CHAT APPLICATION (CHAT ROOM) USING SOCKET PROGRAMMING IN PYTHON

A COURSE PROJECT REPORT

By

KARTIK JAISWAL (RA2011031010060)
SANJU DAS (RA2011031010063)
NAMAN MITHWANI(RA2011031010054)
SAMBHRAM GHOSH(RA2011031010036)

Under the guidance of

S. THENMALAR

In partial fulfilment for the Course

of

18CSC302J - COMPUTER NETWORKS

in NWC

FACULTY OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

Kattankulathur, Chenpalpattu District

NOVEMBER 2022

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that this mini project report "**CHAT APPLICATION (CHAT ROOM) USING SOCKET PROGRAMMING IN PYTHON**" is the bonafide work **KARTIK JAISWAL(RA2011031010060), NAMAN MITHWANI (RA2011031010054), SANJU DAS (RA2011031010063), SAMBHRAM GHOSH(RA2011031010036)** who carried out the project work under my supervision.

SIGNATURE

Dr. S. Thenmalar,
Course Cordinator,
Associate Professor,
Networking and Communication,
SRM Institute of Science and Technology

ABSTRACT

Chat refers to the process of communicating, interacting and/or exchanging messages over the Internet. It involves two or more individuals that communicate through a chat-enabled service or software. Chat may be delivered through text, audio or video communication via the Internet. A chat application has basic two components, viz server and client. A server is a computer program or a device that provides functionality for other programs or devices. Clients who want to chat with each other connect to the server.The chat application we are going to make will be more like a chat room, rather than a peer to peer chat. So this means that multiple users can connect to the chat server and send their messages. Every message is broadcasted to every connected chat user

ACKNOWLEDGEMENT

We express our heartfelt thanks to our honorable **Vice Chancellor Dr. C. MUTHAMIZHCHELVAN**, for being the beacon in all our endeavors.

We would like to express my warmth of gratitude to our **Registrar Dr. S. Ponnusamy,** for his encouragement

We express our profound gratitude to our **Dean (College of Engineering and Technology) Dr. T. V.Gopal,** for bringing out novelty in all executions.

We would like to express my heartfelt thanks to Chairperson, School of Computing **Dr. Revathi Venkataraman,** for imparting confidence to complete my course project

We wish to express my sincere thanks to **Course Audit Professor Dr.Annapurani Panaiyappan, Professor and Head, Department of Networking and Communications** and **Course Coordinators** for their constant encouragement and support.

We are highly thankful to our my Course project Faculty **S.THENMALAR , PROFESSOR , NWC,** for his/her assistance, timely suggestion and guidance throughout the duration of this course project.

We extend my gratitude to our **HoD <Name> <Designation>, <Department>** and my Departmental colleagues for their Support.

Finally, we thank our parents and friends near and dear ones who directly and indirectly contributed to the successful completion of our project. Above all, I thank the almighty for showering his blessings on me to complete my Course project.

TABLE OF CONTENTS

INTRODUCTION

1. Scenario Description

A network has to be designed for a small business organization which has 100 users. The organization hosts an e-commerce application on a server which is accessible to internet users using https and with a public IP address. We have to setup a server which listens on Port 443 (HTTPS) only. The private IP address should be masked by NAT on the company router so that the external users will know only the public IP of the server. This will ensure added security for the server. The employees in the organization will be able to use the server without any restriction since they are directly routed to the same network via the company router. The departments can also intercommunicate with one another without any restriction. The public internet users will only know a public address of the company router to connect to the server. They can communicate with each other, since they are on a public network. However, they cannot know the private addresses of the company devices.

LITERATURE SURVEY

1. Literature Survey

It is important to pay heed to the company's requirements, completing which the design can be improved further upon. Priority number one is to have the functional requirements in place. These are requirements that are essential to the business and cannot be compromised upon. The non -functional requirements can then later be added on demand, as these are not essential for the business to run their absence may not hinder the company operations drastically. The absolute essential features are listed as follows. A high-speed reliable secure internet connection is essential as it will ensure there are no gaps in communication between stakeholders, and will ensure that the transaction take place in a smooth manner.
Switches/ethernet hubs help by ensuring all round connectivity so that everything is available to everyone on demand

Security is also of prime importance as it takes care of two things:

a) that there is no breach in the ecosystem and that the company data is safe and secure, and not out in the open

b) that there is no malware/spyware that will jeopardize the company's future performance by corrupting their data assets

Communication between the organization and the outside world is facilitated by use of a telephone, or other pathways that give the company a specific name or brand value.

2. References

https://smallbusiness.chron.com/advantages-chat-room-70640.html

https://code.visualstudio.com/

https://www.askpython.com/python/examples/create-chatroom-in-python

https://projectsgeek.com/2017/12/simple-chat-room-system-project.html

https://code-projects.org/public-chat-room-in-php-with-source-code/

REQUIREMENT ANALYSIS

1. Software requirements

Software's can be defined as programs which run on our computer.it act as petrol in the vehicle. It provides the relationship between the human and a computer. It is very important to run software to function the computer. Various software's are needed in this project for its development.
Which are as follows-?

Operating system-Windows 11

Others-Visual Studio

We will be using visual basic as our front hand because it is easier to use and provides features to the users which is used for the development of the project.

2. Hardware Requirement

In hardware requirement we require all those components which will provide us the platform for the development of the project. The minimum hardware required for the development of this project is as follows-

- Ram- minimum 128 MB
- Hard disk-minimum 5 GB
- Processor-Pentium 3
- Floppy drive 1.44" inch
- CD drive

These all are the minimum hardware requirement required for our project. We want to make our project to be used in any. Type of computer therefore we have taken minimum configuration to a large extent.128 MB ram is used so that we can execute our project in a least possible RAM.5 GB hard disk is used because project takes less space to be executed or stored. Therefore minimum hard disk is used. Others enhancements are according to the needs.

IMPLEMENTATION

1. Code:

Connected with ('127.0.0.1', 64847)
Nickname is Kartik
Connected with ('127.0.0.1', 64855)
Nickname is Sanju
Connected with ('127.0.0.1', 64861)
Nickname is Sambhram

Server code:

```
import socket
import threading
# Connection Data
host = '127.0.0.1'
port = 55555

# Starting Server
server = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
server.bind((host, port))
server.listen()

# Lists For Clients and Their Nicknames
clients = []
nicknames = []

# Sending Messages To All Connected Clients
```

```
def broadcast(message):
    for client in clients:
        client.send(message)
# Handling Messages From Clients
def handle(client):
    while True:
        try:
            # Broadcasting Messages
            message = client.recv(1024)
            broadcast(message)
        except:
            # Removing And Closing Clients
            index = clients.index(client)
            clients.remove(client)
            client.close()
            nickname = nicknames[index]
            broadcast('{} left!'.format(nickname).encode('ascii'))
            nicknames.remove(nickname)
            break
# Receiving / Listening Function
def receive():
    while True:
        # Accept Connection
        client, address = server.accept()
        print("Connected with {}".format(str(address)))

        # Request And Store Nickname
        client.send('NICK'.encode('ascii'))
        nickname = client.recv(1024).decode('ascii')
        nicknames.append(nickname)
        clients.append(client)
```

```
        # Print And Broadcast Nickname
        print("Nickname is {}".format(nickname))
        broadcast("{} joined!".format(nickname).encode('ascii'))
        client.send('Connected to server!'.encode('ascii'))

        # Start Handling Thread For Client
        thread = threading.Thread(target=handle, args=(client,))
        thread.start()
receive()
```

Client code:

```
import socket
import threading
# Choosing Nickname
nickname = input("Choose your nickname: ")

# Connecting To Server
client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client.connect(('127.0.0.1', 55555))
# Listening to Server and Sending Nickname
def receive():
    while True:
        try:
            # Receive Message From Server
            # If 'NICK' Send Nickname
            message = client.recv(1024).decode('ascii')
            if message == 'NICK':
                client.send(nickname.encode('ascii'))
            else:
                print(message)
        except:
```

```
            # Close Connection When Error
            print("An error occured!")
            client.close()
            break
# Sending Messages To Server
def write():
    while True:
        message = '{}: {}'.format(nickname, input(''))
        client.send(message.encode('ascii'))
# Starting Threads For Listening And Writing
receive_thread = threading.Thread(target=receive)
receive_thread.start()

write_thread = threading.Thread(target=write)
write_thread.start()
```

RESULTS AND DISCUSSION

1. Results:

```
PS C:\Users\kjkar\OneDrive\Desktop\mini proj> python server.py
Connected with ('127.0.0.1', 64847)
Nickname is Kartik
Connected with ('127.0.0.1', 64855)
Nickname is Sanju
Connected with ('127.0.0.1', 64861)
Nickname is Sambhram
```

```
PS C:\Users\kjkar\OneDrive\Desktop\mini proj» python client.py
Choose your nickname: Sanju
Sanju joined!
Connected to server!
Sambhram joined!
Kartik: Hello everyone!!
Hii, I am Sanju
Sanju: Hii, I am Sanju
Sambhram: Jai Shree Ram, friends
```

2. Discussion:

As we can observe from the above results, peer to peer communication between the three participants was successfully established. By using nicknames, it is easier for users to identify other users, and updates the clients as to who is currently present in the chatroom. This proves to be an excellent method of communication amongst users as it is simple, efficient and fast

CONCLUSION AND FUTURE ENHANCEMENT

There is always a room for improvements in any software package, however good and efficient it may be done. But the most important thing should be flexible to accept further modification. Right now we are just dealing with text communication. In future this software may be extended to include features such as:

- File transfer: this will enable the user to send files of different formats to others via the chat application.

- Voice chat: this will enhance the application to a higher level where communication will be possible via voice calling as in telephone.

- Video chat: this will further enhance the feature of calling into video communication.

REFERENCES

- https://smallbusiness.chron.com/advantages-chat-room-70640.html
- https://code.visualstudio.com/
- https://www.askpython.com/python/examples/create-chatroom-in-python
- https://projectsgeek.com/2017/12/simple-chat-room-system-project.html
- https://code-projects.org/public-chat-room-in-php-with-source-code/