

# Prototype Sistem Monitoring Ketinggian Air Menggunakan Telegram

Muannas Nur Abdullah<sup>1</sup>, Firman Saputra<sup>2</sup>, Rasendriya Bagas Utomo<sup>3</sup>, Rasyid Ahmad Mustofa<sup>4</sup>, Herliyani Hasanah<sup>5</sup>

<sup>1</sup> Fakultas Ilmu Komputer, Program Studi D3 Teknik Komputer, Universitas Duta Bangsa, Surakarta, Indonesia  
Email: <sup>1</sup> muannasnur122@gmail.com, <sup>2</sup> firmansptr68@gmail.com, <sup>3</sup> rasendryabagas2004@gmail.com,

<sup>5</sup> rasyidmusthofa9347@gmail.com, <sup>6</sup> herliyani\_hasanah@udb.ac.id

\*) [muannasnur122@gmail.com](mailto:muannasnur122@gmail.com)

**Abstrak**– Sistem monitoring ketinggian air merupakan aspek penting dalam memantau kondisi banjir dan mengambil tindakan yang diperlukan. Dalam penelitian ini, kami mengusulkan pengembangan sistem monitoring ketinggian air yang menggunakan Telegram sebagai platform komunikasi. Tujuan utama dari penelitian ini adalah untuk menyediakan pengguna dengan informasi real-time tentang ketinggian air melalui pesan instan yang disediakan oleh Telegram. Metode yang digunakan dalam penelitian ini melibatkan pemasangan sensor ketinggian air pada lokasi yang strategis. Sensor tersebut mengukur ketinggian air dan mengirimkan data yang terkait ke sebuah mikrokontroler. Mikrokontroler kemudian mengirimkan data tersebut ke perangkat pengirim pesan Telegram melalui koneksi internet.

**Kata Kunci:** Monitoring, Telegram, Real-Time, Mikrokontroler, Internet

**Abstract**– The water level monitoring system is an important aspect in monitoring flood conditions and taking the necessary actions. In this research, we propose to develop a water level monitoring system that uses Telegram as a communication platform. The main aim of this study is to provide users with real-time information about the water level via instant messages provided by Telegram. The method used in this research involves installing water level sensors at strategic locations. The sensor measures the water level and sends related data to a microcontroller. The microcontroller then sends the data to the Telegram messaging device via an internet connection.

**Keywords:** Monitoring, Telegram, Real-Time, Microcontroller, Internet

## 1. PENDAHULUAN

Dalam era di mana teknologi terus berkembang, sistem monitoring yang efisien dan dapat diakses dengan mudah sangat penting untuk menghadapi tantangan terkait ketinggian air, seperti banjir atau tingginya permukaan air sungai. Prototype ini dirancang untuk memanfaatkan kepopuleran aplikasi Telegram sebagai platform komunikasi yang luas digunakan. Dengan menggunakan Telegram sebagai antarmuka, pengguna dapat menerima pembaruan ketinggian air secara real-time dan memberikan tanggapan langsung melalui pesan teks atau perintah yang sudah diprogram. Dalam prototype ini, sistem monitoring ketinggian air dikombinasikan dengan sensor yang terpasang di tempat yang tepat, seperti bendungan atau sungai. Sensor ini secara terus-menerus mengukur ketinggian air dan mengirimkan data tersebut ke server yang terhubung dengan aplikasi Telegram. Kemudian, data yang dikirim oleh sensor diolah oleh sistem dan dikirimkan ke pengguna melalui pesan teks melalui Telegram. Pengguna dapat melihat ketinggian air saat ini, melacak tren, atau menerima peringatan jika ketinggian air mencapai tingkat bahaya. Mereka juga dapat mengirimkan perintah tertentu melalui aplikasi Telegram untuk meminta informasi tambahan atau mengambil tindakan tertentu, seperti membagikan informasi dengan pihak berwenang terkait. Prototipe ini bertujuan untuk memberikan aksesibilitas dan keterlibatan yang lebih besar dalam pemantauan ketinggian air, sehingga dapat meningkatkan kesiapsiagaan dan respon terhadap situasi darurat terkait air. Dengan mengintegrasikan teknologi komunikasi yang sudah ada, seperti Telegram, prototype ini menawarkan solusi yang praktis, efisien, dan mudah digunakan bagi masyarakat luas. Harapannya, dengan pengembangan lebih lanjut dan implementasi prototype ini, kita dapat meningkatkan sistem monitoring ketinggian air secara keseluruhan, mengurangi dampak buruk banjir, dan meningkatkan keamanan dan keberlanjutan lingkungan kita.

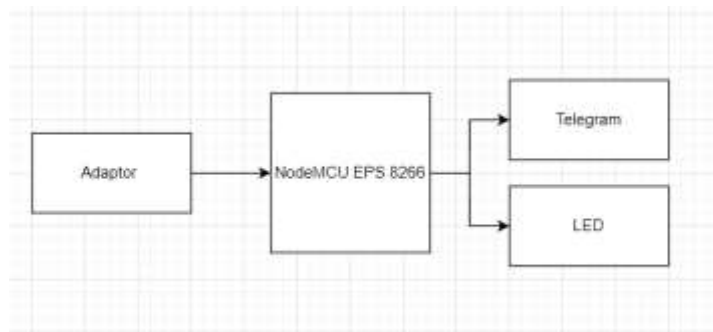
## 2. METODE PENELITIAN

Pembahasan pada bagian ini meliputi perangkat keras, perangkat lunak, dan pengujian perangkat.

### 2.1 Rancangan Sistem

Skema rancangan sistem monitoring ketinggian air. Rancangan sistem monitoring ini terdiri atas bagian-bagian berikut.

1. Bagian input, terdiri dari adaptor. Adaptor berfungsi sebagai suplai tegangan
2. Bagian proses, menggunakan mikrokontroler NodeMCU ESP8266 berfungsi memproses data dan data akan diolah menjadi data output.
3. Bagian output, terdiri dari lampu LED sebagai indikator ketinggian air dan telegram sebagai penerima data output sekaligus monitoring ketinggian air.



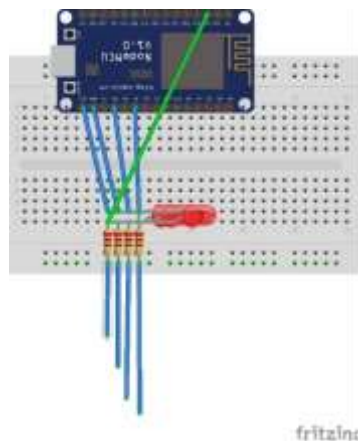
Gambar 1. Blok Diagram Sistem Monitoring Ketinggian Air NodeMCU

### 2.2 Prototype

Prototype merupakan alat yang memberikan ide untuk pembuat dan pengguna tentang sistem akan berfungsi secara lengkap, Prototyping merupakan proses pembuatan model sederhana yang memiliki gambaran dasar tentang program serta pengujian awal. Prototyping memberikan kesempatan bagi pembuat dan pengguna agar saling berinteraksi selama proses pembuatan, sehingga pembuat dapat dengan mudah memodelkan perangkat yang akan dibuat. Prototyping merupakan salah satu metode pengembangan yang banyak digunakan.

### 2.3 Monitoring Ketinggian Air

Monitoring ketinggian air merupakan kegiatan yang dilakukan untuk mengetahui ketinggian air pada suatu tempat. Ketinggian air dapat diketahui dengan menggunakan alat pengukuran air otomatis atau dengan teknologi internet of things. Pada penelitian ini monitoring ketinggian air menggunakan sensor ultrasonik, mikrokontroler nodeMCU8266 dan beberapa modul elektronika untuk output. ESP8266 merupakan sebuah chip mikrokontroler yang sudah bisa terkoneksi WIFI dan System On Chip sehingga dapat di program secara langsung ke NodeMCU ESP8266 tanpa mikrokontroler tambahan.



Gambar 2. Prototype Rangkaian Sistem Monitoring Ketinggian Air

## 2.4 NodeMCU ESP8266

NodeMCU ESP8266 merupakan sebuah modul WIFI yang akhir-akhir ini populer di kalangan hardware developer. NodeMCU merupakan sebuah platform IoT yang bersifat OpenSource dan bersifat SOC (System On Chip), sehingga dapat diprogram langsung ke NodeMCU ESP8266 tanpa mikrokontroler tambahan. ESP 8266 memiliki kelebihan dapat menjalankan peran sebagai adhoc akses poin maupun klien sekaligus.



Gambar 3. NodeMCU ESP8266

## 2.5 PCB Board

PCB Board digunakan untuk menyediakan jalur listrik dan koneksi antara komponen elektronik yang ada pada suatu perangkat. Memungkinkan aliran listrik yang tepat antara komponen, serta memberikan dukungan mekanis bagi komponen tersebut. Pada PCB, jalur listrik terbuat dari lapisan tembaga yang ditempatkan di atas bahan isolator. Jalur-jalur ini biasanya dibentuk menggunakan metode etsa, di mana tembaga yang tidak diinginkan dihilangkan dari permukaan PCB dengan menggunakan bahan kimia. Proses ini membentuk pola jalur listrik yang diperlukan untuk menghubungkan komponen elektronik.



Gambar 4. PCB Board

## 2.6 Lampu LED

LED merupakan jenis lampu yang menggunakan dioda semikonduktor untuk menghasilkan cahaya. LED terdiri dari bahan semikonduktor yang dipadatkan dan memiliki kemampuan untuk mengubah arus listrik menjadi cahaya melalui proses yang disebut efek elektroluminesensi. Pada dasarnya, LED terdiri dari dua lapisan semikonduktor yang berbeda, yaitu lapisan P (positif) dan lapisan N (negatif), yang dipisahkan oleh sebuah lapisan aktif yang menghasilkan cahaya saat dialiri arus listrik. Ketika

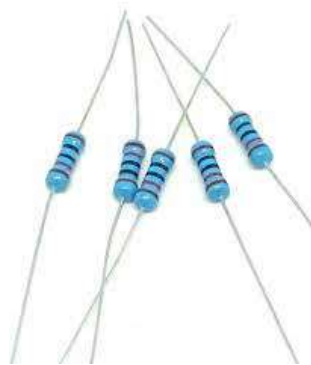
arus mengalir melalui LED, elektron dalam lapisan N dan lubang di lapisan P bertemu di lapisan aktif. Selama proses ini, energi dilepaskan dalam bentuk cahaya.



Gambar 5. Lampu LED

## 2.7 Resistor

Resistor komponen elektronik yang digunakan untuk membatasi aliran arus dalam suatu rangkaian listrik. Resistor memiliki resistansi, yang menunjukkan sejauh mana resistor menghambat arus listrik yang mengalir melaluinya. Resistansi diukur dalam satuan ohm ( $\Omega$ ). Resistor terbuat dari material yang memiliki sifat resistif, seperti karbon, logam, atau keramik. Resistor memiliki dua terminal yang memungkinkan mereka terhubung ke dalam rangkaian listrik. Ketika arus mengalir melalui resistor, sejumlah tegangan jatuh di sepanjang resistor sesuai dengan nilai resistansinya. Resistor digunakan dalam berbagai aplikasi dalam rangkaian elektronik, seperti pembatas arus, pembagi tegangan, penstabil tegangan, filter, pengatur volume, dan masih banyak lagi.



Gambar 6. Resistor

## 2.8 Transistor

Transistor merupakan salah satu komponen elektronik yang sangat penting dalam dunia teknologi. Fungsi utama transistor adalah sebagai penguat sinyal elektronik dan sebagai saklar elektronik yang dapat mengendalikan aliran arus listrik. Transistor terdiri dari tiga lapisan semikonduktor yang saling berhubungan. Lapisan ini umumnya terbuat dari bahan semikonduktor seperti silikon atau germanium.



Gambar 7. Transistor

## 2.9 Telegram

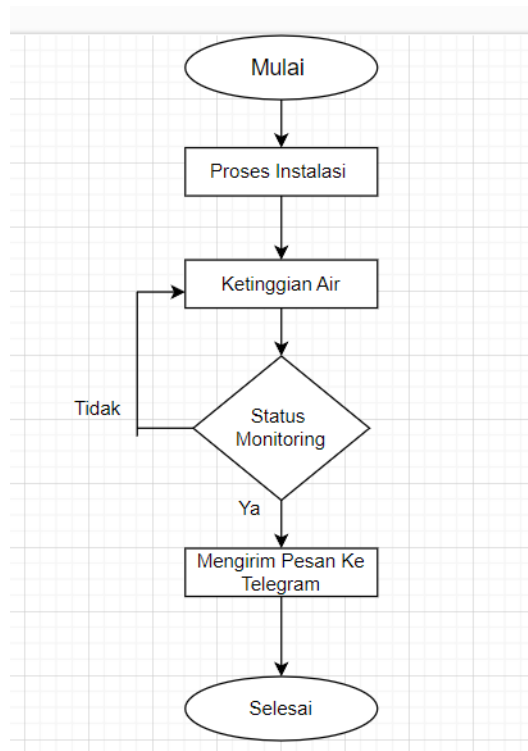
Telegram merupakan sebuah aplikasi pesan instan yang memungkinkan pengguna untuk mengirim pesan teks, suara, video, dan berbagi file dengan cepat dan aman. Aplikasi Telegram pertama kali diluncurkan pada tahun 2013 oleh Pavel Durov dan timnya. Telegram menawarkan berbagai fitur dan keunggulan dibandingkan dengan aplikasi pesan instan lainnya.



Gambar 8. Logo Aplikasi Telegram

## 3.0 Flowchart

Flowchart bertujuan, untuk memberi gambaran bagaimana cara kerja prototype sistem monitoring ketinggian air menggunakan telegram.



Gambar 9. Flowchart Prototype Sistem Monitoring Ketinggian Air Menggunakan Telegram

### 3.1 Source Code

Source Code ini berisikan coding untuk telegram untuk menampilkan hasil dari tes monitoring ketinggian air.

```

#include <ESP8266WiFi.h>
#include <WiFiClientSecure.h>
#include <UniversalTelegramBot.h>

// Wifi network station credentials
#define WIFI_SSID "UAS Elektronika"
#define WIFI_PASSWORD "12345678"
// Telegram BOT Token (Get from Botfather)
#define BOT_TOKEN "5892767980:AAHpyS4k55GuvmUtOyVDc3g6i_WqvQg2nmw"
int a=2;
int b=4;
int c=5;
boolean aman=false;
boolean siaga=false;
boolean bahaya=false;

const unsigned long BOT_MTBS = 1000; // mean time between scan messages

X509List cert(TELEGRAM_CERTIFICATE_ROOT);
WiFiClientSecure secured_client;
UniversalTelegramBot bot(BOT_TOKEN, secured_client);
unsigned long bot_lasttime; // last time messages' scan has been done

void handleNewMessages(int numNewMessages)
    
```

```

{
  Serial.println("handleNewMessages");
  Serial.println(String(numNewMessages));

  for (int i = 0; i < numNewMessages; i++)
  {
    String chat_id = bot.messages[i].chat_id;
    String text = bot.messages[i].text;

    String from_name = bot.messages[i].from_name;
    if (from_name == "")
      from_name = "Guest";
    if (text == "info")
    {
      bot.sendChatAction(chat_id, "typing");
      delay(4000);

      bot.sendMessage(chat_id, "Did you see the action message?");

    }
  }
  /*
  if (text == "info")
  {
    bot.sendChatAction(chat_id, "typing");
    delay(4000);
    bot.sendMessage(chat_id, "Did you see the action message?");

    // You can't use own message, just choose from one of bellow

    //typing for text messages
    //upload_photo for photos
    //record_video or upload_video for videos
    //record_audio or upload_audio for audio files
    //upload_document for general files
    //find_location for location data

    //more info here - https://core.telegram.org/bots/api#sendchataction
  }
  */
  // if (text == "/start")
  // {
  //   String welcome = "Welcome to water level nodemcu, " + from_name + ".\n";
  //   welcome += "info level akan otomatis. Tapi bisa coba perintah ini untuk manual.\n\n";
  //   welcome += "info : untuk dapat info level terbaru\n";
  //   bot.sendMessage(chat_id, welcome);
  // }
  }
}

void setup()
{
  Serial.begin(115200);
  Serial.println();
  pinMode(a,INPUT_PULLUP);
}

```

```

pinMode(b,INPUT_PULLUP);
pinMode(c,INPUT_PULLUP);
// attempt to connect to Wifi network:
Serial.print("Connecting to Wifi SSID ");
Serial.print(WIFI_SSID);
WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
secured_client.setTrustAnchors(&cert); // Add root certificate for api.telegram.org

while (WiFi.status() != WL_CONNECTED)
{
  Serial.print(".");
  delay(500);
}
Serial.print("\nWiFi connected. IP address: ");
Serial.println(WiFi.localIP());

Serial.print("Retrieving time: ");
configTime(0, 0, "pool.ntp.org"); // get UTC time via NTP
time_t now = time(nullptr);
while (now < 24 * 3600)
{
  Serial.print(".");
  delay(100);
  now = time(nullptr);
}
Serial.println(now);
}

void loop()
{
  boolean lv1=digitalRead(a);
  boolean lv2=digitalRead(b);
  boolean lv3=digitalRead(c);
  if(!lv1&&!lv2&&!lv3){
    if(!bahaya){
      Serial.println("Bahaya");
      bot.sendChatAction("1147502750", "typing");
      delay(1000);
      bot.sendMessage("1147502750", "Status Dalam Kondisi Bahaya!");
      bahaya=true;
      siaga=false;
      aman=false;
    }
  }
  else if(!lv1&&!lv2){
    if(!siaga){
      Serial.println("Siaga ");
      bot.sendChatAction("1147502750", "typing");
      delay(1000);
      bot.sendMessage("1147502750", "Status Dalam Kondisi Siaga!");
      siaga=true;
      aman=false;
      bahaya=false;
    }
  }
}

```



```

else if(!lv1){
  if(!aman){
    Serial.println("Aman ");
    bot.sendChatAction("1147502750", "typing");
    delay(1000);
    bot.sendMessage("1147502750", "Status Dalam Kondisi Aman");
    aman=true;
    siaga=false;
    bahaya=false;
  }
}
if (millis() - bot_lasttime > BOT_MTBS)
{
  int numNewMessages = bot.getUpdates(bot.last_message_received + 1);

  while (numNewMessages)
  {
    Serial.println("got response");
    handleNewMessages(numNewMessages);
    numNewMessages = bot.getUpdates(bot.last_message_received + 1);
  }

  bot_lasttime = millis();
}
}

```

### 3.2 Source Code untuk bot di telegram

/\*\*\*\*\*\*

A telegram bot for your ESP8266 that responds with whatever message you send it.

Parts:

D1 Mini ESP8266 \* - <http://s.click.aliexpress.com/e/uzFUnIe>  
(or any ESP8266 board)

= Affilate

If you find what I do useful and would like to support me, please consider becoming a sponsor on Github  
<https://github.com/sponsors/witnessmenow/>

Written by Brian Lough

YouTube: <https://www.youtube.com/brianlough>

Tindie: <https://www.tindie.com/stores/brianlough/>

Twitter: <https://twitter.com/witnessmenow>

\*\*\*\*\*/

### 3. HASIL DAN PEMBAHASAN

#### A. Pengujian NodeMCU ESP8266

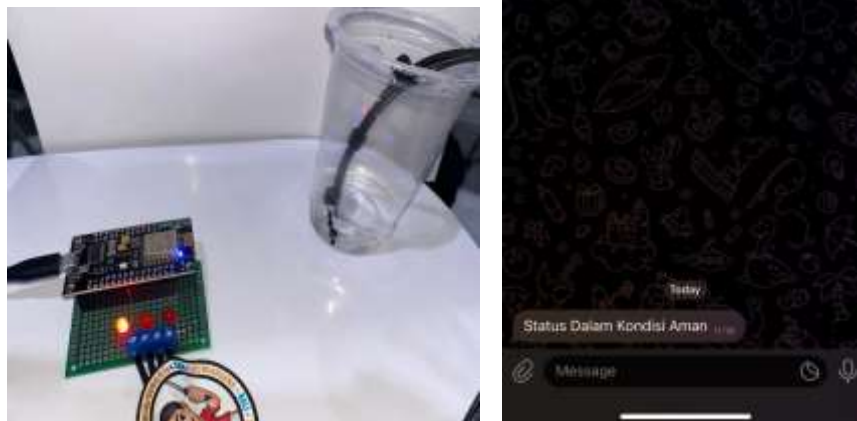
Pengujian NodeMCU Esp8266 dilakukan untuk mengetahui apakah dapat bekerja dengan baik seperti yang diinginkan, lalu hubungkan ke wifi yang sudah disiapkan dan dapat terhubung seperti yang diinginkan. Pengujian NodeMCU ESP 8266 dapat dilihat pada gambar 10.



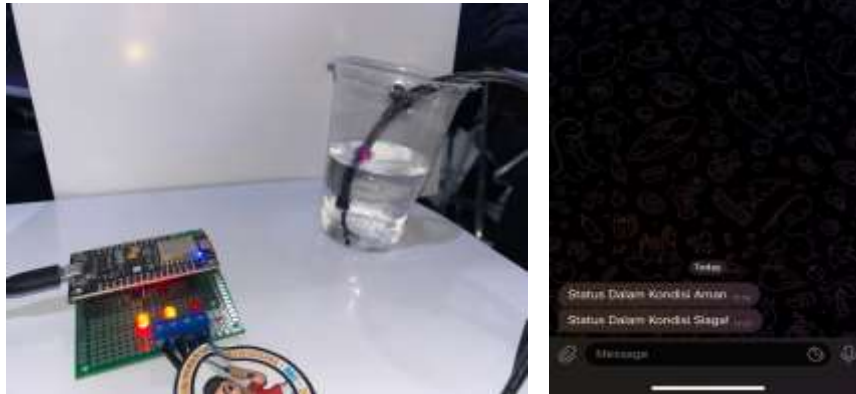
Gambar 10. Pengujian NodeMCU ESP8266

#### B. Pengujian Ketinggian Air

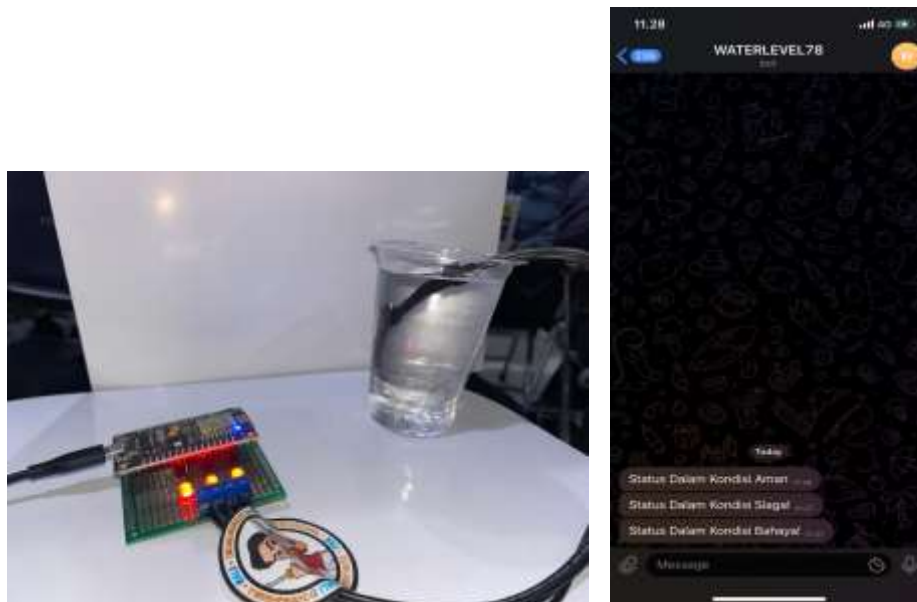
Pengujian ketinggian air ini untuk menampilkan hasil tes apakah status dalam kondisi aman, siaga, atau bahaya.



Gambar 11. Hasil Tes Pertama Status Dalam Kondisi Aman



Gambar 12. Hasil Tes Kedua Status Dalam Kondisi Siaga



Gambar 13. Hasil Tes Ketiga Status Dalam Kondisi Bahaya

#### 4. KESIMPULAN

Prototype ini berhasil mengintegrasikan sistem monitoring ketinggian air dengan platform Telegram. Hal ini memungkinkan pengguna untuk menerima notifikasi tentang perubahan ketinggian air melalui pesan di Telegram. Pengguna dapat dengan mudah mengakses informasi ketinggian air secara langsung melalui aplikasi Telegram yang mereka sudah akrab dengan penggunaannya. Tidak perlu lagi menginstal atau mengoperasikan aplikasi atau perangkat tambahan. Pengguna dapat dengan mudah mengatur notifikasi dan memilih interval pembaruan yang mereka inginkan dan juga dapat meminta informasi tentang ketinggian air pada permintaan. Prototype ini memberikan keunggulan dalam memonitoring ketinggian air secara efisien dan efektif, terutama bagi mereka yang tinggal atau bekerja di daerah yang rentan terhadap banjir atau perubahan tingkat air yang cepat. Namun, penting untuk diingat bahwa ini hanya sebuah prototipe,

dan kesimpulan ini didasarkan pada asumsi bahwa prototipe tersebut berhasil memenuhi tujuan dan kebutuhan pengguna secara efektif. Evaluasi lebih lanjut dan pengujian lapangan mungkin diperlukan sebelum mengimplementasikan secara luas.

## REFERENSI

- [1] Jayadi Akhmad, "Rancang Bangun Alat Monitoring Ketinggian Air Pada Reservoir Berbasis Internet Of Things," *Jurnal ICTEE Vol.3, No.2*, Januari. 2023.
- [2] Herdiana Yudi, "Prototype Monitoring Ketinggian Air Berbasis Internet Of Things Menggunakan Blynk Dan NodeMCU ESP8266 Pada Tangki," *Jurnal Informatika*, Vol. 07, No. 01 Bandung, Juni. 2020.
- [3] Achmad Andani, "Penentuan Level Air Tanki Dengan Sistem Kendali". *Jurnal Ilmiah "Elektrikal Enjiniring"* Vol. 09, No. 02, Mei-Agustus 2011.
- [4] Sadi Sumardi, Putra Syah Ilham, "Rancang Bangun Monitoring Ketinggian Air Dan Sistem Kontrol Pada Pintu Air Berbasis Arduino Dan SMS Gateway," *Jurnal Teknik Universitas Muhammadiyah Tangerang*, vol. 07, no. 01, hal. 77-91, Januari-Juni, 2018.
- [5] Ulumuddin, M Sudrajat, TD Rachmildha, "Prototipe Sistem Monitoring Air Pada Tangki Berbasis Internet Of Things Menggunakan NodeMCU ESP8266 Dan Sensor Ultrasonik", *Seminar Nasional Teknik Elektro UIN Sunan Gunung Djati*, Bandung, 2015.
- [6] Suryantoro Heri, "Prototype Sistem Monitoring Level Air Berbasis Labview dan Arduino Sebagai Sarana Pendukung Praktikum Instrumental Sistem Kendali", *Indonesian Journal of Laboratory*, vol. 1, no. 3, hal. 20-32, 2019.
- [7] Satya Prima Trias, Al Fauzan Rifqi Muhammad, Admoko Dwi Muhammad Estu "Sesnor Ultrasonik HCSR04 Berbasis Arduino Due Untuk Sistem Monitoring Ketinggian," *Jurnal Fisika Dan Aplikasinya*," vol.15, no. 2 hal. 36-39, 2019.
- [8] Bastian Ade, Mardiana Ardi, Riyanto Riki "Pengembangan Prototype Sistem Monitoring Ketinggian Air Untuk Peringatan Dini," *Jurnal J-Ensitem*, vol. 5, no. 2, 2019.
- [9] Poetra Arya Ade, Reza Nandika, Wijaya Kusuma Toni "Prototipe Sistem Monitoring Ketinggian Air Pada Tangki Berbasis Internet of Things," vol. 6, no. 1, hal. 97-108, 2023.
- [10] RB Sagita, A Prapanca "Rancang Bangun Sistem Monitoring Level Air Untuk Mendeteksi Banjir Berbasis Mikrokontroler Arduino Dan Visual Basic. Net ," vol. 8. no. 2, hal 98-104, 2018.
- [11] Sanusi Faiz Achmad, " Propotipe Sistem Pemantau Ketinggian Level Sungai Jarak Jauh Berbasis IoT ( Internet of Things ) dengan NodeMCU", 2018.
- [12] Rindra Karunika Aruna, Widodo Arif, Baskoro Farid, Kholis Nur "Sistem Monitoring Level Ketinggian Air Pada Tandon Rumah Tangga Berbasis IoT ( Internet of Things ) ," *Jurnal Teknik Elektro*., vo 11. 1, no. 1, hal. 17-22, 2022.
- [13] Wiratama Arun Nyoman, Wiharta Made Dewa, Wirastuti Dewi Esta Ary Made "Rancang Bangun Sistem Monitoring Ketinggian Air Berbasis Android Menggunakan Transistor Water Level," *Jurnal Spektrum*, vol. 7, no. 4, 2022.
- [14] Nugraha Satya Kevin, "Prototype Sistem Monitoring Ketinggian Air Di Pintu Air Menggunakan Aplikasi Android Berbasis Internet of Things", Institut Teknologi Telkom Jakarta, 2021.
- [15] Sulaksono Ganjar, " Rancang Bangun Prototype Sistem Monitoring Ketinggian Air Berbasis Microcontroler Arduino Uno R3, Universitas Negeri Jakarta, 2015.