

**By:**

Adil Zainul Syed

## **Socket Network Programming – Client-Server Chat Application**

### **Objective:**

To implement client-server communication using TCP sockets where multiple clients can send and receive messages from a central server.

### **Tools Required:**

- Language: Python 3
- Libraries: socket, threading

### **Theory:**

Socket programming allows communication between two processes — typically on different machines — over a network. A server waits for client requests, while a client initiates a connection. TCP sockets ensure reliable, ordered, and error-checked communication.

### **Algorithm:**

#### **Server:**

1. Create a socket using `socket.socket()`.
2. Bind it to an IP address and port.
3. Listen for incoming connections using `listen()`.
4. Accept client connections in a loop.
5. For each client, start a new thread to handle messages concurrently.
6. Broadcast messages to all connected clients.

#### **Client:**

1. Create a socket and connect to the server's IP and port.
2. Send messages to the server.
3. Continuously listen for messages from the server.

### **Program Code:**

#### **a) Server (server.py):**

```
import socket

import threading
```

```

clients = []

client_ids = {} # Map socket to client ID

client_count = 0 # Incremental ID counter


def handle_client(client_socket):

    client_id = client_ids[client_socket]

    while True:

        try:

            message = client_socket.recv(1024).decode()

            if not message:

                break

            formatted_message = f"{client_id}: {message}"

            print(formatted_message)

            broadcast(formatted_message, client_socket)

        except:

            print(f"{client_id} disconnected.")

            clients.remove(client_socket)

            del client_ids[client_socket]

            client_socket.close()

            break


def broadcast(message, sender_socket):

    for client in clients:

        if client != sender_socket:

            client.send(message.encode())


def main():

    global client_count

```

```

server = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

server.bind(('localhost', 12345))

server.listen(5)

print("Server listening on port 12345...")


while True:

    client_socket, addr = server.accept()

    client_count += 1

    client_id = f"Client-{client_count} (socket-{client_socket.fileno()})"

    client_ids[client_socket] = client_id

    clients.append(client_socket)

    print(f"Connected to {addr} as {client_id}")

    client_socket.send(f"You are connected as {client_id}".encode())


    threading.Thread(target=handle_client, args=(client_socket,)).start()


if __name__ == "__main__":

    main()

```

## b) Client (client.py):

```

import socket

import threading


def receive_messages(client_socket):

    while True:

        try:

            message = client_socket.recv(1024).decode()

            if not message:

                break

            print("\n" + message)

```

```
except:

    print("Disconnected from server.")

    client_socket.close()

    break
```

```
def main():

    client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

    client.connect(('localhost', 12345))

    threading.Thread(target=receive_messages, args=(client,)).start()

    while True:

        message = input("")

        client.send(message.encode())

if __name__ == "__main__":

    main()
```

## Output:

a) Setting up the Server side –

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

Loading personal and system profiles took 3065ms.
(base) PS C:\Users\adilz> cd C:\Users\adilz\OneDrive\Documents\asssignments_or_projects\Socket-Network-Programming-Client-Server-Chat-Application
(base) PS C:\Users\adilz\OneDrive\Documents\asssignments_or_projects\Socket-Network-Programming-Client-Server-Chat-Application> python server.py
Server listening on port 12345...
```

b) Setting up the Client side (2 Clients, same for both) –

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

Loading personal and system profiles took 3031ms.
(base) PS C:\Users\adilz> cd C:\Users\adilz\OneDrive\Documents\asssignments_or_projects\Socket-Network-Programming-Client-Server-Chat-Application
(base) PS C:\Users\adilz\OneDrive\Documents\asssignments_or_projects\Socket-Network-Programming-Client-Server-Chat-Application> python client.py

You are connected as Client-1 (socket-500)
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

Loading personal and system profiles took 3019ms.
(base) PS C:\Users\adilz> cd C:\Users\adilz\OneDrive\Documents\asssignments_or_projects\Socket-Network-Programming-Client-Server-Chat-Application
(base) PS C:\Users\adilz\OneDrive\Documents\asssignments_or_projects\Socket-Network-Programming-Client-Server-Chat-Application> python client.py

You are connected as Client-2 (socket-516)
```

c) On the Server side after establishment of network connection between 2 clients -

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

Loading personal and system profiles took 3079ms.
(base) PS C:\Users\adilz> cd C:\Users\adilz\OneDrive\Documents\asssignments_or_projects\Socket-Network-Programming-Client-Server-Chat-Application
(base) PS C:\Users\adilz\OneDrive\Documents\asssignments_or_projects\Socket-Network-Programming-Client-Server-Chat-Application> python server.py
Server listening on port 12345...
Connected to ('127.0.0.1', 60022) as Client-1 (socket-500)
Connected to ('127.0.0.1', 60028) as Client-2 (socket-516)
```

d) Chatting between the clients after establishment of connection on the network –

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

Loading personal and system profiles took 3079ms.
(base) PS C:\Users\adilz> cd C:\Users\adilz\OneDrive\Documents\asssignments_or_projects\Socket-Network-Programming-Client-Server-Chat-Application
(base) PS C:\Users\adilz\OneDrive\Documents\asssignments_or_projects\Socket-Network-Programming-Client-Server-Chat-Application> python server.py
Server listening on port 12345...
Connected to ('127.0.0.1', 60022) as Client-1 (socket-500)
Connected to ('127.0.0.1', 60028) as Client-2 (socket-516)
Client-1 (socket-500): Hello, who are you?
Client-2 (socket-516): Rajnikanth
Client-1 (socket-500): wait what?!
Client-2 (socket-516): Super Star Rajnikanth
Client-1 (socket-500): Sir, Autograph?!
Client-2 (socket-516): Here.... Take ra (gives autograph in style)
Client-1 (socket-500): Wow thanks sir nice to meet you my name is Adil Zainul Syed from CSE Core Section - C
Client-2 (socket-516): Woahhh, I watch your videos on youtube.... Lore N' Order right?! 😊
Client-1 (socket-500): yes 😊
Client-2 (socket-516): Wow even I am a big fan of yours! Autograph please 🙏😊

Windows PowerShell
Hello, who are you?
Client-2 (socket-516): Rajnikanth
wait what?!

Client-2 (socket-516): Super Star Rajnikanth
Sir, Autograph?!

Client-2 (socket-516): Here.... Take ra (gives autograph in style)
Wow thanks sir nice to meet you my name is Adil Zainul Syed from CSE Core Section - C

Client-2 (socket-516): Woahhh, I watch your videos on youtube.... Lore N' Order right?! 😊
yes 😊

Client-2 (socket-516): Wow even I am a big fan of yours! Autograph please 🙏😊

Windows PowerShell
You are connected as Client-2 (socket-516)
Client-1 (socket-500): Hello, who are you?
Rajnikanth

Client-1 (socket-500): wait what?!
Super Star Rajnikanth

Client-1 (socket-500): Sir, Autograph?!
Here.... Take ra (gives autograph in style)

Client-1 (socket-500): Wow thanks sir nice to meet you my name is Adil Zainul Syed from CSE Core Section - C
Woahhh, I watch your videos on youtube.... Lore N' Order right?! 😊

Client-1 (socket-500): yes 😊
Wow even I am a big fan of yours! Autograph please 🙏😊
```

## Conclusion:

Successfully implemented client-server communication using sockets where multiple clients can interact with each other through a central server in real time.