

# **LAPORAN PRAKTIKUM PEMROGRAMAN**

## **JARINGAN**



**Disusun Oleh :**

**Nur Asiah (231401033)**

**Mata Kuliah : Bahasa Pemrograman Jaringan Komputer  
(DELPHI)**

**Dosen Pengampu : Ucok, S.Kom.,MT**

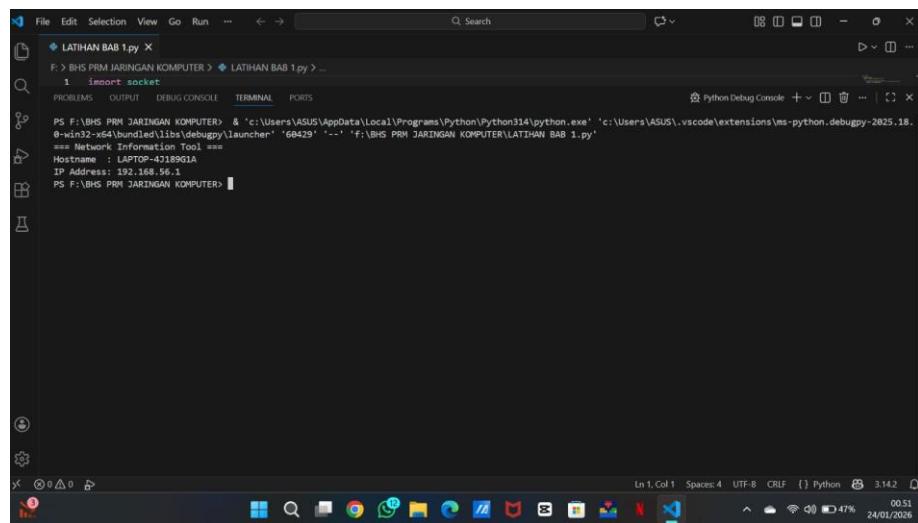
**FAKULTAS ILMU KOMPUTER  
PROGRAM STUDI TEKNIK INFORMATIKA  
UNIVERSITAS INDONESIA TIMUR**

**2026**

## Bab 1: Konsep Dasar Pemrograman Jaringan

Pada gambar praktikum Bab 1 ini membahas konsep fundamental pemrograman jaringan yang menjadi dasar untuk memahami materi selanjutnya. Topik utamanya mencakup jaringan komputer, model client-server, identifikasi alamat IP dan port, serta bagaimana data ditransmisikan antar aplikasi.

### Hasil :

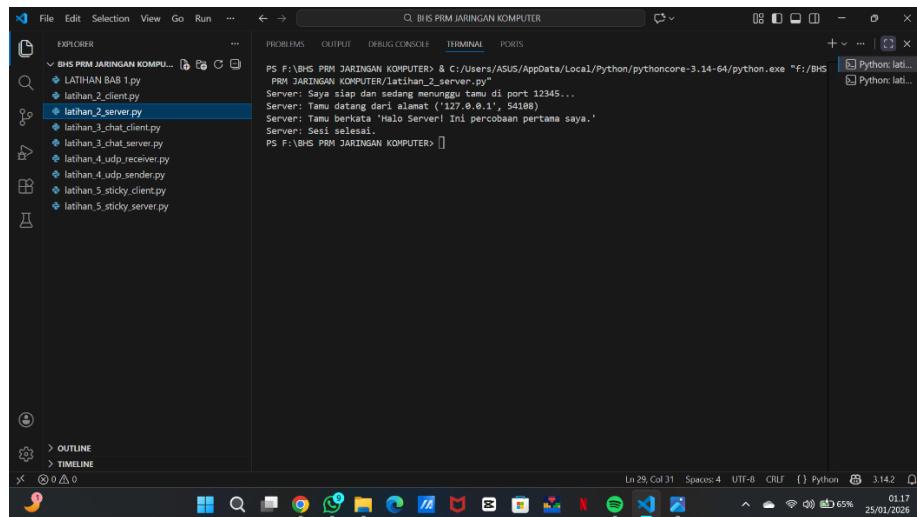


```
LATIHAN BAB 1.py
F: > BHS PRM JARINGAN KOMPUTER > LATIHAN BAB 1.py > ...
1 import socket
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS F:\BHS PRM JARINGAN KOMPUTER> & 'c:\Users\ASUS\AppData\Local\Programs\Python\Python314\python.exe' 'c:\Users\ASUS\.vscode\extensions\ms-python.python.debugpy-2025.18.0-win32-x64\bundled\lib\debugpy\launcher' '68429' '--' 'F:\BHS PRM JARINGAN KOMPUTER\LATIHAN BAB 1.py'
*** Network Information Tool ***
Hostname : LAPTOP-42189GIA
IP Address: 192.168.56.1
PS F:\BHS PRM JARINGAN KOMPUTER>
```

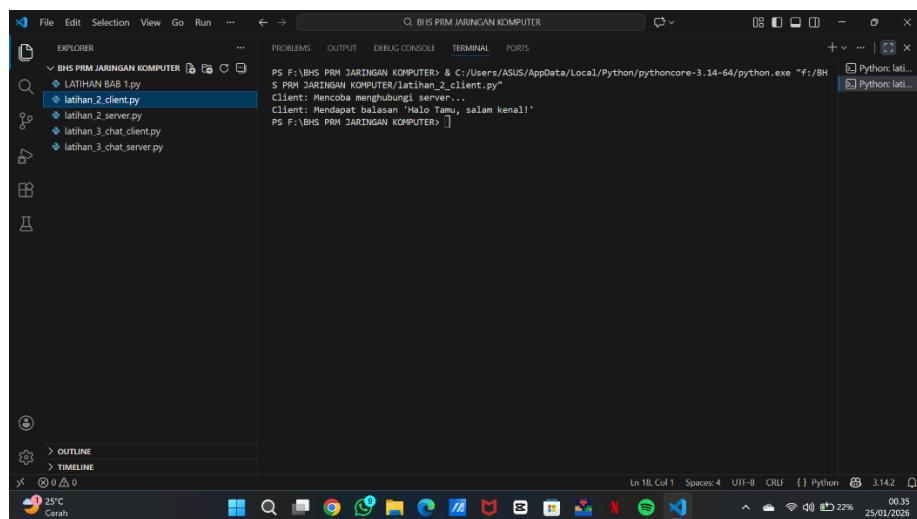
## Bab 2: Socket API Dasar

Pada Bab ini fokus pembahasan bergeser ke API socket, yaitu antarmuka pemrograman yang memungkinkan aplikasi melakukan komunikasi jaringan. Materi mencakup cara membuat socket di Python, mengikat socket ke alamat dan port tertentu, serta operasi dasar seperti mendengarkan koneksi, menerima data, dan mengirim data.

### Hasil :



```
PS F:\BHS PRM JARINGAN KOMPUTER> & C:/Users/ASUS/AppData/Local/Python/pythoncore-3.14-64/python.exe "F:/BHS PRM JARINGAN KOMPUTER/latihan_2_server.py"
PRM JARINGAN KOMPUTER> Saya siap dan sedang menunggu tamu di port 12345...
Server: Tamu datang dari alamat ('127.0.0.1', 54168)
Server: Tamu berkata 'Halo Server! Ini percobaan pertama saya.'
Server: Sesi selesai.
PS F:\BHS PRM JARINGAN KOMPUTER>
```

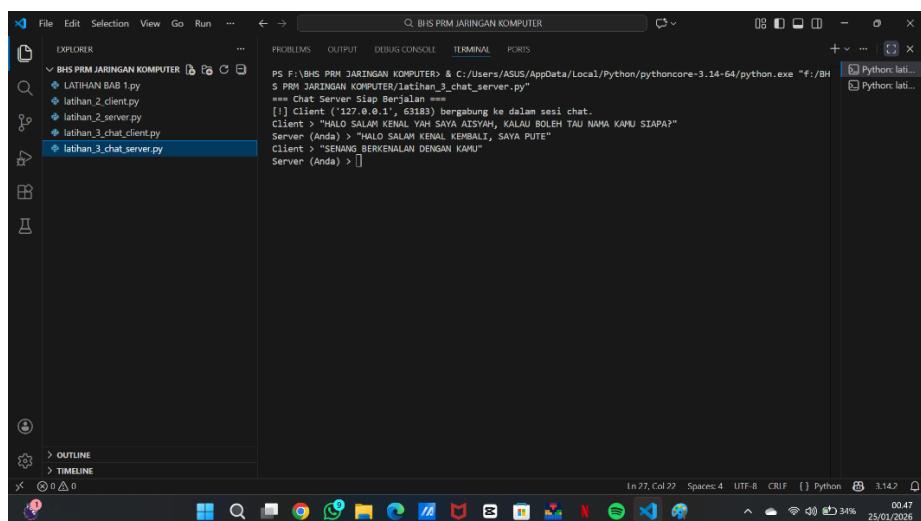


```
PS F:\BHS PRM JARINGAN KOMPUTER> & C:/Users/ASUS/AppData/Local/Python/pythoncore-3.14-64/python.exe "F:/BHS PRM JARINGAN KOMPUTER/latihan_2_client.py"
PRM JARINGAN KOMPUTER> Client: Mencoba menghubungi server...
Client: Mendapat balasan 'Halo Tamu, salam kenal!'
PS F:\BHS PRM JARINGAN KOMPUTER>
```

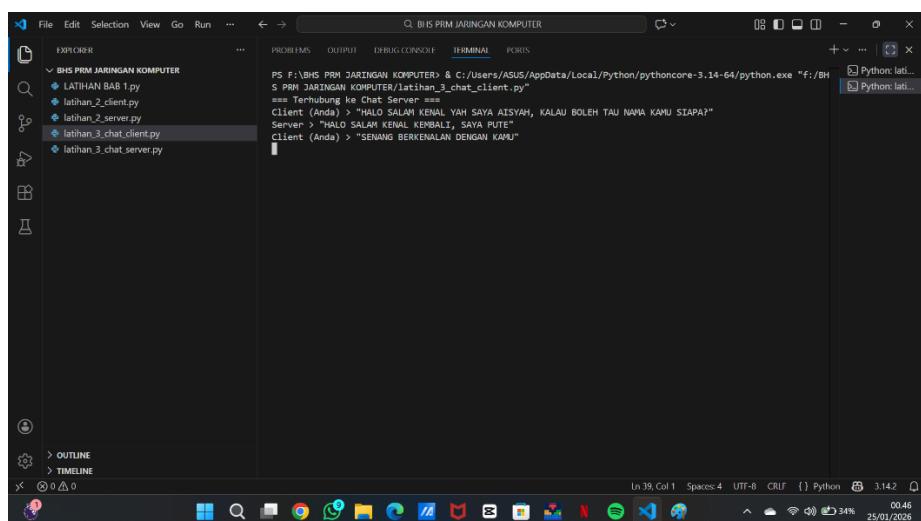
## Bab 3: Protokol TCP (Aplikasi Chat)

Pada Bab ini memfokuskan pada membangun aplikasi chat sederhana menggunakan protokol TCP. Karena TCP menyediakan koneksi yang andal dan berurutan, bab ini menjelaskan bagaimana server TCP dapat mengelola koneksi client, menangani input pengguna, dan menyampaikan pesan secara dua arah. Pembaca belajar bagaimana menulis kode lengkap untuk client dan server yang bisa saling bertukar pesan, serta bagaimana menangani koneksi yang masuk dan keluar secara dinamis.

### Hasil :



```
PS F:\BHS PRM JARINGAN KOMPUTER> & c:/Users/ASUS/AppData/Local/Python/pythoncore-3.14-64/python.exe "f:/BHS PRM JARINGAN KOMPUTER/latihan_3_chat_server.py"
*** Chat Server Siap Berjalan ===
[!] Client ('127.0.0.1', 69183) bergabung ke dalam sesi chat.
Client > "HALO SALAM KENAL YAH SAYA AISYAH, KALAU BOLEH TAU NAMA KAMU SIAPA?"
Server (Anda) > "HALO SALAM KENAL KEMBALI, SAYA PUTE"
Client > "SENANG BERKENALAN DENGAN KAMU"
Server (Anda) > []
```

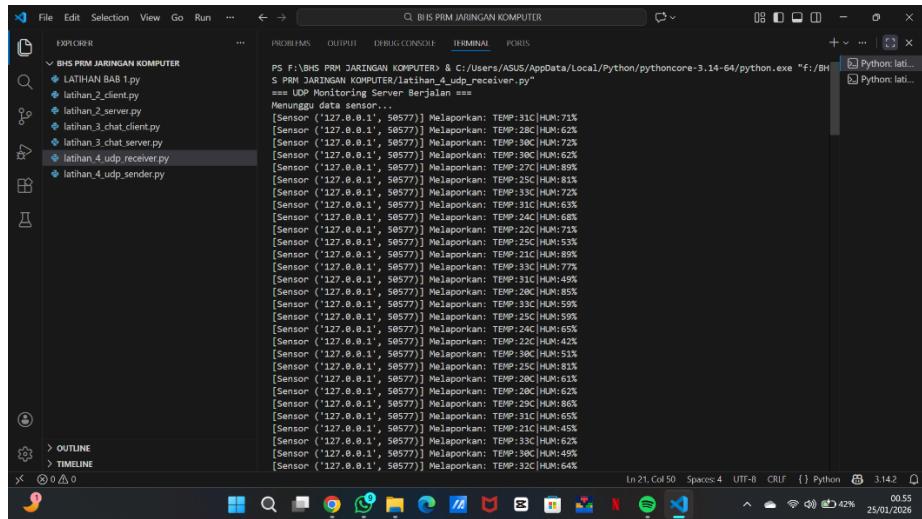


```
PS F:\BHS PRM JARINGAN KOMPUTER> & c:/Users/ASUS/AppData/Local/Python/pythoncore-3.14-64/python.exe "f:/BHS PRM JARINGAN KOMPUTER/latihan_3_chat_client.py"
*** Terhubung ke Chat Server ===
Client (Anda) > "HALO SALAM KENAL YAH SAYA AISYAH, KALAU BOLEH TAU NAMA KAMU SIAPA?"
Server > "HALO SALAM KENAL KEMBALI, SAYA PUTE"
Client (Anda) > "SENANG BERKENALAN DENGAN KAMU"
```

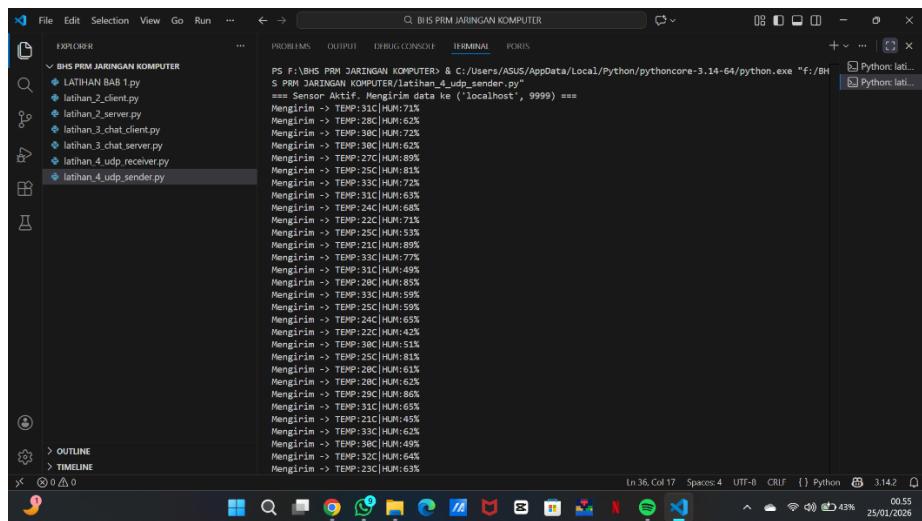
## Bab 4: Protokol UDP (Streaming & Broadcasting)

Pada Bab ini pembahasan beralih ke komunikasi UDP, yang bersifat tanpa koneksi. Bab ini menguraikan penggunaan UDP untuk aplikasi yang membutuhkan latensi rendah dan throughput tinggi, seperti streaming audio/video atau broadcast pesan. Pembaca akan dipandu memahami bagaimana membangun soket UDP, mengirim dan menerima paket data, serta menghadapi tantangan yang muncul karena UDP tidak menjamin pengiriman atau urutan paket.

### Hasil :



```
PS F:\BHS PRM JARINGAN KOMPUTER& C:/Users/ASUS/AppData/Local/Python/pythoncore-3.14-64/python.exe "F:/BHS PRM JARINGAN KOMPUTER/lathian_4_udp_receiver.py"
*** UDP Monitoring Server Berjalan ***
Menunggu data sensor...
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:31C|HUM:71%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:28C|HUM:62%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:30C|HUM:63%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:29C|HUM:62%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:27C|HUM:89%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:25C|HUM:81%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:33C|HUM:72%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:31C|HUM:63%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:24C|HUM:66%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:22C|HUM:71%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:25C|HUM:53%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:34C|HUM:49%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:33C|HUM:77%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:31C|HUM:49%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:28C|HUM:85%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:33C|HUM:59%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:25C|HUM:59%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:24C|HUM:65%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:22C|HUM:42%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:34C|HUM:51%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:33C|HUM:49%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:28C|HUM:61%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:26C|HUM:62%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:29C|HUM:86%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:31C|HUM:65%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:21C|HUM:45%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:33C|HUM:62%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:36C|HUM:49%
[Sensor ('127.0.0.1', 58577)] Melaporkan: TEMP:33C|HUM:64%
```

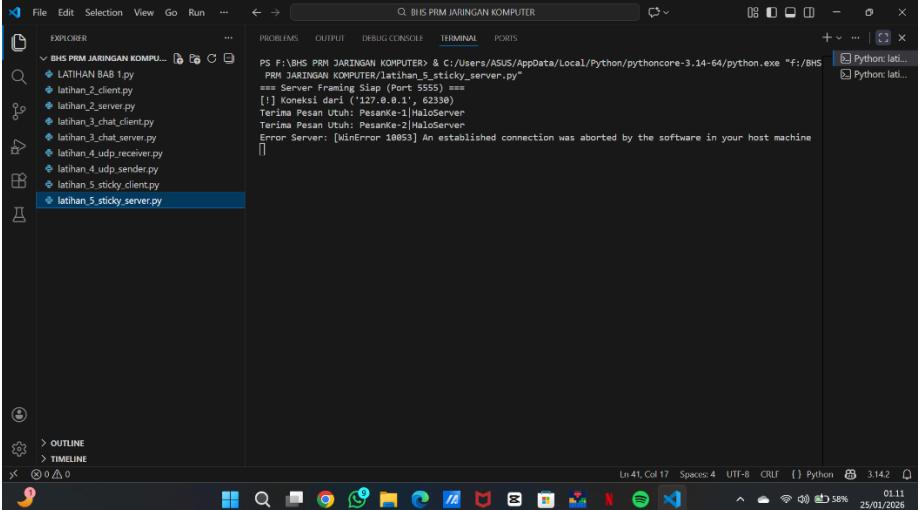


```
PS F:\BHS PRM JARINGAN KOMPUTER& C:/Users/ASUS/AppData/Local/Python/pythoncore-3.14-64/python.exe "F:/BHS PRM JARINGAN KOMPUTER/lathian_4_udp_sender.py"
*** Sensor Aktif: Mengirim data ke ('localhost', 9999) ===
Mengirim -> TEMP:31C|HUM:71%
Mengirim -> TEMP:28C|HUM:62%
Mengirim -> TEMP:30C|HUM:62%
Mengirim -> TEMP:27C|HUM:89%
Mengirim -> TEMP:25C|HUM:51%
Mengirim -> TEMP:34C|HUM:49%
Mengirim -> TEMP:33C|HUM:72%
Mengirim -> TEMP:31C|HUM:63%
Mengirim -> TEMP:24C|HUM:68%
Mengirim -> TEMP:22C|HUM:71%
Mengirim -> TEMP:25C|HUM:53%
Mengirim -> TEMP:21C|HUM:89%
Mengirim -> TEMP:33C|HUM:77%
Mengirim -> TEMP:31C|HUM:69%
Mengirim -> TEMP:28C|HUM:65%
Mengirim -> TEMP:33C|HUM:59%
Mengirim -> TEMP:25C|HUM:59%
Mengirim -> TEMP:24C|HUM:65%
Mengirim -> TEMP:22C|HUM:42%
Mengirim -> TEMP:30C|HUM:51%
Mengirim -> TEMP:25C|HUM:81%
Mengirim -> TEMP:20C|HUM:61%
Mengirim -> TEMP:28C|HUM:62%
Mengirim -> TEMP:30C|HUM:68%
Mengirim -> TEMP:33C|HUM:63%
Mengirim -> TEMP:21C|HUM:45%
Mengirim -> TEMP:33C|HUM:62%
Mengirim -> TEMP:30C|HUM:49%
Mengirim -> TEMP:32C|HUM:64%
Mengirim -> TEMP:23C|HUM:63%
```

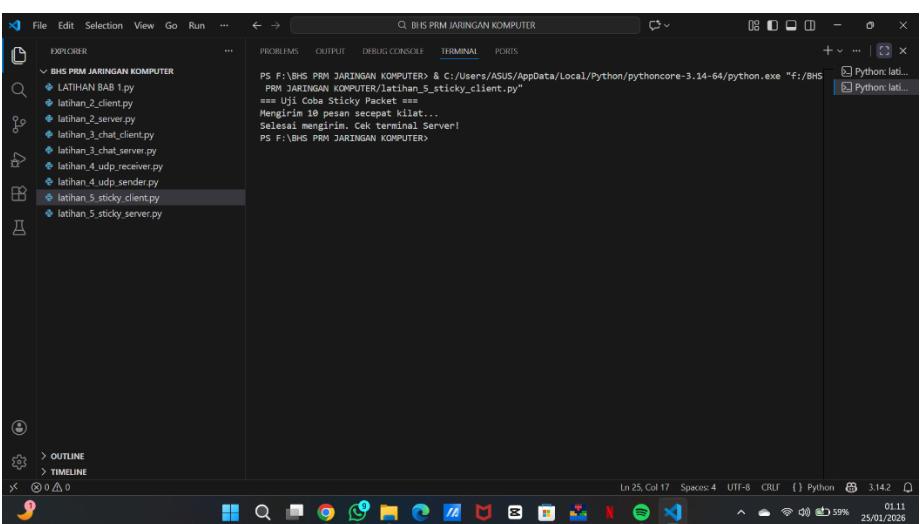
## Bab 5: Error Handling & Framing Data

Pada Bab ini membahas teknik-teknik penting untuk membentuk data (framing) dan menangani kesalahan jaringan. Karena data yang berjalan di jaringan belum tentu utuh atau datang dengan format yang konsisten, penanganan framing membantu memastikan bahwa pesan dapat dipisahkan dan diinterpretasi dengan benar oleh penerima.

### Hasil :



```
PS F:\BHS PRM JARINGAN KOMPUTER> & C:/Users/ASUS/AppData/Local/Python/pythoncore-3.14-64/python.exe "f:/BHS PRM JARINGAN KOMPUTER/latihan_5_sticky_server.py"
PRM JARINGAN KOMPUTER/latihan_5_sticky_server.py"
==> Server Framing Slap (Port 5555)
[!] Koneksi dari "127.0.0.1" port 52390
Terima Pesan Utuh: PesanKe-2 HalloServer
Terima Pesan Utuh: PesanKe-2 HalloServer
Error Server: [MinError 10053] An established connection was aborted by the software in your host machine
```

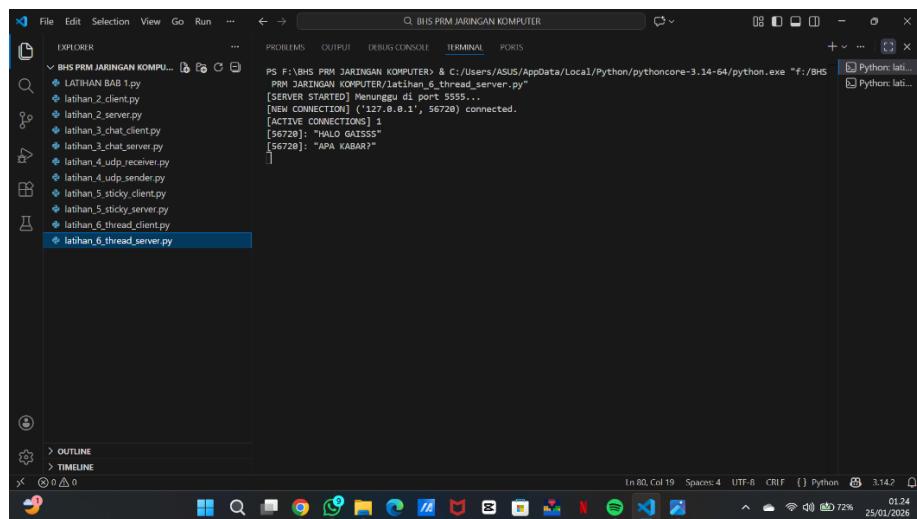


```
PS F:\BHS PRM JARINGAN KOMPUTER> & C:/Users/ASUS/AppData/Local/Python/pythoncore-3.14-64/python.exe "f:/BHS PRM JARINGAN KOMPUTER/latihan_5_sticky_client.py"
PRM JARINGAN KOMPUTER/latihan_5_sticky_client.py"
==> Uji Coba Sticky Packet ==
Mengirim 10 pesan secepat kilat...
Selesai mengirim. Cek terminal Server!
```

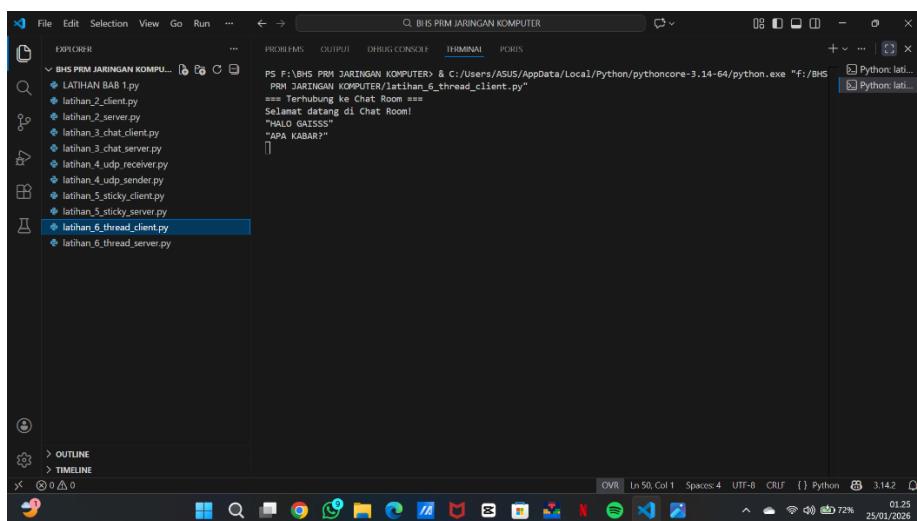
## Bab 6: Concurrency Part I – Threading

Pada Bab ini menjelaskan bagaimana setiap koneksi client dapat ditangani di thread terpisah sehingga server dapat melayani banyak client tanpa blocking satu sama lain. Selain itu dibahas pula bagaimana sinkronisasi sumber daya bersama dilakukan agar tidak terjadi konflik antar thread, serta strategi untuk memaksimalkan performa server pada aplikasi dunia nyata.

### Hasil :



```
PS F:\BHS PRM JARINGAN KOMPUTER> & C:/Users/ASUS/AppData/Local/Python/pythoncore-3.14-64/python.exe "f:/BHS PRM JARINGAN KOMPUTER/latihan_6_thread_server.py"
[SERVER STARTED] Menunggu di port 5555...
[NEW CONNECTION] ('127.0.0.1', 56720) connected.
[ACTIVE CONNECTIONS] 1
[56720]: "HALO GAISS"
[56720]: "APA KABAR?"
```

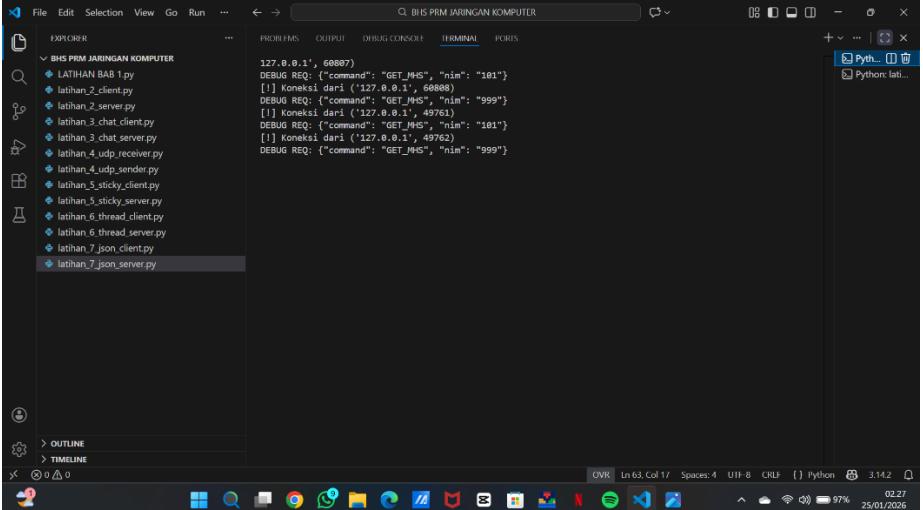


```
PS F:\BHS PRM JARINGAN KOMPUTER> & C:/Users/ASUS/AppData/Local/Python/pythoncore-3.14-64/python.exe "f:/BHS PRM JARINGAN KOMPUTER/latihan_6_thread_client.py"
== Terhubung ke Chat Room ==
Selamat datang di Chat Room!
"HALO GAISS"
"APA KABAR?"
```

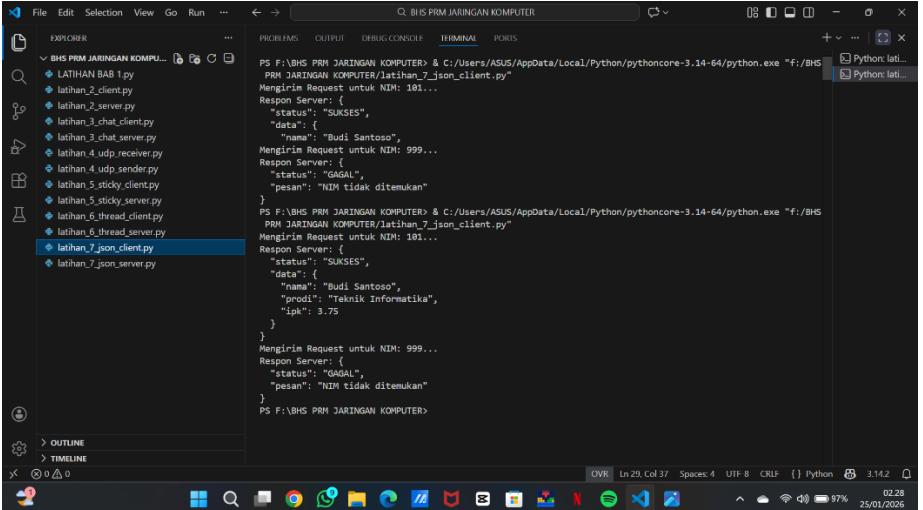
## Bab 7: Serialisasi Data (JSON & Pickle)

Pada Bab ini mengulas pentingnya serialisasi data ketika mengirimkan struktur data yang kompleks melalui jaringan. Dua teknik yang umum dibahas adalah penggunaan format JSON (teks) dan Pickle (biner Python). JSON cocok untuk interoperabilitas dan dibaca manusia, sedangkan Pickle lebih efisien untuk objek Python namun kurang kompatibel di luar Python.

### Hasil :



```
127.0.0.1, 60807
DEBUG REQ: {"command": "GET_NIM", "nim": "101"}
[!] Koneksi dari ('127.0.0.1', 60808)
DEBUG REQ: {"command": "GET_NIM", "nim": "999"}
[!] Koneksi dari ('127.0.0.1', 49761)
DEBUG REQ: {"command": "GET_NIM", "nim": "101"}
[!] Koneksi dari ('127.0.0.1', 49762)
DEBUG REQ: {"command": "GET_NIM", "nim": "999"}  
...
```

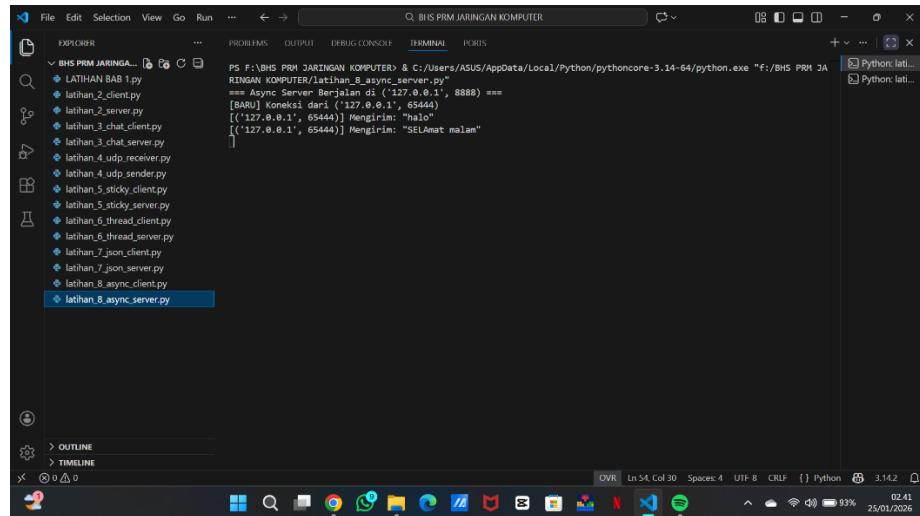


```
Mengirim Request untuk NIM: 101...
Respon Server: {
    "status": "SUKSES",
    "data": [
        {
            "nama": "Budi Santoso",
            "prodi": "Teknik Informatika",
            "ipk": 3.75
        }
    ]
}
Mengirim Request untuk NIM: 999...
Respon Server: {
    "status": "GAGAL",
    "pesan": "NIM tidak ditemukan"
}
PS F:\BHS PRM JARINGAN KOMPUTER>
```

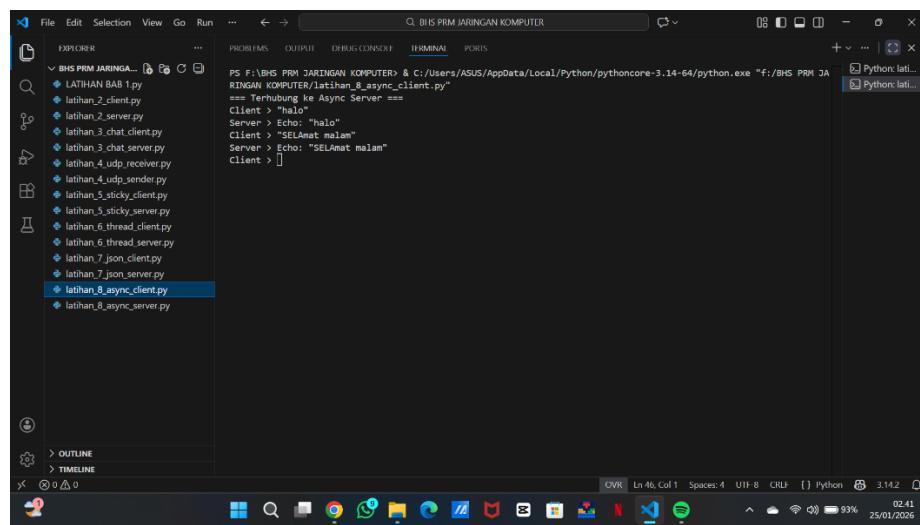
## Bab 8: Asynchronous I/O (Concurrency Part II)

Pada Bab ini memperkenalkan konsep I/O asinkron menggunakan event loop. Teknik ini memungkinkan operasi jaringan berjalan tanpa blocking thread utama, sehingga server bisa mengelola banyak koneksi sekaligus secara efisien. Biasanya diperkenalkan pada modul seperti asyncio di Python, memahami bagaimana coroutine bekerja, serta bagaimana event loop memicu tugas-tugas asinkron sesuai dengan event yang terjadi di jaringan.

### Hasil :



```
PS F:\BHS PRM JARINGAN KOMPUTER> & C:/Users/ASUS/AppData/Local/Python/pythoncore-3.14-64/python.exe "f:/BHS PRM JARINGAN KOMPUTER/latihan_8_async_server.py"
==> Async Server Berjalan di ('127.0.0.1', 8888)
[BARU] Koneksi dari ('127.0.0.1', 65444)
[('127.0.0.1', 65444)] Mengirim: "halo"
[('127.0.0.1', 65444)] Mengirim: "SELamat malam"
```

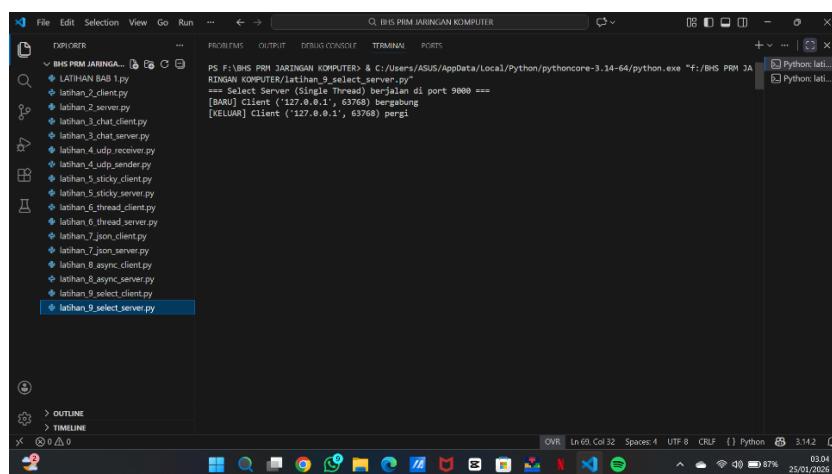


```
PS F:\BHS PRM JARINGAN KOMPUTER> & C:/Users/ASUS/AppData/Local/Python/pythoncore-3.14-64/python.exe "f:/BHS PRM JARINGAN KOMPUTER/latihan_8_async_client.py"
==> Terhubung ke Async Server ...
Client > "halo"
Server > Echo: "halo"
Client > "SELamat malam"
Server > Echo: "SELamat malam"
Client > ]
```

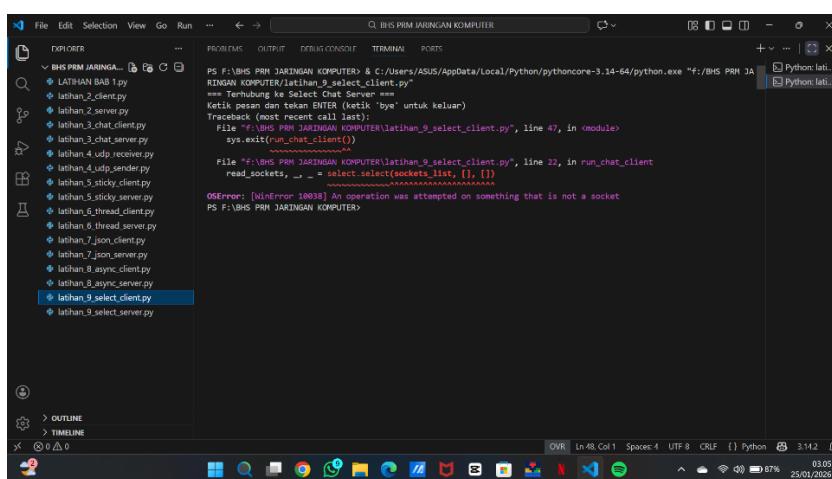
## Bab 9: I/O Multiplexing (select & poll)

Pada Bab ini menjelaskan teknik multiplexing, yakni memantau banyak socket sekaligus tanpa harus menggunakan thread untuk setiap koneksi. Fungsi seperti select() dan poll() memungkinkan program menunggu event pada sekumpulan soket dan merespon hanya ketika ada data yang tersedia. Pendekatan ini sangat efisien untuk server single-threaded yang harus menangani banyak koneksi, dan bab ini memandu memahami bagaimana menggunakan mekanisme tersebut dengan benar untuk meningkatkan kemampuan pemrosesan aplikasi jaringan.

### Hasil :



```
PS F:\BHS PRM JARINGAN KOMPUTER& C:/Users/ASUS/AppData/Local/Python/pythoncore-3.14-64/python.exe "f:/BHS PRM JA
RINGAN KOMPUTER/latihan_9_select_server.py"
*** Select Server (Single Thread) berjalan di port 9000 ===
[BARU] Client ('127.0.0.1', 63768) bergabung
[KELUAR] Client ('127.0.0.1', 63768) pergi
```

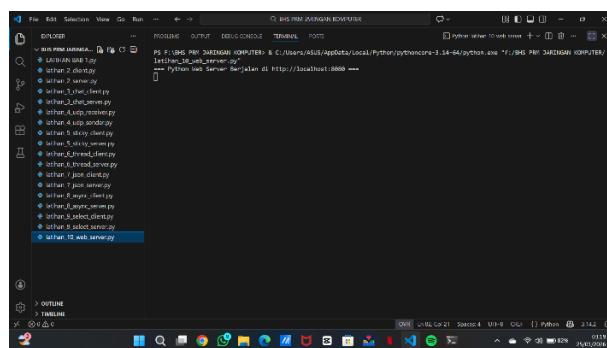


```
PS F:\BHS PRM JARINGAN KOMPUTER& C:/Users/ASUS/AppData/Local/Python/pythoncore-3.14-64/python.exe "f:/BHS PRM JA
RINGAN KOMPUTER/latihan_9_select_client.py"
*** Terhubung ke Select Chat Server ===
Ketik pesan dan tekan ENTER (ketik 'byc' untuk keluar)
Traceback (most recent call last):
  File "<f:\BHS PRM JARINGAN KOMPUTER\latihan_9_select_client.py>", line 47, in <module>
    sys.exit(<__main__.Client>())
  File "<f:\BHS PRM JARINGAN KOMPUTER\latihan_9_select_client.py>", line 22, in run_chat_client
    read_sockets, _, _ = select.select(sockets_list, [], [])
OSERROR: [WinError 10038] An operation was attempted on something that is not a socket
PS F:\BHS PRM JARINGAN KOMPUTER
```

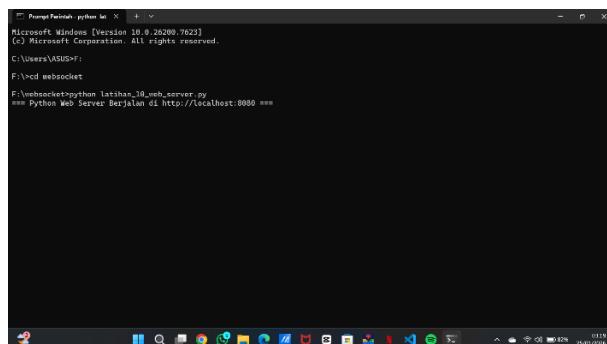
## Bab 10: Protokol HTTP & Web Server

Pada Bab ini memberikan pembahasan mendalam tentang bagaimana HTTP bekerja serta bagaimana merancang web server sederhana menggunakan Python. Materinya mencakup format permintaan dan respon HTTP, status code, header, serta cara menyajikan file HTML dan resource lain ke browser. Pembaca belajar cara memproses request dari browser, membentuk respon yang sesuai, serta menangani berbagai format konten seperti teks, gambar, atau file lainnya.

### Hasil :



A screenshot of a Windows File Explorer window. The path shown is 'C:\Users\AGUS\PycharmProjects\Latihan\_10\_web\_server'. Inside this folder, there are several subfolders and files: 'socket', 'thread', 'http', 'httpd', 'httpd.py', 'httpd\_thread.py', 'httpd\_threaded.py', 'httpd\_select.py', 'httpd\_twisted.py', 'httpd\_wsgi.py', 'httpd\_wsgi\_thread.py', 'httpd\_wsgi\_threaded.py', 'httpd\_wsgi\_select.py', and 'httpd\_wsgi\_twisted.py'. The 'httpd.py' file is highlighted with a blue selection bar at the bottom.



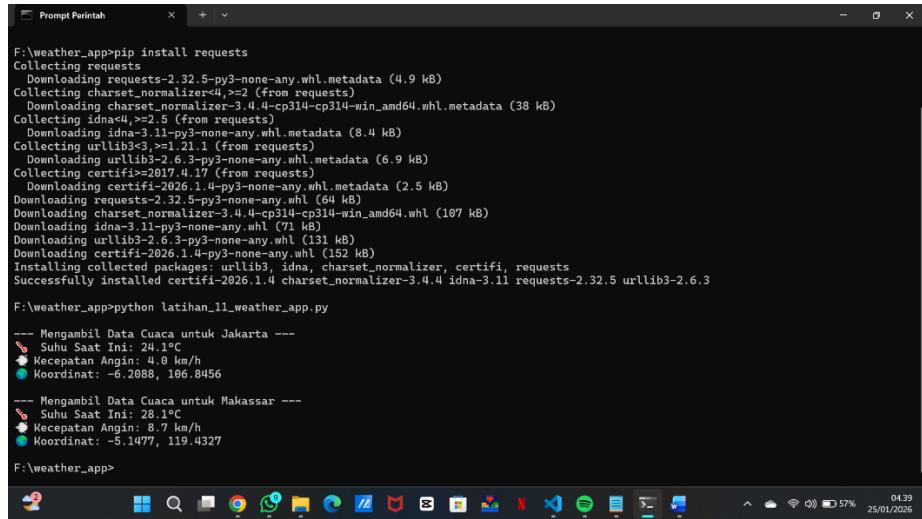
A screenshot of a Windows Command Prompt window titled 'Prompt Pada Kode Web Server'. The command entered is 'F:\websockets>python latihan\_10\_web\_server.py'. The response from the command line is '\*\*\* Python Web Server Berjalan di http://localhost:8080 \*\*\*'.



## Bab 11: REST API & Web Services

Pada Bab ini memperkenalkan konsep REST API, pendekatan modern untuk membangun layanan web yang bisa diakses melalui HTTP dengan format pertukaran data JSON. Bab ini dibimbing bagaimana menyusun API server yang dapat diakses oleh client lain, serta bagaimana client mengambil atau mengirim data ke server melalui permintaan REST.

### Hasil :

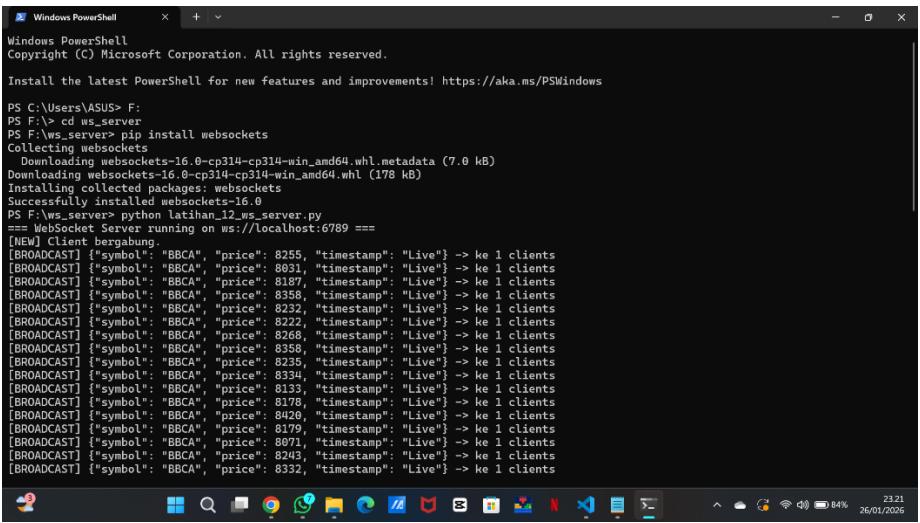


```
F:\weather_app>pip install requests
Collecting requests
  Downloading requests-2.32.5-py3-none-any.whl.metadata (4.9 kB)
Collecting charset_normalizer<4,>=2 (from requests)
  Downloading charset_normalizer-3.4.4-cp311-cp311-win_amd64.whl.metadata (38 kB)
Collecting idna<4,>=2.5 (from requests)
  Downloading idna-3.11-py3-none-any.whl.metadata (8.4 kB)
Collecting urllib3<3,>=1.21.1 (from requests)
  Downloading urllib3-2.6.3-py3-none-any.whl.metadata (6.9 kB)
Collecting certifi=>2017.4.17 (from requests)
  Downloading certifi-2026.1.4-py3-none-any.whl.metadata (2.5 kB)
Downloaded requests-2.32.5-py3-none-any.whl (64 kB)
Downloaded charset_normalizer-3.4.4-cp311-cp311-win_amd64.whl (107 kB)
Downloaded idna-3.11-py3-none-any.whl (71 kB)
Downloaded urllib3-2.6.3-py3-none-any.whl (131 kB)
Downloaded certifi-2026.1.4-py3-none-any.whl (152 kB)
Installing collected packages: urllib3, idna, charset_normalizer, certifi, requests
Successfully installed certifi-2026.1.4 charset_normalizer-3.4.4 idna-3.11 requests-2.32.5 urllib3-2.6.3
F:\weather_app>python latihan_11_weather_app.py
--- Mengambil Data Cuaca untuk Jakarta ---
🌡 Suhu Saat Ini: 24.1°C
💨 Kecepatan Angin: 4.0 km/h
📍 Koordinat: -6.2088, 106.8456
--- Mengambil Data Cuaca untuk Makassar ---
🌡 Suhu Saat Ini: 28.1°C
💨 Kecepatan Angin: 8.7 km/h
📍 Koordinat: -5.1477, 119.4327
F:\weather_app>
```

## Bab 12: Real-time Communication (WebSocket)

Pada Bab ini membahas bagaimana WebSocket berbeda dari HTTP biasa karena koneksiya yang persisten dan full-duplex, sehingga cocok untuk aplikasi chat, game, atau notifikasi langsung. Bab ini juga mempelajari cara membuka koneksi WebSocket, menangani pesan masuk/keluar, serta menjaga koneksi tetap aktif selama sesi berlangsung.

### Hasil :

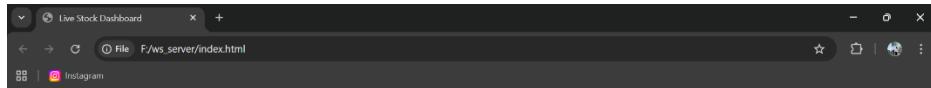


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\ASUS> F:
PS F:>> cd ws_server
PS F:\ws_server> pip install websockets
Collecting websockets
  Downloading websockets-16.0-cp314-cp314-win_amd64.whl.metadata (7.0 kB)
  Downloading websockets-16.0-cp314-cp314-win_amd64.whl (178 kB)
Installing collected packages: websockets
Successfully installed websockets-16.0
PS F:\ws_server> python latihan_12_ws_server.py
== WebSocket Server running on ws://localhost:6789 ==

[NEW] Client bergabung
[BROADCAST] {"symbol": "BBCA", "price": 8255, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8631, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8187, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8358, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8232, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8222, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8268, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8358, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8235, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8334, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8133, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8178, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8420, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8179, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8071, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8243, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8332, "timestamp": "Live"} -> ke 1 clients
```



Pantauan Saham BBCA (Real-time)

Rp 8194

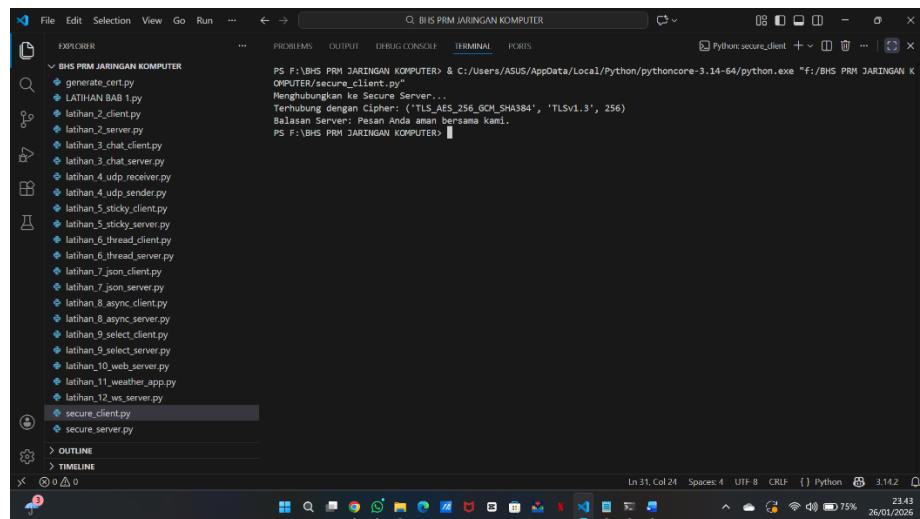
Connected to Server: 



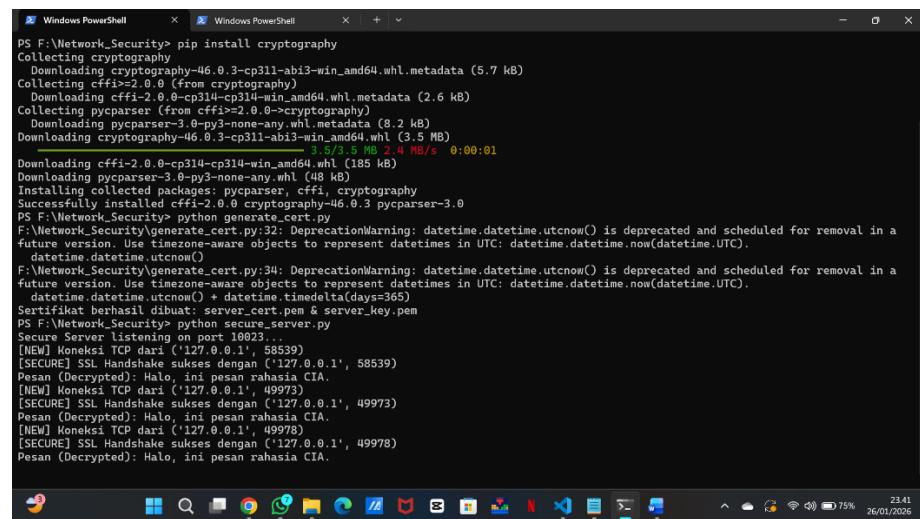
## Bab 13: Keamanan Jaringan (Network Security)

Pada Bab ini menjelaskan berbagai konsep penting seperti enkripsi data, penggunaan SSL/TLS untuk koneksi aman, autentikasi client/server, serta proteksi terhadap serangan umum seperti DoS atau spoofing. Bab ini juga diajarkan praktik terbaik untuk menulis program jaringan yang aman dari eksploitasi, termasuk validasi input, penggunaan sertifikat digital, dan teknik mitigasi risiko agar aplikasi lebih tahan terhadap ancaman keamanan.

### Hasil :



```
File Edit Selection View Go Run ... ← → Q BI S PRM JARINGAN KOMPUTER
EXPLORER PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
F:\BHS PRM JARINGAN KOMPUTER> & C:/Users/ASUS/AppData/Local/Python/pythoncore-3.14-64/python.exe "F:/BHS PRM JARINGAN KOMPUTER/secure_client.py"
Menghubungkan ke Secure Server...
Terhubung dengan Cipher: ('TLS_AES_256_GCM_SHA384', 'TLSv1.3', 256)
Balasan Server: Pesan Anda aman bersama kami.
PS F:\BHS PRM JARINGAN KOMPUTER>
```

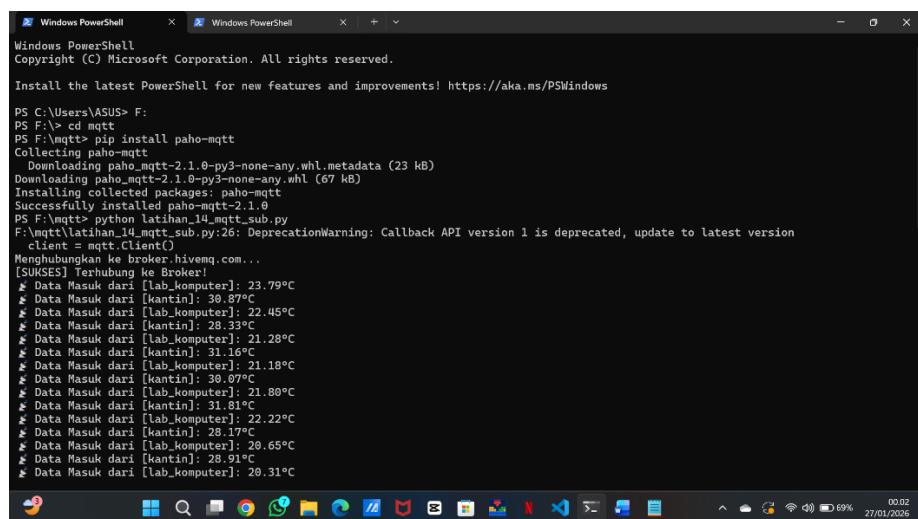


```
PS F:\Network_Security> pip install cryptography
Collecting cryptography
  Downloading cryptography-46.0.3-cp311-abi3-win_amd64.whl.metadata (5.7 kB)
Collecting cffi>=2.0.0 (from cryptography)
  Downloading cffi-2.0.0-cp311-cp311-win_amd64.whl.metadata (2.6 kB)
Collecting pycparser (from cffi>=2.0.0->cryptography)
  Downloading pycparser-3.0-py3-none-any.whl.metadata (8.2 kB)
Collecting pycparser-3.0-py3-none-any.whl (3.5 MB)
  Downloading pycparser-3.0-py3-none-any.whl (3.5 MB)
  Downloading cffi-2.0.0-cp311-abi3-win_amd64.whl (185 kB)
  Downloading pycparser-3.0-py3-none-any.whl (49 kB)
Installing collected packages: pycparser, cffi, cryptography
Successfully installed cffi<=2.0.0 cryptography<=46.0.3 pycparser-3.0
PS F:\Network_Security> python generate_cert.py
F:\Network_Security\generate_cert.py:32: DeprecationWarning: datetime.datetime.utcnow() is deprecated and scheduled for removal in a future version. Use timezone-aware objects to represent datetimes in UTC: datetime.datetime.now(datetime.UTC).
  datetime.datetime.utcnow()
F:\Network_Security\generate_cert.py:34: DeprecationWarning: datetime.datetime.utcnow() is deprecated and scheduled for removal in a future version. Use timezone-aware objects to represent datetimes in UTC: datetime.datetime.now(datetime.UTC).
  datetime.datetime.utcnow()
  datetime.timedelta(days=365)
Sertifikat berhasil dibuat: server_cert.pem & server_key.pem
PS F:\Network_Security> python secure_server.py
Secure Server listening on port 10023...
[NEW] Koneksi TCP dari ('127.0.0.1', 58539)
[SECURE] SSL Handshake sukses dengan ('127.0.0.1', 58539)
Pesan (Decrypted): Halo, ini pesan rahasia CIA.
[NEW] Koneksi TCP dari ('127.0.0.1', 49973)
[SECURE] SSL Handshake sukses dengan ('127.0.0.1', 49973)
Pesan (Decrypted): Halo, ini pesan rahasia CIA.
[NEW] Koneksi TCP dari ('127.0.0.1', 49978)
[SECURE] SSL Handshake sukses dengan ('127.0.0.1', 49978)
Pesan (Decrypted): Halo, ini pesan rahasia CIA.
```

## Bab 14: Arsitektur Sistem Terdistribusi & IoT (MQTT)

Pada Bab ini memperkenalkan protokol ringan seperti MQTT yang dirancang untuk komunikasi efisien antar perangkat resource-limited. Pembaca belajar cara kerja pola publish/subscribe MQTT, penggunaan broker MQTT, serta bagaimana perangkat dan aplikasi berbasis IoT dapat saling bertukar data secara efisien dalam sistem terdistribusi.

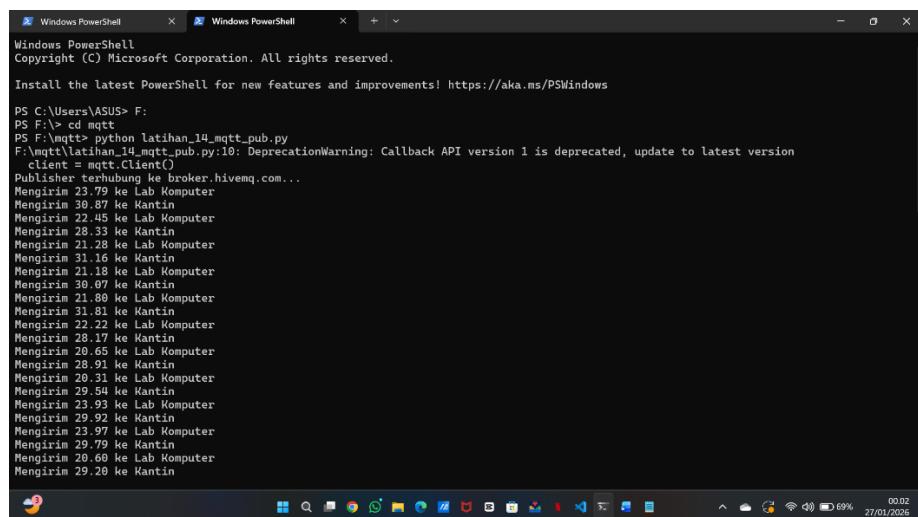
### Hasil :



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\ASUS> F:
PS F:>\ cd mqtt
PS F:\mqtt> pip install paho-mqtt
Collecting paho-mqtt
  Downloading paho_mqtt-2.1.0-py3-none-any.whl.metadata (23 kB)
  Downloading paho_mqtt-2.1.0-py3-none-any.whl (67 kB)
Installing collected packages: paho-mqtt
Successfully installed paho-mqtt-2.1.0
PS F:\mqtt> python latihan_14_mqtt_sub.py
F:\mqtt\latihan_14_mqtt_sub.py:26: DeprecationWarning: Callback API version 1 is deprecated, update to latest version
  client = mqtt.Client()
Menghubungkan ke broker.hivemq.com...
[SUKSES] Terhubung ke Broker!
* Data Masuk dari [Lab_komputer]: 23.79°C
* Data Masuk dari [Kantin]: 30.87°C
* Data Masuk dari [Lab_komputer]: 22.45°C
* Data Masuk dari [Kantin]: 28.33°C
* Data Masuk dari [Lab_komputer]: 21.28°C
* Data Masuk dari [Kantin]: 31.16°C
* Data Masuk dari [Lab_komputer]: 21.18°C
* Data Masuk dari [Kantin]: 30.07°C
* Data Masuk dari [Lab_komputer]: 21.80°C
* Data Masuk dari [Kantin]: 31.81°C
* Data Masuk dari [Lab_komputer]: 22.22°C
* Data Masuk dari [Kantin]: 28.17°C
* Data Masuk dari [Lab_komputer]: 28.65°C
* Data Masuk dari [Kantin]: 28.91°C
* Data Masuk dari [Lab_komputer]: 28.31°C
```



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

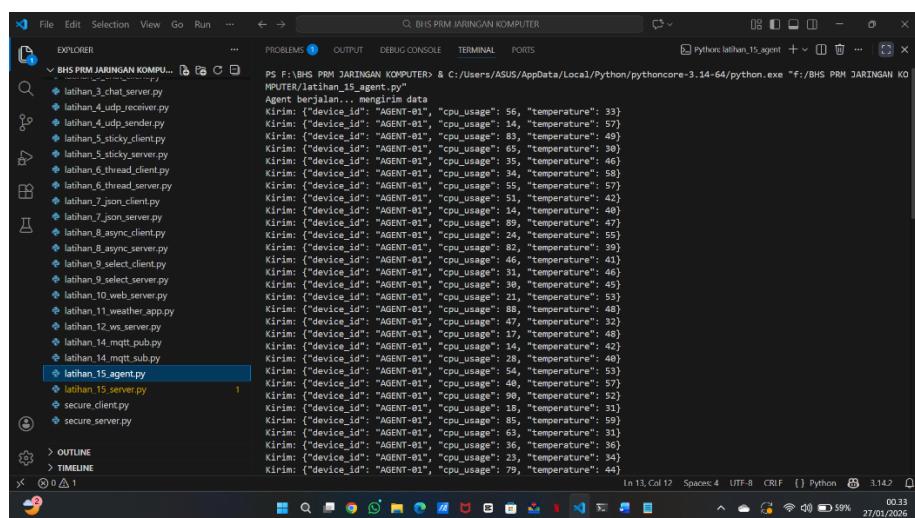
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\ASUS> F:
PS F:>\ cd mqtt
PS F:\mqtt> python latihan_14_mqtt_pub.py:10: DeprecationWarning: Callback API version 1 is deprecated, update to latest version
  client = mqtt.Client()
Publisher terhubung ke broker.hivemq.com...
Mengirim 23.79 ke Lab Komputer
Mengirim 30.87 ke Kantin
Mengirim 22.45 ke Lab Komputer
Mengirim 28.33 ke Kantin
Mengirim 21.28 ke Lab Komputer
Mengirim 31.16 ke Kantin
Mengirim 21.18 ke Lab Komputer
Mengirim 30.07 ke Kantin
Mengirim 21.80 ke Lab Komputer
Mengirim 31.81 ke Kantin
Mengirim 22.22 ke Lab Komputer
Mengirim 28.17 ke Kantin
Mengirim 28.65 ke Lab Komputer
Mengirim 28.91 ke Kantin
Mengirim 20.31 ke Lab Komputer
Mengirim 29.54 ke Kantin
Mengirim 23.93 ke Lab Komputer
Mengirim 29.92 ke Kantin
Mengirim 23.97 ke Lab Komputer
Mengirim 29.79 ke Kantin
Mengirim 20.60 ke Lab Komputer
Mengirim 29.28 ke Kantin
```

## Bab 15: Penutup & Proyek Akhir (Capstone Project)

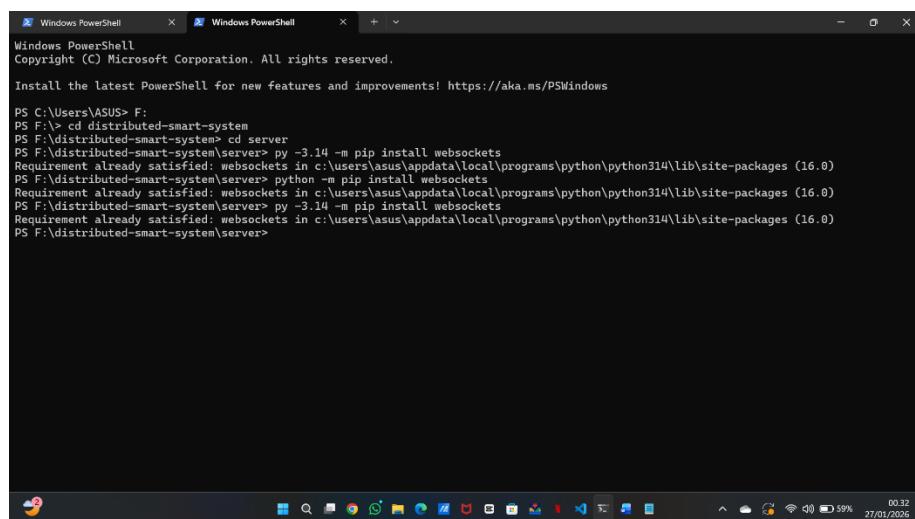
Bagian terakhir ini adalah proyek akhir yang mensintesis semua konsep yang telah dipelajari dari bab-bab sebelumnya. Pembaca diwajibkan merancang dan membangun sebuah aplikasi jaringan end-to-end yang mencakup berbagai teknik seperti socket programming, concurrency, protokol HTTP/REST, dan mungkin WebSocket atau IoT. Tujuannya adalah memastikan pembaca tidak hanya memahami teori, tapi juga mampu mengimplementasikan solusi nyata yang kompleks dan berfungsi penuh.

### Hasil :

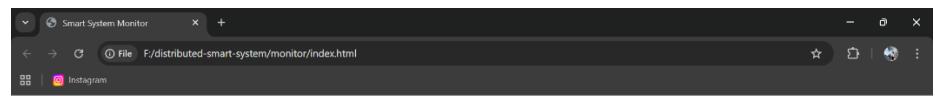


```
PS F:\BHS PRM JARINGAN KOMPUTER> & C:/Users/ASUS/AppData/Local/Python/pythoncore-3.14-64/python.exe "F:/BHS PRM JARINGAN KOMPUTER/latihan_15_agent.py"
Agent berjalan... mengirim data
Kirim: {"device_id": "AGENT-01", "cpu_usage": 56, "temperature": 33}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 14, "temperature": 57}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 83, "temperature": 49}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 65, "temperature": 30}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 35, "temperature": 46}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 34, "temperature": 58}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 55, "temperature": 57}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 51, "temperature": 43}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 14, "temperature": 48}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 89, "temperature": 47}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 24, "temperature": 55}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 82, "temperature": 39}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 46, "temperature": 41}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 31, "temperature": 46}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 25, "temperature": 45}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 21, "temperature": 43}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 81, "temperature": 48}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 47, "temperature": 32}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 17, "temperature": 48}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 14, "temperature": 42}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 54, "temperature": 53}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 29, "temperature": 57}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 99, "temperature": 23}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 18, "temperature": 31}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 85, "temperature": 99}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 63, "temperature": 31}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 36, "temperature": 36}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 23, "temperature": 34}
Kirim: {"device_id": "AGENT-01", "cpu_usage": 79, "temperature": 44}

In 13, Col 12  Spaces: 4  UTF-8  CR/LF  Python  3.14.2  00:33  27/01/2026
```



```
PS C:\Users\ASUS> F:
PS F:>>> cd distributed-smart-system
PS F:\distributed-smart-system> cd server
PS F:\distributed-smart-system\server> py -3.14 -m pip install websockets
Requirement already satisfied: websockets in c:/users/asus/appdata/local\programs\python\python314\lib\site-packages (16.0)
PS F:\distributed-smart-system\server> python -m pip install websockets
Requirement already satisfied: websockets in c:/users/asus/appdata/local\programs\python\python314\lib\site-packages (16.0)
PS F:\distributed-smart-system\server> py -3.14 -m pip install websockets
Requirement already satisfied: websockets in c:/users/asus/appdata/local\programs\python\python314\lib\site-packages (16.0)
PS F:\distributed-smart-system\server>
```



### Login

username password Login

### Live Dashboard

Waiting...

