INSTITUTE OF TECHNICAL EDUCATION AND RESEARCH (ITER), SOA UNIVERSITY
Capstone Project Report – Assignment 4 (LSP)
Subject: Linux System Programming
Project Title: Network File Sharing – Server & Client using C++ Sockets
Submitted by: *Aditya Padhi*
Regd No.: 2241016007
Branch: Computer Science and Engineering (CSE)
Semester: 7th
Under the Guidance of: Dr. Ayes Chinmay

Capstone Project Report

Assignment 4 (LSP): Network File Sharing – Server & Client

1. Title

Network File Sharing – Server & Client using C++ Sockets

2. Objective

The objective of this project is to develop a networked file-sharing system using C++ socket programming that allows users to share files between a server and multiple clients over a TCP/IP connection. The system supports uploading, downloading, and listing files remotely, demonstrating client–server architecture and inter-process communication under Linux.

3. Tools and Technologies

Component	Technology
Language	C++ 17
Platform	Linux (Ubuntu via WSL2)
IDE	Visual Studio Code
Protocol	TCP/IP
Libraries	<sys/socket.h>, <arpa/inet.h>, <netdb.h>, <thread>, <filesystem>

4. Problem Statement

Transferring files between systems often requires manual effort or third-party tools. This project provides a custom-built file-sharing application that can be run entirely on Linux systems, using only C++ sockets for secure and efficient file exchange.

5. Project Description

The project consists of two main modules:

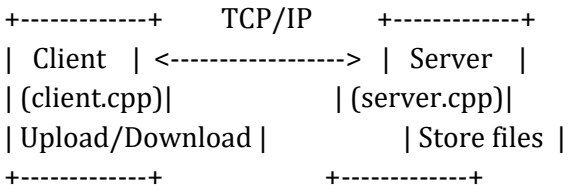
- Server Module (server.cpp)

- Listens for incoming client connections on a specified IP and port.
- Handles client requests for file upload (put), file download (get), and listing (ls).
- Saves uploaded files to a shared directory (shared_root/uploads/).
- Logs client connections and file activity.

- Client Module (client.cpp)

- Connects to the server using TCP sockets.
- Sends commands (put, get, ls, quit) to interact with the server.
- Uploads and downloads files using reliable send/receive mechanisms.
- Displays file lists and transfer status in the terminal.

6. System Architecture



7. Features Implemented

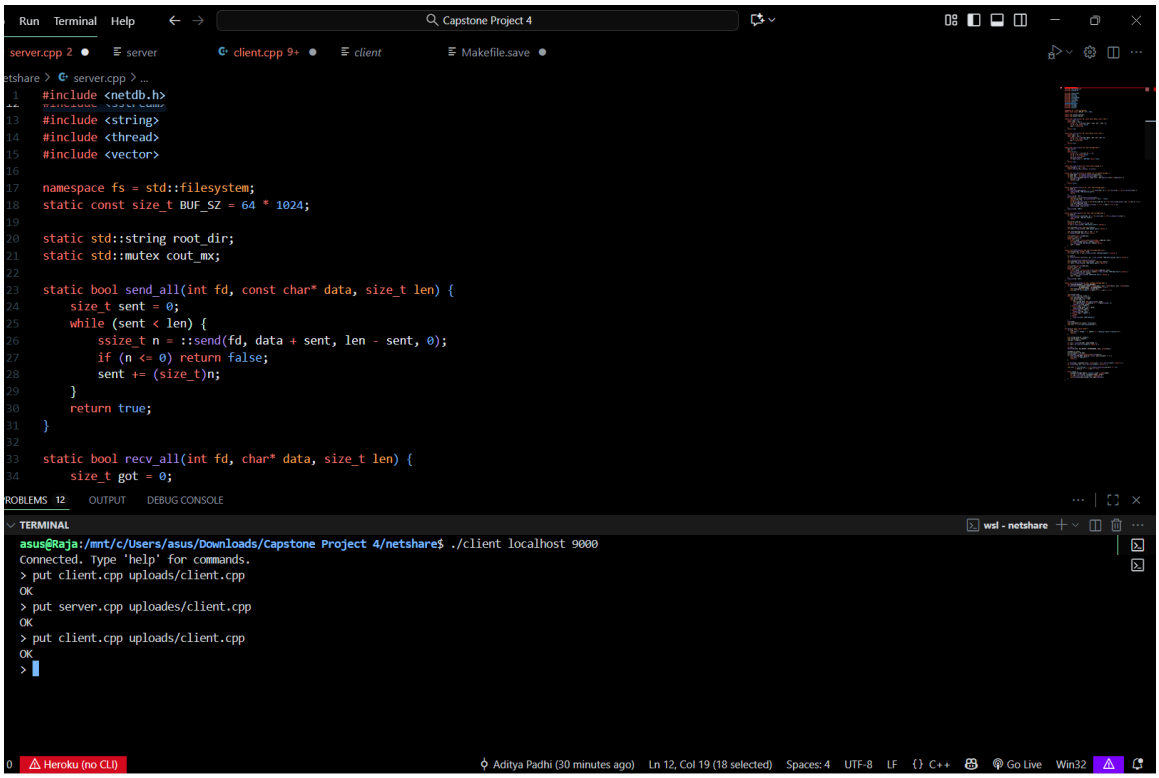
- Server–Client connection via TCP
- File Upload (Client → Server)
- File Download (Server → Client)
- File Listing (ls)
- Multi-directory support (uploads, hii, etc.)
- Real-time server logs for file activity
- Browser access through python3 -m http.server

8. Commands Used

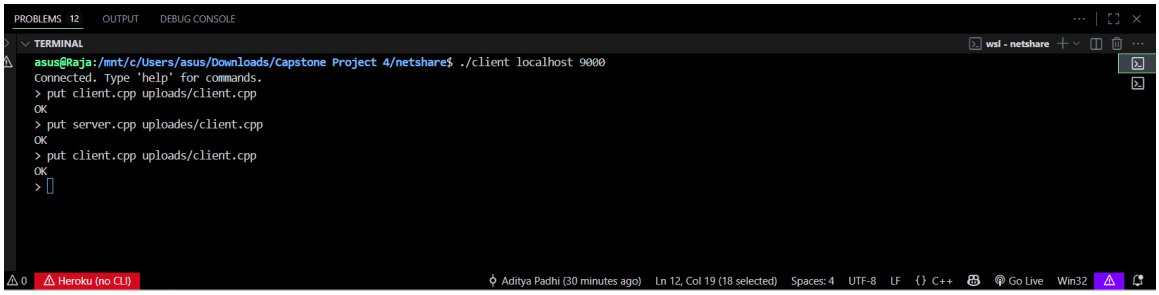
Command	Description
put <local> <remote>	Uploads file to server
get <remote>	Downloads file from server
ls	Lists files/directories on server
quit	Disconnects client
python3 -m http.server 8080	View shared files via browser

9. Output Screenshots

Include screenshots of server start, client connection, uploads, listings, logs, and browser view.


Run Terminal Help
Capstone Project 4
server.cpp 2
server
client.cpp 9+
client
Makefile.save
server.cpp
#include <netdb.h>
#include <string>
#include <thread>
#include <vector>
namespace fs = std::filesystem;
static const size_t BUF_SZ = 64 * 1024;
static std::string root_dir;
static std::mutex cout_mx;
static bool send_all(int fd, const char* data, size_t len) {
size_t sent = 0;
while (sent < len) {
ssize_t n = ::send(fd, data + sent, len - sent, 0);
if (n <= 0) return false;
sent += (size_t)n;
}
return true;
}
static bool recv_all(int fd, char* data, size_t len) {
size_t got = 0;
PROBLEMS 12
OUTPUT
DEBUG CONSOLE
TERMINAL
wsl - netshare
asus@Raja:/mnt/c/Users/asus/Downloads/Capstone Project 4/netshare$ ./client localhost 9000
Connected. Type 'help' for commands.
> put client.cpp uploads/client.cpp
OK
> put server.cpp uploades/client.cpp
OK
> put client.cpp uploads/client.cpp
OK
> Heroku (no CLI)
Aditya Padhi (30 minutes ago)
Ln 12, Col 19 (18 selected)
Spaces: 4
UTF-8
LF
C++
Go Live
Win32


PROBLEMS 12
OUTPUT
DEBUG CONSOLE
TERMINAL
wsl
asus@Raja:/mnt/c/Users/asus/Downloads/Capstone Project 4/netshare$ ./server 0.0.0.0 9000 ./shared_root
[*] Serving "/mnt/c/Users/asus/Downloads/Capstone Project 4/netshare/shared_root" on 0.0.0.0:9000
[+] client 127.0.0.1:51912
Heroku (no CLI)
Aditya Padhi (30 minutes ago)
Ln 12, Col 19 (18 selected)
Spaces: 4
UTF-8
LF
C++
Go Live
Win32


PROBLEMS 12
OUTPUT
DEBUG CONSOLE
TERMINAL
wsl - netshare
asus@Raja:/mnt/c/Users/asus/Downloads/Capstone Project 4/netshare$ ./client localhost 9000
Connected. Type 'help' for commands.
> put client.cpp uploads/client.cpp
OK
> put server.cpp uploades/client.cpp
OK
> put client.cpp uploads/client.cpp
OK
Heroku (no CLI)
Aditya Padhi (30 minutes ago)
Ln 12, Col 19 (18 selected)
Spaces: 4
UTF-8
LF
C++
Go Live
Win32

10. Results

Project successfully implements file-sharing using sockets with upload, download, and web access features.

11. Conclusion

A fully functional file-sharing application was developed using C++, demonstrating client-server communication and socket programming.

12. GitHub Repository

https://github.com/AdityaPadhi-lab/Network-File-Sharing-Server-Client

13. Future Scope

Add user authentication, encryption, and web dashboard features for better usability.

14. References

Beej's Guide to Network Programming, cppreference.com, and Linux man pages for socket functions.

Place: Bhubaneswar
Date: 09/11/2025

Name:- Aditya Padhi
Regd. No.: 2241016007
B.Tech – 7th Semester, CSE
ITER, SOA University