**NATIONAL COLLEGE OF ENGINEERING**

**(Affiliated to Tribhuvan University)**

**Talchhikhel, Lalitpur**



**[Subject Code: CT 654]**

**A MINOR PROJECT MIDTERM REPORT ON**

**“Text-to-Speech Synthesis System”**

**Submitted by**

**Bimal Adhikari [074/BCT/007]**

**Bishnu Khojwar [074/BCT/011]**

**Bishworam Ghimire [074/BCT/012]**

**Samrat Subedi [074/BCT/034]**

**Submitted to**

**Department of Computer and Electronics Engineering**

**Falgun, 2077 B.S**

**Text-to-Speech Synthesis System**

**Subject Code: CT 654**

**Submitted by**

**Bimal Adhikari [074/BCT/007]**

**Bishnu Khojwar [074/BCT/011]**

**Bishworam Ghimire [074/BCT/012]**

**Samrat Subedi [074/BCT/034]**

**A minor project report submitted in fulfillment of the requirement**

**for the**

**Department of Computer and Electronics Engineering**

**Department of Computer and Electronics Engineering**

**National College of Engineering**

**Talchhikhel, Lalitpur**

**Nepal**

**Falgun, 2077 B.S.**

# COPYRIGHT

The author has agreed that the Library, Department of Electronics and Computer Engineering, National college of Engineering and Institute of Engineering may make this report freely available for inspection. Moreover, the author has agreed that for extensive copying of this project report for scholarly purpose may be granted by the supervisors who supervised the project work record here in or, in their absence, by the Head of Department where in the project report was done. It is understood that the recognition will be given to the author of this report and to the Department of Electronics and Computer Engineering, National college of engineering, in any use of the material of this project report, copying or publication or the other use of this report for financial gain without approval of the Department of Electronics and Computer Engineering, National College of Engineering and author’s written permission is prohibited.

Request for the permission to copy or to make any other use of the material in this report in whole or in part should be addresses to:

Head of Department

Department of Electronics and Computer Engineering

National College of Engineering

Talchhikhel, Lalitpur

# ACKNOWLEDGEMENT

Foremost, we would like to express our sincere gratitude to our supervisor **Er. Subash Pandey** for the continuous support of our project study, research, and accomplishment of project, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped us in all the time of project. We could not have imagined having a better supervisor and mentor as **Er. Subash Pandey** for our project.

Besides our supervisor, we would like to thank Head of the Department **Er. Pradip Adhikari** and the teachers: **Er. Mohan Maharjan, Er. Sharmila Bista, Er. Anup Shrestha and Er. Rebanta Aryal** for their encouragement, insightful comments, and hard questions.

We thank our senior students of National College of Engineering for the stimulating discussions, and for all the fun we have had in the project period.

Any attempt at any level can 't be satisfactorily completed without the support and guidance of our parents and friends. Last but not the least, we would also express our deepest gratitude to NCE administration, which provided us with essential and all kind of logistics support and encouragement throughout the project study.

# ABSTRACT

Listening to the texts has become more popular than reading it thoroughly. Some people having some disorder of reading has become problematic. In this busy schedule of every person, listening to the text while doing other task can help a lot. The proposed system is an application that converts text into spoken word, by analyzing and processing the text to convert this processed text into synthesized speech representation of the text. Here, we developed a useful text-to-speech synthesizer in the form of a simple application that converts inputted text or files with different extensions uploaded, into synthesized speech and reads out to the user. The development of a text to speech synthesizer will be of great help to people with visual impairment and make through large volume of text easier to listen. It can also be helpful for those with language barriers.

*Keywords: text, text-to-speech synthesizer, files, extensions*

# TABLE OF CONTENTS

[COPYRIGHT iii](#_Toc64732203)

[ACKNOWLEDGEMENT iv](#_Toc64732204)

[ABSTRACT v](#_Toc64732205)

[TABLE OF CONTENTS vi](#_Toc64732206)

[LIST OF TABLES viii](#_Toc64732207)

[LIST OF FIGURES ix](#_Toc64732208)

[LIST OF ABBREVIATIONS x](#_Toc64732209)

[1. INTRODUCTION 1](#_Toc64732210)

[1.1 Background 1](#_Toc64732211)

[1.2 Problem Statement 8](#_Toc64732212)

[1.3 Aim 9](#_Toc64732213)

[1.4 Objectives 9](#_Toc64732214)

[1.5 Scope 10](#_Toc64732215)

[1.6 Application 11](#_Toc64732216)

[2. LITERATURE REVIEW 12](#_Toc64732217)

[3. REQUIREMENT SPECIFICATION 20](#_Toc64732218)

[3.1 Overall Description 20](#_Toc64732219)

[3.1.1 Product Perspective 20](#_Toc64732220)

[3.1.2 Product Function 20](#_Toc64732221)

[3.1.3 Operating Environment 20](#_Toc64732222)

[3.2External Interface Requirements 20](#_Toc64732223)

[3.2.1 Hardware Interface 20](#_Toc64732224)

[3.2.2 Software Interface 20](#_Toc64732225)

[3.3 System Features 21](#_Toc64732226)

[3.3.1 Direct Text Entry 21](#_Toc64732227)

[3.3.2 Voice Gender Selection 21](#_Toc64732228)

[3.3.3 File upload 22](#_Toc64732229)

[3.3.4 Speech Generation of Direct text input 22](#_Toc64732230)

[3.3.5 Speech Generation of Uploaded file 23](#_Toc64732231)

[3.4Other Non-Functional Requirements 23](#_Toc64732232)

[3.4.1 Performance Requirements 23](#_Toc64732233)

[3.4.2 Safety Requirements 24](#_Toc64732234)

[3.4.3 Software Quality Attributes 24](#_Toc64732235)

[4. SYSTEM ANALYSIS 25](#_Toc64732236)

[a. Activity Diagram of main process 25](#_Toc64732237)

[b. Domain Model 27](#_Toc64732238)

[c. Sequential Model 28](#_Toc64732239)

[5. SYSTEM DESIGN 29](#_Toc64732240)

[a. UI design 29](#_Toc64732241)

[b. Class Diagram 30](#_Toc64732242)

[6. IMPLEMENTATION 31](#_Toc64732243)

[a. Main Algorithm 31](#_Toc64732244)

[b. Main class with functions 32](#_Toc64732245)

[7. UNIT TESTING 33](#_Toc64732246)

[8. SYSTEM TESTING 34](#_Toc64732247)

[9. DISCUSSION 35](#_Toc64732248)

[10. CONCLUSION 36](#_Toc64732249)

[11. REFERENCES 37](#_Toc64732250)

# LIST OF TABLES

[Table 1: List of Consonants 2](#_Toc64364836)

[Table 2:List of Vowels 4](#_Toc64364837)

[Table 3 Comparison of existing systems 6](#_Toc64364838)

# LIST OF FIGURES

[Figure 1 Activity Diagram of main process 25](#_Toc64732251)

[Figure 2 Domain Model 27](#_Toc64732252)

[Figure 3 Sequential Model of System 28](#_Toc64732253)

[Figure 4 Class Diagram of main process 30](#_Toc64732254)

# LIST OF ABBREVIATIONS

API – Application Programming Interface

DNN – Deep Neural Network

DSP – Digital Signal Processing

GUI – Graphical User Interface

HMM – Hidden Markov Model

IEEE - Institute of Electrical and Electronics Engineers

IEICE - Institute of Electronics, Information and Communication Engineers

IPA- International Phonetic Alphabet

LPC – Linear Predictive Coding

MATLAB - Matrix Laboratory

MLLR – Maximum Likelihood Linear Regression

MOS – Mean Opinion Score

MS – Modulation Spectrum

NAIST-Nara Institute of Science and Technology

NLP -Natural Language Processing

SWT – Standard Widget Toolkit

TTS – Text to Speech

VTTS – Visual Text to Speech

WWW – World Wide Web

# 1. INTRODUCTION

## 1.1 Background

Text-to-speech synthesis -TTS is the automatic conversion of a text into speech that resembles, as closely as possible, a native speaker of the language reading that text. The text-to-speech synthesizer (TTS) is the technology that lets the computer speak to you. The TTS system gets the text as the input and then a computer algorithm that calls the TTS engine, analyses the text, pre-processes the text, and synthesizes the speech with some mathematical models. The TTS engine usually generates sound data in an audio format as the output. [1]

Speech synthesis is a field of computer science that deals with designing computer systems that synthesize written text. It is a technology that allows a computer to convert a written text into speech via a speaker. Automatic speech synthesis is one of the fastest developing fields in the framework of speech science and engineering. The basic idea of text-to-speech (TTS) technology is to convert written input to spoken output by generating synthetic speech. Speech synthesis can be described as the artificial production of human speech. A computer system used for this purpose is called a speech synthesizer.

The text-to-speech (TTS) synthesis procedure consists of three main phases:

The first is either direct input to the user or processing of the file. The file of different extensions is uploaded with the validation of required file extension. The text from those files is firstly extracted. In case of pdf files, the number of pages in the file are extracted and the text from all the pages are stored in a string and sent for further processes. In case of docx file, the number of paragraphs is extracted and text from all the paragraph are stored in string and sent for further processing. In case of txt files, normal file read operation can extract the text from the file which are then sent for further processing.

For analyzing and processing the text, we are using Natural Language Processing (NLP). The NLP helps in removal of white spaces, divide into sentences and further dividing into tokens, removal of unwanted symbols, acronym and abbreviation handling.

Finally, using Pyttsx3 module, this processed text is converted into a synthesized speech representation of the text.

Despite there being just 26 letters in the English language there are approximately 44 unique sounds, also known as phonemes. The 44 sounds help distinguish one word or meaning from another. Various letters and letter combinations are known as graphemes are used to represent the sounds.

The 44 English sounds fall into two categories: consonants and vowels. Also, they are distinguished into two types on the basis of vibration they cause voice and voiceless sound. There may be discovered lists with more or less than these 44 sounds.

Below is a list of the 44 phonemes along with their International Phonetic Alphabet symbols.

**Consonants:**

Table 1: List of Consonants

|  |  |  |  |
| --- | --- | --- | --- |
| Phoneme | IPA Symbol | Graphemes | Examples |
| 1 | b | b, bb | bug, bubble |
| 2 | d | d, dd, ed | dad, add, milled |
| 3 | f | f, ff, ph, gh, lf, ft | fat, cliff, phone, enough, half, often |
| 4 | g | g, gg, gh,gu,gue | gun, egg, ghost, guest, prologue |
| 5 | h | h, wh | hop, who |
| 6 | dʒ | j, ge, g, dge, di, gg | jam, wage, giraffe, edge, soldier, exaggerate |
| 7 | k | k, c, ch, cc, lk, qu, q(u), ck, x | kit, cat, chris, accent, folk, bouquet, queen, rack, box |
| 8 | l | l, ll | live, well |
| 9 | m | m, mm, mb, mn, lm | man, summer, comb, column, palm |
| 10 | n | n, nn,kn, gn, pn | net, funny, know, gnat, pneumonic |
| 11 | p | p, pp | pin, dippy |
| 12 | r | r, rr, wr, rh | run, carrot, wrench, rhyme |
| 13 | s | s, ss, c, sc, ps, st, ce, se | sit, less, circle, scene, psycho, listen, pace, course |
| 14 | t | t, tt, th, ed | tip, matter, thomas, ripped |
| 15 | v | v, f, ph, ve | vine, of, stephen, five |
| 16 | w | w, wh, u, o | wit, why, quick, choir |
| 17 | z | z, zz, s, ss, x, ze, se | zed, buzz, his, scissors, xylophone, craze |
| 18 | ʒ | s, si, z | treasure, division, azure |
| 19 | tʃ | ch, tch, tu, ti, te | chip, watch, future, action, righteous |
| 20 | ʃ | sh, ce, s, ci, si, ch, sci, ti | sham, ocean, sure, special, pension, machine, conscience, station |
| 21 | θ | th | Thongs |
| 22 | ð | th | Leather |
| 23 | ŋ | ng, n, ngue | ring, pink, tongue |
| 24 | j | y, i, j | you, onion, hallelujah |

**Vowels:**

Table 2:List of Vowels

|  |  |  |  |
| --- | --- | --- | --- |
| Phoneme | IPA Symbol | Graphemes | Examples |
| 25 | æ | a, ai, au | cat, plaid, laugh |
| 26 | eɪ | a, ai, eigh, aigh, ay, er, et, ei, au, a\_e, ea, ey | bay, maid, weigh, straight, pay, foyer, filet, eight, gauge, mate, break, they |
| 27 | e | e, ea, u, ie, ai, a, eo, ei, ae | end, bread, bury, friend, said, many, leopard, heifer, aesthetic |
| 28 | i: | e, ee, ea, y, ey, oe, ie, i, ei, eo, ay | be, bee, meat, lady, key, phoenix, grief, ski, deceive, people, quay |
| 29 | ɪ | i, e, o, u, ui, y, ie | it, england, women, busy, guild, gym, sieve |
| 30 | aɪ | i, y, igh, ie, uy, ye, ai, is, eigh, i\_e | spider, sky, night, pie, guy, stye, aisle, island, height, kite |
| 31 | ɒ | a, ho, au, aw, ough | swan, honest, maul, slaw, fought |
| 32 | oʊ | o, oa, o\_e, oe, ow, ough, eau, oo, ew | open, moat, bone, toe, sow, dough, beau, brooch, sew |
| 33 | ʊ | o, oo, u,ou | wolf, look, bush, would |
| 34 | ʌ | u, o, oo, ou | lug, monkey, blood, double |
| 35 | u: | o, oo, ew, ue, u\_e, oe, ough, ui, oew, ou | who, loon, dew, blue, flute, shoe, through, fruit, manoeuvre, group |
| 36 | ɔɪ | oi, oy, uoy | join, boy, buoy |
| 37 | aʊ | ow, ou, ough | now, shout, bough |
| 38 | ə | a, er, i, ar, our, ur | about, ladder, pencil, dollar, honour, augur |
| 39 | eəʳ | air, are, ear, ere, eir, ayer | chair, dare, pear, where, their, prayer |
| 40 | ɑ: | a | Arm |
| 41 | ɜ:ʳ | ir, er, ur, ear, or, our, yr | bird, term, burn, pearl, word, journey, myrtle |
| 42 | ɔ: | aw, a, or, oor, ore, oar, our, augh, ar, ough, au | paw, ball, fork, poor, fore, board, four, taught, war, bought, sauce |
| 43 | ɪəʳ | ear, eer, ere, ier | ear, steer, here, tier |
| 44 | ʊəʳ | ure, our | cure, tourist |

The most important qualities of modern speech synthesis systems are its naturalness and intelligibility. By naturalness, we mean how closely the synthesized speech resembles real human speech. Intelligibility, on the other hand, describes the ease with which the speech is understood. The maximization of these two criteria is the main development goal in the TTS field.

Text to speech software can be enormously helpful for anyone who's visually impaired or has a condition like dyslexia that makes reading on screens tricky. It can also help overcome language barriers for people who are in the process of learning. Text to speech software is also ideal if we want to listen to a document while doing something else, or if we want to sense-check something we've written. It is also a valuable computational aid for those with speech disorders.

There are many existing systems available. Some of the existing system with the strength and weakness are:

Table 3 Comparison of existing systems

|  |  |  |  |
| --- | --- | --- | --- |
| S. N | Systems | Strength | Weakness |
| 1 | Acapela | Covers variety of language  Good voice quality  Flexibility of voice control | Chinese language not available |
| 2. | NeoSpeech | Very high-quality natural voice  Ease to use  Chinese, Italian, French, English, etc. accepted | Only available for few languages. |
| 3 | Loquendo | Covers most languages  Great support resources | User interface not friendly  Unstable server |
| 4 | Panopreter basic | Quick and simple to use  Export sound in WAV and MP3 formats | For windows only |
| 5 | Wordtalk | Integrates with MS Word | Not all functions are clear |

Between these systems following features are common:

* Multiple languages available
* User lexicon
* Good voice quality
* Ease of user interface on some systems
* Mostly available for windows.

## 1.2 Problem Statement

In Text-To-Speech-Synthesis systems, there are several problems in text preprocessing, such as numerals, abbreviations, and acronyms. Correct prosody and pronunciation analysis from the written text is also a major problem today. Also, the unclarities of the spoken word is another problem. The other problem is the unattractive and not user-friendly GUI.

For the user who is in the process of learning the correct pronunciations of the word, this problem is severe. It doesn’t help those who want to overcome language barriers.

For those with less technical skills face difficulty in using the software. Due to the difficulty of using the GUIs for the user, the user faces a lot of problems. Also, the unattractiveness of the GUI doesn’t encourage a user to use the system.

Sometimes you may feel tired to read an e-book and you just want to liberate your eyes for a bit but on other hand you don’t want to give up your books. This has been problem for many readers.

It is expected that the new system will reduce and improve most of the problems encountered in the old system.

## 1.3 Aim

* The aim of this project is to convert the text into speech.

## 1.4 Objectives

* To make familiar with correct pronunciation and prosody.
* To make ease on listening e-book.
* To provide verbal communication for individual who are unable to speak.
* To help students work around their reading difficulties and access the classroom materials.

## 1.5 Scope

The scope of this project is to facilitate people for correct prosody and pronunciation. For correct pronunciation and exact prosody for the word, this system can be helpful. Those individuals learning the correct pronunciation, this system can be helpful. Also, the synthesized speech is of good quality. The English language which it delivers is of high quality. So, the sound is clear and realistic. This system overcomes the problem of language barrier.

This system consists of simple user interface. So, the person with less technical skill can easily use the system. The attractive and simple user interface makes easier for the user to use the system. Due to simple and easy user interface, this encourages the user to use the system.

For those users who are tired of reading the e-books and are on long exposure to laptop or desktop and on the other hand don’t want to give up the books, this system is helpful. User can enjoy the eBooks and continue doing other jobs or rest using this system.

## 1.6 Application

This system can be used by everyone who wants to listen text documents instead of reading them. Text files can be easily translated into audio with the help of our system. People with learning disabilities who have difficulty reading large amounts of text due to dyslexia or other problems really benefit from TTS, offering them an easier option for experiencing contents. Those readers suffering from these problems gets benefit from this TTS system.

For those users tired of reading the documents can get real benefit from this software. Also, users exposed to digital gadgets for long time can enjoy this system which provides real time natural listening sound.

Synthesized speech can be used also in many educational situations. It can be used for special tasks like spelling and pronunciation teaching. It can also be used with interactive educational applications.

# 2. LITERATURE REVIEW

CB F. and RD Hudson [1] presented a text to speech system that can convert text from an electronic document into an audio output that includes speech associated with the text as well as audio contextual cues. They implemented there system in numerous ways, including as a method, system, or device (including a computer readable medium or a graphical user interface). As a computer-implemented method for converting text to speech, one embodiment of the invention can, for example, include at least: selecting a document to be converted to speech; parsing the selected document; converting text in the selected document to speech; and creating an audio file based on the converted text. According to another aspect of the invention, an audio summary can be generated for a file. The audio summary for a document can thereafter be presented to a user so that the user can hear a summary of the document without having to process the document to produce its spoken text via text-to-speech conversion. Documents as used herein pertain to electronic documents. The electronic documents are electrically stored in an electronic file on a computer readable medium. For example, a document used herein can be of various types and formats, including documents concerning text, word processing, presentation, webpage, electronic mail (e-mail), markup language, syndication, page description language, portable document format, etc.

Isewon I and his team [2] researched that text- to- speech synthesizer project is done by analyzing and processing the text using Natural Language Processing (NLP) and then using Digital Signal Processing (DSP) technology to convert this processed text into synthesized speech representation of the text. Their system interfaces with a text to speech engine developed for American English. They did it in the form of a simple application that converts inputted text into synthesized speech and reads out to the user which can then be saved as a .mp3 file. The system was developed using Java programming language. They have used a method called Concatenative Synthesis to produces the most natural-sounding synthesized speech. Concatenative synthesis is based on the concatenation (or stringing together) of segments of recorded speech. According to this paper, Text-to-speech synthesis takes place in several steps. The TTS systems get a text as input, which it first must analyze and then transform into a phonetic description. Then in a further step, it generates the prosody. From the information now available, it can produce a speech signal. The structure of the text-to-speech synthesizer can be broken down into major modules:

* Natural Language Processing (NLP) module: It produces a phonetic transcription of the text read, together with prosody.
* Digital Signal Processing (DSP) module: It transforms the symbolic information it receives from NLP into audible and intelligible speech.

Their system converts text to speech either by typing the text into the text field provided or by coping from an external document in the local machine and then pasting it in the text field provided in the application. It also provides functionality that allows the user to browse the World Wide Web (www) on the application. The system is capable of reading any portion of the web page the user browses. This can be achieved by the user highlighting the portion he wants to be read out loud by the system and then clicking on the “Play” button. The application was divided into two main modules - the main application module which includes the basic GUI components which handle the basic operations of the application such as the input of parameters for conversion either via file or direct keyboard input or the browser. This would make use of the open-source API called SWT and DJ Native Swing. In the second module, the main conversion engine integrated into the main module was for the acceptance of data hence the conversion. This would implement the API called free TTS.

Shetake P.S and her team [3] in the paper said that a text to speech converter convert’s normal language text into speech. Text to speech converter is useful in different applications. Customer support dialog systems Interactive voice response (IVR) systems etc. and are also useful in applied research. This application is more helpful in banking, toys, and many other applications like checking marks, railways, aid to the physically challenged persons, language education, and fundamental and applied research. etc. But text to speech conversion is not that easy for the machine as it is for human. Basic steps that machine has to follow for text to speech analysis are database creation, character recognition, and text to speech conversion. This paper surveys methods related to character recognition as well as approaches used for text to speech conversion for the machine. In this paper, there is discussed character recognition and speech synthesis techniques which are very useful to perform the task text to speech conversion. As stated, TTS is divided into two subproblems: character recognition and speech. To get the best speech synthesis rate, the database of the system should be large. So, there is scope to increase the database of the proposed system.

Swetha N. and Anuradha N. [4] In the research paper focused on the conversion of character-to-voice. In this paper, they presented the idea about developing a pc-based text-to-speech synthesizer using MATLAB. Here Text-To-Speech or speech synthesis is taken as an automatic production of speech, by ‘grapheme to phoneme’ transcription. In this paper, it is stated that the database of various recorded alphabets and digits in the form of wave files is created. Then the texts files of the text to be read are created. Then the .txt files are opened in MATLAB. The file is opened and read. For every character read, corresponding wave files are read. In this system, there are various problems in text preprocessing, such as numerals, abbreviations, and acronyms.

Jacob A. and Mythili P. [5] discussed the implementation details of a child-friendly, good quality, English TTS system that is phoneme-based, concatenative, easy to use with little memory. Direct waveform concatenation and linear prediction coding (LPC) are used. Most existing TTS systems are unit-selection based, which use standard speech databases available in neutral adult voices. Here reduced memory is achieved by the concatenation of phonemes and by replacing phonetic wave files with their LPC coefficients. The linguistic analysis was used to reduce the algorithmic complexity instead of signal processing techniques. Suﬃcient degree of customization and generalization catering to the needs of the child user had been included through the provision for vocabulary and voice selection to suit the requisites of the child. Prosody had also been incorporated. This inexpensive TTS system was implemented in MATLAB, with the synthesis presented using a graphical user interface (GUI), thus making it child friendly. This can be used not only as an interesting language learning aid for the normal child but it also serves as a speech aid to the vocally disabled child. The quality of the synthesized speech was evaluated using the mean opinion score (MOS) provision for varying the voice quality. Another feature of this work is that it was implemented using a female voice, whereas most of the successful LPC-based TTS have been implemented in the male voice. This TTS further has an add-on facility in that new words can be synthesized, after adding these words, their transcription, and constituent wave ﬁle names to the irrespective databases. The prosody of the utterance can be designed to vary depending on the nature of the recordings in the speech database from which the phoneme segments are excised. Thus, a simple, ﬂexible, and eﬃcient TTS that can be user deﬁned has been set up with minimum resources to serve multiple purposes. Though this had been developed for English, it can be suitably modiﬁed for any other language. This TTS was found to be a successful vocal aid/language learning aid as the users were able to get a real feel of phonemes, the most basic speech units. The learning environment can be conditioned to any particular accent by using an appropriate combination of database and pronunciation dictionary. Alternately, content speciﬁc learning too can be encouraged implicitly. By suitable design of the TTS vocabulary and database, a child can be familiarized with all common terms associated with any speciﬁc topic. Thus, such a TTS helps the child user get acquainted with the regular as well as any other selective vocabulary.

Tamura M. and his team [6] in the paper describes a technique for synthesizing speech with any desired voice. The technique is based on an HMM (Hidden Markov Model)-based text-to-speech (TTS) system and MLLR (Maximum Likelihood Linear Regression) adaptation algorithm. To generate speech of an arbitrarily given target speaker, speaker-independent speech units, i.e., average voice models, are adapted to the target speaker using the MLLR framework. In addition to spectrum and pitch adaptation, they derived an algorithm for adaptation of state duration. They demonstrated that a few sentences uttered by a target speaker are sufficient to adapt not only voice characteristics but also prosodic features. The synthetic speech generated from adapted models using only four sentences is very close to that from speaker-dependent models trained using a large amount of speech data. To convert the duration model parameters, they derived the MLLR algorithm for duration models. They have shown that synthetic speech generated from adapted models using average voice models becomes closer to the target speaker’s voice. Their future work is a subjective and objective evaluation of this technique.

Toda T. and Tokuda K. [7] present a text-to-speech system developed at Nara Institute of Science and Technology (NAIST) for the Blizzard Challenge 2015. They have developed their TTS system based on a statistical parametric speech synthesis technique using a hidden Markov model (HMM). To improve the quality of synthetic speech, they have newly implemented two techniques for the traditional HMM-based speech synthesis framework, pre-processing for producing smooth parameter trajectories to be modeled with HMM and speech parameter generation considering the modulation spectrum. The developed system has been submitted to the mono-lingual task and its performance has been demonstrated from the results of large-scaled subjective evaluation. Their research group, a speech synthesis group of Augmented Human Communication Laboratory, Nara Institute of Science and Technology (NAIST), studies various speech synthesis techniques, such as high-quality statistical parametric speech synthesis techniques, real-time voice conversion techniques for augmented speech production (e.g., voice/vocal effector or a speaking aid system for laryngectomees, towards the development of technologies to break down existing barriers in their speech communication. To submit a TTS system from their group the Blizzard Challenge 2015, they have developed their system, the NAIST TTS system based on a statistical parametric speech synthesis technique using Hidden Markov Model (HMM). To improve the quality of synthetic speech, two techniques are newly implemented for the traditional HMM-based speech synthesis framework preprocessing for producing smooth parameter trajectories to be modeled with HMM and speech parameter generation considering the modulation spectrum (MS) of speech parameters. The developed system has been submitted to the monolingual task and its performance has been demonstrated from there sorts of large-scaled subjective evaluations. This paper describes the details of the NAIST TTS system. They also brieﬂy discuss the results of large-scaled subjective evaluations on naturalness, similarity to the original speaker, and intelligibility, which they’re provided from the organizers. This paper has presented the NAIST TTS system for the Blizzard Challenge 2015. The pre-processing for smoothing parameter trajectories and the speech parameter generation considering the modulation spectrum have been implemented in their system. The results in the challenge have demonstrated that their system is capable of synthesizing naturally sounding speech.

Tokuda K. and his team [8] in their research paper gave a general overview of Hidden Markov model (HMM)-based speech synthesis, which has recently been demonstrated to be very effective in synthesizing speech. The main advantage of this approach is its ﬂexibility in changing speaker identities, emotions, and speaking styles. This paper also discusses the relation between the HMM-based approach and the more conventional unit-selection approach that has dominated over the last decades. Finally, advanced techniques for future developments are described. Index Terms—text-to-speech synthesis, hidden Markov model, HMM-based speech synthesis, statistical parametric speech synthesis, HTS. This paper gave a general overview of HMM-based speech synthesis and its recent advances. HMM-based speech synthesis has started to be used in daily life, e.g., cellphones, smartphones, in-car navigation systems, and call centers. Although the quality of synthesized speech generated by HMM-based speech synthesis has been drastically improved recently, its naturalness is still far from that of actual human speech. In the conversational speech, the naturalness of prosody is still insufﬁcient to properly convey non-verbal information, e.g., emotional expressions and emphasis. To ﬁll the gap between natural and synthesized speech, the statistical approaches described in the research paper will be more important in the future.

Parker J. and his team [9] presented a Visual text to speech system (VTTS) based on a deep neural network (DNN). Given an input text sentence and a set of expression tags, the VTTS can produce not only the audio speech, but also the accompanying facial movements. The expressions can either be one of the expressions in the training corpus or a blend of expressions from the training corpus. Furthermore, they present a method of adapting a previously trained DNN to include a new expression using a small amount of training data. Experiments show that the proposed DNN-based VTTS is preferred by 57.9% over the baseline hidden Markov model-based VTTS which uses cluster adaptive training. Index Terms used are Expressive Visual Text to Speech, Expression Adaptation, Deep Neural Network. In this paper, they present a system that, given a text sentence and some expressive tag, produces a photorealistic talking head. Talking heads can be used to improve human-machine interaction systems. Such systems have been used in education, in reading news or eBooks, and even in post-processing of ﬁlm production. In this paper, they present a DNN-based text-driven expressive talking head. The contributions of this work include an adaptation of audio DNN-based text to speech (TTS) to VTTS, a novel method of producing expressive visual speech using a DNN, yielding superior results compared to HMM-based VTTS, and a method of adapting a pre-trained TTS system to incorporate a new expression using a small amount of adaptation data. This paper presented a DNN-based expressive talking head. Unlike other systems, the entire face is modeled, which is important for expressive visual speech. Furthermore, new expressions can be added to the model with a small amount of adaptation data and without retraining the network. Their method outperforms an expressive HMM-based talking head. Attention is being turned to investigate alternate methods of modeling different expressions, in particular using a gated ﬁnal layer, where the ﬁnal layer models a three-way relationship between the penultimate layer, the expression, and the output. Furthermore, they wish to investigate the use of similar techniques to jointly model speaker and expression and so perform both speaker and expression adaptation.

Ifeanyi N., Ikenna O. and Izunna O. [10] in the paper presented a Text To Speech Synthesis technology that provides a means of converting written text from a descriptive form to a spoken language that is easily understandable by the end user (Basically in English Language). It runs on JAVA platform, and the methodology used was Object Oriented Analysis and Development Methodology; while Expert System was incorporated for the internal operations of the program. This design was geared towards providing a one-way communication interface whereby the computer communicates with the user by reading out textual document for the purpose of quick assimilation and reading development. The first methodology was Object Oriented Analysis and Development Methodology (OOADM). OOADM was selected because the system had to be represented to the user in a manner that was user-friendly and understandable by the user. They used speech synthesis Module in which they extracted the phonetic components of the message and obtained a string of symbols representing sound-units, boundaries between words, phrases and sentences along with a set of prosody markers (indicating the speed and intonation). They matched the sequence of symbols with items stored in the phonetic inventory and bound them together to form the acoustic signal for the voice output device.

Kamble K, Kagalkar R. [11] in the paper presented a single text-to-speech (TTS) system for Indian languages Viz., Hindi to generate speech .This generally involves two steps, text processing and speech generation. A graphical user interface has been designed for converting Hindi text to speech in Java Swings. This paper present text-to-speech (TTS) system based on the Concatenative synthesis approach. In this paper, they discussed the topics relevant to the development of TTS systems .The text to speech conversion may seem effective and efficient to its users if it produces natural speech and by making several modifications to it. Text to speech synthesis is a critical research and application area in the field of multimedia interfaces. In this paper, a speech synthesis system has been designed and implemented for Hindi Language. A database has been created from the various domain words and syllables. The given text is analyzed and syllabified based on the syllable segmentation rules. The desired speech is produced by the Concatenative speech synthesis approach. Speech synthesis is advantageous for people who are visually handicapped. This paper made a clear and simple overview of working of text to speech system (TTS) in step by step process. The Text to Speech System for Hindi using English Language is able to speak a loud Hindi word which is typed in English. The system read the input data in a natural form. The user types the input string and the system reads it from the database or data store where the words, phones, diaphones, triphone are stored. In this paper, they presented the development of existing TTS system by adding spellchecker module to it for Hindi language. There are many text to speech systems (TTS) available in the market and also much improvisation is going on in the research area to make the speech more effective, and the natural with stress and the emotions.

# 3. REQUIREMENT SPECIFICATION

## 3.1 Overall Description

### 3.1.1 Product Perspective

### 3.1.2 Product Function

* File Processing
* Text Processing
* Abbreviation Handling
* Voice Gender Selection
* Speech Generation

### 3.1.3 Operating Environment

Software Requirements

* Any web browser that supports HTML5
* Python 3.8

Hardware Requirements

* 1GB RAM
* 128MB free memory space

## 3.2External Interface Requirements

### 3.2.1 Hardware Interface

PC: Laptop or Desktop with windows installed. Properly working speaker, optical mouse and keyboard are needed for desktop and laptop.

### 3.2.2 Software Interface

Developing end

* Python v3.9.1
* Pycharm2020.2.2: IDE for Java developing
* Adobe Photoshop CC 2020 –Designing such as User Interface
* Any web browser supporting html5

## 3.3 System Features

### 3.3.1 Direct Text Entry

3.3.1.1 Description and Priority:

This feature provides user to provide direct input to the system.

Priority: High

3.3.1.2 Response sequence:

* User select text box.
* User inputs the data on textbox

3.3.1.3 Functional Requirement

User must be on the landing or main page.

### 3.3.2 Voice Gender Selection

3.3.2.1 Description and Priority:

This feature allows user to select one of two voices: male and female voice

Priority: Medium

3.3.2.2 Response sequence:

* User clicks on voice select assistant.
* User chooses between male and female voice.

3.3.2.3 Functional Requirement

User must be on the landing or main page.

### 3.3.3 File upload

3.3.3.1 Description and Priority:

This feature allows user to upload file. It also checks whether the file is of valid extension.

Priority: High

3.3.3.2 Response sequence:

* User clicks on upload button on main page.
* System redirects to upload page.
* User clicks on browse tab.
* System displays the browse window.
* User select the file to be uploaded

3.3.3.3 Functional Requirement

User must be on the landing or main page and must have clicked upload button.

### 3.3.4 Speech Generation of Direct text input

3.3.4.1 Description and Priority:

This feature allows user to generate the speech of the text entered and the text of the uploaded file of valid extensions.

Priority: High

3.3.4.2 Response Sequence

* User press the Play button.
* User select the gender of voice.
* If gender not selected, system sets default gender of voice as Male.
* System generates the speech.

3.3.4.3 Functional Requirements:

User must have entered the main page and the text must have been entered in the text box. The user selects the gender of voice and press the play button.

### 3.3.5 Speech Generation of Uploaded file

3.3.5.1 Description and Priority:

This feature allows user to generate the speech of extracted text of the uploaded file of valid extensions.

Priority: High

3.3.5.2 Response Sequence:

* User enters the upload page.
* System redirects to upload page.
* User browses and selects file of valid extension.
* System validates the file extension.
* System extracts the text from the file and generate the speech

3.3.5.3 Functional Requirements

User must be on the upload page and selects the file of valid extension and clicks upload button.

## 3.4Other Non-Functional Requirements

### 3.4.1 Performance Requirements

* Response time- The system will generate speech within 3 sec of clicking the play button.
* Capacity- The system must support 10,000 words at a time.

### 3.4.2 Safety Requirements

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage.

### 3.4.3 Software Quality Attributes

Availability: The system shall be available all the time.

Correctness: Bug free software which fulfill the correct need/requirements of the client.

Maintainability: The ability to maintain, modify information and update fix problems of the system.

Usability: software can be used again and again without distortion.

Accessibility: Administrator and many other users can access the system but the access level is controlled for each user according to their work scope.

Stability: The system outcome/output won’t change time to time. Same output will be given always for a given input.

# 4. SYSTEM ANALYSIS

## a. Activity Diagram of main process

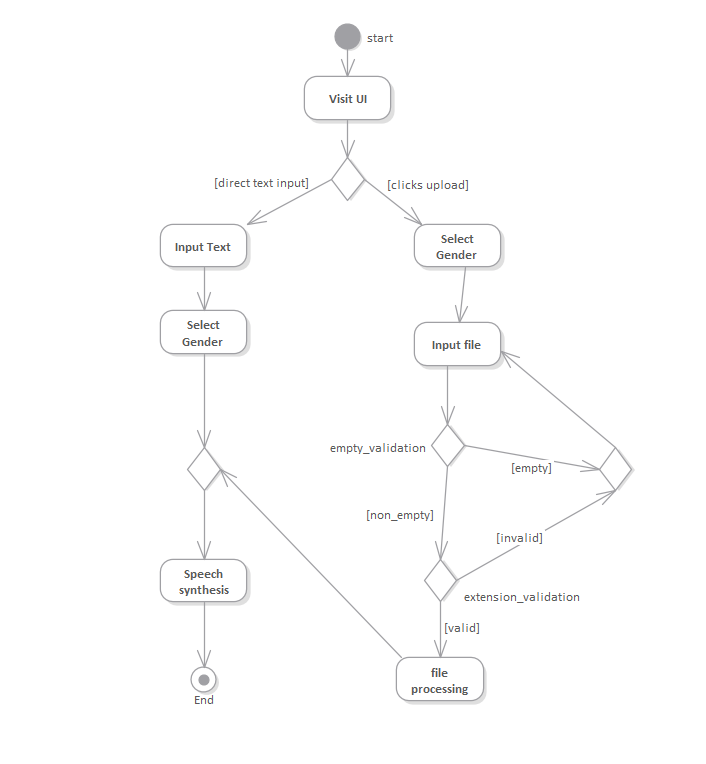


Figure Activity Diagram of main process

As shown in figure 5: Activity Diagram, user visit the user interface. The input can be either direct text or the file upload. The user selects the gender and send the text and gender to synthesize the speech in case of direct input.

For file upload, the user selects the file. The system checks for empty file upload and then for supporting extensions(.pdf,.docx,.txt). If both the conditions are true then the file is processed and sent for speech synthesis. If there is empty upload or invalid extension, the error message is displayed and ask for reuploading the file.

## b. Domain Model

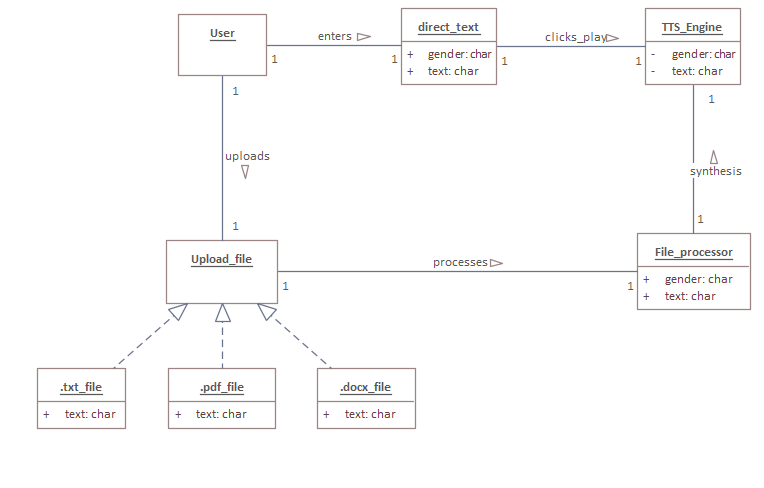


Figure Domain Model

## c. Sequential Model

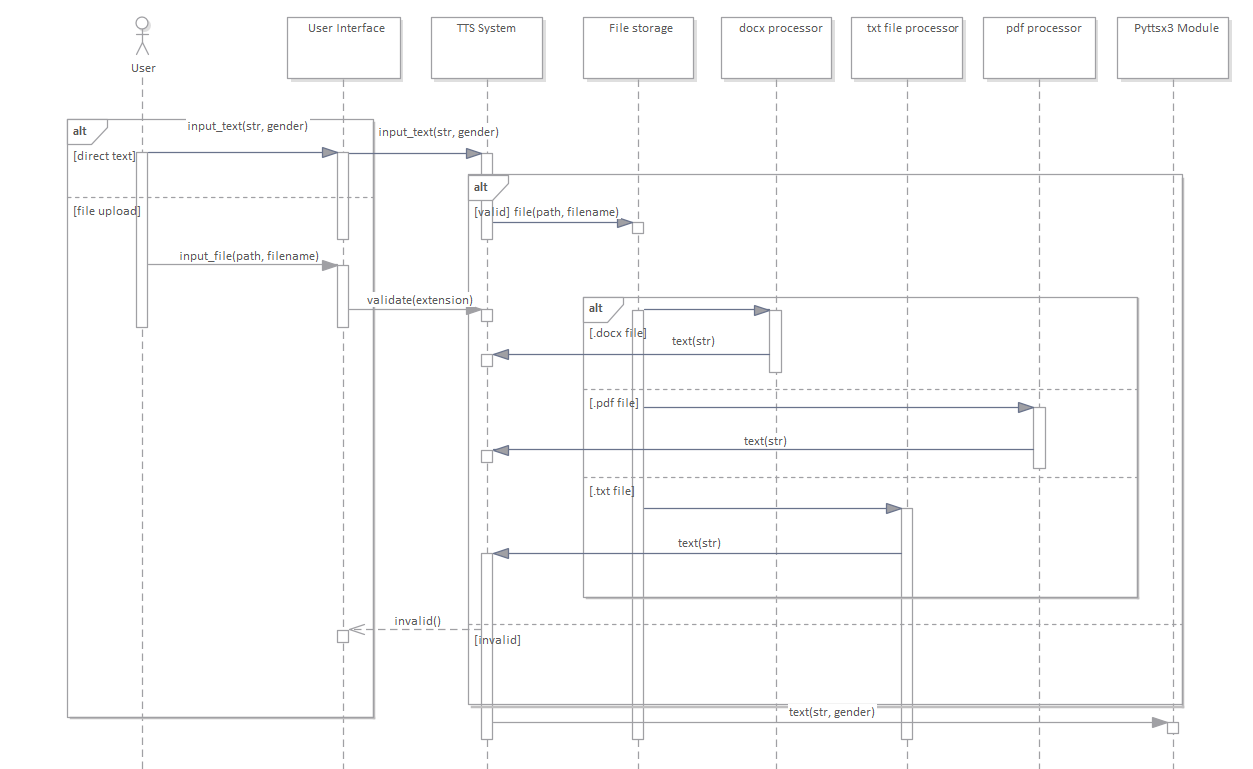


Figure Sequential Model of System

As shown in figure 4: System Sequence Diagram, the input from user to the user interface is either direct input\_text or the input\_file. If it is direct text or file with the valid extension, message is forwarded to TTS System. For the file uploaded, the file is passed to respective processor which extract text as string and passes to TTS System which further passes the text and gender to Pyttsx3 module for speech generation.

If the file extension is invalid, TTS System sends message to user interface.

# 5. SYSTEM DESIGN

## b. Class Diagram

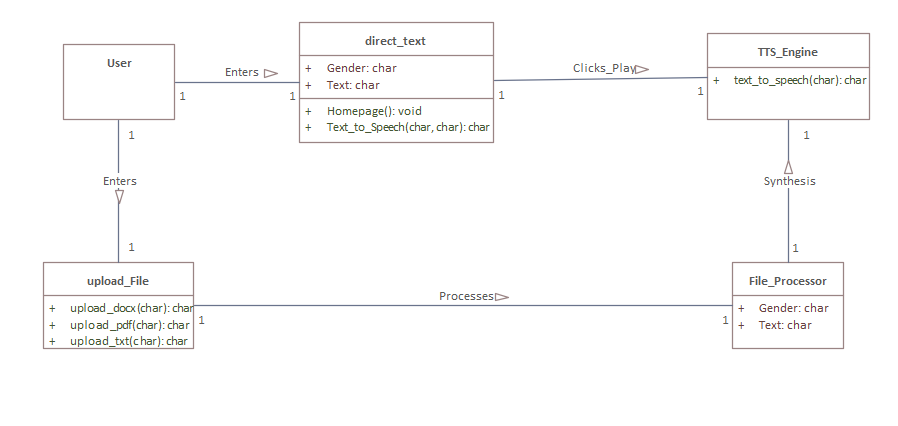


Figure Class Diagram of main process

## a. UI design

# 6. IMPLEMENTATION

## a. Main Algorithm

## b. Main class with functions

# 7. UNIT TESTING

# 8. SYSTEM TESTING

# 9. DISCUSSION

# 10. CONCLUSION

# 11. REFERENCES

[1] Christopher Brian, and Reginald Dean Hudson. "Intelligent text-to-speech conversion." U.S. Patent No. 8,996,376. 31 Mar. 2015.

[2] I.Isewon, J.Oyelade, O.Oladipupo*. "Design and Implementation of Text to Speech Conversion for Visually Impaired People"*, International Journal of Applied Information Systems,2014

[3] P.S.Shetake, S.A.Patil, P.M.Jadhav. *"Review of Text To Speech Conversion Methods"*, International Journal of Industrial Electronics and Electrical Engineering (IJIEEE), 2014

[4] N.Swetha,K.Anuradha, *"Text-to-speech Conversion”,* International Journal of Advanced Trends in Computer Science and Engineering,2013

[5] A.Jacob and P.Mythili. *"Developing a Child Friendly Text-to-Speech System".,* https://doi.org/10.1155/2008/597971,2008

[6] Masatsune Tamura, Takashi Masuko, Keiichi Tokuda, Takao Kobayashi *"Adaptation of pitch and spectrum for HMM-based speech synthesis using MLLR”, IEEE* International Conference on Acoustics, Speech, and Signal Processing. Proceedings, 2001

[7] T.Toda, K.Tokuda, *"A speech parameter generation algorithm considering global variance for HMM-based speech synthesis",* IEICE Transactions on Information and Systems, 2007

[8] K.Tokuda, Y. Nankaku, T.Toda, H.Zen, J.Yamagishi, *"Speech synthesis based on hidden Markov models",* Proceedings of the IEEE, 2013

[9] J.Parker,Y.Stylianou, R.Cipolla,R.Maia, "Expressive visual text to speech and expression adaptation using deep neural networks", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),2018

[10]Ifeanyi N, Ikenna O and Izunna O. Text-To-Speech Synthesis (TTS). Nigeria: IJRIT (International Journal of Research in Information Technology), Volume 2, May 2014.

[11]Kamble K, Kagalkar R. *Translation of Text To Speech Conversion for Hindi language.* India: International Journal of Science and Research (IJSR). 2012.