Development of Jmessenger

[1]Mr.Hande Sushant S, [2]Mr.Tambe Sandip B, [3]Mr. Ghorpade Nilesh N,
[4]Mr.Vaidya Ajinkya S.

[1,2,3,4] Pravara Rural College of Engineering, Loni Tal-Rahata, Dist-Ahmednagar (India)

Abstract: **The topic presents JMessenger an application is one of easiest way to communicate with a users through Local Area network. No network connection is required. It only requires the master machine's IP address and you will be able to communicate to others members through LAN. It can help you to communicate with your friends even you both do not have network connection. As local area network which connect no of users to each other and also client to master computer. So we have used the same concept here we are connecting two client or client and server with each other and by providing the IP address we can chat with each other. and also the file transfer application is also implemented where a user can upload a file or save a file. We are providing the path to the root folder where we have stored our file to the user who wants to download particular file.**

Keywords: **JMessenger, SIGAR API.**

1.   INTRODUCTION

JMessanger is a application developed in java. To start the communication you must be connected with the server after that your message can broadcast to the client though which we want to communicate. Chat system is a complete master/slave chat solution, independent chat system. JMessenger application is a GUI interface. This application makes it easy to found people on a network and communicate with them easily without being online and makes net message transmission easy. The file sharing application is also implemented where a user can save a file or upload a file. We are providing the path to the root folder where we have stored our file to the user who wants to download particular file. This is a beta attachment in this chat system and need to improve in further steps. Some features of the application are as follows: This application is used for sending messages from one client to next client. In this, master machine takes a message from the sender client and sends to reciever client. If any slave wants to join the communication then he runs the client application on client side, enters the user name and hits the login button and starts the chatting with other client. In this we are maintaining the list of user names also. After creating the connection with master PC, slave sends the user name to master PC and it store it in the array list and sends this array list to all slaves. Same like this when any slave has logged out then server remove this name from the array list and send this updated array list to every client.

In this application we are providing a file transfer system where a registered user can view or download files of other user.

1.1 Need:

File sharing on the other hand is good tool to share those file which your friend wants from you or any other file which can be useful for other. The user can see this file after login to system and can download this file .It is one of the easiest way as it doesn't require much knowledge for sharing and downloading.

1.2. Features:

1) This application is used for sending messages from one slave to another slave. In this, master PC takes a message from the sender client and sends to reciever client. If any client wants to join the communication then he runs the client application on slave side, enters the user name and hits the login button and starts the chatting with other client.

2) In this we are maintaining the list of user names also. After creating the connection with master PC, slave sends the user name to master PC and master PC store it in the array list and sends this array list to all slaves. Same like this

when any slave has logged out then server remove this name from the array list and send this updated array list to every client.

3) In this application we are providing a file sharing system where a registered user can view or download files of other user.

1.3 Applications:

1) LAN Messenger is P2P chat application for internet communication and it require a server.

2) A variety of handy features are supported including notification, personal and group messaging with encryption, file transfer and message login.

2.   LITERATURE SURVEY

On the Internet, chatting is talking to other people who are using the Internet at the same time you are. In some cases, a private chat can be arranged between two parties who meet initially in a group chat. The JMessenger can provide the feature of offline chatting. In this messenger there is no need of internet connection.

2.1 Socket:

A socket is one endpoint of a two-way communication link between two programs running on the networked computers. A socket is assign to a port number so that the TCP layer can identify the application that data is destined to be sent to. Normally, a server runs on a specific computer and has a socket that is bound to aperticular port number. The master PC just waits, listening to the socket for a client to make a connection establishment request. If everything goes well, the master PC accepts the connection. Upon acceptance, the master PC gets a new socket bound to the same local port and also has its remote endpoint set to the address and port of the slave. It requires a new socket so that it can continue to listen to the original socket for connection requests while tending to the needs of the connected client. Multi User Chat System is an application through which the user can communicate with other users connected in the same network area (LAN). This works under any operating system and is programmed in java. To establish a communication between the systems, we require simple socket connections in order to connect them in a network. Socket programming uses the client socket and server socket methods to connect the local host to the named host and port. The communication between various users is done using server client model. Several client machines are connected to their dedicated server ports and communication is established.
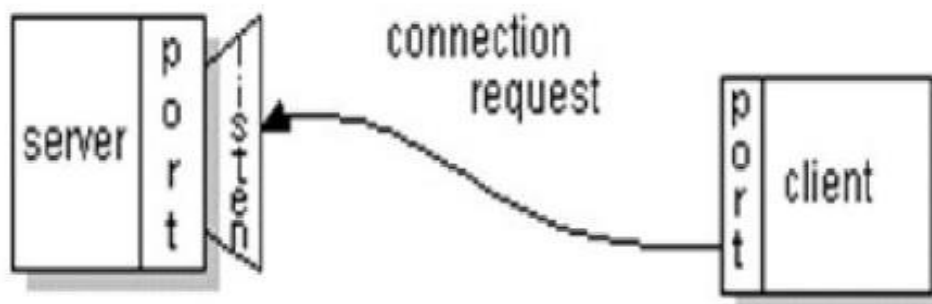
Fig.1 Request from Client to Server

We use the lowest level of networking techniques used in java, though there are no of ways of connecting slave with the master PC. Hence, we need simple classes of networking package in order to establish a server client prototype. To have a basic server client connection through java programming language, we require the concepts of java like,

1. Network packages

2. Thread classes

3. AWT tool kit

4. Event handling.

A precursor of LAN messengers is the Unix command, and similar facilities on privious systems which enabled multiple users on one host system to directly talk with each other.

1. According to the survey made on the network used in medium scale industries and schools.

2. LAN/WLAN with client-server mode is mostly used network in small scale industries.

3. Here we are implementing it in data transfer with file sharing.

4. The server should be connected for login.

5. The client must have knowledge about networking.

6. The database connectivity of the project should be done properly so data can be retrieved easily.

7. He/ she must have knowledge about java.

8. The jdk should be installed in the pc on which the project is running.

9. The Wamp server should be in running state.

CONCLUSION

The primary goal of this project is to give an idea about offline Chat. This project has given us an in depth information about java networking and its applications in day today life. Thus we are going to developed one application which can perform following tasks:

1. Offline Chatting

2. Multimedia File Transfer

3. Network Monitoring

REFERENCES

[1] SumanSrinivasan, ArezuMoghadam, Henning Schulzrinne, BonAHA: Service Discovery Framework for Mobile Ad-Hoc Applications. IEEE Consumer Communications Networking Conference 2009 (CCNC09), Las Vegas, USA, January 2009.

[2] SumanSrinivasan, ArezuMoghadam, Se Gi Hong, Henning G Schulzrinne, 7DS -Node Cooperation and Information Exchange in Mostly Disconnected Networks. IEEE International Conference onCommunications (ICC), Glasgow, Scotland, Jun 2007.

[3] Zero-Con_guration Networking protocols,http://www.zeroconf.org/

[4] ICQ, http://www.icq.com/