

INSTITUTO FEDERAL DO PARANÁ
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO

GUILHERME ZIMERMANN

SISTEMA CHAT MULTIUSUÁRIOS

PALMAS

2017

GUILHERME ZIMERMANN

SISTEMA CHAT MULTIUSUÁRIOS

Trabalho apresentado como requisito parcial de avaliação da disciplina de Sistemas Distribuídos do curso de Sistemas de Informação do Instituto Federal do Paraná, campus Palmas.

Professor: Felipe Teixeira

PALMAS

2017

Descrição – Chat

O sistema desenvolvido trata-se de um chat multithread, modelo cliente-servidor, cuja comunicação ocorre por meio de sockets. Ele aceita a conexão de diversos usuários, que podem comunicar-se entre si, através de mensagens escritas, com, no máximo, 140 caracteres. O sistema funciona para conversações em grupo. Ao passo que um novo usuário é conectado, ele já inicia a interação, recebendo e podendo mandar mensagens. O próximo passo, é implementar uma forma de conversas privadas, onde um usuário pode convidar outro para sair da conversação em grupo, para uma conversa privada (trabalho futuro). Quando um usuário se desconecta, os demais são notificados. A linguagem utilizada para o trabalho foi a linguagem Java.

Thread

Pode ser considerada um programa, que funciona como um subsistema, dividindo tarefas de um processo. Essas tarefas subdivididas podem ser executadas ao mesmo tempo, ou quase, trazendo mais performance do que um único bloco sendo executado.

A utilização de threads facilita o desenvolvimento, com a possibilidade de elaborar programas em módulos. Outro ponto, na questão desempenho, é o fato de não deixar processos parados. Enquanto um aguarda um determinado dispositivo de entrada ou saída, ou outro recurso do sistema, outro thread pode estar trabalhando. Por outro lado, exige atenção, pois quanto mais threads implementadas, o trabalho torna-se mais complexo.

Socket

Meio de comunicação usado para viabilizar a conexão cliente/servidor, onde um cliente informa o endereço de IP e a respectiva porta do servidor (endereçamento direto) criando uma forma de comunicação com esse cliente.

Cliente-Servidor

O modelo de arquitetura Cliente-Servidor possui vários aspectos para a sua definição. Um deles refere-se ao método de aplicações computacionais através de diferentes plataformas.

Outra diz respeito a um modelo onde dois ou mais computadores interagem entre si, oferecendo serviços uns aos outros. Também, é uma arquitetura computacional que envolve requisições de serviços de clientes para servidores.

No caso do presente trabalho, um chat multi-usuário, o servidor atua na gerência e distribuição das mensagens entre os usuários (clientes) do serviço de comunicação simultânea.

Java

Linguagem utilizada para o desenvolvimento trabalho, o Java é orientado a objetos, com comportamentos de seus objetos sendo determinados por classes. Sua sintaxe assemelha-se á das linguagens C e C++. Foi criada no início da década de 1990, pela Sun Microsystems. Em 2009, foi adquirida pela Oracle. Atualmente está na versão 8.

No último mês de julho, a IEEE (Instituto de Engenheiros Eletricistas e Eletrônicos), organização profissional conhecida pelo fomento ao conhecimento nas áreas de Engenharia Elétrica, Eletrônica e da Computação, publicou um ranking das linguagens de programação mais utilizadas neste ano.

No ranking geral, o Java ficou em 3º lugar, atrás das linguagens Python e C. No quesito desenvolvimento Web, foi considerada a 2ª mais utilizada. Para Mobile, também a 2ª. No desenvolvimento de aplicação empresariais e científicas, foi a 3ª.

Funcionamento do Chat

A classe Server centralizará as conexões via socket, sendo a responsável pelo envio de mensagens entre os clientes conectados. Cada conexão cria uma thread, ou seja, cada cliente tem um subsistema, sendo o Server responsável pela gerência de todo esse trabalho. Essa classe deve ser executada primeiro.

Por sua vez, a classe Client criará uma instância e fará uma conexão com o Server. Para a conexão, deve-se informar o endereço IP e a porta do servidor. Nesse caso, o IP utilizado é o localhost (127.0.0.1) e a porta 10001.

Referências

O que é Thread? Disponível em <<https://canaltech.com.br/produtos/o-que-e-thread/>>

Fundamentos da Arquitetura Cliente/Servidor. Disponível em <http://www.netsoft.inf.br/aulas/4_SIN_Programacao_Cliente_Servidor/Fundamentos_de_Cliente-Servidor.pdf>

The Top Programming Languages 2017. Disponível em <https://spectrum.ieee.org/ns/IEEE_TPL_2017/index/2017/1/1/1/1/50/1/50/1/50/1/30/1/30/1/20/1/20/1/5/1/5/1/20/1/100/>