

Projeto de redes de computadores: lista telefônica

Henrique Araújo Ferreira

Instituto de Computação (IC) – Universidade Federal de Alagoas (UFAL)

CEP 57.072-970 – Maceió – AL – Brasil

[{haf@ic.ufal.br}](mailto:haf@ic.ufal.br)

***Resumo.** Este trabalho é uma aplicação dos conceitos aprendidos na disciplina de redes de computadores, em especial o uso de sockets e threads. Foi usada uma implementação multithreading onde é possível consultar o telefone de um indivíduo já cadastrado na lista ou adicioná-lo, caso seu nome não conste.*

Implementação

A linguagem de programação usada foi **python** com suas bibliotecas nativas **socket** e **threading**. A proposta do projeto é de uma lista telefônica com duas funcionalidades: consultar e inserir. O projeto consiste em 3 arquivos, o código-fonte do cliente, o do servidor e a lista telefônica que se trata de um arquivo .csv com as colunas Nome e Telefone.

Na parte do servidor, é iniciado um *socket server* e partir daí, existem 4 funções: checa se o indivíduo dado como entrada consta na lista; adiciona uma pessoa à lista; conexão de cada cliente através de um *thread*; inicia servidor.

No *client side*, é iniciado um socket que é trabalhado ao redor de uma função que recebe a entrada do usuário e com ajuda de variáveis auxiliares, faz todo o trabalho de decisão de consulta ou inserção. Além de que, uma vez decidido o que o usuário deseja, são enviados os pacotes de dados ao servidor e é aguardada uma resposta.

Dificuldades no desenvolvimento

As principais dificuldades encontradas na implementação foram: desconectar o cliente do servidor sem que houvesse *broken pipe error*, isto é, sem que a aplicação do cliente continuasse rodando mesmo sem conexão; adicionar pessoas à lista sem a necessidade de o usuário digitar novamente o nome da pessoa em questão; e por fim, finalizar a *thread* referente ao cliente assim que a conexão com o mesmo é encerrada.

Possíveis melhorias

Nota-se que o modelo de lista telefônica usado neste projeto é simples e não muito informativo e a fim de torná-lo mais completo, poderiam ter sido adicionadas informações como: UF, operadora, possíveis números de telefone adicionais de um indivíduo e tempo com o mesmo número.

No que tange as aplicações, foram usados apenas os métodos READ e CREATE, o que torna a lista estática. Para uma melhor manipulação, dinamizar a lista com a inserção os métodos DELETE e UPDATE se faz necessário.