



**St.PETER'S**  
**ENGINEERING COLLEGE**

Mansamaguda, Dullapally, Kompally Road, Hyderabad, Telangana 500100.

Affiliated to JNTU -Hyderabad & Approved  
By AICTE, New Delhi

## **TEXT-TO-SPEECH**

An Application Development Lab Report Submitted  
In partial fulfillment of the requirements for the award of the degree of

**Bachelor of  
Technology in  
Computer Science and Engineering(AI&ML)**

**SUBMITTED BY:-**

KISTARAM BHUPATHI REDDY 21BK5A6604



## **ABSTRACT**

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. Text-to-speech synthesizer (TTS) is the technology which lets computer speak to you. The TTS system gets the text as the input and then a computer algorithm which called 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.

Speech synthesis can be described as artificial production of human speech [3]. A computer system used for this purpose is called a speech synthesizer, and can be implemented in software or hardware. A text-to-speech (TTS) system converts normal language text into speech [4]. Synthesized speech can be created by concatenating pieces of recorded speech that are stored in a database. Systems differ in the size of the stored speech units; a system that stores phones or di phones provides the largest output range, but may lack clarity. For specific usage domains, the storage of entire words or sentences allows for high-quality output. Alternatively, a synthesizer can incorporate a model of the vocal tract and other human voice characteristics to create a completely "synthetic" voice output [5]. The quality of a speech synthesizer is judged by its similarity to the human voice and by its ability to be understood. An intelligible text-to-speech program allows people with visual impairments or reading disabilities to listen to written works on a home computer.

**Keywords:** Text, audio

## TABLE OF CONTENTS

| <b>S.NO</b> | <b>TITLE</b>  | <b>PG.NO</b>                |
|-------------|---|-----------------------------|
| <b>1</b>    | <b>INTRODUCTION</b><br>PURPOSE,AIM AND OBJECTIVES<br>BACKGROUND OF PROJECT<br>MODULES DESCRIPTION         | <b>09</b><br>09<br>10<br>11 |
| <b>2</b>    | <b>SYSTEM ANALYSIS</b><br>HARDWARE AND SOFTWARE<br>REQUIREMENTS<br>SOFTWARE REQUIREMENTS<br>SPECIFICATION | <b>12</b><br>12<br><br>12   |
| <b>3</b>    | <b>TECHNOLOGIES USED</b><br>JAVASCRIPT & HTML   | <b>13</b><br>13             |
| <b>4</b>    | <b>SYSTEM DESIGN&amp;UML DIAGRAMS</b><br>ARCHITECTURE<br>UNIFIED MODELING LANGUAGE                        | <b>14</b><br>14<br>15       |
| <b>5</b>    | <b>IMPLEMENTATION</b>   | <b>18</b>                   |
| <b>6</b>    | <b>OUTPUTSCREENS</b>  | <b>30</b>                   |
| <b>7</b>    | <b>CONCLUSION</b>   | <b>32</b>                   |

|          |                    |           |
|----------|--------------------|-----------|
| <b>8</b> | <b>FUTURESCOPE</b> | <b>33</b> |
|----------|--------------------|-----------|

## LIST OF FIGURES

| FIGURE.NO | NAME                        | PG.NO |
|-----------|-----------------------------|-------|
| 4.1       | SYSTEM ARCHITECTURE DIAGRAM | 14    |
| 4.2       | CLASS DIAGRAM               | 15    |
| 4.3       | USE CASE DIAGRAM            | 16    |
| 4.4       | SEQUENCE DIAGRAMS           | 17    |

## OUTPUT SCREENS

| FIGURE.NO | NAME            | PG.NO |
|-----------|-----------------|-------|
| 6.1       | OUTPUT SCREEN 1 | 30    |
| 6.2       | OUTPUT SCREEN 2 | 31    |

## 1. 1. INTRODUCTION

### **PURPOSE AND AIM:**

Text-to-speech (TTS) is a type of assistive technology that reads digital text aloud. It's sometimes called "read aloud" technology.

With a click of a button or the touch of a finger, TTS can take words on a computer or other digital device and convert them into audio. TTS is very helpful for kids who struggle with reading. But it can also help kids with writing and editing, and even focusing. Text-to-Speech (TTS) is the ability of a computer to produce spoken words by converting text to voice. In other words Text-to-Speech software is a speech synthesizer that vocalizes text in real time in a natural way. Text-to-Speech technology can be used in various areas development platforms.

**-Education:** Text-to-Speech technology can be used in language teaching applications to vocalize any word in order to improve the pronunciation capabilities of the users.

**-Accessibility:** Text to speech technology can read text out loud and enables the use of computers and mobile devices for the disabled and for people with special needs. For example menus of ATMs are vocalized with Text to Speech technology in order to provide enhanced customer experience especially for the disabled. Web site and newspaper vocalization can also be listed among accessibility-related use of TTS. The voice in TTS is computer-generated, and reading speed can usually be sped up or slowed down. Voice quality varies, but some voices sound human. There are even computer-generated voices that sound like children speaking.



## **OBJECTIVES:**

Text-to-speech refers to the ability to convert computer readable text into natural soundings speech base on in-depth linguistic understanding. It transforms linguistic information stored as data or text into speech .

The objectives of Text-to-Speech are:

1. The system is helpful for persons having learning disabilities or visually challenged.
2. Prevents eye from strain, and user can sit and listen comfortably.
3. Saves time especially while driving, exercising.
4. Easy to use.
5. Help improving spelling, reading, writing skills.

## **BACKGROUND OF PROJECT:**

Speech synthesis, or the artificial production of the human voice, has come a long way over the last 70 years. Whether you currently use text-to-speech services to listen to books, study, or proofread your own written work, there's no doubt that text-to- speech services have made life easier for people in a variety of professions. Text-to- speech technology can help people who are visually impaired and live with other disabilities get the information they need to thrive at work and to communicate with others.

The software also allows students and others with heavy workloads of reading to listen to their information via human speech when they're on the go. Synthetic speech allows people to get more done in less time, and can be useful in a variety of settings, from video game creation to helping people with language processing differences

## **MODULES DESCRIPTION:**

### **Volume:**

The html audio volume property is used for set or return the current volume of the audio

The volume change event occurs each time the volume of a video/audio has been changed. The volume property gets and sets the volume of the utterance. It is a float that represents the volume value, between 0 (lowest) and 1 (highest). The default value is 1 if this property is unset.

This event is invoked by:

- Increasing or decreasing the volume
- Muting or un muting the media player

**Rate:** The rate property gets and sets the rate of the utterance. It is a float representing the rate value which can range between 0.1 (lowest) and 10 (highest). The default value is 1 if this property is unset.

**Pitch:** The pitch property gets and sets the pitch of the utterance. It is a float representing the pitch value that can range between 0 (lowest) and 2 (highest). The default pitch is 1 if this property is unset.

**Voice:** The voice property gets and sets the voice that will be used to speak the utterance. This should be set to one of the Speech Synthesis Voice objects. If it is not set, the most suitable default voice available for the utterance's language setting will be used.



## 2. 2. SYSTEM ANALYSIS

### **HARDWARE AND SOFTWARE REQUIREMENT:**

#### **Hardware Requirements :**

1. Processor: Intel® Core™ i3-9750H
2. CPU@2.6GHz or above
3. RAM: 4 GB RAM or above
4. Storage:2 GB or above
5. Active Internet Connection
6. Microphone
7. Speakers

#### **Software Requirements:**

1. Java
2. CSS
3. HTML

### 3.3. TECHNOLOGIES USED

#### JAVASCRIPT:

JavaScript is a very free-form language compared to Java. You do not have to declare all variables, classes, and methods. You do not have to be concerned with whether methods are public, private, or protected, and you do not have to implement interfaces. Variables, parameters, and function return types are not explicitly typed.

**JavaScript** is a lightweight, cross-platform, and interpreted compiled programming language which is also known as the scripting language for web pages . It is well-known for the development of web pages, many non-browser environments also use it. JavaScript can be used for client-side developments as well as server side developments

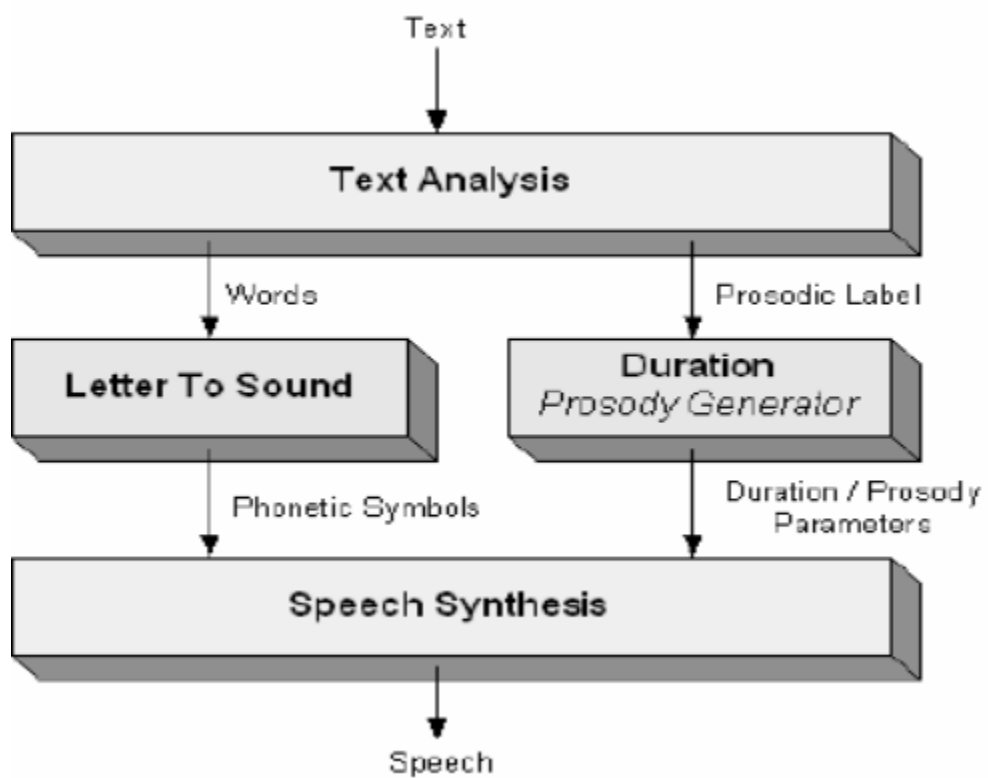
JavaScript is a multi-paradigm, dynamic language with types and operators, standard built-in objects, and methods. Its syntax is based on the Java and C languages — many structures from those languages apply to JavaScript as well. JavaScript supports object-oriented programming with object prototypes and classes. It also supports functional programming since functions are first-class that can be easily created via expressions and passed around like any other object.

#### HTML:-

HTML stands for Hypertext Markup Language and it is a widely used programming language used to develop web pages. Through this HTML Tutorial, we'll understand what HTML is. HTML is a markup language that is used to create web pages. . It defines how the web page looks and how to display content with the help of elements. It forms or defines the structure of our Web Page, thus it forms or defines the structure of our Web Page. We must remember to save your file with .html extension

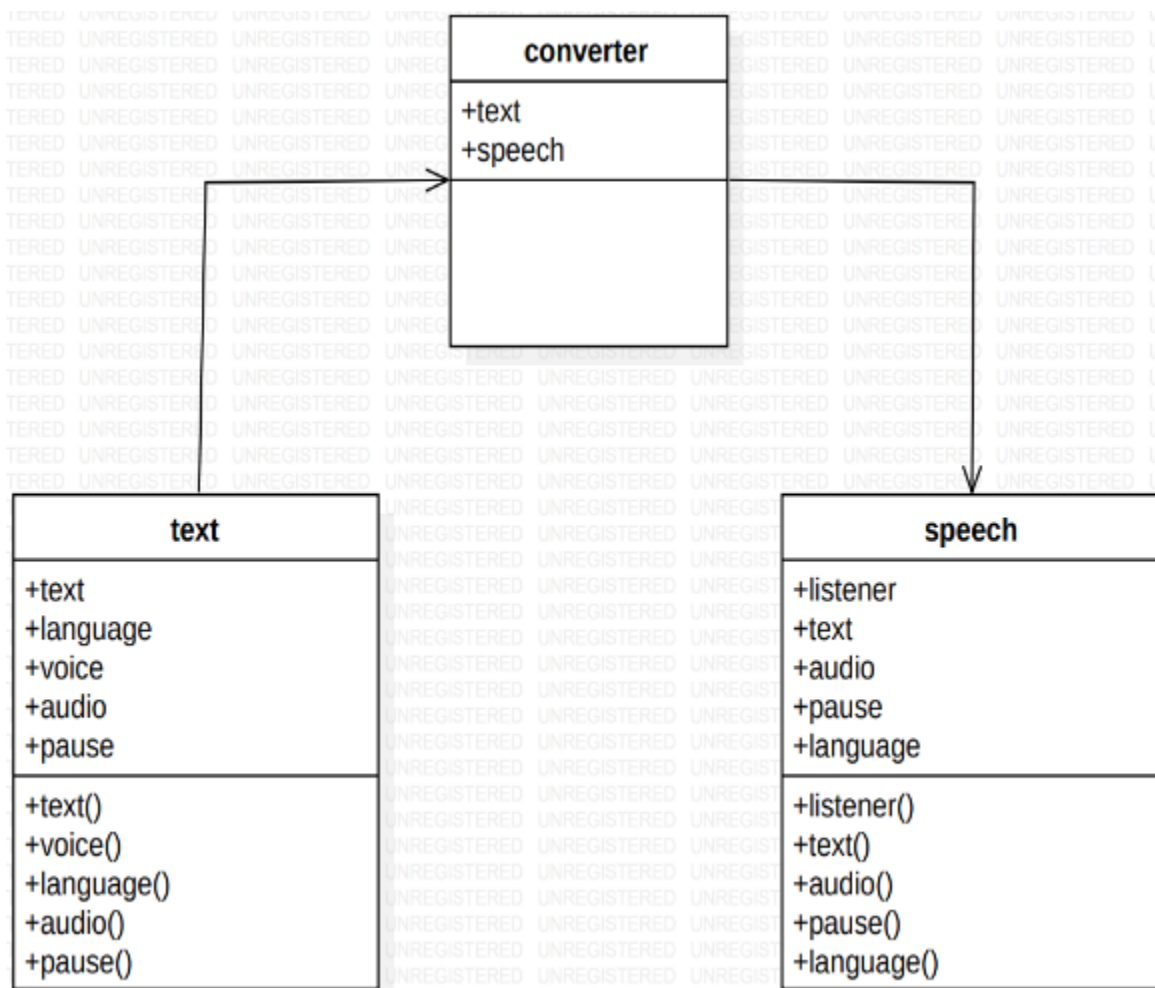
#### 4. 4. SYSTEM DESIGN & UML DIAGRAMS

##### SYSTEM ARCHITECTURE



**Fig.4.1 System architecture**

## CLASS DIAGRAM

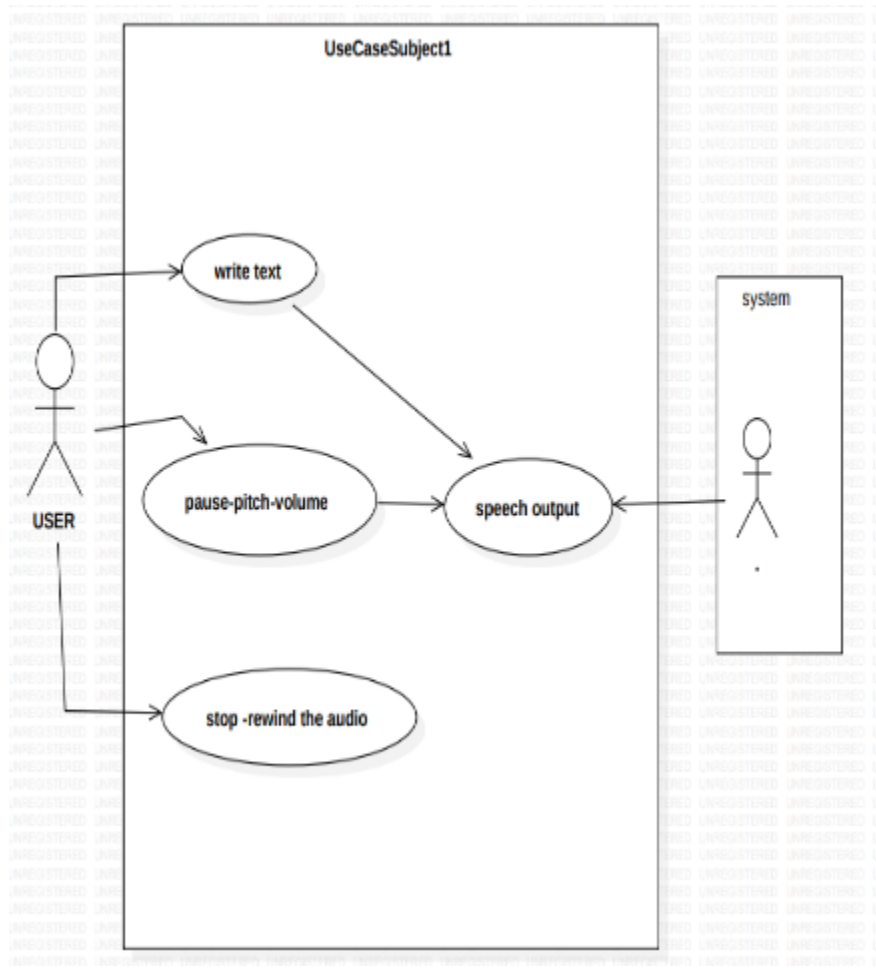


**Fig.4.2 Class diagram**



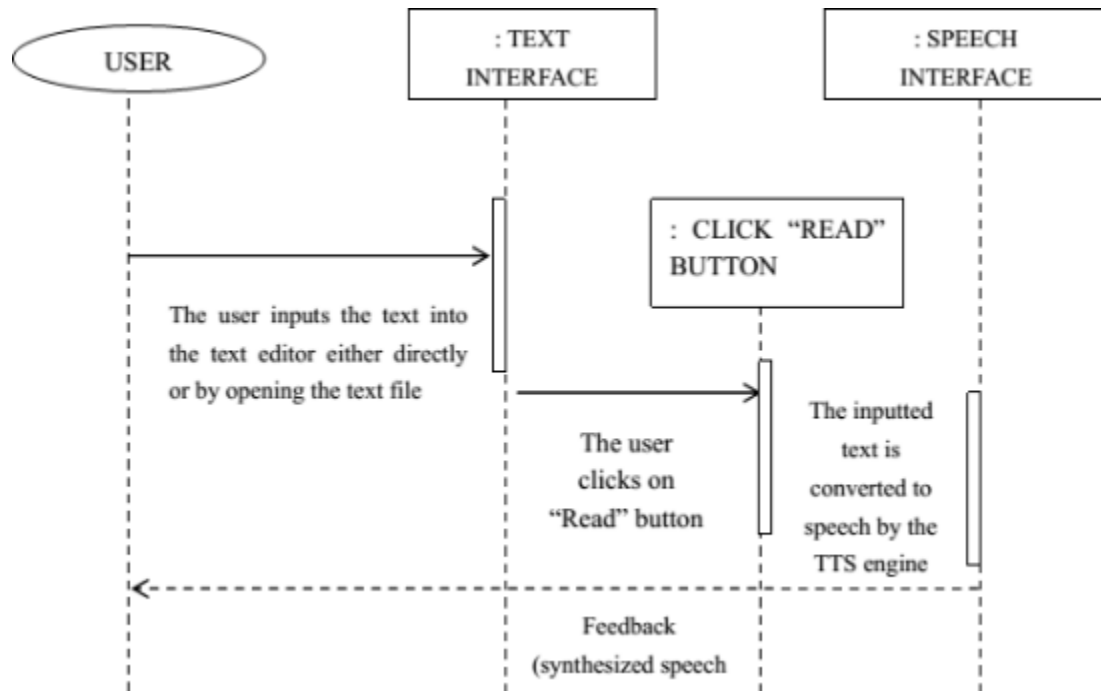


## USE CASE DIAGRAM



**Fig.4.3 Use case diagram**

## SEQUENCE DIAGRAM



**Fig.4.4 sequence diagram**



## 5. IMPLEMENTATION

```
//for speech synthesis

let      speech      =      new
SpeechSynthesisUtterance(); speech.lang
= "en"; //default language letlangInput =
document.getElementById('lang');
//variable calls

lettextVal = document.querySelector("#txt-
val");      letspeechBtn      =
document.querySelector(".play");
letpauseBtn      =
document.querySelector(".pause");
letcancelBtn      =
document.querySelector(".cancel");
letresumeBtn      =
document.querySelector(".resume");
//for voices

letvoiceSelection = []; //global array

letvoiceInput = document.getElementById("voices");
//for text display

let heading = document.querySelector("h3")
//for volume control

letvolControl  =  document.getElementById('vol');
letpitchControl = document.getElementById('pitch');
letrateControl  =  document.getElementById('rate');
letvolumeSpan  =  document.getElementById("vol-
control");      letpitchSpan      =
document.getElementById("pitch-control");
```

```

letrateSpan = document.getElementById("rate-
control");

//spans for language switch

letlangTag = document.querySelector(".lang-div
span");          letvoiceTag          =
document.querySelector(".voice-div      span");
letvolTag = document.querySelector(".volume-div
span");

letrateTag = document.querySelector(".rate-div
span");          letpitchTag          =
document.querySelector(".pitch-div span");
//for      displaying      voice      options
window.speechSynthesis.onvoiceschanged =
() => {
voiceSelection = window.speechSynthesis.getVoices(); //adding voice in global
arrayspeech.voice = voiceSelection[0];
voiceSelection.forEach(voiceOptions); //calling inside the array for each option
functionvoiceOptions(voice, i) {
(voiceInput.options[i] = new Option(voice.name, i));
}
};

//to      read      text
speechBtn.addEventListener('click',
() => { speech.text = textVal.value;
window.speechSynthesis.speak(spe
ech);
});

//to      change      voice
voiceInput.addEventListener('change', ()

```

```

=> { speech.voice =
voiceSelection[voiceInput.value];
console.log(voiceInput.value);
});

//to change language
langInput.addEventListener('change',
() => { console.log(langInput.value);
switch
(langInput.value) {
case 'english':
speech.lang
g = 'en';
break;
case
'hindi':
speech.lang
g = 'hi';

heading.innerHTML = "हन्दीमेंभाषणके हएपाठ";
textVal.setAttribute('placeholder', 'अपनापाठसुननेके हएयाँहखें');
langTag.innerHTML = "भाषा";
voiceTag.innerHTML =
"ध्वहन"; volTag.innerHTML =
"मात्रा"; rateTag.innerHTML =
"गह"; pitchTag.innerHTML =
"स्वरमान";break;
}

});

//to control volume | pitch | rate
volControl.addEventListener('input',
() => { speech.volume =

```

```

volControl.value;
volumeSpan.innerHTML      =
volControl.value;
});

pitchControl.addEventListener('input',
()    =>    {    speech.pitch    =
pitchControl.value;
pitchSpan.innerHTML      =
pitchControl.value;
});

rateControl.addEventListener('input',
()    =>    {    speech.rate    =
rateControl.value;
rateSpan.innerHTML      =
rateControl.value;
});

//button controls

//pause-play
pauseBtn.addEventListener('click'
,                               ()=>{
window.speechSynthesis.pause();
})

resumeBtn.addEventListener('click'
,                               ()=>{
window.speechSynthesis.resume();
})

cancelBtn.addEventListener('click'
,                               ()=>{
window.speechSynthesis.cancel();

```

```
})
```

```
@import
```

```
url('https://fonts.googleapis.com/css2?family=Comfortaa:wght@300&display=swa  
p');
```

```
:root {
```

```
--theme-color: #FA0748;
```

```
}
```

```
*{
```

```
font-family: 'Comfortaa', cursive;
```

```
}
```

```
.conta
```

```
iner {
```

```
displa
```

```
y:
```

```
flex;
```

```
flex-direction:
```

```
column; justify-
```

```
content: center;
```

```
align-content:
```

```
center; align-
```

```
items: center;
```

```
background-color: var(--theme-
```

```
color);color: white;
```

```
margin:
```

```
auto;
```

```
margin-top:
```

```
0.2em;
```

```
width: 20%;
```

```
height:
```



```
88vh;
padding:
2em;
border-radius:
0.5em; gap:
0.5em;
max-height: fit-content;

}
h3 {

font-size:
2em; font-
weight:
100; text-
align:
center;

font-weight: bolder;

}
/* voice options and language options */

.optio
ns {
displa
y:
flex;
flex-direction:
column; justify-
content: center;
align-items:
center; gap:
0.5em;
```

```
color:      var(--
theme-color); font-
weight: bold;
font-size: 1em;

}
```

```
.voice-div,
.lang-div{
display: flex;
justify-content: space-
between;      align-
items:center;
gap: 0.5em;

background-color:
white;  padding:
0.5em;
border-radius:
0.2em; width:
105%;
}
```

```
select {
padding:
0.6em;
border-style:
none; border-
radius: 0.2em;
width: 80%;
font-
weight:
bold;color:
white;
```

```
outline-color:    var(--theme-  
color); background-color: var(-  
-theme-color);  
}
```

```
select:focus{

outline-color: var(--theme-color);

}

option
{
padding
g: 1em;
}

/* output buttons */

.button-
control{
display:
flex;
justify-content:
center;  align-
items:  center;
border-radius:
0.2em;  gap:
1em;
background-color:
white;  padding:
0.5em      1em;
margin-bottom:
1em;
}

.button-control
button{ display:
flex;
```

```
justify-content:
center; align-
items: center;
padding: 0em;
border-
style: none;
transition:
all 0.8s;
color: var(--theme-color);

}

.button-
control i{
font-size:
3em;
background-color:
white; text-align:
center;
/* text-shadow: -1px 1px 5px var(--theme-color); */

}

.button-control button:hover {

cursor:
pointer;
transform:
scale(1.3);
}

/* input field and voice controls */

.input-
control
{
```

```
display:
flex;
flex-direction:
column; justify-
content: center;
align-items:
center; gap:
0.2em;
width: 90%;

}
textar
ea {
width:
100%;
height
: 5em;
resize:
none;
border-radius:
0.2em; border-
style: none;
padding: 1em;
font-
size:
2em;
display:
flex;
justify-content:
center; align-
items: center;
```

```
color:      var(--
theme-color);
}
textarea:focus {

outline-color: var(--theme-color);

}
.voice-
control {
display:
flex;

flex-direction:
column; justify-
content:center;
align-items:
center;      gap:
0.4em;
padding:
0.5em; border-
radius: 0.2em;
color: var(--theme-
color);      width:
100%;
}
.volume-div,

.rate-div,

.pitch-div {

background-color:
white;      display:
flex;
```

```

align-items:
center; width:
120%;
padding:
0.2em; border-
radius: 0.2em;
}

/* for slider styling */

.voice-control input[type="range"] {

    -webkit-
appearance:    none;
border-radius: 2em;
height: 0.4em;

background-color: var(--theme-color);

}

.voice-control
span { border-
radius: 0.2em;
width: 50%;
text-align:
center;
padding:
0.5em;
margin:
auto;

```



```
font-weight: bold;

}

@media screen and (max-width:1000px) and (max-height:1500px) {
.container
{ margin:
auto;
align-self:
center;
width:
90%;
height:
90vh;
padding:
1em; gap:
1em;
text-align: center;

max-height: fit-
content; font-
size: 0.8em;
}

/* voice options and language options */

.opti
ons
{
gap:
0.5e
m;
font-size: 1em;
}
```

```
.voice-div,  
.lang-div{  
width: 85%;  
}
```

```
select  
{  
width:  
100%;  
}
```

```
option  
{  
padding  
g: 1em;  
}
```

```
.button-  
control i{  
font-size:  
2em;z  
background-color:  
white; text-align:  
center;  
/* text-shadow: -1px 1px 5px var(--theme-color); */  
}
```

```
.button-control  
button:hover { cursor:  
pointer;  
transform: scale(1.3);  
}
```

```
/* input field and voice controls */textarea {
  width
  h:
  90%;
  heigh
  t:
  5em;
}
textarea:focus {
  outline-color: var(--theme-color);
}
.voice-
control {
  width:
  70%;
}}
```

## 6. 6. OUTPUT SCREENS



**Fig.6.1**



**Fig.6.2**

## **7. CONCLUSION**

In conclusion, I think Although speech-to-text conversion (STT) machines aim at providing benefits for the deaf or people who can't speak, it is difficult to review, retrieve and reuse speech transcript.this residue.

Text to speech synthesis is a rapidly growing aspect of computer technology and is increasingly playing a more important role in the way we interact with the system and interfaces across a variety of platforms. We have identified the various operations and processes involved in textto speech synthesis. We have also developed a very simple and attractive graphical user interface which allows the user to type in his/her text provided in the text field in the application

## **8. 8. FUTURESCOPE**

The future of wild reality is that it could possibly overthrow some forms of education. The technology is already being used to help people with disabilities and not able to read, but it could advance even more. It could be a possibility that text to speech converters end up taking over the education system because it would be cheaper for a school to pay a converter than a full-time teacher employee! Even though that would be a long way away before it would happen, it is still a crazy thought to ponder