**<YOUR\_UNIVERSITY\_NAME>**

**<YOUR\_UNIVERSITY\_LOGO>**

A

MINI PROJECT-**<PROJECT\_CODE>**

**“TEXT TO SPEECH”**

Partial Fulfilment of degree for the awards of the degree of

**<YOUR\_DEPARTMENT\_NAME>)**

<YOUR\_SESSION>

Under the Guidance of

**<YOUR\_INSTRUCTOR\_NAME>**

Submitted by

**<YOUR\_NAME>**

**<YOUR\_ROLL-NUMBER>**

**<YOUR\_COLLEGE\_LOGO>**

**<YOUR\_COLLEGE\_NAME>**

# DECLARATION

We hereby declare that the project work entitled **“TEXT TO SPEECH”** submitted to **MR/MRS. <INSTRUCTOR\_NAME>** is a record of an original work done by us. This project work is submitted in the partial fulfilment of the requirement for the award of the degree of the **“<DEPARTMENT\_NAME>”.** The result embodied in this thesis has not been submitted to any other university or institute for the any degree or diploma.

**<YOUR\_NAME>**

**<YOUR\_ROLL-NO>**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**<INSTRUCTOR\_NAME>** *(project supervisor)*

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# ACKNOWLEDGEMENT

I would like to express my special thanks of gratitude to my project guide **<INSTRUCTOR\_NAME>** sir who gave me the golden opportunity to do this wonderful project on the topic **TEXT TO SPEECH**, which also helped me in doing a lot of research and i came to know about so many new things I am really thankful to them.

Secondly I would also like to thank my parents and friends who helped me a lot in finalizing this project within the limited time frame.

**<YOUR\_NAME>**

**<YOUR\_ROLL-NO>**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# TABLE OF CONTENTS

[DECLARATION 1](#_Toc129672334)

[ACKNOWLEDGEMENT 2](#_Toc129672335)

[TABLE OF CONTENTS 3](#_Toc129672336)

[ABSTRACT 5](#_Toc129672337)

[SOFTWARE DEVELOPMENT LIFE CYCLE MODEL 7](#_Toc129672338)

[Waterfall SDLC Model 7](#_Toc129672339)

[REQUIREMENT GATHER 9](#_Toc129672340)

[Introduction to Project: 9](#_Toc129672341)

[Objective: 10](#_Toc129672342)

[Existing System: 10](#_Toc129672343)

[Proposed System 10](#_Toc129672344)

[REQUIREMENT ANALYSIS 11](#_Toc129672345)

[Software requirement: 11](#_Toc129672346)

[Features: 11](#_Toc129672347)

[Advantage: 11](#_Toc129672348)

[Disadvantage: 11](#_Toc129672349)

[Applications: 12](#_Toc129672350)

[DESIGNING OF PROJECT 13](#_Toc129672351)

[GUI Designing tools: 13](#_Toc129672352)

[Figma: 13](#_Toc129672353)

[Set up Figma: 13](#_Toc129672354)

[Proxlight Designer: 14](#_Toc129672355)

[Functionality designing tools: 15](#_Toc129672356)

[Project Activity Diagram: 15](#_Toc129672357)

[Use Cases of the project: 15](#_Toc129672358)

[Programming tools: 16](#_Toc129672359)

[Python programming language: 16](#_Toc129672360)

[Downloading and Installation of python: 16](#_Toc129672361)

[VSCODE: 16](#_Toc129672362)

[Installing Visual Studio Code: 16](#_Toc129672363)

[Project Libraries: 18](#_Toc129672364)

[Tkinter: 18](#_Toc129672365)

[Geometry Management 20](#_Toc129672366)

[Pyttsx3: 20](#_Toc129672367)

[Pillow: 21](#_Toc129672368)

[Project Snapshot: 22](#_Toc129672369)

[1. Application Screenshot 22](#_Toc129672370)

[2. Voice change and Voice rate multiple option screenshot. 22](#_Toc129672371)

[How to we add voices:- 23](#_Toc129672372)

[Project Components: 24](#_Toc129672373)

[IMPLEMENTATION 26](#_Toc129672374)

[TESTING 34](#_Toc129672375)

[CONCLUSION 37](#_Toc129672376)

[SCOPE 38](#_Toc129672377)

# ABSTRACT

In this mini project we made GUI software, known as “TEXT TO SPEECH”*.* Current state-of-the-art text-to-speech systems produce intelligible speech but lack the prosody of natural utterances. In recent years, the most popular acoustic model in automatic speech recognition (ASR) and text-to-speech synthesis (TTS) is a hidden Markov model (HMM), due to its ease of implementation and modelling flexibility.

Project TTS(text to speech) is developed using the python programming language, since the python is easy to implement and have a lot of predefined modules for web and desktop app development, we uses the “pyttsx3” module.

In this project we worked upon python text to speech module pyttsx3 version 3, pyttsx3 is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline and is compatible with both Python 2 and 3. An application invokes the pyttsx3.init() factory function to get a reference to a pyttsx3. Engine instance. it is a very easy to use tool which converts the entered text into speech.

TTS project is a gui program, in which we uses TKINTER module is used for designing graphical user interface. The tkinter package (“Tk interface”) is the standard Python interface to the Tcl/Tk GUI toolkit. Both Tk and tkinter are available on most Unix platforms, including macOS, as well as on Windows systems. When we think about GUI then we think about modern look to our project, so in this project we uses FIGMA to create modern GUI to look better. Figma is a collaborative interface design tool that’s taking the design world by storm. Unlike Sketch, which runs as a standalone MacOS app, Figma is entirely browser-based, and therefore works not only on Macs, but also on PCs running Windows or Linux, and even on Chromebooks.

Already, digital voice interactions are driving the growth of new digital spaces like the metaverse and augmented reality. But good business decisions don’t just follow tech trends. New investments must pay off with benefits that drive quick returns. Text to speech does. To show how, here are just 10 of the dozens of TTS benefits, covering everything from brand affinity to corporate learning.

TEXT TO SPEECH

# SOFTWARE DEVELOPMENT LIFE CYCLE MODEL

The Software Development Life Cycle (SDLC) is a structured process that enables the production of high-quality, low-cost software, in the shortest possible production time. The goal of the SDLC is to produce superior software that meets and exceeds all customer expectations and demands. The SDLC defines and outlines a detailed plan with stages, or phases, that each encompasses their own process and deliverables. Adherence to the SDLC enhances development speed and minimizes project risks and costs associated with alternative methods of production

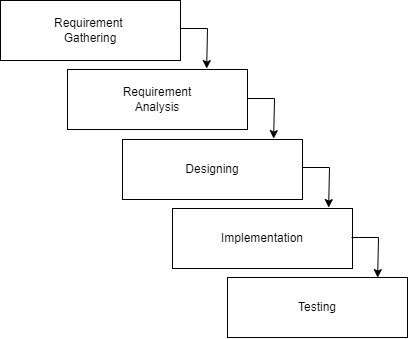
There are several types of SDLC models but in this project, we uses simplest SDLC model-

## Waterfall SDLC Model

Waterfall represents the oldest, simplest, and most structured methodology. Each phase depends on the outcome of the previous phase, and all phases run sequentially. This model provides discipline and gives a tangible output at the end of each phase. However, this model doesn’t work well when flexibility is a requirement. There is little room for change once a phase is deemed complete, as changes can affect the cost, delivery time, and quality of the software.

In this project 5- phases are taken are listed below-

1. Requirement gathering.
2. Requirement analysis.
3. Designing.
4. Implementation.
5. Testing.



The advantages of waterfall development are that it allows for departmentalization and control. A schedule can be set with deadlines for each stage of development and a product can proceed through the development process model phases one by one.

Development moves from concept, through design, implementation, testing, installation, troubleshooting, and ends up at operation and maintenance. Each phase of development proceeds in strict order.

Some of the major advantages of the Waterfall Model are as follows −

* Simple and easy to understand and use
* Easy to manage due to the rigidity of the model. Each phase has specific deliverables and a review process.
* Phases are processed and completed one at a time.
* Works well for smaller projects where requirements are very well understood.
* Clearly defined stages.
* Well understood milestones.
* Easy to arrange tasks.
* Process and results are well documented.

# REQUIREMENT GATHER

Requirements gathering are the process of understanding what you are trying to build and why you are building it. Requirements gathering are often regarded as a part of developing software applications or of cyber-physical systems like aircraft, spacecraft, and automobiles (where specifications cover both software and hardware).

## Introduction to Project:

The text-to-speech (TTS) assistive technology uses artificial intelligence to translate information written in a human-readable form in one language into audio, voice, or speech with a human accent.

Such systems turn text into audio or speech output using AI-driven algorithms as the input. It is also referred to as “real aloud technology” because it reads the text aloud.

This software project is windows and Linux based application that reads a text file to the user. The bot reads a text file and associated pronunciations in its temporary database. The bot then reads an entire word to the user. The pronunciations of articles and basic words have been fed to the bot, the rest of the words and complex ones are calculated and read accordingly.

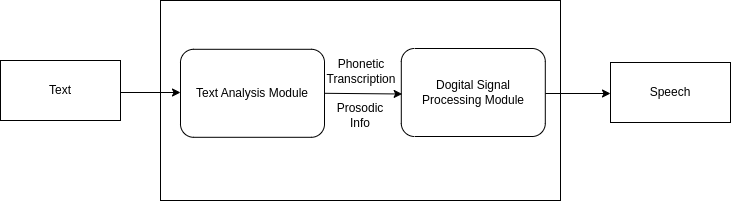
 The bot can be effectively used to help read the text document for the user so that the user does not constantly need to look at the screen and read the entire document. Test to speech converter is a recent software project that allows even the visually challenged to read and understand various documents.

Fig: process of text to speech conversion.

## Objective:

The aim of this tool is to convert the text into audio. It enables the user to read an e-book using this system. It is easy to use. Copy the text which is to be converted to audio and click on the play button. The user shall be able to listen to the audio. The user can also pause the audio at any moment of time. He can replay it. The user has an option to move the application to any position on the desktop. This eases the user to access the application.

A very helpful application which can ease the user to listen to the text rather than reading it continuously on the screen. This interface is developed using Visual basic. There are some APIs which are used to help in text to speech conversion. A very effective application which can be installed on any of the windows system. A very nice system which has a good natural quality of voice. It synthesizes the text and converts in to audio. This is very helpful for e reading of books. It is also great fun for kids. It can be used for educational purposes. Be it emails, documents in any formats and books you can use this application for all your requirements. A simple drag and drop interface which can be used on the desktops.

## Existing System:

There was no tool earlier to convert text to speech. It was difficult for users to continuously read the text on screen. Some users had the tasks of reading extensive content on the screen. There was no automation available to convert the text to speech. For people who are the area of content writing and proof reading, the task is difficult which requires them to read an extensive number of books online.

## Proposed System

This is a simple program which can be moved anywhere on desktop. This application synthesises the text to speech. There are no hassles of login and password. The user can install this application and can start using. It enables the user to copy text into clipboard and converts the selected text to audio. The audio which is played is very natural. The user can pause the audio at any time. The user can replay it back at the point he has stopped. The application can be moved anywhere on the desktop which eases the user to use the program. This application is developed using Visual Basic. This system uses some of the standard text to speech APIs.

# REQUIREMENT ANALYSIS

## Software requirement:

1. **Platform**: Windows 8, 8.1, 10, 11
2. **Processor:** Processor i3 or above with minimum 1.5GHz clock speed.
3. **Memory/space:** Hard disk/SSD 20GB minimum.
4. **RAM:** 4GB or more.

## Features:

* **Text input:** Users are provided with a text box where they can enter the required text in the software.
* **Speech rate:** Users can even alter the speech speed for application to read out text by choosing the appropriate rate provided by the software.
* **Voice change options:** Users can play text in two different voices male & female.
* **Play & download:** Users can play the texts and also can save the audio file of text in mp3, by default file name is “speech.mp3” of the output file

## Advantage:

* The system is helpful for persons having learning disabilities or visually challenged
* Prevents eye from strain, and user can sit and listen comfortably.
* Saves time especially while driving, exercising.
* Easy to use.
* Help improving spelling, reading, writing skills.
* Much helpful for influencer.
* Can embed on any python project.

## Disadvantage:

* Voice are not natural.
* Writing text is like messing with time.

## Applications:

* Can be used for visually and learning disabled people.
* Can be used in learning pronunciations.
* Can be used as reminders in offices, houses.
* Useful to make voice assistants on python.

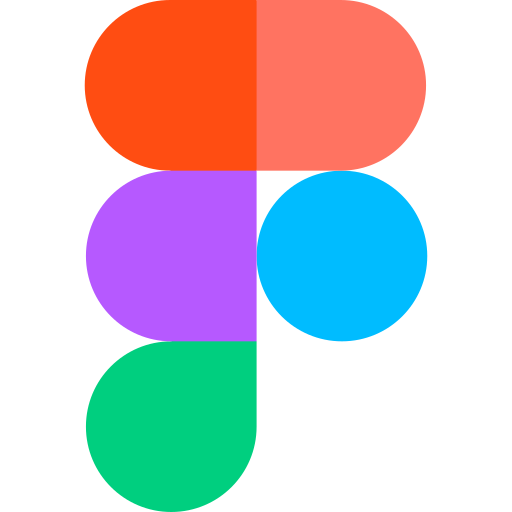
# DESIGNING OF PROJECT

The design phase is a stage where software developers define the technical details of the product. Depending on the project, these details can include screen designs, databases, sketches, system interfaces, and prototypes. Clients use these details to make final product design choices

To design this project we prepared the DFD and Use Case of project and then we selected the tools required to design the software product.

## GUI Designing tools:

### Figma:

Figma is a collaborative interface design tool that’s taking the design world by storm. Unlike Sketch, which runs as a standalone MacOS app, Figma is entirely browser-based, and therefore works not only on Macs, but also on PCs running Windows or Linux, and even on Chromebooks. It also offers a web API, and it’s free!

Another big advantage of Figma is that it allows real-time collaboration on the same file.

When using conventional “offline” apps like Sketch and Photoshop, if designers want to share their work, they typically have to export it to an image file, then send it via email or instant message.

In Figma, instead of exporting static images, we can simply share a link to the Figma file for clients and colleagues to open in their browser. This in itself saves significant time and inconvenience in a designer’s workflow. But more importantly, it means that clients and colleagues can interact more richly with the work, and review the latest version of the file.

### Set up Figma:

* Getting started in Figma is as simple as going to [www.figma.com](http://www.figma.com/), clicking “Sign up”, and entering your details. Once you’ve done that, Figma will open up with a start screen like this. Click on “New File” and we’ll get started!
* Now you are ready

### Proxlight Designer:

Proxlight Designer was created to speed up the GUI development process in Python. It uses the well-known design software Figma to make creating beautiful Tkinter GUIs in Python a piece of cake.

Proxlight Designer uses the Figma API to analyze a design file and create the respective code and files needed for the GUI. Even Proxlight Designer's GUI is created using Proxlight Designer.

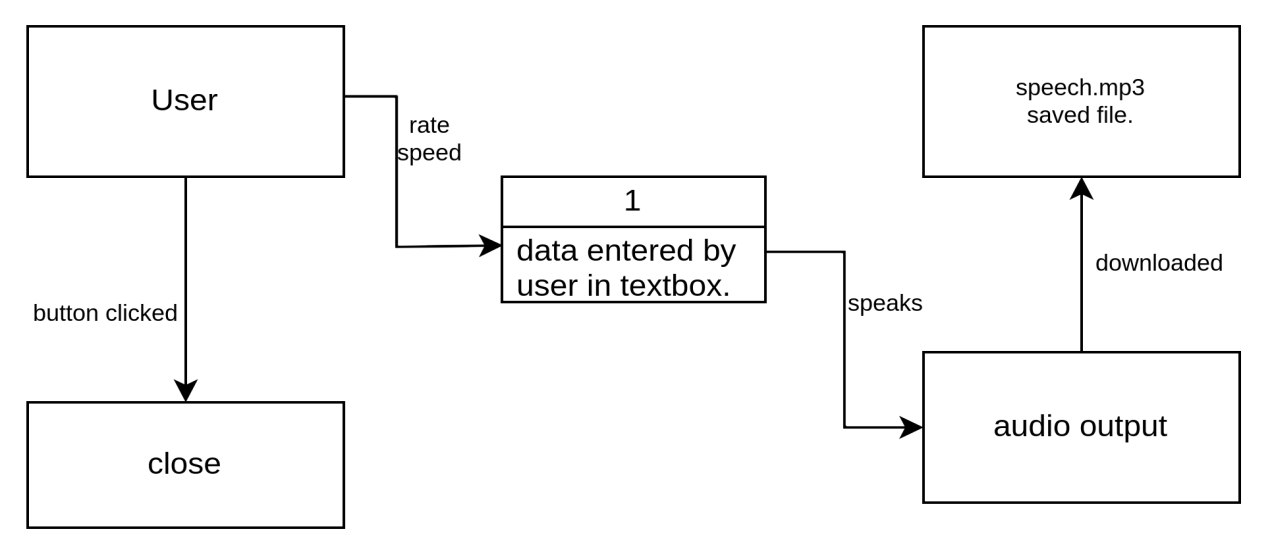
Download link: - https://proxlightapps.gumroad.com/l/Proxlight-Designer

Basically it is used to export the design file created in Figma-

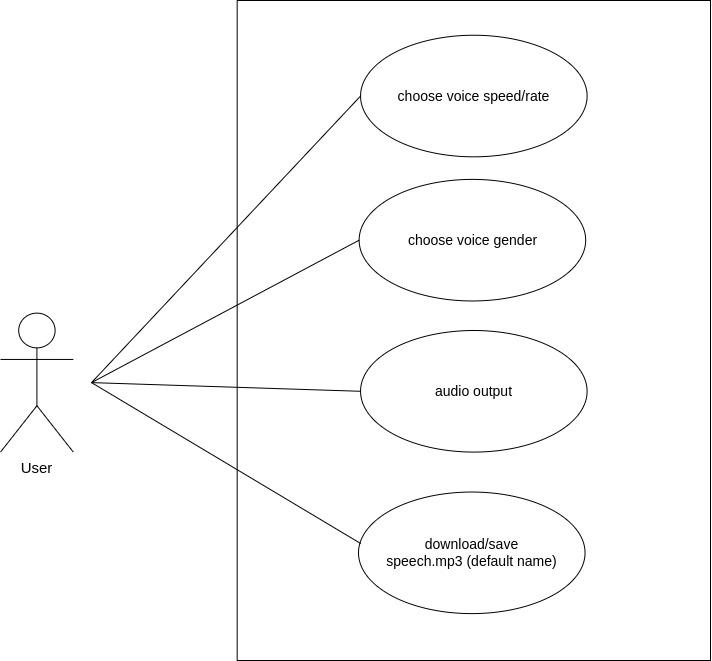
* copy figma design project link.
* Paste to file url field or proxlight.
* Go to figma setting, scrolldown to token section.
* Create token, paste to token field in proxlight.
* Select export path, hit enter.
* Now proxlight generates tkinter project from figma file.

## Functionality designing tools:

### Project Activity Diagram:

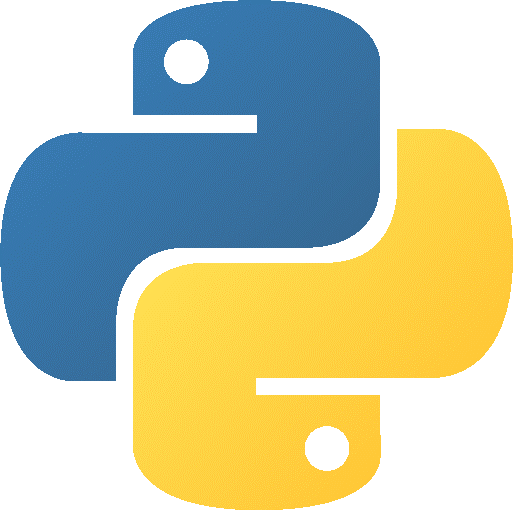


### Use Cases of the project:



## Programming tools:

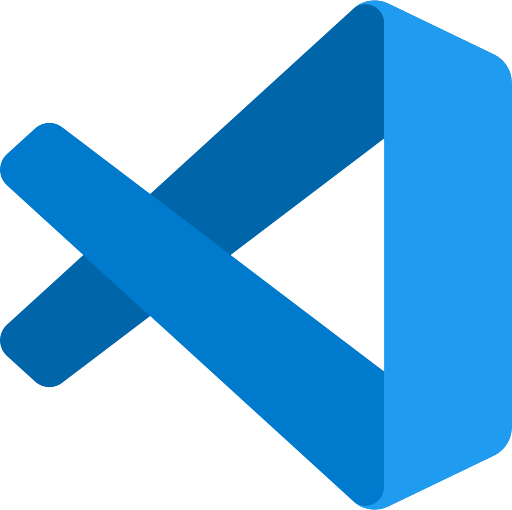
### Python programming language:

 Python is a popular programming language. It was created by Guido van Rossum, and released in 1991. Python is a widely used general-purpose, high level programming language. It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation. It was mainly developed for emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code. Python is a programming language that lets you work quickly and integrate systems more efficiently.

### Downloading and Installation of python:

* Download python software from <http://python.org/>
* Double-click the icon labelling the file python-3.7.4-amd64.exe.(name may be different).
* Ensure that the Install launcher for all users (recommended) and the Add Python 3.7 to PATH checkboxes at the bottom are checked.
* Highlight the Install Now (or Upgrade Now) message, and then click it.
* Click the Yes button. A new Python 3.7.4 (64-bit) Setup pop-up window will appear with a Setup Progress message and a progress bar.

### VSCODE:

Visual Studio Code (VS Code) is a source code editor developed by Microsoft that can run on Windows, macOS, and Linux. It is free, open-source, and is embedded with debugging tools, integrated terminals, built-in Git version control, code navigation, refactoring, and so on. In addition to the built-in features, VS Code is highly customizable as users can install extensions to add additional support for languages, themes, and debuggers – among others.

Although VS Code comes with built-in support for JavaScript, Typescript, and Node.js, it has a vibrant ecosystem of extensions for other languages including Python.

### ****Installing Visual Studio Code:****

* **Download Vscode from this linkhttps://code.visualstudio.com/download.**
* **double click on it to kick start the installation process. Once the installation is started, click next on the welcome screen.**
* **Accept the License Agreement: Go through the license agreement, Click on I accept the agreement and click Next**
* **Choose Installation Directory: By default, VS-Code is installed under C:\users\{username}\AppData\Local\Programs\Microsoft VS Code, but this default location of installation can be changed by clicking on Browse.**
* **Choose Additional Options: Various options to choose and click next.**
* **Create a desktop icon**
* **Add “Open with code” action to Windows Explorer file context menu: This would allow an option to open a file in VS Code on performing the right click on the file.**
* **Add Open with code action to Windows Explorer folder context menu: This would allow an option to open a folder in VS Code on performing the right click on the folder.**
* **Register Code as an editor for supported file types: File supported by VS Code will be automatically opened in Visual Studio Code editor.**
* **Add to PATH (available after restart): Visual Studio Code installed directory path will be added to an environment variable automatically but after the restart.**
* **Once you choose options that you need, click on Next to continue**
* **Select Launch VS Code and click on Finish.**

### Project Libraries:

### Tkinter:

Tkinter is the inbuilt python module that is used to create GUI applications. It is one of the most commonly used modules for creating GUI applications in Python as it is simple and easy to work with. You don’t need to worry about the installation of the Tkinter module separately as it comes with Python already. It gives an object-oriented interface to the Tk GUI toolkit. **Widgets** in Tkinter are the elements of GUI application which provides various controls (such as Labels, Buttons, ComboBoxes, CheckBoxes, MenuBars, RadioButtons and many more) to users to interact with the application.

To install tkinter(in case not found) open command prompt and type-

**pip install tkinter**

There are currently 15 types of widgets in Tkinter. We present these widgets as well as a brief description in the following table −

|  |  |
| --- | --- |
| Sr.No. | Operator & Description |
| 1 | [Button](https://www.tutorialspoint.com/python/tk_button.htm)  The Button widget is used to display buttons in your application. |
| 2 | [Canvas](https://www.tutorialspoint.com/python/tk_canvas.htm)  The Canvas widget is used to draw shapes, such as lines, ovals, polygons and rectangles, in your application. |
| 3 | [Checkbutton](https://www.tutorialspoint.com/python/tk_checkbutton.htm)  The Checkbutton widget is used to display a number of options as checkboxes. The user can select multiple options at a time. |
| 4 | [Entry](https://www.tutorialspoint.com/python/tk_entry.htm)  The Entry widget is used to display a single-line text field for accepting values from a user. |
| 5 | [Frame](https://www.tutorialspoint.com/python/tk_frame.htm)  The Frame widget is used as a container widget to organize other widgets. |
| 6 | [Label](https://www.tutorialspoint.com/python/tk_label.htm)  The Label widget is used to provide a single-line caption for other widgets. It can also contain images. |
| 7 | [Listbox](https://www.tutorialspoint.com/python/tk_listbox.htm)  The Listbox widget is used to provide a list of options to a user. |
| 8 | [Menubutton](https://www.tutorialspoint.com/python/tk_menubutton.htm)  The Menubutton widget is used to display menus in your application. |
| 9 | [Menu](https://www.tutorialspoint.com/python/tk_menu.htm)  The Menu widget is used to provide various commands to a user. These commands are contained inside Menubutton. |
| 10 | [Message](https://www.tutorialspoint.com/python/tk_message.htm)  The Message widget is used to display multiline text fields for accepting values from a user. |
| 11 | [Radiobutton](https://www.tutorialspoint.com/python/tk_radiobutton.htm)  The Radiobutton widget is used to display a number of options as radio buttons. The user can select only one option at a time. |
| 12 | [Scale](https://www.tutorialspoint.com/python/tk_scale.htm)  The Scale widget is used to provide a slider widget. |
| 13 | [Scrollbar](https://www.tutorialspoint.com/python/tk_scrollbar.htm)  The Scrollbar widget is used to add scrolling capability to various widgets, such as list boxes. |
| 14 | [Text](https://www.tutorialspoint.com/python/tk_text.htm)  The Text widget is used to display text in multiple lines. |
| 15 | [Toplevel](https://www.tutorialspoint.com/python/tk_toplevel.htm)  The Toplevel widget is used to provide a separate window container. |
| 16 | [Spinbox](https://www.tutorialspoint.com/python/tk_spinbox.htm)  The Spinbox widget is a variant of the standard Tkinter Entry widget, which can be used to select from a fixed number of values. |
| 17 | [PanedWindow](https://www.tutorialspoint.com/python/tk_panedwindow.htm)  A PanedWindow is a container widget that may contain any number of panes, arranged horizontally or vertically. |
| 18 | [LabelFrame](https://www.tutorialspoint.com/python/tk_labelframe.htm)  A labelframe is a simple container widget. Its primary purpose is to act as a spacer or container for complex window layouts. |
| 19 | [tkMessageBox](https://www.tutorialspoint.com/python/tk_messagebox.htm)  This module is used to display message boxes in your applications. |

### Geometry Management

All Tkinter widgets have access to specific geometry management methods, which have the purpose of organizing widgets throughout the parent widget area. Tkinter exposes the following geometry manager classes: pack, grid, and place.

* [The *pack()* Method](https://www.tutorialspoint.com/python/tk_pack.htm) − This geometry manager organizes widgets in blocks before placing them in the parent widget.
* [The *grid()* Method](https://www.tutorialspoint.com/python/tk_grid.htm) − This geometry manager organizes widgets in a table-like structure in the parent widget.
* [The *place()* Method](https://www.tutorialspoint.com/python/tk_place.htm) − This geometry manager organizes widgets by placing them in a specific position in the parent widget.

### Pyttsx3:

pyttsx3 is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline and is compatible with both Python 2 and 3. An application invokes the pyttsx3.init() factory function to get a reference to a pyttsx3. Engine instance. it is a very easy to use tool which converts the entered text into speech. The pyttsx3 module supports two voices first is female and the second is male which is provided by

“sapi5” for windows. It supports three TTS engines :

* *sapi5* – SAPI5 on Windows
* *nsss* – NSSpeechSynthesizer on Mac OS X
* *espeak* – eSpeak on every other platform

Installation To install the pyttsx3 module, first of all, you have to open the terminal and write

**pip install pyttsx3**

### Pillow:

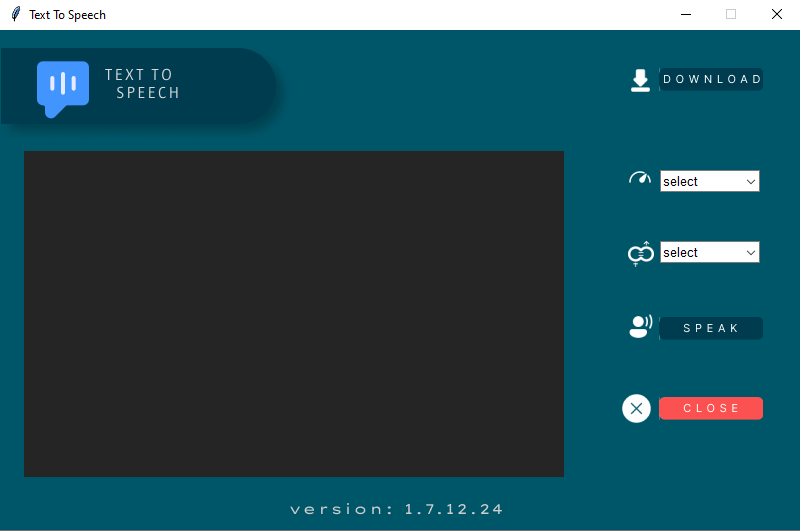
In the programming world, we can process digital images using various libraries or tools. In this article, we will learn about one of the popular tools of Python is Pillow. However, Python provides many other useful libraries such as [OpenCV](https://www.javatpoint.com/opencv), Python Image Library (PIL), and Scikit-image.

Installation To install the pillow module, first of all, you have to open the terminal and write.

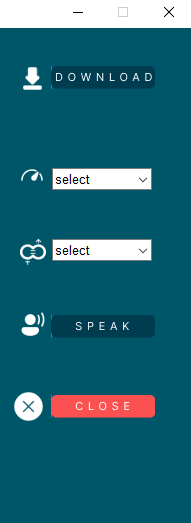
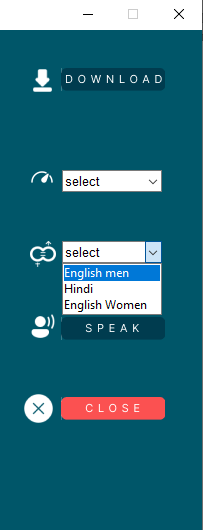
**pip install pillow**

## Project Snapshot:

### 1. Application Screenshot



### 2. Voice change and Voice rate multiple option screenshot.

### How to we add voices:-

Before adding we verify how many voices we have in our system by default.

import pyttsx3

engine = pyttsx3.init()

voices = engine.getProperty('voices')

for voice in voices:

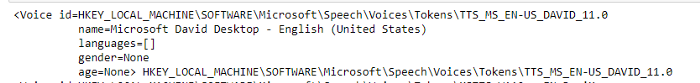
print(voice, voice.id)

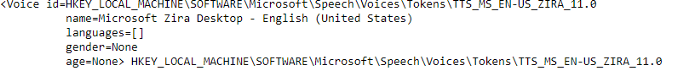
engine.setProperty('voice', voice.id)

engine.say("Hello World!")

engine.runAndWait()

engine.stop()





David and Zira are the two only voices installed in our system (MS windows).

These two are only capable of reading and speaking English characters.

For Hindi language we added Kalpana Assistant Language to read and speak Hindi characters.

Here the steps to add voices in windows OS:-

* Open windows narrator, Press window + Control + n.
* Scroll down and click on add more voices.
* Now click on add more voices.
* Search for Hindi Kalpana.
* Select and Kalpana and click on add button.
* After adding, press win + s, search for registry edit.
* Now browse the path=”HKEY\_LOCAL\_MACHINE/SOFTWARE/MICROSOFT/Speech\_OneCore/Voices/Tokens/
* Now select Kalpana language pack, right click and select export, save the export file desired to your location.
* Now navigate to exported file, right click on the file, and select edit.
* Now replace “Speech\_OneCore” to “Speech” in the file using find and replace notepad program.
* Save the file and double click on it and run the file, each time it asks to yes/no box, choose yes.
* Now Kalpana language pack is successful installed on the system.
* To verify we run the same script to check the installed assistant voice, now this time it will show three voices, DAVID, ZIRA, and KALPANA.
* DAVID on 0th index, ZIRA on 1st index, and KALPANA on 2nd index.



* We renamed DAVID as ENGLISH MEN, ZIRA as ENGLISH WOMEN, and KALPANA as HINDI.

## Project Components:

**Frame**: Frame is used to contain the elements of the user interface of the software.

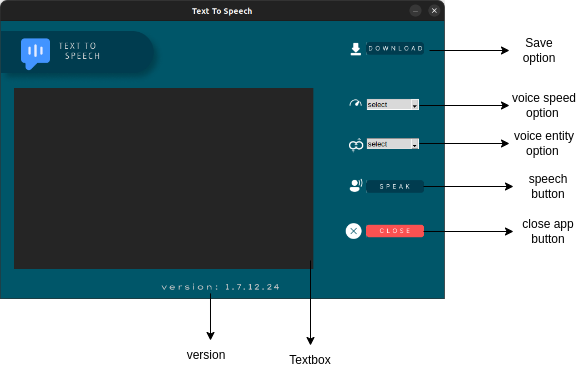
**Textbox**: A text box in required to get the text data of the user.

**Speak Button**: A button which convert the text into audio format and speaks using the operating system default voice encoder/decoder.

**Speed Button**: A button which is used to control the speech speed rate.

**Gender Button**: A button which allows choosing the voice entity either Male or Female, but by defaulting it is set to Male entity.

**Download Button**: A button which allows us to save the speech file into audio format, by default it is named speech.mp3.



# IMPLEMENTATION

#------importing modules----------

from tkinter import \*

import tkinter as tk

from tkinter import \*

import tkinter.ttk

from tkinter.ttk import Combobox

from PIL import \*

from tkinter import filedialog

from tkinter import messagebox

import os

#------creating main window----------

window = Tk()

window.geometry("800x500+200+120")

window.configure(bg = "#005669")

window.title("Text To Speech")

# ---------speak button definitions------------------

def speak():

import pyttsx3

# getting values fo fields.

textInput = textArea.get(1.0, "end-1c")

textGender = genderText.get()

textSpeed = speedText.get()

if textInput == "":

messagebox.showerror("Error","text field is empty !")

elif textGender == "select":

messagebox.showerror("Error","Please select voice gender !")

elif textSpeed == "select":

messagebox.showerror("Error","Please select voice speed !")

else:

# inittialize the pyttsx3 module

engine = pyttsx3.init()

# voice change script

voices = engine.getProperty('voices')

if textGender == 'Hindi':

engine.setProperty('voice', voices[1].id)

elif textGender == "English Women":

engine.setProperty('voice', voices[2].id)

# voice rate or speed script

speed = engine.getProperty('rate')

if textSpeed == 'Slow':

engine.setProperty('rate', 90)

elif textSpeed == 'Fast':

engine.setProperty('rate', 220)

else:

engine.setProperty('rate', 150)

# speak the string given in say().

engine.say(textInput)

engine.runAndWait()

'''for voice in voices:

print("Voice: %s" % voice.name)

print(" - ID: %s" % voice.id)

print(" - Languages: %s" % voice.languages)

print(" - Gender: %s" % voice.gender)

print(" - Age: %s" % voice.age)

print("\n")'''

# ---------save button definitions------------------

def save():

import pyttsx3

# getting values fo fields.

textInput = textArea.get(1.0, "end-1c")

textGender = genderText.get()

textSpeed = speedText.get()

if textInput == "":

messagebox.showerror("Error","text field is empty !")

elif textGender == "select":

messagebox.showerror("Error","Please select voice gender !")

elif textSpeed == "select":

messagebox.showerror("Error","Please select voice speed !")

else:

# inittialize the pyttsx3 module

engine = pyttsx3.init()

# voice change script

voices = engine.getProperty('voices')

if textGender == 'Hindi':

engine.setProperty('voice', voices[1].id)

elif textGender == "English Women":

engine.setProperty('voice', voices[2].id)

# voice rate or speed script

speed = engine.getProperty('rate')

if textSpeed == 'Slow':

engine.setProperty('rate', 90)

elif textSpeed == 'Fast':

engine.setProperty('rate', 220)

else:

engine.setProperty('rate', 150)

# speak the string given in say().

path = filedialog.askdirectory()

os.chdir(path)

engine.save\_to\_file(textInput, 'speech.mp3')

engine.runAndWait()

messagebox.showinfo("info","download successfull !")

# ---------close button definitions------------------

def close():

textInput = textArea.get(1.0, "end-1c")

if textInput != "":

ans = messagebox.askquestion("Confirm","Are you sure?")

if ans == "yes":

window.destroy()

else:

window.destroy()

# ------------field text variable---------------------

genderText = StringVar()

speedText = StringVar()

# ---------canvas on main window------------------

canvas = Canvas(

window,

bg = "#005669",

height = 500,

width = 800,

bd = 0,

highlightthickness = 0,

relief = "ridge"

)

canvas.place(x = 0, y = 0)

# ---------text area script------------------

entry0\_img = PhotoImage(file = f"img\_textBox0.png")

entry0\_bg = canvas.create\_image(

294.0, 283.5,

image = entry0\_img)

textArea = Text(window,

bd = 0, padx=20, pady=20,

bg = "#252525",

fg="white",

font=("Times New Roman", 14),

highlightthickness = 0)

textArea.configure(insertbackground='white')

textArea.place(

x = 24, y = 121,

width = 540,

height = 323)

# ---------gender box script------------------

gender = Combobox(window, values=['English men', 'Hindi','English Women'], font='arial 10',

state='r', width=11, height=40, textvariable=genderText)

gender.place(x=660, y=211)

gender.set('select')

# ---------speed box script------------------

speed = Combobox(window, values=['Slow', 'Normal', 'Fast'],

font='arial 10', state='r', width=11, height=40, textvariable=speedText)

speed.place(x=660, y=140)

speed.set('select')

# ---------background icon UI script------------------

background\_img = PhotoImage(file = f"background.png")

background = canvas.create\_image(

326.5, 253.0,

image=background\_img)

# ---------save button script------------------

img0 = PhotoImage(file = f"img0.png")

b0 = Button(

image = img0,

borderwidth = 0,

highlightthickness = 0,

command=save,

relief = "flat")

b0.place(

x = 659, y = 38,

width = 104,

height = 23)

# ---------speech script------------------

img1 = PhotoImage(file = f"img1.png")

b1 = Button(

image = img1,

borderwidth = 0,

highlightthickness = 0,

command=speak,

relief = "flat")

b1.place(

x = 659, y = 287,

width = 104,

height = 23)

# ---------close button script------------------

img2 = PhotoImage(file = f"img2.png")

b2 = Button(

image = img2,

borderwidth = 0,

highlightthickness = 0,

command=close,

relief = "flat")

b2.place(

x = 659, y = 367,

width = 104,

height = 23)

window.resizable(False, False)

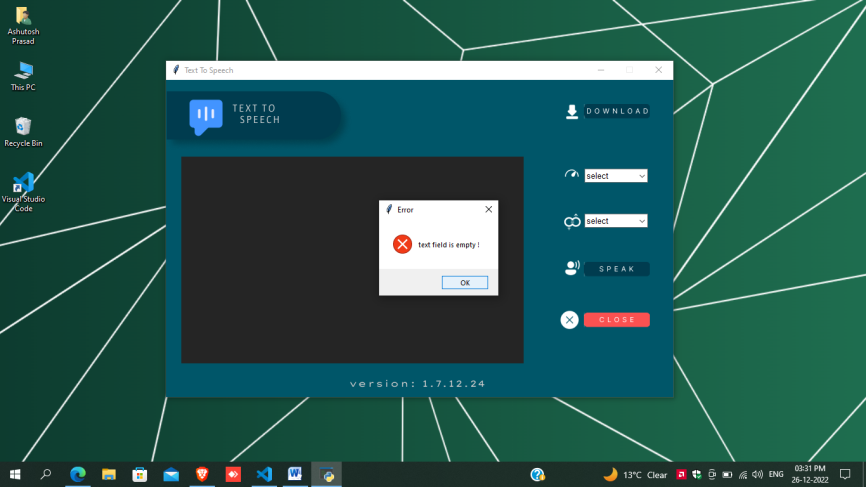
window.mainloop()

# TESTING

Testing is a group of techniques to determine the correctness of the application under the predefined script but, testing cannot find all the defect of application. The main intent of testing is to detect failures of the application so that failures can be discovered and corrected. It does not demonstrate that a product functions properly under all conditions but only that it is not working in some specific conditions.

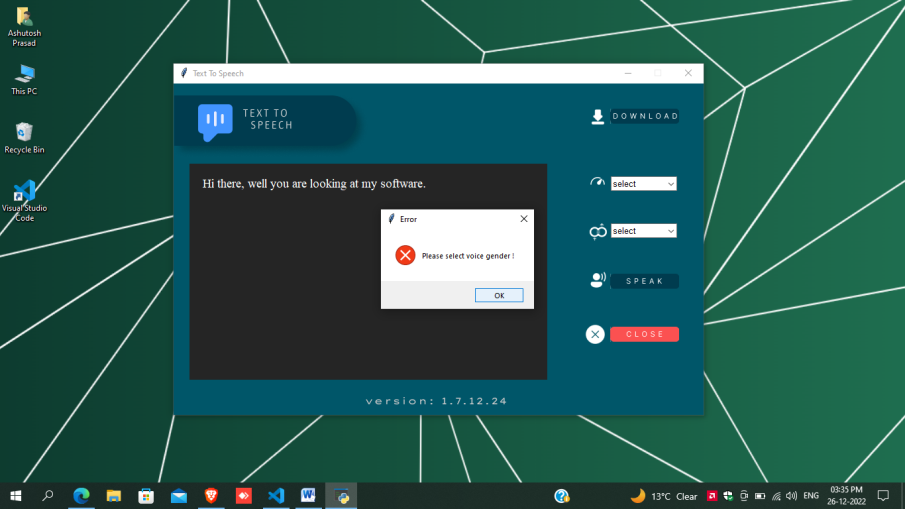
1. **What will happen when user does not enter any data and click on speak button?**

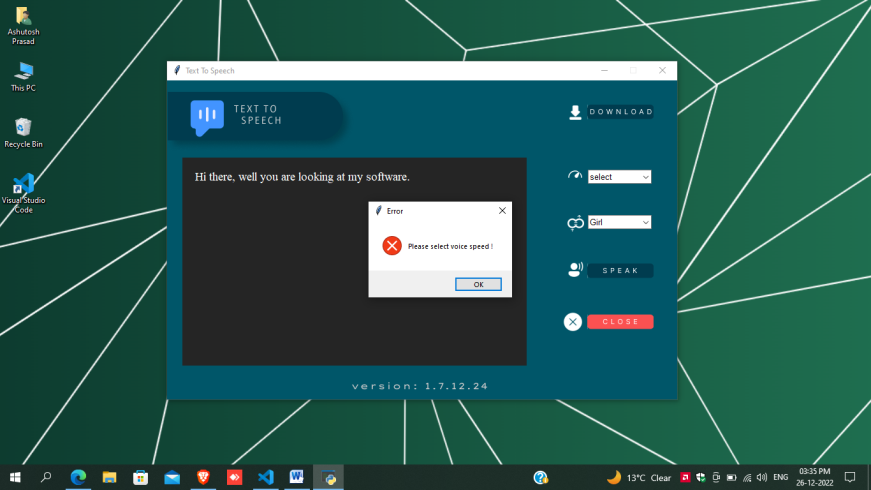
When a user enters nothing in text box then error handler of python showerror() gives error the *“text field is empty”.*



1. **What will happen when user does enter any data and click on speak button?**

User enters some data into text field and click on speak button, then there are error message appears, *“select voice gender”* and when user selects voice gender then again a error message appears*“select voice speed”.*





1. **What will happen when user does enter any data and click on speak button?**

When all requirement fulfilled then interpreter executes engine os TTS and voice speech played.

1. **What will happen when user does not enter any data and click on download button?**

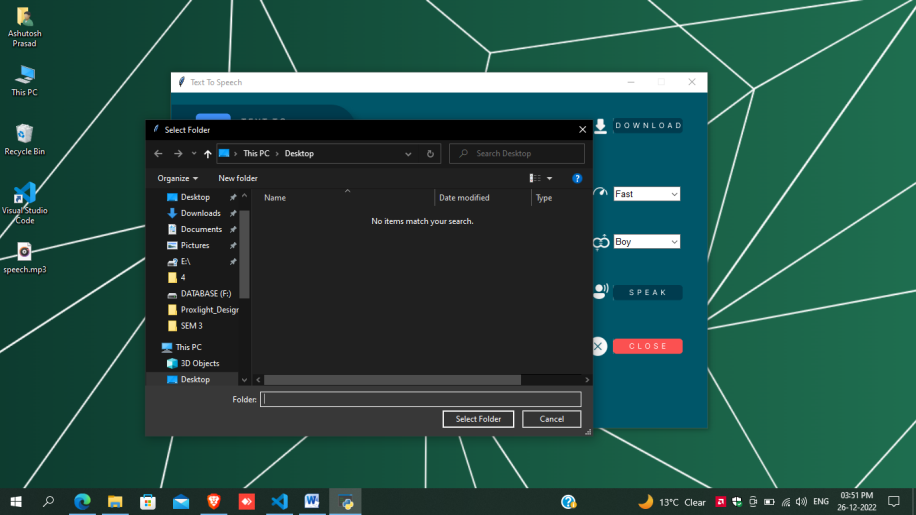
When a user enters nothing in text box then error handler of python showerror() gives error the *“text field is empty” and download nothing.*

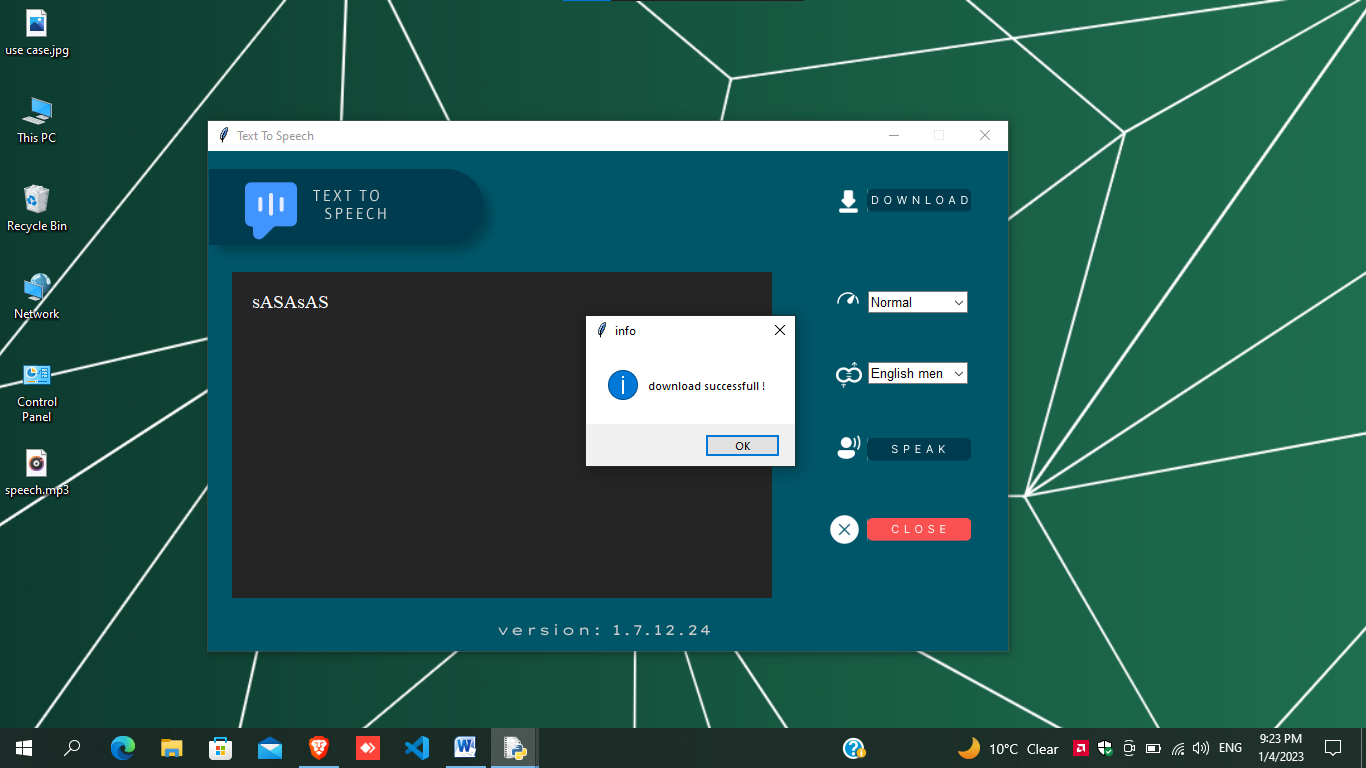
1. **What will happen when user does enter any data and click on speak button?**

User enters some data into text field and click on download button, then there are error message appears, *“select voice gender”* and when user selects voice gender then again a error message appears *“select voice speed”*

1. **What will happen when user every requirement and click on download button?**

What will happen when user every requirement and click on download button, then it will ask for where to save the file and an audio file name=**”speech.mp3”** will be downloaded on desired path or default file path =” C:\Users\ashut\Documents\speech.mp3”



# CONCLUSION

Understanding what is text to speech offers a variety of business benefits. Most people believe that this is no more than a piece of assistive tech, but it’s fast becoming a cornerstone of daily life.

Nearly every type of personal digital device now comes with this type of technology, and it’s never been simpler for businesses to incorporate it into their digital transformation strategies.

Be a business owner that’s on the cutting-edge of technology with WellSaid. Our AI-powered TTS solution is at the forefront of the industry and allows you to harness the full potential of TTS.

# SCOPE

Before we get into the near future, I should lay out how the current climate is.  Normal text to speech converters like [Note vibes](https://notevibes.com/) and [Natural tts](https://naturaltts.com/) are setting the stage for what is expected out of a converter.

Mixing text to speech converters and a machine learning how to speak and mimicking humans is going to lead to some next-level technology.  Just think about a robot being able to have a conversation with you someday.  If that does not freak you out, then you are crazy!

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

It would be a thing of the past to type out your message or paper, because we could just use our voice.  It does make sense in some regards, because we can probably speak much faster than typing in most cases.  But there are certain drawbacks that could hinder the expansion of this idea.

The only thing that we do right now is wait and see how the text to speech world is going to change.  Not only are the converters getting better with their methods, but the entire industry has the ability to make a lasting impact very soon.  Keep your eyes peeled to see which industries it will affect and how many lives it will impact.