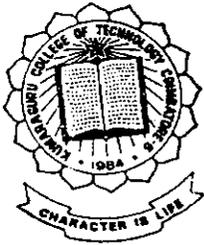


“YOUR COMPANION - THE NLP PERSON”

P-258



PROJECT REPORT

SUBMITTED BY

G. ANUSHA SARAVANA DEVI

L. SUNDHARARAMAN

GUIDED BY

Mr. S.ANDREWS, M.Sc.,PDGPM

Department of Computer Science and Engineering

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE OF

BACHELOR OF SCIENCE IN

APPLIED SCIENCE - COMPUTER TECHNOLOGY

OF THE BHARATHIAR UNIVERSITY, COIMBATORE.

1999 -2000

Department of Computer Science and Engineering

Kumaraguru College of Technology

Coimbatore - 641 006.



CERTIFICATE

**KUMARAGURU COLLEGE OF
TECHNOLOGY**
Coimbatore – 641 006

Department of Computer Science and Engineering

Certificate

This is certify that the report entitled “Your Companion - The NLP Person” has been submitted by L. SUNDHARA RAMAN *Reg.No: 9727Q0045*, G. ANUSHA SARAVANA DEVI *Reg.No: 9727Q0005* in partial fulfillment for the award of the degree of *Bachelor of Science in the Applied Science – Computer Technology*, Branch of Bharathiar University, Coimbatore – 641 046 during the academic year 1999-2000.

.....
Guide

S. Jayaram 28/3/.....
Head of the Department

Certified that the candidates with University Registration No. *9727Q0045*
and *9727Q0005* were examined in the project viva-voce examination held on
28.03.2000

.....
External Examiner

.....
External Examiner

aditi diamonds private limited

Pondicherry
10-03-2000

To Whomsoever It May Concern

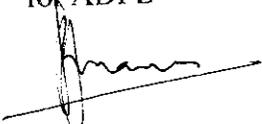
The following students of **Kumaraguru College of Technology, Coimbatore**, studying final year B.Sc. (C.T) have completed their project for our Company for the year 1999 - 2000.

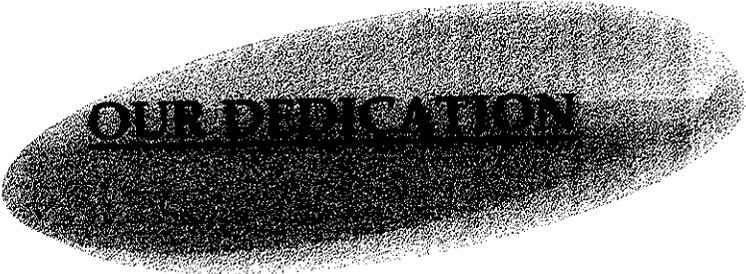
Mr. L. Sundhararaman
Ms. G. Anusha Saravana Devi

Name of the project	: "Your Companion - The NLP person"
Operating system	: Windows 3.x and above
Hardware configuration	: 32 MB RAM, Pentium Class processor, Sound Card, Microphone and Speakers.
Technical utility of the project	: Recognising the users speech and generating responses appropriately.
Duration of the project	: 4 months
Project implementation date	: 10.03.2000

They have put up many hours of hard work to achieve perfection in their project work.
We appreciate their efforts and wish them a bright future and success in all their endeavour.

for ADPL


(J.MANOU)



OUR DEDICATION

*We are Proud to dedicate
Our project to those who brought us
To this world and to those who
Embellish our lives in this world
And at last to the ALMIGHTY*

CONTENTS

	<u>Page no.</u>
Acknowledgement	
1. INTRODUCTION	
1.1 Purpose	4
1.2 Scope	6
1.3 Definitions	7
1.4 Overview	14
2. GENERAL DESCRIPTION	
2.1 Product perspective	15
2.2 Product function	16
2.3 User chart	18
3. REQUIREMENT SPECIFICATION	
3.1 Function requirement	
3.1.1 Introduction	22
3.1.2 List of inputs	23
3.1.3 List of outputs	51
3.2 Performance requirement	
3.2.1 Introduction	65
3.2.2 Security	66
3.2.3 Availability	67
3.2.4 Capacity	68
3.2.5 Response time	69
3.3 Design constraints	
3.3.1 Hardware limitation	70
3.3.2 User interface	72
3.3.3 Screen format	73
APPENDIX	
Sample Source Code	78
FUTURE ENHANCEMENTS	126
CONCLUSION	127
BIBLIOGRAPHY	

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

We wish to express our sincere and heartfelt gratitude to DR.K.K.Padmanabhan, Bsc(Engg.), M.Tech, PhD, Our esteemed Principal, KCT. For giving us the needed encouragement in starting this project and carrying this out successfully.

Our heartfelt thanks and deep sense gratitude to professor to professor DR.S.Thangasamy BE(HONS.), PhD., Head Of the Department of Computer Science and Engineering for his benevolent attitude and spurring encouragement.

We like to thank our Guide and Class Advisor MR.S.Andrews, M.sc., PDGPM, Department of Computer Science and Engineering without whose motivation and guidance we would not have been able to embark on the project of this magnitude. We would also like to thank him for his constant support and for

helping to improve in our spirits towards the aim we had in our project transcendently.

We sincerely appreciate the gratitude of MR.J.MANOU and Kavingar R.MEENAKSHI, for allowing us to perform a project through which our computing talents reach greater heights. The encouragement given to us and the timely spotting and stopping us where we deviated from path that would lead us towards the necessary results were unforgettable. The statement of needs that were given to us made our jobs easier to get the results efficiently.

We reciprocate the kindness shown to us by the staff members of the Department of Computer Science and Engineering.



YOUR COMPANION

- The NLP person

INTRODUCTION

Making computer recognize what humans say is a long-term process involving several complex problems. One among the ways to solve the complexness of this problem is being used in this project, say Your Companion – The NLP (Natural Language Processing) person.

This project is about enabling the computer to generate appropriate responses for the words/sentences spoken by a human. Making him think that computer can do both speaking and listening and moreover **Thinking**. Though it is impossible to make computer understand what ever a human says, this project is aimed to understand the terms that are related with the **Internet**.

The interests of the development lies in the area of providing a path for the **VISION IMPIARED** people to use the computer as their **TUTOR** for teaching them and a companion to clear their doubts about the latest heartthrob subject the Internet. The handicapped people can directly feed in their questions to the system without using a keyboard; the system understands the questions and replies a best answer to the user in audio.

As the subject of the project is internet; e-mail forms the most out of the usage of internet. Another additive to the usage of internet by the vision impaired people is an e-mail utility which can be of real usage to them.

This utility when necessary connects to the mail server automatically, through the internet and checks for the arrival of any new mails, receives all the new mails, stores them in the inbox of the default e-mail client, and first reads the subject line of the e-mail followed by the body of the e-mail. As they could even send e-mails by framing it's contents by dictating through their voice itself. This utility can also be used over a network, since MAPI (Messaging Application Programming Interface) programming is used for developing the utility.

Natural language processing offers the greatest potential rewards because it would allow people to interact with computers without needing any specialized knowledge. You could simply walk up to a computer and talk to it. Unfortunately, programming computers to understand natural languages has proved to be more difficult than originally thought. Some rudimentary translation systems that translate from one human language to another are in existence, but they are not nearly as good as human translators. There are also voice recognition systems that can convert spoken sounds into written words, but they do not understand what they are writing; they simply take dictation. Even these systems are quite limited -- User must speak slowly and distinctly.

In this project the Natural language processing is achieved using the speech recognition technique; the conversion of the users voice into words and sentences that makes the process of producing a NLP package easy. Any person with a little English knowledge can easily use this project to full extent and make most out of the same.

PURPOSE

MISSION FOR VISION

The purpose behind the development of this project is to make the computer more interactive to the user, in the sense that the computer responds to whatever user speaks in relation with a specific domain say, internet. The idea is to enable the computer to understand whatever user says about the internet and generate an appropriate response to the users questions.

The intention of the project is to deliver a solution for the handicapped persons to use the computer for their learning; can be served using this project. Handicapped people who are capable of listening and speaking can use the computer even if they are vision impaired or any other kind of physical inefficiency. This proves to be of a real usage to the handicappers' access to the computer as a first step. In the near future there might be a lot of packages for the usage of the handicapped persons for their personal usage of computers to attain its usage to the fullest extent, as other normal users of the computers; get most of their works done out of the manipulation using the computer.

The secondary purpose is to make the user (Normal) think that he or she is not talking to the computer instead to a human. Masking the human from recognizing that the one he or she talking to be a computer adds to the generation of responses with regards to the user input appropriately.

When ever the user speaks to the computer it is automatically recognized by the computer and the recognized terms are being made to understand by the computer so as to generate the responses from the computer and fake the user so as to make him or her think that the one they were talking to is not just a computer, but a pc which is capable of interacting with humans as one of other human beings in their day-to-day life.

SCOPE

The scope of the project is widened till the ends of the internet domain. All the terms related to the internet technologies are being recognized by this project. The responses are generated only for the recognized words. The recognition of words is suppressed to go beyond the Internet field. Once the user speaks a word that the system understands the system generates a response that the user can understand. The project is not only aimed to generate responses; instead can also be used to create text documents, in which the users can feed the text directly by speaking into a microphone which is connected to the computer.

The computer recognizes the voice of the user and creates a text based on the input voice. As a result the combination of the texts can be saved as a document, and can be retrieved at any instance of time. Similarly the program can intake input from any text file for processing.

There is an e-mail accessing utility provided with this project using which the users can check for new mails and receive if any, can hear the contents of the e-mails including the subject line of the mails. This utility supports the usage over a network. The users who are connected to a network which shares its internet connection to its users can also make use of this utility.

DEFINITIONS:

This is where all the acronyms and the abbreviations used in this project are explained in detail. There might be many acronyms or abbreviations in a project that are understandable only to the persons related to that project or that field. The main aim of this part of the documentation is to explain all the terms that are involved in the project under the assumption of the reader as a fresher to these technologies.

NLP

Abbreviation for Natural language processing .It is the technique using which the computer can be programmed to understand human language. As the result of understanding the language further processing in the understood language. This is a classic **Artificial Intelligence** problem that is being worked on the **FIFTH GENERATION** project by scientists.

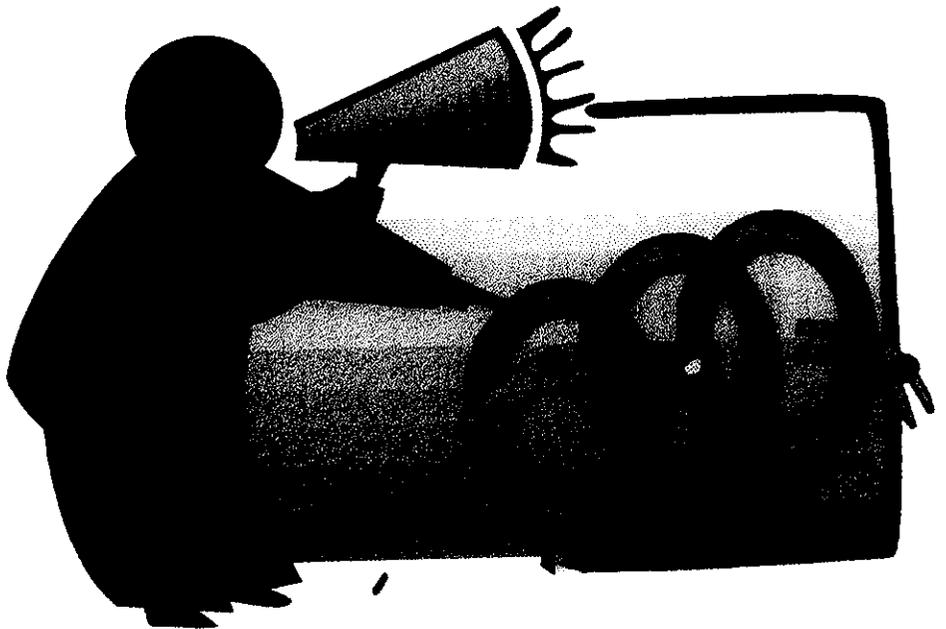
A human language For example, English, French, and Chinese are natural languages where in the other hand the Computer languages, such as FORTRAN and C, are not.

Probably the single most challenging problem in computer science is to develop computers that can understand natural languages. So far, the complete solution to this problem has proved elusive, although a great deal of progress has been made.

FAULT TOLERANCE

Since the languages and slang differ from each and every person, enabling computer to understand all those languages are really impossible in the near future. But a system can developed to understand a whole language completely, yet the process will be consuming a large amount of time in terms of years. The progress in this field indicates that if the time taken to implement this system is going to reduce; then the process will be of larger use, if functioning properly.

SPEECH RECOGNITION





It is the technique by which the words that are spoken by human are being recognized by the computer. Understanding a person's slang is difficult even for many of the humans. Similarly in the case of a computer, understanding different slang's is an uphill task, to overcome that, computer uses some training modules to get comfortable with the users voice. As a result the computer can easily recognize what user is saying.

The field of computer science that deals with designing computer systems that can recognize spoken words. Note that voice recognition implies only that the computer can take dictation, not that it understands what is being said. Comprehending human languages falls under a different field of computer science called natural language processing.

A number of voice recognition systems are available on the market. The most powerful can recognize thousands of words. However, they generally **require an extended training session** during which the computer system becomes accustomed to a particular voice and accent. Such systems are said to be **speaker dependent**.

Many systems also require that the speaker speak slowly and distinctly and separate each word with a short pause. These systems are called discrete speech systems. Recently, great strides have been made in continuous speech systems -- voice recognition systems that allow you to speak naturally. There are several continuous-speech systems available for personal computers, now.

Because of their limitations and high cost, voice recognition systems have traditionally been used only in a few specialized situations. For example, such systems are useful in instances when the user is unable to use a keyboard to enter data because **his or her hands are occupied or disabled**. Instead of typing commands, the user can simply speak into a headset. Increasingly, however, as the cost decreases and performance improves, speech recognition systems are entering the mainstream and are being used as an alternative to keyboards.

TEXT-TO-SPEECH CONVERSION

It is the technique using which the computer can be enabled to speak any text selected. Enabling a computer to speak the text doesn't require much effort as Speech Recognition. Text-to-speech fundamentally functions as a pipeline that converts text into PCM (PULSE CODE MODULATION) digital audio.

It's a computer's ability to produce sound that resembles human speech. Although they can't imitate the full spectrum of human cadences and intonations, speech synthesis systems can read text files and output them in a very intelligible, if somewhat dull, voice. Many systems even allow the user to choose the type of voice -- for example, male or female. Speech synthesis systems are particularly valuable for seeing-impaired individuals.

MAPI

(Messaging Application Programming Interface)

A messaging architecture and a client interface component.

As a messaging architecture, MAPI enables multiple applications to interact with multiple messaging systems across a variety of hardware platforms.

As a client interface component, MAPI is the complete set of functions and object-oriented interfaces that forms the foundation for the MAPI subsystem's client application and service provider interfaces. In comparison with Simple MAPI, Common Messaging Calls (CMC), and the Active Messaging Library, MAPI provides the highest performance and greatest degree of control to messaging-based applications and service providers

OVERVIEW

As explained above the main idea of the project is to make the computer more interactive to the user. The interaction to the user is limited to the internet terms alone. The technologies used in the project are NLP, Speech Recognition, Text-To-Speech conversion. A database containing the keywords is used for generating responses according to the user inputs. The techniques briefed above will be explained in the following passages.

GENERAL DESCRIPTION

PRODUCT PERSPECTIVE

The main concept behind the task, being to accomplish effective communication between user and his or her computer, the plan is to meet the attainable perfection. Granting the freedom of speech to the user eases his or her insight so as to become friendlier with the machine. As a result, the users think their machines as their companions and get pleasure from interacting with the computer.

The project involves speech recognition technique, granting the liberty to the user of communicating directly to the machine. It is easy for any person to speak rather than type in anything. The project can be used for not only to generate responses, but also to create documents of text data, which are typed directly into the computer taking the users voice as the input source. This file can also be saved in the computer, paving the way towards the less usage of the keyboard, while typing any document.

The project also involves text-to-speech conversion which can enable the installed computer to read any sort of text, it has. This utility can come handy in the tight corners like reading a long document , or a lengthy e-mail or irritated of watching the monitor for a long time, the users can use our project to read the textual contents and they can relax their physique.

The people who are so busy that they cant even read the email can use our project to listen to their e-mails, by the time they can do some other work.

PRODUCT FUNCTION

The project function is very simple. The whole process falls in the following steps

- 1) Intializing the necessary resources in the machine
- 2) Recognizing input from the users voice
- 3) Splitting up of the words
- 4) Searching each word in the database
- 5) Displaying the match found from the database
- 6) Reading the response generated
- 7) Releasing the occupied resources at the end

The various steps sated above are executed successively. Each of the steps are explained in the following paragraphs.

1) Initializing the necessary resources in the machine

The speech recognition and text-to-speech conversion needs respective Engines, a half duplex sound card (full duplex recommended) installed, a microphone etc. these are being initialized for the complete governance by program. If any other program does use this resource during initialization, the resources are shared between the applications.

2) Recognizing input from the users voice

As the user speaks through the microphone his voice is converted to PCM (Pulse Code Modulation). Then the transformation of the PCM digital audio into a better acoustic representation is performed. Then the application of a "grammar" so the speech recognizer knows what phonemes to expect. A grammar could be anything from a context-free grammar to full-blown English. The next step is to figure out which phonemes are spoken. Then the Conversion of phonemes into words is performed.

As a result the words that are spoken by the user is converted into words by the speech recognition utility.

3) Splitting up of the words

The sentence parsed from the user input is then split into various separate words with respect to the spaces found in the keyword. Each word after a space is extracted from the sentence and stored separately.

4) Searching each word in the database

The split words are searched in a database containing all the words that are being understood by the computer. Taking one word at a time from the split keywords, the searching is done to find the first match for that word in the database. If there is no match for that word the next word is searched for its match.

5) Displaying the match found from the database

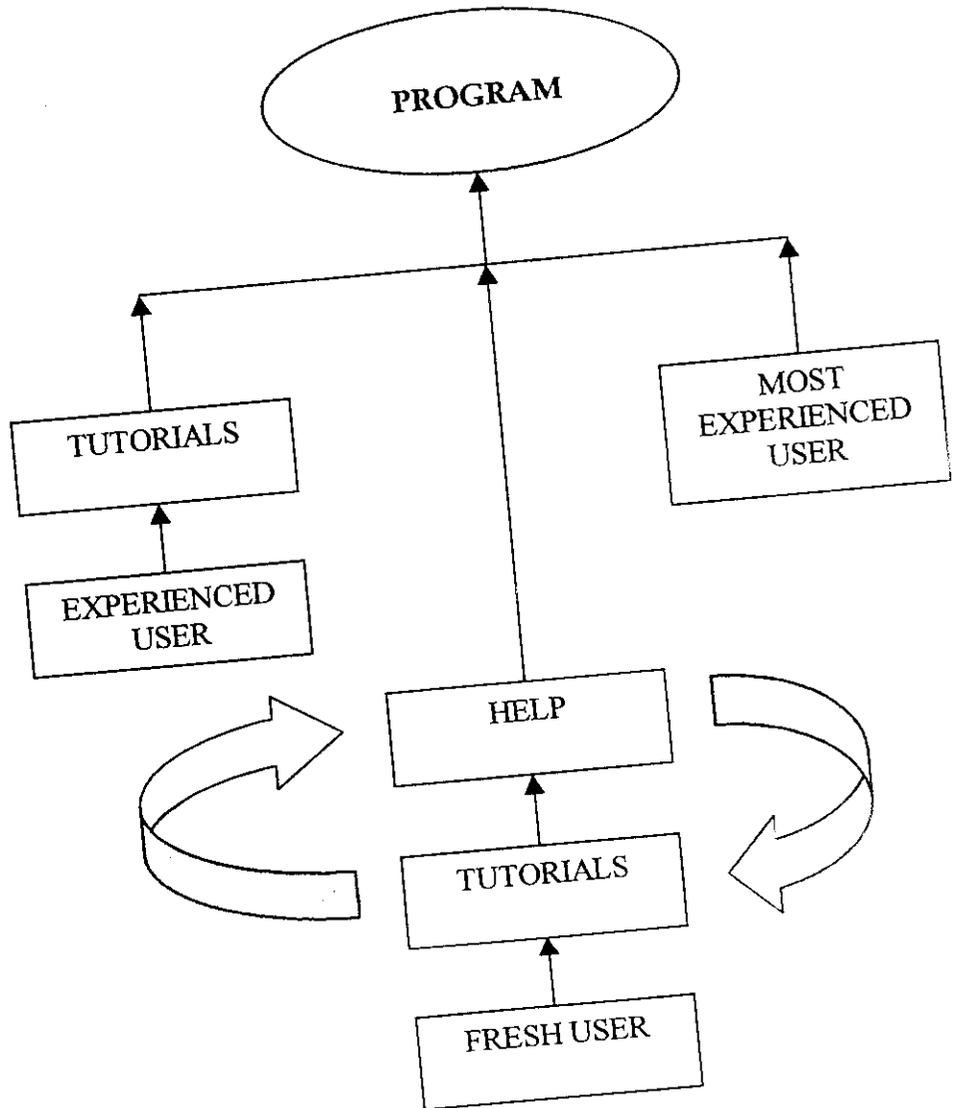
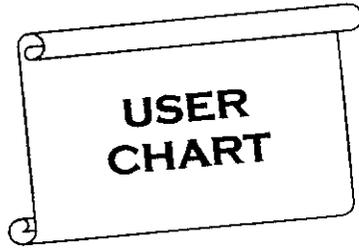
The execution of this step is in accordance with the result of the previous step. If any match is found for the word that is extracted from the user input, then that match is displayed to the user.

6) Reading the response generated

The computer using the text-to-speech conversion technique reads the displayed response. During this step the speech recognition is stopped due to the effective performance of this technique.

7) Releasing the occupied resources at the end

The resources occupied by the speech recognition engines and text-to-speech conversion engines are released at the end of the program termination. This is pave way to other programs to access those resources freely.



The above chart is a brief illustration of the user categories that can be classified according to the practice and experience of the users with the computers. There are three categories of clients that can be classified in accordance with their skills. However the program has been designed for the usage of a fresh user, he or she needs some experience for generating correct response from the computer.

The three user categories are

- 1) Most Experienced user
- 2) Experienced user
- 3) Fresh user

These categories are grouped according to the skill, which are based upon the know-how of the user. The above category divisions are explained underneath, in detail.

1) Most Experienced user

This type of user can directly make use of the project to its fullest scope. i.e., the user belonging to this category doesn't need any kind of training in order to get familiarize with the program. A normal user can convert himself into a Most Experienced user with sufficient experience and training over the program. This level of user is the maximum and beyond this category, no user level can be defined.

The users of this category are masters of the program and can perform tasks with a high suppleness using the functionalities that are available.

2) Experienced user

The user belonging to this category is the kind of person having a little bit of exposure towards the execution and the options available in the project to use its functionalities. There will be help available for him or her along with the project that the user can seek at any time for reference on any topic.

The user of this kind belongs to the intermediate category in the view of usage of the program. The user must know the basics of the program and the guidelines provided to work on it for efficient results.

The user can convert himself directly as a Most Experienced user by using the program a lot of times to get accustomed with its operations. And hence by considerable experience the user can call himself as a Most Experienced user of the program and can perform tasks at a higher degree of speed.

3) Fresh user

The user belonging to this kind of category is a new user to both computers and the program. This user has a very little knowledge of what are process that occur while the program gets executed and the least knowledge about what is available for his or her use from the program, how it can be used for extracting much usage from the program.

The user belonging to this category is of the persons having a least knowledge about the computers and they can convert themselves into a Experienced user by using the tutorials specially created for them, that are supplementary with the program. The help that is available to the Experienced user is also available to this kind of users out of which they could also get guided in a better path for attaining the most valuable experience.

REQUIREMENT SPECIFICATION

The various requirement analysis which are made in the project are being listed in the following subdivision of this section. The functional requirements, performance requirements are listed along with the design constraint. Each requirement are explained in detail so as to list out each and every point to be listed in this documentation. Care is taken to list all the constraints in the project; since the listing of constraints allows other programmers or the users to find out the area where any improvements can be done in case of future development of this project or another project in relation with this project or using the technologies used in this project.

FUNCTIONAL REQUIREMENT

INTRODUCTION

In this module the functional requirements of the project are being explained in detail. The functional requirement is the part in which the functions that are used in the project are decided. This is one of the prelude steps in the forwarding of the project development. The modules that are necessary to complete the project aim are framed in the preliminary part of the requirement phase, where each and every module are designed carefully to make the project containing all the modules to bring out it's efficiency in various issues.

In any project for easy development, the requirements must be analyzed first. In that stage the modules that will suite the requirements are decided, where in the idea is to have a clear base for development of the project. once the requirements of the project is framed, then comes the functional requirements part. In this part the programmer has to decide how to frame the modules as decided in the previous step in functions using a programming language.

In accordance with this project the project is developed using the programming language **Microsoft® VISUAL BASIC 6.0**. The functions are decided in accordance with the modules that are fixed in the requirement analysis phase, were included as per the syntax specified in the programming language. The various properties of the functional analysis are explained in the following paragraphs.

LIST OF INPUTS

The project is a basic system for interaction between the computers and the humans. In any type of interaction the process has something in common between the interacting "objects". In this case the user inputs recognized by the computer are going to be the common criteria between the user and the computer. The user can speak any of words to the computer, in English; which will be converted to text and the following are the words that are being really understood by the computer, as a result of which it generates the responses.

There are single words and multiple words recognized by this system. While opting for multiple keywords the system requires a special syntax to be followed for the generation of the correct response. This syntax is explained soon in the following paragraphs.

For example:

The keyword "INTERNET" can be simply spoken and the response can be generated easily, as

"WHAT IS INTERNET"

(Or)

"EXPLAIN ABOUT INTERNET"

(Or)

"HELP ME KNOW ABOUT WHAT Is The
Meaning Of The INTERNET"

Etc.,

When it comes to the multiple keywords handling the following syntax is to be followed, to get correct response.

Syntax:

<First keyword>--<Second Keyword>[-<Third Keyword>...]

The user has to speak the first keyword, and then must say "HYPHEN", followed by the next keyword, and if has another keyword following must again speak "HYPHEN", and then can speak the next keyword.

The above syntax can be briefly explained by the following example

To get response for "INTERNET SERVICE PROVIDER" the user must follow the above syntax and speak as

"INTERNET" "HYPHEN"

The system will interpret this as

"INTERNET-"

Then "SERVICE" "HYPHEN", as

"INTERNET-SERVICE-"

And at last "PROVIDER", as

"INTERNET-SERVICE-PROVIDER"

By this way the user can feed in multiple keywords. The questions that fall to the machine may framed as

"WHAT IS INTERNET-SERVICE-PROVIDER"

(Or)

"EXPLAIN ABOUT INTERNET-SERVICE-PROVIDER"

Etc.,

The following are the inputs that are recognized by the computers that alone generate responses from the machine.

LIST OF INPUTS
BACKBONE
BANDWIDTH
BANG
BARFMAIL
BASIC
BBS
BCC
BETA
BIBLE
BINARY
BIT
BITNET
BOOKMARK
BOOT
BOUNCE
BOUNCE-MESSAGE
BOX
BOZO-MAIL
BPS
BROWSER
BROWSER-WAR
BTW
BUG
BULLETIN-BOARD-SYSTEM

LIST OF INPUTS

BYTE

C

C++

CASCADE

CC

CD

CENTRAL-PROCESSING-UNIT

CGI

CHAT

CHAT-HISTORY

CHATTER'S-BLOCK

CHECKSUM

CHURN

CHURNING

CLIENT

CLIENT-SERVER-MODEL

CLIPBOARD

COLLISION-DETECTION

COMMERCIAL-ONLINE-SERVICE

COMMON-GATEWAY-INTERFACE

COMPILE

COMPRESSION

COMPUSERVE

COMPUTER-LITERACY

CONGESTION

CONNECTION

COOKIES

COOPETITION

COPY-PROTECTION

CPU

CRACKER

CROSSLOAD

CROSS-POSTING

CRUNCH

CUBE

CU-SEEME

CYBERPUNK

CYBERSPACE

DAEMON

DATA-ENCRYPTION-KEY

DATA-ENCRYPTION-STANDARD

DATABASE

DEDICATED-LINE

LIST OF INPUTS
DELIVERY-RECEIPT
DELURK
DES
DIAL-UP
DIGERATI
DNS
DOMAIN-NAME
DOMAIN-NAME-SYSTEM
DOMAINISM
DOS
DOWNLOAD
DOWNLOAD-CHARGES
DRAG
DUMP
ECASH
EFORM
ELECTRONIC-FRONTIER-FOUNDATION
EMAIL
EMAIL-ADDRESS
EMAIL-NEVER-NEVERLAND
EMAIL-OVERLOAD
EMOTICON
ENCRYPTION
ETHERNET
EXCITE
EXECUTE
EYEBALLS
FACILITATED-CHAT
FAQ
FIDONET
FILE-COMPRESSION
FILE-TRANSFER
FILE-TRANSFER-PROTOCOL
FINGER
FIREWALL
FLAME
FLAME-BAIT
FLAME-ON
FLAME-WAR
FLASH-SESSION
FLAT-FILE
FONT
FORUM

LIST OF INPUTS

FORWARDING

FRAME

FREENET

FREEWARE

FRIED

FTP

FUD

FWIW

FYI

GARBAGE

GET-A-REAL-COMPUTER!

GIF

GIGO

GLITCH

GO-WORD

GOLDEN-RULE-OF-NETIQUETTE

GOPHER

GRAPHICS-INTERCHANGE-FORMAT

GREP

GUI

GURU

HACKER

HANDLE

HANG

HARDWIRED

HELLO-WORLD!

HIT

HOLY-WARS

HOME-PAGE

HOSED

HOST

HOTBOT

HTML

HTTP

HYPE

HYPERLINK

HYPERTEXT

HYPERTEXT-MARKUP-LANGUAGE

HYPERTEXT-TRANSFER-PROTOCOL

ICON

ICONOGRAPHER

IDENTITY-HACKING

IMHO

LIST OF INPUTS

INBOX
INFOBAHN
INFORMATION-SUPERHIGHWAY
INFOSEEK
INSTALL
INTEGRATED-SERVICES-DIGITAL-NETWORK
INTERNET
INTERNET-EXPLORER
INTERNET-RELAY-CHAT
INTERNET-SERVICE-PROVIDER
INTERNET-SOCIETY
INTERNIC
INTEROPERABILITY
INTRANET
IP-ADDRESS
IRC
ISDN
ISOC
ISP
JACK-IN
JAVA
JAVASCRIPT
JOINT-PHOTOGRAPHIC-EXPERTS-GROUP
JPEG
KERMIT
KILL-FILE
KLUGE
KNOWBIE
KNOWBOT
LAN
LEASED-LINE
LINE-NOISE
LINK
LIST-SERVER
LOCAL-AREA-NETWORK
LOL
LOOP
LURK
MACHINE
MACHINE-LANGUAGE
MAILBOMB
MAILING-LIST
MAINFRAME

LIST OF INPUTS

MATRIX

MBONE

MELTDOWN

MENU

MESSAGE-HEADER

MIME

MIRROR-SITE

MISSION-CRITICAL

MODEM

MODEM-MANTRA

MODERATOR

MODS

MOOF

MORF

MOSAIC

MUD

MULTICAST-BACKBONE

MULTIMEDIA-GULCH

MULTIPLEXOR

MULTIPURPOSE-INTERNET-MAIL-EXTENSIONS

MULTITASKING

MULTI-USER-DUNGEON

NANOSECOND

NASTYGRAM

NAVIGATOR

NDA

NET-LINGO

NET-SURFING

NET.GOD

NET.POLICE

NETIQUETTE

NETLAG

NETNEWS

NETWORK

NEWBIE

NEWSGROUP

NON-DISCLOSURE-AGREEMENT

NSFNET

NUKE

OBJECT-ORIENTED-PROGRAMMING

OBJECTS

OFFLINE

ONLINE

LIST OF INPUTS

OOP
OPERATING-SYSTEM
ORTHOGONAL
OUTBOX
PACKET
PACKET-SNIFFING
PARSE
PASSWORD
PATH
PC
PERL
PERSONAL-COMPUTER
PGP
PING
PLUG-INS
POINT-OF-PRESENCE
POINT-TO-POINT-PROTOCOL
POP
POST
POSTMASTER
POWER-NEWBIE
PPP
PRETTY-GOOD-PRIVACY
PRODIGY
PROGRAM
PROGRAMMING-LANGUAGE
PROTOCOL
RAM
RANDOM-ACCESS-MEMORY
READ-RECEIPTS
README-FILE
REAL-SOON-NOW
RECURSIVE
REFRESH
REMOTE-LOGIN
RENDER
RESPONSE-TIME
RFC
RICH-TEXT-FORMAT
ROOT
ROOT-DIRECTORY
ROTFL
ROUTER

LIST OF INPUTS

RSN

RTF

RTFM

SCROLL-BAR

SCROLLING

SEARCH-ENGINE

SECURITY

SENIORNET

SERIAL-LINE-INTERNET-PROTOCOL

SHAREWARE

SHOUTING

SHOVELWARE

SIG

SILICON-ALLEY

SILICON-VALLEY

SLIP

SMILEY

SNA

SNAIL-MAIL

SNEAKERNET

SOHO

SPAM

SPOOL

SPRAYER

STAGE-DIRECTIONS

STATUS-BAR

SUBROUTINE

SUBSCRIBE

SURF

SURGE-PROTECTOR

SYSADMIN

SYSOP

SYSTEMS-NETWORK-ARCHITECTURE

T1

T3

TCP/IP

TELECOMMUNICATIONS

TELECOMMUTING

TELEDILDONICS

TELNET

TEXT-BOX

THRASH

THREAD

LIST OF INPUTS

TILE
TITLE-BAR
TOGGLE
TOKEN-RING
TOOLBAR
TOSSED-OUT
TRAFFIC
TRIPLE-DUB
TROJAN-HORSE
TWEAK
UNICODE
UNIFORM-RESOURCE-LOCATOR
UNIX
UPLOAD
URBAN-LEGEND
URL
USENET
USER
USER-ID
UTILITY
UUCP
UUENCODE
VIRTUAL
VIRTUAL-REALITY-MARKUP-LANGUAGE
VIRUS
VRML
WAIS
WANNABEE
WEBLIOGRAPHY
WEBMASTER
WETWARE
WHOIS
WIDE-AREA-NETWORK
WINDOW
WINDOWS
WINTEL
WIRED
WIZARD
WIZZYWIG
WORLD-WIDE-WEB
WORM
WRT
WWW

LIST OF INPUTS

WYSIWYG

YAHOO

YMMV

FLAME-OFF

NAV-BAR

NAVEX

NAVIGATE

NBIF

NCSA

NET-BUDDY

NET-NANNY

NET-TOOB

NAV-BAR

BEAR-PAW

BETA-BUGS

BFN

BFT

BHOF

BIF

BITD

BITMAP

BLANTHERER

BM

BMP

BODY

BOIL-THE-OCEAN

BOOF

BOOLEAN-LOGIC

BOOLEAN

BOT

BRB

BROADBAND

BORKEN-PIPE

BROWSER-COMPATIBILITIES

BSD

BSOD

BTSOOM

BUFER-PAGE

BULLET

BUTTON

C-SHELL

CADD

CACHE

LIST OF INPUTS

CAD

CAFÉ

CANCELBUNNY

CANCELMOOSE

CANCELBOT

CASE-SENSITIVE

CATCHUP

CATCH-UP

CD-ROM

CELLPADDING

CERF

CHERNOBYL-PACKET

CFML

CGI-BIN

CHANNEL-VIRTUAL-AREA

CHECKBOX

CHICKEN-HEAD

CID

CIO

CIX

CLEAR-TYPE

CLICK-HERE

CLICK-RATE

CLICKABLE-GRAPHIC

IMAGEMAP

CLICKS

CLICKSTREAMS

COLD-FUSION

CMYK

CNET

COBWESITE

COF\$

COMMAND-LINE

COMMERCE-SERVER

COMMUNICATION-SOFTWARE

COM-PORT

CONFIGURE

CONNECTIVITY

CONTENT-PROVIDER

CONTENT

CONTEXT-SENSITIVE-MENU

CONTEXTUAL-COMMERCE

CONVERSATION-VIEW

LIST OF INPUTS

COOL-TALK
CORNEA-GUMBO
COSN
COUNTRY-CODES
CRAWLER
CRC
CROSS-POST
CSS
CTS
CULBR
CYBERAGE-RAIDER
CYBERCAD
CYBERCIDE
CYBERNOIR
CYBERTOWN
CYBRARIAN
DAEMON
DARPA
DATA-GLOVE
DATA-TRAFFIC
DATA-WAREHOUSING
DATA
DATABASE-FRONT-END
DATAGRAM
DDN
DECODE
DECRYPTION
DEFAULT-BROWSER
DEFAULT
DEJA-NEWS-RESEARCH-SERVICE
DENIZEN
DESKTOP-VIDEO
DESKTOP
DIAL-UP-ACCOUNT
DIGITERATI
DIGITAL-AUDIO
DILLIGAS
DIRECTORY
DIRTY-CONNECTION
DISCUSSION-GROUP
DISTRIBUTION
DOT-ADDRESS
DOT-FILE

LIST OF INPUTS

DSS

DUPLEX

DWEEB

DYNAMIC-CONTENT

DYNAMIC-NODE

DYNAMIC-ROTATION

COM

BASEBAND

BATCH

BBIAB

BBL

ABEND

EARN

E-BOOKS

ECRUITING

EDITOR

EDU

E-JOURNAL

ELECTRONIC-PUBLISHING

ELECTRONIC-MALL

STOREFRONT

EMAC

E-MAIL-SHORTHAND

EMBEDDED-HYPERLINK

EMISSARY

EMOTAGS

EMOTICONS

ESP

ETAILING

E-TEXT

EUDORA

EXTENSIONS

EXTERNAL-VIEWER

EXTRANET

E-ZINE

F2F

FASTCGI

FAT-BINARY

FETCH

FIBER-OPTICS

FIDO

FILE-EXTENSION

FILM-AT-11

LIST OF INPUTS

FIRE-OFF

FIREWALL

FIRMWARE

FLAME-WAR

FLOATING-TOOLBAR

FOLLOW-UP

FOO

FOOBAR

FORM

FRAMES

FREE-AGENT

FREE-LIZARD-PROJECT

FRAME-SOURCE

FUBAR

FULL-DUPLEX

FUNCTIONALITY

FUZZY-EXPERT-SYSTEM

FUZZY-LOGIC

FUZZY-OBJECTS

GOOD-THING

GR&D

GRAFFITI

GSM

GTG

GTGB

GTSY

GYPO

GZIP

HANDSHAKE

HACKER-ETHIC

HACKER-JARGON

HAGO

HAYES-COMPATIBLE

HDML

HDTV

HEADER

HELPER-APPLICATION

HIERARCHY

HOLY-WAR

HOSTNAME

HOTDOG

HOTJAVA

HOTLIST

LIST OF INPUTS

HOTMAIL

HOT-PLUGGING

HTH

HTML-REFERENCE-LIBRARY

HTTPS

HUB

HERTZ

IAE

IC

ICQ

IDKY

IDST

IDTS

IETF

IMAP

IMNSHO

IMO

IMPRESSIONS

INCONTEXT-WEBANALYZER

INDEX

INDICES

INFOSPACE

INKTOMI

IN-LINE-IMAGE

INTERNAUT

INTERNET-ACCOUNT

INTERNET-BACKBONE

INTERNET-SECURITY

INTERNETWORKING

INITIALIZATION-STRING

INFORMATION-PACKET

JABBER

JDK

JINI

JPG

KBPS

KEY-PAL

KEYWORD

KILOBYTE

KIT

KYPO

LAG

LAMER

LIST OF INPUTS

LATENCY

LAUNCH

LIVE

LIVE3D

LOGIN

LOGOFF

LOGON

LOGOUT

LTNS

LYCOS

LYNX

NAVEX

NAVIGATE

NBIF

NCSA

NET-BUDDY

NET-NANNY

NET-TOOB

NET ANALYSIS-DESKTOP

NET PERSONALITY

NETARYPUBLIC

NETGRAVITY-ADSERVER

NETIZEN

NETSCAPE-COLOR-PALETTE

NETSITE

ACCESS-NUMER

ACCESS-TIME

ACCESSWARE

ACTIVE

NAV-BAR

ACCESS-CONTROL

NEURAL-NETWORKS

NEWBIE

NEWS

NEWSFEED

NEWSGROUPS

NEWSREADER

NFS

NFW

NIC

NIFOC

NNTP

NOC

LIST OF INPUTS

NODE

NSF

OEM

NORTON-ANTI-VIRUS

NOT-FOUND-404

NP

NREN

NRG

NRN

NSA-LINE-EATER

NSAPI

OFFLINE

OH

OIC

PMFJI

POINT-LISTING

POINT

POINTCAST

ROBOTICS

OLGA

ON-THE-FLY

OOC

OTOH

OOTB

OPEN

OS

OSP

PACKET-SWITCHING

PAGE-REQUESTS

PAGE-VIEWS

PAINT-SHOP-PRO

PALM-PILOT

PARALLEL-PORT

PC-CARD

PCI

PCMCIA

PDA

PDF

PEBCAK

PGML

PH

PKZIP-PKUNZIP

PLANFILE

LIST OF INPUTS

PLATFORM

PLS

PLUG-IN

PORT

PORTAL

POSTING

POSTSCRIPT

POTS

POV

PROLLY

PROMPT

PROXY

PROXY-SERVER

PUSH

QUAKESPY

QUERY

QUEUE

QUICKTIME-VR

RASTER-GRAPHICS

README

REAL-TIME-CHAT

REALAUDIO

REAL-NAME

RECIPROCAL-LINK

RED-BOOK

RED-BRICK-INTELLIGENT-SQL

REMOTE-TERMINAL

RGB

RI&W

RIGHT-CLICK

RINGMASTER

ROT13

ROTFLMAO

RTM

SECURE-CHANNEL

SCALABILITY

SCENEGRAPH

SCRIPT

SCROLL-AND-SCROLL-BAR

SDK

THE-WEB

THIN-CLIENT

THUMBNAIL

LIST OF INPUTS

TIA

TIAL

TIC

SDRAM

SELECT-BOX

SERVER-HAS-NO-DNS-ENTRY

SEX

SGI

SGML

SHELL-ACCOUNT

SHOCKWAVE

SHORTCUT

SHORTHAND

SHTML

SIG-FILE

SLIP/PPP

SMDS

SMILEYS

SMTP

SNAFU

SOLUTION

SORG

SORT

SOUND-CARD

SOUND-PLAYER

SOURCE-CODE

SPEW

SPIDER

SPLASH-PAGE

SPOD

SQL-SERVER

SQL

SSI

SSL

STATIC-IP

STORED-PROCEDURE

STREAMWORKS

STUFFIT-EXPANDER

STYLE-GUIDE

STYLE-SHEETS

SUBNET-MASK

SUYF

SWAG

LIST OF INPUTS

SWDYT
TAG
TALK
TALKER
TANSTAAFL
TARFU
TASK-BAR
TECHNOPHILE
TEOTWAWKI
TERMINAL-EMULATION
TERMINAL-SERVER
TERMINAL
TIER-1-ISP
TIFF
TIF
TRUESPEECH
TLA
TMTOWTDI
UUEE
UUENCODE-UUDECODE
TOOL
TRACKER-BALL
TRANSFER-INTERRUPTED!
TRON
TRUMPET-WINSOCK
TTFN
TTT
TTYL
TX
TYVM
TWAIN
UNPC
UPGRADE-FEVER
USERNAME
USING-A-PREVIOUSLY-CACHED-COPY-INSTEAD
VANILLA
VCALENDER
VDO
VECTOR-GRAPHICS
VERONICA
VERTUAL-CIRCUIT
VIRTUAL-DIRECTORY
VCARD

LIST OF INPUTS

W3

W3C

WAG

WAMBAM

WAN

VRWEB

WEBMISTRESS

WEBRING

WEB-SITE

WEBSMITH

ZINE

ZINE.NET

ZIP

WAREZ

WAV

WB

WEBCRAWLER

WCA

WDYS

WE

WEB-DESIGNER

WEB-DEVELOPER

WEB-GURU

WEB-JAM

WEB-PAGE

WEBCASTING

WEBHIPPIE

WEBTV

WEG

WELCOME-PAGE

HOWWHERE

WIDEBAND

WINDOWS-NT

WINSOCK

WINZIP

WTF

WTG

WYRN

WYS

WYT

X-WINDOW-SYSTEM

XML

XMODEM

LIST OF INPUTS

Y2K

YA

YAHOO!

YAHOO!GANS!

YDKM

YECCH!

YELLOW-BOOK

YERK

YOYO-MODE

YSYD

YTALK

YTTT

ZINE

ZINE.NET

AAF

ABSOLUTE-ADDRESS

ACAP

ACCESS

ACM

ZIP

ACUSTIC-COUPLER

ACROBAT

ACRONYM

ACTIVE-SERVER-PAGES

ACCESS-CODE

ADO

ACTIVEX-DATA-OBJECTS

ADSL

ASYMMETRIC-DIGITAL-SUBSCRIBER-LINE

MAE

MAIL-MERGE

ACTIVE-MOVIE

ACTIVEX

ACTIVEX-CONTROL

ADA

ADDRESS

AGENT

ALERT-BOX

ALGORITHM

ALIAS

ALPHANUMERIC

ALTA-VISTA

ALTAVISTA

LIST OF INPUTS
ANIMATED-GIF
ANONYMOUS-FTP
ANSI
APACHE
APACHE-WEB-SERVER
APPLET
APPLICATION-SHARING
ARCHIE
ARCHIVE
ARCNET
ARP
ADDRESS-RESOLUTION-PROTOCOL
ARPANET
ARTIFICIAL-INTELLIGENCE
AI
ASCII
ASYNCHRONOUS
ATM
ASYNCHRONOUS-TRANSFER-MODE
ATTACHMENT
ATTENUATION
AUTHENTICATION
AUTHORING-TOOL
AUTHORWARE
AUTHORIZATION
AUTO-ANSWER
AUTO-REDIAL
AWT
ABSTRACT-WINDOWING-TOOLKIT
MARQUEE
MASTER/SLAVE
MAU
MAILBOX
MAIL-BOX
MAILING-LIST-SERVER
MAJORDOMO
MAN
METROPOLITAN-AREA-NETWORK
MAP
MAPI
MBPS
MEDIA
META

LIST OF INPUTS

META-TAG

MICROSOFT

MIDDLEWARE

MIME

SERVER

SERVERS

MIPS

MMX

MNP

MODERATED-NEWSGROUP

MODULAR-ARCHITECTURE

MODULATE

MONITOR

MOO

MOORES-LAW

MORPHING

MOSAIC

MOTION-JPEG

MOUSE

MOZILLA

MP

MP3

MPEG

MPOA

MPP

MSN

MTU

MUCK

MULTICAST

MULTIPLEX

MULTIPLEXER

MULTITHREADING

MUSH

B-CHANNEL

BABY-AT

BETA-TEST

BGP

BACKGROUND

BACKLIGHTING

BACKPLANE

BACKSLASH

BACKSPACE

BACKSPACE-KEY

LIST OF INPUTS

BACKUP
BACKWARD-COMPATIBLE
BAD-SECTOR
BAK-FILE
BANDING
BAR-CHART
BASE-ADDRESS
BASEBAND-TRANSMISSION
BASELINE
BCD
BAT-FILE
BATCH-FILE
BATCH-PROCESSING
BATTERY-PACK
BAUD
BAY
BELL-103
BELL-AND-WHISTLES
BENCHMARK
BIG-ENDIAN
BILEVEL-PRINTER
BINHEX
BIOMETRICS
BIOS
BISYNC
BIT-BLOCK-TRANSFER
BINARY-COMPATIBLE
BINARY-FILE
BINARY-FILE-TRANSFER
BINARY-FORMAT
BINARY-TREE
BIND
BOMB
BOOLEAN-EXPRESSION
BITWISE-OPERATOR
BLEED
BLOATWARE
BLOB
BLOCK
BNC-CONNECTOR
BOOLEAN-LOGIC
BOOTP
BRI

LIST OF INPUTS
CAD/CAM
CADDY
CAE
CALCULATOR
BRIDGE
BROADCAST
BROUTER
BUBBLE-MEMORY
BUFFER
BRUST-MODE
BUS
CABLE-MODEM
CALENDAR

LIST OF OUTPUTS

Seen above are the list of inputs, which when given outputs are being generated by the computer. The outputs that are generated by the computer, listed here will be presented to the user both by textual version and through vocal version by the computer. There are no rules or restrictions applied to the output that is generated. The output is as such given to the user in the two forms.

LIST OF OUTPUTS
The top level of a hierarchical network. The main pipes along which data is transferred. The "Internet backbone" is sometimes referred to, though it doesn't exist.
1. The amount of information or data that can be sent over a network connection in a given period of time. Bandwidth is usually stated in bits per second (bps), kilobits per second (kbps), or megabits per second (mps). 2. The technical meaning is generalized in hacker slang. Individuals are said to be "high bandwidth" if they are able to process large volumes of information in short periods of time.
Spoken name for an exclamation point, used in old-style UUCP addresses to delimit the steps in a path from one site to another.
Repeated bounce messages, usually due to mail server or gateway errors, which cause significant annoyance
Acronym for Beginners All-purpose Symbolic Instructional Code. An easy-to-learn, highly flexible computer language invented at Dartmouth University. Different versions of BASIC run on various operating systems. Since each version has its own peculiar quirks, a BASIC program written in one version may not be compatible with another version. The "Hello World!" program in BASIC: 10 PRINT "Hello World!" 20 END
An open computer system that members can dial into in order to send email, join discussion groups, and download files. Since the 1970s, BBS's have provided an early means for home users to get online. Originally, BBS's were freestanding local systems, but now many provide access to Internet email, telnet, FTP, and other Internet services.
Abbreviation for Blind Carbon Copy. To bcc: an email message to someone is to send them a copy of the email message without the knowledge of the person to whom the email message is addressed to.
A version of an application that is made available prior to the official release for the purposes of testing.
A detailed and sometimes authoritative reference book covering a particular operating system, platform, or application. Originally, this was used generically to describe fundamental source books; more recently, it has been embraced by computer book publishers as a marketing ploy.
Mathematical base 2, or numbers composed of a series of zeros and ones

LIST OF OUTPUTS

Since zero's and one's can be easily represented by two voltage levels on an electronic device, the binary number system is widely used in digital computing.

Stands for binary digit. A bit is either on or off and is represented by "1" or "0". A collection of bits are put together to form a byte.

Acronym for Because It's Time NETwork. An obsolete network used by the academic and research community for email, mailing lists, and file transfers. It is distinct from the Internet but connected to it through email and news gateways.

A routine that allows you to save a reference to a site or page that you have already visited. At a later point in time, you can use a bookmark to return to that page. It commonly refers to a feature of Netscape Navigator (a web browser) that allows you to collect and organize bookmarks of your favorite web sites.

To start up or reset a computer. When a computer is booted, a bootstrap routine is automatically executed that looks for and loads the operating system. A cold boot is when the computer is powered up from an off state. A warm boot occurs when an already turned on computer is rebooted.

The return of a piece of email because it could not be delivered to the specified address.

A notification message returned to sender indicating that an email message could not be delivered. Usually the message is automatically generated by the Postmaster at the recipient's site, sometimes with an indication of what went wrong. The most common problem is an incorrect address, but sometimes email sent via the Internet fails for no apparent reason.

A computer.

A feature of some email and newsgroup reader applications that screens out incoming messages from those whose correspondence is not valued.

Abbreviation for bits per second. A measurement of the number of bits of information that can be sent over a network connection.

An application used to view and navigate the World Wide Web and other Internet resources.

A catch phrase that refers to the battles between Netscape and Microsoft for dominance of the web browser market. Both sides seek to maximize their product's marketshare and mindshare in cyberspace. The battles are marked by short product development cycles, publicity campaigns, provocative public statements, appeals for federal intervention, and a general desire to crush the other side.

Abbreviation for By The Way.

A problem with computer software or hardware that causes it to malfunction or crash.

(abbreviation: BBS) An open computer system that members can dial into in order to send email, join discussion groups, and download files. Since the 1970s, BBS's have provided an early means for home users to get online. Originally, BBS's were freestanding local systems, but now many provide access to Internet email, telnet, FTP, and other Internet services.

An open computer system that members can dial into in order to send email, join discussion groups, and download files. Since the 1970s, BBS's have provided an early means for home users to get online. Originally, BBS's were

LIST OF OUTPUTS

freestanding local systems, but now many provide access to Internet email, telnet, FTP, and other Internet services.

An open computer system that members can dial into in order to send email, join discussion groups, and download files. Since the 1970s, BBS's have provided an early means for home users to get online. Originally, BBS's were freestanding local systems, but now many provide access to Internet email, telnet, FTP, and other Internet services.

A collection of bits to form a binary number. The size of bytes varies from system to system. On the original home computers, bytes were composed of 8 bits. Now many computers operate with bytes that are 32 bits long.

The name of a sophisticated computer language commonly used for the creation of professional-grade applications. C is distinguished from other computer languages by its usage of pointers, variables that point at locations in memory. It was invented by Dennis Ritchie in the early 1970s.

This is the C version of "Hello World":

```
#include "stdio.h"
main() {
printf("Hello World!");
}
```

A computer language based on C that uses objected-oriented programming design principles.

A series of reply posts to a USENET message, each adding a trivial or nonsense theme to the corpus of previous replies. Some consider this art; there is a USENET newsgroup devoted to propagating this self-expressive form (alt.cascade).

Abbreviation for Carbon Copy. To cc: an email message to someone is to send them a copy of the email message.

1. Abbreviation for compact disk.

2. Abbreviation for change directory, a command in both the UNIX and DOS operating systems that assists in navigating a hierarchical directory structure.

(abbreviation: CPU) The main silicon chip that runs a computer's operating system and application software. It performs a computer's essential mathematical functions and controls essential operations. Intel CPUs in PCs were numbered by generation: 286, 386, 486. This numbering scheme was abandoned by Intel with the Pentium CPU, which would be the 586, and the Pentium Pro, which would be the 686. Motorola PowerPC CPUs have maintained a consistent numbering convention: 601, 603, 604.

A standard used by programmers that allows their programs to interact with the World Wide Web. CGI scripts can be written in many computer languages, but Perl and C are the most common.

A form of interactive online communication that enables typed conversations to occur in real-time. When participating in a chat discussion, your messages are instantaneously relayed to other members in the chat room while other members' messages are instantaneously relayed to you.

A transcript of a chat session.

A condition characterized by excessive anxiety about sending chat messages. The chatter with chatter's block feels that every message he types in a chat must be perfectly worded and perfectly timed. By striving to meet these standards of perfection, the chatter constantly erases and

LIST OF OUTPUTS

rewords his messages, often deciding not to send the messages at all. The cure for chatter's block is to relax. Everyone in a chat is expressing (and typing) ideas quickly and no one will blink an eye at the odd typo.

A mathematical calculation applied to the contents of a packet before and after it is sent. If the "before" calculation does not match the "after" calculation, there were errors in the transmission.

The turnover of users on an online service, especially after the expiration of a free trial period.

Describes a computer taking a long time to process a particular operation. When a computer is churning, it may seem to be doing nothing.

Any computer system that requests a service of another computer system. A workstation requesting the contents of a file from a file server is a client of the file server.

A configuration in which one computer, designated as a "server", sends information to a number of other "client" computers.

A holding area that temporarily stores information copied or cut from a document. Both the Macintosh and Windows operating systems support this feature.

The process by which a node on a network monitors the communications line to determine when a collision (two nodes attempting to transmit at the same time) has occurred.

A computer network that supplies its members with access to chat rooms, bulletin boards, and other online content on a monthly fee basis. Commercial online services include America Online, CompuServe, The Microsoft Network, and Prodigy. In addition to their own proprietary content, most commercial online services also provide access to the Internet.

(abbreviation: CGI) A standard used by programmers that allows their programs to interact with the World Wide Web. CGI scripts can be written in many computer languages, but Perl and C are the most common.

To have a computer translate code written in a computer language into an executable form.

A technology that reduces the size of a file. Compression programs like WinZip and UNIX compress are valuable to network users because they help save both time and bandwidth.

A commercial online service.

Knowledge about and the ability to learn about computers.

A state occurring in a part of a network when the message traffic is so heavy that it slows down network response time.

When two computers have established a path through which the exchange of information can occur.

Small files that are downloaded to your computer when you browse certain web pages. Cookies hold information that can be retrieved by other web pages on the site. Some cookies are programmed with an expiration date so that they are automatically deleted after a period of time.

The phenomena of computer companies cooperating with their competitors on a project-by-project basis.

A software lock placed on a computer program by its developer to thwart piracy. This preventative measure was widely used in the mid-1980s but later abandoned by many developers because of numerous customer complaints.

LIST OF OUTPUTS

The main silicon chip that runs a computer's operating system and application software. It performs a computer's essential mathematical functions and controls essential operations. Intel CPUs in PCs were numbered by generation: 286, 386, 486. This numbering scheme was abandoned by Intel with the Pentium CPU, which would be the 586, and the Pentium Pro, which would be the 686. Motorola PowerPC CPUs have maintained a consistent numbering convention: 601, 603, 604.

A malicious hacker who breaks (or cracks) the security of computer systems in order to access, steal, or destroy sensitive information. "Hacker" is often incorrectly used instead of cracker, especially by the media.

To send an attached file via email.

To post a single message simultaneously to multiple newsgroups or discussion groups. Gratuitous cross-posting is considered poor Netiquette.

To efficiently process large amounts of information. A number cruncher, for example, is a routine or device optimized for and dedicated to processing numbers.

An original NeXT computer. The motherboard and drives for this machine are packed into a 12-inch matte black cube.

Pronounced "See you, See me," CU-SeeMe is a publicly available videoconferencing program developed at Cornell University. It allows anyone with audio/video capabilities and an Internet connection to videoconference with anyone else with the same capabilities. It also allows multiple people to tie into the same videoconference.

1. A subgenre of science fiction inspired by William Gibson's 1982 novel "Neuromancer".

2. A lifestyle characterized by computer games, Internet surfing, and large doses of attitude.

1. The place where computer networking hardware, network software, and people using them converge. Defined by John Perry Barlow as the place where a telephone call happens.

2. The prefix "cyber" is often combined with other words, as in "cyberpunk".

In the UNIX operating system, a background process that lies dormant waiting to perform some useful task. The sendmail daemon, for example, continually runs but becomes active only when email is sent or received.

A string of characters used to mathematically encode a message so that it can only be read by someone in possession of another related key.

(abbreviation: DES) A popular, standard encryption scheme.

Loosely, any aggregation of data; usually a large collection of data that has been formatted by some user-defined standard.

A telephone line that is reserved for the singular purpose of providing a data connection between two computers.

A telephone line that is reserved for the singular purpose of providing a data connection between two computers.

An optional email feature that notifies you when your email message has been delivered to its recipient.

To come out of online lurking mode; to actively participate in a online discussion after a period of just watching or lurking. This term is derived from episodes of Star Trek that feature Klingon warships that can hide ("cloak") or appear ("decloak") at will.

A popular, standard encryption scheme.

LIST OF OUTPUTS

1. A temporary connection between computers established over a telephone line.

2. To establish a temporary connection to another computer.

The digital elite. Derived from "literati".

A database system which looks up host IP addresses based upon domain names. For example if you ask for "www.thisismyhost.com" it will return "123.45.67.89". Copies of the Domain Name System are distributed through the Internet.

The official name of a computer connected to the Internet. Domain names are derived from a hierarchical system, with a host name followed by a top-level domain category. The top-level domain categories are com (for commercial enterprises), org (for non-profit organizations), net (for network services providers), mil (for the military), and gov (for government). Some Internet domain names include the computer server name, other sub-domains, and/or country abbreviations (e.g., us). Domain names act as easy-to-remember addresses for product or company information. As such, they are often subject to disputes between competing commercial interests. Most domain names are assigned by the InterNIC.

(abbreviation: DNS) A database system which looks up host IP addresses based upon domain names. For example if you ask for "www.thisismyhost.com" it will return "123.45.67.89". Copies of the Domain Name System are distributed through the Internet.

Prejudice against people on the basis of their Internet address. For example, adopting a dismissive attitude towards anyone who posts from a commercial online service. "Why should anyone listen to you, you're posting from: aol.com!"

Acronym for Disk Operating System. A dated operating system for PCs that features a command line interface.

To transfer data from a larger "host" system to a smaller "client" system's hard drive or other local storage device.

Monetary charges associated with downloading a file from a commercial online service. This method of information exchange is not very popular.

To move an image or a window from one place on the screen to another by grabbing it and pulling it to a new location.

1. A large mass of information that shows the exact contents of computer memory at a particular time. Dumps are used by programmers in the debugging process.

2. To send a large volume of information to somewhere specific such as a printer or a screen.

3. A backup or a backup file created by the UNIX dump utility.

Electronic money designed to be used over a network or stored on cards similar to credit cards. Ecash is still more of an idea than a practical reality, largely due to security concerns.

An electronic form that is filled out by a user and sent over a network. They are typically used to place orders or provide feedback. Eforms can be placed on web pages or in Java applets and usually contain text boxes, buttons, and other components.

To quote their mission statement, "The Electronic Frontier Foundation is a non-profit civil liberties organization working in the public interest to protect privacy, free expression, and access to public resources and information online, as well as to promote responsibility in new media." Their WWW site is

LIST OF OUTPUTS

located at <http://www.eff.org>

Electronic mail, the computer-based exchange of mail.

An electronic mail address. Email addresses follow the formula: user-ID@domain-name.

A fictitious "place" where lost email is said to wind up.

The state of being completely overwhelmed by the amount of email one has received.

A cute sideways face created by using special characters on the keyboard. Used to express emotions without words. For example, this winking face ;-) indicates "I'm joking", this sad face :-(expresses sadness or "I'm sulking". If this makes no sense, turn your head sideways and look again. Also known as a "smiley".

A procedure that renders the contents of a message or file unintelligible to anyone not authorized to read it. PGP (Pretty Good Privacy) is a commonly used encryption program.

A standard method of connecting computers to a local area network using coaxial cable invented by Robert Metcalfe at Xerox PARC in the early 1970s.

An Internet search engine at <http://www.excite.com>

To process or run a computer program.

A viewing audience for a WWW site.

In a facilitated chat, a host or facilitator controls the messages that appear on the chat screen. Usually used when there is a guest speaker. Facilitated chats provide an orderly environment for the guest speaker and ensure that she is not overwhelmed with dozens of questions all being asked at once.

Acronym for Frequently Asked Questions. A reference document created for particular topic or group that answers to common beginners' questions. It is considered poor Netiquette to ask a question without first reading the FAQ.

A worldwide hobbyist network of personal computers started in 1984 that exchanges mail, discussion groups, and files.

Reducing the size of files for ease of storage and transfer. WinZip is an example of an application that compresses files; the UNIX compress utility is another.

The process of moving or transmitting a file from one location to another, as between two programs or from one computer to another.

(abbreviation: FTP) . An Internet protocol that enables you to transfer files between computers on the Internet.

A UNIX utility that reports information about other users who have UNIX accounts. Finger can tell you, for example, where and when a person last logged in to the system. It can also be used on a single host or across the Internet.

A set of security procedures that separates and protects data on a LAN from crackers who might access the LAN from the Internet.

A public post or email message that expresses a strong opinion or criticism. Flames can be fun when they allow people to vent their feelings, then return to the topic at hand. Others are simply insulting and can lead to flame wars.

An inflammatory post that is designed to provoke a flame war or flame responses.

Notifiers that surround a flame message and let readers know that the message they are reading is a flame. Although you don't see these as much

LIST OF OUTPUTS

as you used to, they would most commonly be used by an individual known to a particular online group who wishes to do a little ranting and then return to the topic at hand. Note that the original usage of "flame on" was derived from Marvel Comics' Human Torch character.

A series of public posts in which people flame one another rather than contribute useful information.

A feature of America Online that automatically performs online tasks at a designated time. Flash sessions are often used to send/receive email and download large files.

1. As a noun, an ASCII text file consisting of records of a single type, in which there is no embedded structure information governing relationships between records.

2. As an adjective, describes a flattened representation of a database as single file from which the structure could implicitly be rebuilt.

A typographic style used to display or print characters. Times Roman, Courier, and Helvetica are three examples.

A topically-focused discussion group or area. From the traditional Roman forum – a community area where ideas and proposals are discussed.

Sending an email message or post from one person to a third party.

1. An HTML feature that allows web designers to segment the window of a web browser into distinct sections.

2. A single complete picture in a moving picture sequence.

A community-based bulletin board system that serves a local geographic community on a non-profit basis.

Shareware that is openly available to the public without the requirement of user registration fee.

To be totally destroyed or otherwise unusable, as in "my hard drive is fried" or "the network is totally fried."

An Internet protocol that enables you to transfer files between computers on the Internet.

Abbreviation for Fear, Uncertainty, and Doubt. A set of sales tactics employed by market leaders to cast aspersion on competing products. Computer products are often purchased on the basis of perceived market leadership because no one wants to get stuck with a losing product that might not be supported in the near future. The usefulness of using FUD to confuse a market is epitomized by the apocryphal saying, "No one ever got fired for buying IBM." A good example of FUD is Microsoft's tactic of pre-announcing products far in advance of their actual availability. All of a sudden the market for competing products evaporates as customers await a dominating Microsoft product.

Abbreviation of For What It's Worth.

Abbreviation of For Your Information.

1. A string of unwanted, meaningless, or unintelligible characters produced by error.

2. Incorrect input to a computer.

An imperative issued as a response to someone who is complaining about not being able to get work done on an obsolete, single-tasking, graphical, otherwise lame computer. Generally, UNIX systems are considered to be real computers in opposition to Windows-based systems.

A file format for storing images, commonly used on the web.

LIST OF OUTPUTS

Acronym for Garbage In Garbage Out. Usually said in response to complaints that a program didn't "do the right thing" when given bad or flawed input.

A small malfunction.

The word associated with a forum or area on a commercial online service that allows you to get to that place quickly.

"Remember the Human", the rule upon which all Netiquette is based. Often in electronic communications, it is easy to forget that you are communicating with real people, not just networks of computers.

A menu-driven program developed at the University of Minnesota that helps you locate and retrieve information on the Internet.

(abbreviation: GIF). A file format for storing images, commonly used on the web.

1. To rapidly scan a large volume of information looking for a particular string or pattern.

2. A UNIX command used to scan a file or group of files for a matching search string or pattern.

Abbreviation for Graphical User Interface. A set of screen presentations and metaphors that utilize graphic elements such as icons in an attempt to make an operating system easier to use. Unlike UNIX and DOS, which feature a command line interface, the Windows and the Macintosh operating systems present graphical environments for input and output.

An expert who acts as a knowledge resource for others and who is generally venerated by the people whose problems he or she solves.

An expert programmer who likes to spend a lot of time figuring out the finer details of computer systems or networks, as opposed to those who learn only the minimum necessary.

A nickname used in online communications.

1. An unexpected halt of a computer, usually while running an application program. A hung machine is characterized by a total lack of response from the mouse and keyboard. The user can almost never effect a recovery except by turning the computer off and restarting it.

2. In the construction "hang off", to attach a peripheral device to a computer via a cable. "I'm going to hang another hard drive off my home box."

A function or capability that is hardcoded into a system. Generally, anything that can not be modified or customized.

The classic first demonstration program used when teaching a new language. The goal of a "Hello World!" program is simply to print the words "Hello World!" to the screen.

A single user accessing a single file from a web server. A unit of measure often used erroneously to evaluate the popularity of a web site.

Ubiquitous online disputes that never end. Common holy wars are fought over Macintosh vs. Windows, UNIX vs. Windows NT, and everyone vs. Microsoft, as well as over societal issues such as abortion, gun control, and pornography.

A web page that is typically the main source of information about a particular person, group, or concept. Many people on the web create home pages about themselves for fun; these are also known as vanity pages.

To be totally destroyed or otherwise unusable, as in "my hard drive is hosed" or "the network is totally hosed."

LIST OF OUTPUTS

1. A computer that allows users to communicate with other host computers on a network.
2. A chat term for someone who is managing a chat. Hosts often act as referees and have the power to turn participants into spectators and vice versa.

An Internet search engine at <http://www.hotbot.com>

The tag-based ASCII language used to create pages on the World Wide Web.

The protocol used by the World Wide Web to transfer HTML files.

Marketing messages that overstate the truth (as in "Don't believe the hype!").

A highlighted word or picture within a hypertext document that when clicked takes you to another place within the document or to another document altogether.

Text that includes links or shortcuts to other documents, allowing the reader to easily jump from one text to related texts, and consequentially from one idea to another, in a non-linear fashion. Coined by Ted Nelson in 1965.

(abbreviation: HTML). The tag-based ASCII language used to create pages on the World Wide Web.

(abbreviation: HTTP). The protocol used by the World Wide Web to transfer HTML files.

A small graphic image that represents a file or application and when clicked upon produces a programmed result. Use of this mnemonic convention originated at Xerox PARC and was subsequently popularized by the Apple Macintosh. Producing an effective icon is non-trivial because of size and color restraints.

A skillful designer who elevates icon design to an art form.

Posing as someone else. Posting anonymously or pseudonymously, usually with the intent to deceive.

Abbreviation for In My Humble Opinion.

A folder where you receive incoming mail.

A variant of information superhighway.

An unimplemented proposal by Vice President Al Gore to wire the US for hundreds of cable television channels. Now synonymous with the Internet.

An Internet search engine at <http://www.infoseek.com>

To load and configure a piece of software on a computer

(abbreviation: ISDN). A technology offered by telephone carriers that allows for the rapid transfer of voice and data.

A worldwide network of networks that all use the TCP/IP communications protocol and share a common address space. First incarnated as the ARPANET in 1969, the Internet has metamorphosed from a military internetwork to an academic research internetwork to the current commercial Web, file transfer, and Internet Relay Chat. The Internet is experiencing tremendous growth in the number of users, hosts, and domain names. It is gradually subsuming other media, such as proprietary computer networks, newspapers, books, television, and the telephone. Also known as "the net", "the information superhighway", and "cyberspace".

A free web browser application from Microsoft.

(abbreviation: IRC). A chat network that operates over the Internet. Original

LIST OF OUTPUTS

evolved from the UNIX talk program, IRC is similar to the chat systems found on commercial online services.

(abbreviation: ISP). 1. A business that delivers access to the Internet, usually for a monthly fee. PSI, UUNET, and Netcom are examples of established ISPs but there are thousands of smaller ones all around the world.
2. Any business that provides Internet services such as web sites or web site development.

(abbreviation: ISOC). To quote its home page at <http://info.isoc.org>: "The Internet Society is a non-governmental International organization for global cooperation and coordination for the Internet and its internetworking technologies and applications. The Society's individual and organizational members are bound by a common stake in maintaining the viability and global scaling of the Internet. They comprise the companies, government agencies, and foundations that have created the Internet and its technologies as well as innovative new entrepreneurial organizations contributing to maintain that dynamic."

The InterNIC is the entity that controls the registration of most domain names on the Internet. The InterNIC is a cooperative activity between the National Science Foundation, Network Solutions, Inc. and AT&T. Its home page is at <http://internic.net/>

The ability of software and hardware on multiple machines from multiple vendors to communicate meaningfully.

A private network that uses Internet-related technologies to provide services within an organization.

A string of four numbers separated by periods (such as 111.22.3.444) used to represent a computer on the Internet. The format of the address is specified by the Internet Protocol in RFC 791. When a PC accesses the Internet through an ISP, it sometimes receives a temporary IP address.

A chat network that operates over the Internet. Originally evolved from the UNIX talk program, IRC is similar to the chat systems found on commercial online services.

A technology offered by telephone carriers that allows for the rapid transfer of voice and data.

To quote its home page at <http://info.isoc.org>: "The Internet Society is a non-governmental International organization for global cooperation and coordination for the Internet and its internetworking technologies and applications. The Society's individual and organizational members are bound by a common stake in maintaining the viability and global scaling of the Internet. They comprise the companies, government agencies, and foundations that have created the Internet and its technologies as well as innovative new entrepreneurial organizations contributing to maintain that dynamic."

1. A business that delivers access to the Internet, usually for a monthly fee. PSI, UUNET, and Netcom are examples of established ISPs but there are thousands of smaller ones all around the world. 2. Any business that provides Internet services such as web sites or web site development.

To log in to a machine or connect to a network. Derived from cyberpunk fiction.

An object oriented programming language created by Sun Microsystems. Java is a device independent language, meaning that programs compiled in Java can be run on any computer. Java programs can be run as a free-

LIST OF OUTPUTS

standing application or as an applet placed on a web page. Applets written in Java are served from a web site but executed on the client computer. Java applets have a built-in security feature which prevents them from accessing the file system of the client computer.

Here is the Java version of "Hello World!":

```
class HelloWorld {  
public static void main (String args[]) {  
System.out.println("Hello World!");  
}  
}
```

A scripting language that allows lines of Java code to be inserted into HTML scripts.

(abbreviation: JPEG)

An image compression standard for still photographs that is commonly used on the web.

Joint Photographic Experts Group An image compression standard for still photographs that is commonly used on the web.

A protocol used for transferring files over a dial-up connection that is commonly used on BBS systems.

A file used by some USENET reading programs that filters out unwanted messages, usually from a particular author or on a particular subject. If you add someone to your kill file, you arrange for the person to be ignored by your news reader. Originally from Larry Wall's `rn` program.

A quick fix to a problem that places expediency over elegance. Pronounced "klooj". Variant spelling is "kludge".

A person who understands the finer details of computer networking.

A artificially intelligent computer program that automates the search for animation.

Local Area Network. A group of computers at a single location (usually an office or home) that are connected by phone lines or coaxial cable.

A permanently established connection between computers over a dedicated phone line which is leased from a telephone carrier.

Static over a telephone line that interferes with network communications.

A highlighted word or picture within a hypertext document that when clicked bring you to another place within the document or to another document altogether.

An automated mailing list distribution system. List servers maintain a list of email addresses to be used for mass emailing. Subscribing and unsubscribing to the list is accomplished by sending a properly formatted email message to the list server.

(abbreviation: LAN)

A group of computers at a single location (usually an office or home) that are connected by phone lines or coaxial cable.

Abbreviation for Laughing Out Loud.

A programming technique of creating a series of repeating actions. Usually, there is some event specified as part of the loop which triggers the ending of the repetitious action.

To hang out in an area without directly participating. For example, when you're new to a discussion group or chat room, it's a good idea to lurk and become familiar with its scope and general rules before posting. Lurking is

LIST OF OUTPUTS

perfectly acceptable in cyberspace; the negative connotations of the standard English usage does not apply.

Commonly used for "computer".

A program in the form of a series of binary codes that are understandable by the CPU. 99.9% of the time programmers write their code in another "higher level" programming language which in turn translates their code into machine language.

The act of sending massive amounts of email to a single address with the malicious intent of disrupting the system of the recipient. Mailbombing is considered a serious breach of Netiquette and is probably illegal.

A discussion group that occurs via mass email distributions. Mailing lists are usually maintained by individuals utilizing list server software. List servers maintain a list of email addresses to be used for the mailing list. Subscribing and unsubscribing to the list is accomplished by sending a properly formatted email message to the list server. There are two types of mailing lists: moderated and unmoderated. To send a message to an unmoderated list, you email it to the list server which automatically emails your message to every name on the list. To send a message to a moderated list, you email it to the mailing list's moderator who would then send it on to the list server for distribution.

A high-level computer designed for intensive computational tasks and used by large corporations. Mainframes are often shared by multiple users connected to the computer via terminals. Originally refers to the cabinet containing the CPU of a room-sized batch-processing machine.

A superset of the Internet that includes all networks and computers that can exchange email.

Short for Multicast Backbone on the Internet, Mbone is an extension to the Internet to support IP multicasting -- two-way transmission of data between multiple sites. The TCP/IP protocol used by the Internet divides messages into packets and sends each packet independently. Packets can travel different routes to their destination which means that they can arrive in any order and with sizable delays between the first and last packets. In addition, each recipient of the data requires that separate packets be sent from the source to the destination. This works fine for static information, such as text and graphics, but it doesn't work well for real-time audio and video.

With Mbone, a single packet can have multiple destinations and isn't split up until the last possible moment. This means that it can pass through several routers before it needs to be divided to reach its final destinations. This leads to much more efficient transmission and also ensures that packets reach multiple destinations at roughly the same time. The Mbone is an experiment to upgrade the Internet to handle live multimedia messages. Mbone servers have special Class D IP addresses. As of March 1997, there were more than 3,000 Mbone servers on the Internet.

The Mbone was developed by Steve Deering at Xerox PARC and adopted by the Internet Engineering Task Force (IETF) in March 1992.

A state of complete network overload that grinds all traffic to a halt.

A list of options, each of which performs a desired action such as choosing a command or applying a particular format to a part of a document. Menu's are commonly used in graphical interfaces.

The above list is just a sample list of outputs. The computer will generate a lot of terms as outputs for the user given inputs.

PERFORMANCE REQUIREMENT

INTRODUCTION

In this module the performance requirements of the project are explained in detail. The performance requirement is the part in which the performance of the project is explained. This is one of the prelude steps in the forwarding of the project development. The performance of the predetermined functions included in the previous requirement phase are analyzed and checked for their requirements.

After the inclusions of the functions in the functional requirement phase, comes the performance requirement phase. In this phase we do analyze all the operation of those functions. In the following paragraphs various aspects of this phase.

SECURITY

The program has a better security added to it since it is developed in win32API environment. The program additively has security alerts even if any mishaps occur. The cause of problems due the installation of the program or the execution of the program has the probability less than one percent (1%).

The chance of creating blunders by the accompanying engines and other softwares are also very reduced; since they are the products of prestigious companies which have their own user policies.

The users if wishes to check their personal e-mail accounts, he will be secured by providing their passwords at runtime. This feature enables the users to have more security for their e-mails. As a result no intruders can read their e-mails.

Thus the security issue, as far as the project is concerned, can be guaranteed to be the best in optimistic requisites.

AVAILABILITY

The components that are required by the program for its execution are provided as supplementary during the installation of the program. Thus the necessary components are available to the user without any restriction.

Some of the components that are required for the execution of the program are **Microsoft® speech-recognition engine**, **Microsoft® text-to-speech conversion engine**, xvoice.dll, xlisten.dll etc., are supplied in addition with the program, and the user need not install them separately.

CAPACITY

The capacity of the program, database and accompanying installation binaries will need around 6 Megabytes of disk space. Thus the hard disk space required for installing the software will come around the same. If the necessary vicinity is not found in the hard disk, there might be installation problems and execution problems.

During the installation of the program itself the installing process checks the disk on which it is stored for necessary space so that no problems arise due to lack of space. Thus the capacity needed for the installation and execution of the program needs around 6 MB of disk space.

RESPONSE TIME

The time taken for generating responses for the user queries are very less. Though it is generally very less, the response time is normally machine dependent. A machine with a good disk space and a good amount of RAM, powered by a good processing speed will take very less response time in order to generate responses for the user queries.

For example a machine with 6 MB disk space, 16 MB RAM, powered with a 500 MHZ processor, will require a high response time than a machine with 10 MB disk space, 32 MB RAM, powered by a 400 MHZ processor will take a less response time. Thus the response time taken by the program is machine dependent and cannot be predicted as can be done with other properties of the program.

DESIGN CONSTRAINTS

HARDWARE LIMITATION

The hardware requirements of the project are

- ✓ A half-duplex Sound card.
- ✓ Microphone.
- ✓ 16 MB RAM.
- ✓ 6 MB of free hard disk space.
- ✓ A CD-ROM drive for installation.
- ✓ A normal 101 keys keyboard.
- ✓ A mouse.
- ✓ A pair of Speakers (Or),
- ✓ Headphones or Earphones.

The hardware components listed above are the minimum requirements of the project to function. The recommended configuration of the system is

- A full-duplex Sound card.
- A close-talk microphone.
- 32 MB RAM.
- 7 MB of free hard disk space.
- A CD-ROM drive for installation.
- A normal 104 Windows keyboard.
- An INTELLI[®] mouse.
- A pair of amplified speakers.
- Amplified headphones or earphones.

The components are required according to the above configuration, for the efficient working of the program.

USER INTERFACE

The program has good user interfaces, which the user will be presented with when the program is executed. There are two ways of accessing any option that is available in the program. The first one is through the keyboard accessing and the next one is through the mouse clicking.

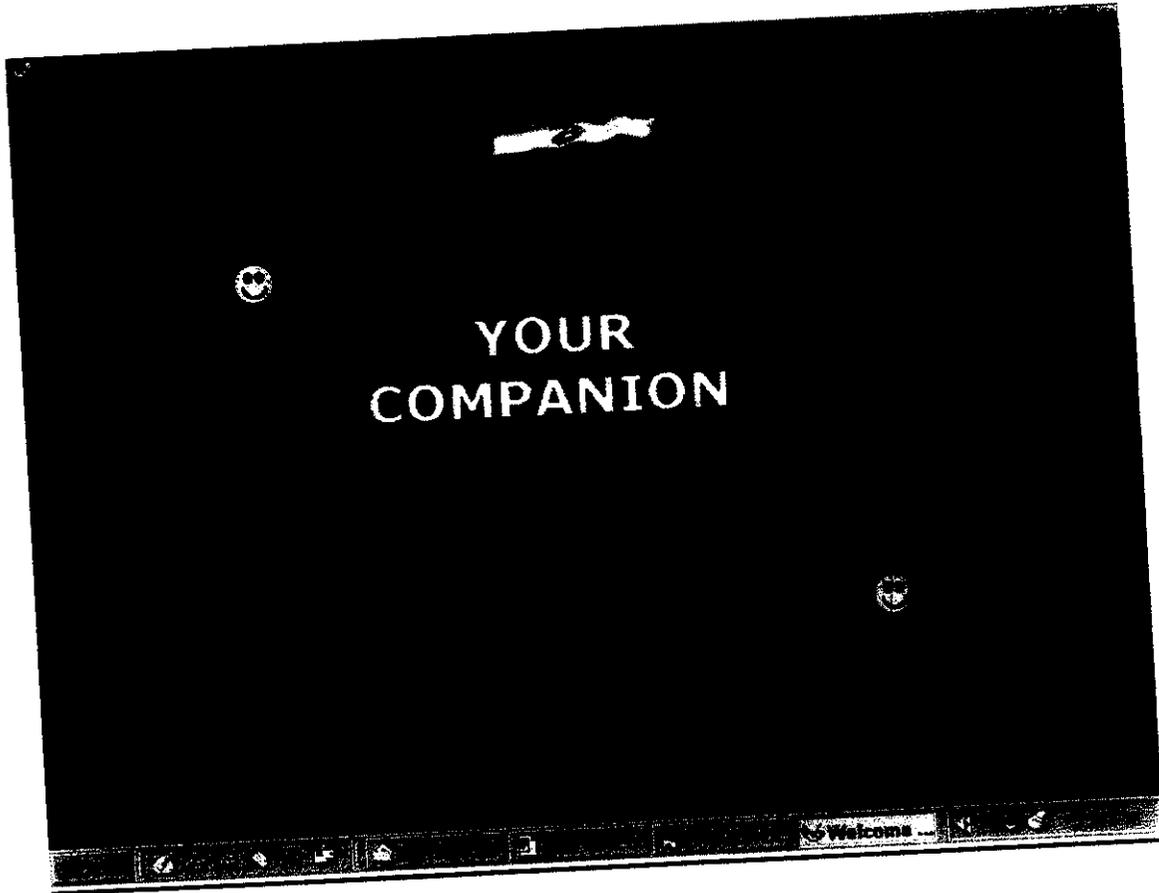
All the options and facilities that are available in the program are presented in pull down menus and buttons. The user can use both the menus and the buttons, which depends on his or her comfort. Providing user interfaces is one of the easiest ways to make the user feel comfortable while he is new to the program. Thus care is taken while designing the user interfaces for the program.

SCREEN INTERFACE

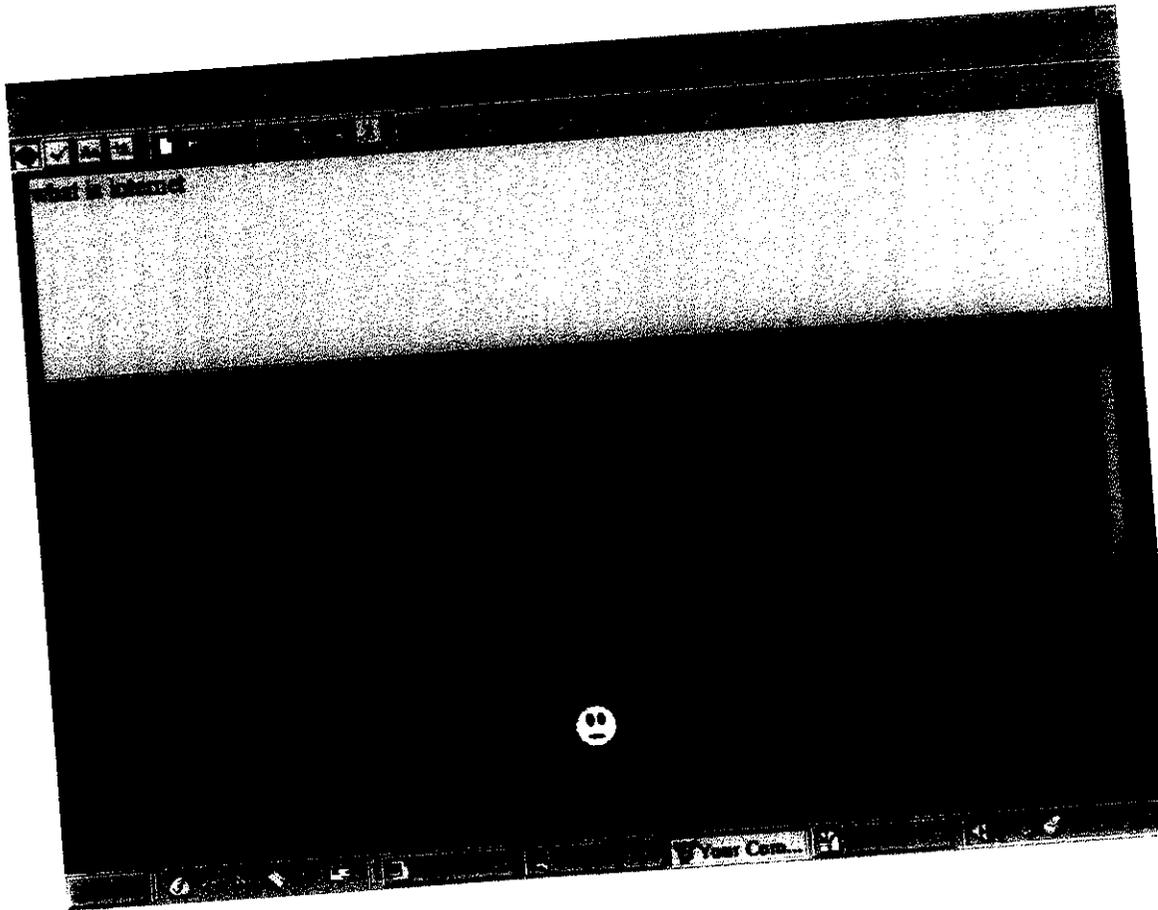
The project has two types of screen format. They are input screen format and output screen format.

The input screen is the one that is presented to the user when the program is in input mode and the user enters some input. In the project the following screen is the input screen.

STARTING SCREEN



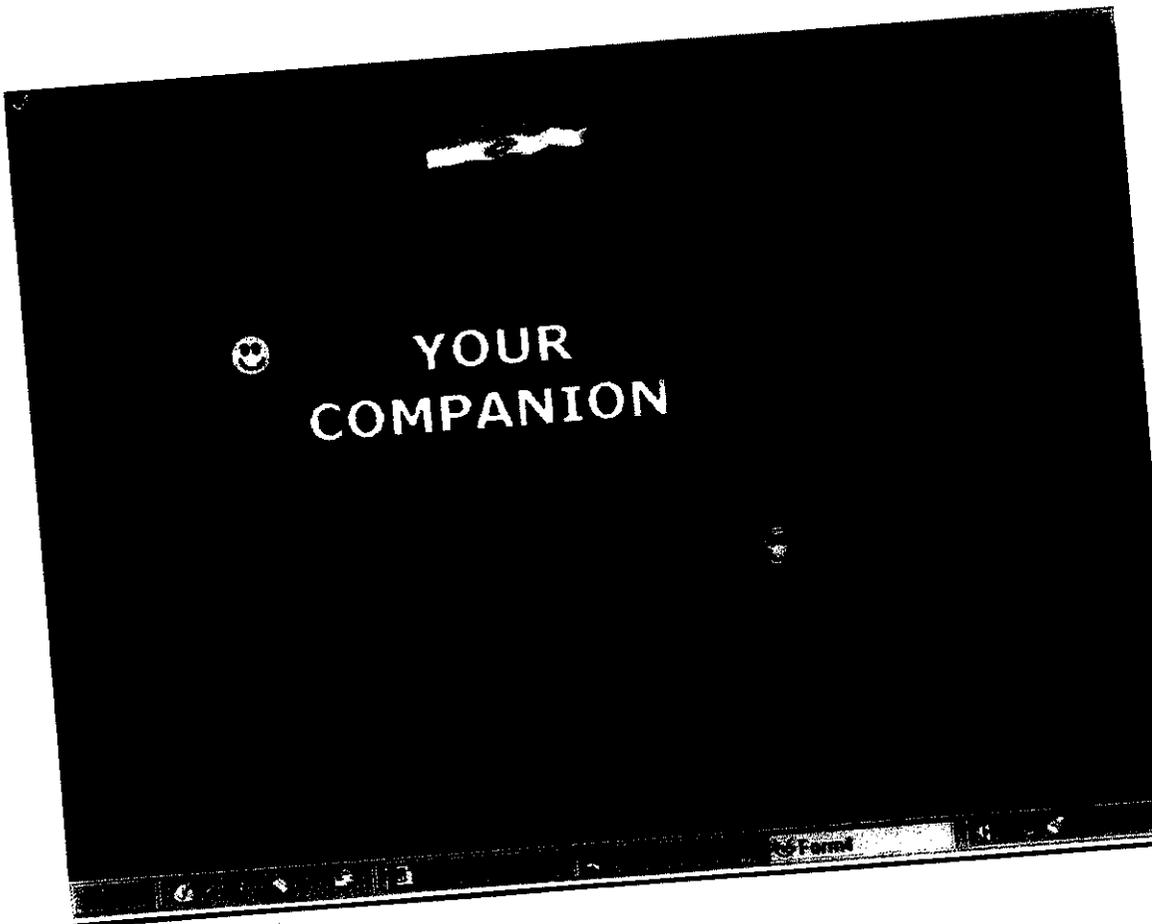
INPUT SCREEN



The output screen is the one that is presented to the user when the program is in output mode. This screen normally contains some results processed using the user inputs provided in the previous screen. In the project the following is the output screen.



EXIT SCREEN



APPENDIX

SAMPLE SOURCE CODE

The following source code is the sample source code for the project. There are five forms and a module according to which they are separated.

Source code in the MAIL READ FORM

Rem This sample will play a signon sound and read the read
the subject of
Rem newly arrived e-mail for network email users. It uses the
MAPI control to
Rem access email, so your email must be MAPI compliant.
Rem Microsoft Exchange is MAPI compliant. So is Microsoft
Outlook, Microsoft Outlook Express
Rem This project has not been tested on any other E-Mail
Clients for its functionality.
Rem so before using this software on your custom e-mail
clients,
Rem make sure that it is MAPI compliant.

Rem NOTIFYICONDATA is data type is needed for creating an
icon on the tray

Private Type NOTIFYICONDATA

cbSize As Long

hwnd As Long

uld As Long

uFlags As Long

```
uCallbackMessage As Long
hIcon As Long
szTip As String * 64
End Type
```

```
Rem constants for tray notifications
```

```
Private Const NIM_ADD = &H0
```

```
Private Const NIM_DELETE = &H2
```

```
Private Const WM_MOUSEMOVE = &H200
```

```
Private Const NIF_MESSAGE = &H1
```

```
Private Const NIF_ICON = &H2
```

```
Private Const NIF_TIP = &H4
```

```
Private Const WM_RBUTTONDOWN = &H204 'Button down
```

```
Private Declare Function Shell_NotifyIcon Lib "shell32" _
```

```
Alias "Shell_NotifyIconA" _
```

```
(ByVal dwMessage As Long, pnid As NOTIFYICONDATA) As
```

```
Boolean
```

```
Dim nid As NOTIFYICONDATA
```

```
Public gStarted As Integer
```

```
Public gMode As Integer
```

```
Public LastReceived As String
```

```
Private Const POLLTIME = 10000
```

```
Private Sub exit_Click(Index As Integer)
```

```
Rem Clean up the MM controls and tray icon before we exit
```

```
MMControl1.command = "Close"
```

```
MMControl2.command = "Close"
```

```
Shell_NotifyIcon NIM_DELETE, nid
```

```
End 'end execution
```

End Sub

```
Private Sub Form_Load()
```

```
Rem This applet's main UI is an icon on the tray.
```

```
nid.cbSize = Len(nid)
```

```
nid.hwnd = Form1.hwnd
```

```
nid.ulid = vbNull
```

```
nid.uFlags = NIF_ICON Or NIF_TIP Or NIF_MESSAGE
```

```
nid.uCallbackMessage = WM_MOUSEMOVE
```

```
nid.szTip = "eMail Notify" & vbNullChar
```

```
nid.hIcon = Form1.Icon
```

```
Shell_NotifyIcon NIM_ADD, nid
```

```
Rem initialize the multimedia controls to play our sounds
```

```
MMControl1.notify = False
```

```
MMControl1.Wait = True
```

```
MMControl1.Shareable = False
```

```
MMControl1.DeviceType = "WaveAudio"
```

```
MMControl1.FileName = App.Path + "\signon.wav"
```

```
MMControl1.command = "Open"
```

```
MMControl2.notify = False
```

```
MMControl2.Wait = True
```

```
MMControl2.Shareable = False
```

```
MMControl2.DeviceType = "WaveAudio"
```

```
MMControl2.FileName = App.Path + "\signoff.wav"
```

```
MMControl2.command = "Open"
```

```
Rem Dont except on errors with speech output..the user may  
have some other
```

Rem sounds playing. Ideally we would catch the errors and do
the notify later.
DirectSS1.SuppressExceptions = 1

Rem set the last mail recieved time to a date in the past.
LastRecieved = "1996/08/05 16:40"

Rem Our state machine startes at mode 0
gMode = 0
gStarted = 0
Timer1.Interval = POLLTIME

Rem Play the startup sound
MMControl1.command = "Prev"
MMControl1.command = "Play"
Rem control picks up at MMControl1_Done after the sound is
played, perhaps
End Sub

Rem this function should only be called from the timer so it
does not get reentered

Rem instead, to call this do Timer1.Interval = 1
Private Sub CheckForNewMail()

Rem This function is called occassionly to check if new mail
has arrived

Dim SayString As String

Rem Just a bit of a precaution..be sure not to start get errors
if we

Rem are called before the program is initialized.

If (gStarted <> 1) Then

Exit Sub

End If

Rem Mark that we are checking mail, so we won't re-enter the process

Rem while were doing it. We need to do this because the sounds and

Rem spoken words are played assynchronously.

Rem Get the new mail list..

MAPIMessages1.FetchSorted = False 'sort the way the user has their mailbox laid out

MAPIMessages1.Fetch

Rem If there is some mail..

tCount = 0

again:

If (MAPIMessages1.MsgCount) Then

Rem see if it is new mail, not just old mail marked as unread...

MAPIMessages1.MsgIndex = MAPIMessages1.MsgCount -

tCount - 1

If ((MAPIMessages1.MsgDateReceived) >= (LastRecieved))

Then

Rem remember the time of this mail..we wont read mail way older than this

Rem from now on, even if it is marked unread.

```
LastRecieved = MAPIMessages1.MsgDateReceived
Rem state change to reading mail..Play the sign-on sound
Rem control will go to MMControl1_Done when the sound is
finished.
```

```
gMode = 2
```

```
MMControl1.command = "Prev"
```

```
MMControl1.command = "Play"
```

```
Else
```

```
If (MAPIMessages1.MsgIndex = 0) Then
```

```
Rem we didn't do anything, so we are finished
```

```
Else
```

```
tCount = tCount + 1
```

```
GoTo again
```

```
End If
```

```
End If
```

```
Else
```

```
Rem we didn't do anything, so we are finished
```

```
End If
```

```
End Sub
```

```
Private Sub MMControl1_Done(NotifyCode As Integer)
```

```
If (gMode = 0) Then
```

```
Rem if we are just starting up, speak our text and control
goes to
```

```
Rem DirectSS1_AudioStop after the text is spoken
```

DirectSS1.Speak "Hi, welcome to the mail notify program. I will notify you about new email when it arrives in your inbox."

Elseif (gMode = 2) Then

Rem We just played the sign on sound..now speak the text.
Control

Rem goes to DirectSS1_AudioStop after the text is spoken

SayString = "You have new mail from " +
MAPIMessages1.MsgOrigDisplayName + ". Subject line," +
MAPIMessages1.MsgSubject

DirectSS1.Speak SayString

If (readbody(1).Checked) Then

Rem we try to skip forwards since reading them yet again
can become tiresome

bodystring = MAPIMessages1.MsgNoteText

forwardsloc = InStr(bodystring, "-----Original Message-----

")

If (forwardsloc = 0) Then

forwardsloc = InStr(bodystring, "From:")

End If

If (forwardsloc > 0) Then

bodystring = Left(bodystring, forwardsloc - 1)

End If

SayString = "Message Body reads: " + bodystring

If (forwardsloc > 0) Then

SayString = SayString + " Skipping forwarded message

body."

```
End If
DirectSS1.Speak SayString
End If
```

```
End If
```

```
End Sub
```

```
Private Sub DirectSS1_AudioStop(ByVal hi As Long, ByVal lo As  
Long)
```

```
If (gMode = 0) Then
```

```
Rem We have just finished speaking our startup text, so now  
we play our
```

```
Rem sign off sound and start waiting for new mail
```

```
gMode = 1
```

```
Rem Since we set our state to 1 nothing much happens when  
control goes to
```

```
Rem MMControl2_Done after the sound is played..
```

```
MMControl2.command = "Prev"
```

```
MMControl2.command = "Play"
```

```
Rem This call signs us on email..if the user is not already  
logged on to email,
```

```
Rem This call will prompt the user to do so.
```

```
MAPIsession1.SignOn
```

```
MAPIMessages1.SessionID = MAPIsession1.SessionID
```

```
Rem note that we are ready to start, and go ahead and check  
for mail right now.
```

```
gStarted = 1
Timer1.Interval = 1
Rem from now our timer will poll for new mail.
Timer1.Enabled = True
Elseif (gMode = 2) Then
Rem We have just finished reading the mail subject. Play sign
off sound.
Rem Control goes to MMControl2_Done which will reset our
mode/state once
Rem the sound is finished.
```

```
MMControl2.command = "Prev"
MMControl2.command = "Play"
```

```
End If
```

```
End Sub
```

```
Private Sub MMControl2_Done(NotifyCode As Integer)
```

```
If (gMode = 2) Then
```

```
Rem we have just played the sign off sound after reading a
subject line.
```

```
Rem go back into ready mode
```

```
gMode = 3
```

```
Timer1.Interval = 1 'check for new mail now!
```

```
End If
```

```
End Sub
```

```
Private Sub readbody_Click(Index As Integer)
Rem always stop the current reading so you have a way of
making the computer shut up
DirectSS1.AudioReset
readbody(1).Checked = Not readbody(1).Checked
End Sub
```

```
Private Sub stopread_Click(Index As Integer)
Rem always stop the current reading so you have a way of
making the computer shut up
DirectSS1.AudioReset
End Sub
```

```
Private Sub Timer1_Timer()
Rem check for new mail every once in a while, according to
how the
Rem timer objects interval property is set
Timer1.Interval = POLLTIME
CheckForNewMail
End Sub
```

This function is for handling a click on the tray icon (code from MSDN)

'we just pop up a menu.

```
Private Sub Form_MouseMove _
```

```
(Button As Integer, _
```

```
Shift As Integer, _
```

```
x As Single, _  
y As Single)  
'Event occurs when the mouse pointer is within the  
rectangular  
'boundaries of the icon in the taskbar status area.  
Dim msg As Long  
Dim sFilter As String  
msg = x / Screen.TwipsPerPixelX  
Select Case msg  
Case WM_RBUTTONDOWN  
PopupMenu notify(1)  
End Select  
End Sub
```

Rem This sample will play a signon sound and read the read
the subject of
Rem newly arrived e-mail for network email users. It uses the
MAPI control to
Rem access email, so your email must be MAPI compliant.
Rem Microsoft Exchange is MAPI compliant. So is Microsoft
Outlook, Microsoft Outlook Express
Rem This project has not been tested on any other E-Mail
Clients for its functionality.
Rem so before using this software on your custom e-mail
clients,
Rem make sure that it is MAPI compliant.

```
Rem NOTIFYICONDATA is data type is needed for creating an  
icon on the tray  
Private Type NOTIFYICONDATA  
cbSize As Long
```

hwnd As Long
uld As Long
uFlags As Long
uCallbackMessage As Long
hIcon As Long
szTip As String * 64
End Type

Rem constants for tray notifications

Private Const NIM_ADD = &H0

Private Const NIM_DELETE = &H2

Private Const WM_MOUSEMOVE = &H200

Private Const NIF_MESSAGE = &H1

Private Const NIF_ICON = &H2

Private Const NIF_TIP = &H4

Private Const WM_RBUTTONDOWN = &H204 'Button down

Private Declare Function Shell_NotifyIcon Lib "shell32" _

Alias "Shell_NotifyIconA" _

(ByVal dwMessage As Long, pnid As NOTIFYICONDATA) As

Boolean

Dim nid As NOTIFYICONDATA

Public gStarted As Integer

Public gMode As Integer

Public LastReceived As String

Private Const POLLTIME = 10000

Private Sub exit_Click(Index As Integer)

Rem Clean up the MM controls and tray icon before we exit

MMControl1.Command = "Close"

```
MMControl2.command = "Close"  
Shell_NotifyIcon NIM_DELETE, nid  
End 'end execution  
End Sub
```

```
Private Sub Form_Load()  
Rem This applet's main UI is an icon on the tray.  
nid.cbSize = Len(nid)  
nid.hwnd = Form1.hwnd  
nid.ulid = vbNull  
nid.uFlags = NIF_ICON Or NIF_TIP Or NIF_MESSAGE  
nid.uCallbackMessage = WM_MOUSEMOVE  
nid.szTip = "eMail Notify" & vbNullChar  
nid.hIcon = Form1.Icon  
Shell_NotifyIcon NIM_ADD, nid
```

```
Rem initialize the multimedia controls to play our sounds  
MMControl1.notify = False  
MMControl1.Wait = True  
MMControl1.Shareable = False  
MMControl1.DeviceType = "WaveAudio"  
MMControl1.FileName = App.Path + "\signon.wav"  
MMControl1.command = "Open"
```

```
MMControl2.notify = False  
MMControl2.Wait = True  
MMControl2.Shareable = False  
MMControl2.DeviceType = "WaveAudio"  
MMControl2.FileName = App.Path + "\signoff.wav"  
MMControl2.command = "Open"
```

Rem Dont except on errors with speech output..the user may
have some other
Rem sounds playing. Ideally we would catch the errors and do
the notify later.
DirectSS1.SuppressExceptions = 1

Rem set the last mail recieved time to a date in the past.
LastRecieved = "1996/08/05 16:40"

Rem Our state machine startes at mode 0
gMode = 0
gStarted = 0
Timer1.Interval = POLLTIME

Rem Play the startup sound
MMControl1.command = "Prev"
MMControl1.command = "Play"
Rem control picks up at MMControl1_Done after the sound is
played, perhaps
End Sub

Rem this function should only be called from the timer so it
does not get reentered
Rem instead, to call this do Timer1.Interval = 1
Private Sub CheckForNewMail()
Rem This function is called occassionly to check if new mail
has arrived

Dim SayString As String

Rem Just a bit of a precaution..be sure not to start get errors
if we

Rem are called before the program is initialized.

If (gStarted <> 1) Then

Exit Sub

End If

Rem Mark that we are checking mail, so we won't re-enter the
process

Rem while were doing it. We need to do this because the
sounds and

Rem spoken words are played assynchronously.

Rem Get the new mail list..

MAPIMessages1.FetchSorted = False 'sort the way the user
has their mailbox laid out

MAPIMessages1.Fetch

Rem If there is some mail..

tCount = 0

again:

If (MAPIMessages1.MsgCount) Then

Rem see if it is new mail, not just old mail marked as
unread...

MAPIMessages1.MsgIndex = MAPIMessages1.MsgCount -

tCount - 1

If ((MAPIMessages1.MsgDateReceived) >= (LastRecieved))

Then

Rem remember the time of this mail..we wont read mail way
older than this

Rem from now on, even if it is marked unread.

LastRecieved = MAPIMessages1.MsgDateReceived
Rem state change to reading mail..Play the sign-on sound
Rem control will go to MMControl1_Done when the sound is
finished.

gMode = 2

MMControl1.command = "Prev"

MMControl1.command = "Play"

Else

If (MAPIMessages1.MsgIndex = 0) Then

Rem we didn't do anything, so we are finished

Else

tCount = tCount + 1

GoTo again

End If

End If

Else

Rem we didn't do anything, so we are finished

End If

End Sub

Private Sub MMControl1_Done(NotifyCode As Integer)

If (gMode = 0) Then

Rem if we are just starting up, speak our text and control
goes to
Rem DirectSS1_AudioStop after the text is spoken

DirectSS1.Speak "Hi, welcome to the mail notify program. I
will notify you about new email when it arrives in your inbox."

Elseif (gMode = 2) Then

Rem We just played the sign on sound..now speak the text.
Control

Rem goes to DirectSS1_AudioStop after the text is spoken

SayString = "You have new mail from " +
MAPIMessages1.MsgOrigDisplayName + ". Subject line," +
MAPIMessages1.MsgSubject
DirectSS1.Speak SayString

If (readbody(1).Checked) Then
Rem we try to skip forwards since reading them yet again
can become tiresome

bodystring = MAPIMessages1.MsgNoteText

forwardsloc = InStr(bodystring, "-----Original Message-----

")

If (forwardsloc = 0) Then

forwardsloc = InStr(bodystring, "From:")

End If

If (forwardsloc > 0) Then

bodystring = Left(bodystring, forwardsloc - 1)

End If

SayString = "Message Body reads: " + bodystring

```
If (forwardsloc > 0) Then  
    SayString = SayString + " Skipping forwarded message  
body."
```

```
End If
```

```
DirectSS1.Speak SayString
```

```
End If
```

```
End If
```

```
End Sub
```

```
Private Sub DirectSS1_AudioStop(ByVal hi As Long, ByVal lo As  
Long)
```

```
If (gMode = 0) Then
```

```
Rem We have just finished speaking our startup text, so now  
we play our
```

```
Rem sign off sound and start waiting for new mail
```

```
gMode = 1
```

```
Rem Since we set our state to 1 nothing much happens when  
control goes to
```

```
Rem MMControl2_Done after the sound is played..
```

```
MMControl2.command = "Prev"
```

```
MMControl2.command = "Play"
```

```
Rem This call signs us on email..if the user is not already  
logged on to email,
```

```
Rem This call will prompt the user to do so.
```

```
MAPISession1.SignOn
```

```
MAPIMessages1.SessionID = MAPISession1.SessionID
Rem note that we are ready to start, and go ahead and check
for mail right now.
```

```
gStarted = 1
```

```
Timer1.Interval = 1
```

```
Rem from now our timer will poll for new mail.
```

```
Timer1.Enabled = True
```

```
Elseif (gMode = 2) Then
```

```
Rem We have just finished reading the mail subject. Play sign
off sound.
```

```
Rem Control goes to MMControl2_Done which will reset our
mode/state once
```

```
Rem the sound is finished.
```

```
MMControl2.command = "Prev"
```

```
MMControl2.command = "Play"
```

```
End If
```

```
End Sub
```

```
Private Sub MMControl2_Done(NotifyCode As Integer)
```

```
If (gMode = 2) Then
```

```
Rem we have just played the sign off sound after reading a
subject line.
```

```
Rem go back into ready mode
```

```
gMode = 3
```

```
Timer1.Interval = 1 'check for new mail now!
```

End If

End Sub

```
Private Sub readbody_Click(Index As Integer)
Rem always stop the current reading so you have a way of
making the computer shut up
DirectSS1.AudioReset
readbody(1).Checked = Not readbody(1).Checked
End Sub
```

```
Private Sub stopread_Click(Index As Integer)
Rem always stop the current reading so you have a way of
making the computer shut up
DirectSS1.AudioReset
End Sub
```

```
Private Sub Timer1_Timer()
Rem check for new mail every once in a while, according to
how the
Rem timer objects interval property is set
Timer1.Interval = POLLTIME
CheckForNewMail
End Sub
```

This function is for handling a click on the tray icon (code from MSDN)
we just pop up a menu.

```
Private Sub Form_MouseMove _
```

```
    (Button As Integer, _
```

```
    Shift As Integer, _
```

```
    x As Single, _
```

```
    y As Single)
```

```
    'Event occurs when the mouse pointer is within the
```

```
    rectangular
```

```
    'boundaries of the icon in the taskbar status area.
```

```
    Dim msg As Long
```

```
    Dim sFilter As String
```

```
    msg = x / Screen.TwipsPerPixelX
```

```
    Select Case msg
```

```
        Case WM_RBUTTONDOWN
```

```
            PopupMenu notify(1)
```

```
    End Select
```

```
End Sub
```

SOURCE CODE FOR START-UP SCREEN

```
Dim xchange As Integer
Dim ychange As Integer
Private Sub Command1_Click()
Unload Form2
Form5.Visible = True
Form5.SetFocus
End Sub
```

```
Private Sub Command2_Click()
Load Form1
End Sub
```

```
Private Sub Command3_Click()
Unload Form2
Form4.Visible = True
Form4.SetFocus
End Sub
```

```
Private Sub Form_Load()
xchange = 100
ychange = 100
```

```
End Sub
```

```
Private Sub Timer1_Timer()
Image1.Left = Image1.Left + xchange
Image1.Top = Image1.Top + ychange
Image2.Left = Image2.Left - xchange
Image2.Top = Image2.Top - ychange
```

Image2.Left = Image2.Left - xchange
Image2.Top = Image2.Top - ychange

xchange3 = xchange
ychange3 = ychange
Image3.Top = Image3.Top - ychange3

xchange2 = xchange
ychange2 = ychange

If Image1.Left > Me.ScaleWidth Then xchange = xchange * -1
If Image1.Left < 0 Then xchange = xchange * -1
If Image1.Top > Me.ScaleHeight Then ychange = ychange * -1
If Image1.Top < 0 Then ychange = ychange * -1

If Image2.Left > Me.ScaleWidth Then xchange2 = xchange2 * -1
If Image2.Left < 0 Then xchange2 = xchange2 * -1
If Image2.Top > Me.ScaleHeight Then ychange2 = ychange2 * -1
If Image2.Top < 0 Then ychange2 = ychange2 * -1

If Image3.Top > Me.ScaleHeight Then ychange3 = ychange3 * -1
If Image3.Top < 0 Then ychange3 = ychange3 * -1

End Sub

SOURCE CODE FOR MAIN PROGRAM

Rem this is the source code for Your Companion - The NLP
person
Rem program written in VISUAL BASIC 6.0
Rem This Code has been written by G.ANUSHA SARAVANA DEVI
and
Rem L.SUNDHARARAMAN of Final Bsc(CT) in Kumaraguru
College of technology
Rem for the project as a part of the DEGREE from
Rem BHARATHIYAR UNIVERSITY

```
Public gMode As Integer
Dim Keyword(400) As String
Dim Reply(10) As String
Dim string1 As String
Dim gThisFile As String
Dim gReadStart As Integer
Public key As Integer
Public resp As Integer
Public db As Database
Public rs As Recordset
Dim words(1 To 500) As String
Dim spaces(1 To 50) As Integer
Dim wordcomb(1 To 50) As String
Public sp As Integer
Public a As Integer
Public b As Integer
```

```
Private Sub About_Click()
```

MsgBox "Project developers are Anusha Saravana Devi and
Sundhararaman"
End Sub

Private Sub checkAndread_Click()
Form1.Enabled = True
Form1.Visible = False
Form5.SetFocus
End Sub

Private Sub Command2_Click()
Load Form1
Form5.SetFocus
End Sub

Private Sub Command4_Click()
RichTextBox1.selstart = 0
RichTextBox1.SelLength = Len(RichTextBox1.Text)
Clipboard.SetText RichTextBox1.SelText
SetText "", True
Text1.Text = Clipboard.GetText

seperatewords Text1

Dim stInput As String
stInput = Trim(Text1.Text)
For i = 1 To sp - 2
wordcomb(i) = Right\$(stInput, spaces(i))
Next
For i = 1 To sp - 2
Text1.Text = wordcomb(i)

Next

```
If Text1.Text <> "" Then  
rs.FindFirst "[keyword] = " & Text1.Text & ""  
If rs.NoMatch Then  
For b = 1 To a  
rs.FindFirst "[Keyword] = " & words(a) & ""  
Next  
GoTo 100  
Else  
If rs.NoMatch Then  
MsgBox "no match was found in the database", vbOKOnly  
End If  
100 Text1.Text = rs!response
```

End If

End If

DirectSS1.Speak Text1.Text

End Sub

```
Private Sub dicttrain_Click()
```

```
Vdict1.TrainGeneralDlg Form5.hwnd, "Dictation Training"
```

```
End Sub
```

```
Private Sub DirectSS1_AudioStop(ByVal hi As Long, ByVal lo As  
Long)
```

```
Toolbar1.Buttons(15).Value = tbrUnpressed
```

```
Toolbar1.Buttons(15).Image = 14
```

```
End Sub
```

```
Private Sub DirectSS1_WordPosition(ByVal hi As Long, ByVal  
low As Long, ByVal bytearray As Long)  
Dim firstspace As Integer
```

VB strings are unicode, so we divide the bytearray by 2 to get
the character offset

```
RichTextBox1.selstart = bytearray / 2 + gReadStart  
firstspace = FindTextBreak(1, RichTextBox1.selstart, "" +  
vbNewLine + vbLf + vbCr + vbCrLf)  
RichTextBox1.SelLength = firstspace - RichTextBox1.selstart
```

```
On Error GoTo done
```

```
Vdict1.Lock
```

```
Vdict1.TextSelSet RichTextBox1.selstart,
```

```
RichTextBox1.SelLength
```

```
Vdict1.Unlock
```

```
done:
```

```
End Sub
```

```
Public Sub Form_Load()  
Dim command As String
```

```
Dim Description As String
```

```
Dim category As String
```

```
Dim action As String
```

```
Dim FLAGS As Long
```

```
Set db = OpenDatabase("c:\my documents\kbase1.mdb")
```

```
Set rs = db.OpenRecordset("tbldata", dbOpenDynaset)
```

```
On Error GoTo ErrorMessage
```

```
If (Vdict1.Initialized = 0) Then
    Vdict1.Initialized = 1
    Vdict1.Mode = 32
End If
```

```
listen (0)
```

```
gThisFile = ""
```

```
ResetText
```

```
Alwaysselect.Checked = True
```

```
Toolbar1.ButtonHeight = ImageList1.ImageHeight
```

```
Toolbar1.ButtonWidth = ImageList1.ImageHeight
```

```
Toolbar1.Height = Toolbar1.ButtonHeight
```

```
RichTextBox1.Top = Toolbar1.Top + Toolbar1.Height
```

```
CommonDialog1.Filter = "Dictation (*.msd)|*.msd|Rich Text  
(*.rtf)|*.rtf|Text (*.txt)|*.txt"
```

```
GoTo NoError
```

```
ErrorMessage:
```

```
MsgBox "Unable to initialize dictation engine. Make sure an  
engine that supports dictation is installed."
```

```
End
```

```
NoError:
```

```
End Sub
```

```
Private Sub Addword_Click()
```

```
On Error GoTo addend
```

```
Vdict1.LexiconDlg Form5.hwnd, "Add Word"
```

```
addend:
```

```
End Sub
```

```
Private Sub alwaysontop_Click()
```

```
alwaysontop.Checked = Not (alwaysontop.Checked)
```

If (alwaysontop.Checked) Then

FloatPad (True)

Else

FloatPad (False)

End If

End Sub

Private Sub ResetText()

RichTextBox1.Font.Name = "Times New Roman"

RichTextBox1.Font.Size = 14

SelectAll_Click

Delete_Click

RichTextBox1.Font.Name = "Times New Roman"

RichTextBox1.Font.Size = 14

End Sub

Private Sub Alwaysselect_Click()

Alwaysselect.Checked = Not (Alwaysselect.Checked)

End Sub

Private Sub capitalizeword_Click()

s = RichTextBox1.selstart

e = RichTextBox1.SelLength

Vdict1.Lock

Vdict1.FX (3) 'toggle word case

Vdict1.Unlock

RichTextBox1.selstart = s

RichTextBox1.SelLength = e

End Sub

Private Sub listening_Click()

listen (1)

End Sub

```
Private Sub notlistening_Click()
```

```
listen (0)
```

```
End Sub
```

```
Private Sub ReadDocument_Click()
```

```
Dim ReadText As String
```

```
On Error Resume Next 'if TTS cannot read because some other  
thing is making sound, dont except
```

```
If (DirectSS1.Speaking) Then
```

```
Stopreading_Click
```

```
Else
```

```
listen (0)
```

```
Toolbar1.Buttons(15).Value = tbrPressed
```

```
Toolbar1.Buttons(15).Image = 13
```

```
gReadStart = RichTextBox1.selstart
```

```
ReadText = Right$(RichTextBox1.Text,
```

```
Len(RichTextBox1.Text) - RichTextBox1.selstart)
```

```
DirectSS1.Speak ReadText
```

```
End If
```

```
End Sub
```

```
Private Function isTextBreak(c As String, breakchars As String)
```

```
For i = 1 To Len(breakchars)
```

```
If c = Mid$(breakchars, i, 1) Then
```

```
isTextBreak = True
```

```
GoTo done
```

```
End If
```

```
Next i
```

```
isTextBreak = False
```

```
done:
```

End Function

Rem This is used instead of the rich text box scan function to
reduce screen flicker

Private Function FindTextBreak(direction As Integer, start As
Integer, breakchars As String)

If (RichTextBox1.Text = "") Then

 FindTextBreak = 1

Elseif (direction = 0) Then

 For i = start To 1 Step -1

 If (isTextBreak(Mid\$(RichTextBox1.Text, i, 1),
breakchars)) Then

 FindTextBreak = i

 GoTo done

 End If

 Next i

 FindTextBreak = i

 GoTo done

Else

 For i = start To Len(RichTextBox1.Text)

 If (isTextBreak(Mid\$(RichTextBox1.Text, i + 1, 1),
breakchars)) Then

 FindTextBreak = i

 GoTo done

 End If

 Next i

 FindTextBreak = i

End If

done:

End Function

```
Private Sub RichTextBox1_Click()  
Dim breakstring As String
```

```
If (Alwaysselect.Checked) Then  
    If RichTextBox1.selstart = Len(RichTextBox1.Text) Then  
        GoTo setsel
```

```
        breakstring = " -_!@#$$%^&*(),.[]+=`'\" + vbNewLine + vbCrLf  
        + vbCr + vbLf  
        sStart = FindTextBreak(0, RichTextBox1.selstart,  
        breakstring)
```

```
        If (sStart <> 0) Then  
            If (sStart = RichTextBox1.selstart And (Not  
            isTextBreak(Mid$(RichTextBox1.Text, (sStart), 1),  
            breakstring))) Then
```

```
                sStart = sStart - 1  
                Send = sStart + 1
```

```
            Else
```

```
                GoTo findit
```

```
            End If
```

```
        Else
```

```
            findit:
```

```
                Send = FindTextBreak(1, RichTextBox1.selstart +  
                RichTextBox1.SelLength, breakstring)
```

```
            End If
```

```
                RichTextBox1.selstart = sStart
```

```
                RichTextBox1.SelLength = Send - sStart
```

```
            setsel:
```

```
                On Error GoTo done
```

```
                Vdict1.Lock
```

```
Vdict1.TextSelSet RichTextBox1.selstart,  
RichTextBox1.SelLength  
Vdict1.Unlock
```

done:

End If

End Sub

```
Private Sub SetText(newText As String, ui As Boolean)
```

```
Vdict1.Lock
```

```
Vdict1.TextSelSet RichTextBox1.selstart, 0
```

```
RichTextBox1.TextSet newText, RichTextBox1.selstart,  
RichTextBox1.SelLength, 65536
```

```
Vdict1.Unlock
```

```
If (ui) Then
```

```
RichTextBox1.SelText = newText
```

```
End If
```

```
End Sub
```

```
Private Sub RichTextBox1_KeyPress(KeyAscii As Integer)
```

```
Dim s As String
```

```
s = Chr$(KeyAscii)
```

```
SetText s, False
```

```
ShowCorrectionWindow (showcorrection.Checked)
```

```
End Sub
```

```
Private Sub RichTextBox1_MouseUp(Button As Integer, Shift As  
Integer, x As Single, y As Single)
```

```
Vdict1.Lock
```

```
Vdict1.TextSelSet RichTextBox1.selstart,  
RichTextBox1.SelLength
```

```
Vdict1.Unlock
```

```
ShowCorrectionWindow (showcorrection.Checked)
End Sub
```

```
Private Sub setfont_Click()
' Set Cancel to True
CommonDialog1.CancelError = True
On Error GoTo ErrHandler
' Set the Flags property
CommonDialog1.FLAGS = cdlCFEffects Or cdlCFBoth
' Display the Font dialog box
CommonDialog1.ShowFont
RichTextBox1.Font.Name = CommonDialog1.FontName
RichTextBox1.Font.Size = CommonDialog1.FontSize
RichTextBox1.Font.Italic = CommonDialog1.FontItalic
RichTextBox1.Font.Bold = CommonDialog1.FontBold
RichTextBox1.Font.Underline = CommonDialog1.FontUnderline
RichTextBox1.Font.StrikeThru =
CommonDialog1.FontStrikethru
```

```
ErrHandler:
```

```
End Sub
Private Sub showcorrection_Click()
If (showcorrection.Checked) Then
ShowCorrectionWindow (0)
Else
ShowCorrectionWindow (1)
End If
End Sub
Private Sub Stopreading_Click()
DirectSS1.AudioReset
```

```
End Sub  
Private Sub Vdict1_AttribChanged(ByVal Attrib As Long)
```

```
    If Attrib And 2 = 2 Then  
        If (Vdict1.Mode And 32) Then  
            ListenUI (1)  
        Else  
            ListenUI (0)  
        End If  
    End If
```

```
End If  
End Sub
```

' This is the core of the program. TextChanged is called
whenever the text
' is changed by the dictation object. Use GetChanges to get
the differences
' and update the rich text control.

```
Private Sub Vdict1_TextChanged(ByVal reason As Long)  
    Dim newStart As Long  
    Dim newend As Long  
    Dim oldStart As Long  
    Dim oldEnd As Long  
    Dim selstart As Long  
    Dim sellen As Long  
    Dim theText As String
```

```
Vdict1.Lock
```

```
On Error GoTo spuriouserror  
Vdict1.GetChanges newStart, newend, oldStart, oldEnd
```

```
If (oldStart < oldEnd) Then
    RichTextBox1.selstart = oldStart
    RichTextBox1.SelLength = oldEnd - oldStart
    RichTextBox1.SelText = ""
End If
```

```
If (newend > newStart) Then
```

```
    RichTextBox1.selstart = newStart
    RichTextBox1.SelLength = 0
```

```
    Vdict1.TextGet newStart, newend - newStart, theText
    RichTextBox1.SelText = theText
End If
```

```
    ShowCorrectionWindow 2
```

```
spuriouserror:
```

```
Vdict1.Unlock
```

```
End Sub
```

```
Private Sub Vdict1_TextSelChanged()
```

```
    Dim selstart As Long
```

```
    Dim sellen As Long
```

```
    Vdict1.TextSelGet selstart, sellen
```

```
    RichTextBox1.selstart = selstart
```

```
    RichTextBox1.SelLength = sellen
```

```
End Sub
```

```
Private Sub Bold_Click()
```

```
    RichTextBox1.SelBold = True
```

```
End Sub
```

```
Private Sub Copy_Click()  
Clipboard.SetText RichTextBox1.SelectedText  
End Sub
```

```
Private Sub Cut_Click()  
Clipboard.SetText RichTextBox1.SelectedText  
SetText "", True  
End Sub
```

```
Private Sub Delete_Click()  
SetText "", True  
End Sub
```

```
Private Sub exit_Click()  
Form5.Visible = False  
Form3.Visible = True  
Form3.SetFocus  
End Sub
```

```
Public Sub ShowCorrectionWindow(Show)  
Static caret As POINTAPI
```

```
If (Show) Then
```

```
If (Show = 2) Then
```

```
If (showcorrection.Checked) Then
```

```
sflags = 1
```

```
Else
```

```
sflags = 0
```

```
End If
```

```
Else
```

```
showcorrection.Checked = True
```

```
sflags = 1
```

```
End If
```

```
Else
```

showcorrection.Checked = False

sflags = 0

End If

Vdict1.FLAGS = (Vdict1.FLAGS And (Not 1)) Or sflags

GetCaretPos caret

caret.x = caret.x + Form5.Left / Screen.TwipsPerPixelX

caret.y = caret.y + Form5.Top / Screen.TwipsPerPixelY +

RichTextBox1.Top / Screen.TwipsPerPixelY + 60

Vdict1.SetSelRect caret.x, caret.y, caret.x, caret.y

End Sub

Private Sub FloatPad(ontop)

If (ontop) Then

tp = SetWindowPos(Form5.hwnd, HWND_TOPMOST, 0, 0, 0,
0, FLAGS)

Else

tp = SetWindowPos(Form5.hwnd, HWND_NOTOPMOST, 0, 0,
0, 0, FLAGS)

End If

End Sub

Private Function Max(a, b)

If (a > b) Then

Max = a

Else

Max = b

End If

End Function

```

Public Sub Form_Unload(Cancel As Integer)
listen (0)
ShowCorrectionWindow (0)
End
End Sub
Private Sub Italic_Click()
RichTextBox1.SelItalic = True
End Sub
Private Sub New_Click()
If MsgBox("This will erase the current text. Are you sure?",
vbOKCancel, "New") = 1 Then
ResetText
gThisFile = ""
End If
End Sub
Private Sub open_Click()
Dim versionstring As String
Dim fn As String
Dim fs As Long
Dim fi As Long
Dim fb As Long
Dim fu As Long
Dim fst As Long

CommonDialog1.ShowOpen
If (CommonDialog1.FileName <> "") Then
New_Click
gThisFile = CommonDialog1.FileName

If (Right(gThisFile, 3) = "msd") Then

```

```
hand = Vdict1.CreateDocFile(gThisFile, 18)
stream2 = Vdict1.CreateStream(hand, "Version", 18)
Vdict1.StreamRead stream2, versionstring, 36
```

```
If versionstring <> "46FC730A-D849-11d0-AB8A-
08002BE4E3B7" Then
```

```
    dummy = MsgBox("This program cannot open the file.
Invalid format", vbOK, "Invalid Format")
```

```
Else
```

```
    stream = Vdict1.CreateStream(hand, "Header", 18)
    Vdict1.ReadStreamFont stream, fn, fs, fi, fb, fu, fst
```

```
    Vdict1.ReleaseStream stream
```

```
    Vdict1.SessionDeserialize hand
```

```
    RichTextBox1.Font.Name = fn
```

```
    If (fs < 0) Then
```

```
        fs = -fs
```

```
    End If
```

```
    RichTextBox1.Font.Size = fs
```

```
    RichTextBox1.Font.Italic = fi
```

```
    RichTextBox1.Font.Bold = fb
```

```
    RichTextBox1.Font.Underline = fu
```

```
    Rem RichTextBox1.Font.StrikeThru = fst
```

```
End If
```

```
Vdict1.ReleaseStream stream2
```

```
Vdict1.ReleaseStore hand
```

```
Else
```

```
    RichTextBox1.FileName = CommonDialog1.FileName
```

```
    gThisFile = RichTextBox1.FileName
```

```
    SetText RichTextBox1.Text, False
```

```
End If
```

```
End If
```

```
End Sub
Private Sub Options_Click()
Vdict1.GeneralDlg Form5.hwnd, "Dictation Options"
```

```
End Sub
Private Sub Paste_Click()
SetText Clipboard.GetText, True
End Sub
```

```
Private Sub DoSave()
Dim hand As Long
Dim fs As Long
Dim fi As Long
Dim fb As Long
Dim fu As Long
Dim fst As Long
```

```
If (Right(gThisFile, 3) = "msd") Then
```

```
fs = -RichTextBox1.Font.Size
fi = RichTextBox1.Font.Italic
fb = RichTextBox1.Font.Bold
fu = RichTextBox1.Font.Underline
fst = False
```

```
hand = Vdict1.CreateDocFile(gThisFile, 18)
stream = Vdict1.CreateStream(hand, "Header", 18)
Vdict1.SetSize stream, 0
Vdict1.WriteStreamFont stream, RichTextBox1.Font.Name,
fs, fi, fb, fu, fst
Vdict1.SessionSerialize hand
```

```

stream2 = Vdict1.CreateStream(hand, "Version", 18)
Vdict1.SetSize stream2, 0
Vdict1.StreamWrite stream2, "46FC730A-D849-11d0-AB8A-
08002BE4E3B7"
Vdict1.ReleaseStream stream
Vdict1.ReleaseStream stream2
Vdict1.ReleaseStore hand
Elseif (Right(gThisFile, 3) = "rtf") Then
    Open gThisFile For Output As 1
    Print #1, RichTextBox1.TextRTF
    Close 1
Else
    Open gThisFile For Output As 1
    Print #1, RichTextBox1.Text
    Close 1
End If

End Sub

Private Sub save_Click()
If (gThisFile <> "") Then
    DoSave
Else
    saveas_Click
End If
End Sub

Private Sub saveas_Click()

CommonDialog1.ShowSave
If (CommonDialog1.FileName <> "") Then
    gThisFile = CommonDialog1.FileName
    DoSave

```

```

End If
End Sub
Public Sub ListenUI(op As Integer)
If (op = 1) Then

    Toolbar1.Buttons(1).Value = tbrPressed
    Toolbar1.Buttons(1).Image = 1
    listening.Checked = True
    notlistening.Checked = False
Else
    Toolbar1.Buttons(1).Value = tbrUnpressed
    Toolbar1.Buttons(1).Image = 12
    listening.Checked = False
    notlistening.Checked = True
End If
End Sub
Public Sub listen(op As Integer)

If (op = 1) Then
    Stopreading_Click
    Vdict1.Mode = 32
On Error GoTo NoActivate
    Vdict1.Activate
NoActivate:
    Else
        Vdict1.Mode = 2
On Error GoTo NoDeactivate
    Vdict1.Deactivate
NoDeactivate:

End If

```

```
ListenUI (op)
GoTo NoError
ErrorMessage:
MsgBox "Unable to initialize dictation engine. Make sure an
engine that supports dictation is installed."
End
NoError:
End Sub
Private Sub toolbar1_ButtonClick(ByVal Button As Button)
    ' Use the Key property with the SelectCase statement to
specify
    ' an action.
    Select Case Button.key
        Case Is = "cut"
            Cut_Click

        Case Is = "cutall"
            SelectAll_Click
            Cut_Click

        Case Is = "copy"
            Copy_Click

        Case Is = "paste"
            Paste_Click

        Case Is = "new"
            New_Click

        Case Is = "save"
            save_Click
```

Case Is = "open"

open_Click

Case Is = "listen"

If (listening.Checked) Then

listen (0)

Else

listen (1)

End If

Case Is = "showhide"

showcorrection_Click

Case Is = "capitalizeword"

capitalizeword_Click

Case Is = "addword"

Addword_Click

Case Is = "read"

ReadDocument_Click

End Select

End Sub

Private Sub SelectAll_Click()

RichTextBox1.selstart = 0

RichTextBox1.SelLength = Len(RichTextBox1.Text)

End Sub

Private Sub Command1_Click()

RichTextBox1.selstart = 0

RichTextBox1.SelLength = Len(RichTextBox1.Text)

```
Clipboard.SetText RichTextBox1.SelectedText  
SetText "", True  
Text1.Text = Clipboard.GetText  
End Sub
```

```
Private Sub Command3_Click()  
Form5.Hide  
Form5.Visible = False  
Form4.Visible = True  
Form4.SetFocus  
End Sub
```

```
Sub separatewords(stInput As String)  
Dim incounter As Integer  
Dim infoundpos As Integer  
Dim length As Integer
```

```
Const parsechar = " "  
length = Len(stInput)
```

```
If Len(stInput) = 0 Then Exit Sub
```

```
incounter = 1
```

```
sp = 1
```

```
infoundpos = InStr(incounter, stInput, parsechar)
```

```
spaces(sp) = infoundpos
```

```
sp = sp + 1
```

```
a = 1
```

```
While infoundpos <> 0
```

```
words(a) = Mid$(stInput, incounter, infoundpos - incounter)
```

```
incounter = infoundpos + 1
infoundpos = InStr(incounter, stInput, parsechar)
spaces(sp) = infoundpos
sp = sp + 1
a = a + 1
Wend
```

```
If incounter < Len(stInput) Then
    words(a) = Mid$(stInput, incounter)
End If
```

```
End Sub
```

SAMPLE SOURCE CODE FOR MODULE USED

```
Public Const SWP_NOMOVE = 2
Public Const SWP_NOSIZE = 1
Public Const FLAGS = SWP_NOMOVE Or SWP_NOSIZE
Public Const HWND_TOPMOST = -1
Public Const HWND_NOTOPMOST = -2
Declare Function SetWindowPos Lib "user32" (ByVal hwnd As
Long, ByVal hWndInsertAfter As Long, ByVal x As Long, ByVal y
As Long, ByVal cx As Long, ByVal cy As Long, ByVal wFlags As
Long) As Long

Type POINTAPI
    x As Long
    y As Long
End Type
Declare Function GetCaretPos Lib "user32" (lpPoint As
POINTAPI) As Long
```

FUTURE ENHANCEMENTS

In future a system that recognizes the users speech irrespective of the context or style, in a particular language will be developed. The user will be awarded the real freedom of speech where he or she can communicate with the computer irrespective of the context to which their interaction belongs.

All the operation will be controlled through voice completely. The user can avoid the usage of the keyboard or mouse to usage for controlling the program. This provides to be more productive for the users who are handicapped or the people who are busy and cannot control keyboard or mouse or the people who don't know how to use keyboard or mouse.

There will be more options to improve the productivity of the program in our future enhancement of the program.

CONCLUSION

Thus the project titled "Your Companion - The NLP person" can be explained in every aspect with respect to its performance. The aim of the project is to make the vision impaired people also to use the computer to learn about it, check and hear their e-mails. From the project development we could learn a lot of areas where development we could as to make the project more useful to the users that will be done in the project's future enhancements.

BIBLIOGRAPHY

- Loren. D. Eidahl, Brian Siler, Jeff Spotts, Francesco Balena, Jeff Cainer, Nelson Howell, Duncan Mackenzie, Lowell Mauer, Brad Shannon, Mike McKley, "**Platinum Edition Using Visual Basic[®] 5**", MacMillan Computer Publishing USA, 1997.
- Brian Siler and Jeff Spotts, "**Special Edition Using Visual Basic[®] 6**", Prentice-Hall of India Private Limited, 1999.
- Jeffery P. McManus, "**DataBase Access With Visual Basic[®]**", MacMillan Computer Publishing, USA, 1998.
- WWW.NETLINGO.COM, 2000.
- WWW.NETDICTIONARY.COM, 2000.

T. 258

