

**Tugas Kelompok Pemrograman Jaringan Membuat Group Chat
Menggunakan Socket**



Abdur Rachman Wahed	05111840000004
Kresna Adhi Pramana	05111840000072
Excel Deo Cornelius	05111840000117

KELAS:

Pemrograman Jaringan D

Departemen Teknik Informatika

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember (ITS) Surabaya

2021

Tugas

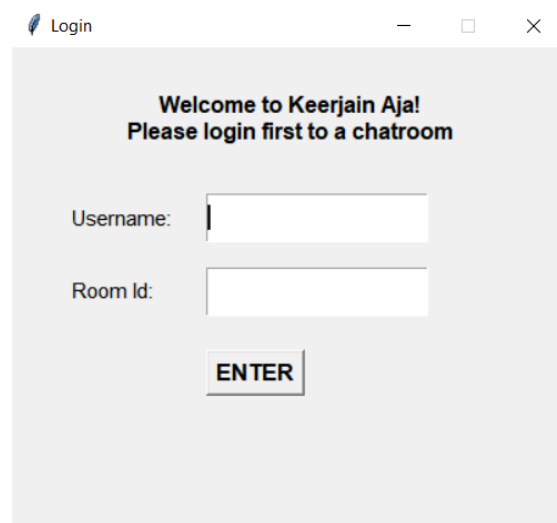
1. Tambahkan kemampuan
 - Graphical User Interface untuk chat client
 - Pengiriman file dari user ke user
 - Group message
2. Dokumentasi
 - Buatlah dokumentasi protokolnya (semacam pada progjar4a) disubmit di masing-masing MyITS classroom
 - Source code tugas ini harap di letakkan di repository git masing-masing anggota kelompok
 - Laporan menyebutkan member kelompok dan tugasnya

Jawaban

1. Graphical User Interface untuk chat client

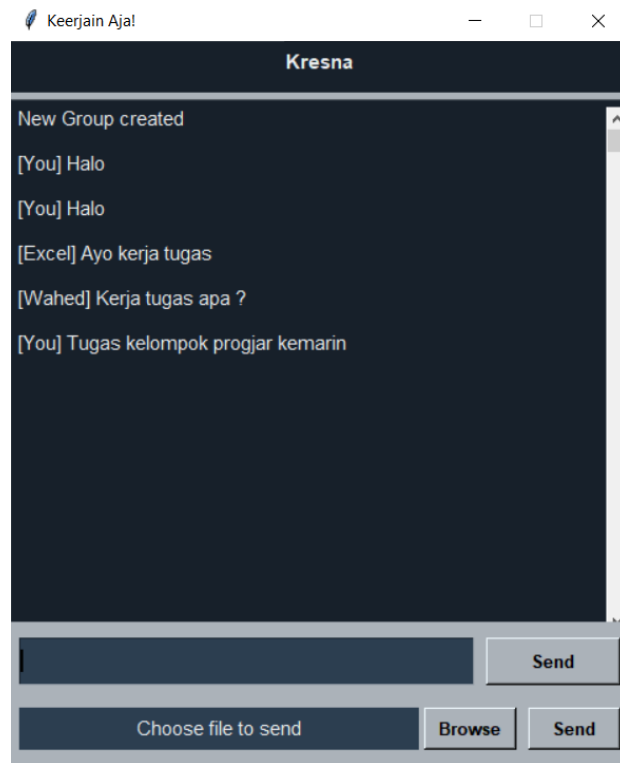
- Menu login

Berikut adalah menu login untuk client. Pada bagian ini kita diminta untuk memasukkan username dan room id.



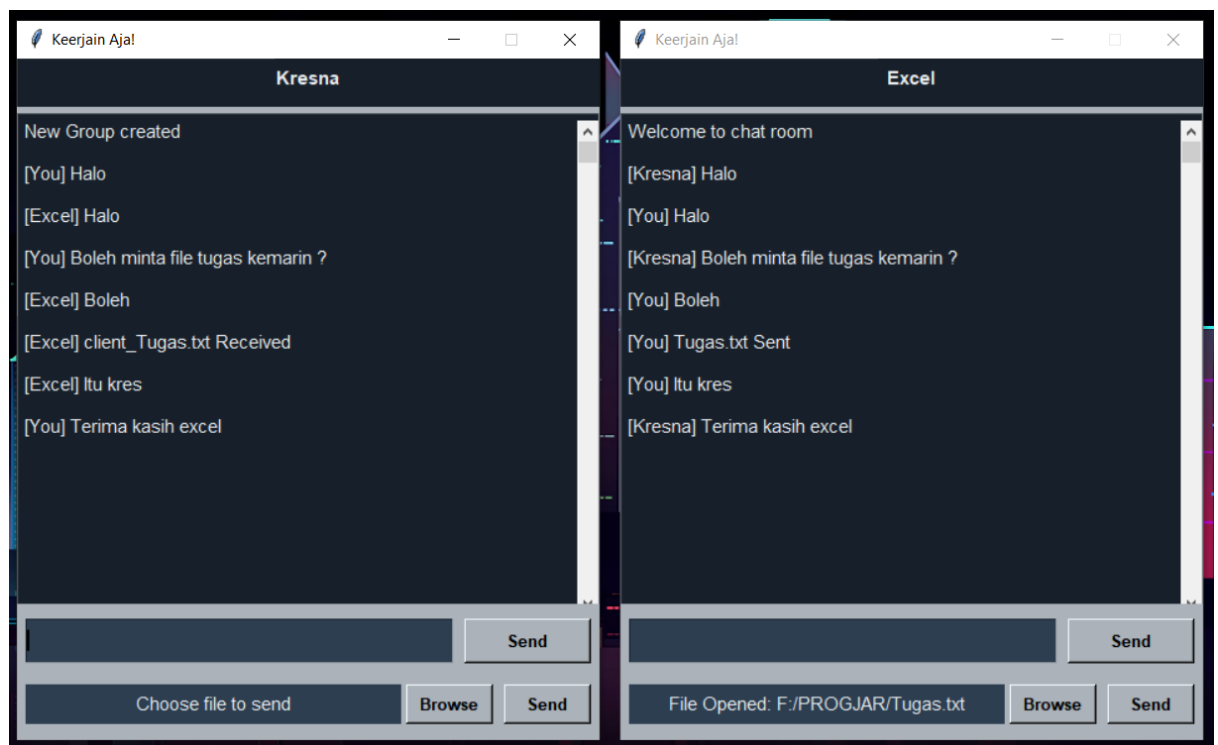
- GUI chat client

Berikut adalah GUI setelah kita memasukkan username dan room id. Disini kita memasuki sebuah group chat dengan room id yang telah kita masukkan di awal.

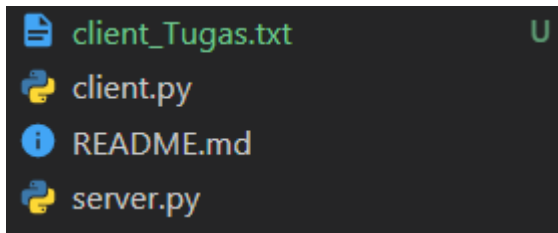


2. Pengiriman file dari user ke user

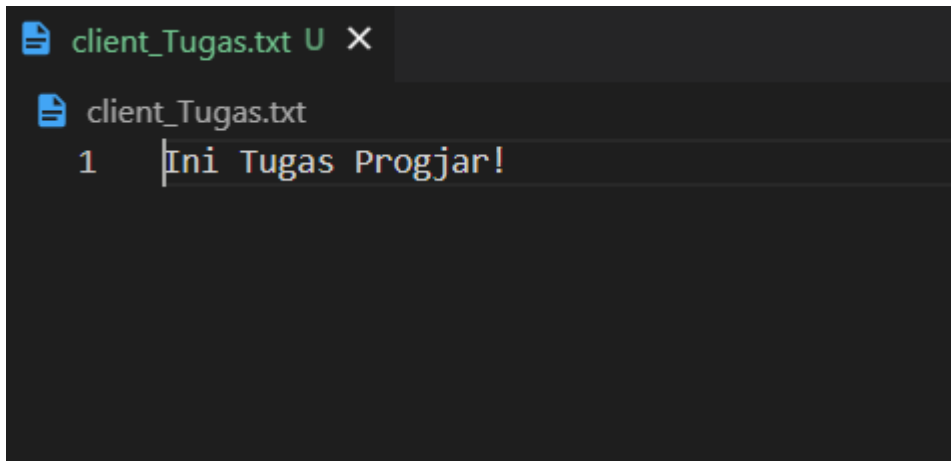
Berikut adalah dokumentasi dari pengiriman file dari user ke user lain. File yang telah dikirim akan masuk ke dalam server.



Dari sisi server, file yang telah dikirim oleh satu client akan masuk ke dalam server.

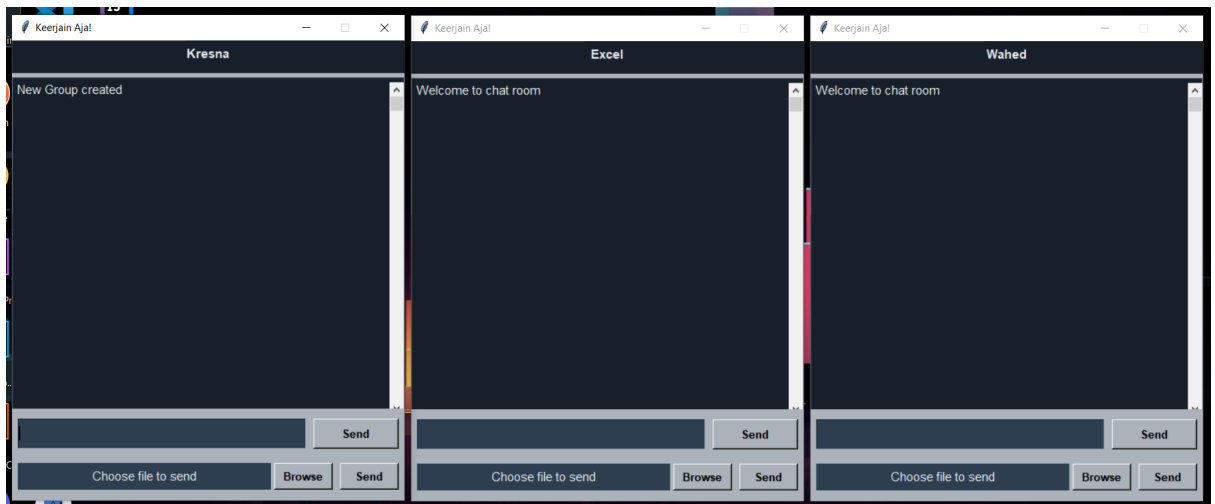


Berikut adalah isi dari file yang telah dikirimkan oleh client.

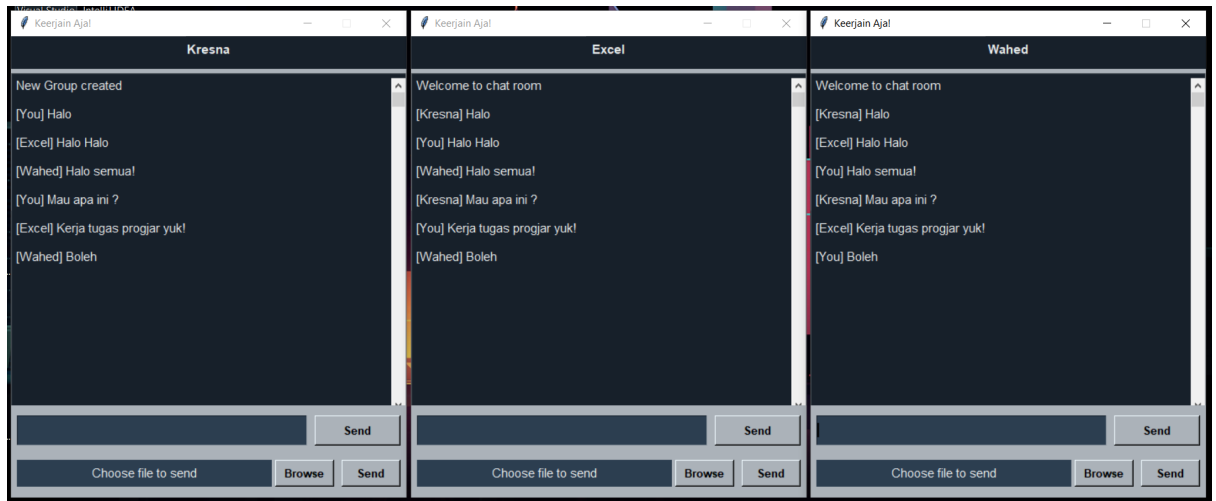


3. Group message

Berikut adalah dokumentasi dari group chat message. Tulisan kalimat pembuka pada GUI client chat yang dimiliki oleh user paling kiri yaitu Kresna memiliki teks yang berbeda karena dalam hal ini, Kresna adalah user yang menginisiasi room group chat.



Selanjutnya adalah dokumentasi dari group chat message dan interaksi yang dapat dilakukan dalam group chat message.



Dokumentasi Protokol

1. File Server

Tujuan : Melayani request dari client serta melakukan broadcasting

Aturan Protokol :

- Client mengirimkan request dalam bentuk string
- Untuk mengirim file harus dimulai dengan request string "FILE" disertai dengan data file, ukuran, nama file

Request yang Dilayani :

a. accept_connection

```
def accept_connections(self, ip_address, port):
    self.ip_address = ip_address
    self.port = port
    self.server.bind((self.ip_address, int(self.port)))
    self.server.listen(100)

    while True:
        connection, address = self.server.accept()
        print(str(address[0]) + ":" + str(address[1]) + " Connected")

        # koneksi yang sudah terbuat di jalankan di thread
        start_new_thread(self.clientThread, (connection,))

    self.server.close()
```

Tujuan : melakukan koneksi ke server

Parameter :

- Self object
- Ip address
- Port

Result

- **Berhasil** : koneksi client terbuat dan dijalankan pada thread
- **Gagal** : koneksi terputus

b. clientThread

```
def clientThread(self, connection):
    user_id = connection.recv(1024).decode().replace("User ", "")
    room_id = connection.recv(1024).decode().replace("Join ", "")

    if room_id not in self.rooms:
        connection.send("New Group created".encode())
    else:
        connection.send("Welcome to chat room".encode())

    # informasi koneksi di simpan sesuai dengan room_id nya
    self.rooms[room_id].append(connection)

    while True:
        try:
            message = connection.recv(1024)
            print(str(message.decode()))
            if message:
                # jika yang dikirim adalah file
                if str(message.decode()) == "FILE":
                    self.broadcastFile(connection, room_id, user_id)

                else:
                    # jika yang di kirim adalah text
                    message_to_send = "<" + str(user_id) + "> " + message.decode()
                    self.broadcast(message_to_send, connection, room_id)

            else:
                self.remove(connection, room_id)
        except Exception as e:
            print(repr(e))
            print("Client disconnected earlier")
            break
```

Tujuan : menerima request dari client berupa pesan string maupun file

Parameter :

- Self object

- Connection object

Result

- **Berhasil :**
 - Message string akan diterima
 - Jika message dikirim dengan string "FILE" maka dilakukan broadcast file
 - Jika message merupakan pesan biasa maka akan dilakukan broadcast pesan
- **Gagal :** error, ditampilkan pesan error dari exception

c. broadcastFile

```
def broadcastFile(self, connection, room_id, user_id):
    file_name = connection.recv(1024).decode()
    lenOfFile = connection.recv(1024).decode()
    for client in self.rooms[room_id]:
        if client != connection:
            try:
                client.send("FILE".encode())
                time.sleep(0.1)
                client.send(file_name.encode())
                time.sleep(0.1)
                client.send(lenOfFile.encode())
                time.sleep(0.1)
                client.send(user_id.encode())
            except:
                client.close()
                self.remove(client, room_id)

    total = 0
    print(file_name, lenOfFile)
    while str(total) != lenOfFile:
        data = connection.recv(1024)
        total = total + len(data)
        for client in self.rooms[room_id]:
            if client != connection:
                try:
                    client.send(data)
                    # time.sleep(0.1)
                except:
                    client.close()
                    self.remove(client, room_id)
    print("Sent")
```

Tujuan : melakukan broadcast file ke semua client

Parameter :

- Self object
- Connection object
- Room id
- User id pengirim

Result

- **Berhasil :**
 - Mengirim pesan “FILE” ke semua client selain pengirim
 - Mengirim string nama file
 - Mengirim string ukuran file
 - Mengirim user id pengirim
 - Mengirim string data file
- **Gagal :** error, ditampilkan pesan error koneksi client terputus

d. broadcast

```
def broadcast(self, message_to_send, connection, room_id):  
    for client in self.rooms[room_id]:  
        if client != connection:  
            try:  
                client.send(message_to_send.encode())  
            except:  
                client.close()  
                self.remove(client, room_id)
```

Tujuan : melakukan broadcast pesan ke semua client

Parameter :

- Self object
- Pesan yang ingin di broadcast
- Connection object
- Room id

Result

- **Berhasil :** Mengirim pesan ke semua client selain pengirim
- **Gagal :** error, ditampilkan pesan error koneksi client terputus

2. File Client

Tujuan : Melakukan request ke server untuk mengirim pesan atau file

Aturan Protokol :

- Client mengirimkan request dalam bentuk string
- Untuk mengirim file harus dimulai dengan request string "FILE" disertai dengan data file, ukuran, nama file

Request yang Dilakukan:

a. goAhead

```
def goAhead(self, username, room_id=0):  
    self.name = username  
    self.server.send(str.encode(username))  
    time.sleep(0.1)  
    self.server.send(str.encode(room_id))  
  
    self.login.destroy()  
    self.layout()  
  
    rcv = threading.Thread(target=self.receive)  
    rcv.start()
```

Tujuan : melakukan koneksi ke server

Parameter :

- Self object
- Username
- Room id

Result

- **Berhasil :** koneksi client terbuat dan dijalankan pada thread
- **Gagal :** -

b. sendMessage

```
def sendMessage(self):  
    self.textCons.config(state=tk.DISABLED)  
    while True:  
        self.server.send(self.msg.encode())  
        self.textCons.config(state=tk.NORMAL)  
        self.textCons.insert(tk.END,  
                             "[You] " + self.msg + "\n\n")  
  
        self.textCons.config(state=tk.DISABLED)  
        self.textCons.see(tk.END)  
        break
```

Tujuan : melakukan request kirim pesan ke server

Parameter :

- Self object

Result

- **Berhasil :** pesan terkirim, dan di tampilkan di room chat
- **Gagal : -**

c. sendFile

```
def sendFile(self):
    self.server.send("FILE".encode())
    time.sleep(0.1)
    self.server.send(str("client_" + os.path.basename(self.filename)).encode())
    time.sleep(0.1)
    self.server.send(str(os.path.getsize(self.filename)).encode())
    time.sleep(0.1)

    file = open(self.filename, "rb")
    data = file.read(1024)
    while data:
        self.server.send(data)
        data = file.read(1024)
    self.textCons.config(state=tk.DISABLED)
    self.textCons.config(state=tk.NORMAL)
    self.textCons.insert(tk.END, "[You] "
                        + str(os.path.basename(self.filename))
                        + " Sent\n\n")
    self.textCons.config(state=tk.DISABLED)
    self.textCons.see(tk.END)
```

Tujuan : melakukan request kirim file ke server

Parameter :

- Self object

Result

- **Berhasil :**
 - Mengirim string "FILE"
 - Mengirim nama file
 - Mengirim ukuran file
 - Mengirimkan data file
- **Gagal : -**

Repository Individu

- Abdur Rachman Wahed 05111840000004
Link: https://github.com/HexaTeam6/Pemrograman_Jaringan_D
- Kresna Adhi Pramana 05111840000072
Link: https://github.com/KresnaAP/Pemrograman_Jaringan_D
- Excel Deo Cornelius 05111840000117
Link: https://github.com/exceldeo/Pemrograman_Jaringan_D

Repository Kelompok

Link: https://github.com/KresnaAP/Pemrograman_Jaringan_D_Kelompok_4

Kelompok dan Tugas

- Abdur Rachman Wahed 05111840000004
Membuat server, pengiriman file
- Kresna Adhi Pramana 05111840000072
Membuat client dengan GUI, pengiriman file
- Excel Deo Cornelius 05111840000117
Membuat client dengan GUI, group chat