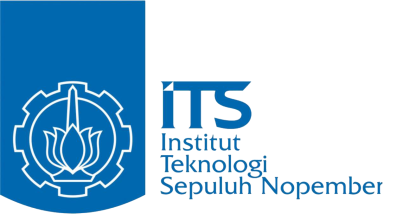
****

**LAPORAN TUGAS CHAT SERVER**

**PEMROGRAMAN JARINGAN E**

**OLEH :**

**KELOMPOK 5**

**RAHMA DINI MAGHFIROTUL L. (5114100027)**

**I GEDE PUTU NOBBY A. P. (5114100048)**

**NURHAMIDAH TYAS P. (5114100057)**

**ADENUAR PURNOMO (5114100079)**

**IRZAL AHMAD S. (5114100123)**

1. **CHATTING SERVER**
2. TUJUAN PEMBUATAN

Membuat server untuk aplikasi chatting dengan protokol yang di rancang sendiri. Aplikasi ini menyediakan fitur berupa chatting antar user, chatting grup, create grup, join grup.

1. RANCANGAN PROTOKOL
   1. REGISTER

Merupakan fitur yang digunakan user untuk mendaftar pertama kali untuk dapat menggunakan aplikasi chat server.

a.1. PARAMETER :

* + userid -> alfanumerik
  + pass -> alfanumerik

a.2. FORMAT :

REGISTER(spasi)userid(spasi)password

a.3 RETURN :

* SUKSES : kondisi ini menyatakan bahwa clien berhasil ter-registrasi

**+REG\_SUCCESS**

* GAGAL :kondisi ini menyatakan bahwa user tidak berhasil teregister karena userid sudah terpakai

**-USR\_REGISTERED**

(\*) terdapat return gagal jika syntax yang dmasukkan salah

* 1. LOGIN

Merupakan fitur yang harus dilakukan user sebelum menggunakan aplikasi chat server dengan cara mengisikan username dan password yang telah di dapatkan pada saat register.

b.1 PARAMETER

* userid -> alfanumerik
* pass -> alfanumerik

b.2 FORMAT

LOGIN(spasi)userid(spasi)password

b.3 RETURN

* SUKSES: **+LOGIN\_SUCCESS**
* GAGAL :
  + - kondisi yang menyatakan jika userid yang dimasukkan belum terdaftar **-USER\_NOT\_FOUND**- kondisi yang menyatakan jika password yang dimasukkan salah **-WRONG\_PASS**
  1. LOGOUT

Merupakan fitur yang disediakan untuk user agar dapat keluar dari aplikasi chat server.

c.1 PARAMETER

-

c.2 FORMAT

LOGOUT

c.3 RETURN

* + - * SUKSES : return yang menyatakan bahwa client berhasil logout **+LOGOUT**
  1. CREATE GROUP

Merupakan fitur yang tersedia di dalam aplikasi chat server untuk membuat grup yang dapat digunakan untuk berdiskusi antar anggotanya.

d.1 PARAMETER

-

d.2 FORMAT

CREATEGROUP(spasi)namagrup(spasi)password

d.3. RETURN

* + - * SUKSES : **+GROUP\_CREATED**
      * GAGAL :kondisi jika nama grup yang akan dibuat sudah terdaftar **-GROUP\_EXISTED**
  1. LIST GROUP

Merupakan fitur yang tersedia di dalam aplikasi chat server untuk melihat daftar grup yang terdaftar

e.1 PARAMETER

-

e.2 FORMAT

LISTGROUP

e.3 RETURN

* SUKSES: <<menampilkan list grup yang sudah ada>>
  1. JOIN GROUP

Merupakan fitur yang tersedia di dalam aplikasi chat server untuk bergabung dengan grup yang sudah terdaftar dalam aplikasi chat server.

f.1 PARAMETER

f.2. FORMAT

JOIN(spasi)nama\_grup(spasi)password

f.3. RETURN

* + - * SUKSES: **+JOINED\_SUCCESS**
      * GAGAL:kondisi yang menyatakan gagal untuk join grup karena jumlah anggota sudah penuh **-GROUP\_FULL** kondisi yang menyatakan bahwa grup ingin diikuti tidak ada **-GROUP\_NOT\_FOUND**kondisi yang menyatakan bahwa password yang dimasukka salah **-WRONG\_PASS**
  1. LIST USER

Merupakan fitur yang tersedia di dalam aplikasi chat server untuk melihat user yang aktif.

g.1. PARAMETER



g.2. FORMAT

LISTUSER(spasi)nama\_grup

g.3. RETURN

* + - * SUKSES : <<Muncul list user dalam grup>>
      * GAGAL: kondisi yang menyatakan bahwa grup tidak ada/belum terdaftar **-NO\_GROUP**
  1. PRIVATE MESSAGE

Merupakan fitur yang tersedia di dalam aplikasi chat server yang dapat digunakan user untuk mengirim pesan kepada user lain secara privat.

h.1. PARAMETER

h.2. FORMAT

PM (spasi) userid\_penerima (spasi) pesan

h.3. RETURN

* + - * SUKSES: **+MESSAGE\_SENT**
      * GAGAL:kondisi yang menyatakan bahwa userid\_penerima tidak ada/belum terdaftar **-USER\_NOT\_FOUND**
  1. GROUP MESSAGE

Merupakan fitur yang tersedia di dalam aplikasi chat server yang dapat digunakan user untuk mengirim pesan kepada user lain secara privat.

i.1. PARAMETER

-

i.2. FORMAT

GM(spasi)namagrup(spasi)pesan

i.3. RETURN

* + - * SUKSES : **+MESSAGE\_SENT**
      * GAGAL: kondisi yang menyatakan bahwa grup belum terdaftar **-GROUP\_NOT\_FOUND** kondisi yang menyatakan bahwa user belum join grup sehingga tidak bisa berkirim pesan **-GROUP\_NOT\_JOINED**

1. RECEIVE PRIVATE MESSAGE FORMAT

RPM(spasi)userID(spasi)pesan

1. RECEIVE GROUP MESSAGE FORMAT

GM(spasi)nama\_grup(spasi)userID(spasi)pesan

SYNTAX ERROR

Jika client memasukkan format yang tidak ada/salah maka akan tampil“**-SYNTAX\_ERROR**”. Tanpa tanda petikTanda (+) (-) pada return juga merupakan syntax

1. Code Server.py
2. #python version: Python 3.5 for Windows
3. **import** socket
4. **import** sys
5. **import** thread
7. sock=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)
9. #menerima koneksi dari semua interface
10. server\_address=('',10000)
11. client\_list=list()
12. auth\_list={}
13. logged={}
15. **def** send\_to\_all\_klien(message):
16. **for** koneksi **in** client\_list:
17. **try**:
18. koneksi.sendall(message)
19. **except**:
20. koneksi.close()
21. client\_list.remove(koneksi)
23. **def** send\_message(koneksi,message):
24. **try**:
25. koneksi.sendall(message)
26. **except**:
27. **print**("Send error")
29. **def** load\_user():
30. f=open("data.dat","r")
32. **for** line **in** f:
33. **try**:
34. line=line.splitlines()
35. #print line[0]
36. tmp=line[0].strip().split(":")
37. #print tmp
38. auth\_list[tmp[0]]=tmp[1]
39. **except**:
40. **pass**
42. **def** register(koneksi,arg):
43. f=open("data.dat","a")
44. str\_user=arg[1]
45. str\_pass=arg[2]
47. **if** auth\_list.get(str\_user,0)==0:
48. str\_auth=str\_user+":"+str\_pass
49. f.write(str\_auth)
50. f.close
51. auth\_list[str\_user]=str\_pass
52. koneksi.sendall("+REG\_SUCCESS")
53. **return**
54. **else**:
55. koneksi.sendall("-USR\_REGISTERED")
56. f.close()
58. **def** login(koneksi,arg):
59. str\_user=arg[1]
60. str\_pass=arg[2].strip()
62. **if** logged.get(str\_user,0)!=0:
63. koneksi.sendall("+ALREADY\_LOGGED\_IN");
64. **return** str\_user
66. pass\_check=auth\_list.get(str\_user,0)
68. **if** pass\_check!=0:
69. **if** str\_pass==pass\_check:
70. koneksi.sendall("+LOGIN\_SUCCESS")
71. logged[str\_user]=koneksi
72. **return** str\_user
73. **else**:
74. koneksi.sendall("-WRONG\_PASS")
75. **return** 0
76. **else**:
77. koneksi.sendall("-USER\_NOT\_FOUND")
78. **return** 0
80. **def** check\_logged(userid):
81. **if** logged.get(userid,0)==0:
82. **return** 0
83. **else**:
84. **return** 1
86. **def** check\_is\_me\_logged(koneksi,userid):
87. **if** logged.get(userid,0)==0:
88. koneksi.sendall("-YOU\_NOT\_LOGGED\_IN")
89. **return** 0
90. **else**:
91. **return** 1
93. **def** pm(koneksi,arg,userid):
94. **if** check\_is\_me\_logged(koneksi,userid)==0:
95. **return**
97. sender=userid
98. receiver=arg[1]
99. message="RPM "+sender+" "+arg[2]
101. **if** check\_logged(receiver)==0:
102. koneksi.sendall("-USER\_NOT\_FOUND")
103. **return** 0
105. koneksi=logged[receiver]
106. send\_message(koneksi,message)
108. **def** koman(koneksi,arg,userid=0):
109. arg=arg.split(" ",2)
111. **if** arg[0]=="REGISTER":
112. register(koneksi,arg)
113. **elif** arg[0]=="LOGIN":
114. login(koneksi,arg)
115. **elif** arg[0]=="PM":
116. pm(koneksi,arg,userid)
118. #------------------------------------------------------------------------------
119. **def** klien(koneksi):
120. userid=0
122. **while** True:
123. data = koneksi.recv(1024)
124. **try**:
125. **if** data:
126. arg=data.split(" ")
127. **if** arg[0]=="LOGIN":
128. userid=login(koneksi,arg)
130. **if** userid!=0:
131. **break**
132. **else**:
133. **continue**
135. koman(koneksi,data)
136. **else**:
137. **break**
138. **except**:
139. koneksi.close()
140. **if** userid!=0:
141. **del** logged[userid]
142. **break**
144. **while** True:
145. **try**:
146. data = koneksi.recv(1024)
147. **if** data:
148. koman(koneksi,data,userid)
149. **else**:
150. **break**
151. **except**:
152. koneksi.close()
153. **break**
155. sock.bind(server\_address)
156. sock.listen(1)
158. load\_user()
160. **print** auth\_list
162. **while** True:
163. koneksi,addr=sock.accept()
164. client\_list.append(koneksi)
165. thread.start\_new\_thread(klien,(koneksi,))
167. koneksi.close()
168. sock.close()
169. Pembagian Kerja

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama** | **NRP** | **Fitur yang dibuat** |
| 1 | Rahma Dini M. Laily | 5114100027 | Register |
| Login |
| 2 | I Gede Putu Nobby A.P. | 5114100048 | Private Message |
| Group Message |
| Create Group |
| 3 | Nurhamidah Tyas P. | 5114100057 | Logout |
| Join grup |
| 4 | Adenuar Purnomo | 5114100079 | List User |
| List grup |
| 5 | Irzal Ahmad S. | 5114100123 | Receive private message |
| Receive group message |

1. **CHATTING CLIENT**
2. TUJUAN PEMBUATAN

Membuat client untuk aplikasi chatting sesuai dengan protokol yang di rancang oleh kelompok 4 kelas pemrograman jaringan E. Aplikasi ini menyediakan fitur berupa chatting antar user, chatting grup, create grup, join grup.

1. DRAFT PROTOCOL
   1. REGISTRASI

Fitur registrasi merupakan fitur pertama kali yang harus dilakukan dalam aplikasi ini yaitu mendaftarkan diri sebagai anggota aplikasi chat server ini, ketika mendaftar maka user akan memiliki username dan password yang berguna untuk dapat masuk ke dalam aplikasi ini.

a.1. COMMAND

REGIST<spasi>USERNAME<spasi>PASSWORD

a.2. PARAMETER

* USERNAME : alfanumerik (tanpa spasi)
* PASSWORD : alfanumerik (tanpa spasi)

a.3. RETURN

* 101 : registrasi berhasil
* 102 : registrasi gagal username sudah ada
* 103 : registrasi gagal kesalahan format
* 101 : REGIST [USERNAME] OK (untuk informasi di server)
* 001 : syntax error
* 003 : failed karena sudah login
  1. LOGIN

Fitur login merupakan fitur kedua yang wajib dijalankan dalam aplikasi ini sebelum masuk ke dalam sistem. User yang telah melakukan registrasi harus melakukan login dengan menggunakan username dan password yang dimilikinya.

b.1. COMMAND

LOGIN<spasi>USERNAME<spasi>PASSWORD

b.2. PARAMETER

* USERNAME : alfanumerik (tanpa spasi)
* PASSWORD : alfanumerik (tanpa spasi)

b.3. RETURN

* 201 : berhasil masuk
* 202 : login gagal username/password salah
* 201 LOGIN [USERNAME] OK (informasi dalam sisi server)
* 001 : syntax error
* 003 : anda sudah login
  1. CHAT PERSONAL

Fitur ini merupakan salah satu fitur yang dapat digunakan untuk mengirimkan pesan personal kepada user yang sedang aktif di dalam aplikasi chat ini. Fitur ini hanya dapat dijalankan ketika user telah masuk ke dalam sistem.

c.1. COMMAND

SENDP<spasi>USERNAME\_PENERIMA<spasi>MESSAGE

c.2. PARAMETER

* USERNAME\_PENERIMA : alfanumerik (tanpa spasi)
* MESSAGE : pesan (256 character)

c.3. RETURN

* 301 : pesan personal terkirim
* 301 SENDP [USERNAME] OK (tampilan dalam sisi server)
* 302 : gagal karena username penerima tidak ada
* 303 : gagal karena username sedang tidak online
* 304 : gagal karena message kosong
* 001 : command salah
* 002 : belum login
  1. CREATE GROUP

Fitur ini merupakan fitur untuk membuat group chat di dalam aplikasi chat ini. Ketika grup dibuat oleh salah satu user maka otomatis user yang membuat grup tersebut akan langsung tergabung ke dalam group chat yang dibuatnya.

d.1. COMMAND

CREATE<spasi>NAMA\_GROUP

d.2. PARAMETER

NAMA\_GROUP : alfanumerik (tanpa spasi)

d.3. RETURN

* 401 : grup berhasil dibuat
* 401 [GROUP] BY [USERNAME] CREATED (tampilan dalam sisi server)
* 402 : gagal karena nama group sudah ada
* 403 : gagal karena kesalahan format parameter
* 001 : command salah
* 002 : anda belum login
  1. JOIN GROUP

Fitur ini merupakan fitur lanjutan ketika group chat sudah ada di dalam sistem. User baru yang melakukan login dapat bergabung ke dalam group chat yang ada dengan menggunakan command seperti di bawah ini.

e.1. COMMAND

JOIN<spasi>NAMA\_GROUP

e.2. PARAMETER

NAMA\_GROUP : alfanumerik (tanpa spasi)

e.3. RETURN

* 501 : berhasil gabung group
* 501 [USERNAME] JOIN [GROUP] (informasi di tampilan server)
* 502 : gagal karena group tidak ada
* 503 : gagal join karena sudah join grup yang dituju
* 001 : command salah
* 002 : anda belum login
  1. LEAVE GROUP

User yang telah tergabung ke dalam suatu group chat dapat meninggalkan group tersebut dengan menggunakan fitur leave group. Nantinya ketika user telah meninggalkan group tersebut maka setiap pesan yang ada dalam group tersebut tidak akan dikirimkan ke user yang sudah meninggalkan group tersebut.

f.1. COMMAND

LEAVE<spasi>NAMA\_GROUP

f.2. PARAMETER

NAMA\_GROUP : alfanumerik (tanpa spasi)

f.3. RETURN

* 601 : berhasil keluar group
* 601 [USERNAME] LEAVE [GROUP] (tampilan dalam sisi server)
* 602 : gagal karena anda belum masuk group
* 603 : gagal karena group tidak ada
* 001 : command salah
* 002 : anda belum login
  1. CHAT GROUP

Ketika group chat sudah ada dan memiliki beberapa member maka, member yang tergabung dalam suatu group chat dapat mengirimkan pesan kepada group tersebut yang nantinya pesan tersebut akan diteruskan ke tiap user yang tergabung dalam suatu group chat tersebut.

g.1. COMMAND

SENDG<spasi>NAMA\_GROUP<spasi>MESSAGE

g.2. PARAMETER

* NAMA\_GROUP : alfanumerik (tanpa spasi)
* MESSAGE : pesan (256 character)

g.3. RETURN

* 701 : SENDG OK pesan terkirim
* 701 [NAMAGRUP] [PENGIRIM] OK
* 702 : SENDG FAILED group tidak ada
* 703 : SENDG FAILED message kosong
* 704 : SENDG FAILED user belum tergabung
* 001 : FAILED command salah
* 002 : FAILED anda belum login
  1. LIST USER AKTIF

Fitur ini berfungsi untuk menampilkan daftar user yang sedang login di dalam sistem yang nantinya fitur ini berhubungan untuk mengirimkan pesan personal. Hal ini karena tanpa adanya fitur ini maka user akan sulit untuk mengetahui user mana saja yang sedang aktif dan dapat dikirimi pesan.

h.1. COMMAND

LISTUS

h.2. PARAMETER

-

h.3. RETURN

* 003 : menampilkan daftar user yang sedang aktif
* 001 : command salah
* 002 : anda belum login
  1. LIST GROUP

Fitur ini berfungsi untuk menampilkan daftar grup yang telah dibuat di dalam sistem ini. Fitur ini memudahkan user untuk bergabung ke dalam suatu grup chat di dalam sistem.

i.1. COMMAND

LISTGR

i.2. PARAMETER

-

i.3. RETURN

* 004 : menampilkan list group yang ada
* 001 : command salah
* 002 : belum login
  1. CHAT PUBLIC/ BROADCAST

Chat public merupakan fitur yang berfungsi untuk mengirimkan pesan seperti broadcast message ke semua user yang sedang login ke dalam sistem.

j.1. COMMAND

SEND<spasi>PUBLIC<spasi>MESSAGE

j.2. PARAMETER

MESSAGE : pesan (256 character)

j.3. RETURN

* 801 : pesan public/broadcast terkirim
* 801 BC BY [USERNAME] SENT (informasi tampilan di sisi server)
* 803 : gagal karena message kosong
* 001 : command salah
* 002 : anda belum login

k. LOGOUT

Fitur ini merupakan fitur terakhir di dalam aplikasi chat ini dimana ketika pengguna berada di dalam sistem maka ia dapat keluar atau logout dari aplikasi chat tersebut dan koneksinya akan terputus.

k.1. COMMAND

LOGOUT

k.2. PARAMETER

-

k.3. RETURN

* 203 : berhasil logout
* 203 [USERNAME] LOGOUT (tampilan dalam sisi server)
* 001 : command salah
* 002 : belum login

1. CODE CLIENT.PY
2. #!/usr/bin/env python
4. **import** npyscreen, curses
5. **import** thread
6. **import** socket
7. **import** sys
8. **from** thread **import** \*
9. **import** os
10. **import** time
12. sock = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

15. server\_address = ('10.151.36.250', 10040)
17. sock.connect(server\_address)
18. nama=''
19. err\_list={}
21. err\_list['501']="berhasil gabung group"
22. err\_list['502']="gagal karena group tidak ada"
23. err\_list['503']="gagal join karena sudah join grup yg dituju"
24. err\_list['401']="grup berhasil dibuat"
25. err\_list['402']="gagal karena nama group sudah ada"
26. err\_list['403']="gagal karena kesalahan format parameter"
27. err\_list['301']="pesan personal terkirim"
28. err\_list['302']="gagal karena username penerima tidak ada"
29. err\_list['303']="gagal karena username sedang tidak online"
30. err\_list['304']="gagal karena message kosong"
31. err\_list['001']="command salah"
32. err\_list['002']="anda belum login"
33. err\_list['003']="anda sudah login"
34. err\_list['004']="menampilkan list grup yang ada"
35. err\_list['101']="registrasi berhasil"
36. err\_list['102']="registrasi gagal username sudah ada"
37. err\_list['103']="registrasi gagal kesalahan format"
38. err\_list['201']="berhasil masuk"
39. err\_list['202']="login gagal username/password salah"
40. err\_list['301']="pesan personal terkirim"
41. err\_list['302']="gagal karena username penerima tidak ada"
42. err\_list['303']="gagal karena username sedang tidak online"
43. err\_list['304']="gagal karena message kosong"
44. err\_list['401']="grup berhasil dibuat"
45. err\_list['402']="gagal karena nama group sudah ada"
46. err\_list['403']="gagal karena kesalahan format parameter"
47. err\_list['501']="berhasil gabung group"
48. err\_list['502']="gagal karena group tidak ada"
49. err\_list['503']="gagal join karena sudah join grup yg dituju"
50. err\_list['601']="berhasil keluar group"
51. err\_list['602']="gagal karena anda belum masuk group"
52. err\_list['603']="gagal karena group tidak ada"
53. err\_list['701']="SENDG OK pesan terkirim"
54. err\_list['702']="SENDG FAILED grup tidak ada"
55. err\_list['703']="SENDG FAILED message kosong"
56. err\_list['704']="SENDG FAILED user belum tergabung"
57. err\_list['801']="pesan public/broadcast terkirim"
58. err\_list['803']="gagal karena message kosong"
59. err\_list['005']="EndList"
61. **class** MyTestApp(npyscreen.NPSAppManaged):
62. **def** onStart(self):
63. #self.keypress\_timeout\_default = 1
64. self.a=self.addForm("MAIN", mainform)
65. self.res=""

68. **def** while\_waiting(self):
69. **pass**
70. #        print "lala"


74. **class** mainform(npyscreen.ActionForm):
75. **def** create(self):
76. self.keypress\_timeout\_default = 1
77. #super(mainform, self).create()
78. self.resp = self.add(npyscreen.MultiLineEdit, max\_height=7)
79. self.usrid=self.add(npyscreen.TitleText, name = "UserId:", value="")
80. self.grpid=self.add(npyscreen.TitleText, name = "GroupId:", value="")
81. self.mes=self.add(npyscreen.TitleText, name = "Message:", value="")
82. self.sendbutton = self.add(npyscreen.ButtonPress, name='Send',rely=13)
83. self.sendgbutton = self.add(npyscreen.ButtonPress, name='Send To Group',rely=13,relx=9)
84. self.regbutton = self.add(npyscreen.ButtonPress, name='Register',rely=16)
85. self.logbutton = self.add(npyscreen.ButtonPress, name='Login',rely=16,relx=13)
86. self.lsuserbutton = self.add(npyscreen.ButtonPress, name='List User',rely=16,relx=20)
87. self.logoutbutton = self.add(npyscreen.ButtonPress, name='Logout',rely=16,relx=31)
88. self.cgroupbutton = self.add(npyscreen.ButtonPress, name='Create Group',rely=18)
89. self.jgroupbutton = self.add(npyscreen.ButtonPress, name='Join Group',rely=18,relx=17)
90. self.lgroupbutton = self.add(npyscreen.ButtonPress, name='Leave Group',rely=18,relx=30)
91. self.lsgroupbutton = self.add(npyscreen.ButtonPress, name='List Group',rely=18,relx=44)
93. self.regbutton.whenPressed = self.register
94. self.logbutton.whenPressed = self.login
95. self.lsuserbutton.whenPressed = self.listuser
96. self.sendbutton.whenPressed = self.pm
97. self.cgroupbutton.whenPressed = self.creategroup
98. self.jgroupbutton.whenPressed = self.joingroup
99. self.lgroupbutton.whenPressed = self.leavegroup
100. self.lsgroupbutton.whenPressed = self.listgroup
101. self.sendgbutton.whenPressed = self.sendgroup
102. self.logoutbutton.whenPressed = self.logout
103. thread.start\_new\_thread(self.receive,())
105. **def** receive(self):
106. **while** True:
107. data = sock.recv(1024)
108. **if** err\_list.get(data.strip(),0)!=0:
109. time.sleep(1)
110. npyscreen.notify\_wait(err\_list[data.strip()], title='Warning')
111. time.sleep(0.5)
112. #            print ps\_error
113. **else**:
114. self.resp.value=self.resp.value+data.decode()
115. self.resp.display()
117. **def** frmreg(\*args):
118. F = npyscreen.Form(name='Register')
119. usr=F.add(npyscreen.TitleText, name="Username: ")
120. pas=F.add(npyscreen.TitleText, name="Password: ")
121. F.edit()
122. **return** (usr.value,pas.value)
124. **def** frmcgroup(\*args):
125. F = npyscreen.Form(name='Create Group')
126. groupname=F.add(npyscreen.TitleText, name="Group Name: ")
127. F.edit()
128. **return** groupname.value
130. **def** frmjgroup(\*args):
131. F = npyscreen.Form(name='Join Group')
132. groupname=F.add(npyscreen.TitleText, name="Group Name: ")
133. F.edit()
134. **return** groupname.value
136. **def** frmlgroup(\*args):
137. F = npyscreen.Form(name='Leave Group')
138. groupname=F.add(npyscreen.TitleText, name="Group Name: ")
139. F.edit()
140. **return** groupname.value
142. **def** frmlog(\*args):
143. F = npyscreen.Form(name='Login')
144. usr=F.add(npyscreen.TitleText, name="Username: ")
145. pas=F.add(npyscreen.TitleText, name="Password: ")
146. F.edit()
147. **return** (usr.value,pas.value)
149. **def** pm(self):
150. **if** (self.mes.value!=''):
151. msg=""
152. **if**(self.usrid.value!=''):
153. msg="SENDP " + self.usrid.value+ " " + self.mes.value
154. **else**:
155. msg="SEND " + "PUBLIC " + self.mes.value
157. sock.sendall(msg)
159. **def** sendgroup(self):
160. **if** (self.mes.value!='' **and** self.grpid.value!=''):
161. msg="SENDG " + self.grpid.value + " " + self.mes.value
162. sock.sendall(msg)
164. **def** logout(self):
165. sock.sendall("LOGOUT")
167. **def** listuser(self):
168. sock.sendall("LISTUS")
170. **def** listgroup(self):
171. sock.sendall("LISTGR")
173. **def** creategroup(self):
174. groupname=npyscreen.wrapper\_basic(self.frmcgroup)
175. sock.sendall("CREATE "+groupname)
177. **def** joingroup(self):
178. groupname=npyscreen.wrapper\_basic(self.frmjgroup)
179. sock.sendall("JOIN "+groupname)
181. **def** leavegroup(self):
182. groupname=npyscreen.wrapper\_basic(self.frmlgroup)
183. sock.sendall("LEAVE "+groupname)
185. **def** login(self):
186. (usr,pas)=npyscreen.wrapper\_basic(self.frmreg)
187. msg="LOGIN "+usr+" "+pas
188. sock.sendall(msg)
190. **def** register(self):
191. (a,b)=npyscreen.wrapper\_basic(self.frmreg)
192. msg="REGIST "+a+" "+b
193. sock.sendall(msg)

196. **def** while\_waiting(self):
197. **pass**
199. **def** display\_mainform():
200. mf=MyTestApp()
201. **print** mf
202. mf.run()
203. **return** 0
205. **def** main():
206. display\_mainform()

209. **if** \_\_name\_\_ == '\_\_main\_\_':
210. main()
212. PEMBAGIAN KERJA

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama** | **NRP** | **Fitur yang dibuat** |
| 1 | Rahma Dini M. Laily | 5114100027 | Registrasi |
| Login |
| 2 | I Gede Putu Nobby A.P. | 5114100048 | Chat Personal |
| Create group |
| Join Group |
| 3 | Nurhamidah Tyas P. | 5114100057 | List group |
| List user aktif |
| 4 | Adenuar Purnomo | 5114100079 | Chat Group |
| Chat public/broadcast |
| 5 | Irzal Ahmad S. | 5114100123 | Logout |
| Leave group |