

Alunos

- Matheus Della Rocca Martins – 12549731
- Thallys Simões de Moraes Oliveira – 11819827

Git: [Trabalho 2 – Redes de Computadores](#)

Drive: [Vídeo de Demonstração](#)

Proposta

O projeto proposto busca implementar um sistema de chat em grupo utilizando sockets TCP e endereçamento com IPv4. Neste sentido, visa-se desenvolver uma aplicação que permite a troca de mensagens entre múltiplos usuários conectados em uma mesma rede. O servidor desempenha um papel central na organização da comunicação, sendo responsável pelo estabelecimento das conexões, recebimento e encaminhamento das mensagens dos clientes para os destinatários adequados.

Dessa forma, o servidor permite que vários clientes se conectem simultaneamente e enviem mensagens broadcast (para todos os clientes conectados) ou mensagens diretas (para um cliente específico). Além disso, foram implementadas funcionalidades adicionais, como a visualização da lista de clientes ativos e um comando de ajuda para orientar os usuários nas diversas ações disponíveis.

A escolha de utilizar o protocolo TCP em vez do UDP baseia-se na necessidade de garantir a confiabilidade na transmissão das mensagens em um chat. O TCP oferece mecanismos robustos de controle de erro, retransmissão de dados e garantia de entrega ordenada, essenciais para a integridade das conversas trocadas pelos usuários em nossa aplicação.

Por fim, vale-se destacar que este projeto foi desenvolvido e testado para que todos os clientes sejam executados em uma mesma máquina e se comuniquem através desta rede local.

Funcionalidades

- **Mensagens Broadcast:** Permite enviar mensagens para todos os clientes conectados ao servidor.
- **Mensagens Diretas:** Permite enviar mensagens diretamente para um cliente específico. Para enviar uma mensagem direta, utilize o comando ***dm nomeDoCliente: mensagem*** no terminal.
- **Mensagens do servidor:** Permite que sejam enviadas mensagens/avisos, destinadas para todos os clientes, através do servidor.
- **Comandos de Ajuda (help):** Fornece um comando para exibir a lista de comandos disponíveis. Para utilizá-lo, envie `help` através dos clientes.
- **Lista de Clientes (client list):** Possibilidade de listar todos os clientes conectados ao servidor. Para utilizá-lo, envie `client list` através dos clientes.

Compilação e Execução

Os requisitos deste projeto normalmente já vem instalados por padrão no linux. Entretanto, caso tenha problemas seguem as devidas instruções:

Para compilar e executar este projeto, são necessárias as seguintes bibliotecas:

- **gcc:** Compilador GNU para C e C++
 - sudo apt update
 - sudo apt install gcc
- **libstdc++:** Biblioteca padrão do C++
 - Instalação adicional não necessária no Ubuntu/Debian, incluída com o GCC.
- **pthread:** Biblioteca para suporte a threads POSIX.
 - sudo apt update

- `sudo apt install libpthread-stubs0-dev`

Compilando o Servidor:

- Navegue até o diretório onde o código do servidor está localizado.
- Compile o servidor com o seguinte comando:
 - **`gcc -o server server.cpp -lstdc++ -lpthread`**

Executando o Servidor:

- Após a compilação, execute o servidor com: **`./server`**

Compilando o Cliente:

- Navegue até o diretório onde o código do cliente está localizado.
- Compile o cliente com o seguinte comando:
 - **`gcc -o client client.cpp -lstdc++ -lpthread`**

Executando o Cliente:

- Após a compilação, execute o cliente com: **`./client`**

Certifique-se de iniciar o servidor antes de executar os clientes. Cada cliente deve se conectar ao servidor especificando o IP e a porta correspondentes.