

Escola Superior de Tecnologia e Gestão

Licenciatura em Engenharia Informática

Sistemas Distribuídos

Ano Letivo 2021/22

Ficha 3

Elaborado em: 2022/03/23

Daniel Santos | N°2019133865

Índice

List of Figures	ii
1. Exercício 1	1

List of Figures

FIGURA 1 - EXECUÇÃO DO CÓDIGO	1
FIGURA 2 - ALERT MESSAGE	1
FIGURA 3 - CÓDIGO RECEIVE	2
FIGURA 4 - FUNÇÃO LOGOUT	2
FIGURA 5 - FUNÇÃO SEND	3
FIGURA 6 – FUNÇÃO/EVENTO JLOGOUT.....	3

1. Exercício 1

Execução do programa:

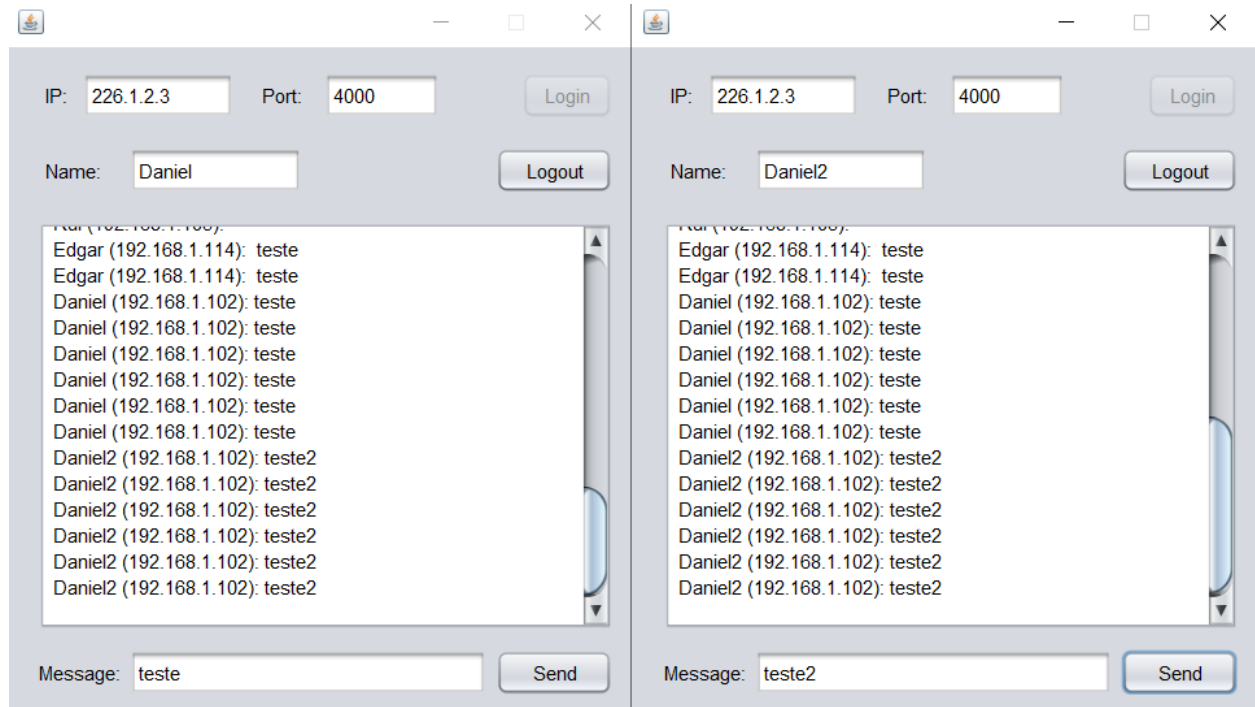


Figura 1 - Execução do código

Comunicação entre duas instâncias e colega de turma “Edgar”.

Como desenvolvi o código:

Comecei por desenvolver a interface gráfica do programa através do drag & drop do Apache NetBeans.

Depois de definir as variáveis das caixas de texto (Ex: Tip,Tport,Tname,...).

Depois defini os eventos dos botões login, logout e send.

O botão login irá verificar se os valores se encontram vazios ou não. Se algum valor estiver vazio irá aparecer a seguinte mensagem:

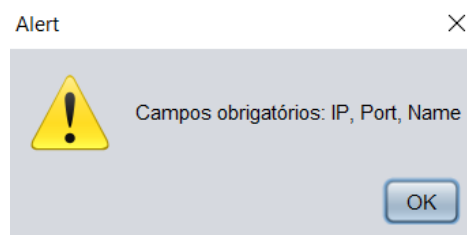


Figura 2 - Alert Message

Se todos os campos estiverem preenchidos corretamente então o botão login será automaticamente desabilitado e o botão logout e send serão automaticamente habilitados. Ainda obtém os valores ip, port e name e executa a função receive da classe ReceiveMulticast.

A função receive é executada através de uma única thread:

```
public void run() {
    receive(ip, port, textarea);
}

public void login(String ip, int port, JTextArea textarea) {
    ReceiveMulticast rm = new ReceiveMulticast(ip, port, textarea);
    t = new Thread(rm);
    t.start();
}

public void receive(String ip, int port, JTextArea textarea) {
    try {
        InetAddress in = InetAddress.getByName(ip);
        MulticastSocket s = new MulticastSocket(port);
        s.joinGroup(in);
        while (true) {
            byte buf[] = new byte[1000];
            DatagramPacket dp = new DatagramPacket(buf, buf.length);
            //while (true) {
                s.receive(dp);
                String str = new String(dp.getData());
                if (str.contains(":")) {
                    String[] message = str.split(":");
                    textarea.append(message[0] + " (" + dp.getAddress().getHostAddress() + "): " + message[1] + "\n");
                    textarea.setCaretPosition(textarea.getDocument().getLength());
                }
                str = "";
            }
        } catch (IOException e) {
        }
    }
```

Figura 3 - Código Receive

O meu código tem uma regra especial para receber mensagens pois só aceita uma mensagem se outra pessoa comunicar da seguinte forma “Nome:Mensagem” pois eu dou split à mensagem através de os “:” para conseguir introduzir na textarea o ip de quem me enviou a mensagem.

Para dar logout é muito simples, assim que o utilizador pressione no botão logout este sairá do chat através desta função:

```
public void logout() {
    try {
        s.close();
        t.stop();
    } catch (Exception e) {
    }
}
```

Figura 4 - Função logout

Esta função fecha a socket e depois termina a thread.

Para enviar mensagens é no botão send. Este irá enviar através da seguinte função:

```
public class SendMulticast {  
  
    public void send(String ip, int port, String name, String message) {  
  
        try {  
            InetAddress in = InetAddress.getByName(ip);  
            MulticastSocket s = new MulticastSocket();  
            String m = name + ":" + message;  
            byte[] send = m.getBytes();  
            DatagramPacket dp = new DatagramPacket(send, send.length, in, port);  
            s.setTimeToLive(5);  
            s.send(dp);  
            s.close();  
        } catch (IOException e) {  
        }  
    }  
}
```

Figura 5 - Função send

Por fim, o utilizador ao dar logout na interface irá acontecer o seguinte:

```
private void JlogoutActionPerformed(java.awt.event.ActionEvent evt) {  
    rm.logout();  
    this.Jlogin.setEnabled(true);  
    this.Jlogout.setEnabled(false);  
    this.Jsend.setEnabled(false);  
    this.Jip.setEnabled(true);  
    this.Jport.setEnabled(true);  
    this.Jname.setEnabled(true);  
    this.textarea.append("Logging out...\n");  
    this.textarea.setText(null);  
}
```

Figura 6 – Função/Evento Jlogout

Irá dar de volta permissões ao utilizador para mexer nas caixas de texto (ip,port,name) e irá limpar o textarea.