

DOKUMENTASI PEMROGRAMAN JARINGAN

PROTOKOL MULTICHATTING



KELAS B

David Victor Giandly

5113100033

Asisten:

Bahrul Halimi

Dosen:

Royyana Muslim I.

INSTITUT TEKNOLOGI SEPULUH NOPEMBER SURABAYA

2015

Program ini adalah suatu program chatting yang bisa digunakan 2 dan atau lebih client. Program chatting ini menggunakan bahasa python. Program ini memiliki 4 protokol yang terdiri dari login, send, sendall, list.

Pertama akan menjelaskan mengenai login.

Login

```
#login
if temp1[0]=="login" :
    log_in(sock, str(temp1[1]))
def log_in (sock, user):
    g = 0
    f = 0
    for name in NAME_LIST:
        if name == user:
            g = 1
        if name == sock:
            f = 1
    if f==1:
        send_msg(sock, "You already has a username\n")
    elif g==1:
        send_msg(sock, "Username already exist\n")
    else:
        #masukkan data user ke array
        NAME_LIST.append(sock)
        NAME_LIST.append(user)
        send_msg(sock, "Login success\n")
chat_server()
```

Dalam code diatas saya membuat sebuah fungsi untuk login.

Contoh inputan untuk login adalah sebagai berikut :

Login(spasi)namauser → Login David

Jika seperti ini yang keluar adalah **Login Success**.

Setelah login success maka secara otomatis alamat ip yang ada akan masuk kedalam sock dan namausernya dimasukkan ke user.

Dan jika sudah login dan akan login lagi maka yang keluar adalah "Username already exist"

Fungsi login disini adalah fungsi utama dalam chatting ini karena tanpa melakukan login maka tidak dapat melakukan chatting dan protokol login ini digunakan dalam protokol send.

Kemudian ke protokol ke 2 yaitu Send

Send

```
#send
elif temp1[0]=="send" :
    logged = 0
    user = ""
    for x in range (len(NAME_LIST)):
        if NAME_LIST[x]==sock:
            logged=1
            user=NAME_LIST[x+1]
    if logged==0:
        send_msg(sock, "Please login first\n")
    else:
        temp2=""
        for x in range (len(temp1)):
            if x>1:
                if not temp2:
                    temp2+=str(temp1[x])
                else:
                    temp2+=" "
                    temp2+=str(temp1[x])
        for x in range (len(NAME_LIST)):
            if NAME_LIST[x]==temp1[1]:
                send_msg(NAME_LIST[x-1], "["+user+"] : "+temp2+"\n")
```

Protokol send ini digunakan untuk chatting antara 2 user.

Sebelum melakukan chatting antara 2 orang maka code diatas akan mengecek untuk pertama kali apakah user tersebut sudah login atau belum. Jika belum maka akan keluar kata "Please login first". Jika sudah maka cara melakukan chatting dengan 1 user yang lain adalah sebagai berikut :

Send(spasi)username_yang_ingin_dichat(spasi)pesan → Send dia Haloo

Jadi kata Haloo akan dikirim server hanya kepada alamat yang dituju dan sedangkan alamat yang dituju sudah diganti dengan nama user agar client atau pengguna tidak perlu bingung untuk mengetahui alamat user yang ingin dituju.

Kemudian ke protokol ke 3 yaitu Sendall

Sendall

```
#sendall
elif temp1[0]=="sendall" :

    logged = 0
    user = ""
    for x in range (len(NAME_LIST)):
        if NAME_LIST[x]==sock:
            logged=1
            user=NAME_LIST[x+1]

    if logged==0:
        send_msg(sock, "Please login first\n")

    else:
        temp2=""
        for x in range(len(temp1)):
            if x!=0:
                if not temp2:
                    temp2=str(temp1[x])
                else:
                    temp2+=" "
                    temp2+=temp1[x]
        broadcast(server_socket, sock, "["+user+"] : "+temp2+"\n")
```

Dalam code diatas hampir sama dengan protokol Send sebelumnya hanya yang beda dalam protokol ini pada cara memasukkan inputannya. Sendall ini semacam Broadcast Message jadi pesan yang diketikkan oleh satu user akan dikirimkan kepada semua user yang sedang online atau sedang terkoneksi dengan server. Format inputan ini adalah

Sendall(spasi)pesan_yang _disampaikan → Sendall Halo semua. Saya David

Maka pesan “Halo semua. Saya David” akan terkirim ke semua user yang sudah login dan yang masih terkoneksi dengan server.

Kemudian ke protokol ke 4 yaitu list

List

```
..
#lihat list
elif temp1[0]=="list" :
    logged = 0
    for x in range (len(NAME_LIST)):
        if NAME_LIST[x]==sock:
            logged=1

    if logged==0:
        send_msg(sock, "Please login first\n")

    else:
        temp2=""
        for x in range (len(NAME_LIST)):
            if x%2==1:
                temp2+=" "
                temp2+=str(NAME_LIST[x])
        send_msg(sock, "[List_User] : "+temp2+"\n")
else:
    print ('Invalid Command')
..
```

Protokol yang terakhir ini adalah protokol list yang digunakan untuk mengecek user yang sedang online atau terkoneksi dengan server.

Contoh inputan pada protokol ini hanya mengetikkan kata “list” maka akan terlihat nama – nama user yang sedang online atau terkoneksi dengan server.

Jika belum login maka tidak dapat melihat list nama user yang sedang login dan akan muncul kalimat “Please login First”

Dan apabila jika salah menginputkan kata maka yang muncul adalah “Invalid Command”