

Nama : Ambadha Dhafa Ganiesa Nugroho

NPM : 50421140

Kelas : 4IA10

Matkul : Pemrograman Jaringan

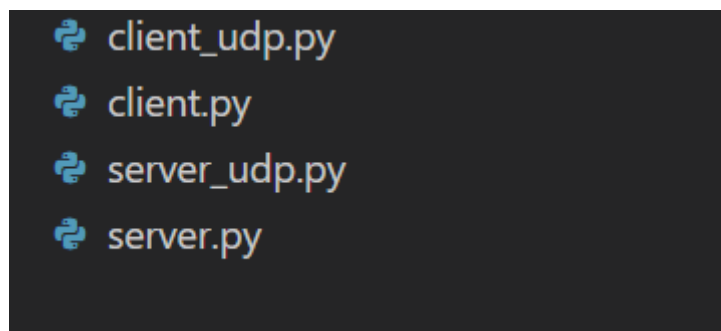
Socket Programming

SockCom (*Socket Communication*)

Program ini merupakan implementasi dari konsep **socket programming** menggunakan bahasa pemrograman Python. Sistem terdiri dari dua pasangan program client-server yang menggunakan protokol komunikasi yang berbeda, yaitu **TCP (Transmission Control Protocol)** dan **UDP (User Datagram Protocol)**.

Tujuan dari program ini adalah untuk menunjukkan bagaimana dua komputer (atau proses) dapat saling berkomunikasi dalam arsitektur **client-server**, baik dalam bentuk komunikasi yang andal (TCP) maupun ringan dan cepat (UDP).

Program



Server.py

```
server.py > ...
1  import socket
2
3  server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
4  server_socket.bind(('localhost', 9000))
5  server_socket.listen()
6  print('Listening on port 9000...')
7
8  while True:
9      client_socket, address = server_socket.accept()
10     print(f'Connection from {address} has been established!')
11     client_socket.send(bytes("welcome to the server!", "utf-8"))
12     client_socket.close()
```

- Membuat server socket dengan protokol SOCK_STREAM (TCP).
- Melakukan bind ke localhost dan port 9000.
- Menunggu koneksi dari client (listen()), lalu menerima koneksi (accept()).
- Mengirim pesan sambutan ke client, lalu menutup koneksi.
- **Karakteristik:** Andalan dalam koneksi stabil dan berurutan.

Client.py

```
client.py > ...
1 import socket
2
3 client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
4 client_socket.connect(('localhost', 9000))
5
6 message = client_socket.recv(1024)
7 print(message.decode("utf-8"))
```

- Membuat client socket dan melakukan koneksi ke localhost port 9000.
- Menerima pesan dari server dan mencetaknya ke layar.
- **Karakteristik:** Membuktikan komunikasi dua arah berbasis TCP.

Server_udp.py

```
server_udp.py > ...
1 import socket
2
3 server_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
4 server_socket.bind(('localhost', 9001))
5
6 print('Listening on port 9001...')
7 while True:
8     data, address = server_socket.recvfrom(1024)
9     print("Received message:", data.decode('utf-8'), 'from', address)
10    server_socket.sendto(data, address)
```

- Membuat server socket menggunakan SOCK_DGRAM (UDP).
- Melakukan bind ke localhost dan port 9001.
- Menunggu pesan dari client menggunakan `recvfrom()`.
- Mencetak pesan yang diterima dan mengirim balasan ke pengirim.
- **Karakteristik:** Ringan, cepat, tapi tanpa jaminan pengiriman.

Client_udp.py

```
client_udp.py > ...
1 import socket
2
3 client_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
4 message = "Hello, saya Ambadha Dhafa Ganiesa Nugroho 50421140!"
5 client_socket.sendto(message.encode("utf-8"), ('localhost', 9001))
6 data, address = client_socket.recvfrom(1024)
7 print(data.decode("utf-8"))
```

- Membuat client socket menggunakan protokol UDP.
- Mengirim pesan ke server UDP pada port 9001.
- Menerima balasan dari server dan mencetaknya ke layar.

Output

Server.py

```
TERMINAL
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\victo\OneDrive\Desktop\pjar> python server.py
Listening on port 9000...
Connection from ('127.0.0.1', 58415) has been established!
[]
```

Menerima koneksi TCP dari client lalu mengirimkan pesan sambutan sebelum menutup koneksi.

Client.py

```
TERMINAL
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\victo\OneDrive\Desktop\pjar> python client.py
welcome to the server!
PS C:\Users\victo\OneDrive\Desktop\pjar> []
```

Menghubungkan ke server TCP, menerima pesan, dan menampilkannya.

Server_udp.py

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

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\victo\OneDrive\Desktop\pjar> python server_udp.py
Listening on port 9001...
Received message: Hello, saya Ambadha Dhafa Ganiesa Nugroho 50421140! from ('127.0.0.1', 51120)
[]
```

Menerima pesan UDP dari client dan mengirimkan kembali pesan tersebut ke client

Client_udp.py

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

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\victo\OneDrive\Desktop\pjar> python client_udp.py
Hello, saya Ambadha Dhafa Ganiesa Nugroho 50421140!
```

Mengirim pesan UDP ke server dan menampilkan balasan dari server.

Cara Kerja

TCP

1. **Server** menunggu koneksi dari client.
2. **Client** terhubung ke server.
3. **Server** mengirim pesan ke client.
4. **Client** menerima dan menampilkan pesan.
5. Koneksi ditutup.

UDP

1. **Client** mengirim pesan ke server.
2. **Server** menerima pesan dari client.
3. **Server** membalas pesan ke client.
4. **Client** menerima dan menampilkan balasan.

Kesimpulan

Program ini berhasil mendemonstrasikan bagaimana socket digunakan dalam komunikasi antar proses secara langsung melalui dua protokol utama, yaitu TCP dan UDP. Melalui implementasi ini, mahasiswa dapat memahami perbedaan mendasar antara komunikasi berbasis koneksi (TCP) dan tanpa koneksi (UDP), cara kerja arsitektur client-server pada level jaringan dasar, serta mendapatkan pengalaman langsung dalam melakukan socket programming menggunakan bahasa Python. Tema utama dari program ini adalah Socket Programming, karena seluruh komunikasi dibangun langsung dengan modul socket, tanpa melibatkan framework atau protokol tingkat tinggi, sehingga memberikan pemahaman yang mendalam tentang mekanisme komunikasi jaringan dasar.