# English Text to Multilingual Speech Translator Using Android

**Vivek Hanumante, Rubi Debnath, Disha Bhattacharjee, Deepti Tripathi, Sahadev Roy**

***Abstract: This paper aims at providing design and development solution of an Android application whose objective is to provide a solution to overcome the barrier of languages by implementing text to speech conversion in different languages. The Android application developed text to speech conversion to facilitate the translation of English language text into speech output in different languages. We have also proposed few improvements which can further advance this system to include more target audiences so as to make it more beneficial and useful. The proposed English Text to Multilingual Speech Translator using Android (T2MSTA) aims at providing assistance to the people lacking the power of speech or non-native speakers like people who do not share a common language.***

***Index Terms— Android, T2MSTA, Text to Speech conversion, Intent, MATLAB.***

## I. INTRODUCTION

Android is probably the most popular operating system being used by millions of smart phones and tablets today and is growing by leaps and bounds. This is one of the most sophisticated and user friendly platform. Thus we have selected android as a platform to develop a system that provides a solution for the people who are unable to read or speak a particular language either because of illiteracy or not sharing a common language, or for other reasons.

There are many procedures available in the market for text to speech conversion [1][2][3]. Each language uses different sets of algorithms and these cannot be employed to translate languages. Some multilingual translators are also available but they are limited for text to text [4][5][6][7]. Speech to speech translation is also available [8][9].

The proposed method is a simple and handy solution to overcome the barrier of languages by developing an android application for multilingual translation implementing text to speech conversion. It provides the methodology to implement text to speech conversion to facilitate the translation of English language text into speech output in different languages as desired. Design and development of this T2MSTA will thus provide assistance to the people those lack the power of speech or non-native speakers.

With the design and development of this smart multilingual translator we strive for developing a system which will in turn embellish the devices which assists aforementioned people. Thus a novel system T2MSTA has been proposed which can take input from user and can translate in various languages. T2MSTA so far can take an input of a single word or a sentence in English language and can translate as output voice in three different languages namely Hindi, English and Bengali as per users' choice. The system can be further upgraded to generate syllable based sounds for the words which are not present in library stored [10] [11].

## II. OVERVIEW OF T2MSTA

The proposed T2MSTA is centralized on an objective which is a twofold: i) Text to Speech conversion and ii) Language Translation. Sound libraries of different languages for T2MSTA has been created using MATLAB, which includes most frequently used words in daily life conversations. In the android application user is asked to enter the text and click on the button named with language to listen the translated speech output. Figure 1 presents the system architecture of T2MSTA:

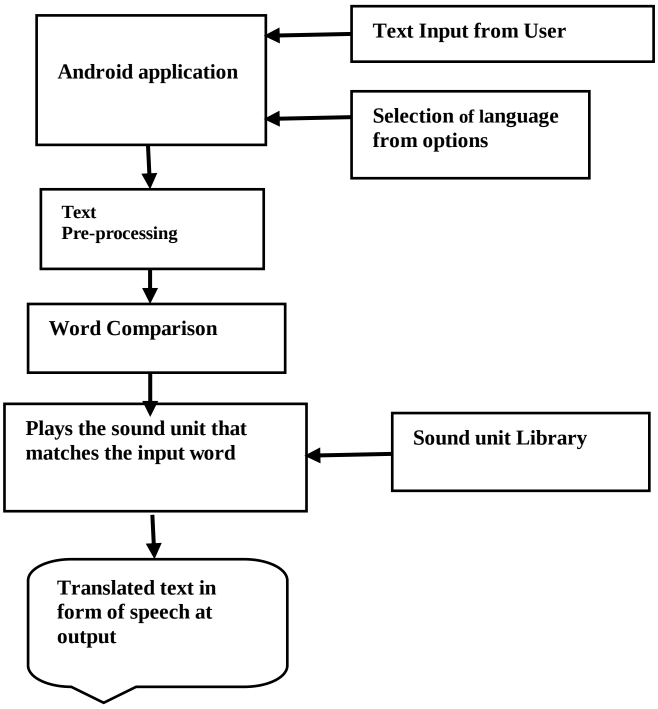

Android application
Text Input from User
Selection of language from options
Text Pre-processing
Word Comparison
Plays the sound unit that matches the input word
Sound unit Library
Translated text in form of speech at output

Fig. 1. **System Architecture**

## III. DEVELOPMENT AND OPTIMIZATION OF SOUND LIBRARY

Libraries for T2MSTA have been developed using MATLAB. Firstly the sound units are

**Manuscript received on April, 2014.**

**Vivek Hanumante**, Department of Electronics & Communication Engineering, NIT Arunachal Pradesh, Yupia, India.
**Rubi Debnath**, Department of Electronics & Communication Engineering, NIT Arunachal Pradesh, Yupia, India.
**Disha Bhattacharjee**, Department of Electronics & Communication Engineering, NIT Arunachal Pradesh, Yupia, India.
**Deepti Tripathi**, Department of Electronics & Communication Engineering, NIT Arunachal Pradesh, Yupia, India.
**Sahadev Roy**, Department of Electronics & Communication Engineering, NIT Arunachal Pradesh, Yupia, India.

stored in .wav format, which then, is converted into .ogg format after certain optimization which are discussed in detail in this section.

The whole sound library comprise of different sub-libraries for different languages. Each of these sub-libraries include word units of that language stored using a MATLAB function *wavrecord* [12], the words are stored in digitized format in the form of matrix. Each library is a collection of English words which stores the sound of corresponding translation of the word pertaining to that language. For example the syntax for storing a word Y using MATLAB is given below:

Y = WAVRECORD(N,FS,'DTYPE');

Where Y is the variable name in which one wants to store voice, having sample frequency FS in Hz, N is sample rate and 'DTYPE' is data type specified. Available data types are 'double', 'single', 'int16' and 'uint8'.

The duration of 2 second has been provided to facilitate a comfortable zone of recording voice for speaker, but one word takes an average of approximately 0.800 second for pronunciation. Providing a longer duration for recording helps in optimizing the sound unit in terms of suitable duration for playing at output and reducing noise. The recorded sound unit in Matlab can be converted into '.wav' format file in order to export. This exported file can be edited to remove the time lapse using a suitable sound editor software. Further this edited file is stored in .ogg format. The conversion from .wav to .ogg after time lapse optimization yields into a file which has a memory of approximately 35% of the original .wav file. For example a file of 45Kb in .wav becomes approximately 15Kb after this optimization. This in turns results into an effective memory minimization.

One sample speech waveform of word “Shaam” (Hindi translation of word “Evening”) is given in Figure 2 which is a raw sound unit containing time lapse and noise.

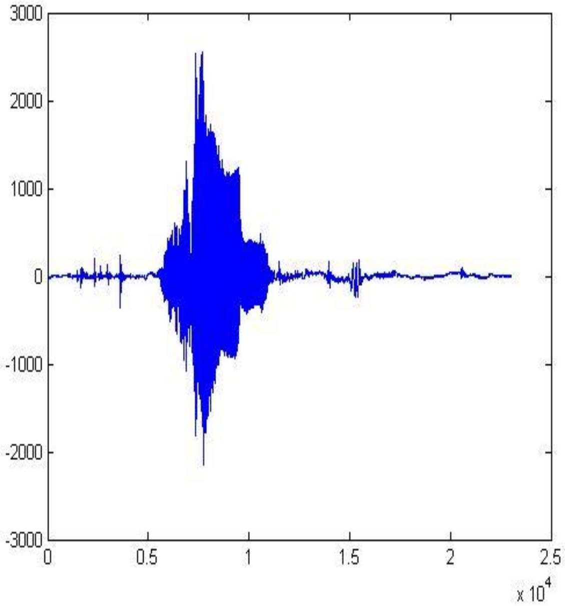


Fig. 2. Speech waveform of Word 'Shaam' (Hindi translation of word “Evening”) having noise and time lapse before optimization.

Speech waveform of Word 'Shaam' (Hindi translation of word “Evening”) having noise and time lapse before optimization.

Figure 3 presents the speech waveform of optimized sound unit after trimming time lapse and noise. While editing the sound unit a Guard time of approximately 20 millisecond is provided before and after the portion containing sound data to provide a suitable pause for pronunciation of words in a sentence.

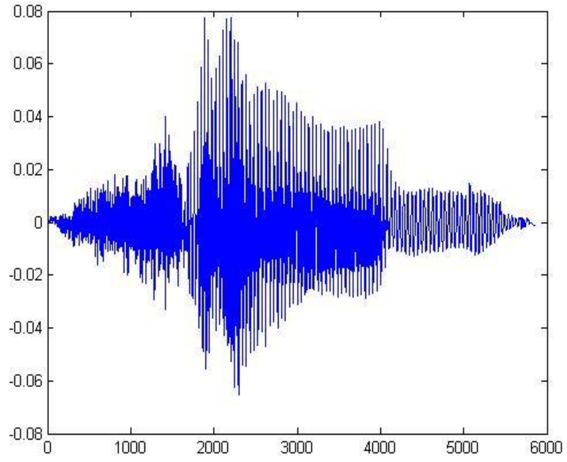


Fig. 3. Speech waveform of Word 'Shaam' after optimization

Speech waveform of Word 'Shaam' after optimization with Guard time

Table I presents the speech waveforms of word Question and its translations in Hindi and Bengali stored in male and female voices. Both the waveforms before and after optimization have been shown for comparison.

## IV. ALGORITHM

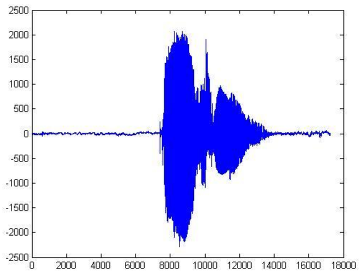
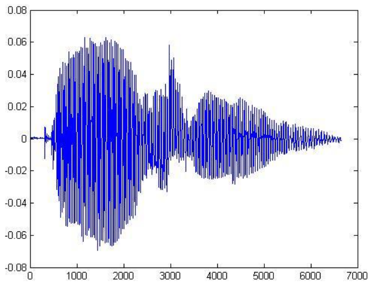
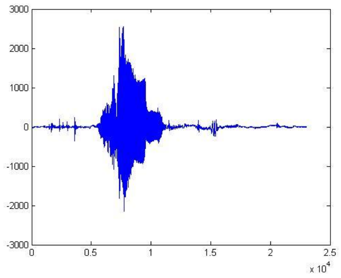
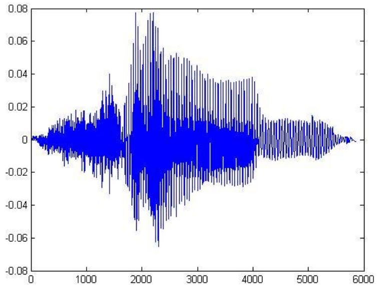
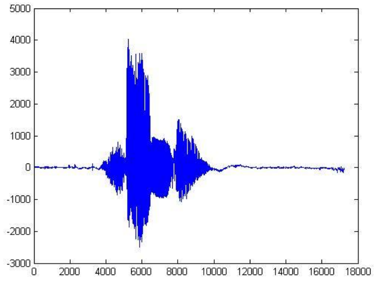
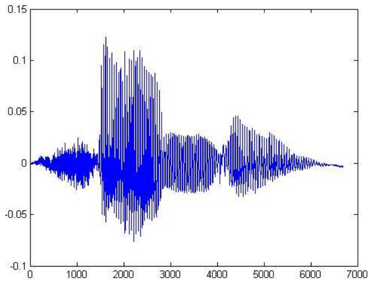
This section presents an algorithm for the working of developed Android application discusses the process of taking input and producing sound output after processing the input text. User is asked to enter the text in English. Entered text may be a single word or a sentence. User has the freedom to enter text in upper, lower or mixed cases. Following is the algorithm to take input and process text.

```
1:  Take user input in EditText field.
2:  Convert Text to string
3:  On click of Hindi Button:
    Start Hindi_Activity for translation.
4:  On click of English Button:
    Start English_Activity for translation.
5:  On click of Bengali Button:
    Start Bengali_Activity for translation
6:  Get string from intent and store in STR.
7:  Split the sentence to words and store in array WordArray.
8:  Find the length of WordArray in ArrLength.
9:  Initialize X=0
10: For X=0
11: Compare string from array.
12: If: word found
13: Get PlayTime duration.
14: Start MediaPlayer MP.
15: Sleep (PlayTime) //Delay for duration of PlayTime for MP.
16: Else: Display-Sorry!! “word” is not present in our library
17: Increment X
```

18: Repeat steps 11 to 17 until X==*ArrLength*.

**TABLE I. Speech Waveforms For Word “Question” And Its Translatio**

string say STR from the Intent.

| Sr No. | Language | Word Translation | Raw Speech Waveform | Optimized speech waveform |
|---|---|---|---|---|
| 1. | English | “Evening” |  |  |
| 2. | Hindi | “Shaam” |  |  |
| 3. | Bengali | “Sondhya” |  |  |

## V.WORKING OF T2MSTA

This android application, apart from having a main Activity (LAUNCHER), contains three DEFAULT activities which are executed based on the language selection of user by clicking on the corresponding button. Main activity asks user for entering the text which may be a word or a sentence without including any special character. After entering the text user can click on any button containing the name of languages to hear the translation of entered text in that particular language. Text entered by user is wrapped up in Intent and is sent to the next activity which starts on click of a button.

We have three DEFAULT activities for language Hindi, English and Bengali. These activities start with the click of a button. The text sent from the main activity is received in a

Then this received sentence is split into the single units of words and these words are stored in an array. These stored words are compared one by one with the strings of words stored in our library. If a match is found the corresponding word is played from library, otherwise a message is displayed for unavailability of word in library.

For example if a user enters “Hello World”, the two words “Hello” and “World” will be stored in array. Both the words will be separately compared with the words stored in library. If a match is found then the system will play the stored voice in language chosen by user, otherwise a message “Sorry!! “Word” is not present in our library now”. Following is the segment of program displaying the comparison of words and playing the corresponding sound unit.

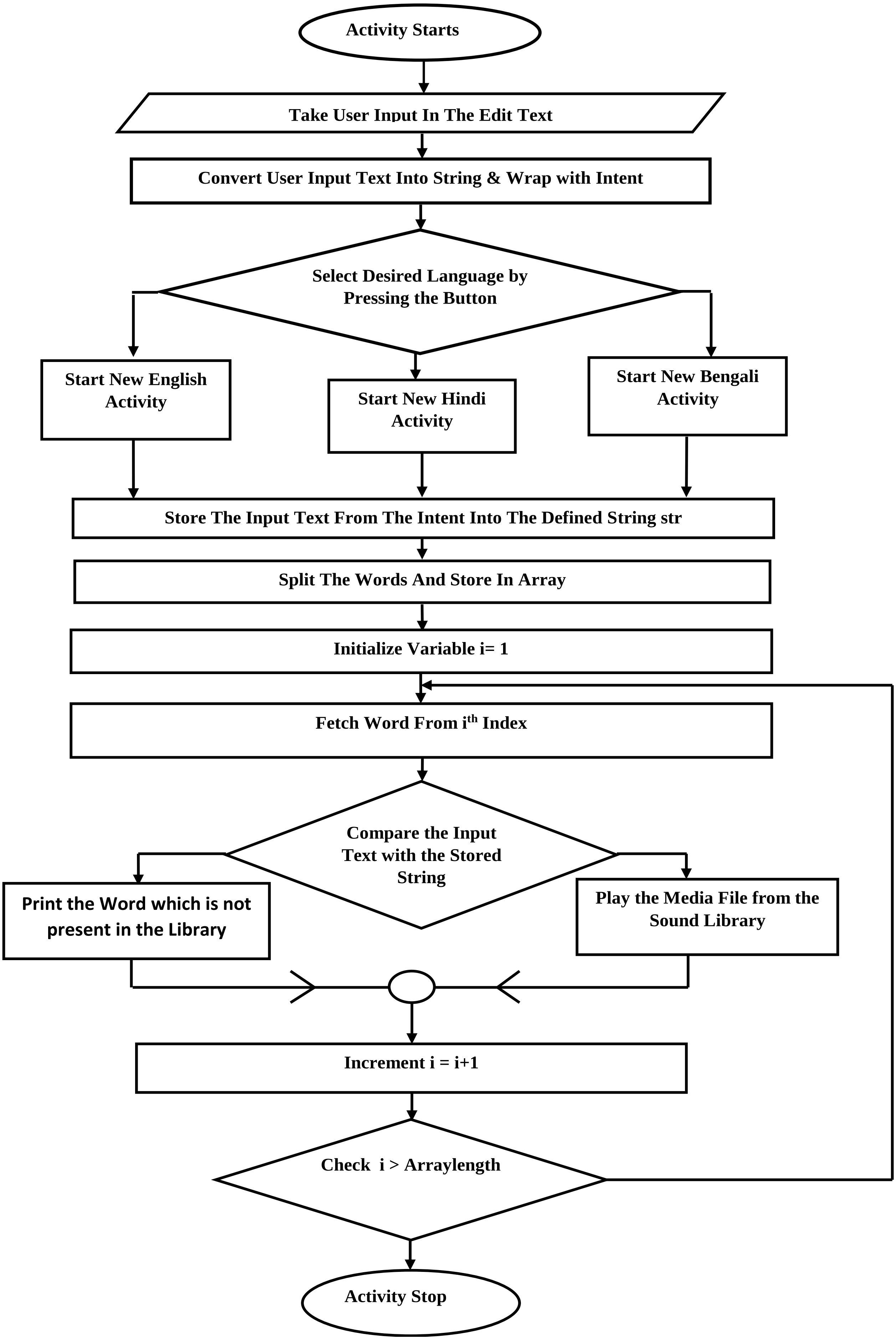
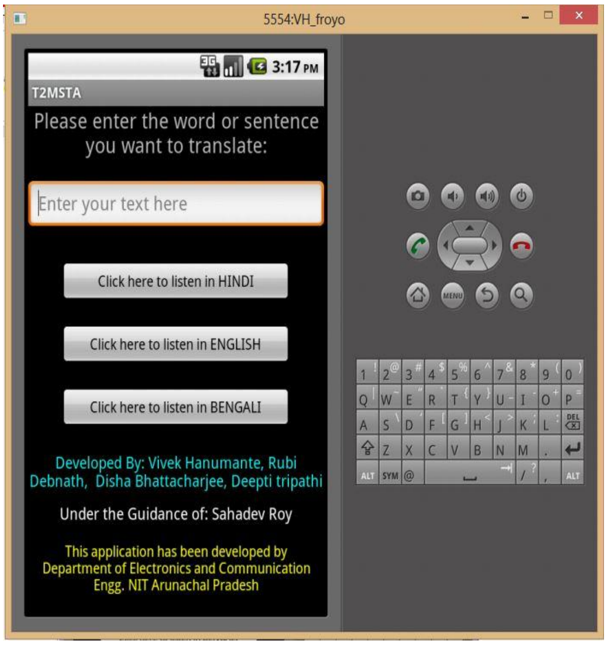

Activity Starts
Take User Input In The Edit Text
Convert User Input Text Into String & Wrap with Intent
Select Desired Language by Pressing the Button
Start New English Activity
Start New Hindi Activity
Start New Bengali Activity
Store The Input Text From The Intent Into The Defined String str
Split The Words And Store In Array
Initialize Variable i= 1
Fetch Word From ith Index
Compare the Input Text with the Stored String
Print the Word which is not present in the Library
Play the Media File from the Sound Library
Increment i = i+1
Check i > Arraylength
Activity Stop

Fig. 4. Flowchart of T2MSTA

```
str = getIntent().getExtras().getString("Text");
String [] mystr= str.split("\\s");
for (int x=0; x<mystr.length; x++)
if(mystr[x].equalsIgnoreCase("afternoon"))
{
mp = MediaPlayer.create(this, R.raw.afternoon);
playtime = mp.getDuration();
mp.start();
try{
Thread.sleep(playtime + 100); }
catch(InterruptedException e){
   e.printStackTrace();}
 }
 else if (mystr[x].equalsIgnoreCase("evening"))
{
   mp = MediaPlayer.create(this, R.raw.evening);
   playtime = mp.getDuration();
   mp.start();
   try{
   Thread.sleep(playtime + 100);
   } catch(InterruptedException e)
 {
   e.printStackTrace();}
   }
else
{
DispH.setText("Sorry!! Word "+mystr[x]+ " is not present in our Library");
}
```

Fig. 5. Program template for steps 3 to 12 in algorithm

Figure 6 displays the main layout where user can enter the text and select the desired language in which user wishes to listen the translation of text. When user clicks on a button out of the provided options the application takes user to another activity as shown in Fig. 7, where by clicking on the provided button user can listen to the translation of text in the form of speech.


5554:VH_froyo

3:17 PM

T2MSTA

Please enter the word or sentence you want to translate:

Enter your text here

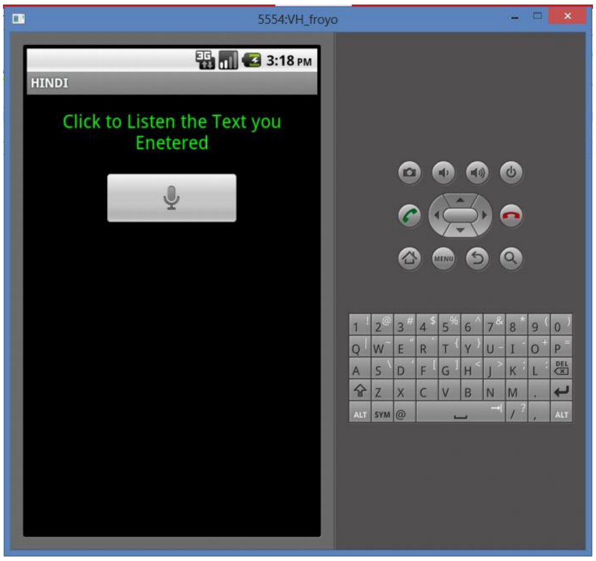
Developed By: Vivek Hanumante, Rubi Debnath, Disha Bhattacharjee, Deepti tripathi

Under the Guidance of: Sahadev Roy

This application has been developed by Department of Electronics and Communication Engg. NIT Arunachal Pradesh

Click here to listen in HINDI
Click here to listen in ENGLISH
Click here to listen in BENGALI
MENU
ALT
SYM
DEL

**Fig. 6. Main layout to enter text and select language.**


5554:VH_froyo

3:18 PM

HINDI

Click to Listen the Text you Enetered

**Fig. 7. New Activity to play the sound**

## VI. SCOPE FOR FUTURE IMPROVEMENTS

The system for multilingual translator proposed by us is an elementary model which incorporates almost all the basic requirements. However to make this system more precise and useful for a wide range of target audience, it demands some further improvements. We are continuously striving to make it frontier in its field. Further we are aiming at following improvements:

1. To take input text from an image of printed English text by implementing character recognition. Presently we are only able to take manual input through virtual keyboard.
2. To implement grammar rules or concepts for more accurate translation in desired languages.
3. Implementation of generation of sound based on syllables for unrecognized words in library like name of a person or place etc. which may not be present in library.
4. The system can be further extended to include more languages and possibly dialects.
5. Generation of Natural sounding voice.

## VII. CONCLUSION

We have thus proposed a novel system for taking input in English language and translating the text to provide a speech output in different languages. The presently developed system provides an option of translating English text in three languages English, Hindi & Bengali which are among the most spoken languages in India. The proposed system with necessary improvements discussed holds the commercial viability. This may lead to a great help to the people lacking the power of speech, dyslexic people and non-native speakers.

## REFERENCES

1. El-Imam, Y.A.; Banat, K., "Text-to-speech conversion on a personal computer," Micro, IEEE , vol.10, no.4, pp.62,74, Aug. 1990
2. O'Malley, M.H., "Text-to-speech conversion technology," Computer, vol.23, no.8, pp.17-23, Aug. 1990.

3. Santos, J.; Nombela, J., "Text-to-speech conversion in Spanish a complete rule-based synthesis system," Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '82. , vol.7, no., pp.1593-1596, May 1982.
4. Sadeque, F.Y.; Yasar, S.; Islam, M.M., "Bangla text to speech conversion: A syllabic unit selection approach," International Conference on Informatics, Electronics & Vision (ICIEV), 2013, pp.1-6, 17-18 May 2013.
5. Bothe, K.; Joachim, S., "Interactive tool-based production of multilingual teaching and learning materials," Fifth IEEE International Conference on Advanced Learning Technologies, ICALT, pp.516,518, 5-8 July 2005.
6. Nowak, G.; Grabowski, S.; Draus, C.; Zarebski, D.; Bieniecki, W., "Designing a computer-assisted translation system for multi-lingual catalogue and advertising brochure translations," Proceedings of VIth International Conference on Perspective Technologies and Methods in MEMS Design (MEMSTECH), pp.175,180, 20-23 April 2010.
7. Linsi Xia; Yamashita, N.; Ishida, Toru, "Analysis on Multilingual Discussion for Wikipedia Translation," Second International Conference on Culture and Computing (Culture Computing), pp.104, 109, 20-22 Oct. 2011.
8. Yuqing Gao; Bowen Zhou; Liang Gu; Sarikaya, R.; Hong-Kwang Kuo; Rosti, A.-V.I.; Afify, M.; Weiihong Zhu, "IBM Mastor: Multilingual Automatic Speech-To-Speech Translator," ICASSP 2006 Proceedings. 2006 IEEE International Conference on Acoustics, Speech and Signal Processing, 2006. , vol.5, 14-19 May 2006.
9. Liang Gu; Yuqing Gao; Liu, F.-H.; Picheny, Michael, "Concept-based speech-to-speech translation using maximum entropy models for statistical natural concept generation," Processing, IEEE Transactions on Audio, Speech, and Language , vol.14, no.2, pp.377-392, March 2006.
10. Evandro Franzen1 and Dante Augusto Couto Barone, “Automatic Discovery of Brazilian Portuguese Letter to Phoneme Conversion Rules through Genetic Programming”, Springer-Verlag Berlin Heidelberg, pp. 62–65, 2003.
11. Hertz, S.R., "English text to speech conversion with delta," Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '86. , vol.11, pp.2427, 2430, Apr 1986.
12. MATLAB and Simulink Release 2013a, The MathWorks, Inc., Natick, Massachusetts, United States, 2013.