Integrated Messaging System

By

Syed Iqbal Haider Kazmi

A report submitted to Quaid-i-Azam University Islamabad
As a partial fulfillment of the requirements for the
Degree of M.Sc. in Computer Science

August 2006

QUAID-I-AZAM UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE

Dated: Sept. 09 ,2006

FINAL APPROVAL

This is to certify that we have read the project report submitted by Syed Iqbal Haider and it is our judgment that this report is of sufficient standard to warrant its acceptance by the Quaid-i-Azam University, Islamabad for the degree of Master of Science in Computer Science.

COMMITTEE:

1. External Examiner

   Dr. A. Sattar
   Ex-Director General
   Pakistan Computer Bureau
   Islamabad.

2. Supervisor

   Ms. Muddassira Arshad,
   Lecturer
   Deptt. of Computer Science
   Quaid-i-Azam University
   Islamabad

   Muddassira Arshad.

3. Chairman

   Dr. Muhammad Afzal Bhatti
   Chairman
   Deptt. of Computer Science
   Quaid-i-Azam University
   Islamabad

Dedicated to...

First of all thanks to Almighty Allah.

My Great loving and Caring Parents ,Teachers and Friends whose pray and affection has enabled me to complete such a difficult task, to my family members whose inspirations was the source of encouragement for me.
And especially to Kinza.

Acknowledgement

I would like to extend my humble and gratitude to Almighty Allah who endowed me potential and ability to complete this project. During my master program I received the help, support and encouragement from several people. It is really a pleasure for me to acknowledge this cooperation.

First and foremost my **parents** and my other Family **Members** who practically freed me from all responsibilities and who constantly prayed for me through out my academic career.

I am greatly thankful to ***Dr.Afzal Bhatti***, Chairman Department of Computer Science Quaid-i-Azam University whose guidelines during the final semester are really great supplement for my future life.

I am greatly thankful to internal supervisor ***Ms Muddassira Arshad*** for her healthy suggestions and constant encouragement to complete this thesis report. She was really there to help me whatever she can do. I greatly appreciate her caring attitude to me. Whenever I went to her for some help, she gave me the time even when she was busy.

Once again I will say Miss thank you for your sincere cooperation.

I would like to thank to **Mr. Junaid Jalib** the **CEO Xorlogics Inc. Islamabad** who guide me through out my project.

I would like to pay special thanks to ***Ajmal Bhai,*** Software Developer Project Vision. Limited for providing me the technical guidance and moral support during the development of project.

I would like to express my sincere gratitude to my best friends **Amir Hassan, Niaz Hussain, Kashif Ali, Aasif** and other my class felows **who** gave me tremendous moral support and encouragement at each and every step during the development of project. I will never forget the company of these cute and closet buddies. Guys I will never forget your association.

Syed Iqbal Haider Kazmi

Project Brief

Project Title:	Integrated Messaging Server
Organization:	Xorlogics Inc. Islamabad
Undertaken by:	Mr. Syed Iqbal Haider Kazmi
Internal Supervisor:	Ms Muddassira Arshad
Starting date:	March 2006
Completion date:	August 2006
Software Used:	Visual Studion.Net
Operating System:	Microsoft ® Windows ® XP Professional 2006
System Used:	Pentium-III
	RAM 192 MB

Abstract

This project is aimed at R&D in the field of Real Time Communications and Voice over Internet protocol by following very rich and robust H.323 protocol for Integrated Messaging using the Microsoft based platforms, like MS Windows XP Professional.

The system has two main parts. The Audio Call Handling, Sending Notifications.

The first part of the system is concerned with audio call handling. This part of the system receives audio calls, plays greetings messages to callers and stores their voice messages. This module behaves like a global answering machine.

The second part of the system provides notification message sending. This is an important part of the Integrated Messaging System. The users are notified by sending different kind of messages to make them aware of their new messages.

The SDK's like openH323 and PWLib are that provide rich set of API's for call features over packet switched networks. The IDE used are Microsoft® Visual Studio.Net and Tool to be used are Visual C++ .NET.

Preface

This report represents description of Analysis, Design, Development, Implementation and Testing of "Integrated Messaging System". The entire work is summarized in chapters and some Appendixes.

Chapter-1	This chapter covers the vision of the project and effects of the product.
Chapter-2	This chapter covers the research work done during the project. It provides the introduction to Voice Over internet protocol concepts and H323 protocol overview and H323 based VOIP Networks architectures.
Chapter-3	This chapter concerns about requirement engineering, project plan resources to be used.
Chapter-4	This chapter contains items regarding analysis of system.
Chapter-5	This chapter introduces Object-Oriented design of the system. It includes class diagram, sequence diagram and their description
Chapter-6	This chapter includes documentation about implementation of system.
Chapter-7	This chapter has documentation about testing of the system.
References	Reference of books used in this documentation.
Webliography	URL's of web sites, those were useful in development process.
Glossary	Glossary of terms used in this documentation.

Contents

1. Introduction 1
1.1 Purpose of document 1
1.2 Introduction to the Organization 1
1.3 Problem Definition 1
1.4 Scope 2
1.4.1 System Context 2
1.4.2 Information Objectives 2
1.4.3 Function and Performance 3
1.5 Objectives 4
1.6 Product Description 4
1.6.1Operating System 4
1.6.2 Software applications 4
1.6.3 Hardware Devices 4
2. Literature Survey 5
2.1 Existing systems 5
2.1.1 NEAXMail® AD-64 5
2.1.2 NEAXMAIL® AD-120 FEATURES (Ver. 2.8.0) 5
2.1.3 Active Voice 5
2.2 Introduction to VoIP 5
2.2.1 Application Of VoIP 6
2.3 Voice Communication Protocols for VoIP 7
2.3.1 H.323 7
2.3.2 SIP 7
2.3.3 MGCP 7
2.4 Factors to consider when designing VoIP Network 7
2.4.1 Delay 7
2.4.2 Jitter: 8
2.4.3 Serialization 8
2.4.4 Bandwidth Consumption 8
2.4.5 Difference between IP Telephony and PSTN 8

2.5 Introduction to H.323 9
2.5.1 Definition 10
2.5.2 H323Versions 10
2.5.3 H.323 in Relation to Other Standards of the H.32x Family 10
2.5.4 Overview 11
2.5.5 Interworking with Other Multimedia Networks 11
2.5.6 H.323 Zone 12
2.5.7 H.323 Components 13
3. Requirement Engineering 17
3.1 Requirement Definition 17
3.1.1 Functional Requirements 17
3.1.2 Non-Functional Requirements 17
3.2 Requirement Specification 17
4. Analysis and Design 23
4.1 User Model View 23
4.1.1 Identification of Actors 23
4.1.2 Identification of Use Cases 23
4.1.3 Use Case Description (Use-Case Template) 24
4.1.4 Use Case Diagram 26
4.2 Structural Model View 27
4.2.1 Identification of Classes 27
4.2.2 Class Diagram 28
4.2.3 Classes Description 29
4.3 Behavioral Model 33
4.4 Architectural View 36
4.5 Design Decisions 37
5. Implementation 39
5.1 Platform Selection 39
5.2 Tools and SDK Selection 39
5.3 Code Documentation Standards 39

5.4 Deployment Diagram.. 41
5.5 Implementation Decisions.. 41
6. Testing.. 42
6.1 Software Testing Introduction.. 42
6.2 Software Testing Objectives... 42
6.3 Selected Testing Strategy.. 42
6.3.1 Black Box Testing.. 42
6.4 Test Cases.. 43
References.. 49
Webliography.. 50
Glossary... 51

Chapter 1

Introduction to VoIP

1. Introduction

1.1 Purpose of This Document

This Software Requirement Specification document describes the major work completed during the Analysis phase of the Project for Xorlogics Inc. Islamabad, A multinational organization. It is designed to express the product specification precisely and accurately in terms of there.

- Purpose
- Scope
- Objectives

1.2 Introduction to the Organization

Xorlogics Inc. Islamabad (Merging Ideas into Technology) is a well-reputed company, which works on mobile applications, web development and, VoIP (Voice over Internet Protocol) on national and international level. Xorlogic`s vision is to become a leading Product Development Company. Xorlogics build on the product concept to help companies realize a commercial product with a focus on their valued proposition consisting of low cost, high quality and on time delivery.

1.3 Problem Definition

People are inundated with messages and communications, both real-time and non-real time, necessitating a better way to reduce messaging chaos and manage all of the voice mail, e-mail, text, and fax messages workers receive every day. The solution is Integrated Messaging (IM), the integration of voice mail, fax, e-mail, allowing users to access any of these messages, anywhere, anytime, from any terminal of choice. The goal of a IM system is to simplify and speed up communication processes to achieve time and cost savings for companies and/or individuals. IM makes it easier for users and system administrators to manage the influx of messages, while providing enhanced voice mail and fax capabilities. IM enables users to access voicemail, email, and fax messages from a single user interface, making messages accessible via telephone, wireless device, or Web interface. It can be premise-based, using a server to bring together the applications, or it can be offered by a service provider as a network-hosted service.

Following are the three features of an Integrated Messaging solution:

1. Voice Messaging over IP

2. E-Mail Messaging over IP
3. Fax Messaging over IP

In the first phase of development Voice Messaging over IP's features will be provided to customers. This is what is provided in current version of the Integrated Messaging Server. This application is based on IP-IP Calling and Messaging features.

1.4 Scope

Software scope is determined by answering the:

- System Context
- Information Objectives
- Functions and Performance

1.4.1 System Context:

The Integrated Messaging Server will be installed on some central location having a unique address where it will communicate with some Gatekeepers or H323 Endpoints to receive the calls refused by their attendants. Gatekeepers and Gateways are the essential parts of the system. Gatekeepers are used to register the Endpoints to rout calls to them. The Gateways will be acting as bridge between traditional analog PSTN network and Digital IP network. The user will make a call through Dialer by providing destination information and Gatekeeper address with which it is registered. When the Gatekeeper will route the call to its destination and call is not received by the resource person then the Gatekeeper will transfer call to Integrated Messaging Server so that the caller's voice message can be stored and the resource person can be notified. The user can retrieve his messages by log in to his mailbox from the web interface provided.

1.4.2 Information Objectives:

The major input and output data objects for the system are following:

Input objects

- Messages codes
- Redirected call information
- IP addresses and Gatekeeper requests

Output Objects

- Audio messages played to the caller
- Notification messages i.e., email
- Email attachments of audio messages

1.4.3 Function and Performance:

The major functions that system will provide are

- VoIP Incoming Call Handling
- VoIP based Simultaneous Multiple Calls Handling
- Personalized Greeting Message Playing
- Voice Message Recording
- Notification Message Sending i.e. email
- Voice Message Retrieval

Incoming Calls

The system will be able to accept call when Gatekeeper will transfer call after sensing that the resource person is not attending call after a specified amount.

Simultaneous Multiple Calls

As Integrated Messaging System will be robust and have the ability to accept more than one call at the same time because it is possible that more than one people can talk to each other at the same time.

Personalized Greeting Message Playing

When the system accepts a call, it will play a personalized greeting message to the caller recorded by the user.

Voice Message Recording

After accepting call and playing greeting message the system will have ability to record the caller's voice message. The caller will leave message for the user so this way the user can never miss any of his messages.

Notification Message Sending

As soon as the voice message of the caller is stored, the user will be notified through an email so as he may be able to retrieve his missing messages as soon as possible.

Message Retrieval

The user will be able to retrieve his messages. So the web interface should be present to retrieve the message globally from any where in any part of the world.

1.5 Objectives

In the light of the problem discussed earlier the developed software solution of an Integrated Messaging that fulfill the following major objectives:

- It should have the functionality of being registered with the gatekeeper automatically, manually or by discovering it.
- It should have the functionality of receiving the call after sensing that the call is not received by the callee. This procedure should take atmost 10 seconds.
- It should have flexibility to accept multiple calls at the same time.
- It should play the personalized messages for each and every caller.
- It should notify the user by email or SMS(if possible).

1.6 Product description

1.6.1 Operating System

- Name: MS windows XP, 2000 or Above (XP Recommended)
- Version: XP with service pack 2.

1.6.2 Software Applications

- Internet Information Server Version 6.0
- Visual Studio.Net

1.6.3 Hardware Devices

- The software product will operate on a client-server environment so an active connection of internet is required for which modem is needed.
- Some type of device to record the voice message.

Chapter 2

Literature Survey

2. Literature Survey

2.1 Existing Systems

2.1.1 NEAXMail® AD-64

The foundation for a powerful unified communications solution is based on the NEAXMail® AD-64, NEC's new Windows® 2003-based voice messaging system. Standard features include advanced unified messaging, voice mail, and automated attendant.

The desktop Mailbox Manager on client machines allows end users to modify their own mailbox greetings, notifications, groups, conversation preferences, and other settings. The Live Record feature enables a subscriber to record incoming calls at any point during the call and save the recording to a personal mailbox and/or forward it to other subscribers. This allows phone conversations to be archived for future reference.

2.1.2 NEAXMAIL® AD-120 FEATURES (Ver. 2.8.0)

- Answer calls on first ring
- Message delivery and call transfer
- Network dial by name
- Network dial by IP
- Future message and delivery
- Multiple addressing of messages
- Record Call Handler Greetings over the phone

(http://www.necunifiedsolutions.com/main/Products/ProProduct.asp?solid=5&prodid=64)

2.1.3 Active Voice

Active Voice, LLC is a global provider of unified messaging, computer telephony and voice messaging solutions that empower people to communicate and manage information with convenience and efficiency. Their business solutions allow people to use virtually any communication device to access and control their messages, data and real-time communications. (http://www.activevoice.com.au/)

2.2 Introduction to VoIP

Voice over IP (VoIP) is the real time delivery of voice across networks using the Internet Protocols. Important driving force behind voice over IP is cost saving. Circuit switching carries voice well, but is expensive. IP is attractive for voice because of lower equipment cost, integration of voice and data, lower bandwidth requirements and widespread availability of it. IP telephony is cheap compared to high cost of long distance and international voice calls. New services can be easily enabled in IP telephony because the intelligence lies in the end systems and IP networks are open in nature. Also many of the existing services like caller-id, call forwarding, multi-line presence will be very easy to implement and also will be offered for free. For corporate carrying voice on existing data networks is cost efficient and also integration of voice and data applications can result in efficient business process. Examples are integrated voice mail and email, teleconferencing, automated and intelligent call distribution etc.

The main challenge for VoIP is the voice quality should be as good as today's telephone networks. Also the ease of operation, functionality should be the same as public switched telephone network (PSTN). This becomes really challenging because IP is not designed to carry voice or real time traffic. The important requirements for quality voice are less delay, minimal jitter, low (or no) packet loss, and speech-coding techniques that maintain natural speech as well as require low bandwidth. Also very important are reliability and scalability of a VoIP system.

(http://ezineararticles.com/?introduction-to-VoIP&id=65549)

2.2.1 Application Of VoIP

Today VoIP is used in following applications.

- Toll bypass – Reduce cost of long distance calls by routing them through IP-based networks.
- Call center integration – Call centers that use one link to talk to customers and one to retrieve data can combine them into one saving costs. Also customers can go find information in web site and make a call directly from there.
- Integrated Messaging – which allows users to receive various forms of messages (voice, email, fax etc) at a single access point?
- IP videoconferencing – which will reduce cost for businesses.
- Corporate intranets – integration of voice and data networks in corporate using IP based PBX solutions, reducing infrastructure and administration costs.
- Hosted PBX solutions for small office/home office (SOHO).

2.3 Voice Communication Protocols for VoIP

2.3.1 H.323:

The International Telecommunication Union-Telecommunications standardization Sector (ITU-T) specification for sending voice, video, and data across a network. The H.323 specification includes several related standards, such as H.225 (call control), H.235 (security), H.245 (media path and parameter negotiation), and H.450 (supplementary services).

2.3.2 SIP:

This is the IETF's standard for establishing VOIP connections. It is an application layer control protocol for creating, modifying and terminating sessions with one or more participants. The architecture of SIP is similar to that of HTTP (client-server protocol). Requests are generated by the client and sent to the server. The server processes the requests and then sends a response to the client. A request and the responses for that request make a transaction [RFC2543].

2.3.3 MGCP:

It is a protocol that defines communication between call control elements (Call Agents) and telephony gateways. Call Agents are also known as Media Gateway Controllers. It is a control protocol, allowing a central coordinator to monitor events in IP phones and gateways and instructs them to send media to specific addresses. It resulted from the merger of the Simple Gateway Control Protocol and Internet Protocol Device Control [Huitema99].

2.4 Factors to consider when designing VoIP Network

2.4.1 Delay:

Delay is the time it takes for VoIP packets to travel between two endpoints. Because of the speed of network links and the processing power of intermediate devices, some delay is expected; however, you should attempt to minimize this delay. The human ear normally accepts a delay of about 150 milliseconds (ms) without noticing problems. (The ITU G.114 standard recommends no more than 150 ms of one-way delay).

2.4.2 Jitter:

Although delay can cause unnatural starting and stopping of conversations, variable-length delays (also known as *jitter*) can cause a conversation to break and become unintelligible. Jitter is not usually a problem with PSTN calls because the bandwidth of calls is fixed. However, in VoIP networks in which existing data traffic might be bursty, jitter can become a problem.

2.4.3 Serialization:

Serialization is a term that describes what happens when a router attempts to send both voice and data packets through an interface. In general, voice packets are very small (80 to 256 bytes), and data packets can be very large (1500 to 18,000 bytes). On relatively slow links, such as WAN connections, large data packets can take a long time to send onto the wire. When these large packets are mixed with smaller voice packets, the excessive transmission time can lead to both delay and jitter. You can use fragmentation to reduce the size of the data packets so that the delay and jitter also decrease.

2.4.4 Bandwidth Consumption:

Traditional voice conversations consume 64 kbps of network bandwidth. When this voice traffic is run though a VoIP network, it can be compressed and digitized by digital signal processors (DSPs built into the routers. This compression can reduce the calls to sizes as small as 5.3 kbps for voice samples. After the packets go onto the IP network, the appropriate IP/UDP/RTP headers must be added. This can add a substantial amount of bandwidth to each call (about 40 bytes per packet). Technologies such as RTP header compression, however, can reduce the IP header overhead to about two bytes.

2.4.6 Difference between IP Telephony and PSTN:

IP telephony relies on "end to end" paradigm for delivery of services. Signaling is done end-to-end; hence the call state is also end-to-end as are the instantiation of many telephony features. This leads to tremendous flexibility and extensibility. IP telephony separates call setup and reserving resources. Hence it breaks the assumption of all or nothing call completion of current telephone system. PSTN addresses are overloaded with multiple functions like end point identification, service indication, indication of who pays for the call and carrier selection. IP telephony addresses are used solely for endpoint identification and basic service indication. Payment and carrier selection are handled by protocols such as RSVP and RTSP.

Phone system employs different signaling protocols between user and Network (UNI) as compared to network elements (NNI), because of which certain features are not available to end users. This distinction does not exist in Internet both at level of data transport and signaling.

The open, end-to-end nature of Internet means that completely different service vendors provide various components of telephony services. The separation of functionality simplifies the number portability problem also.

2.5 Introduction to H.323

The H.323 standard is a cornerstone technology for the transmission of real-time audio, video, and data communications over packet-based networks. It specifies the components, protocols, and procedures providing multimedia communication over packet-based networks. Packet-based networks include IP–based (including the Internet) or Internet packet exchange (IPX)–based local-area networks (LANs), enterprise networks (ENs), metropolitan-area networks (MANs), and wide-area networks (WANs). H.323 can be applied in a variety of mechanisms—audio only (IP telephony); audio and video (video telephony); audio and data; and audio, video and data. H.323 can also be applied to multipoint-multimedia communications. H.323 provides myriad services and, therefore, can be applied in a wide variety of areas—consumer, business, and entertainment applications.

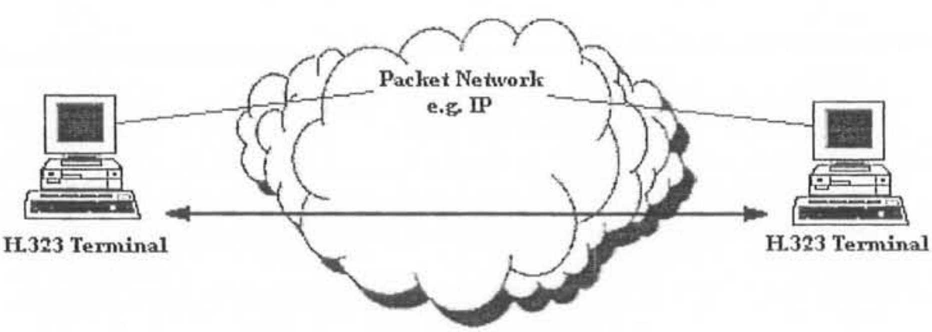


Figure 2.1 H.323 Terminals on a Packet Network
(www.technology/online/tutorials/h323/)

2.5.1 Definition:

H.323 is a standard that specifies the components, protocols and procedures that provide multimedia communication services—real-time audio, video, and data communications—over packet networks, including Internet protocol (IP)–based networks. H.323 is part of a family of ITU—T recommendations called H.32x that provides multimedia communication services over a variety of networks.

2.5.2 H.323 Versions:

The H.323 standard is specified by the ITU–T Study Group 16. Version 1 of the H.323 recommendation—visual telephone systems and equipment for LANs that provide a non-guaranteed quality of service (QoS) was accepted in October 1996. It was, as the name suggests, heavily weighted towards multimedia communications in a LAN environment. Version 1 of the H.323 standard does not provide guaranteed QoS.

The emergence of voice-over–IP (VoIP) applications and IP telephony has paved the way for a revision of the H.323 specification. The absence of a standard for voice over IP resulted in products that were incompatible. With the development of VoIP, new requirements emerged, such as providing communication between a PC–based phone and a phone on a traditional switched circuit network (SCN). Such requirements forced the need for a standard for IP telephony. Version 2 of H.323 packet-based multimedia communications systems—was defined to accommodate these additional requirements and was accepted in January 1998. New features are being added to the H.323 standard. The features being added include fax-over-packet networks, gatekeeper-gatekeeper communications, and fast-connection mechanisms.

2.5.3 H.323 in Relation to Other Standards of the H.32x Family:

The H.323 standard is part of the H.32x family of recommendations specified by ITU–T. The other recommendations of the family specify multimedia communication services over different networks:

- H.324 over SCN.
- H.320 over integrated services digital networks (ISDN).
- H.321 and H.310 over broadband integrated services digital networks (B–ISDN).
- H.322 over LANs that provide guaranteed QoS.

One of the primary goals in the development of the H.323 standard was interoperability with other multimedia-services networks. This interoperability is achieved through the use of a gateway. A gateway performs any network or signaling translation required for interoperability.

2.5.4 Overview:

This is actually a discussion of H.323 protocol standard. So H.323 is explained with an emphasis on gateways and gatekeepers, which are components of an H.323 network. The call flows between entities in an H.323 network are explained, and the interworking aspects of H.323 with H.32x family protocols are discussed.

2.5.5 Interworking with Other Multimedia Networks:

The H.323 standard specifies four kinds of components, which, when networked together, provide the point-to-point and point-to-multipoint multimedia-communication services:

- Terminals
- Gateway
- Gatekeeper
- Multipoint Control Units (MCUs)

Terminals:

Used for real-time bidirectional multimedia communications, an H.323 terminal can either be a personal computer (PC) or a stand-alone device, running an H.323 and the multimedia applications. It supports audio communications and can optionally support video or data communications. Because the basic service provided by an H.323 terminal is audio communications, an H.323 terminal plays a key role in IP–telephony services. An H.323 terminal can either be a PC or a stand-alone device, running an H.323 stack and multimedia applications. The primary goal of H.323 is to interwork with other multimedia terminals. H.323 terminals are compatible with H.324 terminals on SCN and wireless networks, H.310 terminals on B–ISDN, H.320 terminals on ISDN, and H.321 terminals on B–ISDN, and H.322 terminals on guaranteed QoS LANs. H.323 terminals may be used in multipoint conferences.

Gateways:

A gateway connects two dissimilar networks. An H.323 gateway provides connectivity between an H.323 network and a non–H.323 network. For example, a gateway can connect and provide communication between an H.323 terminal and SCN networks (SCN networks include all switched telephony networks, e.g., public switched telephone network [PSTN]). This connectivity of dissimilar networks is achieved by translating protocols for call setup and release; converting media formats between different networks, and transferring information between the networks connected by the gateway. A gateway is not required, however, for communication between two terminals on an H.323 network.

Gatekeepers:

A gatekeeper can be considered the brain of the H.323 network. It is the focal point for all calls within the H.323 network. Although they are not required, gatekeepers provide important services such as addressing, authorization and authentication of terminals and gateways; bandwidth management; accounting; billing; and charging. Gatekeepers may also provide call-routing services.

Multipoint Control Units:

MCUs provide support for conferences of three or more H.323 terminals. All terminals participating in the conference establish a connection with the MCU. The MCU manages the conference resources, negotiates between terminals for the purpose of determining the audio or video coder/decoder (CODEC) to use, and may handle the media stream. The gatekeepers, gateways, and MCUs are logically separate components of the H.323 standard but can be implemented as a single physical device.

2.5.6 H.323 Zone:

An H.323 zone is a collection of all terminals, gateways, and MCUs managed by a single gatekeeper. A zone includes at least one terminal and may include gateways or MCUs. A zone has only one gatekeeper. A zone may be independent of network topology and may be comprised of multiple network segments that are connected using routers or other devices.

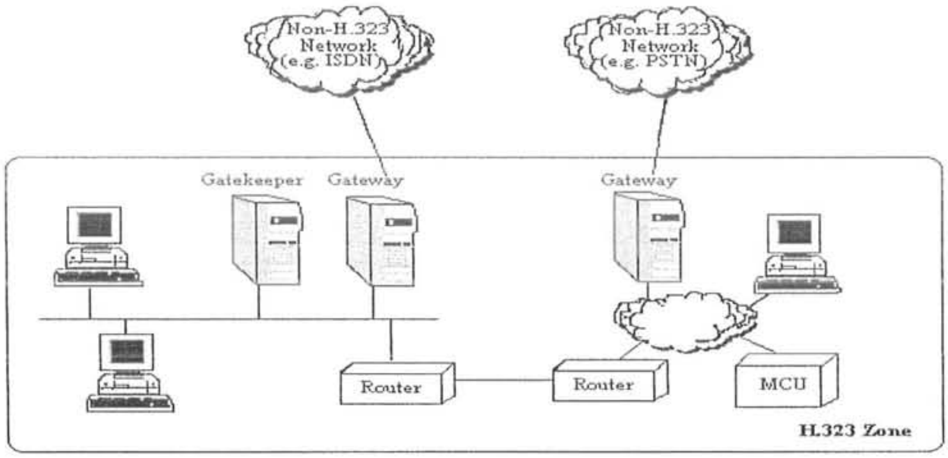


Figure 2.2 an H.323 Zone

2.5.7 H.323 Components:

The protocols specified by H.323 are listed below. H.323 is independent of the packet network and the transport protocols over which it runs and does not specify them.

- audio CODECs
- video CODECs
- H.225 registration, admission, and status (RAS)
- H.225 call signaling
- H.245 control signaling
- real-time transfer protocol (RTP)
- real-time control protocol (RTCP)

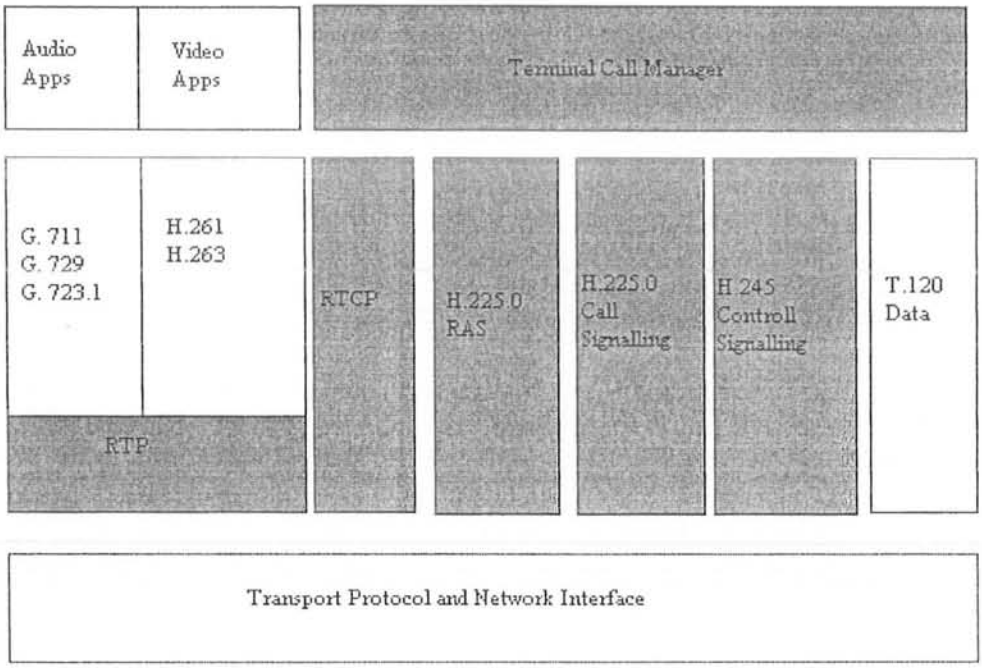


Figure 2.3 H.323 Terminal-Side Protocol Stack

Audio CODEC:

An audio CODEC encodes the audio signal from the microphone for transmission on the transmitting H.323 terminal and decodes the received audio code that is sent to the speaker on the receiving H.323 terminal. Because audio is the minimum service provided by the H.323 standard, all H.323 terminals must have at least one audio CODEC support, as specified in the ITU–T G.711 recommendation (audio coding at 64 kbps). Additional audio CODEC recommendations such as G.722 (64, 56, and 48 kbps), G.723.1 (5.3 and 6.3 kbps), G.728 (16 kbps), and G.729 (8 kbps) may also be supported.

Video Codec:

A video CODEC encodes video from the camera for transmission on the transmitting H.323 terminal and decodes the received video code that is sent to the video display on the receiving H.323 terminal. Because H.323 specifies support of video as optional, the support of video CODECs is optional as well. However, any H.323 terminal providing video communications must support video encoding and decoding as specified in the ITU–T H.261 recommendation.

H.225 Registration, Admission, and Status (RAS):

Registration, admission, and status (RAS) is the protocol between endpoints (terminals and gateways) and gatekeepers. The RAS is used to perform registration, admission control, bandwidth changes, status, and disengage procedures between endpoints and gatekeepers. An RAS channel is used to exchange RAS messages. This signaling channel is opened between an endpoint and a gatekeeper prior to the establishment of any other channels.

H.225 Call Signalling:

The H.225 call signaling is used to establish a connection between two H.323 endpoints. This is achieved by exchanging H.225 protocol messages on the call-signaling channel. The call-signaling channel is opened between two H.323 endpoints or between an endpoint and the gatekeeper.

H.245 Control Signaling:

H.245 control signaling is used to exchange end-to-end control messages governing the operation of the H.323 endpoint. These control messages carry information related to the following:

- Capabilities exchange
- Opening and closing of logical channels used to carry media streams
- Flow-control messages
- General commands and indications

Real-Time Transport Protocols:

Real-time transport protocol (RTP) provides end-to-end delivery services of real-time audio and video. Whereas H.323 is used to transport data over IP–based networks, RTP is typically used to transport data via the user datagram protocol (UDP). RTP, together with UDP, provides transport-protocol functionality. RTP provides payload-type identification, sequence numbering, time-stamping, and delivery monitoring. UDP provides multiplexing and checksum services. RTP can also be used with other transport protocols.

Real-Time Transport Control Protocol:

Real-time transport control protocol (RTCP) is the counterpart of RTP that provides control services. The primary function of RTCP is to provide feedback on the quality of the data distribution. Other RTCP functions include carrying a transport-level identifier for an RTP source, called a canonical name, which is used by receivers to synchronize audio and video.

H.323 terminals must support the following:

- H.245 for exchanging terminal capabilities and creation of media channels
- H.225 for call signaling and call setup
- RAS for registration and other admission control with a gatekeeper
- RTP/RTCP for sequencing audio and video packets

H.323 terminals must also support the G.711 audio CODEC. Optional components in an H.323 terminal are video CODECs, T.120 data-conferencing protocols, and MCU capabilities.

(www.tec.org/online/tutorials/h323)

Chapter 3

Requirement Engineering

3. Requirement Engineering

It is the set of activities that leads to the production of requirements definitions and specifications of system.

System requirement include the following sub headings:

- Requirements Definitions
    - Functional Requirements
    - Non Functional Requirements
- Requirements Specifications.

3.1 Requirement Definition

Requirements definitions of MIS are, what services the system is expected to provide and the constraints under which it must operate. Some of them are functional and some are non-functional which are defined separately in the subsequent sections.

3.1.1 Functional Requirements

- Handle incoming calls
- Play greeting messages after receiving calls
- Voice Message Recording
- Notification Message Sending i.e. email and SMS
- Message Retrieval
- Manage Users

3.1.2 Non-Functional Requirements

- The system should be able to play greeting message provided by respective users.
- The system to be developed should have real time response. That is the system response time should be real time the voice and video delay should be minimized.

3.2 Requirement Specification

Requirement specification is detailed and precise description of the system requirements. The requirement specification is mainly divided into two categories. These are functional requirements specification and non-functional requirements specification. There are many techniques to document the functional requirements specification of the system to be developed. But I have used Structured Natural Language Specification:

R1: Handle Incoming Calls

Requirement Identifier: R1	Description Handle Incoming Calls
Function:	IM Server should handle incoming calls
Description:	The IM Server will be responsible for incoming calls. When it detects an incoming call, it will receive the call and will play the greeting message to the caller.
Inputs:	Incoming Call Request
Source:	Gatekeeper / Other H323 EndPoint
Outputs:	After receiving the incoming call, Greeting Message will be Played to caller.
Destination:	IM Server
Requires:	The system should be registered with Gatekeeper
Pre-conditions:	Gatekeeper should be in active state and IM Server is properly initialized to receive calls.
Post-conditions:	Play greeting message according to time.

Table 3.1: Handle Incoming calls

R2: Play Greeting Message

Requirement Identifier: R2	Description Play Greeting Message
Function:	Play Greeting Message to Caller
Description:	When the system will receive call properly, the system will play a greeting message to the caller so that he will be told that the resource person is not at home and he can record his/her message.
Inputs:	The "destination number" upon which the greeting message will be searched and played.
Source:	The information will be entered by user.
Outputs:	A voice file having the user's message
Destination:	Caller's output device like microphone or speaker
Requires:	Call should be received and destination number should be passed.
Pre-conditions:	Call is connected
Post-conditions:	Prompt to record voice message

Table 3.2: Play Greeting Messages

R3: Voice Message Recording

Requirement Identifier: R3	**Description** Voice Message Recording
Description:	After receiving call and playing a voice message to caller, caller should be asked to record his voice message so that the resource person will be informed. So the system will record caller's voice.
Inputs:	The voice signals sent by the caller
Source:	The network used
Outputs:	A voice file recorded by the caller
Destination:	It can be stored at Permanent storage for further use.
Requires:	It requires name and path of Network.
Pre-conditions:	Call should be received and greeting message is played
Post-conditions:	The notification should be sent to user

Table 3.3: Voice Message Recording

R4: Sending Notification

Requirement Identifier: R4	Description Sending Notification
Description:	After receiving call, greeting message will be played and Caller's message will be record for the callee. After recording the voice message, callee will be notified that his new message has arrived.
Inputs:	Email address
Source:	IM Server
Outputs:	User will be receive notification on his email account
Destination:	Users Email Server
Requires:	Voice message should properly be stored on Permanent storage.
Pre-conditions:	There will be no Firewall restriction on the network to send notifications
Post-conditions:	Callee retrieve the message from the E-mail account

Table3.4: send Notification

R5: Retrieving Voice Message

Requirement Identifier:	Description
R5	Retrieving Voice Message
Description:	When the voice message will be successfully stored then this voice message is also sent to email server as an attachment. Then the user will open the web interface from any location in the world and after log into the system he will be able to see his voice messages stored and then he can download message from there and can listen.
Inputs:	Identification number and password
Source:	The information will be entered by the user
Outputs:	User voice mail box will be displayed from where he can download his voice messages.
Destination:	User's PC
Requires:	Web Browser from where web site can be seen
Pre-conditions:	The User`s PC should be connected to internet and users is successfully log into the system
Post-conditions:	User can play the voice message using standard players.

Table 3.5: Retrieve voice messages

Chapter 4

System Analysis and Design

4. Analysis and Design

For analysis and design purposes the Unified Modeling Language (UML) is used. In UML, a system is represented using different "views" that describe the system from distinctly different perspective. Each view is defined by a set of diagrams. The following views are present in UML.

User Model View:

The Use-case is the modeling approach of choice for the user model view.

Structural model view:

Data and functionality are viewed from inside the system. That is, static structure (classes, objects and relationship) is modeled.

Behavioral Model:

This model represents the dynamic or behavioral aspects of the system. It also depicts the interactions or collaborations between various structural elements described in the user model and structural models view.

4.1 User Model View

4.1.1 Identification of Actors

Actors of this System are:

1) Caller
2) Callee (To whom call is made)

4.1.2 Identification of Use Cases

Following are the possible identified use cases of the system.

- Handle Incoming Calls
- Play Greetings
- Record Messages
- Compose Messages
- Send Notification
- Retrieve Messages

4.1.3 Use Case Description (Use-Case Template)

Here is the description of the few use cases identified above:

Use Case: Handle Incoming Calls

Use case:	**Handle Incoming Calls**
Actors	**Caller**
Trigger	When the Gatekeeper transfers call to the system
Flow of events	
1	The caller calls to the resource person
2	If the resource person does not attend call then call will be transferred to IM Server
3	IM Server will listen the incoming calls
4	IM Server will detect the incoming calls and receives for the further processing

Table 4.1 handle incoming calls

Use Case: Handle Incoming Calls

Use case:	**Play Greetings**
Actors	**Caller**
Trigger	IM Server listens the call
Flow of events	
1	IM Server will listen the call
2	Call will be accepted and call information parsed to get destination call number
3	IM Server will retrieve greeting message according to the time
4	Greeting message will be retrieved and will be played to caller

Table 4.2: Play Greetings

Use Case: Record Voice Message

Use case:	**Record Voice Message**
Actors	**Caller**
Trigger	Messaging Playing completed
Flow of events	
1	When the greeting message is played to caller, he is asked to record his voice message
2	Callers voice message will be stored on HDD
3	Sends message to system that message recording completed

Table 4.3: Record Voice Message

Use Case: Send Notification

Use case:	**Send Notification**
Actors	**Callee**
Trigger	Call record completed
Flow of events	
1	The message of the caller will be recorded
2	A notification message will be composed
3	A notification message will be sent to user so that he may be able to check his messages

Table 4.4: Send Notification

Use Case: Retrieve Message

Use case:	**Retrieve Messages**
Actors	**Callee**
Trigger	User request to retrieve messages
Flow of events	
1.	User will login to system through a web interface
2.	After authentication, user will be shown his voice message on the email server

Table 4.5: Retrieve Message

4.1.4 Use Case Diagram

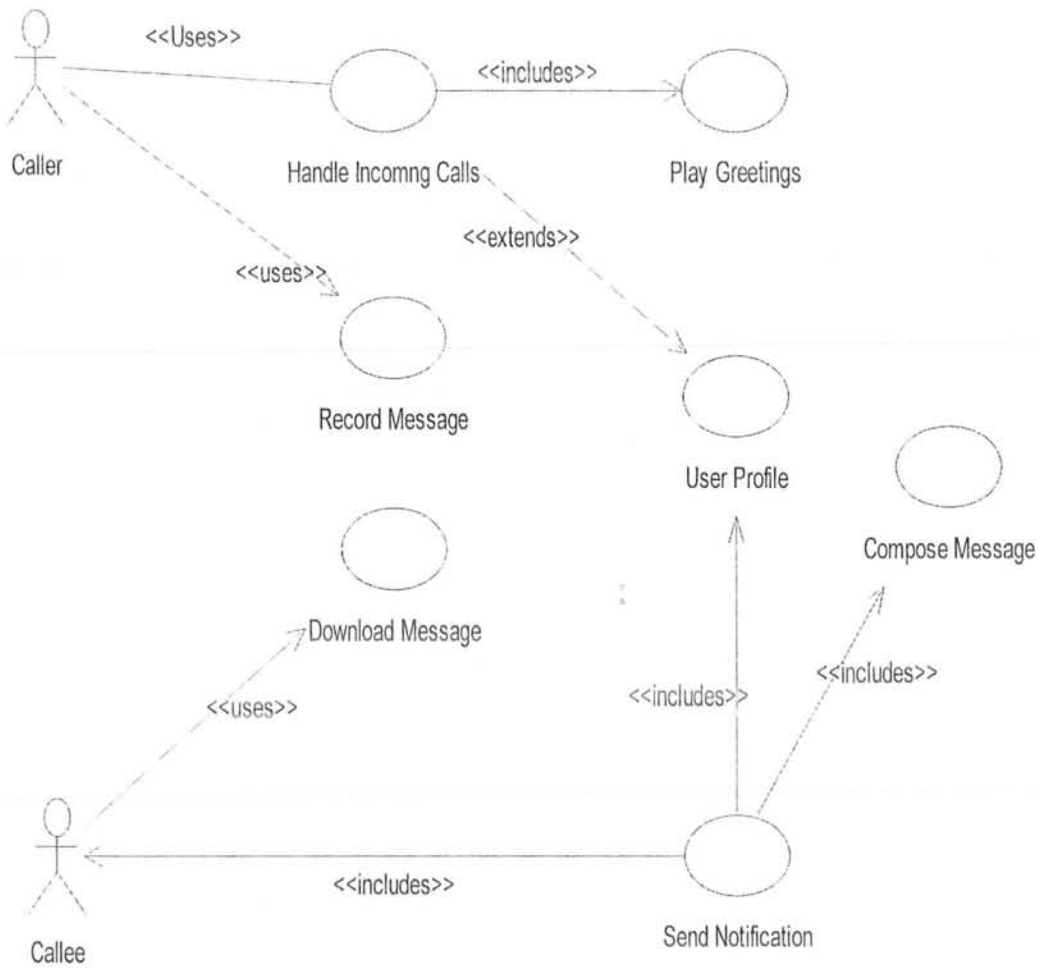


Figure 4.1 Use Case Diagram

Conclusuion

In this chapter, we have discussed about various use cases. All the principles we had followed in designing our application in order to have the clear understanding about the use cases of the application. This included use cas description and use case diagram.

4.2 Structural Model View:

4.2.1 Identification of Classes

- MediaAgent
- AnsMachine
- IMH323Endpoint
- IMH323Connection
- PCM_OGMChannel
- PCM_RecordFile
- IMAgent
- NotificationManager

4.2.2 Class Diagram

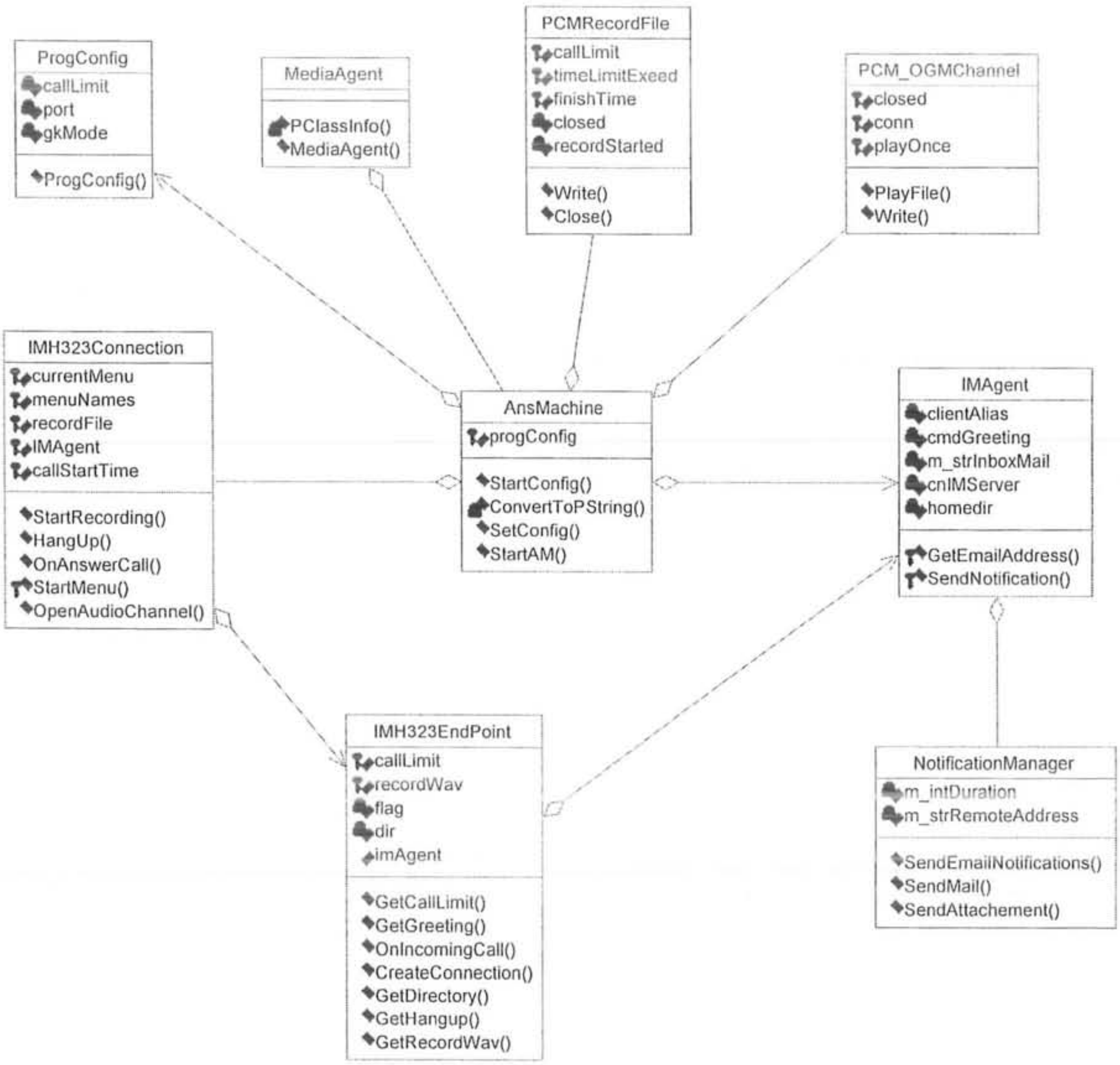


Figure 4.2 Class Diagram

4.2.3 Classes Description

Class Name: Media Agent		
Class Type: Property Class		
Responsibilities	**Collaboration**	
1. This is entry point of the application 2. Gets and sets the application information including version info, operating system info etc. 3. Starts the Answering Machine thread	AnsMachine	MediaAgent PClassInfo() MediaAgent()

Table 4.6: Media Agent class

Class Name: AnsMachine		
Class Type: Property Class		
Responsibilities	**Collaboration**	
1. Configures the application settings 2. Registers the application with the Gatekeeper 3. Starts the answering machine thread to listen the incoming calls	IMH323EndPoint	AnsMachine progConfig StartConfig() ConvertToPString() SetConfig() StartAM()

Table 4.6: AnsMachine Class

Class Name: IMH323EndPoint		
Class Type: Property Class		
Responsibilities	**Collaboration**	
1. Creates the H323EndPoint 2. Initialize the Endpoint e.g. sets the capabilities of the Endpoint 3. Detects the incoming calls 4. Gets the greeting file path on the base of incoming call info	IMH323Connection IMAgent	

Table 4.7: IMH323EndPoint Class

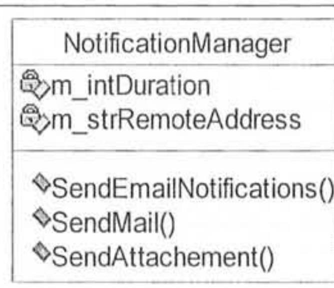
Class Name: NotificationManager		
Class Type: Property Class		
Responsibilities	**Collaboration**	
1. Responsible to send notification messages to users 2. Compose email messages 3. Sends email notification and email attachment of recorded voice message to mail server		

Table 4.8: NotificationManager Class

Class Name: IMH323Connection		
Class Type: Property Class		
Responsibilities	**Collaboration**	
1. It is called by the IMH323EndPoint. 2. Opens and closes the connection with the caller 3. Receives the call 4. Opens the logical channel to carry media over the network 5. Opens and initialize the audio channel for audio data transfer 6. Adjusts the call start, end timings 7. Initialize the file playing and file recording channels 8. Logs the call details in the file	IMAgent IMH323EndPoint PCM_OGMChannel PCM_RecordFile	IMH323Connection currentMenu menuNames recordFile IMAgent callStartTime StartRecording() HangUp() OnAnswerCall() StartMenu() OpenAudioChannel()

Table 4.9: IMH323ConnctionPoint Class

Class Name: PCM_OGMChannel		
Class Type: Property Class		
Responsibilities	**Collaboration**	
Plays Greetings message to the caller	PCM_RecordFile IMH323Connection	PCM_OGMChannel closed conn playOnce PlayFile() Write()

Table 4.10: PCM_OGMChanel Class

Class Name: PCM_RecordFile		
Class Type: Property Class		
Responsibilities	**Collaboration**	
Records the voice message of the caller and stores the recorded voice message on the HDD	IMH323Connection	PCMRecordFile callLimit timeLimitExeed finishTime closed recordStarted Write() Close()

Table 4.11: PCM_RecordFile Class

Class Name: IMAgent		
Class Type: Property Class		
Responsibilities	**Collaboration**	
• This class represents the User Manager Agent. • It manages the email address • Initiates the notification process	NotificationManager	IMAgent clientAlias cmdGreeting m_strInboxMail cnIMServer homedir GetEmailAddress() SendNotification()

Table 4.12: IMAgent Class

4.3 Behavioral Model

Here below three important behaviors are explained through diagrams

- Send Notification Sequence
- Play Greeting Sequence
- Handle Call Sequence

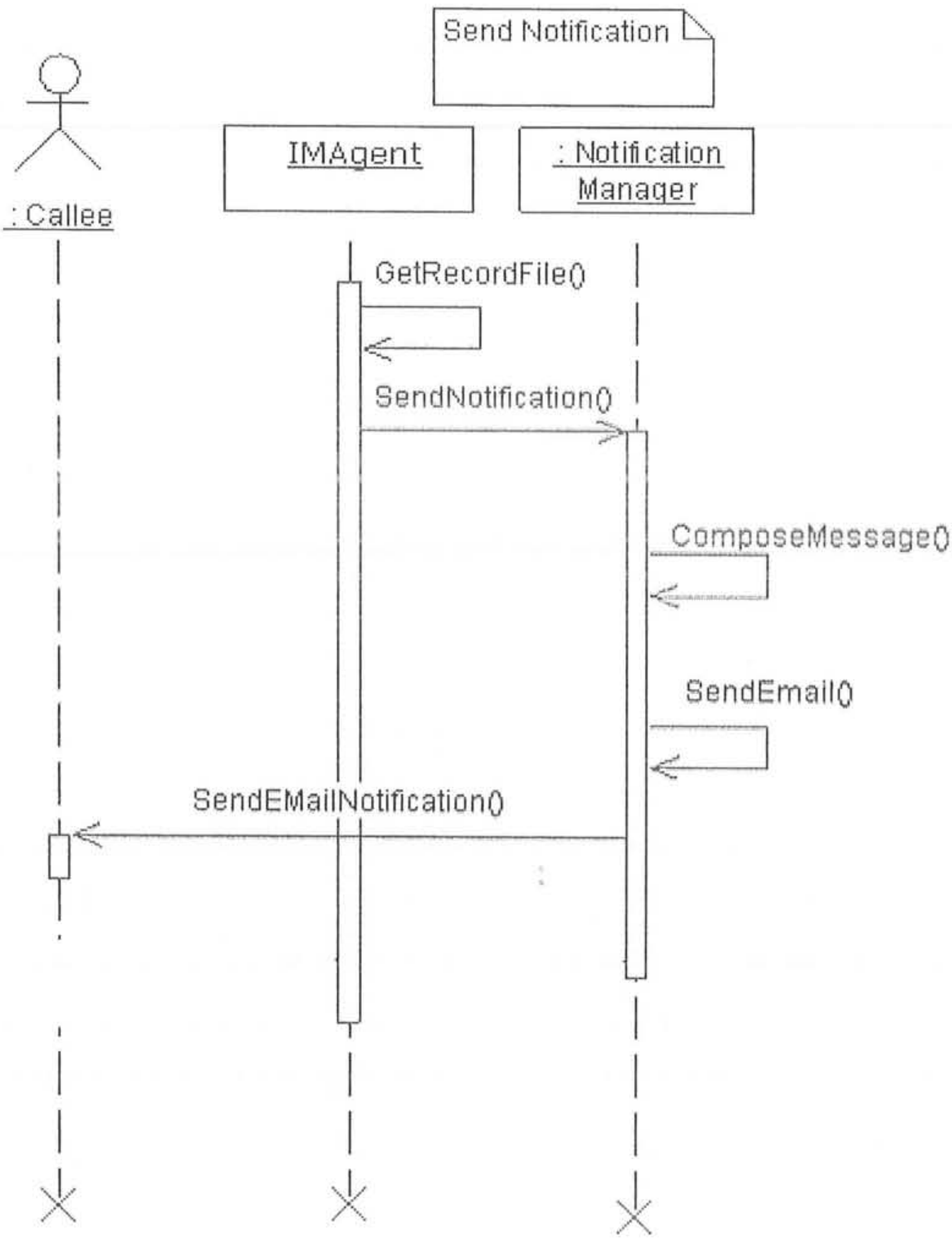


Figure 4.3: Send Notification Sequence

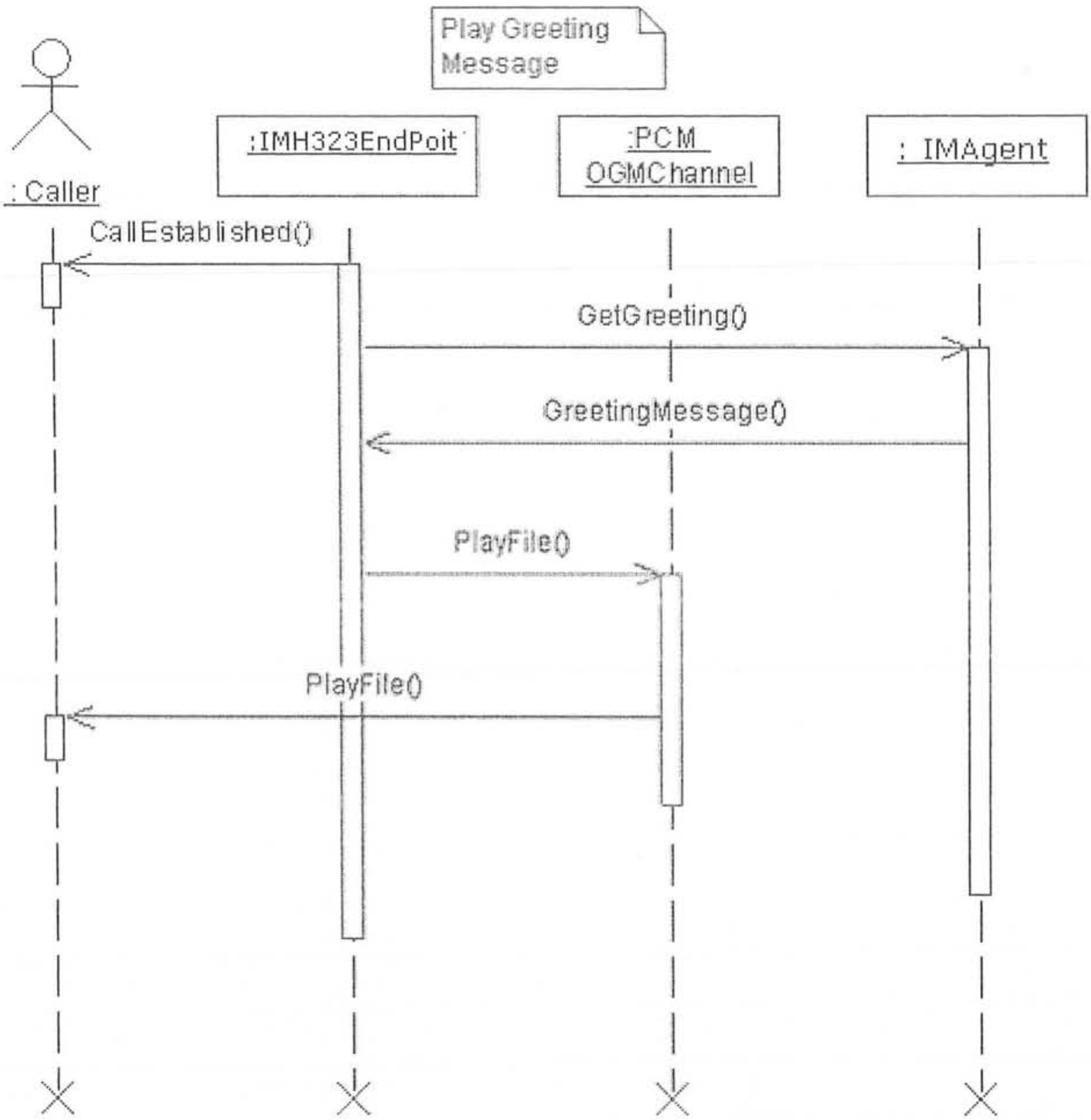


Figure 4.4 Play Greeting Sequence

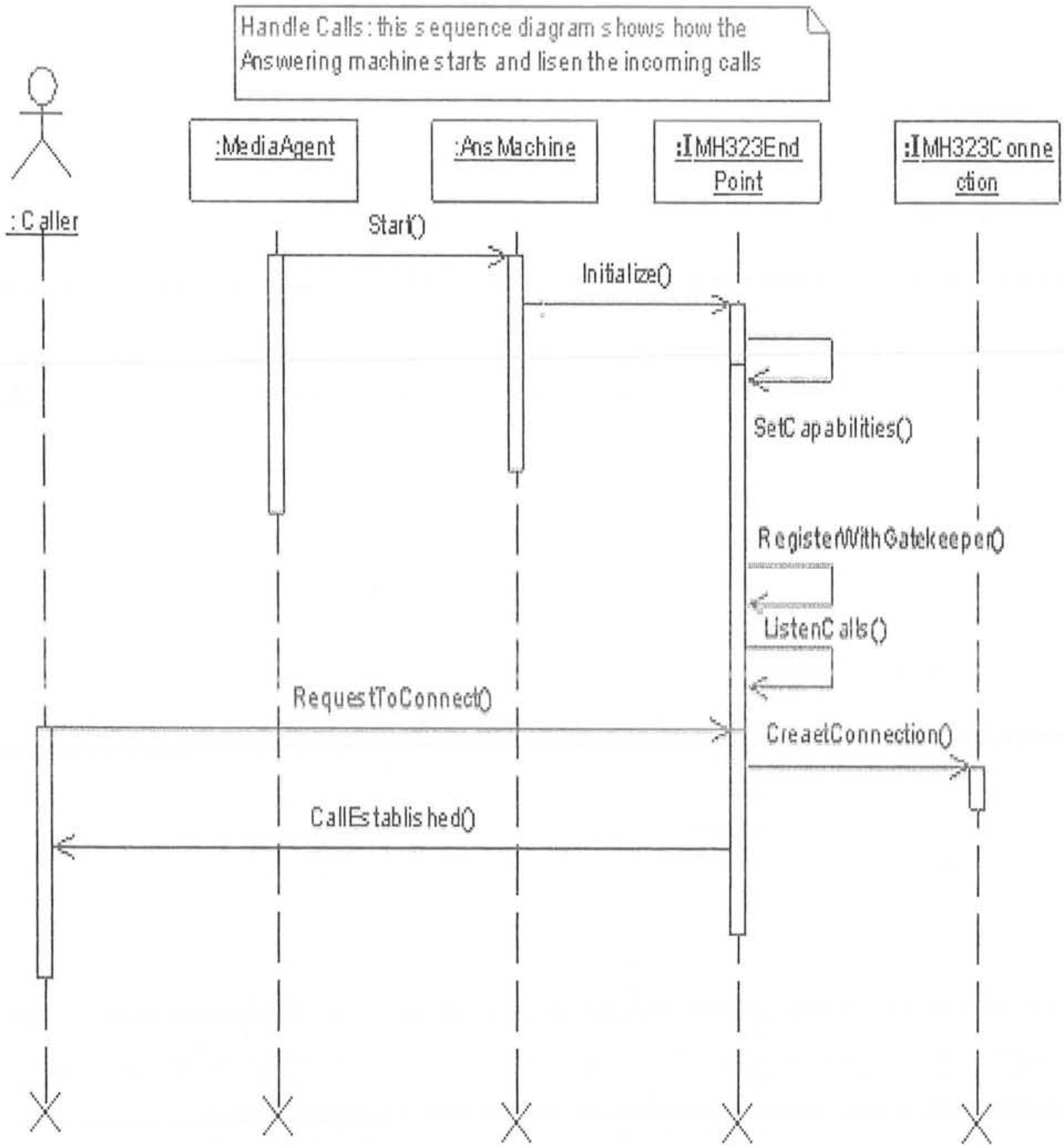
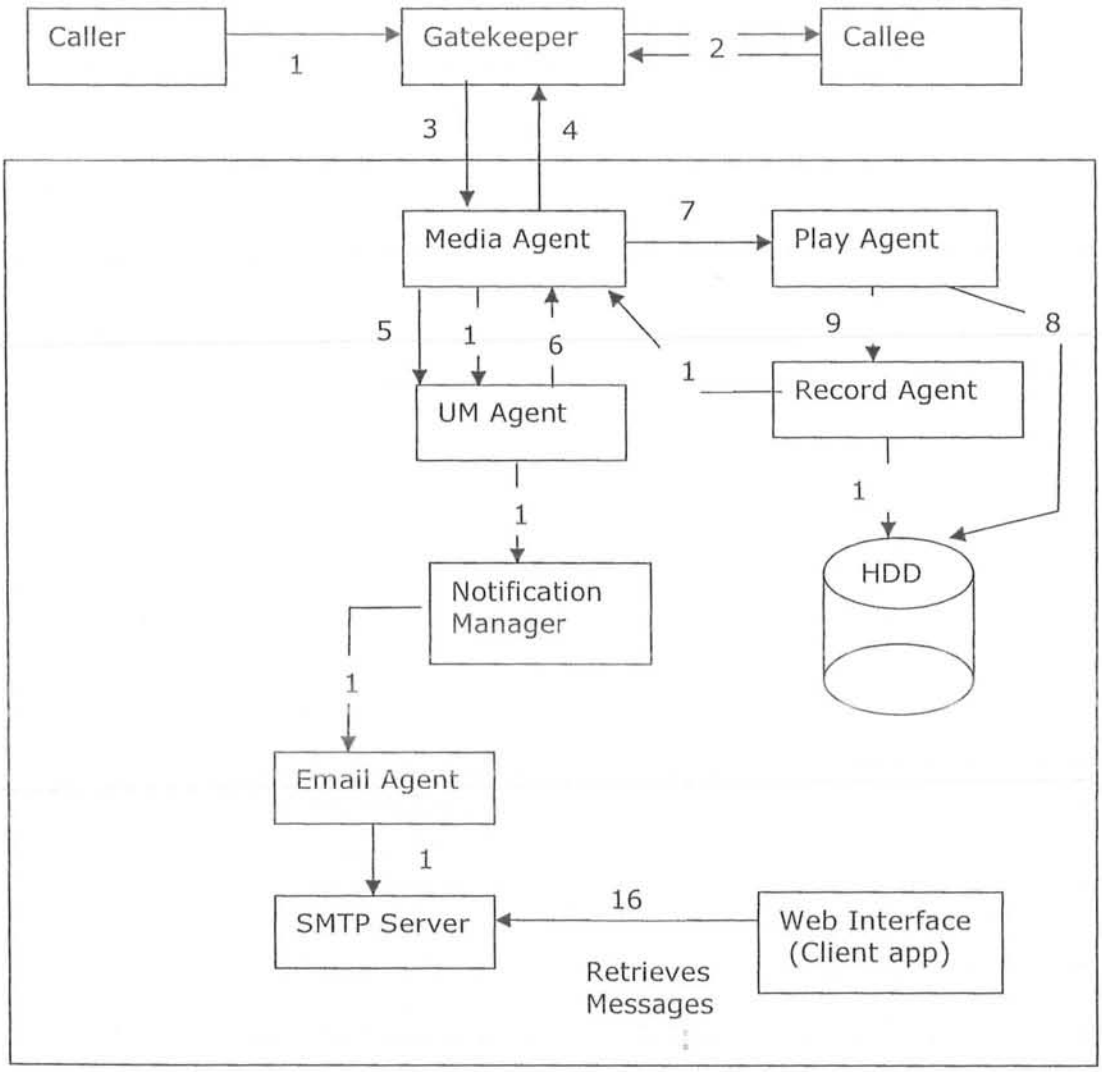


Figure 4.5: Handle Call Sequence

4.4. IM Architectural Model:



Model Description:

1. Caller calls to callee via Gatekeeper
2. Callee does not attends call
3. Gatekeeper then transfers call to IM System's Media Agent Module
4. Media Agent receives call on the behalf of the callee
5. On the base of the call info Media Agent communicates with UM Agent to get the personalized greeting message
6. After getting the correct greeting message info it passes to Media Agent
7. Media Agent then initiates the File Play Agent to play the greeting message
8. Play Agent retrieves that greeting file from Permanent Storage device and play to caller
9. After playing the message it initiates the Record Agent to record the caller's message
10. Record Agent then records the caller's voice message as a file on the permanent storage device
11. After storing the voice message successfully it notifies the media agent that file has been successfully recorded and also provides recorded file info to media agent
12. Media Agent transfer that info to UM Agent to send notification to the Callee
13. IM Agent initiates the Notification Manager
14. Notification Manager initiates the Email Manager to send email notification to Callee
15. Email Manager send email notification to Callee
16. Now the Callee has the facility to retrieve his voice messages from the email server through a web interface

4.5 Design Decisions:

- Use of 'wav' file to save the voice message.
  - The voice file are saved in default 'wav' format on the system.
  - It is native audio file format of Microsoft Windows. It has a better sound quality and can be easily attached to any e-mail.
- Call placed using H323 Protocol.
  - It is an Organizational Requirement.
  - H.323 is more flexible than SIP. Although SIP is a little bit simple then H323 but not mature as H.323 and could not solve more complex problems.
  - H.323 is well defined as compared to SIP.
  - H.323 is modular because it covers all call signaling, user location, and registration.
  - It supports both unicast and multicast communication.

Chapter 5

System Implementation

5. Implementation

5.1 Platform Selection

Platform selection in the software implementation phase is very critical .When choosing an operating System you must ensure that all those features that your software requires are supported by operating system. The selection of Platform should ensure that it will not create any Copy Rights problems when software is deployed.

The Platform I have selected for development is Microsoft® Windows XP.The reason for choosing this operating system is that it provides the feature and capabilities that are required during the software development phase. Also the SDK documentation worked with, suggests that Windows® XP is good choice as for the Operating system selection is concerned.

The built software will be able to run on the Microsoft Windows based operating system i.e., Windows® 98, Windows® 2000 and Windows XP.

5.2 Tools and SDK Selection

The Tool selected for development of this project is Visual C++. NET and SDK selected for the development is PWLib and OpenH323 that implements H323 protocol.

There are very few reasons for choosing OpenH323 which are:

- OpenH323 is an open source library that implements H323 protocol.
- OpenH323 is completely implements features of H323 protocol.
- It is platform independent.
- OpenH323 is developed using C++ and it is very much compatible with VC++.NET.
- PWLib is used for its rich data structure and availability of low level APIs for voice communication.

5.3 Code Documentation Standards

Code of project has been documented well with coding standard as told by external project in charge. These standards are described below

- Each Class name begin with capital letter e.g IMAgent class.
- Fully object oriented design is followed in implementation of this system.
- A general Design structure is used for the system, this design generalization will enable future enhancements, and other types of network interfaces can be easily incorporated in current system without much difficulty.

- Each Class is well documented in form of class documentation. Documentation of these classes is provided in the appendices.

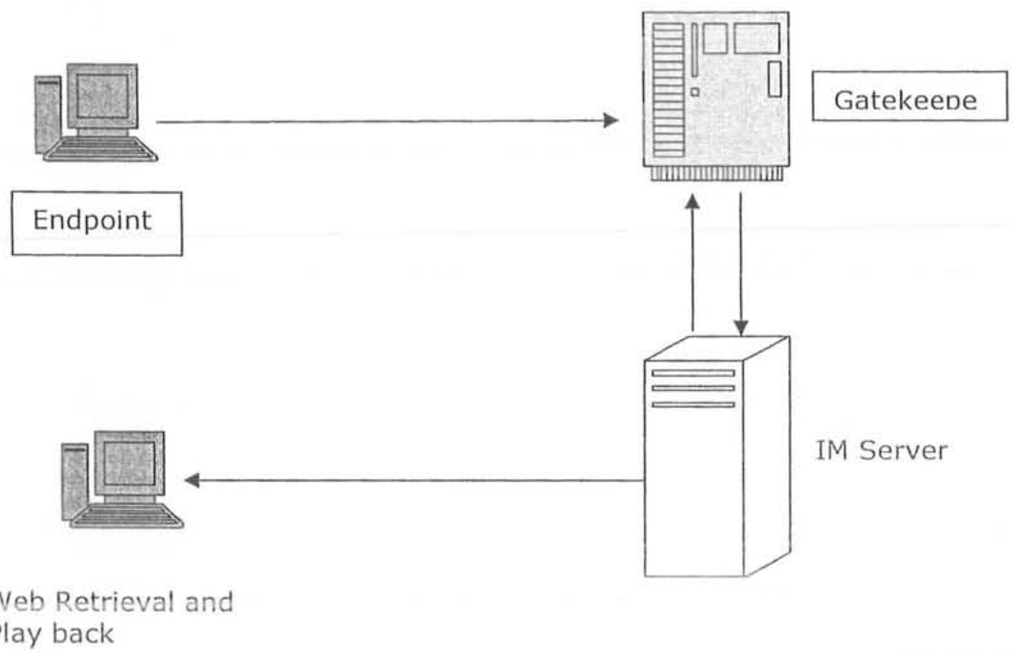
Implementation Code for Recording voice message:

```
BOOL PCM_RecordFile::Write(const void * buf, PINDEX len)
{
// Wait for the mutex, and Signal it at the end of this function
        PWaitAndSignal mutex(pcmrecordMutex);

// If the record file has been closed, or if the time limit has
// been exceeded, then return immediately.
        if (closed || timeLimitExceeded)
              return FALSE;
        if (!recordStarted)
        {
              DelayFrame(len);
              return TRUE;
        }
        PTime now;
        if ((callLimit != 0) && (now >= finishTime))
        {
              PTRACE(1, "Terminating call due to timeout");
              conn.ClearCall();
              timeLimitExceeded = TRUE;
              return TRUE;
        }
        DelayFrame(len);
        dataWritten = TRUE;
        return WriteFrame(buf, len);
}
```

5.4 Deployment Diagram

A deployment diagram shows the allocation of processes to processors in the physical design of a system. A deployment diagram may represent all or part of the process architecture of a system.



5.5 Implementation Decisions

➢ **Microsoft Visual C++.NET**

Microsoft Visual C++ (is also known as MSVC) is an Integrated Environment Development IDE product for the C, C++ and CLI programming languages engineered by Microsoft. It has tools for developing and debugging C++ code. Visual C++ boasts features are such as syntax highlighting and IntelliSense and advanced debugging functionality.[http//en.wikipedia.org/wiki]

- Use of validation controls which are provided by Visual Studio
- Provide the facility of Attachments of voice data with E-mail

➢ **Use of different H323 Libraries as PWLIB and OpenH323.**

The H.323 standard is a cornerstone technology for the transmission of real-time audio, video, and data communications over packet-based networks. It specifies the components, protocols, and procedures providing multimedia communication over packet-based networks.

- H323 is well tested.
- H323 is simple and matured than the SIP.

Chapter 6

System Testing

6. Testing

6.1 Software Testing Introduction

Simply stated, quality is very important. It is much easier to explain to a customer why there is a missing feature than to explain to a customer why the product lacks quality. A customer satisfied with the quality of a product will remain loyal and wait for new functionality in the next version. Quality is a distinguishing attribute of a system indicating the degree of excellence. The importance of software testing can not be overemphasized. Once the source code has been generated, software must be tested to allow errors to be identified and removed before the delivery of software. While it is not possible to remove the every error in large software package, our goal is to remove as many as possible in the early software development cycle. [PRE05]

6.2 Software Testing Objectives

Testing of the application is done to achieve following objectives.

- Execution of the program is done with intent to find the errors in the program.
- Test Cases are designed so that these have high probability of finding an as-yet-undiscovered error.

6.3 Selected Testing Strategy

Although there are plenty of testing strategies which can be applied during the software testing process but each of this approach has its own objective to achieve. As for the testing of Integrated Messaging Server is concerned the preferred testing policy is:

- Black Box Testing

6.3.1 Black Box Testing

Black testing is carried out on the basis of specification of software component or System which is to be put under testing procedure. This type of testing measures behavior of system by studying the supplied input and obtained results. It is same as testing the mathematical functions therefore some time we also call the black box testing as "Functional Testing".

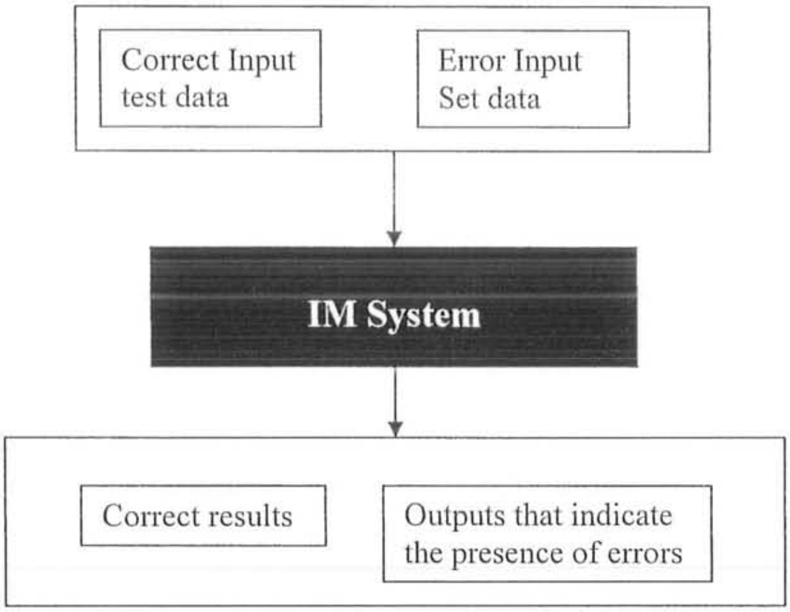


Fig 7.1. System is being treated as black box

6.4 Test Cases

Test cases are designed so that these maps with following functional requirements of the system.

R1- Handle incoming calls

R2- Play greeting messages after receiving calls

R3- Voice Message Recording

R4- Notification Message Sending i.e. email

R5- Download Messages

<u>Test Case for "Handle Incoming Call"</u>

Test Case # 1

Test Case Identifier

Handle Calls

Test Location

D:\Test Cases\Test1.wav

Features to be Tested

Place the call to the User

Connection is created between caller and IM Server

It can handle Incoming Calls

Feature Pass/Fail Criteria

PASS: The call is connected to the system.

FAIL: No Greeting message is played to the caller

Means of Control

DRIVER: AnsMAchine is the Driver Calss

STUB: PCM_OGMChannel, IMH323Connection

Data

INPUT:

call: Primary Sound Capture Driver

OUTPUT:

A connection is created between caller and IM Serevr.

Test Procedure

Steps:

Place the call to the IM Server through any other system.

Special Req:

Filtration is applied.

Validation controls (error providing controls are used).

Test Case for "Playing Greeting Message after receiving call"

Test Case # 2

Test Case Identifier

PlayGreetingMessage

Test Location

D:\Test Cases\Test2.wav

Features to be Tested

Greeting message is played to the caller

Feature Pass/Fail Criteria

PASS: The call is connected to the system and it plays a greeting message to the caller.

FAIL: No Greeting message is played to the caller

Means of Control

DRIVER: IMH323Connection is the Driver Calss

STUB: IMH323EndPoint

Data

INPUT:

call: Primary Sound Capture Driver

OUTPUT:

A greeting message is played to the user.

Test Procedure

Steps:

Place the call to the IM Server through any other system.

On proper call the Server will play the greeting message to the caller.

Special Req:

Filtration is applied.

Validation controls (error providing controls are used).

Test Case for "Recording Message"

Test Case # 3

Test Case Identifier

RecordMessage

Test Location

D:\Test Cases\Test3.wav

Features to be Tested

Voice Message recorded.

Unique name given to the voice message.

Recorded in "wav" format.

Feature Pass/Fail Criteria

PASS: The voice message is recorded in the Incoming calls folder

FAIL: No voice message is saved at the location.

Means of Control

DRIVER: AnsMachine is the Dreiver class

STUB: PCM_RecordFile

Data

INPUT:

call: Primary Sound Capture Driver

OUTPUT:

A wav file is created composed of the voice message from the caller.

Test Procedure

Steps:

After receiving Greeting message start to record your voice message.

Special Req:

Filtration is applied.

Validation controls (error providing controls are used).

Test Case for "Sending Notification"

Test Case # 4

Test Case Identifier

NotificationSend

Test Location

D:\Test Cases\Test4.wav

Features to be Tested

Get E-mail Address of the Callee.

Compose a message for the callee.

Attach the Voice message to the simple message composed.

Send on the callee e-mail address.

Feature Pass/Fail Criteria

PASS: E-mail notification about the message is sent to the callee

FAIL: No message is sent to the callee

Means of Control

DRIVER: IMAgent is the Dreiver class

STUB: NotificationManager

Data

INPUT:

Voice message of the caller saved in the 'Incoming Calls' folder.

OUTPUT:

A message is composed with an attachment of voice message and send to E-mail account (s_iqbal4u@yahoo.com)

Test Procedure

Steps:

Place a call and record your voice message

Special Req:

Filtration is applied.

Validation controls (error providing controls are used).

Test Case for "Downloading voice message"

Test Case # 4

Test Case Identifier

Downloadmessage

Test Location

D:\Test Cases\Test5.wav

Features to be Tested

Voice message is delivered successfully

Downloaded

Feature Pass/Fail Criteria

PASS: Voice message is downloaded from the e-Mail account.

FAIL: Voice message Notification is not present in mail account.

Means of Control

DRIVER: IMAgent is the Dreiver class

STUB: NotificationManager

Data

INPUT:

E-mail address and password

OUTPUT:

The voice message is downloaded as a attachment from the web.

Test Procedure

Steps:

Open the e-mail account. (s_iqbal4u@yahoo.com)

Check the Inbox.

Special Req:

Filtration is applied.

Validation controls (error providing controls are used).

References

REFRENCES:

[SOM00] Somerville, I., Software Engineering, *Publisher*, 6^{th} Edition 2000

[PRE01] Pressman, R.S., Software Engineering A Practitioner's Approach, *McGraw Hills*, 2001

[ALF91] *Alford, M.W.* A Requirement Engineering Methodology for Real Time System, IEEE *Tran on Software Engineering* SE-3 (1) 60-90[123], 1991

[DAV93] *Davis*, A.M., Software Requirements: Analysis and Specification. Englewood Cliff NJ: *Prentice Hall*

[EAS93] *Easter Brook, S.* Domain Modeling with Hierarchies of alternative Viewpoint complete

[GIL98] Gil Held, Voice and Data Internetworking, McGraw-Hill

[HOO98] Hoofer, G., System Analysis and Designing, *Prentice Hall*

Webliography

Webliography

http://www.necunifiedsolutions.com/main/Products/ProProduct.asp?solid=5&prodid=64

http://www.activevoice.com.au/

http://www.fokus.gmd.de/research/cc/glone/projects/ipt/

http://www.imtc.org/h323.htm

http://www.packetizer.com/h323.htm

Glossary

Glossary

IM Server	Integrated Messaging Server
H.323	It is ITU-T specification protocol used on packet switched network for communication purposes.
MAN	Metropolitan Area Network
MCUs	Multipoint Control Unit
MGCP	(Media Gateway protocol)
PSTN	Public Switched Telephone Network
PBX	Private Branch Exchange
QoS	Quality of Service
RAS	Registration, Admission, Status
RTP	Real-time transfer protocol
RTCP	real-time control protocol
SIP	Session Initiation Protocol, a signaling protocol by IETF.
TCP	Transport control Protocol
VoIP	Voice over Internet Protocol
WAN	Wide Area Network