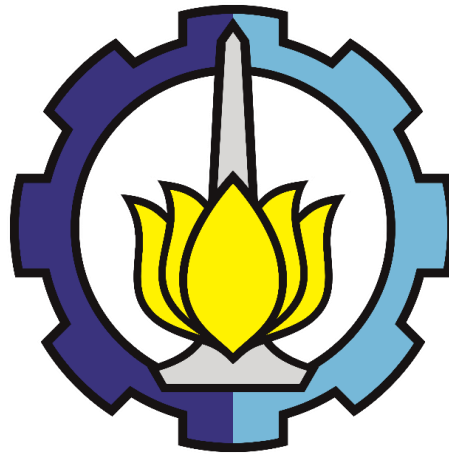


**TUGAS 5**  
**PEMROGRAMAN JARINGAN**



**KELAS: Pemrograman Jaringan - C**

**Dibuat oleh :**

**Ahmad Syauqi**

**05111740000093**

**DEPARTEMEN TEKNIK INFOMATIKA**  
**FAKULTAS TEKNOLOGI INFORMASI DAN KOMUNIKASI**  
**INSTITUT TEKNOLOGI SEPULUH NOPEMBER 2020**

#### A. Multi-threaded Message Service

Service ini merupakan service komunikasi yang memberikan layanan pengiriman pesan antar user-nya. Service ini menggunakan arsitektur server-client yakni dengan menggunakan satu server, lebih dari satu client dapat terhubung ke server tersebut. Hal tersebut dapat dilakukan karena adanya konsep multithreading yang diimplementasikan pada server tersebut.

Secara singkat, server akan menginisialisasi service dimana client bisa terhubung pada server tersebut. Lalu ketika client merequest koneksi ke server tersebut, maka server akan membuat thread baru untuk melayani client tersebut. Selanjutnya client dapat menggunakan service tersebut dengan mengirimkan request berupa command, yang diikuti session dari user dan parameter tambahan apabila ada, dan server akan memberikan respon kepada client berupa JSON.

Command yang dapat digunakan antara lain, **auth** (login), **send** (mengirim pesan), **inbox** (melihat inbox), **activelist** (melihat daftar user yang sedang terhubung pada server), dan **logout**.

#### B. Format Data

Data yang digunakan untuk komunikasi berupa teks alfanumerik dengan format tertentu. Data yang dikirimkan kepada server didapat dari input user yang menggunakan service tersebut. Data yang diterima dari server berupa JSON string yang berisi informasi data tersebut.

Secara umum, format alfanumerik yang diinputkan untuk dikirimkan ke server adalah sebagai berikut

#### **COMMAND [PARAMS[]]**

yakni sebuah urutan string yang diawali dengan **command**, yang juga bisa diikuti oleh parameters yang dipisahkan oleh spasi. Setelah user mengetikkan command tersebut, maka program client akan otomatis mengirimkan string tersebut namun dengan menambahkan token berupa **session id** untuk fitur-fitur selain fitur autentikasi, sehingga format data yang dikirim akan menjadi seperti berikut,

#### **Command [session\_id – except login] [parameters[]] - optional]**

Setelah command tersebut dikirimkan, maka client dapat menerima response yang didapat dari server.

Data yang dikirimkan oleh server berupa string dengan format JSON dengan key berupa **response** dan juga **messages**. **Response** adalah key yang berisi response dari server, yang berupa **OK** apabila server berhasil mengeksekusi command, dan **ERROR** apabila server gagal mengeksekusi command client tersebut.

### C. Daftar Fitur

Fitur-fitur yang tersedia pada service ini antara lain adalah:

#### 1. Login – Autentikasi user

##### a. Deskripsi

Fitur ini adalah fitur untuk membuat session pada service tersebut. Autentikasi user dibutuhkan karena fitur lain hanya bisa digunakan setelah client mendapatkan session id dari server

##### b. Parameter Input

Input berupa string alfanumerik,

**auth [username] [password]**

dimana username adalah username dari client dan password adalah password dari client, yang dipisahkan oleh spasi setiap string nya

##### c. Hasil output

Output dari command tersebut adalah response dari server, yang antara lain berupa

- **OK – username [username] logged in, token [token id]**, apabila login berhasil dilakukan
- **ERROR – [message]**, apabila login gagal di eksekusi

##### d. Contoh input dan output

Client:

```
Command :auth lineker surabaya
diterima dari server b'{"status": "OK", "tokenid": "15b0b0f5-b9ab-4
846-9c1e-ea4c8cc2c16"}'
diterima dari server b'5"}\r\n\r\n'
end of string
username lineker logged in, token 15b0b0f5-b9ab-4846-9c1e-ea4c8cc2c
165
```

Server:

```
WARNING:root:data dari client: auth lineker surabaya

WARNING:root:AUTH: auth lineker surabaya
WARNING:root:balas ke client: {"status": "OK", "tokenid": "15b0b0
f5-b9ab-4846-9c1e-ea4c8cc2c165"}
```

## 2. Mengirim Pesan

### a. Deskripsi

Fitur ini adalah fitur mengirimkan pesan dari user yang terdaftar pada server, ke user lain yang terdaftar pada server. Pesan tersebut akan disimpan pada server yang nantinya dapat dilihat oleh user yang menerima.

### b. Parameter Input

Input berupa string alfanumerik,

**send [username] [pesan yang dikirim]**

dimana username adalah username dari client yang dituju dan pesan yang dikirim adalah pesan yang akan dikirimkan ke client tersebut, yang dipisahkan oleh spasi setiap string nya

### c. Hasil output

Output dari command tersebut adalah response dari server, yang antara lain berupa

- **OK – message sent to [username]**, apabila pesan berhasil dikirim
- **ERROR – [message]**, apabila pengiriman pesan gagal di eksekusi

### d. Contoh input dan output

Client:

```
Command 15b0b0f5-b9ab-4846-9c1e-ea4c8cc2c165:send messi halo, messi
, apa kabar?
send 15b0b0f5-b9ab-4846-9c1e-ea4c8cc2c165 messi halo, messi, apa k
abar?

diterima dari server b'{"status": "OK", "message": "Message Sent"}\
r\n\r\n'
end of string
message sent to messi
```

Server:

```
WARNING:root:data dari client: send 15b0b0f5-b9ab-4846-9c1e-ea4c8c
c2c165 messi halo, messi, apa kabar?

WARNING:root:SEND: session 15b0b0f5-b9ab-4846-9c1e-ea4c8cc2c165 se
nd message from lineker to messi
WARNING:root:balas ke client: {"status": "OK", "message": "Messag
e Sent"}
```

## 3. Membaca inbox

### a. Deskripsi

Fitur ini adalah fitur membaca pesan yang diterima oleh user ini. Pesan yang diterima akan berupa dictionary dengan 2 key value yakni pengirim dan pesan yang diterima.

### b. Parameter Input

Input berupa string alfanumerik,

**inbox**

dimana command yang di inputkan tanpa adanya parameter tambahan

### c. Hasil output

Output dari command tersebut adalah response dari server, yang antara lain berupa

- **OK – dictionary[{username},{messages}]**, apabila inbox bisa ditampilkan

- **ERROR – [message]**, apabila pengiriman command gagal di eksekusi
- d. Contoh input dan output

Client:

```
Command 52cf191b-034b-4e7e-bde9-694b53f32fe3:inbox
diterima dari server b'{"status": "OK", "messages": {"lineker": [{"msg_from": "Gary Lin'
diterima dari server b'eker", "msg_to": "Lionel Messi", "msg": "
halo, messi, apa kaba'
diterima dari server b'r? \\r\\n"}]}}\\r\\n\\r\\n'
end of string
{"lineker": [{"msg_from": "Gary Lineker", "msg_to": "Lionel Messi"
, "msg": " halo, messi, apa kabar? \\r\\n"}]}}
```

Server:

```
WARNING:root:data dari client: inbox 52cf191b-034b-4e7e-bde9-694b5
3f32fe3

WARNING:root:INBOX: 52cf191b-034b-4e7e-bde9-694b53f32fe3
WARNING:root:balas ke client: {"status": "OK", "messages": {"line
ker": [{"msg_from": "Gary Lineker", "msg_to": "Lionel Messi", "msg
": " halo, messi, apa kabar? \\r\\n"}]}}
```

#### 4. Melihat user yang aktif

##### a. Deskripsi

Fitur ini adalah fitur melihat seluruh daftar user yang sedang aktif. Response yang didapat berupa username, nama user, dan negaranya.

##### b. Parameter Input

Input berupa string alfanumerik,

**activeslist**

dimana command yang di inputkan tanpa adanya parameter tambahan

##### c. Hasil output

Output dari command tersebut adalah response dari server, yang antara lain berupa

- **OK – dictionary[{username,nama, negara}]**, apabila ada user yang sedang aktif
- **ERROR – [message]**, apabila pengiriman command gagal di eksekusi

##### d. Contoh input dan output

Client:

```
Command 15b0b0f5-b9ab-4846-9c1e-ea4c8cc2c165:activelist
diterima dari server b'{"status": "OK", "activeusers": [{"username"
: "lineker", "nama":'
diterima dari server b' "Gary Lineker", "negara": "Inggris"}, {"use
rname": "messi", "na'
diterima dari server b'ma": "Lionel Messi", "negara": "Argentina"}]
}\\r\\n\\r\\n'
end of string
username      nama      negara
lineker Gary Lineker Inggris
messi Lionel Messi Argentina
```

Server:

```
WARNING:root:data dari client: activist 15b0b0f5-b9ab-4846-9c1e-ea4c8cc2c165

WARNING:root:ACTIVE_USERS: 15b0b0f5-b9ab-4846-9c1e-ea4c8cc2c165
WARNING:root:balas ke client: {"status": "OK", "activeusers": [{"username": "lineker", "nama": "Gary Lineker", "negara": "Inggris"}, {"username": "messi", "nama": "Lionel Messi", "negara": "Argentina"}]}
```

## 5. Logout

### a. Deskripsi

Fitur ini adalah fitur untuk logout dari server. Fitur ini akan secara otomatis menghapus session kita yang terdaftar pada server sehingga kita secara otomatis terlogout dari server dan tidak bisa menggunakan fitur yang lain. Logout memerlukan session id kita atau dalam arti lain kita telah login kedalam service, apabila belum, maka logout tidak bisa dilakukan

### b. Parameter Input

Input berupa string alfanumerik,

#### **logout**

dimana command yang di inputkan tanpa adanya parameter tambahan

### c. Hasil output

Output dari command tersebut adalah response dari server, yang antara lain berupa

- **OK – logged out**, apabila logout berhasil dilakukan
- **ERROR – [message]**, apabila pengiriman command gagal di eksekusi

### d. Contoh input dan output

Client:

```
Command 15b0b0f5-b9ab-4846-9c1e-ea4c8cc2c165:logout
diterima dari server b'{"status": "OK", "tokenid": "\u0000"}\r\n\r\n'
end of string
logged out
```

Server:

```
WARNING:root:data dari client: logout 15b0b0f5-b9ab-4846-9c1e-ea4c8cc2c165

WARNING:root:LOGGING OUT: 15b0b0f5-b9ab-4846-9c1e-ea4c8cc2c165
WARNING:root:balas ke client: {"status": "OK", "tokenid": "\u0000"}
```