35
24	23:00	0.0925	212.81	0.0658	151.4	0.0711	163.53	0.0245	56.35

Lampiran B.5 Tabel 5 *Monitoring* Penggunaan Air Hari Ke-5

No.	Waktu	Node ID							
		Node 2		Node 3		Node 4		Node 5	
		Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)
1	00:00	0	0	0	0	0	0	0	0
2	01:00	0	0	0	0	0	0	0	0
3	02:00	0	0	0	0	0	0	0	0
4	03:00	0	0	0	0	0	0	0	0
5	04:00	0	0	0	0	0	0	0	0
6	05:00	0	0	0	0	0	0	0	0
7	06:00	0	0	0	0	0	0	0	0
8	07:00	0.0041	9.43	0.0205	47.15	0.0288	66.24	0.0559	128.66
9	08:00	0.0127	29.21	0.0322	74.06	0.0634	145.82	0.1664	382.72
10	09:00	0.0322	74.08	0.0322	74.06	0.0634	145.82	0.1664	382.72
11	10:00	0.051	117.3	0.0322	74.06	0.0634	145.82	0.1664	382.72
12	11:00	0.051	117.3	0.0322	74.06	0.0634	145.82	0.1664	382.72
13	12:00	0.051	117.3	0.0322	74.06	0.0634	145.82	0.1664	382.72
14	13:00	0.051	117.3	0.0322	74.06	0.0634	145.82	0.1664	382.72
15	14:00	0.051	117.3	0.0322	74.06	0.0634	145.82	0.1664	382.72
16	15:00	0.051	117.3	0.0322	74.06	0.0634	145.82	0.1664	382.72
17	16:00	0.0657	151.79	0.0322	74.06	0.0634	145.82	0.1732	398.36
18	17:00	0.0961	221.03	0.0322	74.06	0.0634	145.82	0.1732	398.36
19	18:00	0.1232	283.26	0.0744	171.17	0.0959	220.66	0.1732	398.36
20	19:00	0.1232	283.26	0.0744	171.17	0.0959	220.66	0.1732	398.36
21	20:00	0.1232	283.26	0.0744	171.17	0.0959	220.66	0.1732	398.36
22	21:00	0.1232	283.26	0.0744	171.17	0.0959	220.66	0.1732	398.36
23	22:00	0.1232	283.26	0.0744	171.17	0.0959	220.66	0.1732	398.36
24	23:00	0.1232	283.26	0.0744	171.17	0.0959	220.66	0.1732	398.36

Lampiran B.6 Tabel 6 *Monitoring* Penggunaan Air Hari Ke-6

No.	Waktu	Node ID							
		Node 2		Node 3		Node 4		Node 5	
		Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)
1	00:00	0	0	0	0	0	0	0	0

2	01:00	0	0	0	0	0	0	0	0
3	02:00	0	0	0	0	0	0	0	0
4	03:00	0	0	0	0	0	0	0	0
5	04:00	0	0	0	0	0	0	0	0
6	05:00	0	0	0	0	0	0	0	0
7	06:00	0	0	0	0	0.0209	48.07	0	0
8	07:00	0.0176	40.48	0.0223	51.29	0.0209	48.07	0.0122	28.15
9	08:00	0.0615	141.45	0.0365	83.95	0.0209	48.07	0.0122	28.15
10	09:00	0.0615	141.45	0.0447	102.94	0.0209	48.07	0.0122	28.15
11	10:00	0.0615	141.45	0.0447	102.94	0.0209	48.07	0.0122	28.15
12	11:00	0.0615	141.45	0.0447	102.94	0.0209	48.07	0.0122	28.15
13	12:00	0.0615	141.45	0.0447	102.94	0.0209	48.07	0.0122	28.15
14	13:00	0.0615	141.45	0.0447	102.94	0.0209	48.07	0.0122	28.15
15	14:00	0.0615	141.45	0.0447	102.94	0.0209	48.07	0.0122	28.15
16	15:00	0.0615	141.45	0.0447	102.94	0.0209	48.07	0.0122	28.15
17	16:00	0.0615	141.45	0.0447	102.94	0.0209	48.07	0.0240	55.24
18	17:00	0.1000	230.07	0.0703	161.69	0.0399	91.78	0.0240	55.24
19	18:00	0.1000	230.07	0.0800	184.07	0.0399	91.78	0.0240	55.24
20	19:00	0.1000	230.07	0.0800	184.07	0.0399	91.78	0.0240	55.24
21	20:00	0.1000	230.07	0.0800	184.07	0.0399	91.78	0.0240	55.24
22	21:00	0.1000	230.07	0.0800	184.07	0.0399	91.78	0.0240	55.24
23	22:00	0.1000	230.07	0.0800	184.07	0.0399	91.78	0.0240	55.24
24	23:00	0.1000	230.07	0.0800	184.07	0.0399	91.78	0.0240	55.24

Lampiran B.7 Tabel 7 *Monitoring* Penggunaan Air Hari Ke-7

No.	Waktu	Node ID							
		Node 2		Node 3		Node 4		Node 5	
		Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)
1	00:00	0	0	0	0	0	0	0	0
2	01:00	0	0	0	0	0	0	0	0
3	02:00	0	0	0	0	0	0	0	0
4	03:00	0	0	0	0	0	0	0	0
5	04:00	0	0	0	0	0	0	0	0
6	05:00	0	0	0	0	0	0	0	0
7	06:00	0	0	0	0	0	0	0	0
8	07:00	0.0154	35.42	0.0208	47.84	0	0	0.0166	38.18
9	08:00	0.0444	102.19	0.0299	68.77	0.0189	43.47	0.0166	38.18
10	09:00	0.0628	144.48	0.0299	68.77	0.0189	43.47	0.0166	38.18
11	10:00	0.0628	144.48	0.0299	68.77	0.0189	43.47	0.0166	38.18
12	11:00	0.0628	144.48	0.0299	68.77	0.0189	43.47	0.0166	38.18
13	12:00	0.0628	144.48	0.0299	68.77	0.0189	43.47	0.0166	38.18
14	13:00	0.0628	144.48	0.0299	68.77	0.0189	43.47	0.0166	38.18
15	14:00	0.0628	144.48	0.0299	68.77	0.0189	43.47	0.0166	38.18
16	15:00	0.0628	144.48	0.0299	68.77	0.0189	43.47	0.0166	38.18
17	16:00	0.0628	144.48	0.0299	68.77	0.0473	108.84	0.0273	62.79
18	17:00	0.1112	255.76	0.0618	142.14	0.0778	179.01	0.0273	62.79
19	18:00	0.1112	255.76	0.0618	142.14	0.0778	179.01	0.0273	62.79
20	19:00	0.1112	255.76	0.0618	142.14	0.0778	179.01	0.0273	62.79
21	20:00	0.1112	255.76	0.0618	142.14	0.0778	179.01	0.0273	62.79
22	21:00	0.1112	255.76	0.0618	142.14	0.0778	179.01	0.0273	62.79
23	22:00	0.1112	255.76	0.0618	142.14	0.0778	179.01	0.0273	62.79
24	23:00	0.1112	255.76	0.0618	142.14	0.0778	179.01	0.0273	62.79

LAMPIRAN C LISTING PROGRAM

1. *Listing program Water Meter*
 - a. *Listing Program Menghubungkan Node Dengan Node Lainnya*

```
#include <painlessMesh.h>
#include <ESPAsyncTCP.h>
#include <Arduino_JSON.h>
#include <ESP8266HTTPClient.h>

// Define untuk koneksi dan port jaringan -----
-----
#define MESH_PREFIX "smartwatermeter"
#define MESH_PASSWORD "123456789"
#define MESH_PORT 80

//String to send to other nodes with sensor readings
String readings;

Scheduler userScheduler; // to control your personal task
painlessMesh mesh;

// User stub
void sendMessage(); // Prototype so PlatformIO doesn't
complain
String getReadings(); // Prototype for sending sensor
readings

//Create tasks: to send messages and get readings;
Task taskSendMessage(TASK_SECOND * 60 , TASK_FOREVER,
&sendMessage);

String getReadings () {
    JSONVar jsonReadings;
    jsonReadings["node_id"] = nodeNumber;
    jsonReadings["volume"]= TOTAL;
    jsonReadings["cost"]= cost;
    readings = JSON.stringify(jsonReadings);
    return readings;
}
```

```

void sendMessage () {
    String msg = getReadings();
    mesh.sendBroadcast(msg);
}

//Needed for painless library
void receivedCallback( uint32_t from, String &msg ) {
    Serial.printf("Message received from %u\n", from,
msg.c_str());
    String nodemcuData1 = msg;
    JSONVar myObject = JSON.parse(msg.c_str());
    int node_id = myObject["node_id"];
    double volume = myObject["volume"];
    double cost = myObject["cost"];
    Serial.print("Node: ");
    Serial.print(node_id);
    Serial.print(" Volume: ");
    Serial.println(volume,5);
    Serial.print("Biaya: ");
    Serial.println(cost);
void newConnectionCallback(uint32_t nodeId) {
    Serial.printf("New Connection, NodeId = %u\n", nodeId);
}

void changedConnectionCallback() {
    Serial.printf("Changed connections \n");
}

void nodeTimeAdjustedCallback(int32_t offset) {
    Serial.printf("Adjusted time %u. Offset = %d\n",
mesh.getNodeTime(),offset);
}

void setup(){
    Serial.begin(115200);
    Wire.begin(D1,D2);
    //mesh.setDebugMsgTypes( ERROR | MESH_STATUS | CONNECTION
| SYNC | COMMUNICATION | GENERAL | MSG_TYPES | REMOTE ); //
all types on

```

```

    mesh.setDebugMsgTypes( ERROR | STARTUP | CONNECTION ); //
    set before init() so that you can see startup messages

    // Channel set to 6. Make sure to use the same channel for
    your mesh and for you other
    // network (STATION_SSID)
    mesh.init( MESH_PREFIX, MESH_PASSWORD, &userScheduler,
MESH_PORT);
    mesh.onReceive(&receivedCallback);
    mesh.onNewConnection(&newConnectionCallback);
    mesh.onChangedConnections(&changedConnectionCallback);
    mesh.onNodeTimeAdjusted(&nodeTimeAdjustedCallback);

    userScheduler.addTask(taskSendMessage);
    taskSendMessage.enable();

    pinMode(input, INPUT);
    lcd.begin();
    lcd.backlight();
}

void loop(){
    mesh.update();
    delay(1000)
}

```

b. Listing Program Pembacaan Debit Air Dan Menampilkan Data Pada LCD I2C 16x2

```

#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include <EEPROM.h>
int nodeNumber = 1;

LiquidCrystal_I2C lcd(0x27, 16, 2); //biru pakai 27 hijau 3f

const int input = D6;
int addr = 0;
byte sensorInterrupt = 0;

```

```

float calibrationFactor = 5.5;
volatile byte pulseCount;

float flowRate,liter,TOTAL,cost;

unsigned long oldTime;
unsigned long prevTime= 0;
unsigned long seconds=0;

//biaya tergantung kelompok rumah tangga (berdasarkan daerah
tangerang)
//R1 = 1875; R2 = 2300; R3 = 2800; R4 = 3350;
int R2 = 2300;

void setup(){
  Serial.begin(115200);
  Wire.begin(D1,D2);

  pinMode(input,INPUT);
  pulseCount = 0;
  flowRate = 0;
  liter = 0;
  TOTAL = 0;
  oldTime = 0;
  cost = 0;

  lcd.begin();
  lcd.backlight();

  digitalWrite(input, HIGH);

  attachInterrupt (digitalPinToInterrupt (input),pulseCounter,RI
SING);
}

void loop()
{
  //Reset Data Tiap Hari (24jam)
  unsigned long currentTime = millis();

```

```

if ((currentTime-prevTime) >= 1000){
    seconds++;
    Serial.print("Waktu: ");
    Serial.println(seconds);
    prevTime = currentTime;
}
if (millis()-oldTime>1000){
    detachInterrupt(sensorInterrupt);

    flowRate = ((1000.0/(millis()-
oldTime))*pulseCount)/calibrationFactor;

    oldTime = millis();

    liter += flowRate/60;
    TOTAL = liter/1000;
    cost = R2*TOTAL;

    unsigned int frac;

    Serial.print("Node Id: ");
    Serial.println(nodeNumber);

    Serial.print("Flow Rate: ");
    Serial.print(int(flowRate));
    Serial.print(".");

    frac = (flowRate - int(flowRate))*100;
    Serial.print(frac,DEC);
    Serial.println(" L/min");

    Serial.print("Output Total Quantity: ");
    Serial.print(TOTAL,4);
    Serial.println(" m^3 ");

    Serial.print("Biaya: Rp.");
    Serial.println(cost);
    Serial.print("\n");

```

```

    lcd.setCursor(0,0);
    lcd.print("Speed: ");
    lcd.print(flowRate,2);
    lcd.setCursor(13,0);
    lcd.print("L/M");
    lcd.setCursor(0,1);
    lcd.print("Total: ");
    lcd.setCursor(6,1);
    lcd.print(TOTAL,4);
    lcd.setCursor(13,1);
    lcd.print("m^3");

    pulseCount = 0;

    attachInterrupt(sensorInterrupt, pulseCounter,
FALLING);
}
if (seconds >= 86400){
    seconds = 0;
    TOTAL = 0;
    cost = 0;
    lcd.clear();
}
}

void pulseCounter(){
    pulseCount++;
}

```

c. Listing Program Pengiriman Data Pembacaan Ke Server

```

#include <ESP8266HTTPClient.h>
String url =
"http://10.249.193.100/monitoring/nodemcu_log/webapi/api/cre
ate.php";

StaticJsonDocument<1024> doc;

```

```

String nodemcuData;
int node_id = nodeNumber;
double volume = TOTAL;
double costs = cost;

doc["node_id"] = String(node_id);
doc["volume"] = String(volume,4);
doc["cost"] = String(costs);

HTTPClient http;
http.begin(url);
http.addHeader("Content-Type", "application/json");

serializeJson(doc, nodemcuData);
Serial.print("POST data >> ");
Serial.println(nodemcuData);

int httpCode = http.POST(nodemcuData); //Send the
request
String payload;
if (httpCode > 0) { //Check the returning code
    payload = http.getString(); //Get the request
response payload
    payload.trim();
    if( payload.length() > 0 ){
        Serial.println(payload + "\n");
    }
}
http.end();
}

```

d. Listing Program Pengujian Keseluruhan

```

#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include <EEPROM.h>
#include <painlessMesh.h>
#include <ESPAsyncTCP.h>
#include <Arduino_JSON.h>
#include <ESP8266HTTPClient.h>

```



```

// Define untuk koneksi dan port jaringan -----
-----
#define MESH_PREFIX "smartwatermeter"
#define MESH_PASSWORD "123456789"
#define MESH_PORT 80

//#define STATION_SSID "RedmiN7"
//#define STATION_PASSWORD "12345567"
//#define STATION_PORT 80
//uint8_t station_ip[4] = {192,168,43,1}; //ip server

//node 1=10.249.193.100
//node 2=10.124.71.100
//node 3=10.126.155.100
//node 4=10.22.63.100
//node 5=10.104.253.100
String url =
"http://10.249.193.100/monitoring/nodemcu_log/webapi/api/create.php";
//Number for this node
int nodeNumber = 1;

LiquidCrystal_I2C lcd(0x27, 16, 2); //biru pakai 27 hijau 3f

const int input = D6;
int addr = 0;
byte sensorInterrupt = 0;

float calibrationFactor = 5.5;
volatile byte pulseCount;

float flowRate, liter, TOTAL, cost;
unsigned int flowMilliLitres;
unsigned long totalMilliLitres;

unsigned long oldTime;
unsigned long prevTime= 0;
unsigned long seconds=0;

```

```

//biaya tergantung kelompok rumah tangga (berdasarkan daerah
tangerang)
//R1 = 1875; R2 = 2300; R3 = 2800; R4 = 3350;
int R2 = 2300;

//String to send to other nodes with sensor readings
String readings;

Scheduler userScheduler; // to control your personal task
painlessMesh mesh;

// User stub
void sendMessage(); // Prototype so PlatformIO doesn't
complain
String getReadings(); // Prototype for sending sensor
readings

//Create tasks: to send messages and get readings;
Task taskSendMessage(TASK_SECOND * 5 , TASK_FOREVER,
&sendMessage);

String getReadings () {
    JSONVar jsonReadings;
    jsonReadings["node_id"] = nodeNumber;
    jsonReadings["volume"]= TOTAL;
    jsonReadings["cost"]= cost;
    readings = JSON.stringify(jsonReadings);
    return readings;
}

void sendMessage () {
    String msg = getReadings();
    mesh.sendBroadcast(msg);
}

```

```

//Needed for painless library
void receivedCallback( uint32_t from, String &msg ) {
    Serial.printf("Message received from %u\n", from,
msg.c_str());
    String nodemcuData1 = msg;
    JSONVar myObject = JSON.parse(msg.c_str());
    int node_id = myObject["node_id"];
    double volume = myObject["volume"];
    double cost = myObject["cost"];
    Serial.print("Node: ");
    Serial.print(node_id);
    Serial.print(" Volume: ");
    Serial.println(volume,5);
    Serial.print("Biaya: ");
    Serial.println(cost);

    HTTPClient http;
    http.begin(url);
    http.addHeader("Content-Type", "application/json");

    Serial.print("POST data >> ");
    Serial.println(nodemcuData1);

    int httpCode1 = http.POST(nodemcuData1);//Send the request
    String payload1;
    if (httpCode1 > 0) { //Check the returning code
        payload1 = http.getString(); //Get the request
response payload
        payload1.trim();
        if( payload1.length() > 0 ){
            Serial.println(payload1 + "\n");
        }
    }
}

void newConnectionCallback(uint32_t nodeId) {
    Serial.printf("New Connection, NodeId = %u\n", nodeId);
}

```

```

void changedConnectionCallback() {
    Serial.printf("Changed connections \n");
}

void nodeTimeAdjustedCallback(int32_t offset) {
    Serial.printf("Adjusted time %u. Offset = %d\n",
mesh.getNodeTime(), offset);
}

void setup(){
    Serial.begin(115200);
    Wire.begin(D1,D2);
    //mesh.setDebugMsgTypes( ERROR | MESH_STATUS | CONNECTION
| SYNC | COMMUNICATION | GENERAL | MSG_TYPES | REMOTE ); //
all types on
    mesh.setDebugMsgTypes( ERROR | STARTUP | CONNECTION ); //
set before init() so that you can see startup messages

    // Channel set to 6. Make sure to use the same channel for
your mesh and for you other
    // network (STATION_SSID)
    mesh.init( MESH_PREFIX, MESH_PASSWORD, &userScheduler,
MESH_PORT);
    mesh.onReceive(&receivedCallback);
    mesh.onNewConnection(&newConnectionCallback);
    mesh.onChangedConnections(&changedConnectionCallback);
    mesh.onNodeTimeAdjusted(&nodeTimeAdjustedCallback);

    //khusus untuk node bridge
    //mesh.initOTAResive("bridge");
    //mesh.stationManual(STATION_SSID, STATION_PASSWORD,
STATION_PORT, station_ip);
    //mesh.setRoot(true);
    //mesh.setContainsRoot(true);

    userScheduler.addTask(taskSendMessage);
    taskSendMessage.enable();
    delay(10);
}

```

```

pinMode(input, INPUT);
pulseCount = 0;
flowRate = 0;
flowMilliLitres = 0;
totalMilliLitres = 0;
liter = 0;
TOTAL = 0;
oldTime = 0;
cost = 0;

lcd.begin();
lcd.backlight();

digitalWrite(input, HIGH);

attachInterrupt(digitalPinToInterrupt(input), pulseCounter, RISING);
}

void loop()
{
  unsigned long currentTime = millis();
  if ((currentTime-prevTime) >= 1000){
    seconds++;
    Serial.print("Waktu: ");
    Serial.println(seconds);
    prevTime = currentTime;
  }

  if (millis()-oldTime>1000){
    detachInterrupt(sensorInterrupt);

    flowRate = ((1000.0/(millis()-oldTime))*pulseCount)/calibrationFactor;

    oldTime = millis();

    liter += flowRate/60;
    TOTAL = liter/1000;
  }
}

```

```

    cost = R2*TOTAL;

    unsigned int frac;

    Serial.print("Node Id: ");
    Serial.println(nodeNumber);

    Serial.print("Flow Rate: ");
    Serial.print(int(flowRate));
    Serial.print(".");

    frac = (flowRate - int(flowRate))*100;
    Serial.print(frac,DEC);
    Serial.println(" L/min");

    Serial.print("Output Total Quantity: ");
    Serial.print(TOTAL,4);
    Serial.println(" m^3 ");

    Serial.print("Biaya: Rp.");
    Serial.println(cost);
    Serial.print("\n");

    lcd.setCursor(0,0);
    lcd.print("Speed: ");
    lcd.print(flowRate,2);
    lcd.setCursor(13,0);
    lcd.print("L/M");
    lcd.setCursor(0,1);
    lcd.print("Total: ");
    lcd.setCursor(6,1);
    lcd.print(TOTAL,4);
    lcd.setCursor(13,1);
    lcd.print("m^3");

    pulseCount = 0;

    attachInterrupt(sensorInterrupt, pulseCounter, FALLING);
}

```

```

//Gunakan listing program dibawah jika node AP juga akan
digunakan untk pembacaan air (node ap akan mengirimkan data
pembacaan air miliknya sendiri ke server)//
//StaticJsonDocument<1024> doc;
//String nodemcuData;
//int node_id = nodeNumber;
//double volume = TOTAL;
//double costs = cost;

//doc["node_id"] = String(node_id);
//doc["volume"] = String(volume,4);
//doc["cost"] = String(costs);

//HTTPClient http;
//http.begin(url);
//http.addHeader("Content-Type", "application/json");

//serializeJson(doc, nodemcuData);
//Serial.print("POST data >> ");
//Serial.println(nodemcuData);

//int httpCode = http.POST(nodemcuData);//Send the request
//String payload;
//if (httpCode > 0) { //Check the returning code
//  payload = http.getString(); //Get the request response
payload
//  payload.trim();
//  if( payload.length() > 0 ){
//    Serial.println(payload + "\n");
//  }
// }
//http.end();

mesh.update();
delay(1000);
if (seconds >= 86400){
  seconds = 0;
  pulseCount = 0;
}

```

```

        flowRate = 0;
        liter = 0;
        TOTAL = 0;
        cost = 0;
        lcd.clear();
    }
}

void pulseCounter(){
    pulseCount++;
}

```

2. *Listing program website*

a. *Halaman Dashboard*

```

<?php
    include "config.php";
    $label=
["Januari", "Februari", "Maret", "April", "Mei", "Juni", "Juli", "Agustus",
", "September", "Oktober", "November", "Desember"];
?>

<html>
<head>
    <script type="text/javascript" src="chartjs/Chart.js"></script>
</head>
<hr>
<b>Data Monitoring Air</b>
<hr>

<body>
    <table width= 100% border="1" cellpadding="4">
        <tr bgcolor="00ffffff">
            <th>No</th>
            <th>Node_ID</th>
            <th>Volume</th>
            <th>Biaya</td>
            <th>Created at</th>
        </tr>

```



```

<!-- Input data ke tabel dari Database-->
<?php
    $no=0;
    $query= mysqli_query($connection, "SELECT m1.* FROM
dashboard m1 LEFT JOIN dashboard m2 ON (m1.node_id = m2.node_id AND
m1.id < m2.id) WHERE m2.id IS NULL ORDER BY node_id;");
    while($result= mysqli_fetch_array($query)){
        $no++
        ?>

        <tr>
            <td><?php echo $no?></td>
                <td><b><a    style="text-decoration:none"
href="page/detail-data1.php?node_id=<?=$result['node_id']?>"><?php
echo $result['node_id']?></a></b></td>
            <td><?php echo $result['volume']?></td>
            <td><?php echo $result['cost']?></td>
            <td><?php echo $result['created_at']?></td>
        </tr>
        <?php
        }
        ?>
    </table>

    <div style="width:600px; margin: 10px auto">
        <canvas id="myChart"></canvas>
    </div>

    <script>
        var ctx=
document.getElementById('myChart').getContext('2d');
        var myChart = new Chart(ctx,{
            type:'line',
            data:{
                labels: <?php echo json_encode($label); ?>,
                datasets:[
                    {

```

```

        label: 'Node 1',
        data: [
            <?php
                $year=date('Y');
                for ($i=1; $i<=12 ; $i++) {
                    $data_node_1=mysqli_query($connection,"SELECT
sum(volume) as volume FROM dashboard t1 JOIN (SELECT
DATE(created_at) date_date, node_id, MAX(created_at) max_date FROM
dashboard GROUP BY date_date, node_id ) t2 ON t1.created_at =
t2.max_date WHERE t2.node_id=1 AND YEAR(t1.created_at)=$year AND
MONTH(t1.created_at)=$i");
                    $data1="";
                    while($data_n1=mysqli_fetch_assoc($data_no
de_1)){
                        $dn1=$data_n1['volume'];
                        $data1 .= "'$dn1'","";
                        echo $data1;
                    }
                }
            ?>
        ],
        backgroundColor:[ 'rgba(255, 99, 132, 0.4)'],
        borderColor:['rgba(255, 99, 132, 0.4)'],
        borderWidth: 1
    },
    {
        label: 'Node 2',
        data: [
            <?php
                for ($i=1; $i<=12 ; $i++) {
                    $data_node_2=mysqli_query($connection,"SELECT
sum(volume) as volume FROM dashboard t1 JOIN (SELECT
DATE(created_at) date_date, node_id, MAX(created_at) max_date FROM
dashboard GROUP BY date_date, node_id ) t2 ON t1.created_at =
t2.max_date WHERE t2.node_id=2 AND YEAR(t1.created_at)=$year AND
MONTH(t1.created_at)=$i");
                    $data2="";
                    while($data_n2=mysqli_fetch_assoc($data_no
de_2)){

```

```

                $dn2=$data_n2['volume'];
                $data2 .= "'$dn2'".",";
                echo $data2;
            }
        }
        ?>
    ],
    backgroundColor:[ 'rgba(54, 162, 235, 0.4)',
    borderColor:['rgba(54, 162, 235, 0.4)',
    borderWidth: 1
    },
    {
    label: 'Node 3',
    data: [
        <?php
            for ($i=1; $i<=12 ; $i++) {
                $data_node_3=mysqli_query($connection,"SELECT
sum(volume) as volume FROM dashboard t1 JOIN (SELECT
DATE(created_at) date_date, node_id, MAX(created_at) max_date FROM
dashboard GROUP BY date_date, node_id ) t2 ON t1.created_at =
t2.max_date WHERE t2.node_id=3 AND YEAR(t1.created_at)=$year AND
MONTH(t1.created_at)=$i");
                $data3="";
                while($data_n3=mysqli_fetch_assoc($data_no
de_3)){
                    $dn3=$data_n3['volume'];
                    $data3 .= "'$dn3'".",";
                    echo $data3;
                }
            }
        ?>
    ],
    backgroundColor:[ 'rgba(255, 0, 0, 0.4)',
    borderColor:['rgba(255, 0, 0, 0.4)',
    borderWidth: 1
    },
    {
    label: 'Node 4',
    data: [

```

```

        <?php
            for ($i=1; $i<=12 ; $i++) {
                $data_node_4=mysqli_query($connection,"SELECT
sum(volume) as volume FROM dashboard t1 JOIN (SELECT
DATE(created_at) date_date, node_id, MAX(created_at) max_date FROM
dashboard GROUP BY date_date, node_id ) t2 ON t1.created_at =
t2.max_date WHERE t2.node_id=4 AND YEAR(t1.created_at)=$year AND
MONTH(t1.created_at)=$i");
                $data4="";
                while($data_n4=mysqli_fetch_assoc($data_no
de_4)){
                    $dn4=$data_n4['volume'];
                    $data4 .= "'$dn4'","";
                    echo $data4;
                }
            }
        ?>
    ],
    backgroundColor:[ 'rgba(0, 255, 0, 0.4)',
borderColor:['rgba(0, 255, 0, 0.4)'],
borderWidth: 1
},
{
label: 'Node 5',
data: [
        <?php
            for ($i=1; $i<=12 ; $i++) {
                $data_node_5=mysqli_query($connection,"SELECT
sum(volume) as volume FROM dashboard t1 JOIN (SELECT
DATE(created_at) date_date, node_id, MAX(created_at) max_date FROM
dashboard GROUP BY date_date, node_id ) t2 ON t1.created_at =
t2.max_date WHERE t2.node_id=5 AND YEAR(t1.created_at)=$year AND
MONTH(t1.created_at)=$i");
                $data5="";
                while($data_n5=mysqli_fetch_assoc($data_no
de_5)){
                    $dn5=$data_n5['volume'];
                    $data5 .= "'$dn5'","";
                    echo $data5;

```

```

        }
    }
    ?>
],
backgroundColor:[ 'rgba(255, 0, 255, 0.4)',
borderColor:['rgba(255, 0, 255, 0.4)',
borderWidth: 1
}
]
},
options:{
    scales:{
        yAxes:[{
            ticks:{
                beginAtZero: true
            }
        }]
    }
}
});
</script>
</body>
</html>

```

b. Listing program halaman detail data

```

<html>
<body>
<hr>
<h2><b>Detail Data User Node ID <?php echo
$_GET['node_id'];?></b></h2>
<style>
    @media print{
        #art-sheet #sidebar{
            width:0;
        }
        #art-sheet #article{
            width: 100%;
        }
        .back-page, .print-page, .search-date, .label-pencarian,
        .tahun-pencarian, .submit-pencarian, .bulan-pencarian{
            display: none;
        }
    }

```

```

    }

}

#art-sheet{
border-top: 1px solid #FEFEFE;
clear: both;
width: 100%;
margin: 1px auto 0;
background: white;
color: #000000;
text-align:left;
}

#sidebar{
    float: left;
    width: 15%;
    margin: 0 5px auto;
    padding: 5px;
    position:fixed;
}

#article{
    padding-left:40px;
    float: right;
    width: 80%;
}

.article-header{
float: left;
width: 100%;
height: auto;
line-height:40px;
margin: 10px 0 2px 0;
padding: 0;
}

.article-content{
padding: 10px;
float: left;
width: 100%;
overflow:scroll;
}

</style>
<button onclick="location.href='../';" class="back-
page">Kembali</button>
<hr>

```

```

<div id=art-sheet>
  <div id=sidebar>
    <button onclick="window.print()" class="print-page">Print
or Save to PDF</button>
    <br>
    <br>
    <b class="search-date">Search</b>
    <form action="" method="POST">
    <label class="label-pencarian" for="pencarian">Pilih Bulan
dan Tahun</label>
    <select class="bulan-pencarian" name="bulan" id="cari">
    <option value="01">Januari</option>
    <option value="02">Februari</option>
    <option value="03">Maret</option>
    <option value="04">April</option>
    <option value="05">Mei</option>
    <option value="06">Juni</option>
    <option value="07">Juli</option>
    <option value="08">Agustus</option>
    <option value="09">September</option>
    <option value="10">Oktober</option>
    <option value="11">November</option>
    <option value="12">Desember</option>
    <input class="tahun-pencarian" type="text" name="tahun"
placeholder="Tahun">
    </select>
    <button class="submit-pencarian" type="submit"
name="submit" >Submit</button>
  </form>
</div>
<div id=article>
  <div class=aricle-content>
    DATA BULAN <?php
      if(!isset($_POST['bulan'])) {
        $_POST['bulan']=date('m');
      }
      if(!isset($_POST['tahun'])) {
        $_POST['tahun']=date('Y');
      }
      echo $_POST['bulan'],"- ".$_POST['tahun'];?>
    <table width= 100% border="1" cellpadding="4">
      <tr bgcolor="00ffffff">
        <th>No</th>
        <th>Node_ID</th>
        <th>Volume (m3)</th>
        <th>Biaya (Rp.)</th>
        <th>Created at</th>

```

```

        </tr>
    <?php
        if(!isset($_POST['bulan'])) {
            $bulan = date('m');
        }
        else {
            $bulan = $_POST['bulan'];
        }
        if(!isset($_POST['tahun'])) {
            $tahun = date('Y');
        }
        else {
            $tahun = $_POST['tahun'];
        }
    ?>

    <?php
        if(isset($_GET['node_id'])) {
            $node_id = $_GET['node_id'];
        }
        else {
            die ("Error, No ID Selected!");
        }
    ?>
    <!--Tambah data-->
    <?php
        include "config.php";
        $node_id = $_GET['node_id'];
        $no=0;
        $total=0;
        $tot_bayar=0;
        $tot_volume=0;
        $tot_vol=0 ;
        $query= mysqli_query($connection, "SELECT t1.* FROM
dashboard t1 JOIN (SELECT DATE(created_at) date_date, node_id,
MAX(created_at) max_date FROM dashboard GROUP BY date_date,
node_id ) t2 ON t1.created_at = t2.max_date WHERE
t2.node_id=$node_id AND YEAR(t1.created_at)='$tahun' AND
MONTH(t1.created_at)='$bulan'");
        while($result= mysqli_fetch_array($query)) {
            $total= $result['cost'];
            $tot_vol= $result['volume'];
            $tot_volume += $tot_vol;
            $tot_bayar += $total;
            $no++
        }
    ?>
    <tr>

```



```

        <td><?php echo $no?></td>
        <td><?php echo $result['node id']?></td>
        <td><?php echo $result['volume']?></td>
        <td><?php echo $result['cost']?></td>
        <td><?php echo $result['created_at']?></td>
    </tr>

    <?php
    }
    ?>

    <tr>
        <th colspan="2">Total Bayar</th>
        <th><?php echo $tot_volume ?></th>
        <th><?php echo $tot_bayar ?></th>
    </tr>
</div>
</div>
</table>

</body>
</html>

```

c. Listing program halaman *about us*

```

<hr>
<b>ABOUT US</b>
<hr>

<div class="news">
    <div class="judul">
        Profile
    </div>
    <div class="sinopsis">
        Web yang bertujuan untuk memudahkan monitoring penggunaan air
        di setiap rumah
    </div>
</div>

<div class="news">
    <div class="judul">
        Information
    </div>
    <div class="sinopsis">
        Air membawa nutrisi yang dibutuhkan oleh tubuh. Air sebagai
        sumber pemenuhan kebutuhan hidup, seperti irigasi persawahan,

```

kehidupan binatang, nutrisi tumbuhan, dan lainnya. Penggunaan air yang bersih bisa mencegah berbagai macam penyakit, seperti diare, kolera, dan sebagainya.

Hari Air Sedunia dirayakan setiap tanggal 22 Maret.

Penghematan air dapat dilakukan dengan cara berikut.
1. Matikan keran air jika tidak digunakan.
2. Menggunakan air seperlunya.
3. Merawat peralatan pipa air, keran, dan penampungan air.
4. Pantau tagihan air secara berkala.

</div>

</div>

d. *Listing program halaman contact*

```
<hr>
<b>Admin Profil</b>
<hr>

<div style="width: 200px; float: left;">

</div>

<div style="float: left;">
<table width="100%" cellpadding="5">
  <tr>
    <th width="120" align="left">Nama</th>
    <td>: Dimico</td>
  </tr>
  <tr>
    <th width="120" align="left">Bagian</th>
    <td>: Admin</td>
  </tr>
  <tr>
    <th width="120" align="left">Email</th>
    <td>: 3332170047@untirta.ac.id</td>
  </tr>
  <tr>
    <th width="120" align="left">Phone</th>
    <td>: 0123456789</td>
  </tr>
  <tr>
    <th width="120" align="left">Alamat</th>
    <td>: Tangerang, Banten</td>
  </tr>
</table>
</div>

<hr style="clear: both">
```

e. Listing program config.php

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "monitoring";

$conn = mysqli_connect($servername, $username, $password,
$dbname);
// if (!$conn){
//     die("Connection Failed:".mysqli_connect_error());
// }

function query($query){
    global $conn;
    $result = mysqli_query($conn,$query);
    $row = [];
    while ($row = mysqli_fetch_assoc($result)){
        $row[] = $row;
    }
    return $row;
}
?>
```

f. Listing program database.php

```
<?php
class Database {
    private $host = "localhost";
    private $database_name = "monitoring";
    private $username = "root";
    private $password = "";

    public $conn;

    public function getConnection(){
        $this->conn = null;
        try{
            $this->conn = new PDO("mysql:host=" . $this->host
            . ";dbname=" . $this->database_name, $this->username, $this-
            >password);
            $this->conn->exec("set names utf8");
        }catch(PDOException $exception){
            echo "Database could not be connected: " .
            $exception->getMessage();
        }
    }
}
```

```
        return $this->conn;
    }
}
?>
```

g. Listing program create.php

```
<?php
header("Access-Control-Allow-Origin: *");
header("Content-Type: application/json; charset=UTF-8");

include_once '../config/database.php';
include_once '../class/nodemcu_log.php';

$databse = new Database();
$db = $databse->getConnection();

$item = new monitoring($db);

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    // The request is using the POST method
    $data = json_decode(file_get_contents("php://input"));
    $item->node_id = $data->node_id;
    $item->volume = $data->volume;
    $item->cost = $data->cost;
}

if($item->createLogData()){
    echo 'Data created successfully.';
} else{
    echo 'Data could not be created.';
}

?>
```

h. Listing program nodemcu_log.php

```
<?php
class monitoring{

    // Connection
    private $conn;

    // Table
    private $db_table = "dashboard";

    // Columns
    public $id;
```

```

public $node_id;
public $volume;
public $cost;
public $created_at;

// Db connection
public function __construct($db){
    $this->conn = $db;
}

// CREATE
public function createLogData(){
    $sqlQuery = "INSERT INTO
                ". $this->db_table ."
                SET
                node_id = :node_id,
                volume = :volume,
                cost = :cost";
    $stmt = $this->conn->prepare($sqlQuery);

    // sanitize
    $this->node_id=htmlspecialchars(strip_tags($this-
>node_id));
    $this->volume=htmlspecialchars(strip_tags($this-
>volume));
    $this->cost=htmlspecialchars(strip_tags($this->cost));

    // bind data
    $stmt->bindParam(":node_id", $this->node_id);
    $stmt->bindParam(":volume", $this->volume);
    $stmt->bindParam(":cost", $this->cost);
    if($stmt->execute()){
        return true;
    }
    return false;
}
}
?>

```

i. *Listing program style css*

```

body {
    font-family: verdana;
    font-size: 0.85em;
}

.menu {

```

```

display: block;
background-color:#00A6BB;
height:46px;
}

.content {
clear: both;
padding:5px 0;
min-height:200px;
}

.footer {
clear: both;
margin-top:10px;
background-color: #eee;
padding:10px 8px;
}

#navigasi {
position:relative;
line-height:30px;
margin:0;
padding:0;
list-style-type:none;
list-style-position:outside;
}

#navigasi a {
display:block;
padding:8px 16px;
background-color:#00A6BB;
color:#fff;
text-decoration:none;
}

#navigasi a:hover {
background-color:#00BCD4;
color:#fff;
}

#navigasi li {
position:relative;
float:left;
}

#navigasi ul {

```

```

    position:absolute;
    display:none;
    margin:0;
    padding:0;
    list-style-type:none;
    list-style-position:outside;
}

#navigasi li ul a{
    width:12em;
    height:auto;
    float:left;
}

#navigasi li:hover ul{
    display:block;
}

#navigasi li:hover ul ul{
    display:none;
}

.news {
    background: #ddd;
}

.judul {
    padding: 4px;
    border-top: 1px solid #ccc;
    border-bottom: 1px solid #ccc;
    font-weight: bold;
}

.sinopsis {
    padding: 4px;
    margin-bottom: 5px;
}

table {
    table-layout:fixed;
}

td {
    overflow: hidden;
    text-overflow: ellipsis;
    word-wrap: break-word;
}

```