
DIGITAL CONVERGENCE TECHNOLOGY IN AN AUDIO CHAT SYSTEM

Japheth B. R
Department of Mathematics/Computer Science,
Niger Delta University, Yenagoa, Nigeria
jbunakiye@yahoo.com

ABSTRACT

This paper presents a chat-system as a modern communication system taking advantage of digital convergence that addresses the problems of delays, time and materials wastage, and structures that characterize meetings, personal conversations, group conversation, and advert placements in our society. Communication can be realized through any activity and through specialized forms, such as audio conferencing. The specific task defined here is modern communication enhanced and powered by digital convergence aimed at making easier the communication of multiple users of a chat-system, in a local area network. It allows for simple scripting for devices recognition interactions at the ports and implementation in a Client/Server program platform through multithreading but did not cover bandwidth management and how the network synchronizes various media platforms to create new experiences.

Keywords: *Chat-System, communication, digital convergence, Client/Server, platform, multithreading*

INTRODUCTION

Chat-system is a very popular computer program for online conferencing. A chat-system can also be considered as Instant Messaging system, while the chat-system consists of [8] client program and server program. The client program has mainly two types of user interfaces, chat-room interface and Instant Messenger interfaces. The Instant Messenger allows its users to view a listing of people with whom they frequently communicate, it also determines if the people are currently online and available, and send or receive messages instantly. In the beginning instant messaging or chat-system program gained popularity with home users but soon the technology is quite suitable for corporate and professional communications. The most common difference between a chat-room and an Instant Messaging is that messenger is more private than chat-room. A chat-room is known as a public chatting place where identities of all clients are shown in a list without their permission. And in this paper the use of both interfaces were considered. The advantage of digital convergence [1] in modern communication covered in this paper handles the new efficiencies produced by the synergy of delayed communication technologies such as telegram, and postal services that now share resources and interact with each other that results in a converged communication medium. This is done to enable a thorough review of convergence technology and the trends among partners using computers. This led to the concept of protocols and the rules of communication which interlinked computers are to follow [2, 3].

THE PARADIGM SHIFT

These paradigm shifts are ongoing in the media, and often occur from time to time as the technology to create better devices evolves. Broadcasting is increasingly being replaced by the Internet, enabling consumers all over the world the freedom to access their preferred media content more easily and at a more available rate than ever before. With this knowledge of modern communication; taking advantage of digital convergence, insecurity of private information [6] could be reduced to a barest minimum because digitization and the widespread sharing of digital files over a computer network, does raise significant concerns about the privacy of the individual who is receiving and sending digitized communication. Delays and wastages are negligible because in the former the chat system can make possible meetings without moving from one place to the other, and in the later reference to meetings can be handled easily in the future. This system creates favorable work atmosphere, which makes a business communication more interactive. Where you will have live chat for your business clients and team and this will help to utilize time for some other important tasks instead of organizing hundreds of files, folders and even mails [3, 4 and 5].

THE CONVERGENCE LOGICAL ARCHITECTURE

The logical architecture [9] is a design that depicts the logical building blocks of the components and the infrastructure services needed to support them and the identified usage patterns. Figure 1 shows the high-level logical convergence architecture of the chat-system. The chat system actually is a modern communication suite so the advantage of digital convergence does not change the existing two-tiered client/server architecture exception of the convergence server "sitting" between the communications suite servers and the clients on the network.

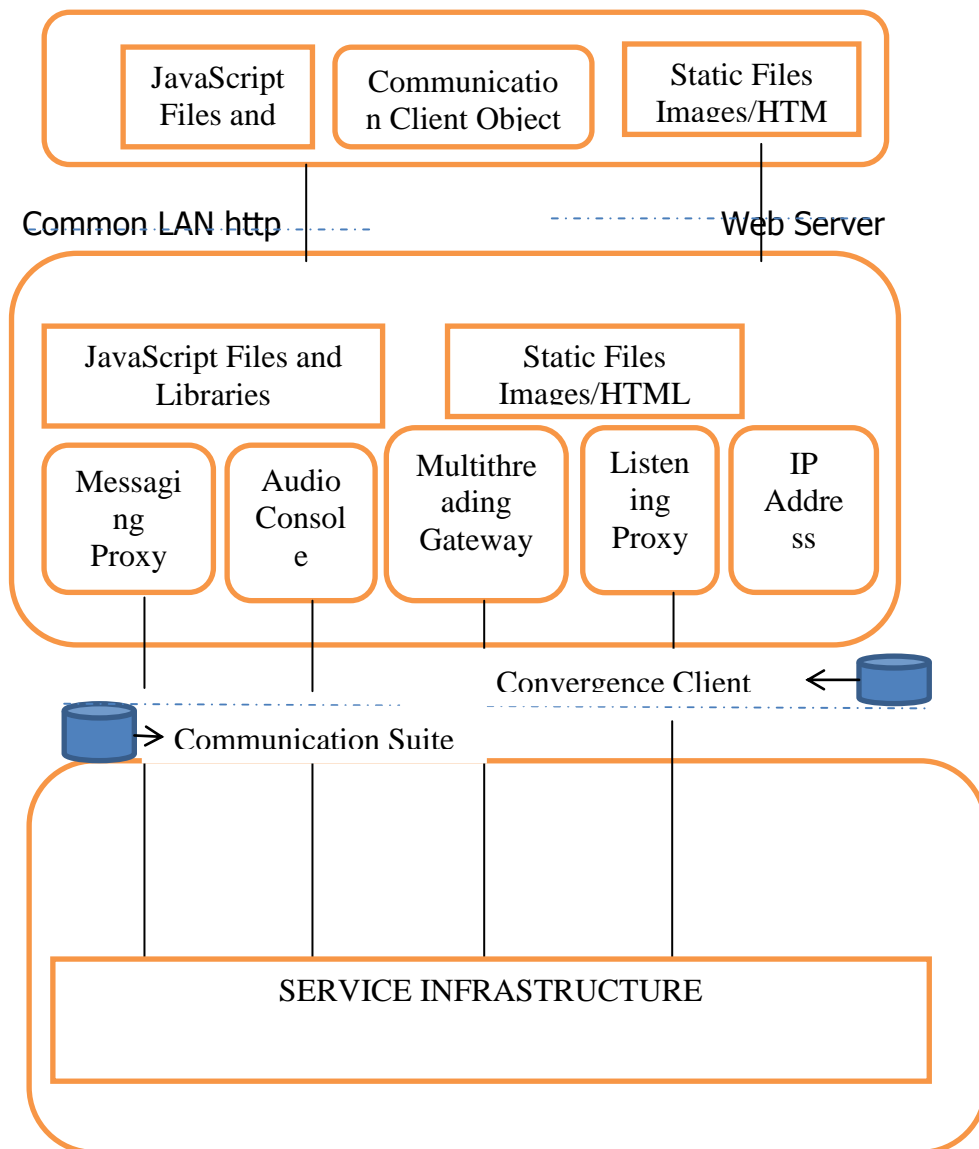


Figure 1: An Overview of the Chart-System Architecture

The architecture of the multiple user chat system is designed to enable transactions managed by software to ensure that all the data associated with the transaction is processed as a single unit. This generally requires a distribution that uses a multithreaded client/server approach. A distributed data and application client/server system is a solution in which (1) the data and data manipulation layers are placed on their own server(s), (2) the application logic is placed on its own server, and (3) only the presentation logic and presentation are placed on the clients.. The entire sum up is the connection of multiple client applications to the server application. The server then echoes messages back to the client applications, making it possible for the chat-system to make audio conferencing [7, 9].

REQUIREMENTS MODEL

There is the Non Functional Requirement Where;

1. Create convenience for the user (Administrator) to use the system
2. Secure processed information in the computer.

AND The Functional Requirements Where;

1. Enable user login to the system
2. Allow users to make voice calls with the system.
3. Enable user to chat with available clients

Table 1 shows the users of the system and their descriptions

USER	DESCRIPTION
ADMINISTRATOR	He is the manager of the system and its database.
OTHER USER	These are Lecturers and students that use or make use of the chat system in one way or the other

Table 1: Categories of Users

There are two categories of users of this system, these are;

- a. The Administrator
- b. Other Users(Students/Lecturers)

The system is to meet the generally known user requirements of a Chat system amongst which are;

- a. Be able to register users.
- b. Be able to log in.
- c. Be able to chat with every user logged in.
- d. Know user by user name.
- e. Easy use of audio conferencing.

PERFORMANCE

The system has a great performance because it uses multithreading to handle the send and receive actions done by each users connected to the server. This is done by threads, creating different processes for each client connected so the clients don't have to wait in a queue. The system is developed using NetBeans IDE for the software development and it's been tested using Windows XP operating system, command prompt would be used to run the server applications for the system. The system was developed in segments, firstly the text

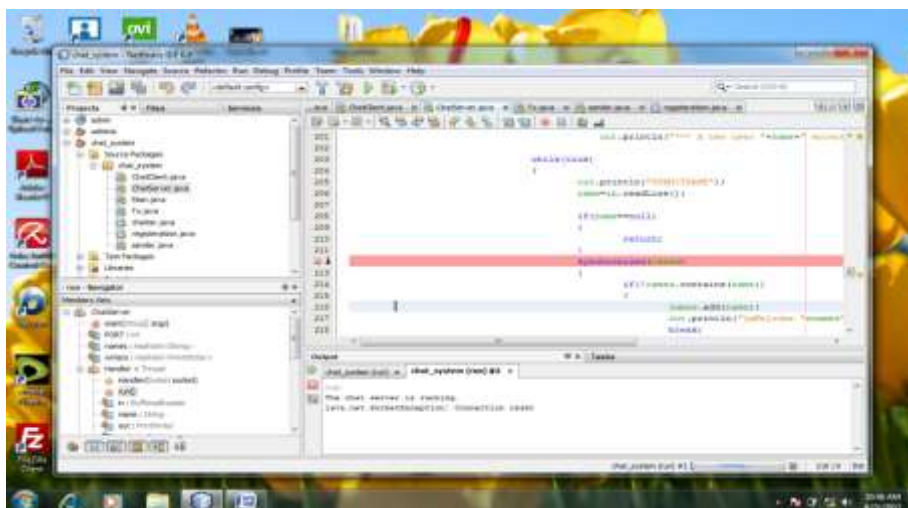
interaction part of the chat-system was developed, afterwards the audio part was and later both were integrated to give the chat-system working efficiently [9].

ALGORITHM: TESTING

```
Start the chat server
Print "the server is running"
While the server awaits incoming calls from clients
Start the chat client
Prompt user to enter IP address
Input IP address
If the IP address is correct
    Connect the client to server
    Prompt user to enter username
    If username is unique
        Set the client to be seen with username
        Input message in text field
        Send message to server
        Server broadcast message from client to all connected clients
        Print the message
        Print messages from all other clients
    Else
        Prompt user to enter username until it's unique
        Input username
Else if the IP address is not correct
    Close connection between client and server
End
```

IMPLEMENTATION OF THE SYSTEM

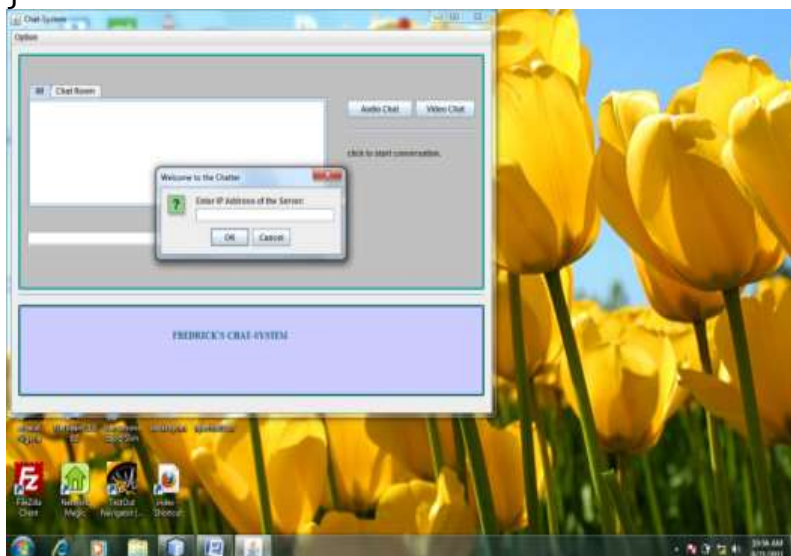
The system implementation was done in phases because it has individual classes that represent the different components of the system thereafter all the components were brought together for an integrated testing. Here below are some of the snapshots as results of the unit testing that was done and the implementation codes



This code snippet is responsible for the server listening and waiting for connection to be made.

```

ServerSocket listener=new ServerSocket(PORT);
try
{
    while(true)
    {
        new Handler(listener.accept()).start();
    }
    finally
    {
        listener.close();
    }
}
    
```



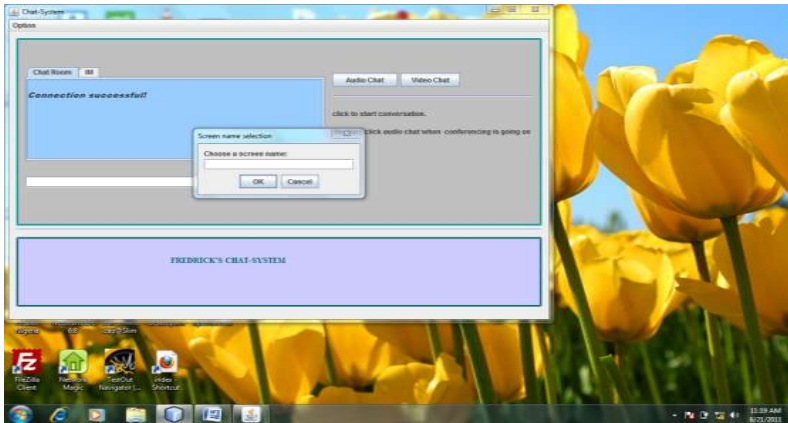
This prompts the user to enter the IP address which the server is listening on.

```

private String getServerAddress()
{
    
```

```

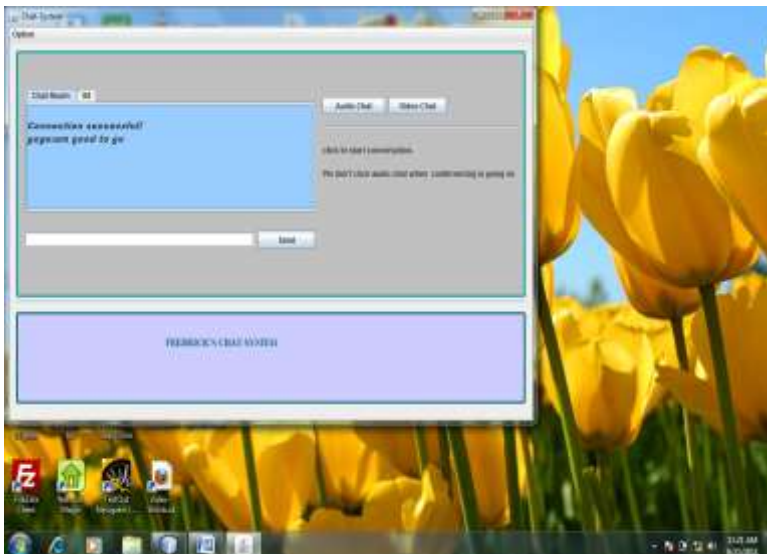
return
    JOptionPane.showInputDialog(this,"Enter IP Address of the Server:","Welcome
to the Chatter",JOptionPane.QUESTION_MESSAGE);
}
    
```



Prompts the user for a screen name so other users will know who is chatting with them.

```

private String getNames()
{
    return
        JOptionPane.showInputDialog(this,"Choose a screen name:","Screen name
selection",JOptionPane.PLAIN_MESSAGE);
}
    
```



These codes are responsible for sending messages to the server.

```

jTextField1.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        out.println(jTextField1.getText());
    }
}
    
```

```
        jTextField1.setText("");
    }
});
jButton1.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        out.println(jTextField1.getText());
        jTextField1.setText("");
    }
});
```

Software Requirements

The development of the system requires the following;

1. Java Development Kit (JDK 1.6)
2. Java Runtime Environment (JRE 1.6)
3. Java NetBeans 6.0 for the design of the system layout.
4. A notepad and Visual studio Editors where used as source code editors
5. Microsoft SQL (MSSQL) Database Management System for database management.
6. Microsoft Dos prompt for compiling and running codes
7. Microsoft Windows XP as Operating System
8. The minimum requirements for a system to run this application is 128MB of RAM

HARDWARE REQUIREMENTS

For a system to run this application it should have the following minimum hardware requirements;

1. 5MB of Hard disk
2. A CD-ROM of at least 700MB
3. Uninterrupted Power Supply (UPS)

CONCLUSION

In this paper, the client/server multithreading program platform is identified and used for the implementation of a chat system most suitable for audio conferencing. It is a modern communication system, taking advantage of digital convergence. Given the high degree of time, noise and resource waste negligibility that characterizes the chat system, it seems that, it is a good package for groups and organizations business communications

REFERENCES

[1] Arora, A. (1993) Closure and convergence: A foundation of fault-tolerant Computing, IEEE Transactions Software Engineering, Special Issue on Software reality, Vol. 19, No 3, 1015-1027

[2] Gouda M.G. (1995) The triumph and Tribulation of system stabilization, Invited paper, Proceedings of the 9th International Workshop on Distributed Algorithms (WDAG'95), to be published by Springer- Veriag 1995.

[3] Multari, N. (1991) Stabilizing Communication Protocols, IEEE Transactions on Computers, Special Issue on Protocol Engineering, Vol. 40, No.4, pp 448-458

See also:

[4] Open MP C and C++ Application Program Interface (version 2.0), Open MP Architecture Review Board, March 2002.

[5] "Multithreading: Taking Advantage of Intel Architecture-based Multiprocessor Workstations," Intel White Paper, 1999.

[6] "Threading Methodology: Principles and Practices," Intel Technical Report, 2002.

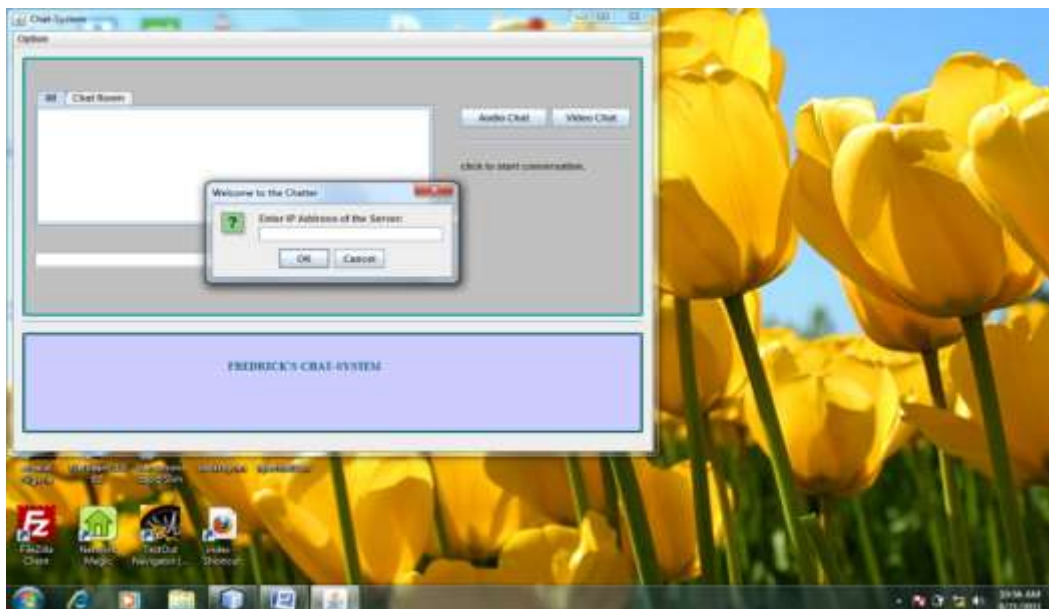
[7] M. Ben-Ari, Principles of Concurrent Programming, Prentice-Hall International, 1982.

[8] Johnson M. Hart, Win32 System Programming (2nd Edition), Addison-Wesley, 2001.

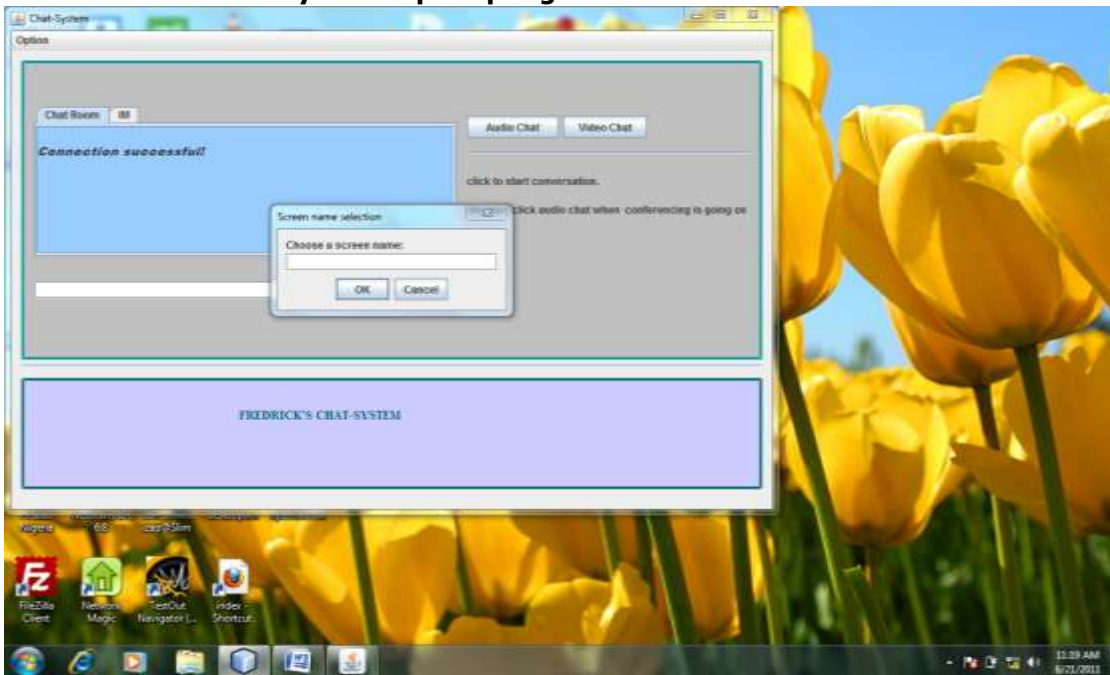
[9] Deitel & Deitel (2007), JAVA "How to program 8th Edition". Pearson Education.

[10] IT edge journal February Edition 2003. Donald's Publications, Abuja, Nigeria

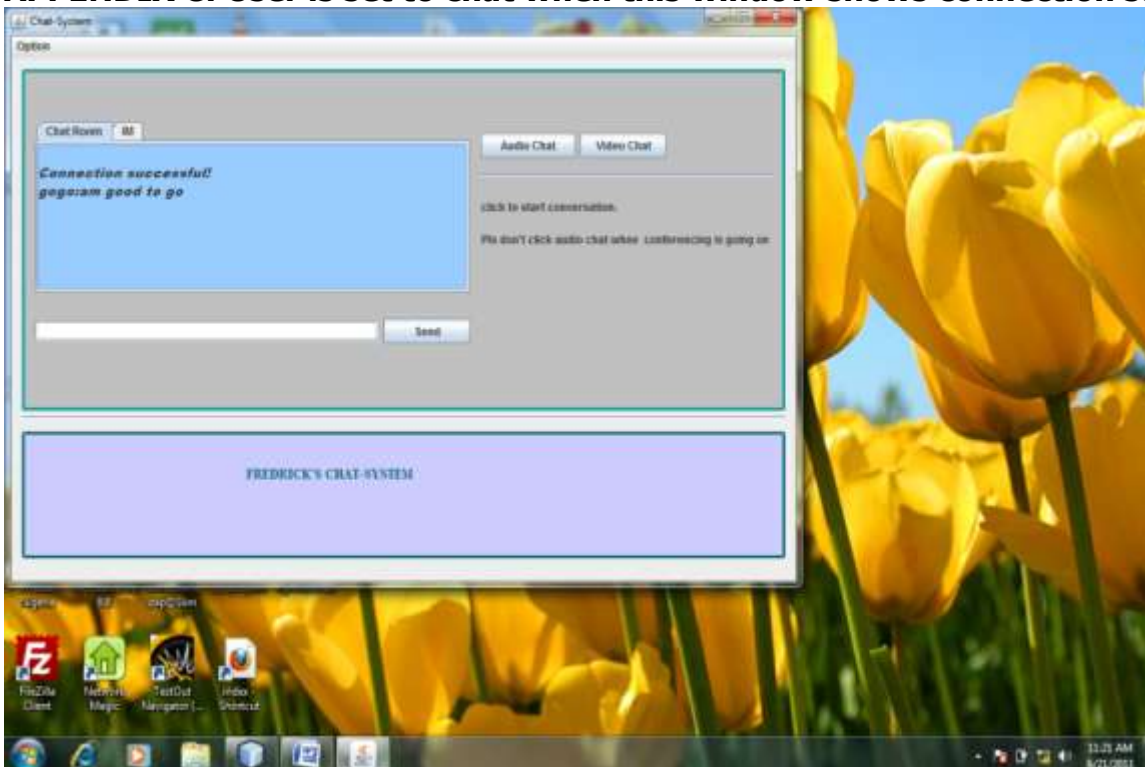
APPENDIX A: The system is prompting the user to enter the IP address which the server listens.



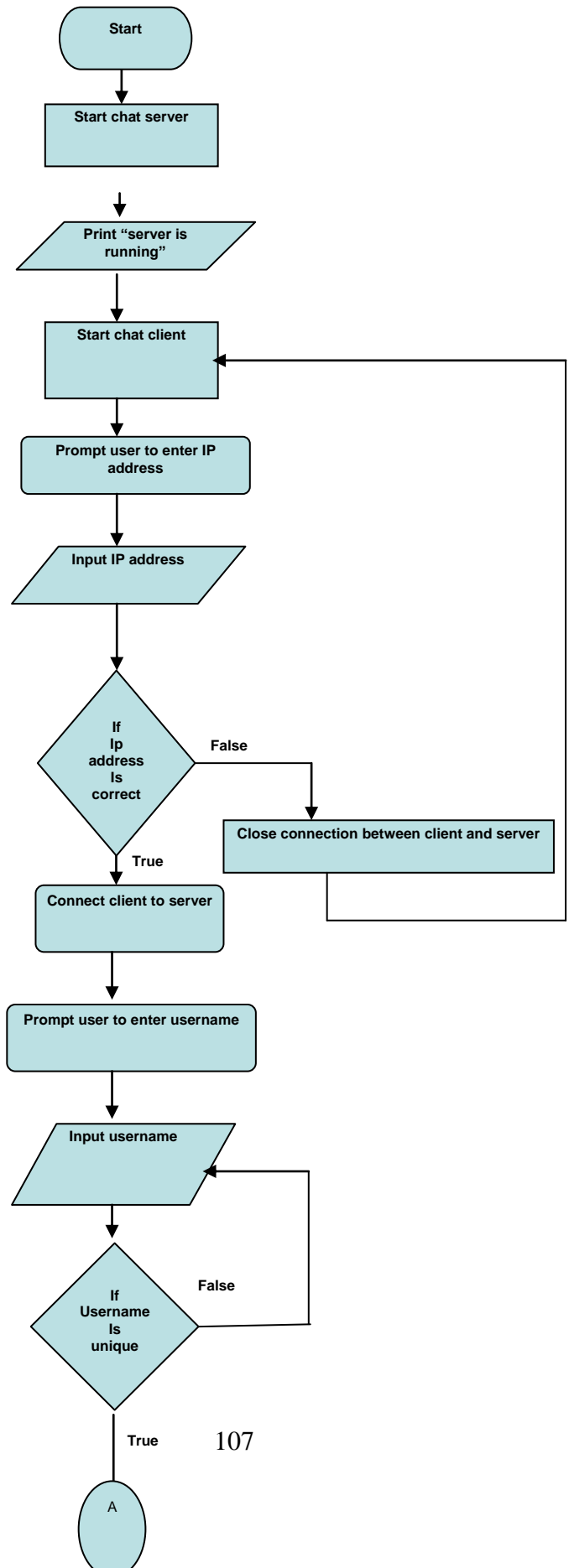
APPENDIX B: The system is prompting the user to enter a screen which the server listens.



APPENDIX C: User is set to chat when this window shows connection successful.



APPENDIX D: Flow Chart for Top down Chat-System Structure



APPENDIX E: Flow Chart for Business Requirements Chat-System Structure

