

PERANCANGAN *SMART HOME* BERBASIS *INTERNET OF THINGS* MENGGUNAKAN *CISCO PACKET TRACER*

Salsa Fadhillah¹, Hanafi², Hanafi³,

^{1,2,3}Jurusan Teknik Elektro Politeknik Negeri Lhokseumawe
Email: salssaa12@gmail.com¹, hanafi_hf@pnl.ac.id², hnfbatubara@yahoo.com³

Corresponding Author : Hanafi

Jurusan Teknik Elektro Politeknik Negeri Lhokseumawe
Email: hanafi_hf@pnl.ac.id

Abstrak – Perkembangan teknologi *Internet of Things* (IoT) telah memungkinkan penerapan sistem otomatisasi rumah atau *smart home* yang semakin canggih. Penelitian ini bertujuan untuk merancang dan mensimulasikan sistem *smart home* menggunakan perangkat IoT melalui *Software Cisco Packet Tracer*. Sistem ini mencakup pengendalian berbagai perangkat seperti sensor suhu, kelembaban, kebakaran, gerak, asap, gas, dan RFID, serta aktuator seperti lampu, kipas, jendela, penyiram tanaman, dan pintu otomatis. Simulasi dilakukan untuk menguji konektivitas jaringan, fungsi otomatisasi, dan kinerja jaringan berdasarkan parameter *delay*, *throughput*, *jitter*, dan *packet loss*. Hasil simulasi menunjukkan bahwa jaringan beroperasi secara stabil dengan rata-rata *delay* 40 ms dan 52 ms, kecepatan transfer data 49,2 Kbps dan 64 Kbps, *jitter* 36,22 ms dan 57,67 ms, serta *packet loss* sebesar 0%. Semua perangkat IoT mampu merespons kondisi secara *real-time*, dan sistem monitoring jarak jauh dapat dijalankan. Keseluruhan hasil menunjukkan bahwa rancangan sistem *smart home* ini mampu mendukung komunikasi antar perangkat dan dapat diimplementasikan sebagai solusi otomatisasi rumah berbasis IoT.

Kata-kata kunci: *Smart Home, Internet of Things, Cisco Packet Tracer, Otomatisasi Rumah, Kinerja Jaringan.*

Abstract – The development of *Internet of Things* (IoT) technology has enabled the implementation of increasingly sophisticated home automation systems or smart homes. This study aims to design and simulate a smart home system using IoT devices through *Cisco Packet Tracer Software*. This system includes controlling various devices such as temperature, humidity, fire, motion, smoke, gas, and RFID sensors, as well as actuators such as lights, fans, windows, plant sprinklers, and automatic doors. Simulations were conducted to test network connectivity, automation functions, and network performance based on *delay*, *throughput*, *jitter*, and *packet loss* parameters. The simulation results show that the network operates stably with an average *delay* of 40 ms and 52 ms, data transfer rates of 49.2 Kbps and 64 Kbps, *jitter* of 36.22 ms and 57.67 ms, and *packet loss* of 0%. All IoT devices are able to respond to conditions in *real-time*, and the remote monitoring system can be run. Overall results show that this smart home system design is able to support communication between devices and can be implemented as an IoT-based home automation solution.

Keywords: *Smart Home, Internet of Things, Cisco Packet Tracer, Home Automation, Network Performance.*

I. PENDAHULUAN

Perkembangan teknologi informasi mendorong munculnya konsep *smart home* yang memungkinkan berbagai perangkat rumah saling terhubung melalui *Internet of Things* (IoT) [1]. Sistem ini mampu mendeteksi kondisi lingkungan seperti suhu, kelembaban, asap, gas, dan gerakan, kemudian mengaktifkan aktuator seperti lampu, kipas, pintu, maupun penyiram tanaman secara otomatis guna mendukung otomatisasi dan pengendalian perangkat rumah yang dapat dipantau serta dikontrol dari jarak jauh melalui jaringan internet [2].

Beberapa penelitian sebelumnya telah mengkaji penerapan *smart home* berbasis IoT menggunakan simulasi. Chaudhari, dkk. merancang model rumah pintar

berbasis *Cisco Packet Tracer* [3], Sinaga, dkk. menekankan efisiensi energi melalui integrasi IoT [4], sedangkan Prayitno dan Yakti melakukan simulasi otomatisasi pintu dan kipas berbasis sensor [5]. Hasil-hasil penelitian tersebut menunjukkan potensi besar IoT dalam mendukung otomatisasi rumah modern. Pada penelitian ini dilakukan perancangan dan simulasi sistem *smart home* menggunakan *Cisco Packet Tracer*.

II. TINJAUAN PUSTAKA

A. *Internet of Things* (IoT)

Internet of Things (IoT) merupakan konsep yang memungkinkan perangkat fisik saling terhubung melalui

jaringan internet untuk bertukar data tanpa interaksi langsung manusia [6]. Teknologi ini mendukung berbagai aplikasi, termasuk pada *smart home*, yang mengintegrasikan sensor dan aktuator untuk memantau serta mengendalikan kondisi rumah secara *real-time* [7].

B. *Smart Home*

Smart home didefinisikan sebagai sistem rumah pintar yang dilengkapi perangkat IoT untuk mengatur pencahayaan, suhu, pintu, jendela, hingga sistem peringatan seperti alarm atau sirene. Sistem ini mampu merespon perubahan kondisi lingkungan melalui sensor, seperti sensor suhu, kelembaban, asap, gas, kebakaran, gerak, dan RFID [8]. Dengan adanya integrasi tersebut, pengguna dapat memantau serta mengendalikan perangkat rumah melalui jaringan internet [9].

C. Parameter Kinerja Jaringan

Parameter kinerja jaringan merupakan ukuran yang digunakan untuk mengevaluasi kinerja jaringan komputer. Parameter ini dapat meliputi *throughput*, *delay*, *packet loss*, dan *jitter* [10]. Standar kualitas layanan jaringan ditetapkan oleh TIPHON (*Telecommunications and Internet Protocol Harmonization Over Networks*).

- *Throughput*

Throughput merupakan besaran total data yang diterima server dalam waktu tertentu dengan fungsi untuk mengukur efektifitas data transfer yang dituliskan dengan satuan bps (*bit per second*) [10]. Persamaan untuk menghitung nilai *throughput* dapat dilihat pada rumus sebagai berikut:

$$Throughput = \frac{\text{Total Paket Data Diterima}}{\text{Waktu Pengamatan}} \quad (1)$$

- *Packet Loss*

Packet loss adalah data yang hilang pada saat pengiriman paket data [11]. Kualitas *packet loss* suatu jaringan menurut TIPHON bisa dilihat pada Tabel 1 [13].

$$Packet\ loss = \frac{(\text{Paket Dikirim} - \text{Paket Diterima})}{(\text{Paket Dikirim})} \times 100\% \quad (2)$$

TABEL I
Standar Kualitas *Packet Loss* Menurut TIPHON [13]

Kategori <i>Packet Loss</i>	<i>Packet Loss</i> (%)	Indeks
Sangat Bagus	0 – 3	4
Bagus	3 – 15	3
Cukup Bagus	15 – 25	2
Buruk	> 25	1

- *Jitter*

Jitter atau variasi *delay*, berhubungan erat dengan *latency*, yang menunjukkan banyaknya variasi *delay*

pada transmisi data di jaringan. Hal ini disebabkan oleh variasi-variasi panjang antrian data [12]. Kualitas *jitter* suatu jaringan menurut TIPHON bisa dilihat pada Tabel 2 [13].

$$Jitter = \frac{\text{Total variasi delay}}{\text{Total paket yang diterima}} \quad (3)$$

TABEL II
Standar Kualitas *Jitter* Menurut TIPHON [13]

Kategori <i>Jitter</i>	<i>Jitter</i> (ms)	Indeks
Sangat Bagus	0	4
Bagus	0 – 75	3
Cukup Bagus	75 – 125	2
Buruk	125 – 225	1

- *Delay (Latency)*

Delay merupakan waktu yang diperlukan oleh suatu paket data untuk melakukan perpindahan dari sumber ke tujuan melalui jaringan [10]. Kualitas *delay* suatu jaringan menurut TIPHON bisa dilihat pada Tabel 3 [13].

$$Latency\ (Rata-Rata) = \frac{\text{Jumlah waktu pengiriman data}}{\text{Jumlah paket}} \quad (4)$$

TABEL III
Standar Kualitas *Latency* Menurut TIPHON [13]

Kategori <i>Latency</i>	<i>Latency</i> (ms)	Indeks
Sangat Bagus	< 150	4
Bagus	150 – 300	3
Cukup Bagus	300 – 450	2
Buruk	> 450	1

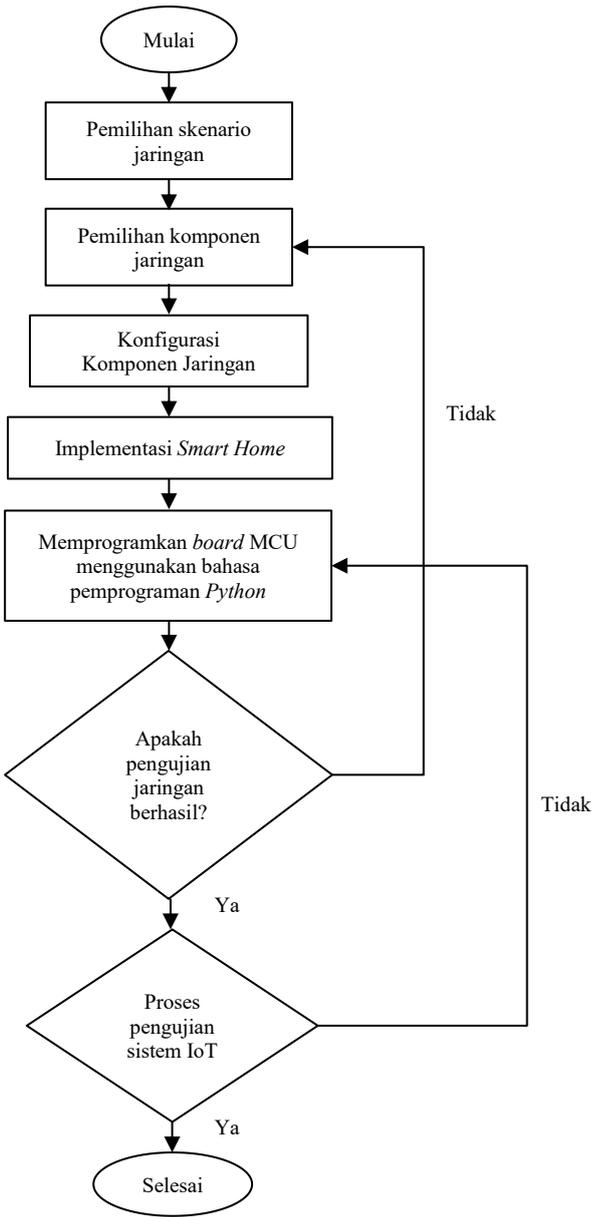
D. *Cisco Packet Tracer*

Untuk keperluan penelitian, *Cisco Packet Tracer* sering digunakan sebagai perangkat lunak simulasi jaringan yang mendukung perancangan dan pengujian sistem IoT. Aplikasi ini memungkinkan simulasi perangkat jaringan, pemrograman mikrokontroler, serta monitoring interaksi antar perangkat IoT dalam lingkungan virtual [14].

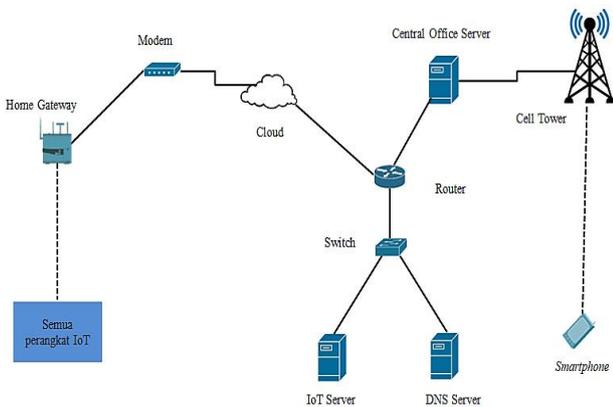
III. METODOLOGI

A. Konsep Desain

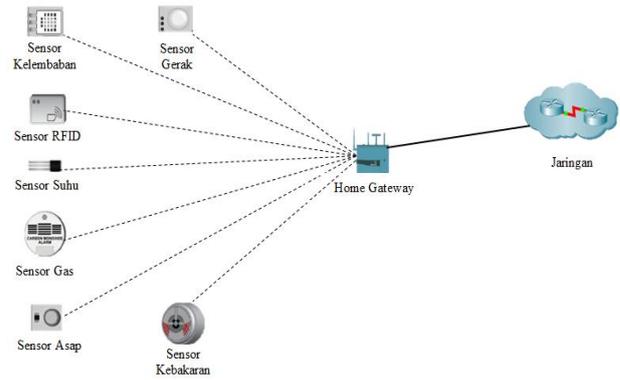
Metode yang digunakan dalam penelitian ini mencakup perancangan dan simulasi menggunakan *software Cisco Packet Tracer versi 8.2.2*. Proses perancangan dimulai dengan membangun struktur jaringan seperti *router*, *switch*, *server* dan lainnya sebagai dasar komunikasi antar perangkat, kemudian dilakukan pemilihan sensor yang akan digunakan untuk *smart home*, seperti sensor gerak, sensor suhu, sensor kelembaban, sensor api, dan alarm.



Gbr. 1 Flowchart Penelitian



Gbr. 2 Diagram Jaringan



Gbr. 3 Diagram Sensor Smart Home

B. Fungsional dan Struktur Alat

Pada sistem *smart home*, setiap perangkat memiliki fungsi dan struktur yang saling mendukung untuk menciptakan otomatisasi dan pengendalian rumah secara cerdas. Berikut penjelasan fungsional dan struktur alat yang digunakan dalam perancangan ini:

- *Cell Tower*, sebagai penghubung awal antara perangkat *mobile* dengan jaringan utama untuk komunikasi nirkabel *smart home*.
- *Central Office Server*, pusat distribusi jaringan dari penyedia layanan ke perangkat rumah.
- *Router*, mengatur lalu lintas data antarjaringan serta berfungsi sebagai DHCP server.
- *Switch* (2960 IOS15), menghubungkan perangkat dalam LAN dan meneruskan data berdasarkan alamat MAC.
- *IoT Server & DNS Server*, IoT Server sebagai pusat monitoring dan kendali perangkat IoT, sedangkan DNS Server menerjemahkan domain ke alamat IP.
- *Cloud*, merepresentasikan layanan komputasi awan untuk aplikasi, penyimpanan, dan layanan jarak jauh.
- *Cable Modem*, penghubung jaringan rumah dengan penyedia layanan internet melalui kabel.
- *Home Gateway* (DLC100), penghubung perangkat rumah dengan internet serta penyedia SSID untuk koneksi nirkabel.
- *Mikrokontroler* (MCU-PT), pusat kendali sensor dan aktuator.
- *Sensor Suhu*, mendeteksi suhu ruangan untuk mengatur kipas secara otomatis atau manual via *smartphone/laptop*.
- *Sensor Kelembaban*, memantau kelembaban tanah dan mengaktifkan penyiram tanaman bila di bawah ambang batas.
- *Sensor Gas*, mendeteksi kebocoran gas dan mengaktifkan alarm, jendela, serta sprinkler.
- *Sensor Asap*, dipasang di garasi untuk mendeteksi asap, membuka pintu otomatis, dan mengaktifkan CCTV.
- *Sensor Kebakaran*, mendeteksi api dan memicu alarm, penyiram air, serta membuka jendela otomatis.

- Sensor Gerak (*Motion Detector*), mendeteksi pergerakan di area depan rumah dan mengaktifkan alarm serta CCTV.
- Sensor RFID, sebagai pengaman pintu, hanya kartu terverifikasi yang dapat membuka pintu, serta otomatis menyalakan lampu.

C. Metode Simulasi

Simulasi *smart home* dilakukan menggunakan *Cisco Packet Tracer* melalui beberapa tahap. Pertama, membuka aplikasi dan membuat proyek baru sebagai dasar perancangan. Kedua, menentukan komponen jaringan pada *Network Devices* dan *End Devices*, serta menambahkan perangkat *smart home* dari menu *Components* dan kategori IoT. Ketiga, melakukan konfigurasi jaringan melalui *Command Line Interface (CLI)* dan pemrograman komponen dengan membuat proyek baru pada menu *Programming*. Tahap terakhir adalah pengujian sistem, meliputi *ping* untuk konektivitas, uji respon sensor terhadap *detector*, serta pemantauan keseluruhan simulasi untuk memastikan sistem berjalan tanpa *error*.

D. Metode Analisis

Setelah berhasil melakukan simulasi *smart home*, langkah selanjutnya adalah menganalisis hasil untuk memastikan bahwa semua sensor berfungsi dengan baik dan berhasil dioperasikan, serta menganalisis kinerja jaringan dengan menggunakan perintah *ping* untuk mengukur *delay*, *throughput*, *jitter*, dan *packet loss*. Analisis ini mengacu pada standar TIPHON yang memberikan kategori kualitas jaringan berdasarkan nilai parameter-parameter tersebut. Data hasil simulasi kemudian dianalisis untuk mengetahui jumlah data yang berhasil ditransfer dalam satu waktu tertentu, waktu yang dibutuhkan untuk mengirimkan paket data antar perangkat, variasi dalam waktu pengiriman paket, dan persentase paket yang hilang selama transmisi.

IV. HASIL DAN PEMBAHASAN

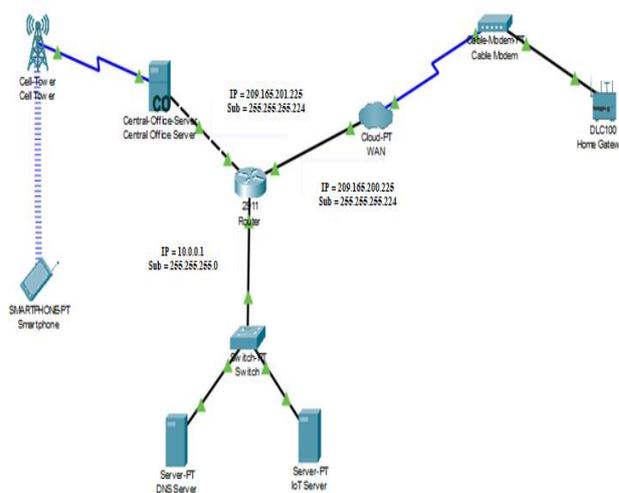
A. Topologi Jaringan *Smart Home*

Topologi jaringan *smart home* seperti ditunjukkan pada Gambar 4. Gambar tersebut merujuk pada struktur jaringan yang dapat dilihat pada Gambar 5. Topologi jaringan ini menunjukkan bagaimana data dari perangkat IoT diteruskan melalui *home gateway* (DLC100).

Topologi ini terdiri dari *smartphone* yang terhubung nirkabel ke menara seluler, lalu ke Central Office (CO) Server via *coaxial*. Dari CO Server, koneksi diteruskan ke router dengan kabel *cross-over*, kemudian ke switch dengan kabel *straight-through*. Switch menghubungkan DNS Server dan IoT Server, sedangkan router juga tersambung ke cloud, modem (*coaxial*), dan akhirnya ke *Home Gateway* (DLC100) melalui kabel *straight-through*.



Gbr. 4 Tampilan *Smart Home* 3D

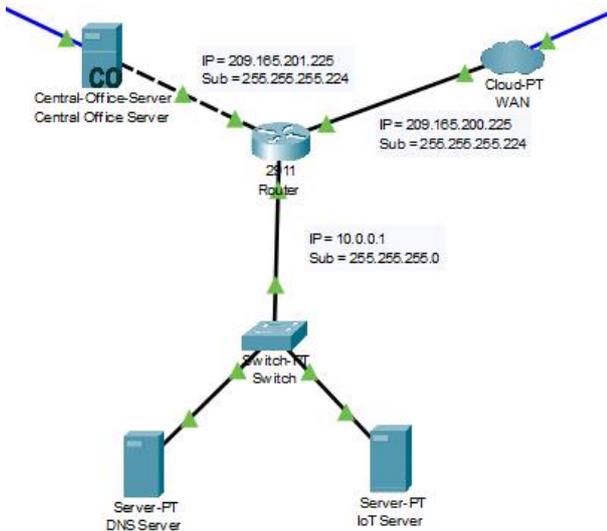


Gbr. 5 Topologi Jaringan

B. Konfigurasi Dasar *Router*

Konfigurasi dasar *router* merupakan langkah awal yang dilakukan untuk memastikan setiap perangkat jaringan dapat saling terhubung dengan baik. Pada topologi jaringan yang telah dirancang, digunakan sebuah perangkat *router Cisco tipe 2911* yang berfungsi sebagai penghubung antara jaringan lokal dan jaringan eksternal (WAN). *Router* ini mengatur lalu lintas data antara berbagai segmen jaringan, termasuk koneksi ke switch, cloud, dan CO Server. Skema jaringan dapat dilihat pada Gambar 6.

Setelah rangkaian dirancang, *router* dikonfigurasi untuk pengaturan alamat IP pada setiap antarmuka, serta pengaktifan *interface* dapat dilihat pada Gambar 7, Gambar 8, dan Gambar 9.



Gbr. 6 Skema Rangkaian Jaringan

```
Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int g0/0
Router(config-if)#ip address 10.0.0.1 255.255.255.0
Router(config-if)#no shutdown
```

Gbr. 7 Tampilan Program Port G0/0

```
Router(config-if)#int g0/1
Router(config-if)#ip address 209.165.200.225 255.255.255.224
Router(config-if)#no shutdown
```

Gbr. 8 Tampilan Program Port G0/1

```
Router(config)#int g0/2
Router(config-if)#ip address 209.165.201.225 255.255.255.224
Router(config-if)#no shutdown
Router(config-if)#ex
```

Gbr. 9 Tampilan Program Port G0/2

• Konfigurasi Router Sebagai Server DHCP

Router dikonfigurasi agar berperan sebagai server DHCP untuk membagikan alamat IP secara otomatis kepada perangkat. Router mengatur pembagian IP kepada perangkat CO Server dan Home Gateway. Pada Gambar 10 dan Gambar 11, konfigurasi DHCP mencakup penentuan rentang IP (IP Address Pool), subnet mask, default gateway, serta DNS server. Dengan menggunakan DHCP, pengalokasian IP menjadi lebih dinamis dan efisien, serta mengurangi kemungkinan konflik IP.

```
Router(config)#ip dhcp excluded-address 209.165.201.225 209.165.201.229
Router(config)#ip dhcp pool CELL
Router(dhcp-config)#network 209.165.201.224 255.255.255.224
Router(dhcp-config)#default-router 209.165.201.225
Router(dhcp-config)#dns-server 10.0.0.254
Router(dhcp-config)#exit
```

Gbr. 10 Tampilan Program DHCP pada Router untuk CO Server.

```
Router(config)#ip dhcp excluded-address 209.165.200.225 209.165.200.229
Router(config)#ip dhcp pool WAN
Router(dhcp-config)#network 209.165.200.224 255.255.255.224
Router(dhcp-config)#default-router 209.165.200.225
Router(dhcp-config)#dns-server 10.0.0.254
Router(dhcp-config)#exit
```

Gbr. 11 Tampilan Program DHCP pada Router untuk Home Gateway.

C. Konfigurasi Server IoT dan DNS

Pada jaringan yang telah dirancang, terdapat dua jenis server utama yang memiliki peran penting dalam mendukung sistem smart home, yaitu IoT server dan DNS server. Server IoT berfungsi sebagai pusat pengendali perangkat-perangkat Internet of Things (IoT) yang ada dalam jaringan. Sementara itu, DNS (Domain Name System) server digunakan untuk menerjemahkan nama domain yang digunakan oleh perangkat pengguna. Tabel 4 adalah konfigurasi IP untuk Server IoT dan DNS Server.

TABEL IV
IP Address Perangkat DNS Server dan IoT Server

Perangkat	Alamat Ip	Subnet	Gateway	DNS Server
DNS Server	10.0.0.254	255.0.0.0	10.0.0.1	10.0.0.254
IoT Server	10.0.0.253	255.255.255.0	10.0.0.1	10.0.0.254

D. Implementasi Smart Home

Setelah pembuatan rangkaian topologi jaringan yang ditampilkan pada Gambar 5, langkah selanjutnya adalah melakukan perancangan perangkat smart home. Perancangan ini mencakup penempatan dan konfigurasi perangkat-perangkat IoT yang digunakan dalam sistem otomatisasi rumah.

E. Pengujian Konektivitas Router

Pengujian konektivitas dilakukan dengan perintah ping melalui CLI dari router ke DNS Server dan IoT Server menggunakan lima variasi ukuran paket (36, 64, 128, 512, dan 1024 byte) untuk memastikan konfigurasi berhasil dibuat dan perangkat dapat terhubung.

• Pengujian Konektivitas dari Router ke DNS Server

Pengujian ini dilakukan dengan mengirimkan perintah ping dari Router G0/0 (10.0.0.1) ke DNS Server (10.0.0.254). Tujuannya untuk memastikan bahwa router dapat terhubung dengan DNS Server dan menerima respon balik dengan ukuran paket yang berbeda.

```
Router#ping
Protocol [ip]:
Target IP address: 10.0.0.254
Repeat count [5]: 10
Datagram size [100]: 36
Timeout in seconds [2]:
Extended commands [n]: n
Sweep range of sizes [n]: n
Type escape sequence to abort.
Sending 10, 36-byte ICMP Echos to 10.0.0.254, timeout is 2 seconds:
!!!!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 0/1/12 ms
```

Gbr. 12 Hasil Ping 36 Bytes

```
Router#ping
Protocol [ip]:
Target IP address: 10.0.0.254
Repeat count [5]: 10
Datagram size [100]: 64
Timeout in seconds [2]:
Extended commands [n]: n
Sweep range of sizes [n]: n
Type escape sequence to abort.
Sending 10, 64-byte ICMP Echos to 10.0.0.254, timeout is 2 seconds:
!!!!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 0/3/22 ms
```

Gbr. 13 Hasil ping 64 Bytes

```
Router#ping
Protocol [ip]:
Target IP address: 10.0.0.254
Repeat count [5]: 10
Datagram size [100]: 128
Timeout in seconds [2]:
Extended commands [n]: n
Sweep range of sizes [n]: n
Type escape sequence to abort.
Sending 10, 128-byte ICMP Echos to 10.0.0.254, timeout is 2 seconds:
!!!!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 0/7/34 ms
```

Gbr. 14 Hasil ping 128 Bytes

```
Router#ping
Protocol [ip]:
Target IP address: 10.0.0.254
Repeat count [5]: 10
Datagram size [100]: 512
Timeout in seconds [2]:
Extended commands [n]: n
Sweep range of sizes [n]: n
Type escape sequence to abort.
Sending 10, 512-byte ICMP Echos to 10.0.0.254, timeout is 2 seconds:
!!!!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 0/9/32 ms
```

Gbr. 15 Hasil ping 512 Bytes

```
Router#ping
Protocol [ip]:
Target IP address: 10.0.0.254
Repeat count [5]: 10
Datagram size [100]: 1024
Timeout in seconds [2]:
Extended commands [n]: n
Sweep range of sizes [n]: n
Type escape sequence to abort.
Sending 10, 1024-byte ICMP Echos to 10.0.0.254, timeout is 2 seconds:
!!!!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 0/10/35 ms
```

Gbr. 16 Hasil ping 1024 Bytes

Gambar di atas menampilkan hasil uji koneksi jaringan dengan menggunakan perintah *ping*. Dalam proses pengujian ini, *router* mengirimkan masing-masing 10 paket ICMP dengan variasi ukuran 36 byte, 128 byte, 512 byte, dan 1024 byte menuju alamat IP 10.0.0.254.

- Pengujian Konektivitas dari *Router* Ke IoT Server

Selanjutnya dilakukan pengujian konektivitas dari *Router* G0/0 (10.0.0.1) ke IoT Server (10.0.0.253) dengan metode dan variasi ukuran byte yang sama seperti sebelumnya.

```
Router#ping
Protocol [ip]:
Target IP address: 10.0.0.253
Repeat count [5]: 10
Datagram size [100]: 36
Timeout in seconds [2]:
Extended commands [n]: n
Sweep range of sizes [n]: n
Type escape sequence to abort.
Sending 10, 36-byte ICMP Echos to 10.0.0.253, timeout is 2 seconds:
!!!!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 0/1/11 ms
```

Gbr. 17 Hasil ping 36 Bytes

```
Router#ping
Protocol [ip]:
Target IP address: 10.0.0.253
Repeat count [5]: 10
Datagram size [100]: 64
Timeout in seconds [2]:
Extended commands [n]: n
Sweep range of sizes [n]: n
Type escape sequence to abort.
Sending 10, 64-byte ICMP Echos to 10.0.0.253, timeout is 2 seconds:
!!!!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 0/4/44 ms
```

Gbr. 18 Hasil ping 64 Bytes

```
Router#ping
Protocol [ip]:
Target IP address: 10.0.0.253
Repeat count [5]: 10
Datagram size [100]: 128
Timeout in seconds [2]:
Extended commands [n]: n
Sweep range of sizes [n]: n
Type escape sequence to abort.
Sending 10, 128-byte ICMP Echos to 10.0.0.253, timeout is 2 seconds:
!!!!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 0/4/38 ms
```

Gbr. 19 Hasil ping 128 Bytes

```
Router#ping
Protocol [ip]:
Target IP address: 10.0.0.253
Repeat count [5]: 10
Datagram size [100]: 512
Timeout in seconds [2]:
Extended commands [n]: n
Sweep range of sizes [n]: n
Type escape sequence to abort.
Sending 10, 512-byte ICMP Echos to 10.0.0.253, timeout is 2 seconds:
!!!!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 0/5/45 ms
```

Gbr. 20 Hasil ping 512 Bytes

```
Router#ping
Protocol [ip]:
Target IP address: 10.0.0.253
Repeat count [5]: 10
Datagram size [100]: 1024
Timeout in seconds [2]:
Extended commands [n]: n
Sweep range of sizes [n]: n
Type escape sequence to abort.
Sending 10, 1024-byte ICMP Echos to 10.0.0.253, timeout is 2 seconds:
!!!!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 0/13/56 ms
```

Gbr. 21 Hasil ping 1024 Bytes

Gambar diatas menunjukkan hasil pengujian konektivitas jaringan menggunakan perintah *ping*. Dalam pengujian ini, *router* mengirimkan 10 paket data ICMP berukuran 36 byte, 64 byte, 128 byte, 512 byte, dan 1024 byte ke alamat IP tujuan. Pengujian ini memastikan setiap paket terkirim tanpa kehilangan data serta mengukur stabilitas dan kecepatan respons jaringan.

- Pengukuran Throughput
 Pengukuran dilakukan dengan mengirimkan 10 paket data dari *router* ke DNS Server dan IoT Server menggunakan variasi ukuran paket. Adapun persamaan untuk menghitung parameter *Throughput* dapat dilihat pada Persamaan 1.

TABEL V
 Hasil Pengukuran *Throughput*

Skenario Pengujian	Jumlah Ping	Ukuran Paket (byte)	Total Data (bit)	Total Waktu (detik)	Throughput (Kbps)
<i>Router</i> ke DNS Server	10	36	2880	0,001	2.880
<i>Router</i> ke IoT Server	10	36	2880	0,001	2.880
<i>Router</i> ke DNS Server	10	64	5120	0,003	1.706
<i>Router</i> ke IoT Server	10	64	5120	0,004	1.280

Skenario Pengujian	Jumlah Ping	Ukuran Paket (byte)	Total Data (bit)	Total Waktu (detik)	Throughput (Kbps)
Router ke DNS Server	10	128	1024	0,007	1.463
Router ke IoT Server	10	128	1024	0,004	2.560
Router ke DNS Server	10	512	40960	0,009	4.551
Router ke IoT Server	10	512	40960	0,005	8.192
Router ke DNS Server	10	1024	81920	0,01	8.192
Router ke IoT Server	10	1024	81920	0,013	6.301

Berdasarkan Tabel 5, *throughput* dipengaruhi oleh ukuran paket dan waktu pengiriman. Pada paket 36 byte, *throughput* tinggi yaitu 2.880 Kbps, namun menurun pada 64–128 byte dengan nilai terendah 1.463 Kbps ke DNS Server. Sebaliknya, pada paket lebih besar *throughput* meningkat, dengan nilai tertinggi 8.192 Kbps pada paket 1024 byte ke DNS Server dan stabil pada IoT Server, terutama 512 byte yang juga mencapai 8.192 Kbps. Secara keseluruhan, jaringan ke IoT Server lebih stabil untuk data besar.

- Pengukuran *Packet Loss*

Tabel 6 menampilkan hasil pengukuran *packet loss* dari router ke DNS Server dan IoT Server dengan variasi ukuran paket, menggunakan Persamaan 2.

TABEL VI
Hasil Pengukuran *Packet Loss*

Skenario Pengujian	Ukuran Paket (byte)	Jumlah Paket Dikirim	Paket Hilang	Persentase Paket hilang
Router ke DNS Server	36	10	0	0%
Router ke IoT Server	36	10	0	0%
Router ke DNS Server	64	10	0	0%
Router ke IoT Server	64	10	0	0%
Router ke DNS Server	128	10	0	0%
Router ke IoT Server	128	10	0	0%
Router ke DNS Server	512	10	0	0%
Router ke IoT Server	512	10	0	0%
Router ke DNS Server	1024	10	0	0%
Router ke IoT Server	1024	10	0	0%

Tabel 6 menunjukkan hasil pengujian *packet loss* pada jaringan. Semua skenario menunjukkan hasil yang sangat baik. Dari pengujian yang dilakukan antara router ke DNS Server dan IoT Server dengan berbagai ukuran paket data, tidak ditemukan adanya paket yang hilang.

Setiap skenario menunjukkan bahwa dari 10 paket yang dikirim, seluruhnya diterima dengan baik, menghasilkan *presentase packet loss* sebesar 0%.

- Pengukuran Latency

Tabel 7 menampilkan hasil pengukuran *latency* yang diperbolehkan dari pengiriman paket antara router ke DNS Server dan IoT Server. Latency dihitung menggunakan Persamaan 4.

TABEL VII
Hasil Pengukuran *Latency*

Skenario Pengujian	Ukuran Paket (byte)	Jumlah sampel	Rata-rata Latency (ms)	Latency Minimum (ms)	Latency Maksimum (ms)
Router ke DNS Server	36	10	1	0	12
Router ke IoT Server	36	10	1	0	11
Router ke DNS Server	64	10	3	0	22
Router ke IoT Server	64	10	4	0	44
Router ke DNS Server	128	10	7	0	34
Router ke IoT Server	128	10	4	0	38
Router ke DNS Server	512	10	9	0	32
Router ke IoT Server	512	10	5	0	45
Router ke DNS Server	1024	10	10	0	35
Router ke IoT Server	1024	10	13	0	56

Dari Tabel 7 menunjukkan hasil pengujian bahwa rata-rata *latency* cenderung meningkat seiring dengan bertambahnya ukuran paket. Pada pengujian dari router ke DNS Server dengan ukuran paket 36 byte, rata – rata *latency* hanya 1 ms, sedangkan pada ukuran 1024 byte meningkat menjadi 10 ms. Hal yang sama juga terlihat pada koneksi ke IoT Server, di mana rata-rata *latency* meningkat dari 1 ms pada 36 byte menjadi 13 ms pada 1024 byte. Meskipun ada variasi dalam rata-rata *latency*, nilai *latency* minimum selalu 0 ms untuk semua ukuran paket, yang menunjukkan bahwa beberapa paket dapat dikirim dan diterima dengan sangat cepat tanpa *delay*. Sedangkan nilai *latency* maksimum cenderung bervariasi dan meningkat seiring bertambahnya ukuran paket, contohnya dari 12 ms menjadi 56 ms pada koneksi ke IoT Server.

F. Pengujian Konektivitas Perangkat Jaringan

Pengujian konektivitas dilakukan untuk memastikan semua perangkat dalam jaringan sudah terhubung dan konfigurasi yang dilakukan telah berhasil. Pengujian dilakukan dengan metode *ping* antar perangkat.

- Pengujian Konektivitas dari Laptop ke DNS Server

Gambar 17 menunjukkan hasil pengujian konektivitas jaringan menggunakan perintah *ping*, dengan waktu respon pengiriman 10 paket data adalah 15 ms hingga 120 ms. Ini masuk ke dalam kategori sangat bagus berdasarkan Standar TIPHON.

```
C:\>ping 10.0.0.254 -n 10

Pinging 10.0.0.254 with 32 bytes of data:

Reply from 10.0.0.254: bytes=32 time=66ms TTL=126
Reply from 10.0.0.254: bytes=32 time=21ms TTL=126
Reply from 10.0.0.254: bytes=32 time=16ms TTL=126
Reply from 10.0.0.254: bytes=32 time=27ms TTL=126
Reply from 10.0.0.254: bytes=32 time=120ms TTL=126
Reply from 10.0.0.254: bytes=32 time=39ms TTL=126
Reply from 10.0.0.254: bytes=32 time=15ms TTL=126
Reply from 10.0.0.254: bytes=32 time=21ms TTL=126
Reply from 10.0.0.254: bytes=32 time=55ms TTL=126
Reply from 10.0.0.254: bytes=32 time=28ms TTL=126

Ping statistics for 10.0.0.254:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 15ms, Maximum = 120ms, Average = 40ms
```

Gbr. 17 Pengujian Laptop Ke DNS Server

- Pengujian Konektivitas dari Laptop ke IoT Server

Gambar 18 menunjukkan hasil pengujian konektivitas pengiriman 10 paket data, dengan waktu respon 52 ms hingga 150 ms. Ini masuk ke dalam kategori sangat bagus berdasarkan Standar TIPHON.

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.0.0.253 -n 10

Pinging 10.0.0.253 with 32 bytes of data:

Reply from 10.0.0.253: bytes=32 time=225ms TTL=126
Reply from 10.0.0.253: bytes=32 time=19ms TTL=126
Reply from 10.0.0.253: bytes=32 time=125ms TTL=126
Reply from 10.0.0.253: bytes=32 time=17ms TTL=126
Reply from 10.0.0.253: bytes=32 time=23ms TTL=126
Reply from 10.0.0.253: bytes=32 time=42ms TTL=126
Reply from 10.0.0.253: bytes=32 time=8ms TTL=126
Reply from 10.0.0.253: bytes=32 time=9ms TTL=126
Reply from 10.0.0.253: bytes=32 time=9ms TTL=126
Reply from 10.0.0.253: bytes=32 time=48ms TTL=126

Ping statistics for 10.0.0.253:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 8ms, Maximum = 225ms, Average = 52ms
```

Gbr. 18 Pengujian Laptop Ke IoT Server

- Pengukuran *Throughput*

Throughput dihitung menggunakan Persamaan 1. Hasil *throughput* dapat dilihat pada Tabel 8.

TABEL VIII
Hasil Pengukuran *Throughput*

Skenario Pengujian	Jumlah Ping	Ukuran Paket (byte)	Total Data (bit)	Total Waktu (detik)	<i>Throughput</i> (Kbps)
Laptop ke DNS Server	10	32	2560	0,04	64
Laptop ke IoT Server	10	32	2560	0,052	49,2

Tabel 8 menunjukkan hasil pengujian koneksi antara laptop ke DNS Server dan IoT Server memiliki nilai kecepatan data 64 Kbps dan 49,2 Kbps, yang mengindikasikan bahwa performansi jaringan sangat baik untuk aktivitas seperti pengiriman data berkelanjutan atau aplikasi *real-time*.

- Pengukuran *Packet Loss*

Dalam pengukuran ini, dilakukan pengiriman data dari Laptop ke dua tujuan, yaitu DNS Server dan IoT Server, masing-masing sebanyak 10 paket data. *Packet loss* dapat dihitung menggunakan Persamaan 2, dengan hasil seperti pada Tabel 9. *Packet Loss* diperoleh 0%, ini termasuk dalam kategori Sangat Bagus berdasarkan Standar TIPHON.

TABEL IX
Hasil Pengukuran *Packet Loss*

Skenario Pengujian	Jumlah Paket Dikirim	Paket Hilang	Persentase Paket hilang	Standar Tiphon
Laptop ke DNS Server	10	0	0%	Sangat Bagus
Laptop ke IoT Server	10	0	0%	Sangat Bagus

- Pengukuran *Jitter*

Perhitungan *jitter* dilakukan dengan menggunakan Persamaan 3. Hasil *jitter* dapat dilihat pada Tabel 10.

TABEL X
Hasil Pengukuran *Jitter*

No	Skenario Pengujian	Tujuan IP	Jumlah Paket	<i>Jitter</i> (ms)	Standar Tiphon
1	Laptop ke DNS Server	10.0.0.254	10	36,22	Bagus
2	Laptop ke IoT Server	10.0.0.253	10	57,67	Bagus

Berdasarkan Tabel 10, diperoleh nilai *jitter* sebesar 36,22 ms dari total 10 paket data yang dikirim dari Laptop ke DNS Server. Sementara itu, pada pengujian dari Laptop ke IoT Server diperoleh *jitter* sebesar 57,67 ms. Kedua nilai ini masih dalam kategori Bagus berdasarkan Standar TIPHON.

- Pengukuran *Latency*

Latency dihitung menggunakan Persamaan 4, dengan hasil seperti pada Tabel 11. Berdasarkan tabel tersebut diketahui bahwa dari Laptop ke DNS Server, diperoleh rata-rata *latency* sebesar 40ms, dengan *latency* minimum 15 ms dan maksimum 120 ms dari total 10 kali pengiriman paket data. Sedangkan dari Laptop ke IoT Server, didapatkan rata-rata *latency* 52 ms, dengan nilai minimum 8 ms dan maksimum 225 ms. Ini masih termasuk ke dalam kategori Sangat Bagus berdasarkan Standar TIPHON.

TABEL XI
Hasil Pengukuran *Latency*

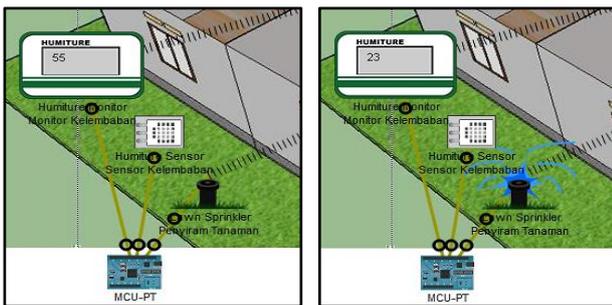
Skenario Pengujian	Jumlah sampel	Rata-rata <i>Latency</i> (ms)	<i>Latency</i> Minimum (ms)	<i>Latency</i> Maksimum (ms)	Standar Tiphon
Laptop ke DNS Server	10	40	15	120	Sangat Bagus
Laptop ke IoT Server	10	52	8	225	Sangat Bagus

G. Pengujian Sistem *Smart Home*

Pengujian ini dilakukan dengan tujuan untuk memastikan bahwa seluruh perangkat yang telah dikonfigurasi dalam sistem *smart home* dapat berfungsi dengan baik sesuai skenario yang dirancang. Gambar keseluruhan dari *smart home* seperti diperlihatkan pada Gambar 4.

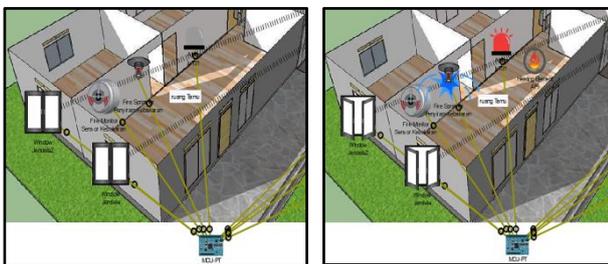
- Perangkat IoT

Pada perancangan ini, perangkat IoT yang digunakan seperti sensor kelembaban, sensor kebakaran, sensor RFID, sensor gerak, sensor gas, sensor asap, dan sensor suhu.



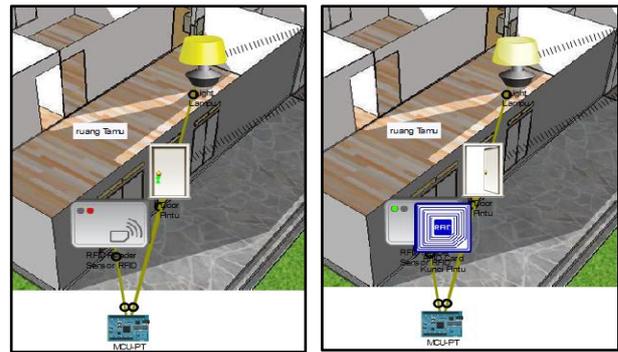
Gbr. 19 Sensor Kelembaban Sebelum dan Sesudah Dijalankan

Gambar 19 menunjukkan kontrol penyiraman tanaman otomatis berdasarkan kelembaban tanah. Saat sensor membaca 55 (tanah lembab), penyiram tidak aktif, sedangkan pada nilai 23 (tanah kering), sistem otomatis menyalakan penyiram untuk menjaga kelembaban.



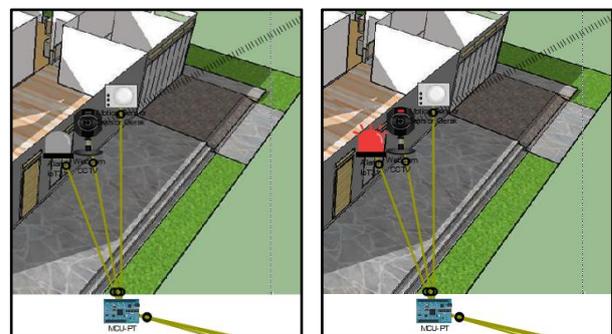
Gbr. 20 Sensor Kebakaran Sebelum dan Sesudah Terdeteksi

Gambar 20 menampilkan hasil simulasi sistem kebakaran. Terdapat sensor kebakaran yang terhubung ke MCU (*microcontroller*) bersama dengan penyiram kebakaran, alarm, dan jendela otomatis. Semua perangkat ini siap mendeteksi dan merespon kejadian darurat kebakaran.



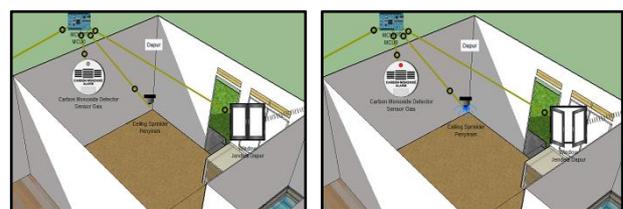
Gbr. 21 Sensor RFID Sebelum dan Sesudah Dijalankan

Gambar 21 sistem ini banyak digunakan dalam *smart home*. Ketika pengguna menempelkan kartu RFID yang valid ke reader, data ID kartu dikirim ke MCU-PT. jika ID tersebut sesuai dengan daftar ID yang telah terdaftar di sistem, maka pintu akan terbuka dan lampu akan menyala secara otomatis.



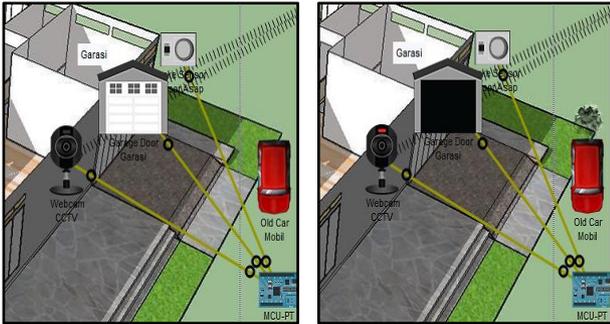
Gbr. 22 Sensor Gerak Sebelum dan Sesudah Terdeteksi

Gambar 22 menunjukkan pengujian sensor gerak di area pintu. Pada kondisi normal, sensor tidak mendeteksi pergerakan sehingga CCTV dan alarm tidak aktif. Saat ada gerakan terdeteksi, sistem otomatis menyalakan alarm dan mengaktifkan CCTV untuk memantau situasi.



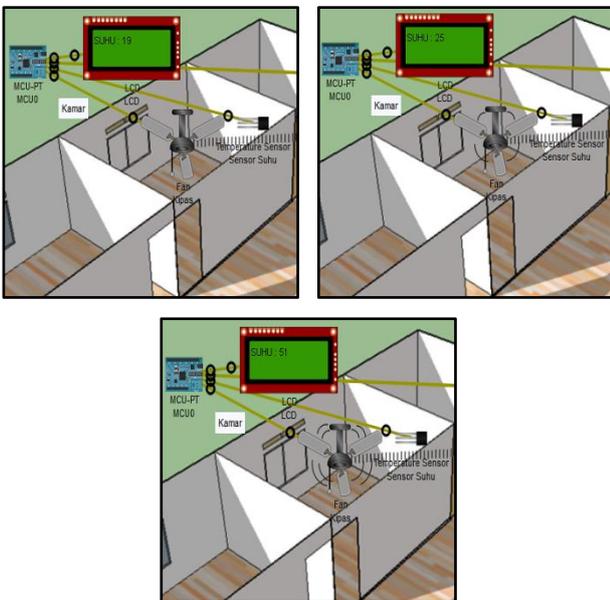
Gbr. 23 Sensor Gas Sebelum dan Sesudah Terdeteksi

Gambar 23 menunjukkan simulasi deteksi gas karbon monoksida (CO) di dapur. Pada kondisi normal sensor mendeteksi udara aman, sedangkan pada gambar kedua terdeteksi gas CO dari mobil yang menyala di dekat dapur. Sistem bekerja berdasarkan ambang batas CO, di mana kadar <9 ppm aman dan >35 ppm berbahaya sehingga memerlukan penanganan segera.



Gbr. 24 Sensor Asap Sebelum Dan Sesudah Terdeteksi

Pada Gambar 24 terdapat beberapa komponen yang saling terhubung, yaitu sensor asap, CCTV, garasi otomatis. Ketika mobil mendekati area garasi dan terdeteksi adanya asap mobil sistem akan secara otomatis mengaktifkan kamera CCTV untuk merekam aktivitas di sekitar, tetapi jika tidak terdeteksi adanya asap mobil di sekitar garasi maka garasi akan tetap tertutup.

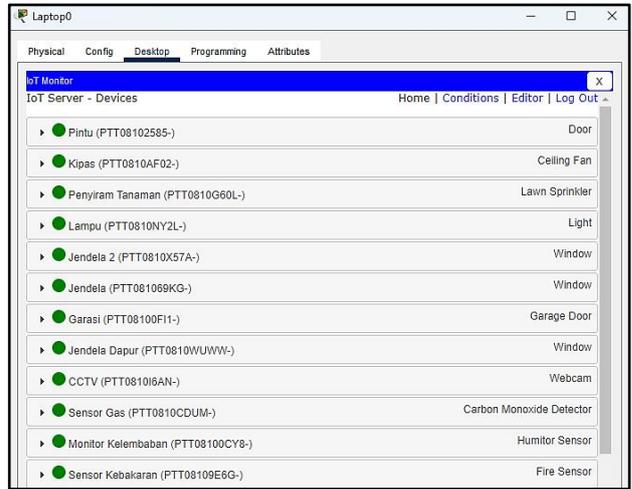
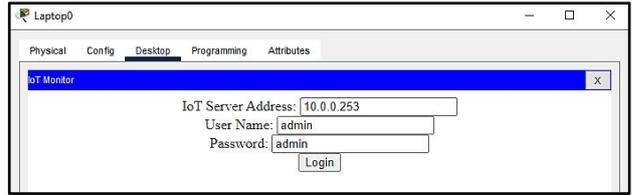


Gbr. 25 Sensor Suhu Sebelum dan Sesudah Terdeteksi

Gambar 25 menunjukkan pengujian sensor suhu dengan kipas dan LCD, di mana pada suhu 19°C kipas tidak aktif karena masih normal, pada 25°C kipas menyala dengan kecepatan sedang untuk menstabilkan suhu, dan pada 51°C kipas berputar dengan kecepatan tinggi sebagai respon kondisi sangat panas.

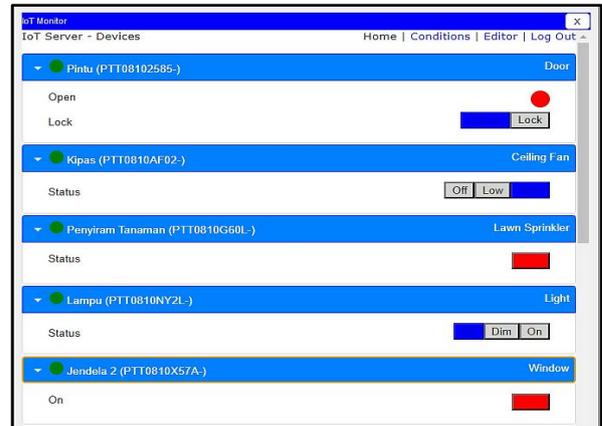
- Monitoring IoT

Monitoring ini memungkinkan pengguna memantau dan mengendalikan perangkat rumah dari mana saja melalui koneksi internet menggunakan *smartphone* atau laptop, termasuk melihat status sensor serta mengaktifkan atau menonaktifkan perangkat dengan mudah.

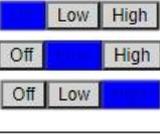
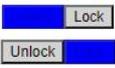


Gbr. 26 Tampilan Halaman Login IoT Monitor

Gambar 26 menunjukkan tampilan antarmuka *IoT Monitor* di *Cisco Packet Tracer*. Pada tahap awal, pengguna *login* ke server IoT (10.0.0.253) dengan username dan password *admin*. Setelah berhasil masuk, ditampilkan daftar perangkat *smart home* seperti jendela, pintu, lampu, kipas, CCTV, hingga penyiram tanaman. Setiap perangkat ditandai indikator hijau yang menandakan aktif dan terkoneksi, serta dapat dipantau dan dikendalikan melalui antarmuka tersebut.



Keterangan:

Tampilan	Kondisi
	On / Aktif
	Off / Tidak Aktif
	Terkoneksi / Online
	Tidak Terkoneksi / Offline
	<ul style="list-style-type: none"> Perangkat dalam kondisi Mati (Off) Perangkat dalam mode Low (rendah) Perangkat dalam mode High (tinggi)
	<ul style="list-style-type: none"> Perangkat dalam kondisi Mati (Off) Perangkat dalam mode Redup (Dim) Perangkat dalam kondisi Nyala (On)
	<ul style="list-style-type: none"> Perangkat dalam kondisi Tidak Terkunci (Unlock) Perangkat dalam mode Terkunci (Lock)
	<ul style="list-style-type: none"> Perangkat dalam mode Sesuai (Valid) Perangkat dalam mode Tidak Sesuai (Invalid) Perangkat dalam mode Menunggu (Waiting)

H. Analisis Data

Pada hasil pengujian konektivitas jaringan dari *router* ke DNS Server dan IoT Server, *throughput* menunjukkan nilai tertinggi sebesar 8192 Kbps pada pengiriman paket berukuran 1024 byte, karena waktu pengirimannya hanya sekitar 0,01 detik. Sebaliknya, *throughput* terendah terjadi pada pengiriman paket 128 byte ke DNS Server yang hanya mencapai 146,3 Kbps, seperti yang ditampilkan pada Tabel 5. Pada penelitian ini, *throughput* meningkat seiring bertambahnya ukuran paket. Selain itu, hasil pengujian *packet loss* pada Tabel 6 menunjukkan bahwa tidak ada paket yang hilang pada semua skenario pengiriman data, baik ke DNS Server maupun IoT Server. Semua paket berhasil diterima dengan baik, menghasilkan persentase *packet loss* sebesar 0% yang menunjukkan bahwa koneksi jaringan tanpa gangguan dalam pengiriman data.

Pengujian *latency* menunjukkan bahwa semakin besar ukuran paket yang dikirimkan, maka rata-rata *latency* juga meningkat. Hal ini dapat dilihat pada Tabel 7, dimana rata-rata *latency* meningkat dari 1 ms pada ukuran 36 byte menjadi 10 ms pada ukuran 1024 byte. Ini juga terjadi pada koneksi ke IoT Server, dari 1 ms menjadi 13 ms. Peningkatan *latency* dikarenakan semakin besar ukuran paket data, waktu yang dibutuhkan untuk dikirim dan diproses juga lebih lama. Namun, berdasarkan Gambar 12 sampai 21, tidak dilakukan perhitungan *jitter* karena data *delay* per paket tidak tersedia. Sedangkan penghitungan *jitter* membutuhkan waktu *delay* tiap paket secara individu agar dapat diketahui selisih antar *delay* paket.

Sementara itu, hasil pengujian konektivitas jaringan dari laptop ke DNS Server dan IoT Server, memiliki nilai *throughput* masing masing sebesar 64 Kbps dan 49,2 Kbps, seperti terlihat pada Tabel 8. Selain itu, hasil pengukuran *packet loss* pada Tabel 9 dengan hasil 0% paket hilang pada kedua skenario. Pada pengukuran *jitter*, nilai yang dihasilkan yaitu 36,22 ms ke DNS Server dan 57,67 ms ke IoT Server yang dapat dilihat pada Tabel 10, keduanya masih dalam kategori “bagus”, meskipun terdapat sedikit variasi waktu antar paket. Pada hasil pengukuran *latency* yang dapat dilihat pada Tabel 11 menunjukkan rata-rata 40 ms dan 52 ms, yang termasuk dalam kategori “sangat bagus”, menunjukkan bahwa waktu respon jaringan yang cepat.

Secara keseluruhan, hasil analisis data menunjukkan bahwa jaringan *smart home* yang dirancang melalui *cisco packet tracer* berjalan dengan baik, stabil, dan mampu mendukung komunikasi antar perangkat IoT. Semua sensor merespons sesuai dengan fungsinya, dan sistem mampu menjalankan perintah otomatisasi seperti menyalakan lampu atau alarm dengan lancar.

V. KESIMPULAN

- Throughput* dari laptop ke DNS Server dan IoT Server berada pada 64 Kbps dan 49,2 Kbps, yang termasuk cocok untuk mendukung komunikasi antar perangkat IoT.
- Hasil pengujian menunjukkan 0% *packet loss*, yang berarti tidak ada data yang hilang selama proses transmisi, ini menandakan koneksi jaringan sangat stabil.
- Pengujian *jitter* menghasilkan nilai yang rendah 36,22 ms dan 57,67 ms artinya variasi waktu pengiriman antar paket data kecil.
- Rata-rata nilai *delay* dari perangkat rumah (laptop/smartphone) ke DNS Server dan IoT Server tercatat berada pada 40 ms dan 52 ms, menunjukkan bahwa respon antar perangkat berlangsung cepat dalam menjalankan otomatisasi.
- Seluruh perangkat *smart home* seperti kipas, penyiram tanaman, lampu, jendela, pintu otomatis, serta sensor suhu, asap, gas, kebakaran, kelembaban, RFID dan gerak berhasil merespons kondisi secara otomatis.
- Sistem IoT monitoring memungkinkan pemantauan dan pengendalian perangkat secara *real-time* melalui perangkat *mobile* seperti *smartphone* dan laptop.

REFERENSI

- Shemsi, I. (2018). Implementing smart home using cisco packet tracer simulator. *International Journal of Engineering Science Invention Research & Development*, 4(VII).
- Nababan, L., Elnovreny, J., Sinambela, L., & Syaputra, A. (2023). Pelatihan Pembuatan Simulasi IoT Smart Home dengan Cisco Packet Tracer di

- SMK Budi Agung Medan. *ABDIKAN: Jurnal Pengabdian Masyarakat Bidang Sains dan Teknologi*, 2(3), 350-360.
- [3] Chaudhari, R. R., Joshi, K. K., Joshi, N., & Kumar, M. (2020). Smart And Secure Home Using IoT Simulations With Cisco Packet Tracer. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 6(3), 88-93.
- [4] Sinaga, D. C. P., Tampubolon, G. J., & Ndruru, I. (2024). Implementation of a Smart Home Based on Internet of Things using CISCO Packet Tracer. *Journal of Computer Networks, Architecture and High Performance Computing*, 6(1), 407-418.
- [5] Prayitno, R. H., & Yakti, B. K. (2020). Simulasi Smart Home Menggunakan Cisco Packet Tracer. *Jurnal Ilmiah Informatika Komputer*, 25(2), 115-126.
- [6] Aziz, T., & Liza, R. (2022). Model Simulasi Smarthome Berbasis Internet of Things dan Cloud Computing Menggunakan Cisco Packet Tracer. *SNASTIKOM*, 1(01), 62-71.
- [7] Jie, Y., Pei, J. Y., Jun, L., Yun, G., & Wei, X. (2013, June). Smart home system based on iot technologies. In *2013 International conference on computational and information sciences* (pp. 1789-1791). IEEE.
- [8] Nuradi, A. (2018). Pengembangan Arsitektur Otomatisasi Smart Home dengan Internet of Things. *Jurnal Sistem Cerdas*, 1(2), 51-61.
- [9] Soliman, M., Abiodun, T., Hamouda, T., Zhou, J., & Lung, C. H. (2013, December). Smart home: Integrating internet of things with web services and cloud computing. In *2013 IEEE 5th international conference on cloud computing technology and science* (Vol. 2, pp. 317-320). IEEE.
- [10] Amrullah, A., Al Rasyid, M. U. H., & Winarno, I. (2022). Implementasi dan Analisis Protokol Komunikasi IoT untuk Crowdsensing pada Bidang Kesehatan. *Jurnal Inovtek Polbeng Seri Informatika*, 7(1), 122-135.
- [11] Burhani, M. U., Novianto, D., & Yuliantari, R. V. (2023). Analisis Komunikasi Nirkabel Menggunakan IoT Network Protocols pada Stasiun Cuaca Mini. *Jurnal Teknik Elektro Uniba (JTE UNIBA)*, 7(2), 325-328.
- [12] Akbar, R. A., Bahri, S., & Hasfani, H. Optimasi Pengiriman Data Sensor Menggunakan Pemrosesan Paralel Berbasis Internet of Things (IoT). *Coding: Jurnal Komputer dan Aplikasi*, 13(1), 45-54.
- [13] TIPHON, R. (1999). *Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) General aspects of Quality of Service (QoS)*. DTR/TIPHON-05006 (cb0010cs. PDF).
- [14] Gwangwava, N., & Mubvirwi, T. B. (2021). Design and simulation of IoT systems using the Cisco Packet Tracer. *Advances in Internet of Things*, 11(02), 59.