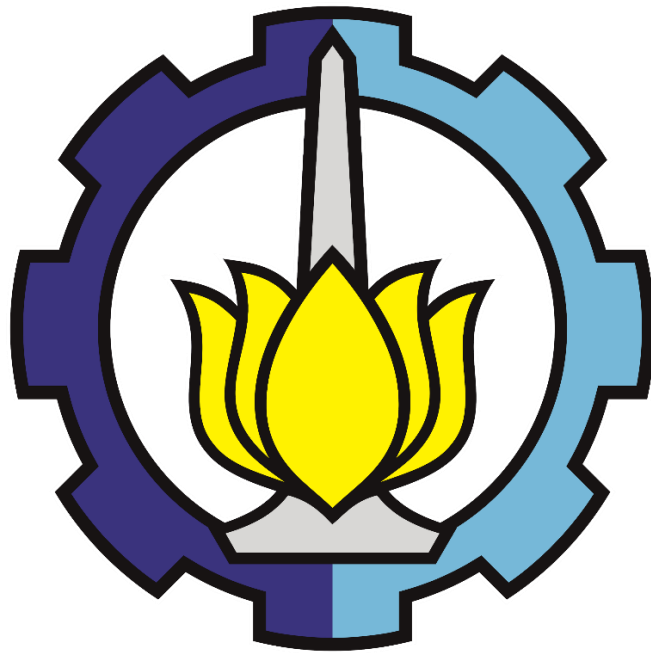


Dokumentasi Tugas 5 – Program Chat dengan *Multi-Threading* Server

Pemrograman Jaringan B



Ditulis Oleh:

Bayu Laksana

05111740000020

**DEPARTEMEN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI ELEKTRO DAN INFORMATIKA CERDAS
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
2020**

Dokumentasi Program Chat Menggunakan *Multi-Threading*

Nama Service

Program chat menggunakan Multi-Threading

Deskripsi

Program yang dibangun merupakan program berbasis *command line interface* sederhana yang berfungsi sebagai program perpesanan antar pengguna (*chat*). Disini pengguna dapat bertukar informasi menggunakan pesan teks sederhana.

Service dalam program chat ini menyediakan fungsionalitas dasar untuk mengirim pesan berupa teks sederhana kepada pengguna lain layaknya aplikasi perpesanan pada umumnya, melakukan login/logout, melihat isi pesan masuk pengguna, dan melihat pengguna lain yang sedang aktif (*online*).

Format Data

Komunikasi data dikirimkan melalui *socket* dengan data yang diinputkan oleh client berupa string yang berisi perintah (*command*) dan kumpulan argumen sesuai dengan perintahnya. Perintah dan argumen-argumen dipisahkan oleh spasi.

1. Format General

```
[command]<spasi>[argumen1]<spasi>[argumen2]...[argumen]
```

Daftar Fitur

Fitur-fitur yang tersedia dalam program chat berbasis *command line interface* ini meliputi :

1. Autentikasi Login

a. Deskripsi dan Tujuan

Fitur Autentikasi Login ini merupakan fitur dimana memungkinkan pengguna untuk melakukan login ke dalam program menggunakan kredensial berupa username dan password yang telah terdaftar pada program. Ketika pengguna telah berhasil login, maka program akan membuatkan *session* kepada program client dan pengguna pun dapat mengakses fitur-fitur lainnya.

Tujuan dari fitur ini adalah untuk memvalidasi pengguna yang masuk ke dalam sistem apakah sudah terdaftar atau belum agar tidak mengakses data dari pengguna lainnya dan menjamin hanya dapat mengakses data miliknya saja.

Format perintah yang dituliskan oleh pengguna untuk melakukan login adalah:

```
auth login [username] [password]
```

b. Parameter Input

Perintah yang digunakan adalah **auth** . Kemudian, terdapat 2 parameter command, yakni **[username]** dan **[password]** masing-masing merupakan string tanpa spasi.

c. Hasil Keluaran

Hasil keluaran dari perintah tersebut bervariasi tergantung kondisi saat ini.

Username tidak ditemukan

```
Error, User tidak ada
```

Password salah/tidak sesuai atau gagal login

```
Error, Login gagal
```

Username dan password benar

```
Masuk sebagai [username]. Token ID [token id]
```

d. Contoh Message Input dan Keluaran

Username tidak ditemukan

```
[Command] >> auth login xxx xxx
```

```
[Command] >> auth login xxx xxx
Diterima dari server {"status": "Error", "message": "User tidak ada"}
end of string

Error, User tidak ada
```

Password salah/tidak sesuai atau gagal login

```
[Command] >> auth login bayulaxana xxx
```

```
[Command] >> auth login bayulaxana xxx
Diterima dari server {"status": "Error", "message": "Login gagal"}
end of string

Error, Login gagal
```

Username dan password benar

```
[Command] >> auth login bayulaxana password123
```

```
[Command] >> auth login bayulaxana password123
Diterima dari server {"status": "Ok", "tokenId": "cf96ecdf-a38b-4d6f-ad
0c-d7367219c45
Diterima dari server c"}
end of string

Masuk sebagai bayulaxana. Token ID cf96ecdf-a38b-4d6f-ad0c-d7367219c45c
```

2. Autentikasi Logout

a. Deskripsi dan Tujuan

Fitur Autentikasi Logout merupakan fitur dimana memungkinkan pengguna yang sudah login ke program agar bisa logout (keluar) dari program. Ketika pengguna melakukan logout dari sistem, maka session yang telah dibuat program akan dihapus dan status pengguna menjadi tidak aktif (offline).

Tujuan dari fitur ini adalah untuk keluar dari sistem, menghapus session pengguna yang kemudian pengguna menjadi offline.

Format perintah yang dituliskan oleh pengguna untuk melakukan logout adalah:

```
auth logout
```

b. Parameter Input

Perintah yang digunakan adalah **auth**. Dalam fitur ini tidak ada parameter command yang dibutuhkan.

c. Hasil Keluaran

Hasil keluaran dari perintah tersebut bervariasi tergantung kondisi saat ini.

Pengguna belum login

```
Anda belum login
```

Pengguna login dan berhasil logout

```
Logout berhasil
```

d. Contoh Message Input dan Keluaran

Pengguna belum login

```
[Command] >> auth logout
```

```
[Command] >> auth logout  
Anda belum login
```

Pengguna login dan berhasil logout

```
[Command] >> auth logout
```

```
[Command bayulaxana/cf96ecdf-a38b-4d6f-ad0c-d7367219c45c] >> auth logout  
Diterima dari server {"status": "Ok", "message": "Berhasil logout"}  
end of string  
  
Logout berhasil
```

3. Mengirimkan Pesan

a. Deskripsi dan Tujuan

Dalam fitur Mengirimkan Pesan, jelas sekali disini bahwa pengguna yang aktif (telah melakukan login) dapat mengirimkan pesan sederhana berupa teks kepada pengguna lain (tidak harus aktif) yang nantinya dapat dibaca oleh pengguna lain tersebut.

Tujuannya jelas, yakni agar pengguna dapat saling bertukar informasi dengan pengguna lain melalui pesan teks sederhana tersebut.

Format perintah yang dituliskan oleh pengguna untuk mengirimkan pesan adalah:

```
send<spasi>[username tujuan]<spasi>[isi pesan teks]
```

Isi pesan teks disini bisa dipisahkan spasi dan panjangnya dapat bervariasi menyesuaikan input dari pengguna.

b. Parameter Input

Perintah yang digunakan adalah **send**. Dalam fitur ini terdapat 2 parameter yang dibutuhkan yakni berupa:

- **[username tujuan]** yakni username pengguna yang akan dikirim pesan.
- **[isi pesan teks]**

c. Hasil Keluaran

Hasil keluaran dari perintah tersebut bervariasi tergantung kondisi saat ini.

Pengguna belum login

```
Error, login terlebih dahulu
```

Username tujuan tidak ditemukan

```
Error, User tidak ditemukan
```

Pesan berhasil dikirimkan

```
Pesan telah terkirim ke [username tujuan]
```

d. Contoh Message Input dan Keluaran

Pengguna belum login

```
[Command] >> send messi hai
```

```
[Command] >> send messi hai  
Error, login terlebih dahulu
```

Username tujuan tidak ditemukan

```
[Command] >> send asus hai asus
```

```
[Command bayulaxana/ad008a76-5962-4854-9937-f8f888cf4005] >> send asus hai asus  
Diterima dari server {"status": "Error", "message": "User tidak ditemukan"}  
end of string  
  
Error, User tidak ditemukan
```

Pesan berhasil dikirimkan

```
[Command] >> send messi Hai Messi
```

```
[Command bayulaxana/ad008a76-5962-4854-9937-f8f888cf4005] >> send messi Hai Messi  
Diterima dari server {"status": "Ok", "message": "Pesan terkirim"}  
end of string  
  
Pesan telah terkirim ke messi
```

4. Melihat Inbox

a. Deskripsi dan Tujuan

Inbox merupakan istilah yang sering digunakan untuk mengartikan kotak pesan yang masuk. Program chat ini juga menyediakan fitur untuk Melihat Inbox pengguna tentunya dengan syarat pengguna tersebut harus sudah melakukan login ke dalam program.

Tujuannya adalah agar pengguna bisa melihat isi pesan-pesan yang masuk dari pengguna lain dan dapat dibaca oleh pengguna tersebut.

Format perintah yang dituliskan oleh pengguna untuk melihat inbox adalah:

```
inbox
```

b. Parameter Input

Perintah yang digunakan adalah **inbox**. Dalam fitur ini tidak ada parameter command yang dibutuhkan.

c. Hasil Keluaran

Pengguna belum login

```
Error, login terlebih dahulu
```

Pengguna berhasil melihat inbox

```
{
  [username asal]: [
    {
      "msg_from": [username asal],
      "msg_to": [username tujuan],
      "msg": [isi pesan]
    }
  ]
},
....
....
```

d. Contoh Message Input dan Keluaran

Pengguna belum login

```
[Command] >> inbox
```

```
[Command] >> inbox
Error, login terlebih dahulu
```

Pengguna berhasil melihat inbox

```
[Command] >> inbox
```

```
[Command messi/ec025d57-d94d-41fc-8cd0-1e533825e219] >> inbox
Diterima dari server {"status": "Ok", "message": {"bayulaxana": [{"msg_from": "Bayu L
Diterima dari server aksana", "msg_to": "Lionel Messi", "msg": " Hai Messi \r\n"}]}}
Diterima dari server
end of string

{
  "bayulaxana": [
    {
      "msg_from": "Bayu Laksana",
      "msg_to": "Lionel Messi",
      "msg": " Hai Messi \r\n"
    }
  ]
}
```

5. Melihat Pengguna Aktif

a. Deskripsi dan Tujuan

Fitur ini memungkinkan pengguna yang telah login dapat melihat daftar pengguna lain yang juga sedang aktif pada saat yang sama. Daftar yang diperlihatkan berupa Nama pengguna, username dan session id dari pengguna yang sedang aktif.

Tujuannya adalah agar ketika pengguna ingin berkomunikasi dengan pengguna lain, maka ia bisa mengetahui siapa saja pengguna yang sedang aktif.

Format perintah yang dituliskan oleh pengguna untuk melihat pengguna aktif adalah:

```
active_user
```

b. Parameter Input

Perintah yang digunakan adalah **active_user**. Dalam fitur ini tidak ada parameter command yang dibutuhkan.

c. Hasil Keluaran

Pengguna belum login

```
[Error, login terlebih dahulu
```

Pengguna berhasil melihat daftar

```
Daftar user yang aktif
=====
[Nama] [username] / [token id]
...
...
...
```


d. Contoh Message Input dan Keluaran

Pengguna belum login

```
[Command] >> active_user
```

```
[Command] >> active_user  
Error, login terlebih dahulu
```

Pengguna berhasil melihat daftar

```
[Command] >> active_user
```

```
[Command bayulaxana/3a1eafca-da58-48d1-bc1b-cf30e950dac0] >> active_user  
Diterima dari server {"status": "Ok", "message": {"9fac396f-65ca-437c-91ad-a2ac0f3cbd  
Diterima dari server 46": {"username": "messi", "userdetail": {"username": "messi", "  
Diterima dari server nama": "Lionel Messi", "password": "surabaya", "incoming": {}, "  
Diterima dari server outgoing": {}}, "3a1eafca-da58-48d1-bc1b-cf30e950dac0": {"usern  
Diterima dari server ame": "bayulaxana", "userdetail": {"username": "bayulaxana", "na  
Diterima dari server ma": "Bayu Laksana", "password": "password123", "incoming": {},  
Diterima dari server "outgoing": {}}}} }  
end of string  
  
Daftar user yang aktif  
=====
```

Lionel Messi	[messi]	/	9fac396f-65ca-437c-91ad-a2ac0f3cbd46
Bayu Laksana	[bayulaxana]	/	3a1eafca-da58-48d1-bc1b-cf30e950dac0