

**DOKUMENTASI LAYANAN PENGIRIMAN PESAN (*CHATTING*)
MENGUNAKAN MULTITHREADED SERVER
PEMROGRAMAN JARINGAN – B**



Dipersiapkan oleh :

Tubagus Irkham Izzata Apriwibowo

05111740000012

Departemen Informatika – Institut Teknologi Sepuluh Nopember

Jl. Teknik Kimia, Keputih, Kec. Sukolilo, Kota SBY, Jawa Timur 60111

a. Nama Service

Nama dari sistem ini adalah Layanan Pengiriman Pesan (*Chatting*) Dengan Menggunakan *Multithreaded Server*.

b. Deskripsi Sistem

Sistem ini adalah sistem untuk memberikan layanan pengiriman pesan atau biasa disebut chatting antar pengguna yang sudah terdaftar di dalam sistem. Sistem ini berbasis *server* dan *client* yang dihubungkan melalui *socket* dan *server* pada sistem ini sendiri adalah *multithreaded server* yang memungkinkan antar pengguna dapat berkomunikasi melalui *server* yang ada.

Seperti yang telah disebutkan di atas, sistem ini berbasis *server* dan *client* yang mana kedua aktor tersebut memiliki fungsinya masing-masing. *Server* akan bertindak sebagai inti dari sistem dikarenakan semua alur data yang dikirimkan maupun terkirim akan disimpan dan diproses di dalam server. Sedangkan, *client* disini adalah *user* yang menggunakan sistem ini. *User* sendiri dapat menggunakan beberapa fitur yang disediakan di dalam sistem ini. Fitur tersebut antara lain ada fitur untuk *login*, fitur untuk mengirim pesan, fitur untuk melihat pesan yang dikirimkan oleh *user* lain, fitur untuk melihat daftar user yang sedang aktif, dan juga fitur untuk *logout* dari sistem.

c. Format Data

Berikut adalah format data yang digunakan dalam sistem ini untuk fitur-fitur yang tersedia :

- Untuk fitur *login* :
auth *spasi* username *spasi* password (ex : auth messi surabaya)
- Untuk fitur mengirim pesan atau *send* :
send *spasi* username penerima pesan *spasi* pesan (ex : send henderson halo!)
- Untuk fitur melihat pesan yang dikirimkan oleh *user* lain atau *inbox* :
inbox
- Untuk fitur melihat daftar *user* yang sedang aktif atau *list user* :

list

- Untuk fitur *logout* :
logout

d. Daftar Fitur

Untuk menggunakan sistem ini sebagai user, maka jalankan file **chat-cli.py**. Berikut adalah daftar dan deskripsi lengkap dari fitur-fitur yang tersedia di dalam sistem ini :

1. Fitur *Login*

a. Deskripsi dan Tujuan

Fitur ini adalah fitur pertama yang akan dijumpai oleh *user* jika ingin memasuki sistem. Selain fitur ini berguna untuk memasuki sistem, fitur ini juga berguna untuk melakukan autentikasi *user* yang masuk. Autentikasi *user* tersebut dilakukan dengan cara menyamakan data *user* yang diinputkan dengan data *user* yang terdaftar pada file *chat.py*. Maka dari itu, jika ingin menggunakan sistem layanan *chat* ini, maka jika *user* belum terdaftar harus didaftarkan terlebih dahulu dengan menambahkan biodata *user* ke dalam file *chat.py*.

b. Parameter *Input*

Parameter *input* untuk fitur ini adalah :

auth *spasi* username *spasi* password (ex : auth messi Surabaya)

c. Hasil Keluaran dan Penjelasan Status Keluaran

- Jika *user* berhasil *login*, maka sistem akan menampilkan status **OK** beserta token-id *user* tersebut. Sistem juga akan menampilkan pesan yang berisi (**username *user* logged in, token *token-id-user*.**)
- Jika *user* gagal untuk melakukan *login*, maka sistem akan menampilkan status **ERROR** dengan pesan sesuai kesalahannya (**Password Salah atau User Tidak Ada**).

d. Contoh *Message Input* dan Keluaran

- Contoh *input* yang pertama adalah jika *user* berhasil masuk ke dalam sistem (ex : auth messi surabaya).

```
C:\Users\asus\Anaconda3\python.exe "D:/Kuliah/Semester 6/ProgJar/Tugas 5/chat-cli.py"
Command :auth messi surabaya
diterima dari server b'{"status": "OK", "tokenid": "9526a9a9-5e81-4a62-b4a7-039646fd748"}\r\n\r\n'
diterima dari server b'e')\r\n\r\n'
end of string
username messi logged in, token 9526a9a9-5e81-4a62-b4a7-039646fd748e
```

- Contoh *input* yang kedua adalah jika *user* gagal masuk ke dalam sistem dengan *username* yang salah atau tidak terdaftar.

```
Command :auth pepe surabaya
diterima dari server b'{"status": "ERROR", "message": "User Tidak Ada"}\r\n\r\n'
end of string
Error, User Tidak Ada
```

- Contoh *input* yang ketiga adalah jika *user* gagal masuk ke dalam sistem dengan *password* yang salah.

```
Command :auth messi jakarta
diterima dari server b'{"status": "ERROR", "message": "Password Salah"}\r\n\r\n'
end of string
Error, Password Salah
```

2. Fitur Pengiriman Pesan atau *Send*

a. Deskripsi dan Tujuan

Fitur ini adalah fitur untuk pengiriman pesan antar *user* yang terdaftar di dalam sistem ini. Fitur ini dapat dianggap sebagai fitur utama dari sistem layanan *chatting* ini sendiri. *User* dapat saling mengirim pesan yang mereka inginkan kepada *user* lain yang berhak menerima pesan tersebut. Komunikasi antar *user* pada fitur ini dihubungkan melalui *socket* dan juga menggunakan *multithread* agar banyak *user* dapat terhubung satu sama lain dalam satu *server*.

b. Parameter *Input*

Parameter *input* untuk fitur ini adalah :

send *user-tujuan* pesan

c. Hasil Keluaran dan Status Keluaran

- Jika proses pengiriman pesan kepada *user* tujuan berhasil, sistem akan menampilkan pesan **“message sent to *user-tujuan*”**.
- Jika proses pengiriman pesan kepada *user* tujuan gagal, maka sistem akan mengembalikan status **ERROR** dan juga menampilkan pesan **“Error, User Tidak Ditemukan”**.

d. Contoh *Message Input* dan Keluaran

- Contoh *input* yang pertama adalah jika *user* berhasil mengirim pesan kepada *user* tujuan (ex : send henderson halo hen!).

```
Command 9526a9a9-5e81-4a62-b4a7-039646fd748e: send henderson halo hen!
send 9526a9a9-5e81-4a62-b4a7-039646fd748e henderson halo hen!

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

- Contoh *input* yang kedua adalah jika *user* gagal mengirim pesan kepada *user* tujuan dikarenakan *user* tujuan belum terdaftar.

```
Command 7e6c9d41-41c0-4ade-81c2-e9717809e777: send bepe halo pe!
send 7e6c9d41-41c0-4ade-81c2-e9717809e777 bepe halo pe!

diterima dari server b'{"status": "ERROR", "message": "User Tidak Ditemukan"}\r\n\r\n'
end of string
Error, User Tidak Ditemukan
```

3. Fitur Melihat Pesan yang Dikirimkan oleh *User* Lain atau *Inbox*

a. Deskripsi dan Tujuan

Fitur ini bertujuan untuk melihat pesan yang dikirimkan oleh *user* lain. Fitur ini sendiri akan dapat digunakan jika *user* lain tersebut telah berhasil mengirimkan pesan kepada *user* penerima.

b. Parameter *Input*

Parameter *input* dari fitur ini adalah :

inbox

c. Hasil Keluaran dan Status Keluaran

Hasil keluaran dari fitur ini adalah sistem akan mengembalikan status **OK** serta menampilkan nama *user* yang mengirimkan pesan, nama *user* yang menerima pesan, dan juga pesan yang terkirim.

d. Contoh *Message Input* dan Keluaran

- Contoh *input* yang pertama adalah jika di dalam *inbox* tidak ada pesan.

```
Command b4b6b4e4-226a-4461-b284-b60e688cee04: inbox
diterima dari server b'{"status": "OK", "messages": {}}\r\n\r\n'
end of string
{}
```

- Contoh input yang kedua adalah jika di dalam *inbox* sudah ada pesan yang berhasil terkirim.

```
Command 44e66bc2-9aad-4970-be97-5fda34fd4184: list
diterima dari server b'{"status": "OK", "messages": {"henderson": [{"msg_from": "Jordan"
diterima dari server b' Henderson', "msg_to": "Lionel Messi", "msg": " halo messi! \r\n"
diterima dari server b'n"]}}}\r\n\r\n'
end of string
{"henderson": [{"msg_from": "Jordan Henderson", "msg_to": "Lionel Messi", "msg": " halo messi! \r\n"}]}
```

4. Fitur Melihat Daftar *User* yang Sedang Aktif atau *List User*

a. Deskripsi dan Tujuan

Fitur ini berfungsi untuk melihat daftar *user* yang sedang aktif menggunakan sistem ini. Fitur ini memiliki tujuan agar *user* dapat saling melakukan pengiriman pesan pada *user* yang juga sedang menggunakan sistem layanan *chatting* ini.

b. Parameter *Input*

Parameter input dari fitur ini adalah :

list

c. Hasil Keluaran dan Status Keluaran

Fitur ini akan mengembalikan status **OK** jika fitur ini dapat berhasil digunakan dan juga akan menampilkan daftar nama *user* yang sedang aktif menggunakan sistem ini.

d. Contoh *Message Input* dan Keluaran

- Contoh keluaran yang pertama adalah jika hanya 1 *user* yang sedang aktif menggunakan sistem ini.

```
Command b4b6b4e4-226a-4461-b284-b60e688cee04: list
diterima dari server b'{"status": "OK", "user": ["lineker"]}\r\n\r\n'
end of string
Active users are : ["lineker"]
```

- Contoh keluaran yang kedua adalah jika terdapat beberapa *user* yang sedang aktif menggunakan sistem ini.

```
Command b4b6b4e4-226a-4461-b284-b60e688cee04: list
diterima dari server b'{"status": "OK", "user": ["lineker", "messi"]}\r\n\r\n'
end of string
Active users are : ["lineker", "messi"]
```

5. Fitur *Logout*

a. Deskripsi dan Tujuan

Fitur ini adalah *fitur* untuk *logout* agar *user* dapat keluar dari sistem. Jika *user* ingin menyudahi penggunaan sistem, maka *user* harus melakukan *logout* agar dapat dikeluarkan dari sistem. Setelah *user* melakukan *logout*, maka sistem akan menghapus *session id* dari *user* yang menggunakan sistem ini sebelumnya.

b. Parameter *Input*

Parameter *input* untuk fitur ini adalah :

logout

c. Hasil Keluaran dan Status Keluaran

Keluaran dari fitur ini adalah sistem akan mengembalikan status **OK** sekaligus menampilkan pesan **“User *user* has succesfully logged out”**.

d. Contoh *Message Input* dan Keluaran

Berikut adalah contoh keluaran dari fitur *logout* ini :

```
Command b4b6b4e4-226a-4461-b284-b60e688cee04:logout
diterima dari server b'{"status": "OK", "messages": "User lineker has succesfully logge'
diterima dari server b'd out"}\r\n\r\n'
end of string
"User lineker has succesfully logged out"
```