

Developing and Implementation of Distributed Chat Applications using WPF and WCF

Motaz Daadoo

*Department of Computer Systems Engineering
Palestine Technical University- Kadoorie (PTUK), Tulkarm, Palestine
P.O. Box 7 Tulkarm City, Palestine,
E-mail: m.daadoo@ptuk.edu.ps
Tel: +972-9-2688175, Fax: +972-2-2677922*

Abstract

This paper deals with the issues involved in distributed application. The development of distributed chat applications involves several tools and technologies. In particular, the researcher describes an application that provides multi chatting between computers on the Internet. A server realizes the connection. It is important to mention that the addresses of clients are not known. Server listen and be ready for any request both devices are allowed to send messages to the other party without the knowledge of Internet Protocol (IP) each and there is a special whisper who says that it allows users to send anonymous letters, and receive responses, known as the "Whispers", which sends the message to a specific person. The technology used in this application in C# Windows Communication Foundation (WCF). In this paper is presented a solution that involves the use of Windows Presentation Foundation (WPF) and WCF frameworks to develop distributed chat applications.

Keywords: Chat Application, .NET, Service, WCF, WPF.

1. Introduction

My application is about communication between clients through server. The server is always ready to listen to any request, and waits the clients to connect. The clients connect to the server and start communicating to each other clients, but no one knows the other client's IP address, only the server IP address they knows. I will describe the application and how it works through the paper, and the tools, API's which are used and why.

.NET Framework as it is shown in the Figure 1 is a software framework developed by Microsoft that runs primarily on Microsoft Windows. It includes a large class library known as Framework Class Library (FCL) and provides language interoperability (each language can use code written in other languages) across several programming languages. Programs written for .NET Framework execute in a software environment (as contrasted to hardware environment) known as Common Language Runtime (CLR), an application virtual machine that provides services such as security, memory management, and exception handling. (As such, computer code written using .NET Framework is called "managed code"). FCL and CLR together constitute .NET Framework.

In this application the researcher used .NET because it has different API's which are important to use, such as WCF and WPF. WCF is a framework for building service-oriented applications. Using WCF, you can send data as asynchronous messages from one service endpoint to another. A service endpoint can be part of a continuously available service hosted by Internet Information Services (IIS)

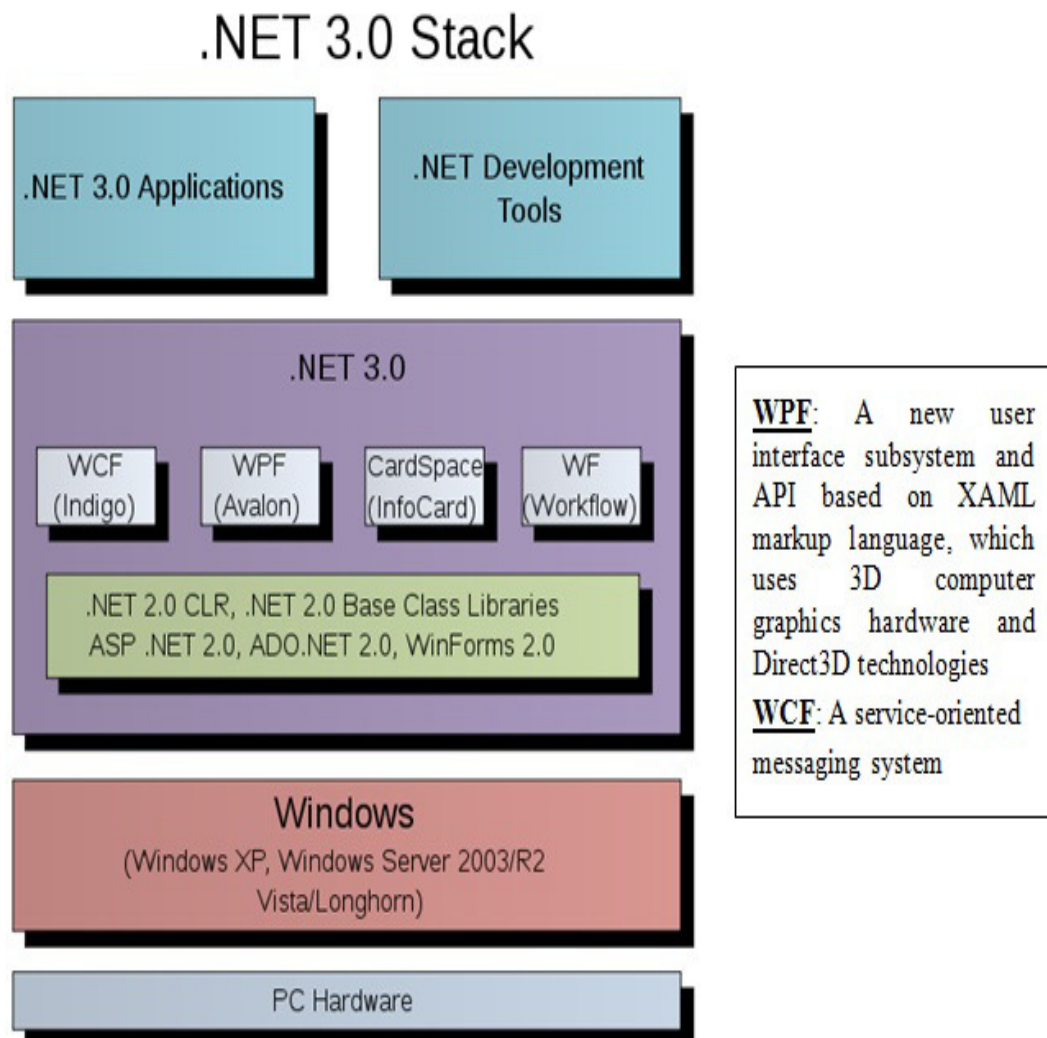
or it can be a service hosted in an application. An endpoint can be a client of a service that requests data from a service endpoint. The messages can be as simple as a single character or word sent as XML, or as complex as a stream of binary data [Rouse, 2007].

The WCF, previously known as "Indigo", is a runtime and a set of APIs in the .NET Framework for building connected, service-oriented applications. While creating such applications was possible prior to the existence of WCF, WCF makes the development of endpoints easier than ever. In summary, WCF is designed to offer a manageable approach to creating Web services and Web service clients.

WPF is a graphical subsystem for rendering user interfaces in Windows-based applications. WPF, previously known as "Avalon", was initially released as part of .NET Framework 3.0. Rather than relying on the older GDI subsystem, WPF uses DirectX. WPF attempts to provide a consistent programming model for building applications and separates the user interface from business logic. It resembles similar XML-oriented object models, such as those implemented in XUL and SVG.

WPF employs XAML, an XML-based language, to define and link various interface elements. WPF applications can be deployed as standalone desktop programs or hosted as an embedded object in a website. WPF aims to unify a number of common user interface elements, such as 2D/3D rendering, fixed and adaptive documents, typography, vector graphics, runtime animation, and pre-rendered media. These elements can then be linked and manipulated based on various events, user interactions, and data bindings [Mancas, C., & Mocanu, M., 2013].

Figure 1: The architecture of .NET



2. WCF

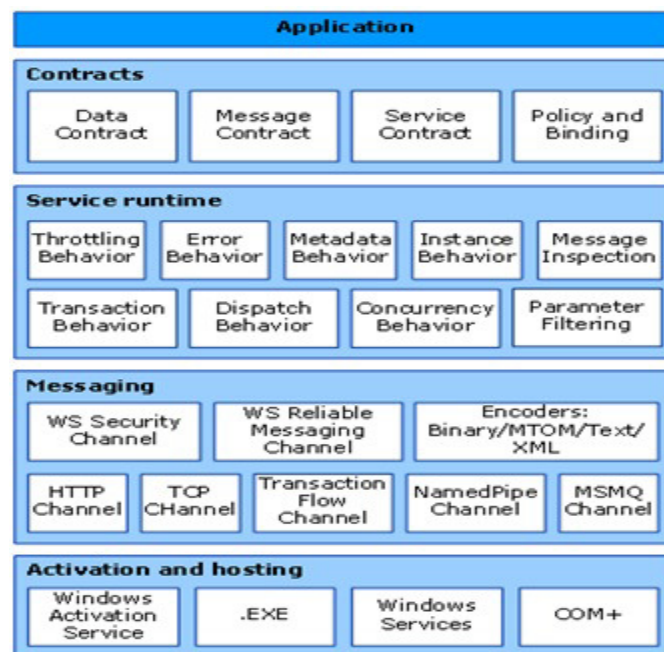
WCF which stands for Windows Communication Foundation is a collection of APIs used for building Service Oriented Architecture (SOA). A WCF host provides the services from a network and can serve many WCF clients. WCF clients on the other hand can also consume services from many WCF hosts at the same time. WCF is platform independent, which means it is possible to host or consume Services from any kind of platform [Purwati, Y., & Utomo, F. S., 2014].

2.1 WCF at an Architectural Level

WCF is a tool, often used to implement and deploy a SOA. It is designed using service-oriented architecture principles to support distributed computing where services have remote consumers as shown in Figure 2 Clients can consume multiple services; services can be consumed by multiple clients.

Services are loosely coupled to each other. Services typically have a Web Services Description Language (WSDLinterface) that any WCF client can use to consume the service, regardless of which platform the service is hosted on. WCF implements many advanced Web Services (WS) standards such as WS-Addressing, WS-Reliablemessaging and WS-Security. With the release of .NET Framework 4.0, WCF also provides RSS Syndication Services, WS-Discovery, routing and better support for REST services, as it is shown in the Figure 2.

Figure 2: Architecture of a WCF application



2.2 Endpoint

A WCF client connects to WCF service via an endpoint. Each service exposes its contract via one or more endpoints. An endpoint has an address (which is a URL specifying where the endpoint can be accessed) and binding properties that specify how the data will be transferred.

The mnemonic "ABC" can be used to remember address/binding/contract. Binding specifies what communication protocols are used to access the service, whether security mechanisms are to be used, and the like. WCF includes predefined bindings for most common communication protocols such as SOAP over HTTP, SOAP over TCP, and SOAP over Message Queues, etc.

Interaction between WCF endpoint and client is done using a SOAP envelope. SOAP envelopes are in simple XML form, which makes WCF platform-independent. When a client wants to access the

service via an endpoint, it not only needs to know the contract, but it also has to adhere to the binding specified by the endpoint. Thus, both client and server must have compatible endpoints.

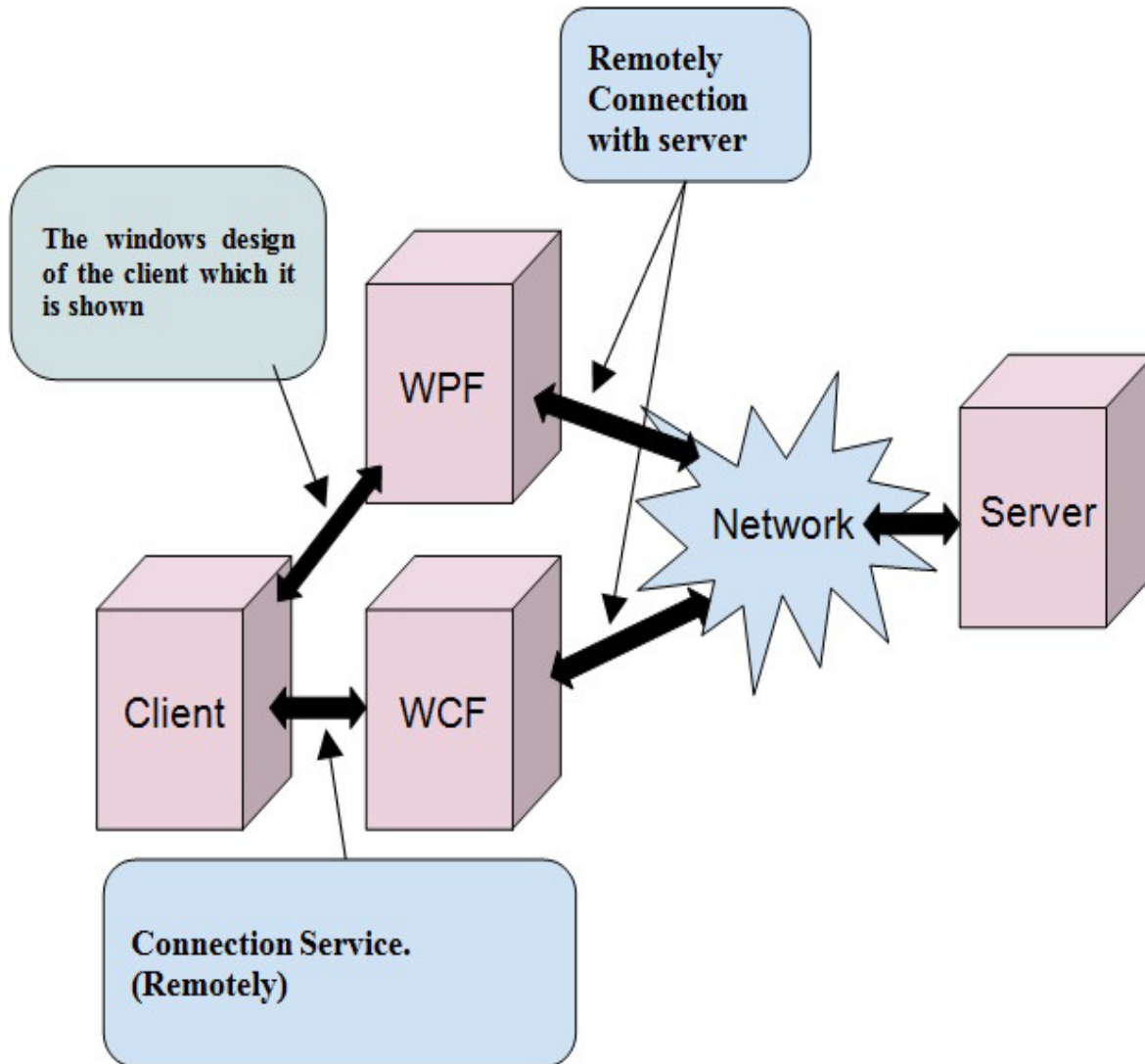
2.3 Features of WCF

WCF includes the following set of features:

- **Service Orientation:** One consequence of using WS standards is that WCF enables you to create service oriented applications. SOA is the reliance on Web services to send and receive data.
- **Interoperability:** WCF implements modern industry standards for Web service interoperability.
- **Multiple Message Patterns:** Messages are exchanged in one of several patterns. The most common pattern is the request/reply pattern, where one endpoint requests data from a second endpoint. The second endpoint replies.
- **Service Metadata:** WCF supports publishing service metadata using formats specified in industry standards such as WSDL, XML Schema and WS-Policy. This metadata can be used to automatically generate and configure clients for accessing WCF services. Metadata can be published over HTTP and HTTPS or using the Web Service Metadata Exchange standard.
- **Data Contracts:** Because WCF is built using the .NET Framework, it also includes code-friendly methods of supplying the contracts you want to enforce. One of the universal types of contracts is the data contract.
- **Security:** Messages can be encrypted to protect privacy and you can require users to authenticate themselves before being allowed to receive messages. Security can be implemented using well-known standards such as SSL or WS-Secure Conversation.
- **Multiple Transports and Encodings:** Messages can be sent on any of several built-in transport protocols and encodings.
- **Reliable and Queued Messages:** WCF supports reliable message exchange using reliable sessions implemented over WS-Reliable Messaging and using MSMQ.
- **Durable Messages:** A durable message is one that is never lost due to a disruption in the communication. The messages in a durable message pattern are always saved to a database.
- **Transactions:** WCF also supports transactions using one of three transaction models: WS-AtomicTransactions, the APIs in the System, transactions namespace, and Microsoft Distributed Transaction Coordinator.
- **AJAX and REST Support:** REST is an example of an evolving Web 2.0 technology. WCF can be configured to process "plain" XML data that is not wrapped in a SOAP envelope.
- **Extensibility:** The WCF architecture has a number of extensibility points. If extra capability is required, there are a number of entry points that allow you to customize the behavior of a service.

2.4 Behaviors

Behaviors implement the `IServiceBehavior` interface for service extensions, the `IEndpointBehavior` for endpoints, the `IContractBehavior` interface for service contracts, or the `IOperationBehavior` for operations. Service behaviors are used for message processing across a service, rather than processing that would be specific to a single operation, as it is shown in the Figure 3.

Figure 3: A WCF methods on a client

2.5 Interoperability

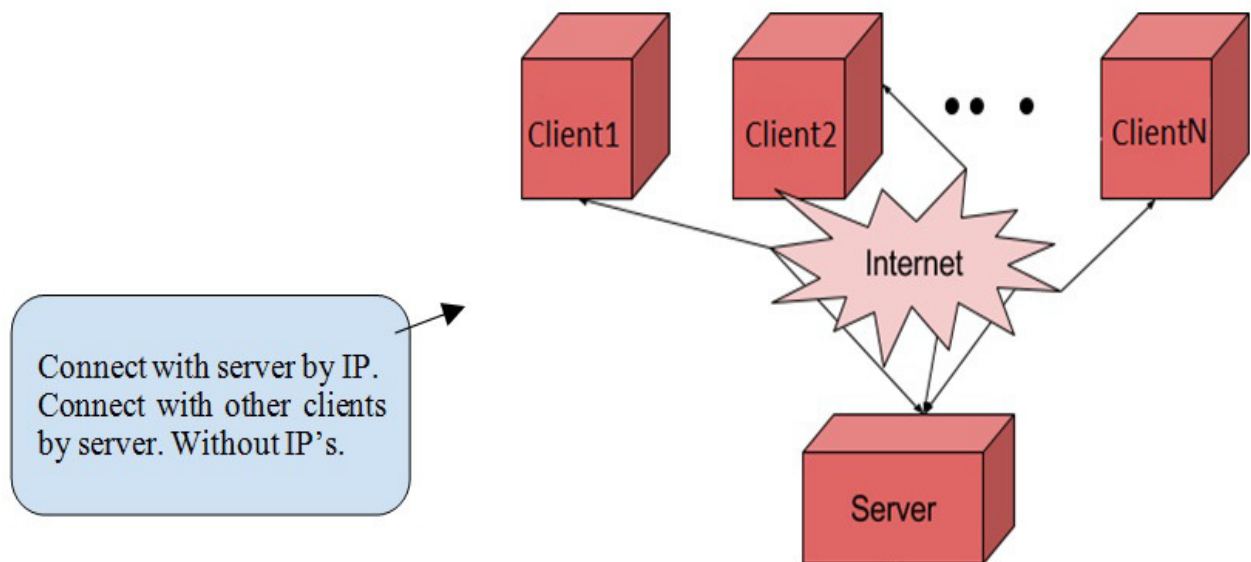
WCF supports interoperability with WCF applications running on the same Windows machine or WCF running on a different Windows machines or standard Web services built on platforms such as Java running on Windows or other operating systems.

In addition to SOAP, WCF 4 supports non-SOAP XML, RSS, JSON, and binary formats for external communication via HTTP or HTTPS.

3. The Application Description

The application related in this paper contains two parts: server application, client application, as it is shown in the Figure 4. The server runs on a computer as an executable application (.exe), the client runs on other computers that can have an unknown address. We can connect many computers from anywhere in Internet that they have an unknown IP address.

The client is connected to server (which has a known address), then the other Clients can “see” each other. Clients has the server IP address by default, we add it in the code so we don’t need to add it dynamic, so the server and the clients remain connected with each other. The server participates to make the connection between many clients, it doesn’t participate to transfer.

Figure 4: The block schema for the application

The server knows all the connections that are done. It has a list in which it retains the client's name.

We made the following suppositions:

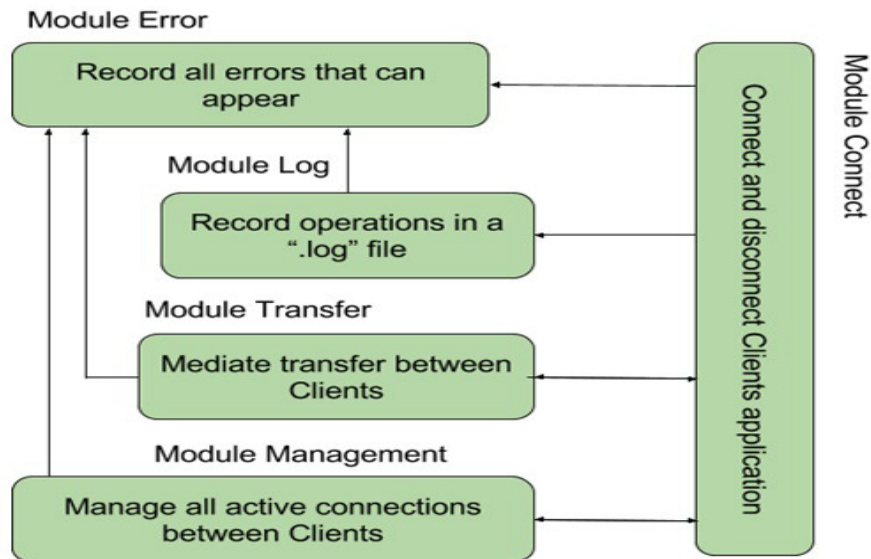
- When a client initiates a connection, the server verifies the existence of the connection in the list and accepts that connection or not.
- The communication between many clients is made through server (only if neither of them have a valid IP address in Internet).

3.1 Server Application

The server application makes the connection between clients applications. The following diagram in Figure 5 represents the interaction between the Server's modules.

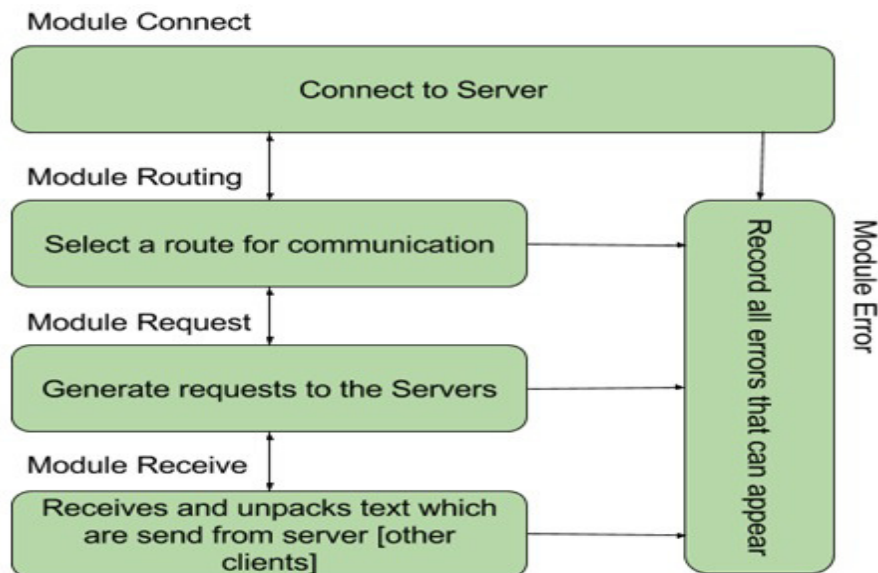
The server application is made up of five composing modules:

- **Module Connect** - used to connect and disconnect client.
- **Module Error** - used to record all errors that can appear.
- **Module Log** - used to record all operations for connection and disconnection between clients.
- **Module Transfer** - used to mediate the transfer between clients when these cannot connect directly.
- **Module Management**- used to view all active connections between clients.

Figure 5: The interaction between the server's modules

3.2 Client Application

The client application runs on a computer that represents the source from where we want to send or receive messages. The following diagram in Figure 6 represents the interaction between the client modules

Figure 6: The interaction between the client modules

The client application is made up of five composing modules:

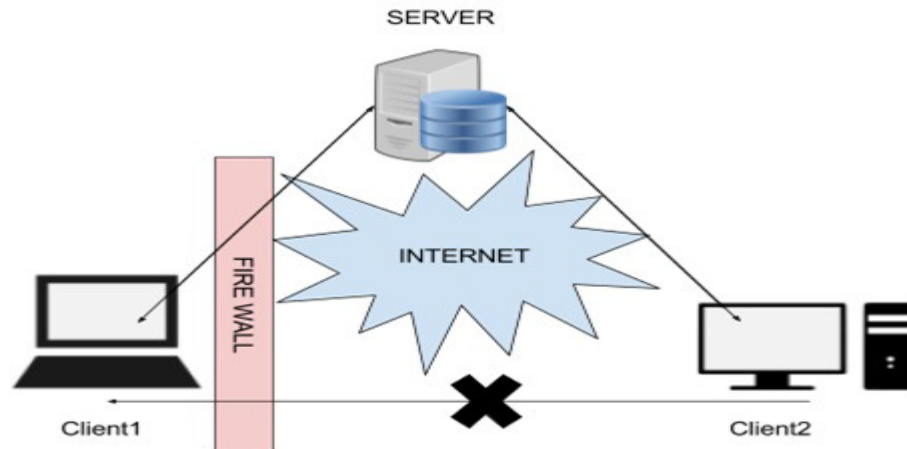
- **Module Connect** - used to connect and disconnect to server.
- **Module Error** - used to record all errors that can appear.
- **Module Routing** - used to select a route for communication between many clients. The selection of a route is explained below.
- **Module Request** - used to generate requests to the remote, the requests to send or receive a file.
- **Module Receive** - used to receive and unpack message from other client.

There are three cases in which a client computer can be:

- The client1 is behind a firewall or has an IP address unreachable in Internet and the client N-side is not behind a firewall and has an IP address reachable in Internet, as it is shown in the Figure 7.

In this case, because of the firewall or of the IP Address unreachable in Internet on the client N-side the client1 cannot communicate with the other client's computer.

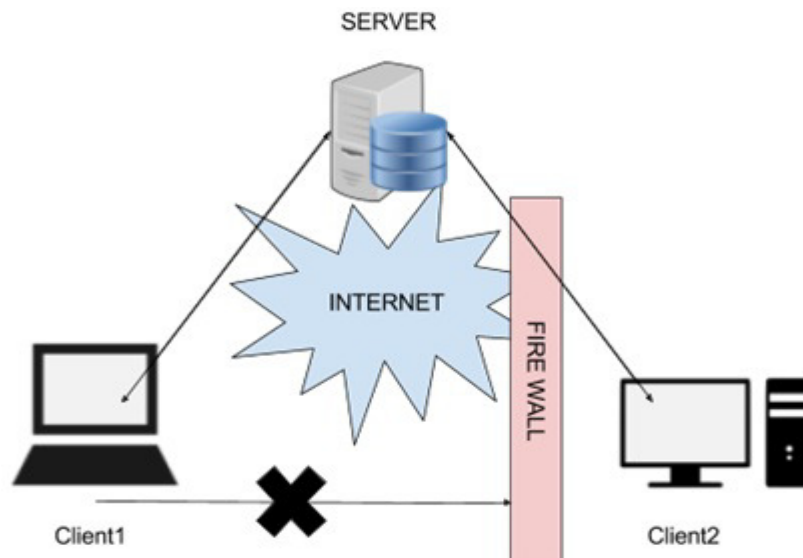
Figure 7: The client1 is behind a firewall



A solution for this problem in Figure 7 is that the client1 must initiate communication with the client2. If the client1 cannot find the client2 then the communication must be made via Server.

- The client1 is behind a firewall or has an IP address unreachable in Internet and the client N is not behind a firewall and has an IP Address reachable in Internet, as it is shown in the Figure 8. In this case, because of the firewall or of the IP address unreachable in Internet on the client1-side, the other client cannot communicate through with the client's computer.

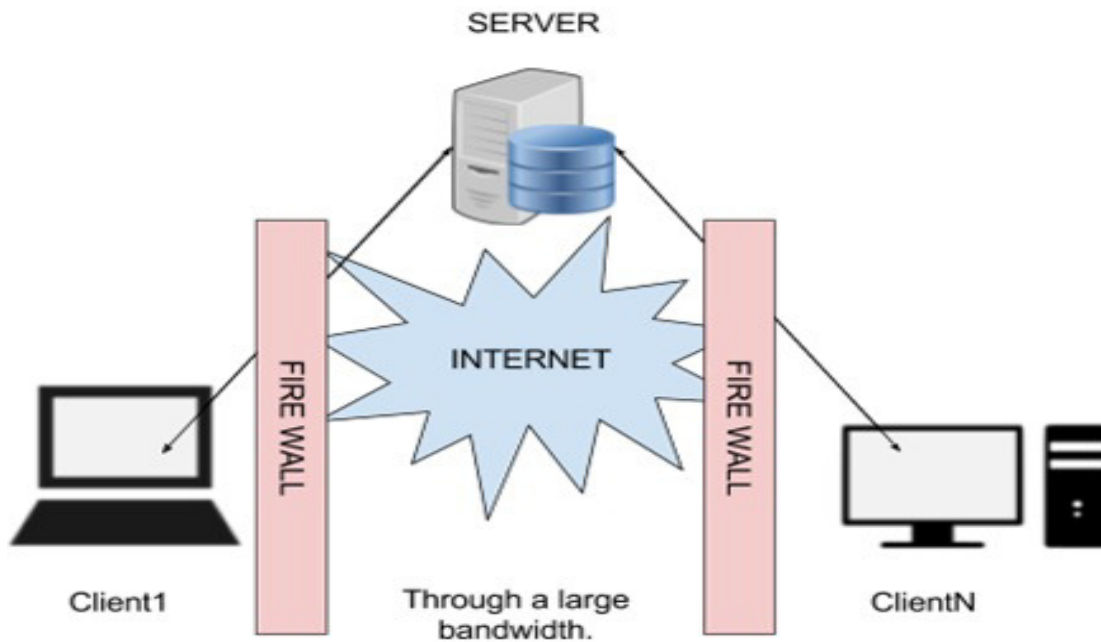
Figure 8: The client2 is behind a firewall



A solution for this problem is that the client must initiate communication with the other client, If the client cannot find the other then the communication must be made via server.

- The clients are both behind a firewall or have IP address unreachable in Internet, as it is shown in the Figure 9 In this case, because of the firewall or the unreachable IP Address in Internet on the client1-side and the client N-side, the client cannot communicate with the other, and the other cannot communicate with the client1, either.

Figure 9: The Clients behind a firewall



A solution for this problem is that the communication between the multi computers must be made via server.

For this reason, the Server must be a very fast computer, which must be connected to the Internet.

The researcher implements all this cases employing a very simple solution that is uses plain sockets with WCF. All these cases are implemented in clients by module routing.

3.3 Important Libraries used

- **System. Service Model:** This configuration section contains all the WCF ServiceModel configuration elements. And can using it only in Console application.
- **System.Runtime. InteropServices:** provides a wide variety of members that support COM interop and platform invoke services.

3.4 Diagrams and Flow Chart of the Application

Figure 10: Functional Diagram of the application

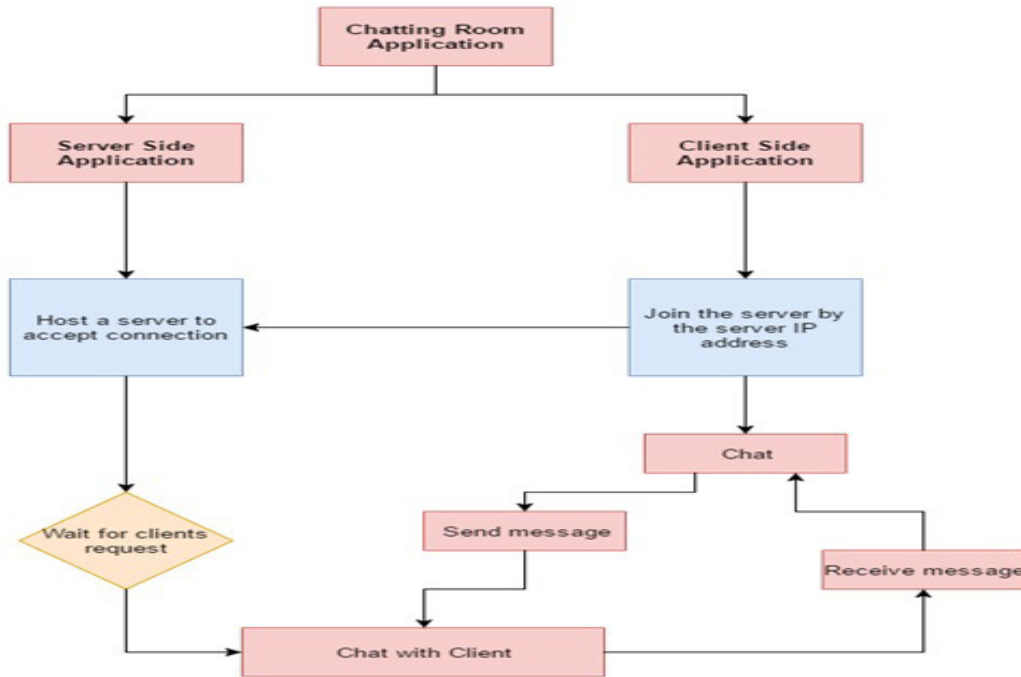


Figure 11: Flow Chart of the application

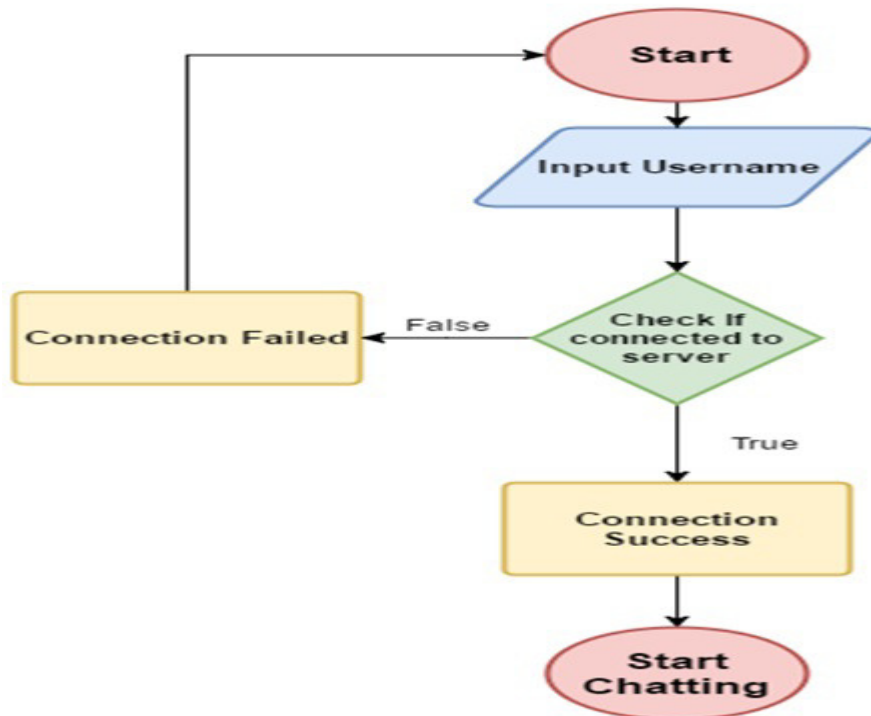


Figure 12: Use Case Diagram of the application

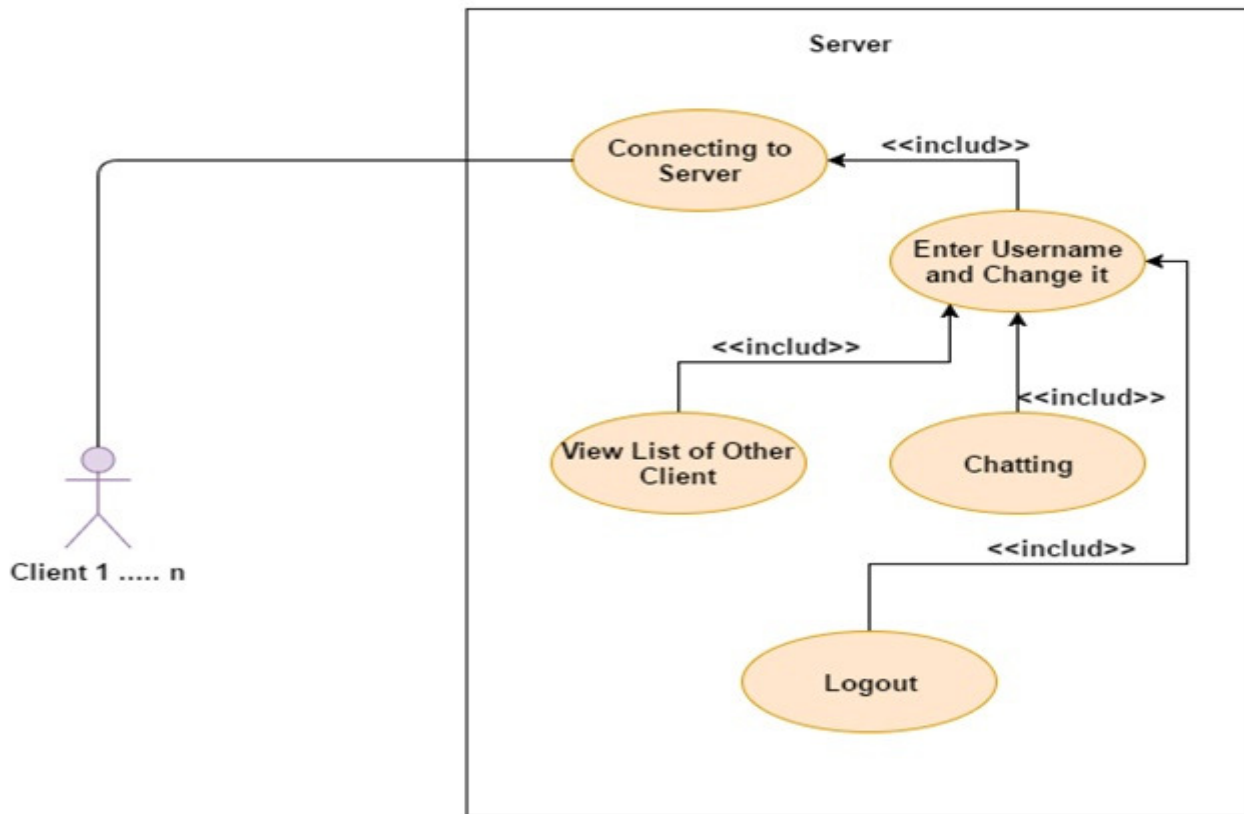


Figure 13: Sequence Diagram of the application

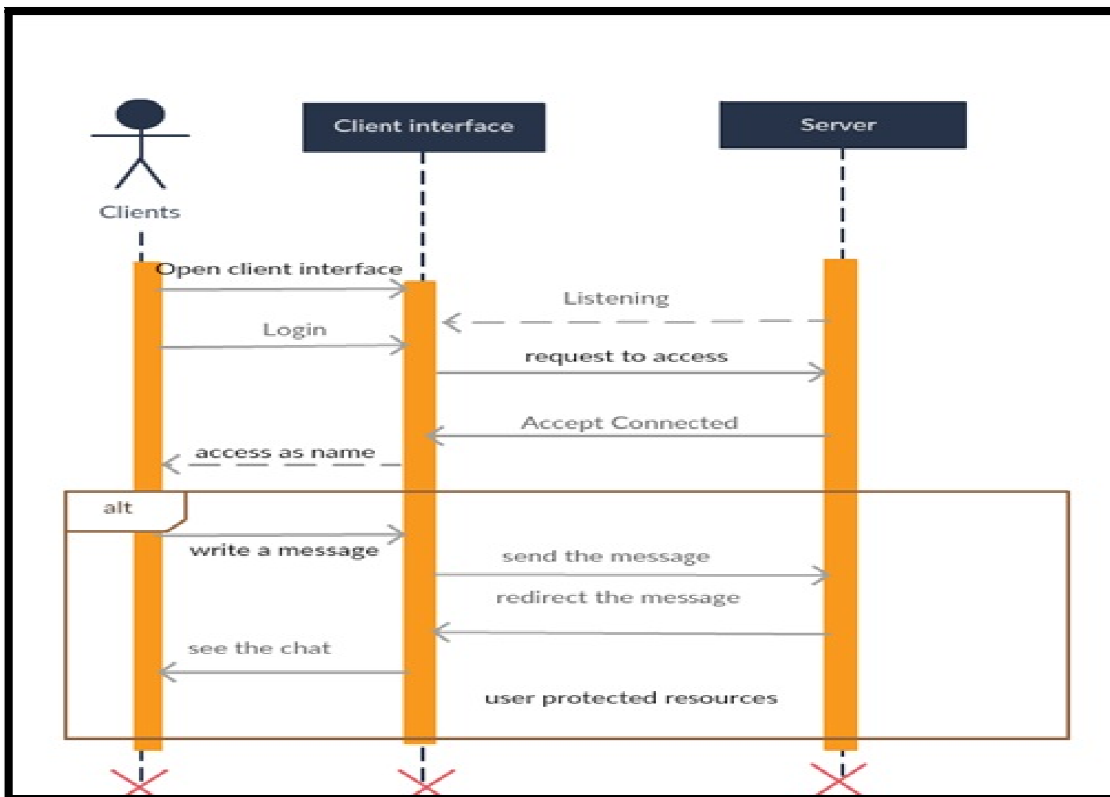
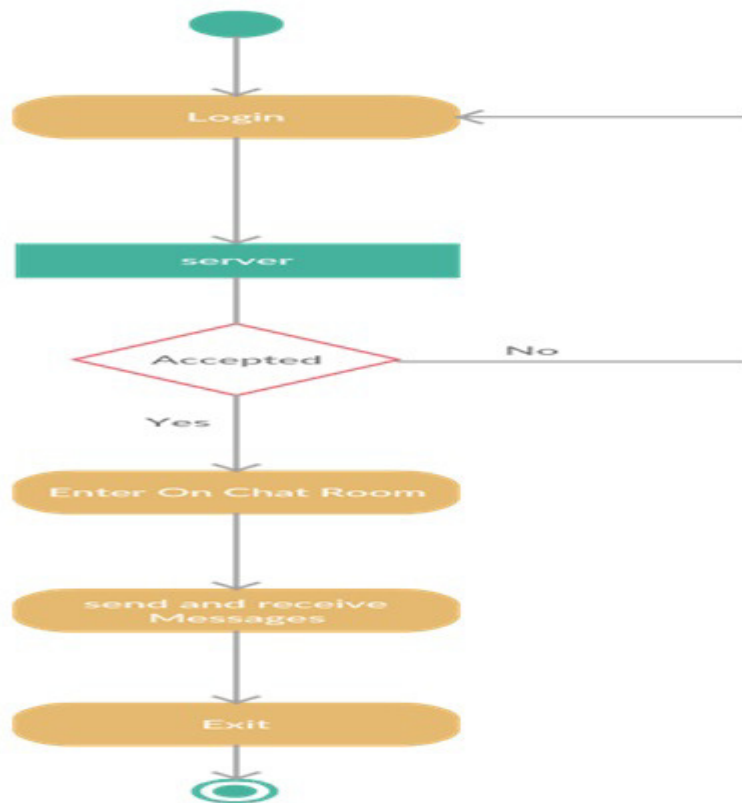


Figure 14: Activity Diagram of the application



4. Result and Interfaces

Figure 15 represent form server start hosting a clients, there have a server WCF services that drive the connection with clients and use **System.ServiceModel** Library.

Figure 15: Server start-form

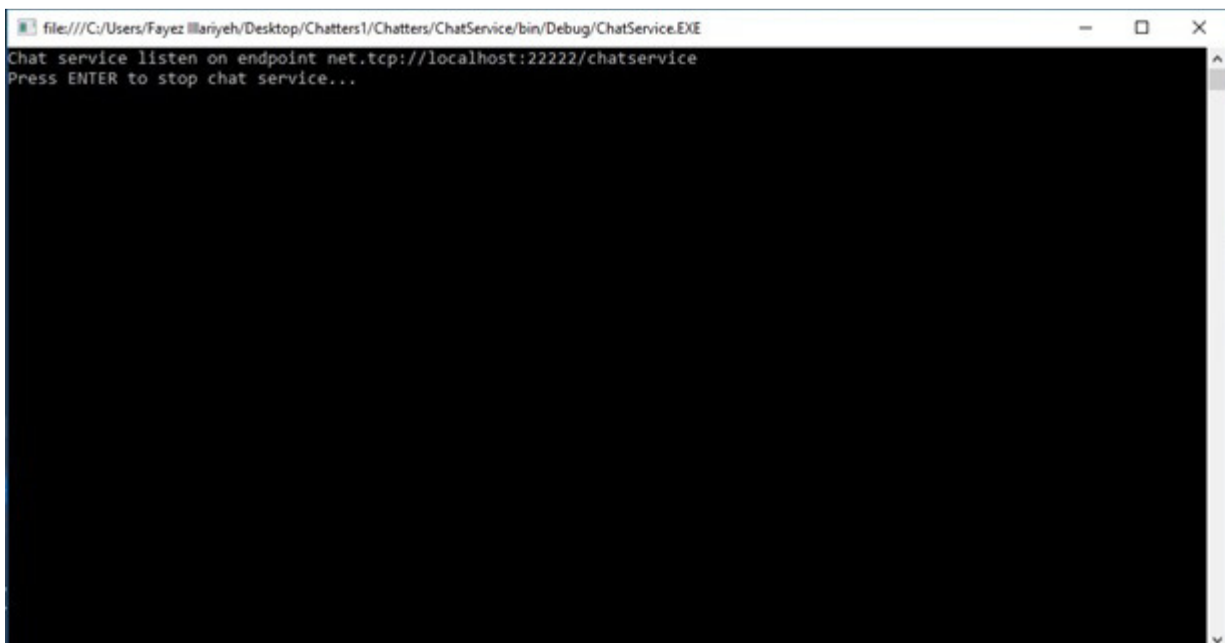


Figure 16 the start window of client application, the client must login into chat room by name and enter his photo.

Figure 16: Client application - start form

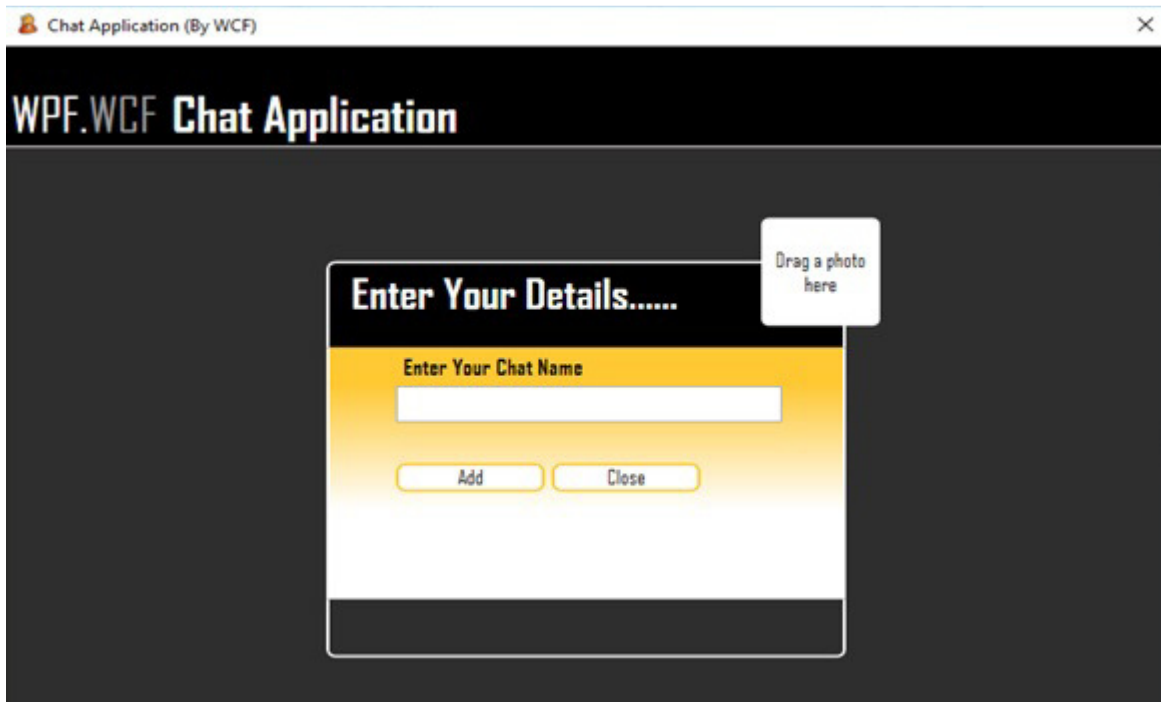
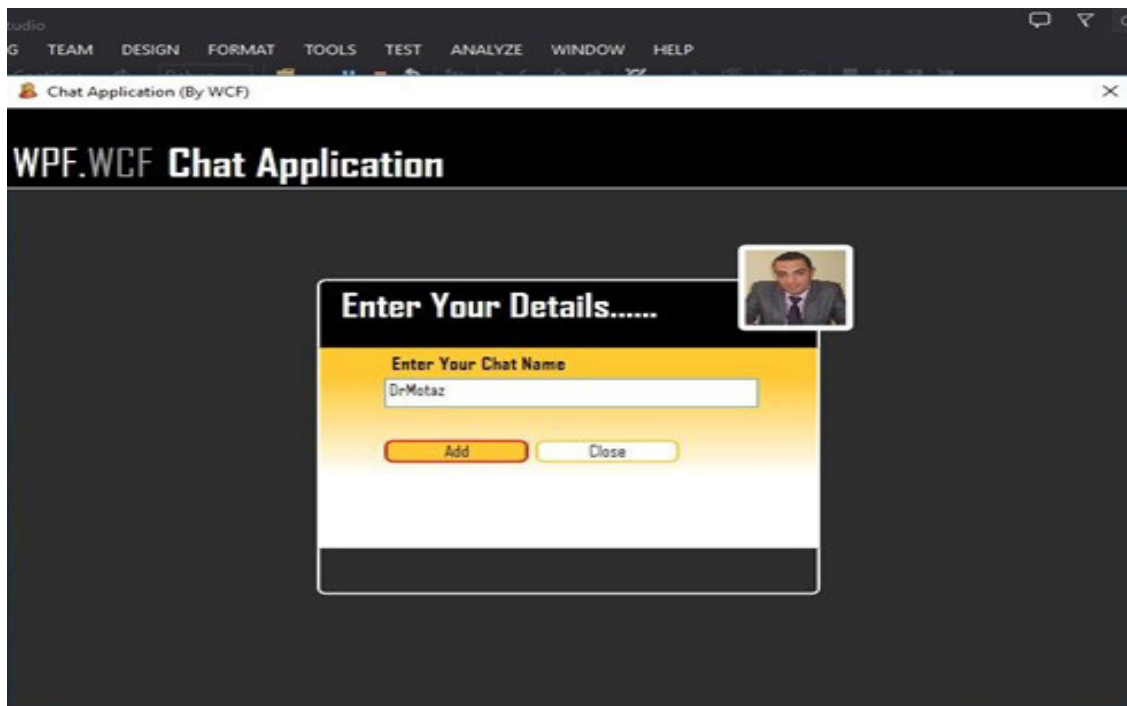


Figure 17: The client entered his name and chose his photo



In Figure 18 client has connected with server and enter to a chatting window, he can see a client list, messages, and send a message. The client name is shown in bellow of the interface.

Figure 18: Client entered to the chat

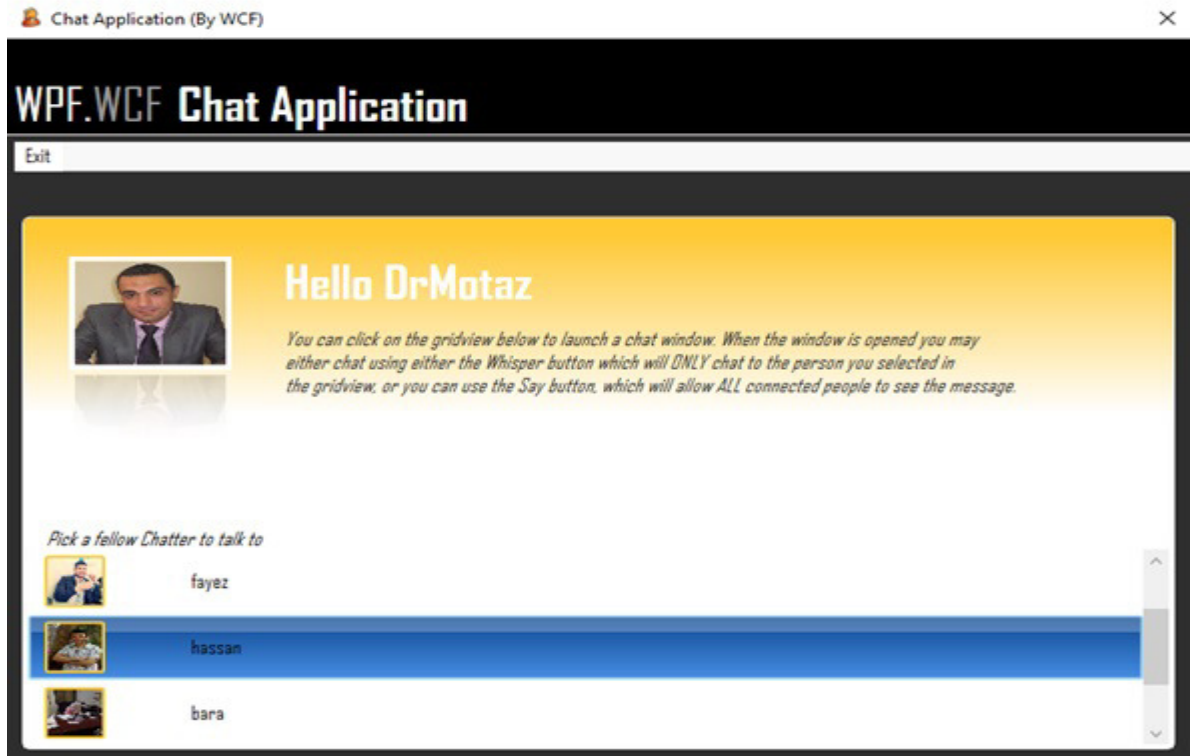


Figure 19: Button say send the message so all the clients can see the message

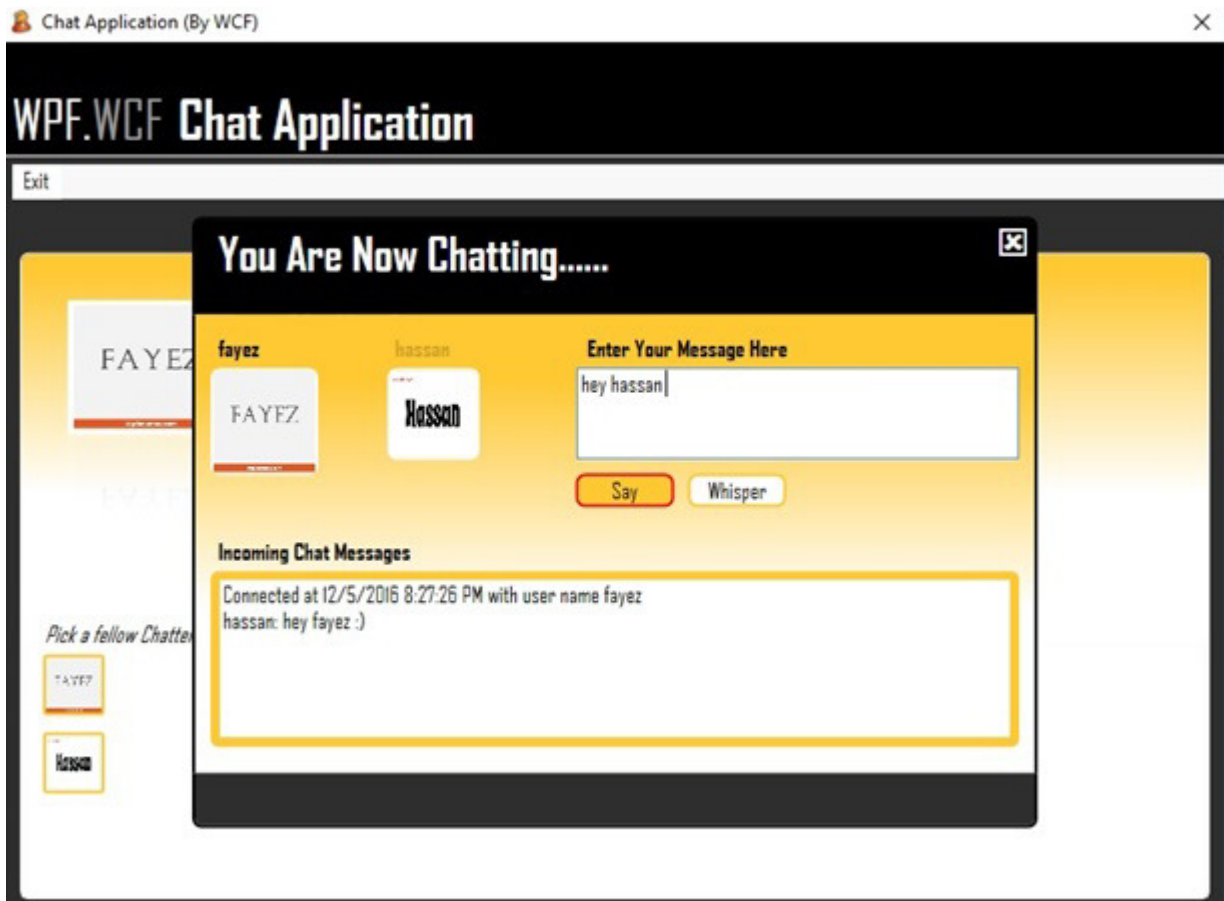


Figure 20: Button whisper sends the message to specific client so the other can't see the message

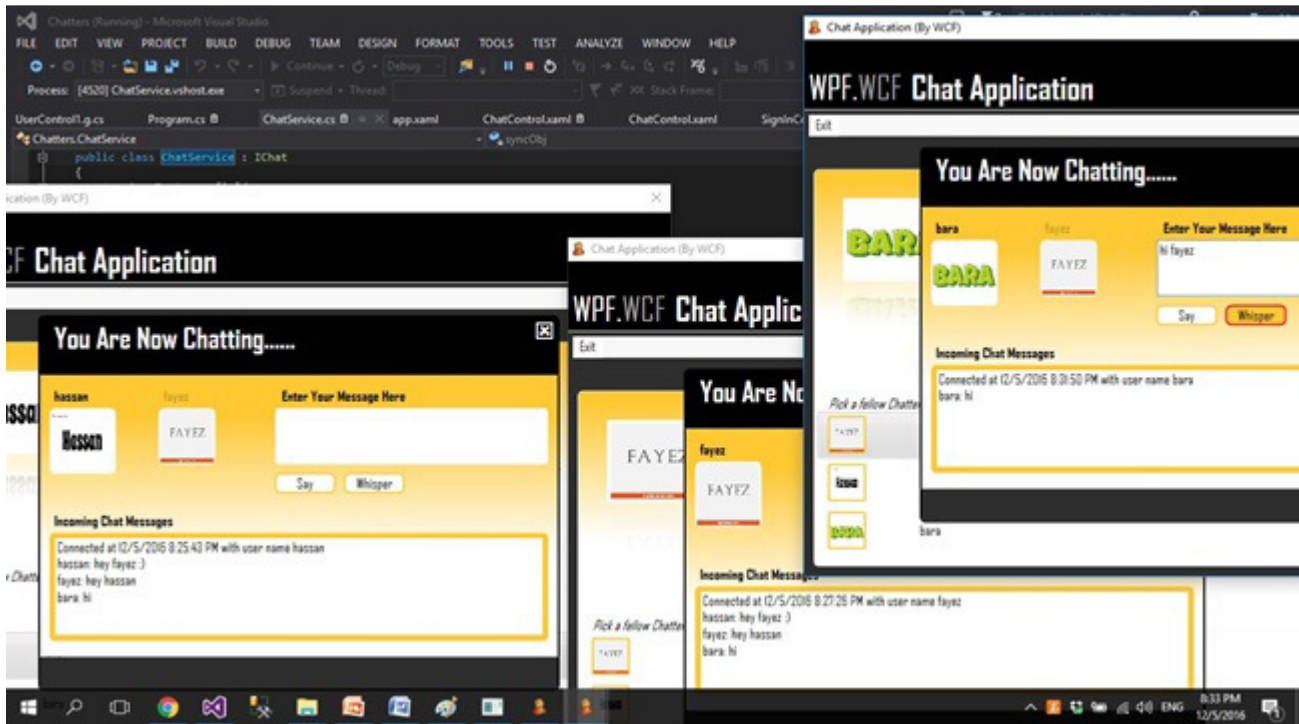


Figure 21: The clients talk to each other through the application

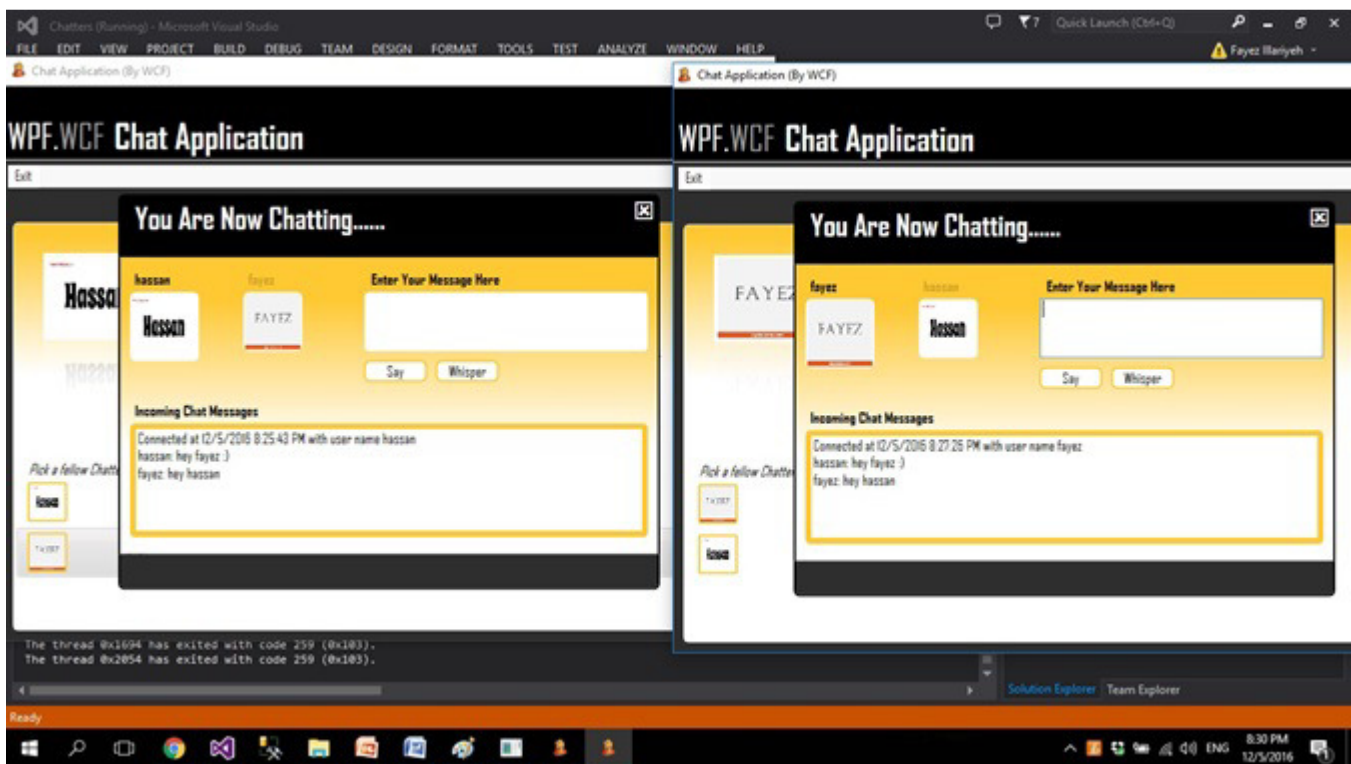


Figure 22: The chat between the clients

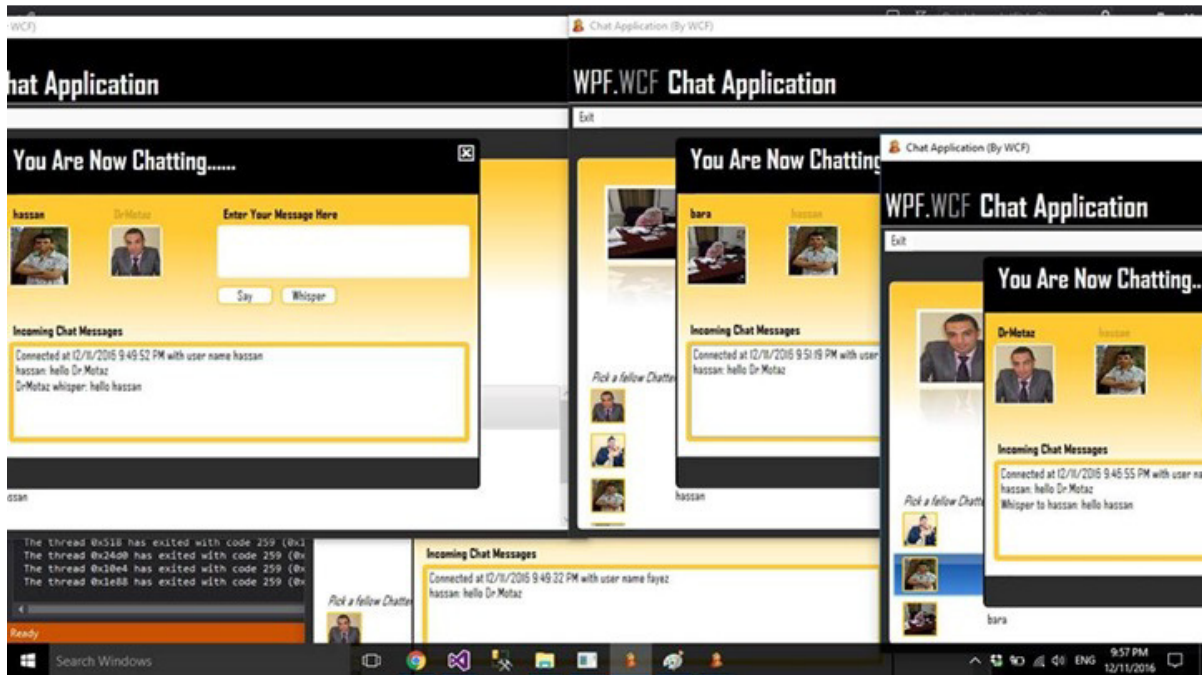


Figure 23: Statistic about WCF services



5. Conclusion

The application described in this paper is made for sending messages from a computer that is somewhere in Internet (at home or at office) to another computer or devices that can be connected to Internet wired or wireless (laptop, PDA, Pocket PC).

WCF service is one of the most noteworthy ways to implement an client-server application in .NET environment. It allows effective separation of the server side and everything inside the server side from the client side. Client-server application using WCF service can have both a single or multiple clients connecting to a single or multiple services.

The use of a service is not restricted to a single application and can be used across multiple platforms as well. Since the WCF service uses contracts to identify data and methods, multiple application types, using the same or different programming language, can access the service because data types and other useful information can be identified from the contracts.

The WCF service can easily be implemented with few simple steps, the basic structure of which can be automatically generated by Visual Studio. The most important parts left for a programmer to do are to specify service configurations and assign correct contracts to methods, properties and classes, to be used by the service. If the WCF service is done using the best practice and up to SOA standards, the contracts of the service should be separated into their own interface, which the actual service implements. Result of the paper was a working implementation of the WCF service and when it was compared to other possible implementation methods, the WCF service was an optimal choice for applications with multiple concurrent users.

Acknowledgment

The author would like to thank Palestine Technical University – Kadoorie (PTUK) for supporting this research and allowing him to conduct this work in the university labs. The application is developed for educational purpose and is currently in use at Computer Networks Lab at University.

References

- [1] Purwati, Y., & Utomo, F. S. (2014). The Development of Windows Communication Foundation Application as Middleware for Banyumas Tourism and Culinary Information System. TELKOMNIKA (Telecommunication Computing Electronics and Control), 12(4), 1031-1038.
- [2] Rouse, M. 2007. Requirements analysis (requirements engineering).
- [3] URL: <http://searchsoftwarequality.techtarget.com/definition/requirements-analysis>. Accessed: 18.10.2015.
- [4] Mancas, C., & Mocanu, M. (2013, July). QoS optimization in congested multimedia networks. In Telecommunications and Signal Processing (TSP), 2013 36th International Conference on (pp. 38-42). IEEE.