

LAPORAN PRAKTIKUM

{PEMOGRAMAN BERBASIS JARINGAN}

Disusun untuk Memenuhi Matakuliah Praktikum {Pemograman Berbasis Jaringan}
Dibimbing oleh { Pelsri Ramadar N.S. M.Kom }



Oleh:

Arisna Nova Surya Teguh Saputra

1123102115

PROGRAM STUDI S1 TEKNIK INFORMATIKA
SEKOLAH TINGGI ILMU KOMPUTER PGRI BANYUWANGI
2022

LEMBAR PENGESAHAN

LAPORAN PRAKTIKUM

Matakuliah : Pemograman Berbasis Jaringan

Oleh : Arisna Nova Surya Teguh Saputra

NIM : 1123102115

Telah disahkan pada

Hari : -

Tanggal : dd MM yyyy

Mengetahui/ Menyetujui :

Pelsri Ramadar N.S M.Kom

Pelsri Ramadar N.S M.Kom

NIDN.0704068408

KATA PENGANTAR

Laporan ini disusun untuk memenuhi salah satu tugas yang diberikan dalam mata kuliah terkait jaringan komputer. Tujuan utama dari laporan ini adalah untuk memberikan pemahaman kepada mahasiswa tentang konsep socket berbasis User Datagram Protocol (UDP) dan implementasinya. Selain itu, laporan ini juga mengupas manipulasi data serta pengukuran performa pengiriman data, yang merupakan aspek penting dalam pengembangan teknologi berbasis jaringan.

Dalam proses penyelesaian laporan ini, kami memperoleh dukungan, arahan, dan masukan dari berbagai pihak. Oleh karena itu, kami mengucapkan terima kasih kepada dosen pembimbing atas bimbingannya yang telah memberikan dukungan dan pembelajaran pada materi ini.

Saya menyadari bahwa laporan ini masih memiliki banyak kekurangan. Oleh sebab itu, kami dengan senang hati menerima kritik dan saran yang membangun untuk perbaikan di masa yang akan datang. Semoga laporan ini dapat memberikan manfaat, baik bagi kami sebagai penyusun maupun bagi pembaca lainnya dalam memperdalam ilmu di bidang jaringan komputer.

DAFTAR ISI

Daftar Isi	Halaman
LEMBAR PENGESAHAN LAPORAN PRAKTIKUM	ii
KATA PENGANTAR	iii
DAFTAR ISI	iv
DAFTAR GAMBAR	v
Modul Praktikum 1	1
1.1 Capaian Praktikum Modul 1	1
1.2 Indikator Capaian	1
Dan seterusnya	

DAFTAR GAMBAR

Daftar Gambar	Halaman
1.1 Gambar Flowchart Modul 1.....	2
1.2 Gambar Hasil Pengujian Modul 1.....	5
2.1 Gambar Pseudocode Modul 2.....	7
Dan seterusnya	

DAFTAR TABEL

Daftar Tabel	Halaman
1.1 Tabel Hasil Pengujian Modul 1	5
2.1 Tabel Hasil Pengujian Modul 2	8
Dan seterusnya	

BAB 1

*JUDUL PRAKTIKUM MODUL 1

1.1 Capaian Praktikum Pertemuan 1

Mahasiswa memahami dan mampu mengimplementasikan dari socket UDP serta memanipulasi data. Mahasiswa juga mampu memahami tentang threading dan mampu mengimplementasikan.

1.2 Indikator Capaian

Mahasiswa melakukan percobaan pemograman thread untuk server serta melakukan pada sisi client agar dapat mengirimkan pesan kepada server.

1.3 Landasan Teori

Pemograman berbasis jaringan pada materi ini adalah mengenai pembahasan tentang threading (pengontrol aliran program).

1.4 Pelaksanaan Praktikum

1.4.1 Percobaan Pertama

Pada percobaan kali ini mahasiswa melakukan percobaan agar sisi server dan client dapat terkoneksi serta sisi client bisa mengirimkan pesan terhadap server.

a. Script / Setting Program

```
Server :  
  
import socket  
  
import threading  
  
import time  
  
  
socketClient = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)  
  
nomer = 0
```

```
socketClient.bind(("127.0.0.1", 12345))
```

```
print("Server menunggu...")
```

```
def func(number):
```

```
    x = socketClient.recvfrom(1024)
```

```
    addrs = x[1][0]
```

```
    msg = x[0].decode()
```

```
    no = str(number)
```

```
    print(str("User-"+no) + " : " + msg)
```

```
    time.sleep(int(msg))
```

```
    print(str("User-"+no) + " : " + msg)
```

```
while True:
```

```
    nomer += 1
```

```
    t = threading.Thread(target=func, args=(nomer, ))
```

```
    t.start()
```

```
Client:
```

```
import socket
```

```
import threading
```

```
import time
```

```
def send_messages(socketClient):
```

```
    while True:
```

```
        angka = input('Masukkan angka (atau "exit" untuk keluar): ')
```

```
        if angka.lower() == 'exit':
```

```
            print("Client disconnected.")
```

```
            socketClient.close()
```

```
            break
```

```
    try:
```



```
int(angka)

b = angka.encode('utf-8')

socketClient.send(b)

time.sleep(1)

except ValueError:

    print("Input harus berupa angka. Coba lagi.")

socketClient = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)

server_address = ("127.0.0.1", 12345)

socketClient.connect(server_address)

thread_send = threading.Thread(target=send_messages, args=(socketClient,))

thread_send.start()

thread_send.join()
```

b. Langkah Uji Coba

- Langkah pertama mahasiswa melakukan copy dan paste codingan yang telah dosen berikan.
- Langkah kedua mahasiswa merubah codingan sesuai apa yang diinginkan oleh dosen pengampu.

c. Hasil Uji Coba

Tampilkan hasil pengujian yang telah dilakukan dalam percobaan ini.

```

PS D:\KULIAH\semester 3\praktikum
n.exe "d:/KULIAH/semester 3/prakti
Server menunggu...
User-1 : 1
User-1 : 1
User-2 : 2
User-3 : 3
User-2 : 2
User-4 : 4
User-3 : 3
User-5 : 5
User-4 : 4
User-5 : 5

```

```

PS D:\KULIAH\semester 3\praktikum pemograman berbasis jaringan> & C:/U
n.exe "d:/KULIAH/semester 3/praktikum pemograman berbasis jaringan/cli
Masukkan angka (atau "exit" untuk keluar): 1
Masukkan angka (atau "exit" untuk keluar): 2
Masukkan angka (atau "exit" untuk keluar): 3
Masukkan angka (atau "exit" untuk keluar): 4
Masukkan angka (atau "exit" untuk keluar): 5
Masukkan angka (atau "exit" untuk keluar): exit
Client disconnected.
PS D:\KULIAH\semester 3\praktikum pemograman berbasis jaringan>

```

d. Analisa Hasil

Pada percobaan ini berhasil dilakukan dengan sisi client mengirimkan pesan kepada server.

1.5 Kesimpulan

1.5.1 Kesimpulan Percobaan 1

Server UDP Multithreaded

- Server menggunakan socket UDP untuk menerima data dari client.
- Setiap koneksi client ditangani dalam thread terpisah untuk memungkinkan penerimaan dan pemrosesan pesan secara paralel.

- Server menampilkan pesan yang diterima dari client bersama dengan nomor identifikasi client (*User-n*).
- Pesan yang diterima diproses dengan jeda waktu sesuai isi pesan, menunjukkan kemampuan server untuk memproses data dengan logika tambahan, seperti delay berbasis input.

Client UDP Multithreaded

- Client mengirimkan pesan berupa angka ke server menggunakan protokol UDP.
- Terdapat mekanisme input dari pengguna untuk memasukkan angka yang akan dikirim, dengan validasi agar hanya angka yang dapat diterima.
- Thread digunakan untuk menjalankan proses pengiriman pesan secara terus-menerus, memungkinkan client berfungsi secara asinkron terhadap server.
- Client dapat diakhiri dengan mengetikkan "exit", yang menutup socket dan menghentikan program.

Keseluruhan Sistem

- Sistem ini menunjukkan bagaimana komunikasi berbasis UDP dapat dilakukan antara server dan client dengan pendekatan multithreading.
- Protokol UDP yang bersifat *connectionless* memungkinkan pengiriman data tanpa perlu membangun koneksi yang persisten, sehingga ringan dan cepat.
- Namun, tidak ada mekanisme pengelolaan error atau pengiriman balasan, yang berarti komunikasi hanya satu arah dari client ke server. Hal ini cukup untuk skenario sederhana tetapi kurang ideal untuk kebutuhan yang memerlukan keandalan lebih tinggi

Mengetahui:

Pelsri Ramadar N.S M.Kom

Pelsri Ramadar N.S M.Kom

NIDN.0704068408

***NB. Halaman ini dibuat sebanyak pertemuan yang diberikan**

BAB **

PENUTUP

****.1 Kesimpulan**

Implementasi ini menggambarkan contoh sederhana dan efektif dari komunikasi berbasis UDP dengan multithreading, memungkinkan pengiriman dan pemrosesan pesan secara paralel. Untuk keperluan yang lebih kompleks, perlu ditambahkan fitur seperti pengelolaan error, konfirmasi penerimaan pesan, atau validasi koneksi untuk meningkatkan keandalan sistem.

****.2 Saran**

-

DAFTAR PUSTAKA

Tuliskan rujukan yang anda gunakan baik website maupun buku seperti contoh dibawah.

1. Tim Asisten Dosen. 2014. Modul 1 Pengenalan Sistem Operasi, Ide Visual C++, Dan Algoritma Pemrograman. Malang: Universitas Negeri Malang.
2. Program Konversi Suhu (online)
<http://bondanoky.blogspot.com/2012/10/program-konversi-suhu-c.html>. Di akses 8 September.

LAMPIRAN

Berisikan syntax atau gambar yang dibutuhkan dalam tiap pertemuan praktikum.

```
PS D:\KULIAH\semester 3\praktikum
n.exe "d:/KULIAH/semester 3/prakti
Server menunggu...
User-1 : 1
User-1 : 1
User-2 : 2
User-3 : 3
User-2 : 2
User-4 : 4
User-3 : 3
User-5 : 5
User-4 : 4
User-5 : 5
```

```
PS D:\KULIAH\semester 3\praktikum pemograman berbasis jaringan> & C:/U
n.exe "d:/KULIAH/semester 3/praktikum pemograman berbasis jaringan/cli
Masukkan angka (atau "exit" untuk keluar): 1
Masukkan angka (atau "exit" untuk keluar): 2
Masukkan angka (atau "exit" untuk keluar): 3
Masukkan angka (atau "exit" untuk keluar): 4
Masukkan angka (atau "exit" untuk keluar): 5
Masukkan angka (atau "exit" untuk keluar): exit
Client disconnected.
PS D:\KULIAH\semester 3\praktikum pemograman berbasis jaringan> █
```