

Leveraging VOIP on Local Area Network using Java Media Framework

George Asante
Department of Information
Technology Education
University of Education,
Winneba
Kumasi-Ghana

J.B. Hayfron-Acquah
Department of Computer
Science
KNUST
Kumasi-Ghana

K. Riverson
CSIR
Accra

ABSTRACT

Voice over Internet Protocol (VoIP) is a standard for taking analogue audio signals, and turning them into digital data that can be transmitted over a network. VoIP has become an important factor in network communication. It has a lower operational cost, greater flexibility, and a variety of enhanced applications. VoIP is time – based. To ensure real-time transmission, Real-Time Transmission Protocol (RTP) is used on top of User Datagram Protocol (UDP). RTP provides end-to-end network transport functions suitable for applications transmitting real-time data, such as audio, video or simulation data, over multicast or unicast network services. Java Media Framework (JMF) is an Application Programming Interface (API) that uses RTP and therefore ideal for time-based media. Thus, the topic: *leveraging VOIP on LAN using Java Media Framework*. The research aims at designing a system that will allow users to communicate over a data network. That is to be able to send text, make voice and video call, and transfer file over a network. The system uses client/server architecture. The architecture is a 3-tier: the client, the main server and the database server. The system designed could be used on Android mobile phones and computers with Windows operating system. The outcome of the research will allow users to communicate at virtually no cost. The product will also put the network of various organizations into full utilization.

Keywords

VOIP, Java Media Framework, Public Switched Telephone Network (PSTN), User Datagram Protocol (UDP), Real-Time Transmission Protocol (RTP), Application Programming Interface (APIs).

1. INTRODUCTION

1.1 Background Study

VOIP is a standard for converting analog audio signals into digital data that can be transmitted over the Internet (or simply a data network) rather than traditional Public Switched Telephone Network (PSTN). VoIP has become an important development in network communication. It is a rapidly growing Internet service. It has lower operational costs, greater flexibility, and a variety of enhanced applications. VoIP technology uses the Internet's packet-switching capabilities to provide phone service.

The internet is an Internet Protocol (IP)-based network. IP-based networks are connectionless, packets switched networks. Transmission by the connectionless technology has no guarantee that the data is received by the destination. To avert this problem, VOIP communication should be connection oriented. In a connection-oriented communication, there is a guaranteed delivery of data: any data that is not received by the destination system is re-sent by the sending device.

Connection-oriented approach introduces a lot of delay in transmission. Since VoIP is time – based, using the connection –oriented approach is not the best. The known protocol that ensures connectionless-oriented transmission is User Datagram Protocol (UDP). Though UDP transmission is not secured, it is ideal protocol for transmitting time – based media.

To ensure real-time transmission, Real-Time Transmission Protocol (RTP) is used on top of UDP. RTP provides end-to-end network transport functions suitable for applications transmitting real-time data, such as audio, video or simulation data, over multicast or unicast network services

Java has a lot of Application Programming Interface (APIs) that help programmers during coding. These APIs provide a lot of abstraction for the programmer. One of these APIs is the Java Media Framework (JMF). JMF uses RTP and it is therefore ideal for time-based media such as audio and video. JMF rich features make it prudent to integrate it in VOIP applications. Thus, the topic: *leveraging VOIP on LAN using Java Media Framework*.

1.2 Statement of the Problem

Some of the communication methods used at various institutions include electronic mailing, chatting, text messaging and voice calls using mobile phones. These methods are comparably costly and many a time unavailable to the users.

When staff wants to share files, some of the options available to them are to use a Universal Serial Bus (USB) drive or external hard disk drive. The problem with these options is that both parties need to be at the source or destination of the file. Although one can use e-mail attachment technology to transfer files, the size of the file to be transmitted is always limited to say 25 MB.

When a staff wants to call other staff at different department, the staff needs to use his or her mobile phone which he or she will be charged by his or her network operator.

To avert this situation, the researcher decided to embark on a research that will allow users to communicate (make voice call, make video call, send text, and transfer files) over a network without paying any fee.

1.3 The Purpose of the Study

The purpose of the study was to design a system that will allow users to communicate over a data network.

The specific objectives are:

- I. To design a client software to be installed on participating mobile phones and computers.
- II. To make client software communicate (i.e. transfer

file, send text, make video and voice call) with other clients over a network.

- III. To have a server application that will keep track of registered clients.
- IV. To make clients detect the availability of other clients.

2. RELATED WORKS

2.1 Analysis of Existing System

There are a lot of systems that enable one to make call, send text and transfer files over a network. Such systems include Skype, Yahoo! messenger, Google Talk, Facebook, Twitter, and Inter- com system. But these systems have their own drawbacks.

Below are some of these systems, their features and major drawbacks.

2.1.1 Skype

Skype is a peer-to-peer VoIP client developed in 2003 by the organization that created Kazaa. Skype claims that it can work almost seamlessly across Network Address Translations (NATs) and firewalls and has better voice quality than other VoIP clients. It encrypts calls end-to-end, and stores user information in a decentralized fashion. Skype also supports instant messaging and conferencing (Salman, 2004).

Though Skype has a lot of features such as text chat, voice and video call, calling landlines and sending files, the major problem with Skype is that it is purely internet-based. That is, if you are not connected to the internet you cannot use the application. Even when connected to the internet, call quality from Skype may suffer if you have a slower-than-average Internet connection.

Users of Skype are provided with only the client application while the developer controls the server application. This architecture makes it unsuitable for organisations that want to communicate internally.

2.1.2 Yahoo! Messenger

Yahoo! Messenger is an all-in-one communication tool, and anyone, anywhere in the world, with a computer and Internet connection can use it. With Yahoo! Messenger, you can:

- I. Instantly message friends in real time.
- II. Call and talk with your friends. You can call a friend's computer, mobile phone, and even the old-fashioned home phone.
- III. Share your favorite photos.

Although Yahoo Messenger provides many cool features such as SMS messages and video chat, there are severe limitations that will make you regret your choice if you go with Yahoo Messenger.

I. Need a Yahoo Account

The first requirement you will notice when you sign up is that you need to sign up for a Yahoo account. That means a new username and a new password to remember. While this is perfectly reasonable, not everyone has a Yahoo account.

II. Need Your Friends to Have a Yahoo Account

How many of your friends are using Yahoo Messenger themselves? Hopefully at least one or you will mostly be chatting with yourself.

III. SMS Limit

Although Yahoo advertises that they let you communicate with your friends' cell phones via SMS, there appears to be a limitation on how many messages you can send. After you have sent 3 messages to a particular cell phone in a single session, you will begin receiving error messages.

IV. No Customer Support

If you are having trouble with the Yahoo Messenger service, their support team consists of an internet bot that you can chat with. If you have ever chatted with a bot, you probably know that they are almost never helpful. The only function that these bots seem to serve is to ask you to re-word your question because it cannot understand you.

(http://help.yahoo.com/tutorials/ms8/mess/im_setup1.html
Accessed: 9/7/12 8:40 pm)

2.1.3 Google Talk

Google Talk is an instant messaging service that provides both text and voice communication. Because the Google Talk servers communicate with clients using an open protocol, Extensible Messaging and Presence Protocol (XMPP), the service can also be accessed using any other client that supports XMPP (http://en.wikipedia.org/wiki/Google_Talk
Accessed 9/7/12 8:43 pm).

However, Google Talk requires a Gmail username and password before use. And it is also internet – based only.

All these existing systems are internet – based. This means they cannot be used without internet connectivity. It was based on this drawback that the researcher decided to embark on the research “*Leveraging VoIP on LAN using JMF*”.

2.2 Overview of VOIP

Voice communication has been continually evolving since Alexander Bell's discovery. For a long period of time, circuit switched networks dominated the transmission of voice. Circuit switched networks were also used as a medium for data transmission. The picture today is totally different, with packet switched networks supporting both data and voice communications.

An emerging technology that uses packet switched networks for voice transmission is Voice over Internet Protocol (VoIP) telephony. VoIP is the family of technologies that allows IP networks to be used for voice applications, such as telephony, voice instant messaging, and teleconferencing. VoIP defines a way to carry voice calls over an IP network, including the digitization and packetization of the voice streams (Tiantioukas, 2007).

The advantages that make this technology preferable to traditional telephony are low cost, use of the existing infrastructure, and the ability to add new applications without additional cost.

2.2.1 Different Forms of VOIP

VoIP is basically a clever "reinvention of the wheel." VoIP is a revolutionary technology that has the potential to completely rework the world's phone systems (<http://communication.howstuffworks.com/ip-telephony.htm>.
Accessed 5/10/11 9:00 am).

The interesting thing about VoIP is that there is not just one way to place a call. There are three different "flavors" of VoIP service in common use today:

- i. Analog Telephone Adaptor (ATA): The simplest and most common way is through the use of a device

called an ATA. The ATA allows you to connect a standard phone to your computer or your Internet connection for use with VoIP. The ATA is an analog-to-digital converter. It takes the analog signal from your traditional phone and converts it into digital data for transmission over the Internet. Providers like Vonage and AT&T Call Vantage are bundling ATAs free with their service. You simply crack the ATA out of the box, plug the cable from your phone that would normally go in the wall socket into the ATA, and you're ready to make VoIP calls. Some ATAs may ship with additional software that is loaded onto the host computer to configure it; but in any case, it is a very straightforward setup

- ii. IP Phones: These specialized phones look just like normal phones with a handset, cradle and buttons. But instead of having the standard RJ-11 phone connectors, IP phones have an RJ-45 Ethernet connector. IP phones connect directly to your router and have all the hardware and software necessary right onboard to handle the IP call. Mobile phones with Wi-Fi allow subscribing callers to make VoIP calls from any Wi-Fi hotspot.
- iii. Computer-to-computer: This is certainly the easiest way to use VoIP. You do not even have to pay for long-distance calls. There are several companies offering free or very low-cost software that you can use for this type of VoIP. All you need is the software, a microphone, speakers, a sound card and an Internet connection; preferably a fast one like you would get through a cable or Digital Subscriber Line (DSL) modem. Except for your normal monthly Internet Service Provider (ISP) fee, there is usually no charge for computer-to-computer calls, no matter the distance
 (<http://www.tecnicontrol.pt/en/wiki/item.html?id=40-the-voip-pabx-or-ip-pabx>. Accessed 24/02/15 1:00 pm.).

2.2.2 How VOIP Works

VoIP works by digitizing voice in data packets, sending them and reconvert them in voice at destination. Digital format can be better controlled: it can be compressed, routed, converted to a new better format, and so on; also digital signal is more noise tolerant than the analog one.

TCP/IP networks are made of IP packets containing a header (to control communication) and a payload to transport data: VoIP use it to go across the network to destination.
Voice (source) -- ADC ----- Network ----- DAC -- Voice (destination)

As shown in Figure 2.1, the voice of one user is digitized, compressed and then packetized before being sent through a wired or wireless communication channel to the other user. At the other end, the opposite procedure is followed: the received packet is depacketized, decompressed, converted to analog form and then played back to the user. In order for the conversation to be natural, the same procedure must be followed in both directions so a full duplex communication is established. The Internet Voice can allow you to make a call directly from a computer, a special VoIP phone, or a traditional phone using an adapter. If you make a call using a phone with an adapter, you will be able to dial just as you always have, and the service provider may also provide a dial tone. If your service assigns you a regular phone number, then a person can call you from his or her regular phone without using special equipment.

Some services using VoIP may only allow you to call other people using the same service, but others may allow you to call anyone who has a telephone number - including local, long distance, mobile, and international numbers. Also, while some services only work over your computer or a special VoIP phone, other services allow you to use a traditional phone through an adapter.

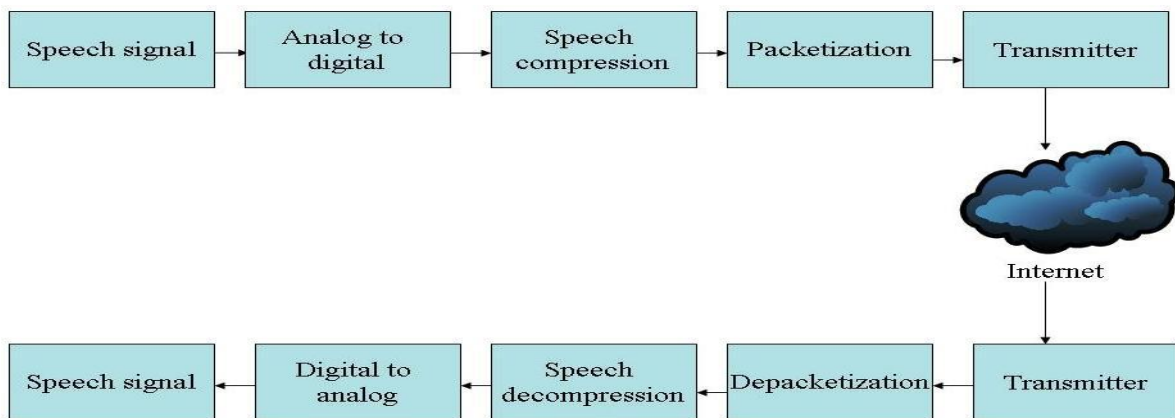


Figure 2.1 Basic VoIP Communication Systems

2.3 Socket Programming

Before you can talk to someone on the phone, you must supply a phone number to the telephone system. In a similar way, before a program can communicate with another program, it must tell the network something to identify the other program. In TCP/IP, it takes two pieces of information to identify a particular program: an *Internet address*, used by IP, and a *port number*, the additional address interpreted by the transport protocol (TCP or UDP) (Calvert K.L and

Donahoo M.J, 2008). The IP address identifies a particular device on the network and the port number uniquely identifies a program running on that device. Programs communicate with each other on the network using a communication channel (wired or wireless).

The Java language was designed (from the start) for use over the Internet. It provides many useful abstractions for implementing programs that communicate via the application programming interface (API) known as *sockets*.

There a lot of APIs available in Java that helps programmers during coding. Java API is not but a set of classes and interfaces that comes with the Java Development Kit (JDK). Java API is actually a huge collection of library routines that performs basic programming tasks such as looping, and displaying GUI form
 (http://www.roseindia.net/java/javaapi/java-api.shtml Accessed 28/01/12 10:53 PM).

Java media framework (JMF), according to Cory Janssen, is API developed to enable adding audio, video, and other time-based media to Java applications and applets. JMF extends the Java Platform Standard Edition (J2SE). It supports recording, streaming, playing, and converting between multiple media formats. Additionally, it enables developing cross-platform Java multimedia applications
 (http://www.techopedia.com/definition/23996/java-media-framework-jmf Accessed 24/2/2015 2:21 pm).

Sun Microsystems, Silicon Graphics, and Intel Corporation collaborated to release JMF 1.0 in 1997. However, the next release of JMF, that is 2.0, was developed by Sun and IBM and released in 1999. This version supported new features, including recording, streaming, and conversion among

different media formats.

3. METHODOLOGY

3.1 System Analysis and Design

The system uses client/server architecture. The architecture is a 3-tier: the client, the main server and the database server as shown in Figure 3.1. The clients consist of the various computers. The main server contains the address-to-location mappings for each user in the system. The database server contains information about users: First name, Surname, Other names, username, password, Gender, date of registration and date of modification. The database resides on the same computer that hosts the main server. This architecture enables the system to track all users and log any transaction that goes on in the system. Each user begins by establishing network links with their friends, now called network peers. Effectively, each user only accepts calls from people they trust.

Beside the 3-tier, the system can also be viewed as a three layer architecture. There are three major layers: Graphical User Interface Layer (GUI Layer), Control Layer and Data Layer as shown in Figure 3.2.

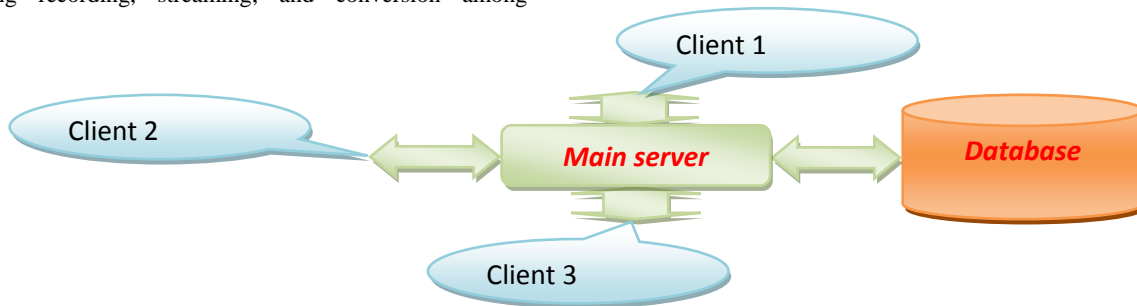


Figure 3.1 The 3-tier architecture

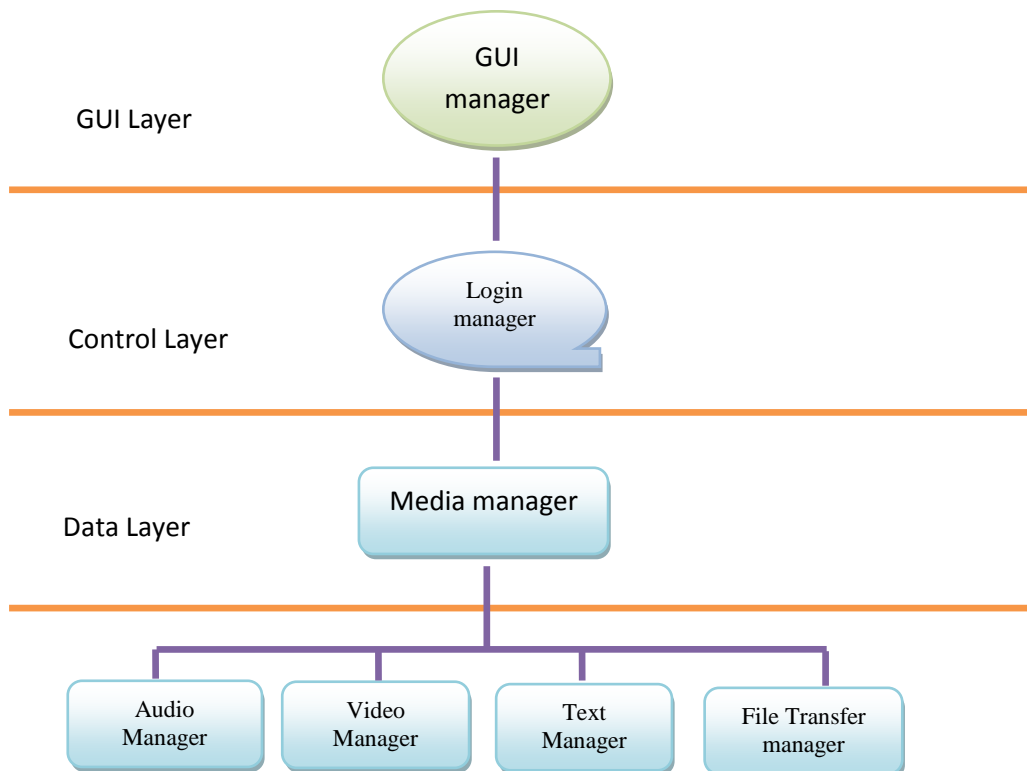


Figure 3.2 The layered architecture

3.2 Text Chat

A user can use text chat by selecting a user available online to chat with. After selecting a partner, the user type a message at the text area provided and click the *send* button. The messages of both the receiver and the sender are displayed in a text area for both parties to see. During chatting, the sender and the receiver's user names are distinguished from each other with different colours. There is no restriction on the number of characters one can send during text chat.

With text chat, the recipient's username and the message is sent to the server. The server search for the recipient's IP address and forward the message to the recipient. The code for the text chat is found at the Appendix A and its actual source code is on the attached CD.

3.3 Audio Chat

Here, JMF RTP Manager is used to send and receive audio streams. When a user selects an online client, the user then clicks on voice call button. A tone rings at the recipient's end. The recipient accepts or rejects the call. When the user accepts the call, a session is created between the two parties. When the user on one system speaks or generates any sound it is captured by the audio device (here, the microphone – referred to as *capture device* in JMF architecture) and gets *DataSource* from it. A *Player* takes as input the stream of audio and renders it to a speaker; much like a CD player reads a CD and outputs music to the speaker as shown in figure 3.3.

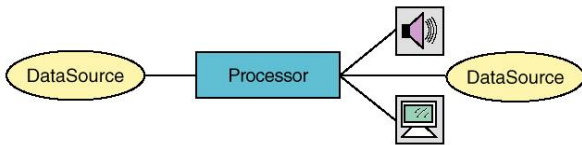


Figure 3.3 An Audio Call Process

A Player has states. A Player steps through each state until it reaches the final state. JMF defines six states in a Player:

- **Unrealized:** In this state, the Player object has been instantiated. Like a newborn baby who does not yet recognize its environment, a newly instantiated Player does not yet know anything about its media.
- **Realizing:** A Player moves from the unrealized state to the realizing state when you call the Player's *realize()* method. In the realizing state, the Player is in the process of determining its resource requirements.
- **Realized:** Transitioning from the realizing state, the Player comes into the realized state. In this state the Player knows what resources it needs and has information about the type of media it is to present. It can also provide visual components and controls, and its connections to other objects in the system are in place.
- **Prefetching:** When the *prefetch()* method is called, a Player moves from the realized state into the prefetching state. A prefetching Player is preparing to present its media. During this phase, the Player preloads its media data, obtains exclusive-use resources, and does whatever else is needed to play the media data.
- **Prefetched:** The state where the Player has finished prefetching media data – it is ready to start.

- **Started:** This state is entered when you call the *start()* method. The Player is now ready to present the media data. The order of the player states can be seen from Figure 3.4.

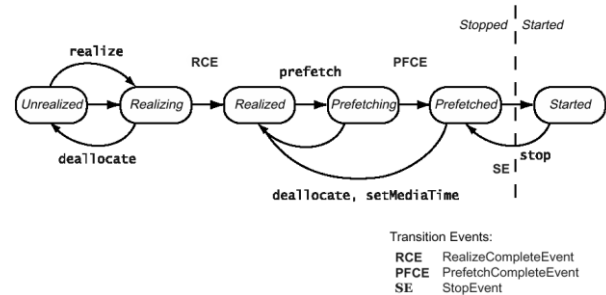


Figure 3.4 Player States [Source: <http://www.javaworld.com/jw-04-2001/jw-0406-jmf1.html?page=2> Accessed 26/06/12 9:49 pm]

Before a Player plays a media, first an instance player is created through the media manager. Next the visual component and the control-panel components of the Player are displayed. Then the player object is prepared to start. The player can be started and stopped using *start()* and *stop()* methods.

A *mediaProcessor* which is a variable processor is created which is responsible for converting our audio media from a given file format say mp3, to a format suitable for transmission over RTP protocol.

The *dataSink* variable is created which will be the output block. When a *dataSink* is created a *mediaLocator* is specified which is its actual destination. When our processed media is run through *dataSink*, it will be transmitted through *mediaTransmitter* instance to whatever location(s) specified in *mediaLocator*. In this way the desired audio stream is sent to desired destination and vice-versa (Radhika, 2012).

The codes for the audio chat could be found in Appendix B and its actual source code is on the attached CD.

3.4 Video Chat

Here too, there is a sender side and a receiver side. Each user will maintain an online list of other users present in the network and will be indicated through their username. A caller selects an online client and clicks the video call button. Same as the voice call, the recipient can accept or reject the call. When a recipient accepts a call, a session is created. This is followed by checking for supported capture devices and finding the media locator for them. A *DataSource* object for that media is then created. Next an instance processor is created through the media manager for transmitting. The processor goes through its phases –configuring, configured, unrealized, realizing, realized, prefetching, prefetched. This is followed by capturing the video tracks which will then be encoded, compressed and then passed onto RTP Manager. RTP session is then initialized by specifying the IP address and port number. At the receiver side a special thread will be listening to a special port for receiving purpose; if it detects new media streams it will create a player which goes through the stages mentioned in audio chat. The *dataSink* variable is created which is the output block. When a *dataSink* is created, a *mediaLocator* is specified which is its actual destination. When the processed media is run through *dataSink*, it will be transmitted through *mediaTransmitter* instance to whatever location(s) specified in the *mediaLocator*. In this way the desired video stream is sent to desired destination and vice-

versa (Radhika, 2012).

This process is similar to a scenario where a video camera captures video, stores it on a magnetic tape or CD, plays the media with VCR Player and the VCR player outputs the media to a screen or speaker as shown in figure 3.5.

The codes for the video chat could be found at the Appendix B and the actual source code could be found on the attached CD.

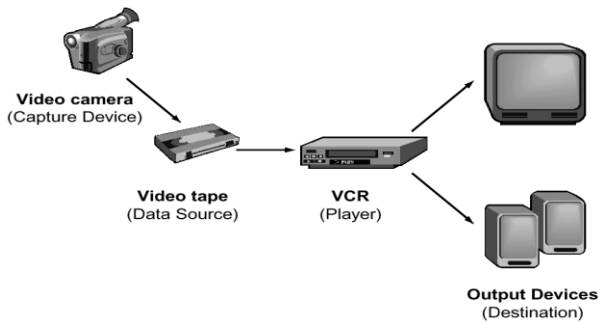


Figure 3.5 A simple Video Player Scenario

3.5 Transfer

Here too there is a sender and receiver side. A sender will select a recipient from the online list. The sender clicks the *transfer file button* and this will display *choose file* dialog. The sender browses to select the file to be transferred and click *ok* to send the file name, file path and the recipient's username to the server. The Server forwards the file to the recipient. The recipient rename file if necessary and the recipient accept to save file.

The transmission of the file started when the recipient clicks the save button. The file will not be transferred when the recipient clicks the cancel button. Clicking the cancel button sends a message to the sender.

4. IMPLEMENTATION, TESTING AND EVALUATION

4.1 The User Interface

The Graphical User Interface is composed of three components: Server-Side GUI, Client-Side GUI, and Database and Table creation GUI.

4.1.1 Server-side Graphical User Interface (GUI)

This GUI allows the administrator to configure the server side of the application. The server must start before any client can communicate to one another. The administrator can select the number of users to use the system at a time. If there is error in loading the application, the administrator can view the server log for more information.

4.1.2 Client-Side GUI

Client-side GUI has been created for the following:

- (1) Login form – for users to login into their account (if an existing users). As shown in fig.4.1
- (2) Registration form – for users to create a new account if not existing.
- (3) Services form – gives access to authenticated users to use the services provided by the Application (options like for text chat, data transfer, audio and video call are provided here).

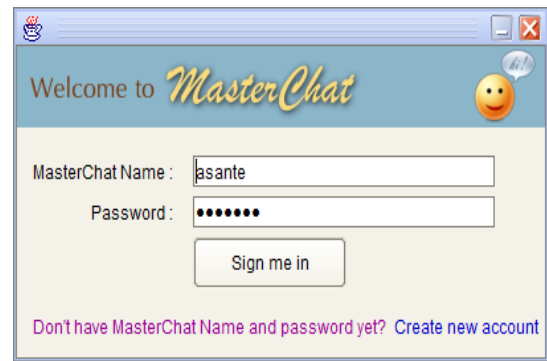


Figure 4.1 Login form

4.2 Testing

The test cases were designed in the following format:

Condition:

Input Criteria:

Input:

Result:

The **condition** describes the rule of the system. **Input criteria** specify the valid input boundaries. **Input** defines the sample input. **Result** gives the output of the sample input.

4.2.1 Test case for 'Server should run for a client when a client want to connect to the Server'

Condition: "The Server must be under running state when a client get Connected".

Input Criteria: Any client can connect to Server, which is not running.

Input: Client 1 wants to connect.

Result: Error message as shown in Figure 4.2 is displayed.

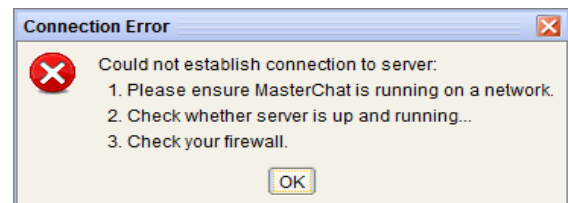


Figure 4.2 Connection Error Form

4.2.2 Test case for 'A client wants to connect twice to the Server'

Condition: "A Client should not be connected twice to the Server".

Input Criteria: Allow any client connect to the Server twice.

Input: Client 1 is already connected to the Server, and wants to connect again.

Result: Error message as shown in Figure 4.3 is displayed.

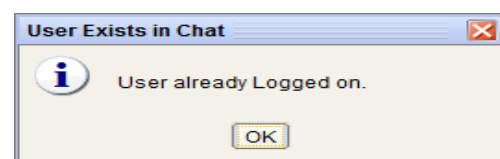


Figure 4.3 User Exists Form

4.2.3 Test case for “Entering username and password accurately”

Condition: “The username and password must be entered”.

Input Criteria: Log in new user.

Input: do not enter correct username or password or both.

Result: Error message as shown in figure 4.13 is displayed.

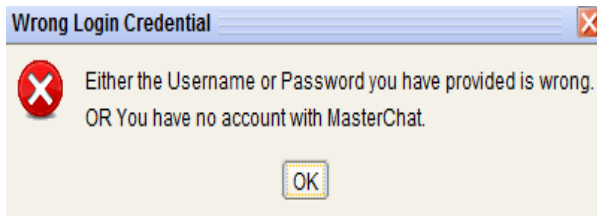


Figure 4.4 Wrong Login Credentials

4.3 Evaluation

The proposed system was evaluated based on the extent of leveraging VOIP using JMF to improve communication on campus and its cost effectiveness.

Since users do not pay for text chat and calls made, they can make any number of calls within a day. The table 4.1 below shows call and SMS charges for selected network operators in Ghana. Users of the various networks are charged for the number of minutes used but users who use the proposed system, *MasterChat*, will pay no fee.

Table 4.1: Call charges for the various network operators in Ghana (as at November, 2014)

OPERATOR	Cost per minute call (on-net)	Cost per minute Call (off-net)	Cost per sms (on-net)	Cost per sms (off-net)
Tigo	3p	8p	4.1p	4.7p
MTN	9p	10p	4p	5p
GLO	6p	6p	4p	4p
Vodafone	9p	14p	4p	4.2p
Airtel	6.5p	8p	2p	5p
MasterChat	0	-	0	-

An alternate approach used by various institutions to communicate internally is the “inter-com”. According to home wyse, in 2014, the average cost to install an Intercom

System ranges from \$1296.73 to \$1893.67 per each intercom system

(http://www.homewyse.com/services/cost_to_install_intercom_system.html. Accessed 26/1/15 8:52 am). Since a user of this application has its own network with, the installation of the proposed system, *MasterChat*, will cost no fee.

5. SUMMARY, CONCLUSION, AND RECOMMENDATION

5.1 Summary

The purpose of this design research was to design a system that will allow people to communicate over a data network. Users using the proposed system will be able to send text, make voice and video call, and transfer file from one user to the other. Though, there are other existing systems that can perform the above mentioned tasks, those systems are internet-based which makes it impossible to use without internet connection. The proposed system runs on intranet and this makes it ideal application for organisations who want to communicate internally.

Java Programming language was used in the design of the application. JMF, a Java Media API, was used to abstract the internal implementation of media streaming. Client application for mobile devices was designed using Android SDK and eclipse IDE. Client/Server architecture was used in the design of the application. With this 3-tier architecture, users’ information is stored in the database server. Address – to- location mapping is done by the main server. Many clients can connect to the main server application using required IP address and port number. The application could be used on Android mobile phones and computers with Windows operating system.

5.2 Conclusion

The research aimed at the design of a system that allows users to send text, make voice call, make video call, and transfer file over a Local Area Network. Client software, server software and Database were created and integrated. Java Media Framework was the API used to handle media streaming. The system designed was evaluated to be cost effective, faster, reliable and always available as compared to using inter-com system or cellular phone.

5.3 Recommendations

The targeted task was to design a system that allow users to send text, make voice call, make video call, and transfer file over a Local Area Network. The system designed has a number of advantages over the existing systems. These advantages include the following:

The system was evaluated to be cost effective as compared with the existing systems such as inter-com system and Cellular Phones.

Technical support could easily be provided locally unlike Yahoo! Messenger and Google Talk that the user needs to communicate with a bot.

The system is faster and always available. Unlike the internet-based systems that have intermittent connections, this system is purely intranet- based and will always be on when the intranet is available. Here, no bandwidth is needed. There is low traffic, high transmitted packets, low drop packets, high speed of transmission, and there is low packet collision.

The users of the proposed system have at their disposal both the client and Server applications. With systems such as Yahoo messenger, Google Talk, Facebook and Skype, users are provided with only the client application. Therefore if there is any problem with the Server application, users need to wait for it to be fixed from the developers end. With the *MasterChat*, the system could be easily monitored and troubleshoot is done locally.

It is therefore recommended that, within a local area network, Master Chat should be used instead of inter-com system, cellular phones or internet-based system such as Skype, Yahoo! Messenger and Google Talk.

6. REFERENCES

- [1] Calvert, K. L. and Donahoo, J. M. (2008). TCP/IP Sockets in Java: Practical Guide for Programmers, Second Edition, Morgan Kaufmann Publishers, USA.
- [2] Comer, D. (2006). Internetworking with TCP/IP: Principles, Protocols, and Architecture, Volume 1, Prentice Hall, Upper Saddle River.
- [3] DeveloperWorks, Java sockets
101:ibm.com/developerWorks
- [4] Forouzan, B. A. (2010). TCP/IP Protocol Suite, Fourth Edition, McGraw-Hill Companies, Inc., New York, USA.
- [5] Gomillion, D. and Dempster, B. (2005). Building Telephony Systems with Asterisk, Packet Publishing Ltd. Birmingham, UK.
- [6] <http://compnetworking.about.com/od/networkprotocolsip/g/udp-user-datagram-protocol.htm>. Accessed 23/01/12 11:22 PM
- [7] [http:// docs.oracle.com/javase/tutorial/sound/index.html](http://docs.oracle.com/javase/tutorial/sound/index.html)
Accessed 20/05/11 8:30 am
- [8] http://en.wikipedia.org/wiki/Google_Talk. Accessed 9/7/12 8:43 pm
- [9] http://en.wikipedia.org/wiki/User_Datagram_Protocol. Accessed 23/01/12 11:17 pm
- [10] http://grack.com/downloads/school/enel619.10/report/java_media_framework.html. Accessed 5/7/2012 4:47 pm
- [11] http://help.yahoo.com/tutorials/ms8/mess/im_setup1.htm
1 Accessed 9/7/12 8:40 pm
- [12] <http://voip.about.com/od/voipbasics/a/ReasonsForVoIP.htm>. Accessed 20/05/11 8:20 am
- [13] http://www.homewyse.com/services/cost_to_install_intercom_system.html. Accessed 2/11/12 10:25 am.
- [14] <http://www.javaworld.com/jw-04-2001/jw-0406-jmf1.html?page=2>. Accessed 26/06/12 9:49 pm
- [15] <http://www.roseindia.net/java/javaapi/java-api.shtml>. Accessed 28/01/12 10:53 PM
- [16] <http://www.techopedia.com/definition/23996/java-media-framework-jmf>. Accessed 24/2/2015 1:21 pm
- [17] Kevin, W. (2005). Voice over IP First-Step, Cisco Press.
- [18] Knútur, B. O. (2011). Risk analysis on VoIP systems, Master's thesis.
- [19] Pawel, L. (2007). VoIP Security in Public Networks, master's thesis.
- [20] Perea, R. M. (2008). Internet Multimedia Communications Using SIP: A Modern Approach Including Java Practice, Elsevier's Science & Technology Rights Department in Oxford, UK.
- [21] Radhika, P.(2012) Multimedia conferencing using java media framework, International Conference on Computing and Control Engineering (ICCCE 2012), 12 & 13 April, 2012
www.iccce.co.in/Papers/ICCCECS696.pdf
- [22] Salman, A. B. and Schulzrinne, H. G. (2004). An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol, Columbia University, New York NY 10027
- [23] Tiantioukas, N. (2007). Effects of the Wireless Channel, Signal Compression and Network Architecture on Speech Quality in VoIP Networks, Master's thesis, Naval Postgraduate School, Monterey, California, USA.