

TUGAS AKHIR
RANCANG BANGUN PURWARUPA DATABASE CURAH
HUJAN DAN SUHU MENGGUNAKAN ESP32 BERBASIS
INTERNET OF THINGS (IoT)

Diajukan Sebagai Persyaratan untuk Menyelesaikan Program Pendidikan
Diploma III Program Studi Teknik Informatika Jurusan Teknik Elektro
Politeknik Negeri Pontianak



Disusun Oleh:
OSCAR RIZALDI
NIM: 3201916118

PROGRAM STUDI D3 TEKNIK INFORMATIKA
JURUSAN TEKNIK ELEKTRO
POLITEKNIK NEGERI PONTIANAK
2022

HALAMAN PENGESAHAN
RANCANG BANGUN PURWARUPA DATABASE CURAH HUJAN DAN
SUHU MENGGUNAKAN ESP32 BERBASIS *INTERNET OF THINGS* (IoT)

Oleh:

OSCAR RIZALDI

320916118

**Tugas Akhir ini telah diterima dan disahkan sebagai salah satu syarat untuk
menyelesaikan Program Pendidikan Diploma III pada Program Studi
Teknik Informatika Jurusan Teknik Elektro Politeknik Negeri Pontianak**

Disahkan Oleh:

**Ketua Jurusan
Teknik Elektro**

**Ketua Program Studi
D-III Teknik Informatika**

**Hasan, ST,.MT.
NIP.19710821199903003**

**Suheri, ST.M.Cs.
NIP.198307172008121005**

Mengetahui,

Direktur Politeknik Negeri Pontianak

**Dr. Ir. H. Muhammad Toasin Asha, M.Si.
NIP. 196112251990111001**

HALAMAN PERNYATAAN
RANCANG BANGUN PURWARUPA DATABASE CURAH HUJAN DAN
SUHU MENGGUNAKAN ESP32 BERBASIS *INTERNET OF THINGS* (IoT)

Oleh:

OSCAR RIZALDI
3201916118

Dosen Pembimbing

Wendhi Yuniarto, ST.,MT.
NIP. 197406231999031001

Telah dipertahankan di depan penguji pada tanggal 25 Agustus 2022 dan
dinyatakan memenuhi syarat sebagai Laporan Tugas Akhir

Penguji 1,

Penguji 2,

Satriyo, S.T.,M.Kom
NIP. 197609232006041001

Fitri Wibowo, S.S.T.,M.T.
NIP. 198512282015041002

HALAMAN PERNYATAAN
RANCANG BANGUN PURWARUPA DATABASE CURAH HUJAN DAN
SUHU MENGGUNAKAN ESP32 BERBASIS *INTERNET OF THINGS* (IoT)

Oleh:

OSCAR RIZALDI

3201916118

Dosen Pembimbing

Wendhi Yunarto, ST.,MT.

NIP. 197406231999031001

**Telah dipertahankan di depan penguji pada tanggal 25 Agustus 2022 dan
dinyatakan memenuhi syarat sebagai Laporan Tugas Akhir**

Penguji 1,



Satriyo, S.T.,M.Kom

NIP. 197609232006041001

Penguji 2,



Fitri Wibowo, S.S.T.,M.T.

NIP. 198512282015041002

HALAMAN PENGESAHAN

RANCANG BANGUN PURWARUPA DATABASE CURAH HUJAN DAN
SUHU MENGGUNAKAN ESP32 BERBASIS *INTERNET OF THINGS* (IoT)

Oleh:

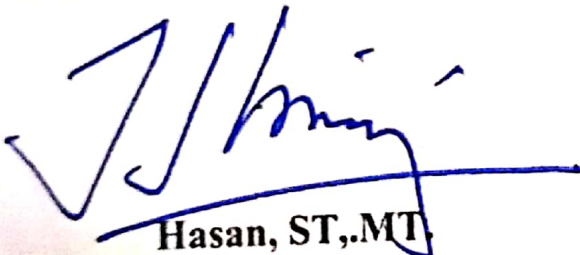
OSCAR RIZALDI

320916118

Tugas Akhir ini telah diterima dan disahkan sebagai salah satu syarat untuk
menyelesaikan Program Pendidikan Diploma III pada Program Studi
Teknik Informatika Jurusan Teknik Elektro Politeknik Negeri Pontianak

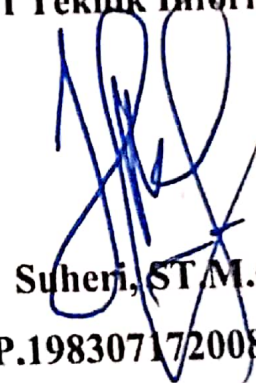
Disahkan Oleh:

Ketua Jurusan
Teknik Elektro



Hasan, ST., MT.
NIP.19710821199903003

Ketua Program Studi
D-III Teknik Informatika



Suheri, ST. M. Cs.
NIP.198307172008121005

Mengetahui,

Direktur Politeknik Negeri Pontianak

Dr. Ir. H. Muhammad Toasin Asha, M.Si.

NIP. 196112251990111001

HALAMAN PERNYATAAN ORISINALITAS

Saya yang bertanda tangan dibawah ini:

Nam : Oscar Rizaldi

NIM : 3201916118

Jurusan / Program Studi : Teknik Elektro / D-III Teknik Informatika

Judul Tugas Akhir : Rancang Bangun Purwarupa Database Curah Hujan
Dan Suhu Menggunakan ESP32 Berbasis *Internet of Things* (IoT)

Menyatakan dengan sebenar-benarnya bahwa penulisan laporan Tugas Akhir ini berdasarkan hasil penelitian, pemikiran dan pemaparan asli dari saya sendiri, baik untuk naskah laporan maupun kegiatan yang tercantum sebagai bagian dari laporan Tugas Akhir ini. Jika terdapat karya orang lain, saya akan mencantumkan sumber yang jelas.

Demikian pernyataan ini saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini dan sanksi lain sesuai dengan peraturan yang berlaku di Politeknik Negeri Pontianak.

Demikian pernyataan ini saya buat dalam keadaan sadar tanpa paksaan dari pihak manapun.

Pontianak, 25 September 2022

Yang membuat pernyataan

Oscar Rizaldi

NIM. 3201916118

RIWAYAT HIDUP



Biodata Mahasiswa

Nama Mahasiswa	: Oscar Rizaldi
NIM	: 3201916118
Tempat/Tanggal Lahir	: Muara Gerunggang, 26 Juli 2001
Jenis Kelamin	: Laki-laki
Agama	: Islam
Alamat	: Jl. Sepakat 2 gg Mutiara no A16
No. Telpn	: 082157071773
Email	: oscarizaldi0@gmail.com

ABSTRAK

Cuaca adalah keadaan atau kondisi atmosfer di wilayah tertentu yang relative sempit dan pada jangka waktu yang singkat. Hujan merupakan peristiwa jatuhnya air dalam bentuk cairan maupun padat yang dicurahkan atmosfer ke permukaan bumi. Curah hujan adalah jumlah air yang jatuh di permukaan tanah datar selama periode tertentu yang diukur dengan satuan tinggi milimeter (mm) di atas permukaan horizontal. Penakar hujan adalah instrumen yang digunakan untuk mendapatkan dan mengukur jumlah curah hujan pada satuan waktu tertentu. Dalam penelitian ini akan dilakukan suatu sistem yang digunakan untuk mengukur curah hujan dimana proses *monitoring* dapat dilakukan secara *real time* menggunakan aplikasi berbasis *web*. Curah hujan diklasifikasikan menjadi 4 kondisi, yaitu hujan ringan, hujan sedang, hujan lebat, dan hujan ekstrim. Hujan ringan berada pada level ketinggian air 0-20 mm/hari, hujan sedang berada pada level ketinggian air 20-50 mm/hari, hujan lebat berada pada level ketinggian air 50-100 mm/hari, dan hujan ekstrim berada pada level ketinggian air >100 mm/hari.

Curah Hujan, hujan, cuaca, *real time*, *web*

ABSTRACT

Weather is a condition or condition of the atmosphere in a certain area that is relatively narrow and in a short time. Rain is the event of the fall of water in liquid or solid form which is poured out by the atmosphere onto the earth's surface. Rainfall is the amount of water that falls on a flat land surface for a certain period which is measured in millimeters (mm) above the horizontal surface. Rain gauge is a tool used to obtain and measure the amount of rainfall at a certain time. In this study a system will be used to measure rainfall where the monitoring process can be carried out in real time using a web-based application. Rainfall is classified into 4 conditions, namely light rain, moderate rain, heavy rain and extreme rain. Light rain with a water level of 0-20 mm/day, moderate rain with a water level of 20-50 mm/day, heavy rain with a water level of 50-100 mm/day, and extreme rain with a water level of >100 mm/day.

Precipitation, rain, weather, real time, web

PRAKATA

Alhamdulillah, puji syukur kami panjatkan kehadiran Allah SWT, yang telah memberikan rahmat dan hidayah-Nya. Shalawat serta salam kami curahkan kepada Nabi Besar Muhammad SAW, keluarga, sahabat, serta umatnya hingga akhir zaman, sehingga pada akhirnya penulis dapat menyelesaikan tugas akhir ini dengan baik. Pada kesempatan ini penulis menyampaikan rasa terima kasih yang sebesar-besarnya kepada orang-orang yang telah berperan sehingga dapat terselesaikannya tugas akhir ini, antara lain :

1. Orang tua yang selalu memberikan nasihat dan dukungan selama berjalannya proses penyusunan Tugas Akhir.
2. Bapak Dr. Ir. H. Muh. Toasin Asha, M. Si selaku Direktur Politeknik Negeri Pontianak.
3. Bapak Hasan, ST., MT selaku Ketua Jurusan Teknik Elektro.
4. Bapak Suheri, S.T., M.Cs selaku Ketua Program Studi Teknik Informatika.
5. Bapak Muhammad Diponegoro, S.Kom., M.Cs. selaku Koordinator Tugas Akhir Program Studi Teknik Informatika.
6. Bapak Wendhi Yuniarto, S.T., M.T. selaku Dosen Pembimbing yang tiada henti memberikan bimbingan, dorongan, saran, kritik dan motivasi kepada penulis.
7. Bapak Satriyo, S.T., M.Kom. selaku Dosen Penguji 1 yang telah memberikan saran dan kritik selama proses penyusunan Tugas Akhir ini.
8. Bapak Fitri Wibowo, S.ST., M.T. selaku Dosen Penguji 2 yang juga telah memberikan saran dan kritik selama proses penyusunan Tugas Akhir ini.
9. Dosen Program Studi Diploma III Teknik Informatika Politeknik Negeri Pontianak yang telah memberi bekal ilmu kepada penulis.
10. Seluruh *staff* pengajar dan administrasi Politeknik Negeri Pontianak, khususnya di Jurusan Teknik Elektro Program Studi Teknik Informatika.
11. Teman-teman Program Studi Teknik Informatika yang selalu memberi semangat, bantuan, dan saran selama proses penyusunan Tugas Akhir ini serta lain-lainya yang tidak dapat disebutkan satu-persatu.

Penulis menyadari bahwa dalam penyusunan laporan Tugas Akhir ini masih jauh dari sempurna, untuk itu kritik dan saran dari pembaca yang membangun sangat penulis harapkan untuk menjadi pelajaran di kemudian hari. Penulis berharap semoga laporan Tugas Akhir ini dapat memberikan manfaat bagi kita semua. Terima kasih .

Pontianak, 25 September 2022

Oscar Rizaldi

DAFTAR ISI

HALAMAN PENGESAHAN.....	i
HALAMAN PERNYATAAN	ii
HALAMAN PERNYATAAN ORISINALITAS.....	iii
RIWAYAT HIDUP.....	iv
ABSTRAK	v
ABSTRACT.....	vi
PRAKATA.....	vii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xii
DAFTAR GAMBAR	xiii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan.....	2
1.5 Manfaat.....	2
1.6 Metodologi	3
1.6.1 Metodologi Pengumpulan Data.....	3
1.6.2 Metodologi Pengembangan Sistem.....	3
BAB II LANDASAN TEORI	5
2.1 Tinjauan Pustaka	5
2.2 Teori Dasar	5
2.2.1 Pengukuran Curah Hujan.....	5
2.2.2 Sensor Curah Hujan Tipe <i>Tipping Bucket</i>	6
2.2.3 ESP32	8
2.2.4 LCD 16x2 (Liquid Crystal Display)	10
2.2.5 Sensor Suhu DHT11	11
2.2.6 MQTT	12

2.2.7 ThingSpeak	12
2.2.8 PHP	13
2.2.9 Xampp.....	13
2.2.10 HTML	13
2.2.11 Laravel	13
2.2.12 MySQL	13
BAB III HASIL DAN PERANCANGAN SISTEM.....	14
3.1 Rancangan Sistem	14
3.2 Diagram Blok	14
3.3 Flowchart.....	16
3.4 Use Case Diagram	17
3.5 Perancangan Sistem Perangkat Keras	18
3.5.1 Sensor <i>Tipping Bucket</i> ke ESP32.....	18
3.5.2 DHT11 ke ESP32	19
3.5.3 LCD 16x2, dan I2C ke ESP32	19
3.5.4 RTC DS3231 ke ESP32.....	20
3.6 Perancangan Sistem IoT	20
3.7 Rancangan Sistem Perangkat Lunak	23
3.7.1 Perancangan Halaman	23
BAB IV HASIL DAN PEMBAHASAN	24
4.1 Sensor <i>Tipping Bucket</i>	24
4.2 Sensor Suhu DHT11	25
4.3 Hasil Pengujian.....	25
4.3.1 <i>Database</i>	27
4.3.2 ThingSpeak	29
4.4 Bentuk Perangkat Keras	29
4.5 Implementasi Sistem <i>Monitoring</i>	31
4.5.1 Sistem <i>Internet of Things</i>	31
4.5.2 LCD 16x2 dan I2C	33
4.6 Sistem Perangkat Lunak	33

4.6.1 Program Pembaca Sensor Curah Hujan, Suhu, LCD 16x2, dan RTC..	33
4.7 Analisa.....	43
BAB V KESIMPULAN DAN SARAN.....	44
5.1. Kesimpulan.....	44
5.2. Saran	44
DAFTAR PUSTAKA	45
LAMPIRAN.....	47
Lampiran 1.....	48
Dokumentasi.....	48
Lampiran 2.....	51
Program Arduino IDE	51
Program <i>Index.php</i>	57
Program cekhujan.php.....	60
Program cek suhu.php	60
Program cekhujanmenit.php.....	60
Program cekhujanjam.php.....	61
Program kirimdata.php	61

DAFTAR TABEL

Tabel 2. 1 Klasifikasi Curah Hujan Menurut Standar World Meteorological Organization (WMO) [4]	6
Tabel 2. 2 Spesifikasi ESP32	9
Tabel 4. 1 Spesifikasi Tipping Bucket	27
Tabel 4. 2 Data Database Hasil Pengukuran 27 Agustus 2022.....	27

DAFTAR GAMBAR

Gambar 1. 1 Gamabar Metodologi Pengembangan	3
Gambar 2. 1 Mekanisme Sensor Tipping Bucket	7
Gambar 2. 2 Sensor Tipping Bucket	8
Gambar 2. 3 Modul ESP32	9
Gambar 2. 4 LCD 16x2 (Liquid Crystal Display).....	10
Gambar 2. 5 Modul Sensor Suhu DHT11	12
Gambar 3. 1 Blok Diagram Sistem	14
Gambar 3. 2 Flowchart.....	16
Gambar 3. 3 Use Case Diagram	17
Gambar 3. 4 Skematik Tipping Bucket	18
Gambar 3. 5 Cloud Server ThingSpeak	20
Gambar 3. 6 Tampilan awal ThingSpeak.....	21
Gambar 3. 7 Tampilan Channel ThingSpeak	21
Gambar 3. 8 Tampilan data keluaran	22
Gambar 3. 9 Tampilan Website	23
Gambar 4. 1 Sensor Curah Hujan	24
Gambar 4. 2 Tutup Sensor Curah Hujan	25
Gambar 4. 3 Sensor DHT11	25
Gambar 4. 4 Proses Pengambilan Data	26
Gambar 4. 5 Grafik Data Curah Hujan Dan Suhu	28
Gambar 4. 6 Monitoring Data Sistem IoT ThingSpeak	29
Gambar 4. 7 Implementasi Alat	30
Gambar 4. 8 Implementasi Sensor Hujan atau Tipping Bucket.....	31
Gambar 4. 9 Tampilan Channel ThingSpeak	32
Gambar 4. 10 Implementasi LCD 16x2 dan ESP32	33
Dokumentasi 1	48
Dokumentasi 2	48
Dokumentasi 3	49
Dokumentasi 4	49
Dokumentasi 5	50
Dokumentasi 6	50

BAB I

PENDAHULUAN

1.1 Latar Belakang

Hujan secara kondisi geografis Indonesia adalah salah satu kejadian alam yang disebabkan oleh angin Muson Barat, yaitu angin yang bergerak dari arah utara dan atau barat laut menuju ke selatan atau dari benua Asia menuju benua Australia. Angin Muson Barat ini membawa banyak uap air sehingga ketika melewati Indonesia menyebabkan turunnya hujan, kondisi ini biasa disebut dengan musim penghujan. Musim penghujan di Indonesia biasanya memiliki periode antara bulan Oktober sampai April. Hujan adalah paremeter, sehingga hujan dapat di ukur kuantitasnya.

Pemanfaatan teknologi dalam kehidupan sehari-hari manusia tidak terlepas dari berbagai sensor. Pemanfaatan sistem sensor ini mendukung manusia dalam melakukan pekerjaan. Contohnya dalam melakukan pemantauan curah hujan yaitu dengan menggunakan sensor curah hujan. Sensor curah hujan merupakan sensor yang tersusun dari flat yang apabila diberi tetesan air akan memberikan perubahan nilai tegangan outputnya. Secara klimatologi curah hujan mempunyai siklus. Memulai analisis dan pengamatan curah hujan yang cukup panjang maka penulis dapat mengetahui data curah hujan beserta sifat-sifatnya.

Seiring dengan perkembangan teknologi informasi yang sangat pesat terkadang informasi cuaca dapat diakses dengan mudah oleh siapa pun dan tidak ada batasan waktu juga ketika ingin mengakses dan mengambil data informasi cuaca tersebut. Infomasi curah hujan yang terjadi (*realtime*) sangat penting sekali bagi masyarakat yang membutuhkan, terutama bagi para nelayan yang akan melaut, pengendara roda dua akan melakukan perjalanan, maskapai penerbangan yang akan melakukan penerbangan. Sedangkan data curah hujan tahunan sangat dibutuhkan oleh dinas pekerjaan umum yang akan melakukan pembangunan pada suatu daerah, dinas pertanian yang akan melakukan penelitian pada suatu daerah, dan pihak-pihak lain yang ingin melakukan analisis ketinggian curah hujan pada suatu daerah tertentu. Dalam penelitian ini juga sebagai informasi apabila intensitas curah hujannya tinggi kemungkinan air di permukaan akan meluap dan akan menyebabkan banjir. Untuk itu dengan adanya latar belakang diatas dibuatlah alat dengan menggunakan sensor

curah hujan yaitu “ Rancang Bangun Purwarupa Database Curah Hujan Dan Suhu Menggunakan ESP32 Berbasis *Internet of Things* (IoT) ”.

1.2 Rumusan Masalah

Rumusan masalah yang diusulkan dalam penulisan tugas akhir ini adalah : Bagaimana rancang bangun purwarupa database Curah Hujan dan Suhu Menggunakan ESP32 Berbasis *Internet of Things* (IoT).

1.3 Batasan Masalah

Dalam pembuatan tugas akhir ini penulis membatasi permasalahan sebagai berikut:

1. Unsur keadaan lingkungan yang akan di *monitoring* ini adalah curah hujan.
2. Sensor yang digunakan dalam penelitian ini adalah sensor *tipping bucket* dan sensor suhu DHT11.
3. Menggunakan ESP32 sebagai mikrokontroler pengolah data curah hujan.
4. Menggunakan *ThingSpeak* sebagai *platform* IoT.
5. Menggunakan protokol MQTT.
6. Hasil pengukuran dari intensitas curah hujan ini akan ditampilkan pada website.
7. Data hasil pengukuran bisa diakses oleh pengguna secara *online* menggunakan koneksi *internet*.

1.4 Tujuan

Berdasarkan latar belakang yang telah dipaparkan, tujuan dari Sistem Monitoring Curah Hujan adalah sebagai berikut:

1. Menghasilkan alat yang dapat memonitor besarnya curah hujan yang terjadi pada suatu tempat secara *real time* berbasis IoT.
2. Memberikan data informasi curah hujan yang ada dipermukaan.

1.5 Manfaat

Manfaat dari pembuatan tugas akhir ini adalah :

1. Sebagai acuan atau rujukan untuk pengembangan cara pengaksesan data curah hujan secara jarak jauh dan *real time* bagi peneliti.
2. Memantau data curah hujan sehingga bisa digunakan untuk memberikan informasi besarnya curah hujan dan juga sebagai peringatan untuk persiapan datangnya banjir jika besarnya curah hujan sudah melebihi batas aman bagi masyarakat di daerah rawan banjir.

1.6 Metodologi

Metodologi yang digunakan dalam penyusunan Tugas Akhir ini adalah sebagai berikut:

1.6.1 Metodologi Pengumpulan Data

Teknik pengumpulan data yang dilakukan dalam penyusunan proposal tugas akhir ini adalah sebagai berikut :

1. Studi Literatur

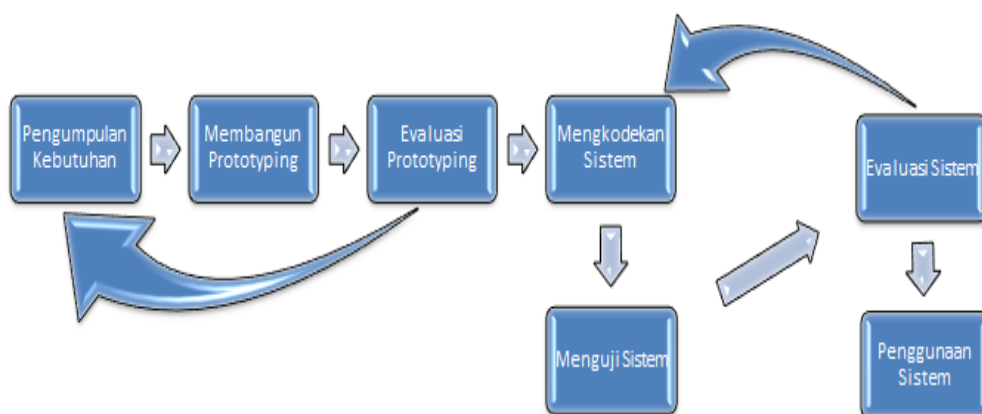
Studi Literatur, pengumpulan data yang berhubungan dengan judul tugas akhir dengan cara mengumpulkan dan membaca buku-buku, *e-book*, jurnal yang mencakup tentang permasalahan yang ada pada latar belakang, teori-teori tentang curah hujan, ESP32, *platform* IoT serta penunjang lainnya tentang implementasi alat yang dibuat.

2. Observasi

Observasi merupakan salah satu metode pengumpulan data yang dilakukan dengan cara pengamatan langsung di Kabupaten Ketapang.

1.6.2 Metodologi Pengembangan Sistem

Adapun metodologi penelitian yang dilakukan untuk menyelesaikan berbagai permasalahan yang terjadi dengan menggunakan metode *prototype* serta dengan langkah-langkah berikut:



Gambar 1. 1 Gamabar Metodologi Pengembangan

a. Pengumpulan Kebutuhan

Penulis melakukan pendefinisikan format seluruh *hardware* dan *software*, mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.

b. Membangun *Prototype*

Dalam perancangan penulis membangun *prototyping* dengan membuat perancangan sementara yang berfokus pada penyajian (misalnya dengan membuat input dan format output). Sistem *monitoring* curah hujan terdiri atas perangkat keras (*hardware*) dan perangkat lunak (*software*). Perangkat keras berupa sensor curah hujan tipe *tipping bucket*, sensor suhu DHT11, ESP32, dan LCD 16×2 (*Liquid Crystal Display*). Sedangkan perangkat lunak berupa program yang akan menjalankan fungsional sistem sehingga data dapat terbaca dan akan dikirim ke *website*.

c. Evaluasi *Prototyping*

Penulis melakukan evaluasi apakah *prototyping* yang sudah dibangun sudah sesuai. Jika sudah sesuai maka langkah 4 akan di ambil. Jika tidak *prototyping* direvisi dengan mengulang langkah 1, 2, dan 3.

d. Mengkodekan Sistem

Dalam tahap ini *prototyping* yang sudah sesuai akan diterjemahkan ke dalam Bahasa pemrograman yang sesuai.

e. Menguji Sistem

Pada tahap ini penulis melakukan pengujian atau di tes pada sistem yang sudah menjadi satu perangkat lunak (*software*) yang siap pakai.

f. Evaluasi Sistem

Penulis melakukan evaluasi sistem untuk memastikan apakah sistem yang sudah dibuat telah sesuai dengan yang di inginkan.

g. Menggunakan Sistem

Disini sistem telah di uji dan telah sesuai dengan yang diharapkan maka sistem siap untuk digunakan.

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Beberapa referensi yang telah didapatkan yaitu Ignatus Arias Nurdiyanto (2019) telah melakukan penelitian dengan judul “*Monitoring Data Curah Hujan Berbasis Internet of Things* “. Pada penelitian tersebut membuat tentang perancangan sistem data curah hujan yang dapat di *monitoring* dimanapun dan kapanpun melalui perangkat *android* selama tersedia koneksi *internet* [1].

Kasus kedua Rio Simpana Putra (2017) telah melakukan penelitian dengan judul “Rancang Bangun Sistem *Monitoring* Intensitas Curah Hujan Harian Berbasis IoT (*Internet of Things*) Menggunakan Android”. Dalam penelitian tersebut akan dilakukan rancang bangun sistem yang digunakan untuk mengukur intensitas curah hujan dimana proses *monitoring* dapat dilakukan secara *real time* menggunakan aplikasi pada *Android* [2].

Kasus ketiga telah melakukan penelitian dengan judul “*Prototype Buka Tutup Atap Otomatis Untuk Menjemur Padi Dan Solar Panel Berbasis Arduino Uno*”. Dalam penelitian ini dengan menggunakan sensor curah hujan dan sensor LDR juga menggunakan motor stepper sebagai penggerak atap berfungsi memudahkan petani untuk menjemur padi tanpa khawatir hujan dan menggunakan solar panel untuk memanfaatkan sinar matahari saat menjemur [3].

Adapun perbedaan dari 3 kasus di atas dengan pembuatan proposal TA yang ditulis adalah pada 3 kasus di atas menggunakan sistem aplikasi berbasis *android* sedangkan proposal TA dalam penelitian ini penulis menggunakan *website*.

2.2 Teori Dasar

2.2.1 Pengukuran Curah Hujan

Hujan merupakan peristiwa turunnya air atau kristal (hujan es/salju) dari langit sampai ke permukaan bumi atau tanah akibat dari penguapan air laut. Hujan adalah parameter, sehingga hujan dapat diukur kuantitasnya. Sedangkan menurut BMKG dalam websitenya “Curah Hujan (mm) adalah ketinggian air hujan yang terkumpul dalam penakar hujan pada tempat yang datar, tidak menyerap, tidak meresap dan tidak mengalir”. Pengukuran curah hujan adalah mengetahui tinggi air hujan yang

menggenangi suatu bidang datar atau tanah pada suatu daerah. Dalam satu milimeter curah hujan berarti pada luasan satu meter persegi bidang datar tertampung air hujan setinggi satu milimeter atau tertampung volume air hujan sebesar satu liter. Besarnya nilai intensitas curah hujan digunakan untuk menentukan kriteria hujan yang terjadi, yaitu dari kriteria Sangat Ringan sampai Sangat Lebat. Untuk klasifikasi curah hujan ditunjukkan pada Tabel 1 [4].

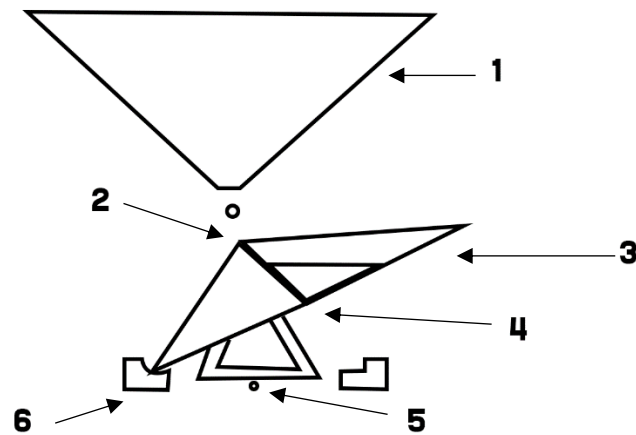
Tabel 2. 1 Klasifikasi Curah Hujan Menurut *Standar World Meteorological Organization (WMO)* [4]

Kriteria Hujan	Intensitas Hujan
Sangat Ringan	<5.0 mm
Ringan	5.0 – 20 mm
Sedang / Normal	20 – 50 mm
Lebat	50 – 100 mm
Sangat Lebat	>100 mm

2.2.2 Sensor Curah Hujan Tipe *Tipping Bucket*

Penakar curah hujan adalah alat yang digunakan untuk mengukur besarnya curah hujan yang terjadi pada satuan waktu tertentu. Alat pengukur curah hujan dibagi menjadi dua jenis yaitu tipe manual dan tipe otomatis. *Sensor* curah hujan tipe *Tipping Bucket* adalah termasuk alat penakar hujan otomatis. Prinsip kerjanya adalah ketika hujan turun, air akan masuk melalui corong yang biasanya berbentuk kerucut terbalik, lalu tetesan air dari ujung corong ditampung oleh penampung berayun atau biasa disebut dengan *Tipping Bucket*. Alat penampung berayun ini terdiri dari dua wadah yang diisi bergantian, setiap kali salah satu sisi wadah terisi penuh maka alat menjadi tidak seimbang dan bagian yang terisi air akan turun ke bawah untuk mengosongkan air dalam wadah dan membuang air ke saluran pembuangan dan wadah pada sisi yang lain akan naik untuk menampung tetesan air hujan seperti wadah sebelumnya. Setiap jatuhnya wadah penampung air akan mengaktifkan *reed switch* magnetik yang hasilnya direkam oleh *data logger*. Cara kerja dari *reed switch* adalah ketika ada medan magnet mengenai bagian depan *sensor*, maka *sensor* akan bekerja sehingga menghubungkan kontak. Posisi *sensor* menempel dengan badan silinder, sehingga saat silinder bergerak naik atau turun

maka akan ada medan magnet yang mengenai *reed switch* [5]. Mekanisme sensor *tipping bucket* ditunjukkan pada Gambar 2.1.



Gambar 2. 1 Mekanisme Sensor *Tipping Bucket*

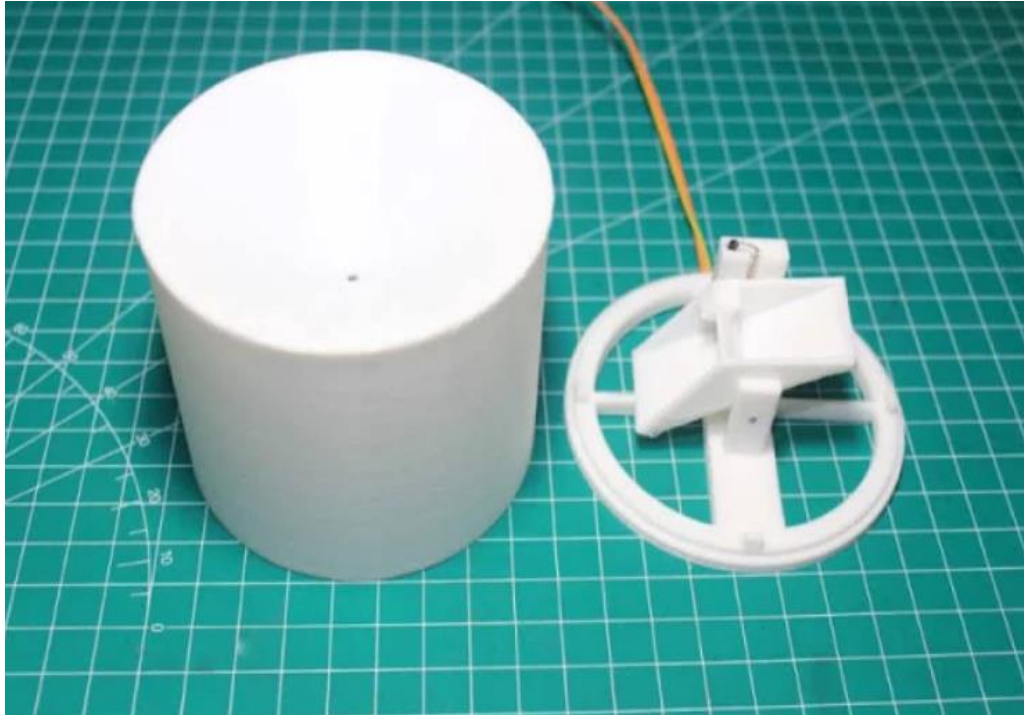
Keterangan nomor pada Gambar 2.1 :

1. Bagian kerucut terbalik atau penerima hujan
2. Engsel pemutar
3. TB (*Tipping Bucket*) wadah yang menampung air hujan
4. Magnet
5. *Sensor Reed Switch*
6. Saluran pembuangan air hujan

Nilai curah hujan pada penakar hujan tipe *Tipping Bucket* dapat diketahui menggunakan persamaan :

$$CH = \frac{JT \times V}{L}$$

Dengan CH adalah Curah Hujan (mm), JT adalah Jumlah *Tipping*, V adalah volume per *tipping* (ml) , dan L adalah luas mulut corong (cm²).



Gambar 2. 2 Sensor *Tipping Bucket*

2.2.3 ESP32

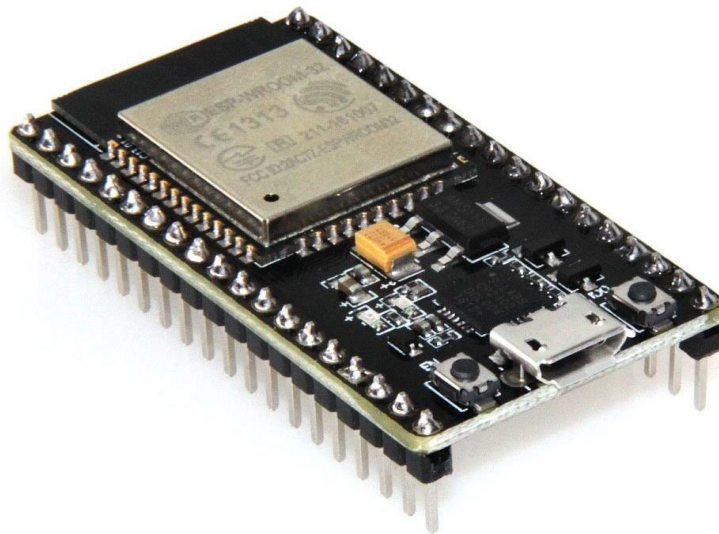
Mikrokontroler ESP32 dibuat oleh perusahaan bernama *Espressif Systems*. Salah satu kelebihan yang dimiliki oleh ESP32 yaitu sudah terdapat *Wi-Fi* dan *Bluetooth* di dalamnya, sehingga akan sangat memudahkan ketika penulis belajar membuat sistem IoT yang memerlukan koneksi wireless. Mikrokontroler ESP32 memiliki keunggulan yaitu sistem berbiaya rendah, dan juga berdaya rendah dengan modul *WiFi* yang terintegrasi dengan chip mikrokontroler serta memiliki bluetooth dengan mode ganda dan fitur hemat daya menjadikannya lebih fleksibel.

ESP32 adalah mikrokontroler yang dikenalkan oleh *Espressif System* merupakan penerus dari mikrokontroler ESP8266. Pada mikrokontroler ini sudah tersedia modul *WiFi* dalam chip sehingga sangat mendukung untuk membuat sistem aplikasi *Internet of Things*. ESP32 sendiri tidak jauh berbeda dengan ESP8266 yang familiar di pasaran, hanya saja ESP32 lebih kompleks dibandingkan ESP8266, cocok untuk sobat dengan proyek yang besar [6].

berikut ini merupakan spesifikasi yang dimiliki oleh mikrokontroler ESP32:

Tabel 2. 2 Spesifikasi ESP32

Atribut	Detail
CPU	Tensilica Xtensa LX6 32bit Dual-Core di 160/240MHz
SRAM	520 KB
FLASH	2MB (max. 64MB)
Tegangan	2.2V sampai 3.6V
Arus Kerja	Rata-rata 80mA
Dapat diprogram	Ya (C,C++.Phyton, Lua, dll)
Open Source	Ya
Konektivitas	
Wi-Fi	802.11 b/g/n
Bluetooth	4.2BR/EDR + BLE
UART	3
I/O	
GPIO	32
SPI	4
I2C	2
PWM	8
ADC	18 (12-bit)
DAC	2 (8-bit)



Gambar 2. 3 Modul ESP32

Keterangan :

- GPIO-0 – GPIO-3 : Input Output
- VCC : Tegangan masuk 3.3 Vdc
- GND : Ground
- Reset
- Chip Enable

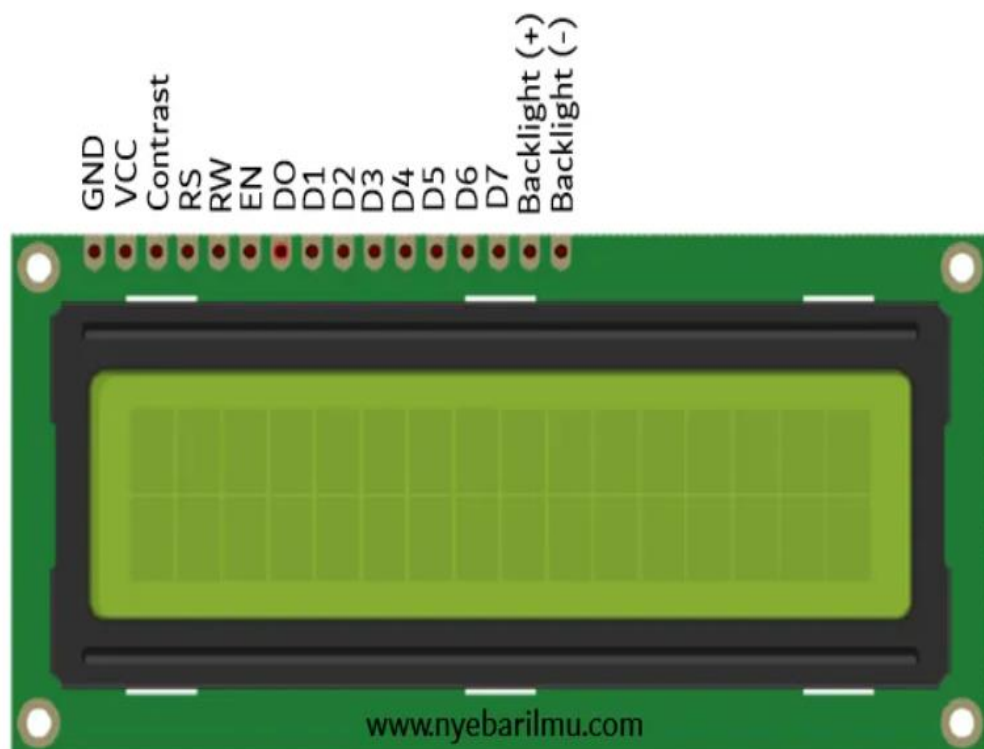
2.2.4 LCD 16x2 (Liquid Crystal Display)

LCD 16×2 (*Liquid Crystal Display*) merupakan modul penampil data yang mepergunakan kristal cair sebagai bahan untuk penampil data yang berupa tulisan maupun gambar. Pengaplikasian pada kehidupan sehari – hari yang mudah dijumpai antara lain pada kalkulator, gamebot, televisi, atau pun layar komputer.

Jenis dari perangkat ini ada yang dan pada postingan ini akan dibahas tentang Tutorial Arduino mengakses LCD 16×2 dengan mudah, dimana mudah didapatkan ditoko elektronik terdekat [7].

Adapun fitur – fitur yang tersedia di LCD 16 x 2 antara lain sebagai berikut :

- Terdiri dari 16 kolom dan 2 baris
- Dilengkapi dengan back light
- Mempunyai 192 karakter tersimpan
- Dapat dialamati dengan mode 4-bit dan 8-bit
- Terdapat karakter generator terprogram



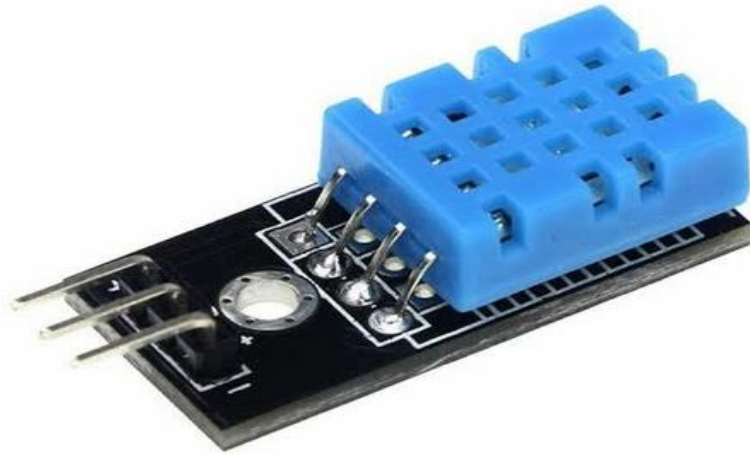
Gambar 2. 4 LCD 16x2 (Liquid Crystal Display)

Keterangan :

1. GND : catu daya 0Vdc
2. VCC : catu daya positif
3. *Constrate* : untuk kontras tulisan pada LCD
4. RS atau *Register Select* :
 - *High* : untuk mengirim data
 - *Low* : untuk mengirim instruksi
5. R/W atau *Read/Write*
 - *High* : mengirim data
 - *Low* : mengirim instruksi
 - Disambungkan dengan LOW untuk pengiriman data ke layar
6. E (*enable*) : untuk mengontrol ke LCD ketika bernilai LOW, LCD tidak dapat diakses
7. D0 – D7 = Data Bus 0 – 7
8. *Backlight +* : disambungkan ke VCC untuk menyalakan lampu latar
9. *Backlight –* : disambungkan ke GND untuk menyalakan lampu latar

2.2.5 Sensor Suhu DHT11

Sensor DHT11 adalah salah satu jenis sensor yang banyak digunakan pada project berbasis Arduino. Sensor ini memiliki keunikan yaitu dapat membaca suhu (temperature) ruangan dan kelembapan udara (humidity). Sensor ini dikemas dalam bentuk kecil dan ringkas, serta harganya yang terjangkau. Harga sensor DHT11 ini hanya 1,2\$ USD atau sekitar 15 ribu rupiah saja. Kegunaan sensor DHT11 ini biasanya dipakai pada project monitoring suhu ruangan maupun kelembapan udara pada ruangan oven [8].



Gambar 2. 5 Modul Sensor Suhu DHT11

2.2.6 MQTT

Message Queue Telemetry Transport (MQTT) adalah sebuah protokol komunikasi data *machine to machine* (M2M) yang berada pada layer aplikasi, MQTT bersifat *lightweight message* artinya MQTT berkomunikasi dengan mengirimkan data pesan yang memiliki *header* berukuran kecil yaitu hanya sebesar 2 bytes untuk setiap jenis data, sehingga dapat bekerja di dalam lingkungan yang terbatas sumber dayanya seperti kecilnya *bandwidth* dan terbatasnya sumber daya listrik, selain itu protokol ini juga menjamin terkirim nya semua pesan walaupun koneksi terputus sementara, protokol MQTT menggunakan metode *publish/subscribe* untuk metode komunikasinya. *Publish/subscribe* sendiri adalah sebuah pola pertukaran pesan didalam komunikasi jaringan dimana pengirim data disebut *publisher* dan penerima data disebut dengan *subscriber* [9].

2.2.7 ThingSpeak

ThingSpeak adalah platform IOT yang memungkinkan peneliti untuk mengumpulkan, menyimpan, menganalisis, memvisualisasikan, dan bertindak atas data dari sensor atau aktuator, seperti Arduino, Raspberry Pi , BeagleBone Hitam, dan perangkat keras lainnya. Misalnya, dengan *ThingSpeak* dapat membuat aplikasi sensor-logging, aplikasi pelacakan lokasi. *ThingSpeak* berfungsi sebagai pengumpul data yang mengumpulkan data dari perangkat node dan juga memungkinkan data yang akan diambil ke dalam lingkungan perangkat lunak untuk analisis historis data [5].

2.2.8 PHP

PHP merupakan singkatan dari *Hypertext Preprocessor* adalah Bahasa pemrograman yang digunakan untuk mengembangkan situs *website* statis atau situs *website* dinamis [10].

2.2.9 Xampp

Xampp adalah perangkat lunak berbasis web server yang bersifat *open source* (bebas), serta mendukung di berbagai sistem operasi, baik Windows, Linux, atau Mac OS. Xampp digunakan sebagai *standalone server* (berdiri sendiri) atau biasa disebut dengan *localhost*. Hal tersebut memudahkan dalam proses pengeditan, desain, dan pengembangan aplikasi [11].

2.2.10 HTML

HTML adalah singkatan dari Hypertext Markup Language yang memiliki pengertian bahasa markup standar untuk membuat dan menyusun halaman pada aplikasi website. Atau HTML juga bisa diartikan sebagai pondasi awal dalam mengembangkan halaman website yang disusun secara terstruktur dari segi tampilan desain dan fungsinya [12].

2.2.11 Laravel

Laravel adalah salah satu framework PHP yang dirancang untuk membangun situs website mulai dari proyek kecil hingga besar. Laravel mudah dipahami karena mengikuti pola MVC (Model View Controller). Framework PHP ini menjadi pilihan yang dapat diandalkan karena memiliki kinerja, fitur serta skalabilitas yang baik [13].

2.2.12 MySQL

MySQL adalah sebuah DBMS (*Database Management System*) menggunakan perintah SQL (*Structured Query Language*) yang banyak digunakan saat ini dalam pembuatan aplikasi berbasis website. MySQL dibagi menjadi dua lisensi, pertama adalah *Free Software* dimana perangkat lunak dapat diakses oleh siapa saja. Dan kedua adalah *Shareware* dimana perangkat lunak berpelik memiliki batasan dalam penggunaannya [6].

BAB III

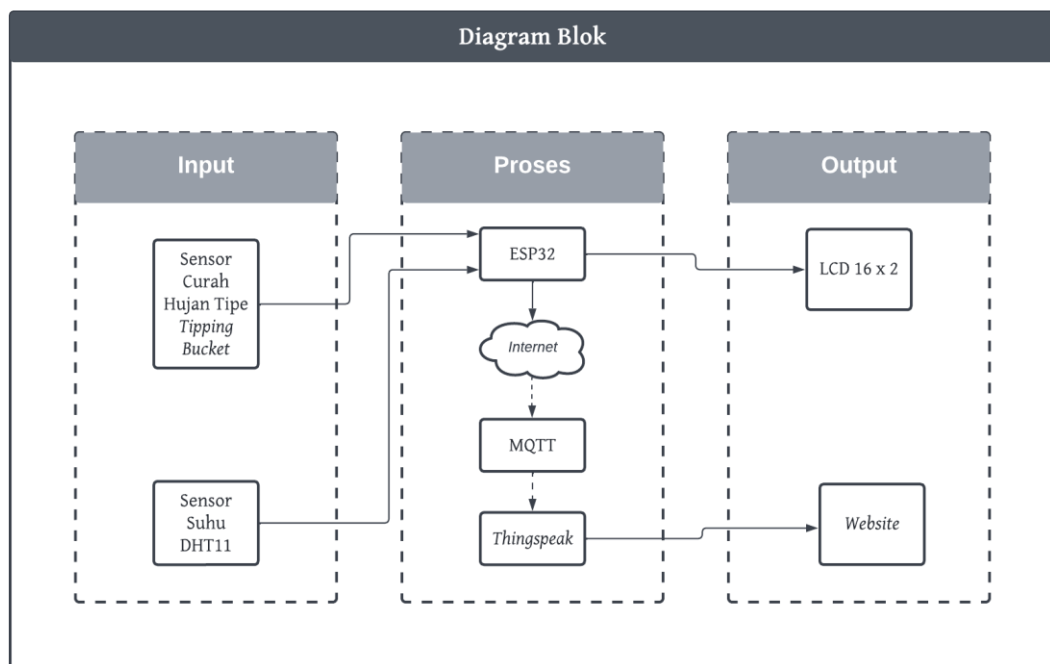
HASIL DAN PERANCANGAN SISTEM

3.1 Rancangan Sistem

Adapun rancangan sistem dalam penyusunan proposal tugas akhir ini terjadi dari *Block Diagram*, *Flowchart*, dan *Use Case Diagram*.

3.2 Diagram Blok

Diagram blok merupakan suatu pernyataan hubungann yang berurutan dari suatu atau lebih komponen yang memiliki kesatuan kerja tersendiri, dan setiap blok komponen mempengaruhi komponen yang lainnya. Diagram blok merupakan salah satu cara yang paling sederhana untuk menjelas cara kerja dari suatu sistem. Dengan diagram blok kita dapat menganalisa cara kerja rangkaian dari merancang hardware yang akan dibuat secara umum. Adapun diagram blok dari sistem yang dirancang seperti yang diperlihatkan pada gambar 3.1.



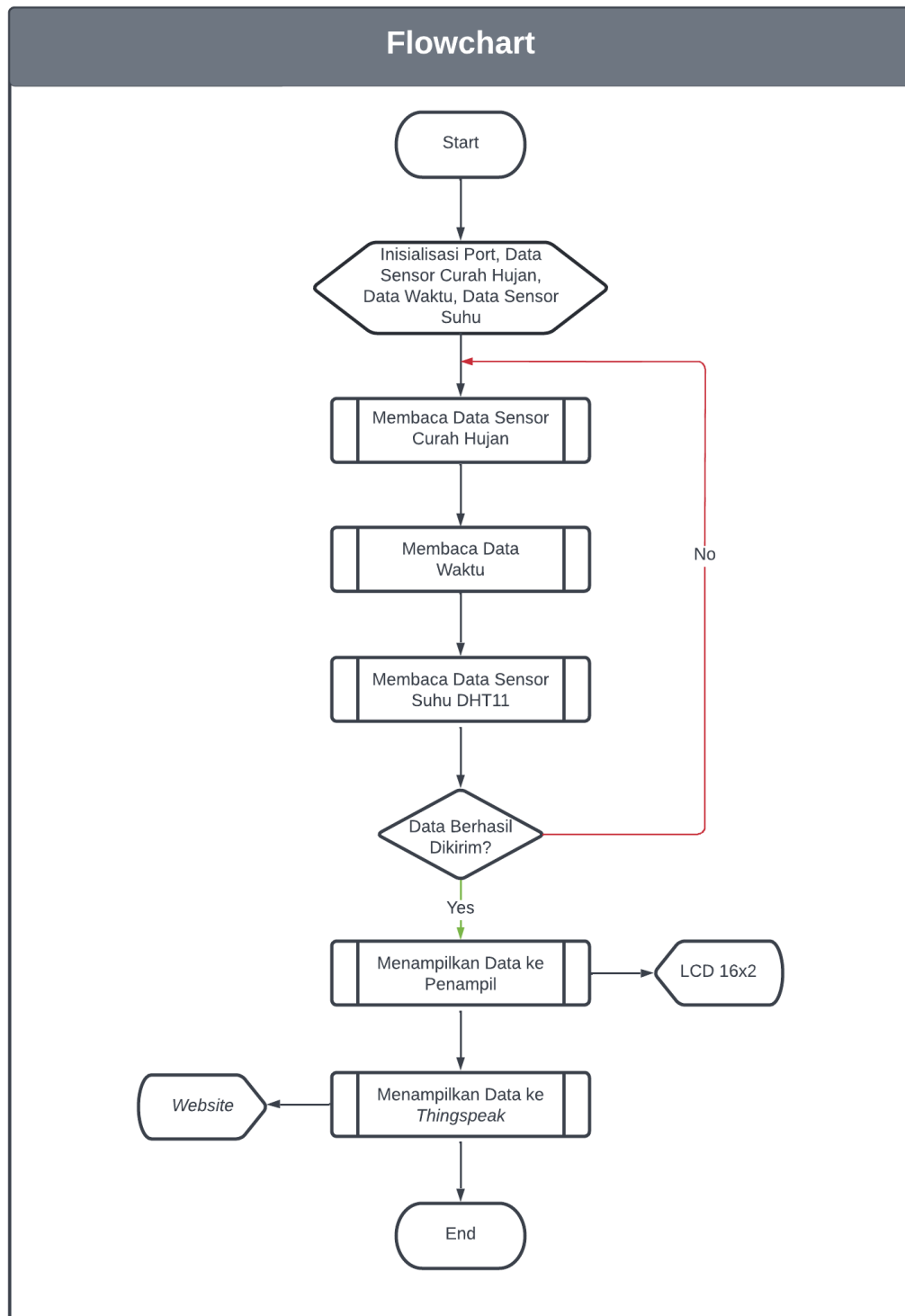
Gambar 3. 1 Blok Diagram Sistem

1. *Tipping Bucket* berfungsi sebagai penampung curah hujan, dimana pada saat hujan turun, maka tetesan hujan akan masuk melalui corong yang

terdapat dalam *tipping bucket*. Sensor suhu DHT11 berfungsi sebagai inputan data suhu yang ada di tempat tersebut, dan dari 2 sensor ini akan dikirimkan ke mikrokontroler.

2. Mikrokontroler akan mengolah data dan menampilkannya di LCD 16x2.
3. Mikrokontroler akan tersambung dengan wifi atau *internet* melalui ESP32 dan mengirimkan data ke *website*.

3.3 Flowchart

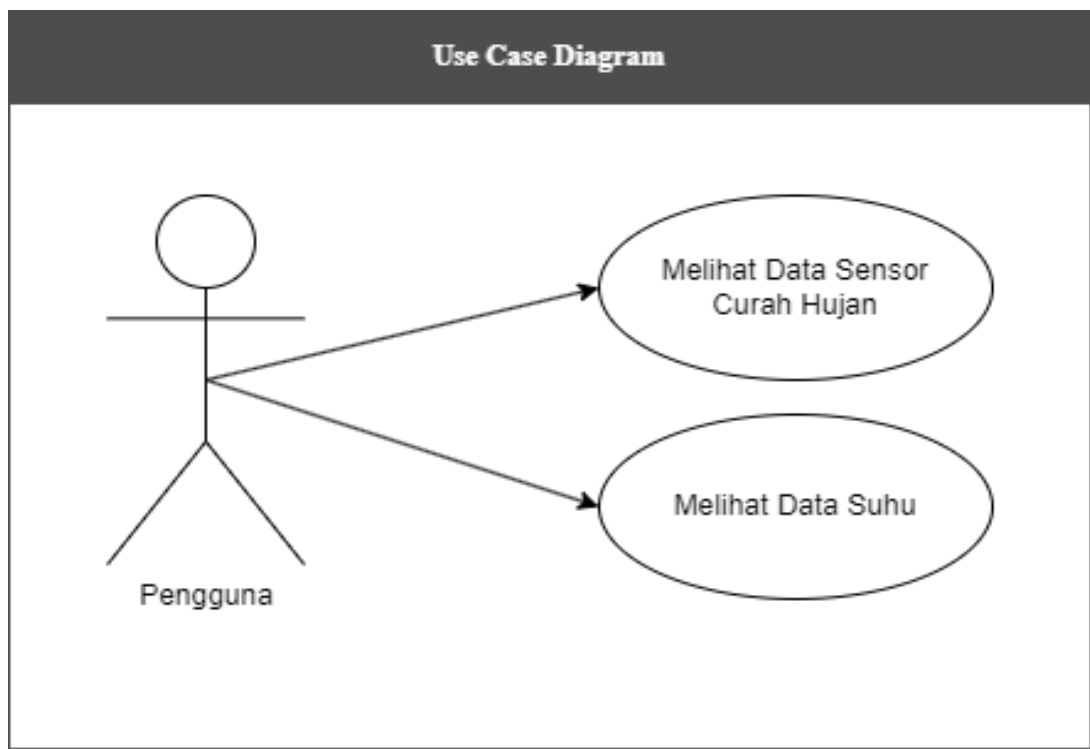


Gambar 3. 2 Flowchart

Rencana Diagram *Flowchart* Pemrosesan Data ini ditunjukkan oleh Gambar 10. Program pengolahan data akan berjalan ketika alat dihidupkan atau alat

mendapat daya masukan. Setelah *Start/Mulai* program akan menginisialisasi variable, terminal atau *port-port* yang digunakan sebagai masukan dan keluaran untuk menghubungkan peralatan sensor dan penampil keluaran dengan ESP32 yang digunakan sebagai *server*. Masukan pada *server* ini yaitu data curah hujan hasil pengukuran dari sensor hujan atau *rain gauge*, data waktu, dan data suhu dari sensor DHT11. Ketika data masukan sudah tersedia, *server* akan membaca data masukan yang berupa data curah hujan, data waktu, dan data suhu tersebut.

3.4 Use Case Diagram

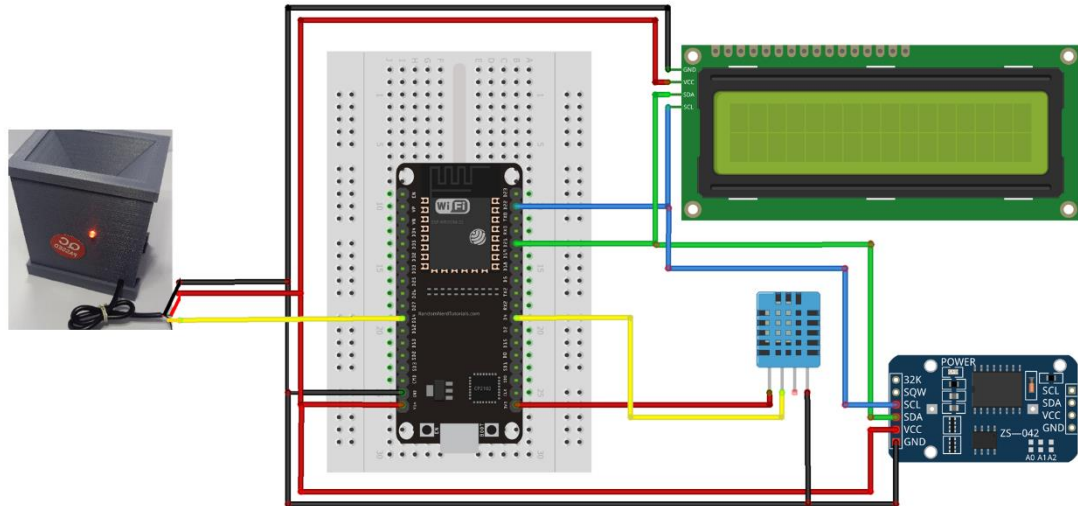


Gambar 3. 3 *Use Case Diagram*

Use case diagram menggambarkan hubungan antara pengguna dan kegiatan yang dapat dilakukannya terhadap aplikasi berbasis *web*. Pada Gambar 11 bisa diketahui pengguna bisa melihat di dalam aplikasi berbasis *web* tersebut yaitu data sensor curah hujan tipe *tipping bucket* data waktu, dan data suhu dan dari DHT11.

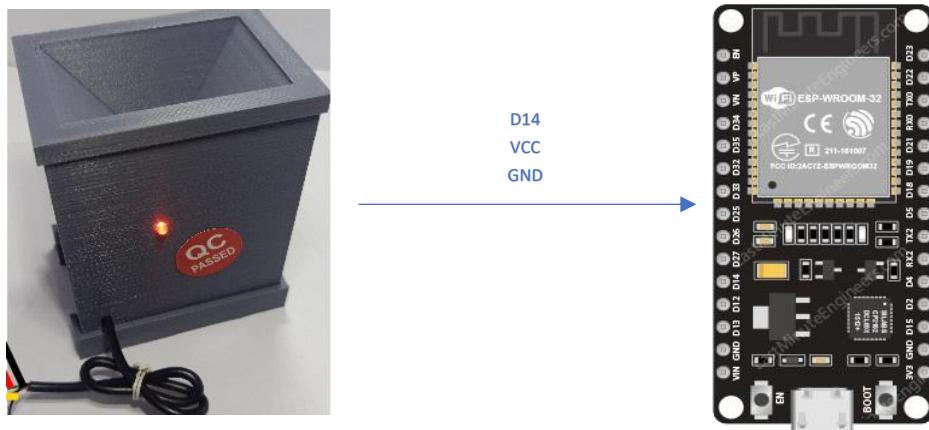
3.5 Perancangan Sistem Perangkat Keras

Berikut hubungan antara setiap input dan output terhadap mikrokontroler,



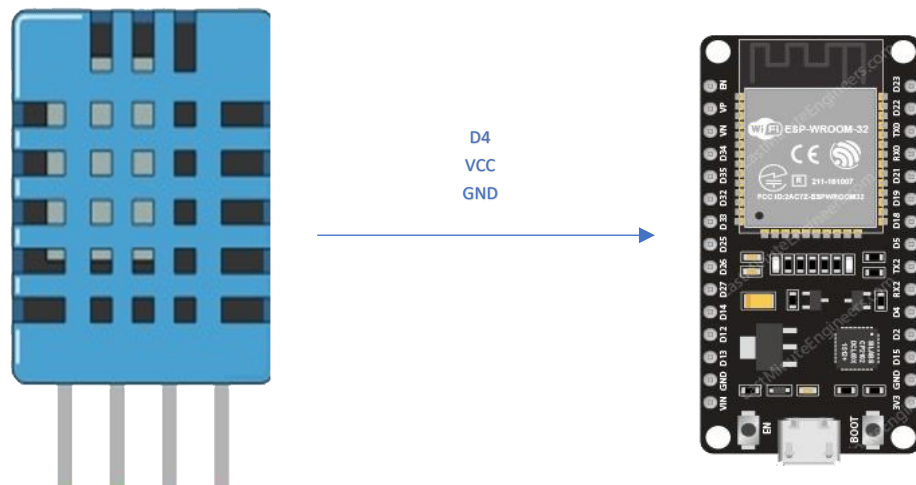
Gambar 3. 4 Skematic Tipping Bucket

3.5.1 Sensor *Tipping Bucket* ke ESP32



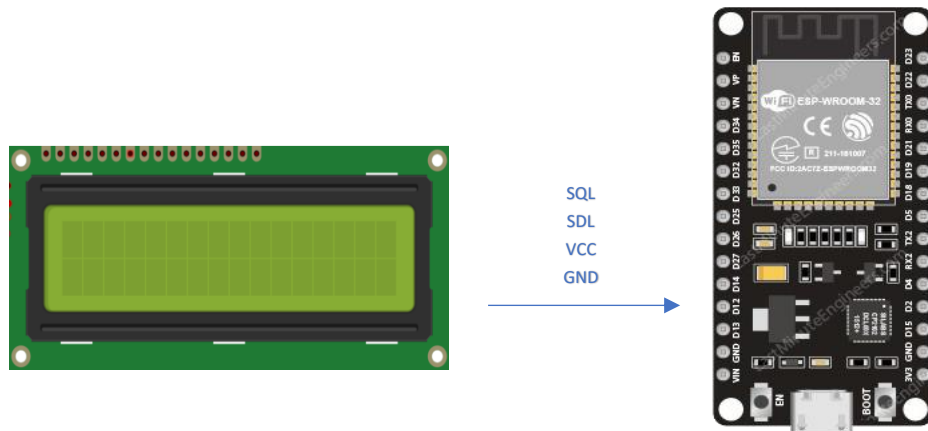
Pin data yang terhubung dari sensor *tipping bucket* ke ESP32 yaitu D14, VCC, GND.

3.5.2 DHT11 ke ESP32



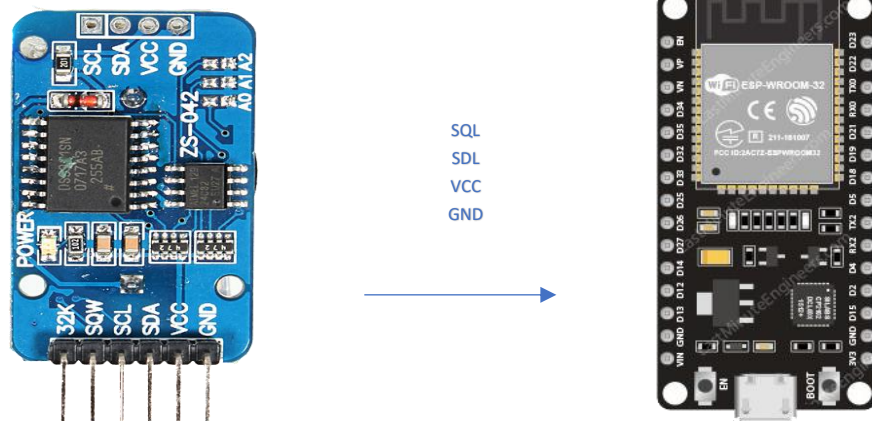
Pin data yang terhubung dari sensor DHT11 ke ESP32 yaitu D4, VCC, GND.

3.5.3 LCD 16x2, dan I2C ke ESP32



Pin pada LCD 16x2 yang menggunakan I2C terhubung dengan ESP32 terhubung ke pin SQL, SDL, VCC, GND.

3.5.4 RTC DS3231 ke ESP32



Pin pada RTC DS3231 terhubung dengan ESP32 terhubung ke pin SQL, SDL, VCC, GND.

3.6 Perancangan Sistem IoT

Proses monitoring pada alat dirancang untuk dapat diakses jarak jauh tanpa perantara kabel penghubung. Untuk dapat melakukan hal tersebut sistem dirancang menggunakan sistem IoT. *Cloud server* menggunakan ThingSpeak yang merupakan salah satu *platform* IoT yang memungkinkan untuk menampilkan dan menganalisis aliran data langsung di *cloud* secara *real time*. Untuk dapat membaca data dari ThingSpeak sebuah PC/laptop yang terhubung dengan koneksi *internet*. Skema sistem IoT ini ditunjukkan oleh Gambr 3.4.



Gambar 3. 5 *Cloud Server ThingSpeak*

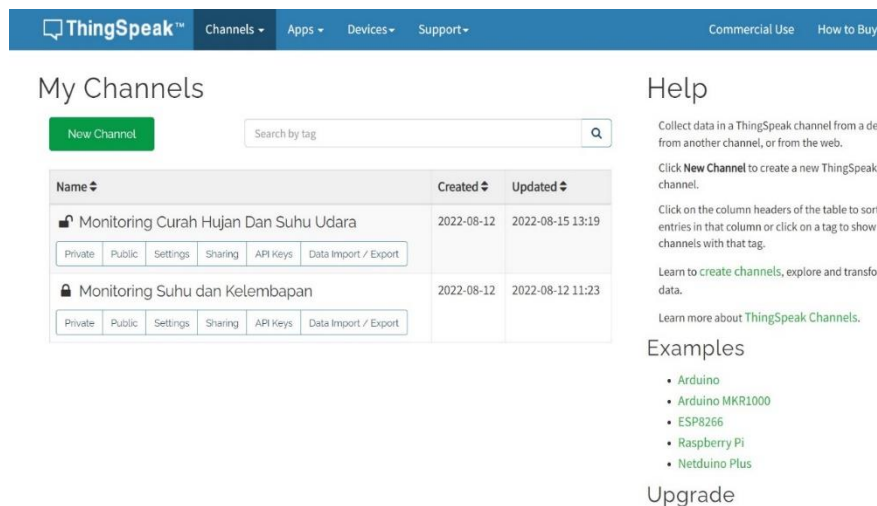
Untuk dapat menggunakan ThingSpeak pengguna harus membuat akun terlebih dahulu. Penulis selaku pengguna telah membuat akun ThingSpeak. Langkah pertama dalam membuat akun adalah dengan mengunjungi *website*

thingspeak.com lalu pilih menu *Sign Up* yang terdapat pada tampilan awal *website*. Langkah-langkah membuat akun ThingSpeak secara rinci ditunjukkan pada Gambar 3.5 – Gambar 3.6. Gambar 3.5 menunjukkan tampilan awal pada *website* *thingspeak.com*.



Gambar 3. 6 Tampilan awal *ThingSpeak*

Setelah membuat akun langkah selanjutnya akan diminta untuk membuat saluran atau *Channel*. *Channel* ini berfungsi untuk menyimpan data yang dikumpulkan oleh aplikasi ThingSpeak. Langkah membuat *channel* ditunjukkan oleh Gambar 3.6. Tampilan *channel* yang telah dibuat ditunjukkan Gambar 3.6



Gambar 3. 7 Tampilan *Channel ThingSpeak*

Setiap *channel* memiliki enam tab menu dengan fungsi yang berbeda-beda. Fungsi menu *Private* dan *Public* adalah untuk mengatur tampilan atau *Graphical User Interface* (GUI). Menu *Private* sebagai akses tampilan untuk pribadi dan *Publics* sebagai akses tampilan untuk umum. Menu *Setting* memiliki beberapa

fungsi diantaranya adalah untuk mengubahb nama *channel*, menambah atau mengurangi *field* yang digunakan, mengganti metadata, mencantumkan *link* ke situs eksternal. Menu *Sharing* berfungsi untuk mengatur siapa saja yang bisa melihat data pada *channel*, misal pribadi, umum atau tertentu. Menu *API Keys* memungkinkan untuk membaca dan menulis data. Karena pada *API Keys* tersedia *Write API Key* dan *Read API Key*. *Write API Key* memungkinkan untuk menulis data dari sebuah mikrokontroler ke *channel*. *Read API Key* digunakan untuk mengijinkan orang lain melihat *channel* pribadi. Menu *Data import/export* berfungsi untuk memasukkan atau mengeluarkan data pada *channel* dengan format CSV.

ThingSpeak menyediakan dua pilihan untuk pengguna yaitu yang membayar atau yang gratis. Di sini penulis menggunakan yang versi gratis. Karena gratis pasti aka nada batasan-batasan penggunaanya, berikut batasan-batasan tersebut adalah pengguna dibatasi untuk pengiriman data tidak lebih dari 3 juta pesan setiap tahun ke layanan ThingSpeak, penggunaan *channel* dibatasi hingga 4 saluran saja, batas interval pembaharuan pesan tetap terbatas pada 15 detik.



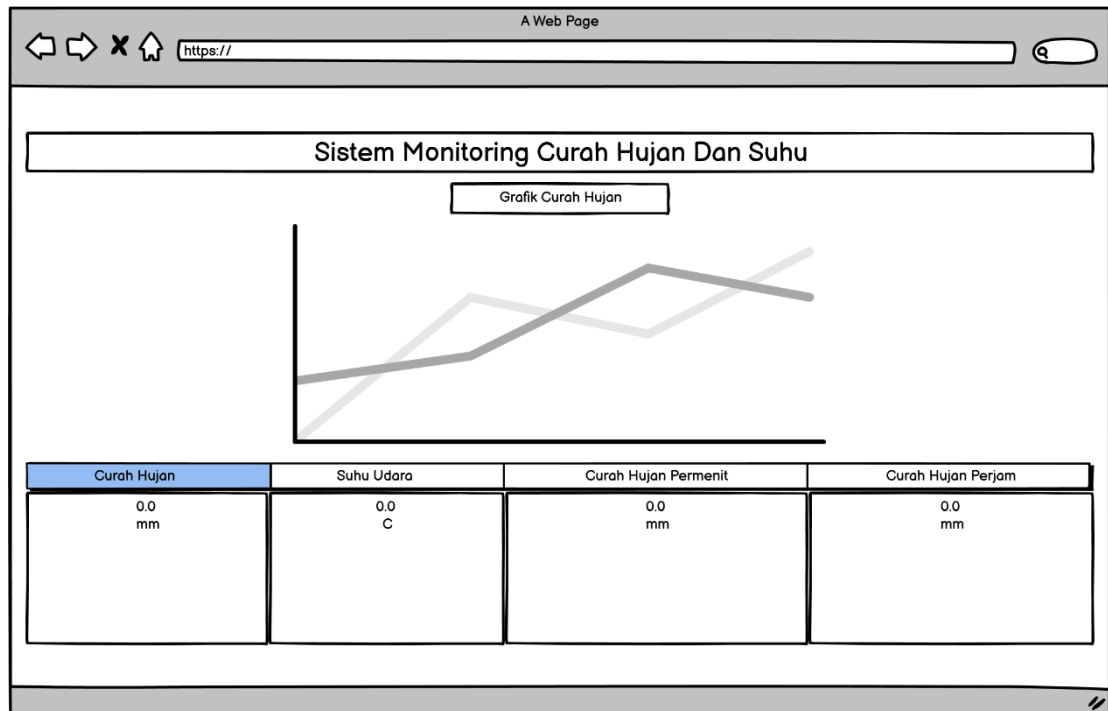
Gambar 3. 8 Tampilan data keluaran

Gambar 3.7 menunjukkan tampilan data keluaran pada sistem IoT yaitu yang berbentuk grafik yang menunjukkan perbandingan antara besarnya data curah hujan dengan waktu pengembalian data.

3.7 Rancangan Sistem Perangkat Lunak

3.7.1 Perancangan Halaman

Tampilan utama pada *website* Sistem Monitoring Curah Hujan ini memiliki 4 data yang ditampilkan yaitu, data curah hujan, data suhu, data curah hujan permenit, data curah hujan perjam, yang bisa dilihat pada Gambar 3.8.



Gambar 3. 9 Tampilan *Website*

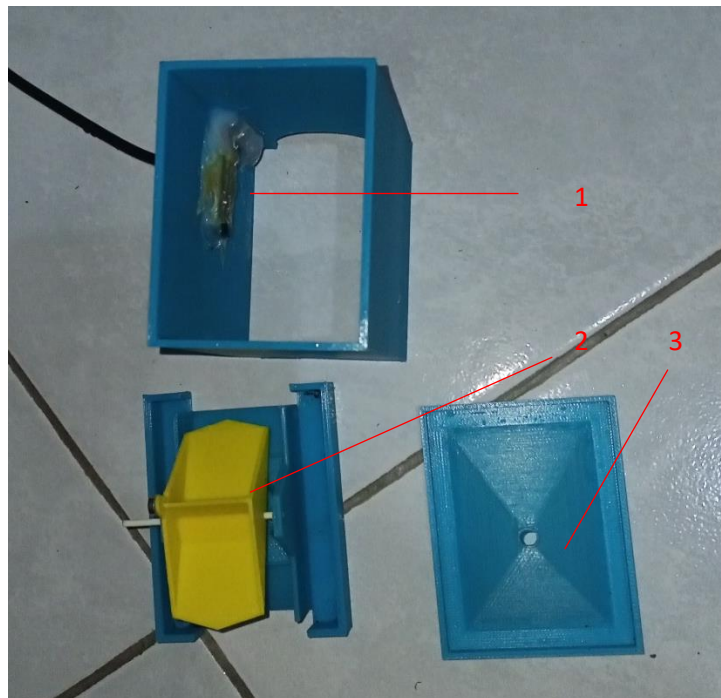
BAB IV

HASIL DAN PEMBAHASAN

4.1 Sensor *Tipping Bucket*

Prinsip kerja sensor curah hujan atau *tipping bucket* menggunakan sensor *hall effect*. Memiliki fungsi yaitu untuk menghitung setiap terjadinya pergerakan pada *bucket*.

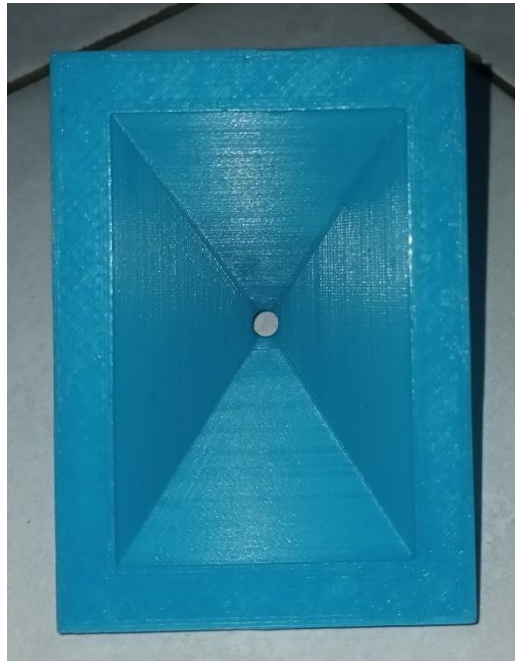
Sensor ditunjukkan pada Gambar 4.1. Sensor *hall effect* digunakan untuk membaca pergerakan *bucket*, pergerakan tersebut bisa dibaca oleh sensor *hall effect*. *Bucket* berfungsi untuk menampung setiap tetesan air hujan yang masuk dari penutup sensor hujan yang berbentuk kerucut terbalik. Bentuk penutup *sensor* ditunjukkan pada Gambar 4.2.



Gambar 4. 1 Sensor Curah Hujan

Keterangan Gambar 4.1 :

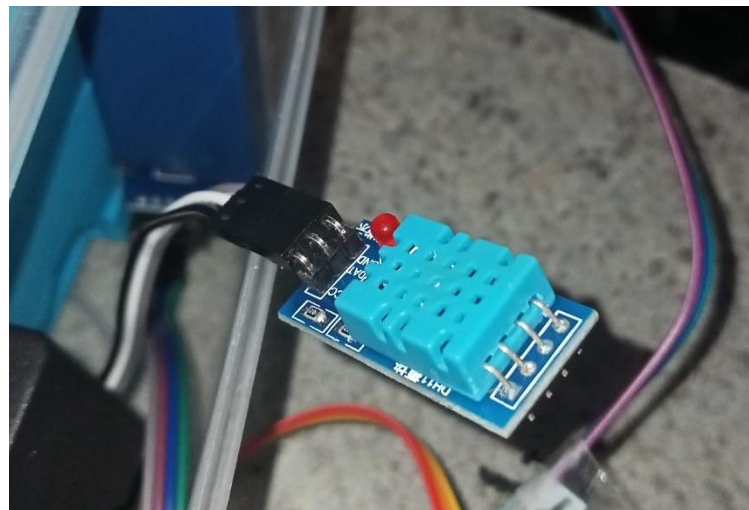
1. Sensor *hall effect*
2. *Bucket*
3. Tutup sensor curah hujan



Gambar 4. 2 Tutup Sensor Curah Hujan

4.2 Sensor Suhu DHT11

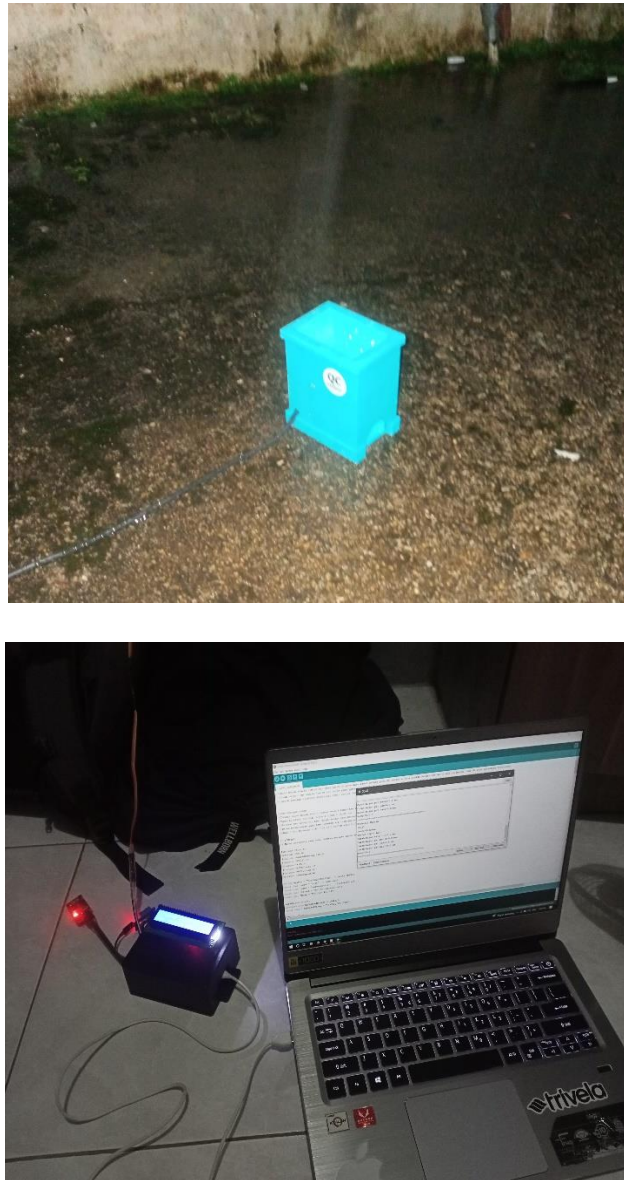
Prinsip kerja sensor suhu atau DHT11 memiliki fungsi yaitu untuk menghitung suhu udara pada suatu tempat tertentu.



Gambar 4. 3 Sensor DHT11

4.3 Hasil Pengujian

Pengujian alat dilakukan di halaman rumah kos. Hal ini dilakukan untuk mengambil data. Proses pengambilan data ditunjukkan pada Gambar 4.4.



Gambar 4. 4 Proses Pengambilan Data

Pada Gambar 4.4, alat yang di atas adalah sensor *tipping bucket* dan yang adalah laptop untuk menjalankan program. Proses pengambilan data adalah dengan pengukur curah hujan tugas akhir ditaruh ditempat terbuka, sehingga sensor hujan atau *tipping bucket* akan mengukur setiap air hujan yang tertampung ke dalam sensor dan menggerakkan *bucket*. Setiap data yang terukur akan disimpan ke *database* dan ke ESP32 sehingga data dapat ditampilkan ke LCD 16x2, ThingSpeak, dan *website*.

Sensor pengukur hujan yang diuji memiliki spesifikasi, untuk detail spesifikasi sensor hujan atau *tipping bucket* ditampilkan pada Tabel 4.1.

Tabel 4. 1 Spesifikasi Tipping Bucket

Keterangan	Sensor Hujan
Tipe	<i>Tipping Bucket</i>
Diameter penerima hujan (tutup sensor)	19,25 cm ²
Nilai pertip	0,7 mm of rain

Dari tabel 4.1 proses pengambilan data dilakukan dengan scenario hujan alami atau hujan sungguhan. Hal ini karena waktu pengambilan data dilakukan pada bulan Agustus 2022.

4.3.1 Database

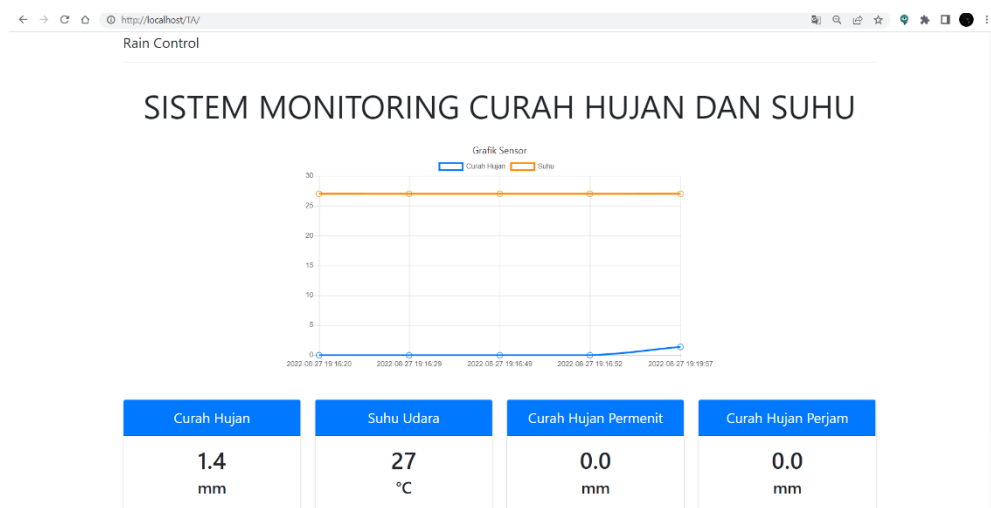
Penyimpanan *database* bertujuan untuk menyimpan seluruh data yang terukur oleh sensor setiap saat dan setiap waktu. Jumlah data yang mampu tersimpan tergantung dengan kapasitas penyimpanan yang tersedia. Pada pengujian ini, dilakukan di dalam *database*. Perekaman atau pengujian alat rancang bangun purwarupa database curah hujan dan suhu berbasis *internet of things* dilakukan pada 26 Agustus 2022. Data dari *database* pada Tabel 4.2 didapatkan berdasarkan hasil pengujian pada alat tugas akhir. Data yang disimpan meliputi urutan data, curah hujan, suhu, dan waktu.

Tabel 4. 2 Data Database Hasil Pengukuran 27 Agustus 2022

Data Ke	Curah Hujan (mm)	Suhu (C)	Waktu
1	1.4	27	2022-08-27 19:19:57
2	1.4	27	2022-08-27 19:21:34
3	1.4	27	2022-08-27 19:21:40
4	1.4	27	2022-08-27 19:21:44

5	1.4	27	2022-08-27 19:21:50
6	1.4	27	2022-08-27 19:21:54
7	1.4	27	2022-08-27 19:22:00
8	1.4	27	2022-08-27 19:22:03
9	1.4	27	2022-08-27 19:22:09
10	1.4	27	2022-08-27 19:22:15
11	1.4	27	2022-08-27 19:23:23

Berdasarkan data hasil pengujian pada tanggal 27 Agustus 2022, besar hujan permenit adalah 1.4 mm. Hal ini bisa dilihat pada Tabel 4.2. Perhitungan pada hujan permenit dimulai pada pukul 19:19:57 WIB sampai dengan 19:23:23 WIB. Jadi data yang terukur sebelum pukul 19:24 WIB.

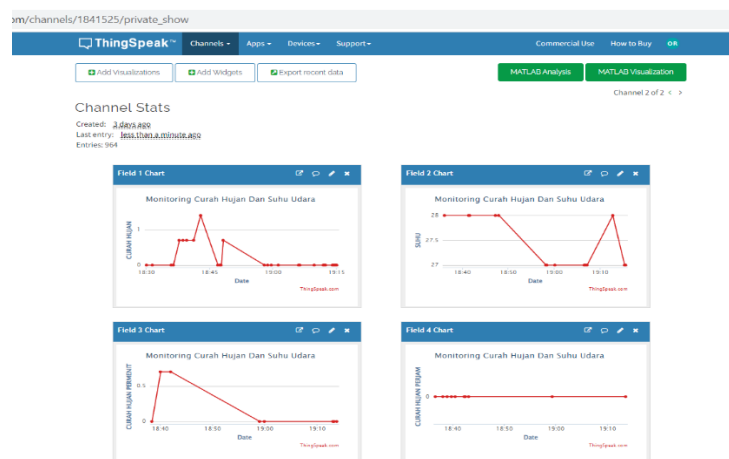


Gambar 4. 5 Grafik Data Curah Hujan Dan Suhu

Gambar 4.5 menunjukkan grafik data curah hujan dan suhu pada alat tugas akhir. Berdasarkan grafik tersebut nilai data hujan permenit sudah tampil pada website secara *realtime*.

4.3.2 ThingSpeak

Bagian ini akan menjelaskan hasil pengambilan data yang tertampil pada ThingSpeak dan juga data yang didapat dari sistem IoT yang dibuat untuk tugas akhir.



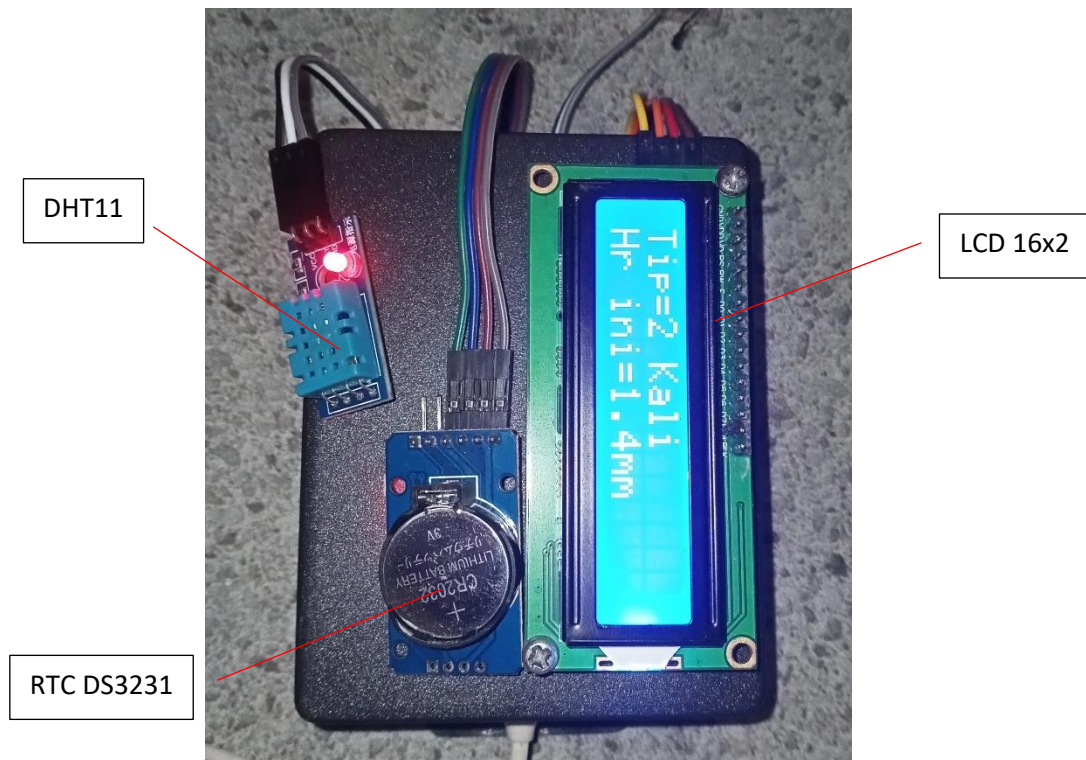
Gambar 4. 6 Monitoring Data Sistem IoT ThingSpeak

Gambar 4.6 adalah hasil *monitoring* data sistem IoT pada ThingSpeak tanggal 26 Agustus 2022, dimulai pada pukul 19:19 WIB sampai dengan 19:23 WIB.

4.4 Bentuk Perangkat Keras

Bagian ini menunjukkan bentuk fisik dari hasil implementasi alat Rancang Bangun Purwarupa Database Curah Hujan dan Suhu Berbasis *Internet of Things* (IoT). Bentuk perangkat keras meliputi bentuk keseluruhan alat, dan sensor *tipping bucket*.

Gambar 4.7 adalah kotak pengaman keseluruhan alat yang berbentuk kotak. Adanya kotak pengaman bertujuan agar alat tidak rusak karena pengujian alat menggunakan air.

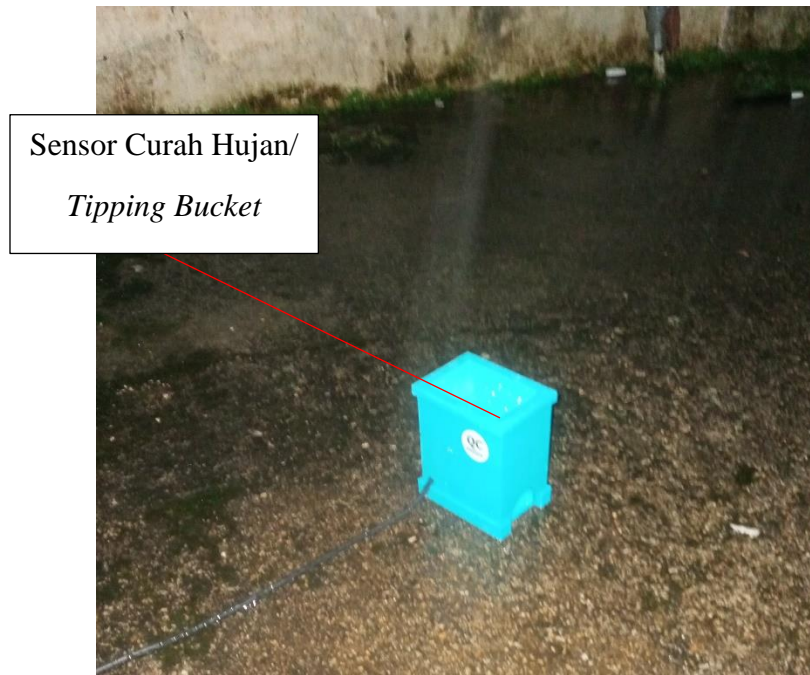


Gambar 4. 7 Implementasi Alat

Pada implementasi, RTC digunakan sebagai *backup* data waktu ketika alat tidak terhubung dengan koneksi *internet* pada laptop. Karena jika alat terhubung dengan koneksi *internet* pada laptop, data waktu akan otomatis mengambil waktu dari laptop/komputer sesuai dengan zona waktu yang telah diset. DHT11 bertujuan untuk mengambil data suhu pada suatu wilayah tertentu. LCD 16x2 berfungsi untuk menampilkan data curah hujan, suhu, dan jam.

4.4.1 Sensor Curah Hujan atau *Tipping Bucket*

Sensor curah hujan menggunakan tipe *tipping bucket* dengan nilai pertipnya adalah 0,7 mm. Bentuk fisik dari sensor ditunjukkan pada Gambar 4.1 dan Gambar 4.2. Sensor hujan atau *tipping bucket* pada alat tugas akhir menggunakan sensor *hall effect* sebagai pembaca setiap gerakan atau setiap perubahan yang terjadi pada *bucket*. Sensor *hall effect* ini memiliki 3 pin, yaitu Vcc, Gnd, dan *output*. Sensor *hall effect* termasuk jenis sensor *digital*, sehingga data masukan bisa langsung dibaca oleh mikrokontroler ESP32. Penjelasan implementasi pin sensor hujan terhadap ESP32 bisa dilihat pada perancangan sistem perangkat keras.



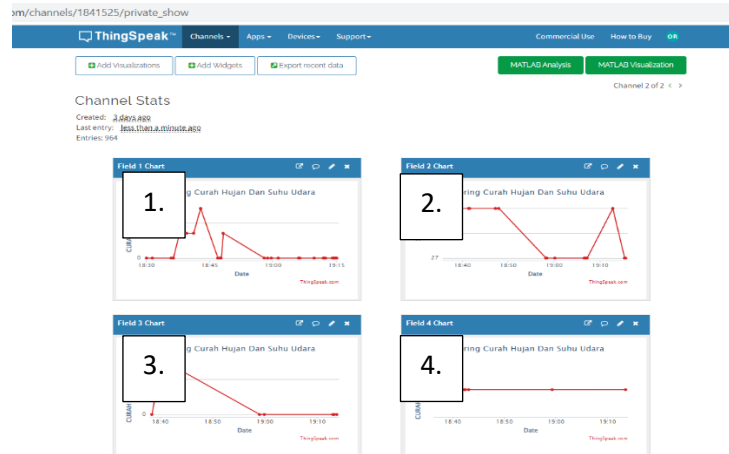
Gambar 4. 8 Implementasi Sensor Hujan atau *Tipping Bucket*

Gambar 4.8 menunjukkan implementasi sensor hujan atau *tipping bucket* ini terletak di halaman rumah kos agar bisa menerima setiap hujan yang turun dan masuke ke sensor *tipping bucket*.

4.5 Implementasi Sistem *Monitoring*

4.5.1 Sistem *Internet of Things*

Bagian ini membahas tentang implementasi *monitoring* data, menggunakan sistem *internet of things* (IoT). Sistem IoT pada alat tugas akhir menggunakan *ThingSpeak* sebagai *cloud server*. Prinsip kerja *monitoring* data berbasis IoT ini adalah menampilkan data hujan yang dikirim oleh ESP32 menggunakan koneksi *internet* ke *ThingSpeak*. Penyajian data disediakan oleh *ThingSpeak* menggunakan tampilan grafik seperti yang ditunjukkan pada Gambar 4.8.



Gambar 4. 9 Tampilan *Channel ThingSpeak*

Keterangan Gambar 4.9:

1. Curah Hujan
2. Suhu Udara
3. Curah Hujan Permenit
4. Curah Hujan Perjam

Implementasi *monitoring* pada *ThibgSpeak* menggunakan 4 *field*. *Field 1* digunakan untuk menampilkan data hujan hari ini, sedangkan *field 2* untuk menampilkan data suhu. Jadi ketika ketika data *field 1* dan *field 2* akan tertampil 1 kali dalam sehari. *Field 3* adalah curah hujan permenit, yang digunakan sebagai data curah hujan permenit, dan *field 4* adalah curah hujan perjam yang digunakan sebagai data curah hujan yang ditampilkan dalam 1 jam sekali.

Setelah melakukan percobaan pengambilan data sekaligus melakukan tes ketahanan alat, didapatkan pada tampilan data *channel ThingSpeak* hanya bisa menampilkan data dalam setiap menit saja. Misal pengambilan data dilakukan antara jam 18.30 WIB sampai dengan 19.15 WIB 26 Agustus 2022.

4.5.2 LCD 16x2 dan I2C



Gambar 4. 10 Implementasi LCD 16x2 dan ESP32

Gambar 4.10 adalah hasil implementasi pada penampil LCD 16x2. Nilai yang ditampilkan pada bagian atas adalah nilai tip yang sudah terjadi dan yang bawah adalah data curah hujan yang sudah terukur.

4.6 Sistem Perangkat Lunak

Bagian ini akan menjelaskan implementasi perangkat lunak pada alat Rancang Bangun Purwarupa Database Curah Hujan dan Suhu Berbasis *Internet of Things* (IoT) termasuk program – program yang digunakan. Pemrograman yang digunakan adalah pemrograman C++ dan PHP.

4.6.1 Program Pembaca Sensor Curah Hujan, Suhu, LCD 16x2, dan RTC

Sebelum melakukan pembuatan sebuah program, terlebih dahulu harus memasukkan modul-modul yang digunakan untuk menjalankan alat. Modul – modul dan *library* yang digunakan untuk menjalankan alat dapat dilihat pada program berikut.

```

1. #include <Wire.h>
2. #include <DHT.h>
3. #include <LiquidCrystal_I2C.h>
4. #include "RTCLib.h"
5. #include <WiFi.h>
6. #include "HTTPClient.h"
7. #include <WiFiClient.h>
8. #include <ThingSpeak.h>

```

Pemrograman untuk pengoprasian ESP32 menggunakan Arduino IDE. Program Arduino IDE untuk menerima data dari sensor hujan atau *tipping bucket*, DHT11, RTC DS3231, dan LCD 16x2. Selain itu pemrograman Arduino IDE juga bisa untuk mengirimkan data ke *ThingSpeak* dan *SQL database*. Sebelum memulai sebuah program, diperlukan deklarasi *library* yang digunakan, yaitu *library* yang dibutuhkan untuk pengoprasian, yang dilihat pada program di atas.

```

1. if (detik.equals("0")) // Curah hujan per menit, jam, hari dihitung
   ketika detik 0
2. {
3.     curah_hujan_per_menit = temp_curah_hujan_per_menit;
4.     temp_curah_hujan_per_jam += curah_hujan_per_menit; // Curah hujan
   per jam dihitung dari penjumlahan curah hujan per menit namun disimpan
   dulu dalam variabel temp
5.     if (menit.equals("0"))// Curah hujan per jam baru dihitung ketika
   menit 0
6.     {
7.         curah_hujan_per_jam = temp_curah_hujan_per_jam;
8.         temp_curah_hujan_per_hari += curah_hujan_per_jam; //// Curah hujan
   per hari dihitung dari penjumlahan curah hujan per jam namun disimpan dulu
   dalam variabel temp
9.         temp_curah_hujan_per_jam = 0.00; // Reset temp
   curah hujan per jam
10.    }
11.    if (menit.equals("0") && jam.equals("0"))// Curah hujan per jam baru
   dihitung ketika menit 0 dan jam 0 (Tengah malam)
12.    {
13.        curah_hujan_per_hari = temp_curah_hujan_per_hari;
14.        temp_curah_hujan_per_hari = 0.00; // Reset temp
   curah hujan per hari
15.        curah_hujan_hari_ini = 0.00; // Reset curah
   hujan hari ini
16.        jumlah_tip = 0; // Jumlah tip di
   reset setiap 24 jam sekali (Tengah malam)
17.    }
18.    temp_curah_hujan_per_menit = 0.00;
19.    curah_hujan = 0.00;
20.    delay(1000);
21. }

```

Pada program di atas menunjukkan kalibrasi program untuk menghitung curah hujan permenit, perjam, dan perhari.

```

1. void setup()
2. {
3.   Serial.begin(9600);
4.   // Inisialisasi LCD
5.   lcd.begin();
6.
7.   // Menyalakan lampu latar LCD
8.   lcd.backlight();
9.   //Clear LCD
10.  lcd.clear();
11.
12.  pinMode(pin_interrupt, INPUT);
13.  attachInterrupt(digitalPinToInterrupt(pin_interrupt),
    hitung_curah_hujan, FALLING); // Akan menghitung tip jika pin berlogika
    dari HIGH ke LOW
14.  // Inisialisasi RTC
15.  if (!rtc.begin())
16.  {
17.    Serial.println("Couldn't find RTC");
18.    while (1)
19.      ;
20.  }
21.  // Inisialisasi DHT11
22.  dht.begin();
23.  delay(10);
24.  WiFi.begin(ssid, pass);
25.  ThingSpeak.begin(client);

```

Pada program diatas menunjukkan pengaturan beberapa perangkat yang digunakan, seperti membuka *port* serial, menentukan *baud rate* yang digunakan untuk komunikasi antara ESP32 dengan pengaturan DHT11, LCD 16x2, dan RTC.

```

1. void loop()
2. {
3.   suhu = dht.readTemperature();
4.   if (isnan(suhu))
5.   {
6.     Serial.println("Tidak Terdeteksi");
7.     return;
8.   }

```

Program diatas menunjukkan pembaca data suhu yang telah dikirimkan oleh sensor DHT11. Data yang diterima oleh ESP32 yaitu data suhu dari sensor DHT11.

```

1. void bacaRTC()
2. {
3.   DateTime now = rtc.now(); // Ambil data waktu dari DS3231
4.   tanggal = String(now.day(), DEC);
5.   bulan = String(now.month(), DEC);
6.   itahun = now.year() - 2000;
7.   tahun = String(itahun);
8.   jam = String(now.hour(), DEC);
9.   menit = String(now.minute(), DEC);
10.  detik = String(now.second(), DEC);
11. }

```

Program sebelumnya menunjukkan pembaca data waktu yang telah dikirimkan oleh sensor RTC DS3231.

```
1. String apikey = "78STM2QYZTC153ME"; // write ApiKey
2. const char *ssid = "oscr"; // nama wifi
3. const char *pass = "satusampai10"; //password wifi
4. const char *server = "api.thingspeak.com";
5. const char *host = "192.168.43.204";
6.
7. WiFiClient client;
8. unsigned long myChannelNumber = 1841525;
9. const char * myWriteAPIKey = "78STM2QYZTC153ME";
```

Program diatas menunjukkan program untuk memasukkan *username* dan *password* dari suatu wifi. Serta program untuk memasukkan *APIKey*, *server* dari *ThingSpeak*, dan *NumberChannel* dari *ThingSpeak*.

```
1. if (tampil == 0)
2. {
3.     lcd.clear();
4.     lcd.setCursor(0, 0); // Menampilkan pada kolom 0, baris 1
5.     lcd.print(konversi(tanggal) + "/" + konversi(bulan) + "/" +
konversi(tahun) + " " + konversi(jam) + ":" + konversi(menit));
6.     lcd.setCursor(0, 1); // Menampilkan pada kolom 0, baris 0
7.     lcd.print(cuaca); // Print cuaca hari ini (Ini bukan ramalan
cuaca tapi membaca cuaca yang sudah terjadi/ sedang terjadi hari ini)
8.     previousMillis_1 = millis();
9.     tampil = 1;
10.    goto end;
11. }
12. if (tampil == 1)
13. {
14.     lcd.clear();
15.     lcd.setCursor(0, 0); // Menampilkan pada kolom 0, baris 1
16.     lcd.print("Tip=");
17.     lcd.print(jumlah_tip);
18.     lcd.print(" Kali");
19.     lcd.setCursor(0, 1); // Menampilkan pada kolom 0, baris 1
20.     lcd.print("Hr ini=");
21.     lcd.print(curah_hujan_hari_ini, 1); // 1 adalah 1 angka dibelakang
koma
22.     lcd.print("mm");
23.     previousMillis_1 = millis();
24.     tampil = 2;
25.     goto end;
26. }
27. if (tampil == 2)
28. {
29.     lcd.clear();
30.     lcd.setCursor(0, 0);
31.     lcd.print("/Mnt=");
32.     lcd.print(curah_hujan_per_menit, 1);
33.     lcd.print("mm");
34.     lcd.setCursor(0, 1);
35.     lcd.print("/Jam=");
36.     lcd.print(curah_hujan_per_jam, 1);
37.     lcd.print("mm");
```

```

38.     previousMillis_1 = millis();
39.     tampil = 3;
40.     goto end;
41. }
42.
43. if (tampil == 3)
44. {
45.     lcd.clear();
46.     lcd.setCursor(0, 0);
47.     lcd.print("/Hari=");
48.     lcd.print(curah_hujan_per_hari, 1);
49.     lcd.print("mm");
50.     lcd.setCursor(0, 1);
51.     lcd.print("Suhu=");
52.     lcd.print(suhu, 1);
53.     lcd.print(" C ");
54.     previousMillis_1 = millis();
55.     tampil = 0;
56. }
57. end:

```

Pada program diatas untuk menampilkan waktu, curah hujan, jumlah tip, suhu, curah hujan permenit, curah hujan perjam, dan curah hujan perhari. Sistem penampil data yang hanya tampil selama 2 detik dalam 3 frame.

```

1. ThingSpeak.writeField(myChannelNumber, 1, curah_hujan_hari_ini,
   myWriteAPIKey);
2. ThingSpeak.writeField(myChannelNumber, 2, suhu, myWriteAPIKey);
3. ThingSpeak.writeField(myChannelNumber, 3, curah_hujan_per_menit,
   myWriteAPIKey);
4. ThingSpeak.writeField(myChannelNumber, 4, curah_hujan_per_jam,
   myWriteAPIKey);

```

Program diatas adalah program untuk mengirim data curah hujan, suhu, curah hujan permenit, dan curah hujan perjam. Dalam data ini akan dikirim dengan *delay* 15 detik sekali di penampil *thingspeak* nya.

```

1. String Link;
2. HTTPClient http;
3.
4. Link = "http://" + String(host) + "/TA/kirimdata.php?curah_hujan=" +
   String(curah_hujan_hari_ini) + "&suhu=" + String(suhu) +
   "&curah_hujan_permenit=" + String(curah_hujan_per_menit) +
   "&curah_hujan_perjam=" + String(curah_hujan_per_jam);
5.
6. // eksekusi
7. http.begin(Link);
8. http.GET();
9.
10. // baca respon setelah berhasil kirim nilai sensor
11. String respon = http.getString();
12. Serial.println(respon);
13. http.end();

```

Program sebelumnya adalah program untuk mengirim data curah hujan, suhu, curah hujan permenit, dan curah hujan perjam. Dalam data ini akan dikirim dengan *delay* 1 detik sekali ke *database*.

4.6.2 Program Pembaca Data Ke Website

```

1. <!doctype html>
2. <html lang="en">
3.   <head>
4.
5.     <meta charset="utf-8">
6.     <meta name="viewport" content="width=device-width, initial-scale=1">
7.     <meta name="description" content="">
8.     <meta name="author" content="Mark Otto, Jacob Thornton, and
Bootstrap contributors">
9.     <meta name="generator" content="Hugo 0.84.0">
10.    <link                                rel="canonical"
href="https://getbootstrap.com/docs/5.0/examples/pricing/">
11.    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"
crossorigin="anonymous">
12.    <!-- Custom styles for this template -->
13.    <link href="pricing.css" rel="stylesheet">
14.    <!--                                rel="stylesheet"          type="text/css"
href="assets/css/bootstrap.min.css"> -->
15.    <script type="text/javascript" src="assets/js/jquery-
3.4.0.min.js"></script>
16.    <script type="text/javascript" src="assets/js/mdb.min.js"></script>
17.    <script type="text/javascript" src="jquery-latest.js"></script>
18.
19.    <title>RainControl</title>
20.    <script type="text/javascript" src="jquery/jquery.min.js"></script>
21.    <!-- load/realtime -->
22.
23.    <script type="text/javascript">
24.      $(document).ready( function() {
25.        setInterval( function() {
26.          $("#cekhujan").load("cekhujan.php");
27.          $("#ceksuhu").load("ceksuhu.php");
28.          $("#cekhujanmenit").load("cekhujanmenit.php");
29.          $("#cekhujanjam").load("cekhujanjam.php");
30.        }, 1000 );
31.
32.      } );
33.    </script>
34.
35.  </head>
36.  <body>
37.
38.  <div class="container">
39.    <header>
40.      <div class="d-flex flex-column flex-md-row align-items-center pb-3
mb-4 border-bottom">
41.        <a href="/" class="d-flex align-items-center text-dark text-
decoration-none">
42.          <span class="fs-4">Rain Control</span>
43.        </a>
44.      </div>

```

```

45.
46.     <div class="pricing-header p-3 pb-md-4 mx-auto text-center">
47.         <h1 class="display-4 fw-normal" style="text-align: bold;">SISTEM
MONITORING CURAH HUJAN DAN SUHU </h1>
48.     </div>
49. </header>
50.
51. <main>
52.
53.     <script type="text/javascript">
54.         var refreshid = setInterval(function(){
55.             $('#responsecontainer').load('data.php');
56.         }, 1000);
57.     </script>
58. </head>
59. <body>
60.     <!-- div untuk grafik -->
61.     <div class="container">
62.         <div class="container" id="responsecontainer" style="width: 60%"></div>
63.     </div>
64.     <script type="text/javascript">
65.         var refreshid = setInterval(function(){
66.             $('#responsecontainer').load('data.php');
67.         }, 1000);
68.     </script>
69.
70.     <!-- div untuk grafik -->
71.     <div class="container">
72.         <div class="container" id="responsecontainer" style="width: 50%"></div>
73.     </div>
74.     <script type="text/javascript">
75.         var refreshid = setInterval(function(){
76.             $('#responsecontainer').load('data.php');
77.         }, 1000);
78.     </script>
79.     <br></br>
80.
81.
82.     <div class="row row-cols-1 row-cols-md-4 mb-4 text-center">
83.     <!-- card curah hujan -->
84.         <div class="col">
85.             <div class="card mb-4 rounded-3 shadow-sm">
86.                 <div class="card-header py-3" style="background-color:
#0078ff">
87.                     <h4 class="my-0 fw-normal" style="color: white">Curah
Hujan</h4>
88.                 </div>
89.                 <div class="card-body">
90.                     <h1><span id="cekhujan">0</span></h1>
91.                     <h3 class="card-title pricing-card-title">mm</h3>
92.                 </div>
93.             </div>
94.         </div>
95.     <!-- card suhu udara -->
96.         <div class="col">
97.             <div class="card mb-4 rounded-3 shadow-sm">
98.                 <div class="card-header py-3" style="background-color:
#0078ff">
99.                     <h4 class="my-0 fw-normal" style="color: white">Suhu
Udara</h4>
100.                 </div>
101.                 <div class="card-body">
102.                     <h1><span id="ceksuhu">0</span></h1>

```

```

103.         <h3 class="card-title pricing-card-title">°C</h3>
104.     </div>
105. </div>
106. </div>
107. <!-- card curah hujan permenit -->
108. <div class="col">
109.     <div class="card mb-4 rounded-3 shadow-sm">
110.         <div class="card-header py-3" style="background-color:
111.             #0078ff">
112.             <h4 class="my-0 fw-normal" style="color: white">Curah
113.             Hujan Permenit</h4>
114.         </div>
115.         <div class="card-body">
116.             <h1><span id="cekhujanmenit">0</span></h1>
117.             <h3 class="card-title pricing-card-title">mm</h3>
118.         </div>
119.     </div>
120. </div>
121. <!-- card curah hujan perjam -->
122. <div class="col">
123.     <div class="card mb-4 rounded-3 shadow-sm">
124.         <div class="card-header py-3" style="background-color:
125.             #0078ff">
126.             <h4 class="my-0 fw-normal" style="color: white">Curah
127.             Hujan Perjam</h4>
128.         </div>
129.         <div class="card-body">
130.             <h1><span id="cekhujanjam">0</span></h1>
131.             <h3 class="card-title pricing-card-title">mm</h3>
132.         </div>
133.     </div>
134. </div>
135. </main>
136. <!-- footer -->
137. <footer class="pt-4 my-md-5 pt-md-5 border-top">
138.     <div class="row">
139.         <div class="col-12 col-md">
140.             <!--  -->
143.             <small class="d-block mb-3 text-muted">&copy; 2022</small>
144.         </div>
145.     </footer>
146. </div>
147.
148.
149.
150. </body>
151. </html>

```

Pada program diatas adalah program untuk menampilkan data curah hujan, data suhu, data curah hujan permenit, dan data curah hujan perjam.

```

1. <?php
2.     // koneksi ke database

```



```

3.     $konek = mysqli_connect("localhost", "root", "", "cobasensor");
4.
5.     // baca data dari tabel tb_sensor_curah_hujan
6.     $sql = mysqli_query($konek, "select * from tb_sensor_curah_hujan
    order by ID desc");
7.
8.     // baca data paling terakhir
9.     $data = mysqli_fetch_array($sql);
10.    $curah_hujan = $data['curah_hujan'];
11.
12.    // apabila nilai curah hujan belum ada maka anggap suhunya sama
    dengan 0
13.    if( $curah_hujan == "") $curah_hujan = 0;
14.
15.    // cetak nilai curah hujan
16.    echo $curah_hujan ;
17.
18.    ?>

```

Pada program diatas adalah program untuk mengambil data curah hujan yang ada di *database*.

```

1. <?php
2.     // koneksi ke database
3.     $konek = mysqli_connect("localhost", "root", "", "cobasensor");
4.
5.     // baca data dari tabel tb_sensor_curah_hujan
6.     $sql = mysqli_query($konek, "select * from tb_sensor_curah_hujan
    order by ID desc");
7.
8.     // baca data paling terakhir
9.     $data = mysqli_fetch_array($sql);
10.    $suhu = $data['suhu'];
11.
12.    // apabila nilai suhu belum ada maka anggap suhunya sama dengan 0
13.    if( $suhu == "") $suhu = 0;
14.
15.    // cetak nilai suhu
16.    echo $suhu ;
17.
18.    ?>

```

Program diatas adalah program untuk mengambil data suhu yang ada di *database*.

```

1. <?php
2.     // koneksi ke database
3.     $konek = mysqli_connect("localhost", "root", "", "cobasensor");
4.
5.     // baca data dari tabel tb_sensor_curah_hujan
6.     $sql = mysqli_query($konek, "select * from tb_sensor_curah_hujan
    order by ID desc");
7.
8.     // baca data paling terakhir
9.     $data = mysqli_fetch_array($sql);
10.    $curah_hujan_permenit = $data['curah_hujan_permenit'];
11.

```

```

12. // apabila nilai curah hujan permenit belum ada maka anggap suhunya
    sama dengam 0
13. if( $curah_hujan_permenit == "" ) $curah_hujan_permenit = 0;
14.
15. // cetak nilai curah hujan permenit
16. echo $curah_hujan_permenit ;
17.
18. ?>

```

Program diatas adalah program untuk mengambil data curah hujan permenit yang ada di *database*.

```

1. <?php
2. // koneksi ke database
3. $koneksi = mysqli_connect("localhost", "root", "", "cobasensor");
4.
5. // baca data dari tabel tb_sensor_curah_hujan
6. $sql = mysqli_query($koneksi, "select * from tb_sensor_curah_hujan
    order by ID desc");
7.
8. // baca data paling terakhir
9. $data = mysqli_fetch_array($sql);
10. $curah_hujan_perjam = $data['curah_hujan_perjam'];
11.
12. // apabila nilai curah hujan perjam belum ada maka anggap suhunya
    sama dengam 0
13. if( $curah_hujan_perjam == "" ) $curah_hujan_perjam = 0;
14.
15. // cetak nilai curah hujan perjam
16. echo $curah_hujan_perjam ;
17.
18. ?>

```

Program diatas adalah program untuk mengambil data curah hujan perjam yang ada di *database*.

```

1. <?php
2. // koneksi ke database
3. $koneksi = mysqli_connect("localhost", "root", "", "cobasensor");
4.
5. // baca data yang dikirim ke esp32
6. $curah_hujan = $_GET['curah_hujan'];
7. $suhu = $_GET['suhu'];
8. $curah_hujan_permenit = $_GET['curah_hujan_permenit'];
9. $curah_hujan_perjam = $_GET['curah_hujan_perjam'];
10.
11. // simpan ke tb_sensor_curah_hujan
12.
13. // auto increment = 1
14. mysqli_query($koneksi, "ALTER TABLE tb_sensor_curah_hujan
    AUTO_INCREMENT=1");
15.
16. // simpan data sensor
17. $simpan = mysqli_query($koneksi, "insert into
    tb_sensor_curah_hujan(curah_hujan, suhu, curah_hujan_permenit,
    curah_hujan_perjam)values('$curah_hujan', '$suhu',
    '$curah_hujan_permenit', '$curah_hujan_perjam')");
18. // uji simpan untuk memberikan respon

```

```

19.     if($simpan)
20.         echo "Berhasil Dikirim";
21.     else
22.         echo "Gagal";
23.
24.
25. ?>

```

Program diatas adalah program untuk mengambil data curah hujan, suhu, curah hujan permenit, dan curah hujan perjam dari Arduino IDE yang akan dikirimkan ke *database*.

4.7 Analisa

Berdasarkan pengujian yang telah dilaksanakan maka dapat dibuat analisa yang berdasarkan data-data hasil pengujian tersebut:

- a. Berdasarkan pengujian alat yang telah dilakukan secara keseluruhan menunjukkan bahwa alat dapat bekerja sesuai dengan prinsip kerja alat yang telah dirancang. Menggunakan sistem *control* mikrokontroller ESP32.
- b. Dari pengujian rangkaian sensor curah hujan atau *tipping bucket* adalah kurang lebih 1,4 mm
- c. Sensor suhu atau DHT11 digunakan untuk membaca suhu ruangan yang ada ditempat tersebut.
- d. Dari ThingSpeak kita akan membaca sistem data curah hujan, data suhu, data curah hujan permenit, dan data curah hujan perjam.
- e. Dari *website* kita akan membaca sistem data curah hujan, data suhu, data curah hujan permenit, dan data curah hujan perjam.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil analisa data dan pengujian alat Rancang Bangun Database Curah Hujan dan Suhu Berbasis *Intener of Things* (IoT) diperoleh kesimpulan sebagai berikut:

1. Pada sistem ini menggunakan mikrokontroller ESP32 untuk mengatur data curah hujan dan suhu.
2. Alat telah mampu melakukan fungsi-fungsi utama sesuai tujuan pembuatan alat dengan baik, seperti membaca data dari sensor, menampilkan data ke LCD 16x2, dan menampilkan data ke ThingSpeak secara *real time*.

5.2. Saran

Pembuatan alat ini masih jauh dari kata sempurna, untuk pengembangan alat pada tahap selanjutnya terdapat beberapa saran sebagai berikut :

1. Alat ini adalah alat ukur, sehingga alat diharapkan selalu berfungsi setiap adanya *variable* yang di ukur. Oleh karena itu, untuk pembuatan selanjutnya disarankan melakukan pengaturan atau program yang mendalam agar alat bisa menjalankan suatu program dengan otomatis ketika alat hidup yang tidak tergantung pada jaringan listrik PLN.
2. Mengeksplor lebih banyak *platform* IoT agar mendapatkan suatu *platform* yang lebih mudah diakses dan lebih menarik dalam penyajian data.
3. Pada alat ini adalah ukur, sehingga harus sesuai dengan data nyata yang ada di lapangan. Oleh karena itu, untuk pembuatan selanjutnya disarankan untuk melakukan kalibrasi yang sesuai dengan data yang ada di BMKG dan juga mencari tahu tentang API ThingSpeak.

DAFTAR PUSTAKA

- [1] I. A. Nurdyanto, *Monitoring Data Curah Hujan Berbasis IoT*, 2019.
- [2] R. S. Putra, *Rancang Bangun Sistem Monitoring Intensitas Curah Hujan Berbasis IoT*, 2017.
- [3] M. A. D. Kusuma, *Prototype Buka Tutup Atap Otomatis Untuk Menjemur Padi Dan Solar Panel Berbasis Arduino Uno*, 2022.
- [4] “World Meteorological Organization (WMO),” 2022. [Online]. Available: <https://www.bmkg.go.id/berita/?p=panduan-wmo-1203-tentang-perhitungan-normal-iklim-apa-yang-berubah&lang=ID&tag=artikel>.
- [5] V. F. Fu, Binus University, [Online]. Available: student-activity.binus.ac.id/himtek/2022/07/27/esp32/. [Accessed 1 September 2022].
- [6] Nyebarilmu.com, [Online]. Available: www.nyebarilmu.com/cara-mengakses-modul-display-lcd-16x2/. [Accessed 4 September 2022].
- [7] “Cara mengakses sensor DHT11 menggunakan Arduino,” Nyebarilmu.com, [Online]. Available: <https://www.nyebarilmu.com/cara-mengakses-sensor-dht11/>. [Accessed 3 September 2022].
- [8] “MQTT adalah protokol terkemuka untuk menghubungkan perangkat IoT,” [Online]. Available: <https://www.emqx.com/id/blog/mqtt-is-the-leading-protocol-for-connecting-iot-devices>. [Accessed 15 Agustus 2022].
- [9] “Pengertian Laravel,” [Online]. Available: <https://www.dewaweb.com/blog/apa-itu-laravel/>. [Accessed 3 September 2022].
- [10] “Apa itu Mysqli,” [Online]. Available: <https://www.niagahoster.co.id/blog/mysql-adalah/>. [Accessed 2 September 2022].
- [11] IQBAL, PURWARUPA SISTEM KONTROL MULTI SOURCE BERBASIS ENERGI SURYA DAN ANGIN UNTUK PEMANFAATAN KAPAL NRLAYAN, IQBAL, 2021.
- [12] ThingSpeak, “ThingSpeak,” 2022. [Online]. Available: <https://thingspeak.com/>.
- [13] A. IDE, “Arduino,” [Online]. Available: <https://arduino.c>. [Accessed 2022].

- [14] BMKG, “BMKG,” [Online]. Available: <https://www.bmkg.go.id/>. [Accessed 2022].
- [15] “Thingspeak.com, Web Server IoT Gratis Buat Kamu Yang Suka Gratisan,” [Online]. Available: <https://www.anakteknik.co.id/krysnayudhamaulana/articles/thingspeakcom-web-server-iot-gratis-buat-kamu-yang-suka-gratisan>. [Accessed 30 September 2022].
- [16] “Apa itu PHP?,” [Online]. Available: <https://www.hostinger.co.id/tutorial/apa-itu-php/>. [Accessed 2 September 2022].
- [17] “Pengertian Xampp,” [Online]. Available: <https://www.ekrut.com/media/xampp-adalah>. [Accessed 3 Agustus 2022].
- [18] K. Azzizah, “Pengertian HTML Lengkap dengan Fungsi dan Sejarah Kemunculannya,” [Online]. Available: <https://www.merdeka.com/trending/pengertian-html-lengkap-dengan-fungsi-dan-sejarah-kemunculannya-kln.html>. [Accessed 3 Agustus 2022].

LAMPIRAN

Lampiran 1

Dokumentasi



Dokumentasi 1



Dokumentasi 2



Dokumentasi 3



Dokumentasi 4



Dokumentasi 5



Dokumentasi 6

Lampiran 2

Program Arduino IDE

```

1. //Curah hujan adalah jumlah air yang jatuh di permukaan tanah selama
   periode tertentu yang diukur dengan satuan tinggi milimeter (mm) di atas
   permukaan horizontal.
2. //Curah hujan 1 mm adalah jumlah air hujan yang jatuh di permukaan per
   satuan luas (m2) dengan volume sebanyak 1 liter tanpa ada yang menguap,
   meresap atau mengalir.
3. // Lebih lengkap silahkan dipelajari lebih lanjut disini
   https://www.climate4life.info/2015/12/hujan-1-milimeter-yang-jatuh-di-jakarta.html
4.
5.
6. //Perhitungan rumus
7. //Tinggi curah hujan (cm) = volume yang dikumpulkan (mL) / area
   pengumpulan (cm2)
8. //Luas kolektor (Corong) 5,5cm x 3,5cm = 19,25 cm2
9. //Koleksi per ujung tip kami dapat dengan cara menuangkan 100ml air ke
   kolektor kemudian menghitung berapa kali air terbuang dari tip,
10. //Dalam perhitungan yang kami lakukan air terbuang sebanyak 70 kali.
    100ml / 70= 1.42mL per tip.
11. //Jadi 1 tip bernilai 1.42 / 19.25 = 0,07cm atau 0.70 mm curah hujan.
12.
13. // PENTING
14. // Nilai kalibrasi yang kami lakukan berlaku untuk semua sensor curah
   hujan yang kami jual tentu Anda dapat melakukan kalibrasi ulang sendiri
   jika dibutuhkan.
15.
16. #include <Wire.h>
17. #include <DHT.h>
18. #include <LiquidCrystal_I2C.h>
19. #include "RTClib.h"
20. #include <WiFi.h>
21. #include "HTTPClient.h"
22. #include <WiFiClient.h>
23. #include <ThingSpeak.h>
24.
25. String apikey = "78STM2QYZTC153ME"; // write ApiKey
26. const char *ssid = "oscr"; // nama wifi
27. const char *pass = "satusampai10"; //password wifi
28. const char *server = "api.thingspeak.com";
29. const char *host = "192.168.43.204";
30.
31. WiFiClient client;
32. unsigned long myChannelNumber = 1841525;
33. const char * myWriteAPIKey = "78STM2QYZTC153ME";
34.
35. RTC_DS1307 rtc;
36.
37. // Gunakan pin D5 pada NodeMCU, GPIO14 pada ESP32, Tegangan 3,3V
   Kemudian upload code ini
38. const int pin_interrupt = 14; // Menggunakan pin interrupt
   https://www.arduino.cc/reference/en/language/functions/external-interrupts/attachinterrupt/
39. long int jumlah_tip = 0;
40. long int temp_jumlah_tip = 0;
41. float curah_hujan = 0.00;
42. float curah_hujan_per_menit = 0.00;
43. float curah_hujan_per_jam = 0.00;
44. float curah_hujan_per_hari = 0.00;

```

```

45. float curah_hujan_hari_ini = 0.00;
46. float temp_curah_hujan_per_menit = 0.00;
47. float temp_curah_hujan_per_jam = 0.00;
48. float temp_curah_hujan_per_hari = 0.00;
49. float milimeter_per_tip = 0.70;
50. String cuaca = "Berawan ";
51.
52. volatile boolean flag = false;
53.
54. // Set alamat I2C LCD 0x27
55. LiquidCrystal_I2C lcd(0x27, 16, 2);
56.
57. // Inisialisasi struktur waktu
58. String tanggal, bulan, tahun, jam, menit, detik;
59.
60. unsigned long previousMillis_1 = 0;
61. unsigned long interval_1 = 2000; // 2 detik , untuk tampil bergantian lcd
62. unsigned long currentMillis_1;
63. int tampil = 0;
64. int itahun;
65.
66. #define DHTPIN 4
67. DHT dht(DHTPIN, DHT11);
68. float suhu;
69.
70. void ICACHE_RAM_ATTR hitung_curah_hujan()
71. {
72.   flag = true;
73. }
74.
75. void setup()
76. {
77.   Serial.begin(9600);
78.   // Inisialisasi LCD
79.   lcd.begin();
80.
81.   // Menyalakan lampu latar LCD
82.   lcd.backlight();
83.   //Clear LCD
84.   lcd.clear();
85.
86.   pinMode(pin_interrupt, INPUT);
87.   attachInterrupt(digitalPinToInterrupt(pin_interrupt),
    hitung_curah_hujan, FALLING); // Akan menghitung tip jika pin berlogika
    dari HIGH ke LOW
88.   // Inisialisasi RTC
89.   if (!rtc.begin())
90.   {
91.     Serial.println("Couldn't find RTC");
92.     while (1)
93.       ;
94.   }
95.   // Inisialisasi DHT11
96.   dht.begin();
97.   delay(10);
98.   WiFi.begin(ssid, pass);
99.   ThingSpeak.begin(client);
100.   //=====Hanya dibuka komen nya jika akan kalibrasi waktu saja
    (hanya sekali) setelah itu harus di tutup komennya kembali supaya tidak
    set waktu terus menerus=====
101.   rtc.adjust(DateTime(F(__DATE__), F(__TIME__))); // Set waktu
    langsung dari waktu PC

```

```

102.    //rtc.adjust(DateTime(2021, 12, 2, 8, 57, 0)); // Set Tahun,
    bulan, tanggal, jam, menit, detik secara manual
103.    // Cukup dibuka salah satu dari 2 baris diatas, pilih set waktu
    secara manual atau dari PC
104.    //=====
    ====
105.    previousMillis_1 = millis();
106.    }
107.
108.    void loop()
109.    {
110.        suhu = dht.readTemperature();
111.        if (isnan(suhu))
112.        {
113.            Serial.println("Tidak Terdeteksi");
114.            return;
115.        }
116.
117.        if (flag == true) // don't really need the == true but makes
    intent clear for new users
118.        {
119.            curah_hujan += milimeter_per_tip; // Akan bertambah nilainya
    saat tip penuh
120.            jumlah_tip++;
121.            delay(500);
122.            flag = false; // reset flag
123.        }
124.
125.        bacaRTC();
126.        curah_hujan_hari_ini = jumlah_tip * milimeter_per_tip;
127.        currentMillis_1 = millis();
128.        if (currentMillis_1 - previousMillis_1 >= interval_1) // Tampil
    bergantian setiap 2 detik sekali
129.        {
130.            if (tampil == 0)
131.            {
132.                lcd.clear();
133.                lcd.setCursor(0, 0); // Menampilkan pada kolom 0, baris 1
134.                lcd.print(konversi(tanggal) + "/" + konversi(bulan) + "/" +
    konversi(tahun) + " " + konversi(jam) + ":" + konversi(menit));
135.                lcd.setCursor(0, 1); // Menampilkan pada kolom 0, baris 0
136.                lcd.print(cuaca); // Print cuaca hari ini (Ini bukan
    ramalan cuaca tapi membaca cuaca yang sudah terjadi/ sedang terjadi hari
    ini)
137.                previousMillis_1 = millis();
138.                tampil = 1;
139.                goto end;
140.            }
141.            if (tampil == 1)
142.            {
143.                lcd.clear();
144.                lcd.setCursor(0, 0); // Menampilkan pada kolom 0, baris 1
145.                lcd.print("Tip=");
146.                lcd.print(jumlah_tip);
147.                lcd.print(" Kali");
148.                lcd.setCursor(0, 1); // Menampilkan pada kolom 0, baris 1
149.                lcd.print("Hr ini=");
150.                lcd.print(curah_hujan_hari_ini, 1); // 1 adalah 1 angka
    dibelakang koma
151.                lcd.print("mm");
152.                previousMillis_1 = millis();
153.                tampil = 2;
154.                goto end;

```

```

155.     }
156.     if (tampil == 2)
157.     {
158.         lcd.clear();
159.         lcd.setCursor(0, 0);
160.         lcd.print("/Mnt=");
161.         lcd.print(curah_hujan_per_menit, 1);
162.         lcd.print("mm");
163.         lcd.setCursor(0, 1);
164.         lcd.print("/Jam=");
165.         lcd.print(curah_hujan_per_jam, 1);
166.         lcd.print("mm");
167.         previousMillis_1 = millis();
168.         tampil = 3;
169.         goto end;
170.     }
171.
172.     if (tampil == 3)
173.     {
174.         lcd.clear();
175.         lcd.setCursor(0, 0);
176.         lcd.print("/Hari=");
177.         lcd.print(curah_hujan_per_hari, 1);
178.         lcd.print("mm");
179.         lcd.setCursor(0, 1);
180.         lcd.print("Suhu=");
181.         lcd.print(suhu, 1);
182.         lcd.print(" C ");
183.         previousMillis_1 = millis();
184.         tampil = 0;
185.     }
186.     end:
187.     delay(1);
188. }
189.
190.     temp_curah_hujan_per_menit = curah_hujan;
191.
192.     //Probabilistik           Curah           Hujan
193.     https://www.bmkg.go.id/cuaca/probabilistik-curah-hujan.bmkg
194.     if (curah_hujan_hari_ini <= 0.00 && curah_hujan_hari_ini <= 0.50)
195.     {
196.         cuaca = "Berawan ";
197.     }
198.     if (curah_hujan_hari_ini > 0.50 && curah_hujan_hari_ini <= 20.00)
199.     {
200.         cuaca = "Hujan Ringan ";
201.     }
202.     if (curah_hujan_hari_ini > 20.00 && curah_hujan_hari_ini <=
203.         50.00)
204.     {
205.         cuaca = "Hujan Sedang ";
206.     }
207.     if (curah_hujan_hari_ini > 50.00 && curah_hujan_hari_ini <=
208.         100.00)
209.     {
210.         cuaca = "Hujan Lebat ";
211.     }
212.     if (curah_hujan_hari_ini > 100.00 && curah_hujan_hari_ini <=
213.         150.00)
214.     {
215.         cuaca = "Hujan Sglt Lebat ";
216.     }
217.     if (curah_hujan_hari_ini > 150.00)
218.     {
219.         cuaca = "Hujan Sangat Lebat ";
220.     }
221. }

```

```

214.    {
215.        cuaca = "Hujan ekstrem ";
216.    }
217.    if (detik.equals("0")) // Curah hujan per menit, jam, hari
    dihitung ketika detik 0
218.    {
219.        curah_hujan_per_menit = temp_curah_hujan_per_menit;
220.        temp_curah_hujan_per_jam += curah_hujan_per_menit; // Curah
    hujan per jam dihitung dari penjumlahan curah hujan per menit namun
    disimpan dulu dalam variabel temp
221.        if (menit.equals("0"))// Curah hujan per jam baru dihitung
    ketika menit 0
222.        {
223.            curah_hujan_per_jam = temp_curah_hujan_per_jam;
224.            temp_curah_hujan_per_hari += curah_hujan_per_jam; ////
    Curah hujan per hari dihitung dari penjumlahan curah hujan per jam namun
    disimpan dulu dalam variabel temp
225.            temp_curah_hujan_per_jam = 0.00; // Reset
    temp curah hujan per jam
226.        }
227.        if (menit.equals("0") && jam.equals("0"))// Curah hujan per
    jam baru dihitung ketika menit 0 dan jam 0 (Tengah malam)
228.        {
229.            curah_hujan_per_hari = temp_curah_hujan_per_hari;
230.            temp_curah_hujan_per_hari = 0.00; // Reset
    temp curah hujan per hari
231.            curah_hujan_hari_ini = 0.00; // Reset
    curah hujan hari ini
232.            jumlah_tip = 0; // Jumlah
    tip di reset setiap 24 jam sekali (Tengah malam)
233.        }
234.        temp_curah_hujan_per_menit = 0.00;
235.        curah_hujan = 0.00;
236.        delay(1000);
237.    }
238.    printSerial();
239.    ThingSpeak.writeField(myChannelNumber, 1, curah_hujan_hari_ini,
    myWriteAPIKey);
240.    ThingSpeak.writeField(myChannelNumber, 2, suhu, myWriteAPIKey);
241.    ThingSpeak.writeField(myChannelNumber, 3,
    curah_hujan_per_menit, myWriteAPIKey);
242.    ThingSpeak.writeField(myChannelNumber, 4, curah_hujan_per_jam,
    myWriteAPIKey);
243.    //temp_jumlah_tip = jumlah_tip;
244.    WiFiClient client;
245.    //inisialisasi port web server
246.    const int httpPort = 80;
247.    if( !client.connect(host, httpPort))
248.    {
249.        Serial.println("Connection Failed");
250.        return;
251.    }
252.
253.    // kondisi terkoneksi
254.    // kirim data sensor ke database
255.
256.    String Link;
257.    HTTPClient http;
258.
259.    Link = "http://" + String(host)
    + "/TA/kirimdata.php?curah_hujan=" + String(curah_hujan_hari_ini) +
    "&suhu=" + String(suhu) + "&curah_hujan_permenit=" +

```

```

    String(curah_hujan_per_menit)      +      "&curah_hujan_perjam="      +
    String(curah_hujan_per_jam);
260.
261.      // eksekusi
262.      http.begin(Link);
263.      http.GET();
264.
265.      // baca respon setelah berhasil kirim nilai sensor
266.      String respon = http.getString();
267.      Serial.println(respon);
268.      http.end();
269.  }
270.
271.  void bacaRTC()
272.  {
273.      DateTime now = rtc.now(); // Ambil data waktu dari DS3231
274.      tanggal = String(now.day(), DEC);
275.      bulan = String(now.month(), DEC);
276.      itahun = now.year() - 2000;
277.      tahun = String(itahun);
278.      jam = String(now.hour(), DEC);
279.      menit = String(now.minute(), DEC);
280.      detik = String(now.second(), DEC);
281.  }
282.
283.  String konversi(String angka) // Fungsi untuk supaya jika angka
    satuan ditambah 0 di depannya, Misalkan jam 1 maka jadi 01 pada LCD
284.  {
285.      if (angka.length() == 1)
286.      {
287.          angka = "0" + angka;
288.      }
289.      else
290.      {
291.          angka = angka;
292.      }
293.      return angka;
294.  }
295.
296.  void printSerial()
297.  {
298.      Serial.println(konversi(jam) + ":" + konversi(menit));
299.      Serial.print("Cuaca=");
300.      Serial.println(cuaca); // Print cuaca hari ini (Ini bukan
    ramalan cuaca tapi membaca cuaca yang sudah terjadi/ sedang terjadi hari
    ini)
301.      Serial.print("Jumlah tip=");
302.      Serial.print(jumlah_tip);
303.      Serial.println(" kali ");
304.      Serial.print("Curah hujan hari ini=");
305.      Serial.print(curah_hujan_hari_ini, 1);
306.      Serial.println(" mm ");
307.      Serial.print("Curah hujan per menit=");
308.      Serial.print(curah_hujan_per_menit, 1);
309.      Serial.println(" mm ");
310.      Serial.print("Curah hujan per jam=");
311.      Serial.print(curah_hujan_per_jam, 1);
312.      Serial.println(" mm ");
313.      Serial.print("Curah hujan per hari=");
314.      Serial.print(curah_hujan_per_hari, 1);
315.      Serial.println(" mm ");
316.      Serial.print("Suhu=");
317.      Serial.print(suhu,1);

```



```

318.         Serial.println(" C");
319.         Serial.print("");
320.         Serial.println("=====
=");
321.     }

```

Program *Index.php*

```

1. <!doctype html>
2. <html lang="en">
3.   <head>
4.
5.     <meta charset="utf-8">
6.     <meta name="viewport" content="width=device-width, initial-scale=1">
7.     <meta name="description" content="">
8.     <meta name="author" content="Mark Otto, Jacob Thornton, and
Bootstrap contributors">
9.     <meta name="generator" content="Hugo 0.84.0">
10.    <link                                rel="canonical"
href="https://getbootstrap.com/docs/5.0/examples/pricing/">
11.    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.mi
n.css"                                rel="stylesheet"                                integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTWfspd3yD65VohhpuuCOMLASjC"
crossorigin="anonymous">
12.    <!-- Custom styles for this template -->
13.    <link href="pricing.css" rel="stylesheet">
14.    <!--                                <link                                rel="stylesheet"                                type="text/css"
href="assets/css/bootstrap.min.css"> -->
15.    <script                                type="text/javascript"                                src="assets/js/jquery-
3.4.0.min.js"></script>
16.    <script type="text/javascript" src="assets/js/mdb.min.js"></script>
17.    <script type="text/javascript" src="jquery-latest.js"></script>
18.
19.    <title>RainControl</title>
20.    <script type="text/javascript" src="jquery/jquery.min.js"></script>
21.    <!-- load/realtime -->
22.
23.    <script type="text/javascript">
24.      $(document).ready( function() {
25.        setInterval( function() {
26.          $("#cekhujan").load("cekhujan.php");
27.          $("#ceksuhu").load("ceksuhu.php");
28.          $("#cekhujanmenit").load("cekhujanmenit.php");
29.          $("#cekhujanjam").load("cekhujanjam.php");
30.        }, 1000 );
31.
32.      } );
33.    </script>
34.
35.  </head>
36.  <body>
37.
38.  <div class="container">
39.    <header>
40.      <div class="d-flex flex-column flex-md-row align-items-center pb-3
mb-4 border-bottom">

```

```

41.     <a href="/" class="d-flex align-items-center text-dark text-
decoration-none">
42.         <span class="fs-4">Rain Control</span>
43.     </a>
44. </div>
45.
46. <div class="pricing-header p-3 pb-md-4 mx-auto text-center">
47.     <h1 class="display-4 fw-normal" style="text-align: bold;">SISTEM
MONITORING CURAH HUJAN DAN SUHU </h1>
48. </div>
49. </header>
50.
51. <main>
52.
53.     <script type="text/javascript">
54.         var refreshid = setInterval(function(){
55.             $('#responsecontainer').load('data.php');
56.         }, 1000);
57.     </script>
58. </head>
59. <body>
60.     <!-- div untuk grafik -->
61.     <div class="container">
62.         <div class="container" id="responsecontainer" style="width: 60%"></div>
63.     </div>
64.     <script type="text/javascript">
65.         var refreshid = setInterval(function(){
66.             $('#responsecontainer').load('data.php');
67.         }, 1000);
68.     </script>
69.
70.     <!-- div untuk grafik -->
71.     <div class="container">
72.         <div class="container" id="responsecontainer" style="width: 50%"></div>
73.     </div>
74.     <script type="text/javascript">
75.         var refreshid = setInterval(function(){
76.             $('#responsecontainer').load('data.php');
77.         }, 1000);
78.     </script>
79.     <br></br>
80.
81.
82.     <div class="row row-cols-1 row-cols-md-4 mb-4 text-center">
83.     <!-- card curah hujan -->
84.         <div class="col">
85.             <div class="card mb-4 rounded-3 shadow-sm">
86.                 <div class="card-header py-3" style="background-color:
#0078ff">
87.                     <h4 class="my-0 fw-normal" style="color: white">Curah
Hujan</h4>
88.                 </div>
89.                 <div class="card-body">
90.                     <h1><span id="cekhujan">0</span></h1>
91.                     <h3 class="card-title pricing-card-title">mm</h3>
92.                 </div>
93.             </div>
94.         </div>
95.     <!-- card suhu udara -->
96.         <div class="col">
97.             <div class="card mb-4 rounded-3 shadow-sm">
98.                 <div class="card-header py-3" style="background-color:
#0078ff">

```

```

99.         <h4 class="my-0 fw-normal" style="color: white">Suhu
        Udara</h4>
100.        </div>
101.        <div class="card-body">
102.            <h1><span id="ceksuhu">0</span></h1>
103.            <h3 class="card-title pricing-card-title">°C</h3>
104.        </div>
105.    </div>
106. </div>
107. <!-- card curah hujan permenit -->
108. <div class="col">
109.     <div class="card mb-4 rounded-3 shadow-sm">
110.         <div class="card-header py-3" style="background-color:
#0078ff">
111.             <h4 class="my-0 fw-normal" style="color: white">Curah
        Hujan Permenit</h4>
112.         </div>
113.         <div class="card-body">
114.             <h1><span id="cekhujanmenit">0</span></h1>
115.             <h3 class="card-title pricing-card-title">mm</h3>
116.         </div>
117.     </div>
118. </div>
119. <!-- card curah hujan perjam -->
120. <div class="col">
121.     <div class="card mb-4 rounded-3 shadow-sm">
122.         <div class="card-header py-3" style="background-color:
#0078ff">
123.             <h4 class="my-0 fw-normal" style="color: white">Curah
        Hujan Perjam</h4>
124.         </div>
125.         <div class="card-body">
126.             <h1><span id="cekhujanjam">0</span></h1>
127.             <h3 class="card-title pricing-card-title">mm</h3>
128.         </div>
129.     </div>
130. </div>
131.
132.
133. </div>
134.
135. </main>
136. <!-- footer -->
137. <footer class="pt-4 my-md-5 pt-md-5 border-top">
138.     <div class="row">
139.         <div class="col-12 col-md">
140.             <!--  -->
141.             <small class="d-block mb-3 text-muted">&copy; 2022</small>
142.         </div>
143.
144.     </div>
145. </footer>
146. </div>
147. </body>
148. </html>

```

Program cekhujan.php

```

1. <?php
2.     // koneksi ke database
3.     $konek = mysqli_connect("localhost", "root", "", "cobasensor");
4.
5.     // baca data dari tabel tb_sensor_curah_hujan
6.     $sql = mysqli_query($konek, "select * from tb_sensor_curah_hujan
    order by ID desc");
7.
8.     // baca data paling terakhir
9.     $data = mysqli_fetch_array($sql);
10.    $curah_hujan = $data['curah_hujan'];
11.
12.    // apabila nilai curah hujan belum ada maka anggap suhunya sama
    dengam 0
13.    if( $curah_hujan == "" ) $curah_hujan = 0;
14.
15.    // cetak nilai curah hujan
16.    echo $curah_hujan ;
17. ?>

```

Program cek suhu.php

```

1. <?php
2.     // koneksi ke database
3.     $konek = mysqli_connect("localhost", "root", "", "cobasensor");
4.     // baca data dari tabel tb_sensor_curah_hujan
5.     $sql = mysqli_query($konek, "select * from tb_sensor_curah_hujan
    order by ID desc");
6.     // baca data paling terakhir
7.     $data = mysqli_fetch_array($sql);
8.     $suhu = $data['suhu'];
9.     // apabila nilai suhu belum ada maka anggap suhunya sama dengam 0
10.    if( $suhu == "" ) $suhu = 0;
11.    // cetak nilai suhu
12.    echo $suhu ;
13. ?>

```

Program cekhujanmenit.php

```

1. <?php
2.     // koneksi ke database
3.     $konek = mysqli_connect("localhost", "root", "", "cobasensor");
4.     // baca data dari tabel tb_sensor_curah_hujan
5.     $sql = mysqli_query($konek, "select * from tb_sensor_curah_hujan
    order by ID desc");
6.     // baca data paling terakhir
7.     $data = mysqli_fetch_array($sql);
8.     $curah_hujan_permenit = $data['curah_hujan_permenit'];
9.     // apabila nilai curah hujan permenit belum ada maka anggap suhunya
    sama dengam 0
10.    if( $curah_hujan_permenit == "" ) $curah_hujan_permenit = 0;
11.    // cetak nilai curah hujan permenit
12.    echo $curah_hujan_permenit ;
13. ?>

```

Program cekhujanjam.php

```

1. <?php
2.     // koneksi ke database
3.     $koneksi = mysqli_connect("localhost", "root", "", "cobasensor");
4.     // baca data dari tabel tb_sensor_curah_hujan
5.     $sql = mysqli_query($koneksi, "select * from tb_sensor_curah_hujan
    order by ID desc");
6.
7.     // baca data paling terakhir
8.     $data = mysqli_fetch_array($sql);
9.     $curah_hujan_perjam = $data['curah_hujan_perjam'];
10.    // apabila nilai curah hujan perjam belum ada maka anggap suhunya
    sama dengan 0
11.    if( $curah_hujan_perjam == "" ) $curah_hujan_perjam = 0;
12.    // cetak nilai curah hujan perjam
13.    echo $curah_hujan_perjam ;
14. ?>

```

Program kirimdata.php

```

1. <?php
2. // koneksi ke database
3. $koneksi = mysqli_connect("localhost", "root", "", "cobasensor");
4. // baca data yang dikirim ke esp32
5. $curah_hujan = $_GET['curah_hujan'];
6. $suhu = $_GET['suhu'];
7. $curah_hujan_permenit = $_GET['curah_hujan_permenit'];
8. $curah_hujan_perjam = $_GET['curah_hujan_perjam'];
9. // simpan ke tb_sensor_curah_hujan
10. // auto increment = 1
11. mysqli_query($koneksi, "ALTER TABLE tb_sensor_curah_hujan
    AUTO_INCREMENT=1");
12. // simpan data sensor
13. $simpan = mysqli_query($koneksi, "insert into
    tb_sensor_curah_hujan(curah_hujan, suhu, curah_hujan_permenit,
    curah_hujan_perjam)values('$curah_hujan', '$suhu',
    '$curah_hujan_permenit', '$curah_hujan_perjam')");
14. // uji simpan untuk memberikan respon
15. if($simpan)
16.     echo "Berhasil Dikirim";
17. else
18.     echo "Gagal";
19. ?>

```