**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 comman line interface sederhana yang berfungsi sebagai program perpesanana 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.

## Format General

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

# Daftar Fitur

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

## Autentikasi Login

1. **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]

1. **Parameter Input**

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

1. **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

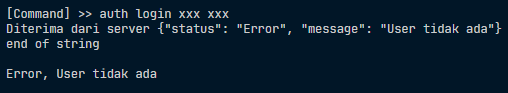
**Username dan password benar**

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

1. **Contoh Message Input dan Keluaran**

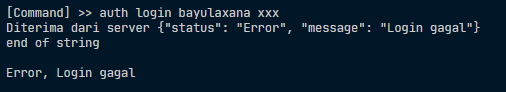
**Username tidak ditemukan**

[Command] >> auth login xxx xxx



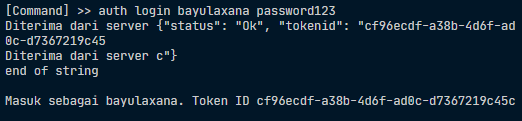
**Password salah/tidak sesuai atau gagal login**

[Command] >> auth login bayulaxana xxx



**Username dan password benar**

[Command] >> auth login bayulaxana password123



## Autentikasi Logout

1. **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

1. **Parameter Input**

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

1. **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

1. **Contoh Message Input dan Keluaran**

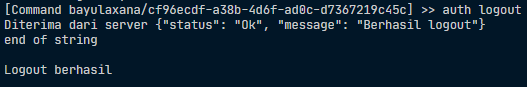
**Pengguna belum login**

[Command] >> auth logout



**Pengguna login dan berhasil logout**

[Command] >> auth logout



## Mengirimkan Pesan

1. **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.

1. **Parameter Input**

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

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

1. **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***]

1. **Contoh Message Input dan Keluaran**

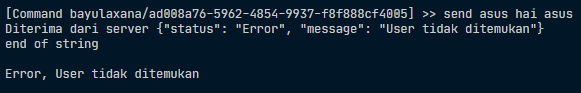
**Pengguna belum login**

[Command] >> send messi hai



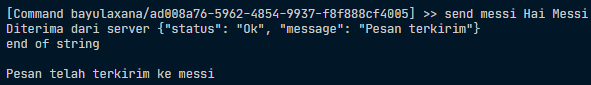
**Username tujuan tidak ditemukan**

[Command] >> send asus hai asus



**Pesan berhasil dikirimkan**

[Command] >> send messi Hai Messi



## Melihat Inbox

1. **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

1. **Parameter Input**

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

1. **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]

}

]

},

....

....

1. **Contoh Message Input dan Keluaran**

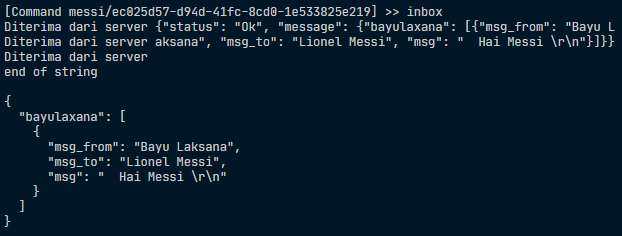
**Pengguna belum login**

[Command] >> inbox



**Pengguna berhasil melihat inbox**

[Command] >> inbox



## Melihat Pengguna Aktif

1. **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

1. **Parameter Input**

Perintah yang digunakan adalah **active\_user**. Dalam fitur ini tidak ada parameter command yang dibutuhkan.

1. **Hasil Keluaran**

**Pengguna belum login**

[Error, login terlebih dahulu

**Pengguna berhasil melihat daftar**

Daftar user yang aktif

======================

[Nama] [username] / [token id]

...

...

...

1. **Contoh Message Input dan Keluaran**

**Pengguna belum login**

[Command] >> active\_user



**Pengguna berhasil melihat daftar**

[Command] >> active\_user

