

INTEGRATION REPORT

TESTCASE 1: (Authentication)

```
cg83-user21@instance-1:~/capg_sprint-2_project/Remote-Editor-Sprint-2/CUT/Cod
e$ make client
gcc client.c clientFunctions.c -o bin/client.out -g
./bin/client.out
Connected to server

Enter username: suraj
Enter password:
.....AUTHENTICATED.....
Enter command-$: help
List of commands available: ls pwd cd select print edit clear bye help
Enter command-$: ls
-      suraj
Enter command-$: pwd
./data/home/suraj
Enter command-$: bye
cg83-user21@instance-1:~/capg_sprint-2_project/Remote-Editor-Sprint-2/CUT/Cod
e$ make client
gcc client.c clientFunctions.c -o bin/client.out -g
./bin/client.out
Connected to server

Enter username: nikita
Enter password:
.....AUTHENTICATED.....
Enter command-$: ls
NO_FILES_FOUND
Enter command-$: pwd
./data/home/nikita
Enter command-$: █
```

TESTCASE 2: (Anonymous user)

```
cg83-user21@instance-1:~/capg_sprint-2_project/Remote-Editor-Sprint-2/CUT/Cod
e$ make client
gcc client.c clientFunctions.c -o bin/client.out -g
./bin/client.out
Connected to server

Enter username:
Enter password:
.....Anonymous.....
Enter command-$: ls
d      some2/
-      some.c
Enter command-$: pwd
./data/home/anonymous
Enter command-$: █
```

TESTCASE 3: (cd and pwd command)

```
Enter username:
Enter password:
.....Anonymous.....
Enter command-$: ls
d      some2/
-      some.c
Enter command-$: pwd
./data/home/anonymous
Enter command-$: cd some2
DIRECTORY_CHANGED
Enter command-$: pwd
./data/home/anonymous/some2
Enter command-$: ls
-      s.c
Enter command-$: cd ..
DIRECTORY_CHANGED
Enter command-$: pwd
./data/home/anonymous
Enter command-$: ls
d      some2/
-      some.c
Enter command-$: █
```

TESTCASE 4: (print command)

```
e$ make client
gcc client.c clientFunctions.c -o bin/client.out -g
./bin/client.out
Connected to server

Enter username:
Enter password:
.....Anonymous.....
Enter command-$: ls
d      some2/
-      some.c
Enter command-$: select some.c
FILE_SELECTED
Enter command-$: print 1
1 #include <stdio.h>
2 int main(int argc, char const *argv[]) {
3     //code
4     return 00;
5 }

Enter command-$: print 1 3
1 #include <stdio.h>
2 int main(int argc, char const *argv[]) {
3     //code

Enter command-$: print 2 4
2 int main(int argc, char const *argv[]) {
3     //code
4     return 00;

Enter command-$: █
```

TESTCASE 5: (edit command and invalid line number)

```
Enter command-$: print 1 5
1 #include <stdio.h>
2 int main(int argc, char const *argv[]) {
3     //code
4     return 00;
5 }

Enter command-$: edit 4
4:return 00;
Enter changes to the line: return 1;
Enter command-$: print 1 5
1 #include <stdio.h>
2 int main(int argc, char const *argv[]) {
3     //code
4     return 1;
5 }

Enter command-$: print 1 8
INVALID_LINE_NUMBER : Choose between 1 and 5

Enter command-$: █
```

TESTCASE 6: (bye command)

```
cg83-user21@instance-1:~/capg_sprint-2_project/Remote-Editor-Sprint-2/CUT/Cod
e$ make client
gcc client.c clientFunctions.c -o bin/client.out -g
./bin/client.out
Connected to server

Enter username:
Enter password:
.....Anonymous.....
Enter command-$: ls
d      some2/
-      some.c
Enter command-$: bye
cg83-user21@instance-1:~/capg_sprint-2_project/Remote-Editor-Sprint-2/CUT/Cod
e$ █
```

TESTCASE 7: (help command and invalid command)

```
Enter command-$: ls
d      some2/
-
some.c
Enter command-$: help
List of commands available: ls pwd cd select print edit clear bye help
Enter command-$: help
List of commands available: ls pwd cd select print edit clear bye help
Enter command-$: help ls
ls - list files in current directory
No arguments are needed
Enter command-$: help pwd
pwd - print current directory
No arguments are needed
Enter command-$: help cd
cd <directory> - change current directory
It takes directory name as argument
Enter command-$: help select
select <filename> - select file to edit
It takes filename name as argument
Enter command-$: help print
print <start> <end> - print the file
It takes 2 numbers (starting and ending number) as optional arguments
Enter command-$: help edit
edit <linenum> - edit the file
It takes line number as arguments and edit the line on the selected file
Enter command-$: help clear
clear - clear the screen
No arguments are needed
Enter command-$: help bye
bye - logout the current user
No arguments are needed
Enter command-$: he
Invalid command
Enter command-$: █
```

UNIT TESTING REPORT

- ❖ We are using Cunit for testing user authentication and validity of loading details from user.txt
- ❖ We have made various header files along with .c and main file. At the end we are using makefile to compile and run the program.

CUnit Testcases for user authentication function:

- **3 Sunny testcases** (Valid input files) and **3 rainy testcases** (Invalid input files)

```
void testSunnyCases1(void) {
    CU_ASSERT_EQUAL(authentication("data/usercu.txt","nikita","nikita15"), 2);
    CU_ASSERT_EQUAL(authentication("data/usercu.txt","suraj","suraj25"), 2);
    CU_ASSERT_EQUAL(authentication("data/usercu.txt","dinesh","dinesh16"), 2);
}
void testRainyCases1(void) {
    CU_ASSERT_EQUAL(authentication("data/usercu.txt","anony","anony"), 1);
    CU_ASSERT_EQUAL(authentication("data/usercu.txt","author","author23"), 1);
    CU_ASSERT_EQUAL(authentication("data/usercu.txt","ruby","ruby7"), 1);
}
```

Cunit Testcases for checking validity of loading user data function:

- **3 Sunny testcases** and **3 Rainy testcases**

```
ser server;
void testSunnyCases2(void) {
    CU_ASSERT_EQUAL(LoadUsersData(&server,"data/usercu.txt"), 1);
    CU_ASSERT_EQUAL(LoadUsersData(&server,"data/usercu.txt"), 1);
    CU_ASSERT_EQUAL(LoadUsersData(&server,"data/usercu.txt"), 1);
}
void testRainyCases2(void) {
    CU_ASSERT_EQUAL(LoadUsersData(&server,"data/usercv.txt"), 0);
    CU_ASSERT_EQUAL(LoadUsersData(&server,"data/usercl.txt"), 0);
    CU_ASSERT_EQUAL(LoadUsersData(&server,"data/usercd.txt"), 0);
}
```

Execution as well as Running the program via makefile

```
cg83-user21@instance-1:~/capg_sprint-2_project/Remote-Editor-Sprint-2/CUT/ToolReport/Cunit$ make
gcc -c src/main.c -o bin/main.o -lcunit -g
gcc -c src/authenticate.c -o bin/authenticate.o -lcunit -g
gcc -c src/authenticate_suite.c -o bin/authenticate_suite.o -lcunit -g
gcc -c src/load.c -o bin/load.o -lcunit -g
gcc -c src/load_suite.c -o bin/load_suite.o -lcunit -g
gcc bin/main.o bin/authenticate.o bin/authenticate_suite.o bin/load.o bin/load_suite.o -o bin/file.tests -lcunit -g
./bin/file.tests

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

Suite: Suite Authenticate file
  Test: test of authenticate() in Sunny cases ...passed
  Test: test of authenticate() in Rainy cases ...passed
Suite: Suite Load file
  Test: test of load() in Sunny cases ...passed
  Test: test of load() in Rainy cases ...passed

Run Summary:
  Type      Total      Ran Passed Failed Inactive
  suites         2         2    n/a      0         0
  tests         4         4      4      0         0
  asserts       12       12     12      0        n/a

Elapsed time =    0.000 seconds
```

IT PLAN

1	TEST CASE ID STEP		PRECONDITIONS/DEPENDENCIES	TEST CASE	TEST DATA	EXPECTED RESULTS	RESULTS OBTAINED
2	IT_01	Accept Connec	No preconditions for User	Connection should be established	Anonymous and Authenticated User	Connection established	DONE
3	IT_02	Load User Data	User should only be Anonymous/Authenticated.	Classify the user based on credentials	Anonymous: Username and password null, Authenticated users credentials should match	User able to login as Anc	DONE
4	IT_03	cd and pwd cor	Restriction for users to go back from the h	Change Directory and present working	Both Authenticated and Anonymous User can access.	Directory changes displa	DONE
5	IT_04	print command	User should select a file before operating this command	Files inside present working directory	Both Authenticated and Anonymous User can access.	Print the contents inside	DONE
6	IT_05	Edit command	User should select a file before operating	tfiles inside present working directory	Both Authenticated and Anonymous User can access	Edit the specified line in t	DONE
7	IT_06	Bye Command	No preconditions for User	Connection should be disconnected	Anonymous and Authenticated User	Disconnected successful	DONE
8							
9							

UT PLAN

1	TEST CASE ID	STEP	PRECONDITIONS/DEPENDENCIES	TEST CASE	TEST DATA	EXPECTED RESULTS	RESULTS OBTAINED	STATUS(PASS/FAIL)
2	UT_01	Sunny case	Authentication should be successful	Authentication	Username, Pas	Authenticated User able to	DONE	PASS
3	UT_02	Rainy case	Authentication is not successful	Authentication	Username, Pas	Invalid credentials, User n	DONE	PASS
4	UT_03	Sunny case	User Data should be able to load inside the user data base.	LoadUserData	User Databse	it Loading of data inside use	DONE	PASS
5	UT_04	Rainy case	User Data does not load inside the user data base.	LoadUserData	User Databse	it Loading of data inside use	DONE	PASS
6								
7								
8								