

Internet of Things Analytics Menggunakan ELK Stacks Dalam Pabrik Perangkat Server

Kent Bryant¹, Melisa Mulyadi^{2*}, Lanny Panjaitan³, Maria Angela Kartawidjaja⁴,
Ferry Rippun G.M⁵

^{1,2,3,4,5}Prodi Teknik Elektro, Fakultas Teknik
Universitas Katolik Indonesia Atma Jaya

Article Info

Article history:

Received
05 12 2023

Accepted
09 12 2023

Keywords:

DHT11, ELK, Internet
of Things, micro-
controller, monitoring

Abstract

The internet is an information and communication technology that makes it easier for human activities to access data and send information into networks. For industry, the internet is also used to store data both internally and externally. There are several application facilities that are open source to help the industry save data via the internet to the cloud, one of which is the ELK stack. In this research, a simulation of a temperature and humidity monitoring system is designed for a server device factory and simultaneously calculates the amount of production. The data obtained is sent to Elasticsearch as a cloud so that the measurement data can be monitored and visualized in the form of graphs. The system is realized using a microcontroller, ESP8266, DHT11 sensor and proximity sensor. This design is equipped with an ELK system that can collect data and filter the required data so that it can be monitored and visualized in real time. The results of testing the temperature and humidity obtained are in the optimal value range and also proves that this research can be used on a large scale. It is recommended that this design be implemented in a factory or smart home to make it easier to obtain and retrieve the required data.

Info Artikel

Histori Artikel:

Diterima:
05 12 2023

Disetujui:
09 12 2023

Kata Kunci:

DHT11, ELK, Internet
of Things, micro-
controller, monitoring

Abstrak

Internet merupakan teknologi informasi dan komunikasi yang mempermudah aktivitas manusia untuk mengakses data dan mengirimkan informasi ke dalam jaringan. Bagi industri, internet juga dimanfaatkan untuk menyimpan data baik internal maupun eksternal. Terdapat beberapa fasilitas aplikasi yang bersifat open source untuk membantu industri menyimpan data melalui internet ke cloud, salah satunya ELK stack. Pada penelitian ini dirancang simulasi sistem monitoring suhu dan kelembapan untuk pabrik perangkat server dan sekaligus menghitung jumlah produksi. Data yang diperoleh dikirimkan ke Elasticsearch sebagai cloud sehingga data pengukuran dapat dimonitor dan divisualisasikan dalam bentuk grafik. Sistem direalisasi menggunakan mikrokontroler, ESP8266, sensor DHT11 dan sensor proximity. Perancangan ini dilengkapi dengan sistem ELK yang dapat mengumpulkan data serta menyaring data yang dibutuhkan sehingga dapat di monitoring dan divisualisasi secara real time. Hasil pengujian suhu dan kelembapan yang didapat masuk dalam rentang nilai optimal dan juga membuktikan bahwa penelitian ini dapat digunakan untuk skala besar. Sebaiknya perancangan ini diimplementasikan pada pabrik atau smart home agar lebih mudah untuk mendapatkan serta mengambil data yang dibutuhkan.

1. PENDAHULUAN

Pada abad ke-21, perkembangan ilmu pengetahuan Teknologi Informasi dan Komunikasi atau yang lebih dikenal dengan singkatan TIK meningkat pesat. Para ahli terus melahirkan inovasi baru yang mendukung kemajuan teknologi yang berkelanjutan. Internet menjadi bukti salah satu kemajuan teknologi yang mempermudah aktivitas manusia dan melahirkan masyarakat digital [1]. Keuntungan dari internet yaitu membuat akses data dalam jaringan menjadi lebih mudah. Masyarakat juga memanfaatkan internet

*Corresponding author : Melisa Mulyadi
Email address: melisa.mulyadi@atmajaya.ac.id

sebagai media untuk penyebaran informasi. Demikian pula bagi perdagangan/industri internet dimanfaatkan untuk penyimpanan data baik internal maupun eksternal.

Di industri yang memproduksi perangkat *server* memerlukan kondisi ruang produksi dengan kelembapan dan suhu tertentu agar tidak menimbulkan hal-hal yang fatal. Biasanya server membutuhkan kondisi ruang dengan suhu pada 18-27 °C [2] dan kelembapan pada 40-55% rH [3].

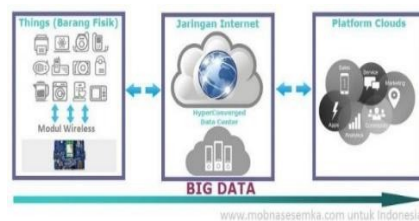
Untuk menangani hal tersebut, dibutuhkan sistem yang dapat memantau suhu dan kelembapan secara *realtime*. Pada penelitian ini, sistem yang demikian akan disimulasikan dengan ELK stack. ELK stack adalah suatu sistem *open-source* yang merupakan kombinasi dari *elasticsearch*, *logstash* dan *kibana*. Sistem ELK stack dapat mencatat data log dari sejumlah sensor secara *real-time* dan mengolahnya untuk divisualisasi menjadi beragam bentuk grafik.

Penelitian ini disimulasikan pada ruang *server* PT. X menggunakan *software Elasticsearch* yang dibantu dengan sensor kelembapan udara dan suhu yaitu sensor DHT11. Simulasi ini menggunakan jaringan lokal dan *cloud*. Jaringan lokal berfungsi sebagai komunikasi antar perangkat yang digunakan sedangkan *cloud* digunakan untuk penyimpanan data serta visualisasi dari data yang diterima.

2. TEORI DASAR

2.1 Internet of Things

Internet of Things atau IoT merupakan suatu deskripsi dari jaringan yang menghubungkan sensor, *software*, dan juga teknologi lain agar dapat menyimpan data di *cloud* dengan menggunakan internet [4]. Dalam penelitian ini, IoT digunakan agar perusahaan dapat menerima data produksi secara *real-time* dengan menggunakan jaringan internet, yang nantinya akan dianalisis di tempat lain yang terkoneksi dengan jaringan internet. Gambar 1 menunjukkan konsep dari *Internet of Things* (IoT) [5]



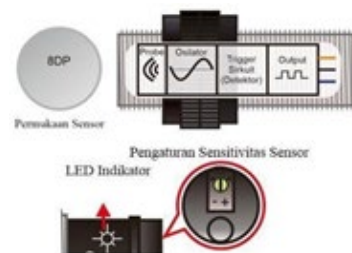
Gambar 1. Konsep IoT [5]

2.2 Sensor Proximity

Sensor proximity yang digunakan bersifat kapasitif. Sensor ini adalah salah satu jenis sensor yang dapat digunakan untuk mendeteksi ada atau tidaknya suatu objek tanpa adanya kontak fisik [6]. Simbol dan bentuk fisik sensor kapasitif ditunjukkan pada Gambar 2 sedangkan modul sensor kapasitif ditunjukkan pada Gambar 3. Objek yang dapat terdeteksi oleh sensor Kapasitif proximity yaitu metal maupun non-metal.



Gambar 2. Sensor proximity kapasitif



Gambar 3. Modul sensor proximity kapasitif

2.3 Sensor Suhu dan Kelembapan(DHT11)

Sensor Suhu dan Kelembapan yang digunakan merupakan sensor DHT11 yang merupakan sensor yang memiliki kalibrasi sinyal digital yang mampu memberikan informasi suhu dan kelembapan. Sensor ini tergolong komponen yang memiliki tingkat stabilitas yang sangat baik.

2.4 ELK Stacks

ELK Stacks merupakan aplikasi *open-source* yang berfungsi untuk melakukan pengindeksan data dan memvisualisasikannya menjadi sebuah grafik, metrik, tabel atau gambar. ELK Stacks terdiri dari 3 aplikasi *open-source*, yaitu: *Elasticsearch*, *Logstash*, dan *Kibana*. Dapat dilihat pada Gambar 4 yang merupakan alur dari proses pengolahan data menggunakan ELK Stacks.



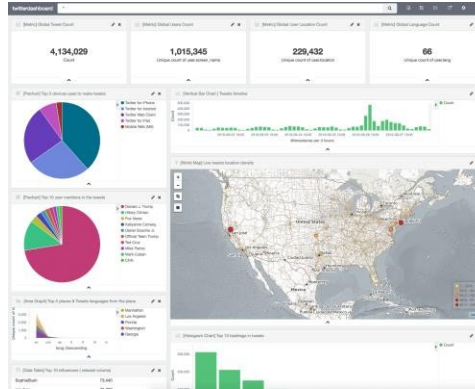
Gambar 4. Alur pengolahan data ELK Stacks

Elasticsearch merupakan sebuah mesin mencari dan analisis untuk semua jenis data termasuk teks, numerik, geospasial, terstruktur dan tidak terstruktur dari perangkat dan data log yang telah ditentukan. Analisis yang dilakukan *Elasticsearch* terdiri dari dua proses yaitu mengatur urutan data dan mengorganisasikannya ke dalam suatu pola, kategori, serta satuan uraian dasar. Pada *Elasticsearch* terdapat proses *data ingestion* yang memproses data mentah melalui tahapan penguraian, normalisasi, dan diperkaya dengan data tambahan sebelum dilakukan proses pengindeksan.

Logstash merupakan pipa pemrosesan data yang memungkinkan untuk mengambil data dari berbagai sumber secara bersamaan dan mengubahnya ke dalam format JSON, sebelum diindeks ke dalam *Elasticsearch*. Format JSON atau *JavaScript object notation* yang dimaksud adalah format yang digunakan dalam transfer dan penyimpanan data. Format JSON memiliki *file* yang lebih ringan dibandingkan dengan XML, Format JSON memiliki struktur kode yang lebih sederhana dan mudah untuk dipahami.

Filebeat adalah pengirim data log untuk *file local*. *Filebeat* berperan sebagai agen pada sebuah server yang berguna memonitor direktori log atau *file log* yang telah ditentukan, mengumpulkan log, dan meneruskannya ke *Logstash* maupun langsung ke *Elasticsearch* untuk memulai melakukan pengindeksan.

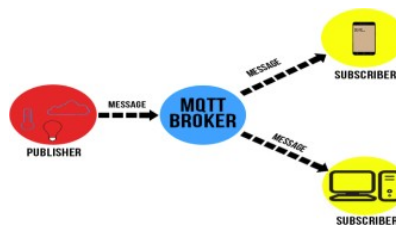
Kibana merupakan sebuah sistem visualisasi dan manajemen data untuk *Elasticsearch*. Data dapat divisualisasikan dalam berbagai bentuk berupa grafik, metrik, table, ataupun gambar. *Dashboard* merupakan salah satu fitur yang disediakan oleh *kibana*, fitur ini berfungsi untuk mengumpulkan data yang telah divisualisasikan ke dalam satu halaman dan dapat dikonfigurasi sesuai kebutuhan. *Kibana* juga bertindak sebagai antarmuka pengguna untuk memantau data dari *Elasticsearch* yang telah dikonfigurasi kebutuhannya. Dapat dilihat pada Gambar 5 merupakan contoh dari *dashboard* yang merupakan fitur yang disediakan *Kibana*.



Gambar 5. Dashboard Kibana

2.5 MQTT Mosquitto Broker

Message Queue Telemetry Transport (MQTT) merupakan protokol yang ringan dan sederhana untuk digunakan sehingga dapat melakukan komunikasi dengan skala besar. MQTT bekerja pada layer *transport* dan menggunakan konsep *server publish/subscribe* dimana *publish* merupakan *client* yang mengirimkan data sedangkan *subscribe* merupakan server atau *client* yang menerima data dari pihak pengirim. Protokol bekerja dengan metode *machine-to-machine* (M2M) dan menggunakan TCP/IP pada layer aplikasi sehingga komunikasi antar *client* berpusat pada server yang biasa disebut dengan MQTT *broker*.

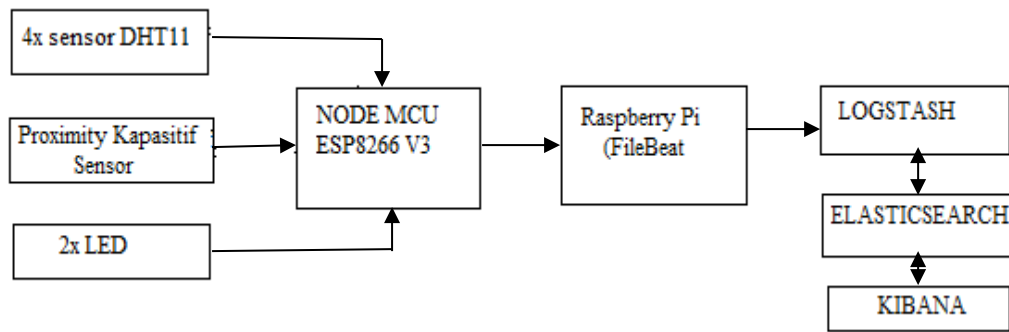


Gambar 6. Konsep kerja MQTT

Broker Mosquitto merupakan salah satu *broker/server* yang digunakan pada protokol MQTT karena dapat menyediakan standard implementasi untuk *server* dan *client* MQTT. *Mosquitto* dapat digunakan dalam segala situasi terutama dalam kebutuhan mengolah pesan ringan apda perangkat dengan sumber daya yang terbatas seperti *Raspberry pi*.

3. METODOLOGI PENELITIAN

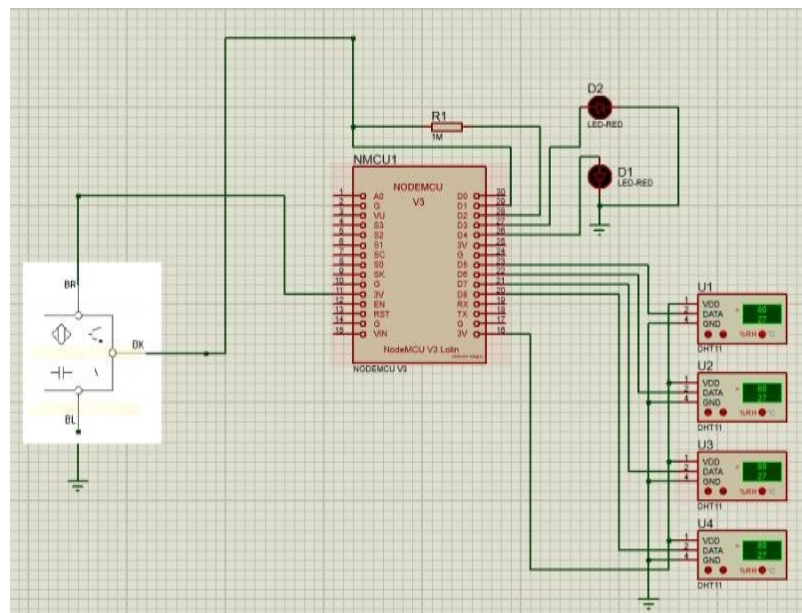
Untuk sistem perancangan ini, NodeMCU ESP8266 menerima data dari 4 buah sensor untuk mendeteksi suhu serta kelembapan yang kemudian dikirimkan ke Raspberry Pi secara *wireless* dalam jaringan yang sama. Pada Raspberry Pi telah di-*install* sebuah agen bernama *filebeat* yang berfungsi sebagai penerima data log sensor, mengumpulkan data log yang dibutuhkan, dan juga mengirimkan datalog menuju *logstash*. *Logstash* bekerja untuk mengumpulkan seluruh data log dari berbagai *server* yang kemudian format dari data log tersebut dikonversikan menjadi format JSON yang merupakan format yang memiliki struktur kode yang lebih sederhana dibandingkan format lainnya sebelum dilakukan indeks pada proses *Elasticsearch* yang merupakan tahap utama dari perancangan ini. Pada *Elasticsearch* terdapat proses data *ingestion* yang memproses data mentah melalui tahapan penguraian, normalisasi, dan diperkaya dengan data tambahan sebelum dilakukan proses indeks. Proses indeks dilakukan pada tahap *Elasticsearch* yang merupakan mesin pencari dan analisis dari data yang telah diterima. Tahap *kibana* merupakan tahap untuk visualisasi seluruh data yang telah diindeks pada *Elasticsearch*.



Gambar 7. Diagram blok perancangan

3.1 Perancangan Perangkat Keras

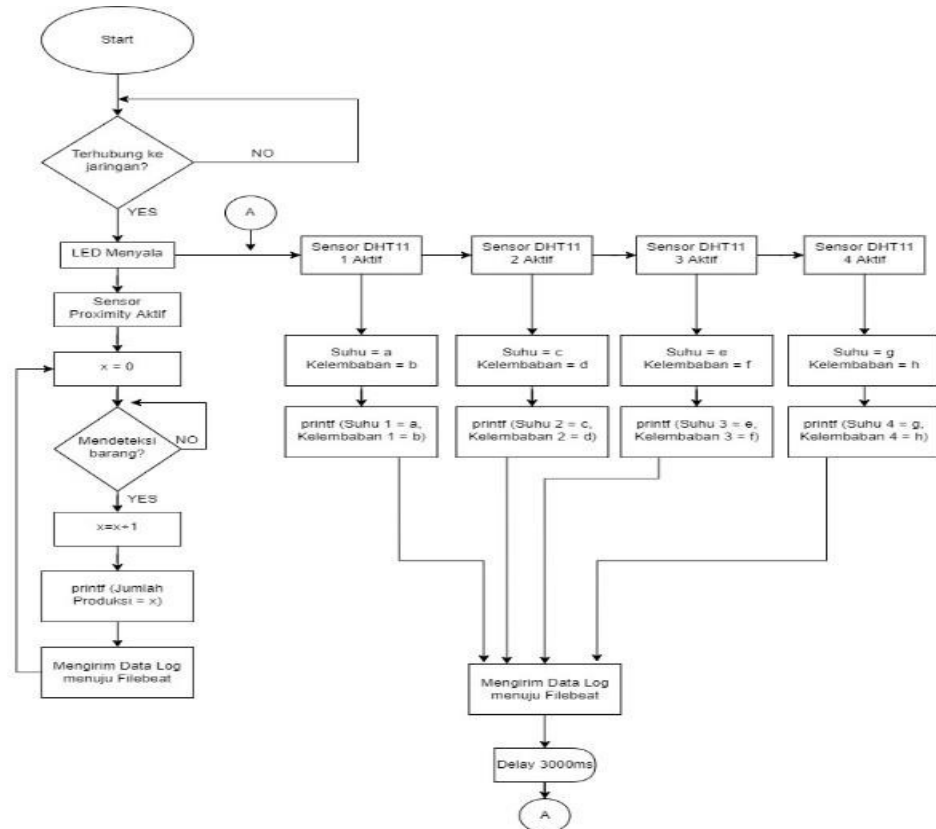
Pada penelitian ini komponen utamanya adalah mikrokontroler NodeMCU ESP8266 untuk mengontrol semua sensor. LED terhubung dengan *port* D4 atau GPIO2 yang berfungsi sebagai indikator menunjukkan NodeMCU ESP8266 telah terhubung dengan jaringan internet. Terdapat 4 sensor suhu dan kelembapan (DHT 11) yang terhubung dengan *port* D5 (GPIO14), D6 (GPIO12), D7 (GPIO13), dan D8 (GPIO15). Sensor *proximity* berfungsi sebagai penghitung barang yang dihubungkan dengan port D1 (GPIO5) dan port D2 (GPIO4) dengan tambahan hambatan $1M\Omega$. Karena pada simulasi ini tidak dibutuhkan data analog, data yang digunakan hanya data digital yang mengidentifikasi apakah sensor tersebut aktif atau tidak. Ketika status port D2 (GPIO4) berubah dari LOW menjadi HIGH, akan membuat input D1 (GPIO5) akan tetap pada status LOW untuk sementara waktu. Sistem akan menambahkan penghitungan secara otomatis bila status pin D1 berubah menjadi HIGH dan terdapat LED pada pin D3 sebagai indikator yang akan menyala bila sensor mendeteksi sebuah benda..



Gambar 8. Skematik perangkat

Sistem ini dimulai dengan memastikan perangkat NodeMCU telah terhubung dengan jaringan internet yang telah ditentukan. Sebagai indikator lampu LED akan menyala bila telah terhubung dengan jaringan internet dan akan mati bila jaringan terputus. Sensor *proximity* digunakan untuk menghitung jumlah produksi barang. Jumlah barang didefinisikan sebagai x . Saat belum ada barang, maka $x = 0$ dan bila terdapat barang yang melewati sensor *proximity* maka x akan menjadi 1 dan kembali 0 bila objek telah melewati sensor. Perhitungan akan diulang dari 0 bila *timestamp* pada *elasticsearch* telah berganti

hari. Kondisi suhu dan kelembapan dideteksi oleh DHT11. Sensor ini memiliki *delay* sebesar 3000 ms setelah mengirim data ke Filebeat untuk mendeteksi kembali suhu dan kelembapan.



Gambar 9. Diagram alir perangkat keras

3.2 Perancangan Perangkat Lunak

Data yang dihasilkan oleh setiap sensor akan diberikan sebuah topik yang berbeda-beda dan dikirimkan menuju Raspberry Pi dalam bentuk string dengan MQTT broker yaitu Mosquitto. Pada perancangan ini topik yang diberikan adalah sensor1 (Sensor DHT11), sensor2 (Sensor DHT11), sensor3 (Sensor DHT11), sensor4 (Sensor DHT11), dan product (Sensor Proximity Kapasitif).

```
if (client.publish("/esp8266/sensor1",
String(sensor1).c_str())) {
Serial.println("Sensor DHT11 1 sent!"),
}
```

Gambar 10. Konfigurasi topik

Pada *Raspberry pi*, penyimpanan berkas log terdapat pada folder `/var/log/sensor`. Untuk log yang telah diterima *client* disimpan dengan cara memasukkan *command* “`mosquitto sub -h (ip server) -t (topik) >> /var/log/sensor/(nama berkas).log &`” pada terminal maka log yang diterima akan tersimpan pada berkas tersebut secara otomatis.

Filebeat diletakkan pada *Raspberry Pi* dan dilakukan konfigurasi agar terhubung dengan *elastic cloud*. Konfigurasi pertama yang dilakukan adalah menetapkan host id, cloud id, dan password cloud yang dilakukan pada berkas `filebeat.yml`. Host id didapatkan pada URL project yang telah disediakan oleh *Elasticsearch*. Untuk cloud id dan password credential diberikan oleh *elasticsearch* pada pembuat project pada cloud. Untuk log yang telah diterima dari Arduino diteruskan menuju *elasticsearch* melewati *filebeat*. Untuk konfigurasi ini, dilakukan pada berkas `filebeat.yml` bagian `file beat.input`. Yang dikonfigurasi pada bagian ini adalah *type* dan *paths*. Untuk *type* memiliki beberapa pilihan, pada

perancangan ini dipilih log karena yang diterima oleh raspberry adalah data log. Untuk *path* dilengkapi dengan lokasi berkas dari log pada setiap sensor yang telah disimpan pada Raspberry.

Data yang diterima oleh Raspberry merupakan data dalam bentuk string sedangkan pada *kibana* dibutuhkan data dalam bentuk float dimana data tersebut berguna untuk membuat visualisasi. Oleh karena itu data yang telah diterima diubah menjadi float. Konfigurasi untuk mengubah data dari string menjadi float terdapat pada *filebeat.yml*. Data yang diterima oleh *elasticsearch* merupakan data yang telah terpisah menjadi suhu dan kelembapan pada setiap sensornya. Sedangkan untuk sensor *Proximity Capacitive* yang merupakan *counter* tidak perlu dipisah karena hanya menghasilkan 1 data log. Data hasil produksi, penghitung terdapat langsung dari *kibana*. *Kibana* akan menambahkan jumlah log yang diterima dari sensor *Proximity capacitive* secara *real time* dan di konfigurasi untuk penghitungan selama 1 hari. Visualisasi yang dibuat merupakan data setiap sensor secara *real-time*, untuk setiap sensor DHT1, dan *Line Graph* untuk *Temperature* dan *Humidity*. Tabel akan mengeluarkan data baru setiap 5 detik yang dihasilkan dari setiap sensor. *Line graph* merupakan rata-rata dari nilai yang didapatkan dalam kurun waktu 1 menit sedangkan *counter* hanya memvisualisasikan jumlah produksi secara *real-time*.

4. PEMBAHASAN DAN HASIL

Alat yang dirancang ditaruh pada ruangan *server* selama beberapa hari, kemudian pendingin ruangan dimatikan selama beberapa menit dan dinyalakan kembali agar dapat melihat hasil dari sensor DHT11 yaitu temperatur dan kelembapan. Hasil pengujian menunjukkan bahwa ke-empat sensor DHT11 berjalan dengan baik, hal ini dapat dilihat dari hasil percobaan yang terdapat pada *kibana* saat pendingin dimatikan, nilai temperatur yang dihasilkan oleh sensor DHT11 meningkat serta kelembapan pada daerah tersebut berkurang, hal ini dimulai pada waktu menunjukkan 18.30 sampai dengan 18.38. Terjadi penurunan suhu dan kenaikan kelembapan setelah pendingin kembali dinyalakan pada waktu menunjukkan 18.39. Pengujian sensor *proximity* dilakukan dengan menggunakan cara melewati beberapa objek agar dapat terdeteksi oleh sensor, bila sensor mendeteksi barang tersebut maka penghitungan akan bertambah pada *kibana* dan perhitungan akan kembali menjadi 0 bila hari berganti.

Time per minut	Temperature	Humidity
18:30	25.50	42.00
18:30	25.80	41.00
18:30	25.80	40.00
18:30	25.80	39.00
18:31	25.30	42.00
18:32	25.30	42.00
18:33	25.40	42.00
18:33	25.30	42.00
18:34	25.80	42.00
18:35	25.80	42.00
18:36	26.20	42.00
18:36	25.90	42.00
18:36	25.80	42.00
18:37	26.20	42.00
18:38	26.20	44.00
18:38	26.20	43.00
18:38	26.20	42.00
18:39	25.30	47.00
18:39	25.30	46.00
18:39	25.80	46.00
18:39	25.80	45.00
18:39	25.80	44.00
18:40	24.80	52.00

Gambar 11. Hasil pengujian DHT11



Gambar 12. Visualisasi perhitungan Real-Time

5. KESIMPULAN

Berdasarkan hasil pengujian dapat disimpulkan sebagai berikut:

1. Mikrokontroler dapat mengolah data dari sensor suhu, kelembapan dan sensor proximity dan mengirimkan data tersebut ke internet dengan baik.
2. Simulasi menggunakan ELKstack dapat memantau kondisi kerja yaitu suhu, kelembapan dan menghitung hasil produksi kemudian menampilkan data tersebut secara *real time*.

Penulis menyarankan untuk menggunakan sensor dengan kualitas yang lebih baik agar data yang didapatkan lebih akurat dan valid. Sebaiknya simulasi ini diimplementasikan pada pabrik atau *smart home* agar lebih mudah untuk mendapatkan serta mengambil data yang dibutuhkan.

DAFTAR PUSTAKA

- [1] M. Ngafifi, "Kemajuan Teknologi Dan Pola Hidup Manusia Dalam Perspektif Sosial Budaya," *Jurnal Pembangunan Pendidikan Fondasi dan Aplikasi*, vol. 2, no. 1, 2014.
- [2] M. F. Awaj, A. F. Rochim and E. D. Widiyanto, "Sistem Pengukur Suhu dan Kelembaban Ruang Server," *Jurnal Teknologi dan Sistem Komputer*, vol. 2, no. 1, pp. 40-47, 2014.
- [3] J. M. Waworundeng, O. Dumanaw and T. Rumawouw, "Prototipe Detektor Suhu dan Kelembaban Berbasis IoT di Ruang Server Sistem Informasi Universitas Klabat," *Cogito Smart Journal*, vol. 7, no. 1, 2021.
- [4] Panduardi, F., & Haq, E. S. (2016), "Wireless Smart Home System Menggunakan Raspberry Pi", *Jurnal Teknologi Informasi dan Terapan*, vol 3, no. 1, pp. 320-325, 2016.
- [5] Mobile," *Jurnal Ilmiah Ilmu Komputer*, vol. 4, no. 1, 2018.
- [6] Instrument and Automation Engineers' Handbook. Volume I, Measurement and Safety