

Projecto de Introdução às Redes e Comunicação

Relatório Técnico

server.py:

Compilar -> "python server.py"

Variáveis:

- SOCKET_Dict = {} → Dicionário {sock : {username, código de status}}.
- groups = {} → Dicionário {nomegrupo:[users do grupo]}.
- SOCKET_LIST = [] → Lista com todas as conexões ativas.
- Block = {} → Dicionário {conexão: [users bloqueados]}

Funções:

- chat_server(): Função principal do servidor, em se usou um select.select(SOCKET_LIST,[],[],0)

Cujo objetivo é o controlo das conexões que estão prontas para ser lidas e aceita conexões novas. Sempre que há uma nova conexão, é pedido ao utilizador o username e, preenchidos os respectivos dicionários, este recebe informação de todos os utilizadores que estão online. Como resposta há nova ligação - o servidor envia para todos os utilizadores conectados ao servidor informação sobre o utilizador que se acabou de ligar ao servidor. Usou-se um try/except para analisar o que é enviado pelos utilizadores, gravando no SOCKET_DICT[sock][1] o código de status (/pm/<user/ ou /group/<nomegrupo>/) que permite que qualquer informação enviada pelo utilizador seja enviada correctamente para os utilizadores com a mesma conversa aberta ou, se os utilizadores não tiverem a conversa aberta, enviada a respectiva notificação.

Qualquer código inválido chama a função help().

- readg(): Função chamada quando é recebido o código /group/<nomegrupo>/ - se o grupo estiver criado, abre, enviando a informação do ficheiro para o utilizador; se não estiver criado, alerta para o facto do grupo ainda não estar criado.
- block(): Função chamada quando é recebido o código /block/<user>/ (com flag 0) que adiciona o <user> ao dicionário de bloqueados pelo o utilizador. Função requerida quando é recebido o código /unblock/<user>/ (com flag 1) removendo o <user> do dicionário de bloqueados pelo o utilizador. Função chamada internamente com (flag 2), quando um utilizador tenta enviar mensagens a outro, devolvendo "False" se houver correspondência no dicionário Block entre os utilizadores.

- **ban():** Função chamada quando é recebido o código `/ban/<nomeg>/<user>/` que permite ao criador banir do grupo qualquer utilizador.
- **message():** Função chamada quando o código status é `/group/<nomegrupo>/` cuja função é a do envio da mensagem para todos os utilizadores pertencentes ao grupo com a conversa aberta ou notificação para os utilizadores que não têm a conversa aberta.
- **create():** Função chamada quando é recebido o código `/create/<nomegrupo>/` que permite ao utilizador criar um grupo.
- **invite():** Função chamada quando é recebido o código `/invite/<nomegrupo>/<user>/` - se o utilizador pertencer ao grupo `<nomegrupo>`, permite ao utilizador convidar o `<user>` para o grupo - o `<user>` não precisa de aceitar, fica automaticamente com permissões.
- **broadcast():** Serve para enviar mensagens para todos os utilizadores.
- **help():** Função chamada quando é recebido o código `/help/` ou quando o utilizador insere um código inválido que mostra informação sobre a utilização do programa. Envia ao utilizador uma mensagem de erro quando internamente recebe erro 1 ou 2.

client.py:

Compilar → “python cliente.py 127.0.0.1 9009”

(port do servidor definido como 9009)

Ao iniciar, o programa pede ao utilizador para definir o seu username. O `client.py` serve simplesmente para enviar e receber informação, todo o controlo e análise de informação enviada é feito pelo servidor. O `cliente.py` usa o mesmo método `select()` que o servidor, para saber se o `cliente.py` vai enviar ou receber mensagem, todo o que é recebido do servidor é mostrado na consola.

Elementos do grupo:

Cláudio André Ventura Alves, aluno nº2016225581

Francisco Forte, aluno nº 2016225674