

Manual do Projeto - Chat com Sockets TCP

Projeto de Chat com Sockets TCP

Este projeto é uma aplicação de chat via terminal, desenvolvida com Python e usando sockets TCP para comunicação entre um servidor e múltiplos clientes.

Estrutura do Projeto

- trabalho-sockets/

server.py # Código do servidor

client.py # Código do cliente

README.md # Este arquivo

Pré-requisitos

- Python 3.6+ instalado (<https://www.python.org/downloads/>)

Como executar manualmente

1. Clone o repositório:

- git clone <https://github.com/seu-usuario/trabalho-sockets.git>
- cd trabalho-sockets

2. Inicie o servidor em um terminal:

- python server.py

3. Inicie um cliente em outro terminal:

- python client.py

4. Repita o passo 3 para cada novo cliente.

Manual do Projeto - Chat com Sockets TCP

Como utilizar

Após executar a aplicação, os terminais do servidor e dos clientes estarão ativos.

Terminal do Servidor

O servidor exibe mensagens indicando que recebeu novas conexões. Ele não envia mensagens, apenas repassa o que recebe de um cliente para todos os outros.

Exemplo:

Servidor rodando em 0.0.0.0:5555

Nova conexão de ('127.0.0.1', 53010)

Nova conexão de ('127.0.0.1', 53014)

Terminal do Cliente

- Quando o cliente inicia, ele pede que você digite seu nome.
- Depois, o cliente entra no modo de chat.
- A partir daí, você pode digitar mensagens no terminal.
- Cada mensagem será enviada ao servidor, que a retransmite para todos os outros clientes conectados.
- O chat continua em tempo real enquanto todos os clientes estiverem conectados.
- Para sair, use CTRL+C no terminal do cliente.

Exemplo:

Cliente 1:

Digite seu nome: Ana

Ana: Oi, tudo bem?

Manual do Projeto - Chat com Sockets TCP

Cliente 2 (recebe automaticamente):

Ana: Ol, tudo bem?

Explicação técnica

- server.py: aguarda conexões de clientes, recebe mensagens e retransmite para os outros.
- client.py: conecta ao servidor, permite enviar e receber mensagens simultaneamente via terminal.
- Comunicação via TCP socket (socket.SOCK_STREAM).

Observações

- Mensagens digitadas s aparecem para os outros, nunca para o próprio remetente.
- Se um cliente se desconectar, os outros continuam conectados normalmente.
- No possui histórico de mensagens, é um sistema simples que funciona somente em tempo real.