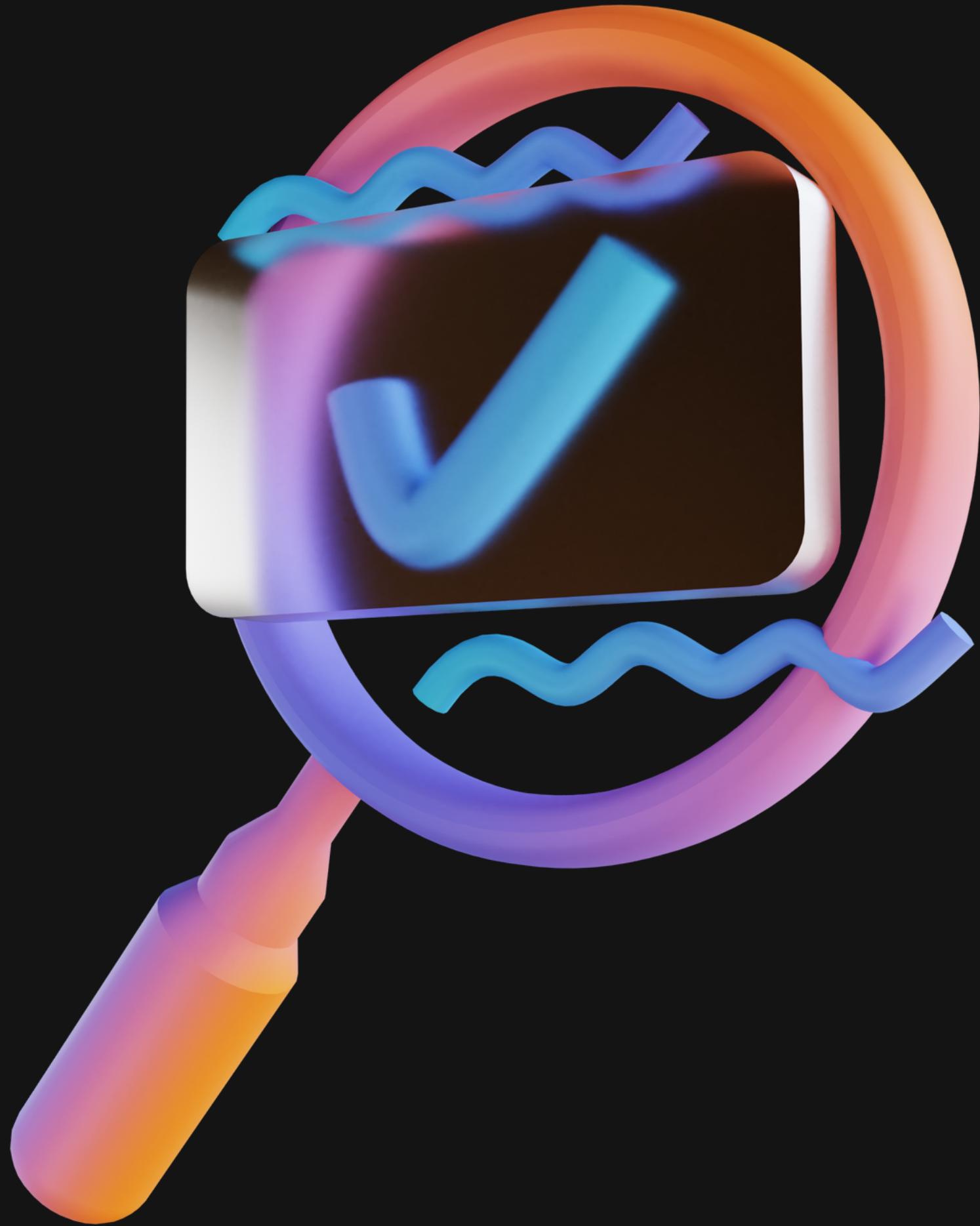




Sistemas Distribuídos

Componentes: Valtercio Santiago, Mayra Pepétua



PROJETO

Projeto criado com socket com o intuito
de enviar e receber arquivos de um
servidor usando a conexão TCP

DUAS APLICAÇÕES



**Envio de arquivos para um servidor
usando a conexão TCP**



**Recebimento de arquivos de um servidor
usando a conexão TCP**



Envio de Arquivos

Client code

```
● ● ●  
import socket  
  
# Cria um objeto de socket  
client_socket = socket.socket(socket.AF_INET,  
socket.SOCK_STREAM)  
  
# Conecta-se ao servidor  
client_socket.connect(('localhost', 8080))  
  
# Exibe mensagem de confirmação de conexão  
print("Cliente conectado ao servidor!")  
  
# Nome do arquivo que será enviado  
filename = str(input("Digite o nome do arquivo a ser  
enviado: "))  
  
# Envio do nome do arquivo ao servidor  
client_socket.send(filename.encode())
```

```
● ● ●  
# Abre o arquivo no modo de leitura binária  
with open(filename, 'rb') as file:  
  
    # Lê o conteúdo do arquivo em blocos de 1024 bytes  
    data = file.read(1024)  
  
    while data:  
  
        # Envio dos dados ao servidor  
        client_socket.send(data)  
  
        # Lê o próximo bloco de dados do arquivo  
        data = file.read(1024)  
  
    # Exibe mensagem de conclusão  
    print("Arquivo enviado com sucesso!")  
  
    # Fecha a conexão com o servidor  
    client_socket.close()
```

Server code

```
import socket

# Cria um objeto de socket
server_socket = socket.socket(socket.AF_INET,
socket.SOCK_STREAM)

# Associa o socket a um endereço e porta
server_socket.bind(('localhost', 8080))

# Espera conexões do cliente
server_socket.listen(1)

# Exibe mensagem de confirmação
print("Servidor aguardando conexão...")

# Aceita uma conexão do cliente
connection, address = server_socket.accept()

# Recebe o nome do arquivo do cliente
filename = connection.recv(1024).decode()

# Abre o arquivo no modo de escrita binária
with open(filename, 'wb') as file:

    # Recebe os dados do arquivo em blocos de 1024 bytes
    data = connection.recv(1024)

    # Enquanto houver dados a serem recebidos
    while data:
```

```
# Escreve os dados recebidos no arquivo
file.write(data)

# Recebe o próximo bloco de dados
data = connection.recv(1024)

# Recebe os dados escritos do cliente em blocos de 1024 bytes
data_written = connection.recv(1024)

# Abre o arquivo no modo de escrita
with open("escrito.txt", "w") as file:

    # Enquanto houver dados a serem recebidos
    while data_written:

        # Escreve os dados recebidos no arquivo
        file.write(data_written.decode())

        # Recebe o próximo bloco de dados
        data_written = connection.recv(1024)

    # Exibe mensagem de conclusão
    print("Arquivo recebido e salvo com sucesso!")

    # Fecha a conexão com o cliente
    connection.close()

    # Fecha o servidor
    server_socket.close()
```



Recebimento de Arquivos

Client code

```
● ● ●  
import socket  
  
# Cria um objeto de socket do tipo TCP/IP  
client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)  
  
# Conecta ao servidor que está em execução localmente na  
porta 8080  
client.connect(('localhost', 8080))  
  
# Imprime uma mensagem indicando que a conexão foi  
estabelecida com sucesso  
print("O cliente está conectado ao servidor!")  
  
# Solicita que o usuário informe o nome do arquivo que será  
enviado  
filename = str(input("Digite o nome do arquivo a ser  
enviado: "))  
  
# Envia o nome do arquivo codificado para o servidor  
client.send(filename.encode())
```

```
● ● ●  
# Abre o arquivo em modo de escrita binária  
with open(filename, 'wb') as file:  
    while 1:  
        # Recebe os dados enviados pelo servidor  
        data = client.recv(1000000)  
  
        # Se não houver mais dados, encerra o loop  
        if not data:  
            break  
  
        # Escreve os dados no arquivo  
        file.write(data)  
  
    # Imprime uma mensagem indicando que o arquivo foi enviado  
    # com sucesso  
    print("Arquivo enviado com sucesso!")  
  
    # Encerra a conexão com o servidor  
    client.close()
```

Server Code

```
import socket

# Cria um objeto socket com IPv4 e protocolo TCP
server = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

# Associa o servidor ao endereço IP e porta específicos
server.bind(('localhost', 8080))

# Coloca o servidor em modo escuta para conexões
server.listen(1)

print("O servidor funcionando na porta 8080...")
```

```
# Aceita uma conexão de entrada e retorna uma nova tupla de
# objeto do socket
connection, address = server.accept()

# Recebe o nome do arquivo a ser enviado pelo cliente
nameFile = connection.recv(1024).decode()

with open(nameFile, 'rb') as file: # Abre o arquivo em modo
    # de leitura binário

        for data in file: # Lê o arquivo em blocos de tamanho
            # definido

                # Envia cada bloco do arquivo para o cliente
                connection.sendall(data)

print("Arquivo enviado com sucesso!")
```