

# DESIGN OF INTERNET OF THINGS (IOT) BASED BEARING MONITORING TOOL

Yudi Siswanto<sup>1</sup>, Agus Salim Opu<sup>2</sup>, Ika Nurjanah<sup>3</sup>, Henly Pakiding<sup>4</sup>

<sup>1,2,4</sup>D3 Teknik Perawatan Mesin, Politeknik Industri Logam Morowali,  
[yudi@pilm.ac.id](mailto:yudi@pilm.ac.id)

<sup>2</sup>[opuagussalim@gmail.com](mailto:opuagussalim@gmail.com)

<sup>4</sup>[henlypakiding@gmail.com](mailto:henlypakiding@gmail.com)

<sup>3</sup>S1 Teknik Mesin, Fakultas Teknik, Universitas Negeri Surabaya

[ikajannah@unesa.ac.id](mailto:ikajannah@unesa.ac.id)

**Abstrak**— Kerusakan bearing merupakan salah satu penyebab utama gangguan operasional pada sistem mekanis akibat kurangnya pemantauan kondisi secara berkelanjutan. Deteksi dini terhadap peningkatan getaran dan suhu menjadi langkah penting dalam mencegah downtime serta mengurangi biaya perawatan. Penelitian ini bertujuan merancang dan mengembangkan prototipe sistem pemantauan kondisi bearing berbasis *Internet of Things* (IoT) menggunakan metode *Research and Development* (R&D). Sistem terdiri atas mikrokontroler Arduino Uno dan NodeMCU ESP8266 sebagai pengendali utama, sensor piezoelektrik untuk mendeteksi getaran, serta sensor inframerah MLX90614 untuk mengukur suhu permukaan bearing. Data hasil pengukuran dikirim secara real-time ke platform *ThingSpeak* untuk visualisasi dan analisis jarak jauh. Hasil pengujian selama tiga jam menunjukkan rata-rata getaran sebesar 7,25 Hz dan suhu rata-rata 35,87 °C, di mana indikator kondisi pada LED dan LCD berfungsi sesuai dengan ambang batas yang telah ditentukan. Sistem mampu memberikan peringatan dini terhadap potensi kerusakan melalui pembacaan parameter secara kontinu. Kebaruan penelitian ini terletak pada integrasi multi-sensor berbiaya rendah dengan platform *ThingSpeak* untuk pemantauan kondisi bearing secara real-time dan *end-to-end*, yang mendukung konsep *predictive maintenance*.

**Kata kunci:** Bearing, IoT, ThingSpeak, Getaran, Suhu

**Abstract**— Bearing failure is one of the main causes of operational disruptions in mechanical systems due to the lack of continuous condition monitoring. Early detection of vibration and temperature increases is essential to prevent downtime and reduce maintenance costs. This study aims to design and develop a bearing condition monitoring prototype based on the *Internet of Things* (IoT) using the *Research and Development* (R&D) approach. The system employs an Arduino Uno and NodeMCU ESP8266 as the main controllers, a piezoelectric sensor to detect vibration, and an MLX90614 infrared sensor to measure the bearing surface temperature. The measured data are transmitted in real time to the *ThingSpeak* platform for remote visualization and analysis. Experimental testing over three hours showed an average vibration of 7.25 Hz and an average temperature of 35.87 °C, where the condition indicators on the LED and LCD operated according to the predefined thresholds. The system successfully provided early warnings of potential bearing failure through continuous parameter monitoring. The novelty of this research lies in the integration of low-cost multi-sensor technology with the *ThingSpeak* platform for real-time, end-to-end bearing condition monitoring, supporting the concept of *predictive maintenance*.

**Keywords:** Bearing, IoT, ThingSpeak, Vibration, Temperature

## PENDAHULUAN

Bearing merupakan salah satu bagian dari elemen mesin yang berperan untuk menopang poros di dalam mesin sehingga poros dapat berputar dengan lancar, bantalan memungkinkan poros dan elemen mesin lainnya berfungsi dengan baik[1]. Bantalan memiliki elemen dan bagian yang berputar yang senyap saat berputar, terletak di antara poros dan rumah bantalan. Pada bantalan bola, gesekan bergulir terjadi antara bagian yang berputar dan yang stasioner melalui elemen bergulir seperti bola, proses akan terus terjadi selama mesin beroperasi [2] [3]. Jika bantalan digunakan terus menerus tanpa tindakan pemantauan, dapat menyebabkan potensi kerusakan pada bantalan[4]. Kerusakan bantalan dapat menyebabkan penghentian peralatan mekanis yang

menciptakan waktu henti, peningkatan biaya untuk perbaikan penuh, dan bahkan menyebabkan kecelakaan besar[5] [6].

Oleh karena itu, pemantauan bantalan secara berkelanjutan sangat penting untuk memberikan informasi untuk mendeteksi cacat dini, serta untuk melakukan pelacakan kesalahan yang efisien, dan tindakan pemeliharaan yang optimal untuk mencegah kerusakan bantalan. Salah satu metode yang dapat digunakan dalam melakukan tindakan pemantauan bantalan adalah menggunakan teknologi sensor, pemantauan bantalan menggunakan sensor adalah metode yang lebih efektif, menawarkan berbagai keunggulan dibandingkan teknik tradisional yang terkadang hanya mengandalkan inspeksi visual atau dengan indera [7][8]. Mengintegrasikan sensor dengan *Internet of Things* (IoT)

adalah salah satu metode yang tepat dalam memantau bantalan, karena meningkatnya jumlah node sensor yang menghasilkan lebih banyak data dalam proses pemantauan. *Internet of Things* atau dikenal dengan singkatan IOT adalah konsep yang bertujuan untuk menghubungkan perangkat fisik dengan internet dan memperluas manfaat konektivitas internet yang membuat perangkat fisik seperti sensor terus terhubung [9][10].

Integrasi sensor dengan teknologi IoT menawarkan keuntungan seperti lalu lintas rendah karena pengguna dapat memantau dari jarak jauh, tanpa perlu memantau langsung di lokasi sehingga menghasilkan stabilitas, dan cakupan yang lebih besar, sehingga menghemat biaya inspeksi konvensional. Secara keseluruhan, dengan penerapan teknologi IoT dapat membantu menyederhanakan tugas manual dan meningkatkan jangkauan pemantauan[11][12]. Pada penelitian sebelumnya yang dilakukan oleh Adam satria, pemantauan kondisi bantalan tahun 2016 dilakukan dengan sensor yang akan menampilkan hasil pemantauan pada layar LCD[13]. Namun, penelitian membatasi pemantauan hanya pada lokasi dimana alat tersebut berada, pada penelitian Sukumar menggunakan sensor tunggal tanpa adanya integrasi dengan *Internet of Things*[14]. Penelitian ini membawa perkembangan dengan mengintegrasikan sensor dengan teknologi *Internet of Things*, memungkinkan data pemantauan bantalan dipantau secara real-time, sehingga pengguna dapat segera mengetahui kondisi bantalan dan memberikan solusi yang diperlukan tanpa harus berada di lokasi. Penelitian ini bertujuan merancang dan mewujudkan alat pemantau bearing berbasis IoT yang mampu mengukur frekuensi getaran dan suhu secara bersamaan, lalu mengunggahnya ke ThingSpeak secara real time. Penulis juga mengecek fungsi pembacaan sensor, keandalan komunikasi antara Arduino dan ESP8266, agar teknisi bisa mendapat peringatan dini mengenai kondisi bearing di lapangan. penelitian ini meliputi cara merancang sistem yang ringkas, murah, dan mudah dipasang untuk memantau getaran dan suhu secara real time; cara memastikan data dapat dikirim dan ditampilkan dengan andal di ThingSpeak untuk pemantauan jarak jauh; seberapa efektif indikator LED/LCD/buzzer memberi peringatan saat ambang terlampaui; serta seberapa stabil proses akuisisi dan transmisi data ketika sistem dijalankan terus-menerus.

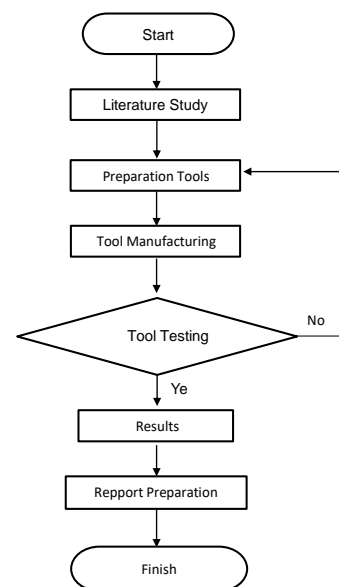
Penelitian ini mencakup rancangan end-to-end pemantauan bearing multi sensor mulai dari akuisisi, pengiriman, hingga visualisasi real time menggunakan ThingSpeak; pembuktian eksperimental bahwa pembacaan dan aliran data berjalan stabil beserta penyajian metrik evaluasi yang ringkas (rata-rata frekuensi getaran dan suhu); serta penerapan indikator kondisi yang praktis dan mudah direplikasi untuk mendukung perawatan prediktif di lapangan.

## METODE

Metode Metodologi yang digunakan dalam penelitian ini adalah Penelitian dan Pengembangan. Research and Development (R&D) adalah proses atau langkah-langkah

untuk mengembangkan produk baru atau meningkatkan produk yang ada dengan langkah-langkah yang terstruktur dan sistematis mulai dari desain, manufaktur, pengujian, hingga evaluasi [14]. Secara rinci, tahapan dalam desain penelitian ini dijelaskan sebagai berikut:

1. Tahap awal dilakukan dengan memulai penelitian.
2. Setelah memulai penelitian, dilanjutkan dengan penelitian yang sedang dilakukan, baik itu melalui jurnal ilmiah atau karya ilmiah lainnya.
3. Membuat perencanaan alat sebagai fondasi awal yang meliputi Perancangan arsitektur (perangkat keras, firmware, alur IoT, dan skema data).
4. Setelah melalui tahap perencanaan, proses pembuatan alat dapat dilakukan yang mencakup perakitan modul sensor, mikrokontroler, antarmuka lokal, dan konektivitas.
5. Setelah alat dibuat, tahap selanjutnya adalah uji coba alat yang meliputi fungsi sensor getaran, sensor suhu, komunikasi serial, LCD/LED/buzzer, jika alat tidak berfungsi maka penelitian akan kembali ke tahap pembuatan alat, jika berhasil maka penelitian dapat dilanjutkan ke tahap selanjutnya.
6. Setelah melakukan pengujian alat, maka akan didapatkan hasil
7. Hasil yang diperoleh kemudian akan digunakan untuk membuat laporan
8. Setelah proses penulisan laporan selesai, penelitian telah mencapai tahap akhir.



Gbr 1. Bagan alur penelitian

## Kerusakan Bantalan

Kerusakan pada bearing dapat diindikasikan dan dianalisa secara visual dengan mata telanjang setelah bearing dilepas dari

poros atau rumah bearing. Ada beberapa kasus kerusakan bearing yang dapat dianalisa secara visual, yaitu karena kelebihan beban, panas yang berlebihan, pemasangan yang tidak benar, kontaminasi, kesalahan pelumas dan lain-lain. Teknologi terbaru telah menyediakan alat untuk analisis kerusakan elemen pada bearing berdasarkan getaran yang dihasilkan. Pengukuran getaran yang dihasilkan oleh sebuah bearing harus dilakukan pada saat mesin beroperasi [15].

### Sistem Pemantauan

Sistem pemantauan adalah proses untuk mengumpulkan data secara real-time dari berbagai sumber daya. Secara garis besar, tahapan dalam sistem monitoring dibagi menjadi tiga proses, yaitu [16]:

1. Proses dalam mengumpulkan data pemantauan
2. Proses dalam menganalisis data pemantauan
3. Proses dalam menampilkan data hasil

### Internet of Things

*Internet of Things* (IoT) adalah platform revolusioner yang menghubungkan jaringan pintar dengan sejumlah besar perangkat elektronik melalui internet, memungkinkan koneksi yang andal dan real-time, pengumpulan data dari sensor, komputasi, dan aktuasi [17]. Perangkat IoT, seperti mikrokontroler berjejaring dengan sensor dan output yang terpasang, telah menjadi kebutuhan di rumah tangga, transportasi, industri, dan perawatan kesehatan, sehingga memberikan kesempatan untuk menunjukkan relevansi komputasi sosial [18]. IoT memungkinkan interaksi antara berbagai perangkat untuk mencapai tujuan yang sama, membuka jalan bagi berbagai aplikasi berdampak tinggi di berbagai bidang kehidupan nyata di masa depan [19]. Protokol komunikasi seperti MQTT, AMQP, CoAP, dan lainnya digunakan dalam aplikasi IoT, dengan fokus pada pengembangan protokol pesan multi-topik untuk meningkatkan kinerja dan kecerdasan sistem IoT [20].

### Thingspeak

*ThingSpeak* merupakan sebuah platform open source berupa website yang menyediakan layanan untuk kebutuhan IOT dan dapat menerima data menggunakan protokol HTTP melalui jaringan internet. ThingSpeak memungkinkan pembuatan aplikasi logging, aplikasi pelacakan lokasi, dan jejaring sosial dengan pembaruan status [21].

### Spesifikasi Sensor Piezoelektrik

1. Prinsip kerja: cakram piezoelektrik menghasilkan tegangan proporsional terhadap percepatan/defleksi
2. Tegangan kerja modul: 3,3–5 V (umum 5 V dari Arduino)
3. Keluaran: analog 0–5 V (tanpa beban), impedansi tinggi; disarankan resistor beban 1 M $\Omega$  paralel ke piezo untuk membuang muatan

4. Frekuensi kerja yang relevan untuk kasus Anda: 1–100 Hz (putaran 60–600 rpm). Cakram piezo sanggup jauh lebih tinggi, tetapi untuk monitoring putaran rendah cukup di rentang ini
5. Sensitivitas: sangat bergantung rangkaian kondisioning (jika pakai op-amp/penyearah). Untuk pengukuran “frekuensi” via zero-crossing, fokus pada kestabilan sinyal, bukan skala mutlak
6. Sampling yang disarankan: 200–500 Hz di Arduino (memberi  $\geq 20$ –50 sampel per siklus pada 5–10 Hz)
7. Rekomendasi rangkaian:
  - 7.1. Seri resistor 100 k $\Omega$ –1 M $\Omega$  ke A0 untuk membatasi arus puncak
  - 7.2. Dioda penjepit atau gunakan proteksi internal AVR (jangan biarkan tegangan di atas Vcc+0,5V)
  - 7.3. Opsional: penyearah + filter RC bila ingin membaca amplitudo, bukan frekuensi
8. Pemasangan mekanik: klem/epoxy ke housing dekat bantalan; hindari longgar karena akan mempengaruhi hasil
9. Noise: kurangi dengan pengkabelan pendek, ground tunggal, dan rata-rata bergerak 3–5 jendela

### Spesifikasi Sensor Suhu Infra merah

1. Model umum: MLX90614ESF (contoh varian BAA, FOV  $\pm 90^\circ$ ; varian lain bisa lebih sempit)
2. Tegangan kerja: 3,3–5 V
3. Antarmuka: SMBus/I2C kompatibel, alamat default 0x5A, kecepatan tipikal 100 kHz
4. Koneksi Arduino Uno: SDA = A4, SCL = A5; tambah pull-up 4,7–10 k $\Omega$  ke Vcc jika modul belum memilikinya
5. Rentang ukur suhu objek: sekitar  $-70$  hingga  $+380^\circ\text{C}$
6. Rentang ukur suhu ambient internal: sekitar  $-40$  hingga  $+125^\circ\text{C}$
7. Akurasi tipikal:  $\pm 0,5^\circ\text{C}$  (sekitar  $0$ – $50^\circ\text{C}$ , kondisi kalibrasi pabrik)
8. Resolusi:  $0,02^\circ\text{C}$  (14-bit)
9. Waktu respons optik/termal: sekitar 0,1–0,7 s (tergantung filter internal)
10. *Field of view* (FOV):  $\pm 90^\circ$  untuk varian BAA; gunakan varian FOV sempit bila ingin area ukur lebih fokus
11. Emisivitas: dapat diprogram (default 1,0); untuk permukaan logam mengkilap gunakan cat/matte tape agar bacaan akurat
12. Interval baca yang disarankan: 5–10 Hz (setiap 100–200 ms) dengan smoothing moving average 5 sampel

### Sampling Rate

1. Sampling getaran: 200 Hz pada Arduino Uno (ADC 10-bit). Estimasi frekuensi dilakukan per jendela 1 detik dengan overlap 50% menggunakan metode zero-

HARI	FLUKTUASI GETARAN (Hz)	FLUKTUASI SUHU (°C)
1	0,48	0,19
2	0,52	0,21
3	0,50	0,20
4	0,47	0,18
5	0,53	0,22
6	0,49	0,20
7	0,51	0,20
RATA-RATA	0,50 Hz	0,20 °C

crossing/ambang ganda; hasil dirata-ratakan 3 jendela untuk meredam noise.

2. Sampling suhu inframerah (MLX90614): 5 Hz (pembacaan setiap 200 ms) dengan smoothing moving average 5 sampel; latensi efektif sekitar 1 detik.
3. Komunikasi serial Arduino–ESP8266: UART 9600 bps, 8N1.
4. Pengiriman ke ThingSpeak: HTTP GET setiap 15 detik; data yang dikirim adalah nilai rata-rata selama interval tersebut (Field1 untuk getaran, Field2 untuk suhu

## HASIL DAN DISKUSI

TABEL 1  
KALIBRASI SENSOR PIEZZOELEKTRIK

Suhu Referensi (°C)	Suhu Sensor (°C)	Suhu Koreksi (°C)	Error Absolut (°C)	Error (%)
25.00.00	24.84	25.18.00	00.18	0,0506944
30.00.00	29.85	30.18.00	00.18	0,0416667
35.00.00	34.50.00	34.82	00.18	00.51
40.00.00	39.63	39.95	00.05	00.12
45.00.00	44.66	44.98	00.02	00.04
50.00.00	49.61	49.92	00.08	00.16
60.00.00	59.77	60.07.00	00.07	00.12
70.00.00	69.69	69.99	00.01	00.01
80.00.00	79.80	80.04.00	00.04	00.05
90.00.00	89.82	90.04.00	00.04	00.05
100.00.00	99.83	100.03.00	00.03	00.03
Rata-rata	—	—	00.11	00.28

Dari tabel, error absolut rata-rata = 0.44 Hz.

Error relatif rata-rata  $\approx 2.34\%$ .

Artinya, sensor Anda cenderung meleset sekitar 0.4 Hz dari nilai sebenarnya pada rentang 5–50 Hz.

### Rumus koreksi:

$$F_{\text{corr}} = 0.08627 + 0.99542 \times 24.4$$

$$F_{\text{corr}} \approx 24.37 \text{ Hz}$$

### Koding untuk arduino:

float Fs = measuredHz; (hasil hitung frekuensi sensor)

float Fcorr = 0.08627 + 0.99542 \* Fs; (frekuensi terkalibrasi)

TABEL 2  
KALIBRASI SENSOR SUHU INFRA MERAH

No	Frekuensi Referensi (Hz)	Frekuensi Sensor (Hz)	Frekuensi Koreksi (Hz)	Error Absolut (Hz)	Error (%)
1	5	05.02	05.26	00.26	05.20
2	10	10.03	10.34	00.34	03.40
3	15	14.07	0,634027778	00.27	0,097222
4	20	20.05	20.49	00.49	02.45
5	25	24.04.00	24.42.00	00.58	02.32
6	30	29.08.00	29.72	00.28	0,064583
7	35	34.06.00	34.50.00	00.50	01.43
8	40	39.05.00	39.47.00	00.53	01.33
9	45	45.08.00	45.70	0,048611111	01.56
10	50	50.06.00	50.48.00	00.48	0,066667
	Rata-rata	—	—	0.44 Hz	2.34 %

menunjukkan error absolut rata-rata  $\approx 0.10\text{ }^{\circ}\text{C}$  ( $\approx 0.24\%$ )

### Rumus koreksi:

$$\text{Suhu Koreksi} = 0.3887 + 0.99811 \times (\text{Suhu Sensor})$$

### Koding untuk arduino:

$$\text{float Tcorr} = 0.3887 + 0.99811 * \text{Ts};$$

### Pengujian Jangka Panjang (1 Minggu)

Sensor diuji ulang secara berkala (setiap hari).

Hasil pengukuran dibandingkan dengan nilai referensi.

Dari pemantauan beberapa hari, error rata-rata tetap stabil:

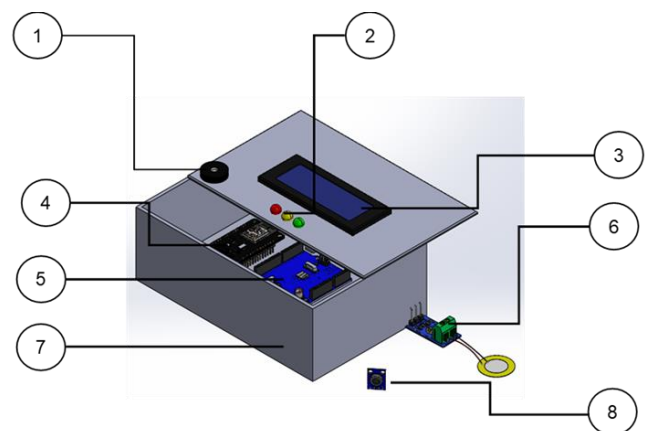
Sensor getaran: fluktuasi  $\pm 0.5$  Hz

Sensor suhu IR: fluktuasi  $\pm 0.2\text{ }^{\circ}\text{C}$

Tidak ditemukan drift yang signifikan, sehingga persamaan kalibrasi awal tetap berlaku.

### Perencanaan Alat

Perencanaan alat dimulai dengan membuat desain menggunakan perangkat lunak *Computer Aided Design* (CAD). Perancangan pada software CAD dapat memastikan bahwa setiap komponen alat didesain dengan presisi. Di bawah ini adalah hasil desain beserta keterangan untuk setiap bagian yang telah didesain.



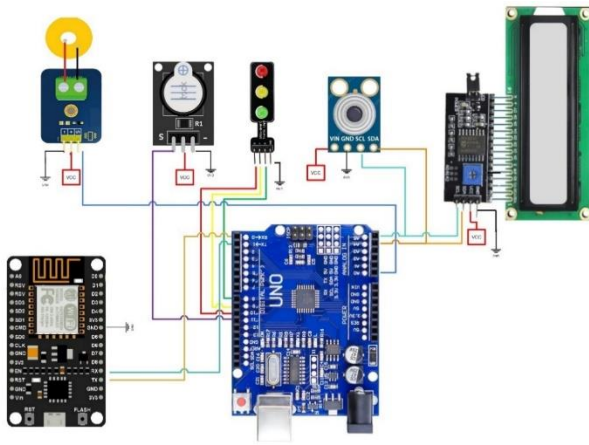
Gbr. 2 Desain Alat

### Deskripsi

- |                    |                            |
|--------------------|----------------------------|
| 1. Bel             | 5. Bel Arduino Uno         |
| 2. LED Indikator   | 6. Piezoelektrik           |
| 3. LCD             | 7. Kotak Panel             |
| 4. Nodemcu ESP8266 | 8. Sensor Suhu Infra Merah |

### Perencanaan rangkaian skematik

Pada bagian ini, kami akan menjelaskan rangkaian skematik perangkat yang akan dibuat. Rangkaian ini memberikan gambaran menyeluruh mengenai komponen-komponen utama, serta memvisualisasikan jalur setiap perangkat untuk saling terhubung sehingga mencapai fungsi yang diinginkan.



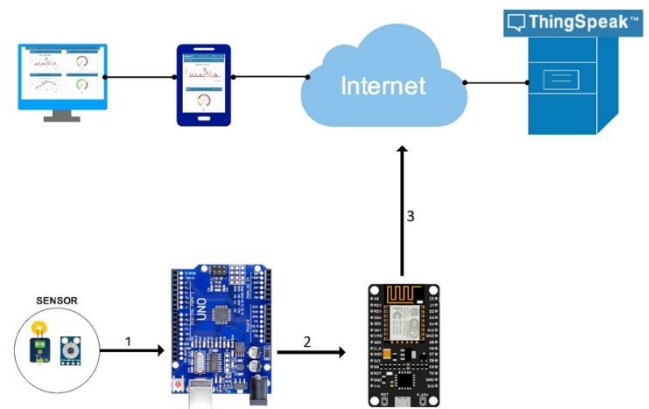
Gbr.3. Perencanaan sirkuit skematik

Berdasarkan rangkaian skematik, setiap komponen mendapatkan VCC dan GND untuk memberikan daya pada masing-masing komponen. Adapun pin analog, digital, dan sinyal yang masuk ke Arduino Uno adalah sebagai berikut:

Untuk menghubungkan NodeMCU ke Arduino Uno, Rx pada NodeMCU dihubungkan ke Tx Arduino Uno, dan Tx pada NodeMCU dihubungkan ke pin Rx Arduino Uno. Selain itu, GND pada NodeMCU terhubung ke GND Arduino Uno. Untuk menghubungkan LCD I2C, SDA pada LCD I2C dihubungkan ke A4 Arduino Uno, dan SCL pada LCD I2C dihubungkan ke A5 Arduino Uno. Pin VCC pada LCD I2C dihubungkan ke 5V Arduino Uno, dan GND pada LCD I2C dihubungkan ke GND Arduino Uno.

Untuk menghubungkan sensor piezoelektrik, S pada sensor piezoelektrik dihubungkan ke pin A0 Arduino Uno, pin Vcc pada sensor piezoelektrik dihubungkan ke 5V Arduino Uno, dan GND pada sensor piezoelektrik dihubungkan ke GND Arduino Uno. Untuk menghubungkan buzzer, S pada buzzer dihubungkan ke A11 Arduino Uno, Vcc pada buzzer dihubungkan ke pin 5V Arduino Uno, dan GND pada buzzer dihubungkan ke GND Arduino Uno.

LED indikator dihubungkan dengan cara berikut: G pada LED indikator dihubungkan ke pin 8 Arduino Uno, Y pada LED indikator dihubungkan ke D9 Arduino Uno, dan R pada LED indikator dihubungkan ke D10 Arduino Uno. GND pada LED indikator dihubungkan ke GND Arduino Uno. Untuk menghubungkan sensor suhu inframerah, SDA pada sensor suhu inframerah dihubungkan ke A4 Arduino Uno, pin SCL pada sensor suhu inframerah dihubungkan ke A5 Arduino Uno, VCC pada sensor suhu inframerah dihubungkan ke 5V Arduino Uno, dan GND pada sensor suhu inframerah dihubungkan ke GND Arduino Uno.

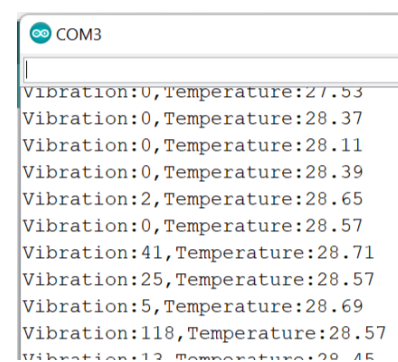


Gbr. 4 Skema IOT

Skema IOT yang digunakan dalam penelitian ini dibagi menjadi 3 tahap utama, sebagai berikut:

1. Sensor getaran dan sensor suhu akan mengirimkan kondisi getaran dan suhu bantalan ke Arduino Uno
2. Setelah Arduino Uno mendapatkan Nilai Getaran, Arduino Uno akan mengirimkan nilai getaran dan suhu melalui komunikasi serial.
3. Nilai-nilai yang diperoleh dari Arduino uno akan diteruskan ke Internet dan disimpan di server cloud Thingspeak sehingga pengguna dapat memonitor secara langsung melalui komputer atau smartphone.

Hasil pengujian sensor piezoelektrik dalam mengukur frekuensi getaran dan sensor suhu inframerah dalam mengukur nilai suhu menunjukkan bahwa kedua sensor tersebut dapat bekerja dan merespon sesuai dengan fungsinya. Sensor piezoelektrik berhasil mendeteksi dan mengukur getaran dengan akurasi yang cukup baik, menunjukkan kemampuan untuk menangkap perubahan frekuensi getaran yang terjadi pada objek yang dimonitor. Di sisi lain, sensor suhu inframerah mampu mengukur suhu sesuai dengan fungsinya.



Gbr 5. Monitor serial Pembacaan sensor

Hasil pengujian komunikasi serial antara Arduino dengan NodeMCU ESP8266 dalam mengirimkan data hasil pembacaan sensor getaran dan sensor suhu menunjukkan hasil yang baik. Pada serial monitor terlihat bahwa Arduino berhasil mengirimkan data dengan metode komunikasi serial ke NodeMCU ESP8266. Pengujian ini melibatkan serangkaian langkah untuk memastikan bahwa data yang dikirim dari



Arduino dapat diterima dengan baik oleh NodeMCU ESP8266 tanpa ada informasi yang hilang atau terjadi kesalahan.

```
COM3

Data berhasil dikirim
Nilai Sensor: 4 Tegangan: 0.02
Data berhasil dikirim
Nilai Sensor: 4 Tegangan: 0.02
Data berhasil dikirim
Nilai Sensor: 4 Tegangan: 0.02
Data berhasil dikirim
Nilai Sensor: 3 Tegangan: 0.01
Data berhasil dikirim
Nilai Sensor: 0 Tegangan: 0.00
Data berhasil dikirim
```

Gbr. 6. Tampilan komunikasi serial

Hasil pengujian LED indikator dalam menampilkan warna yang berbeda pada setiap kondisi yang telah ditentukan, yaitu hijau untuk kondisi baik, kuning untuk kondisi rusak sedang, dan merah untuk kondisi rusak berat, seperti yang terlihat pada gambar, LED indikator akan menyala secara bergantian pada kondisi yang telah ditentukan.



Gbr.7 Tampilan LED indikator

Pengujian yang dilakukan bertujuan untuk mengevaluasi kemampuan LCD dalam menampilkan data yang diperoleh dari sensor getaran dan sensor suhu. Pada penelitian ini, sistem dirancang untuk memberikan tampilan yang interaktif dan informatif dengan menampilkan pesan yang berbeda berdasarkan pembacaan sensor. LCD akan menampilkan nilai getaran dan suhu yang terukur secara bergantian, dan memberikan notifikasi sesuai dengan kondisi yang terdeteksi. Kondisi ini dibagi menjadi tiga kategori: kondisi baik, rusak sedang, dan rusak berat.



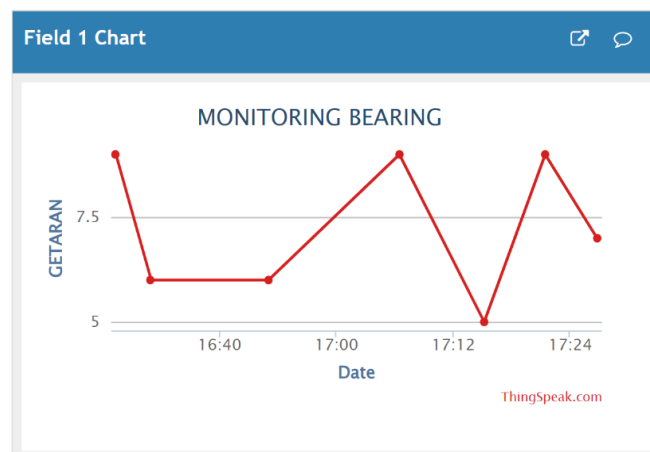
Gbr. 8 Tampilan LCD

Hasil pengujian untuk buzzer menunjukkan bahwa perangkat ini akan aktif ketika pembacaan sensor melewati batas nilai tertinggi yang telah ditetapkan. Pengujian ini dilakukan dengan mengatur ambang batas untuk suhu dan getaran, dimana jika sensor suhu atau sensor getaran mendeteksi nilai yang melebihi ambang batas, maka buzzer akan diaktifkan sebagai tanda peringatan.

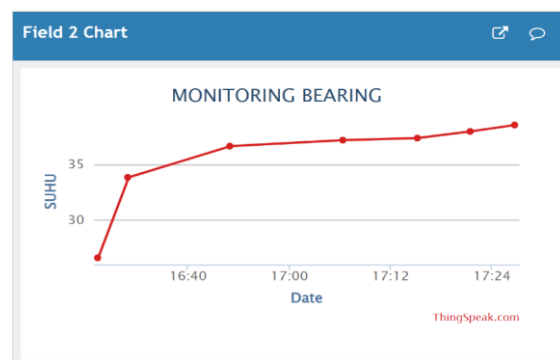


Gbr. 9 Tampilan Buzzer

Pengujian sistem *Internet of Things* (IoT) menunjukkan bahwa NodeMCU ESP8266 berhasil mengirimkan data getaran dan suhu secara akurat ke *ThingSpeak platform*. Pengiriman data ini membuktikan bahwa sistem berhasil memproses dan mengirimkan informasi yang diperoleh NodeMCU ESP8266 dari sensor ke *platform cloud* untuk pemantauan jarak jauh.



Gbr. 10 Tampilan platform *Thingspeak*



Gbr. 11. Tampilan platform *Thingspeak*

Tabel di bawah ini menunjukkan hasil pengukuran getaran dan suhu dari sistem pemantauan bearing berbasis IoT dalam berbagai interval waktu. Pada awal pengukuran, getaran terdeteksi pada 9 Hz dan suhu 26,61°C. Setelah 40 menit, getaran turun menjadi 6 Hz dan suhu meningkat menjadi 33,83°C. Satu jam setelah pengukuran dimulai, getaran tetap pada 6 Hz dan suhu naik lagi menjadi 36,63°C.

Setelah 1,2 jam, getaran naik kembali ke 9 Hz dengan suhu 37,17°C. Pada 1,4 jam, getaran menurun menjadi 5 Hz dan suhu sedikit meningkat menjadi 37,35°C. Ketika

pengukuran mencapai 2 jam, getaran kembali ke 9 Hz dan suhu mencapai 37,95°C. Setelah 2,4 jam, getaran turun sedikit menjadi 7 Hz dan suhu naik menjadi 38,53°C. Pada akhir pengukuran setelah 3 jam, getaran tetap pada 7 Hz dan suhu mencapai 38,93°C.

Nilai rata-rata getaran selama periode pengukuran adalah 7,25 Hz, dan suhu rata-rata adalah 35,87°C. Kondisi bearing secara keseluruhan dinilai baik, dengan semua pengukuran menunjukkan indikator hijau pada lampu dan tampilan “Kondisi Baik” pada LCD. Hal ini menunjukkan bahwa selama periode pengukuran, sistem berfungsi dalam batas normal dan tidak ada indikasi masalah pada bearing.

TABEL 3  
HASIL PENGUJIAN

NO	Operation time	Vibration Value	Temperature Value	Display on Lcd	Indicator Light
1.	0 minute	9 Hz	26,61°C	Good Condition	green
2.	40 minute	6 Hz	33,83°C	Good Condition	green
3.	1 hour	6 Hz	36,63°C	Good Condition	green
4.	1,2 hour	9 Hz	37,17°C	Good Condition	green
5.	1,4 hour	5 Hz	37,35°C	Good Condition	green
6.	2 hour	9 Hz	37,95°C	Good Condition	green
7.	2,4 hour	7 Hz	38,53°C	Good Condition	green
8.	3 hour	7 Hz	38,93°C	Good Condition	green
Nilai Rata-Rata		7,25 Hz	35,87°C	Good Condition	green

Berdasarkan delapan titik ukur selama 3 jam, statistik deskriptifnya adalah sebagai berikut.

1. Getaran: rata-rata 7,25 Hz; simpangan baku 1,58 Hz; median 7,00 Hz; rentang 5–9 Hz; koefisien variasi 21,8%. Secara tren, kemiringan garis waktu  $\approx -0,15$  Hz/jam (mencerminkan kestabilan—turun tipis dan tidak signifikan secara praktis).
2. Suhu: rata-rata 35,87°C; simpangan baku 4,06°C; median 37,26°C; rentang 26,61–38,93°C; koefisien variasi 11,3%. Suhu menunjukkan kenaikan seiring waktu dengan kemiringan  $\approx 3,44^\circ\text{C}/\text{jam}$  (perkiraan interval kepercayaan  $\pm 1,9^\circ\text{C}/\text{jam}$  pada data ini), konsisten dengan pemanasan bertahap selama operasi.
3. Hubungan getaran–suhu dalam rentang uji ini lemah (korelasi mendekati nol), sehingga perubahan suhu yang teramati terutama mencerminkan pemanasan operasi, bukan perubahan pola getaran.

#### Interpretasi teknis

1. Variasi getaran ( $CV \approx 21,8\%$ ) masih dalam batas kondisi baik yang Anda tetapkan; tidak terlihat lonjakan frekuensi yang berkelanjutan.

2. Kenaikan suhu yang hampir linear ( $\sim 3,4^\circ\text{C}/\text{jam}$ ) selaras dengan fase pemanasan awal sistem; mendekati keadaan tunak di akhir uji ( $38\text{--}39^\circ\text{C}$ ).
3. Dengan indikator berbasis ambang (LED/LCD/*buzzer*), seluruh pembacaan jatuh pada kategori baik, memvalidasi bahwa ambang konservatif Anda efektif untuk skenario pengujian ini.

Validitas dan akurasi pengukuran untuk memperjelas keabsahan data, tambahkan subbagian berikut pada metode atau hasil:

1. Validasi suhu (MLX90614).
  - a) Prosedur: lakukan kalibrasi dua titik sederhana pada permukaan uji bercat hitam (emisivitas  $\approx 0,95\text{--}1,0$ ): titik dingin (es–air  $\sim 0^\circ\text{C}$ ) dan titik panas (air mendidih  $\sim 100^\circ\text{C}$  di permukaan laut), atau bandingkan langsung terhadap termometer referensi (mis. termokopel K berkalamasi).
  - b) Koreksi emisivitas: jika objek logam mengilap, tempelkan pita hitam matte pada titik ukur untuk mengurangi bias pancaran.
  - c) Metrik akurasi: hitung MAE, RMSE, dan MAPE terhadap alat referensi.
    - $\text{MAE} = \text{rata-rata}(|T_i(\text{measured}) - T_i(\text{ref})|)$
    - $\text{RMSE} = \sqrt{\text{rata-rata}((T_i(\text{measured}) - T_i(\text{ref}))^2)}$
    - $\text{MAPE} = 100\% \times \text{rata-rata}(|T_i(\text{measured}) - T_i(\text{ref})|/T_i(\text{ref}))$
  - d) Pelaporan: tampilkan tabel ringkas (n, MAE, RMSE, MAPE) dan sebutkan batas akurasi tipikal sensor ( $\approx \pm 0,5^\circ\text{C}$  pada  $0\text{--}50^\circ\text{C}$ ); jelaskan apakah hasil validasi Anda berada dalam atau mendekati kisaran tersebut.

2. Validasi frekuensi getaran (*piezo*).

- a) Prosedur: bandingkan estimasi frekuensi dari Arduino (metode *zero-crossing*) dengan referensi tachometer optik pada poros atau dengan FFT dari akselerometer/HPN yang dikalibrasi. Alternatif murah: uji pada motor dengan putaran diketahui (mis. 600 rpm  $\rightarrow$  10 Hz) dan verifikasi deviasi.
- b) Metrik akurasi: gunakan MAE, RMSE, dan galat relatif (%) per titik uji.
- c) *Robustness*: laporkan keberhasilan deteksi pada beberapa amplitudo/ketegangan pemasangan sensor (longgar vs klem kuat) untuk menunjukkan ketahanan metode terhadap variasi mekanik.

Keandalan transmisi dan integritas data laporkan metrik jaringan agar *end-to-end*-nya jelas.

1. Interval kirim: 15 detik; dalam 3 jam diharapkan 720 *update*.
2. Keberhasilan unggah: hitung persentase *update* sukses =  $(\text{terkirim}/\text{diupayakan}) \times 100\%$ . Target  $\geq 95\%$ .
3. Latensi dan *jitter*: jika dicatat, sebutkan median waktu respons server dan penyebarannya.
4. Konsistensi: cek ada/tidaknya paket hilang, duplikasi, atau urutan yang tidak sesuai *timestamp*.

Perbandingan dengan literatur, temuan dalam penelitian ini berkaitan dengan kecenderungan umum pada penelitian sejenis yaitu:

1. Kenaikan suhu selama jam operasi awal lazim dilaporkan pada sistem berputar akibat pemanasan bantalan dan pelumas; temuan  $\sim 3\text{--}4\text{ }^{\circ}\text{C/jam}$  Anda sejalan dengan pola tersebut.
2. Stabilitas frekuensi getaran pada kondisi normal dan tidak ditemukannya kenaikan yang persisten juga konsisten dengan laporan bahwa anomali getaran yang berarti biasanya muncul saat kerusakan mulai berkembang (*mis. misalignment atau pitting*).
3. Integrasi IoT berbasis *ThingSpeak* untuk *logging* dan visualisasi real-time telah digunakan pada berbagai aplikasi monitoring; kontribusi Anda terletak pada penyatuan multi-sensor murah, indikator lokal, dan evaluasi kontinu 3 jam pada skenario bengkel/laboratorium.

## HASIL

Selama pengujian 3 jam, suhu meningkat hampir linear dengan kemiringan sekitar  $3,44\text{ }^{\circ}\text{C}$  per jam, menunjukkan fase pemanasan operasi hingga mendekati  $39\text{ }^{\circ}\text{C}$  pada akhir uji. Sebaliknya, frekuensi getaran relatif stabil di sekitar  $7\text{--}9\text{ Hz}$  dengan simpangan baku  $1,58\text{ Hz}$  dan tidak menunjukkan tren kenaikan yang berkelanjutan. Temuan ini mendukung kesimpulan bahwa kondisi bantalan berada pada kategori baik menurut ambang yang diterapkan; indikator LED/LCD aktif konsisten, dan buzzer tidak terpicu.

## KESIMPULAN

Sistem pemantauan *bearing* berbasis IoT yang dikembangkan pada penelitian ini berhasil mengukur dan mengirimkan data getaran serta suhu secara *real-time* ke *ThingSpeak*, sekaligus menampilkan status kondisi melalui LED/LCD dan *buzzer*. Pada pengujian 3 jam, nilai rata-rata getaran tercatat  $7,25\text{ Hz}$  dan suhu  $35,87\text{ }^{\circ}\text{C}$ ; seluruh pembacaan berada pada kategori baik sesuai ambang yang ditetapkan, tanpa memicu *buzzer*.

## Keunggulan

1. Arsitektur *end-to-end* yang sederhana dan berbiaya rendah, menggabungkan multi-sensor (getaran dan suhu), akuisisi di Arduino, konektivitas ESP8266, hingga penyimpanan/visualisasi cloud.
2. Pemantauan *real-time* jarak jauh melalui *ThingSpeak*, memungkinkan rekam jejak historis untuk evaluasi kondisi.
3. Indikator lapangan (LED/LCD/*buzzer*) yang langsung memberi notifikasi dini tanpa harus membuka *dashboard*.
4. Parameter implementasi yang jelas (*mis. sampling getaran, interval kirim*) sehingga mudah direplikasi di bengkel/laboratorium.

## Keterbatasan

1. Ambang indikator masih bersifat *heuristik* dan belum diturunkan dari basis data yang luas.
2. Ketergantungan pada *Wi-Fi* dan batas interval unggah *ThingSpeak* berpotensi menimbulkan kehilangan paket pada jaringan yang tidak stabil.

## Kontribusi ilmiah

1. Bukti eksperimental berkelanjutan yang menunjukkan kestabilan pembacaan dan aliran data beserta ringkasan metrik kuantitatif (rata-rata getaran dan suhu) sebagai dasar evaluasi kondisi.
2. Skema indikator kondisi yang praktis dan mudah direplikasi untuk mendukung perawatan prediktif di lingkungan pendidikan dan bengkel.

## Arah pengembangan

1. Melakukan kalibrasi dan validasi akurasi terhadap alat referensi (tachometer optik/termokopel) serta melaporkan metrik error (MAE, RMSE, galat relatif).
2. Memperluas pengujian ke durasi yang lebih panjang, variasi beban, dan skenario kerusakan terkontrol untuk melihat sensitivitas deteksi.
3. Menambah fitur analitik: RMS, FFT, envelope, dan normalisasi terhadap putaran, agar lebih peka pada cacat awal.
4. Menyusun ambang dan model peringatan berbasis data (*mis. pembelajaran mesin/anomali*) dari dataset waktu nyata.
5. Meningkatkan keandalan komunikasi (*mis. MQTT dengan buffer offline dan retry*) serta penguatan perangkat keras (braket sensor, proteksi, catu cadangan).
6. Mengembangkan dashboard dan notifikasi yang lebih kaya (*trigger otomatis, email/SMS*) serta mempertimbangkan penambahan sensor lain bila diperlukan.

## Potensi aplikasi

Sistem ini relevan untuk pendidikan vokasi, bengkel perawatan, dan industri skala kecil–menengah yang membutuhkan pemantauan sederhana namun *real-time* sebagai langkah awal menuju perawatan prediktif berbasis data.

## REFERENCE

- [1] G. Fu, A. Zheng, Y. Zhao, and H. Liu, "Research and Design of Automatic Pipe Blanking Device Based on Adaptive Control," *Acad. J. Sci. Technol.*, vol. 4, no. 2, pp. 133–136, 2022, doi: 10.54097/ajst.v4i2.4189.
- [2] S. Kurniawan and J. Aldo, "ANALISIS KEGAGALAN PADA PROSES PEMOTONGAN MATERIAL MENGGUNAKAN MESIN TRULASER 5040," *Armatur*, vol. 04, no. September, pp. 164–174, 2023, doi: 10.094.63/armatur.v4i2.3932.
- [3] B. Lee and K. Saitou, "Assembly Synthesis With Subassembly Partitioning for Optimal In-Process Dimensional Adjustability," *AI EDAM*, vol. 21, no. 1, pp. 31–43, Jan. 2007, doi: 10.1017/S0890060407070126.
- [4] F. Yang, P. Zhang, X. Zhang, J. Cao, and Y. Xing, "Modeling of the Variation Propagation for Complex-Shaped Workpieces in Multi-Stage Machining Processes," *Machines*, vol. 11, no. 6, p. 603, 2023. doi: 10.3390/machines11060603.
- [5] A. Y. Rahmawati, "Pengaruh Penggunaan Kendali Ketinggian



- Torch Dalam Proses Pemotongan Plat Baja St 37 Menggunakan Cnc Plasma Cutting Terhadap Nilai Kekasaran Permukaan Dan Topografi Permukaan Hasil Potongan,” 2020, pp. 1–23.
- [6] T. Trzepieciński, S. M. Najm, and H. G. Lemu, “Current Concepts for Cutting Metal-Based and Polymer-Based Composite Materials,” *Journal of Composites Science*, vol. 6, no. 5, p. 150, 2022. doi: 10.3390/jcs6050150.
- [7] B. Suhardi, *Perancangan Sistem Kerja dan Ergonomi Industri*. 2016.
- [8] Y. Chang, J. Zhang, S. Han, X. Li, and S. Yu, “Influence of Cutting Process on the Flanging Formability of the Cut Edge for DP980 Steel,” *Metals*, vol. 13, no. 5, p. 948, 2023. doi: 10.3390/met13050948.
- [9] Y. Yetri, O. Ganumba, A. Kurnia, D. Budiman, and J. Junaidi, “Rancang Bangun Welding Fixture Table Untuk Pemotongan Dan Pegelasan Plat,” *Manutech J. Teknol. Manufaktur*, vol. 14, pp. 103–109, Dec. 2022, doi: 10.33504/manutech.v14i02.217.
- [10] A. S. Wibowo, T. A. Kristiawan, and A. Hidayat, “Rancang Bangun dan Analisis S creen Cutting Machine Guna Meningkatkan Produktivitas Pemotongan,” *J. Mech. Eng. Appl. Technol.*, vol. 1, no. 1, pp. 13–18, 2023.
- [11] and S. [X. Xu, Hengzhou, Z. Runze, *Manual Cutting Machine*, 2nd ed. 2021.
- [12] N. H. Ummah and S. S. Dahda, “Analisis Efektifitas Kinerja Mesin Cutting Manual Dan Otomatis Menggunakan Metode OEE ( Overall Equipment Effectiveness ) Di PT . XYZ,” *J. Tek. Ind.*, vol. 8, no. 2, pp. 345–354, 2022, doi: 10.24014/jti.v8i2.19765.
- [13] E. Riyadi and E. Kusumawati, “Rancang Bangun Sliding Cutting Jig Guna Mengoptimalkan Fungsi Kerja Mesin Gerinda Tangan Sebagai Alat Potong Plat Lembaran,” *J. Pengelolaan Lab. Pendidik.*, vol. 4, pp. 82–89, Jun. 2022, doi: 10.14710/jplp.4.2.82-89.
- [14] M. D. Ellianto, Y. E. Sariski; Nurcahyo, and K. Rizqi, “Rancang Bangun Mesin Pemotong Keramik dengan Sistem Sliding,” *J. Mech. Manuf. Technol.*, vol. 3, no. 1, pp. 61–68, 2022, doi: 10.35891/jmmt.v3i1.3000.
- [15] Okpatrioka, “Research And Development ( R & D ) Penelitian Yang Inovatif Dalam Pendidikan,” vol. 1, no. 1, pp. 86–100, 2023.
- [16] A. D. Mustaqiem, “Analisis Perbandingan Faktor Keamanan Rangka Scooter Menggunakan Perangkat Lunak Solidwork 2015,” *J. Tek. Mesin Mercu Buana*, vol. 9, no. 3, pp. 164–172, 2020, doi: 10.22441/jtm.v9i3.9567.
- [17] M. Ficki, K. Kardiman, and N. Fauji, “SIMULASI BEBAN RANGKA PADA MESIN PENGILING SEKAM PADI MENGGUNAKAN PERANGKAT LUNAK,” *ROTOR*, vol. 15, p. 44, Nov. 2022, doi: 10.19184/rotor.v15i2.32447.
- [18] B. Siever and M. P. Rogers, “An IoTa of IoT,” in *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, in SIGCSE '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 742. doi: 10.1145/3017680.3017820.
- [19] A. C. F. J. M. G. M. Iera, “The Internet of Things [Guest Editorial],” *IEEE Wirel. Commun.*, vol. 17, no. 6, pp. 8–9, 2010, doi: 10.1109/MWC.2010.5675772.
- [20] M. Hussein, A. I. Galal, E. Abd-Elrahman, and M. Zorkany, “Internet of Things (IoT) Platform for Multi-Topic Messaging,” *Energies*, vol. 13, no. 13, p. 3346, 2020. doi: 10.3390/en13133346.
- [21] A. N. A. Zumaroh *et al.*, “Development of Application Programming Interface (API) for AMIKOM Purwokerto Hand Sanitizer (AMPUH) Data Logger Visualization,” *J. Tek. Inform.*, vol. 3, no. 3, pp. 791–796, 2022, doi: 10.20884/1.jutif.2022.3.3.222.