

# REDES DE COMPUTADORES

## Testes de Comunicação em Rede

O teste tem como objetivo efetuar a verificação da comunicação entre o cliente e o servidor por meio do protocolo TCP (Transmission Control Protocol) protocolo que garante que os dados enviados de um computador a outro chegue na ordem certa sem erros, sem perda de informações e sem falhas de conexão.

### Componentes de Teste:

**Servidor TCP (SmartHomeServerTest.cs):** Ele vai efetuar a conexão em uma porta específica, quando ele receber uma mensagem dos sensores ele responde com "OK".

### Código Sever:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Net.Sockets;
using System.Net;
using System.Text;
using System.Threading.Tasks;

namespace TesteComunicacaoRede
{
    internal class SmartHomeServer
    {
        public static void Iniciar()
        {
            TcpListener servidor = new TcpListener(IPAddress.Any, 5000);
            servidor.Start();
            Console.WriteLine("Servidor aguardando conexões...");

            while (true)
            {
                TcpClient cliente = servidor.AcceptTcpClient();
                NetworkStream stream = cliente.GetStream();

                byte[] buffer = new byte[1024];
                int bytesLidos = stream.Read(buffer, 0, buffer.Length);
                string mensagemRecebida = Encoding.UTF8.GetString(buffer, 0, bytesLidos);
```

```

        Console.WriteLine($"Recebido do cliente: {mensagemRecebida}");

        string resposta = "OK";
        byte[] dadosResposta = Encoding.UTF8.GetBytes(resposta);
        stream.Write(dadosResposta, 0, dadosResposta.Length);

        cliente.Close();
    }
}
}
}
}

```

**Cliente TCP (SmartHomeClientTest.cs):** Quando ele se conectar ao servidor ele retorna uma mensagem simulando a leitura de um sensor (ex: sensor de temperatura).

#### Código Cliente:

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Net.Sockets;
using System.Text;
using System.Threading.Tasks;

namespace TesteComunicacaoRede
{
    internal class SmartHomeCliente
    {
        public static void Iniciar()
        {
            try
            {
                TcpClient cliente = new TcpClient("127.0.0.1", 5000); // IP do servidor
                NetworkStream stream = cliente.GetStream();

                string mensagem = "Sensor: Temperatura = 25°C";
                byte[] dados = Encoding.UTF8.GetBytes(mensagem);
                stream.Write(dados, 0, dados.Length);
                Console.WriteLine("Mensagem enviada ao servidor.");

                byte[] buffer = new byte[1024];
                int bytesLidos = stream.Read(buffer, 0, buffer.Length);
                string resposta = Encoding.UTF8.GetString(buffer, 0, bytesLidos);
                Console.WriteLine($"Resposta do servidor: {resposta}");
            }
            catch { }
        }
    }
}

```

```

        cliente.Close();
    }
    catch (Exception ex)
    {
        Console.WriteLine("Erro de conexão: " + ex.Message);
    }
}
}
}

```

### Programa:

```

using System;
using TesteComunicacaoRede;

class Program
{
    static void Main()
    {
        Console.WriteLine("Digite 1 para iniciar o Servidor, 2 para iniciar o Cliente:");
        string escolha = Console.ReadLine();

        if (escolha == "1")
            SmartHomeServer.Iniciar();
        else if (escolha == "2")
            SmartHomeCliente.Iniciar();
        else
            Console.WriteLine("Opção inválida.");
    }
}

```

### Funcionamento do Sistema:

1. Inicie o servidor (SmartHomeServerTest.cs).  
Ele ficará aguardando conexões na porta 5001.
2. Execute o cliente (SmartHomeClientTest.cs).  
O cliente se conecta ao servidor e envia a mensagem:  
"Sensor: Temperatura = 25°C"
3. O servidor recebe a mensagem e responde: "OK"
4. O cliente exibe a resposta no console.

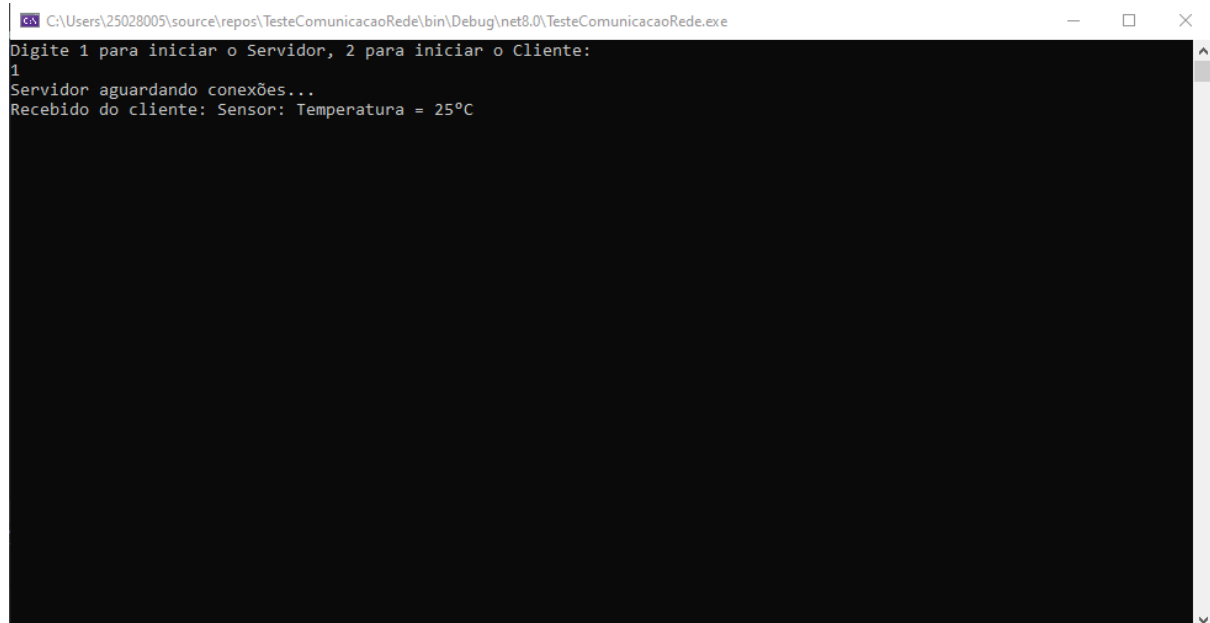
### Resultado Esperado:

O console do servidor exibe:

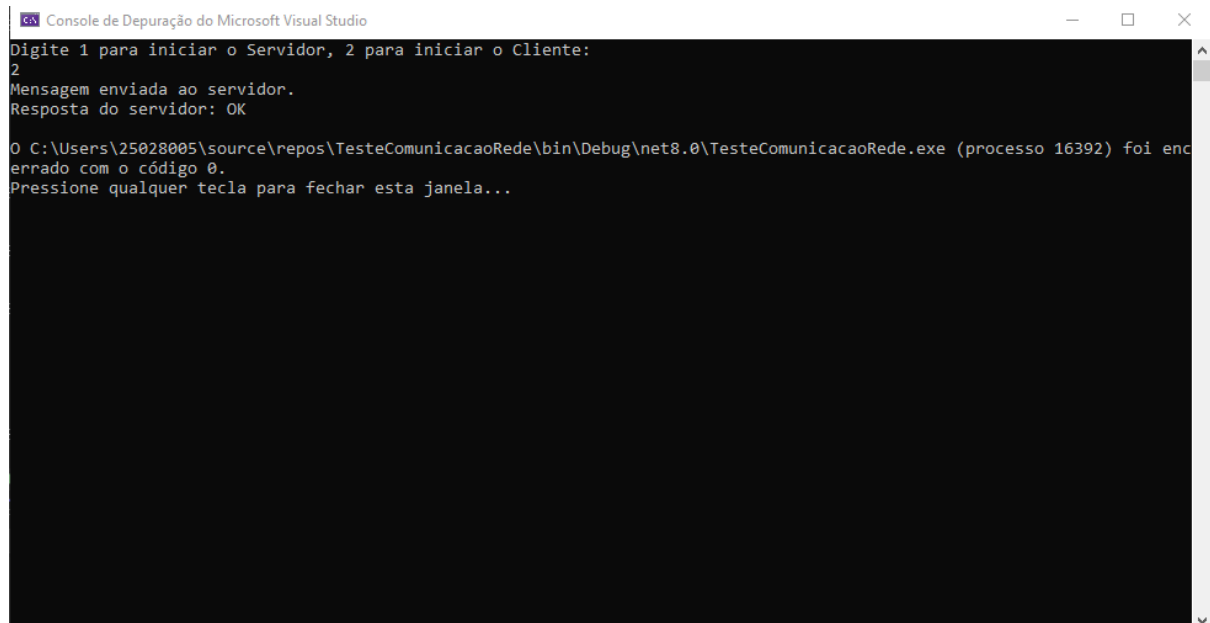
Mensagem recebida: Sensor: Temperatura = 25°C

O console do cliente exibe:

Resposta do servidor: OK



```
C:\Users\25028005\source\repos\TesteComunicacaoRede\bin\Debug\net8.0\TesteComunicacaoRede.exe
Digite 1 para iniciar o Servidor, 2 para iniciar o Cliente:
1
Servidor aguardando conexões...
Recebido do cliente: Sensor: Temperatura = 25°C
```



```
Console de Depuração do Microsoft Visual Studio
Digite 1 para iniciar o Servidor, 2 para iniciar o Cliente:
2
Mensagem enviada ao servidor.
Resposta do servidor: OK

O C:\Users\25028005\source\repos\TesteComunicacaoRede\bin\Debug\net8.0\TesteComunicacaoRede.exe (processo 16392) foi encerrado com o código 0.
Pressione qualquer tecla para fechar esta janela...
```