

Manual do Chat em Rede via Sockets TCP

Este documento descreve um sistema de chat em rede construído com sockets TCP em Python 3. Ele permite que várias pessoas conversem em tempo real usando o terminal. O sistema é dividido em duas partes principais: o servidor, que organiza as mensagens e conexões, e o cliente, que permite aos usuários se conectarem e interagirem.

O que você vai precisar:

- Python 3 instalado.
- Uma rede local ou internet com IP e porta acessíveis para a conexão.
- Um terminal ou prompt de comando para rodar os programas.

Como funciona:

1. O Servidor (`servidor.py`)

O “`servidor.py`” é o cérebro do chat. Ele fica “ouvindo” na porta 12345 (que pode ser mudada) e aceita conexões de vários usuários ao mesmo tempo. Para cada pessoa que se conecta, ele cria um canal de comunicação separado.

- Quando alguém se conecta, o servidor pede um nome de usuário.
- Mensagens enviadas por todos são repassadas para todos os participantes.
- O servidor também entende comandos para enviar mensagens privadas e para mostrar quem está online.

2. O Cliente (`client.py`)

O “`client.py`” é o programa que você usa para entrar no chat. Ele se conecta ao servidor usando o IP e a porta configurados.

- Ao se conectar, você será solicitado a digitar um nome de usuário.
- Você verá as mensagens enviadas por outras pessoas.
- Você pode enviar mensagens públicas ou privadas.
- Há comandos especiais para ver quem está online e para sair do chat.

Como usar:

1. Para iniciar o servidor:

Abra um terminal e digite:

```
python servidor.py
```

O servidor vai começar e esperar pelas conexões.

2. Para iniciar o cliente:

Abra outro terminal (ou use outra máquina) e digite:

```
python client.py
```

Quando for pedido, digite seu nome de usuário. Pronto! Agora você pode usar o chat.

Comandos do Chat:

- **Mensagem pública:** Digite o que quiser e aperte Enter.
- **Mensagem privada:** Use o comando `"/private nome_do_usuario sua mensagem"` para enviar uma mensagem apenas para a pessoa escolhida.
- **Ver quem está online:** Digite `"/online"`.
- **Sair do chat:** Digite `"/quit"`

Importante:

- O servidor precisa estar ligado para que os clientes consigam se conectar.
- Se você tentar enviar uma mensagem privada para alguém que não existe, o sistema avisará.
- O comando `"/quit"` fecha sua sessão, mas o servidor continua funcionando para os outros.

Conclusão:

Este projeto mostra como usar sockets TCP para comunicação em rede, como o servidor pode lidar com várias conexões ao mesmo tempo e como criar uma interação via terminal.