

# Rancang Bangun *Web Server* Sistem Akuisisi Data Kondisi Udara Ruang Berbasis Internet Menggunakan Raspberry Pi

Dwi Andre Rajendradika<sup>1</sup>, Pratikto<sup>2</sup>, Windy Hermawan Mitrakusuma<sup>3</sup>

<sup>1,2,3</sup>Jurusan Teknik Refrigerasi dan Tata Udara, Politeknik Negeri Bandung, Bandung 40012

<sup>1</sup>E-mail: dwi.andre.tptu418@polban.ac.id

<sup>2</sup>E-mail: pratikto@polban.ac.id

<sup>3</sup>E-mail: windyhm@polban.ac.id

## ABSTRAK

Sistem akuisisi data adalah sistem yang digunakan untuk memperoleh, mengumpulkan, dan menyiapkan data untuk diproses. Seiring dengan perkembangan teknologi, otomatisasi sistem akuisisi data diharapkan dapat membuat proses pengakuisisian data menjadi lebih efisien. Kondisi udara dan kualitas udara dalam ruangan dapat mempengaruhi kenyamanan, dan tingkat stres manusia saat beraktivitas di dalam ruangan. Kondisi udara dan kualitas udara dalam ruangan juga dapat mempengaruhi kondisi peralatan elektronik di dalamnya. Oleh karena itu, perlu dilakukan pengakuisisian data parameter-parameter kondisi udara dan kualitas udara di dalam ruangan. Tujuan dari penelitian ini adalah membuat sistem akuisisi data kondisi udara dalam ruangan berupa temperatur udara, kelembaban relatif udara, dan konsentrasi CO<sub>2</sub> dengan *web server* menggunakan Raspberry Pi 3b+, Arduino Mega 2560, sensor DHT11, dan sensor MQ135. *Microframework* Flask dan *library* Flask-SocketIO digunakan sebagai kerangka kerja *web server* dan penyedia komunikasi dua arah antara *server* dan *client* untuk transmisi data. Data hasil pengukuran disimpan pada penyimpanan lokal Raspberry Pi sebagai berkas CSV dan juga dikirimkan pada laman *Google Spreadsheet*. Data hasil pengukuran ditampilkan secara *realtime* pada laman web. Dari hasil pengujian, nilai galat absolut pengukuran temperatur dan kelembaban berturut-turut  $\pm 0.34$  °C dan  $\pm 0.66\%$ . Rata-rata *delay* transmisi data pada sistem adalah 387.87 milisekon dengan persentase *packet loss* 0%.

## Kata Kunci

Akuisisi Data, Raspberry Pi, Kondisi Udara Ruang

## 1. PENDAHULUAN

Kondisi udara dan kualitas udara dalam ruangan dapat mempengaruhi kenyamanan, kemampuan beraktivitas, dan tingkat stres manusia saat melakukan aktivitas di dalam ruangan. Paparan temperatur lingkungan kerja diatas nilai ambang batas memberikan tekanan pada pekerja dimana pekerja akan merasa tidak nyaman berada pada lingkungan kerjanya sendiri [1]. Kondisi udara dan kualitas udara dalam ruangan juga dapat mempengaruhi kondisi peralatan elektronik di dalamnya. Oleh karena itu, diperlukan sistem akuisisi data yang dapat melakukan pengukuran, pencatatan dan pemantauan parameter-parameter kondisi udara dan kualitas udara di dalam ruangan.

Sistem akuisisi data adalah sistem yang berguna untuk mengambil, mengumpulkan dan menyiapkan data, hingga memprosesnya

untuk menghasilkan data yang dikehendaki [2]. Seiring dengan perkembangan teknologi, otomatisasi sistem akuisisi data diharapkan dapat membuat proses pengakuisisian data menjadi lebih efisien dari segi sumber daya manusia, tenaga, maupun waktu.

Raspberry Pi adalah *single board computer* berdimensi sebesar kartu kredit yang dikembangkan oleh Raspberry Pi *Fondation*. Raspberry Pi memiliki beragam fungsi dalam penggunaannya, antara lain, *microcontroller*, *server*, dan alat untuk proyek *home automation*.

Tujuan dari penelitian ini adalah membuat sistem akuisisi data kondisi udara dalam ruangan berupa temperatur udara, kelembaban relatif udara, dan konsentrasi CO<sub>2</sub> dengan *web server* menggunakan Raspberry Pi 3b+,

Arduino Mega 2560, sensor DHT11, dan sensor MQ135. *Microframework* Flask dan *library* Flask-SocketIO digunakan sebagai kerangka kerja *web server* dan penyedia komunikasi dua arah antara *server* dan *client* untuk transmisi data hasil pengukuran. Data hasil pengukuran disimpan pada penyimpanan lokal Raspberry Pi sebagai berkas CSV (*comma separated values*) dan juga dikirimkan pada laman *Google Spreadsheet* agar dapat diakses secara daring dari jarak jauh. Data hasil pengukuran ditampilkan secara *realtime* pada laman web berupa tampilan nilai numerik dan tampilan grafik.

## 2. LANDASAN TEORI

### 2.1 Flask

Flask merupakan *microframework* berbasis Python untuk pengembangan aplikasi berbasis web. Flask dianggap sebagai *microframework* karena minimnya dependensi pada *library* Python lainnya dalam menjalankan fungsi tertentu.

### 2.2 Flask-SocketIO

Flask-SocketIO adalah *library* Python yang menyediakan aplikasi web berbasis Flask sistem komunikasi *full duplex* berlatensi rendah antara *server* dan *client* [3]. Penggunaan Flask-SocketIO diterapkan pada sisi *server*. Pada sisi *client*, digunakan *library* Socket.IO sebagai penyedia koneksi permanen ke sisi *server*. Protokol komunikasi WebSocket sebagai protokol komunikasi utama dan HTTP (*Hypertext Transfer Protocol*) *long polling* sebagai protokol komunikasi cadangan adalah protokol komunikasi yang digunakan pada Flask-SocketIO. Protokol komunikasi WebSocket memungkinkan komunikasi dua arah antara *client* dan *server* secara *realtime*. Flask-SocketIO memiliki dependensi terhadap *concurrent networking library* seperti Eventlet ataupun Gevent agar protokol komunikasi WebSocket dapat digunakan.

### 2.3 Raspberry Pi 3b+

Raspberry Pi adalah *single board computer* yang dikembangkan oleh *Raspberry Pi Foundation*. Raspberry Pi memiliki beragam fungsi dalam penggunaannya, antara lain, *microcontroller*, *server*, dan alat untuk proyek *home automation*. Tipe Raspberry Pi yang digunakan pada penelitian ini adalah Raspberry Pi model 3b+.

### 2.4 Website

*Website* adalah kumpulan halaman-halaman yang digunakan untuk mempublikasikan informasi berupa teks, gambar, dan program multimedia lainnya berupa animasi, suara, dan atau gabungan dari semuanya itu baik yang bersifat statis maupun dinamis yang membentuk suatu rangkaian bangunan yang saling terkait antara satu halaman dengan halaman yang lain yang sering disebut dengan *hyperlink* dan dapat diakses melalui protokol HTTP (*Hyper Text Transfer Protocol*) [4]. Arsitektur dari website adalah *client-server* yang terdiri dari *web server* dan *web client*.

### 2.5 Kenyamanan Termal dan Kualitas Udara Ruang

Kenyamanan termal adalah kondisi pikiran yang mengekspresikan keterpuasan dengan kondisi lingkungan termal yang dinilai oleh evaluasi subjektif [5]. Kenyamanan termal dipengaruhi oleh enam faktor, yaitu tingkat metabolisme tubuh, temperatur udara, temperatur radiant, kecepatan udara, kelembaban udara, dan insulasi pakaian yang dikenakan.

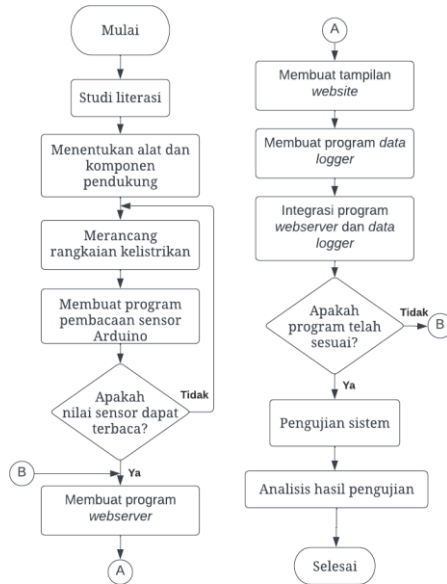
Menurut Kementerian Kesehatan Republik Indonesia yang tercantum pada Peraturan Menteri Kesehatan Republik Indonesia Nomor 1077 tentang Pedoman Penyehatan Udara Dalam Ruang Rumah, kualitas udara ruangan terdiri dari tiga kategori, yaitu kualitas fisik udara ruangan, kualitas kimiawi udara ruangan, dan kualitas biologi udara ruangan [6].

Kualitas fisik udara merupakan nilai parameter yang menunjukkan kondisi fisik udara di dalam ruangan seperti kelembaban, pencahayaan, temperatur, dan partikulat. Kualitas kimiawi udara merupakan nilai parameter yang menunjukkan kondisi udara berdasarkan kandungan unsur kimiawi pada udara seperti sulfur dioksida (SO<sub>2</sub>), nitrogen dioksida (NO<sub>2</sub>), ozon, karbon dioksida (CO<sub>2</sub>), timbal, asbestos, dan karbon monoksida (CO). Kualitas biologi udara adalah nilai parameter yang menunjukkan kondisi biologi udara dalam ruangan seperti keberadaan jamur dan bakteri.

## 3. METODE PENELITIAN

Metode yang digunakan pada penelitian ini adalah metode rancang bangun yang terdiri dari perancangan dan pembuatan perangkat

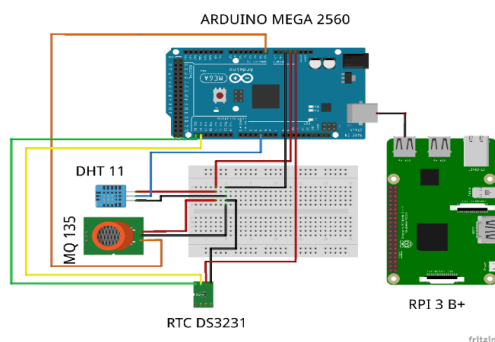
keras dan perancangan dan pembuatan perangkat lunak. Gambar 1 menunjukkan diagram alir tahapan penelitian yang dilakukan.



Gambar 1. Diagram alir tahapan penelitian

### 3.1 Perancangan Perangkat Keras

Perancangan perangkat keras yang dilakukan adalah perancangan rangkaian kelistrikan yang terdiri dari Arduino Mega 2560, Raspberry Pi 3B+, sensor temperatur dan kelembaban berupa DHT11, sensor konsentrasi CO<sub>2</sub> berupa MQ135, dan modul pencatat waktu dan tanggal berupa RTC (*Real Time Clock*) DS3231 seperti yang ditunjukkan gambar 2.



Gambar 2. Rangkaian kelistrikan

### 3.2 Perancangan Perangkat Lunak

Perancangan perangkat lunak terdiri dari pembuatan tiga program utama, yaitu

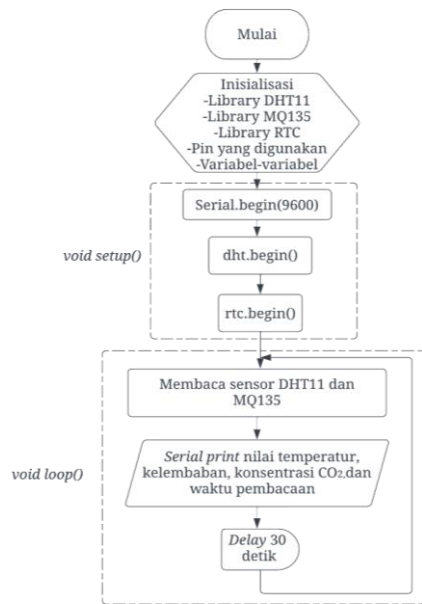
pembuatan program pembacaan sensor, program *data logger*, dan program *web server*.

#### A. Perancangan Program Pembacaan Sensor

Gambar 3 menunjukkan diagram alir dari algoritma program pembacaan sensor. Program ini adalah program yang dijalankan pada perangkat Arduino untuk membaca dan mengirimkan data hasil pengukuran kondisi udara ruangan berupa temperatur udara, kelembaban relatif udara, konsentrasi CO<sub>2</sub>, dan waktu pembacaan sensor.

Proses pertama yang dilakukan program pada saat pertama kali dijalankan adalah menginisialisasi *library* dan mendefinisikan variable-variabel yang diperlukan. Proses kedua adalah menjalankan perintah-perintah yang terdapat pada bagian *void setup()*. Perintah *Serial.begin(9600)* adalah perintah untuk menginisialisasi serial pada *baud rate* 9600. Perintah *dht.begin()* adalah perintah untuk menginisialisasi pembacaan sensor DHT11. Perintah *rtc.begin()* adalah perintah untuk menginisialisasi pembacaan waktu pada RTC DS3231.

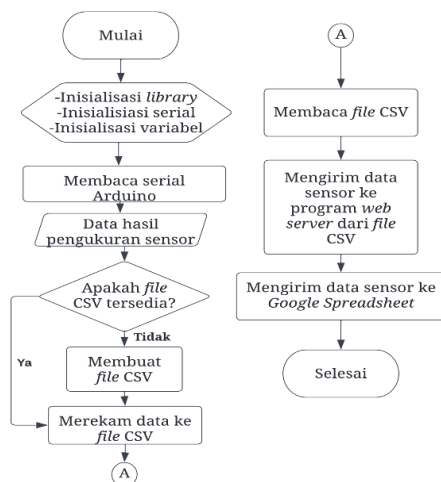
Setelah menjalankan perintah *void setup()*, Arduino akan menjalankan perintah pada bagian *void loop()* secara berulang. Arduino akan membaca nilai temperatur dan kelembaban dari sensor DHT11, nilai konsentrasi CO<sub>2</sub> dari sensor MQ135, dan waktu pembacaan sensor dari modul RTC DS3231. Hasil pembacaan sensor akan dikirimkan ke Raspberry Pi secara serial. Setelah itu, Arduino akan menunggu selama 30 detik untuk melakukan pembacaan berikutnya.



Gambar 3 Algoritma program pembacaan sensor

### B. Perancangan Program Data Logger

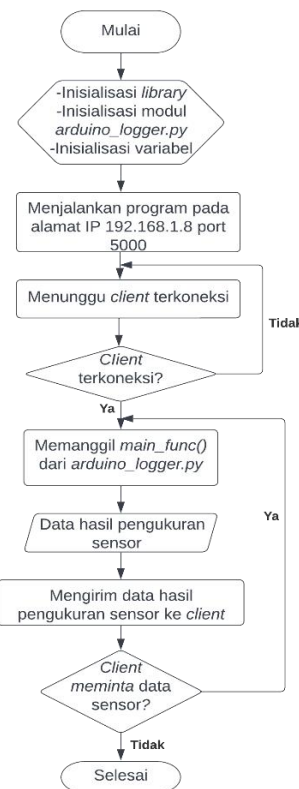
Gambar 4 menunjukkan algoritma dari program *data logger*. Program *data logger* akan bertindak sebagai modul yang dipanggil pada program *web server*. Program akan membaca data hasil pengukuran yang dikirimkan Arduino secara serial. Data hasil pengukuran direkam pada berkas CSV pada penyimpanan lokal Raspberry Pi. Program lalu mengirimkan data hasil pengukuran dari berkas CSV ke program *web server* dan ke laman *Google Spreadsheet*. Program akan dijalankan berulang selama program *web server* dijalankan.



Gambar 4 Algoritma program *data logger*

### C. Perancangan Program Web Server

Gambar 5 menunjukkan algoritma proses transmisi data hasil pengukuran dari program *web server* yang dibuat. Ketika program dijalankan, program akan menyediakan layanan pada alamat IP (*Internet Protocol*) 192.168.1.8 pada *port* 5000. *Server* akan menunggu *client* terkoneksi sebelum melakukan transmisi data. Setelah terkoneksi, *server* akan memanggil *main\_func()* yang terdapat pada modul program *data logger* dan mendapatkan data hasil pengukuran berbentuk *array*. Jika data hasil pengukuran berhasil terkirim, sisi *client* akan merespon dengan meminta data hasil pengukuran berikutnya. Proses transmisi data hasil pengukuran akan terus berulang selama *client* dan *server* saling terkoneksi.



Gambar 5 Algoritma program *web server*

### 3.3 Metode Pengujian Alat

Pengujian yang dilakukan pada sistem akuisisi data yang dibuat terdiri dari tiga jenis pengujian, yaitu pengujian galat pengukuran, pengujian *delay* transmisi data hasil pengukuran, dan pengujian *packet loss*. Pengujian dilakukan selama 30 menit dengan

jangka waktu pembacaan sensor 30 detik sekali. Pengujian dilakukan sebanyak dua kali percobaan. Percobaan pertama dilaksanakan siang hari pada pukul 14.39 WIB sampai dengan 15.09 WIB. Percobaan kedua dilaksanakan pada malam hari pukul 19.15 WIB sampai dengan 19.45 WIB.

Lokasi pengujian alat bertempat di ruangan dengan dimensi 3 m × 4 m × 3 m yang berfungsi sebagai kamar hunian mahasiswa. Ruang kamar tempat pengujian memiliki satu jendela dan ventilasi menghadap ke arah barat laut dengan jumlah penghuni satu orang.

#### A. Pengujian Galat Pengukuran

Pengujian galat (kesalahan) pengukuran yang dilakukan pada penelitian ini adalah dengan membandingkan hasil pengukuran temperatur dan kelembaban pada sensor DHT11 dengan alat ukur pembanding berupa termohigrometer. Jenis galat pengukuran yang diukur berupa galat pengukuran absolut dan galat pengukuran relatif. Persamaan (1) digunakan untuk memperoleh nilai galat pengukuran absolut. Nilai galat pengukuran relatif diperoleh dengan persamaan (2).

$$\text{Galat absolut} = |x - x_0| \quad \dots (1)$$

$$\text{Galat relatif} = \frac{|x - x_0|}{x_0} \cdot 100\% \quad \dots (2)$$

Keterangan:

$x$  = nilai hasil pengukuran alat yang diuji

$x_0$  = nilai alat ukur pembanding

#### B. Pengujian Delay Transmisi Data

Delay transmisi data adalah lama waktu yang dibutuhkan untuk mengirimkan data antara *server* dan *client*. Pada penelitian ini, *delay* transmisi data yang diukur adalah lama waktu yang dibutuhkan *server* untuk mengirimkan data hasil pengukuran menuju *client* yang diperoleh dari selisih antara waktu penerimaan data pada *client* dengan waktu pengiriman data dari sisi *server*. Persamaan (3) digunakan untuk memperoleh nilai *delay* transmisi data.

$$\text{Delay} = t_1 - t_0 \quad (3)$$

Keterangan:

$t_1$  = waktu penerimaan data (milisekon)

$t_0$  = waktu pengiriman data (milisekon)

#### C. Pengujian Packet Loss

*Packet loss* adalah perbandingan antara jumlah paket data yang gagal diterima dengan jumlah paket data yang dikirim. Pada penelitian ini *packet loss* didapat dari perbandingan jumlah data hasil pengukuran yang gagal diterima pada sisi *client* dengan jumlah data hasil pengukuran yang dikirim dari sisi *server*. Persamaan (4) digunakan untuk memperoleh presentase *packet loss*.

$$\text{Packet Loss} = \frac{pt - pr}{pr} \cdot 100\% \quad (3)$$

Keterangan:

$pt$  = jumlah paket data diterima

$pr$  = jumlah paket data dikirim

### 4. HASIL DAN PEMBAHASAN

#### 4.1 Hasil Rancang Bangun Perangkat Keras

Gambar 6 menyajikan kolase foto dari hasil perancangan perangkat keras berupa rangkaian kelistrikan. Rangkaian kelistrikan, modul RTC DS3231, dan unit Arduino ditempatkan di dalam kotak yang dapat ditutup. Gambar 6 juga menunjukkan penempatan sensor DHT11 dan sensor MQ135 yang ditempatkan pada sisi luar kotak. Hasil rancang bangun perangkat keras telah sesuai dengan desain awal rangkaian kelistrikan.

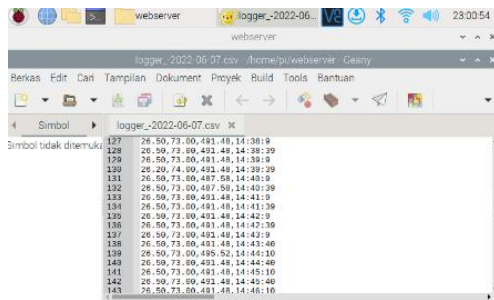


Gambar 6 Hasil perancangan perangkat keras

#### 4.2 Hasil Rancang Bangun Perangkat Lunak

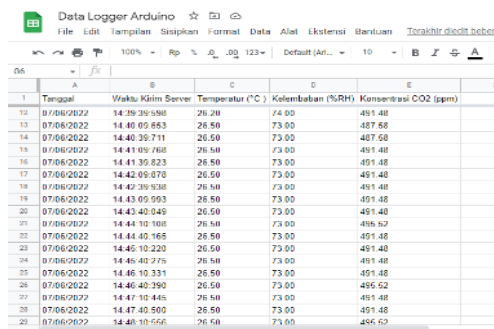
Gambar 7 menunjukkan bahwa sistem yang dibuat dapat melakukan perekaman data hasil pengukuran yang dikirim Arduino pada *file* CSV di dalam penyimpanan lokal Raspberry Pi. Terdapat empat parameter yang direkam, yaitu temperatur udara (°C), kelembaban relatif udara (% RH), konsentrasi CO<sub>2</sub> (ppm),

dan waktu pembacaan sensor. *File* CSV disimpan dengan nama sesuai dengan tanggal pembacaan sensor.

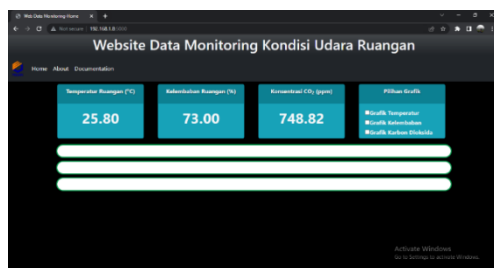


Gambar 7 Hasil Perrekaman Data pada File CSV

Gambar 8 menunjukkan bahwa sistem dapat mengirimkan data hasil pengukuran ke laman *Google Spreadsheet*. Tiap parameter yang diukur direkam pada kolom terpisah. Laman *Google Spreadsheet* yang memuat data hasil pengukuran dapat diakses menggunakan *link* yang dibagikan oleh pengguna.



Gambar 8 Hasil perekaman data pada *Google Spreadsheet*



Gambar 9 Halaman *Home* pada kondisi awal

Gambar 9 menunjukkan tampilan halaman *Home* pada *website* yang telah dibuat. *Website* dapat diakses pada perangkat lain di jaringan internet lokal yang sama dengan mengakses alamat <http://192.168.1.8:5000>. Halaman *Home* menampilkan data hasil pengukuran secara *realtime*. Data hasil pengukuran dapat ditampilkan dalam tampilan grafik dengan

mengeklik pilihan grafik yang ingin ditampilkan. Gambar 10 menunjukkan contoh tampilan grafik pada halaman *Home*.



Gambar 10 Tampilan grafik pada halaman *Home*

### 4.3 Hasil Pengujian Galat Pengukuran

Tabel 1 menunjukkan hasil pengujian galat pengukuran temperatur udara oleh sensor DHT11 pada sistem dengan membandingkan nilai hasil pengukuran temperatur DHT11 dengan alat ukur termohigrometer HTC-11. Dari pengujian yang dilakukan, didapatkan nilai rata-rata galat pengukuran absolut pada pembacaan temperatur oleh DHT11 adalah  $\pm 0.34$  °C. Nilai rata-rata galat pengukuran relatif pembacaan temperatur adalah 1.25%. Nilai galat absolut yang diperoleh dapat dikategorikan baik karena penyimpangan yang terjadi masih di bawah spesifikasi akurasi DHT11 untuk pembacaan temperatur, yaitu  $\pm 2$ °C.

Tabel 1 Pengujian galat pengukuran temperatur

Pengujian	Rata-rata galat pengukuran absolut (°C)	Rata-rata galat pengukuran relatif (%)
Percobaan 1	0.48	1.78
Percobaan 2	0.19	0.73
Rata-rata	0.34	1.25

Tabel 2 Pengujian galat pengukuran kelembaban relatif

Pengujian	Rata-rata galat pengukuran absolut (% RH)	Rata-rata galat pengukuran relatif (%)
Percobaan 1	0.81	1.78
Percobaan 2	0.19	0.73
Rata-rata	0.66	0.90

Tabel 2 menunjukkan hasil pengujian galat pengukuran kelembaban relatif udara oleh sensor DHT11. Dari pengujian yang dilakukan, didapatkan nilai rata-rata galat pengukuran absolut pembacaan kelembaban relatif adalah  $\pm 0.66\%$ . Nilai rata-rata galat pengukuran relatif pembacaan kelembaban relatif adalah  $0.90\%$ . Nilai galat absolut yang diperoleh dapat dikategorikan baik karena penyimpangan yang terjadi masih di bawah spesifikasi akurasi DHT11 untuk pembacaan kelembaban, yaitu  $\pm 5\%$ .

#### 4.4 Hasil Pengujian Delay Transmisi Data

Tabel 3 menunjukkan hasil pengujian *delay* transmisi data hasil pengukuran dari *server* menuju *client*. Dari hasil pengujian, diperoleh nilai rata-rata *delay* transmisi data pada sistem adalah 387.87 milisekon dengan nilai maksimum *delay* yang terjadi selama pengujian adalah 676 milisekon.

Tabel 3 Pengujian *delay* transmisi data

Pengujian	Rata-rata <i>delay</i> transmisi data (milisekon)	Maksimum <i>delay</i> transmisi data (milisekon)
Percobaan 1	350.97	676
Percobaan 2	412.77	644
Rata-rata	381.87	

#### 4.5 Hasil Pengujian Packet Loss

Tabel 4 menunjukkan hasil pengujian *packet loss* pada sistem yang telah dibuat. Selama pengujian, *server* berhasil mengirimkan data hasil pengukuran ke *client* sejumlah 120 data dari 120 data yang perlu dikirim. Persentase

*packet loss* yang diperoleh pada pengujian adalah  $0\%$  sehingga dapat dikategorikan baik.

Tabel 4 Pengujian *packet loss*

Pengujian	Jumlah data dikirim <i>server</i>	Jumlah data diterima <i>client</i>	<i>Packet Loss</i> (%)
Percobaan 1	60	60	0
Percobaan 2	60	60	0

## 5. KESIMPULAN

Berdasarkan rancang bangun dan hasil pengujian sistem akuisisi data kondisi udara ruangan diperoleh kesimpulan sebagai berikut:

1. Sistem yang dibuat dapat mengukur, menyimpan, dan menampilkan data hasil pengukuran kondisi udara ruangan secara *realtime* sesuai dengan desain awal sistem.
2. Rata-rata galat pengukuran absolut dan relatif pembacaan temperatur oleh sistem berturut-turut  $\pm 0.34\text{ }^{\circ}\text{C}$  dan  $1.25\%$ . Rata-rata galat pengukuran absolut dan relatif pembacaan kelembaban oleh sistem berturut-turut  $\pm 0.66\%$  dan  $0.90\%$ .
3. Rata-rata nilai *delay* transmisi data hasil pengukuran pada sistem adalah 387.87 milisekon.
4. Persentase *packet loss* pada sistem selama pengujian adalah  $0\%$ .

## DAFTAR PUSTAKA

- [1] L. Lukas, L. F. Suoth, dan R. Wowor, "Hubungan Antara Suhu Lingkungan Kerja dan Jam Kerja dengan Stres Kerja di PT . Adhi Karya ( PERSERO ) TBK Unit Manado Proyek Universitas Sam Ratulangi," *J. KESMAS*, vol. 7, 2018.
- [2] N. Nagara dan P. I. Yazid, "Perangkat Lunak Sistem Akuisisi Data Menggunakan Delphi," *J. Otomasi Kontrol dan Instrumentasi*, 2012.
- [3] M. Grinberg, "Flask-SocketIO Introduction," 2018. <https://flask-socketio.readthedocs.io/en/latest/intro.html> (diakses Juni 08, 2022).
- [4] F. Marisa, *Web Programming (Client Side and Server Side)*. Yogyakarta: Deepublish, 2017.

Prosiding The 13th Industrial Research Workshop and National Seminar  
Bandung, 13-14 Juli 2022

- [5] ASHRAE, *Thermal Environmental Conditions for Human Occupancy (ASHRAE Standard 55-2010)*. Atlanta: ASHRAE, 2010.
- [6] Republik Indonesia Nomor 1077/MENKES/PER/V/2011 Tentang *Pedoman Penyehatan Udara Dalam Ruang Rumah*. Jakarta: Kementerian Kesehatan Republik Indonesia, 2011.
- Kemntrian Kesehatan Republik Indonesia, *Peraturan Menteri Kesehatan*