

IT TESTING REPORT

IT_TEST_CASE-01 : Client started before the server

```
Jonas Edward@LAPTOP-3BICFCE0 ~/InstantChatters/CUT/Code/bin
$ ./client.exe 8080
Please enter your name: Jonas
ERROR: connect
```

IT_TEST_CASE-02: When number of arguments is less than 2 , User has not provided port number for server or client.

```
USER@DESKTOP-DI40DVI ~/I
$ ./server.exe
Usage: ./server <port>

USER@DESKTOP-DI40DVI ~/I
$ ./client.exe
Usage: ./client <port>
```

IT_TEST_CASE-03: When User provides port number at the time of execution.

```
USER@DESKTOP-DI40DVI ~/InstantCha
$ ./server.exe 8080
=== WELCOME TO THE CHATROOM ===

USER@DESKTOP-DI40DVI ~/InstantCha
$ ./client.exe 8080
Please enter your name: Licency
```

IT_TEST_CASE-04: Writing failed because there are no processes which could read from the pipe.

```
ERROR: write to descriptor failed: Broken pipe
ERROR: -1
```

IT_TEST_CASE-05: If the socket is not properly closed or the address is already in use

```
USER@DESKTOP-DI40DVI ~/InstantChatters/CUT/Code/bin
$ ./server.exe 8080
ERROR: Socket binding failed: Address already in use
```

IT_TEST_CASE-06: If number of clients exceeded MAX_CLIENTS

```
$ ./server.exe 8000
=== WELCOME TO THE CHATROOM ===

Active users list is from queue Add :
Max clients reached. Rejected: 127.0.0.1:43736
```

IT_TEST_CASE-07: Taking note of joining and leaving users.

```
Jonas Edward@LAPTOP-3BICFCE0 ~/InstantChatters/CUT/Code/src
$ ./a.exe 8080
=== WELCOME TO THE CHATROOM ===

Active users are :
clients credential = Jonas : Edward

Active users are : Jonas
clients credential = Laxmikanth: 1234

Jonas has left

Active users are : Laxmikanth
Laxmikanth has left

Active users are :
Aborted (core dumped)
```

IT_TEST_CASE-08: Chat history of the active server session will be stored in the logfile.txt file

```
$ ./client.exe 8080
Please enter your name: user11
=== WELCOME TO THE CHATROOM ===
> Enter the pass now user11 :> pass11
> user11 are you a new or old user? Please give 'o' or 'n' only : > n
> user11 Thanks for credentials , Enjoy chatting !!!
> How are you??
> user15: I am fine.
> user15: Thank you for asking
> Most welcome
>
```

```
$ ./client.exe 8080
Please enter your name: user15
=== WELCOME TO THE CHATROOM ===
> Enter the pass now user15 :> pass15
> user15 are you a new or old user? Please give 'o' or 'n' only :
> n
> user15 Thanks for credentials , Enjoy chatting !!!
> user11: How are you??
> I am fine.
> Thank you for asking
> user11: Most welcome
```

```
$ cat logfile.txt
user2: hey
user1: hello
user2: how are you
user1: Good
user1: And what about you?
user2: I'm good.
user2: Ok. Talk to you later
user1: Bye
user3: Good to see you
user4: Yeah
user5: Hello again
user4: Hii all
user3: How is everything going?
user5: Everything is going on smoothly
user4: Good to know
user55: Hey user20
user20: Hello bro
user55: Having fun?
user20: Yeah! very much
user11: How are you??
user15: I am fine.
user15: Thank you for asking
user11: Most welcome
```

IT TEST CASES

Module	Test Case Id	Test Description	Expected Result	Traceability	Test Situation
Client Module	IT_TEST_CASE-01	Client started before the server	Program should exit and throws a connect error	Req ICS_10	Pass
	IT_TEST_CASE-02	When number of arguments is less than 2	Usage error : ./server <port>	Req ICS_10	Pass
	IT_TEST_CASE-03	When number of arguments is equal to 2	Program should run further	Req ICS_10	Pass
Server Module	IT_TEST_CASE-04	Writing failed because there are no processes which could read from the pipe.	Prints "Writer to descriptor failed: Broken pipe", exits from the program	Req ICS_04	Pass
	IT_TEST_CASE-05	If the socket is not properly closed or the address is already in use	Prints "ERROR: Socket binding failed", exits from program	Req ICS_01	Pass
	IT_TEST_CASE-06	If number of clients exceeded MAX_CLIENTS	Prints "ERROR: Maximum clients reached" , exits from the program	Req ICS_01	Pass
	IT_TEST_CASE-07	Taking note of joining and leaving users	prints "Active users are :xyz" , "abc has left"	Req ICS_03	Pass
	IT_TEST_CASE-08	Chat history of the active server session will be stored in the logfile.txt file	logfile.txt is filled with chat history	Req ICS_09	Pass

UT TESTING REPORT

LOGIN_001 : Valid name

```
void testSunnyCases1(void)
{
    CU_ASSERT_EQUAL(validName("user1"), 1);
    CU_ASSERT_EQUAL(validName("user12"), 1);
}
```

LOGIN_002, LOGIN_003 : Invalid name

```
void testRainyCases1(void)
{
    CU_ASSERT_EQUAL(validName("u"), 0);
    CU_ASSERT_EQUAL(validName("Jonas Laxmi Licency Sukanya Vibhu"), 0);
}
```

LOGIN_004 : Valid password

```
void testSunnyCases2(void)
{
    CU_ASSERT_EQUAL(validPass("pass1"), 1);
    CU_ASSERT_EQUAL(validPass("Pass123"), 1);
}
```

LOGIN_005, LOGIN_006 : Invalid password

```
void testRainyCases2(void)
{
    CU_ASSERT_EQUAL(validPass("p"), 0);
    CU_ASSERT_EQUAL(validPass("pppppaaaaaasssssss112225544kkk1"), 0);
}
```

LOGIN_007 : Validate user

```
void testSunnyCases3(void)
{
    CU_ASSERT_EQUAL(testValid('n', "user44", "passdd"), 1);
    CU_ASSERT_EQUAL(testValid('o', "user55", "pass77"), 1);
}
```

```
void testRainyCases3(void)
{
    CU_ASSERT_EQUAL(testValid('p', "usertt", "passgg"), 0);
    CU_ASSERT_EQUAL(testValid('z', "useryy", "passyy"), 0);
}
```

LOGIN_008 : New user

```
void testSunnyCases4(void)
{
    CU_ASSERT_EQUAL(newUser('n', "user11", "pass33"), 1);
}
```

LOGIN_009 : Old user

```
void testRainyCases4(void)
{
    CU_ASSERT_EQUAL(newUser('o', "user22", "pass33"), 0);
}
```

ADD_001 : Adding to queue

```
void testSunnyCases5(void)
{
    CU_ASSERT_EQUAL(queue_add("user11"), 1);
    CU_ASSERT_EQUAL(queue_add("user30"), 1);
}
```

REMOVE_001 : Removing users from the list

```
void testSunnyCases6(void)
{
    CU_ASSERT_EQUAL(queue_remove("user11"), 1);
    CU_ASSERT_EQUAL(queue_remove("user30"), 1);
}
```

FAIL_001 : Login Success/Failure

```
void testSunnyCases7(void)
{
    CU_ASSERT_EQUAL(loginSuccess("user12", "pass11"), 1);
    CU_ASSERT_EQUAL(loginSuccess("user3", "pass30"), 1);
}
```

FAIL_002 : Login Success/Failure

```
void testRainyCases7(void)
{
    CU_ASSERT_EQUAL(loginSuccess("user22", "pass11"), 0);
    CU_ASSERT_EQUAL(loginSuccess("userzzz", "pzzzz"), 0);
}
```

RESULT :

```
CUnit - A unit testing framework for C - Version 2.1-3
http://cunit.sourceforge.net/

Suite: Name valid or invalid
  Test: test of validName() in normal cases ...passed
  Test: test of invalidName() in error cases ...passed
Suite: Password valid or invalid
  Test: test of validPass() in normal cases ...passed
  Test: test of invalidPass() in error cases ...passed
Suite: User Validation
  Test: test of userValidation() in normal cases ...passed
  Test: test of userValidation() in error cases ...passed
Suite: New user
  Test: test of newUser() in normal cases ...passed
  Test: test of newUser() in error cases ...passed
Suite: Add to active user list
  Test: test of addActiveUser() in normal cases ...passed
Suite: Remove user from active user list
  Test: test of removeUser() in normal cases ...passed
Suite: Login failure
  Test: test of loginSuccessful() in normal cases ...passed
  Test: test of loginFailure() in error cases ...passed

Run Summary:
  Type      Total   Ran   Passed   Failed   Inactive
  suites      7      7      n/a      0      0
  tests     12     12     12      0      0
  asserts    22     22     22      0     n/a

Elapsed time = 0.000 seconds
```

UT TEST CASES

Test Case Id	Step	Preconditions / Dependencies	Scenario Description / Test Case	Test Data	Expected Result	Actual Result (Iteration I)	Status (Pass or Fail)	Bug Severity	Comments/Defect No.	Actual Result (Iteration II)	Status (Pass or Fail)
LOGIN_001	Enter your name	Server should be run first	User enters username of length within 2 to 30 characters	username ="user12"	Password must be less than 30 and more than 2 characters.	Password must be less than 30 and more than 2 characters.	Pass	No		Password must be less than 30 and more than 2 characters.	Pass

LOGIN_002	Enter your name	Server should be run first	User enters the username of length greater than 30 characters	username="Jonas Laxmi Licency Sukanya Vibhu"	Name must be less than 30 and more than 2 characters	Name must be less than 30 and more than 2 characters	Fail	Minor	Syntax error	Name must be less than 30 and more than 2 characters	Pass
LOGIN_003	Enter your name	Server should be run first	User enters the username of length less than 2 characters	username="u"	Name must be less than 30 and more than 2 characters	Name must be less than 30 and more than 2 characters	Pass	No		Name must be less than 30 and more than 2 characters	Pass
LOGIN_004	Enter password	Valid username should be entered	User enters password of length within 2 to 30 characters	password="pass123"	Password must be less than 30 and more than 2 characters	Password must be less than 30 and more than 2 characters	Pass	No		Password must be less than 30 and more than 2 characters	Pass
LOGIN_005	Enter password	Valid username should be entered	User enters password greater than 30 characters	password="ppppppaa aaassssss 11444kkjjj"	Password must be less than 30 and more than 2 characters.	Password must be less than 30 and more than 2 characters.	Pass	No		Password must be less than 30 and more than 2 characters.	Pass
LOGIN_006	Enter password	Valid username should be entered	User enters password less than 2 characters	password="p"	Password must be less than 30 and more than 2 characters.	Password must be less than 30 and more than 2 characters.	Pass	No		Password must be less than 30 and more than 2 characters.	Pass
LOGIN_007	Validate users	Valid username and password should be entered	User enters valid credentials	Is new = "n" , Is old = "o" , Invalid input = "p"/"z", along with username & password	User will be added to the list	User will be added to the list	Fail	Minor	Unable to take input	User will be added to the list	Pass
LOGIN_008	New user	Check for new or old	User enters	Is new = "n" ,	User will be added to	User will be added to	Pass	No		User will be added to	Pass

		user	valid credentials	Invalid input = "p"/"z" alongwith username & password	logfile	logfile				logfile	
LOGIN_009	New user	Check for new or old user	User enters valid credentials	Is old = "o", Invalid input = "p"/"z" alongwith username & password	User will not be added to logfile	User will not be added to logfile	Pass	No		User will not be added to logfile	Pass
ADD_001	Adding to the active list of users	User should be inside chatroom	User equips a place in structure array	Username ="user1", Username ="user30"	User will be added to the structure array	User added to structure array	Pass	No		User added to structure array	Pass
REMOVE_001	Removing from the active list of users	User should not be present inside chatroom	User gets removed from structure array	Username ="user1", Username ="user30"	User will be removed from the structure array	User removed from structure array	Pass	No		User removed from structure array	Pass
FAIL_001	Login failure	User should enter correct password	Registered user enters correct password	Username ="user12" //valid, Password ="pass11" //valid	User is able to connect	User is able to connect	Fail	Minor	Connection issue	User is able to connect	Pass
FAIL_002	Login failure	User should enter incorrect password	Registered user enters wrong password	Username ="user22" //valid, Password ="pass11" //invalid	Login failed : Password incorrect	Login failed : Password incorrect	Pass	No		Login failed : Password incorrect	Pass

Link to excel file :

https://docs.google.com/spreadsheets/d/1LIcwwQwgv98WMSO-wbPfOHfRk_ka3RatEiWTP79pvTo/edit?usp=s
haring