

UNIVERSIDADE FEDERAL DE ALAGOAS
INSTITUTO DE COMPUTAÇÃO

BRUNO LUIZ FLORENTINO SILVA

CHAT GLOBAL EM PYTHON UTILIZANDO SOCKET

MACEIÓ-AL
2022

BRUNO LUIZ FLORENTINO SILVA

CHAT GLOBAL EM PYTHON UTILIZANDO SOCKET

Relatório realizado com o intuito de obtenção de nota parcial na disciplina de Redes de Computadores, sob a orientação do professor Doutor Leandro de Melo Sales.

MACEIÓ-AL
2022

1 INTRODUÇÃO

A prosta é criar um chat global, assim como os canais de comunicação presentes na maioria dos jogos MMORPG. A partir do momento em que um usuário se conecta ao servidor do chat ele passa a receber mensagens de outros usuários, precedido pelo seu nome de usuário (nickname: Mensagem). Ou seja, as mensagens trocadas pelos usuários antes da conexão de um terceiro não serão “reprisesadas”.

A criação do chat foi feita utilizando a linguagem Python, e utilizado o protocolo TCP, onde dois arquivos foram criados: Um para o cliente e um para o servidor. As instruções para utilizar o chat estão descritas no repositório Git.

2 FUNCIONALIDADES

Basicamente, a aplicação consegue receber múltiplas conexões de usuários, os quais enviam mensagens para os demais. Além disso, são salvos os nomes e os endereços dos usuários no momento da conexão. Cada usuário utiliza duas *threads*, uma para “escutar” o servidor, e outra para enviar mensagens para todos que estão conectados.

O servidor avisa quando alguém se conecta ao chat global, e mostra seu *nickname* e endereço IP. No processo de desconexão (feito ao fechar o prompt) o servidor também notifica quem foi desconectado e seu endereço IP.

3 O QUE PODIA SER IMPLEMENTADO

Algo que é interessante, e que não foi implementado, é o fato de que ao se conectar ao servidor o usuário não recebe as mensagens anteriores a sua conexão (assim como em grupos de WhatsApp).

Outra ideia era utilizar mensagens em privado para usuários na rede. Para alguns usuários mais antigos é a função de alguns jogos conhecida como

Whisper(sussurro). Assim, colocar */Whisper* + nome do usuário iria enviar a mensagem exclusivamente para o usuário desejado.

4 DIFICULDADES

Foi difícil resolver o problema de desconexão inesperada de um cliente. Antes, a implementação era totalmente dependente do cliente para encerrar uma conexão (digitar uma palavra para encerrar). Posteriormente foi feita uma alteração, onde a conexão é encerrada devidamente quando o cliente fecha o prompt de comando, não retornando erro no servidor.

Não seria considerado bem uma dificuldade, mas ocorreu algo inusitado ao implementar os sockets. Um *while True* colocado no lugar errado fez com que a aplicação gerasse “infinitas” threads. Isso fez o computador consumir toda a memória RAM, e consequentemente travar.

Ademais, é possível citar a dificuldade de programar a intercomunicação entre o cliente e servidor. A dificuldade talvez exista pelo fato de ser a primeira vez ao programar este modelo. Acredito que após um tempo fique mais intuitivo.