

***PROTOTYPE DETEKSI BANJIR MENGGUNAKAN PLATFORM
THINGER.IO DAN NOTIFIKASI TELEGRAM***

SKRIPSI



DIMAS ALDI ROMADONI
NPM. 20020018

PROGRAM STUDI REKAYASA SISTEM KOMPUTER
FAKULTAS ILMU KOMPUTER
DEHASEN BENGKULU
TAHUN 2025

***PROTOTYPE DETEKSI BANJIR MENGGUNAKAN PLATFORM
THINGER.IO DAN NOTIFIKASI TELEGRAM***

SKRIPSI

**DIMAS ALDI ROMADONI
NPM. 20020018**

*Diajukan Sebagai Salah Satu Syarat Memperoleh Gelar sarjanah Strata 1 Pada
Program Studi Rekayasa Sistem Komputer Fakultas Ilmu Komputer
Universitas Dehasen Bengkulu*

PROGRAM STUDI REKAYASA SISTEM KOMPUTER

FAKULTAS ILMU KOMPUTER

DEHASEN BENGKULU

TAHUN 2025

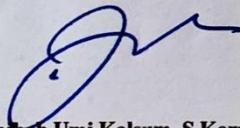
**PROTOTYPE DETEKSI BANJIR MENGGUNAKAN PLATFORM
THINGER.IO DAN NOTIFIKASI TELEGRAM**

SKRIPSI

DIMAS ALDI ROMADONI
NPM. 20020018

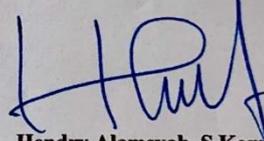
DISETUJUI OLEH :

PEMBIMBING UTAMA



Tolbah Umi Kalsum, S.Kom M.Kom
NIDN: 02.060573.01

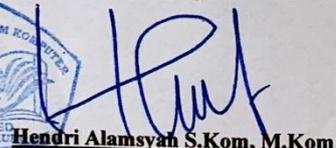
PEMBIMBING PENDAMPING



Hendry Alamsyah, S.Kom, M.Kom
NIDN: 02.110391.02

Mengetahui :

Ketua Program Studi Rekayasa Sistem Komputer



Hendri Alamsyah S.Kom, M.Kom
NIDN: 02.110391.02

**PROTOTYPE DETEKSI BANJIR MENGGUNAKAN PLATFORM
THINGER.IO DAN NOTIFIKASI TELEGRAM**

SKRIPSI

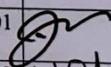
DIMAS ALDI ROMADONI
NPM. 20020018

Telah dipertahankan didepan TIM penguji Universitas Dehasen Bengkulu pada :

Hari : Kamis

Tanggal : 08 Mei 2025

Skripsi ini telah diperiksa dan disahkan oleh :

Penguji	Nama	NIDN	Tanda Tangan
Ketua Penguji	Toibah Umi Kalsum S.Kom., M.Kom	02.060573.01	
Anggota	Hendri Alamsyah, S.Kom., M.Kom	02.110391.02	
Anggota	Yessi Mardiana, S.Kom., M.Kom	02.030288.02	
Anggota	Yoli Andi Rozzi, S.,T., M.T	02.150187.04	

Mengetahui :

Dekan Fakultas Ilmu Komputer



RIWAYAT HIDUP



Penulis dilahirkan di Kecamatan Megang Sakti III, Kabupaten Musi Rawas, Sumatera Selatan pada tanggal 28 November 2001. Penulis merupakan anak pertama dari dua bersaudara yang merupakan putra dari Bapak Mulat purnomo dan Ibu Haryati.

Penulis menamatkan pendidikan Sekolah Dasar di SDN Sungai Dangku pada tahun 2014 dan melanjutkan Sekolah Menengah Pertama di SMPN Pagar Ayu yang diselesaikan pada tahun 2017. Pendidikan Sekolah Menengah Atas diselesaikan di SMAN Megang Sakti pada tahun 2020 dengan jurusan IPS. Pada tahun yang sama penulis masuk Universitas Dehasen Bengkulu dengan mengambil Program Studi Ilmu Komputer.

Selama perkuliahan penulis aktif dan menjadi anggota Himpunan Mahasiswa Rekayasa Sistem Komputer di bidang Humas Priode 2021/2022, dan menjadi ketua Himpunan Mahasiswa Rekayasa Sistem Komputer Priode 2022/2023, selama satu priode. serta penulis aktif dalam beberapa kegiatan kampus, Penulis melaksanakan kegiatan Praktik Kerja Lapangan Di Toko Al fiqih computer 2023.

MOTTO

(FORTIS FORTUNA ADIUVAT)

“Keberuntungan Akan Berpihak Kepada Mereka Yang Berani”

(Q.S AL – Insyirah ; 5)

“Sesungguhnya Bersama Kesulitan Ada Kemudahan”

Terlambat bukan berti gagal, cepat bukan berti hebat. terlambat bukan menjadi alasan untuk menyerah, setiap orang memiliki proses yang beda. percaya Proses itu yang paling penting karena Allah SWT telah mempersiapkan hal baik di balik kata proses yang kamu anggap rumit.

PERSEMBAHAN

Dengan mengucapkan Syukur Alhamdulillahirobbi alaminn sungguh sebuah perjuangan yang cukup Panjang telah aku lalui untuk mendapatkan gelar sarjana ini. Rasa Syukur dan bahagia yang kurasakan ini dengan hati tulus penulis mengucapkan terimakasih kepada:

- ❖ Allah SWT atas segala nikmat dan Rahmat serta karunia pertolongannya selama penulis menyusun skripsi.
- ❖ Nabi Muhammad SAW yang telah menjadi contoh sekaligus panutan bagi penulis.
- ❖ Bapak Mulat Purnomo dan Ibu Haryati. Yang telah melahirkan,merawat, membimbing dan melindungi dengan tulus serta penuh ke Ikhlasan, mencurahkan segala kasi sayang dan cintanya serta yang senantiasa me Doa kan dan memberi semangat dan juga dukungan sepenuh hati.
- ❖ Terimakasih untuk teman-teman ReKayasa sistem komputer angkatan 2020 kepada:hepni,endah adike putri,ovi susanti,agung satrio,afrizal kurniawan, dendy ilham,deby rahmat syukron,mas agus,rahmat agung,diki yandra, nanda eka,razon muhaimin dan Pak suwarto. Yang telah berperan banyak memberikan pengalaman dan pembelajaran selama masa-masa sulit di perkuliahan hingga si penulis ini sampai selesai menjadi sarjana.
- ❖ Terimakasih untuk diri sendiri karena telah mampu berusaha keras dan berjuang sejauh ini. Mampu mengendalikan ego dari segala masalah.

PROGRAM STUDI REKAYASA SISTEM KOMPUTER
FAKULTAS ILMU KOMPUTER UNIVERSITAS DEHASEN BENGKULU

PERNYATAAN KE ASLIAAN

Yang bertanda tangan dibawah ini:

Nama : Dimas Aldi Romadoni
Npm : 20020018
Program Studi : Rekayasa Sistem Komputer
Fakultas : Ilmu Komputer

Dengan ini menyatakan dengan sesungguhnya bahwa :

1. Selama melakukan penelitian dan pembuatan skripsi ini saya tidak melakukan pelanggaran etika akademik
2. Jika dikemudian hari diketahui bukti-bukti yang kuat ternyata Skripsi/TA tersebut dibuat oleh orang lain atau diketahui Skripsi/TA tersebut merupakan Plagiat/Mencontek/Menjiplak hasil karya ilmiah orang lain, maka dengan ini saya bersedia menerima sanksi pencabutan gelar akademik yang saya peroleh dan sanksi sanksi lainnya sesuai dengan peraturan yang berlaku.
3. Saya bersedia menanggung secara pribadi tanpa melibatkan Universitas Dehasen Bengkulu segala bentuk tuntutan hukum yang timbul atas pelanggaran hak cipta dalam karya ilmiah demikian surat pernyataan ini dibuat untuk digunakan sebagaimana mestinya.

Bengkulu, 08 Mei 2025
Hormat saya



Dimas Aldi Romadoni
NPM. 20020018

***PROTOTYPE DETEKSI BANJIR MENGGUNAKAN PLATFORM
THINGER.IO DAN NOTIFIKASI TELEGRAM***

ABSTRAK

Oleh :
Dimas Aldi Romadoni ¹
Toibah Umi Kalsum, M.Kom ²
Hendri Alamsyah, M.Kom ³

Perubahan iklim dan urbanisasi yang pesat telah meningkatkan resiko banjir di berbagai wilayah. Sistem deteksi dini menjadi krusial untuk meminimalisir kerugian material dan korban jiwa. Penelitian ini bertujuan merancang dan membangun *prototype* deteksi banjir berbasis *internet of things* (IoT) menggunakan *platform thinger Io* dan Notifikasi Telegram. Metode yang digunakan adalah *Prototype*, ini dilengkapi dengan sensor ketinggian air dan data yang mengirim secara *real – time* ke *thinger Io* untuk pantau dan di olah. Jika data menunjukkan potensi banjir, sistem secara otomatis mengirim peringatan ke pengguna melalui bot telegram agar dapat segera melakukan Tindakan pencegahan. Pengujian menunjukkan bawa sistem mampu mendeteksi kenaikan level air dengan akurasi tinggi dan mengirim Notifikasi dengan waktu respon kurang lebih 5 detik dan dengan ketinggian air 128-168cm menandai bahaya lampu led berwarna merah, lalu dengan ketinggian 73-127cm lampu led berwarna kuning bertanda waspada, kemudian aman di tunjukan dengan ketinggian 0-72cm lampu berwarna hijau. *Prototype* ini di harapkan menjadi Solusi efektif dan ekonomis dalam mendukung sistem peringatan dini bencana banjir di berbagai skala wilayah.

Kata kunci : **Internet of Things, Deteksi banjir, Thinger Io, Telegram.**
Keterangan :

¹Calon Sarjana Komputer (Sistem Komputer)

²Dosen Pembimbing 1

³Dosen Pembimbing 2

**FLOOD DETECTION PROTOTYPE USING THINGER.IO PLATFORM
AND TELEGRAM NOTIFICATION**

ABSTRACT

By:

Dimas Aldi Romadoni¹
Toibah Umi Kalsum, M.Kom²
Hendri Alamsyah, M.Kom³

Climate change and rapid urbanization have increased the risk of flooding in various regions. Early detection systems are crucial to minimize material losses and casualties. This study aims to design and build a flood detection prototype based on the internet of things (IoT) using the thinger Io platform and telegram notifications. The method used is Prototype, which is equipped with a water level sensor and data that is sent in real time to the thinger Io for monitoring and processing. If the data shows the potential for flooding, the system automatically sends a warning to the user via a telegram bot therefore preventive measures can be taken immediately. Testing shows that the system is able to detect rising water levels with high accuracy and send notifications with a response time of approximately 5 seconds and with a water height of 128-168cm marking the danger of a red LED light, then with a height of 73-127cm the yellow LED light indicates alert, then safe is indicated with a height of 0-72cm the green light. This prototype is expected to be an effective and economical solution in supporting early warning systems for flood disasters in various regional scales.

Keywords: Internet of Things, Flood detection, Thinger Io, Telegram.

Description:

1. Student
2. Supervisor 1
3. Supervisor 2



KATA PENGANTAR

Puji Syukur penulis ucapkan atas rahmat dan karunia Allah SWT, karena atas rahmat dan karunia-nya sehingga Skripsi yang berjudul *Prototype* Deteksi Banjir Menggunakan *Platform Thinger.io* Dan Notifikasi Telegram ini dapat diselesaikan dalam waktu yang telah ditetapkan.

Pada kesempatan ini penulis ingin menyampaikan ucapan terimakasih kepada pihak-pihak yang telah banyak membantu penulis selama penyusunan Skripsi ini terutama kepada :

1. Prof. Dr. Husaini, SE., M.Si., Ak., CRP selaku rektor Universitas Dehasen Bengkulu (UNIVED).
2. Khairil M.Kom selaku Dekan Fakultas Ilmu Komputer Universitas Dehasen Bengkulu.
3. Hendri Alamsyah S.Kom., M.Kom selaku ketua Program Studi Rekayasa Sistem Komputer Universitas Dehasen Bengkulu.
4. Ibu Toibah Umi Kalsum, S.Kom., M.Kom selaku Dosen Pembimbing I yang telah banyak memberikan arahan dan bimbingan kepada penulis.
5. Bapak Hendri Alamsyah S.Kom., M.Kom selaku Dosen Pembimbing II yang telah banyak memberikan arahan dan bimbingan kepada penulis.
6. Ibu Yessi Mardiana, S.Kom., M.Kom selaku Dosen Penguji I yang telah banyak memberikan masukan.
7. Ibu Yoli Andi Rozzi, S.,T., M.T selaku Dosen Penguji II yang telah memberi arahan.

Semoga segala bantuan dan bimbingan yang telah diberikan kepada penulis mendapat imbalan yang berlimpah dari Tuhan YME.

Penulis mengharapkan kritik dan saran yang sifatnya membangun dari berbagai pihak. Akhirnya semoga Skripsi ini dapat bermanfaat bagi penulis khususnya, dan bagi pembaca umumnya.

Bengkulu, 08 Mei 2025

Dimas Aldi Romadoni
NPM. 20020018

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PERSETUJUAN PEMBINGBING	iii
LEMBAR PENGESAHAN	iv
RIWAYAT HIDUP	v
MOTTO DAN PERSEMBAHAN	vi
SURAT PERNYATAAN	vii
ABSTRAK	viii
ABSTRACK	ix
KATA PENGANTAR	ix
DAFTAR ISI	xi
DAFTAR GAMBAR	xiv
DAFTAR TABEL	xvii
DAFTAR LAMPIRAN	xviii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	4
1.3 Batasan Masalah	4
1.4 Tujuan	4
1.5 Manfaat	5
BAB II LANDASAN TEORI	6
2.1 <i>Prototype</i>	6
2.2. Banjir	7
2.3 Thinger.io.....	7
2.4 Telegram	13
2.5 <i>Internet of Things (IoT)</i>	14
2.6 Modul Sensor Ultrasonik	15
2.7 Mikrokontroler	17
2.8 Nodemcu ESP8266	18
2.9 Modul <i>Relay</i>	21
2.10 <i>Buzzer</i>	22

BAB III METODOLOGI PENELITIAN	24
3.1 Subjek penelitian.....	24
3.1.1. Tempat Penelitian.....	24
3.1.2. Sejarah Berdirinya Tempat Penelitian	24
3.1.3. Struktur Organisasi	25
3.1.4. Tugas Dan Wewenang	26
3.2. Metode Penelitian	28
3.3. Instrumen Perangkat Keras dan Perangkat Lunak	30
3.3.1. Perangkat Keras	30
3.3.2. Perangkat Lunak.....	32
3.4. Metode Pengumpulan Data.....	33
3.5. Metode Pengumpulan Data.....	33
3.5.1 Diagram Global Alat	33
3.5.2 Blok Diagram	34
3.5.3 Rancangan Rangkaian Alat	35
3.5.4 Rancangan <i>User Interface</i> Sistem	35
3.5.5 Prinsip Kerja Alat	37
3.5.6 Rencana Kerja Alat	38
3.5.7 Perancangan Pengujian.....	40
BAB IV HASIL DAN PEMBAHASAN	41
4.1 Hasil	41
4.1.1 Hasil <i>Prototype</i> Deteksi Banjir	41
4.1.2 Hasil Aplikasi IoT dan Notifikasi Telegram.....	48
4.2 Pembahasan.....	49
4.2.1 Persiapan Alat dan Bahan	49
4.2.2 Perancangan, Pembuatan Alat dan Aplikasi IoT.....	51
4.2.3 Integrasi Alat dan Aplikasi IoT	59
4.2.4 Uji Coba	65
4.2.5 Hasil & Analisa	78
BAB V KESIMPULAN DAN SARAN	81
5.1 Kesimpulan	81
5.2 Saran	81

DAFTAR PUSTAKA
LAMPIRAN

DAFTAR GAMBAR

Gambar	Halaman
2.1 Tampilan <i>Dashboard</i>	8
2.2 Tampilan <i>Devices</i>	9
2.3 Tampilan <i>Bucket</i>	9
2.4 Tampilan <i>Endpoint</i>	10
2.5 Tampilan <i>Plugins</i>	11
2.6 Tampilan <i>Setting</i>	12
2.7 Logo Aplikasi Telegram	13
2.8 Sensor Ultrasonik HC-SR04	16
2.9 Nodemcu ESP 8266	20
2.10 <i>Relay</i>	22
2.11 <i>Buzzer</i>	23
3.1 Struktur Organisasi	26
3.2 Metode <i>Prototype</i>	29
3.3 Diagram Global Alat	33
3.4 Blok Diagram	34
3.5 Rancangan Rangkaian Alat	35
3.6 Rancangan <i>User Interface</i> Sistem	36
3.7 Rencana Kerja Alat	38
4.1 Rangkaian Perangkat Keras	41
4.2 Tampilan Sensor Ultrasonik Pada Serial Monitor	42
4.3 Tampilan Data Sensor Pada <i>User Interface</i>	43
4.4 <i>List Program Buzzer</i>	44
4.5 Tampilan Awal <i>User Interface</i>	45
4.6 Tampilan Notifikasi Telegram Status Aman	45
4.7 Tampilan Notifikasi Telegram Status Waspada & Bahaya	46
4.8 Tampilan Indikator LED Merah	47
4.9 Tampilan Indikator LED Kuning	47
4.10 Tampilan Indikator LED Hijau	48

4.11 Tampilan <i>User Interface</i> Deteksi Banjir Kondisi Awal.....	49
4.12 Perancangan dan Pembuatan Alat.....	52
4.13 Tampilan Halaman Pencarian Thingier.io	53
4.14 Tampilan <i>Form</i> Thingier.io.....	53
4.15 Tampilan <i>Device Connected</i>	54
4.16 Tampilan <i>Add Devices</i>	55
4.17 Tampilan <i>Dashboard</i>	55
4.18 Tampilan <i>Add Dashboard</i>	56
4.19 Tampilan <i>Dashboard Off</i>	56
4.20 Tampilan <i>Dashboard On</i>	56
4.21 Tampilan <i>Edit Dashboard</i>	57
4.22 Tampilan <i>Widget Setting Spidometer</i>	57
4.23 Tampilan <i>Widget Setting Angka</i>	58
4.24 Tampilan <i>Widget Setting LED</i>	58
4.25 Tampilan <i>Widget Setting Led Indikator</i>	59
4.26 <i>Download</i> Aplikasi Arduino IDE	60
4.27 Tampilan Persetujuan <i>Instalasi</i> Arduino.....	60
4.28 Tampilan Persetujuan <i>Instalasi</i> Arduino.....	61
4.29 Tampilan Memilih <i>Folder</i>	61
4.30 Tampilan <i>Extrack</i> ke <i>Windows</i>	62
4.31 Tampilan <i>Install USB Driver</i>	62
4.32 Tampilan Selesai <i>Install</i>	63
4.33 Tampilan Buka Arduino IDE	63
4.34 Tampilan Menu <i>Board</i>	64
4.35 Tampilan Menu <i>Port</i>	64
4.36 Tampilan <i>Listing</i> Program.....	65
4.37 Tampilan <i>Done Uploading</i>	65
4.38 Tampilan NodeMCU ESP8266 <i>Off</i>	66
4.39 Tampilan NodeMCU ESP8266 <i>On</i>	66
4.40 Tampilan Kode Program NodeMCU	67
4.41 Jarak Sensor Ultrasonik Pada Serial Monitor	68
4.42 Hasil Tampilan Awal LCD	69

4.43 Hasil Tampilan Ketinggian 8 Cm	69
4.44 Program Pada Ketinggian 8 Cm.....	70
4.45 <i>List Program Buzzer</i>	71
4.46 Tampilan Status <i>Device</i> Sistem.....	72
4.47 Tampilan lampu indikator warna merah.....	73
4.48 Tampilan lampu indikator warna hijau.....	73
4.49 Tampilan lampu indikator warna hijau.....	73
4.50 Pengambilan Data Kategori Ketinggian Air Banjir Tinggi	74
4.51 Tampilan <i>User Interface</i> Ketinggian Air Banjir Tinggi	74
4.52 Pengambilan Data Kategori Ketinggian Air Banjir Sedang	75
4.53 Tampilan User Interface Ketinggian Air Banjir Sedang.....	76
4.54 Pengambilan Data Kategori Ketinggian Air Banjir Rendah.....	77
4.55 Tampilan User Interface Ketinggian Air Banjir Rendah	77

DAFTAR TABEL

Tabel	Halaman
3.1 Spesifikasi Perangkat Keras	30
3.2 Spesifikasi Perangkat Lunak	32
3.3 Rancangan Pengujian	40
4.1 Data Indikator Lampu, Notifikasi Telegram & Level Air Banjir	73
4.2 Hasil Kriteria Pengujian	78

DAFTAR LAMPIRAN

1. Time Schedule.
2. Surat Keputusan Bimbingan.
3. Surat Keputusan Dekan Fakultas Ilmu Komputer.
4. Surat Persetujuan Penelitian.
5. Surat Balasan Izin Penelitian.
6. Data-Data Penelitian.
7. Lampiran Dokumentasi.
8. Coding *Think.io*.
9. Coding Telegram.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Indonesia merupakan salah satu negara yang dapat dikategorikan rawan terhadap bencana alam. Hal tersebut dapat terjadi dikarenakan letak posisi Geografis dari wilayah Indonesia sendiri. Selanjutnya negara Indonesia juga berada pada daerah yang memiliki iklim tropis, di mana terdiri dari dua musim berbeda yaitu panas dan hujan. Negara yang beriklim tropis memiliki ciri diantaranya terdapat perubahan cuaca, suhu dan arah angin yang cukup ekstrim. Kondisi iklim tropis yang ekstrim ini jika digabungkan dengan kondisi Geografis permukaan dan batuan yang terdapat di wilayah Indonesia, kemudian ditambah dengan aktivitas manusia terhadap lingkungan yang cenderung semakin destruktif maka akan dapat menyebabkan peristiwa bencana alam khususnya banjir.

Berdasarkan data yang diperoleh terdapat tiga jenis bencana alam yang memiliki tingkat kejadian cukup tinggi di Indonesia, diantaranya puting beliung, tanah longsor serta banjir. Kejadian bencana alam tersebut tercatat telah terjadi di Indonesia sejak tahun 2010 sampai dengan 2019 (Adiyoso, 2018). Pada data tersebut dapat dilihat bahwa hingga akhir bulan September 2019, mencatat jumlah kejadian banjir di Indonesia telah mencapai 549 kasus, tanah longsor 550 kasus, serta puting beliung 725 kasus. Akan tetapi dari ketiga jenis bencana alam tersebut, tercatat bencana banjir yang paling banyak menimbulkan jumlah korban jiwa yang meninggal atau hilang yaitu

sekitar 326 orang.

Seperti yang dikutip dari laman bnpb.go.id bencana banjir dikota Bengkulu sendiri terjadi terakhir kali pada tanggal 23 November 2023 yang melanda beberapa kecamatan di Kota Bengkulu telah surut. BNPB mencatat ada 8 kecamatan yang dilanda banjir yaitu diantaranya adalah kecamatan Ratu Agung, Gading Cemaka, Selebar, Kampung Melayu, Ratu Samban, Bangkahulu, Singaran Pati dan Palaran. Pada saat terjadinya banjir, ketinggian muka air mencapai 100 CM di beberapa titik. Saat terjadinya banjir tim dari BNPB yaitu Tim Reaksi Cepat (TRC) segera melakukan penanganan darurat seperti evakuasi dan kaji cepat. Warga dihimbau untuk dapat siap siaga dalam menghadapi potensi bahaya hidrometeorologi, hendaknya setiap individu dapat mempersiapkan diri dalam menghadapi bahaya tersebut.

Jika berkaca dari kejadian banjir yang telah terjadi, sehingga menimbulkan jumlah korban jiwa meninggal atau hilang yang tidak sedikit, seharusnya dampak dari bencana banjir tersebut dapat dicegah apabila masyarakat lebih siap dan siaga dalam menghadapi datangnya bencana banjir tersebut. Adapun salah satu upaya yang dapat dilakukan oleh pemerintah adalah dengan memberikan informasi serta peringatan dini kepada masyarakat. Kegiatan pemberian informasi serta peringatan dini tersebut diharapkan dapat menjadi salah satu cara yang digunakan untuk mengurangi jumlah korban jiwa apabila terjadi banjir yang disebabkan oleh kurangnya persiapan masyarakat dalam menghadapi bencana banjir tersebut. Media yang digunakan dapat berupa notifikasi Telegram, di mana data yang

akan diterima merupakan data *real-time* di lapangan.

Penelitian terdahulu mengenai teknologi deteksi banjir pernah dilakukan sebelumnya diantaranya dilakukan oleh Abdullah (2010). Penelitian tersebut dilakukan dengan dikonstruksikan menggunakan kombinasi sensor water level, sensor ultrasonik (SRF-05) dan *water flow* untuk mengetahui unjuk kerja sistem monitoring permukaan dan debit air sungai serta intensitas curah hujan sebagai peringatan dini banjir, menghasilkan sebuah program hasil monitoring dan mengetahui keefektifan mikrokontroler dalam mengukur dan mengendalikan sistem. Kemudian penelitian yang dilakukan oleh Rut Dias (2021) membuat sebuah sistem deteksi dini banjir dengan menggunakan mikrokontroler yang dihubungkan dengan sensor ultrasonik sebagai pengukur ketinggian air pada saluran drainase.

Dari beberapa penelitian-penelitian yang telah dipaparkan, bahwasanya penelitian yang sebelumnya telah di buat dapat disimpulkan bahwa alat yang berbasis mikrokontroler arduino tidak dapat di kendalikan dari jarak jauh dan data yang di dapat belum secara *Real-time*. Maka perlu adanya pengembangan yang bisa mengontrol alat dari jarak jauh yaitu alat yang berbasis *Internet of Things*. Menurut Sandi dan Fatma (2023) IoT merupakan suatu konsep yang bertujuan untuk dapat memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus. Selanjutnya pada penelitian ini akan menggunakan *Thinger.io* sebagai *user interface* dengan notifikasi Telegram sehingga dapat memberikan informasi data secara *real time*. Berdasarkan uraian latar belakang di atas, maka dari itu penulis

mengangkat judul “*Prototype* Deteksi Banjir Dengan *Platform Thinger.io* dan Notifikasi Telegram”.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, maka penulis mengangkat rumusan masalah yaitu Bagaimana *Prototype* Deteksi Banjir Dengan *Platform Thinger.io* dan Notifikasi Telegram?

1.3 Batasan Masalah

Batasan masalah pada penelitian ini yaitu:

- a. Data yang didapatkan berupa ketinggian level air.
- b. Notifikasi atau pemberitahuan ketinggian level air dapat diketahui pengguna melalui Telegram.
- c. Pengujian yang dilakukan adalah menguji sensor ultrasonik HCSR04 dan NodeMCU Esp 8266.
- d. Sistem deteksi banjir dengan *platform Thinger.io* dan notifikasi Telegram dalam bentuk *Prototype*.

1.4 Tujuan

Adapun tujuan sebagai berikut:

1. Tujuan Umum

Tujuan dari penulisan proposal Skripsi ini adalah untuk memenuhi salah satu syarat untuk melanjutkan Skripsi pada Program Studi Rekayasa Sistem Komputer Universitas Dehasen Bengkulu.

2. Tujuan Khusus

Adapun tujuan khusus dari pembuatan proposal ini adalah :

- a. Membuat dan merancang sistem deteksi banjir dengan menggunakan *platform Thinger.io* dan notifikasi Telegram.
- b. Mengetahui akurasi sitem deteksi banjir dengan menggunakan *platform Thinger.io* dan notifikasi Telegram.

1.5 Manfaat

- a. Untuk mengetahui cara pembuatan *prototype* deteksi banjir dengan menggunakan *platform Thinger.io* dan notifikasi Telegram.
- b. Untuk mengetahui cara kerja dan cara pengoperasian pengukur ketinggian level air pada deteksi banjir.
- c. Mengetahui cara menerapkan sistem *Internet of Things* (IoT) pada sistem deteksi banjir dengan menggunakan *platform Thinger.io* dan notifikasi Telegram.

BAB II

LANDASAN TEORI

2.1 *Prototype*

Menurut Siswidiyanto, S., dkk (2020 : 16), *prototype* adalah tipe yang asli, bentuk, atau contoh dari sesuatu yang dipakai sebagai contoh yang khas, dasar, atau standar untuk hal-hal lain dari kategori yang sama. Dalam bidang desain, sebuah prototipe dibuat sebelum dikembangkan atau justru dibuat khusus untuk pengembangan sebelum dibuat dalam skala sebenarnya atau sebelum diproduksi secara masal. Kategori prototipe dasar, tidak ada kesepakatan umum tentang apa yang merupakan prototipe dan kata tersebut sering digunakan bergantian dengan kata “model”.

Menurut Sutha (2018 : 1689), *prototype* merupakan salah satu dari implementasi sebuah desain produk yang akan dibangun. *Prototype* adalah sebuah bukti fisik atau konsep dari sebuah konsep perancangan. *Prototype* dapat membantu untuk menentukan apakah pengembang aplikasi telah berada pada jalur yang benar selama proses pengembangan. Pelatih dan pengembang profesional menggunakan teknik *prototype* untuk menghasilkan umpan balik dari pengguna akhir selama proses perancangan sebelum akhirnya mulai membuat code aplikasi. Mendapatkan umpan balik yang kritis akan membantu dalam menghasilkan desain akhir yang pada akhirnya berdampak besar pada pengguna.

Berdasarkan pengertian *prototype* menurut 2 penelitian sebelumnya dapat disimpulkan bahwa *prorotype* adalah sebuah bentuk atau contoh dari

sesuatu yang dipakai atau standar untuk hal-hal lain dari kategori yang sama dari sebuah perancangan.

2.2. Banjir

Menurut Firmansyah (2019 : 26) banjir merupakan suatu aliran air atau penggenangan air yang berasal dari badan utama (badan sungai) atau anak sungai secara berlebihan sehingga dapat membuat kerusakan pada lingkungan sekitarnya. Perbedaan aliran banjir dengan debit normal ditentukan oleh tinggi aliran air dimana banjir ditunjukkan aliran air yang melampaui kapasitas tampung tebing/tanggul sungai sehingga menggenangi daerah sekitarnya.

Menurut Yulaelawati (2019:21) untuk memahami pengertian banjir dengan baik, maka di sini disajikan gambaran mengenai daerah penguasaan sungai. Di dalam suatu ekosistem sungai terdapat bagian-bagian tidak terpisahkan satu dengan yang lainnya, yaitu palung sungai yang selalu tergenang oleh aliran sungai, dataran banjir yang akan tergenang jika air sungai meluap, dan bantaran atau sempadan sungai.

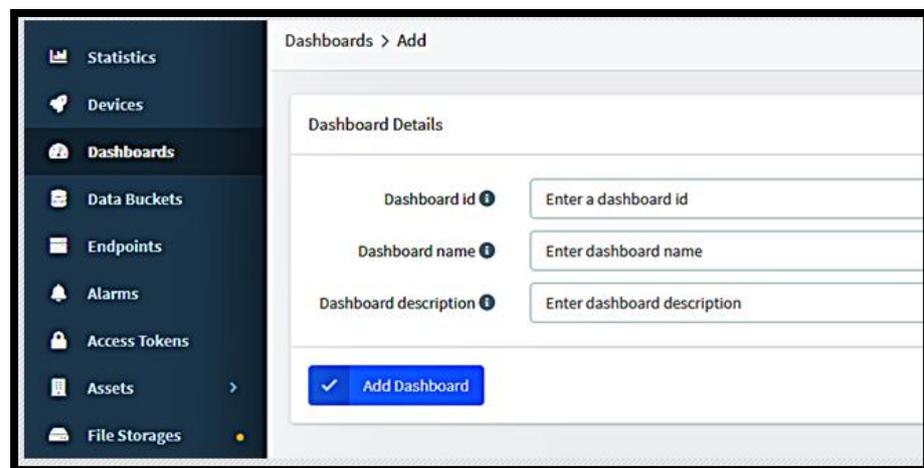
Berdasarkan pengertian banjir yang telah dijelaskan sebelumnya, maka dapat disimpulkan bahwa banjir adalah suatu aliran air yang mengalami keadaan di mana suatu daerah tergenang oleh air dalam jumlah yang banyak.

2.3 Thinger.io

Menurut Agus (2020 :142), *platform* Thinger.io adalah *platform* IoT yang menyediakan setiap alat yang diperlukan untuk membuat prototipe, skala, dan mengelola produk yang terhubung dengan cara yang sangat sederhana. Berikut adalah beberapa menu utama yang terdapat di Thinger.io :

1. *Dashboard*:

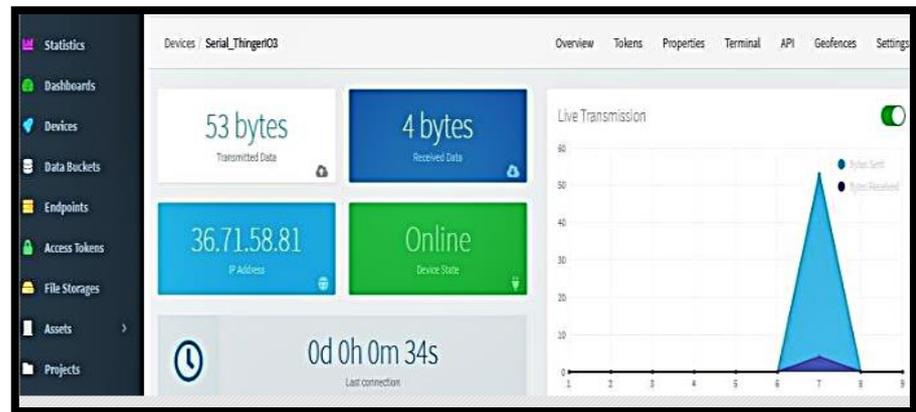
- a. **Overview**: Tampilan umum dari semua perangkat dan data yang terhubung ke akun pengguna.
- b. **Custom Widgets**: Membuat dan mengatur widget untuk menampilkan data sensor secara visual, seperti grafik, indikator, dan peta.



Gambar 2.1 Tampilan *Dashboard*

2. *Devices*:

- a. **Device Manager**: Mengelola perangkat yang terhubung, termasuk menambah, mengedit, dan menghapus perangkat.
- b. **Device Details**: Menampilkan detail spesifik dari masing-masing perangkat, termasuk status konektivitas, data real-time, dan logs.



Gambar 2.2 Tampilan *Devices*

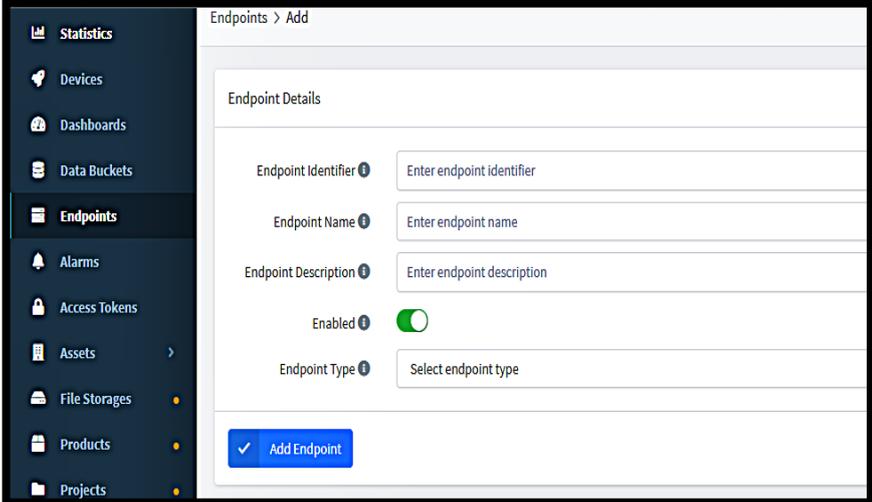
3. *Buckets*:

- a. ***Data Buckets***: Menyimpan dan mengelola data yang dikumpulkan dari perangkat. Data buckets berfungsi sebagai repositori untuk data historis yang bisa dianalisis atau diambil kapan saja.
- b. ***Bucket Settings***: Konfigurasi pengaturan penyimpanan data, termasuk interval pengambilan data dan kapasitas penyimpanan.

Gambar 2.3 Tampilan *Bucket*

4. *Endpoints*:

- a. **HTTP *Endpoints***: Membuat dan mengelola endpoints untuk berkomunikasi dengan layanan lain melalui HTTP, memungkinkan integrasi dengan API pihak ketiga.
- b. **MQTT *Endpoints***: Mengelola endpoints untuk komunikasi MQTT, yang sering digunakan untuk komunikasi antara perangkat IoT.



The screenshot shows a web interface for adding a new endpoint. On the left is a dark sidebar with navigation items: Statistics, Devices, Dashboards, Data Buckets, Endpoints (highlighted), Alarms, Access Tokens, Assets, File Storages, Products, and Projects. The main content area is titled 'Endpoints > Add' and contains a form with the following fields:

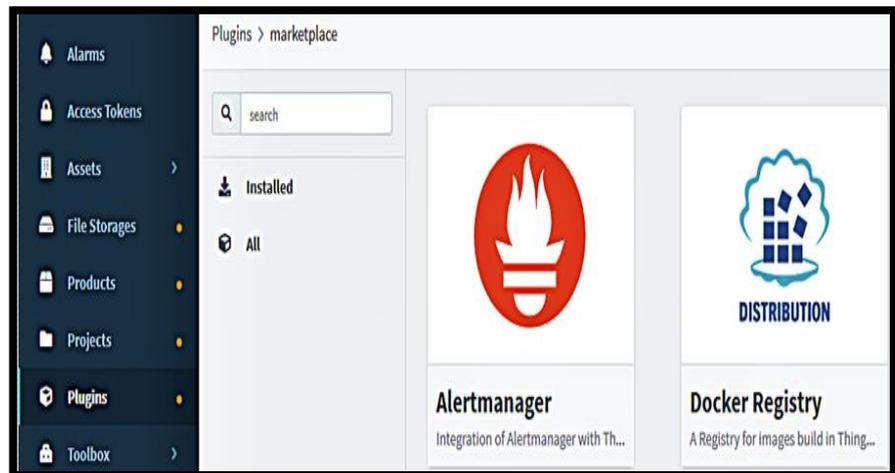
- Endpoint Identifier: Text input with placeholder 'Enter endpoint identifier'
- Endpoint Name: Text input with placeholder 'Enter endpoint name'
- Endpoint Description: Text input with placeholder 'Enter endpoint description'
- Enabled: Toggle switch, currently turned on (green)
- Endpoint Type: Select dropdown with placeholder 'Select endpoint type'

At the bottom of the form is a blue button with a checkmark icon and the text 'Add Endpoint'.

Gambar 2.4 Tampilan *Endpoint*

5. *Plugins*:

- a. **Integrations**: Mengintegrasikan Thingier.io dengan layanan lain seperti IFTTT, Zapier, atau *platform cloud* lainnya.
- b. **Extensions**: Memperluas fungsionalitas platform dengan plugin khusus.



Gambar 2.5 Tampilan *Plugins*

6. *Console*:

- a. *Interactive Console*: Mengakses perangkat secara interaktif untuk melakukan debugging, pengujian, dan konfigurasi langsung dari platform Thinger.io.
- b. *Logs*: Melihat log aktivitas dan data untuk memantau kinerja perangkat dan menemukan masalah.

7. *API Explorer*:

- a. *API Access*: Menjelajahi dan menguji API yang tersedia untuk mengakses data dan mengontrol perangkat melalui permintaan HTTP.
- b. *Documentation*: Panduan dan dokumentasi tentang penggunaan API Thinger.io.

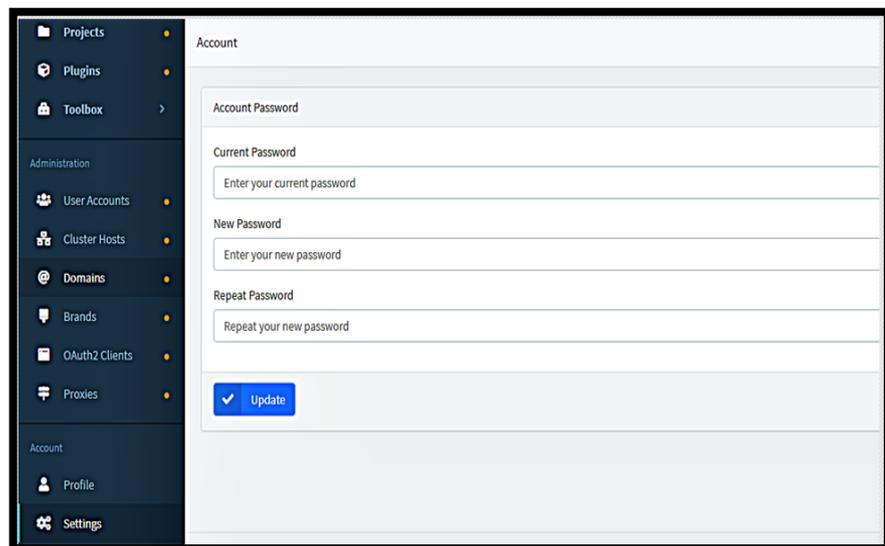
8. *Account*:

- a. *User Profile*: Mengelola informasi profil pengguna, termasuk detail login dan preferensi.

- b. **Subscription**: Mengelola paket langganan dan opsi pembayaran untuk layanan Thinger.io.

9. **Settings**:

- a. **Platform Settings**: Pengaturan umum platform, termasuk konfigurasi server, keamanan, dan lainnya.
- b. **Notifications**: Mengatur notifikasi untuk berbagai kondisi, seperti perubahan status perangkat atau data sensor yang melebihi ambang batas.



Gambar 2.6 Tampilan *Setting*

Menurut wahyu (2021 :8) *Platform* Thinger.io adalah *platform Internet of Things (IoT)* yang menyediakan fitur *cloud* untuk menghubungkan berbagai perangkat yang terkoneksi dengan internet. Thinger.io juga dapat memvisualisasikan hasil pembacaan sensor dalam bentuk nilai atau grafik.

Berdasarkan pengertian *platform* Thinger.io menurut 2 penelitian sebelumnya dapat disimpulkan bahwa *platform* Thinger.io adalah *cloud* yang

menyediakan setiap alat yang di perlukan untuk membuat prototipe dan menyediakan fitur *cloud* untuk menghubungkan berbagai perangkat yang terkoneksi dengan internet.

2.4 Telegram

Menurut Sutrahitu dkk (2021: 346) aplikasi telegram adalah aplikasi *messenger* berbasis *cloud* untuk *smartphone* dan laptop serta fokus pada keamanan dan kecepatan. Sistem operasi *smartphone* yang dapat menggunakan aplikasi Telegram adalah iPhone/iPad, Android, *Windows Phone*, serta dapat digunakan pada laptop dengan sistem operasi PC/Mac/Linux, macOS dan melalui aplikasi Web-browser. Adapun logo dari aplikasi Telegram seperti yang ditunjukkan pada gambar 2.7.



Gambar 2.7 Logo Aplikasi Telegram

Menurut Pasaribu & Susanti (2021 : 29), telegram adalah fitur telegram yang mempunyai fungsi khusus dan berjalan otomatis sesuai dengan perintah atau request user. Secara umum Aplikasi Telegram tidak jauh berbeda dengan Aplikasi Whatsapp, akan tetapi tidak sebanyak pengguna Aplikasi Whatsapp. Aplikasi Telegram di kancah aplikasi messenger telah mengundang perhatian para pengguna aplikasi messenger dan para

pengembang aplikasi messenger, termasuk pendiri Aplikasi Whatsapp. Beberapa fitur dari Aplikasi Telegram yang dapat membantu komunikasi dalam proses pembelajaran adalah fitur *Reply*, *Mention*, *Hashtag*.

- a. *Reply* berguna untuk membalas chat tertentu dari seseorang dengan ikut menyertakan *chat* tersebut dalam balasan.
- b. Salah satu kegunaan mention (dengan format *@username*) pada *chat group* adalah apabila ada hal yang penting untuk diketahui orang yang di-*mention*, maka orang tersebut akan mendapatkan *notification* (pemberitahuan) walaupun orang tersebut mematikan *notification (mute)* dari *group* yang bersangkutan.
- c. Bagi yang sudah terbiasa dengan dunia internet tentu sudah mengenal hashtag. Hashtag membuat chat pada group lebih terstruktur. Kata yang dimulai dengan # apabila diklik atau di-tap akan menghasilkan daftar hasil pencarian sesuai dengan kata tersebut.

Berdasarkan pengertian telegram yang telah dijelaskan sebelumnya, maka dapat disimpulkan bahwa telegram adalah aplikasi *messenger* berbasis *cloud* untuk *smartphone* dan laptop serta fokus pada keamanan dan kecepatan.

2.5 Internet of Things (IoT)

Menurut Metha dalam Arafat (2016:264), *Internet of Things* atau dikenal juga dengan singkatan IoT, adalah sebuah teknologi yang menginovasi benda-benda yang ada disekitar menggunakan internet sehingga kegiatan sehari-hari dapat menjadi lebih praktis dan efisien. Dengan semakin berkembangnya infrastruktur internet, maka kita menuju babak berikutnya,

di mana bukan hanya *smartphone* atau komputer saja yang dapat terkoneksi dengan internet. Namun berbagai macam benda nyata akan terkoneksi dengan internet. Sebagai contohnya dapat berupa: mesin produksi, mobil, peralatan elektronik, peralatan yang dapat dikenakan manusia (*wearables*), dan termasuk benda nyata apa saja yang semuanya tersambung ke jaringan lokal dan global menggunakan sensor dan atau aktuator yang tertanam.

Menurut Rifky (2020:187), *Internet of Things* merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus. Adapun kemampuan seperti berbagi data, *remote control*, dan sebagainya, termasuk juga pada benda di dunia nyata. Contohnya bahan pangan, elektronik, koleksi, peralatan apa saja, termasuk benda hidup yang semuanya tersambung ke jaringan lokal dan global melalui sensor yang tertanam dan selalu aktif. Pada dasarnya, IoT mengacu pada benda yang dapat diidentifikasi secara unik sebagai representasi virtual dalam struktur berbasis Internet.

Berdasarkan pengertian *Internet of Things* menurut beberapa para ahli diatas dapat disimpulkan bahwa *Internet of Things* merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus.

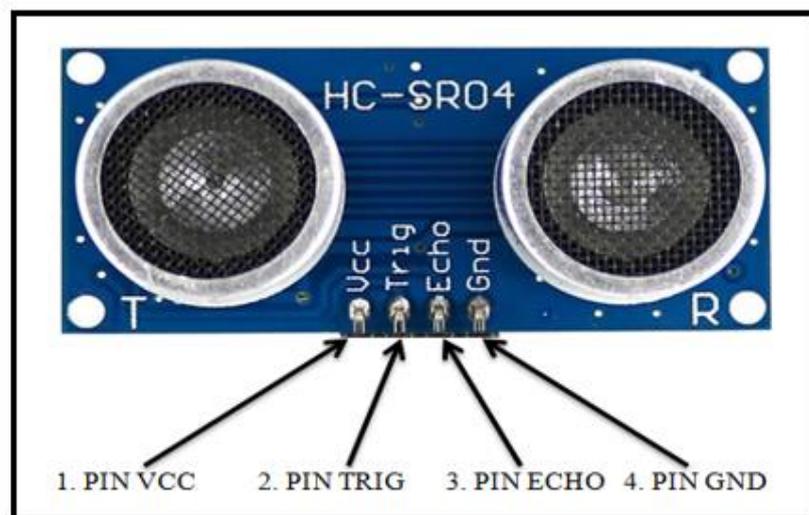
2.6 Modul Sensor Ultrasonik

Menurut Arta (2019: 3), sensor ultrasonik adalah perangkat elektronika yang kemampuannya bisa mengubah energi listrik menjadi energi mekanik dalam bentuk gelombang suara ultrasonik. Sensor ini terdiri dari

rangkaian pemancar ultrasonik yang dinamakan *transmitter* dan penerima ultrasonik yang disebut *receiver*.

Menurut Tryan (2020: 3) Sensor ultrasonik adalah sebuah sensor yang berfungsi untuk mengubah besaran fisis (bunyi) menjadi listrik. Sensor ini gelombang ultrasonik dibangkitkan melalui benda yang disebut dengan *piezoelektrik*. *Piezoelektrik* ini akan menghasilkan gelombang ultrasonik dengan frekuensi 40kHz ketika sebuah *osilator* diterapkan pada benda tersebut.

Berdasarkan menurut 2 para ahli di atas pengertian sensor ultrasonik dapat di simpulkan adalah sensor yang berfungsi untuk mengubah besaran fisis (bunyi) menjadi besaran listrik dan sebaliknya.



Gambar 2.8 Sensor Ultrasonik HC-SR04

Spesifikasi kegunaan setiap Pin:

1. Pin VCC, berfungsi untuk mengoneksikan sensor ke *power supply* 5 volt Arduino. Jadi kamu bisa langsung mengoneksikan pin VCC ke pin 5V di Arduino.

2. Pin Trig (*Trigger*), trigpin Arduino berfungsi untuk memicu pemancaran gelombang ultrasonik. Gelombang akan terpancarkan saat pin ini diberikan logika *HIGH*.
3. Pin *Echo*, berfungsi untuk mendeteksi pantulan gelombang ultrasonik apakah sudah diterima atau belum. Pin *Echo* bernilai *HIGH* jika gelombang pantulan belum diterima dan bernilai *LOW* jika pantulan sudah diterima.
4. Pin GND, berfungsi untuk mengoneksikan sensor ke *power supply ground*. Sama dengan pin VCC, kamu juga bisa langsung menghubungkan pin GND ini ke pin GND Arduino.

2.7 Mikrokontroler

Menurut Nugroho dkk (2021:3), Mikrokontroler adalah bagian – bagian port yang digunakan untuk memasang bagian komponen yang diperlukan. Pada Mikrokontroler terdapat 14 port, dan digunakan sesuai dengan tujuan penelitian. Mikrokontroler dapat dikatakan sebagai sebuah chip yang memiliki tugas dalam mengatur rangkaian elektronika dan umumnya dapat ditanamkan program didalamnya.

Menurut Rifky (2020:188) mikrokontroler atau kadang dinamakan pengontrol tertanam (*Embedded Controller*) adalah suatu sistem yang mengandung masukan atau keluaran, memori, dan prosesor yang digunakan pada produk seperti mesin cuci, pemutar video, mobil dan telepon. Pada prinsipnya, Mikrokontroler adalah sebuah komputer berukuran kecil yang dapat digunakan untuk mengambil keputusan, melakukan hal-hal bersifat berulang dan dapat berinteraksi dengan peranti-peranti eksternal, seperti

sensor ultrasonik untuk mengukur jarak terhadap suatu objek, penerima GPS untuk memperoleh data posisi kebumihan dari satelit dan motor untuk mengontrol gerak pada robot. Sebagai komputer yang berukuran kecil, Mikrokontroler cocok diaplikasikan pada benda-benda yang berukuran kecil, misalnya sebagai pengendali pada robot .

Menurut Sanjaya dkk (2022:451), Mikrokontroler merupakan otak dari pada sistem yang dirancang, yang berfungsi sebagai pusat pengontrol / kendali untuk mengelola data dari semua *periferal* yang terhubung dengan sistem ini, baik itu sensor, penampil maupun perangkat komunikasi nirkabel.

Berdasarkan definisi tersebut bisa dinyatakan bahwa Mikrokontroller merupakan suatu integrated circuit yang dirancang dengan kepadatan tinggi, dimana bagian yang dibutuhkan suatu Mikrokontroler sudah dibuat menjadi kepingan, biasanya mencakup CPU (*Central Processing Unit*), EEPROM/EPROM/PROM/ROM, RAM (*Random Access Memory*), Parallel & Serial, *Timer* dan *Interrupt Controller* yang berfungsi sebagai pengatur rangkaian elektronik serta secara umum dapat ditanamkan program di dalamnya.

2.8 Nodemcu ESP8266

Menurut Rostini (2020:3), *NodeMCU* ESP8266 merupakan chip yang terintegrasi dan dirancang guna menghubungkan Mikrokontroler dengan internet melalui *Wi-Fi*. Ia menawarkan solusi jaringan *Wi-Fi* yang lengkap dan mandiri, yang memungkinkan untuk menjadi *host* ataupun sebagai *Wi-Fi client*. ESP8266 memiliki kemampuan pengolahan dan penyimpanan *on-board* yang kuat, yang memungkinkannya untuk diintegrasikan dengan sensor

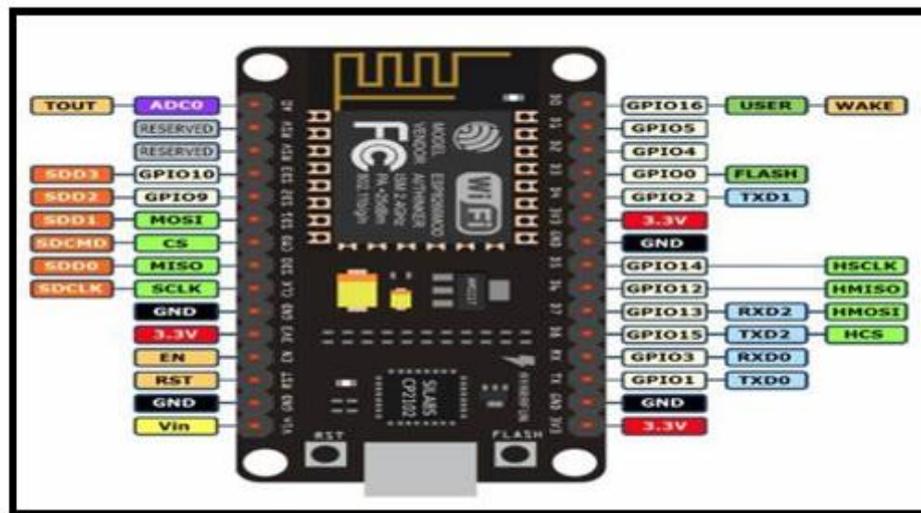
dan aplikasi perangkat khusus lain melalui GPIOs dengan pengembangan yang mudah serta waktu *loading* yang minimal.

Menurut Doni & Rahman (2020:518), ESP8266 merupakan satu chip yang telah lengkap didalamnya sudah termasuk *processor*, memori dan juga akses ke GPIO. Hal ini menyebabkan ESP8266 dapat secara langsung menggantikan Arduino dan ditambah lagi dengan kemampuannya untuk *mensupport* koneksi *wifi* secara langsung.

Menurut Zega (2022:03), ESP8266 adalah modul *wifi* yang memiliki fungsi sebagai perangkat tambahan Mikrokontroler seperti Arduino agar dapat terhubung langsung dengan Wi-Fi dan membuat koneksi TCP/IP. Modul Wi-Fi ini sudah bersifat SoC (*System on Chip*), sehingga kita bisa melakukan *programming* langsung ke ESP8266 tanpa memerlukan Mikrokontroler tambahan.

ESP8266 memiliki kemampuan *on-board processing* dan *storage* yang memungkinkan chip tersebut untuk diintegrasikan dengan sensor-sensor atau dengan aplikasi alat tertentu melalui pin input output hanya dengan pemrograman singkat. Dengan level yang tinggi berupa *onchip* yang terintegrasi memungkinkan *external* sirkuit yang ramping dan semua solusi, termasuk modul sisi depan, didesain untuk menempati area PCB yang sempit.

Berdasarkan pengertian Modul ESP8266 menurut beberapa para ahli diatas dapat disimpulkan bahwa modul ESP8266 adalah chip terintegrasi yang dirancang untuk menghubungkan Mikrokontroler dengan internet melalui Wi-Fi.



Gambar 2.9 Nodemcu ESP 8266

Keterangan pin Nodemcu ESP8266 :

- a. Micro-USB : Fungsinya sebagai power yang dapat terhubung dengan USB *port*. Selain itu, biasanya juga digunakan untuk melakukan pengiriman *sketch* atau memantau data serial dengan serial monitor di aplikasi Arduino IDE.
- b. 3.3V : Digunakan sebagai tegangan untuk *device* lainnya. ada 3 tempat untuk 3.3v. Biasanya juga dituliskan hanya 3V (Sebenarnya tetap 3,3V)
- c. GND : *Ground*. Sebagai tegangan 0 atau nilai negatif untuk mengalirkan arus.
- d. Vin : Sebagai *External Power* yang akan mempengaruhi *Output* dari seluruh pin. Cara menggunakannya yaitu dengan menghubungkannya dengan tegangan 7 hingga 12 volt.
- e. EN, RST : Pin yang digunakan untuk *reset* program di Mikrokontroler.
- f. A0 : Analog pin, digunakan untuk membaca input secara analog.

- g. GPIO 1 – GPIO 16 : Pin yang dapat digunakan sebagai *input* dan *output*. Pin ini dapat melakukan pembacaan dan pengiriman data secara analog juga.
- h. *SD1,CMD, SD0,CLK* : SPI Pin untuk komunikasi SPI (Serial *Peripheral Interface*) dimana kita akan menggunakan *clock* untuk sinkronisasi deteksi bit pada *receiver*.
- i. TXD0, RXD0, TXD2, RXD2 : Sebagai *interface UART*, Pasangannya adalah TXD0 dengan RXD0 dan TXD2 dengan RXD2. TXD1 digunakan untuk *upload firmware/program*.
- j. SDA, SCL (*I2C Pins*) : Digunakan untuk *device* yang membutuhkan I2C.

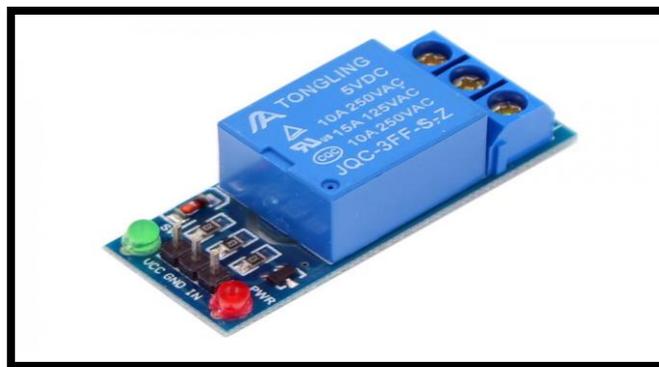
2.9 Modul Relay

Menurut Riyanto (2019:5), Relay merupakan sebuah komponen elektronika berupa saklar elektronik yang digerakkan oleh arus listrik. Secara prinsip, *relay* merupakan tuas saklar dengan lilitan kawat pada batang besi (solenoid) di dekatnya. Ketika solenoid dialiri arus listrik, tuas akan tertarik karena adanya gaya magnet yang terjadi pada solenoid sehingga kontak saklar akan menutup.

Menurut Anata (2022:125), Relay adalah Saklar (*Switch*) yang dioperasikan secara listrik dan merupakan komponen *Electromechanical* (*Elektromekanikal*) yang terdiri dari 2 bagian utama yakni Elektromagnet (Coil) dan Mekanikal (seperangkat Kontak Saklar/*Switch*).

Menurut Gozal (2020:41), *Relay* pengaman merupakan suatu alat baik elektronik maupun magnetik yang dirancang untuk merasakan dan mendeteksi suatu kondisi ketidaknormalan pada sistem tenaga listrik.

Maka dapat disimpulkan bahwa *Relay* adalah komponen elektronika berupa saklar elektronik yang digerakkan oleh arus listrik yang dioperasikan secara listrik dan merupakan komponen *Electromechanical* (Elektromekanikal) yang dirancang untuk merasakan dan mendeteksi suatu kondisi ketidaknormalan pada sistem tenaga listrik.



Gambar 2.10 *Relay*

2.10 *Buzzer*

Menurut Raharja (2021:189), *Buzzer* adalah sebuah komponen elektronika yang berfungsi untuk mengubah getaran listrik menjadi getaran suara. *Buzzer* terdiri dari sebuah diafragma yang memiliki kumparan, jika kumparan tersebut dialiri arus listrik kumparan akan tertarik kedalam tergantung dari polaritas magnetnya

Menurut Hartanto dan Prabowo (2021 : 31), *buzzer* adalah sebuah komponen elektronika yang dapat mengubah sinyal listrik menjadi getaran suara. *Buzzer* ini biasa dipakai pada sistem *alarm*.

Menurut Lestari dkk (2020: 214), *Buzzer* merupakan sebuah modul komponen elektronika kategori transduser, yang bekerja dengan cara mengubah sinyal elektrik menjadi sebuah gelombang suara *Buzzer* biasa difungsikan sebagai alarm sinyal. Biasa di implementasikan pada *project* penelitian sebagai sebuah indikator terhadap suatu kondisi.

Jadi dapat disimpulkan bahwa *Buzzer* adalah sebuah komponen elektronika yang berfungsi untuk mengubah getaran listrik menjadi getaran suara yang dapat menjadi sistem alarm, dengan modul komponen elektronika kategori transduser, yang bekerja dengan cara mengubah sinyal elektrik



Gambar 2.11 *Buzzer*

Berikut adalah spesifikasi yang dimiliki oleh *Buzzer* :

1. Tegangan kerja: 4v-8v DC (optimal 5v)
2. Arus max: 30mA / 5vDC
3. Kekuatan suara max: 85dB / 10cm
4. Frek resonansi: 2500 +/- 300hz
5. Suhu kerja: -20 ~ +70 C
6. Warna: hitam
7. Diameter: 1cm

BAB III

METODOLOGI PENELITIAN

3.1 Subjek penelitian

3.1.1. Tempat Penelitian

Penelitian dilaksanakan di Kementrian Pekerja Umum Perumahan Rakyat Direktorat Jenderal Sumber Daya Air Bengkulu yang beralamatkan di Jl. Batang Hari No. 25 Kecamatan Ratu Agung Kota Bengkulu.

3.1.2. Sejarah Berdirinya Tempat Penelitian

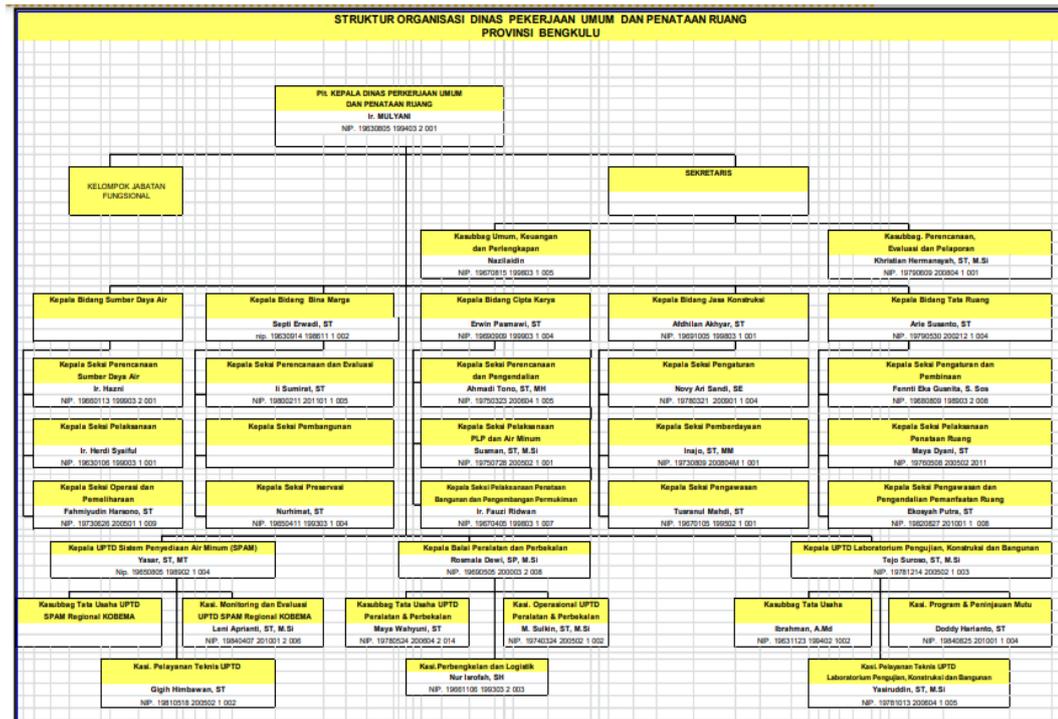
Dinas Pekerjaan Umum dan Penataan Ruang dan Penataan Ruang Provinsi Bengkulu, merupakan suatu Organisasi Perangkat Daerah yang dibentuk berdasarkan Peraturan Daerah Provinsi Bengkulu No.8 Tahun 2016 tanggal 29 November Tahun 2016 Tentang “Pembentukan dan Susunan Perangkat Daerah Provinsi Bengkulu”. Dimana Dinas ini merupakan unsur pelaksana urusan pemerintahan bidang pekerjaan umum dan penataan ruang dengan tipeologi B yang dipimpin oleh seorang Kepala Dinas Eselon II/A yang berada di bawah dan bertanggung jawab kepada Gubernur Bengkulu melalui Sekretaris Daerah. Dinas Pekerjaan Umum dan Penataan Ruang dan Penataan Ruang Provinsi adalah unsur pelaksana Pemerintah Provinsi di Bidang Pekerjaan Umum yang mempunyai tugas membantu Gubernur melaksanakan urusan pemerintahan dan tugas pembantuan bidang pekerjaan umum dan

penataan ruang yang menjadi kewenangan provinsi. Dalam penyelenggaraan tugas tersebut, Dinas Pekerjaan Umum dan Penataan Ruang dan Penataan Ruang Provinsi Bengkulu berdasarkan Peraturan Gubernur Bengkulu Nomor 59 Tahun 2016 tentang “Kedudukan, Susunan Organisasi, Tugas dan Fungsi, serta Tata Kerja Dinas Pekerjaan Umum dan Penataan Ruang dan Penataan Ruang Provinsi Bengkulu” mempunyai fungsi :

1. Perumusan kebijakan Teknis di Bidang Pekerjaan Umum dan Penataan Ruang.
2. Pelaksanaan kebijakan teknis di Bidang Pekerjaan Umum dan Penataan Ruang.
3. Pelaksanaan evaluasi dan pelaporan sesuai dengan lingkup tugasnya.
4. Pelaksanaan administrasi Dinas Pekerjaan Umum dan Penataan Ruang dan Penataan Ruang.

3.1.3. Struktur Organisasi

Adapun Struktur Organisasi Dinas Pekerjaan Umum dan Penataan Ruang dan Penataan Ruang Provinsi Bengkulu dapat dilihat pada gambar 3.1 dalam gambar terlampir.



Gambar 3.1 Struktur Organisasi Direktorat Jenderal Sumber Daya Air Bengkulu

3.1.4. Tugas Dan Wewenang

Untuk melaksanakan tugas serta fungsi, Dinas Pekerjaan Umum dan Penataan Ruang Provinsi Bengkulu dibentuk organisasi yang mampu mendorong semua kegiatan dinas. Berkenaan dengan hal tersebut di atas, maka Organisasi Dinas Pekerjaan Umum dan Penataan Ruang dan Penataan Ruang Provinsi Bengkulu terdiri dari :

A. Kepala Dinas

Mempunyai tugas yaitu : Melaksanakan urusan pemerintahan dan tugas pembantuan dibidang pekerjaan umum dan penataan ruang yang menjadi kewenangan daerah provinsi.

B. Sekretariat

Dipimpin oleh Sekretaris yang mempunyai tugas yaitu: Mengarahkan penyusunan program kerja, mengelola urusan keuangan, kepegawaian, kelembagaan, serta menginventarisir permasalahan persuratan, kearsipan, rumah tangga, perlengkapan, dokumentasi, dan informasi sesuai ketentuan peraturan perundang – undangan.

C. Bidang Sumber Daya Air

Dipimpin oleh Kepala Bidang yang mempunyai tugas: Melaksanakan pengelolaan sumber daya air diwilayah sungai yang meliputi perencanaan, pelaksanaan konstruksi, operasi dan pemeliharaan dalam rangka konservasi, dan pendayagunaan sumber daya air dan pengendalian daya rusak air pada sungai, pantai, bendungan, danau, embung, dan tampungan air lainnya, irigasi, rawa, tambak, air tanah, air baku yang menjadi urusan pemerintah daerah provinsi.

D. Bidang Bina Marga

Dipimpin Kepala Bidang yang mempunyai tugas: Melaksanakan penyusunan perencanaan, pemograman, pelaksanaan pembangunan, dan persevasi jalan dan jembatan, pengamanan pemanfaatan bagian – bagian jalan dan penerangan jalan umum, pengendalian mutu, dan hasil pelaksanaan pekerjaan serta penyediaan dan pengujian bahan dan peralatan.

E. Bidang Cipta Karya

Dipimpin oleh Kepala Bidang yang mempunyai tugas, yaitu: Melaksanakan penyelenggaraan infrastruktur permukiman, bangunan gedung, penataan bangunan dan lingkungan pada kawasan strategis provinsi dan pengelolaan dan pengembangan sistem penyediaan air minum dan sistem penyediaan air minum dan sistem drainase lintas daerah serta pengelolaan dan pengembangan sistem air limbah domestic dan persampahan regional.

F. Bidang Jasa Konstruksi

Dipimpin oleh Kepala Bidang yang mempunyai tugas, yaitu: Menyenggarakan pelaksanaan kebijakan di bidang pembinaan jasa konstruksi sesuai dengan ketentuan peraturan – peraturan.

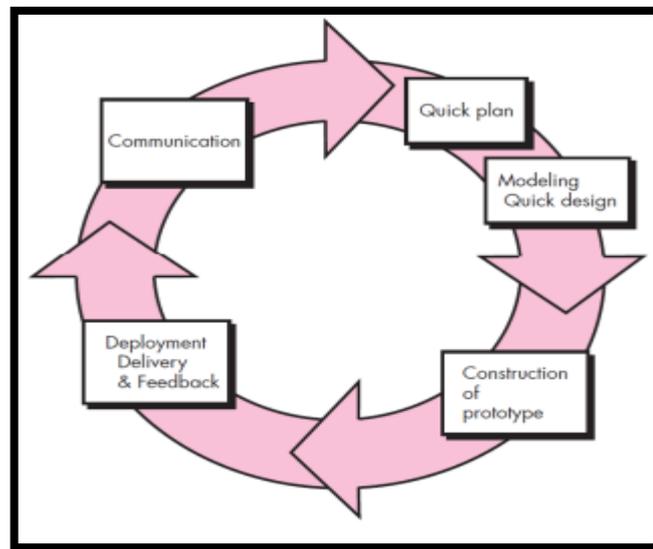
G. Bidang Tata Ruang

Dipimpin oleh Kepala Bidang yang mempunyai tugas, yaitu: Melaksanakan penyiapan perumusan dan pelaksanaan pengaturan, perencanaan, tata ruang, pemanfaatan ruang, dan pembinaan bidang penataan ruang sesuai dengan kewenangan pemerintah provinsi berdasarkan peraturan perundang – undangan yang berlaku.

3.2. Metode Penelitian

Metode Penelitian yang digunakan adalah metode *Prototype*. Sebuah prototype adalah versi awal dari sistem perangkat lunak yang digunakan untuk mendemonstrasikan konsep-konsep, percobaan rancangan, dan

menemukan lebih banyak masalah dan solusi yang memungkinkan. Sistem dengan model prototype memperbolehkan pengguna untuk mengetahui bagaimana sistem berjalan dengan baik. Metode prototyping yang digunakan di dalam penelitian ini bertujuan untuk mendapatkan gambaran aplikasi yang akan dibangun melalui rancangan aplikasi *prototype* terlebih dahulu kemudian akan dievaluasi oleh *user*. Aplikasi *prototype* yang telah dievaluasi oleh *user* selanjutnya akan dijadikan acuan untuk membuat aplikasi yang dijadikan produk akhir sebagai output dari penelitian ini.



Gambar 3.2 Metode *Prototype*

Adapun Tahapan dari Metode *Prototype* yang digunakan adalah sebagai berikut :

1. *Communication*

Pada tahap ini dilakukan dalam pengumpulan data kebutuhan sistem yaitu dengan metode observasi dan wawancara.

2. *Quick Plan and Modelling Quick Design*

Pada tahap ini dilakukan perancangan *prototype* sesuai dengan kebutuhan pengguna, sesuai data diperoleh dari proses identifikasi kebutuhan pengguna. Pembuatan *prototype* dilakukan dengan dua tahapan yaitu perancangan perangkat keras dan perancangan perangkat lunak.

3. *Construction Of Prototype*

Pada tahapan ini dilakukan proses penulisan kode program dan pembuatan *prototype*.

4. *Deployment Delivery and Feedback*

Pada tahapan ini dilakukan pengujian *prototype* yang telah dibuat untuk mengetahui apakah *prototype* telah sesuai dengan kebutuhan yang diharapkan oleh pengguna. Jika *prototype* belum memenuhi kebutuhan pengguna, maka dilakukan kembali identifikasi kebutuhan pengguna. Selanjutnya *prototype* dirancang kembali sesuai dengan saran yang diberikan oleh pengguna. Berikut pengujian yang akan dilakukan pada *prototype*.

3.3. Instrumen Perangkat Keras dan Perangkat Lunak

3.3.1. Perangkat Keras

Perangkat keras merupakan suatu peralatan fisik komputer yang dijalankan untuk menjalankan program. Sistem perangkat keras terdiri dari unit masukan, unit pengelolah dan unit keluaran. Perangkat keras yang diperlukan dalam penelitian ini antara lain.

Tabel 3.1 Spesifikasi Perangkat Keras

No	Alat / Bahan	Spesifikasi	Jumlah
1	Komputer	Minimal Prosesor Gen3	1 unit
2	<i>NodeMCU</i>	Tegangan 3.3V - 5V, Menggunakan jaringan 802.11 b /g /n, <i>Wi-Fi SoC (Single on Chip)</i> , soft-AP, Matikan arus kebocoran <10uA, Memori <i>Flash</i> 4MB, kecepatan pengiriman data <2ms, Konsumsi daya siaga <1.0mW	1 pcs
3	Sensor Ultrasonik	Tegangan 5V, Arus 30mA Typ. 35mA Max. <i>Max Range</i> - 3 m <i>Min Range</i> - 2 cm, 3 pin VCC, <i>Ground</i> , <i>Signal</i>	1 pcs
4	LCD	Tegangan 5 V, pin SDA dan SCL	1 pcs
5	Modul <i>relay</i>	Tegangan 5 V, pin digital	1 pcs
6	<i>Buzzer</i>	Tegangan kerja: 4v-8v DC (optimal 5v), Arus max: 30mA / 5vDC	1 pcs

7	Kabel <i>Jumper</i>	<i>Male & Female</i>	Secukupnya
8	Solder	Berat : 100 Gram Type:DS-40N Watt: 40 Watt	1 unit
9	Timah solder	diameter 0.8 mm / 42 gram	Secukupnya
10	Box	Box plastik	1 pcs
11	Kabel USB	USB 2.0 Fast Transfer Data – Tipey USB: USB Type A Male To Mini USB Type A Male – Panjang: 30 cm	1 pcs
12	Papan PCB	ukuran 10 cm x 20 cm	1 pcs

3.3.2. Perangkat Lunak

Perangkat Lunak merupakan program pendukung yang diperlukan dalam menjalankan perangkat keras. *Software* sebagai penerjemah suatu bahasa mesin (*analog*) yang akhirnya menghasilkan suatu informasi yang dapat dikenal oleh manusia.

Adapun perangkat lunak yang mendukung program ini adalah:

1. Sistem Operasi : *Windows 10 Ultimate*.
2. Aplikasi : IoT (*Internet Of Things*)

Tabel 3.2 Spesifikasi Perangkat Lunak

No	Perangkat Lunak	Spesifikasi
1	<i>Windows</i>	Minimal <i>Windows 10</i>
2	Arduino IDE	Versi 1.8.18
3	Telegram	Notifikasi

4	<i>Thingier.io</i>	<i>User Interface</i>

3.4. Metode Pengumpulan Data

Data penelitian yang di perlukan dalam penyusunan proposal laporan tugas akhir diperoleh melalui metode studi pustaka dan studi laboratorium.

1. Studi Pustaka

Data penelitian pada metode studi pustaka diperoleh dari sumber pustaka yang meliputi buku, majalah, atau arsip mengenai topik yang dibahas dalam penelitian. Data penelitian ini juga diperoleh dari internet.

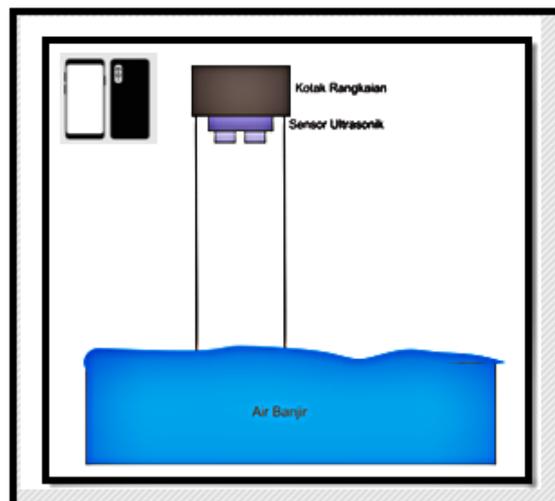
2. Studi Laboratorium

Data penelitian pada metode Studi Laboratorium diperoleh dari melakukan riset ilmiah, eksperimen, pengukuran ataupun pelatihan ilmiah.

3.5. Metode Pengumpulan Data

3.5.1 Diagram Global Alat

Diagram global alat yang digunakan dalam penelitian ini dapat dilihat pada gambar berikut :

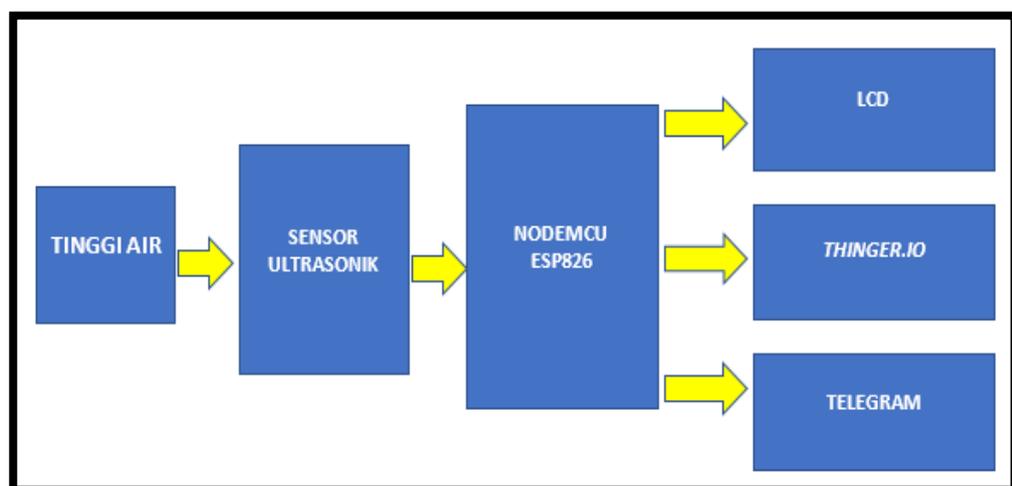


Gambar 3.3 Diagram Global Alat

Sistem pada alat ini menggunakan gelombang ultrasonik yang dipancarkan kemudian diterima kembali pada permukaan air. Setelah diketahui ketinggian air yang diukur maka data ketinggian air akan dikirim ke Nodemcu ESP8266. Data ketinggian air tersebut akan ditampilkan di LCD dan dapat dimonitoring dengan menggunakan *platform Thinger.io* yang telah terhubung oleh sistem. Apabila batas ketinggian air mencapai titik maksimal, sistem akan melakukan pengiriman notifikasi ke Telegram sesuai *user id* yang telah ditentukan oleh sistem untuk membunyikan *buzzer*.

3.5.2 Blok Diagram

Berikut desain blok diagram deteksi banjir dengan menggunakan *platform Thinger.io* dan notifikasi Telegram.



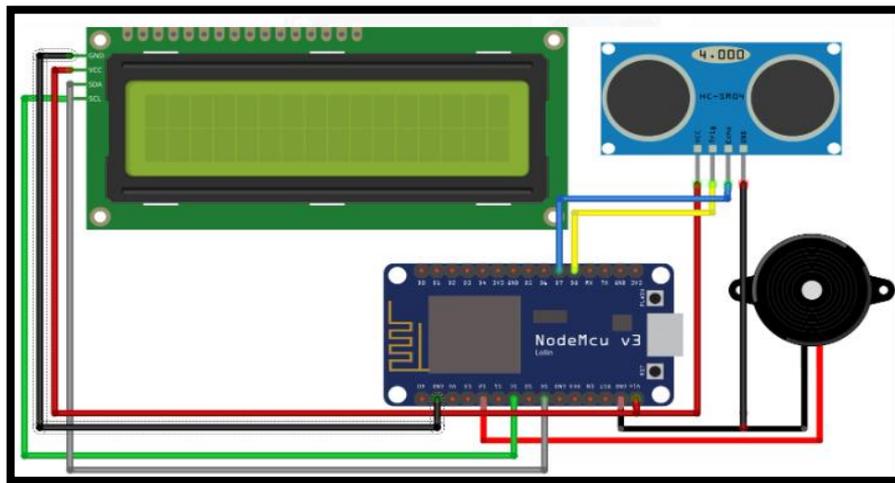
Gambar 3.4 Blok Diagram

Pada saat sistem dihidupkan, Nodemcu ESP8266 akan mendeteksi ketinggian air menggunakan sensor ultrasonik, setelah didapat berapa ketinggian air, maka ketinggian air akan ditampilkan pada LCD dan dapat dimonitoring melalui web yang telah terhubung oleh sistem. Dan sistem

juga akan mengirimkan SMS ke no yang telah terdaftar oleh sistem dan mengirimkan sms ke sistem penerima untuk menghidupkan sirine.

3.5.3 Rancangan Rangkaian Alat

Gambar dibawah ini merupakan gambar dari diagram rangkaian *prototype* deteksi banjir menggunakan *platform* Thinger.io dan notifikasi Telegram.

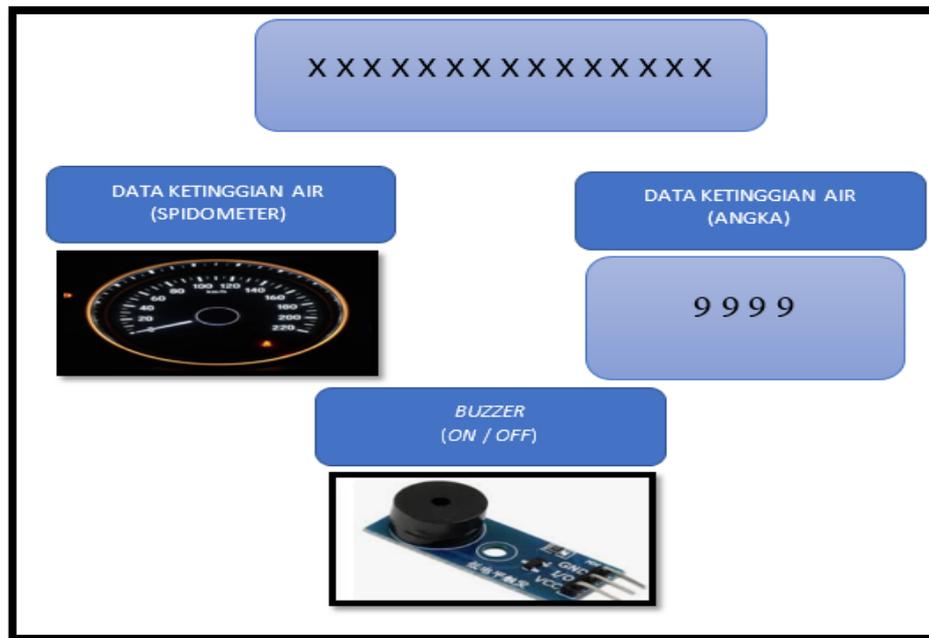


Gambar 3.5 Rancangan Rangkaian Alat

Pada gambar 3.5 yang merupakan rancangan rangkaian alat dapat terlihat Nodemcu ESP8266 akan dihubungkan dengan sensor ultrasonik, LCD dan *buzzer*.

3.5.4 Rancangan *User Interface* Sistem

Platform Thinger.io akan digunakan sebagai *user interface* yang dapat menampilkan data *monitoring* ketinggian level air secara *real time* dari hasil pembacaan data melalui sensor ultrasonik HCSR04.



Gambar 3.6 Rancangan *User Interface* Sistem

Gambar 3.6 merupakan rancangan *user interface* Sistem, adapun tampilan *user interface* yang dirancang terdiri dari judul penelitian, data ketinggian air dalam bentuk *spidometer*, data ketinggian air dalam bentuk angka. Selain itu pada *user interface* akan menampilkan kondisi *buzzer* dalam kondisi *on/off*.

Keterangan :

a. Data ketinggian air dalam bentuk *spidometer*

Tujuan *user interface* menampilkan data ketinggian air dalam bentuk *spidometer* adalah dari sisi Visualisasi yang Mudah Dipahami karena *spidometer* memberikan representasi visual yang intuitif dan mudah dipahami oleh pengguna.

b. Data ketinggian air dalam bentuk angka

Tujuan *user interface* menampilkan data ketinggian air dalam bentuk angka adalah memberikan Akurasi Tinggi, di mana

tampilan tersebut dapat memberikan nilai yang tepat dan akurat, sehingga pengguna dapat mengetahui informasi secara *detail*.

c. Kondisi *Buzzer (On / Off)*

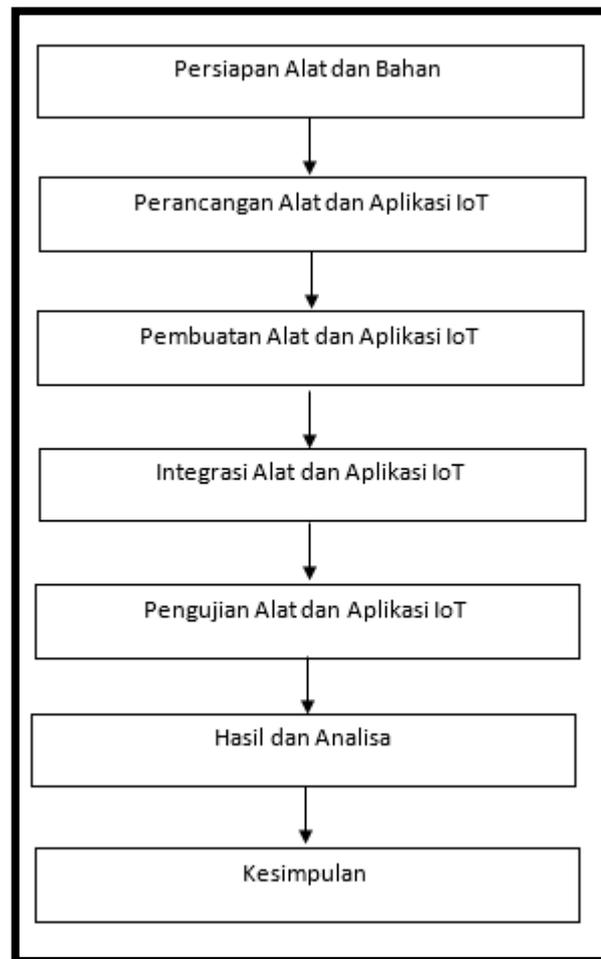
Tujuan *user interface* menampilkan kondisi *Buzzer (On / Off)* adalah untuk mengetahui kondisi *Buzzer* yang aktif.

3.5.5 Prinsip Kerja Alat

Sistem pada alat ini menggunakan gelombang ultrasonik yang dipancarkan kemudian diterima kembali pada permukaan air. Setelah diketahui ketinggian air yang diukur maka data ketinggian air akan dikirim ke Nodemcu ESP8266. Data ketinggian air tersebut akan ditampilkan di LCD dan dapat dimonitoring dengan menggunakan *platform Thinger.io* yang telah terhubung oleh sistem. Apabila batas ketinggian air mencapai titik maksimal, sistem akan melakukan pengiriman notifikasi ke Telegram sesuai *user id* yang telah ditentukan oleh sistem untuk membunyikan *buzzer*.

3.5.6 Rencana Kerja Alat

Rencana kerja dalam melakukan penelitian ini adalah sebagai berikut:



Gambar 3.7 Rencana Kerja Alat

Keterangan :

- a. Persiapan Alat dan Bahan
 1. Penulis mencari serta mengumpulkan komponen-komponen yang akan dipakai dalam rangkaian alat penelitian
 2. Penulis mengunduh *software* arduino untuk memprogram alat.

b. Perancangan Alat dan Aplikasi IoT

Penulis melakukan perancangan dengan referensi dari buku, internet, dan bertanya secara langsung kepada sumber yang dianggap biasa membantu dalam penulisan ini.

c. Pembuatan Alat dan Aplikasi IoT

1. Penulis mulai merakit komponen-komponen yang penulis kumpulkan untuk dapat menjadi satu alat yang sesuai dengan penelitian.

2. Penulis mulai membuat kode program di *software* arduino.

d. Integrasi Alat dan Aplikasi IoT

Setelah alat dan kode program sudah dibuat, maka penulis mulai memasukan kode program ke alat menggunakan kabel mikro USB dan komputer.

e. Pengujian Alat dan Aplikasi IoT

Penulis melakukan pengujian alat dengan mencoba langsung dan melihat serta menganalisis hasilnya sebagai hasil penelitian.

f. Hasil dan Analisa

Setelah pengujian dilakukan maka akan diperoleh hasil dan analisa sistem.

g. Kesimpulan

Setelah seluruh proses dilakukan maka penulis dapat menyimpulkan hasil dari pengujian *prototype* deteksi banjir dengan web server dan notifikasi telegram bahwa apakah alat tersebut sudah berjalan

dengan fungsinya dan jika belum sesuai dengan rencana kerja maka penulis kembali melakukan pengecekan rancangan tersebut.

3.5.7 Perancangan Pengujian

Perancangan pengujian dilakukan setelah merancang dan membangun *prototype* deteksi banjir dengan web server dan notifikasi telegram yang dibuat telah selesai. Adapun beberapa hal yang akan diuji pada *prototype* deteksi banjir dengan web server dan notifikasi telegram adalah sebagai berikut :

Tabel 3.3 Perancangan Pengujian

No	Instrumen Penguji	Hasil	Ket
1	Kemampuan sensor ultrasonik dalam membaca ketinggian air.		
2	Kemampuan Nodemcu ESP8266 menerima data dari sensor Ultrasonik		
3	Pengujian <i>Buzzer</i>		
4	Pengujian Notifikasi Telegram		
5	Pengujian <i>Thingier.io</i>		
6	Pengujian Keseluruhan		

BAB IV

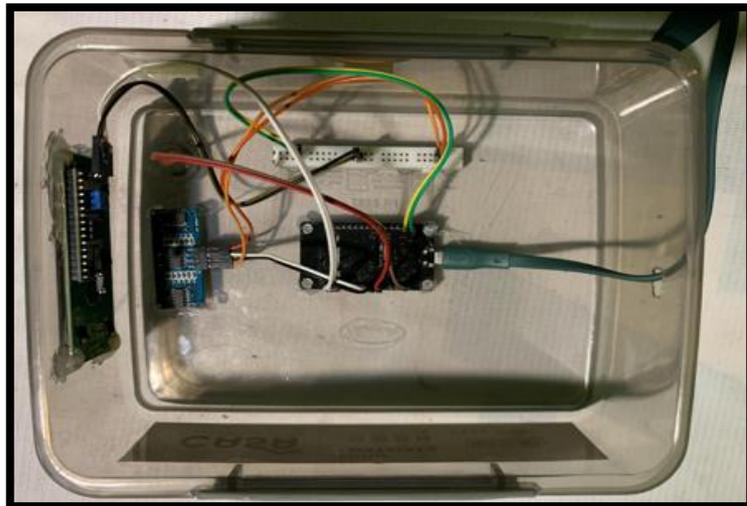
HASIL DAN PEMBAHASAN

4.1 Hasil

Pada bagian ini akan dibahas mengenai proses serta hasil penelitian dari “*Prototype* Deteksi Banjir Menggunakan *Platform Thinger.io* dan Notifikasi Telegram”.

4.1.1 Hasil *Prototype* Deteksi Banjir

Gambar 4.1 merupakan hasil dari *prototype* deteksi banjir menggunakan *Platform Thinger.io* dan notifikasi telegram.



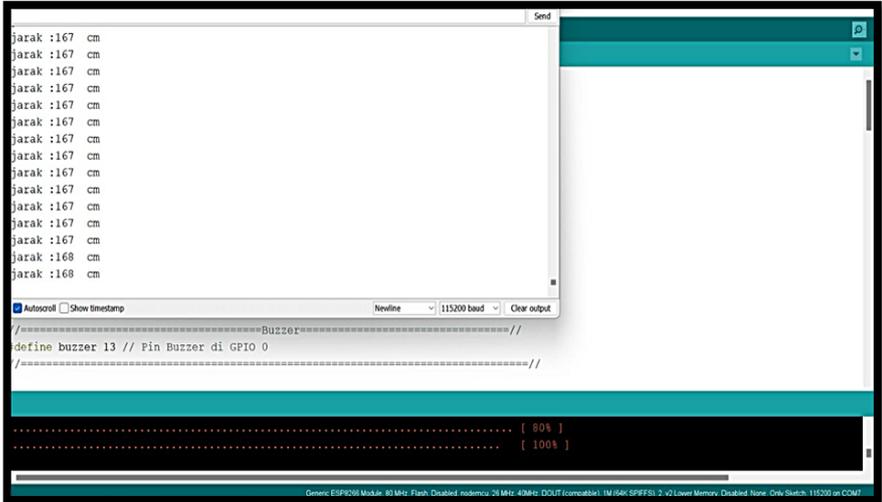
Gambar 4.1 Rangkaian Perangkat Keras

Pada gambar 4.1 dapat terlihat bahwa rangkaian alat utama yang digunakan pada sistem “*Prototype* deteksi banjir menggunakan *Platform Thinger.io* dan notifikasi telegram” terdiri dari NodeMCU ESP-8266, sensor ultrasonik, *buzzer* dan LED. Pengambilan data dapat dilakukan setelah semua komponen sistem “*Prototype* Deteksi

Banjir Menggunakan *Platform thinger.io* dan notifikasi telegram” telah terpasang.

A. Kemampuan Sensor Ultrasonik

Kriteria pengujian yang pertama memiliki tujuan untuk mengetahui kemampuan sensor ultrasonik HC-SR04 dalam mengukur ketinggian air banjir. Hasil yang diperoleh adalah sensor ultrasonik HC-SR04 telah mampu membaca data ketinggian air banjir. Ketinggian air banjir yang dibaca sensor ultrasonik pada kondisi aman, waspada dan bahaya dengan ketinggian 0–72cm, 73–128 cm, dan 129-168cm. Dapat dilihat pada serial monitor.



```

jarak :167 cm
jarak :168 cm
jarak :168 cm

Autoscroll [ ] Show timestamp
Newline [ ] 115200 baud [ ] Clear output

//=====Buzzer=====//
define buzzer 13 // Pin Buzzer di GPIO 0
//=====//

[ 89% ]
[ 100% ]

Generic ESP8266 Module 80 MHz Flash Disabled nodemcu 28 MHz 4MB: D0/JT (compatible) 1M (04) SPFFS 2 v2 Lower Memory Disabled None 0x0 Sketch: 115200 on COM7

```

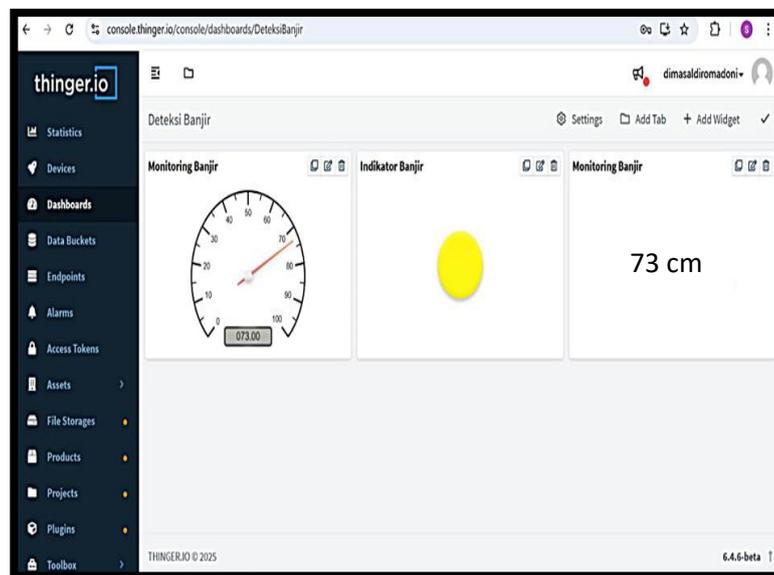
Gambar 4.2 Tampilan Sensor Ultrasonik Pada Serial Monitor

Gambar 4.2 merupakan tampilan dari sensor ultrasonik pada saat membaca data ketinggian air banjir. Hal ini membuktikan kalau kemampuan sensor ultrasonik dalam

membaca data sistem *monitoring* dan pendeteksi ketinggian air banjir secara *real time* berbasis IoT.

B. Kemampuan Nodemcu ESP-8266

Kriteria pengujian kedua yaitu kemampuan Nodemcu ESP-8266 dalam menerima dan mengirim data dari sensor ultrasonik. Berdasarkan pengujian telah diperoleh data bahwa Nodemcu ESP-8266 telah berhasil menerima dan mengirim data dari sensor ultrasonik berupa data ketinggian air banjir.



Gambar 4.3 Tampilan Data Sensor Pada *User Interface*

Gambar 4.3 merupakan hasil tampilan pada saat nodemcu ESP-8266 membaca data ketinggian air banjir dari sensor ultrasonik, kemudian Nodemcu ESP-8266 mengirimkan data ketinggian air banjir dalam bentuk *Spidometer* dan angka, serta tampilan warna lampu LED kuning pada *Thingr.io*.

C. Kemampuan *Buzzer*

Pengujian *buzzer* dilakukan dengan bertujuan untuk mengetahui respon pada saat air banjir dalam kondisi tinggi atau bahaya.

```
int buzzer = 2; // Pin GPIO 16 untuk LED Buzzer
void setup() {
  pinMode(2, OUTPUT);
}

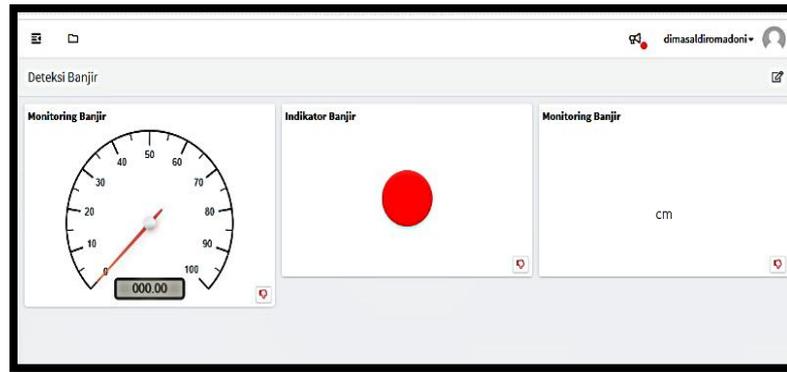
void loop() {
  digitalWrite(2, HIGH); // untuk menhidupkan LED Buzzer
  delay(1000);           // menunggu 1 detik
  digitalWrite(2, LOW); // untuk mematikan LED Buzzer
  delay(1000);          // menunggu 1 detik
}
```

Gambar 4.4 *List Program Buzzer*

Gambar 4.4 merupakan *coding* program yang digunakan pada saat pengujian *buzzer*. Pada prakteknya *buzzer* akan digunakan sebagai sumber alarm yang menghasilkan bunyi pada saat air banjir pada ketinggian level bahaya.

D. Kemampuan Thinger.io

Kriteria pengujian keempat adalah mengenai kemampuan IoT dalam melakukan *monitoring* ketinggian deteksi air banjir jarak jauh secara *realtime*. Hasil yang didapatkan bahwa melalui pengujian tersebut diperoleh hasil bahwa IoT memiliki kemampuan dalam melakukan *monitoring* ketinggian deteksi air banjir dari jarak jauh secara *realtime*.

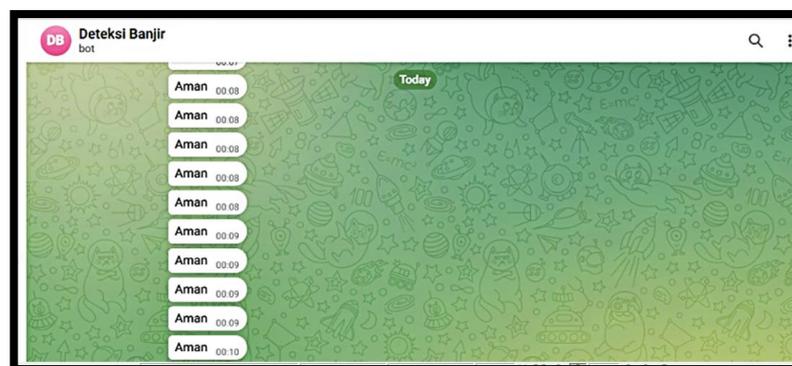


Gambar 4.5 Tampilan Awal *User Interface*

Gambar 4.5 adalah tampilan awal *user interface* pada platform Thingier.io. Hal ini membuktikan bahwa sistem *monitoring* ketinggian dan kekeruhan air tanggul secara *realtime* berbasis IoT dengan jarak jauh. Berdasarkan tampilan yang ada dapat terlihat data ketinggian dan kekeruhan air tanggul secara *realtime* telah bekerja dengan baik.

E. Kemampuan Notifikasi Telegram

Adapun kriteria pengujian kelima yaitu pengujian kemampuan telegram dalam menerima pesan. Pengujian dilakukan dengan melihat keberhasilan telegram dalam mengirimkan notifikasi ke pengguna.



Gambar 4.6 Tampilan Notifikasi Telegram Status Aman

Gambar 4.6 merupakan tampilan notifikasi Telegram status ketinggian banjir dalam kondisi aman. Kondisi aman pada ketinggian 0-72cm.

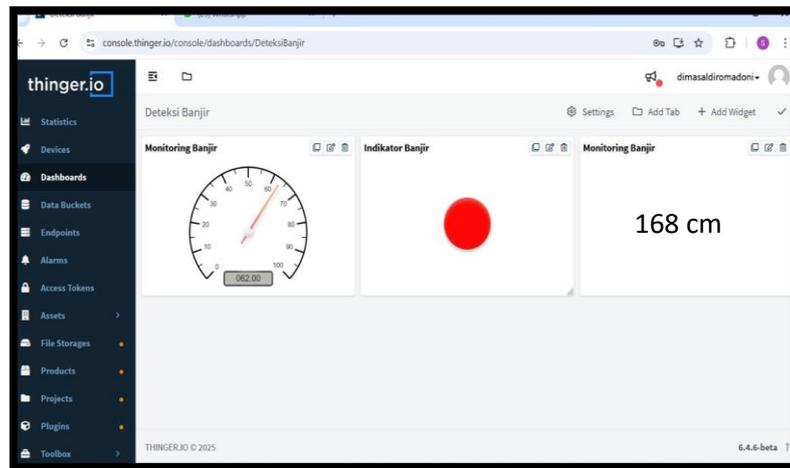


Gambar 4.7 Tampilan Notifikasi Telegram Status Waspada & Bahaya

Selanjutnya gambar 4.7 merupakan tampilan notifikasi Telegram status ketinggian banjir dalam kondisi waspada dan bahaya. Kondisi waspada dan bahaya untuk ketinggian air banjir 73-127cm waspasda dan bahaya 128-168cm.

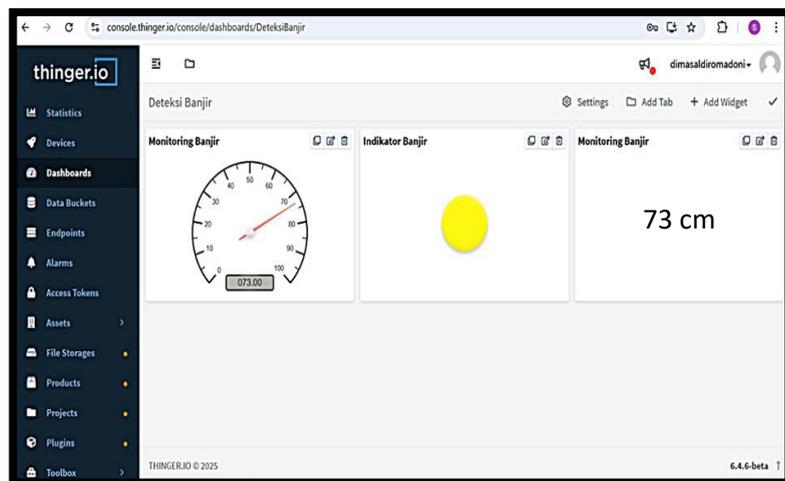
F. Kemampuan Indikator

Kriteria pengujian keenam yaitu pengujian tampilan lampu LED yang digunakan sebagai indikator sesuai dengan warna lampu yang digunakan sebagai level ketinggian air banjir. Warna lampu LED yang digunakan pada tampilan adalah warna merah dan hijau.



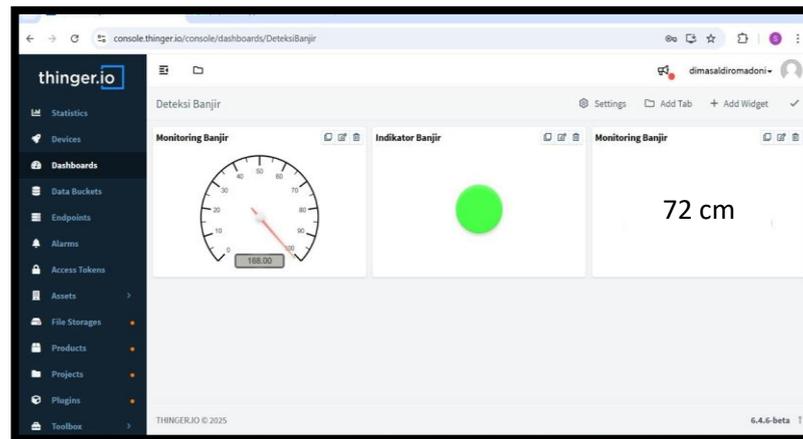
Gambar 4.8 Tampilan Indikator LED Merah

Gambar 4.8 merupakan tampilan indikator LED warna merah, *range* indikator LED merah dengan ketinggian 128-168cm. LED warna merah menandakan pada level bahaya.



Gambar 4.9 Tampilan Indikator LED Kuning

Gambar 4.9 merupakan tampilan indikator LED warna kuning, *range* indikator LED kuning dengan ketinggian 73-128cm. LED warna kuning menandakan pada level waspada.



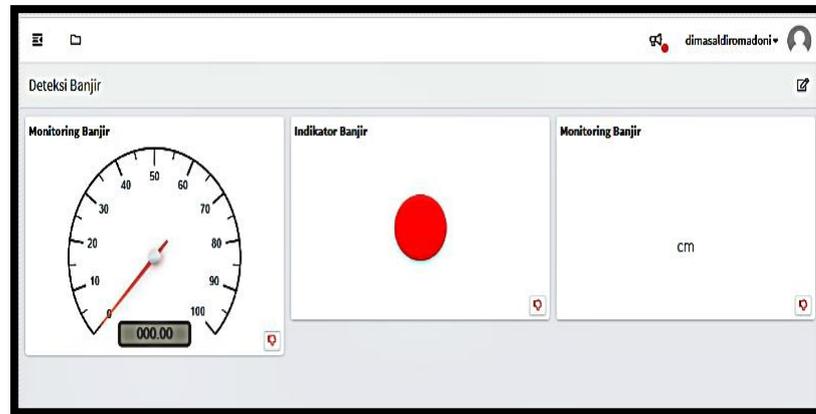
Gambar 4.10 Tampilan Indikator LED Hijau

Gambar 4.10 merupakan tampilan indikator LED warna hijau, *range* indikator LED hijau dengan ketinggian 0-72cm. LED warna hijau menandakan pada level aman. Kemudian untuk mengetahui sistem deteksi banjir menggunakan *platform thinger.io* dan notifikasi telegram telah bekerja sesuai dengan perencanaan, maka akan dilakukan pengujian sistem secara keseluruhan. Pengujian sistem secara keseluruhan terdiri dari dua kondisi diantaranya adalah pengambilan data kategori pengujian tanpa halangan serta pengambilan data kategori pengujian menggunakan halangan.

4.1.2 Hasil Aplikasi IoT dan Notifikasi Telegram

Kemudian setelah semua komponen *prototype* deteksi banjir menggunakan *Platform Thinger.io* dan Notifikasi Telegram selesai terpasang pada sebuah wadah, maka alat akan saling dihubungkan. Adapun *platform* IoT yang dipakai pada sistem deteksi banjir adalah

Thinger.io. Gambar 4.2 merupakan hasil tampilan awal status *dashboard* dari sistem IoT yang menggunakan *Platform thinger.io*.



Gambar 4.11 Tampilan *User Interface* Deteksi Banjir Kondisi Awal

4.2 Pembahasan

Sebelum masuk pada langkah pengujian sistem deteksi banjir menggunakan *Platform Thinger.io* dan Notifikasi Telegram langkah yang dilakukan ialah mempersiapkan alat dan bahan terlebih dahulu yang nantinya akan dipakai dalam pembuatan rangkaian sistem.

4.2.1 Alat dan Bahan

Rangkaian sistem deteksi banjir menggunakan *Platform Thinger.io* dan notifikasi telegram dilakukan dengan cara menghubungkan masing-masing perangkat keras sesuai dengan desain yang telah rancang sebelumnya. Pada sistem deteksi banjir menggunakan *Platform Thinger.io* dan Notifikasi Telegram menggunakan perangkat keras diantaranya sebagai berikut:

A. Alat Yang Digunakan Pada Sistem

1. Obeng digunakan untuk memasang mur dan baut pada rangkaian.
2. Multimeter digunakan untuk mengukur tegangan listrik pada komponen.
3. Lem lilin digunakan sebagai perekat komponen pada box kaca

B. Bahan Yang Digunakan Pada Sistem

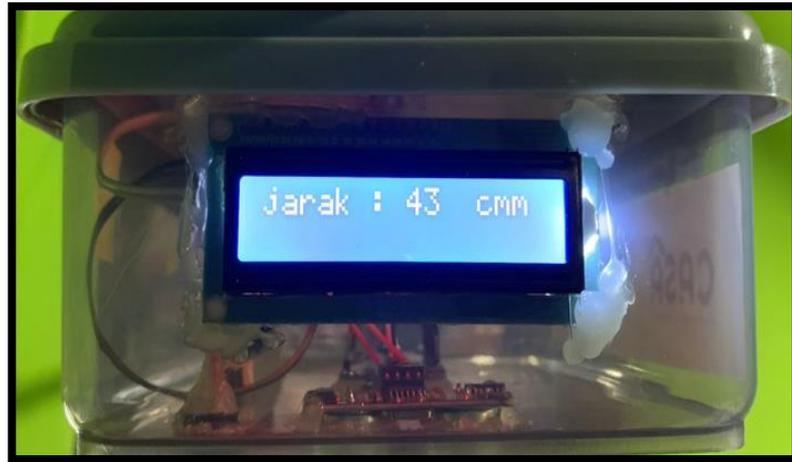
1. Nodemcu ESP-8266 digunakan sebagai otak dalam pengolahan dan pemrosesan data.
2. Kabel USB digunakan untuk *upload* program dari laptop ke dalam nodemcu ESP-8266.
3. Sensor ultrasonik digunakan untuk membaca data ketinggian air tanggul
4. LCD digunakan untuk menampilkan data sensor yang terbaca.
5. *Buzzer* digunakan untuk membunyikan *alarm* level ketinggian air banjir.
6. Relai digunakan sebagai sakelar pemutus arus pada *buzzer*.
7. Kabel *jumper* digunakan untuk menghubungkan komponen pada rangkaian.
8. Komputer dan Android berfungsi sebagai media yang digunakan dalam *monitoring* tampilan pada *user interface*.
9. Box Plastik digunakan sebagai wadah menampung air

4.2.2 Perancangan, Pembuatan Alat dan Aplikasi IoT

A. Perancangan dan Pembuatan Alat

Selanjutnya tahapan yang dilakukan setelah mempersiapkan alat dan bahan yang diperlukan adalah proses merangkai sistem deteksi banjir menggunakan *platform thinger.io* dan notifikasi telegram. Adapun tata caranya adalah dengan menghubungkan masing-masing pin komponen menggunakan kabel jumper. Adapun pin yang digunakan pada LCD adalah pin 4 dan 5. Pin yang digunakan pada sensor ultrasonik adalah pin 14 dan 12. Selanjutnya pin yang digunakan pada *buzzer* adalah pin 13. Setelah komponen saling terhubung satu sama lain maka langkah selanjutnya adalah menghubungkan rangkaian ke laptop dengan menggunakan kabel USB untuk proses *upload* program. Kemudian setelah proses *upload* program selesai, rangkaian akan diletakan atau dihubungkan langsung ke box yang telah dipersiapkan. Berikut adalah langkah-langkah menghubungkan rangkai alat.

1. Sengsor ultrasonik di hubungkan di Nodemcu ESP8266.
2. Nodemcu ESP8266 mengirimka data ketinggian air ke LCD.
3. LCD lalu memonitoring dengan menggunakan *platfrom thinger.io*
4. Telegram akan menerima pengiriman notifikasi sesuai *user id*.
5. *Buzzer* akan membunyikan alaram ketika tanda bahaya.



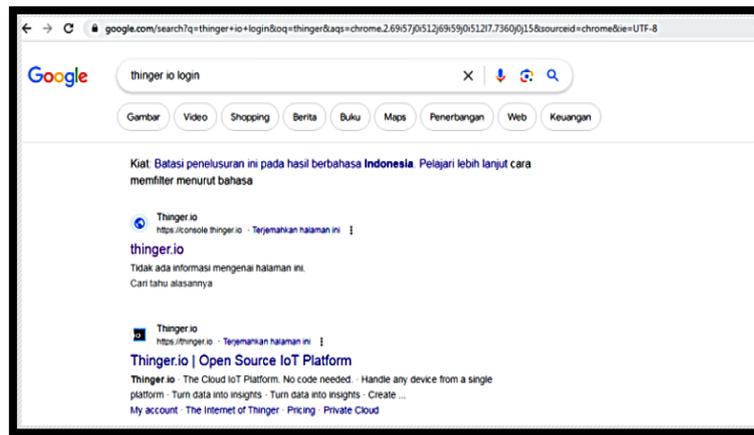
Gambar 4.12 Perancangan dan Pembuatan Alat

B. Perancangan dan Pembuatan Aplikasi IoT

Selanjutnya akan dilakukan perancangan aplikasi IoT untuk sistem deteksi banjir menggunakan *Platform Thinger.io* dan Notifikasi Telegram. *Platform Thinger.io* akan digunakan sebagai *interface* yang dapat menampilkan data ketinggian air banjir. *User interface* yang dirancang akan menampilkan tiga data. Pertama data ketinggian air banjir dalam bentuk *spidometer*, kedua data ketinggian air banjir dalam bentuk angka, serta indikator lampu LED yang ditandai dengan warna merah, kuning atau hijau.

1. Pembuatan Akun *Thinger.io*

Sebelum membuat akun *Thinger.io* Langkah yang pertama dilakukan adalah masuk ke Google *chrome* dan ketika *key word Thinger.io*.



Gambar 4.13 Tampilan Halaman Pencarian *Thingier.io*

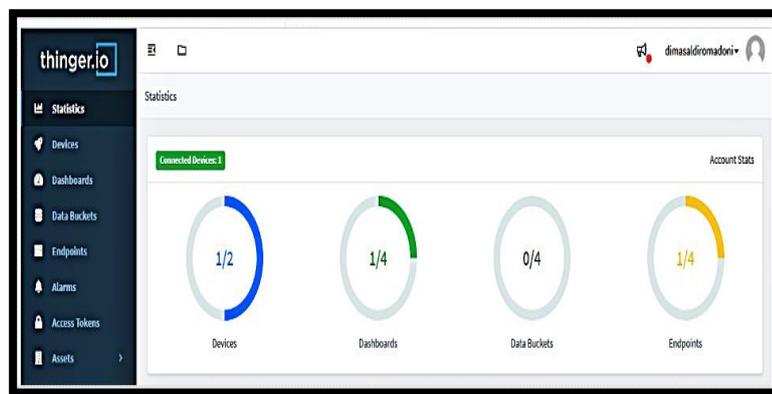
Berdasarkan gambar 4.13 dapat dilihat merupakan hasil pencarian halaman *Thingier.io* pada *Google Chrome*, yang pertama ketikkan *Thingier.io login* pada halaman pencarian, kemudian selanjutnya klik halaman yang paling atas Untuk masuk tampilan halaman pembuatan akun *Thingier.io*

Langkah selanjutnya adalah membuat akun *Thingier.io* terlebih dahulu, lalu melakukan verifikasi *email* agar akun *Thingier.io* dapat digunakan.

Gambar 4.14 Tampilan *Form Thingier.io*

Setelah *log in*, pengguna sudah dapat masuk ke tampilan awal *thingier.io, statistic* merupakan tampilan awal saat *log in*.

Selanjutnya klik menu *device*, dan pilih *add device*. *Device* merupakan laman yang menampilkan nama perangkat yang terkoneksi atau memiliki akses dengan akun *thinger.io* yang digunakan saat itu juga. Jika perangkat sudah terdaftar dan sedang dalam keadaan *online*, maka pada kolom *state* akan berwarna hijau dengan tulisan *connected*. Sementara saat *offline* akan tertulis *disconnected*.



Gambar 4.15 Tampilan *Device Connected*

Selanjutnya isi data *Device* meliputi *Device Id*, *Device Description*, *Device credential* dengan klik *generate random*. kemudian klik *add device*

The screenshot shows the 'Devices > Add' form in the Thinger.io interface. It includes the following fields and options:

- Device Credentials:** A text input field with a 'Random' button.
- Device Information:**
 - Device Name:** A text input field with the label 'Optional device name'.
 - Device Description:** A text input field with the label 'Optional device description'.
- Advanced Options:**
 - Asset Type:** A dropdown menu with 'Select Type...' as the current selection.
 - Asset Group:** A dropdown menu with 'Select Group...' as the current selection.
 - Product:** A dropdown menu with 'Select Product...' as the current selection.
 - Enabled:** A toggle switch that is currently turned on (green).

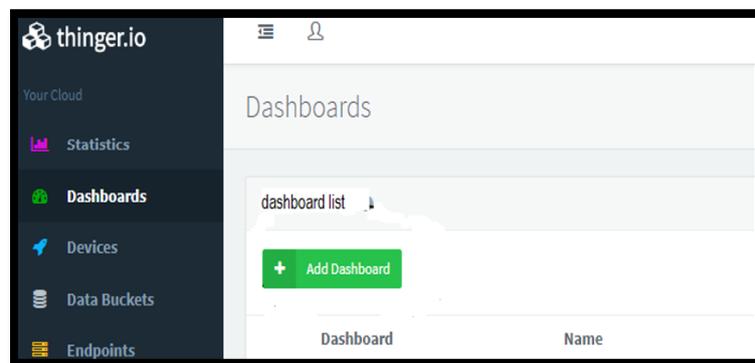
 The sidebar menu on the left is the same as in the previous screenshot.

Gambar 4.16 Tampilan *Add Devices*

2. Pembuatan *Dashboard* pada *Platform Thinger.io*

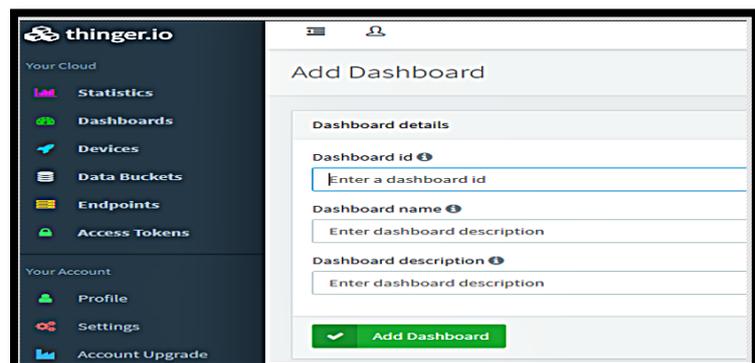
Dashboards merupakan tampilan *user interface* yang digunakan oleh pengguna untuk menampilkan informasi dalam berbagai bentuk grafik maupun angka. Tampilan pada *dashboards* dapat diatur sesuai kebutuhan dengan cara sebagai berikut :

Pertama-tama, klik menu *Dashboard*, lalu pilih *Add Dashboard*.



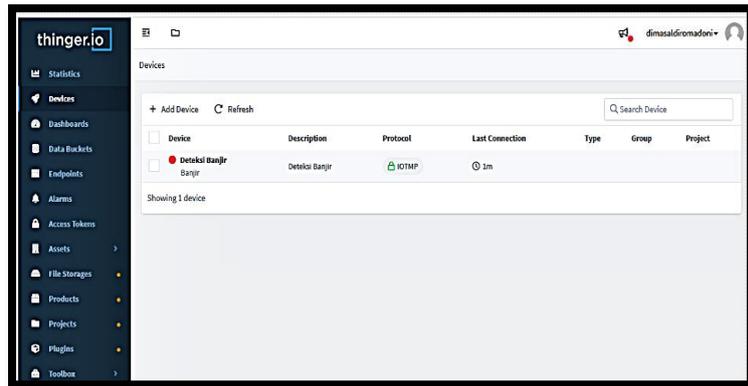
Gambar 4.17 Tampilan *Dashboard*

Kemudian isi *Dashboard* lalu klik *Add Dashboard*



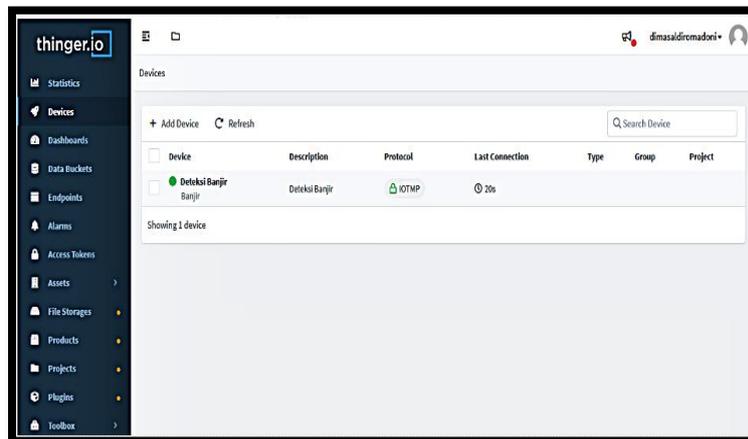
Gambar 4.18 Tampilan *Add Dashboard*

Lalu klik nama *Dashboard* yang telah dibuat.



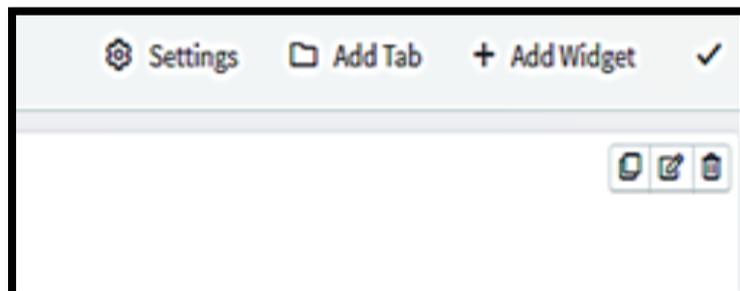
Gambar 4.19 Tampilan *Dashboard Off*

Gambar 4.19 dapat dilihat sebagai hasil nama yang telah di buat pada *dashboard*.



Gambar 4.20 Tampilan *Dashboard On*

Selanjutnya klik tombol pojok kanan atas untuk mengedit tampilan *Dasboard*.



Gambar 4.21 Tampilan Edit *Dashboard*

Kemudian tahapan yang dilakukan adalah klik *Add Widget* dan isi *Widget Setting* nya. Setelah data yang diperlukan selesai diisi maka klik *save* untuk menyimpan tampilan yang telah didesain.

The screenshot shows the 'Widget Settings' dialog box with the 'Widget' tab selected. The 'Tachometer' widget type is chosen. The settings are as follows:

- Title:** Monitoring Banjir
- Subtitle:** Widget Subtitle
- Link To:** Select Dashboard...
- Show Update:** Off
- Show Offline:** None
- Show Fullscreen:** Off
- Background:** #ffffff
- Type:** Tachometer

Gambar 4.22 Tampilan *Widget Setting Spidometer*

Pada gambar 4.22 dapat terlihat bahwa merupakan tampilan *widget setting* data ketinggian air banjir dengan bentuk *spidometer*.

The screenshot shows the 'Widget Settings' dialog box with the 'Widget' tab selected. The 'Text/Value' widget type is chosen. The settings are as follows:

- Title:** Monitoring Banjir
- Subtitle:** Widget Subtitle
- Link To:** Select Dashboard...
- Show Update:** Off
- Show Offline:** None
- Show Fullscreen:** Off
- Background:** #ffffff
- Type:** Text/Value

Gambar 4.23 Tampilan *Widget Setting Angka*

Pada gambar 4.23 dapat terlihat bahwa merupakan tampilan *widget* setting data ketinggian air banjir dengan bentuk *spidometer*.

The screenshot shows the 'Widget Settings' dialog box with the 'Led Indicator' tab selected. The settings are as follows:

- Title:** Indikator Banjir
- Subtitle:** Widget Subtitle
- Link To:** Select Dashboard...
- Show Update:** Off
- Show Offline:** None
- Show Fullscreen:** Off
- Background:** #ffffff
- Type:** Led Indicator

Gambar 4.24 Tampilan *Widget Setting* LED

Pada gambar 4.24 dapat terlihat bahwa merupakan tampilan *widget* setting data ketinggian air banjir dengan bentuk *spidometer*.

The screenshot shows the 'Widget Settings' dialog box with the 'Display Options' tab selected. The settings are as follows:

- Led Size:** 75px
- Color:** #00aa00
- Color Swatches:**
 - 0 (Green) - 72 - [On]
 - 73 (Yellow) - 128 - [On]
 - 129 (Red) - 169 - [On]

Gambar 4.25 Tampilan *Widget Setting* Led Indikator

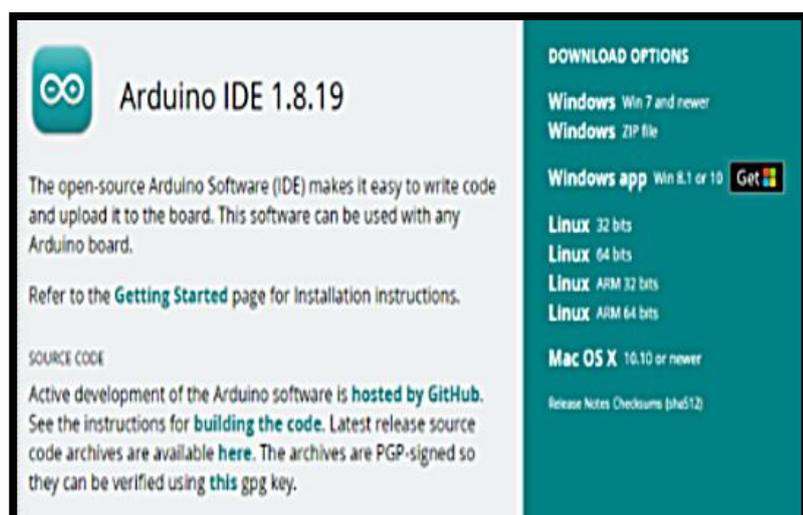
Berdasarkan gambar 4.25 merupakan tampilan *Widget Setting* Led Indikator yang bewarna merah, kuning serta hijau yang menandakan tiga kondisi ketinggian air.

4.2.3 Integrasi Alat dan Aplikasi IoT

A. Penginstalan Aplikasi Arduino IDE

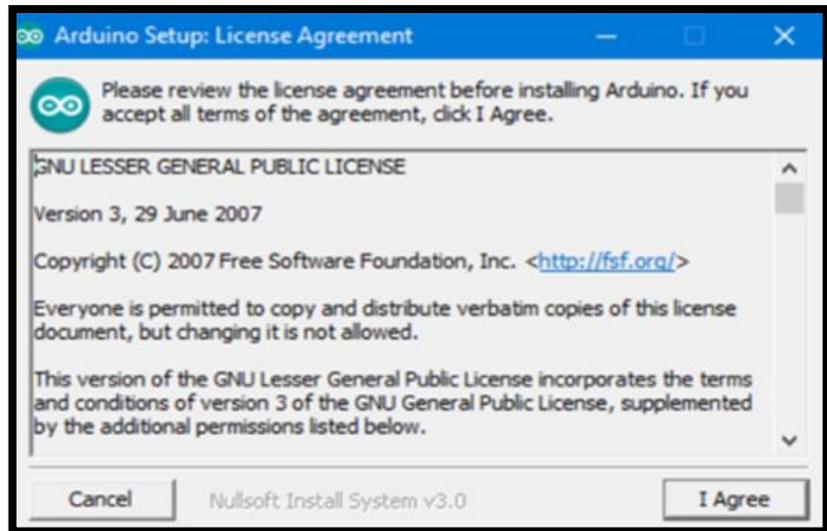
Penginstalan aplikasi Arduino IDE digunakan sebagai tempat untuk dilakukannya pembuatan *listing* program yang digunakan pada sistem deteksi banjir menggunakan *Platform Thinger.io* dan notifikasi telegram. Berikut merupakan tata cara penginstalan Arduino IDE adalah:

1. Hal pertama yang dilakukan adalah menginstal aplikasi Arduino Uno IDE dengan cara *download* situs <https://www.arduino.cc/en/software> lalu pilih *windows installer*. Maka akan tampil pihan *download* seperti pada gambar 4.18 di bawah ini.



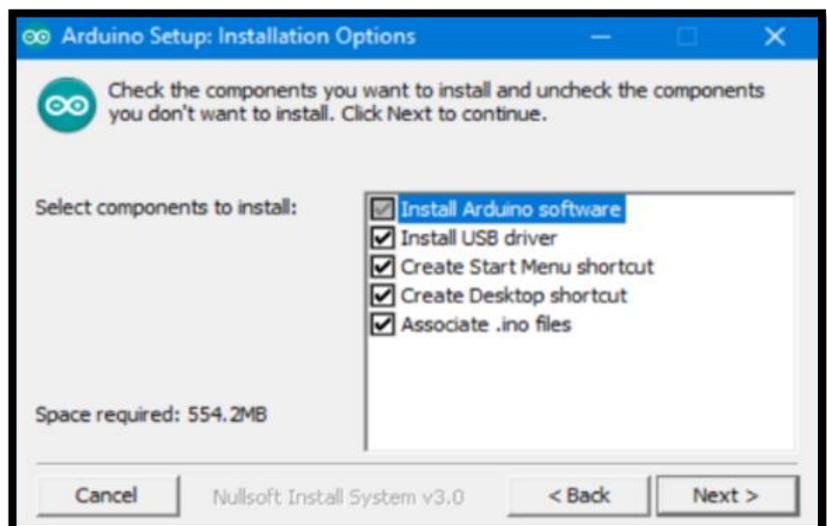
Gambar 4.26 *Download* Aplikasi Arduino IDE

2. Kemudian melakukan persetujuan untuk instalasi *software* Arduino IDE, dengan mengklik tombol “*I Agree*”.



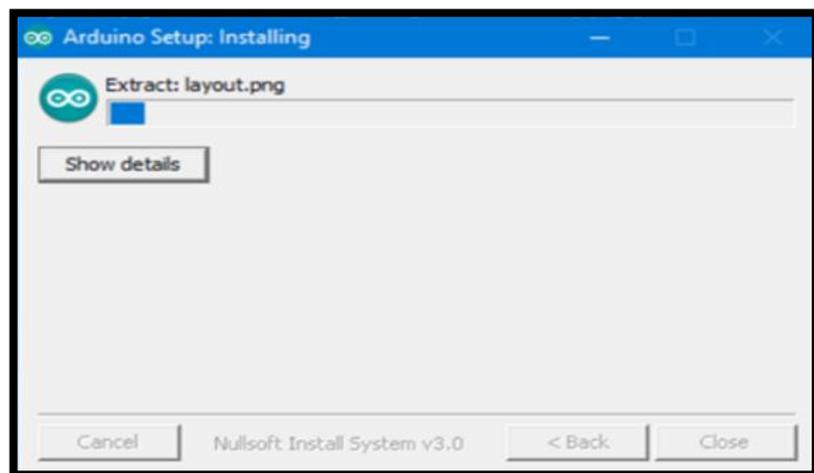
Gambar 4.27 Tampilan Persetujuan *Instalasi* Arduino

3. Untuk pilihan opsi *instalation option* dengan memilih semua komponen yang akan ditampilkan dilakukan dengan memilih semua (tanda centang) pada opsi *instalation option*.



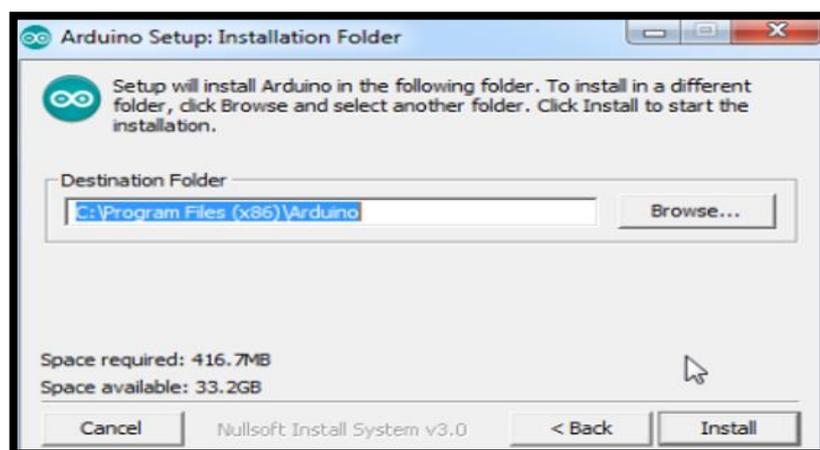
Gambar 4.28 Tampilan Persetujuan *Instalasi* Arduino

4. Tahapan selanjutnya adalah melakukan *instalation folder* yang akan digunakan untuk memilih *folder* tempat penyimpanan program Arduino dengan cara klik tombol *install* dalam memulai proses *instalasi software*.



Gambar 4.29 Tampilan Memilih *Folder*

5. Lalu dilakukan proses *extract* program ke *windows* untuk memulai proses *instalasi*.



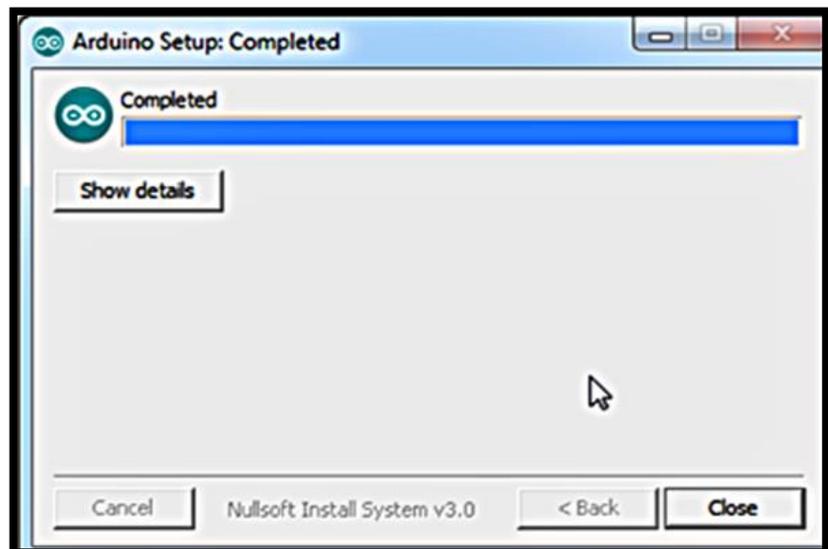
Gambar 4.30 Tampilan *Extrack* ke *Windows*

6. Selanjutnya dilakukan *install usb driver*, ditandai dengan adanya tampilan pilihan untuk *install driver*. Proses ini digunakan untuk mengenali dan melakukan komunikasi dengan *board*.



Gambar 4.31 Tampilan *Install USB Driver*

7. Kemudian yang terakhir adalah tampilan ketika Arduino telah selesai di *install*.



Gambar 4.32 Tampilan Selesai *Install*

B. Upload Program Arduino IDE ke NodeMCU ESP-8266

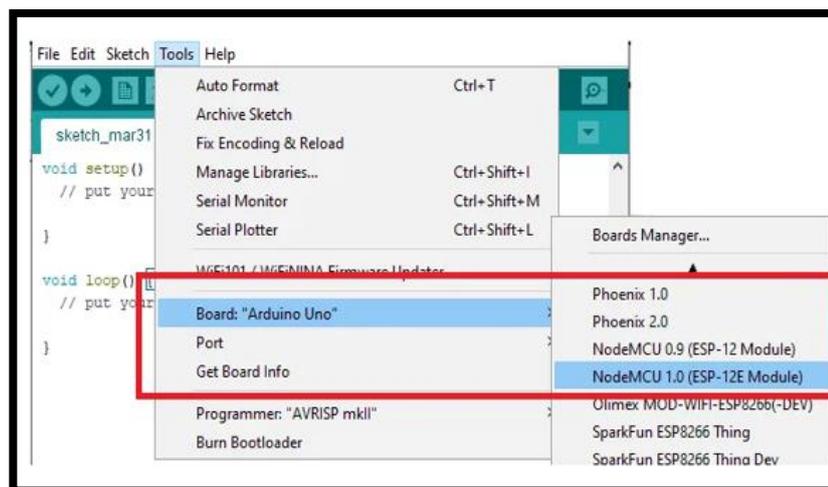
Pada penelitian deteksi banjir menggunakan *Platform Thinger.io* dan notifikasi Telegram. Adapun langkah-langkah yang digunakan untuk *upload* program Arduino IDE ke NodeMCU ESP-8266 adalah sebagai berikut :

1. Hal yang pertama dilakukan pengguna adalah membuka aplikasi program Arduino IDE.



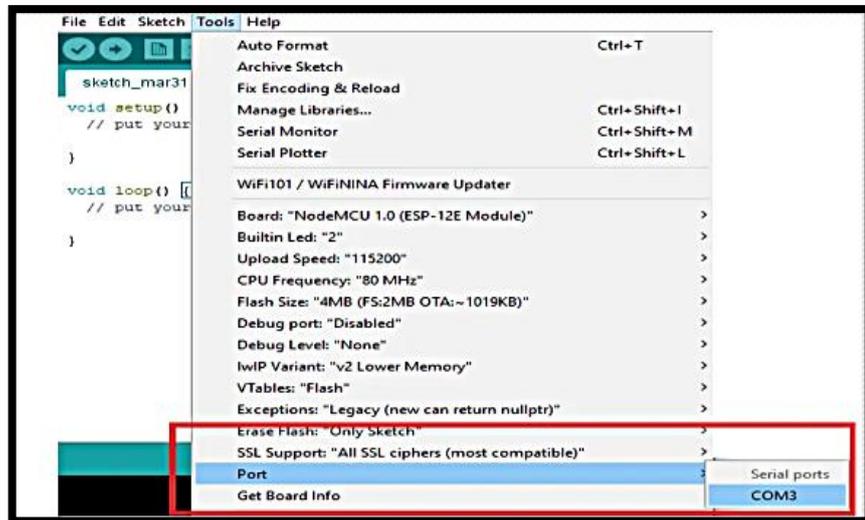
Gambar 4.33 Tampilan Buka Arduino IDE

2. Kemudian buka menu *Tool* → *Board* → *Boards Manager* → pilih



Gambar 4.34 Tampilan Buka Arduino IDE

3. Selanjutnya pengguna harus memilih Port, buka menu **Tools > Port > COM3** (*Sesuaikan dengan COM pada Device Manager*).



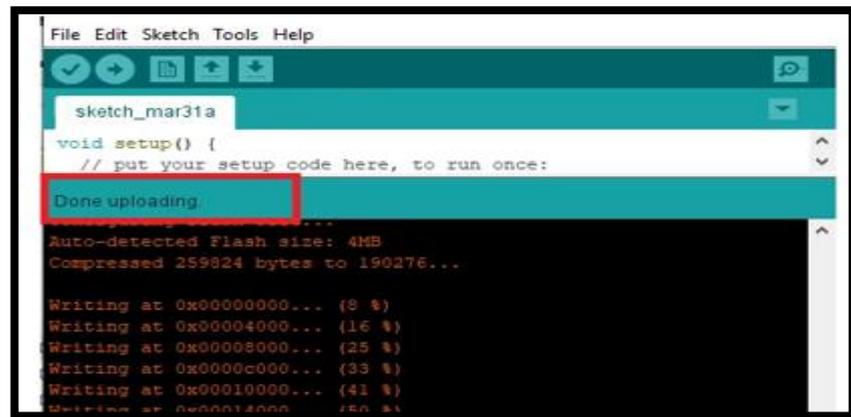
Gambar 4.35 Tampilan Menu *Port*

6. Selanjutnya ketikkan *listing* program yang akan digunakan.

```
//
Serial.println("Memulai Telegram Bot. Koneksi ke Wifi");
myBot.wifiConnect(ssid,pass);
myBot.setTelegramToken(token);
if(myBot.testConnection())
{ Serial.println("Koneksi Berhasil");}
else
{ Serial.println("Koneksi Gagal");}
myBot.sendMessage(id, "notification test");
Serial.println("Pesan Terkirim");
//
  lcd.setCursor(0,0);
  lcd.print("Deteksi Banjir ");
  delay(4000);
  // Penambahan Wifi
  thing.add_wifi(SSID, SSID_PASSWORD);
  thing["jarak"] >> outputValue(jarak);// Variabel Output Jarak
```

Gambar 4.36 Tampilan *Listing* Program

7. Setelah itu klik *verify* dan *upload*, tunggu sampai proses *upload* selesai (terdaypat tulisan *done uploading*).

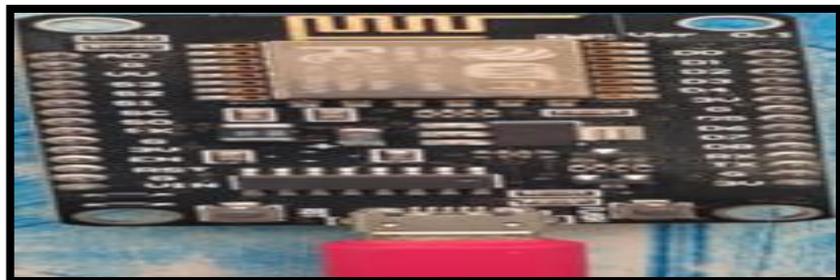


Gambar 4.37 Tampilan *Done Uploading*

4.2.4 Uji coba

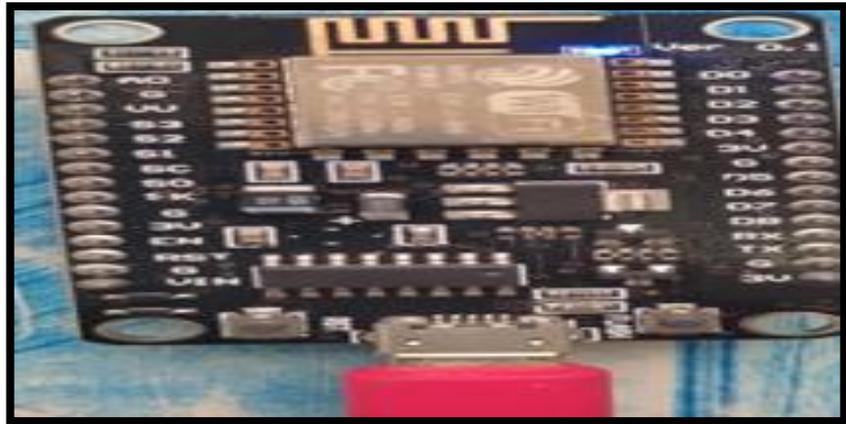
1. Pengujian NodeMCU ESP-8266

Pengujian NodeMCU ESP8266 ini bertujuan untuk mengetahui kemampuan NodeMCU ESP-8266 dalam mengirim dan menerima data sehingga dapat diproses oleh Nodemcu ESP8266.



4.38 Tampilan NodeMCU ESP8266 *Off*

Pada gambar 4.38 dapat terlihat kondisi NodeMCU ESP8266 dalam keadaan *off*, di mana hal tersebut ditandai dengan lampu indikator yang ada di Nodemcu ESP8266 dalam keadaan mati.



4.39 Tampilan NodeMCU ESP8266 *On*

Pada gambar 4.39 dapat terlihat kondisi NodeMCU ESP8266 dalam keadaan *on*, di mana hal tersebut ditandai dengan lampu indikator yang ada di Nodemcu ESP8266 dalam keadaan menyala.

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

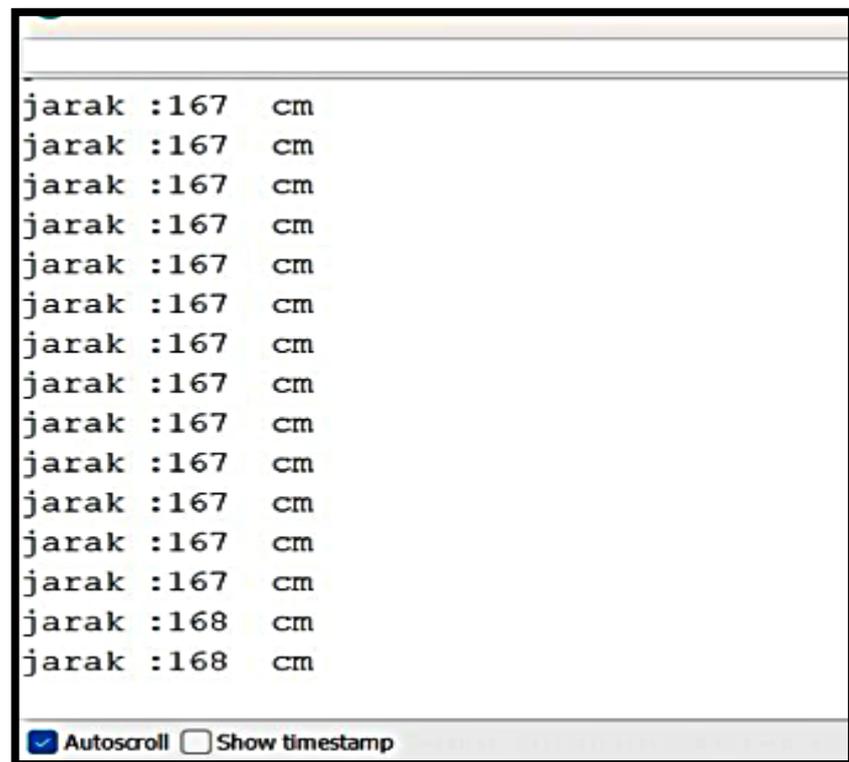
// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000); // wait for a second
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
  delay(1000); // wait for a second
}
```

Gambar 4.40 Tampilan Kode Program NodeMCU

Selanjutnya berdasarkan gambar 4.40 di atas merupakan *listing* program yang digunakan dalam pengujian Nodemcu ESP8266.

2. Pengujian Sensor Ultrasonik

Pengujian pembacaan sensor ultrasonik dengan nodemcu memiliki tujuan yaitu agar diperoleh tingkat akurasi dari modul sensor ultrasonik. Uji coba dimulai dengan mengukur ketelitian sensor ultrasonik dalam membaca pengukuran parameter berupa ketinggian ombak. Kemudian pada serial monitor yang ada pada Arduino IDE akan dapat terlihat hasil dari pembacaan data sensor ultrasonik. Namun sebelumnya dibutuhkan data pembandingan dalam pengujian sensor ultrasonik, yaitu dengan cara mengukur jarak sensor ultrasonik terhadap benda.



```
jarak :167 cm
jarak :168 cm
jarak :168 cm
```

Autoscroll Show timestamp

Gambar 4.41 Jarak Sensor Ultrasonik Pada Serial Monitor

Selanjutnya gambar 4.41 adalah program yang digunakan dalam pengujian pembacaan sensor ultrasonik untuk pengujian sistem

deteksi banjir menggunakan *Platform Thinger.io* dan notifikasi Telegram

3. Pengujian LCD

Gambar 4.35 merupakan hasil tampilan awal LCD, di mana tampilan tersebut adalah kondisi awal pada saat sistem deteksi banjir menggunakan *Platform Thinger.io* dan notifikasi Telegram dilakukan pengujian.



Gambar 4.42 Hasil Tampilan Awal LCD

Selanjutnya pada gambar 4.42 merupakan hasil tampilan LCD dengan hasil pengujian awal. Pengujian awal LCD dilakukan dengan menampilkan kata “ Hello, world ! Tampilan LCD ini yang nanti nya akan digunakan untuk *monitoring* air banjir menggunakan *Platform Thinger.io* dan notifikasi Telegram.



Gambar 4.43 Hasil Tampilan Ketinggian 43 Cm

Pada gambar 4.43 merupakan hasil tampilan LCD dengan hasil pengujian ketinggian air banjir 43 cm. Tampilan LCD ini yang nantinya akan digunakan untuk *monitoring* air banjir menggunakan *Platform Thinger.io* dan notifikasi Telegram.

```
//Library LCD I2C
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27,20,4);

void setup()
{
  lcd.init();
  lcd.init();
  lcd.backlight();
  lcd.setCursor(2,0);
  lcd.print("Hello, world!");
  lcd.setCursor(4,1);
  lcd.print("Test LCD");
}
void loop() {
}
```

Gambar 4.44 Hasil Tampilan Program Pengujian LCD

Selanjutnya gambar 4.44 adalah program yang digunakan dalam pengujian LCD untuk pengujian sistem deteksi banjir menggunakan *Platform Thinger.io* dan notifikasi Telegram.

4. Pengujian *Buzzer*

Pengujian *buzzer* dilakukan dengan bertujuan untuk mengetahui respon pada saat air banjir dalam kondisi tinggi atau bahaya.

```
int buzzer = 2; // Pin GPIO 16 untuk LED Buzzer
void setup() {
  pinMode(2, OUTPUT);
}

void loop() {
  digitalWrite(2, HIGH); // untuk menhidupkan LED Buzzer
  delay(1000);           // menunggu 1 detik
  digitalWrite(2, LOW);  // untuk mematikan LED Buzzer
  delay(1000);           // menunggu 1 detik
}
```

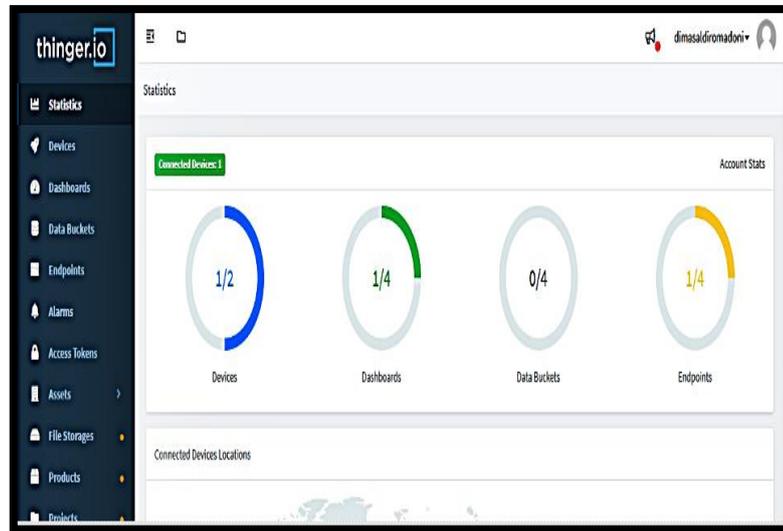
Gambar 4.45 *List Program Buzzer*

Gambar 4.45 merupakan *coding* program yang digunakan pada saat pengujian *buzzer*. Pada prakteknya *buzzer* akan digunakan sebagai sumber alarm yang menghasilkan bunyi pada saat air banjir pada ketinggian level bahaya.

5. Pengujian Kemampuan *Platform Thinger.io* Dalam *Monitoring Sistem*.

Platform yang akan digunakan pada sistem deteksi banjir menggunakan *Platform Thinger.io* dan notifikasi Telegram. Tujuan dari pengujian kemampuan *platform Thinger.io* adalah untuk dapat

menampilkan data yang telah dikirim dari nodeMCU. Tampilan data yang dihasilkan adalah berupa *user interface* secara *realtime*.



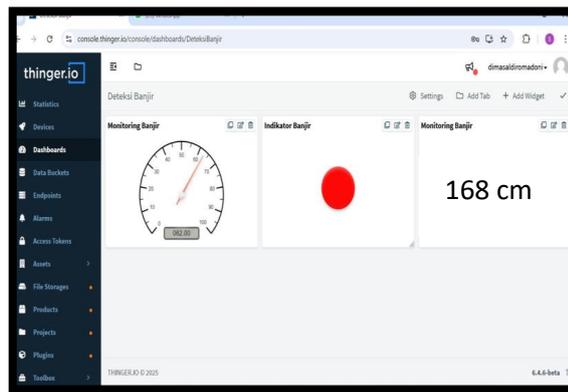
Gambar 4.46 Tampilan Status *Device* Sistem

Gambar 4.46 merupakan tampilan status *device* sistem deteksi banjir menggunakan *Platform Thinger.io* dan notifikasi Telegram. Pada status *device* terlihat bahwa status *device* Thinger.io sistem dalam keadaan terkoneksi dengan internet.

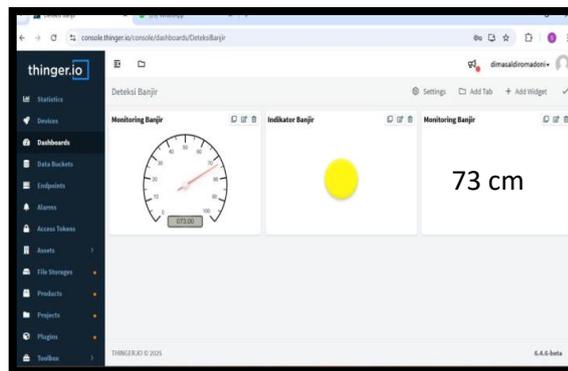
6. Pengambilan Data Tiga Kondisi Ketinggian Air Bendungan Tanjung Agung

Hasil pengujian sistem “*Prototype* Deteksi Banjir Menggunakan *Platform Thinger.io* dan notifikasi Telegram dilakukan melalui tiga kondisi air yaitu tinggi, sedang dan rendah. Pengujian pertama dilakukan pada ketinggian air dengan skala kondisi tinggi, pengujian ke dua untuk skala kondisi sedang dan pengujian ke tiga untuk skala kondisi rendah. Selanjutnya tiga kondisi ketinggian air akan ditandai dengan lampu indikator warna merah, kuning dan hijau sesuai dengan

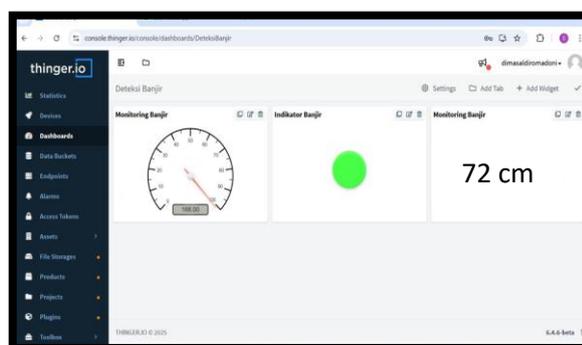
skala yang digunakan. Data deteksi akan dapat di *monitoring* secara *real-time* menggunakan *Platform Thinger.io* dan notifikasi pada Telegram. Berikut adalah gamabar indikator lampu warna merah,kuning dan hijau.



Gambar 4.47 Tampilan Indikator lampu warna merah



Gambar 4.48 Tampilan indikator lampu warna kuning



Gambar 4.49 Tampilan indikator lampu warna hijau

Selanjutnya untuk penjelasan lengkap mengenai data ketinggian air banjir sesuai dengan indikator lampu yang digunakan dapat dilihat pada tabel 4.1 berikut.

Tabel 4. 1 Data Indikator Lampu, Notifikasi Telegram & Level Air Banjir Beserta Skala Pada *Prototype*

No	Indikator Lampu	Notifikasi Telegram	Level Air Banjir	Skala <i>Prototype</i>
1	Merah	Bahaya	5-7cm	128-168cm
2	Kuning	Waspada	3-4cm	73-127cm
3	Hijau	Aman	0-2cm	0-72cm

Berdasarkan pada tabel 4.1 yang merupakan data indikator lampu, Notifikasi Telegram & level air beserta skala digunakan pada *prototype*. Berdasarkan tabel diketahui bahwa untuk lampu indikator merah ketinggian air adalah antara 128–168cm. Selanjutnya untuk indikator lampu kuning ketinggian air adalah antara 73–127cm. Lalu untuk lampu indikator hijau ketinggian air adalah antara 0-72cm.

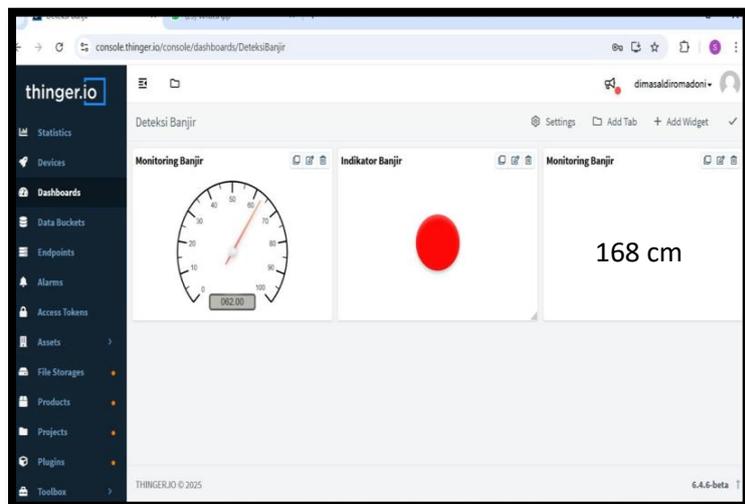
a. Kategori Ketinggian Air Bendungan Tanjung Agung Tinggi

Pengambilan data yang pertama akan dilakukan untuk kategori tinggi yaitu pada saat ketinggian air banjir pada nilai 128-168cm. Lampu indikator yang akan tampil pada *user interface* adalah warna merah. Pengambilan data ini dilakukan untuk mengetahui bahwa sistem yang dirancang untuk deteksi

banjir menggunakan *Platform Thinger.io* dan notifikasi Telegram telah berjalan dengan baik.



Gambar 4.50 Pengambilan Data Kategori Ketinggian Air Tinggi



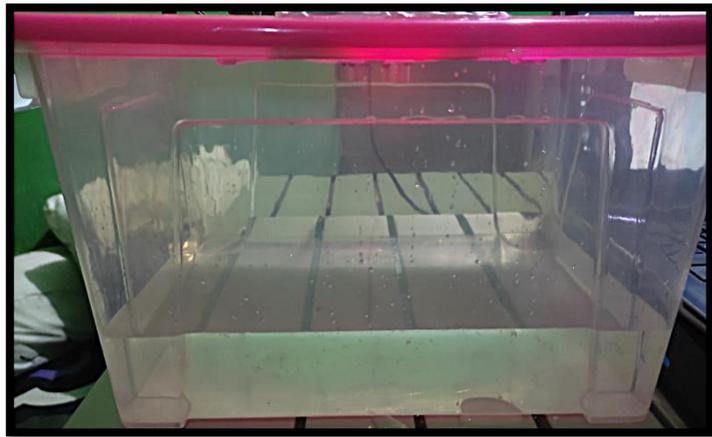
Gambar 4.51 Tampilan *User Interface* Ketinggian Air Tinggi

Gambar 4.51 merupakan gambar pada saat pengambilan data kategori ketinggian air kondisi tinggi. Pada *User Interface* sistem deteksi terbaca data ketinggian air tinggi adalah 128-168cm.

Selain itu terlihat bahwa lampu indikator berwarna merah yang menandakan bahwa ketinggian banjir pada level tinggi.

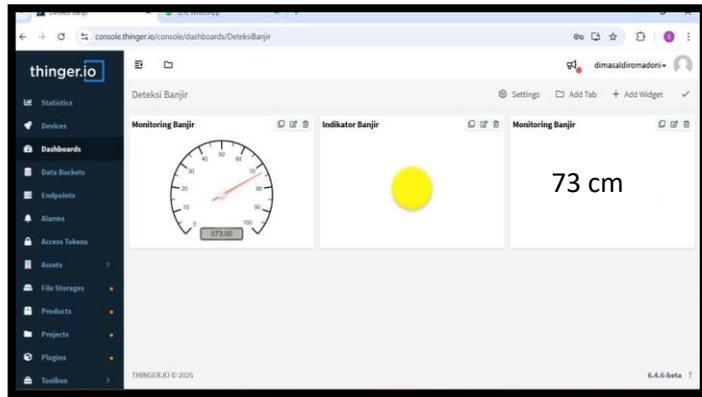
b. Kategori Ketinggian Air Bendungan Tanjung Agung Sedang

Selanjutnya pengambilan data yang kedua akan dilakukan untuk kategori sedang yaitu pada saat ketinggian air pada nilai 73-127cm. Lampu indikator yang akan tampil pada *user interface* adalah warna kuning. Pengambilan data ini dilakukan untuk mengetahui bahwa sistem yang dirancang untuk deteksi banjir menggunakan *Platform Thinger.io* dan notifikasi Telegram telah berjalan dengan baik.



Gambar 4.52 Pengambilan Data Kategori Ketinggian Air

Sedang



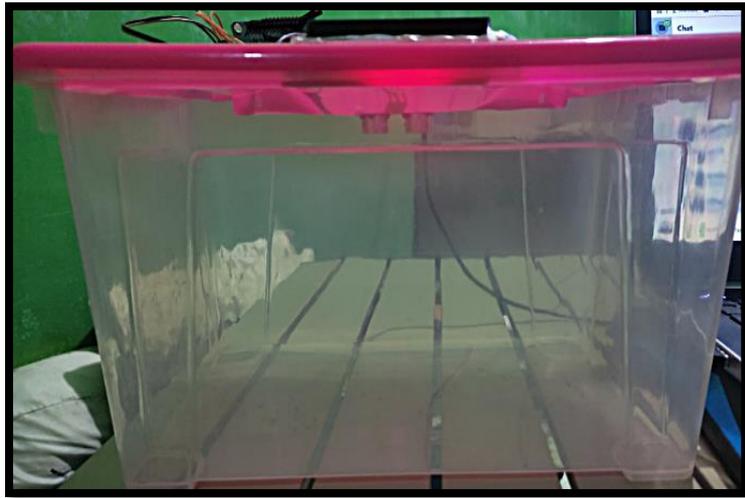
Gambar 4.53 Tampilan *User Interface* Ketinggian Air

Sedang

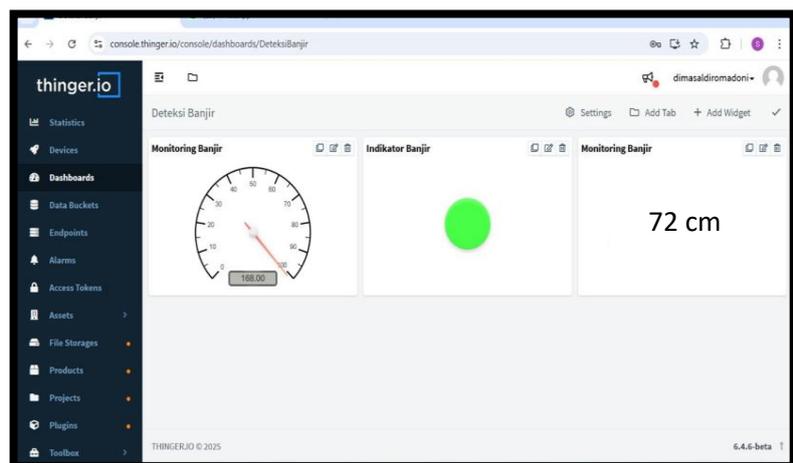
Gambar 4.53 merupakan gambar pada saat pengambilan data kategori ketinggian air pada kondisi sedang. Pada *User Interface* sistem deteksi terbaca data ketinggian air sedang adalah 73-127cm. Selain itu terlihat bahwa lampu indikator berwarna kuning yang menandakan bahwa ketinggian banjir pada level sedang.

c. Kategori Ketinggian Air Bendungan Tanjung Rendah

Selanjutnya pengambilan data yang ketiga akan dilakukan untuk kategori rendah yaitu pada saat ketinggian air pada nilai 0-72cm. Lampu indikator yang akan tampil pada *user interface* adalah warna hijau. Pengambilan data ini dilakukan untuk mengetahui bahwa sistem yang dirancang untuk deteksi banjir menggunakan *Platform Thinger.io* dan notifikasi Telegram telah berjalan dengan baik .



Gambar 4.54 Pengambilan Data Kategori Ketinggian Air Rendah



Gambar 4.55 Tampilan *User Interface* Ketinggian Air Rendah

Gambar 4.55 merupakan gambar pada saat pengambilan data kategori ketinggian air pada kondisi rendah. Pada *User Interface* sistem deteksi terbaca data ketinggian air rendah adalah 0-72cm. Selain itu terlihat bahwa lampu indikator berwarna hijau yang menandakan bahwa ketinggian banjir pada level rendah.

4.2.5 Hasil & Analisa

Selanjutnya berdasarkan hasil pengujian sistem deteksi banjir menggunakan *Platform Thinger.io* dan notifikasi Telegram yang telah dilakukan dapat diketahui bahwa telah memenuhi kriteria pengujian keseluruhan sistem dapat dilihat pada tabel berikut:

Tabel 4.2 Hasil Kriteria Pengujian

No	Instrumen Pengujian	Hasil	Ket
1	Kemampuan sensor ultrasonik dalam membaca ketinggian air.	Sensor ultrasonik dapat membaca ketinggian air.	Berhasil membaca data ketinggian air kondisi (Bahaya, Waspada, Aman)
2	Kemampuan Nodemcu ESP8266 menerima data dari sensor Ultrasonik	NodeMCU telah mampu menerima dan mengirimkan data yang digunakan untuk diolah dan ditampilkan di Thinger.io	Berhasil menerima dan mengirimkan data sensor ultrasonik.
3	Pengujian <i>Buzzer</i>	<i>Buzzer</i> berbunyi pada saat air berada	Berhasil menyala yang menandakan air kondisi bahaya pada ketinggian

		pada kondisi bahaya	128-168
4	Pengujian Notifikasi Telegram	Telegram telah mampu menerima notifikasi mengenai kondisi ketinggian air (bahaya, waspada dan aman)	Berhasil mengirimkan pesan berupa status (Aman, Waspada, Bahaya).
5	Pengujian <i>Thinger.io</i>	<i>Thinger.io</i> berfungsi dengan baik dalam menampilkan data nilai ketinggian air dalam bentuk spidometer dan angka, serta mampu menampilkan	Berhasil menampilkan data dalam 3 kondisi (Aman,Waspada,Bahaya)

		lampu indikator.	
6	Pengujian Keseluruhan	Pengujian keseluruhan sudah sesuai dengan sistem yang dirancang	Sistem telah beroperasi sesuai dengan perancangan yang telah dibuat.

Dari tabel 4.2 yang merupakan hasil pengujian sistem keseluruhan dapat disimpulkan bahwa perancangan sistem deteksi banjir menggunakan *Platform Thinger.io* dan notifikasi Telegram telah berhasil dilakukan karena sistem telah mampu melakukan *monitoring* ketinggian air berdasarkan tiga kondisi berbeda. Tiga kondisi tersebut yaitu pada ketinggian rendah, sedang dan tinggi atau dalam kondisi aman, waspada dan bahaya.

BAB V

KESIMPULAN DAN SARAN

9.1 Kesimpulan

Berdasarkan hasil penelitian mengenai *prototype* deteksi banjir menggunakan *Platform Thinger.io* dan notifikasi Telegram, maka dapat disimpulkan beberapa kesimpulan diantaranya adalah sebagai berikut:

1. Sistem deteksi banjir berbasis *thinger.io* mampu mengirimkan data secara real – time dan memberikan notifikasi kepada pengguna melalui telegram.
2. Implementasi sensor yang digunakan dapat mendeteksi perubahan ketinggian air dengan cukup akurasi, sehingga memungkinkan peringatan dini terhadap potensi banjir.
3. Integrasi antara perangkat keras dan *platform .Io* memungkinkan sistem berjalan secara otomatis dan dapat diakses dari jarak jauh.
4. Dengan adanya sistem ini, di harapkan Masyarakat dapat lebih siap menghadapi ancaman banjir dengan menerima notifikasi secara cepat dan akurat.

9.2 Saran

Adapun saran yang dapat disampaikan pada penelitian *prototype* deteksi banjir menggunakan *Platform Thinger.io* dan Notifikasi Telegram adalah pada sistem hendaknya ditambah perangkat *google map*, sehingga pengguna dapat mengetahui titik lokasi kejadian.

DAFTAR PUSTAKA

- Agus, *Aplikasi Sensor Hc-Sr04 Untuk Mengukur Jarak Ketinggian Air Dengan Mikrokontrol Wemos D1 R2 Berbasis Iot (Internet of Things)*, JURNAL TEKNIKA, Volume 6 Nomor 4 September 2020.
- Anata, P. (2022). Sistem Kerja Relay Buchholz Pada Transformator Daya 150 KV di PT. PLN (Persero) Unit Pelaksana Pengendalian Pembangkitan Belawan. *JET (Journal of Electrical Technology)*, 7(3), 124-127.
- Arta, *Sistem Monitoring Ketinggian Gelombang Air Laut Pada Pelabuhan Berbasis Web*, e-Proceeding of Applied Science : Hlm 3, Vol.5, Desember 2019 *Bendungan Untuk Mengatur Ketinggian Air Berbasis Arduino*. Surakarta: Universitas Muhammadiyah Surakarta, 47 halaman.
- Doni, R., & Rahman, M. (2020). Sistem monitoring tanaman hidroponik berbasis IoT (Internet of Thing) menggunakan Nodemcu ESP8266. *J-SAKTI (Jurnal Sains Komputer dan Informatika)*, 4(2), 516-522.
- Firmansyah 2019. *Prototype Sistem Buka Tutup Otomatis Pada Pintu Air Gempa, Kebakaran*. Jakarta: PT Gramedia Widiasarana Indonesia, 39 halaman.
- Gozal, R. P., Setiawan, A., & Khoswanto, H. (2020). Aplikasi SmartRoom Berbasis Blynk untuk Mengurangi Pemakaian Tenaga Listrik. *Jurnal Infra*, 8(1), 39-45.
- Lestari, H., & Rahmawati, I. (2020). Pemahaman NOS Peserta Didik Sekolah Dasar. *Jurnal Sains Indonesia*, 1(1), 18-25.
- Nugroho, R. A., Gunawan, R. D., Prasetyawan, P., & Wijayanto, D. (2021). Sistem Keamanan Kap Mobil Menggunakan Fingerprint Berbasis Mikrokontroler. *Journal ICTEE*, 2(1), 1-9.
- Parwati, N. K. D., Wiharta, D. M., & Setiawan, W. (2018). Rancang Bangun Sistem Peringatan Dini Bahaya Tanah Longsor Dengan Sensor Hygrometer Dan Piezoelectric. *E-Journal SPEKTRUM*, 5(2), 183-190.
- Pasaribu, B., & Susanti, W. (2021 : 29). Sistem Informasi Pengajuan Rancangan Usulan Penelitian Menggunakan PHP Native dan Bot Telegram. *Jurnal Mahasiswa Aplikasi Teknologi Komputer dan Informasi (JMApTeKsi)*, 3(1), 29-38.
- Rifky, *Sistem Monitoring dan Pemberian Pakan Otomatis Pada Budidaya Ikan Menggunakan WEMOS dengan Konsep Internet of Things (IoT)*, Jurnal Digit 10.2 (2020): 185-195.

- Riyanto, E. (2019). Sistem keamanan rumah berbasis android dengan raspberry pi. *Jurnal Informatika Upgris*, 5(1).
- Rostini, A. N., & Junfithrana, A. P. (2020). Aplikasi smart home node mcu iot untuk blynk. *Jurnal Rekayasa Teknologi Nusa Putra*, 7(1), 1-7.
- Sanjaya, H., Daulay, N. K., Trianto, J., & Andri, R. (2022). Tempat Sampah Otomatis Berbasis Mikrokontroler Arduino. *JURIKOM (Jurnal Riset Komputer)*, 9(2), 451-455.
- Siswidiyanto, S., Wijayanti, D., & Haryadi, E. (2020). Sistem Informasi Penyewaan Rumah Kontrakan Berbasis Web Dengan Menggunakan Metode Prototype. *Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi dan Komunikasi*, 15(1), 16-23.
- Sutrahitu, M. E., Kuahaty, S. S., & Balik, A. (2021). Perlindungan Hukum Pemegang Hak Cipta terhadap Pelanggaran Melalui Aplikasi Telegram. *Tatohi: Jurnal Ilmu Hukum*, 1(4), 346-355.
- Sutha. (2018). Bab II Landasan Teori. *Journal of chemical information and modeling*, 53(9), 1689-1699.
- Susanto, F., Prasiani, N. K., & Darmawan, P. (2022). Implementasi Internet of Things Dalam Kehidupan Sehari-Hari. *Jurnal Imagine*, 2(1), 35-40.
- Tryian, *Sistem Monitoring Ketinggian Air Dan Pengendalian Pintu Air Berbasis Microcontroller Nodecode Mcu Esp8266*, /Jurnal Teknologi Informasi dan Komunikas, Hlm 3, Vol 1. 2020.
- Wahyu, *Purwarupa Alat Pendeteksi Kebakaran Jarak Jauh Menggunakan Platform Thinger.io Prototype Of Remote Fire Detection Using The Thinger.io Platform*, Jurnal Elektro Luceat, JELC Vol.7 No.2, [November] [2021]
- Yulaelawati, 2019. *Mencerdasi Bencana: Banjir, Tanah Longsor, Tsunami, Zega, Alat Monitoring Pemakaian Listrik Menggunakan Arduino Uno*. (2022).

L

A

M

P

I

R

A

N

Rencana Kegiatan
(Time Schedule)

No Uraian Kegiatan	JADWAL / BULAN (Minggu Ke)																			
	Februari				Maret				April				Mei				Juni			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1. Pengajuan Judul	■	■																		
2. Perbaikan Judul			■	■																
3. Penyusunan					■	■	■	■												
4. Konsultasi dengan Tim Pembimbing									■	■	■	■								
6. Seminar																				
7. Perbaikan													■							
8. Penelitian / Pengumpulan Data																	■	■		
9. Pembuatan Program																				
11. Sidang Skripsi																				

Bengkulu, m e 2024
Penulis



**Dimas aldi
romadoni**
NPM. 20020018



UNIVERSITAS DEHASEN BENGKULU

FAKULTAS ILMU KOMPUTER

Jl. Meranti Raya No. 32 Sawah Lebar Bengkulu 38228 Telp. (0736) 22027, Fax 60736) 341139 www.unived.ac.id

SURAT KEPUTUSAN
DEKAN FAKULTAS ILMU KOMPUTER
Nomor : 010 /UNIVED.F.1/A-6/1/2025
PENETAPAN DOSEN PEMBIMBING SKRIPSI PROGRAM STUDI REKAYASA SISTEM KOMPUTER (S1)
PADA FAKULTAS ILMU KOMPUTER UNIVERSITAS DEHASEN BENGKULU TA. 2023/2024

DEKAN FAKULTAS ILMU KOMPUTER

- Menimbang** :
- Bahwa untuk menunjang kelancaran pelaksanaan penyusunan Skripsi mahasiswa Program Studi Rekayasa Sistem Komputer Fakultas Ilmu Komputer Jenjang Strata Satu (S-1) Universitas Dehasen Bengkulu, perlu ditunjuk Dosen pembimbing penyusunan Skripsi TA. 2023/2024.
 - Bahwa nama dosen yang tercantum dalam Surat Keputusan ini dianggap cakap dan mampu untuk melaksanakan tugas yang diembannya.
 - Bahwa untuk keperluan sebagaimana tersebut pada butir (a) di atas perlu ditetapkan dengan keputusan Dekan.
- Mengingat** :
- Undang-undang Nomor 20 Tahun 2003 tentang Sistem Pendidikan Nasional.
 - Peraturan Pemerintah RI Nomor 37 tahun 2009 tentang Dosen.
 - Peraturan Pemerintah RI Nomor 4 tahun 2014 tentang Penyelenggaraan Pendidikan Tinggi dan Pengelolaan Perguruan Tinggi.
 - Peraturan Menteri Riset, Teknologi dan Pendidikan Tinggi RI Nomor 44 tahun 2015 tentang standar Nasional Pendidikan Tinggi.
 - Peraturan Menteri Riset, Teknologi dan Pendidikan Tinggi Republik Indonesia Nomor 62 Tahun 2016 tentang Sistem Penjaminan Mutu Pendidikan Tinggi;
 - Keputusan Yayasan Nomor : 007/Y-D/B-2/1/2024 tentang Pemberhentian dan Pengangkatan Dekan Fakultas Ilmu Komputer Universitas Dehasen Bengkulu;

MEMUTUSKAN

- Menetapkan** :
- Pertama** :
- Menunjuk saudara-saudara :
- | | | |
|--|---|--|
| Nama | : | Toibah Umi Kalsum, S. Kom., M. Kom |
| NIDN | : | 02.060573.01 |
| Jabatan Akademik | : | Lektor |
| Sebagai Dosen Pembimbing Utama (I) | | |
| Nama | : | Hendri Alamsyah, S. Kom., M. Kom |
| NIDN | : | 02.110391.01 |
| Jabatan Akademik | : | Asisten Ahli |
| Sebagai Dosen Pembimbing Pendamping (II) | | |
| Untuk membimbing Skripsi mahasiswa : | | |
| Nama | : | Dimas Aldi Romadoni |
| NPM | : | 20020018 |
| Judul Skripsi | : | Prototype Deteksi Banjir Dengan WEB Server Dan Notifikasi Telegram |
- Kedua** : Mengesahkan judul skripsi tersebut di atas dengan ketentuan bahwa judul tersebut dapat dirubah atas petunjuk dan saran dari pembimbing serta melaporkannya kepada Ketua Prodi.
- Ketiga** : Lamanya waktu bimbingan 6 (Enam) bulan terhitung sejak ditetapkannya keputusan ini.
- Keempat** : Mahasiswa yang tidak dapat menyelesaikan skripsi sampai batas waktu yang telah ditetapkan, dinyatakan batal dan harus diajukan kembali seperti persyaratan baru.
- Kelima** : Biaya bimbingan skripsi dibebankan kepada mahasiswa yang bersangkutan.
- Keenam** : Keputusan ini berlaku sejak tanggal ditetapkan dengan ketentuan akan diperbaiki apabila dikemudian hari ternyata terdapat kekeliruan.

Ditetapkan di : Bengkulu

Pada Tanggal : 03 Januari 2025

Dekan


KHANIL S. KOM, M. KOM
NIDN: 02.130475.01



UNIVERSITAS DEHASEN BENGKULU
FAKULTAS ILMU KOMPUTER

Jalan Meranti Raya Nomor 32 Kota Bengkulu 38228 Telpun (0736) 22027, 26957
Fax. (0736) 341139

Bengkulu, 13 Mei 2024

Nomor : 181 / UNIVED-F.1 / A-6/V / 2024
Lamp : -
Hal : Surat Izin Penelitian

Kepada Yth,
Bpk/Ibu Kepala Balai Wilayah Sungai Serut Sumatera VII
Jl. Batang Hari No. 25 Bengkulu, Kec. Ratu Agung Bengkulu

di-
Bengkulu

Dengan Hormat,

Berdasarkan kurikulum Program Stud Rekayasa Sistem Komputer Jenjang Strata 1 (S1) Fakultas Ilmu Komputer Universitas Dehasen Bengkulu, mahasiswa yang akan menyelesaikan studi diwajibkan menyusun Skripsi. Sehubungan dengan itu mahasiswa kami :

Nama Mahasiswa : Dimas Aldi Romadoni
NPM : 20020018

Bermaksud mengadakan penelitian pada KEMENTRIAN PEKERJA UMUM PERUMAHAN RAKYAT DIREKTORAT JENDRAL SUMBER DAYA AIR BENGKULU Guna memperoleh data dalam rangka menyusun Skripsi yang berjudul :

“ **Prototype Deteksi Banjir Dengan WEB Server Dan Notifikasi Telegram** “

Demikian kami sampaikan atas bantuan dan kerjasamanya diucapkan terima kasih.

Hormat Kami,
a.n. Dekan FILKOM
Wakil Dekan FILKOM



Deni Nitalia Zulita, S.Kom, M.Kom
NIDN : 02.251283.02



UNIVERSITAS DEHASEN BENGKULU

FAKULTAS ILMU KOMPUTER

Jln. Meranti Raya No.32 Sawah Lebar Kota Bengkulu , Telp. (0736) 22027

KARTU BIMBINGAN TUGAS AKHIR

Nama Mahasiswa : DIMAS ALDI ROMADONI **Jenis Kelamin** : L
NPM : 20020018
Program Studi : REKAYASA SISTEM KOMPUTER
Alamat : DANGKU SUMATRA SELATAN
No. Telp / HP :
Judul Tugas Akhir : PROTOTYPE DETEKSI BANJIR DENGAN WEB SERVER DAN NOTIFIKASI TELEGRAM
Semester Mulai : Genap 2023/2024
Dosen Pembimbing : TOIBAH UMI KALSUM, M.KOM (Pembimbing Utama)
 Hendri Alamsyah, S.Kom, M.Kom (Pembimbing Pendamping)
Dosen Penguji :
Riwayat Bimbingan : Pembimbing Utama

KE	TANGGAL BIMBINGAN	DOSEN	URAIAN BIMBINGAN	TTD	
				MHS	PEMB
1	2	3	4	5	6
1	16 Maret 2024	TOIBAH UMI KALSUM, M.KOM	ACC Judul silakan segera lakukan proses bimbingan ke pembimbing 2		
2	06 April 2024	TOIBAH UMI KALSUM, M.KOM	Latar belakang belum menggambarkan alasan kenapa mengangkat judul tersebut serta untuk implementasi alat nantinya cari tempat/instansi yang sesuai sehingga apa yang akan dibuat bernilai dan bermanfaat		
3	19 April 2024	TOIBAH UMI KALSUM, M.KOM	Batasan masalah masih salah perbaiki dan uraikan apa saja yang akan dibahas dalam penelitian ini beserta penggunaan komponen hardware dan software yang benar dalam penelitian ini		
4	06 Mei 2024	TOIBAH UMI KALSUM, M.KOM	Pada bab 2 referensi yang digunakan hanya sedikit tambahkan lagi terutama referensi yang berhubungan dengan komponen hardware dan software sesuai dengan materi penelitian		
5	19 Mei 2024	TOIBAH UMI KALSUM, M.KOM	Pada bab3 struktur penulisan belum benar ikuti sesuai pedoman penulisan skripsi dan rancangan global, blok diagram alat serta design rangkaian diperbaiki sesuai saran dan koreksi		
6	03 Juni 2024	TOIBAH UMI KALSUM, M.KOM	Rencana pengujian diperbaiki sesuai saran serta penulisan didaftar pustaka masih salah lihat lagi aturan penulisan dipedoman		
7	15 Juli 2024	TOIBAH UMI KALSUM, M.KOM	Scadule belum ada lengkapi semua lampiran sesuai kebutuhan dalam penelitian		
8	19 Juli 2024	TOIBAH UMI KALSUM, M.KOM	ACC ujian proposal		
9	11 April 2025	TOIBAH UMI KALSUM, M.KOM	Silakan demokan alat yang sudah dibuat disesuaikan dengan yang dirancang pada proposal		
10	12 April 2025	TOIBAH UMI KALSUM, M.KOM	Hasil Output pendeksian banjir belum sesuai dengan yang diharapkan perbaiki logikanya sehingga hasil output yang menyatakan kondisi banjir itu level airnya seharusnya memang ukuran tertinggi perbaiki lagi sesuai intruksi		
11	17 April 2025	TOIBAH UMI KALSUM, M.KOM	Pada hasil ujicoba masih salah logikanya perbaiki lagi serta tulisan2 yang masih salah diperbaiki lagi		
12	21 April 2025	TOIBAH UMI KALSUM, M.KOM	Riwayat hidup, ringkasan serta lampiran dilengkapi		
13	29 April 2025	TOIBAH UMI KALSUM, M.KOM	Acc silakan lakukan proses untuk ujian skripsi		

Mengetahui,

Ketua Program Studi
 Hendri Alamsyah, S.Kom, M.Kom
 NIDN: 0211039102

Pembimbing Utama

TOIBAH UMI KALSUM, M.KOM
 NIDN: 02.060573.01



PEMERINTAH KOTA BENGKULU
KEMENTERIAN PEKERJA UMUM PERUMAHAN RAKYAT
DIREKTORAT SUMBER DAYA AIR BENGKULU



Jl, Batang Hari No. 25 Kecamatan Ratu Agung Kota Bengkulu

SURAT KETERANGAN

Nomor : 070/90/PUPR-1/2025

Yang bertanda tangan di bawa ini :

Nama : Medya Ramdhan, ST

NIP : 1969112219970031006

Jabatan : Kepala Balai Wilayah Sungai Sumatra VII Bengkulu

Dengan ini menerangkan bawa :

Nama : Dimas Aldi Romadoni

NPM : 20020018

Prodi : Rekayasa Sistem Komputer

Universitas : Dehasen Bengkulu

Telah melakukan penelitian di Kementerian Pekerja Umum Perumahan Rakyat Direktorat Sumber Daya Air Bengkulu dengan julul **“PROTOTYPE DETEKSI BANJIR MENGGUNAKAN *PLATFROM THINGER.IO* DAN NOTIFIKASI TELEGRAM”**. Yang berlangsung mulai dari 11 januari 2025 s.d 08 Maret 2025.

Demikian Surat keterangan ini di buat untuk dapat di pergunakan sebagaimana mestinya.

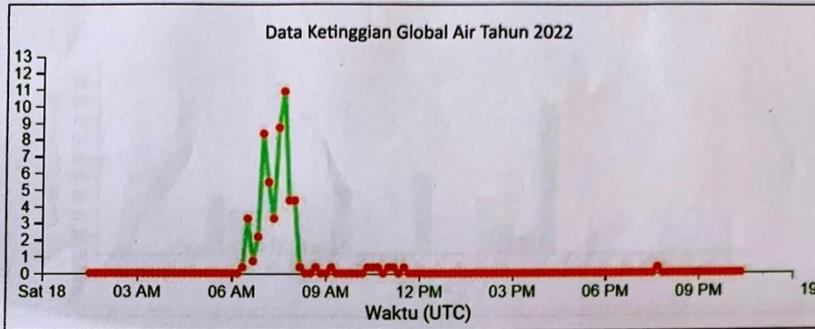
Bengkulu, 13 Mei 2025

Kepala Balai Wilayah
Sungai Sumatra VII Bengkulu

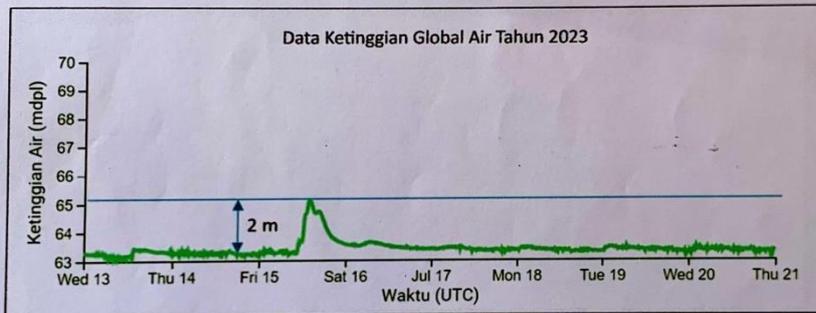

Medya Ramdhan, ST

NIP : 1969112219970031006

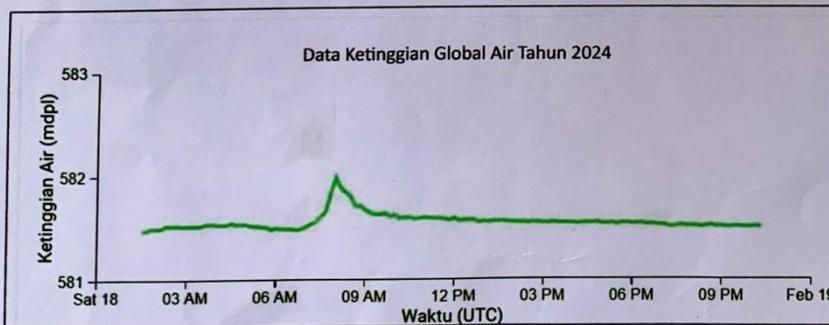
Data Global Naik Turun Ketinggian Air Di Bendungan Tanjung Agung Tahun 2022-2024



air berpotensi turun di tahun 2022

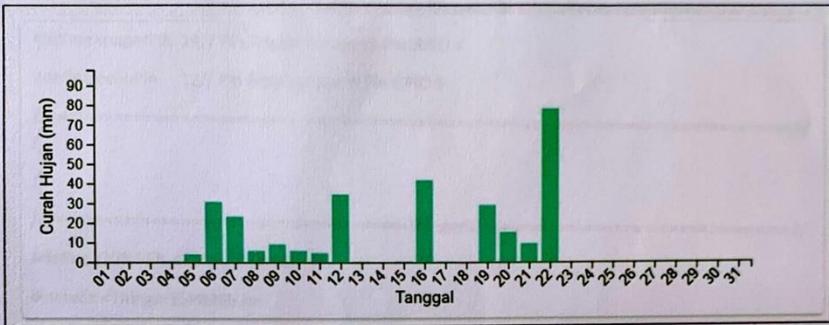


air berpotensi naik di tahun 2023



Air berpotensi naik tinggi di tahun 2024

Curah Hujan 5 Bulan Terakhir Di 2025



Coding telegram

```
//=====Sensor Ultrasonik=====//
#define triggerPin 14// Pin Trigger Sensor di Pin GPIO 4
#define echoPin 12// Pin Echo Sensor di Pin GPIO 5
//=====/
/
/*
//=====ThingerIO=====//
#define THINGER_SERIAL_DEBUG
#include <ThingerESP8266.h>

#define USERNAME "dimasaldiromadoni" // USERNAME ThingerIO
#define DEVICE_ID "Banjir" //DIVICE ID ThingerIO
#define DEVICE_CREDENTIAL "33wmcDgDHUdn3&9B" //DEVICE CREDENTIAL ThingerIO
ThingerESP8266 thing(USERNAME, DEVICE_ID, DEVICE_CREDENTIAL);
//=====/
/
*/
//=====Buzzer=====//
#define buzzer 13 // Pin Buzzer di GPIO 0
//=====/
/

//=====Telegram=====//
#include "CTBot.h"
CTBot myBot;
String ssid = "Amanah kanan";
String pass = "kosong3kamar";
String token = "7549286227:AAEKQ05U0fr-5RBQFcEPXGKglyk3MhjBx2g";
const int id = 8082998049;
```

```

//=====Wifi=====
/
/*
//=====Wifi=====
#define SSID "Banjir" //Nama WIFI yang akan digunakan
#define SSID_PASSWORD "12345678" // Password Wifi
//=====Wifi=====
/
*/

#include <Wire.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27,20,4);

long duration, jarak;
void setup() {
  Serial.begin (115200);
  pinMode(13, OUTPUT);
  lcd.init();          // initialize the lcd
  lcd.init();
  // Print a message to the LCD.
  lcd.backlight();
//
  Serial.println("Memulai Telegram Bot. Koneksi ke Wifi");
  myBot.wifiConnect(ssid,pass);
  myBot.setTelegramToken(token);
  if(myBot.testConnection())
  { Serial.println("Koneksi Berhasil");}
  else
  { Serial.println("Koneksi Gagal");}
  myBot.sendMessage(8082998049, "notification test");
  Serial.println("Pesan Terkirim");
}

```

```

//
    lcd.setCursor(0,0);
    lcd.print("Deteksi Banjir ");
    delay(4000);
    /* // Penambahan Wifi
    thing.add_wifi(SSID, SSID_PASSWORD);
    thing["jarak"] >> outputValue(jarak);// Variabel Output Jarak
    */

//=====Sensor Ultrasonik PinMode=====//
    pinMode(triggerPin, OUTPUT);
    pinMode(echoPin, INPUT);
    lcd.clear();
    digitalWrite(13, LOW);
}
void loop() {
    /*
    //=====ThingerIO=====//
    thing.handle();
    thing.stream(thing["jarak"]);

    //=====//
    /
    */
    //=====membaca Jarak Ultrasonik=====//
    digitalWrite(triggerPin, LOW);
    delayMicroseconds(2);
    digitalWrite(triggerPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(triggerPin, LOW);
    duration = pulseIn(echoPin, HIGH);

```

```

//
    lcd.setCursor(0,0);
    lcd.print("Deteksi Banjir ");
    delay(4000);
    /* // Penambahan Wifi
    thing.add_wifi(SSID, SSID_PASSWORD);
    thing["jarak"] >> outputValue(jarak);// Variabel Output Jarak
    */

//=====Sensor Ultrasonik PinMode=====//
pinMode(triggerPin, OUTPUT);
pinMode(echoPin, INPUT);
lcd.clear();
digitalWrite(13, LOW);
}
void loop() {
/*
//=====ThingerIO=====//
thing.handle();
thing.stream(thing["jarak"]);

//=====/
/
*/
//=====membaca Jarak Ultrasonik=====//
digitalWrite(triggerPin, LOW);
delayMicroseconds(2);
digitalWrite(triggerPin, HIGH);
delayMicroseconds(10);
digitalWrite(triggerPin, LOW);
duration = pulseIn(echoPin, HIGH);

```

```
jarak = (duration/2) / 29.1;
lcd.setCursor(0,0);
lcd.print("jarak : ");
lcd.print((168-jarak));
lcd.print(" cm");
Serial.print("jarak :");
Serial.print(jarak);
Serial.println(" cm");
delay(1000);

//=====
/

if ((168-jarak)<72){
  myBot.sendMessage (8082998049, "Bahaya");
  digitalWrite(13, HIGH);
  delay(3000);
}
else if ((168-jarak)<128){
  myBot.sendMessage (8082998049, "Waspada");
  digitalWrite(13, LOW);
}
else if ((168-jarak)<=168){
  myBot.sendMessage (8082998049, "Aman");
  digitalWrite(13, LOW);
}
}
```

```

coding Think.io

//=====Sensor Ultrasonik=====//
#define triggerPin 14// Pin Trigger Sensor di Pin GPIO 4
#define echoPin 12// Pin Echo Sensor di Pin GPIO 5

//=====/
/
/*
//=====ThingierIO=====//
#define THINGER_SERIAL_DEBUG
#include <ThingierESP8266.h>

#define USERNAME "dimasaldiromadoni" // USERNAME ThingierIO
#define DEVICE_ID "Banjir" //DEVICE ID ThingierIO
#define DEVICE_CREDENTIAL "33wmcDgDHUdn3&9B" //DEVICE CREDENTIAL ThingierIO
ThingierESP8266 thing(USERNAME, DEVICE_ID, DEVICE_CREDENTIAL);
//=====/
/
*/
//=====Buzzer=====//
#define buzzer 13 // Pin Buzzer di GPIO 0

//=====/
/

//=====Telegram=====//
#include "CTBot.h"
CTBot myBot;
String ssid = "Amanah kanan";
String pass = "kosong3kamar";
String token = "7549286227:AAEKQ05U0fr-5RBQFcEPXGKgIyk3MhjBx2g";
const int id = 8082998049;

```

```

//
    lcd.setCursor(0,0);
    lcd.print("Deteksi Banjir ");
    delay(4000);
    /* // Penambahan Wifi
    thing.add_wifi(SSID, SSID_PASSWORD);
    thing["jarak"] >> outputValue(jarak);// Variabel Output Jarak
    */

//=====Sensor Ultrasonik PinMode=====//
    pinMode(triggerPin, OUTPUT);
    pinMode(echoPin, INPUT);
    lcd.clear();
    digitalWrite(13, LOW);
}
void loop() {
    /*
    //=====ThingerIO=====//
    thing.handle();
    thing.stream(thing["jarak"]);

    //=====//
    /
    */
    //=====membaca Jarak Ultrasonik=====//
    digitalWrite(triggerPin, LOW);
    delayMicroseconds(2);
    digitalWrite(triggerPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(triggerPin, LOW);
    duration = pulseIn(echoPin, HIGH);

```

```
jarak = (duration/2) / 29.1;
lcd.setCursor(0,0);
lcd.print("jarak : ");
lcd.print(jarak);
lcd.print(" cm");
Serial.print("jarak :");
Serial.print(jarak);
Serial.println(" cm");
delay(1000);

//=====
/

if (jarak<72){
  myBot.sendMessage (8082998049, "Bahaya");
  digitalWrite(13, HIGH);
  delay(3000);
}
else if (jarak<128){
  myBot.sendMessage (8082998049, "Waspada");
  digitalWrite(13, LOW);
}
else if (jarak<=168){
  myBot.sendMessage (8082998049, "Aman");
  digitalWrite(13, LOW);
}
}
```

Lampiran Dokumentasi

