Chat Server Test Document

Contents

[1. Functional Test Case 3](#_Toc444081712)

[1.1. Connect to Server 3](#_Toc444081713)

[1.2. Close connection of server 3](#_Toc444081714)

[1.3. Input nick name 3](#_Toc444081715)

[1.4. Input duplicated nick name 3](#_Toc444081716)

[1.5. Join Room 3](#_Toc444081717)

[1.6. Display Rooms 3](#_Toc444081718)

[1.7. Send Message 4](#_Toc444081719)

[1.8. Receive Message 4](#_Toc444081720)

[1.9. Receive User entered Room Message 4](#_Toc444081721)

[1.10. Leave Room 4](#_Toc444081722)

[1.11. Receive User left Room Message 4](#_Toc444081723)

[1.12. Unknown command 5](#_Toc444081724)

[1.13. Repeat join same room 5](#_Toc444081725)

[2. Performance Test Case 6](#_Toc444081726)

[2.1. One room maximum users support 6](#_Toc444081727)

[2.2. Maximum rooms support 6](#_Toc444081728)

[2.3. Message delay limitation 6](#_Toc444081729)

[3. Stability Test Case 6](#_Toc444081730)

[3.1. Server stability time 6](#_Toc444081731)

# Functional Test Case

## Connect to Server

1. Step :
   1. For Linux input telnet address port
2. Expectation: Can connect to server

## Close connection of server

1. Precondition: TC 1.1 passed
2. Step:
   1. input “CTRL + C” or “/quit”
3. Expectation : Server can close connection

## Input nick name

1. Precondition: TC1.1 passed
2. Step:
   1. input a nick name “tester”
3. Expectation : show “ Welcome tester”

## Input duplicated nick name

1. Precondition: TC1.3 passed
2. Step:
   1. Open new terminal.
   2. telnet ip port
   3. input nick name “tester”
3. Expectation: Sorry, name take.

## Join Room

1. Precondition: TC1.3 passed
2. Step:
   1. Input /join test\_room
3. Expectation: “entering root: test\_room” and user list

## Display Rooms

1. Precondition: TC 1.5 passed
2. Step:
   1. /rooms
3. Expectation: room list( at least exist test\_room)

## Send Message

1. Precondition: TC1.5 passed
2. Step:
   1. Input “ test message”
3. Expectation: no error.

## Receive Message

1. Step:
   1. Open first terminal by telnet address port
   2. Input nick name “test1”
   3. /join room1
   4. Open Second terminal by telnet address port
   5. Input nick name “test2”
   6. /join room1
   7. Input “ test2 send test message”
2. Expectation: in first terminal, you should see” [test2] :test2 send test message”

## Receive User entered Room Message

1. Step:
   1. Open first terminal by telnet address port
   2. Input nick name “test1”
   3. /join room1
   4. Open Second terminal by telnet address port
   5. Input nick name “test2”
   6. /join room1
2. Expectation: in first terminal, you should see” new user entered chat room room1”

## Leave Room

1. Step:
   1. Open first terminal by telnet address port
   2. Input nick name “test1”
   3. /leave
2. Expectation: in first terminal, you should see” User has left chat test1 (\*\* this is you)”

## Receive User left Room Message

1. Step:
   1. Open first terminal by telnet address port
   2. Input nick name “test1”
   3. /join room1
   4. Open Second terminal by telnet address port
   5. Input nick name “test2”
   6. /join room1
   7. /leave
2. Expectation: in first terminal, you should see “User has left chat test1”

## Unknown command

1. Precondition: TC1.3 passed
2. Step:
   1. Input “ /room”
3. Expectation: “Unknown command :[/join /leave /quit /rooms]

## Repeat join same room

1. Precondition: TC1.5 passed
2. Step:
   1. Input “ /join test\_room”
3. Expectation: You have in this room

# Performance Test Case

## One room maximum users support

Target 200 users

## Maximum rooms support

Unlimited, test 200 rooms, works well.

## Message delay limitation

2 Cores 1.6G 4G memory 2Mb bandwidths.

When connect more than 100 concurrent users send message, sometimes TCP ACK lost..

# Stability Test Case

## Server stability time

Until now: 3\* 24 hour.