Group Member: 1. Mihir Dalvi

2. Sumeet Patle

User Side design implementation(participant)

1. Write a user program in C language and start chatroom by using open system call.
2. As participants joined chatroom then save the all participant(users) information.
3. Once above all process done then create multiple thread.
4. First thread will be used for sending the message and it can exit by chatroom by writing ‘BYE’ or taking input from participant to end the chatroom and close the chatroom.
5. Second thread will be use to read on the message in queue on the open device by using read system call.

Kernel mode design implementation

1. Write a kernel program in C language and create or register a character chatroom.
2. All system calls like open, read, close…etc. supported by the device.
3. Here First store all the information regarding all users registered on the device.
4. After that create a buffer for all message that should store in FIFO manner.
5. With the help of system call the user have option to join the chatroom or leave the chatroom or make changes in information.
6. After that write system call send and display message to all registered user.
7. Read system call read one message at a time from buffer.
8. Also to maintain synchronization between all users(process) implement semaphore with locks so no other user(process) can interfere with message of other user (process) message.