

*Skripsi Geofisika*

**PERANCANGAN SISTEM PERINGATAN DINI BANJIR BERBASIS  
*INTERNET OF THINGS (IOT)* MENGGUNAKAN ARDUINO UNO  
DENGAN METODE LOGIKA *FUZZY* MAMDANI**



**OLEH:  
MUHAMMAD YAHYA  
H061181334**

**DEPARTEMEN GEOFISIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS HASANUDDIN  
MAKASSAR  
2023**

**HALAMAN JUDUL**

**PERANCANGAN SISTEM PERINGATAN DINI BANJIR BERBASIS  
*INTERNET OF THINGS (IOT)* MENGGUNAKAN ARDUINO UNO  
DENGAN METODE LOGIKA *FUZZY* MAMDANI**

**SKRIPSI**

*Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Sains  
Pada Departemen Geofisika  
Fakultas Matematika dan Ilmu Pengetahuan Alam  
Universitas Hasanuddin*

**OLEH:**

**MUHAMMAD YAHYA  
H061181334**

**DEPARTEMEN GEOFISIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS HASANUDDIN  
MAKASSAR  
2023**

**HALAMAN PENGESAHAN**

**“PERANCANGAN SISTEM PERINGATAN DINI BANJIR BERBASIS  
INTERNET OF THINGS (IOT) MENGGUNAKAN ARDUINO UNO  
DENGAN METODE LOGIKA FUZZY MAMDANI”**

**Disusun dan diajukan oleh:**

**Muhammad Yahya  
H061181334**

Telah dipertahankan di depan Panitia Ujian yang dibentuk dalam rangka  
Penyelesaian Program Sarjana Program Studi Geofisika Fakultas Matematika dan  
Ilmu Pengetahuan Alam Universitas Hasanuddin

Pada 31 Juli 2023  
Dinyatakan telah memenuhi syarat kelulusan

Menyetujui,

**Pembimbing Utama**

**Pembimbing Pertama**

  
**Dr. Samsu Arif, M.Si**  
NIP. 196305181991031011

  
**Dr. Muh. Alimuddin Hamzah, M.Eng**  
NIP. 19670929291993031003

**Ketua Departemen Geofisika  
Fakultas Matematika dan Ilmu Pengetahuan Alam  
Universitas Hasanuddin Makassar**

  
**Dr. Muh. Alimuddin Hamzah, M.Eng**  
NIP. 19670929291993031003

## PERNYATAAN KEASLIAN

Saya yang bertandatangan di bawah ini:

N a m a : Muhammad Yahya  
NIM : H061181334  
Departemen : Geofisika  
Judul Skripsi : Perancangan Sistem Peringatan Dini Banjir Berbasis *Internet of Things* (IoT) Menggunakan Arduino Uno dengan Metode Logika *Fuzzy* Mamdani.

Menyatakan bahwa skripsi ini benar-benar hasil karya saya sendiri dan belum pernah diajukan untuk mendapatkan gelar sarjana di Universitas Hasanuddin atau Lembaga Penelitian lain kecuali kutipan dengan mengikuti tata penulisan karya ilmiah yang sudah lazim digunakan, karya tulis ini merupakan murni dari gagasan penelitian saya sendiri, kecuali arahan dari Tim Pembimbing dan masukan Tim Penguji.

Makassar, 7 Juli 2023

Yang membuat pernyataan,



Muhammad Yahya

## SARI BACAAN

### **Perancangan Sistem Peringatan Dini Banjir Berbasis *Internet of Things* (IoT) Menggunakan Arduino Uno dengan Metode Logika *Fuzzy* Mamdani**

Banjir salah satu jenis bencana yang paling sering terjadi di Indonesia. Perlu adanya peringatan dini banjir agar masyarakat maupun pemerintah dapat mempersiapkan diri. Penelitian ini bertujuan merancang sistem peringatan dini banjir berbasis *Internet of Things* (IoT) menggunakan Arduino Uno dengan metode logika *fuzzy* Mamdani. Metode penelitian ini adalah merancang penggunaan mikrokontroler Arduino Uno sebagai pengendali sensor ultrasonik untuk mengukur tinggi air dan sensor *reed switch* untuk mengukur curah hujan. Tinggi air dan curah hujan digunakan untuk perhitungan potensi banjir metode logika *fuzzy* Mamdani. Informasi berupa tinggi air, curah hujan, dan potensi banjir ditampilkan pada *website*, ICD 16x2 dan disimpan pada data *logger*. Sistem dirancang membunyikan *buzzer* dan mengirim notifikasi Telegram. Rancangan kemudian diuji untuk memperoleh akurasi sensor dan memastikan sistem dapat berfungsi. Hasil penelitian ini adalah terciptanya sistem peringatan dini banjir yang menampilkan informasi pada LCD dan *website* dengan *delay* pengiriman data ke *database* rata-rata 0,78 detik. Selain itu, informasi juga disimpan pada data *logger*. Saat potensi tinggi terjadinya banjir, sistem berhasil membunyikan *buzzer* dan mengirim notifikasi Telegram. Pengujian menunjukkan akurasi pengukur tinggi air dan penakar curah hujan masing-masing sebesar 96,99% dan 98,61%. Dapat disimpulkan bahwa hasil perhitungan potensi banjir oleh sistem peringatan dini sesuai dengan simulasi MatLAB.

**Kata kunci:** *Early Warning System*, *Monitoring* Banjir, *Prediksi* Banjir, *Sensor* Ultrasonik, *Tipping Bucket*.

## **ABSTRACT**

### ***Designing an Internet of Things (IoT) Based Flood Early Warning System Using Arduino Uno with Mamdani Fuzzy Logic Method***

*Floods are one of the most common types of disasters in Indonesia. Flood early warning is needed so the community and government can prepare themselves. This research aims to design an Internet of Things (IoT)-based flood early warning system using Arduino Uno with the Mamdani fuzzy logic method. This research method is to design the use of the Arduino Uno microcontroller as a controller for ultrasonic sensors to measure water levels and reed switch sensors to measure rainfall. Water level and rainfall are used for calculating flood potential using the Mamdani fuzzy logic method. Water level, rainfall, and flood potential information are displayed on the website's 16x2 LCD and stored in the data logger. The system is designed to sound a buzzer and send Telegram notifications. The design is then tested to obtain sensor accuracy and ensure the system can function. The result of this research is the creation of a flood early warning system that displays information on the LCD and website with an average data transmission delay to the database of 0.78 seconds. In addition, the information is recorded and stored on the data logger. When there is a high potential for flooding, the system successfully sounds a buzzer and sends a Telegram notification. Tests show that the water level meter and rainfall gauge accuracy is 96.99% and 98.61%, respectively. In conclusion, the flood potential calculation results by the early warning system are in accordance with MatLAB simulations.*

**Keywords:** *Early Warning System, Flood Monitoring, Flood Prediction, Ultrasonic Sensor, Tipping Bucket.*

## KATA PENGANTAR

*Bismillahirrahmanirrahim*

*Assalamu'alaykum warahmatullahi wabarakatuh*

Alhamdulillah, segala puji bagi Allah Subhanahu wa Ta'ala yang senantiasa memberikan limpahan rahmat dan kesehatan kepada saya selaku penulis sehingga dapat mampu menyelesaikan skripsi dengan judul “**Perancangan Sistem Peringatan Dini Banjir Berbasis *Internet of Things* (IoT) Menggunakan Arduino Uno dengan Metode Logika Fuzzy Mamdani**”. Shalawat serta salam tak lupa dihaturkan kepada baginda Rasulullah Shallallahu ‘Alaihi wa Sallam, keluarga, para sahabat, dan para pengikutnya.

Selama penulisan dan penyelesaian skripsi ini tidak terlepas dari berbagai rintangan, hambatan dan keterbatasan penulis dalam penyusunannya. Namun, berkat bantuan dan motivasi dari berbagai pihak sehingga semua masalah dalam penyelesaian skripsi penulis dapat teratasi dengan perlahan. Dalam kesempatan ini, dengan segala kerendahan hati penulis mengucapkan terima kasih banyak kepada kedua orangtua tercinta, yaitu ayah **Bakri** dan Ibu **Dumrah**. Yang merupakan *support system* terbesar sehingga penulis bisa berada pada tahap ini. Penulis juga mengucapkan terima kasih atas dukungan dari kakak tercinta, yaitu **Kak Udi**. Tidak lupa pula ucapan terima kasih kepada **Adik Ekky** dan **Nenek Bongkasa** sebagai motivasi penulis untuk segera menyelesaikan studi.

Ucapan terima kasih juga penulis haturkan kepada seluruh pihak-pihak yang terlibat dan memberikan bantuan serta bimbingan dalam proses penyelesaian

studi. Oleh karena itu, dengan segala kerendahan hati penulis mengucapkan terima kasih dan penghargaan sebesar-besarnya kepada:

1. Bapak **Dr. Samsu Arif, M.Si** dan Bapak **Dr. Muh. Alimuddin Hamzah, M.Eng**, selaku pembimbing utama dan pembimbing pertama yang senantiasa selalu memberikan arahan, bimbingan, serta motivasi kepada penulis untuk dapat menyelesaikan skripsi ini.
2. Bapak **Dr. Erfan Syamsuddin, M.Si**, dan Bapak **Aswar Syafnur, S.Si, M.Eng**, selaku penguji yang telah memberikan saran dan koreksi kepada penulis dalam penyelesaian skripsi agar lebih baik.
3. Bapak **Dr. Muh. Alimuddin Hamzah, M.Eng.**, dan Bapak **Erfan Syamsuddin, M.Si.**, selaku Ketua dan Sekretaris Geofisika FMIPA Unhas.
4. Bapak **Sabrianto Aswad, S.Si, MT**, selaku dosen penasihat akademik yang telah memberikan nasehat dan senantiasa bersikap ramah kepada penulis.
5. Seluruh **Dosen Departemen Geofisika** dan **Staf Departemen Geofisika**, atas ilmu pengetahuan selama masa studi penulis dan bantuan pelayanan administrasi yang telah diberikan.
6. **Muhammad Fadil**, Kakak yang selalu memberikan motivasi, arahan dan bantuan dalam menyelesaikan skripsi ini. Tanpa bantuannya, mungkin penulis tidak bisa menyelesaikan skripsi ini dengan baik. Terima kasih atas bantuannya selama ini.
7. **Ainul Fajri**, selaku sahabat dan teman kosan penulis yang selalu bersama.

8. **Hisbullah** dan **Rahmatullah**, selaku sahabat yang telah membantu penulis dalam mendesain alat.
9. **Abdul Rahman** dan **Iqbal Firdi**, selaku sahabat yang selalu mengantar dan menjemput penulis dari kost ke kampus ataupun sebaliknya.
10. **Yusran**, selaku sahabat yang selalu kebersamai penulis semenjak mahasiswa baru. Terima kasih sudah mau bersahabat dengan penulis.
11. Bapak **Wawan**, selaku tetangga sekaligus keluarga yang telah membantu dalam proses pembuatan perangkat keras alat.
12. **Prayoga** dan **Safira Fauzianingsih**, selaku sahabat yang selalu bersedia menjawab berbagai pertanyaan penulis selama menyelesaikan tugas akhir.
13. Seluruh teman-teman **Geofisika 2018**, atas dukungan dan saling membantu dalam penyelesaian studi penulis.
14. Teman-teman **KKN Gel. 107 Posko 04 Bulukumba**, atas keseruan dan pengabdian bersama selama KKN.

Serta kepada semua pihak yang telah membantu penulis selama menempuh masa studinya. Jika terdapat kesalahan dan kekurangan dalam penelitian ini, penulis memohon maaf yang sebesar-besarnya. Saran dan kritik dari pembaca akan sangat membantu untuk kemajuan penulis di masa mendatang. Akhir kata, semoga skripsi ini dapat bermanfaat bagi semua orang yang membacanya.

*Wassalamualaykum warahmatullahi wabarakatuh*

**Makassar, 7 Juli 2023**

**Penulis**

## DAFTAR ISI

<b>HALAMAN JUDUL .....</b>	<b>ii</b>
<b>HALAMAN PENGESAHAN.....</b>	<b>iii</b>
<b>PERNYATAAN KEASLIAN.....</b>	<b>iv</b>
<b>SARI BACAAN.....</b>	<b>v</b>
<b>ABSTRACT .....</b>	<b>vi</b>
<b>KATA PENGANTAR.....</b>	<b>vii</b>
<b>DAFTAR ISI .....</b>	<b>x</b>
<b>DAFTAR GAMBAR.....</b>	<b>xiii</b>
<b>DAFTAR TABEL .....</b>	<b>xv</b>
<b>DAFTAR LAMPIRAN.....</b>	<b>xvi</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
I.1 Latar Belakang.....	1
I.2 Rumusan Masalah .....	3
I.3 Ruang Lingkup Penelitian .....	3
I.4 Tujuan Penelitian.....	4
<b>BAB II TINJAUAN PUSTAKA .....</b>	<b>5</b>
II.1 Pengertian Sistem.....	5
II.2 Pengertian Perancangan Sistem .....	5
II.3 Peringatan Dini Banjir.....	6
II.4 <i>Internet of Things</i> .....	7
II.5 Arduino Uno.....	8
II.5.1 Komponen Arduino .....	9
II.5.1.1 <i>Hardware</i> .....	9
II.5.1.2 <i>Software</i> .....	10
II.5.2 Dasar Pemrograman .....	11
II.6 Sensor Ultrasonik HC-SR04 .....	13
II.7 Sensor <i>Reed Switch</i> .....	15
II.8 <i>Tipping Bucket</i> .....	17

II.9 <i>Liquid Crystal Display</i> (LCD) .....	19
II.10 I2C LCD .....	19
II.11 NodeMCU .....	20
II.12 Modul Micro SD <i>Card</i> .....	21
II.13 Modul RTC DS1302 ( <i>Real Time Clock</i> ) .....	22
II.14 <i>Buzzer</i> .....	23
II.15 Logika <i>Fuzzy</i> .....	24
II.15.1 Himpunan Tegas ( <i>Crisp</i> ) dan Himpunan <i>Fuzzy</i> .....	25
II.15.2 Fungsi Keanggotaan .....	26
II.15.3 Sistem Inferensia <i>Fuzzy</i> .....	28
II.15.4 Metode Mamdani .....	29
II.16 <i>Database</i> .....	31
II.17 MySQL .....	32
II.18 <i>Hypertext Markup Language</i> (HTML) .....	32
II.19 <i>Hipertext Preprocessor</i> (PHP) .....	33
II.20 JavaScript .....	34
II.21 <i>Cascading Style Sheets</i> (CSS) .....	35
<b>BAB III METODOLOGI PENELITIAN .....</b>	<b>36</b>
III.1 Lokasi dan Waktu Penelitian .....	36
III.2 Alat dan Bahan .....	36
III.3 Blok Diagram .....	37
III.4 <i>Flow Chart</i> Sistem .....	39
III.5 <i>Flow Chart</i> Logika <i>Fuzzy</i> .....	41
III.6 Perancangan Alat .....	41
III.6.1 Perancangan <i>Hardware</i> Mekanik .....	42
III.6.1.1 Perancangan Pengukur Tinggi Air .....	42
III.6.1.2 Perancangan Penakar Curah Hujan <i>Tipping Bucket</i> .....	43
III.6.2 Perancangan <i>Hardware</i> Elektrik .....	44
III.6.3 Perancangan <i>Software</i> .....	45
III.6.3.1 Perancangan <i>Database</i> .....	45
III.6.3.2 Perancangan <i>Website</i> .....	46

III.6.3.3 Perancangan Logika <i>Fuzzy</i> .....	46
III.7 Pengujian Kinerja Sistem.....	49
III.7.1 Pengujian <i>Hardware</i> .....	49
III.7.2 Pengujian <i>Software</i> .....	49
III.7.2.1 Pengujian <i>Database</i> .....	49
III.7.2.2 Pengujian <i>Website</i> .....	49
III.7.2.3 Pengujian Logika <i>Fuzzy</i> .....	50
<b>BAB IV HASIL DAN PEMBAHASAN .....</b>	<b>51</b>
IV. 1 Hasil Pembuatan Alat Secara Keseluruhan .....	51
IV.2 Pengujian Kinerja <i>Hardware</i> .....	54
IV.2.2 Pengujian dan Kalibrasi Sensor Ultrasonik .....	55
IV.2.2.1 Pengukuran Pulsa Echo Terhadap Jarak.....	55
IV.2.2.2 Pengukuran Jarak.....	57
IV.2.3 Pengujian Sensor <i>Reed Switch</i> .....	59
IV.2.4 Pengujian Alat Pengukur Ketinggian Air .....	60
IV.2.5 Pengujian Alat Penakar Curah Hujan .....	62
IV.2.6 Pengujian <i>Buzzer</i> .....	64
IV.2.7 Pengujian <i>Liquid Crystal Display (LCD)</i> .....	65
IV.2.8 Pengujian Data <i>Logger</i> .....	65
IV.2.9 Pengujian <i>Real Time Clock (RTC)</i> .....	66
IV.3 Pengujian Kinerja <i>Software</i> .....	67
IV.3.1 Pengujian <i>Database</i> .....	67
IV.3.2 Pengujian <i>Website</i> .....	68
IV.3.3 Pengujian Notifikasi Telegram .....	69
IV.4 Pengujian Logika <i>Fuzzy</i> .....	70
<b>BAB V PENUTUP .....</b>	<b>73</b>
V.1 Kesimpulan .....	73
V.2 Keterbatasan Alat.....	74
V.3 Saran .....	75
<b>DAFTAR PUSTAKA.....</b>	<b>77</b>
<b>LAMPIRAN</b>	

## DAFTAR GAMBAR

<b>Gambar 2.1</b> Komponen Arduino Uno .....	9
<b>Gambar 2.2</b> Arduino IDE <i>Interface</i> .....	11
<b>Gambar 2.3</b> Inisialisasi <i>pin</i> yang digunakan pada fungsi “ <i>void setup ()</i> ” .....	12
<b>Gambar 2.4</b> Inisialisasi serial monitor yang digunakan pada fungsi “ <i>void setup()</i> ” .....	12
<b>Gambar 2.5</b> Fungsi “ <i>void loop()</i> ” .....	13
<b>Gambar 2.6</b> Fungsi penundaan .....	13
<b>Gambar 2.7</b> Sensor ultrasonik dan prinsip kerjanya.....	14
<b>Gambar 2.8</b> Gelombang yang dipantulkan dari berbagai target permukaan miring.....	15
<b>Gambar 2.9</b> <i>Normally open Reed Switch</i> dan <i>Normally closed Reed Switch</i> . ....	16
<b>Gambar 2.10</b> Prinsip kerja <i>Reed Switch</i> . ....	17
<b>Gambar 2.11</b> Prinsip Kerja <i>Tipping Bucket</i> .....	18
<b>Gambar 2.12</b> LCD 16x2 <i>Display</i> .....	19
<b>Gambar 2.13</b> Modul 12C LCD .....	20
<b>Gambar 2.14</b> NodeMCU <i>Devkit v1.0 (Versi 2)</i> .....	21
<b>Gambar 2.15</b> Modul Micro SD <i>Card</i> .....	22
<b>Gambar 2.16</b> Modul RTC DS1302.....	23
<b>Gambar 2.17</b> <i>Buzzer</i> .....	24
<b>Gambar 2.18</b> Representasi Linear Naik.....	26
<b>Gambar 2.19</b> Representasi Linear Turun.....	27
<b>Gambar 2.20</b> Representasi Kurva Segitiga.....	27
<b>Gambar 2.21</b> Representasi Kurva Trapesium.....	28
<b>Gambar 2.22</b> Struktur Dasar Sistem Inferensia <i>Fuzzy</i> .....	29
<b>Gambar 2.23</b> Struktur program HTML .....	33
<b>Gambar 3.1</b> Diagram Blok Sistem Peringatan Dini Banjir .....	38
<b>Gambar 3.2</b> <i>Flowchart</i> sistem peringatan dini banjir.....	39
<b>Gambar 3.3</b> <i>Flowchart</i> Logika <i>Fuzzy</i> .....	41
<b>Gambar 3.4</b> Perancangan Alat Pengukur Tinggi Air.....	42

<b>Gambar 3.5</b> Rancangan bejana .....	43
<b>Gambar 3.6</b> Perancangan <i>Hardware</i> Elektrik.....	44
<b>Gambar 3.7</b> Rancangan <i>interface website</i> Sistem Peringatan Dini Banjir .....	46
<b>Gambar 3.8</b> Fungsi keanggotaan himpunan masing-masing variabel.....	47
<b>Gambar 4.1</b> Hasil Alat Secara Keseluruhan .....	51
<b>Gambar 4.2</b> <i>Box</i> Mikrokontroler .....	52
<b>Gambar 4.3</b> <i>Box Power Supply</i> .....	52
<b>Gambar 4.4</b> Alat Penakar Curah Hujan Model <i>Tipping Bucket</i> .....	53
<b>Gambar 4.5</b> Alat Pengukur Ketinggian Air .....	54
<b>Gambar 4.6</b> Proses pengujian sensor ultrasonik menggunakan dinding penghalang .....	55
<b>Gambar 4.7</b> Kurva kalibrasi pulsa echo dengan jarak .....	56
<b>Gambar 4.8</b> Kurva korelasi jarak hasil <i>sensing</i> dengan jarak referensi.....	57
<b>Gambar 4.9</b> Plot relatif <i>error</i> data hasil pengukuran sensor.....	58
<b>Gambar 4.10</b> Proses pengujian alat pengukur ketinggian air .....	61
<b>Gambar 4.11</b> Kurva korelasi antara tinggi air hasil pengukuran alat dengan skala referensi.....	62
<b>Gambar 4.12</b> Proses kalibrasi <i>Tipping Bucket</i> .....	63
<b>Gambar 4.13</b> Pengujian tampilan LCD .....	65
<b>Gambar 4.14</b> <i>Website</i> sistem peringatan dini banjir .....	68
<b>Gambar 4.15</b> Pengujian pesan notifikasi Telegram.....	69
<b>Gambar 4.16</b> Fungsi keanggotaan tinggi air.....	70
<b>Gambar 4.17</b> Fungsi keanggotaan curah hujan .....	71
<b>Gambar 4.18</b> Fungsi keanggotaan <i>output</i> potensi banjir .....	71

## DAFTAR TABEL

<b>Tabel 3.1</b> Perancangan <i>Database</i> .....	45
<b>Tabel 4.1</b> Pengujian toleransi jarak sensor <i>reed switch</i> dengan magnet .....	59
<b>Tabel 4.2</b> Hasil Pengujian <i>Tipping Bucket</i> .....	63
<b>Tabel 4.3</b> Hasil Pengujian <i>Buzzer</i> .....	64
<b>Tabel 4.4</b> Hasil Pengujian logika <i>fuzzy</i> .....	72

## DAFTAR LAMPIRAN

<b>Lampiran 1.</b> Hasil pengukuran pulsa echo terhadap jarak .....	83
<b>Lampiran 2.</b> Hasil pengukuran jarak <i>sensing</i> terhadap jarak referensi .....	85
<b>Lampiran 3.</b> Pengukuran tinggi air oleh alat dengan tinggi air referensi .....	87
<b>Lampiran 4.</b> Hasil pengujian Data <i>Logger</i> .....	88
<b>Lampiran 5.</b> Hasil pengujian RTC DS1302 .....	89
<b>Lampiran 6.</b> Hasil pengujian <i>database</i> .....	90
<b>Lampiran 7.</b> Hasil pengujian <i>website</i> .....	91
<b>Lampiran 8.</b> <i>Script</i> Arduino Uno.....	93
<b>Lampiran 9.</b> <i>Script</i> NodeMCU .....	101s

# **BAB I**

## **PENDAHULUAN**

### **I.1 Latar Belakang**

Bencana hidrometeorologi menjadi salah satu jenis bencana alam yang sering terjadi di berbagai Wilayah Negara Kesatuan Republik Indonesia. Kejadian bencana hidrometeorologi merupakan kelompok kejadian bencana banjir, gelombang ektrim, kebakaran lahan dan hutan, kekeringan, dan cuaca ektrim (Amri *et al.*, 2016). Pada periode tahun 2017 sampai dengan tahun 2021, Indonesia mengalami 17.296 bencana alam, dimana bencana hidrometeorologi menyumbang 74% (12.828 kejadian), menurut data yang dihimpun oleh Data Informasi Bencana Indonesia (DIBI)-BNPB. Banjir dan cuaca ektrim merupakan jenis bencana yang paling sering terjadi.

Banjir dapat menimbulkan kerugian besar terhadap pemerintah maupun warga setempat. Banjir tidak hanya mengakibatkan kehilangan materi seperti kerusakan rumah, kehilangan harta benda, dan bahkan korban jiwa. Tetapi banjir juga dapat merusak fasilitas umum. Sebagaimana data yang dihimpun dari Badan Penanggulangan Bencana Daerah Provinsi Sulawesi Selatan, banjir pada tahun 2020 mengakibatkan 2 orang meninggal, 1.094 orang mengungsi dan 25.530 rumah warga rusak. Bahkan dari segi fasilitas umum terdapat 49 jembatan dan 81.526 jalan yang rusak akibat banjir.

Dampak banjir yang besar membuat bencana ini perlu mendapat perhatian serius, salah satunya peringatan dini banjir. Peringatan dini adalah serangkaian kegiatan

pemberian peringatan sesegera mungkin kepada masyarakat tentang kemungkinan terjadinya bencana pada suatu tempat oleh lembaga yang berwenang (BNPB, 2012). Permasalahan saat ini adalah kurangnya informasi peringatan dini banjir pada suatu Daerah Aliran Sungai (DAS). Padahal, dampak banjir dapat ditekan seminim mungkin dengan adanya peringatan dini tersebut karena masyarakat dapat mempersiapkan diri.

Seiring berkembangnya ilmu pengetahuan dan teknologi, sistem peringatan dini sudah berkembang pesat memanfaatkan *Internet of Things* (IoT). Konsep IoT diartikan sebagai sebuah kemampuan untuk menghubungkan objek-objek cerdas dan memungkinkannya untuk berinteraksi dengan objek lain, lingkungan, maupun dengan peralatan komputasi cerdas lainnya melalui jaringan internet (Iqbal *et al.*, 2016). Teknologi IoT yang tersedia saat ini sudah cukup matang dan berpotensi sangat berguna dalam situasi bencana (Sinha *et al.*, 2019). Teknologi IoT tidak dapat mencegah terjadinya banjir, tetapi dapat membantu dalam kesiapsiagaan menghadapi bencana (Samikwa *et al.*, 2020).

Pemanfaatan teknologi IoT untuk prediksi banjir umumnya menggunakan berbagai metode prediksi. Diantaranya Samikwa *et al.* (2020) melakukan prediksi banjir menggunakan IoT dan jaringan saraf tiruan. Hasilnya *prototype* sistem memprediksi banjir secara *real-time*. Yuwono *et al.* (2020), menggunakan metode logika *fuzzy* pada sistem IoT sebagai sistem pendukung keputusan penetapan status bencana banjir. Hasilnya model logika *fuzzy* tersebut dapat menentukan status bencana banjir dengan sangat baik. Adapula Mousavi *et al.*, (2021)

melakukan perancangan sistem deteksi dini banjir berbasis IoT menggunakan *machine learning*.

Pada penelitian ini akan dirancang sistem peringatan dini banjir berbasis *IoT* menggunakan metode logika *fuzzy*. Metode ini dipilih karena logika *fuzzy* adalah salah satu pendekatan yang sederhana dan fleksibel yang dapat diimplementasikan pada DAS dimana data hidrologi yang memadai tersedia dan tidak cukup baik digunakan pada model yang lebih kompleks (Perera, 2014). Pada penelitian ini akan digunakan Sistem Inferensia *Fuzzy* metode Mamdani. Metode ini merupakan metode yang paling sering dimanfaatkan peranannya karena dalam pengaplikasiannya memiliki struktur yang paling sederhana, yaitu operasi *Min-Max* atau *Max-Product* sehingga memudahkan dalam penggunaan dan penerapannya (Yuliantika dan Kartika, 2022).

## **I.2 Rumusan Masalah**

Berdasarkan latar belakang masalah yang telah diuraikan di atas, maka pokok permasalahannya sebagai berikut:

1. Bagaimana merancang bagian *hardware*, *software*, dan logika *fuzzy* Mamdani untuk sistem peringatan dini banjir berbasis *Internet of Things*?
2. Bagaimana hasil pengujian terhadap seluruh komponen sistem peringatan dini banjir?

## **I.3 Ruang Lingkup Penelitian**

Agar dalam pengerjaan tugas akhir ini dapat lebih terarah, maka ruang lingkup penelitian ini mencakup sebagai berikut:

1. Alat ini berupa sistem peringatan dini banjir yang menggunakan mikrokontroler Arduino Uno berbasis bahasa C.
2. Alat ini mengolah data masukan ketinggian air dan curah hujan yang diperoleh dari alat pengukur tinggi air dan penakar curah hujan *tipping bucket* yang dirancang.
3. Alat ini melakukan perhitungan potensi banjir menggunakan metode logika *Fuzzy Mamdani*.
4. Data ketinggian air, curah hujan, dan potensi banjir akan ditampilkan pada layar LCD dan *Website*, serta disimpan pada modul *Micro SD Card*.
5. Notifikasi Telegram dan *buzzer* akan berfungsi hanya ketika prediksi tinggi terjadinya banjir.

#### **I.4 Tujuan Penelitian**

Adapun tujuan dari penelitian ini adalah sebagai berikut:

1. Melakukan perancangan *hardwarre*, *software*, dan logika *fuzzy Mamdani* untuk sistem peringatan dini banjir berbasis *Internet of Things*.
2. Melakukan pengujian terhadap seluruh komponen sistem peringatan dini banjir.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1 Pengertian Sistem**

Menurut Hutahaean (2014:2), “Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran yang tertentu”. Menurut Mulyadi (2016:1), “Sistem pada dasarnya adalah sekelompok unsur yang erat berhubungan dengan yang lainnya, yang berfungsi bersama-sama untuk mencapai tujuan tertentu.” Menurut Maniah dan Hamidin (2017:1), mengatakan bahwa “Sistem adalah kumpulan dari elemen-elemen berupa data, jaringan kerja dari prosedur-prosedur yang saling berhubungan, sumber daya manusia, teknologi baik *hardware* maupun *software* yang saling berinteraksi sebagai satu kesatuan untuk mencapai tujuan/sasaran tertentu yang sama”.

#### **II.2 Pengertian Perancangan Sistem**

Menurut Burch dan Grudniski (2013:227) “Perancangan Sistem dapat didefinisikan sebagai gambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi.” Menurut Scott (2013:228) “Perancangan Sistem menentukan bagaimana suatu sistem akan menyelesaikan apa yang mesti diselesaikan. Tahapan ini mencakup mengkonfigurasi komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem, sehingga setelah instalasi dari sistem

akan benar-benar memuaskan rancang bangun yang telah ditetapkan pada akhir tahap analisis sistem.”

### **II.3 Peringatan Dini Banjir**

Untuk melakukan peringatan dini banjir (*flood early warning*) terdapat beberapa tahapan untuk dapat tercapainya hasil secara efektif. Tahapa-tahapan tersebut (Werner & Kwadijk, 2005):

1. *Detection.*

Pada tahap ini, data tepat waktu (*real time*) di monitor dan di proses untuk mendapatkan informasi tentang banjir yang mungkin terjadi. Informasi tersebut selanjutnya diteruskan untuk melakukan peringatan (*warning*) tanpa melalui *forecasting*. Pada tahapan ini diperlukan juga *filter* terhadap data yang ada karena data yang diperoleh dari lapangan belum tentu memiliki kualitas yang baik.

2. *Forecasting*

Tahapan ini dilakukan prakiraan terhadap curah hujan, tinggi muka air atau debit aliran banjir serta waktu datangnya banjir tersebut. Dengan diketahuinya kejadian banjir tersebut maka dapat diteruskan untuk melakukan peringatan (*warning*).

3. *Warning dan dissemination*

Tahapan ini merupakan faktor kunci sukses dalam sistem peringatan dini banjir (*flood early warning*). Tahapan ini menggunakan informasi yang diperoleh dari tahapan *detection* ataupun *forecasting*. Pihak yang

bertanggung jawab menyebarluaskan informasi tersebut untuk dapat meminimalisasi resiko yang ditimbulkannya.

#### 4. *Response*

Tanggap terhadap isu peringatan banjir, dan hal ini merupakan yang sangat penting untuk tercapainya tujuan pelaksanaan peringatan dini banjir. Jika tujuan dari peringatan dini banjir adalah untuk mengurangi kerugian materil maupun non materil, maka diperlukan personil yang tanggap secara cepat dan tepat dalam melakukan evakuasi apabila banjir benar-benar terjadi.

### **II.4 *Internet of Things***

*Internet of Things* adalah *platform* di mana perangkat berkembang secara terus menerus menjadi lebih pintar, pemrosesan menjadi cerdas, dan komunikasi semakin menjadi informatif (Ray, 2018). *Internet of Things* (IoT) mengacu pada keterhubungan yang ketat antara dunia digital dan fisik (Iera dan Moratobi, 2010) (Sterling, 2005) (ITU Internet Reports, 2005). Objek yang terhubung ke internet menjadi bagian integral dengan kehidupan kita. IoT menjadi solusi yang digunakan dalam menyelesaikan masalah di berbagai bidang, misalnya, pertanian, bisnis, kedokteran, dan komunitas perkotaan (Geetanjali *et al.*, 2019). Selama dekade terakhir istilah *Internet of Things* (IoT) telah menarik perhatian dengan memproyeksikan visi infrastruktur global objek fisik jaringan, memungkinkan konektivitas kapan saja, di mana saja dan menggunakan media apa saja (Madakam *et al.*, 2015) (Srivastava, 2006). Infrastruktur IoT diproyeksikan akan terus berkembang secara eksponensial dari beberapa tahun terakhir ini dengan

jumlah perangkat IoT yang terinstal meningkat dari kisaran 15 Milyar perangkat hingga melebihi jumlah 75 Miliar perangkat di Tahun 2025 (Fortino *et al.*, 2018). Sedangkan, menurut IoT *Analytics* memperkirakan jumlah global perangkat IoT yang terhubung yaitu 12,3 miliar endpoints yang aktif. Pada tahun 2025, kemungkinan akan ada lebih dari 27 miliar koneksi IoT (Lasse, 2020).

## II.5 Arduino Uno

Menurut Saptaji (2015:23) Arduino merupakan papan elektronik berbasis mikrokontroler ATmega yang memenuhi sistem minimum mikrokontroler agar dapat bekerja secara mandiri (*standalone controller*). Menurut Djuandi (2011:8) Komponen utama didalam papan Arduino adalah sebuah mikrokontroler 8-bit dengan merk ATmega yang dibuat oleh Atmel *corporation*. Berbagai papan Arduino menggunakan tipe Atmega yang berbeda-beda tergantung dari spesifikasinya, sebagai contoh Arduino Uno menggunakan ATmega328 sedangkan Arduino Mega 2560 yang lebih canggih menggunakan ATmega2560.

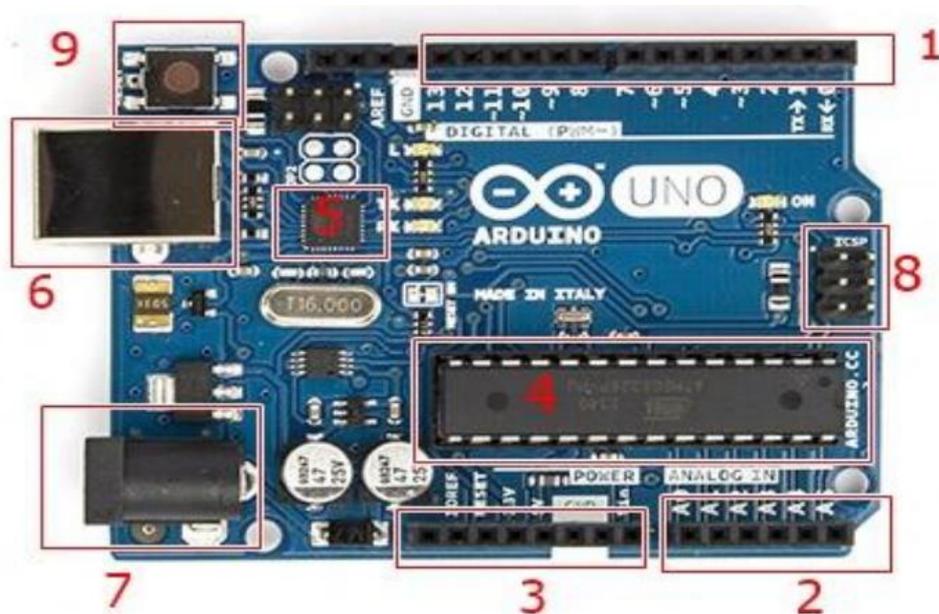
Arduino Uno adalah papan mikrokontroler berbasis ATmega328 yang Memiliki 14 pin *input/output* digital (6 di antaranya dapat digunakan sebagai *output* PWM), 6 *input* analog, osilator kristal 16 MHz, koneksi USB, colokan listrik, *header* ICSP, dan tombol *reset*. Arduino Uno memuat segala hal yang diperlukan untuk mendukung mikrokontroler; cukup sambungkan ke komputer dengan kabel USB atau nyalakan dengan adaptor AC ke DC atau baterai untuk memulai (Handoko, 2017).

## II.5.1 Komponen Arduino

Komponen Arduino dibagi menjadi 2 kategori, yaitu *hardware* dan *software*.

### II.5.1.1 Hardware

Papan arduino terdiri dari banyak komponen yang bekerja sama untuk bisa membuatnya bekerja (Louis, 2016). Saptaji (2015:38) menjelaskan pada papan arduino uno terdapat bagian-bagian seperti terlihat pada Gambar 2.1.



**Gambar 2.1** Komponen Arduino Uno (Saptaji, 2015).

#### 1. *Pin input/output* digital (diberi label '0 sampai 13')

Secara umum *pin I/O* ini adalah pin digital, yakni *pin* yang bekerja pada level tegangan digital (0V sampai 5V) baik untuk *input* atau *output*. Namun pada beberapa *pin output* analog, yang dapat mengeluarkan tegangan analog 0V sampai 5V, pin tersebut adalah pin 3, 5, 6, 9, 10 dan 11. Selain itu untuk pin 0 dan 1 juga memiliki fungsi khusus sebagai pin komunikasi serial.

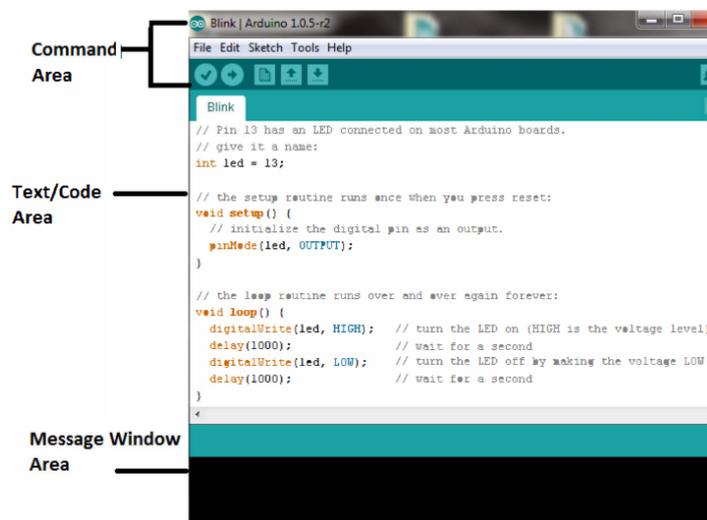
2. *Pin input* analog (diberi label 'A0 sampai A5')  
*Pin* tersebut dapat menerima *input* tegangan analog antara 0V sampai 5V, tegangan ini akan direpresentasikan sebagai bilangan 0-1023 dalam program.
3. *Pin* untuk sumber tegangan  
Kelompok *pin* ini merupakan kumpulan *pin* yang berhubungan dengan sumber tenaga, misalnya output 5V, output 3,3V, GND (2 *pin*) dan Vref (tegangan referensi untuk pembacaan ADC internal).
4. IC ATmega328: IC (*Integrated Circuit*) ini bertinfak sebagai pusat kendali pemrosesan data.
5. IC ATmega16U: IC ini diprogram untuk menangani komunikasi PC melalui *port* USB.
6. *Jack* USB  
Merupakan soket USB tipe B sebagai penghubung data serial dengan PC.
7. *Jack* Power  
Merupakan soket untuk catu daya eksternal antara 9V sampai 12V DC.
8. *Port* ICSP (*In-Circuit Serial Programming*): *Port* ini digunakan untuk memprogram Arduino tanpa *bootloader*.
9. Tombol Reset  
Digunakan untuk mereset papan mikrokontroler Arduino untuk memulai program dari awal.

### **II.5.1.2 Software**

*Software* atau perangkat lunak adalah seperangkat instruksi yang memberi tahu perangkat keras tentang apa yang harus dilakukan dan bagaimana melakukannya.

Arduino IDE (*Integrated Development Environment*) dibagi menjadi tiga bagian utama seperti terlihat pada Gambar 2.2 (Boxall, 2013).

1. *Command Area*: Ini adalah area item menu seperti *File*, *Edit*, *Sketch*, *Tools*, *Help*. Terdapat pula ikon *Verify* untuk verifikasi dan ikon *Upload* untuk mengunggah program serta *Serial Monitor* yang digunakan untuk mengirim dan menerima data antara Arduino dan IDE.
2. *Text Area*: Pada bagian inilah kode dapat ditulis menggunakan versi sederhana dari bahasa pemrograman C++ yang membuatnya lebih mudah untuk menulis program, yang juga disebut sketsa.
3. *Message Window Area*: Bagian ini menampilkan pesan dari IDE di latar hitam, terutama selama validasi kode.



**Gambar 2.2** Arduino IDE *Interface* (Sumber: Boxall, 2013).

## II.5.2 Dasar Pemrograman

Louis (2016) menjelaskan ada dua bagian utama yang selalu dimiliki setiap pemrograman Arduino di Arduino IDE, yaitu *void setup ()* dan *void loop ()*.

1. *void setup ()*:

Ini adalah fungsi pertama yang dijalankan saat Arduino mulai berfungsi. Fungsi ini hanya dijalankan sekali. Fungsi setup berisi inisialisasi setiap *pin* yang ingin kita gunakan dalam proyek kita untuk *input* dan *output*. Gambar 2.3 memperlihatkan contoh cara penulisannya.

```
void setup()
{
  pinMode(pin, INPUT);
  pinMode(pin, OUTPUT);
}
```

**Gambar 2.3** Inisialisasi *pin* yang digunakan pada fungsi “*void setup ()*”

(sumber: Louis, 2016)

Fungsi ini juga berisi inisialisasi serial monitor. Serial monitor digunakan untuk mengetahui data yang dikirim secara serial ke perangkat *peripheral* apa pun. Sebelum menggunakan variabel untuk pemrograman, perlu untuk mendefinisikannya pada fungsi “*void setup()*” seperti yang ditunjukkan pada Gambar 2.4.

```
void setup()
{
  Serial.begin(9600);
}
```

**Gambar 2.4** Inisialisasi serial monitor yang digunakan pada fungsi “*void*

*setup ()*” (sumber: Louis, 2016)

2. *void loop ()*:

Ini adalah fungsi penting selanjutnya dalam sketsa. Kode pada bagian ini harus terus dijalankan tidak seperti kode yang ditulis dalam fungsi *setup*. Contoh fungsi “*void loop ()*” seperti pada Gambar 2.5.

```
void loop()
{
  digitalWrite(pin, HIGH);
}
```

**Gambar 2.5** Fungsi “*void loop ()*” (sumber: Louis, 2016)

Pada Gambar 2.5, *digitalWrite* adalah fungsi yang menuliskan nilai tinggi atau rendah ke *pin* digital. Jika *pin* telah dikonfigurasi sebagai *output* dengan *pinMode ()*, maka voltasenya akan diatur ke nilai yang sesuai: 5V (atau 3,3V pada *board* 3,3V) untuk *HIGH* dan 0V (*ground*) untuk *LOW*. Demikian pula jika ada kebutuhan untuk menunda sketsa maka ada fungsi lain yang membuat penundaan dalam eksekusi kode seperti pada Gambar 2.6.

```
delay(1000); //delay for a second
```

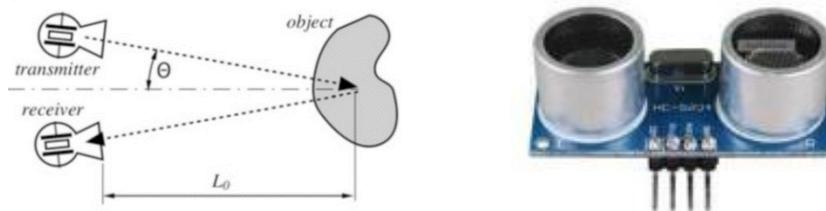
**Gambar 2.6** Fungsi penundaan (sumber: Louis, 2016)

Pada Gambar 2.6, fungsi tersebut akan menimbulkan penundaan dalam pelaksanaan program untuk jangka waktu yang ditentukan (dalam milidetik).

## II.6 Sensor Ultrasonik HC-SR04

Sensor ultrasonik merupakan perangkat yang digunakan untuk mengukur jarak dari suatu objek. Kisaran jarak yang dapat diukur sekitar 2-450 cm. Perangkat menggunakan dua pin digital untuk mengkomunikasikan jarak yang diukur. Sensor membutuhkan catu daya 5V untuk bekerja. Pada sensor ultrasonik HC-SR04 terdapat empat koneksi yaitu +5v, Gnd, *trigger* dan *Echo* (Soni dan Aman, 2018). Pemancar ultrasonik mengirimkan frekuensi suara di atas 18 kHz di udara dengan kecepatan 344 meter per detik (pada 20°C) dan penerima menerima suara

yang dipantulkan dari objek (Kelemen *et al.*, 2015). Gambar sensor dan proses kerja sensor ditunjukkan pada Gambar 2.7.



**Gambar 2.7** Sensor ultrasonik dan prinsip kerjanya (Sumber: Fraden, 2015).

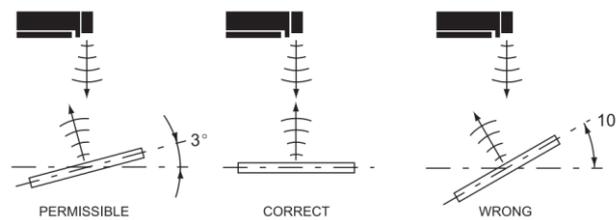
Jarak objek  $L_0$  dapat dihitung melalui kecepatan  $v$  gelombang ultrasonik di media dan sudut  $\theta$  pada persamaan 2.1.

$$L_0 = \frac{vt \cos \theta}{2} \quad (2.1)$$

Dimana  $t$  adalah waktu gelombang ultrasonik untuk melakukan perjalanan ke objek dan kembali ke penerima. Jika pemancar dan penerima diposisikan dekat satu sama lain dibandingkan dengan jarak ke objek, maka  $\cos \theta \approx 1$  (Fraden, 2015).

Sensor ultrasonik tidak mengganggu karena tidak memerlukan kontak fisik dengan targetnya. Namun, pengukurannya sangat sensitif terhadap suhu dan sudut target. Suhu dan kelembaban dapat mempengaruhi kecepatan suara di udara. Sehingga, sensor mungkin perlu dikalibrasi ulang untuk membuat pengukuran yang akurat di lingkungan baru. Variasi suhu dan arus udara dapat menciptakan batas tak terlihat yang akan memantulkan gelombang ultrasonik, jadi harus berhati-hati untuk menghindarinya. Agar gelombang yang ditransmisikan bergema kembali ke penerima, permukaan target harus tegak lurus terhadap

pemancar. Oleh karena itu, benda bulat paling mudah dideteksi karena selalu menunjukkan wajah tegak lurus. Saat membidik objek datar, pastikan sudutnya sehubungan dengan sensor tidak melebihi rentang tertentu seperti pada Gambar 2.8 (Kelemen *et al.*, 2015).



**Gambar 2.8** Gelombang yang dipantulkan dari berbagai target permukaan miring (Sumber: Wilson, 2005).

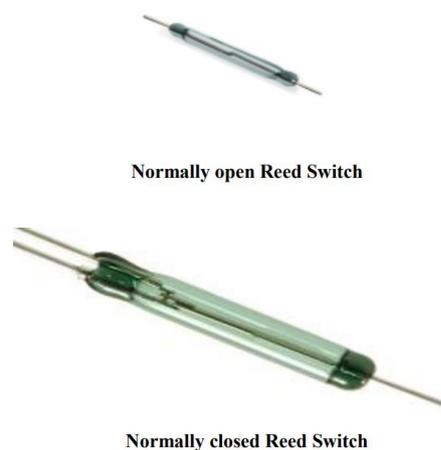
Jika target miring lebih dari +3 derajat ke sumbu normal balok, sebagian sinyal dibelokkan menjauh dari sensor dan jarak penginderaan berkurang. Namun, untuk target kecil yang terletak dekat dengan sensor, deviasi dari normal dapat dinaikkan menjadi +8 derajat. Jika target dimiringkan lebih dari kira-kira 12 derajat terhadap garis normal sumbu pancaran, seluruh sinyal dibelokkan menjauh dari sensor, dan sensor tidak akan merespon. Sinar yang mengenai target dengan permukaan kasar (seperti bahan granular) disebar dan dipantulkan ke segala arah dan sebagian energi kembali ke sensor sebagai gema yang melemah (Wilson, 2005).

## II.7 Sensor *Reed Switch*

*Reed switch* adalah saklar listrik yang dikendalikan oleh medan magnet. *Reed switch* memiliki dua kontak listrik di dalamnya yang dapat terbuka ataupun tertutup, terbuat dari bahan feromagnetik (sesuatu yang mudah dimagnetisasi

seperti besi), dilapisi dengan logam yang tahan lama seperti rhodium atau ruthenium. Kontak listrik tersebut disegel dalam selubung kaca tipis berisi gas (biasanya nitrogen) untuk menjaganya bebas dari debu dan kotoran. Terkadang kaca tipis tersebut memiliki selubung luar yang terbuat dari plastik untuk perlindungan yang lebih baik (Avinash *et al.*, 2021).

Gambar 2.9 memperlihatkan dua varietas utama sensor *Reed switch*, yaitu NO (*Normally Open*) dan NC (*Normally Close*). Kunci untuk memahami cara kerjanya adalah dengan menyadari bahwa mereka tidak hanya berfungsi sebagai jembatan listrik tetapi sebagai jembatan magnet (Avinash *et al.*, 2021).

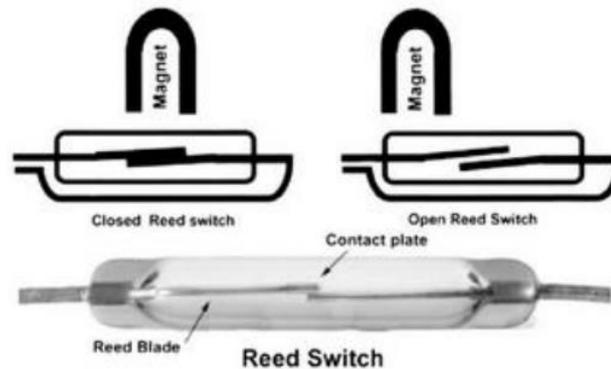


**Gambar 2.9** *Normally open Reed Switch* dan *Normally closed Reed Switch*

(Sumber: Avinash *et al.*, 2021).

*Reed switch* adalah saklar elektromagnetik yang digunakan untuk mengontrol aliran listrik dalam suatu rangkaian. Cara kerja sensor *Reed switch* terlihat pada Gambar 2.10. *Reed switch* secara efektif berfungsi seperti gerbang atau jembatan pada sirkuit listrik sehingga ketika kedua lempengan atau kontak listrik bersentuhan, listrik dapat mengalir. Tidak seperti sakelar mekanis, sakelar ini

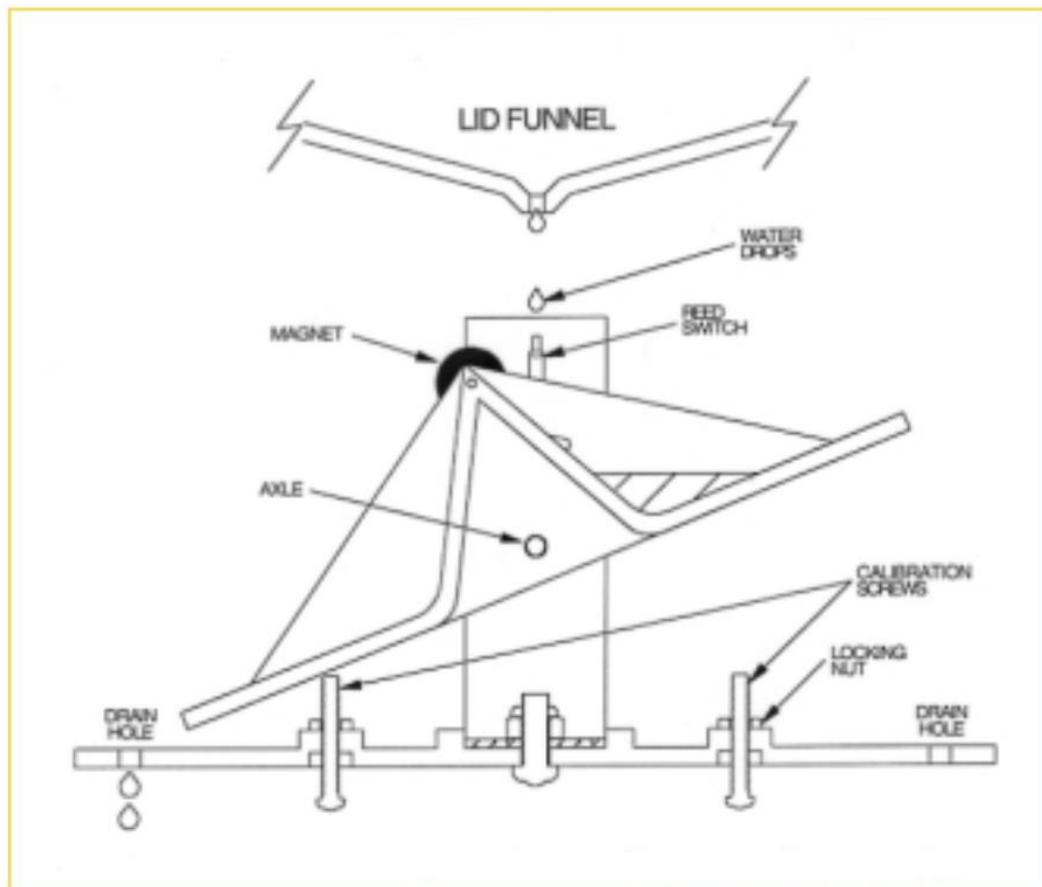
tidak memerlukan sesuatu atau seseorang untuk menghidupkan atau mematikannya secara fisik, sakelar dikendalikan sepenuhnya oleh medan magnet yang tidak terlihat (Tun *et al.*, 2020).



**Gambar 2.10** Prinsip kerja *Reed Switch* (Sumber: Tun *et al.*, 2020).

## II.8 *Tipping Bucket*

Prinsip kerja *Tipping Bucket* mengingatkan pada permainan jungkat-jungkit seperti pada Gambar 2.11. Air hujan yang terkumpul pada bagian penutup alat pengukur menetes ke dalam bejana melalui lubang berdiameter kecil di bagian bawah penutup yang berbentuk corong (Awtrey, 1999). Proses mekanis pada *Tipping Bucket* dapat dipisahkan menjadi dua tahap, yaitu: tahap stasioner dan tahap rotasi bejana. Pertama, bejana didiamkan sampai air hujan yang terisi lebih besar dari volume nominal di salah satu sisi bejana. Saat permukaan air meningkat, pusat massa air secara bertahap menjauh dari poros. Jika air mengisi wadah jungkit dengan curah hujan yang tinggi, jungkit akan terbalik dan airnya dibuang. Ada dua bejana yang menampung air hujan secara bergantian. Setiap gerakan bejana direkam secara mekanis pada mikrokontroler (*counter*) yang menggunakan sensor *reed switch* (Liu *et al.*, 2019).



**Gambar 2.11** Prinsip Kerja *Tipping Bucket* (Sumber: Awtrey, 1999).

Kalibrasi pada *Tipping Bucket* dapat dilakukan dengan memutar sekrup sedemikian rupa. Memutar sekrup untuk menaikkan bejana lebih tinggi di atas alas mengurangi jumlah air yang dibutuhkan untuk menyebabkan penuangan dan dengan demikian meningkatkan hitungan. Sebaliknya, memutar sekrup untuk menurunkan bejana lebih dekat ke alas akan mengurangi hitungan. Mencoba mengkalibrasi unit dengan laju tetesan yang terlalu cepat dapat menyebabkan kesalahan yang tidak dapat diterima (Awtrey, 1999). Menurut Fankhauser (1997), resolusi kedalaman TBR 0,2 – 0,3 mm per tip tampaknya memenuhi persyaratan dalam hidrologi perkotaan. TBR dengan resolusi kedalaman hingga 0,254 mm

dapat digunakan dalam hidrologi perkotaan tanpa menimbulkan kesalahan yang signifikan (Fankhauser, 1998).

## II.9 *Liquid Crystal Display (LCD)*

Penampil (*display*) elektronik adalah salah satu komponen elektronika yang berfungsi untuk menampilkan angka, huruf, atau simbol-simbol lainnya. LCD (*Liquid Crystal Display*) adalah salah satu *display* elektronika yang umum digunakan. LCD dibuat dengan CMOS *logic* yang bekerja dengan tidak menghasilkan cahaya melainkan memantulkan cahaya yang ada di sekitarnya terhadap *front-lit* atau mentransmisikan cahaya dari *back-lit*. Jumlah karakter yang dapat ditampilkan oleh sebuah LCD tergantung dari spesifikasi yang dimiliki (Subagyo & Suprianto, 2017).

LCD (*Liquid Crystal Display*) 16x2 adalah suatu *display* dari bahan cairan kristal yang pengoperasiannya menggunakan sistem dot matriks. LCD 16x2 berarti 16 karakter kali 2 baris sehingga LCD 16x2 dapat menampilkan 32 karakter (Andrianto & Darmawan, 2016). Bentuk LCD pada Gambar 2.12.

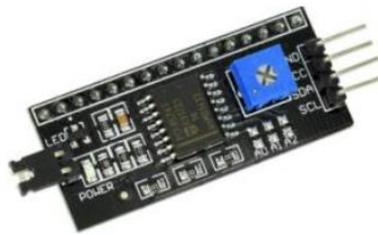


**Gambar 2.12** LCD 16x2 *Display* (Sumber: Andrianto & Darmawan, 2016)

## II.10 I2C LCD

I2C LCD adalah modul LCD yang dikendalikan secara *serial sinkron* dengan protokol I2C/IIC (*Inter Integrated Circuit*) atau TWI (*Two Wire Interface*). Modul

LCD pada normalnya dikendalikan secara paralel baik untuk jalur data maupun kontrolnya. Namun jalur paralel akan memakan banyak pin di sisi controller (misal Arduino, komputer, dll). Setidaknya akan embutuhkan 6 atau 7 *pin* untuk mengendalikan sebuah modul LCD. Dengan demikian untuk sebuah controller yang harus mengendalikan banyak I/O, menggunakan jalur paralel adalah solusi yang kurang tepat (Suryantoro dan Budiyanto, 2019). Modul I2C *converter* diperlihatkan pada Gambar 2.13.



**Gambar 2.13** Modul I2C LCD (Sumber: Suryanto & Budiyanto, 2019)

## II.11 NodeMCU

NodeMCU ESP-8266 merupakan sebuah *opensource platform IoT* dan pengembangan kit dari ESP8266 yang menggunakan bahasa pemrograman eLua untuk membantu pembuat dalam membuat produk IoT atau dengan memakai *sketch* dengan arduino IDE. ESP8266 sendiri merupakan chip WiFi dengan protokol TCP/IP yang lengkap. NodeMCU ESP8266 juga memiliki board yang berukuran sangat kecil yaitu panjang 6.83 cm, lebar 2.54 cm. Node MCU ESP8266 menawarkan kemudahan untuk melakukan pengembangan perangkat berbasis internet karena sudah dilengkapi modul komunikasi *Wireless Firewall* (WiFi). NodeMCU dapat dianalogikan sebagai *board* arduino-nya ESP8266. Program ESP8266 sedikit merepotkan karena diperlukan beberapa titik *wiring*

serta tambahan modul *USB to serial* untuk mengunduh program. Namun NodeMCU ESP8266 telah menyatukan ESP8266 ke dalam sebuah *board* yang minimalis dengan berbagai fitur layaknya mikrokontroler yang mempunyai akses terhadap WiFi juga chip komunikasi *USB to serial*. Sehingga untuk memprogramnya hanya diperlukan kabel data USB (Siswanto, 2020).

Secara umum ada tiga produsen NodeMCU yang produknya kini beredar di pasaran: Amica, DOIT dan Lolin/WeMos. Dengan beberapa varian *board* yang diproduksi yakni V1, V2 dan V3. Generasi kedua atau V2 adalah pengembangan dari versi sebelumnya (V1), dengan *chip* yang ditingkatkan dari sebelumnya ESP-12 menjadi ESP-12E dan IC *USB to Serial* diubah dari CHG340 menjadi CH2102. Berikut ini ialah tampilan NodeMCU v1.0 yang dapat dilihat pada Gambar 2.14.



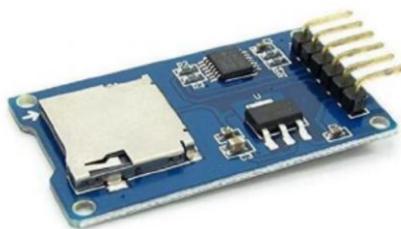
**Gambar 2.14** NodeMCU Devkit v1.0 (Versi 2) (Sumber: Parihar, 2019)

## II.12 Modul Micro SD Card

Proyek yang melibatkan audio, video, grafik, dan pencatatan data, penting untuk memiliki opsi penyimpanan yang dapat dilepas. Kartu SD atau microSD adalah pilihan yang ideal karena dapat menyimpan gigabyte data dalam ruang yang sangat kecil, bahkan lebih kecil dari ukuran sebuah koin. Selain itu kartu-kartu tersebut tersedia di hampir semua toko elektronik. Banyak komputer juga

dilengkapi dengan pembaca kartu SD atau microSD, memudahkan mentransfer data antara berbagai perangkat, seperti misalnya memindahkan data dari data *logger* Arduino ke komputer. Hampir setiap *library* Arduino untuk kartu SD, mengharuskan kartu SD diformat ke FAT16 atau FAT32. Kartu SD hadir dalam dua varian populer yaitu microSD dan SD. Antarmuka, kode, struktur, dll semuanya sama. Satu-satunya perbedaan adalah ukurannya. MicroSD jauh lebih kecil dalam ukuran fisik.

Komunikasi antara mikrokontroler dan *SD card* dapat dilakukan dengan 2 cara yaitu mode SPI dan mode SDIO. Mode SDIO lebih cepat tetapi jauh lebih kompleks. Sebagai gantinya, setiap kartu SD memiliki mode SPI (kecepatan lebih rendah) namun mudah digunakan oleh mikrokontroler manapun (Lattice Semiconductor Corporation, 2019). SPI merupakan protokol komunikasi dengan *interfacing* yang sederhana dan kecepatan yang dimilikinya masih memungkinkan untuk terjadinya komunikasi transfer data dengan mudah (Choudhury dkk, 2014).



**Gambar 2.15** Modul Micro SD Card (Mnati dkk, 2021)

### **II.13 Modul RTC DS1302 (*Real Time Clock*)**

RTC (*Real Time Clock*) merupakan *chip* IC yang mempunyai fungsi menghitung waktu yang dimulai dari detik, menit, jam, hari, tanggal, bulan, hingga tahun dengan akurat. Untuk menjaga atau menyimpan data waktu yang telah di-*ON*-kan

pada modul terdapat sumber catu daya sendiri yaitu baterai jam kancing, serta keakuratan data waktu yang ditampilkan digunakan osilator kristal eksternal. Contoh yang dapat ditemui dalam kehidupan sehari-hari yaitu pada *motherboard* PC yang biasanya letaknya berdekatan dengan *chip* BIOS. Difungsikan guna menyimpan sumber informasi waktu terkini sehingga jam akan tetap *up to data* walaupun komputer tersebut dimatikan (Suryanto, 2019). Berikut bentuk RTC (*Real Time Clock*) pada Gambar 2.16.



**Gambar 2.16** Modul RTC DS1302 (Sumber: Ahmed & Wei, 2014)

## II.14 *Buzzer*

Menurut beberapa ahli Menyatakan “*Buzzer* adalah komponen elektronik yang dapat mengubah sinyal listrik menjadi getaran suara.” Bunyi berasal dari membran yang memiliki kumparan (Manurung *et al.*, 2021). Pada umumnya, *Buzzer* yang merupakan sebuah perangkat audio ini sering digunakan pada rangkaian anti-maling, alarm pada jam tangan, bel rumah, peringatan mundur pada truk dan perangkat peringatan bahaya lainnya.

*Buzzer* yang sering ditemukan dan digunakan adalah *Buzzer* yang berjenis *Piezoelectric*, hal ini dikarenakan *Buzzer Piezoelectric* memiliki berbagai kelebihan seperti lebih murah, relatif lebih ringan dan lebih mudah dalam menggabungkannya ke rangkaian elektronika lainnya (Soleman *et al.*, 2019).

*Buzzer* bekerja pada tegangan DC berbanding terbalik dengan *speaker* yang menggunakan tegangan AC. Setiap kumparan yang terpasang pada diafragma yang bergerak maju mundur akan membuat udara bergetar sehingga menghasilkan suara atau *buzzer* contoh dapat dilihat pada Gambar 2.17 dibawah ini (Manurung *et al.*, 2021).



**Gambar 2.17** *Buzzer* (Sumber: Manurung dkk, 2021)

## II.15 Logika *Fuzzy*

Logika *Fuzzy* merupakan salah satu komponen pembentuk *soft computing*. Logika *fuzzy* pertama kali diperkenalkan oleh Prof. Lotfi A. Zadeh pada tahun 1965. Dasar logika *fuzzy* adalah teori himpunan *fuzzy*. Pada teori himpunan *fuzzy*, peranan derajat keanggotaan sebagai penentu keberadaan elemen dalam suatu himpunan sangatlah penting. Nilai keanggotaan atau derajat keanggotaan atau *membership function* menjadi ciri utama dari penalaran dengan logika *fuzzy* tersebut (Kusumadewi dan Purnomo, 2010).

Menurut Caraka *et al.* (2015), logika *fuzzy* adalah metodologi sistem kontrol pemecahan masalah yang cocok diterapkan pada sistem, mulai dari sistem yang sederhana sampai sistem yang rumit atau kompleks. Metodologi ini dapat diterapkan pada perangkat keras, perangkat lunak, atau kombinasi keduanya.

### II.15.1 Himpunan Tegas (*Crisp*) dan Himpunan *Fuzzy*

Himpunan tegas (*crisp*) disebut juga sebagai himpunan klasik. Himpunan ini memiliki batasan yang jelas, tidak ada ketidakpastian dalam menentukan atau mengalokasikan batasan dari himpunan. Keanggotaan pada himpunan ini memiliki dua kemungkinan yaitu (Ross, 2004):

1. Bernilai 1 yang berarti suatu item menjadi anggota dalam himpunan.
2. Bernilai 0 yang berarti bahwa suatu item tidak menjadi anggota dalam suatu himpunan.

Pada himpunan *fuzzy* batasan anggota bersifat tidak jelas, samar dan ambigu (Castillo dan Melin, 2008). Keanggotaan dari anggota himpunan *fuzzy* bernilai antara 0 hingga 1. Himpunan *fuzzy* dinotasikan sebagai berikut:

$$A = \{(x, \mu_A(x) \mid x \in X)\} \quad (2.2)$$

Himpunan *fuzzy* memiliki 2 atribut yaitu (Kusumadewi dan Purnomo, 2013):

1. Linguistik

Penamaan suatu grub yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami seperti muda, parobaya, tua. Linguistik ini merupakan himpunana *fuzzy*.

2. Numeris

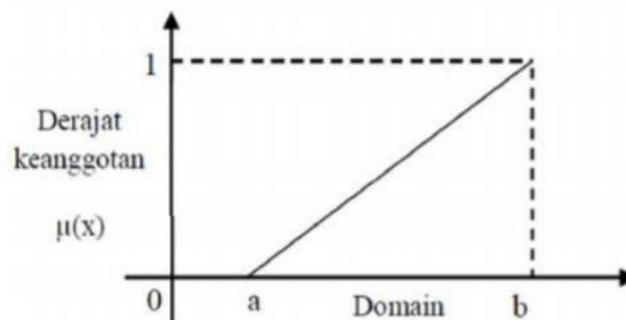
Nilai (angka) yang menunjukkan suatu variabel. Variabel *fuzzy* merupakan variabel yang hendak dibahas dalam suatu sistem *fuzzy*, misalkan: usia, berat, temperatur dan lain-lain.

## II.15.2 Fungsi Keanggotaan

Fungsi keanggotaan (*membership function*) merupakan suatu kurva yang menunjukkan pemetaan titik-titik *input* data ke dalam nilai keanggotaannya (disebut derajat keanggotaan) yang memiliki interval antara 0 hingga 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi (Widaningsih, 2017). Representasi bentuk kurva *membership function* dapat dibagi sebagai berikut (Saputra, 2020).

### 1. Representasi Linear Naik

Pada representasi linear, pemetaan input ke derajat keanggotaannya digambarkan sebagai sebuah garis lurus seperti pada Gambar 2.18.



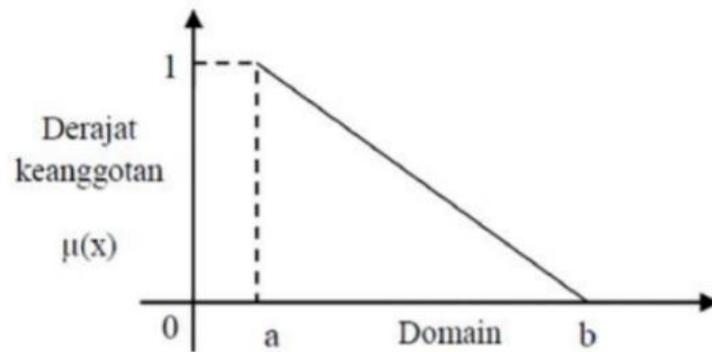
**Gambar 2.18** Representasi Linear Naik (Sumber: Saputra, 2020)

Fungsi keanggotaan:

$$\mu[x] = \begin{cases} 0; & x \leq a \\ (x - a)/(b - a); & a \leq x \leq b \\ 1; & x \geq b \end{cases}$$

### 2. Representasi Linear Turun

Representasi ini merupakan kebalikan dari representasi linear naik seperti yang terlihat pada Gambar 2.19.



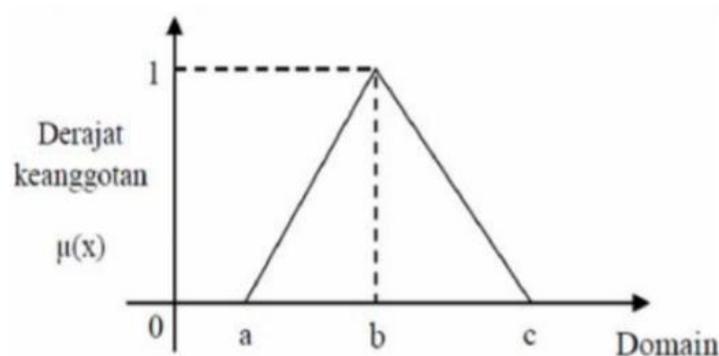
**Gambar 2.19** Representasi Linear Turun (Sumber: Saputra, 2020)

Fungsi keanggotaan:

$$\mu[x] = \begin{cases} 1; & x \leq a \\ (a - x)/(b - a); & a \leq x \leq b \\ 0; & x \geq b \end{cases}$$

### 3. Representasi Kurva Segitiga

Representasi kurva segitiga pada dasarnya merupakan gabungan antara linear naik dan linear turun seperti yang terlihat pada Gambar 2.20.



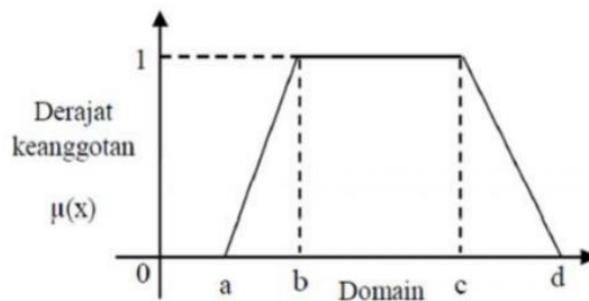
**Gambar 2.20** Representasi Kurva Segitiga (Sumber: Saputra, 2020)

Fungsi keanggotaan:

$$\mu[x] = \begin{cases} 0; & x \leq a \text{ atau } x \geq c \\ (x - a)/(b - a); & a \leq x \leq b \\ (c - x)/(c - b); & b \leq x \leq c \end{cases}$$

#### 4. Representasi Kurva Trapesium

Representasi kurva trapesium memiliki domain lebih luas dari representasi kurva segitiga seperti yang terlihat pada Gambar 2.21.



**Gambar 2.21** Representasi Kurva Trapesium (Sumber: Saputra, 2020)

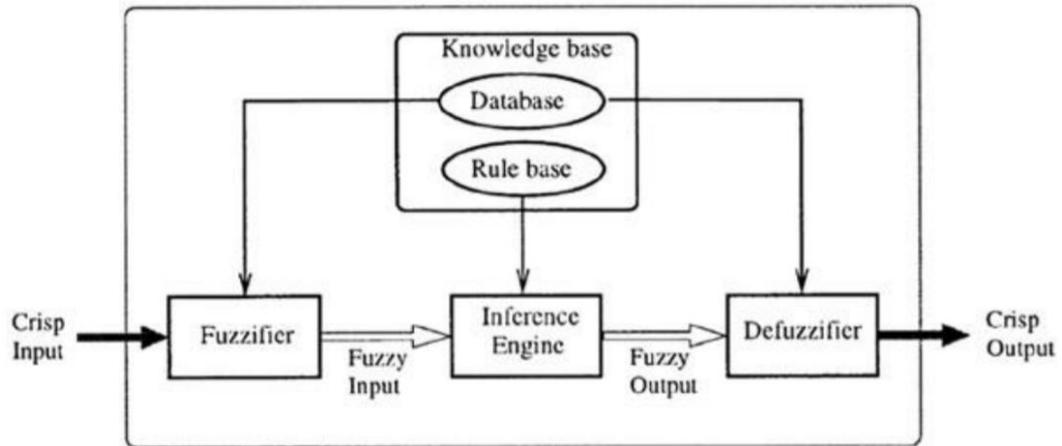
Fungsi keanggotaan:

$$\mu[x] = \begin{cases} 0; & x \leq a \text{ atau } x \geq d \\ (x - a)/(b - a); & a \leq x \leq b \\ 1; & b \leq x \leq c \\ (d - x)/(d - c); & c \leq x \leq d \end{cases}$$

#### II.15.3 Sistem Inferensia *Fuzzy*

Sistem inferensia *fuzzy* merupakan sistem yang menggunakan teori himpunan *fuzzy* untuk memetakan input menjadi output. Pada Gambar 2.22 memperlihatkan struktur dasar dari sistem inferensia *fuzzy* yang terdiri dari tiga komponen konseptual yaitu (Castillo dan Melin, 2008):

1. Aturan dasar, terdiri dari aturan *fuzzy* yang telah diseleksi.
2. Basis data (kamus), dimana mendefinisikan fungsi keanggotaan yang digunakan pada aturan *fuzzy*.
3. Mekanisme penalaran, yang melakukan prosedur inferensia pada aturan dan diberikan fakta untuk mendapatkan output yang wajar atau kesimpulan.



**Gambar 2.22** Struktur Dasar Sistem Inferensia *Fuzzy* (Sumber: Castillo & Melin, 2008)

Secara umum sistem inferensia *fuzzy* mengimplementasikan pemetaan non-linear dari ruang input ke ruang output. Pemetaan ini disampaikan oleh beberapa aturan *if-then*. Dalam sistem inferensia *fuzzy* terdapat beberapa metode diantaranya terdapat tiga metode yang sering digunakan. Perbedaan dari ketiga metode tersebut terletak pada perbedaan konsekuensi aturan *fuzzy*, agregasi dan prosedur defuzzifikasi. Ketiga metode tersebut yaitu metode Tsukamoto, Mamdani, dan Sugeno.

#### II.15.4 Metode Mamdani

Metode Mamdani pertama kali dikemukakan oleh Ebrahim Mamdani pada tahun 1975, metode ini digunakan untuk mengendalikan mesin uap dan boiler yang dikombinasikan dengan sekumpulan aturan kontrol linguistik yang diperoleh dari operator yang berpengalaman (Castillo dan Melin, 2008). Metode ini merupakan metode yang paling sering dimanfaatkan peranannya karena dalam pengaplikasiannya memiliki struktur yang paling sederhana, yaitu operasi MIN-

MAX atau MAX-PRODUCT sehingga memudahkan dalam penggunaan dan penerapannya (Yuliantika dan Kartika, 2022). Untuk mendapatkan *output*, diperlukan beberapa tahapan (Kastina K. dan Silalahi M. 2016) (Yudihartanti, 2011):

1. Pembentukan himpunan *fuzzy*, baik variabel *input* maupun variabel *output* dibagi menjadi satu atau lebih himpunan *fuzzy*.
2. Fuzzyfikasi, yaitu menentukan derajat keanggotaan variabel *input*.
3. Operasi logika *fuzzy*, perlu dilakukan jika bagian *antecedent* lebih dari satu pernyataan melakukan operasi-operasi logika *fuzzy*. Hasil akhir dari operasi ini adalah derajat kebenaran *antecedent* yang berupa bilangan tunggal. Operator *fuzzy* untuk melakukan operasi *and* dan *or* bisa dibuat sendiri.
4. Implikasi: menerapkan metode implikasi untuk menentukan bentuk akhir *fuzzy set* keluaran. *Consequent* atau keluaran dari aturan *fuzzy* ditentukan dengan mengisikan kumpulan *fuzzy* keluaran ke variabel keluaran. Fungsi implikasi yang digunakan adalah Min.
5. Agregasi atau komposisi aturan: yaitu proses mengkombinasikan keluaran semua aturan *if - then* menjadi sebuah kumpulan *fuzzy* tunggal menggunakan fungsi Max. Apabila digunakan fungsi implikasi min maka metode agregasi ini disebut dengan nama max-min atau Mamdani. Fungsi max (*maximum*) secara umum dapat dituliskan:

$$\mu_{sf}[X_i] = \max(\mu_{sf}[X_i], \mu_{kf}[X_i]) \quad (2.3)$$

dengan:

$\mu_{sf}[Xi] =$  nilai keanggotaan solusi *fuzzy* sampai aturan ke *i*

$\mu_{kf}[Xi] =$  nilai keanggotaan konsekuen *fuzzy* aturan ke *i*

6. Penegasan (defuzzyfikasi): *input* dari proses ini adalah suatu himpunan *fuzzy* yang diperoleh dari komposisi aturan-aturan *fuzzy*, sedangkan outputnya adalah bilangan pada domain himpunan *fuzzy* tersebut. Salah satu metode untuk defuzzifikasi yaitu metode *centroid* atau *composite moment*. Dimana pada metode ini, solusi *crisp* diperoleh dengan cara mengambil titik pusat daerah *fuzzy*. Secara umum dirumuskan:

$$\mu(x) = \frac{\int_a^b x\mu(x)dx}{\int_a^b \mu(x)dx} \quad (2.4)$$

atau

$$\mu(x) = \frac{\sum_{i=1}^n x_i\mu(x_i)}{\sum_{i=1}^n \mu(x_i)} \quad (2.5)$$

## II.16 Database

*Database* atau basis data digunakan untuk menyimpan kumpulan informasi atau data yang tersimpan di dalam komputer secara sistematis sehingga dapat diperiksa dengan menggunakan suatu program komputer untuk mendapatkan informasi atau data dari database tersebut (Solochin, 2016). *Database* (Basis Data), tempat penyimpanan data yang besar dimana dapat digunakan oleh banyak pengguna. Seluruh item basis data tidak lagi dimiliki oleh satu departemen, melainkan menjadi sumber daya perusahaan yang dapat digunakan bersama-sama (Indrajani, 2011). Perangkat lunak yang digunakan untuk mengelola dan memanggil (*query*) basis data disebut sistem manajemen basis data adalah *Database Management System* (DBMS) (Ismail *et al.*, 2019).

## II.17 MySQL

MySQL adalah perangkat lunak sistem manajemen basis data SQL. Manajemen *Database* sistem ini *multithreaded* dan *multiuser* dengan sekitar 6 juta instalasi di seluruh dunia. MySQL juga merupakan implementasi dari *Relational Database Management System* (RDBMS), yang didistribusikan secara gratis di bawah *General Public License* (GPL) (Sari dan Abdilah, 2015).

Dengan adanya DBMS ini, pengguna akan lebih mudah mengontrol dan memanipulasi data yang ada. Adapun kelebihan dari MySQL, antara lain (Madakam dan Lake, 2015):

- a. *Free* (Bebas di download)
- b. Fleksibel dengan berbagai pemrograman
- c. Mudah untuk manajemen database
- d. Stabil dan tangguh
- e. Security yang baik
- f. Dukungan dari banyak komunitas
- g. Dapat mendukung transaksi
- h. Perkembangan perangkat lunak yang cukup cepat.

## II.18 *Hypertext Markup Language* (HTML)

*Hypertext Markup Language* (HTML) adalah bahasa standar yang digunakan untuk menampilkan konten pada halaman *website*. Fungsi-fungsi yang dapat dilakukan dengan bahasa programan HTML adalah:

1. Mengatur serta mendesain tampilan isi halaman *website*.

2. Membuat tabel pada halaman *website*.
3. Mempublikasikan halaman *website* secara online.
4. Membuat *form* yang dapat menjadi *input* serta menangani registrasi dan transaksi via *website*.
5. Menampilkan area gambar pada *browser*.

Penanda perintah pada bahasa pemrograman *Hypertext Markup Language* (HTML) disebut dengan *tag*. *Tag* digunakan untuk menentukan tampilan dari dokumen HTML. *Tag* HTML berfungsi untuk mendefinisikan bahwa isi dalam file tersebut adalah dokumen. Elemen *head* merupakan kepala dari dokumen HTML. Elemen *head* digunakan untuk menempatkan identitas file, sedangkan *tag body* digunakan untuk menentukan konten yang akan ditampilkan pada halaman *website* (Hidayatullah dan Kawistara, 2015). Struktur dari pemrograman HTML dapat dilihat pada Gambar 2.23.

```

<html>
<head> ....</head>
<body>
.....
.....
</body>
</html>

```

**Gambar 2.23** Struktur program HTML (Sumber: Mariko, 2019)

## **II.19 *Hipertext Preprocessor* (PHP)**

Menurut Bimo Sunarfrihantono, PHP merupakan merupakan bahasa pemrograman *server-side scripting* yang menyatu dengan HTML untuk membuat halaman web dinamis. Maksud dari *server-side scripting* adalah sintaks dan perintah-perintah yang diberikan akan sepenuhnya dijalankan di *server* tetapi

disertakan pada dokumen HTML. Pembuatan web ini merupakan kombinasi antara PHP dan HTML sebagai pembangunan halaman web. Ketika seorang pengguna internet akan membuka suatu situs yang menggunakan fasilitas *server-side scripting* PHP, maka terlebih dahulu *server* yang bersangkutan akan memproses semua perintah PHP di *server* kemudian mengirimkan hasilnya dalam format HTML ke *web browser* pengguna internet tadi. Dengan demikian seorang pengguna internet tidak dapat melihat kode program yang ditulis dalam PHP sehingga keamanan dari halaman web menjadi lebih terjamin (Sunarfrihantono, 2002).

## II.20 JavaScript

JavaScript adalah bahasa pemrograman berbentuk kumpulan *script* yang berjalan pada suatu dokumen HTML (Kustiyahningsih dan Anamisa, 2011). Dengan JavaScript kita dapat dengan mudah membuat sebuah halaman web yang interaktif (Hardjono, 2006). JavaScript dapat menyempurnakan tampilan dan sistem pada halaman *web-based application* yang dikembangkan. Adapun karakteristik dari bahasa pemrograman JavaScript adalah:

1. Bahasa pemrograman berjenis *high-level* programming.
2. Bersifat *client-side*.
3. Berorientasi pada objek.
4. Bersifat *loosely typed*.

Aplikasi atau *tools* standar yang digunakan dalam mengimplementasikan JavaScript adalah:

1. *Software text-editor*, seperti: Notepad++, Adobe Dreamweaver dan NetBeans.
2. *Web browser*, seperti: Mozilla Firefox, Google Chrome, Internet Explorer, OperaMini, Safari, dan lain sebagainya.
3. HTML.

Kelemahan dari bahasa pemrograman JavaScript ini sendiri adalah JavaScript tidak didukung oleh *browser* versi lama (Enterprise, 2017).

## **II.21 Cascading Style Sheets (CSS)**

CSS berisi rangkaian instruksi yang menentukan bagaimana suatu *text* akan tertampil di halaman web. Perancangan desain *text* dapat dilakukan dengan mendefinisikan *fonts* (huruf), *colors* (warna), *margins* (ukuran), *background* (latar belakang), *font sizes* (ukuran font) (Lay, 2017). Supaya aspek artistik suatu halaman web lebih optimal dan bisa dikelola dengan pola *template* tertentu, diperlukan CSS. Peran CSS untuk memperindah tampilan halaman web, khususnya untuk pengaturan format yang tidak dapat dilakukan hanya dengan mengandalkan HTML (Maryanto, 2009).