INTRANET COMMUNICATION USING VoIP

Project Report

Submitted in partial fulfillment of the
Requirement for the award of the degree of the

BACHELOR OF ENGINEERING IN INFORMATION TECHNOLOGY

OF

BHARATHIAR UNIVERSITY, COIMBATORE.

Submitted by

B. Gomathi
0027S0074

P. A. Rukmini
0027S0100

Under the able guidance of

Mrs. N.Chitra Devi M.E.
Senior Lecturer

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING,

KUMARAGURU COLLEGE OF TECHNOLOGY,
(Affiliated to Bharathiar University, Coimbatore)
COIMBATORE – 641006.

MARCH 2004.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY
(Affiliated to Bharathiar University, Coimbatore)

TÜV
MANAGEMENT SERVICE
ISO 9001

CERTIFICATE

This is to certify that the project entitled

INTRANET COMMUINCATION USING VoIP

is done by

B.Gomathi
0027S0074

P.A.Rukmini
0027S0100

And submitted in partial fulfillment of the
Requirement for the award of the degree of the

Bachelor of Engineering in Information Technology
of
Bharathiar University, Coimbatore.

Professor & Head Of the Department
(Dr.S.THANGASAMY)

Project Guide
(Mrs. N. CHITRA DEVI)

Certified that the candidates were examined by us in the project work
Viva voce examination held on 27.3.2004.

27/3

Internal Examiner

External Examiner

BHARAT SANCHAR NIGAM LIMITED

O/O PRINCIPAL GENERAL MANAGER, COIMBATORE-43.

TO WHOMSOEVER IT MAY CONCERN

It is certified that Miss Gomathi B and Miss Rukmini.P.A, Final year students of B.E. Information Technology of Kumaraguru College of Technology have undertaken and successfully completed and demonstrated project viz. **"Intranet Communication using VoIP"** for Bharat Sanchar Nigam Limited, Coimbatore.

During the project period, their performance, conduct and character were found to be good. I wish them all success in their future endeavors.

Date: 24.03.04

For Bharat Sanchar Nigam Limited,

DE (COMPUTERS)

मंडल अभियंता (कंप्यूटर)
भा.सं.नि.लि., दूरसंचार केन्द्र, कोयम्बत्तूर
Divisional Engineer (Computer)
B.S.N.L. Central Tele.Phone Exchange
COIMBATORE 641 018

DECLARATION

Declaration

We,

B.Gomathi	**0027S0074**
P.A.Rukmini	**0027S0100**

declare that the project entitled **"Intranet communication using VoIP"**, is done by us and to the best of our knowledge, a similar work has not been submitted earlier to the Bharathiar University or any other institution, for fulfillment of the requirement of the course study.

This project report is submitted as the partial fulfillment of the requirement for all awards of the degree of Bachelor of Engineering in Information Technology of Bharathiar University.

Place: Coimbatore.

[B.Gomathi]

Date: 24·03·2004

[P.A.Rukmini]

Project Guided by

Mrs. N.Chitra Devi, M. E.
Senior Lecturer, Department of Computer Science and Engineering,
Kumaraguru College of Technology,
Coimbatore – 641006.

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

The exhilaration achieved upon the successful completion of any task should be definitely shared with the people behind the venture. This project is an amalgamation of study and experience of many people without whose help this project would not have taken shape.

At the outset, we take this opportunity to thank the management of our college for having provided us excellent facilities to work with. We express our deep gratitude from the bottom of our hearts to our Principal **Dr.K.K.Padmanabhan** for ushering us in the path of triumph.

We are always thankful to our beloved Professor and the Head Of the Department, **Dr.S.Thangasamy** whose consistent support and enthusiastic involvement helped us a great deal.

We are greatly indebted to our beloved guide **Mrs. N.Chithra Devi M.E.**, Senior Lecturer, Department of Information Technology for her excellent guidance and timely support during the course of this project. As a token of our esteem and gratitude, we would like to honour her for her assistance towards this cause.

We also thank our Project Coordinator **Mrs. S. Devaki M.S.**, and our beloved class advisor **Ms. P. Sudha B.E.,** for their invaluable assistance.

We also thank Bharat Sanchar Nigam Limited, Coimbatore for giving us an opportunity to do our project work in their organization.

We also feel hereby record our deep sense of gratitude to all the staff and lab technicians in the Department of Information Technology.

We feel proud to pay our respectful thanks to our **Parents** for their enthusiasm and encouragement and also we thank our **friends** who have associated themselves in bringing out this project successfully.

SYNOPSIS

SYNOPSIS

Intranet Communication using VoIP is based on Client-Server architecture. Modern day offices are quite large and in most cases are multistoried. Employees and managers have offices at different locations in the same building. Communication among them becomes a problem.

This software is developed in Visual C++ 6.0. It is simple and easy to use and avoids brain wrecking protocols or ambiguous working methods. The project serves the communication needs of the office.

An online full duplex voice communication is the most natural way of interaction and is a powerful means of communication. Collection, storage, distribution and processing of information are a day to day activity in any enterprise. The successful assimilation and dissemination of information is the major factor deciding the success or failure of a business institution. In such a scenario the need for a cheap, reliable, simple communication software is generally felt. Though not a pioneer in the communication field, Intranet has proved itself to be leader.

CONTENTS

CONTENTS

INTRODUCTION

1. INTRODUCTION

1.1. Existing System and Limitations:

The existing system available in industry can perform voice chat only by using Internet. Using Internet leads to delays which have to be minimized when it comes to real time communication. Data transmission is also not assured to be secure. The existing VoIP technology can be used only for making international calls. The major problem identified is how to communicate efficiently and cost-effectively within an organization. If intranet voice chat facility is implemented, it will be boon to most of the commercial organizations.

1.2. Proposed system and Advantages:

Of late, public networks are moving from circuit-switched networks to packet switched networks. This will cause telecommunications networks and data communication networks, which is mainly IP based, to converge to same networks. This will cause that voice calls are transmitted over IP based networks, as well. Voice over IP means that voice calls are transmitted over IP network rather than over the familiar public-switched telephone network.

The proposed system provides a cheap and efficient means of communication. Though the computers connected to each other through cables or wires can view each other's files, directories etc., but some software has to be installed that would provide a means for the users to communicate with each other.

The proposed system is christened as "Intranet Communication using VoIP". It seeks to provide voice communication to cater to the needs of all executives and managers in the organization. In some situations talking to one another is more effective than sending textual messages. Without using phone or mobile cell phones which can be easily tapped, the personals require to transfer sensitive information. Voice chatting system caters to this need. It is an online communication system that allows users to talk to each other using the multimedia kit. A server program runs in all the systems and users can communicate with each other by running client programs on their systems. The proposed system meets the communication needs of the organization. It removes the communication barrier which had been a major deterrent to the growth of the organization.

SYSTEM REQUIREMENTS

2. SYSTEM REQUIREMENTS

2.1. PRODUCT DEFINITION:

Modern day offices are quite large and in most cases are multi- storied. Employees and managers have offices at different locations in the same building. Communication among them becomes a problem. Conventional communication leads to wastage of time. Corporate offices need some efficient method of communication. Intranet communication software provides the answers to all these problems.

Voice chatting is an online program which connects clients. A server program runs in `Listen` mode. Client program run in `Connect` mode. A client can call another client and talk to him using headphones and mike sets. A successful connection will involve client A sending a message to client B requesting for voice chat. On acceptance, client B sends an acknowledgement. Then client A comes to know that no other client is currently chatting to client B and connects to client B. Voice chatting is the most natural way of interaction and is a powerful means of communication.

2.2. PROJECT PLAN:

It is the initial stage of project development activity. The first step is to understand the user requirements within the framework of the objective and the environment in which the system is to be installed. Consideration is given to the user resources and finance. After requirement analysis a clear picture of the chatting system became visible. Different issues were considered:

The major requirements that became visible are:

a) There has to be a server part.

b) There has to be a client part.

c) Server must be able to accept connection request from the client.

d) Enabling full-duplex voice communication.

SOFTWARE REQUIREMENTS SPECIFICATION

3. SOFTWARE REQUIREMENTS SPECIFICATION

3.1 INTRODUCTION:

3.1.1 Purpose:

The ability to offer cost optimized telephony services by bypassing the circuit-switched networks enables price discrimination as well as entirely new business opportunities for challenging ISP. Our project **"Intranet communication using VoIP"** is aimed at exploiting this ability.

3.1.2 Scope:

Proposed project is helpful for the employees in an organization to communicate with each other through their computers with out the use of internet regardless of which LAN their PC is connected. Any commercial organization can incorporate this project to achieve voice communication free of cost.

3.1.3 Definitions, Acronyms, Abbreviations:

IP : Internet Protocol.

VoIP : Voice over Internet Protocol. This is a technology for transmitting voice over IP network.

ISP	:Internet Service Provider
LAN	: Local Area Network
PBX	: Private Branch Exchange
PSTN	:Public switched telephone network.

3.2. GENERAL DESCRIPTION:

3.2.1 Product Perspective:

The existing technology uses Internet as the communication medium which has certain disadvantages due to delay incurred, packet loss etc., Our project overcomes these disadvantages to a certain extent since it doesn't use Internet. If implemented, the additional overhead of using telephones can be avoided.

3.2.2 Product Functions:

Our project aims at implementing full-duplex voice communication between two PC's. An address book for user reference is also maintained.

3.2.3 User Characteristics:

The user need not have any additional knowledge if he has a general idea of working with a computer.

3.2.4 General Constraints:

If two systems need to have interactive communication it is necessary that the systems should be powered on.

3.3. SPECIFIC REQUIREMENTS:

3.3.1 Functional Requirements:

The basic function of this project is to provide cost free telephone service for an organization. This product also eliminates the need of having internet connection. Finally it also provides a user interface for making calls. An address book consisting of the users and their respective systems IP addresses along with their other personal information is also maintained.

3.3.2 List of Inputs:

a) The IP address of the callee's system.
b) Actual voice message.

3.3.3 Information processing requirement:

The voice message is read by the sound card using appropriate multimedia kit and the message is stored in buffer. This message is appended with source and destination addresses to form IP packets which are directly sent to the destination system.

3.3.4 Development and Operating Environment:

The development environment gives the minimum hardware and software requirements.

Hardware Specification

a) Processor	Pentium III or Pentium IV
b) RAM	64 MB
c) Cache	128MB
d) Hard Disk	10 GB
e) Floppy Drive	1.44 FDD
f) Monitor	14" Monitor

Software Specification

a) Operating System	Windows 98, Windows XP.
b) Platform	Visual C++ (Win32 API)

3.4. PERFORMANCE REQUIREMENT:

3.4.1 Memory:

The memory required to install dialer software is sufficient in a client machine.

3.4.2 Speed:

The client and the server code should run the processor at their maximum speed to achieve real time communication.

3.5. DESIGN CONSTRANINTS:

3.5.1 Hardware Limitations:

Our project compulsorily needs multimedia kit ie., speakers and microphone without which this project would be of no use.

3.5.2 User Interfaces, Screen Formats:

The user interface is to be created such that the caller doesn't have any problems in making calls. There should be provisions for storing the IP addresses in address book.

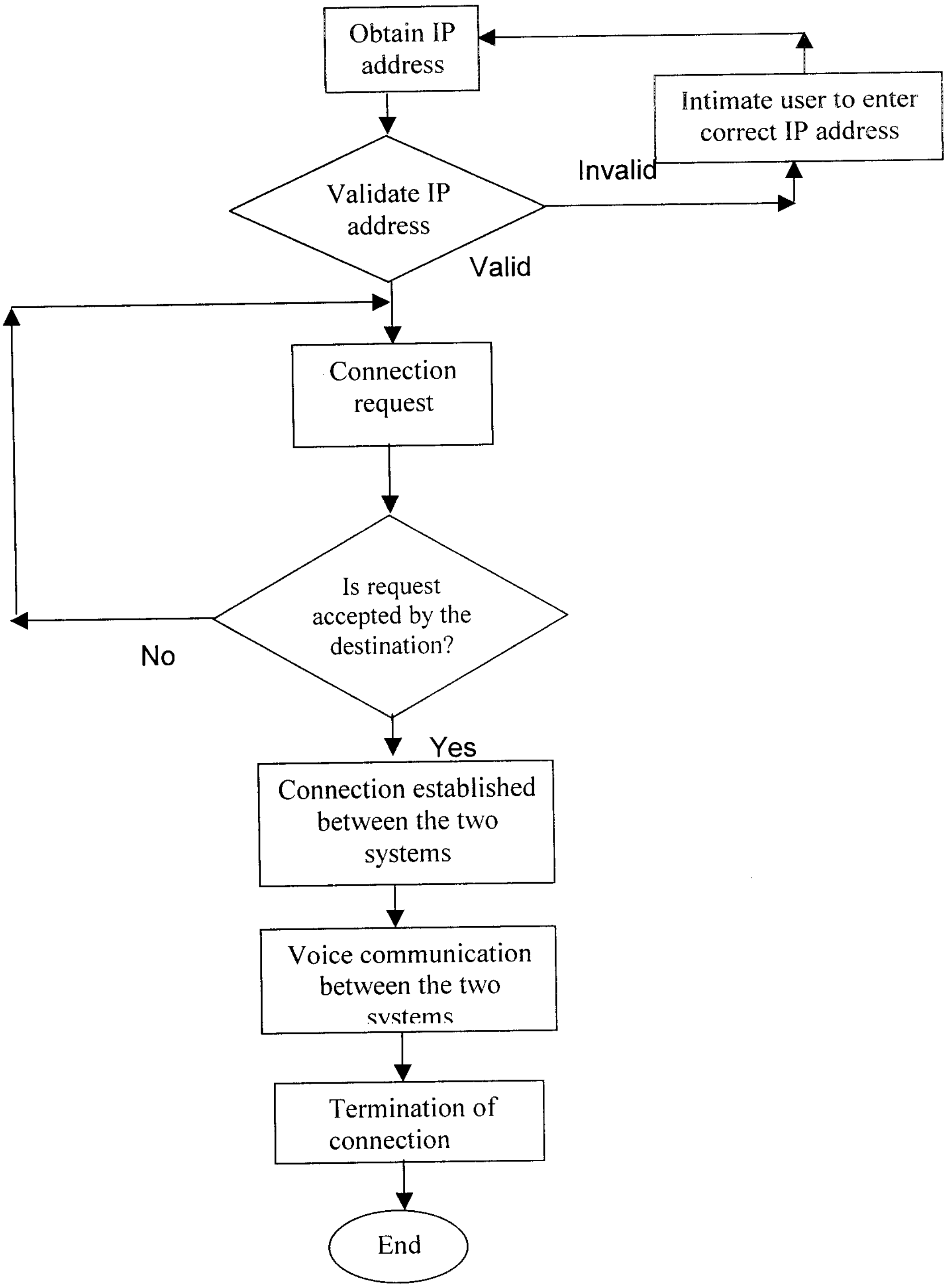
3.5.3 Hardware and Software Interfaces with other Systems:

The hardware interface consists of multimedia kit which is connected to the PC. The software interface consists of a client dialer by which the client enters the destination IP address.

DESIGN DOCUMENTATION

4. DESIGN DOCUMENTATION

4.1 SYSTEM DESIGN:


Obtain IP address
Intimate user to enter correct IP address
Validate IP address
Invalid
Valid
Connection request
Is request accepted by the destination?
No
Yes
Connection established between the two systems
Voice communication between the two systems
Termination of connection
End

4.2 INPUT DESIGN:

The input design is the link between the information system and the user. It comprises the developing specification and those steps necessary to put transaction data into a usable form for processing data entry. The design of input focuses on controlling the amount of input required, controlled errors, avoiding delay, avoiding extra steps and keeping the process simple.

The following things are considered.

a) What data should be given as input.

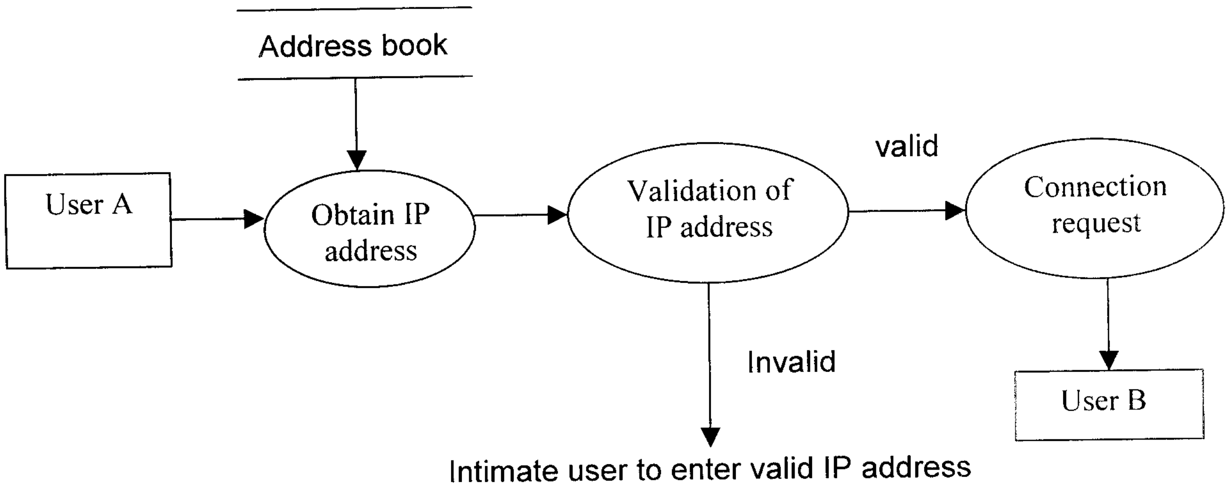
b) How the data should be coded.

c) Methods for preparing input validation and steps to follow when errors occur.

d) IP address of the client with which communication is to take place is entered in the IP address control.

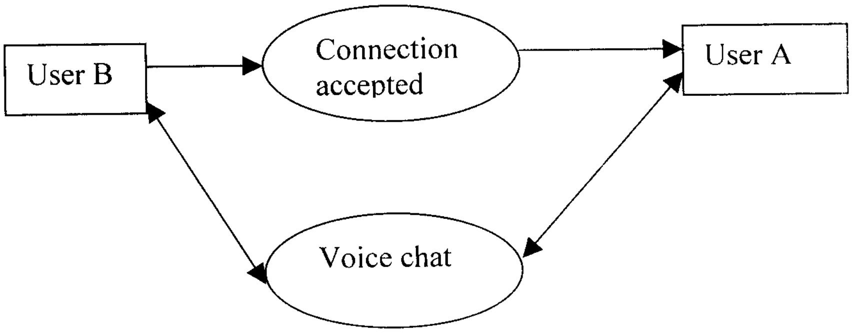
4.3 DETAILED DESIGN:

Connection module:



IP Address of the destination system is obtained from the user. This IP address is for its validation as IP Address follows the unique standard notation. The client tries to connect to the destination PC by creating a socket. If the destination accepts the connection, it sends the source an acknowledgement. Hence a connection between the two systems wishing to communicate is established.

Voice communication module:



Once connection is established, the multimedia kit is enabled and the voice from both the ends is transmitted and received as chunks of data. The transmission and reception of data should be done fast enough so that real time voice communication can be effective.

Client Interface module:

A client interface similar to that of a dialer which enables the user to make a call is implemented. It also includes features like address book for user reference. New addresses can be added. Existing addresses can be modified and deleted.

SYSTEM TESTING

5. SYSTEM TESTING

Testing is an activity to verify that a correct system is being built and is performed with the intent of finding faults in the system. Testing is an activity, however not restricted to being performed after the development phase is complete. But this is to be carried out in parallel with all stages of system development, starting with requirement specification. Testing results once gathered and evaluated, provide a qualitative indication of software quality and reliability and serve as a basis for design modification if required.

In our project “Intranet Communication using VoIP”, the IP address entered by the user is checked for it’s validity. If the IP address entered is valid then the connection and the subsequent voice communication process follows. Else an appropriate message box is displayed to intimate the user that the IP address he entered is no valid and he has to again enter a valid IP address. Also a test case is designed to check whether the host is reachable or not.

5.1 TESTING OBJECTIVES:

Our project testing includes two tests:

a) Validating IP addresses.

b) Determining whether the host is reachable or not.

FUTURE ENHANCEMENTS

6.FUTURE ENHNACEMENTS

Our project is so flexible that it can be enhanced so easily. Some more features could also be added to our project. The future IP telephony equipment will be capable for mobility and then users are not tied to a particular location. It includes:

- Voice conferencing
- PC to telephone communication
- Telephone to telephone communication using the existing data network

We will also focus our attention on making our code platform independent. This can be done by coding our project in any of the platform independent languages like Java. This makes the code to be executable globally.

CONCLUSION

7. CONCLUSION

The complete design and development of the system "Intranet communication using VoIP" is presented in this tract. The system has user friendly features. It is possible for any novice user to use the system. Our project allows a company to eliminate voice network altogether. If implemented, this project would serve to be fruitful to many business organizations which would like to make best use of Intranet VoIP facility.

The programming techniques used in the design of the system provide a scope for further expansion and implementation of any changes, which may occur in the future. The system has been tested for the hosts running on any Windows Operating System.

Since, the requirements of any organization and their standards are changing day by day; the system has been designed in such a way that its scope and boundaries could be expanded in the future with little modifications. As a further enhancement, this system can include the other features specified in the Future Enhancement Section.

BIBLIOGRAPHY

8.BIBLIOGRAPHY

Reference Books:

1. "Win32 Programming" by Brent E. Rector, Joseph M. Newcomer, 1997 Edition, Addison Wesley Publications.
2. "Programming Windows" by Charles Petzold, Fifth Edition, Microsoft Press.

Websites Visited:

1. www.microsoft.com
2. www.cisco.com
3. www.networkcomputing.com
4. www.winprog.com
5. www.protocols.com

ANNEXURE

9.ANNEXURE

9.1 Sample code:

```
#include<windows.h>
#include<winsock.h>
#include<commctrl.h>
#include<stdio.h>
#include<io.h>
#include<fcntl.h>
#include<sys/types.h>
#include<sys/stat.h>
#include "mmsystem.h"
#include"resource.h"
#include "tapi.h"
#include<process.h>

void receivethread(PVOID);
void sendthread(PVOID);
void  RegisterControl();
void addmember();
void CALLBACK waveInProc(HWAVEIN hwi,UINT uMsg,DWORD dwInstance,
                        DWORD dwParam1, DWORD dwParam2);
void CALLBACK waveOutProc( HWAVEOUT hwo, UINT uMsg, DWORD
                dwInstance,  DWORD dwParam1,   DWORD dwParam2 );
BOOL CALLBACK dlgproc(HWND,UINT,WPARAM,LPARAM);
BOOL CALLBACK dlgprocaddr(HWND,UINT,WPARAM,LPARAM);
BOOL CALLBACK dlgprocaddrdet(HWND,UINT,WPARAM,LPARAM);
BOOL CALLBACK dlgprocdelmem(HWND,UINT,WPARAM,LPARAM);
BOOL CALLBACK Splash_Dlgproc(HWND,UINT,WPARAM,LPARAM);

int x=150, y=300, id=100, i, t, fp2, sock, nsock, fedit=0, code;
int iSelect=0, index=0,flag=0, naddr=0, iSlected=0;
char name[50],ipaddr[20],phone[20],email[50],addr[50];
char szError[100],tchBuffer[100],buffer[17000],buf[50];
char recvbuf[17000], smem[50]="", mem[50], str[20];
char menufile[]="menufile.txt", addrfile[]="addr.txt";
char tempstr[100]="", s[50];

WSADATA wsa;
BYTE b1, b2, b3, b4;
LPARAM lip,lib;
DWORD ipbuff;
FILE * fp,*fp1;
LVCOLUMN LvCol;
LVITEM LvItem;
LV_DISPINFO lvd;
bool escKey=0;
MSG msg1;

HWND hname,hemail,hphone,haddr,hipaddr;
HWAVEIN m_hWaveIn, hWaveIn;
HWAVEOUT   hWaveOut;
HWND hlist,hwndg,hEdit;
HPSTR   m_pData;
HINSTANCE hcg;
HMENU hmenu;

static TCHAR szfilename[]=TEXT("Record2.wav") , tchar;
static HWND hList=NULL;
static BOOL brec,bplay,bpause;
static WORD wDeviceID;
```

```
DWORD dwError;
MCI_GENERIC_PARMS mcigeneric;
MCI_OPEN_PARMS mciopen;
MCI_PLAY_PARMS mciplay;
MCI_RECORD_PARMS mcirecord;
MCI_SAVE_PARMS mcisave;
WAVEHDR*    m_pWaveHdr,pWaveHdr;
WAVEFORMATEX*   m_pFormat;
BOOL set=false;

struct addrbook
{
        char name[50];
        char ipaddr[20];
        char phone[20];
        char email[50];
        char addr[50];
}maddrbook;
struct sockaddr_in server;

int WINAPI WinMain(HINSTANCE hc,HINSTANCE hp,LPSTR cmd,int cshow)
{
        RegisterControl();
        hcg=hc;
        DialogBox(hc,MAKEINTRESOURCE(IDD_DIAMAIN),NULL,dlgproc);
        return 0;
}

BOOL CALLBACK dlgproc(HWND hwnd,UINT msg,WPARAM wp,LPARAM lp)
{
```

```
LPPOINT pt;
switch(msg)
{
  case WM_LBUTTONDOWN:
        GetCursorPos(pt);
        x=pt->x;
        y=pt->y;
        break;

  case WM_COMMAND:
        switch(wp)
        {

          case IDC_CALL:
                SendMessage(GetDlgItem(hwnd,IDC_IPADDR),
                   IPM_GETADDRESS,0,(LPARAM)(LPDWORD)&ipbuff);
                 lip=(LPARAM)(LPDWORD)ipbuff;
                 b1=FIRST_IPADDRESS(lip);
                b2=SECOND_IPADDRESS(lip);
                b3=THIRD_IPADDRESS(lip);
                b4=FOURTH_IPADDRESS(lip);
                sprintf(tchBuffer,"%d.%d.%d.%d",b1,b2,b3,b4);

                if(WSAStartup(MAKEWORD(2,2), &wsa) != 0)
                {
                     wsprintf (szError, TEXT("WSAStartup failed. Error: %d"),
                                    WSAGetLastError ());
                     MessageBox (NULL, szError, TEXT("Error"), MB_OK);
                     return FALSE;
                  }
```

```
if((sock = socket(AF_INET, SOCK_STREAM, 0)) ==
                INVALID_SOCKET)
{
     wsprintf (szError, TEXT("socket failed. Error: %d"),
                    WSAGetLastError ());
     MessageBox (NULL, szError, TEXT("Error"), MB_OK);
     return FALSE;
}

  server.sin_family = AF_INET;
  server.sin_port=htons(5600);
  server.sin_addr.S_un.S_addr=inet_addr(tchBuffer);
  if((code=connect(sock,(SOCKADDR *) &server,
                      sizeof(server )))<0)
  {
        PlaySound("busy.wav",NULL,
             SND_FILENAME|SND_ASYNC|SND_LOOP);
        t=MessageBox(NULL,"Line is busy ","Err",0);
        if(t==IDOK)
                 PlaySound(NULL,NULL,SND_FILENAME|
                   SND_ASYNC|SND_LOOP);
         return 0;
      }

      send(sock,"connect",8,0);
      send(sock,tchBuffer,strlen(tchBuffer),0);
      recv(sock,buf,sizeof(buf),0);
      MessageBox(NULL,buf,"Msg",0);
      _beginthread(receivethread,0,NULL);
      _beginthread(sendthread,0,NULL);
   break;
   case IDCANCEL:
             PostQuitMessage(0);
             break;
   case IDC_ADDR:
            DialogBox(hcg,MAKEINTRESOURCE(IDD_DIAADDR),
                  NULL,dlgprocaddr);
             break;
   case IDC_MEMORY:
              id=100;
              hmenu=CreatePopupMenu();
              fp1=fopen(menufile,"r");
              fgets(smem,sizeof(smem),fp1);
              while(!feof(fp1))
              {
                  AppendMenu(hmenu,MF_STRING,id++,smem);
                  fgets(smem,sizeof(smem),fp1);
                }
                fclose(fp1);
               TrackPopupMenuEx(hmenu,TPM_VERTICAL,
                      x,y,hwnd,NULL);
               break;

    case IDC_CLEAR:
                SendMessage(GetDlgItem(hwnd,IDC_IPADDR),
                      IPM_CLEARADDRESS,0,0);
                break;
}
fp1=fopen(menufile,"r");
for(i=100;i<id;i++)
{
```

```
            fgets(smem,sizeof(smem),fp1);
            if(i==wp)
            {
                strcpy(mem,smem);
                strtok(mem,":");
                strcpy(mem,strtok(NULL," "));
                strtok(mem,".");
                b1=atoi(mem);
                strcpy(mem,strtok(NULL,""));
                strtok(mem,".");
                b2=atoi(mem);
                strcpy(mem,strtok(NULL,""));
                strtok(mem,".");
                b3=atoi(mem);
                strcpy(mem,strtok(NULL,""));
                b4=atoi(mem);
                lip=MAKEIPADDRESS(b1,b2,b3,b4);
                SendMessage(GetDlgItem(hwnd,IDC_IPADDR),
                    IPM_SETADDRESS,0,lip);
            }
        }
        fclose(fp1);
        break;
    }
    return 0;
}

BOOL CALLBACK dlgprocaddr(HWND hwnd,UINT msg,WPARAM wp,
                                                LPARAM lp)
{
```

```
switch(msg)
{
case WM_INITDIALOG:
        {
                        int i;
                        char Temp[255];
                        LVBKIMAGE plvbki={0};
                        InitCommonControls();
                        hList=GetDlgItem(hwnd,IDC_LISTADDR);
                        memset(&plvbki,0,sizeof(plvbki));
                        plvbki.ulFlags=LVBKIF_SOURCE_URL;
                        plvbki.xOffsetPercent=40;
                        plvbki.yOffsetPercent=15;
                        OleInitialize(NULL);
                        SendMessage(hList,LVM_SETTEXTBKCOLOR,
                                0,(LPARAM)CLR_NONE);
                        SendMessage(hList,LVM_SETBKIMAGE,0,
                                (LPARAM)(LPLVBKIMAGE)&plvbki);
                        SendMessage(hList,LVM_SETEXTENDEDLISTVIEW
                                STYLE, 0, LVS_EX_FULLROWSELECT);
                         memset(&LvCol,0,sizeof(LvCol));
                        LvCol.mask=LVCF_TEXT|LVCF_WIDTH|
                                        LVCF_SUBITEM;
                        LvCol.cx=0x28;
                        LvCol.pszText="S NO";
                        LvCol.cx=0x42;

                        SendMessage(hList,LVM_INSERTCOLUMN,0,
                                (LPARAM)&LvCol);
                        SendMessage(hList,LVM_INSERTCOLUMN,1,
                                        (LPARAM)&LvCol);
                                SendMessage(hList,LVM_INSERTCOLUMN,2,
                                        (LPARAM)&LvCol);
                                SendMessage(hList,LVM_INSERTCOLUMN,3,
                                        (LPARAM)&LvCol);
                                SendMessage(hList,LVM_INSERTCOLUMN,4,
                                        (LPARAM)&LvCol);
                                SendMessage(hList,LVM_INSERTCOLUMN,5,
                                        (LPARAM)&LvCol);
                                memset(&LvItem,0,sizeof(LvItem));

                                TranslateMessage(&msg1);
                                DispatchMessage(&msg1);

                                fp=fopen(addrfile,"r");
                                fread(&maddrbook,sizeof(maddrbook),1,fp);
                                while(!feof(fp))
                                {
                                        addmember();
                                        fread(&maddrbook,sizeof(maddrbook),1,fp);
                                }
                        }
                        break;

        case WM_COMMAND:
                switch(LOWORD(wp))
                {
                case IDC_CLOSE:
                        EndDialog(hwnd,0);
                        break;
```

```
case IDC_ADDADDR:
        DialogBox(hcg,MAKEINTRESOURCE(IDD_DIAADDRDET),
                NULL,dlgprocaddrdet);
        if(naddr==1)
        addmember();
        break;

case IDC_DELADDR:
        iSlected=0;
        iSlected=SendMessage(hList,LVM_GETNEXTITEM,
              -1,LVNI_FOCUSED);
        if(iSlected==-1)
        {
           MessageBox(hwnd,"No Items in ListView",
                "Error",MB_OK|MB_ICONINFORMATION);
           break;
        }
        SendMessage(hList,LVM_DELETEITEM,iSlected,0);
        break;

case IDC_EDITADDR:
        iSlected=0;
        iSlected=SendMessage(hList,LVM_GETNEXTITEM,
              -1,LVNI_FOCUSED);
        if(iSlected==-1)
        {
                MessageBox(hwnd,"No Items in ListView",
                        "Error",MB_OK|MB_ICONINFORMATION);
                break;
        }
```

```
naddr=0;
LvItem.mask=LVIF_TEXT;
LvItem.cchTextMax = 256;
LvItem.iItem=iSlected;
LvItem.iSubItem=1;
LvItem.pszText=maddrbook.name;
SendMessage(hList,LVM_GETITEM,0,(LPARAM)&LvItem);
LvItem.iSubItem=2;
LvItem.pszText=maddrbook.ipaddr;
SendMessage(hList,LVM_GETITEM,0,(LPARAM)&LvItem);
LvItem.iSubItem=3;
LvItem.pszText=maddrbook.phone;
SendMessage(hList,LVM_GETITEM,0,(LPARAM)&LvItem);

LvItem.iSubItem=4;
LvItem.pszText=maddrbook.email;
SendMessage(hList,LVM_GETITEM,0,(LPARAM)&LvItem);
LvItem.iSubItem=5;
LvItem.pszText=maddrbook.addr;
SendMessage(hList,LVM_GETITEM,0,(LPARAM)&LvItem);
fedit=1;
DialogBox(hcg,MAKEINTRESOURCE
          (IDD_DIAADDRDET),NULL, dlgprocddrdet);
while(!naddr);
char sno[10];
sprintf(sno,"%d",iSlected);
LvItem.mask=LVIF_TEXT;
LvItem.cchTextMax = 256;
LvItem.iItem=iSlected;
LvItem.iSubItem=0;
```

```
        LvItem.pszText=sno;
        SendMessage(hList,LVM_SETITEM,0,(LPARAM)&LvItem);
        LvItem.mask=LVIF_TEXT;
        LvItem.cchTextMax = 256;
        LvItem.iItem=iSlected;
        LvItem.iSubItem=1;
        LvItem.pszText=maddrbook.name;
        SendMessage(hList,LVM_SETITEM,0,(LPARAM)&LvItem);
        LvItem.iSubItem=2;
        LvItem.pszText=maddrbook.ipaddr;
        SendMessage(hList,LVM_SETITEM,0,(LPARAM)&LvItem);
        LvItem.iSubItem=3;
        LvItem.pszText=maddrbook.phone;
        SendMessage(hList,LVM_SETITEM,0,(LPARAM)&LvItem);
        LvItem.iSubItem=4;
        LvItem.pszText=maddrbook.email;
        SendMessage(hList,LVM_SETITEM,0,(LPARAM)&LvItem);
        LvItem.iSubItem=5;
        LvItem.pszText=maddrbook.addr;
        SendMessage(hList,LVM_SETITEM,0,(LPARAM)&LvItem);
   }
  break;

case IDC_SAVEADDR:
    {
       int i;
       DeleteFile(addrfile);
       _creat(addrfile,_S_IREAD | _S_IWRITE);
       fp=fopen(addrfile,"w");
              strcpy(maddrbook.addr,"");
strcpy(maddrbook.email,"");
strcpy(maddrbook.ipaddr,"");
strcpy(maddrbook.name,"");
strcpy(maddrbook.phone,"");

int item=SendMessage(hList,LVM_GETITEMCOUNT,0,0);
for(i=0;i<item;i++)
{
  LvItem.mask=LVIF_TEXT;
  LvItem.cchTextMax = 256;
  LvItem.iItem=i;
  LvItem.iSubItem=1;
  LvItem.pszText=maddrbook.name;
  SendMessage(hList,LVM_GETITEM,0,(LPARAM)&LvItem);

  LvItem.iSubItem=2;
  LvItem.pszText=maddrbook.ipaddr;
  SendMessage(hList,LVM_GETITEM,0,(LPARAM)&LvItem);
  LvItem.iSubItem=3;
  LvItem.pszText=maddrbook.phone;
  SendMessage(hList,LVM_GETITEM,0,(LPARAM)&LvItem);
  LvItem.iSubItem=4;
  LvItem.pszText=maddrbook.email;
  SendMessage(hList,LVM_GETITEM,0,(LPARAM)&LvItem);
  LvItem.iSubItem=5;
  LvItem.pszText=maddrbook.addr;
  SendMessage(hList,LVM_GETITEM,0,(LPARAM)&LvItem);
  fwrite(&maddrbook,sizeof(maddrbook),1,fp);
}
fclose(fp);
```

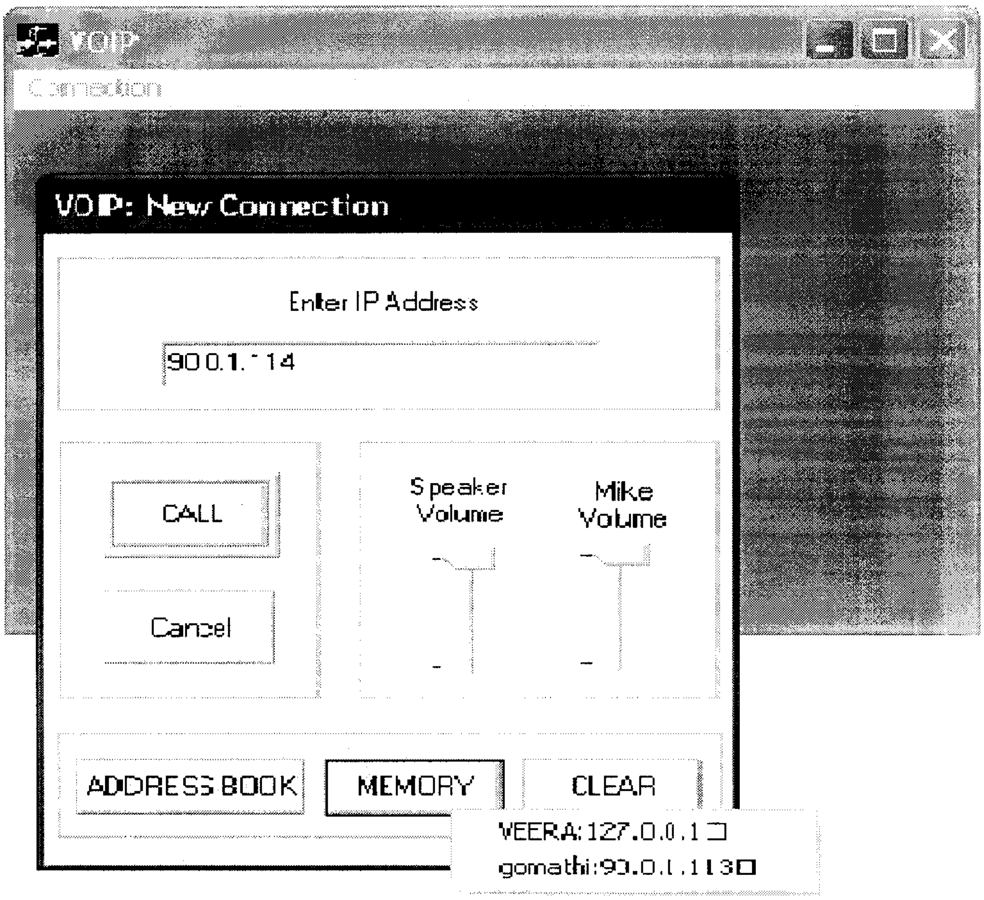
```
                break;
            }
            return 0;
}

void receivethread(PVOID pv)
{
            int flag=0;
             while(1)
              {
                        fp2=_creat("soun2.wav",_S_IREAD | _S_IWRITE);
                        fp2=_open("soun2.wav",_O_APPEND,_S_IREAD|_S_IWRITE);
                        while((recv(nsock,buffer,sizeof(buffer),0))!=-1)
                        {
                          _write(fp2,buffer,sizeof(buffer));
                          MessageBox(NULL,buffer,"received",0);
                          PlaySound("soun2.wav",NULL,SND_FILENAME|SND_ASYNC);
                          flag=1;
                        }
                         _write(fp2,"-1",2);
                         _close(fp2);
                        Sleep(2000);
                        }
             _endthread();
}

void sendthread(void * pv)
{
        while(1)
        {
            send(nsock,"-1",sizeof(int),0);
            brec=FALSE;
            _close(fp2);
            Sleep(2000);
        }
        _endthread();
}
```

9.2 Sample output:

Output of dialer screen:

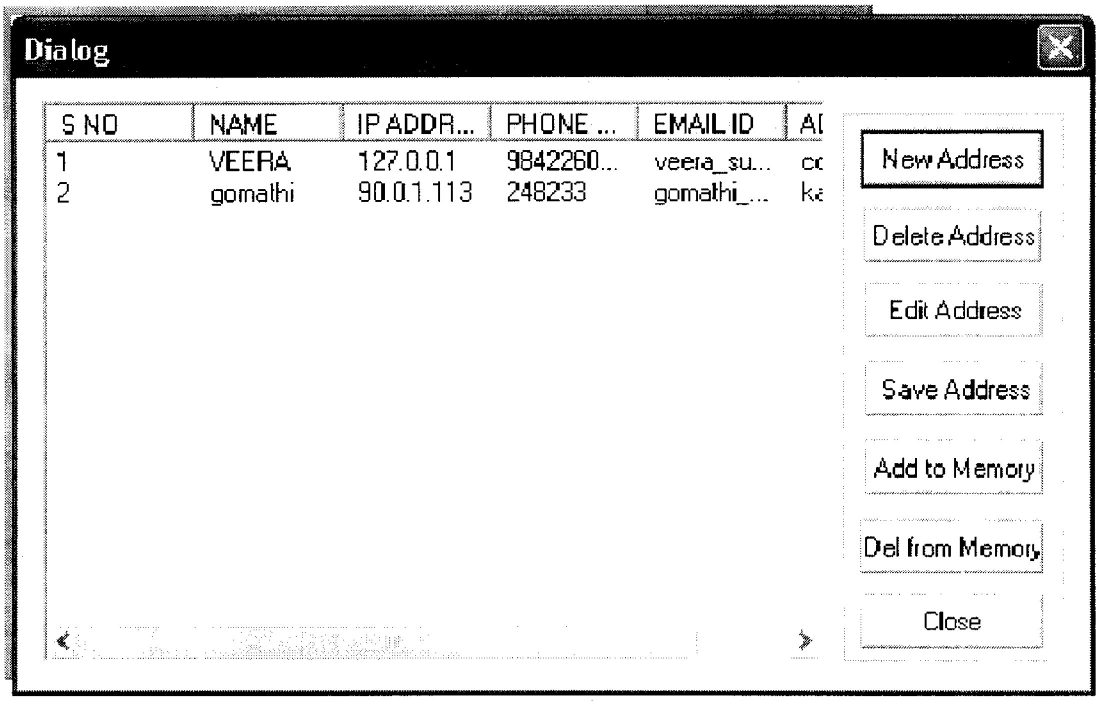


Enter IP Address

900.1.114

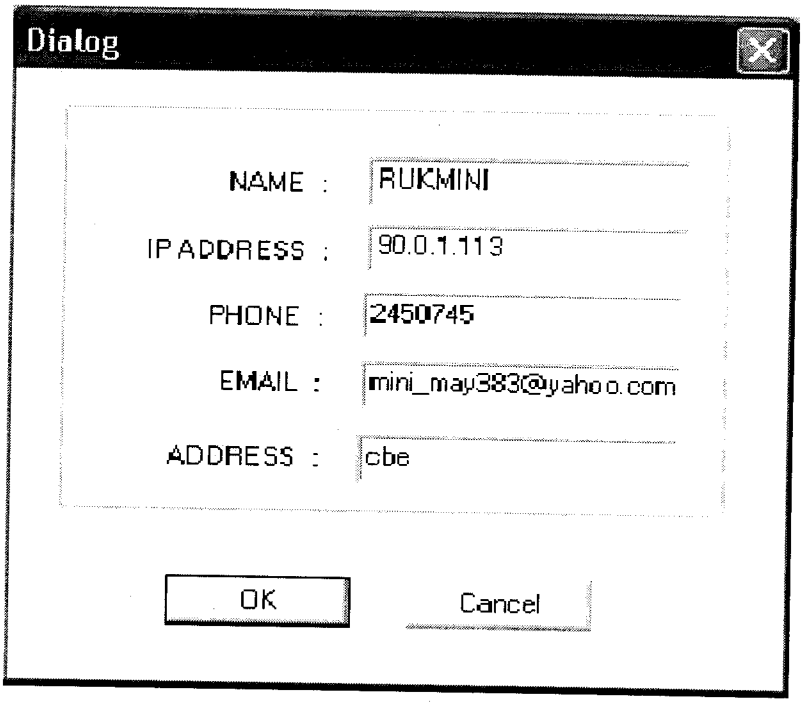
VEERA:127.0.0.1□
gomathi:93.0.1.113□

Sample Address Book screen:



S NO	NAME	IP ADDR...	PHONE ...	EMAIL ID	AI
1	VEERA	127.0.0.1	9842260...	veera_su...	cc
2	gomathi	90.0.1.113	248233	gomathi_...	ka

Sample Address Details Screen:



NAME : RUKMINI

IPADDRESS : 90.0.1.113

PHONE : 2450745

EMAIL : mini_may383@yahoo.com

ADDRESS : cbe