**LAPORAN EVALUASI TENGAH SEMESTER**

**PEMROGRAMAN JARINGAN D**



Oleh :

M Yusuf Mukharom 05111740000051

**DEPARTEMEN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI ELEKTRO DAN INFORMATIKA CERDAS**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**SURABAYA**

**2020**

1. **Penjelasan Singkat**
2. Deskripsi

Blablabla

1. Daftar Fitur

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Keterangan Fitur | Berhasil | Tidak Berhasil | Belum Dikerjakan |
| 1. | Chat | ✔ |  |  |
| 2. | LIST | ✔ |  |  |
| 3. | PWD | ✔ |  |  |
| 4. | CD | ✔ |  |  |
| 5. | MKDIR | ✔ |  |  |
| 6. | SENDALL |  | ✔ |  |
| 7. | DOWNZIP | ✔ |  |  |

1. **Source Code dan Dokumentasi**
2. Server

Tugas utama server disini adalah mengirimkan pesan dan file ke seluruh user

# import library utama

import socket

import time

import select

import sys

def send\_file(filename):

    f = open(filename, "rb")

    data = f.read(1024)

    while(data):

        if( server\_socket.sendto( data, client ) ):

            data = f.read(1024)

            time.sleep(0.02) # Give receiver a bit time to save

    f.close()

def recv\_file(filename):

    f = open(filename, 'wb')

    while True:

        ready = select.select([server\_socket], [], [], 3)

        if ready[0]:

            data, addr = server\_socket.recvfrom(1024)

            f.write(data)

        else:

            f.close()

            break

# membuat socket server udp

server\_address = ('localhost', 5000)

server\_socket = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

server\_socket.setsockopt(socket.SOL\_SOCKET, socket.SO\_REUSEADDR,1)

server\_socket.bind(server\_address)

# set yg akan menyimpan client yg terhubung dengan server

set\_client\_addr = set(())

while True:

    try:

        # recv data yg dikirim client

        data, client\_address = server\_socket.recvfrom(1024)

        data\_msg = data.decode()

        # jika kata kunci adalah connect (client baru connect) maka akan memasukkan client ke set

        # dan mengirim pesan connected... ke client

        if data\_msg.split()[2] == "connect":

            server\_socket.sendto("connected...".encode(), client\_address)

            set\_client\_addr.add(client\_address)

            print(str(client\_address) + ' join the chat')

            continue

        # jika kata kunci adalah close (client left chat) maka akan menghapus client pada set

        elif data\_msg.split()[2] == "closed":

            set\_client\_addr.discard(client\_address)

            print(str(client\_address) + ' left the chat')

            continue

        # jika kata kunci SEND (client kirim file) maka akan menyimpan file pada server

        elif data\_msg.split()[2] == "SENDALL":

            recv\_file(data\_msg.split()[3])

        # broadcast message kesemua user yg ada dalam set

        for client in set\_client\_addr:

            msg\_usr = data\_msg.split()[:2]

            user = str(client).split()

            # akan mengirim pesan kecuali user yg mengirim

            if msg\_usr != user:

                server\_socket.sendto(data\_msg.encode(), client)

                print('message : ' + str(data\_msg) + ' to : ' + str(client))

                # read file dan mengirimkan ke client"

                if data\_msg.split()[2] == "SENDALL":

                    send\_file(data\_msg.split()[3])

    except KeyboardInterrupt:

        print(str(server\_socket.getsockname()) + ' has down')

        server\_socket.close()

        sys.exit(0)

        break

1. Client

Client akan terhubung dengan ftp. Jika client mengirimkan perintah-perintah tertentu maka dia tidak akan mengirimkan message ke server.

# import library yg dibutuhkan

import socket

import time

import sys

import threading

import select

import os

import zipfile

from ftplib import FTP

import shutil

# fungsi receive file

def recv\_file(fname):

    f = open(fname, 'wb')

    while True:

        ready = select.select([client\_socket], [], [], 3)

        if ready[0]:

            data, addr = client\_socket.recvfrom(1024)

            f.write(data)

        else:

            f.close()

            break

#fungsi send file

def send\_file(fname):

    f = open(fname, "rb")

    data = f.read(1024)

    while(data):

        if( client\_socket.send( data ) ):

            data = f.read(1024)

            time.sleep(0.02) # Give receiver a bit time to save

    f.close()

# fungsi untuk mencetak message yg dikirim server

# fungsi ini akan dijalankan oleh thread

def print\_msg():

    while True:

        recv\_msg = client\_socket.recv(1024).decode()

        if recv\_msg.split()[2] == "SENDALL":

            recv\_file(recv\_msg.split()[3])

            continue

        print(str(recv\_msg).rjust(80))

# buat socket client dan menghubungkan ke server

server\_address = ('localhost', 5000)

client\_socket = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

client\_socket.connect(server\_address)

if \_\_name\_\_ == "\_\_main\_\_":

    print('Input user ftp: ')

    user = input()

    print('Input password ftp: ')

    pswd = input()

    print('Input IP FTP: ')

    ipftp = input()

    f = FTP(ipftp)

    f.login(user, pswd)

    act1 = 'LIST'

    act2 = 'PWD'

    act3 = 'CD'

    act4 = 'MKDIR'

    act5 = 'SENDALL'

    act6 = 'DOWNZIP'

    # thread untuk print message

    t1 = threading.Thread(target=print\_msg)

    # mengirimkan connect jika awal konek server

    message = str(client\_socket.getsockname()) + ' ' + 'connect'

    client\_socket.send(message.encode())

    recv\_msg = client\_socket.recv(1024).decode()

    print(recv\_msg)

    t1.start()

    while True:

        try:

            # read inputan pada terminal

            message = sys.stdin.readline()

            message = str(client\_socket.getsockname()) + ' ' + message

            # client\_socket.send(message.encode()) loloolo

            if message.split()[2] == "SENDALL":

                inpt2 = message.split()[3]

                f.retrbinary("RETR " + inpt2, open(inpt2, 'wb').write)

                send\_file(inpt2)

                client\_socket.send(message.encode())

            else :

                if message.split()[2].count(act1) == 1:

                    names = f.nlst()                                        #LIST DIRECTORY

                    print('List : ' + str(names))

                elif message.split()[2].count(act2) == 1:

                    print('PWD :' + f.pwd())           #PRESENT WORK DIRECTORY

                # elif message.split()[2].count(act2) == 1:

                #     inpt2 = message.split()[2].replace('RETR ', '')                        #DOWNLOAD

                #     f.retrbinary("RETR " + inpt2, open(inpt2, 'wb').write)

                # elif message.split()[2].count(act3) == 1:

                #     inpt2 = message.split()[2].replace('STOR ', '')

                #     f.storbinary('STOR ' + inpt2, open(inpt2, 'rb'))              #UPLOAD

                elif message.split()[2].count(act3) == 1:

                    inpt2 = message.split()[3]

                    f.cwd(inpt2)                                                   #CD

                elif message.split()[2].count(act4) == 1:

                    inpt2 = message.split()[3]

                    f.mkd(inpt2)                                               #BUAT DIRECTORY

                elif message.split()[2].count(act5) == 1:

                    print('PWD :' + f.pwd())           #PRESENT WORK DIRECTORY

                elif message.split()[2].count(act6) == 1:                               #DOWNPRESS (FILE DIRECTORY DIUBAH SECARA MANUAL)

                    inpt2 = message.split()[3]

                    shutil.make\_archive('./filezilla/' + inpt2, 'zip', './filezilla/' + inpt2)

                    fz = inpt2 + '.zip'

                    f.retrbinary("RETR " + fz, open(fz, 'wb').write)

                    f.delete(fz)

                else:

                    print('Salah Command')

                continue

            client\_socket.send(message.encode())

            # jika kata kunci SEND maka akan melakukan fungsi send file ke server

            # if message.split()[2] == "SENDALL":

            #     send\_file(message.split()[3])

        except KeyboardInterrupt:

            print('you left the chat')

            message = str(client\_socket.getsockname()) + ' ' + 'closed'

            client\_socket.send(message.encode())

            client\_socket.close()

            sys.exit(0)

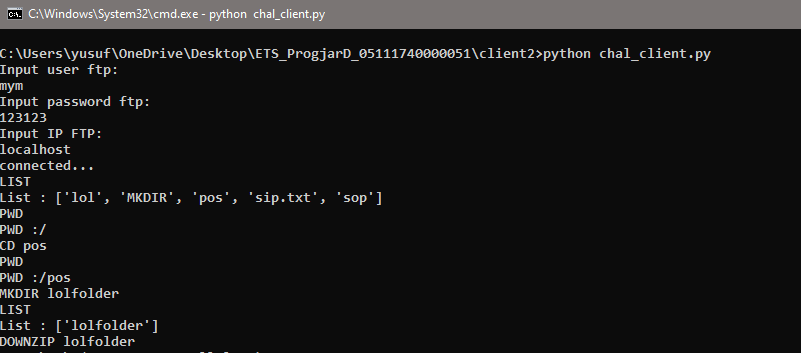
            break

1. **Teknis Pengoperasian**

Berikut merupakan langkah-langkah dalam pengoperasian program diatas;

1. Jalankan file server
2. Jalankan file client

Contoh perintah client



1. Chat sudah berhasil

