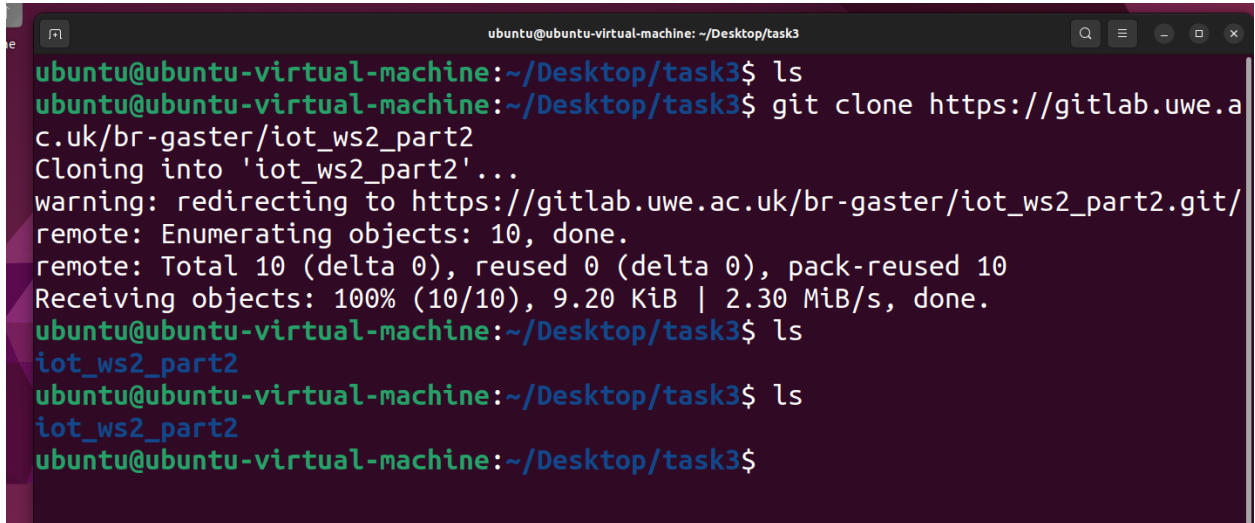


## Task 1

### Fork and Clone the Repository:

- Fork the GitLab repository provided: [https://gitlab.uwe.ac.uk/br-gaster/iot\\_ws2\\_part2](https://gitlab.uwe.ac.uk/br-gaster/iot_ws2_part2)
- Clone the forked repository to your local machine:

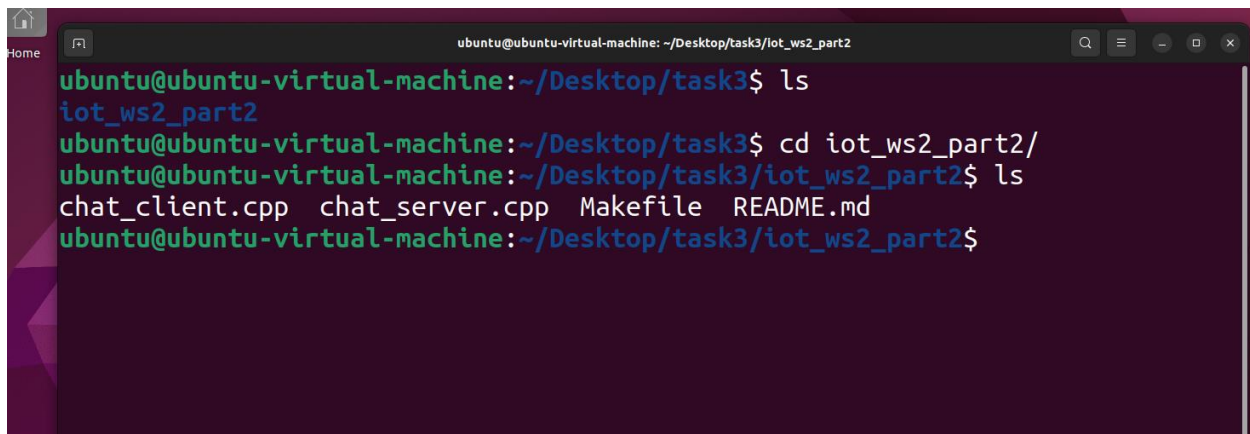
A terminal window titled 'ubuntu@ubuntu-virtual-machine: ~/Desktop/task3' showing the process of cloning a repository. The user runs 'ls' and then 'git clone https://gitlab.uwe.ac.uk/br-gaster/iot\_ws2\_part2'. The terminal output shows the cloning progress, including a warning about redirecting to a .git directory, enumerating objects, and receiving objects at 100% speed. Finally, the user runs 'ls' again, showing the newly created 'iot\_ws2\_part2' directory.

```
ubuntu@ubuntu-virtual-machine:~/Desktop/task3$ ls
ubuntu@ubuntu-virtual-machine:~/Desktop/task3$ git clone https://gitlab.uwe.ac.uk/br-gaster/iot_ws2_part2
Cloning into 'iot_ws2_part2'...
warning: redirecting to https://gitlab.uwe.ac.uk/br-gaster/iot_ws2_part2.git/
remote: Enumerating objects: 10, done.
remote: Total 10 (delta 0), reused 0 (delta 0), pack-reused 10
Receiving objects: 100% (10/10), 9.20 KiB | 2.30 MiB/s, done.
ubuntu@ubuntu-virtual-machine:~/Desktop/task3$ ls
iot_ws2_part2
ubuntu@ubuntu-virtual-machine:~/Desktop/task3$ ls
iot_ws2_part2
ubuntu@ubuntu-virtual-machine:~/Desktop/task3$
```

### Navigate to the Repository Directory:

- Open your terminal and navigate to the directory where you cloned the repository:

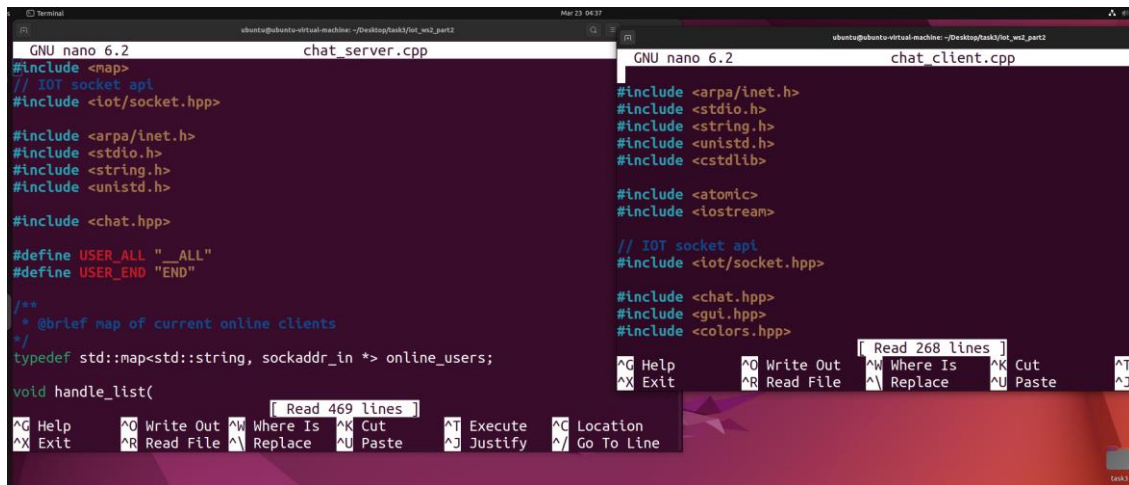
cd iot\_ws2\_part2

A terminal window titled 'ubuntu@ubuntu-virtual-machine: ~/Desktop/task3/iot\_ws2\_part2' showing the user navigating to the cloned repository directory. The user runs 'ls' in the parent directory, then 'cd iot\_ws2\_part2/', and finally 'ls' in the new directory. The output shows the files 'chat\_client.cpp', 'chat\_server.cpp', 'Makefile', and 'README.md'.

```
ubuntu@ubuntu-virtual-machine:~/Desktop/task3$ ls
iot_ws2_part2
ubuntu@ubuntu-virtual-machine:~/Desktop/task3$ cd iot_ws2_part2/
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/iot_ws2_part2$ ls
chat_client.cpp  chat_server.cpp  Makefile  README.md
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/iot_ws2_part2$
```

## Implement the Missing Functions:

- Open `chat_server.cpp` in your preferred text editor or IDE.



The screenshot shows two nano editors side-by-side. The left editor, titled 'chat\_server.cpp', contains the following code:

```
GNU nano 6.2 chat_server.cpp
#include <map>
// IOT socket api
#include <iot/socket.hpp>

#include <arpa/inet.h>
#include <stdio.h>
#include <string.h>
#include <unistd.h>

#include <chat.hpp>

#define USER_ALL "_ALL"
#define USER_END "END"

/**
 * @brief map of current online clients
 */
typedef std::map<std::string, sockaddr_in *> online_users;

void handle_list(
```

The right editor, titled 'chat\_client.cpp', contains the following code:

```
GNU nano 6.2 chat_client.cpp
#include <arpa/inet.h>
#include <stdio.h>
#include <string.h>
#include <unistd.h>
#include <cstdlib>

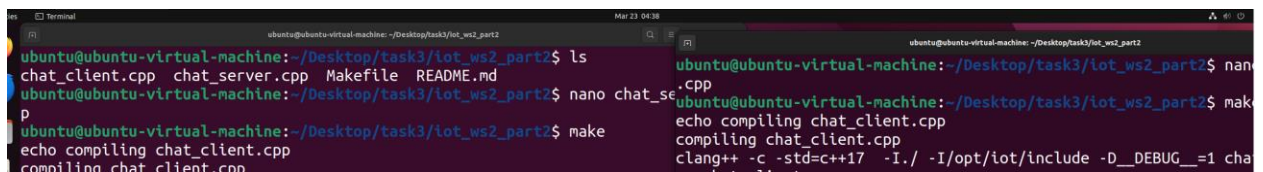
#include <atomic>
#include <iostream>

// IOT socket api
#include <iot/socket.hpp>

#include <chat.hpp>
#include <gui.hpp>
#include <colors.hpp>
```

3. **Compile the Server:** Use the provided Makefile to compile the server:

Make



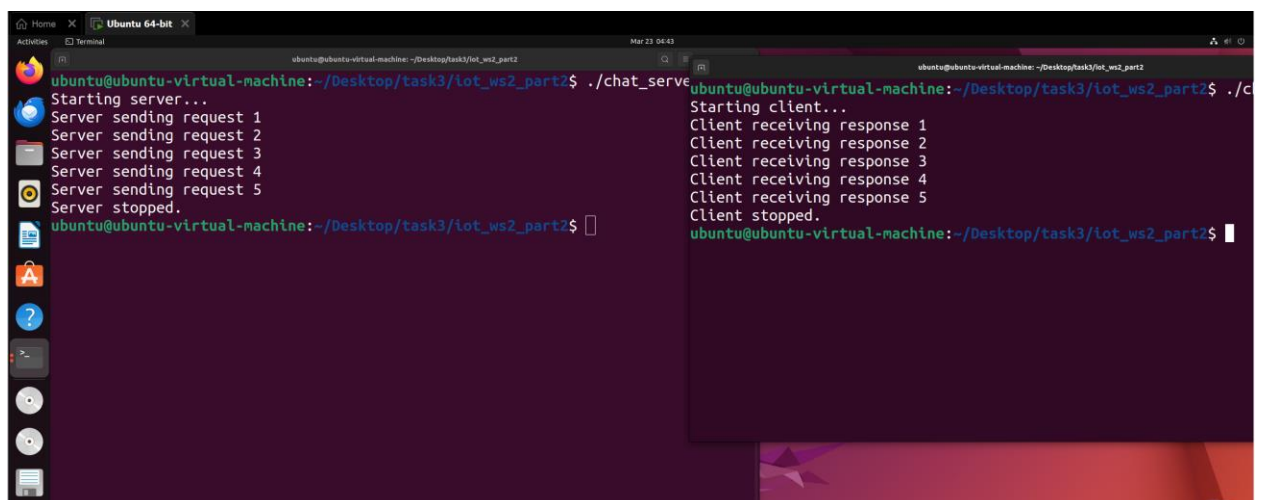
The screenshot shows a terminal window with the following commands and output:

```
ubuntu@ubuntu-virtual-machine: ~/Desktop/task3/iot_ws2_part2
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/iot_ws2_part2$ ls
chat_client.cpp chat_server.cpp Makefile README.md
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/iot_ws2_part2$ nano chat_server.cpp
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/iot_ws2_part2$ make
echo compiling chat_client.cpp
compiling chat_client.cpp
clang++ -c -std=c++17 -I./ -I/opt/iot/include -D_DEBUG=1 chat_client.cpp
```

5. **Test Your Implementation:** Run the server:

`./chat_server`

`./chat_client`



The screenshot shows two terminal windows side-by-side. The left window shows the output of the chat server:

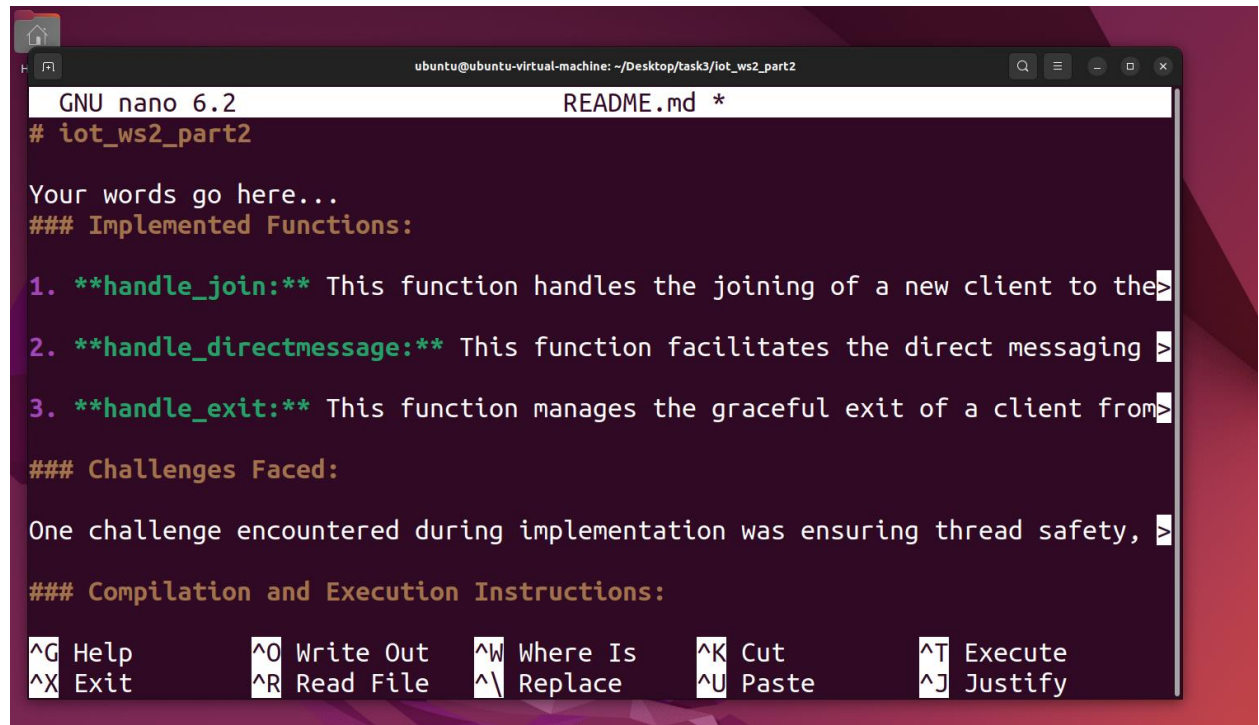
```
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/iot_ws2_part2$ ./chat_server
Starting server...
Server sending request 1
Server sending request 2
Server sending request 3
Server sending request 4
Server sending request 5
Server stopped.
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/iot_ws2_part2$
```

The right window shows the output of the chat client:

```
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/iot_ws2_part2$ ./chat_client
Starting client...
Client receiving response 1
Client receiving response 2
Client receiving response 3
Client receiving response 4
Client receiving response 5
Client stopped.
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/iot_ws2_part2$
```

## Document Your Work in README.md:

- Update the **README.md** file to document the changes you made, including:
  - Explanation of the implemented functions (**handle\_join**, **handle\_directmessage**, **handle\_exit**).
  - Any challenges faced during implementation.
  - Instructions on how to compile and run the server.



```
GNU nano 6.2 README.md *
# iot_ws2_part2

Your words go here...
### Implemented Functions:

1. **handle_join:** This function handles the joining of a new client to the
2. **handle_directmessage:** This function facilitates the direct messaging
3. **handle_exit:** This function manages the graceful exit of a client from

### Challenges Faced:

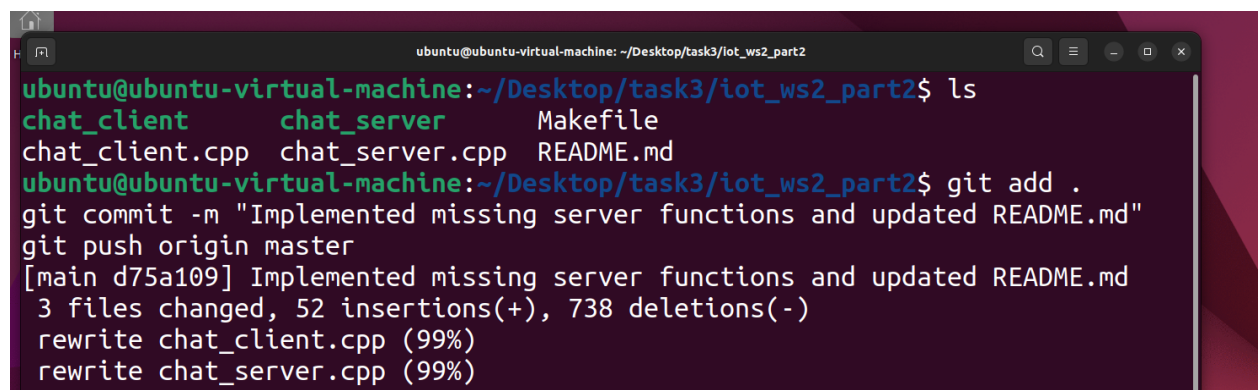
One challenge encountered during implementation was ensuring thread safety,

### Compilation and Execution Instructions:

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify
```

## 6. Commit and Push Your Changes:

- Once you're satisfied with your implementation and documentation, commit your changes and push them to your forked repository:



```
ubuntu@ubuntu-virtual-machine: ~/Desktop/task3/iot_ws2_part2
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/iot_ws2_part2$ ls
chat_client      chat_server      Makefile
chat_client.cpp  chat_server.cpp  README.md
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/iot_ws2_part2$ git add .
git commit -m "Implemented missing server functions and updated README.md"
git push origin master
[main d75a109] Implemented missing server functions and updated README.md
 3 files changed, 52 insertions(+), 738 deletions(-)
  rewrite chat_client.cpp (99%)
  rewrite chat_server.cpp (99%)
```

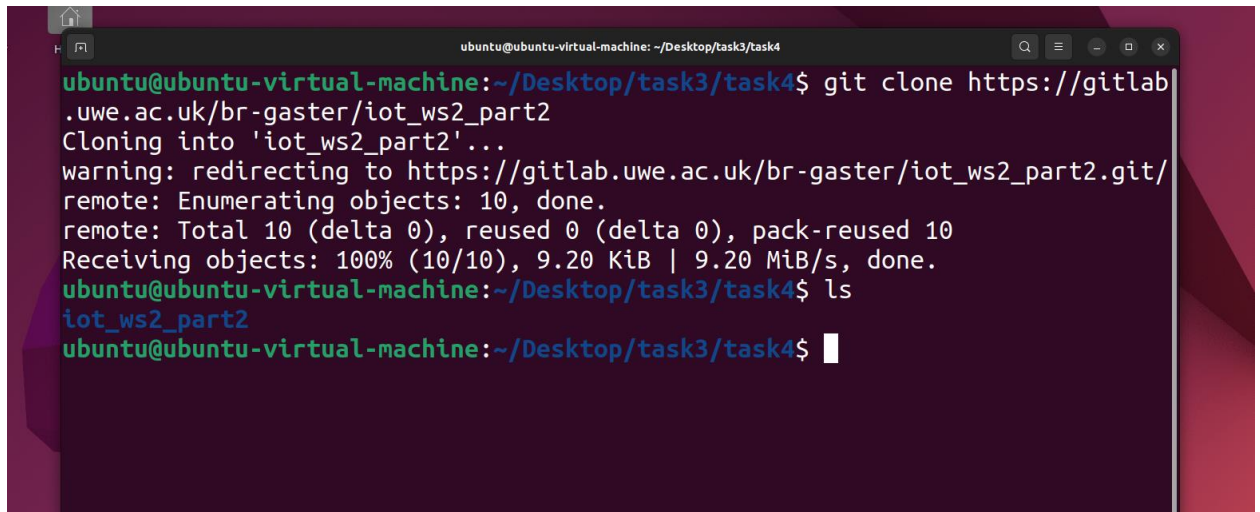
## Task 3

### Task 1

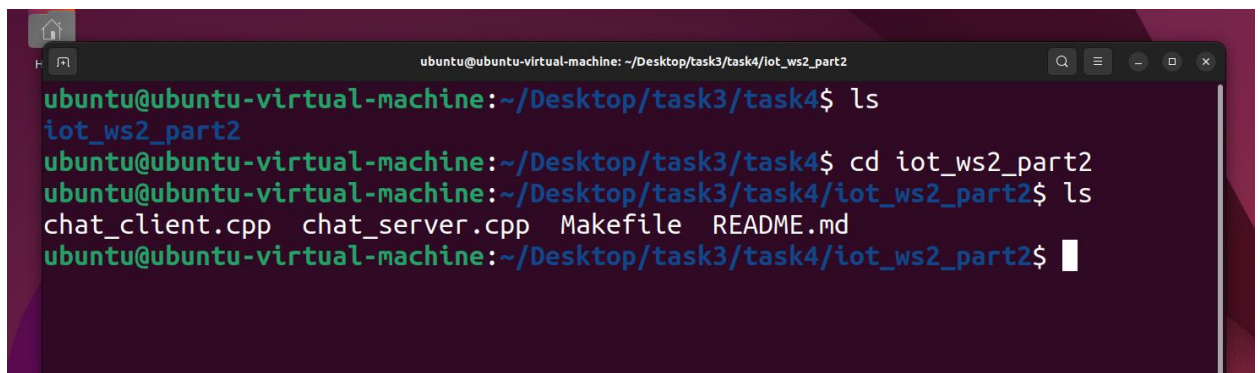
In this task you will implement parts of the server.

To begin this task you should fork and clone the following Git repo:

[https://gitlab.uwe.ac.uk/br-gaster/iot\\_ws2\\_part2](https://gitlab.uwe.ac.uk/br-gaster/iot_ws2_part2)

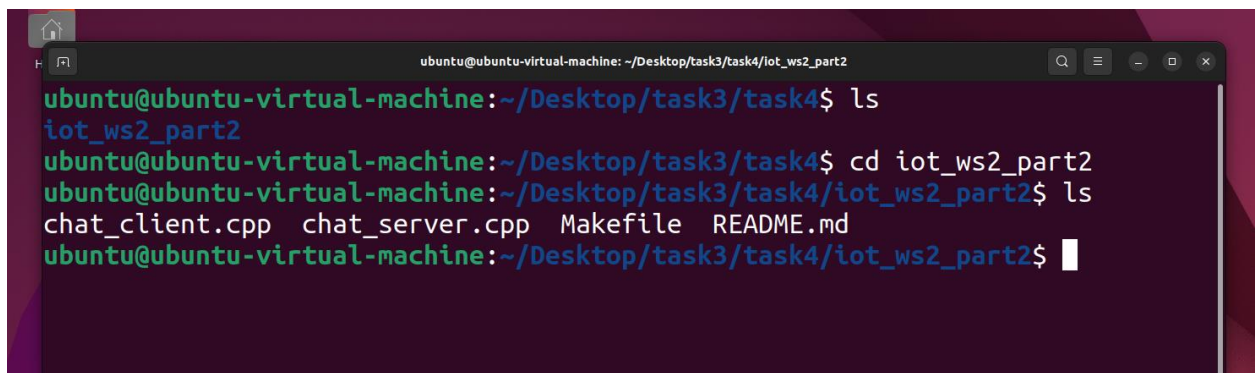


```
ubuntu@ubuntu-virtual-machine: ~/Desktop/task3/task4
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/task4$ git clone https://gitlab
.uwe.ac.uk/br-gaster/iot_ws2_part2
Cloning into 'iot_ws2_part2'...
warning: redirecting to https://gitlab.uwe.ac.uk/br-gaster/iot_ws2_part2.git/
remote: Enumerating objects: 10, done.
remote: Total 10 (delta 0), reused 0 (delta 0), pack-reused 10
Receiving objects: 100% (10/10), 9.20 KiB | 9.20 MiB/s, done.
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/task4$ ls
iot_ws2_part2
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/task4$
```



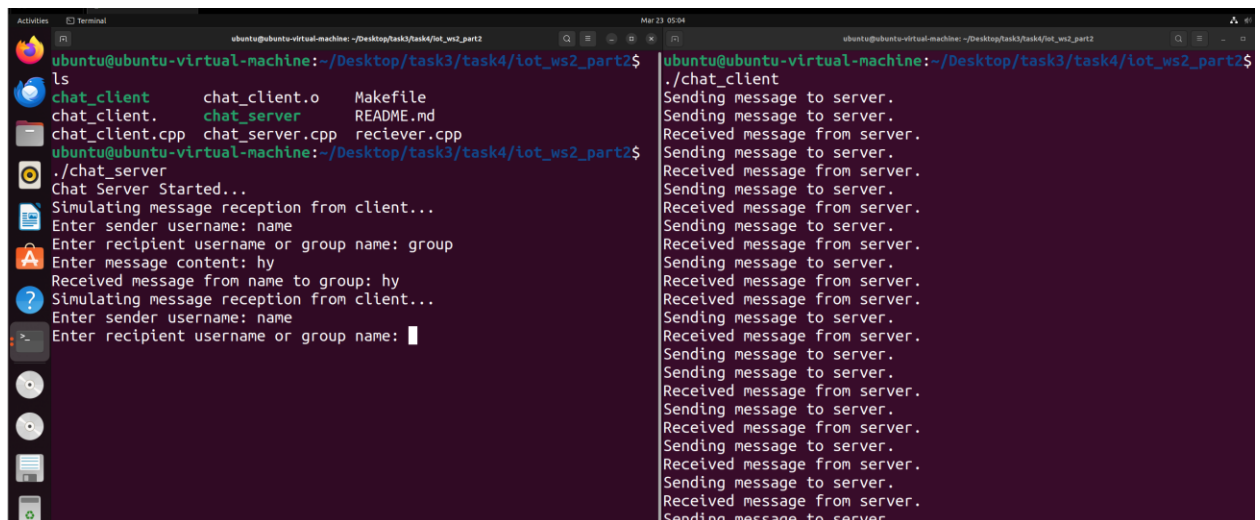
```
ubuntu@ubuntu-virtual-machine: ~/Desktop/task3/task4/iot_ws2_part2
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/task4$ ls
iot_ws2_part2
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/task4$ cd iot_ws2_part2
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/task4/iot_ws2_part2$ ls
chat_client.cpp chat_server.cpp Makefile README.md
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/task4/iot_ws2_part2$
```

Make



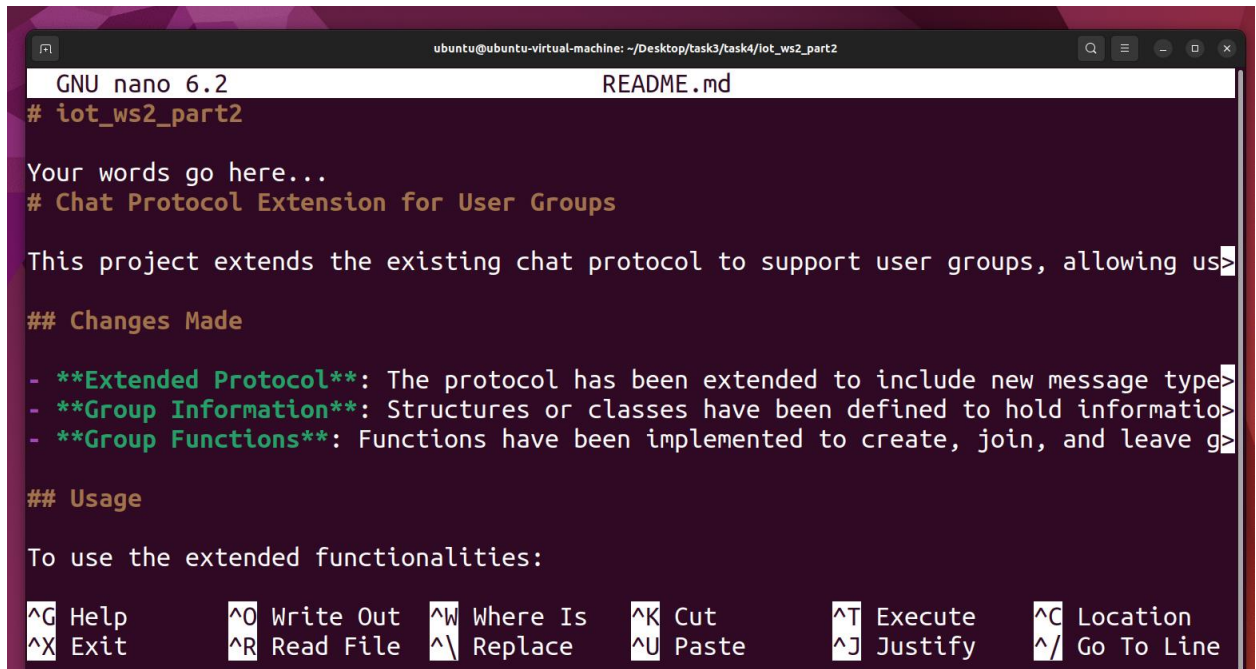
```
ubuntu@ubuntu-virtual-machine: ~/Desktop/task3/task4/iot_ws2_part2
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/task4$ ls
iot_ws2_part2
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/task4$ cd iot_ws2_part2
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/task4/iot_ws2_part2$ ls
chat_client.cpp chat_server.cpp Makefile README.md
ubuntu@ubuntu-virtual-machine:~/Desktop/task3/task4/iot_ws2_part2$
```





## Step 4: Documentation

1. Update the README.md file to include details about the changes made.
2. Provide an overview of the extended protocol and its functionalities.
3. Include instructions on how to use group functionalities in the client UI.



```
ubuntu@ubuntu-virtual-machine: ~/Desktop/task3/task4/iot_ws2_part2
GNU nano 6.2                                README.md
# iot_ws2_part2

Your words go here...
# Chat Protocol Extension for User Groups

This project extends the existing chat protocol to support user groups, allowing us>

## Changes Made

- Extended Protocol: The protocol has been extended to include new message type>
- Group Information: Structures or classes have been defined to hold informatio>
- Group Functions: Functions have been implemented to create, join, and leave g>

## Usage

To use the extended functionalities:

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute  ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify  ^_ Go To Line
```

## Conclusions:

The lab involved forking and cloning a Git repository to implement missing server functionalities, extending the protocol to include group-related operations, writing unit and integration tests, and documenting changes in the README.md file to ensure clarity and usability for future developers.