

JARINGAN KOMPUTER
LAPORAN TUGAS BESAR
KELAS : IF-45-09



ANGGOTA KELOMPOK 13:

Brilliant Jordan	1301213318
Akmal Sidki Razaka	1301210547
Adrian Yoris Mbake Woka	1301210484

FAKULTAS INFORMATIKA
UNIVERSITAS TELKOM
BANDUNG
2023

DAFTAR ISI

Daftar isi.....	2
Latar Belakang	3
Batasan Masalah	4
Sistem yang dibangun	5
Hasil Program	8
Kesimpulan	10

LATAR BELAKANG

Web server adalah perangkat keras atau perangkat lunak yang menjalankan aplikasi server yang bertugas menyimpan, mengelola, dan menyediakan konten serta layanan melalui internet kepada pengguna. Secara umum, web server melayani permintaan pengguna yang dikirim melalui browser web untuk mengakses halaman web, berkas, atau sumber daya lainnya. Web server berperan dalam memproses permintaan yang diterima, mengirimkan respon yang sesuai kepada pengguna, dan mengelola berbagai protokol komunikasi seperti HTTP (Hypertext Transfer Protocol) dan HTTPS (HTTP Secure) untuk menjaga keamanan data yang ditransfer antara server dan pengguna.

TCP socket programming adalah pendekatan atau metode pemrograman yang digunakan untuk membuat aplikasi jaringan menggunakan protokol TCP (Transmission Control Protocol). Dalam TCP socket programming, aplikasi dapat membuat koneksi TCP ke perangkat jaringan lainnya untuk mengirim dan menerima data.

Tujuan dari tugas besar ini adalah membuat sebuah web server dengan metode TCP socket programming. Yang dimana, web server ini akan menggunakan protokol TCP dan Bahasa pemrograman Python. Web server yang akan dibangun diharapkan bisa menerima permintaan HTTP GET dan memarsing HTTP REQUEST yang dikirimkan oleh browser dan tentunya web server bisa dapat mencari dan mengambil file yang diminta oleh client. Manfaat dari tugas besar ini adalah untuk memberi kita pemahaman mengenai web server dan cara kerja sebuah web server serta untuk kedepannya bisa digunakan sebagai publikasi web site kelompok maupun pribadi.

BATASAN MASALAH

Bedasarkan spesifikasi tugas besar ini, ada beberapa spesifikasi web server yang harus diimplementasikan untuk memenuhi kebutuhan web server, antara lain yaitu:

1. Implementasi pembuatan TCP socket dan mengaitkannya ke alamat dan port tertentu
2. Program web server dapat menerima dan memarsing HTTP request yang dikirimkan
3. Web server dapat mencari dan mengambil file (dari file system) yang diminta oleh client
4. Web server dapat membuat HTTP response message yang terdiri dari header dan konten file yang diminta
5. Web server dapat mengirimkan response message yang sudah dibuat ke browser (client) dan dapat ditampilkan dengan benar di sisi client
6. Jika file yang diminta oleh client tidak tersedia, web server dapat mengirimkan pesan“404 Not Found” dan dapat ditampilkan dengan benar di sisi client.

SISTEM YANG DIBANGUN

3.1 Modul yang Digunakan

```
1  from http.server import HTTPServer, BaseHTTPRequestHandler
2  import os
3  import mimetypes
4  import socket
```

Pada Tugas ini, kami menggunakan 4 modul untuk membantu kami dalam pembuatan web server, yakni `http.server`, `os`, `mimetypes`, dan `socket`. Adapun masing-masing penggunaannya yakni:

1. **`http.server`** untuk membuat HTTP Server yang memudahkan pengelolaan permintaan (request), serta menyediakan fitur-fitur server HTTP seperti parsing permintaan HTTP
2. **`os`** untuk berinteraksi dengandirektori file
3. **`mimetypes`** untuk mengenali tipe konten yang dicari berdasarkan dengan ekstensi file
4. **`socket`** untuk berinteraksi dengan client melewati socket, dimana pada tugas kali ini menggunakan TCP Socket Programming.

3.2 Pembuatan Socket

```
HOST = 'localhost'
PORT = 8000
server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
try:
    server_socket.bind((HOST, PORT))
    server_socket.listen()
    print(f'Server is running on {HOST}:{PORT}')
    while True:
        client_socket, client_address = server_socket.accept()
        handler = echoHandler(client_socket, client_address, server_socket)
except Exception as e:
    print(f"Error: {e}")
finally:
    server_socket.close()
```

Menginisialisasi host dengan *localhost*, kemudian menginisialisasi nomor port dengan 8000, lalu membuat *socket tcp*. Kemudian, saat dibuka, socket kemudian mengikat host dan port, lalu siap untuk menerima request dari client dengan *server_socket.listen()*. Lalu saat ada request yang masuk, *server_socket* akan menerima *socket* dan *address* dari client, lalu request tersebut

diproses oleh sistem. Exception tersebut merupakan proteksi bilamana terjadi error, dan finally *server_socket* ditutup.

3.3 Melakukan penanganan (handle) pada request

```
class echoHandler(BaseHTTPRequestHandler):
    def do_GET(self):
        if self.path == '/':
            self.path = '/index.html'
        elif self.path == '/pdf':
            self.path = '/TCPSocketProgramming.pdf'
        try:
            file_path = os.path.abspath(self.path[1:])
            if os.path.isfile(file_path):
                with open(file_path, 'rb') as file:
                    content = file.read()
                self.send_response(200)
                content_type, _ = mimetypes.guess_type(file_path)
                self.send_header('Content-type', content_type)
            else:
                raise FileNotFoundError
        except FileNotFoundError:
            content = b"<html><body><h1>404</h1><h3>File Not Found</h3></body></html>"
            self.send_response(404)
            self.send_header('Content-type', 'text/html')

        self.send_header('Content-length', len(content))
        self.end_headers()
        self.wfile.write(content)
```

Yoris - Personal > Documents > Kuliah > TCP

Name	Status	Date modified	Type	Size
.vscode	✓	5/7/2023 11:47 AM	File folder	
index	✓	5/7/2023 10:54 PM	Microsoft Edge HTM...	1 KB
TCPSocketProgramming	✓	5/28/2023 8:16 PM	Microsoft Edge PDF ...	40 KB
webserver	✓	5/28/2023 8:17 PM	Python Source File	2 KB

Ada 2 file yang dapat diakses, yakni **index.html** dan **TCPSocketProgramming.pdf**. Kedua itu ditangani dalam class `echoHandler` (penggunaan modul `http.server - BaseHTTPRequestHandler`), lebih tepatnya pada fungsi `do_get`, yang merupakan fungsi yang akan dipanggil saat request diterima. Saat pathnya adalah '/', maka file yang diakses adalah `index.html` sedangkan saat pathnya diubah menjadi 'pdf', maka yang diakses adalah `TCPSocketProgramming.pdf`. Pada command prompt server, terekam jejak akses, dimana saat file terdapat pada direktori dan pathnya adalah '/', akan terekam **127.0.0.1 - - [28/May/2023**

20:30:00] "GET / HTTP/1.1" 200 - ; Sedangkan saat path yang diakses adalah '/pdf', maka akan terekam 127.0.0.1 - - [28/May/2023 20:31:04] "GET /pdf HTTP/1.1" 200 -. Dimana 200 adalah balasan saat tidak ada error dan file ditemukan pada direktori.

```
C:\Windows\System32\cmd.e X + v
Microsoft Windows [Version 10.0.22621.1702]
(c) Microsoft Corporation. All rights reserved.

C:\Users\oiswo\OneDrive\Documents\Kuliah\TCP>py webserver.py
Server is running on localhost:8000
127.0.0.1 - - [28/May/2023 20:30:00] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [28/May/2023 20:31:04] "GET /pdf HTTP/1.1" 200 -
127.0.0.1 - - [28/May/2023 20:31:05] "GET /pdf HTTP/1.1" 200 -
|
```

Kemudian penggunaan `os`, seperti `os.path.abspath(self.path[1:])` dan `os.path.isfile(file.path)`, merupakan operasi pada direktori server. Penggunaan `os` ini untuk mengakses direktori, membuka direktori dan membuka file, yang kemudian dibantu oleh `mimetypes` untuk menentukan file mana yang akan dibaca berdasarkan ekstensinya. Jika berkas ditemukan, maka akan file akan dibaca dan dikirim respon kode status 200 (OK). Pada program juga terdapat header *Content-length* yang menyatakan panjang konten, kemudian diakhiri dengan `self.end_headers()`. Terakhir, `self.wfile_write(content)` untuk menuliskan konten ke koneksi socket.

3.4 Proteksi ketika error atau file tidak ditemukan

```
else:
    raise FileNotFoundError
except FileNotFoundError:
    content = b"<html><body><h1>404</h1><h3>File Not Found</h3></body></html>"
    self.send_response(404)
    self.send_header('Content-type', 'text/html')
```

Pada program terdapat proteksi ketika file yang diakses tidak ada pada direktori, yakni dengan mengirimkan respon File not found dengan kode status 404.

```
C:\Windows\System32\cmd.e X + v
Microsoft Windows [Version 10.0.22621.1702]
(c) Microsoft Corporation. All rights reserved.

C:\Users\oiswo\OneDrive\Documents\Kuliah\TCP>py webserver.py
Server is running on localhost:8000
127.0.0.1 - - [28/May/2023 21:00:22] "GET /asd HTTP/1.1" 404 -
|
```

HASIL PROGRAM



Adrian Yoris Mbake Woka

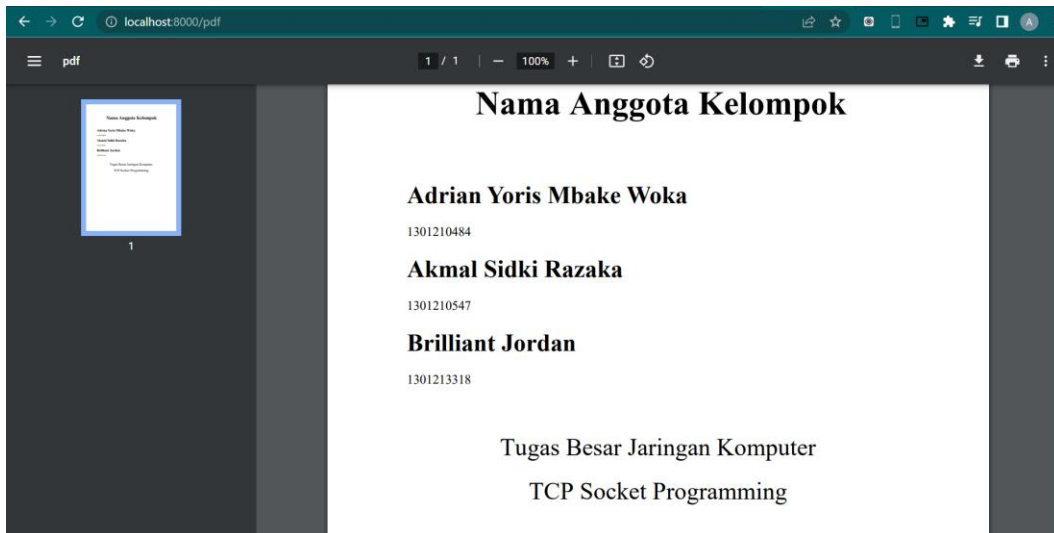
1301210484

Akmal Sidki Razaka

1301210547

Brilliant Jordan

1301213318





404

File Not Found

KESIMPULAN

Berdasarkan hasil pengerjaan tugas besar ini, kami bertiga dapat menyimpulkan bahwa:

1. Membuat TCP Socket: Pertama, kita perlu membuat socket menggunakan protokol TCP. Socket ini akan digunakan untuk berkomunikasi antara server dan klien.
2. Mengaitkan Socket dengan Alamat dan Port: Selanjutnya, kita mengaitkan socket dengan alamat IP dan nomor port tertentu. Ini akan menentukan di mana server akan menerima koneksi masuk.
3. Menunggu Koneksi Masuk: Setelah socket terikat, server akan memanggil metode `listen()` untuk menunggu koneksi masuk. Dalam langkah ini, server siap menerima koneksi dari klien.
4. Menerima Koneksi dan Menghubungkan dengan Klien: Ketika ada koneksi masuk, server akan memanggil metode `accept()` untuk menerima koneksi tersebut. Hal ini akan mengembalikan objek socket baru yang akan digunakan untuk berkomunikasi dengan klien.
5. Mengelola Koneksi dengan Klien: Setelah koneksi dengan klien terhubung, server akan membuat thread baru untuk menangani koneksi tersebut. Thread ini akan menjalankan fungsi khusus yang bertanggung jawab untuk mengelola koneksi dengan klien.
6. Setelah mengelola koneksi dengan klien, server akan kembali ke langkah 4 untuk menerima koneksi masuk dari klien berikutnya. Proses ini berulang terus menerus hingga server dapat melayani banyak koneksi secara bersamaan.

Maka web server dapat menerima dan mengelola koneksi dari berbagai klien. Ini memungkinkan server untuk merespons permintaan dari klien dan memberikan konten web yang sesuai dengan permintaan tersebut.

Kontribusi Anggota Kelompok:

Brilliant Jordan (1301213318) Laporan dan Sistem yang dibangun

Akmal Sidki Razaka (1301210547) Laporan dan Sistem yang dibangun

Adrian Yoris Mbake Woka (1301210484) Laporan dan Sistem yang dibangun