POLITECNICO DI MILANO

Facoltà di Ingegneria dell'Informazione

Polo Regionale Di COMO

Master of Science in Computer Engineering

Laurea Specialistica in Ingegneria Informatica

"POLILIPS: APPLICATION DEAF & HEARING DISABLE STUDENTS"

Supervised by: Professor Sbatella Licia

Prof: Roberto Tedesco

Master Thesis by:

Yaprak Yakar: 737744

Muhammad Usman: 733804

Academic Year 2010/11

Abstract

Speech Recognition (SR) is a very useful technology, which makes life easier and has much to offer in our daily life and for the future. Generally, speech recognizer is a machine which understands humans and their spoken by converting to text. In SR makes communicating with the computer faster than manual interfaces do, like for example the keyboard means that the main advantage of SR is that it can save time. A different aspect of speech recognition is to facilitate for people with functional disability or other kinds of handicap.

Typical applications include dictation, automatic transcription of large audio or video databases, speech-controlled user interfaces, and automated telephone services. If the recognition system is not limited to a certain topic and vocabulary, covering the words in the target languages as well as possible while maintaining high recognition accuracy becomes an issue. The conventional way to model the target language, especially in English recognition systems, is to limit the recognition to the most common words of the language. A vocabulary of 60 000 words is usually enough to cover the language adequately for arbitrary topics.[1]

This thesis introduces the potential of Automatic Speech Recognition (ASR) technology that is one of the most efficient input methods for human-computer in the challenge of inclusive education. ASR technology combined with Information and Communication Technology (ICT) enhances the learning of disabled people both in the classroom. In the classroom, deaf and hearing-impaired students can benefit from a real time transcription of what teacher is saying in English.

[1] http://lib.tkk.fi/Diss/2009/isbn9789512299775/

Acknowledgements

We are very thankful to our supervisor, Prof. Sbatella Licia, for her guidance throughout the thesis work. Her enthusiasm in research and guidance directed us to teach a new and inspiring field, i.e. Knowledge Representation. She was always accessible and willing to help us to overcome difficulties.

Also I would like to pay thanks to Prof. Roberto Tedesco (Post Doctoral Researcher at Politecnico Di Milano University) who helped us in understanding the idea and to lead basic objectives for this project especially to work with core aspects of projects, and also to all our friends, colleagues whose inspiration and moral support helped us.

We express our gratitude with heart and soul to all our family, to our closed friends who stood side by side with us in all aspects of our life, for their continued love and support throughout our life, this dissertation won’t have been possible without them.

October 2011

Yaprak Yakar

Usman Muhammad

Table of Contents

Abstract

Acknowledgements

1 Introduction

1.1 Problem Statement

1.2 Project Goals

1.3 Motivation

1.4 Structure of this document

2 Background Study

2.1 Introduction to Speech Recognition

2.1.1 History of Speech Recognition

2.1.2 Automatic Speech Recognition (ASR)

2.1.2.1The types of Automatic Speech Recognition

2.1.3. The Practical Side of ASR

3 Design

3.1 Stakeholder

3.2 Operating Environment

3.3 Architecture

3.4 Operational States

3.4.1 Main Functional Constraints

3.4.2 Assumptions

3.5 UI Prototypes

3.6 Use Cases

3.6.1 Use case 0:

3.6.2 Use case 1:

3.6.3 Use case 2

3.7 Dynamic Aspects of the System

3.8 POLILIPS Perspectives

3.8.1 POLILIPS Features

3.8.2 Operating Environment

3.8.3 Design and Implementation Constraints

3.8.4 Assumptions and Dependencies

3.8.5 Functional Requirements

3.9 User Interface

3.9.1 Hardware Interface

3.9.2 Software Interface

4 Implementation
4.1 Overview
4.1.1 Teacher is server-based module.
4.1.2 Student is client-based module.
4.2 Technologies used
4.3 Usability Evaluation for POLILIPS
4.3.1 Usability Test for Student Module:
4.3.2 Usability Test for Teacher Module:
5 Future Work
5.1 Future Work
5.2 Knowledge Sharing
5.3 Real World Applications
5.4 Extended Version Features
Conclusion

List of Tables:

Table 3.1: Functional Constraints ……..…………………………49

Table 3.2: Features of POLILIPS ……….……………………… 59

Table 5.1: Features of extended POLILIPS ……………………. 83

List of Figures:

Fig-1.1: The Signal Model of Speech Recognition 22
Fig-1.2: Today's Technology 24
Fig-1.3: Today's technology Vocabulary Size (words) 25
Fig-1.4: A schematic of an automated speech dialog system...........................26
Fig-1.5:A generic Solution 33
Fig-1.6: Isolated Noise class 34
Fig-1.7: Continuous Speech 35
Fig 3.1: Over all architecture for POLLIPS 45
Fig 3.2 : POLILIPS Server operate by Teacher 46
Fig 3.3: POLILIPS client operate by Student 48
Fig 3.4: Over System for POLILIPS 50
Fig 3.5: Teacher Module 52
Fig 3.6: Sequence Diagram for teacher module 54
Fig 3.7: Use-case for Student Module 55
Fig 3.8: Sequence diagram for Student Module 57
Fig 3.9: TELECAMERA CMOS COLORE OBIETTIVO 3,6mm 61
Fig 3.10: Transmitter and Receiver with Microphone 62
Fig 3.11: Receiver for Audio & Video Signals 63
Fig 3.12: E-CAPTURE: DATA ACQUISITION OF AUDIO / VIDEO TO PC 64
Fig 3.13: Batter Pack 65
Fig 3.14: Charger for battery cells 66
Fig 4.1: Login details for Teacher 69
Fig 4.2: Select Camera Device 70
Fig 4.3: POLILIPS Teacher Module 71

Fig 4.4: Login details for Student 72

Fig 4.5: Interface for Student Module With Server side streaming 73

Fig 4.6: Battery Charger Devices 74

Fig 4.7: Assembled Hardware for teacher. 75

Fig 4.7.1: Assembled Hardware for teacher. 75

Fig 4.8: POLILIPS STUDENT MODULE 77

Fig 4.9: Usability test for Teacher Module 78

Chapter 1: Introduction

1.1 Problem Statement

In real life time we have experience deaf people only can read sign language either its movement of hand or movement of lips, which helping to understand motive to deaf people. In example lets talk about a deaf student is attending class but can read the lips movement to understand words, if lips movement is closely captured then information flow will be more easier to understand.

Speech is the primary means of communication between people. For this reasons ranging from technological curiosity about the mechanisms for technical understanding of human speech capabilities, to the desire to automate simple tasks inherently requiring human-machine interactions,

Through a application which shows lips and subtitles by ASR software or APIs, (Automatic Speech recognition) movement of teacher on smartphone (or notebook), The needs teacher wears a small camera and a microphone, which capture the audio/video stream; a teacher notebook receives the stream, on a network adds subtitles and sends the enhanced stream to the user's devices.

In the last years and also for the future, computers have become an integral part of modern daily lives, and our expectations of a user-friendly interface have increased considerably. Automatic speech recognition (ASR) is one of the most efficient input methods for human-computer interface because it is natural for humans to communicate through speech. The input speech signal of the traditional ASR process is usually recorded with a microphone, e.g., a microphone of a closetalking headset or a telephone. In order to make their daily routine easier, voice control could be helpful.

People could operate the light switch, turn on/off the electrical appliances with their voice. Incredibly, this leads to the discussion about intelligent homes where these operations can be made available for common man as well as for the handicapped. [2]ASR is an automatic computerised speech-to-text process that converts human speech signals into written words. It has various applications, such as voice command and control, dictation, dialog systems, audio indexing, speech-to-speech translation, etc. In our application, we used Windows Speech API to provide the user speech to text and text to speech facilities.

1.2 Project Goals

- Broadcast video and audio text streaming on network to words client devices.
- Client devices able to receive streaming from server.
- Client device have to communicate through text streaming with server. (extended goal)

1.3 Motivation

ASR (Automatic Speech Recognition) research has been primarily focused towards large-scale systems and industry and the researchers of ASR continue to make significant technological advances in the area. In the past, speech has been available but very costly. Today speech recognition software for computers is not only commercially available but also reasonably priced or even some of them are free as Microsoft Speech API.

[2] http://umpir.ump.edu.my/409/1/Ardian_Syah_B_Mohd_Yusof_3282.pdf

Today in most schools, businesses, and increasingly in homes, computers are being used to supplement daily life. Individuals use computers to manage everything from business transactions to completing homework assignments. Despite the fact that automatic speech recognition can be used in an increasing number of applications, certain physical and emotional environments are still believed unsuitable for this technology.[3]

Every student has his or her own individual needs. In this thesis, we will try to focus on how speech recognition can be used to assist teaching and learning especially and provide an overview of the main types of technologies that can be used to meet the needs of students with a hearing disability. Finally, the importance of having an open dialogue between student and institution about how their needs can be met is highlighted.

The most of deaf people were born to hearing parents and were taught at school without sign language support and so may not have learnt to use sign languages. They use lip-reading combined with their hearing, aided by hearing aids or cochlear implants (which encourage nerves in the ear electrically). At school or college they may have received support from staff and fellow students and may have been taught in acoustically treated rooms. At university they may find it more difficult to manage using just hearing and lip reading due to background noise, reverberant rooms, poor lighting conditions and staff and students unused to talking to deaf students.

Institutions can provide technology, based on knowledge and information about what that technology can in theory achieve or has in practice achieved for others. Institutions can also do a great deal to ensure their policies and practice assists technology in helping removing barriers to learning and participation. However, since only the individual student can decide whether any particular technology is appropriate to meet their particular individual needs it is important

[3] http://scholar.lib.vt.edu/theses/available/etd-7598-165040/unrestricted/thesis1.pdf

for student and institution to discuss these needs and how they can best be met. It is also important to discuss what the student can provide (e.g. through the Disabled Students Allowance) and what the institution can provide[4].

In real life time we have experience hear impaired people only can read sign language either its movement of hand or movement of lips, which helping to understand motive to hear disabled people. In example let's talk about a deaf student is attending class but can read the lips movement to understand words, if lips movement is closely captured then information flow will be more easier to understand.

Through a application which shows lips and subtitles by ASR software or APIs, (Automatic Speech recognition) movement of teacher on smart phone (or notebook), The needs teacher wears a small camera and a microphone, which capture the audio/video stream; a teacher notebook receives the stream, on a network adds subtitles and sends the enhanced stream to the user's devices. In our application, we will provide that

- **Remote real time communication support**

The people who have hear impaired problem does not have to travel to the lecture by providing record or unravelling services. Therefore, it would be possible to pay for a shorter session and have the choice of employing people from a much wider geographical area who may have more appropriate skills and knowledge. Thanks to our application, the students who do not understand something about the lecture can ask their questions to their teacher in real time while they follow their teacher by reading the lips of their teacher and the text which teacher mentioned at a distance in their preferred way.

- **Text communication**

Text to speech recognition which we have fixed to our application can also allow text communication between teacher and students by asking their questions who's the students do not understand clearly when teacher tried to teach. When the student ask a question as a text, the other students who have signed in to the system for the lecture can see the question of the student and the teacher can hear the question which the student has asked as speech.

- **Using speech recognition to assist teaching and learning**

Current speech recognition applications are relatively inexpensive and capable of accurate and fast responses on standard computers for normal rates of speech, with minimal training of the system to the speaker's voice or training of the user of the technology. Speaker-independent systems that require no enrolment/training may be available in the near future.

Staff and students may have preferences regarding whether and when they find the spoken or written forms of language easier or more useful for teaching and learning. Text to speech applications can automatically change text into speech while speech recognition technology can be used to automatically change speech into text. The application assist teaching and learning effectively to the hear disabled students by providing speech to text recognition and text to speech recognition

- **Real time text transcription in lectures**

SR (Speech Recognition) can be used for providing real time text transcription in lectures to provide a text display of what is being spoken as well as a precise transcript for later reference. To achieve a similar result without the technology would involve the use of expensive, highly trained real-time speech to text reporters who are in great demand for court reporting and real time subtitling of television programs. Standard speech recognition applications require the user to dictate punctuation to break up the text into 'readable' portions. [5]

Real time speech to text transcription can assist deaf students who find it difficult to follow the speaker through hearing alone. It can also be of benefit for students or speakers whose first language is not English and when there are poor acoustics. In addition to deaf students who need to watch to lip-read can assist any students who find it difficult to take notes during a lecture, for example visually impaired students, or students who have a physical disability affecting writing or typing. In addition many students who have no disability or learning difficulty find it difficult to take notes at the same time as listening, watching and thinking.

Speech-recognition can also be used to support distance learning by providing automatic speech to text transcription for online text chat and video streaming.

[5] http://www.soton.ac.uk/~shec/Section16shfecwapICTadvicedocument.html

- **Wireless video camera to aid lip-reading**

A wireless video camera worn round the neck of the person speaking could provide a clear large video image of the speaker's face even if the speaker turns away, is at a distance or has their face obscured by other students etc.

1.4 Structure of this document

This documents presents the accomplished work starting from an overview of the learning application and proof of concept, describing the design of the prototype and implementation phases, and concluding with the description of the case study, based on application future direction of learning application of deaf people.

More precisely, the document is structured as follows. Chapter 2 contains a background study and concepts what ASR speech to text and text to speech, text streaming, what work has been done in this research field till now. Chapter 3 presents the design based on proof of concepts discussed in previous chapter, while Chapter 4 discusses implementation phases has been overcome through my work, and introduces a working application. Chapter 5 introduces the foreseen future evolutions and, finally, chapter 6 draws conclusions.

References and list of figures, tables, bibliography is supporting more document illustration.

Chapter 2: Background Study

2.1 Introduction to Speech Recognition

" Who uses Speech to Text?
STT (Speech to text) is often used by people who were born hearing and later have hearing loss. It is appropriate for someone whose chosen language is English rather than British Sign Language and who is comfortable reading scrolling text from a computer screen."[6]

If you need access services for:

- Conferences and symposia, with live subtitling on large screens
- One-on-one support at conferences or small group meetings
- Union meetings or public meetings where access provision is a matter of policy
- Interviews
- Law Courts
- Webcasts
- Training courses

Natural language interfaces with voice recognition are going to play an important role in our world. The speed of typing and handwriting is usually one word per second; so speaking may be the fastest communication form with a computer. Applications with voice recognition can also be a very helpful tool for handicapped people who have difficulties with typing. The number of organizations that widen knowledge of voice communication with computer is still growing. This is very important because each language needs an individual approach for setting the style grammar or continuous speech.

[6] http://www.ubiqus.ie/site/GB/Corporate__PR_sectors/Disability_access_services/What_is_Speech_to_Text,I4044.htm

SR (Speech recognition) has been defined as the ability for a computer to understand spoken commands is an important factor in the HCI (Human Computer Interaction). SR has been available for many years, but it has not been improved due to the high cost of applications and computing resources. The SR had significant growth in telephony (voice mail, call centre management) and voice-to-text (VTT) applications. Speech recognition is a broader solution that refers to technology that can recognize speech without being targeted at single speaker—such as a call system that can recognize even arbitrary voices.[7]

[1]http://www.ubiqus.ie/site/GB/Corporate__PR_sectors/Disability_access_services/What_is_Speech_to_Text,I4044.htm
[1] http://www.faqs.org/docs/Linux-HOWTO/Speech-Recognition-HOWTO.html#BASICS

Speech recognition is a broader solution that refers to technology that can recognize speech without being targeted at single speaker—such as a call system that can recognize even arbitrary voices.[8]

2.1.1 History of Speech Recognition

The general public's "understanding" of speech recognition comes from such things as the HAL 9000 computer in Stanley Kubrick's film *2001: A Space Odyssey.* Notice that HAL is a perversion of IBM. At the time of the movie's release (1968) IBM was just getting started with a large speech recognition project that led to a very successful large vocabulary *isolated word* dictation system and several small vocabulary control systems. [9]

[8] Wikipedia http://en.wikipedia.org/wiki/Speech_recognition
[9] Illinois Institute of Technology,2001

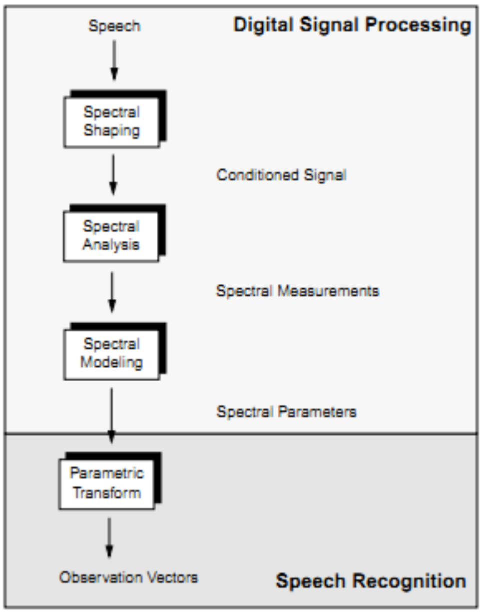


Fig-1.1: The Signal Model of Speech Recognition

In the middle 19's IBM's *Voice type*, Dragon Systems' *Dragon Dictate*, and Kurzweil Applied Intelligence's *Voice Plus* were the popular personal computer speech recognition products on the market. These "early" packages typically required additional (non-standard) DSP computer hardware. They were about 90% accurate for general dictation and required a short pause between words. They were called *discrete* speech recognition systems.

2.1.2 What is the context for Speech Sympathetic research in the 90`s?

- **The Application Need**
- Interaction with on-line data (Internet)
- Automated information agents (“24 hour Help Desk”)
- Global multilingual interactions

- **The Technology Challenge**
- Create natural transparent interaction with computers (“2001”,”Star Trek”)
- Bring computing to the masses (vanishing window)
- Intelligent presentation of information (“hands/eyes busy applications”)

- **Application Areas**
- Command/Control (Telecom/Workstations)
- Database Query (Internet)
- Dictation (Workstations)
- Machine Translation (Workstations)
- Real time interpretation (Telecom)

2.1.2 The Technical Challenge

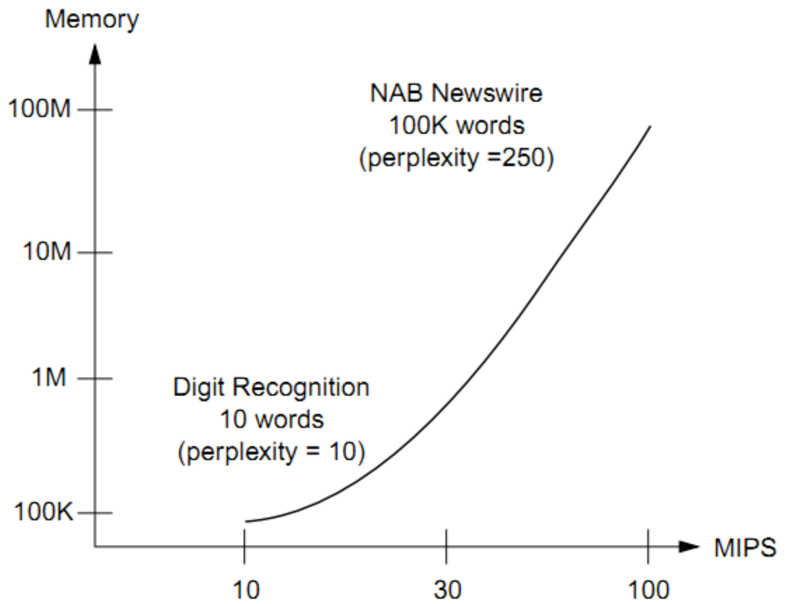
- Pyramids of hidden representations produce immensely complex models.
- Training of complex models requires huge and dynamic knowledge bases
- Interconnected and interdependent levels of representation:
- Correct recognition and transcription of speech depends on understanding the meaning encoded in speech.
- Correct understanding and interpretation of text depends on the domain of discourse.

- *Approach*

  Capitalize on exponential growth in computer power and memory:
    - Statistical modelling and automatic training
    - Shared resources and infrastructures

2.1.3 Today`s Technology

The performance of speech recognition systems is usually evaluated in terms of accuracy and speed. Accuracy is usually rated with word error rate (WER), whereas speed is measured with the real time factor. Other measures of accuracy include Single Word Error Rate (SWER) and Command Success Rate (CSR).



[10]

Fig-1.2: Today's Technology

[10] http://speech.tifr.res.in/tutorials/fundamentalOfASR_picone96.pdf

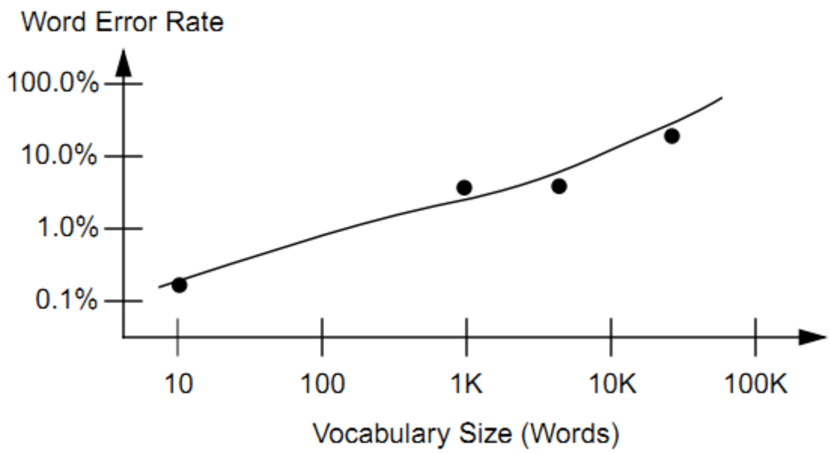


Fig-1.3: Today's technology Vocabulary Size (words)

[11]

2.1.4 Automatic Speech Recognition (ASR)

ASR (Automatic speech recognition) systems for computer access allow users to enter text and commands into the computer using their voice. These systems have the potential to greatly improve the productivity and comfort of performing computer-based tasks for a wide variety of users. ASR may be specifically attractive to people with physical disabilities, when non-speech methods of computer input, such as the keyboard or mouse, may be too slow or too painful to fully meet their needs. This study explores how well ASR systems are meeting the needs of users who have physical disabilities.

While specific speech recognition systems can vary significantly, most may be described along at least five dimensions, including training requirements, the ability to handle continuous speech, size of the system's active vocabulary, error-correction procedures and overall accuracy. Distinctions between these features are somewhat arbitrary, however, as they overlap with each other (e.g., accuracy and training or vocabulary size).

[11] http://speech.tifr.res.in/tutorials/fundamentalOfASR_picone96.pdf

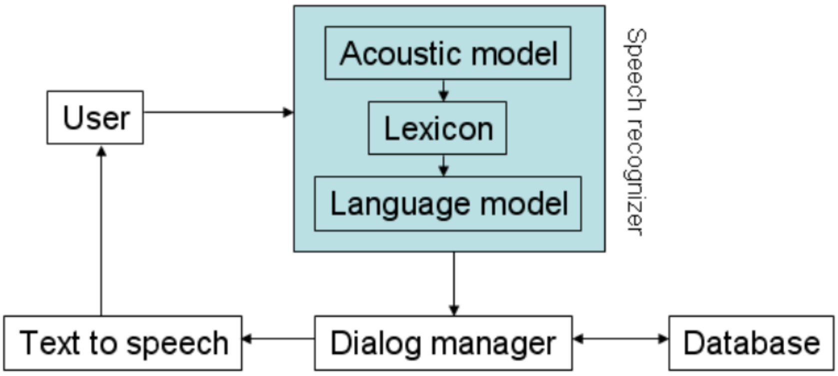


Fig-1.4: A schematic of an automated speech dialog system[12]

2.1.4 Training requirements

Speech recognition systems differ by the degree to which they require an enrolment procedure before using the program. Speaker dependent systems require every user to build a recognition template in the system. This training can require up to several hours to complete, although the processes used differ according to the system. For some systems, a given vocabulary has to be repeated several times, word after word, whereas others end the training as soon as the recognition rate is acceptable, and still others adapt themselves progressively to each speaker during the first hours or days of use. In contrast, speaker-independent systems use previously produced templates provided by system manufacturers, which are generated from a statistical sample of potential users (Milheim, 1993). These systems are especially important in situations in which a large number of people need to interact with the same system . Finally, some SR products (e.g., Naturally Speaking and ViaVoice) work out of the box but their performance improves with training.

[12] http://www.kbs.twi.tudelft.nl/docs/MSc/2006/DALiauwKieFa/thesis.pdf

2.1.5 Speech Continuity

Until recently, most systems used isolated word recognition, which required the user to segment speech into discrete units, pausing briefly between individual words or utterances. Affordable continuous speech recognition systems are now available, allowing users to speak continuously and more naturally, although they also are not completely natural because accuracy is dependent on the user having consistent patterns of pronunciation. Each system requires the user to learn and use a series of commands for routine use, such as saying, "Delete that" followed by a word or phrase that the speaker wants to cut, or "Cap word" to capitalize the first letter of a proper noun in the middle of a sentence. Dragon NaturallySpeaking allows users to choose between using speech or a mouse to format text, and to use commands to control the microphone, move around a document, edit and format text, and play back recorded speech.[13]

2.1.6 Vocabulary

Although systems with small vocabularies are still available, active vocabularies can range from 20,000 to 55,000 immediately recognizable words, depending on the system's cost, and less common words are retrieved from a larger, back-up dictionary. When first starting systems such as Dragon NaturallySpeaking, users receive a standard 30,000-word general vocabulary file containing available words and a language model. Users then customize the active vocabulary and its language model by using a "vocabulary builder," so it more closely matches the content they dictate. In addition, the more consistently a user dictates vocabulary items, the better the system's recognition rate.

13 http://www.teachwise.liepu.lv/faili/composing_via_dictation_and_speach_recognition_sistems.pdf

2.1.7 Error-correction procedures

Mobile access to on-line information is crucial for traveling professionals who often feel out of touch when separated from their computer. Missed messages can cause serious inconvenience or even spell disaster when a decision are delayed or plans change.

A portable computer can empower the nomad to some degree, yet connecting to the network (by modem, for example) can often range from impractical to impossible. The universal telephone, on the other hand, is necessarily networked. Telephone access to on-line data using touch-tone interfaces is already common. A web of invisible, however, often characterizes these interfaces, and tedious orders which result when menu options outnumber telephone keys or when choices overload users' short-term memory.

Conversational speech offers an attractive alternative to keypad input for telephone-based interaction. Implementing a usable conversational interface, however, involves overcoming substantial obstacles, as speech recognition is still a difficult problem, largely because of the many sources of variability associated with the signal. First, the acoustic realizations of phonemes, the smallest sound units of which words are composed, are highly dependent on the context in which they appear. These *phonetic variability's* are exemplified by the audio differences of the phoneme “t” in *two, true,* and *butter* in American English. At word boundaries, contextual variations can be quite dramatic---making *gas shortage* sound like *gash shortage* in American English, and *devo andare* sound like *devandare* in Italian.

Second, *acoustic variability's* can result from changes in the environment as well as in the position and characteristics of the transducer. Third, *within-speaker variability's* can result from changes in the speaker's physical and emotional state, speaking rate, or voice quality. Finally, differences in sociolinguistic background, dialect, and vocal tract size and shape can contribute to *across-speaker variability's*.

2.1.7 Recognition Errors:

Unluckily, the bane of speech-driven interfaces is the very tool which makes them possible: *the speech recognizer.* One can never be completely sure that the recognizer has understood correctly. Interacting with a recognizer over the telephone is not unlike conversing with a beginning student of your native language: since it is easy for your conversational counterpart to misunderstand, you must continually check and verify, often repeating or rephrasing until you are understood.

Not only are the recognition errors trying, but so are the recognizer's inconsistent responses. It is common for the user to say something once and have it recognized, then say it again and have it misrecognized. This lack of likelihood is dangerous. It not only makes the recognizer seem less cooperative than a non-native speaker, but, more importantly, the unpredictability makes it difficult for the user to construct and maintain a useful conceptual model of the applications' behaviour's. When the user says something and the computer performs the correct action, the user makes many assumptions about cause and effect. When the user says the same thing again and some random action occurs due to misrecognition, all the valuable assumptions are now called into question. Not only are users frustrated by the recognition errors, but they are frustrated by their inability to figure out how the applications work.

A variety of facts result in recognition errors. If the user speaks before the system is ready to listen, only part of the speech is captured and thus almost surely misunderstood. An accent, a cold, or an exaggerated tone can result in speech, which does not match the voice model of the recognizer. Background noise, especially words spoken by passersby, can be mistaken for the user's voice. Finally, out- of-vocabulary utterances - i.e., the user says something not covered by the grammar or the dictionary - necessarily result in errors.

Recognition errors can be divided into ***three categories: rejection, substitution, and insertion.*** A *rejection* error is said to occur when the recognizer has no hypothesis about what the user said. A substitution error involves the recognizer mistaking the user's utterance for a different legal utterance, as when "send a message" is interpreted as "seventh message." With an insertion error, the recognizer interprets noise as a legal utterance - perhaps others in the room were talking, or the user inadvertently tapped the telephone.

a. Rejection errors:

In handling rejection errors, we want to avoid the "brick wall" effect - that every rejection is met with the same "I didn't understand" response. Based on user complaints as well as our observation of how quickly defeat levels increased when faced with repetitive errors, we eliminated the repetition. In its place, we give progressive assistance: we give a short error message the first couple of times, and if errors persist, we offer more assistance. For example, here is one progression of error messages that a user might encounter:

“What did you say? Sorry? Sorry. Please rephrase. I didn't understand. Speak clearly, but don't overemphasize. Still no luck. Wait for the prompt tone before speaking. “

As background noise and early starts are common causes of misrecognition, simply repeating the command often solves the problem. Persistent errors are often a sign of out-of-vocabulary utterances, so we escalate to asking the user to try rephrasing the request. Another common problem is that users respond to repeated rejection errors by exaggerating; thus they must be reminded to speak normally and clearly.

Progressive assistance does more than bring the error to the user's attention; the user is guided towards speaking a legal utterance by successively more informative error messages which consider the probable context of the misunderstanding. Repetitiveness and frustration are reduced.

b. Substitution errors:

Where rejection errors are frustrating, substitution errors can be damaging. If the user asks the weather application for "Kuai" but the recognizer hears "Good-bye" and then hangs up, the interaction could be completely terminated. Hence, in some situations, one wants to explicitly verify that the user's utterance was correctly understood.

Verifying every utterance, however, is much too tedious. Where commands consist of short queries, as in asking about calendar entries, verification can take longer than presentation. For example, if a user asks "What do I have today?", responding with "Did you say `what do I have today'?", adds too much to the interaction. Utterance could be implicitly verified by echoing back part of the command in the answer: "Today, at 10:00, you have a meeting with..."

Verification should be commensurate with the cost of the action that would be effected by the recognized utterance. Reading the wrong stock quote or calendar entry will make the user wait a few seconds, but sending a confidential message to the wrong person by mistake could have serious consequences.

c. Insertion errors:

Spurious recognition typically occurs due to background noise. The illusory utterance will either be rejected or mistaken for an actual command; in either case, the previous methods can be applied. The real challenge is to prevent insertion errors. Users can press a keypad command to turn off the speech recognizer in order to talk to someone, sneeze, or simply gather their thoughts. Another keypad command restarts the recognizer and prompts the user with "What now?" to indicate that it is listening again. [14]

2.1.8 Accuracy

An likeness can be made between hand- writing and spelling accuracy in texts produced when writing by hand and in the overall accuracy of SR systems in taking a user's dictation. That is, users are not likely to use SR systems which fail to reach a high level of accuracy in text translation, just as poor spelling and handwriting interfere with the overall quality of a composition. Multiple factors influence accuracy including the similarity of words, the fact that connected speech involves co-articulation (i.e., overlapping phonemes),variability in a user's speech (e.g., having a cold, being fatigued, and saying mispronunciations) or in physical environment (e.g., variable placement of the microphone).

Using words that are spelled differently from the way they are pronounced also leads to system errors. Although Dragon NaturallySpeaking is advertised as taking dictation with high rates of recognition accuracy in the general population, it remains to be seen whether persons, and particularly students with LD, can achieve the claims made in current promotional literature. What kinds of difficulties SR users who have LD may anticipate? First, it is apparent that SR imposes a substantial cognitive load on individuals wishing to compose using an oral mode of production. At this point, it frees users from worrying about spelling and

[14] http://ewh.ieee.org/r10/bombay/news6/AutoSpeechRecog/ASR.htm

handwriting, but it imposes new burdens-careful speech, vocabulary building, explicit punctuation, error correction, and playback and editing procedures, not to mention the initial training requirements.

2.2 The types of Automatic Speech Recognition

Speech recognition systems can be separated in several different classes by describing what types of utterances they have the ability to recognize. These classes are based on the fact that one of the difficulties of ASR is the ability to determine when a speaker starts and finishes an utterance. Most packages can fit into more than one class, depending on which mode they're using.

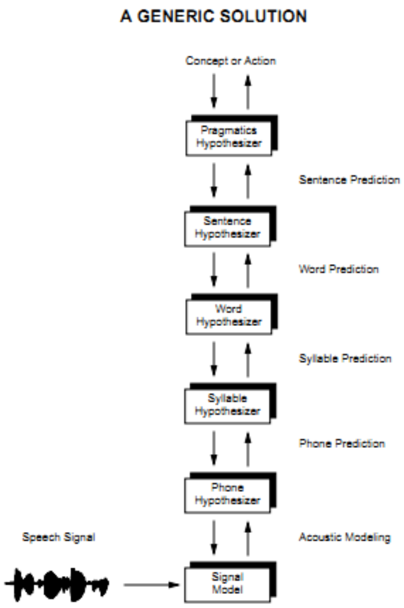


Fig-1.5:A generic Solution [15]

[15] http://speech.tifr.res.in/tutorials/fundamentalOfASR_picone96.pdf

2.2.1 Isolated

Isolated word recognizers usually require each noise to have quiet (lack of an audio signal) on both sides of the sample window. It doesn't mean that it accepts single words, but does require a single utterance at a time. Often, these systems have "Listen/Not-Listen" states, where they require the speaker to wait between utterances (usually doing processing during the pauses). This class of systems might be better called as Isolated Noise class.

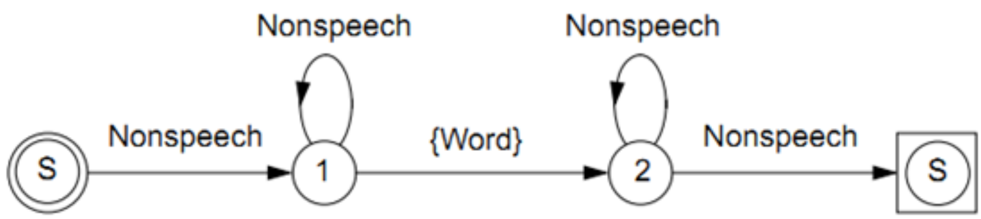


Fig-1.6: Isolated Noise class

No speech: typically an acoustic model of one frame in duration that models the background noise.

{Word}: any word from the set of possible words that can be spoken.

The key point here is that, with such a system, the recognizer finds the optimal start/stop times of the utterance with respect to the acoustic model inventory (a hypothesis-directed search).

2.2.2 Connected Words

Connect word systems (or more correctly 'connected utterances') are similar to isolated words, but allow separate utterances to be 'run-together' with a minimal pause between them.

2.2.3 Continuous Speech

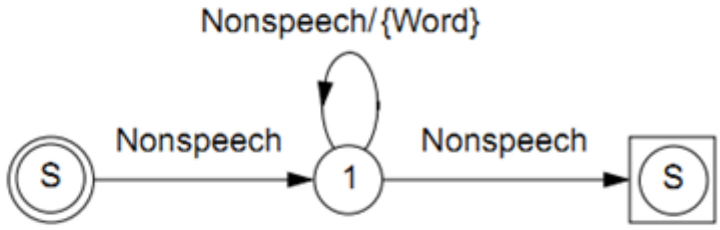


Fig-1.7: Continuous Speech

System recognizes randomly long sequences of words or non-speech events. Continuous recognition is the next step. Recognizers with continuous speech capabilities are some of the most difficult to create because they must utilize special methods to determine noise boundaries. Continuous speech recognizers allow users to speak almost naturally, while the computer determines the content. Basically, it's computer dictation.

2.2.4 Spontaneous Speech

There appears to be a variety of definitions for what impulsive speech actually is. At a basic level, it can be thought of as a speech, which is natural sounding and not rehearsed. An ASR system with spontaneous speech ability should be able to handle a variety of natural speech features such as words being run together, "ums" and "ahs", and even slight stutters. Impulsive, or extemporaneously generated, speech contains disfluencies, and is much more difficult to recognize than speech read from script.

2.2.5 Voice Verification/Identification

Some ASR systems have the ability to identify specific users. Such a class of verification systems is used for security and similar systems.[16]

[16] http://ewh.ieee.org/r10/bombay/news6/AutoSpeechRecog/ASR.htm

If an application is speaker-dependent (single speaker) it means that user has to train the program to recognize his speech, this type of application has got the highest recognition rates. Independent software uses a default set of discrete sounds what cause lower recognition rates, this is usually used in telephony applications. In IWR (Isolated) systems speaker has to make long breaks between words, almost the same as in CWR (Connected words), but there those pauses can be much shorter. In CSR (Continuous Speech Recognition) the speaker can speak fluently, without any breaks. This type of systems has problems detecting individual words for e.g. in the phrase “I scream for ice cream”.

2.3 The Practical Side of ASR

Although any task that involves interfacing with a computer can potentially use ASR, the following applications are the most common right now.

- **Dictation**

  Dictation is the most common use for ASR systems today. This includes medical transcriptions, legal and business dictation, as well as general word processing. In some cases special vocabularies are used to increase the accuracy of the system.

- **Command and Control**

  ASR systems that are designed to perform functions and actions on the system are defined as Command and Control systems. Utterances like "Open Netscape" and "Start a new xterm" will do just that.

- **Telephony**

  Some PBX/Voice Mail systems allow callers to speak commands instead of pressing buttons to send specific tones.

- **Wearable**

  Because inputs are limited for wearable devices, speaking is a natural possibility.

- **Medical/Disabilities**

  Many people have difficulty typing due to physical limitations such as repetitive strain injuries (RSI), muscular dystrophy, and many others. For example, people with difficulty hearing could use a system connected to their telephone to convert the caller's speech to text.

- **Embedded Applications**

  Some newer cellular phones include C&C speech recognition that allow utterances such as "Call Home". This could be a major factor in the future of ASR and Linux. Why can't I talk to my television yet?

Speech-to-text software, sometimes known as dictation software, is something that lets you talk to the computer in some form and have the computer reacts appropriately to what you are saying. This is totally different to text-to-speech software, which is software can read out text already in the computer. There are wide varieties of companies working on the issue of speech recognition. Today the most popular speech recognition software's for Windows are ViaVoice (IBM product), Dragon Naturally Speaking (ScanSoft) and Microsoft Speech API.

- **ViaVoice (IBM product):**

IBM ViaVoice is a range of language-specific continuous speech recognition software products offered by IBM. The current version is designed primarily for use in embedded devices. **Embedded ViaVoice delivers IBM speech technology to mobile devices and automobile components.** In 2003, IBM awarded ScanSoft, which owned the competitive product Dragon

NaturallySpeaking, exclusive global distribution rights to ViaVoice Desktop products for Windows and Mac OS X. Two years later, Nuance merged with ScanSoft.[17]

- Provides fully integrated, automatic speech-recognition and text-to-speech capabilities for small mobile devices, including automotive telematics systems and hands-free phones

- Helps minimize the skills and time needed to develop advanced voice applications for devices and remote systems

- Includes freeform command support for natural and intuitive command phrases that do not need to be known ahead of time or memorized for future use

- Recognizes vocabulary lists exceeding 200,000 spoken words in real time and across a broad range of languages

- Includes porting, integration, testing and consulting services provided by IBM, along with customized development workshops

- Provides an intuitive, easy-to-use developer toolkit powered by Eclipse technology

2.3.1 Functionality

- Portable, event-driven architecture
- Fully integrated automatic speech recognition (ASR) and text-to-speech (TTS)
- Low processor utilization
- Small static and dynamic footprint

[17] http://en.wikipedia.org/wiki/IBM_ViaVoice

- Scalable, modular architecture
- Single-threading and multithreading support
- Runtime event notification
- Unsupervised adaptation to speakers
- Optional speaker enrollment
- Phoneme-based
- Speaker-independent[18]

Operating Systems Supported

- Windows XP
- Windows 2000
- Windows CE / Windows Mobile
- QNX
- Linux
- Embedded Linux
- T-Engine
- MicroItron
- VxWorks
- RTXC

Languages Offered

Automatic Speech Recognition (ASR)

- US English
- North American Spanish
- Canadian French
- UK English
- French
- Italian
- German

[18] http://www-01.ibm.com/software/pervasive/embedded_viavoice/about/

- Spanish
- Dutch
- Japanese
- Mandarin Chinese
- European Portuguese
- Swedish
- Korean

Concatenative Text-to-speech (eCTTS)

- US English
- North American Spanish
- Canadian French
- UK English
- German
- French
- Italian
- Spanish
- Japanese
- Dutch

Formant Text-to-speech

- US English
- North American Spanish
- Canadian French
- UK English
- German
- French
- Italian
- Spanish
- Japanese
- Dutch

- Simplified Chinese
- Brazilian Portuguese
- Korean

Dragon Naturally Speaking (ScanSoft):

Dragon Naturally Speaking is a speech recognition software package developed and sold by Nuance Communications for Windows personal computers. The most recent package is version 11.5, which supports 32-bit and 64-bit editions of Windows XP, Vista and 7. The Mac OS version is called Dragon Dictate.

Key Benefits

- *Turn Talk Into Text*: Say words and watch them appear on your computer screen in Word, Word Perfect, Excel, Outlook, and more -- three times faster than typing -- with up to 99% recognition accuracy right out of the box.
- *Unlock Your Creativity*: Transform ideas into text at the speed of thought; don’t let typing slow you down. Play back what you’ve written for easy proofing.
- *Work Comfortably*: Control your PC in a relaxed, hands-free mode without being tied to your keyboard. Say goodbye to repetitive stress injuries.
- *Use Virtually Any Windows Application*: Create reports, spreadsheets or presentations; send email or schedule meetings; surf the Web; update your Facebook status or download your favorite music -- using just your voice.
- *Use Your Favorite Applications*: Dictate documents; send email and instant messages, surf the Web, and more -- using just your voice.
- *Multi-Task Like Never Before:* Tell your PC what to do, like “Send email to Jon Smith and Raphael Sanchez” or “Search the Web for Internet

marketing companies in Boston, Massachusetts” to work faster and smarter. Complete multiple steps with a single voice command.

- *Work Your Way*: Personalize Dragon with custom word lists and voice commands that reflect the kind of work you do; set formatting preferences, too.
- *Be Productive On The Go*: Use a Nuance-approved digital voice recorder from anywhere, at anytime and Dragon will automatically transcribe the audio files when you return to your PC. Wireless microphone support delivers even more convenience.

Chapter 3: Design

Overview

Thus this chapter, whose purpose is to present the prototype application for deaf students, begins with a brief description of the overall architecture, focusing on our system application how we will design with help of deployment, stake holders, use cases and real time scenario's etc.

POLILIPS is windows based application.

3.1 Stakeholder

Stakeholders are individuals, groups of people, that are actively involved in our project, are affected by its working, or can influence its outcome. The main two stakeholder profiles are speech impairment students and teachers identify the user for this application.

3.2 Operating Environment

This section mainly defines the environment in which the system application will be used and it also defines the major availability, reliability, and performance and integrity requirements of the system.

- The users may access the system application at any time during the class hours to take part in class discussions.

- Different types of this application user will be arranged into several user groups with different kinds of authorities. Such as the application administrator will be granted the highest authority to ensure the successful running of this application, and the student users have the rights to access the system as client operating environments on their desktop applications.

3.3 Architecture

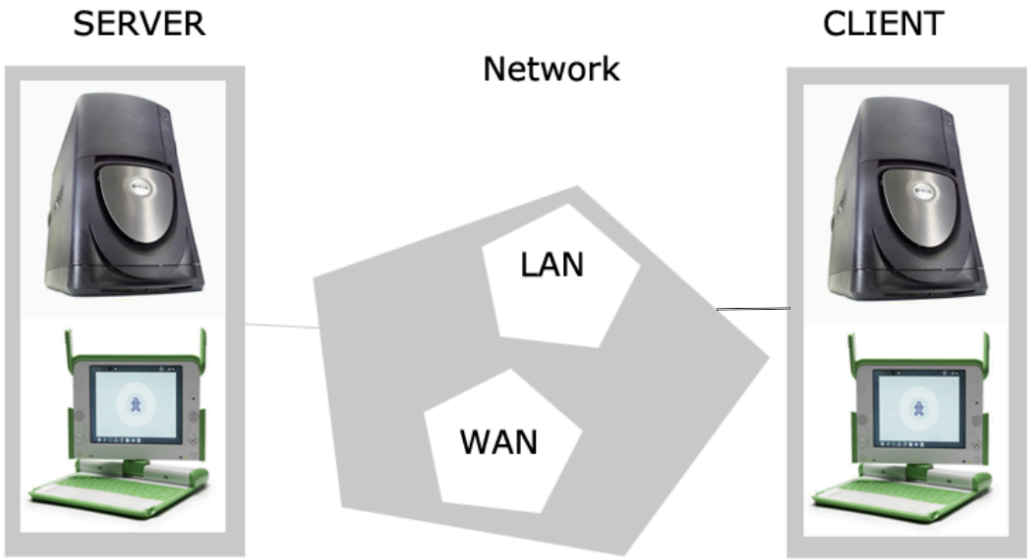


Fig 3.1: Over all architecture for POLLIPS

Comments
The deployment diagram captures the remote access and centred control principles of the POLILIPS. Generally, the remote device communicates with the POLILIPS server through a wireless interface. POLILIPS server or the main hub is physically connected to all the computer devices either on wireless access notebooks or desktop computers in network. All information about the login student users, devices, is stored on the runtime of the application. The functions of all the main nodes are described as below:

- Remote or desktop computer are the running and configures environment of the POLILIPS. It can be Laptops, desktop PCs.

  - **Client Computer:** It will enter username and server ip to connect his device, displays main display panel of class-room.
  - **Server Computer:** It will focused on the and send and recieve coomunication by the mean of recieving text or stream speech converted into text message and video to client computer.

- Server computer is the pivot of the POLILIPS application. It can connect/disconnect devices from the network and give privileges the access main application features, track the status of the users in class room and communicatewith them by server.

3.4 Operational States

The POLILIPS is operated under the control of administrator and students only. There is no interface allowing other people to interact directly to POLILIPS. We can assume both modules are under one polilmi system.

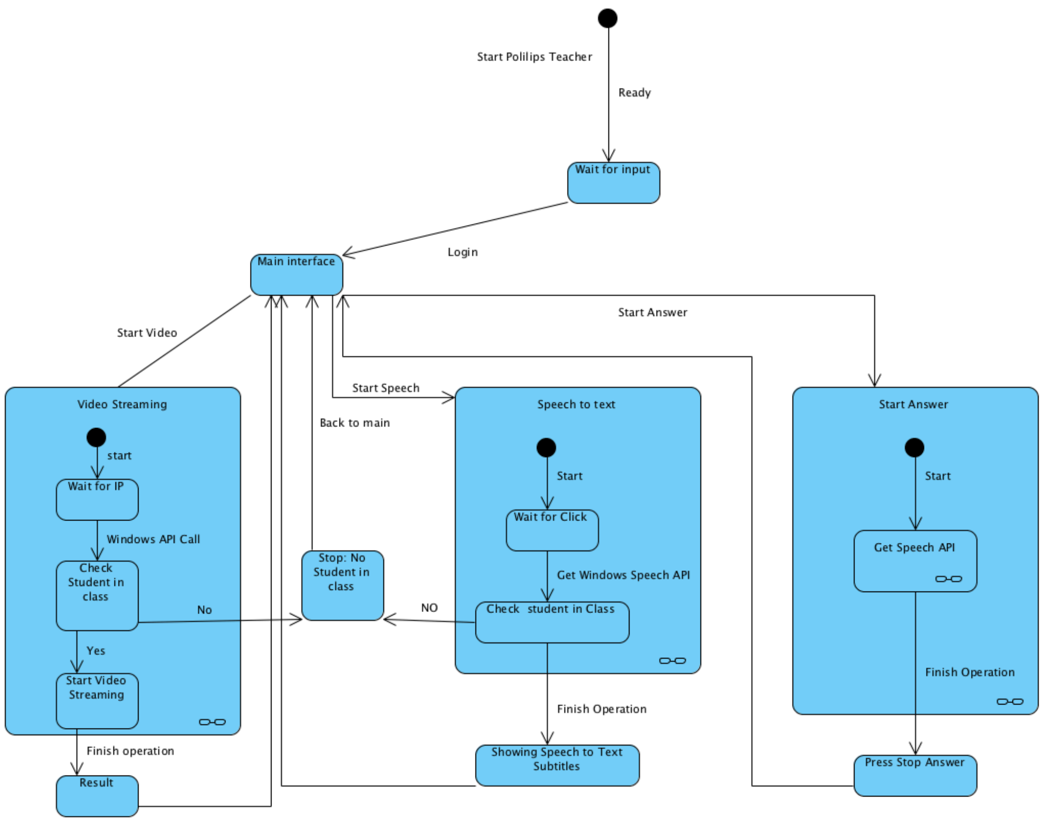


Fig 3.2 : POLILIPS Server operate by Teacher.

This is the overall state chart diagram of our application, showing the general states POLILIPS goes through during its operational lifetime.

First we have to run server application. It's operate by teacher.

- When the system is initiated, it enters to the “Wait input event” state. The system will be waiting for further instructions to login in our case user will enter name.

- Once there is a login request performed, POLILIPS enters the ‘Main User interface’ state in which the user obtains a choice opportunity with POLILIPS functions. If the user wants to access to the “video streaming” state, he is prompted to enter the his system IP and press for start video streaming through client users before streaming it will verify at one student client system is connected to server if yes streaming will perform if no it will come back to main interface by giving errors.

- Once the user perform video streaming now need to enable speech, by pressing start speech, before performing successfully it will check if system at least one client connected. It will enable function from POLILIPS for Speech to Text, to display text subtitle under the video screen.

- If client user ask answer server will get alert in form of voice and if wish to answer will press answer button form main interface and then will perform function same speech to text for client user, after completion user must press stop and then this answer for client computer will store in answer archival box.

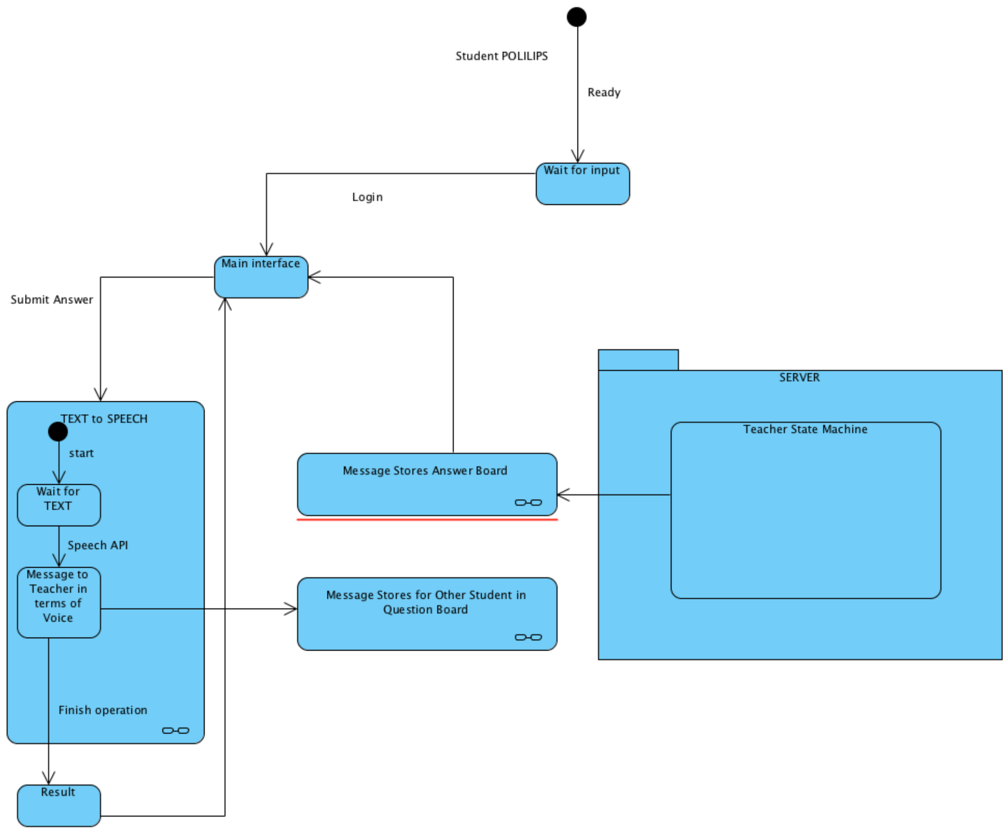


Fig 3.3: POLILIPS client operate by Student.

This is the overall state chart diagram of client side module, showing the general states POLILIPS goes through during its operational lifetime.

Before start this application client must make sure server is running.

- When the system is initiated, it enters to the "Wait input event" state. The system will be waiting for further instructions to login in our case user will enter name and server IP to connect with server.

- Once there is a login request performed, POLILIPS student enters the 'Main User interface' state in which the user obtains a choice opportunity with POLILIPS functions. If the user wants to ask any question form teacher "server" will tupe answer and will send to teacher teacher will hear answer and after submission perform successfully message will store in Question Asked by student archival which only client side computer can see, for alerting all studying in class for communication monitoring.

The following operational states exist for the system instance:

- **Network:** The system instance has been started, and users can log on and operate.
- **Offline:** The system instance is not running on network.

3.4.1 Main Functional Constraints

Code	Constraint	Description
C001	Distributed System	The working environment is based on a distributed system structure.
C002	Dot Net technology	Developing technology using Dot Net programming language.
C003	Hardware	The hardware to stream video and audio signals.

Table 3.1: Functional Constraints

3.4.2 Assumptions

The project is constrained by the following assumptions:

- User can operate computer, configured on LAN network.
- Server is always on.
- Communication Link exists between client and the server devices.

3.7 UI Prototypes

The user interface requirements for POLILIPS are as follows:

- The user interface will be efficient in terms of both speed and user interaction.
- To make sure each group of users such as administrator and clients perform the corresponding activities; each user group will be granted with different user interfaces according to the group's authority, i-e student and teacher module.
- There is an error prevention to avoid problems occurring while system is on process.
- There is an easy-to-use interface to ensure smooth navigation as the user works with the application.

3.6 Use Cases

Use case diagrams are intended to model the functional requirements of POLILIPS. It shows a set of use cases, actors and their relationships.

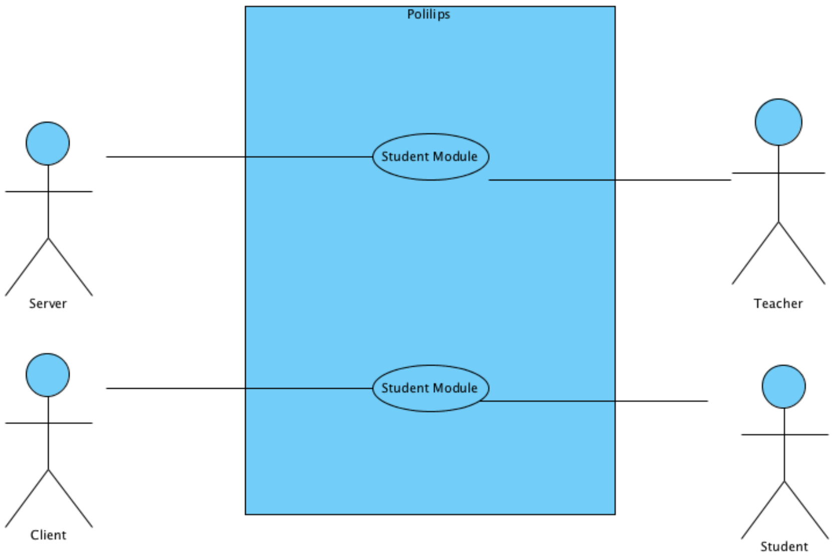


Fig 3.4: Over System for POLILIPS

3.6.1 Use case 0: Overall system of the POLILIPS

Brief description

The user case shows the whole view of POLILIPS process.

Scope

This focuses on the main hub of application, with two descriptive use-case

Level

Summary Level

Primary Actor

Server, Client

Preconditions

- Server will login and create streaming.
- Clients will connected with server to get streaming

Minimal guarantee

Users can view the working status of each module.

Success guarantee

Users can not only view the working status of each module but also perform changes and make action to do communication.

Related Information: Use case diagram

Extensions

- The teacher module activated.
- The student module activated

3.6.2 Use case 1: Teacher Module

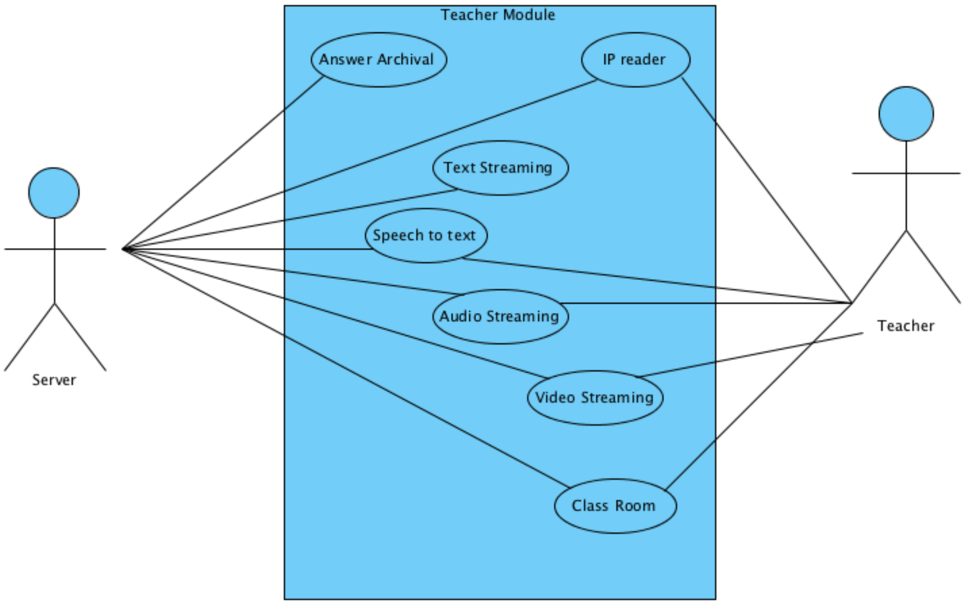


Fig 3.5: Teacher Module

Primary Actor

Server / Teacher

Scope

This use case is for the initial view of the teacher server side application. It is used to send and receive audio to text subtitle, video streaming, and can control stop and start streaming communication.

Level

User-goal level

Minimal Guarantees

No one is login to server.

Success Guarantees

Teacher is login to server.

Preconditions

- Only the administrator / teacher is authorized to make live streaming and connect to clients.
- The server will identify and connect to the added devices correctly.

Trigger

There is a need to through streaming to client devices.

Main Success Scenario

- The administrator teacher logs into the application.
- The system will display teacher control panel.
- The administrator then through streaming with given of server IP to client user devices and allow to connect to server to see video streaming and subtitle and communication channel for discussion.
- Teacher can see who is entering into classroom.
- Teacher will be busy to deliver the lecture but any question if client student ask so teacher will hear into voice.

Sequence Diagram

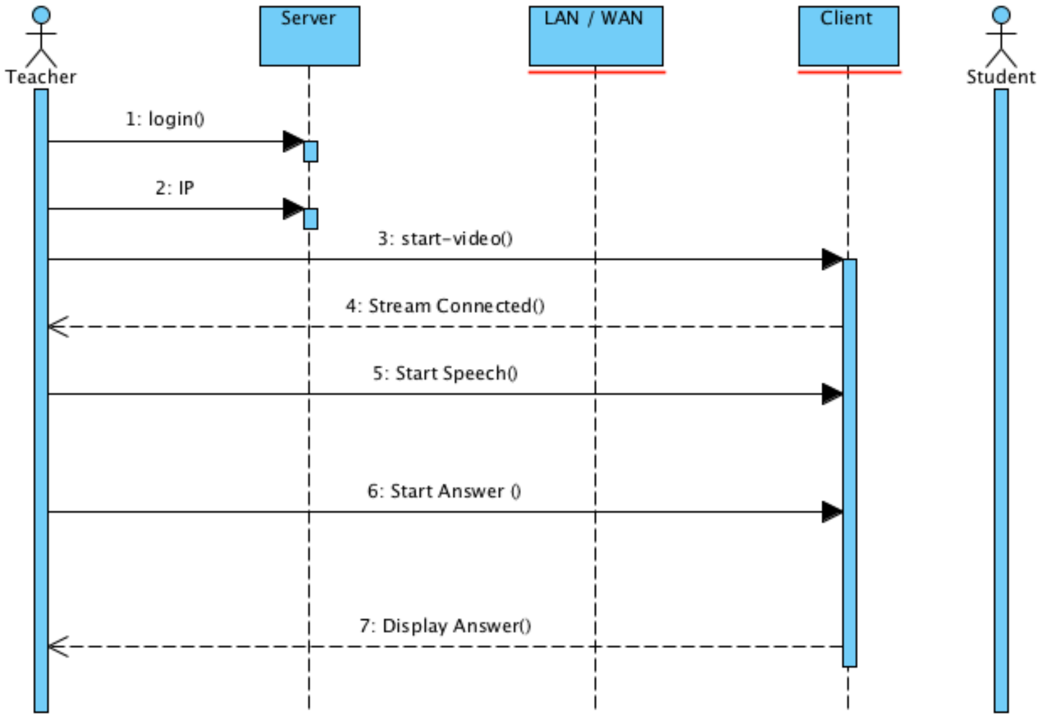


Fig 3.6: Sequence Diagram for teacher module

3.6.3 Use case 2: Student Module

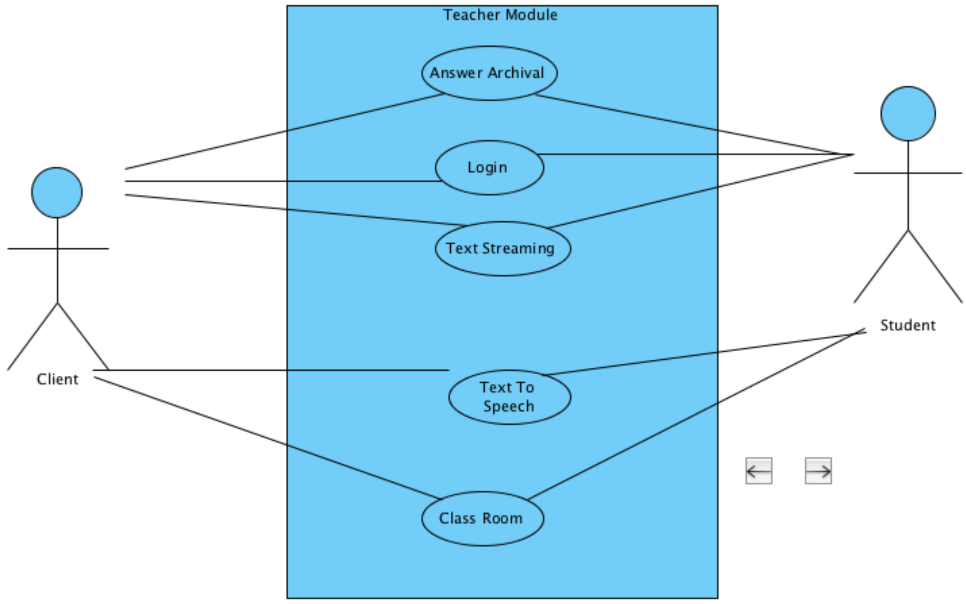


Fig 3.7: Use-case for Student Module

Primary Actor

Client / Student

Scope

This use case is for the initial control of the student client side application. It is used to send message in terms of question during the lecture in form of text to audio.

Level

User-goal level

Minimal Guarantees

No one is connected to server.

Success Guarantees

Student is connected to server.

Preconditions

- Only the student know server IP to get live streaming and communication.

Trigger

When student will ask question will type and submit question. But on server side teacher will hear inform of voice.

Main Success Scenario

- The student logs into the application, with provided information of IP and name.
- The system will display Student control panel.
- When connection is established with server, will automatically start getting video stream, audio-text converted subtitles will be displaying.
- Student can see who is entering into classroom.
- Student can see previous archive for lectures as well asked question in the classroom on student board.
- Teacher will be busy to deliver the lecture but any question prompt student will submit and send message to server for prompt answer as well will be stored to notice board.

Sequence Diagram

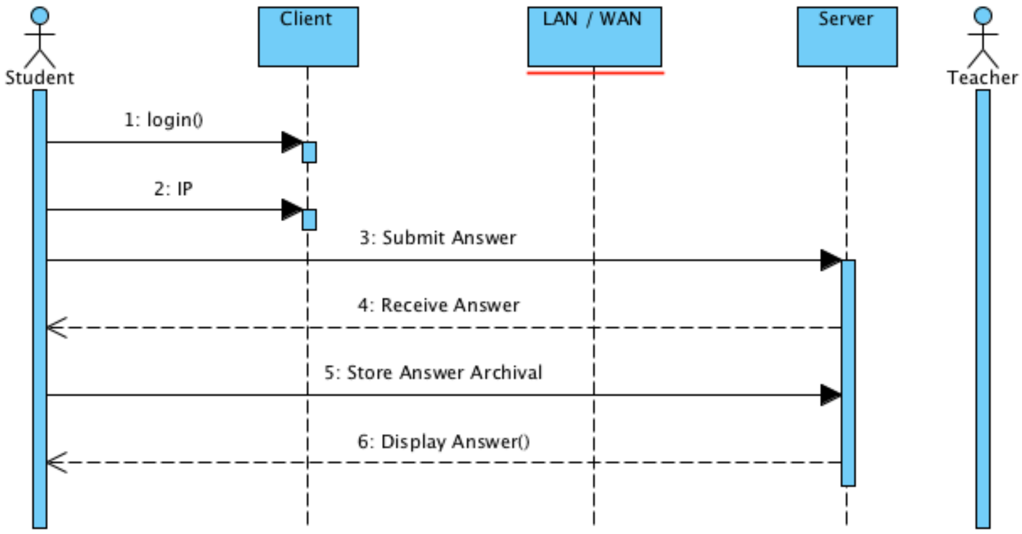


Fig 3.8: Sequence diagram for Student Module

3.7 Dynamic Aspects of the System

For the most part, modelling the dynamic aspects involves modelling the sequential steps in a Polilips process. An activity is an on-going structured execution of behaviour.

3.8 POLILIPS Perspectives

POLILIPS is a application which is controlled by a server and client, respectively as teacher and student on laptops through wireless and desktop computers through LAN network.

3.8.1 POLILIPS Features

Code	Function name	Description
HS001	Polilips Student	Client user will run this application, need to insert login details with server IP and Student name to access control panel.
HS002	Polilips Teacher	Server user will run this application, need to insert login details with teacher name to access control panel.
HS003	Video Streaming	This function in server side taking video streams from builds in or wireless camera and publishing this stream to client computers.
HS004	Speech to Text	This is available on server side, when person is speaking its automatically converting into text and rending this text subtitle to network for client computers.
HS005	Text to speech	This is a client side function when user has to ask question they will type message and submit to server end will receive into voice.
HS006	Text streaming	This is a communicating runtime text to both server and client end users.
HS007	Lecture Archival	Send all messages to archival for all previous lectures.
HS008	A&Q Archival	Client end for other user infomration this is storing Q& A archival.
HS009	Class Room	When client will login by given Server IP connection will established with server and will show presence

		in clasroom tab.
HS0010	Date	Showing cruntt date of the day.

Table 3.2: Features of POLILIPS

3.8.2 Operating Environment

- Architecture: Distributed System
- Local Operating System: Windows XP and later version
- Remote Operating System: Samsung windows tablet, Desktop users, Labtops.

3.8.3 Design and Implementation Constraints

- **Security**

  Security for the POLILIPS includes authentication, to access control panel for serve and client end. Authentication of the user at server end is by entering name.

- **Usability**

  Easy to use, simple interface for disable person to use).

- **Responsiveness**

  System responds quickly to user requests or changes in the environment. System responds within max 2 seconds delay on average to local user requests and changes in the environment.

3.8.4 Assumptions and Dependencies

- Only a future work will leads to POLILIPS is available with full benefits based on our basic version.

- User can operate computer on LAN, Wireless enabled remote devices.
- Servers are always on.
- Communication Link between server and the client devices.
- While designing the initial system I have assumed there are no power failures.

3.8.5 Functional Requirements

- POLILIPS will be able to communicate with all kinds of windows operation systems.
- Provides user authentication on server side with IP confirmation.
- Allows the user to add, server IP, start and stop streaming for video and audio.

3.9 User Interface

User interface is crucial to the use of the application for the users. The user interface must be secure, convenient and extensible. Efficiency: The user interface will be efficient in terms of both speed and user interaction.

- Error Prevention: There is an error prevention to avoid problems occurring while system is on process

- User Friendly Interface: Easy-to-use interface to ensure smooth navigation as the user works with the application Flexible and efficient menus, labels, controls, easy to remember and recognize.

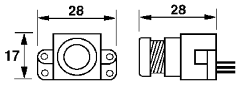
3.9.1 Hardware Interface

There are list of hardware we are using to operate POLILPS are given below

Fig 3.9: TELECAMERA CMOS COLORE OBIETTIVO 3,6mm

SPECIFICATIONS

- Operating system: PAL
- Sensing element: CMOS COLOR 1 / 3 "
- Number of pixels: 330K
- Horizontal resolution: 380 TV lines
- Sensitivity: 3 Lux (F1.2 With)
- Sync: Internal
- Electronic Shutter: 1 / 50 ÷ 1/15.000
- Viewing angle: 92 °
- Lens: f = 3.6 mm/F2.0
- Range: 0.45
- Video signal level: 1 Vp-p 75 ohm composite
- Supply voltage: 12 Vdc to 50 mA
- Working temperature: -10 ° C to + 45 ° C



Measurements in millimeters.

Transmitter and Receiver TX A/V 4

Fig 3.10: Transmitter and Receiver with Microphone

Miniature video transmitter with a microphone input operating at 2.4 GHz where the signal can be received by the receiver (code FR137).
The module includes the stages of the composite video input (1 Vpp to 75) while the audio is taken from the supplied microphone capsule, the transmit frequency is selectable from 4 different values : 2.413 / 2.432 / 2.451 / 2.470 GHz by means of a switch.

It has an output stage that ensures a power of 10 mW @ 75 ohm antenna granted to 1 / 4 Wave (supplied with the module). Specifications: Power supply 12Vdc, 140 mA consumption, size 40 x 30 x 7.5 mm, weight 17 g.

Receiver

Fig 3.11: Receiver for Auido & Video Signals

This receiver will allow us to receive audio and video signal of 4 frequency bands, selectable by a switch (from 2.413 to 2.432 - 2,451 to 2,470 GHz) transmitted from the module FR135, FR170, FR171 or FR172.

Are received through the tuned antenna supplied with the product. Composite video output: 1 Vpp into 75 Ohm, audio output: 2 Vpp max. that also can be applied at a monitor or can be sent to a regular TV using the SCART socket. 12Vdc Dimensions: 115 x 23 x 80 mm, the handset comes with connection cables.

E-CAPTURE: DATA ACQUISITION OF AUDIO / VIDEO TO PC WITH USB PORT

Fig 3.12: E-CAPTURE: DATA ACQUISITION OF AUDIO / VIDEO TO PC

This device hardware allows you to turn you PC into a true DVR. It has RCA video input and S-Video and two audio input. Complete with software for image management.

SPECIFICATIONS

Interfaces: USB 2.0

- Data transfer rates up to 480 Mbit / s, 48 times faster than USB 1.1 interface
- Video formats supported: 176x144, 320x240, 352x288, 720 × 576, 1440x1152
- High image quality

Image Adjustment:

Brightness, contrast, hue, saturation Two video inputs: RCA female, S-VIDEO

- Audio Inputs: AUDIO-L / R AUDIO (RCA female)
- MINIMUM SYSTEM REQUIREMENTS
- Processor speed 1.8 GHz or higher
- 256 MB of RAM
- Hard drive with 10 GB of free space
- Operating System: Win2000, WinXP
- USB port 64-bit graphics card

Rechargeable Battery 1.2V: 4 P.zi

Fig 3.13: Batter Pack

Charger for battery CELLS FROM 2 TO 10 (2.8 to 14 V)

Fig 3.14: Charger for battery cells

Headphone

Fig 3.15: Headphone for PLOLILIPS

We will connect our headphone and microphone through receiver and transmitter.

3.9.2 Software Interface

We will use visual Studio, for Dot net framework 3.0.

- Usability
  - Simple to Operate: The software is easy to learn and operate; the user will not require special skills or training to operate the POLILIPS.
  - Simple design: We try our best to designed in such a way that the user will find it easy to interact with the POLILIPS.
- Reliability
  - POLILIPS up and can run 18hours/day and 365 days and will be crash safe during 95% of its runtime.
- Performance
  - Short response time: The POLILIPS application will not take more than 3 seconds to load onto the Computer Device.

Chapter 4: Implementation

4.1 Overview

POLILIPS is a " basic foundation leads to future direction Proof of Concept" for development of a digital platform to teach deaf students with the help of visual and speech recognition technologies.

The application has two modules, basis on design chapter discuss previously we will implement our application as given below are two main modules of our application.

- Teacher
- Student

4.1.1 Teacher is server-based module.

This module is for teachers to deliver lectures and interact with students. Teacher delivers lectures through a PC with a wireless video camera and microphone attached. Video camera captures his live video lips reading and renders that to student applications on network. While his voice recorded through microphone is converted to text (by speech to text technology) and appears below the teacher's video on "POLILIPS Teacher" and "POLILIPS Student" both application screens.

Teacher's module also shows a questions text stream, which is asked by students through their application. Teacher answers those questions, which appear in answers text stream on student applications screen.

4.1.2 Student is client-based module.

This module is for students, which they log in simply by inserting their name and server IP as login credentials. It shows live video rendered from teacher module and text stream of his audio. It also shows questions from all students and answers from teacher.

4.2 Technologies used

- .NET Framework 3.5
- Windows Speech API
- Windows Sockets API
- Windows Video (avicap32) API

All speech to text features is developed using Windows Speech API and rendered on network using Windows Sockets API. Question and answer streams are also rendered on network using Windows Sockets API.

Video capturing feature is developed using Windows Video API and rendered on network using Windows Sockets API.

Multi threading is used to control all real-time activity.

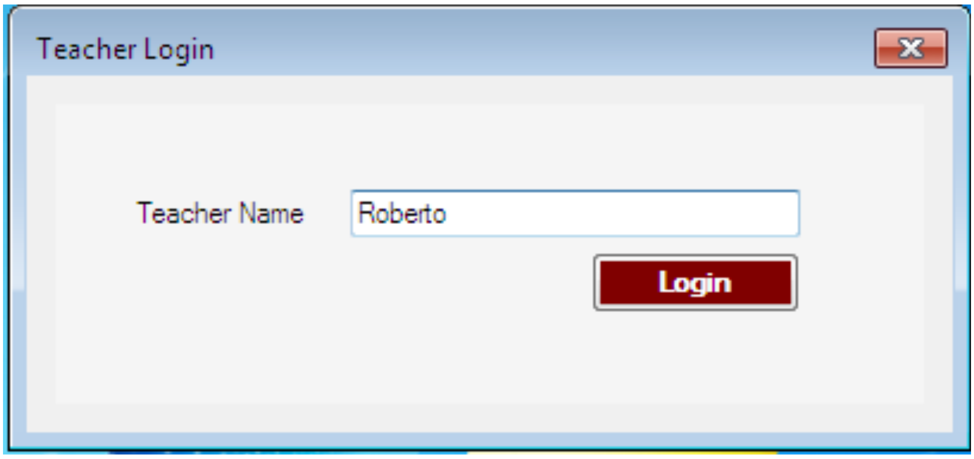


Fig 4.1: Login details for Teacher

Teacher, module login details need to insert, after insertion user will see control panel for POLILIPS application.

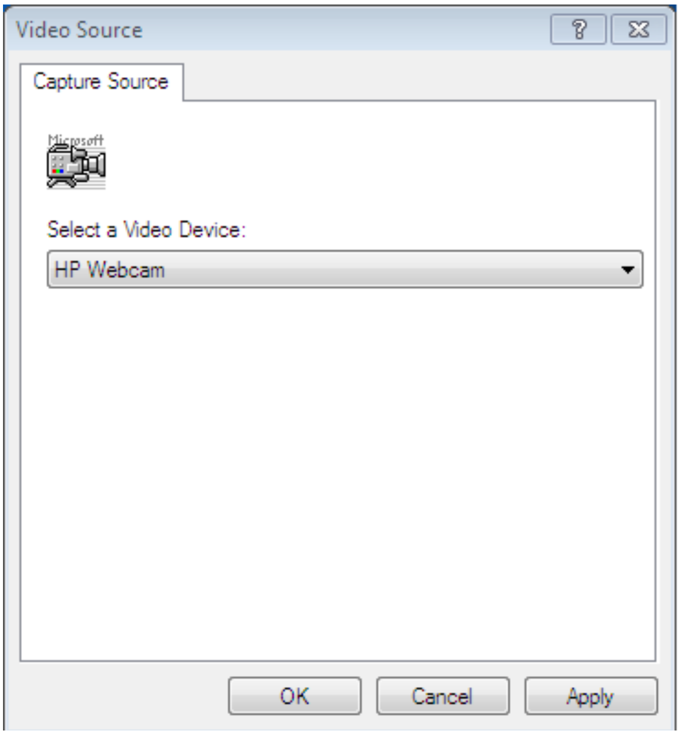


Fig 4.2: Select Camera Device

before running application, system will ask source for video camera, in our case we are using wireless camera, so we will not use default camera device, we will explain later how to configure wireless camera, and how to receive camera stream, using our above given hardware in design chapter.

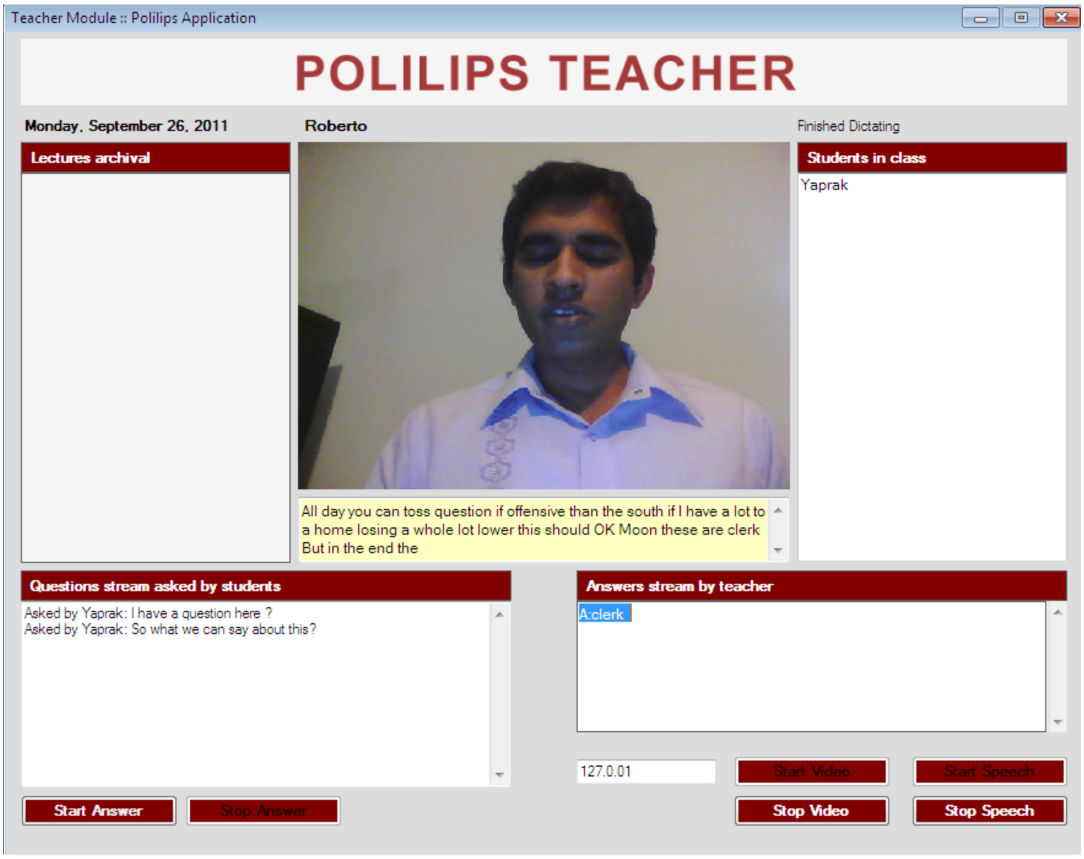


Fig 4.3: POLILIPS Teacher Module

This is POLILIPS teacher control panel screen short where user will navigate through the application, In this we can see discussed features in design chapter, in implementation, in top left corner is date and just under below is Lectures Archival, in centre With name of teacher, and streaming coming from wearing camera device, under video camera screen, is given speech to text subtitles, this subtitles is speech converted text, using speech API, its accuracy is deepened on training of for API, training wizard module will start when we will press on start speech button in interface, this wizard a new user have to compile once after that user will can use this API, In bottom left corner button are given “Start Answer” and “Stop Answer”, We will describe later functionality of this feature in this chapter. In Bottom right corner button are given for start streaming after enabled connection with at least one client machine. Above that is archival of answer given by teacher. We will explain how these feature are working later in this chapter.

Now before go live for video camera streaming, we need client connection for communication at least one user, in our case we are using single standalone machine so we will use this application as localhost computer, it means our machine is behaving like server and client on same machine.

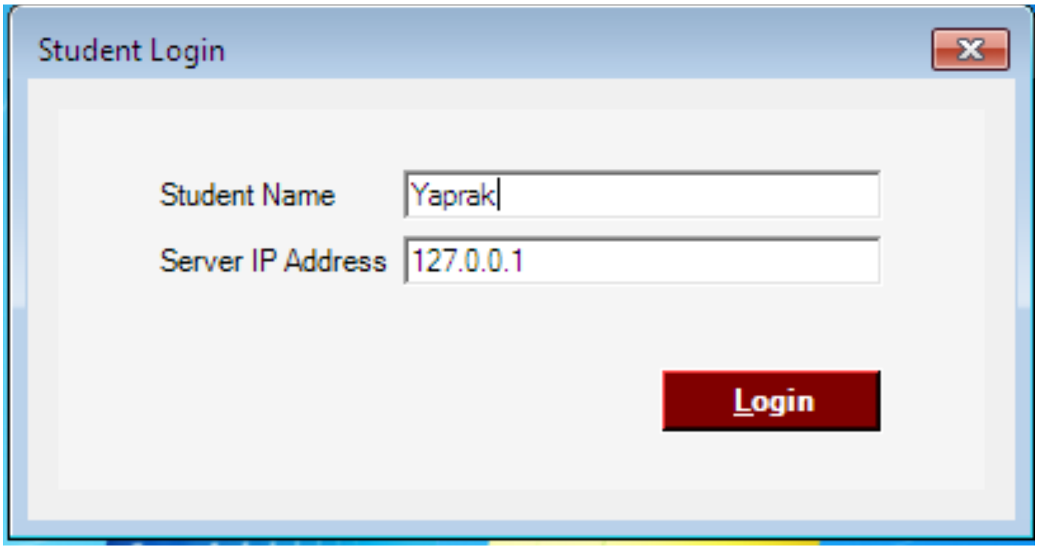


Fig 4.4: Login details for Student

Now we will run student module, to make connection to server we will insert IP of server machine in our case we will insert localhost server IP.

In fig you can see student will insert his/her name and will give server IP details to access control panel.

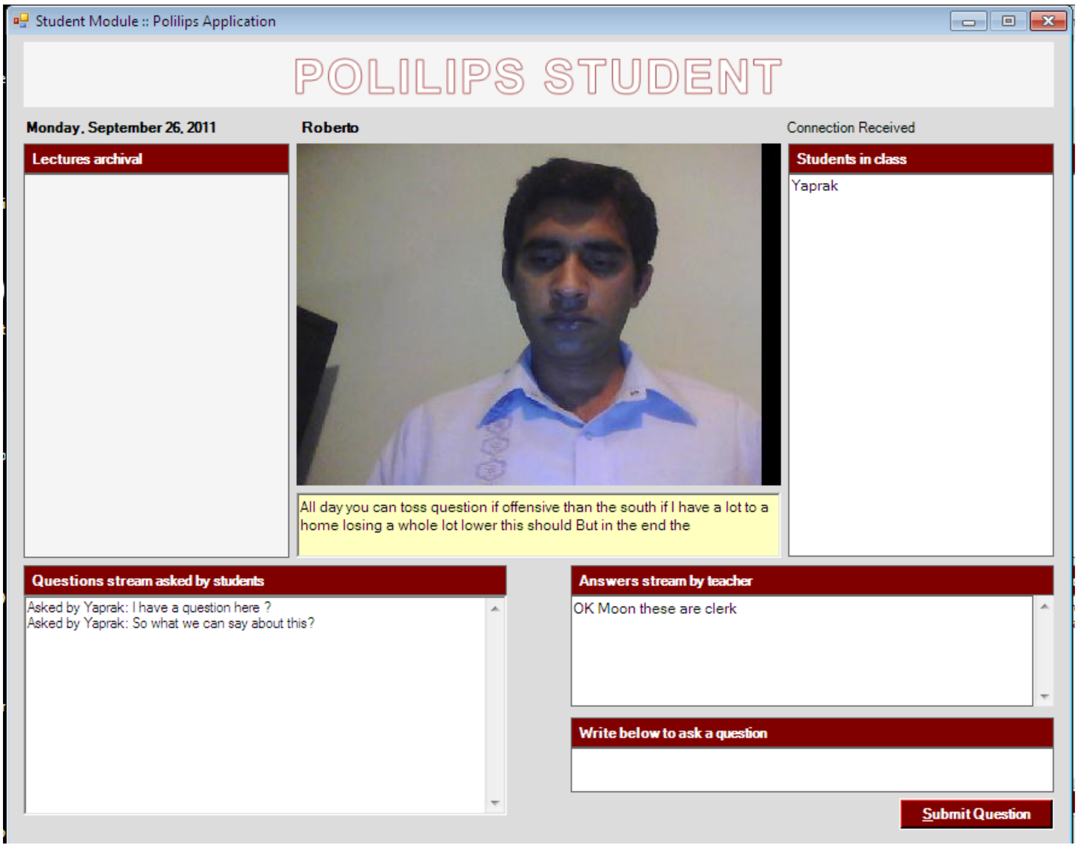


Fig 4.5: Interface for Student Module With Server side streaming

This is a POLILIPS student module control panel in above given fig, where user will navigate through the application, in top left corner is date and just under below is Lectures Archival, in centre With name of teacher, and streaming coming from wearing camera device by server, under video camera screen, is given speech to text subtitles, this subtitles is speech converted text, using speech API are real time streaming, In bottom right corner button are given “Submit Question”, here user will enter text to ask question because user is deaf so cant speak also so privileges give to user for easy communication with teacher, We will describe later functionality of this feature in this chapter.

Now we will describe how to navigate though out POLILIPS application, we try our best to make interface user-friendly, we will describe more about usability and accessibility after this section.

Now we need to assemble our wireless streaming hardware which teacher will be wearing, without any problem teacher can deliver his/her lecture using this hardware. We already discussed our required hardware, in design chapter now needs to implement in right way to communicate audio, video streaming for client users.

Fig 4.6: Battery Charger Devices

In above given fig we are implementing battery case to give power to camera and microphone and hence camera and microphone can send signals to receiver, before using devices we have to make sure our battery is perfectly charged.

Now we will assemble, tele camera, transmitter and receiver. As you can see in fig given below we attached our camera and microphone transmitter with headphone.

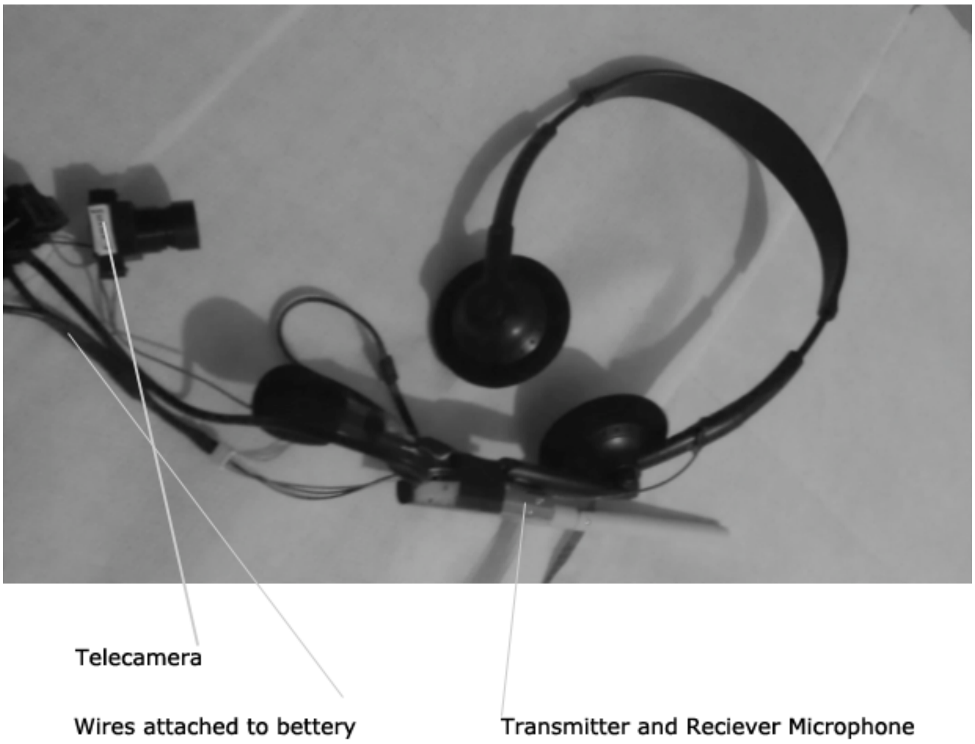


Fig 4.7: Assembled Hardware for teacher.

We can see given below picture how it looks physically after wearing hardware, we make angle for camera with 90* focus to face for catching lips reading.

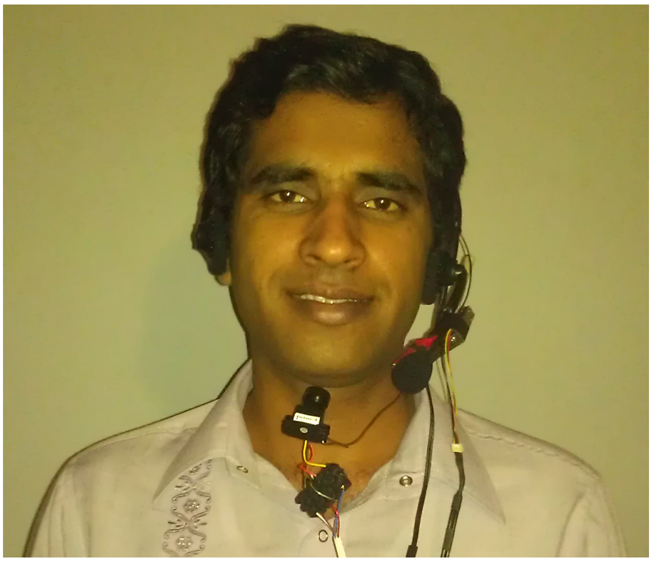
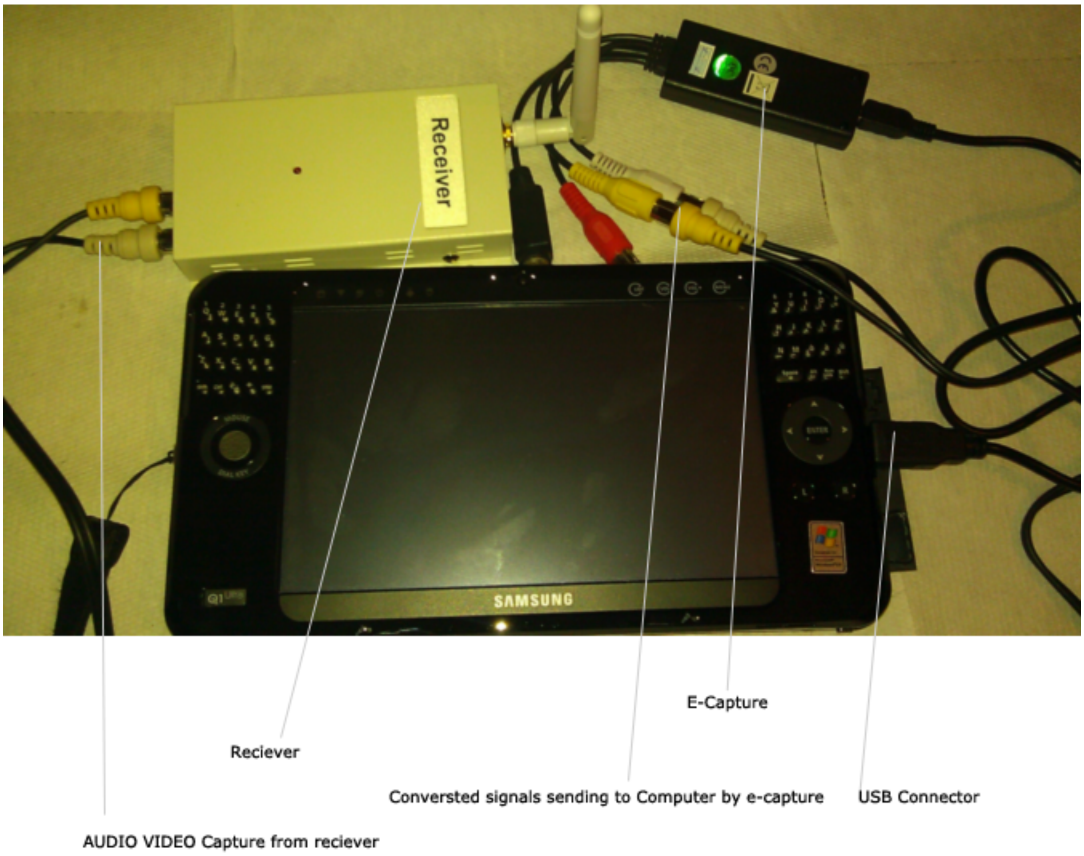


Fig 4.7.1: Assembled Hardware for teacher.



Now we will attach receiver to transmit digital signals with computer device, we are using SAMSUNG tablet device that will receive converted signals by digital audio, video receiver.

Real time Demo screenshots

4.3 Usability Evaluation for POLILIPS

We try our best to make POLILIPS usable and accessible by the users specially client side student module, this was important to get maximum success from this application, we will describe both modules in terms of accessibility and usability approaches.

4.3.1 Usability Test for Student Module:

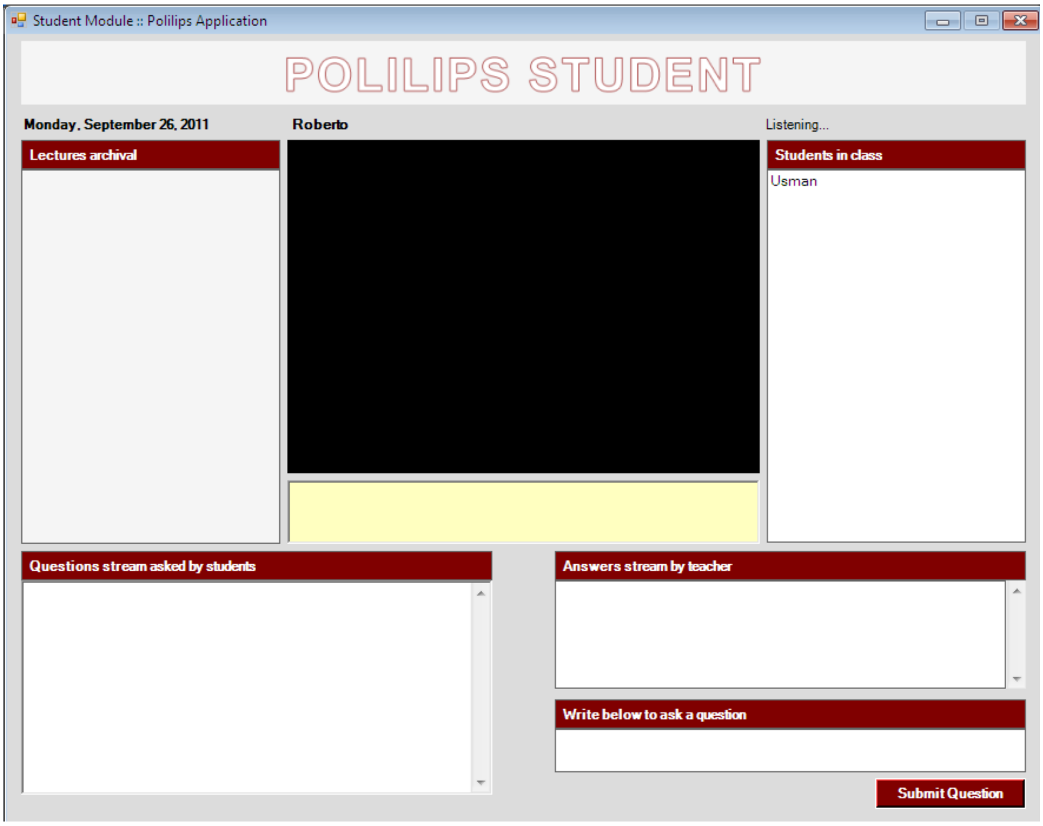


Fig 4.8: POLILIPS STUDENT MODULE.

- **Performance:** The POLILIPS application will not take more than 3 seconds to load onto the Computer Device. Only two steps user can access application contents.
- **Emotional response:** POLILIPS easy to use inter face in student module user is confident to perform action, rather than confuse and not to recall his/ her memory, User is recommending use of this module.

- **Reliability:** POLILIPS Student module is up and can run 18hours/day and 365 days and will be crash safe during 95% of its runtime.
- **Usability:** Simple to Operate, The software is easy to learn and operate; the user will not require special skills or training to operate the POLILIPS users. Simple design, make this application more usable user can access whole functionality with minimum number of clicks; this best to get more positive results from users. We try our best to designed in such a way that the user will find it easy to interact with the POLILIPS student module.
- **Eye Tracking:** Eye tracking of user is acceptable in most of case we got feedback, as henry said I always look any software to find features in bottom of application while its multimedia contents, because this is very general for all such video contents i-e YouTube controls or other video running application, hence in above module we can use all clicking features in bottom area, it was focused to placed here, since user most attention is to read lips and converted subtitles.

4.3.2 Usability Test for Teacher Module:

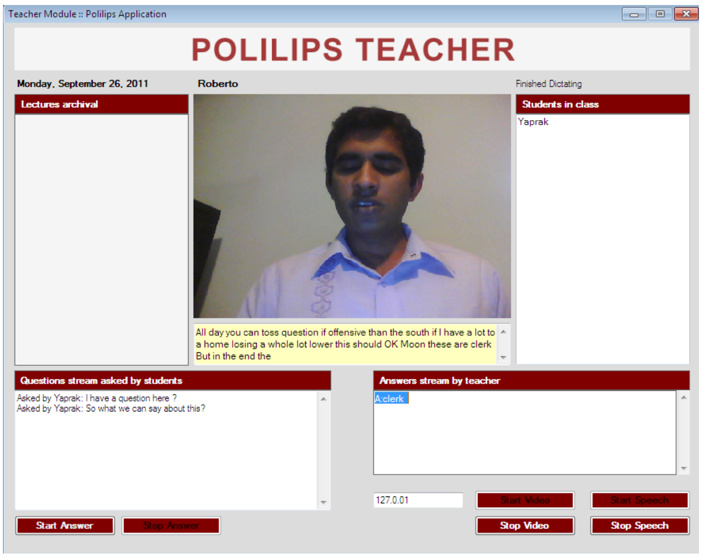


Fig 4.9: Usability test for Teacher Module

- **Performance:** The POLILIPS application will not take more than 3 seconds to load onto the Computer Device. Only two steps requires to user to access main interface to access application contents.
- **Emotional response:** POLILIPS usability in terms of handling wireless equipment for teacher was not fusible feedback, as this application more based in proof of concept so we have to used provided hardware in market search, but with more sensible search in future work, we can improve usability teacher module.
- **Usability:** Simple to Operate, we are using approach how easy to learn and operate; the user will not require special skills or training to operate the POLILIPS users. We are assuming user for this application are professionals and know at least basic IT for using devices, Simple design, make this application more usable user can access whole functionality with minimum number of clicks; this best to get more positive results from users. We try our best to designed in such a way that the user will find it easy to interact with the POLILIPS teacher module.
- **Eye Tracking:** Eye tracking of user is acceptable in most of case we got feedback, as Adriana as teacher said I always look any software to find features in bottom of application while its multimedia contents, because this is very general for all such video contents i-e YouTube controls or other video running application, hence in above module we can use all clicking features in bottom area, it was focused to placed here.

Chapter 5: Future Work

5.1 Future Work

We implemented this application according to achieve our given goals, In future with extended goals following objectives will be really useful to get best benefits from our working application.

- Creating web services of this application on web platform.
- Create a deaf social community from different part of world with extended features to provide education on Learning management system.
- With extended feature for multiple classrooms, with multiple languages.
- Online learning management system, with multiple languages.

We would like to describe more above stated for future work objectives. This application can be implement to extend web services of our application using third party ASR dragon software, for public SDK is available with high cost for developers. This web application can create a big break through in e learning market for disable people, always is most difficult to approach best education system for education, beyond there is no existence work in field of objectives. This web application platform can build international community to provide real-time learning experience from distance learning.

In this era as online real-time web meetings software and websites are available for obtaining business objectives and also much expensive; as well Google, phrasebook.com and Facebook launched online real-time web conference tool publically.

5.2 Knowledge Sharing

We have often mentioned along this document how many advantages would come from a complete and concrete implementation of knowledge sharing among deaf user community. The bases for it have already been thrown, but the work is not complete. Some decisions must be made about what to share, when, how many choices will be left to the user, and how to manage what is shown externally.

5.3 Real World Applications

We would like to describe google “hangout” Application which is already online for public to have real time meetings experience from distance at a time one to many users support. Somehow similar we can create community circles “classes”, socially on POLILIPS learning web management system. With extended features as our application is introduced. List of features can have in future real world application are given below in a table

Extended conceptual model.

5.4 Extended Version Features

Function name	Description
Web Module	This feature can get more focused users from different part of the world.
Social Login	Social Community for particular disable people can have more benefits with the social ID so they can share their knowledge with different communities, in the world.
Unique Courses Classrooms	For each course there can have unique classrooms. And stream will could from different part of the world, In class room there can be workshops for deaf people. And some informational video contents.
Different Language	Dragon is available with different language Support for real World Web Application To manage Cost factor it could have big break through as its best ever Software with good accurate results.
Download Lectures Streams	Lectures for each course class would be downloadable so they can use it later. It can store automatically.
Distance Learning	This could be great feature to extend POLILIPS, with distance learing approach and it can solve only by real world knowledge sharing web platform.

Table 5.1: Features of extended POLILIPS

In other words we looking for future direction for extended version of this application work online social community for deaf and hear impairments people to have full advantages from our basic idea to extend in further future work objectives.

Chapter 6: Conclusions

In this chapter we will conclude about our project. In the beginning of this document (see chapter 1) We claimed that this work had three main goals: to overcome the limits of the POLILIPS, Let us point out now how each of these goals has been achieved throughout this work.

Broadcast video and audio text streaming on network to words client devices, we accomplished this goal to stream video and audio text streaming on network, which we explained in implementation part in detail about success, as our proof of concept we used Windows Speech API, which is free to used and build in in windows system, accuracy is not expected results but with later future work Dragon speech SDK can be embed for better functionality on availably of baring cost of Dragon SDK:

Client devices able to receive streaming from server, this goal we achieved perfectly in our application and accuracy & performance is acceptable by user, but for more user accuracy and performance can be affected.

Client device have to communicate through text streaming with server. This is our (extended goal) to complete role of our project application, we accomplished because deaf or hear impairment user cant communicate in previous goal with teacher for having any questions in his mind, as is discussed in details this document, teacher cant see computer screen to make it more usable and accessible so after submission teacher will hear voice of asked question. So can get intentions somebody asked questions, similarly to complete loop teacher will record answer it will be alerting to student this is answer of a question and it will saved in archival.

In future this application can achieved further advance level goals that will lead application to create break through in industry and great help for disable community.