"VOICE TO TEXT APPLICATION"

A Mini-Project report submitted in partial fulfillment of the requirements to complete the 3rd Year of

Bachelor of Technology

In

Computer Science &
Engineering

By

ANSH KUMAR
(2202920100028)

Under the supervision
of

DR. HIMANSHU SIROHI

MEERUT
INSTITUTE OF
TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
MEERUT INSTITUTE OF TECHNOLOGY,
MEERUT

Affiliated to

DR. A.P.J. ABDUL KALAM TECHNICAL
UNIVERSITY, UTTAR PRADESH, LUCKNOW

JANUARY 2025

CERTIFICATE

I hereby declare that the work which is being presented in the minor project report entitled, "***VOICE TO TEXT APPLICATION***", in partial fulfilment of the requirements to complete the **3rd year of Bachelor of Technology** submitted in **Computer Science and Engineering of Meerut Institute of Technology, Meerut,** is an authentic record of my own work carried out under the supervision of **Dr Himanshu Sirohi** and refers others works which are duly listed in the reference section.

The matter presented in this Project has not been submitted for the award of any other degree of this or any other university.

ANSH KUMAR

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Dr Himanshu Sirohi
Meerut Institute of Technology

TABLE OF CONTENTS

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the **B.Tech** Minor Project undertaken during **B.Tech. 3rd Year**. I owe special debt of gratitude to **Dr Himanshu Sirohi, Department of Computer Science & Engineering, Meerut Institute of Technology, Meerut** for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavors have seen light of the day. I also take the opportunity to acknowledge the contribution of **Prof. M I H Ansari, Head, Department of Computer Science & Engineering, Meerut Institute of Technology, Meerut** for his full support and assistance during the development of the project. I also do not like to miss the opportunity to acknowledge the contribution of all project cocoordinators and faculty members of the department for their kind assistance and cooperation during the development of our project. Last but not the least, I acknowledge our friends for their contribution in the completion of the project.

Signature:
Name: Ansh Kumar
Roll: 2202920100028
Date: 28-12-2024

ABSTRACT

This project presents the development of a Voice to Text system designed to accurately convert spoken language into written text. By utilizing Python and the SpeechRecognition library in combination with the Google Speech-to-Text API, the system delivers real-time transcription while addressing common challenges in speech recognition, such as background noise, varying accents, and latency issues.

The project employs robust data preprocessing techniques, including noise filtering, standardization, and segmentation, ensuring high-quality inputs for the transcription process. Advanced feature extraction methodologies are implemented to enhance accuracy and scalability, making the system suitable for small to medium-scale applications.

The Voice to Text system has practical applications in fields such as transcription services, accessibility tools, and productivity software, where efficiency and reliability are crucial. The successful implementation of this project highlights the potential for future enhancements, including the integration of deep learning models and multilingual support to further improve accuracy and usability

INTRODUCTION

Voice-to-text technology has revolutionized the way we interact with machines, enabling hands-free and accessible interfaces. Such systems are vital in fields like healthcare, customer service, and education.

1.1 Review of Literature

Existing tools like Google Speech-to-Text API, IBM Watson Speech Services, and Deep Speech employ advanced algorithms for accurate transcription. These systems face challenges like multi-speaker environments and noisy data.

1.2 Motivation

The motivation behind this project stems from the increasing demand for voice-activated interfaces. These systems enhance user productivity, enable accessibility for differently abled individuals, and streamline workflows.

1.3 Objective

The primary objective is to create a system that performs efficient and accurate transcription, addressing real-world challenges such as noisy environments and diverse accents.

METHODOLOGY

The development of the Voice to Text system followed a structured methodology:

2.1 Data Collection

Audio samples were sourced from datasets such as Common Voice, which provide diverse accents, languages, and environmental settings. Additional recordings were made to simulate real-world scenarios.

2.2 Preprocessing

Data preprocessing involved:

- Noise reduction using audio filters.
- Standardizing audio formats to WAV, 16-bit mono.
- Segmenting long recordings into smaller chunks.

2.3 Feature Extraction

Key features were extracted using SpeechRecognition, identifying phonetic patterns and frequencies crucial for transcription.

2.4 Implementation

Python libraries like SpeechRecognition and PyDub, alongside the Google Speech-to-Text API, were employed for speech recognition and processing.

2.5 Testing and Evaluation

The system was evaluated on metrics such as Word Error Rate (WER), processing speed, and accuracy across different environments.

DISCUSSION

1. Challenges:

a. Background Noise: External sounds significantly impacted the accuracy of transcription, especially in uncontrolled environments.

b. Accents and Dialects: The system struggled with diverse pronunciations, leading to lower performance across varying accents.

c. Latency: Achieving real-time processing required extensive optimization to reduce delays and improve user experience.

2. Advantages:

a. Ease of Integration: The system can be seamlessly incorporated into different platforms like accessibility tools or transcription services.

b. Scalability: It demonstrated the ability to process large datasets without major performance bottlenecks.

c. User-Friendliness: The system features a simple interface, making it accessible even to non-technical users.

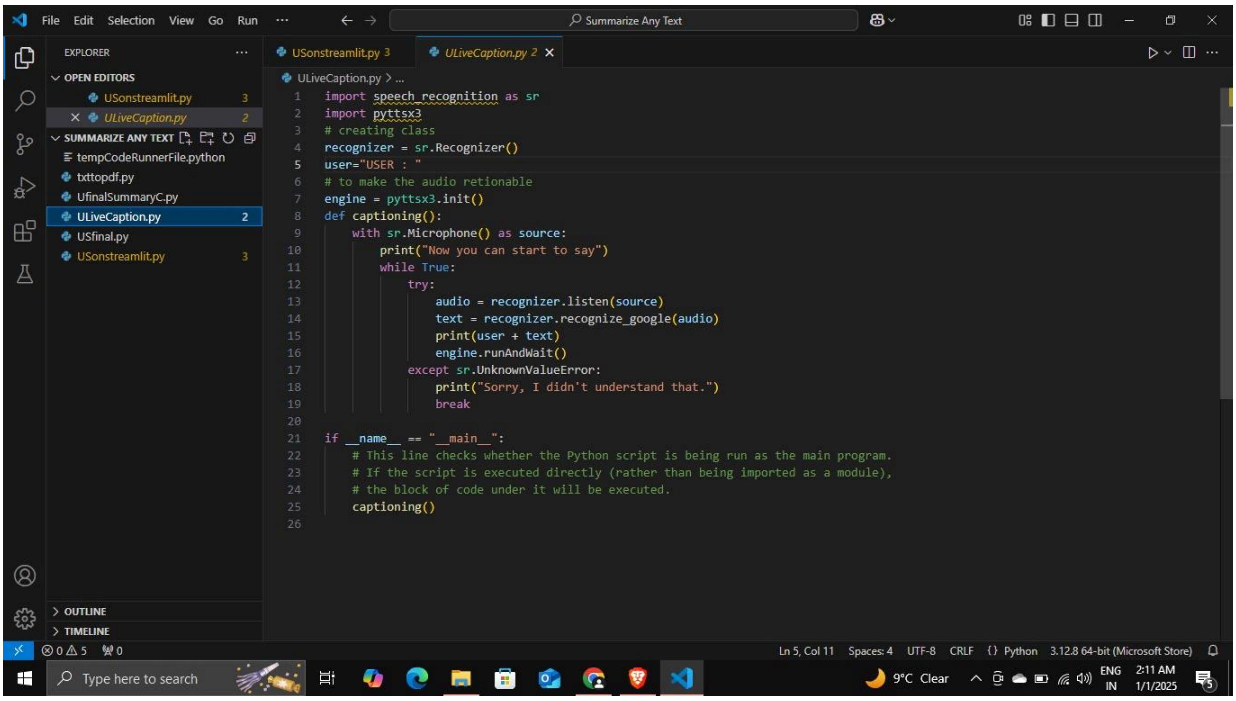
3. Future Scope:

a. Implementing advanced neural network models for enhanced contextual understanding and better accuracy in noisy or complex environments.

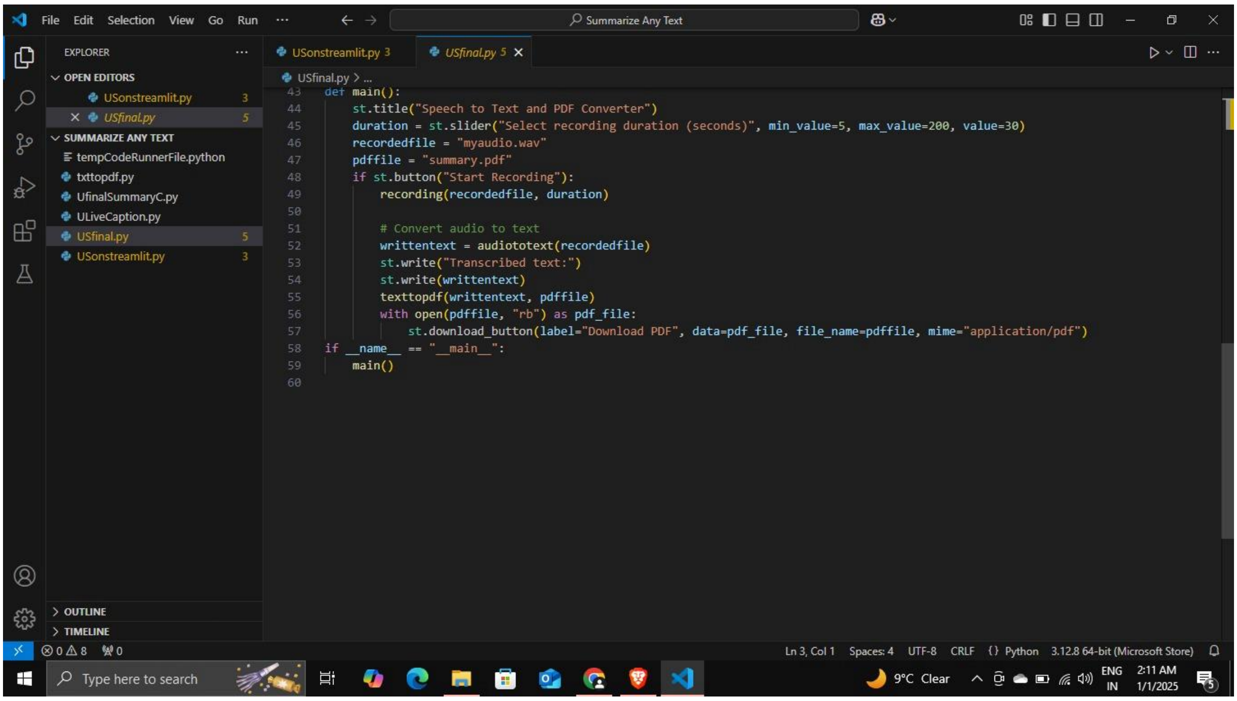
b. Expanding support to include additional languages and offline processing capabilities, making it more versatile and widely usable.

c. Exploring real-time multi-speaker transcription for broader applications in meetings or collaborative settings.

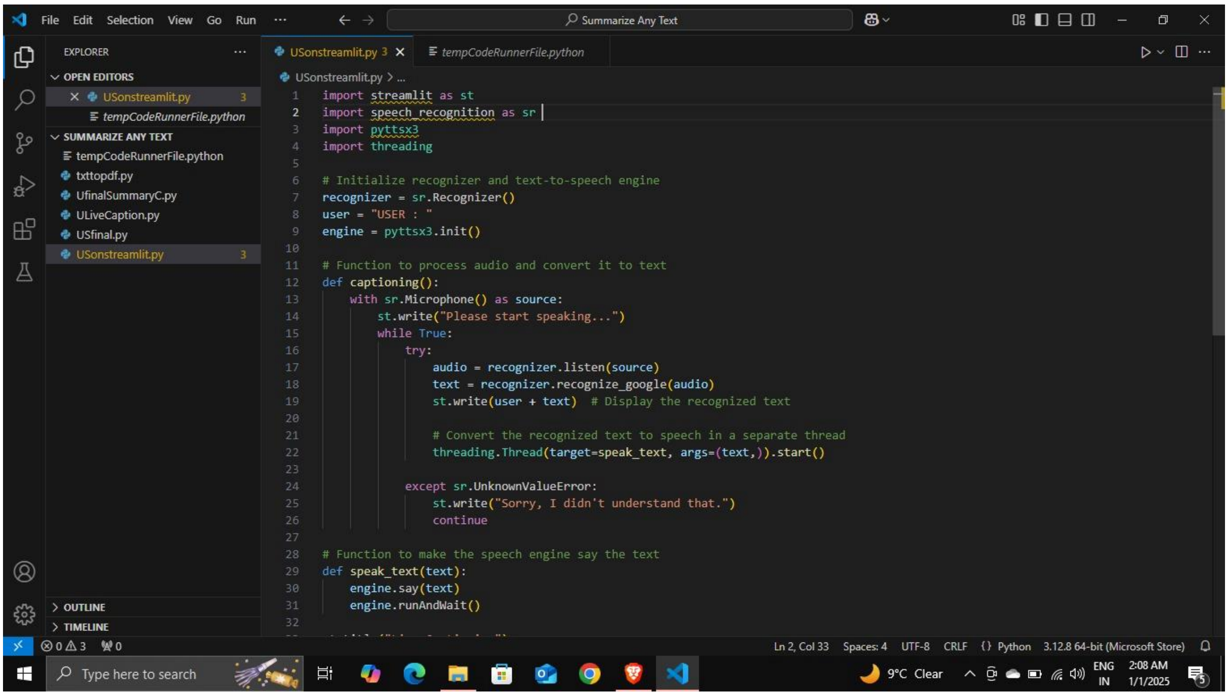
SNAPSHOTS OF CODE



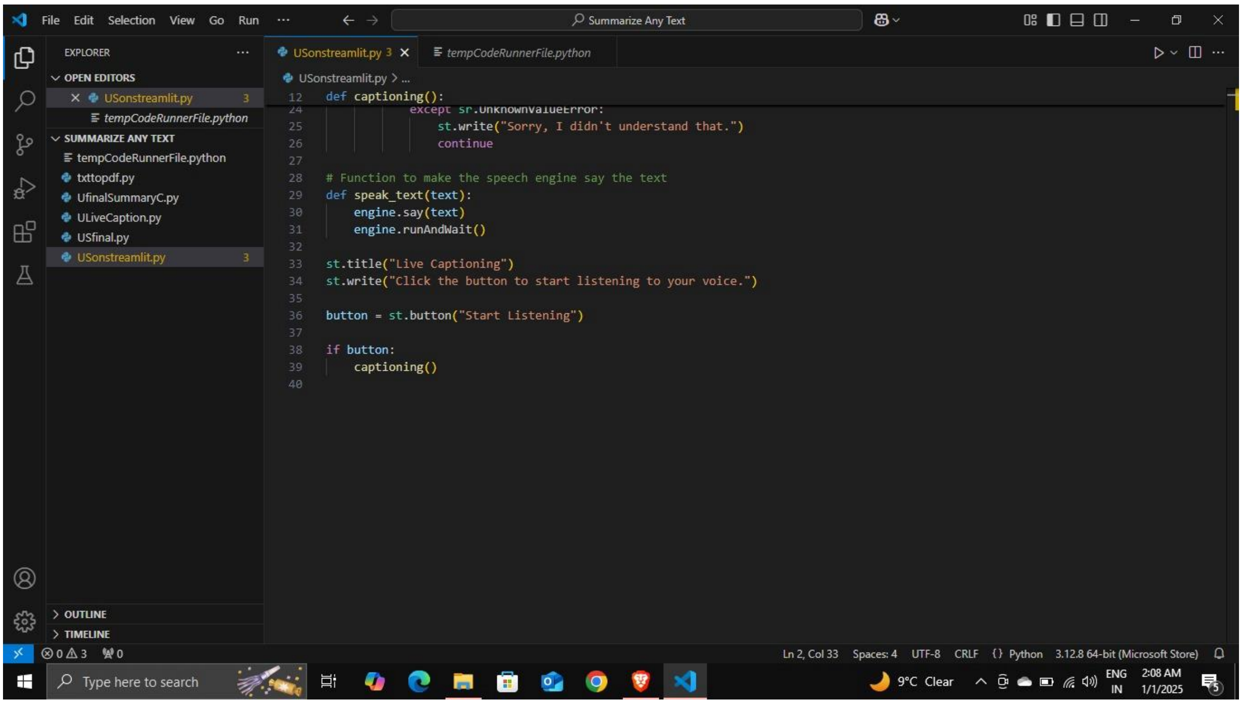
```python
import speech_recognition as sr
import pyttsx3
# creating class
recognizer = sr.Recognizer()
user="USER : "
# to make the audio retionable
engine = pyttsx3.init()
def captioning():
    with sr.Microphone() as source:
        print("Now you can start to say")
        while True:
            try:
                audio = recognizer.listen(source)
                text = recognizer.recognize_google(audio)
                print(user + text)
                engine.runAndWait()
            except sr.UnknownValueError:
                print("Sorry, I didn't understand that.")
                break

if __name__ == "__main__":
    # This line checks whether the Python script is being run as the main program.
    # If the script is executed directly (rather than being imported as a module),
    # the block of code under it will be executed.
    captioning()
```



```python
def main():
    st.title("Speech to Text and PDF Converter")
    duration = st.slider("Select recording duration (seconds)", min_value=5, max_value=200, value=30)
    recordedfile = "myaudio.wav"
    pdffile = "summary.pdf"
    if st.button("Start Recording"):
        recording(recordedfile, duration)

        # Convert audio to text
        writtentext = audiototext(recordedfile)
        st.write("Transcribed text:")
        st.write(writtentext)
        texttopdf(writtentext, pdffile)
        with open(pdffile, "rb") as pdf_file:
            st.download_button(label="Download PDF", data=pdf_file, file_name=pdffile, mime="application/pdf")
if __name__ == "__main__":
    main()
```



```python
import streamlit as st
import speech_recognition as sr
import pyttsx3
import threading

# Initialize recognizer and text-to-speech engine
recognizer = sr.Recognizer()
user = "USER : "
engine = pyttsx3.init()

# Function to process audio and convert it to text
def captioning():
    with sr.Microphone() as source:
        st.write("Please start speaking...")
        while True:
            try:
                audio = recognizer.listen(source)
                text = recognizer.recognize_google(audio)
                st.write(user + text)  # Display the recognized text

                # Convert the recognized text to speech in a separate thread
                threading.Thread(target=speak_text, args=(text,)).start()

            except sr.UnknownValueError:
                st.write("Sorry, I didn't understand that.")
                continue

# Function to make the speech engine say the text
def speak_text(text):
    engine.say(text)
    engine.runAndWait()

```



```python
def captioning():
            except sr.UnknownValueError:
                st.write("Sorry, I didn't understand that.")
                continue

# Function to make the speech engine say the text
def speak_text(text):
    engine.say(text)
    engine.runAndWait()

st.title("Live Captioning")
st.write("Click the button to start listening to your voice.")

button = st.button("Start Listening")

if button:
    captioning()

```

REFERENCES

1. Books and Documentation:

Python Speech Recognition Library Documentation: https://pypi.org/project/SpeechRecognition/

Google Speech-to-Text API Documentation: https://cloud.google.com/speech-to-text/docs

PyDub Library Documentation: https://pydub.com/

3. Research Papers:

Amodei, D., Ananthanarayanan, S., Anubhai, R., et al. (2016). "Deep Speech 2: End-to-End Speech Recognition in English and Mandarin." Proceedings of the 33rd International Conference on Machine Learning.

Hinton, G., Deng, L., Yu, D., et al. (2012). "Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups." IEEE Signal Processing Magazine.

4. Articles and Tutorials:

"Speech Recognition with Python" by Real Python: https://realpython.com/python-speech-recognition/

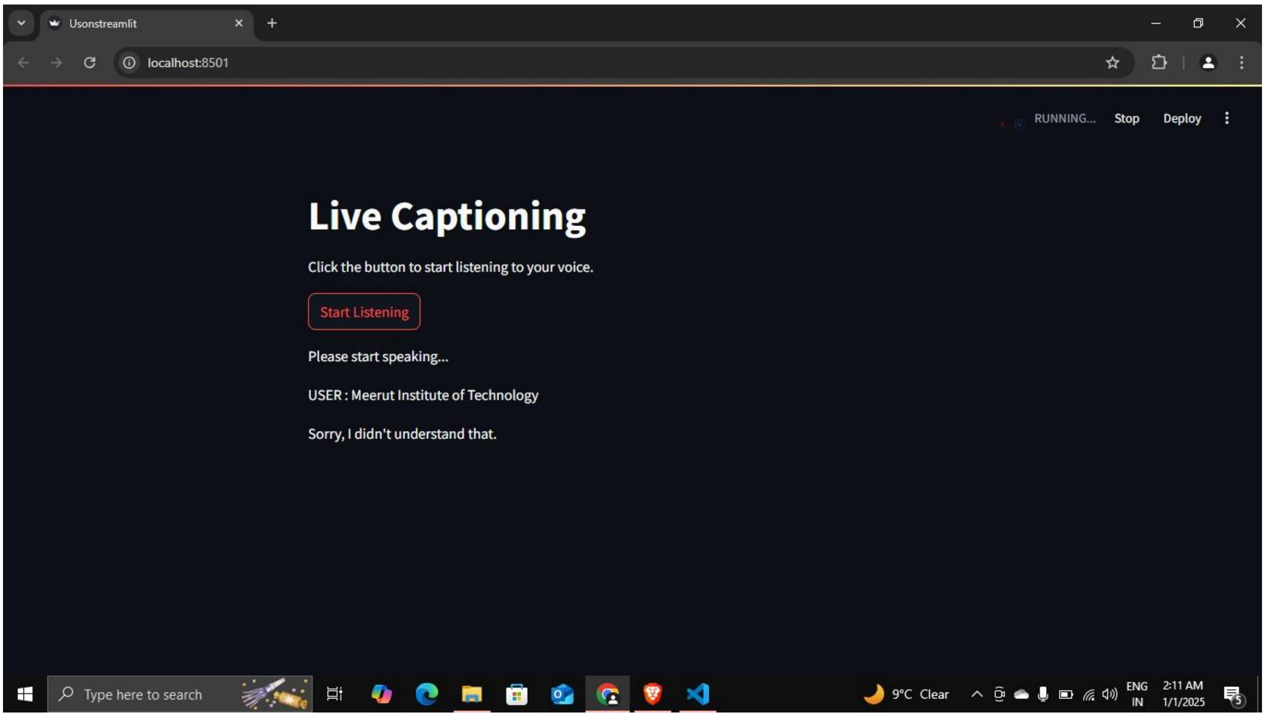
"Understanding the Google Speech-to-Text API" by Towards Data Science: https://towardsdatascience.com/google-speech-to-text-api-implementing-and-using-dcfd5ed71826

5. Tools and Libraries:

NumPy: https://numpy.org/

LibROSA: https://librosa.org/

OUTPUT



Live Captioning

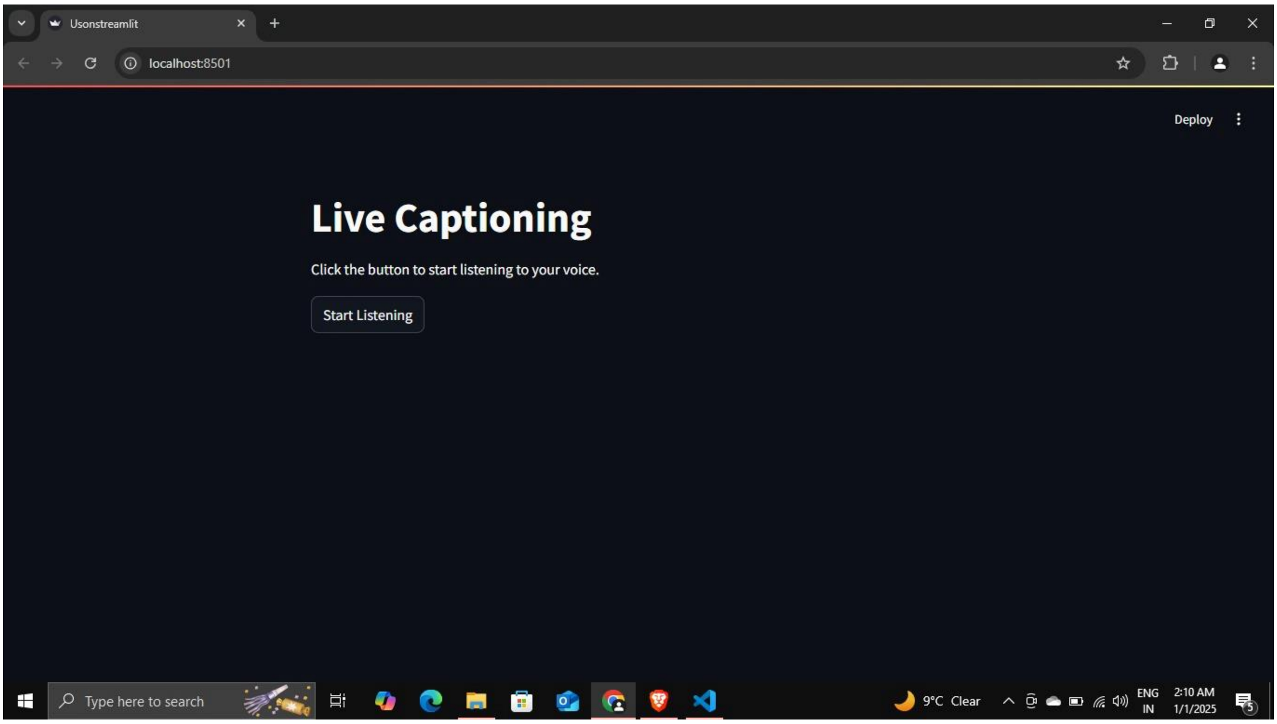
Click the button to start listening to your voice.

Start Listening

Please start speaking...

USER : Meerut Institute of Technology

Sorry, I didn't understand that.



Live Captioning

Click the button to start listening to your voice.

Start Listening