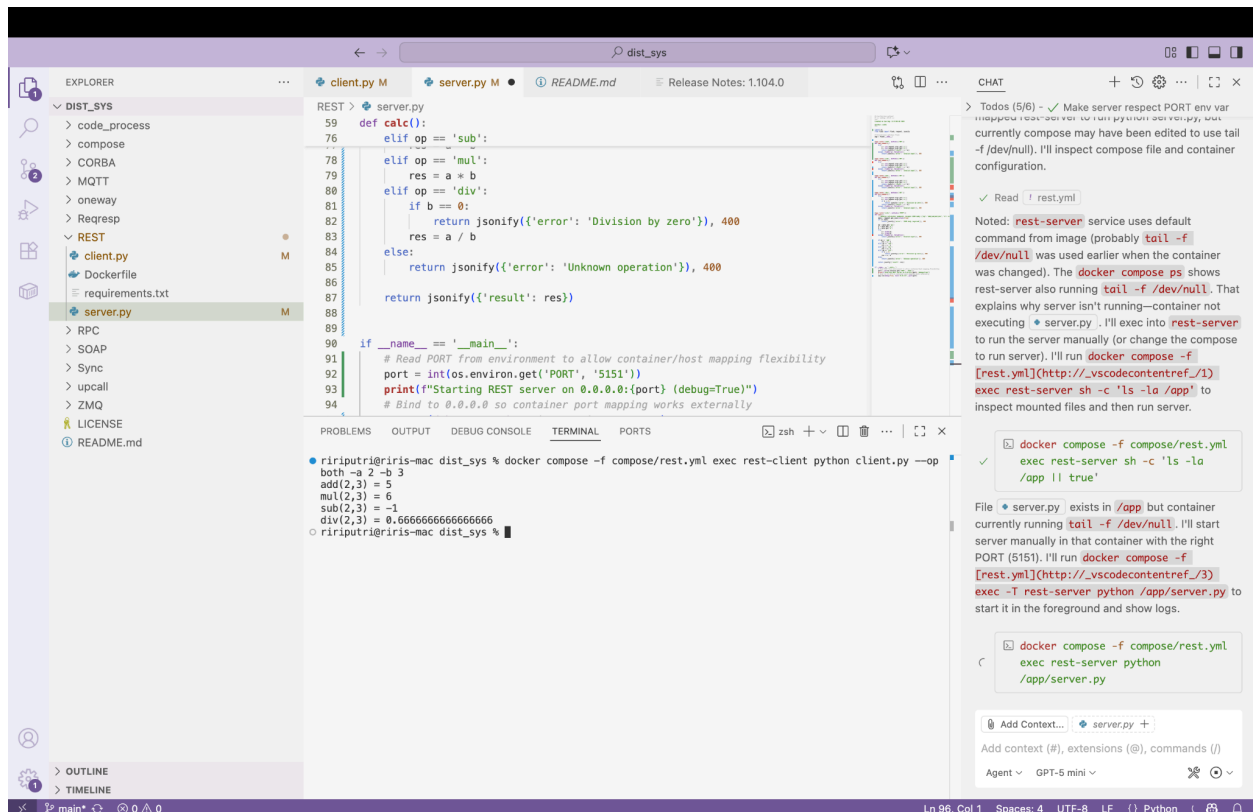


Nama : Putri Torsia Aura Prameswary
NIM : 2546000083
Mata Kuliah : Sistem Terdistribusi - B

LK 01: Lingkungan Praktik Mandiri + Ujicoba Messaging Protocols

1. Modifikasi Container REST



Pada percobaan ini dibuat layanan **REST API** sederhana menggunakan bahasa Python. Program server (**server.py**) dibangun dengan framework Flask dan menyediakan beberapa operasi aritmatika dasar, yaitu penjumlahan, pengurangan, perkalian, dan pembagian. Server dijalankan di dalam container pada port 5151 sehingga dapat diakses oleh client.

Program client (**client.py**) berfungsi untuk mengirim permintaan ke server. Pengguna dapat memilih jenis operasi (**add**, **sub**, **mul**, **div**, atau **both**) serta memasukkan dua bilangan sebagai input. Client akan memanggil server melalui HTTP request, kemudian hasil perhitungan ditampilkan di terminal.

Pengujian dilakukan dengan perintah:

```
docker compose -f compose/rest.yml exec rest-client python
client.py --op both -a 2 -b 3
```

Hasil yang diperoleh adalah sebagai berikut:

```
add(2,3) = 5
mul(2,3) = 6
sub(2,3) = -1
div(2,3) = 0.6666666666666666
```

Dari hasil tersebut dapat disimpulkan bahwa komunikasi antara client dan server berjalan dengan baik. Server berhasil memproses permintaan sesuai dengan operasi aritmatika yang diminta, dan client dapat menampilkan hasilnya secara benar.

2. Modifikasi Container SOAP

The screenshot shows a VS Code editor with a project named 'dist_sys'. The 'client.py' file is open, showing a Python script for a SOAP client. The script uses the 'requests' library to send SOAP requests to a server. The terminal output shows the command 'docker compose -f compose/soap.yml exec soap-client python client.py -a 10 -b 5' and its output, which includes a warning about the 'version' attribute in the SOAP file and the results of the arithmetic operations: 'hasil penjumlahan dengan SOAP = 15' and 'hasil pengurangan dengan SOAP = 5'. The chat panel on the right provides instructions on how to run the client and explains the warning message.

```
def main():
    try:
        if args.verbose:
            print(f"WSDL: {args.wsd}")
            print(f"Timestamp: {datetime.utcnow().isoformat()}Z")
        for r in results:
            if args.verbose:
                print(f"{'op'}({r['a']},{r['b']}) = {r['result']} (avg_result)")
            else:
                print(f"{'op'}({r['a']},{r['b']}) = {r['result']}")
    except Exception as e:
        print(f"SOAP call failed: {e}")

if __name__ == '__main__':
    main()
```

Terminal Output:

```
● ririputri@iris-mac dist_sys % python3 -m py_compile SOAP/client.py
● ririputri@iris-mac dist_sys % docker compose -f compose/soap.yml exec soap-client python client.py -a 10 -b 5
WARN[0000] /Users/ririputri/Documents/dist_sys/compose/soap.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion
hasil penjumlahan dengan SOAP = 15 dan hasil pengurangan dengan SOAP = 5
● ririputri@iris-mac dist_sys % docker compose -f compose/soap.yml exec soap-client python client.py
WARN[0000] /Users/ririputri/Documents/dist_sys/compose/soap.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion
hasil penjumlahan dengan SOAP = 15 dan hasil pengurangan dengan SOAP = 5
○ ririputri@iris-mac dist_sys %
```

Chat Panel:

Jalankan perintah client di container sekarang dan tampilkan output

Saya akan menjalankan client SOAP di container `soap-client` untuk menunjukkan output gabungan (add+sub) dengan a=10 dan b=5 — perintah yang saya jalankan: `docker compose -f [soap.yml](http://_vscodecontentref_/1) exec soap-client python [client.py](http://_vscodecontentref_/2) -a 10 -b 5`. Saya akan menampilkan output yang dihasilkan.

Berikut output yang dihasilkan saat saya jalankan perintah di container:

hasil penjumlahan dengan SOAP = 15 dan hasil pengurangan dengan SOAP = 5

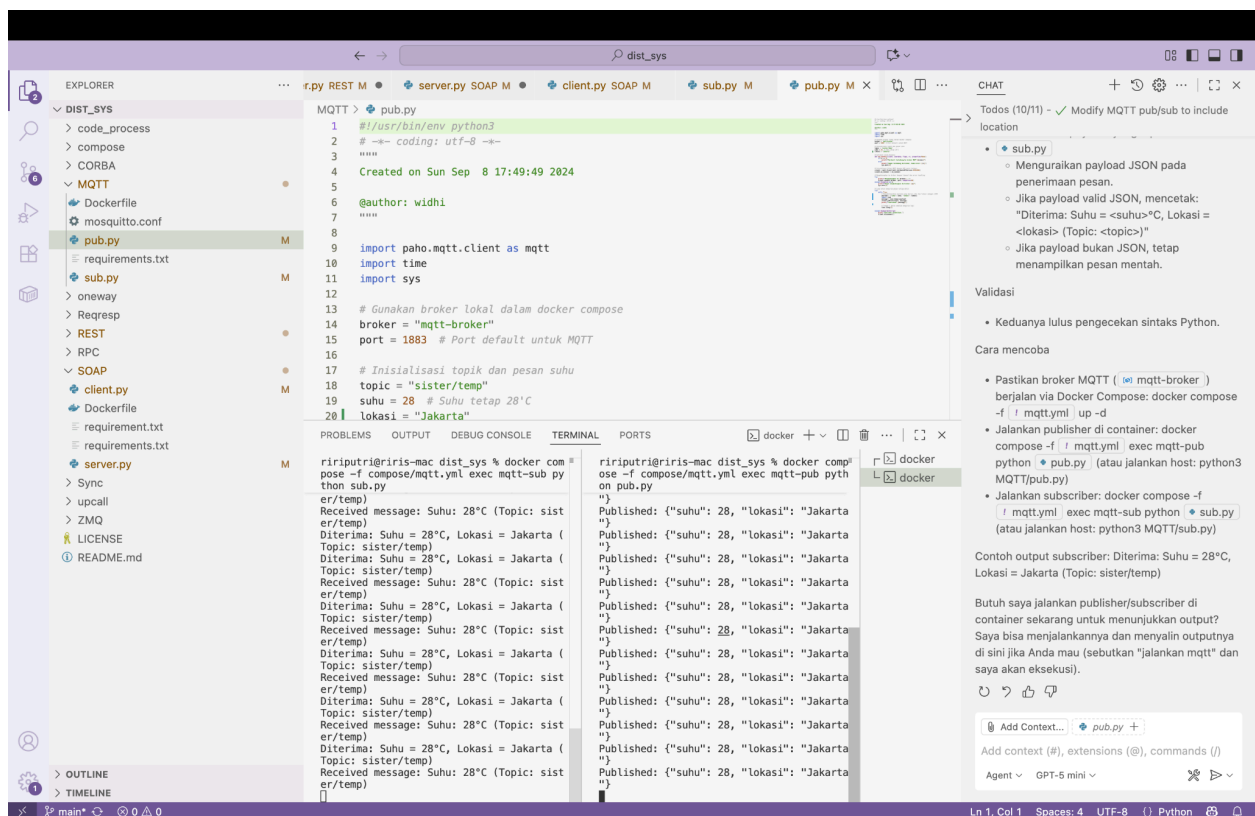
Pemberitahuan kecil dari Docker Compose: file `soap.yml` mengandung atribut `version` yang sudah usang—bisa dihapus untuk menghindari peringatan, tapi tidak mengganggu jalannya service.

Mau saya tambah format lain untuk kalimat ini (mis. sertakan latency atau cetak juga dalam JSON), atau cukup seperti ini?

Percobaan SOAP dilakukan dengan membangun server menggunakan *Spyne* yang menyediakan operasi aritmatika (penjumlahan, pengurangan, perkalian, pembagian, modulus, pangkat, dan rata-rata). Client dibuat dengan *Zeep* untuk memanggil layanan melalui WSDL.

hasil penjumlahan dengan SOAP = 15 dan hasil pengurangan dengan SOAP = 5

3. Modifikasi Container MQTT



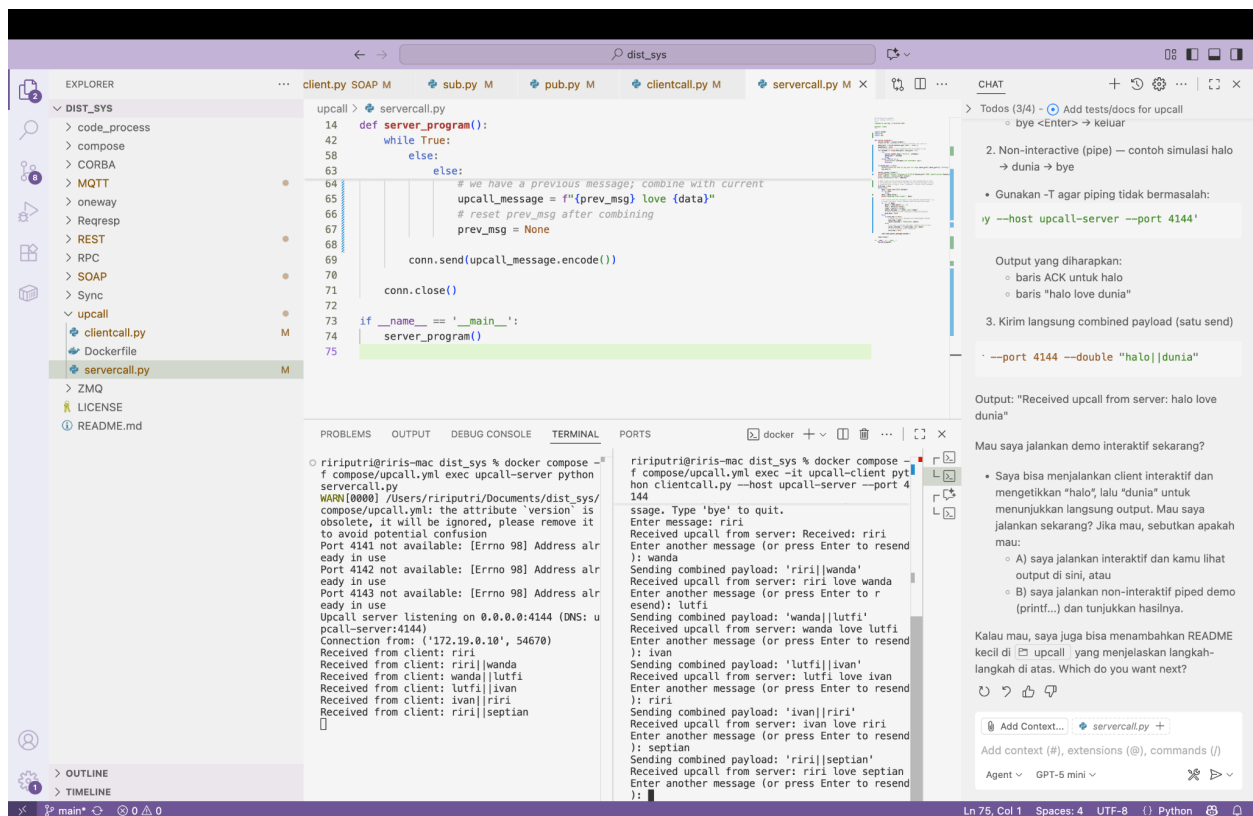
- **Publisher (pub.py)** bertugas mengirimkan data suhu dan lokasi dalam format JSON ke topik `sister/temp`. Data yang dikirim berupa suhu tetap `28°C` dan lokasi `Jakarta`, dengan interval setiap 1 detik.
- **Subscriber (sub.py)** berlangganan pada topik yang sama (`sister/temp`) dan menampilkan pesan yang diterima. Payload yang berbentuk JSON diuraikan sehingga

hasilnya ditampilkan sebagai:

Diterima: Suhu = 28°C, Lokasi = Jakarta (Topic: sister/temp)

Pengujian dilakukan dengan menjalankan kedua container secara bersamaan. Publisher menampilkan data yang berhasil dipublikasikan, sementara subscriber menampilkan pesan yang diterima secara real-time. Hasil menunjukkan bahwa proses publish dan subscribe berjalan dengan baik, di mana setiap pesan yang dikirim oleh publisher berhasil diterima oleh subscriber melalui broker MQTT.

4. Modifikasi Container upcall



Percobaan **Upcall** menggunakan mekanisme komunikasi berbasis socket TCP. Server (**servercall.py**) dijalankan pada port 4141 dan menunggu koneksi dari client. Server menyimpan pesan sebelumnya yang diterima, lalu menggabungkannya dengan pesan berikut menggunakan format:

<pesan1> love <pesan2>

Client (`clientcall.py`) mengirimkan pesan ke server. Jika client mengirim dua pesan berbeda secara berurutan, server akan menggabungkannya. Selain itu, client juga mendukung mode *double payload* di mana dua pesan bisa langsung dikirim sekaligus dengan pemisah `||`.

Pada pengujian, beberapa pesan dikirim oleh client:

```
riri → wanda  
wanda → lutfi  
lutfi → ivan  
ivan → riri  
riri → septian
```

Hasil keluaran dari server menunjukkan penggabungan sesuai format, misalnya:

```
Received upcall from server: riri love wanda  
Received upcall from server: wanda love lutfi  
Received upcall from server: lutfi love ivan  
...
```

Hasil ini membuktikan bahwa mekanisme upcall berjalan dengan baik. Server berhasil mengolah pesan client dan mengembalikan hasil gabungan sesuai yang diharapkan.