**Disciplina**: Conectividade de Sistemas Ciberfísicos

**Curso:**

**Nome Estudante:**

**Atividade Prática / Relatório**

**Socket API (UDP)**

# **Descrição da Atividade:**

Esta atividade consiste em implementar em Python a comunicação UDP utilizando a interface socket. O intuito desta atividade é demostrar o funcionamento do protocolo UDP, bem como as portas são mapeadas aos processos por meio da interface sockets.

**Entrega:**

O estudante deverá entregar um arquivo “.pdf” contendo as respostas da atividade proposta no item especificação.

**Especificação:**

**Exercício 1:**

1. Execute o programa (servidor.py), correspondente ao “Código Python – Servidor UDP”.
   1. Forneça uma porta para o servidor;
   2. Execute o comando “netstat -a” e localize a porta que você forneceu para servidor;
   3. Efetue um printscreen do comando anterior (item b);
2. Execute o programa (cliente.py), correspondente ao “Código Python – Cliente UDP”.
   1. Forneça um endereço IP para o servidor;
   2. Forneça uma porta para o servidor;
   3. Forneça a mensagem a ser enviada;
3. Com o servidor ativo, envie uma mensagem do cliente UDP;
4. Encerre o programa do servidor;
5. Envie outra mensagem do cliente para o servidor

**Relatório:**

Questão 1:

1. O cliente indica erro se o servidor não tiver sido iniciado primeiro?
2. O cliente é notificado se enviar uma mensagem para um servidor que não existe?
3. O cliente é notificado se o servidor for encerrado?
4. O servidor precisa de threads para ouvir mensagens de múltiplos clientes?

**Exercício 2:**

1. Altere o código do servidor para que o IP possa ser passado como argumento do BIND.

**Dica:** a porta precisa ser convertida de *string* para inteiro

porta = int(input('Entre com a porta do servidor'))

1. Lance dois servidores na mesma porta, 9999, mas em endereços diferentes: 127.0.0.2 e 127.0.0.3:

cmd

python servidor.py

python servidor.py

1. Com os servidores ativos, envie uma mensagem do cliente UDP;
2. Efetuar um *printscreen* dos testes realizados.

**Relatório:**

Questão 2:

1. O cliente pode enviar mensagens para servidores diferentes no mesmo socket?
2. O cliente consegue enviar mensagens para o endereço de Broadcast 127.255.255.255?
3. Explique a dificuldade criar um mecanismo confiável para mensagens em BROADCAST.