A

Project Report

on

EMOTION-PRESERVING MULTILINGUAL SPEECH-TO-SPEECH TRANSLATION SYSTEM

A Project Report (Stage-I) submitted in partial fulfilment of the Requirements for the

Award of the Degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

B NAMRATHA	**22VD1A0527**
M HARSHITHA	**22VD1A0514**
B GANESH	**22VD1A0512**
M VIVEK	**22VD1A0566**
A ABHIRAM	**22VD1A0501**

Under the Guidance of

Mr. N. VAMSHI KRISHNA

Assistant Professor (c)

Department of Computer Science and Engineering.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

UNIVERSITY COLLEGE OF ENGINEERING MANTHANI

Pannur (Vil), Ramagiri (Mdl), Peddapally-505212, Telangana (India).

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

UNIVERSITY COLLEGE OF ENGINEERING MANTHANI

Pannur (Vil), Ramagiri (Mdl), Peddapally-505212, Telangana (India).

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION BY THE CANDIDATE

We,

B NAMRATHA	**22VD1A0527**
M HARSHITHA	**22VD1A0514**
B GANESH	**22VD1A0512**
M VIVEK	**22VD1A0566**
A ABHIRAM	**22VD1A0501**

hereby declare that the project report stage entitled ***EMOTION-PRESERVING MULTILINGUAL SPEECH-TO-SPEECH TRANSLATION SYSTEM*** under the guidance of **Mr. N.VAMSHI KRISHNA,** Assistant Professor (c), Department of Computer Science and Engineering, JNTUH University College of Engineering Manthani submitted in partial fulfilment for the award of the Degree of Bachelor of Technology in Computer Science and Engineering.

This is a record of bonafide work carried out by us and the results embodied in this project report have not been reproduced or copied from any source. The results embodied in this project have not been submitted to any other University or Institute for the award of any degree .

B NAMRATHA	**22VD1A0527**
M HARSHITHA	**22VD1A0514**
B GANESH	**22VD1A0512**
M VIVEK	**22VD1A0566**
A ABHIRAM	**22VD1A0501**

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

UNIVERSITY COLLEGE OF ENGINEERING MANTHANI

Pannur (Vil), Ramagiri (Mdl), Peddapally-505212, Telangana (India).

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project stage report entitled ***EMOTION-PRESERVING MULTILINGUAL SPEECH-TO-SPEECH TRANSLATION SYSTEM*** a bonafide work carried out by

BAJJURI NAMRATHA	**22VD1A0527**
MARKA HARSHITHA	**22VD1A0514**
BAVANDLAPALLY GANESH	**22VD1A0512**
MARRI VIVEK	**22VD1A0566**
AKULA ABHIRAM	**22VA1A0501**

in partial fulfilment of the requirements for the degree of **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE AND ENGINEERING** by the **Jawaharlal Nehru Technological University Hyderabad University College of Engineering Manthani** during the academic year 2025-26.

The results of investigation enclosed in this report have been verified and found satisfactory. The results embodied in this project report have not been submitted to any other University or Institute for the award of any degree.

PROJECT GUIDE

Mr. N.VAMSHI KRISHNA, Assistant Professor(c)

HEAD OF THE DEPARTMENT

EXTERNAL EXAMINER

DATE:

Project Report(Stage-I)

ACKNOWLEDGMENT

We feel honoured and privileged to place our warm salutation to our **Principal Dr. B. VISHNU VARDHAN, Senior Professor of Computer Science and Engineering, JNTUH University College of Engineering Manthani,** who gave us the opportunity to have experience in engineering and profound technical knowledge.

We feel honoured and privileged to place our warm salutation to our **Vice Principal Mr. M. UDAY KUMAR, Associate Professor of Computer Science and Engineering, JNTUH University College of Engineering Manthani,** who gave us the opportunity to have experience in engineering and profound technical knowledge.

We pay our deep sense of gratitude to our project Guide **Mr. N.VAMSHI KRISHNA , Assistant Professor(c) of Computer Science and Engineering, JNTUH University College of Engineering Manthani,** to encourage us to the highest peak and to provide us the opportunity to prepare the project. We are extremely grateful to his valuable suggestions and unflinching cooperation throughout project work.

We are immensely obliged to other faculty members of Department of CSE for their kind co-operation.

We wish to convey our thanks to one and all those who have extended their helping hands directly and indirectly in completion of our project.

Last, but not least, our parents are also an important inspiration for us. So, with due regards, we express our gratitude to them.

B NAMRATHA	**22VD1A0527**
M HARSHITHA	**22VD1A0514**
B GANESH	**22VD1A0512**
M VIVEK	**22VD1A0566**
A ABHIRAM	**22VA1A0501**

ABSTRACT

In today's interconnected world, language barriers, regional dialects, and emotional nuances often limit effective speech communication. Traditional translation systems, though capable of accurate transcription and multilingual conversion, usually neglect emotional tone, speaker intent, and cultural context. This often results in technically correct but socially inadequate communication, which poses challenges in areas like international collaboration, education, healthcare, customer support, and tourism.

Existing approaches, including tools such as Google Translate, Azure Speech Translation, and advanced pipelines like Whisper or Coqui-TTS, focus mainly on linguistic accuracy. However, they fail to preserve critical aspects like emotion, dialect-specific expressions, and context sensitive adaptation, leading to loss of meaning and empathy in realworld conversations.

The project proposes an Emotion-Preserving Multilingual Speech-to-Speech Translation System (EPMSSTS) that integrates dialect detection, multimodal emotion recognition, and context-aware NLP into the translation pipeline. The system captures emotions from both audio prosody and textual cues, ensuring the translated output maintains both linguistic accuracy and emotional fidelity. The architecture includes preprocessing of speech input, feature extraction for emotion and dialect analysis, and a fusion-based translation model that generates natural, emotion-preserved speech in the target language.

The expected outcome is a scalable and reliable translation system capable of delivering human-like, empathetic communication across multiple languages and dialects. The system is designed to operate on commodity hardware while offering a production-ready framework for future enhancements in inclusive global communication.

Keywords: Speech Translation, Emotion Preservation, Dialect Detection, Multilingual, Communication, Context-Aware NLP, Prosody Analysis, Human-Centric AI

TABLE OF CONTENTS

LIST OF FIGURES

INTRODUCTION

1. INTRODUCTION

Speaking is the most natural way people communicate, but it quickly becomes difficult when two individuals don't share the same language, dialect, or cultural background. Even when words can be translated, the real meaning often lies in how something is said—whether the person sounds happy, worried, frustrated, excited, or calm. Today's translation tools mostly focus on getting the words right, but they ignore the feelings and tone behind those words. Because of this, conversations can lose their warmth, empathy, and intention. In situations like a patient explaining symptoms, a student asking for help, a traveler seeking directions, or a customer expressing dissatisfaction, losing emotional meaning can affect understanding and trust.

Most existing speech translation systems work in a simple sequence—convert speech to text, translate the text, and then speak the translation aloud. While this works for basic communication, the output often sounds robotic and emotionless. It also doesn't reflect regional accents or dialects, which are a big part of how people naturally speak. Someone from a village, a city, or a different region may use completely different expressions for the same message, and current translation tools fail to recognize or adapt to these differences.

The Emotion-Preserving Multilingual Speech-to-Speech Translation System (EPMSSTS) aims to solve this problem by translating speech in a way that keeps not just the meaning of the words, but also the emotion, tone, and dialect of the speaker. The system listens to a person's voice, identifies emotional cues like pitch and stress, detects dialect patterns, and then produces translated speech that sounds expressive and natural in another language. This makes conversations feel more human, respectful, and relatable.

One of the biggest advantages of this system is accessibility. People who cannot read, write, or use text-based apps can still communicate effortlessly because everything happens through speech. The user simply talks in their own language and dialect, and the system handles the rest. EPMSSTS is designed to work on regular hardware, making it practical for real-world use and future deployment. By combining emotion recognition, dialect understanding, and natural-sounding translation, this project takes a step toward communication that feels more like talking to a real person rather than a machine.

1.1 PROBLEM STATEMENT

People rely on more than just words when they speak — their emotions, tone, and way of expressing themselves play a major role in how their message is understood. However, most existing speech translation systems only focus on converting spoken words from one language to another, without paying attention to how those words are said. As a result, important emotional cues such as urgency, frustration, happiness, confusion, or concern are lost during translation. This can lead to misunderstandings, lack of empathy, and communication that feels robotic or insensitive.

Another major issue is that people speak in different dialects and regional variations, which affect pronunciation, vocabulary, and expression. Current translation tools do not recognize or adapt to these dialect differences, making translation less accurate and sometimes confusing. For people who are uneducated, visually challenged, or unable to read text-based translations, current systems are even less helpful because they depend heavily on written input or output.

Therefore, there is a need for a speech-to-speech translation system that can understand and translate spoken language while preserving emotional tone, speaker intent, and dialectal variations. The problem this project addresses is the absence of a natural, expressive, and context-aware multilingual translation system that allows people to communicate smoothly across languages without losing the emotional meaning and human qualities of speech..

LITERATURE SURVEY

2. LITERATURE SURVEY

A literature survey is a comprehensive review of existing knowledge, research, and developments related to a specific topic or problem. It involves analyzing and synthesizing information from scholarly articles, books, conference papers, and other credible sources to:

- Understand the State of the art
- Identify Research Gaps
- Justify the Research

2.1 INTRODUCTION TO EMOTION AWARE SPEECH TRANSLATION:

Communication becomes challenging when people do not share the same language, dialect, or cultural background. Most translation tools focus only on converting words and fail to capture emotions, tone, and natural speaking style. As a result, the translated output often sounds robotic and loses the warmth, intention, and clarity needed for real human interaction.

The Emotion-Preserving Multilingual Speech-to-Speech Translation System (EPMSSTS) addresses this by recognizing emotional cues, understanding regional dialects, and producing expressive, natural-sounding translated speech. This makes communication more human-like and accessible, especially for people who rely entirely on voice. By combining emotion detection, dialect recognition, and natural speech synthesis, the system helps create meaningful and inclusive multilingual conversations..

2.2 MOTIVATION FOR NATURAL MULTILINGUAL COMMUNICATION:

Human communication is not just about exchanging words—it is about expressing feelings, intentions, and attitudes. When people from different languages or regions try to communicate, language barriers often cause confusion, misunderstanding, and emotional disconnect. Existing translation tools can convert speech into another language, but they sound flat and emotionless, making conversations feel unnatural and sometimes insensitive. This becomes even more challenging for individuals who cannot read or write, or who speak in strong regional dialects.

The motivation behind this project is to create a translation system that feels more human and relatable. By preserving emotional tone and recognizing dialect variations, the system can help people communicate more accurately and comfortably in real-life situations such as health consultations, education support, emergency communication, travel assistance, and customer

interaction. The idea is to make technology adapt to human communication, rather than forcing people to adapt to technology. This project aims to reduce social and linguistic barriers, promote inclusiveness, and enable meaningful multilingual conversations for everyone.

2.3 HISTORY OF SPEECH-TO-SPEECH TRANSLATION SYSTEMS:

Agricultural market systems have transformed significantly over time. The following five stages clearly explain the evolution in a simple and well-structured manner:

2.3.1 Early Text Based Translation(Before 2000):

- Translation required manual typing
- Focus was only on word meaning
- No speech input or expressive output
- Emotions and tone were ignored

2.3.2 Online Translation And Speech Input Era(2000-2015):

- Introduction of tools like Google Translate and Bing Translator
- Speech input became possible
- Output was still robotic and monotone
- Dialects and emotional cues were not considered

2.3.3 Neural Speech Recognition and Voice Synthesis (2015–2020):

- Emergence of systems like Whisper, deep learning ASR, TTS models
- Better accuracy and pronunciation
- Still lacked emotion retention and dialect understanding
- Speech felt artificial and disconnected

2.3.4 Emotion-Aware and Context-Sensitive Speech Translation (2020–Present)

- Growing research in prosody analysis and expressive synthesis
- Emotion recognition models developed alongside speech translation
- Multilingual and dialect-aware embeddings introduced
- Real-time translation with voice preservation became possible

EPMSSTS falls into this fourth phase by combining:

- Emotion recognition
- Dialect detection
- Context-aware translation
- Natural expressive speech output.

2.4 APPLICATIONS OF EPMSSTS:

Emotion-preserving multilingual speech translation has widespread use in communication, accessibility, safety, and service interaction.

- Healthcare and Medical Assistance
- Education, Teaching, and Learning Support
- Customer Support and Service Centers
- Tourism and Travel Communication
- Emergency and Crisis Response
- Accessibility and Social Inclusion

SYSTEM ANALYSIS

3. SYSTEM ANALYSIS

System analysis focuses on understanding the communication challenges that exist when translating speech across different languages, emotions, and dialects. Present translation technologies mainly prioritize linguistic accuracy, but human communication depends heavily on vocal expression, tone, and cultural variations. When these elements are removed during translation, the listener receives an incomplete message, which may alter meaning, emotional intent, and context.

The EPMSSTS system analyzes speech not only for words, but also for:

Prosody (pitch, rhythm, loudness, stress)

Emotional tone (happy, sad, angry, calm, fearful, etc.)

Dialect patterns (regional vocabulary, accent, pronunciation)

Contextual meaning (politeness, formality, urgency)

By addressing these dimensions, the system aims to provide communication that feels natural and human rather than mechanical. System analysis reveals a clear need for a translation solution capable of preserving expressive qualities of speech, improving accessibility, and supporting real-time multilingual communication without losing emotional meaning

.3.1 EXISTING SYSTEM

Current multilingual translation systems fall into three primary categories:

3.1.1 Text-based Translation Tools

Examples: Google Translate (text mode), online dictionaries

Users must type input manually

Translations lack emotional indicators

Not suitable for non-literate users

3.1.2 Speech-to-Text-to-Speech Translation Tools

Examples: Google Translate (voice mode), Microsoft Azure Speech Translation

Speech is converted to text, translated, then spoken aloud

Output sounds robotic and neutral

Emotional tone, urgency, sarcasm, and emphasis are lost

3.1.3 Neural Speech Translation Models

Examples: Whisper-based pipelines, open-source S2S models

Improved speech recognition and quality

Still lack emotion preservation and dialect sensitivity

Focus remains on accuracy, not human expressiveness

Across all these systems, communication is treated as a purely linguistic process, ignoring expressive and social elements. Users receive translations that are technically correct but emotionally disconnected, reducing clarity and natural interaction.

3.1.4 Limitations:

No Emotion Preservation

Lack of Dialect Recognition

Robotic and Unnatural Speech Output

Not Suitable for Non-literate Users

No Context or Cultural Adaptation

Limited Real-Time Usability

Misinterpretation in Critical Situations

3.2 PROPOSED SYSTEM

The proposed system, called the Emotion-Preserving Multilingual Speech-to-Speech Translation System (EPMSSTS), aims to translate spoken language from one language to another while keeping the speaker's emotional tone, intent, and natural expression intact. Instead of treating speech as plain text, the system listens to how the person sounds, identifies the emotion and dialect, translates the meaning, and then produces speech in the target language that still feels expressive and human.

In this system, the user simply speaks normally in their own language and dialect. The system automatically captures the audio, recognizes the spoken content, detects emotional cues such as happiness, sadness, frustration, urgency, or calmness, and then identifies any dialect influence. After understanding these elements, the system generates a translated output that reflects both the meaning and emotional quality of the original speech. The final spoken output sounds natural, expressive, and contextually appropriate rather than robotic or emotionally flat.

The proposed system is especially useful for users who cannot read or type, since the entire interaction happens through speech. It supports real-time communication, making it suitable for live conversations between people who do not share a common language. The architecture is designed to run on standard hardware while still being scalable for future deployment in mobile apps, customer service systems, healthcare communication tools, and educational environments.

Overall, EPMSSTS bridges the gap left by existing translation systems by focusing not only on the words being spoken, but also on how they are spoken. The system creates a more natural, relatable, and human communication experience across different languages and cultural backgrounds.

3.3 METHODOLOGY OF PROPOSED SYSTEM

- **Speech Input and Audio Capture:** The system captures the user's voice, removes background noise, and prepares clean audio for processing..
- **Speech Recognition and Content Extraction:** The captured audio is converted into text to extract the spoken words, sentence structure, and linguistic meaning.
- **Emotion Detection from Voice and Text**: The system analyzes tone, pitch, and vocal patterns to identify the speaker's emotional state.
- **Dialect Identification and Adaptation:** The regional accent and dialect features are detected so the translation can match natural speaking styles.
- **Context-Aware Translation:** The text is translated by considering context, tone, politeness, and emotional intention for accurate meaning.

SYSTEM REQUIREMENT SPECIFICATION

4. SYSTEM REQUIREMENT SPECIFICATION

Software Requirement Specification plays an important role in creating quality software solutions. Specification is basically a representation process. Requirements are represented in a manner that ultimately leads to successful software implementation. Requirements may be specified in a variety of ways. However, there are some guidelines worth following:

- Representation format and content should be relevant to the problem.
- Information contained within the specification should be nested
- Diagrams and other notational forms should be restricted in number and consistent in use.

4.1 FUNCTIONAL REQUIREMENTS

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. So, it's important to make them clear both for the development team and the stakeholders. Generally, functional requirements describe system behaviour under specific conditions

4.2 NON-FUNCTIONAL REQUIREMENTS:

Non-functional requirements define the quality attributes, constraints, and standards that a system must meet to ensure it operates effectively and efficiently. Unlike functional requirements, which specify what a system should do, non-functional requirements focus on how the system performs its tasks.

- **Usability**

  The system should offer a simple voice-based interface that users of all ages, literacy levels, and languages can operate easily without training.

- **Reliability**

  The system must consistently process speech, detect emotions, and produce accurate translations even in varied environments and accents.

- **Performance**

  Translation, emotion detection, and speech synthesis must occur in real-time or near real-time with minimal delay.

- **Supportability**

  The system should be easy to update with new languages, dialects, or emotion models without major redesign.

- **Implementation**

  The solution must run efficiently on standard hardware and be deployable on mobile devices or embedded systems.

- **Interface**

  The UI should support clear microphone access, playback controls, and smooth interaction between input, processing, and output modules.

- **Legal**

  The system must comply with data privacy laws by ensuring secure handling of voice recordings and personal speech data.

4.3 HARDWARE REQUIREMENTS

Hardware requirements refer to the physical components and specifications of a computer system or device that are necessary for running a software application, system, or program effectively. These requirements ensure the system has the necessary computational power, storage, and peripherals to support the application's operation.

4.3.1 CPU Specifications

- CPU Type — Intel i7 or AMD Ryzen 7/9

4.3.2 Memory Specifications

- System Memory — 16 GB (DDR4 SDRAM)
- GPU VRAM — 4 GB to 6 GB
- Module Size — 8 GB
- Memory Type — DDR4 SDRAM

4.4 SOFTWARE REQUIREMENTS:

Software requirements refer to the specific software, tools, libraries, and frameworks that are necessary to run, develop, or maintain a software application. These requirements ensure that the underlying system or environment supports the proper functioning of the application.

- **Operating System** : Windows 10/11
- **Platform** : Integrated Development Environment(IDE) or VSCode
- **Database** : PostgreSQL,Redis,MinIO/AWS S3
- **Programming Languages** : Python 3.x
- **Frontend** : Streamlit or ReactJS
- **Backend** : Fast API or Flask

4.4.2 Modules

- PyTorch or TensorFlow
- Whisper or faster-whisper
- Wav2Vec2, Silero VAD
- Librosa
- Wav2Vec2 SER
- BERT/ROBERTa
- MarianMT, mBART-50 or NLLB-200
- Tacotron 2 or VITS
- CUDA/CuDNN
- HuggingFace Transformers, Docker

4.4.3 Tools:

- Python 3.x (Core backend language.)
- FastAPI / Flask (API services orchestration.)
- PyTorch / TensorFlow (Deep learning backends.)
- ReactJS / Streamlit (User interface display.)
- Hugging Face Transformers (Access large models.)
- Whisper / faster-whisper (Accurate speech transcription.)
- Tacotron 2 / VITS (Expressive speech synthesis)
- PostgreSQL (Logs and glossary.)
- Redis (Real-time data buffer.)
- CUDA/CuDNN (GPU acceleration needed.)
- Docker (Containerized deployment.)
- Git (Version control)

SYSTEM DESIGN

5. SYSTEM DESIGN

System design is the process of defining the elements of a system such as the architecture, modules and components, the different interfaces of those components and the data that goes through that system. It is meant to satisfy specific needs and requirements through the engineering of a coherent and well-running System.

CONCEPT OF UML

UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems.ML stands for Unified Modeling Language is different from the other common programming languages such as C++, Java, COBOL, etc.UML is a pictorial language used to make software blueprints.There are a number of goals for developing UML but the most important is to define some general purpose modelling language, which all modelers can use and it also needs to be made simple to understand and use.

UML DIAGRAMS:

UML (Unified Modeling Language) diagrams are a set of standardized visual representations used to model and describe the structure, behaviour, and interactions within a system. UML is a general-purpose modelling language used in software engineering to visualize, specify, construct, and document the design of a software system.

5.1 USE CASE DIAGRAM

A use case diagram in the Unified Modeling language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals(represented as use case),and any dependencies between those cases. The main purpose of a use case diagram is to show system functions are performed for which actor. Roles of the actors in the system can be depicted.

Fig 5.1.1 shows the use case diagram of our system which describes the interaction between actors which are the one will interact with the subjects .

5.1.1 OVERVIEW OF THE USE CASE DIAGRAM

The Use Case Diagram illustrates the interaction between the user and the Emotion-Preserving Multilingual Speech-to-Speech Translation System. It shows the major functions such as speech input, speech-to-text conversion, emotion and dialect detection, translation, and final speech output.

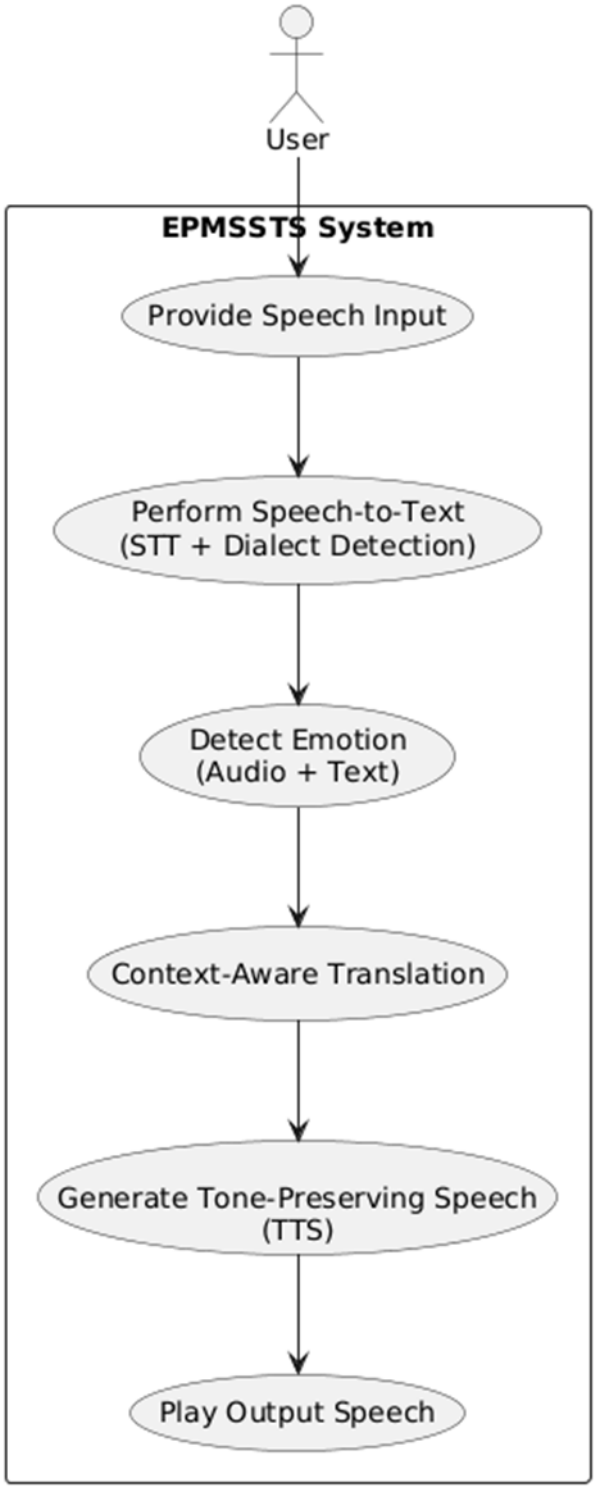

Figure 1 : USE CASE DIAGRAM

5.2 ACTIVITY DIAGRAM

An activity diagram is a type of Unified Modeling Language (UML) diagram used to model workflows and business processes. It visually represents the sequential and parallel activities within a system, providing a high-level view of the dynamic aspects of the system. Here are key points about activity diagrams:

5.2.1 OVERVIEW OF ACTIVITY DIAGRAM

The Activity Diagram shows the step-by-step workflow of the translation process, starting from speech capture to the generation of emotion-preserved speech output. It represents the flow of control between different processing stages of the system.

Explanation:

1. Speech Input & Preprocessing

The user first gives speech input to the system. The system captures the voice, removes background noise, and prepares clean audio for processing.

2. Speech-to-Text & Dialect Detection

The audio is passed to the STT module, which converts speech into text. At the same time, the system analyses the audio/text to detect the speaker's dialect (e.g., Telangana / Andhra).

3. Emotion Detection (Audio + Text)

The system then checks the tone of the voice and the meaning of the text to detect the user's emotion (happy, sad, urgent, etc.) and attaches emotion tags to the text.

4. Context-Aware Translation

Using the text, dialect tag, and emotion tags, the translation module generates a meaning-preserving and emotion-aware translation in the target language.

5. Tone-Preserving Speech Output

Finally, the translated text goes to the TTS module, which produces natural speech in the target language, keeping the original emotion and style, and this audio is played back to the user.

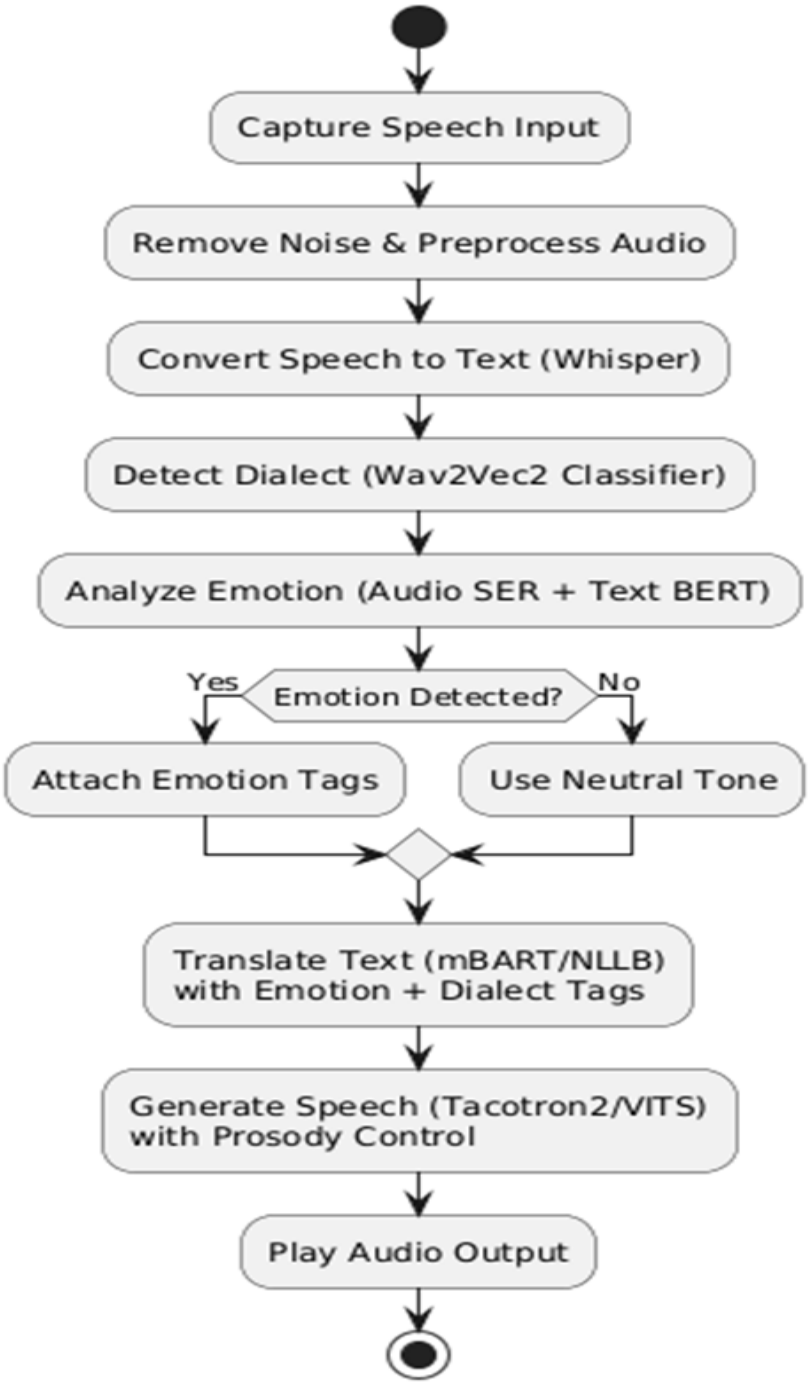

Figure 2 : ACTIVITY DIAGRAM.

5.3 DATA FLOW DESIGN

5.3.1 Data Flow Diagram

The **Data Flow Diagram (DFD)** shows how data moves across different modules of the EPMSSTS system. It represents the flow of audio, text, emotion tags, dialect information, and translated output through the processing components and data stores.

5.3.2 Level 0

This part of the Data Flow Diagram explains how the system begins processing user input. It shows how the user provides speech, how the audio is captured, cleaned, and sent to the Speech-to-Text (STT) module for further processing.

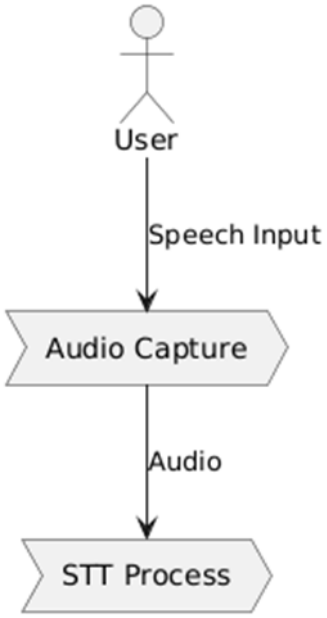

Figure 3 : Audio Capture + STT Input Flow

5.3.3 Level 1

This section illustrates how the captured speech is converted into text and analyzed.It shows how the system extracts the transcript, identifies the speaker's dialect, and detects emotional cues from both audio and text before sending them to the translation module.

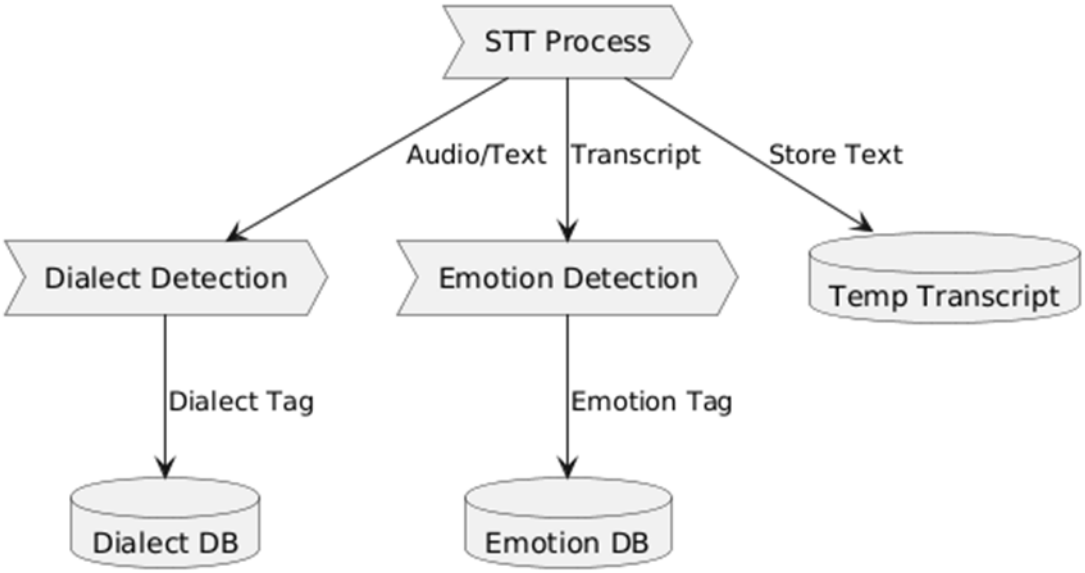

Figure 4 : STT+Dialect+Emotion Data Flow

Explanation:

- **Processes**:

1. STT converts the user's audio into a text transcript.

2. The generated transcript is stored temporarily for further processing.

3. Dialect Detection identifies the speaker's regional accent using audio/text.

4. Emotion Detection analyzes audio and text to detect the speaker's emotion.

5. All detected tags (text, dialect, emotion) are combined and sent to translation.

5.3.4 Level 2

This part shows how the system uses the transcript, dialect information, and emotion tags to generate an accurate translation. It also depicts how the translated text is converted into speech with preserved emotion and finally delivered back to the user as the output.

Explanation:

1.The Translation Engine receives the text, dialect tag, and emotion tag as input.

2.Context Analysis interprets meaning, tone, and sentence structure for accurate translation.

3. The Machine Translation model generates emotion-preserved translated text.

4. The TTS module applies prosody settings based on the detected emotion.

5. The Voice Synthesizer converts the translated text into natural-sounding speech.

6. The Output Delivery module plays the final emotion-preserved speech to the user.

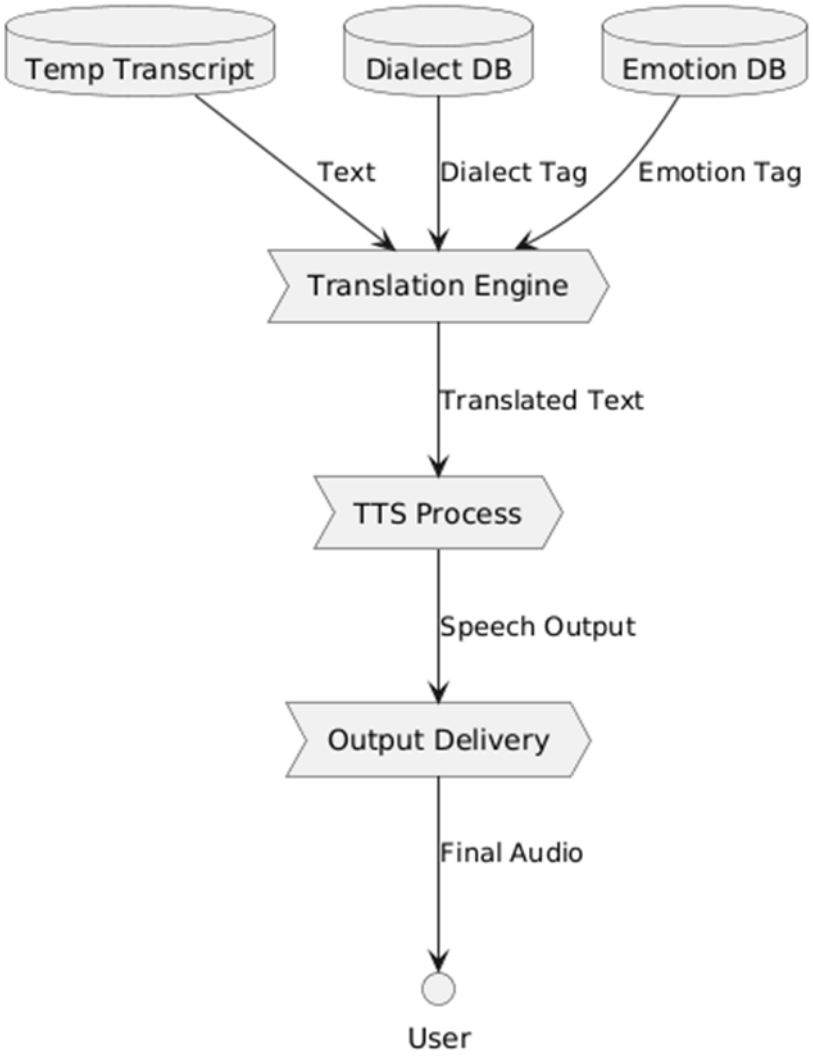

Figure 5 : Translation+TTS +Output Data Flow

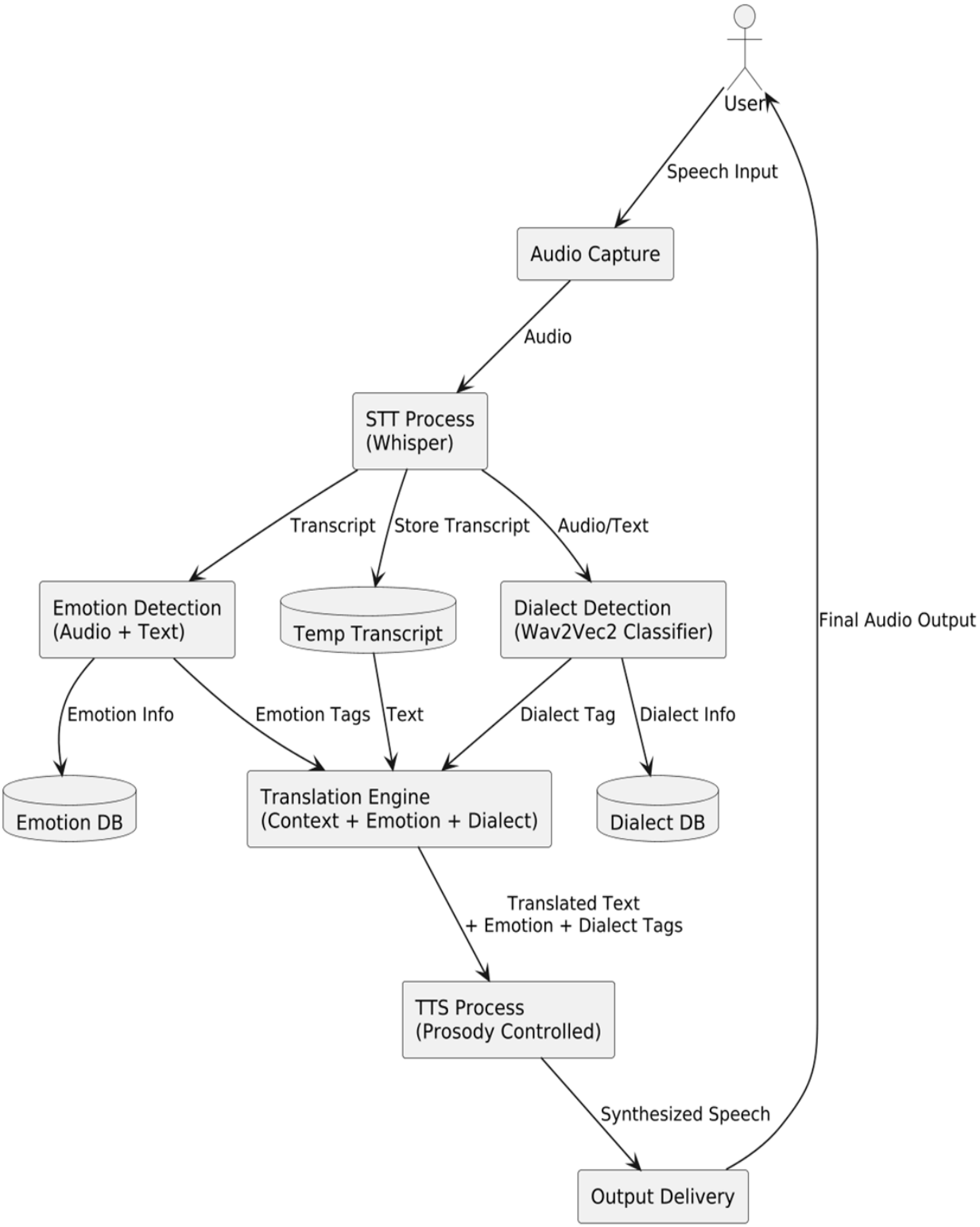

Figure 6 : DATA FLOW DIAGRAM

5.4 SEQUENCE DIAGRAM

The **Sequence Diagram** illustrates the chronological interaction between system components during translation. It shows how the user input flows through STT, dialect identification, emotion detection, translation, and TTS modules.

Explanation:

1. The user gives speech input through the UI, which captures and sends the audio to the STT module.

2. The STT module converts the audio into text and returns the transcript to the UI.

3. The UI sends the transcript and audio to the Dialect Detection and Emotion Detection modules to identify dialect and emotional tone.

4. All detected tags (emotion + dialect) along with the text are sent to the Translation Engine to generate the translated output.

5. The translated text is forwarded to the TTS module, which synthesizes speech with the preserved emotion.

6. Finally, the Output Module plays the translated emotion-aware speech back to the user.

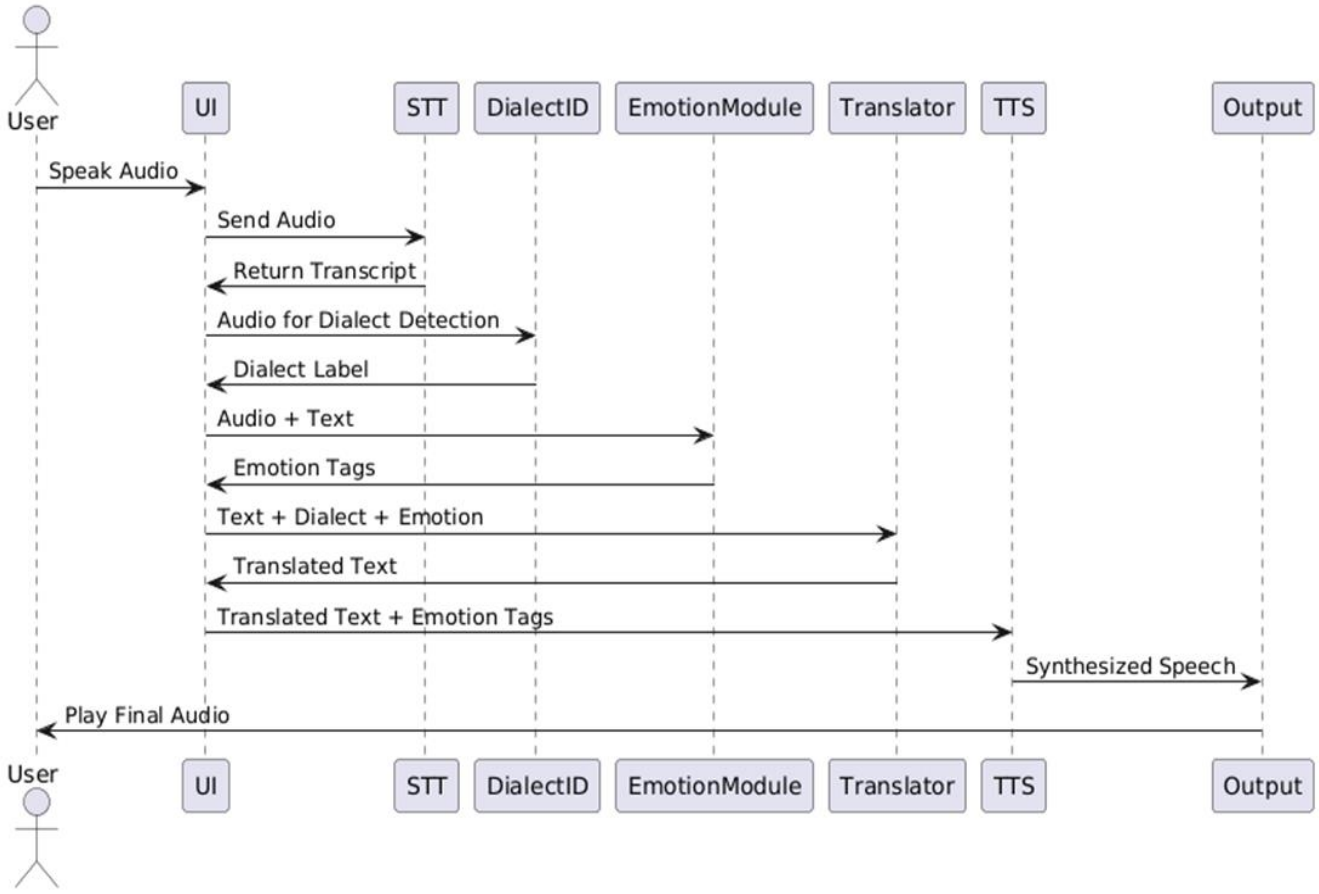

Figure 7 : SEQUENCE DIAGRAM

5.5 CLASS DIAGRAM

5.5.1 OVERVIEW OF THE CLASS DIAGRAM

The Class Diagram defines the structural design of the system by showing the key classes, their attributes, and the relationships among processing modules. It provides a high-level view of how different components interact within the system architecture.

Explanation:

1. The AudioInput class handles capturing the user's voice and preprocessing it before analysis.

2. The STT class converts the preprocessed audio into text using speech recognition techniques.

3. The DialectClassifier class identifies the speaker's regional dialect from audio or text features.

4. The EmotionDetector class detects the speaker's emotional state using both audio and text inputs.

5. The Translator class generates an accurate, emotion-preserved translation using text, emotion tags, and dialect information.

6. The TTS class converts the translated text into natural-sounding speech with proper prosody.

7. The OutputModule class manages the playback of the final speech output to the user.

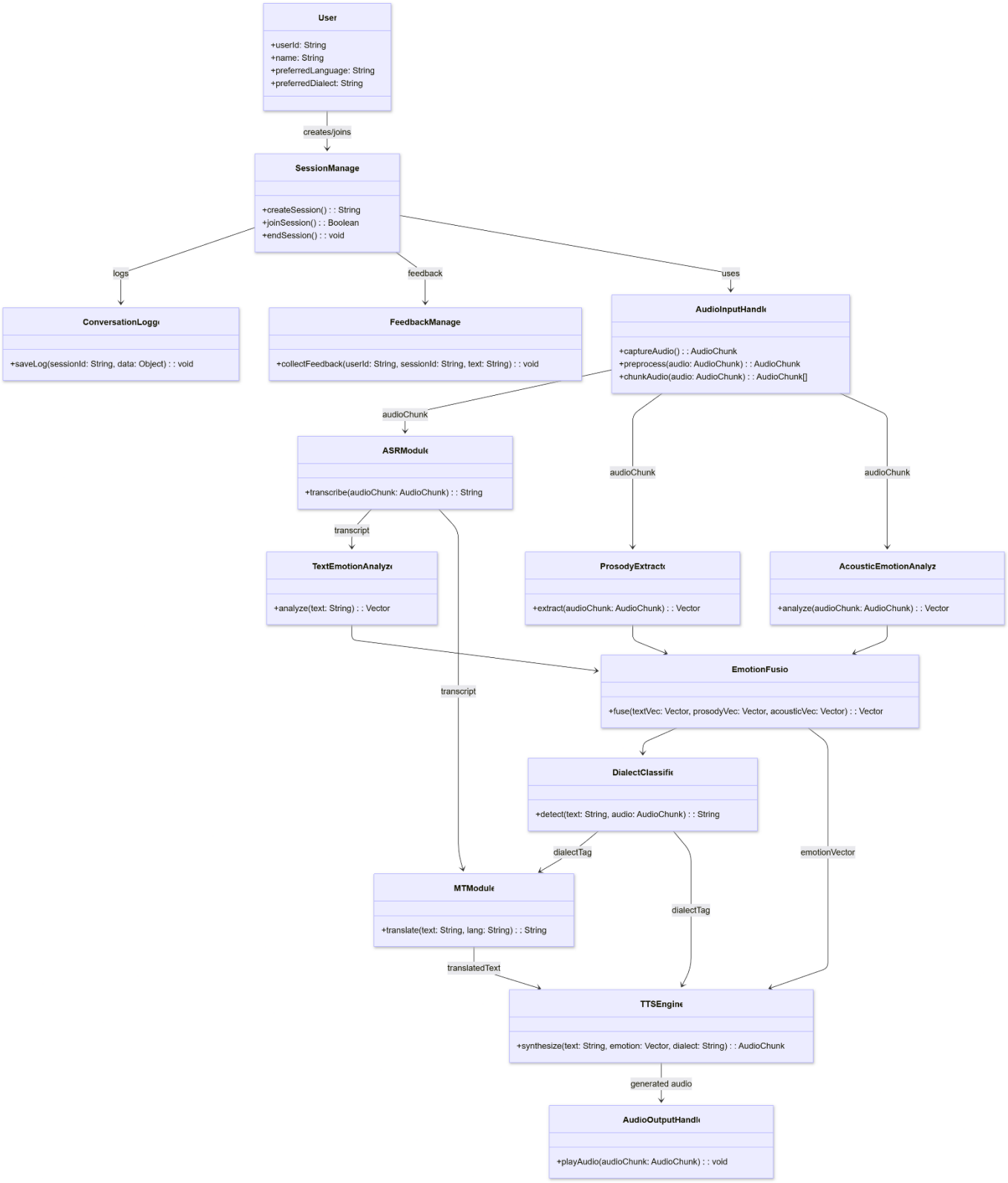

Figure 8 : CLASS DIAGRAM

5.6 SYSTEM ARCHITECTURE

The system architecture is built as a modular pipeline where the user interacts through a frontend interface that sends audio input to the backend API. The backend processes this input through key AI modules such as Speech-to-Text, Dialect Detection, Emotion Detection, Translation, and Text-to-Speech synthesis. A dedicated data layer using PostgreSQL, Redis, and cloud storage supports efficient processing and storage. All components work together to generate accurate, emotion-preserved multilingual speech output. This design ensures scalability, reliability, and smooth communication between modules.

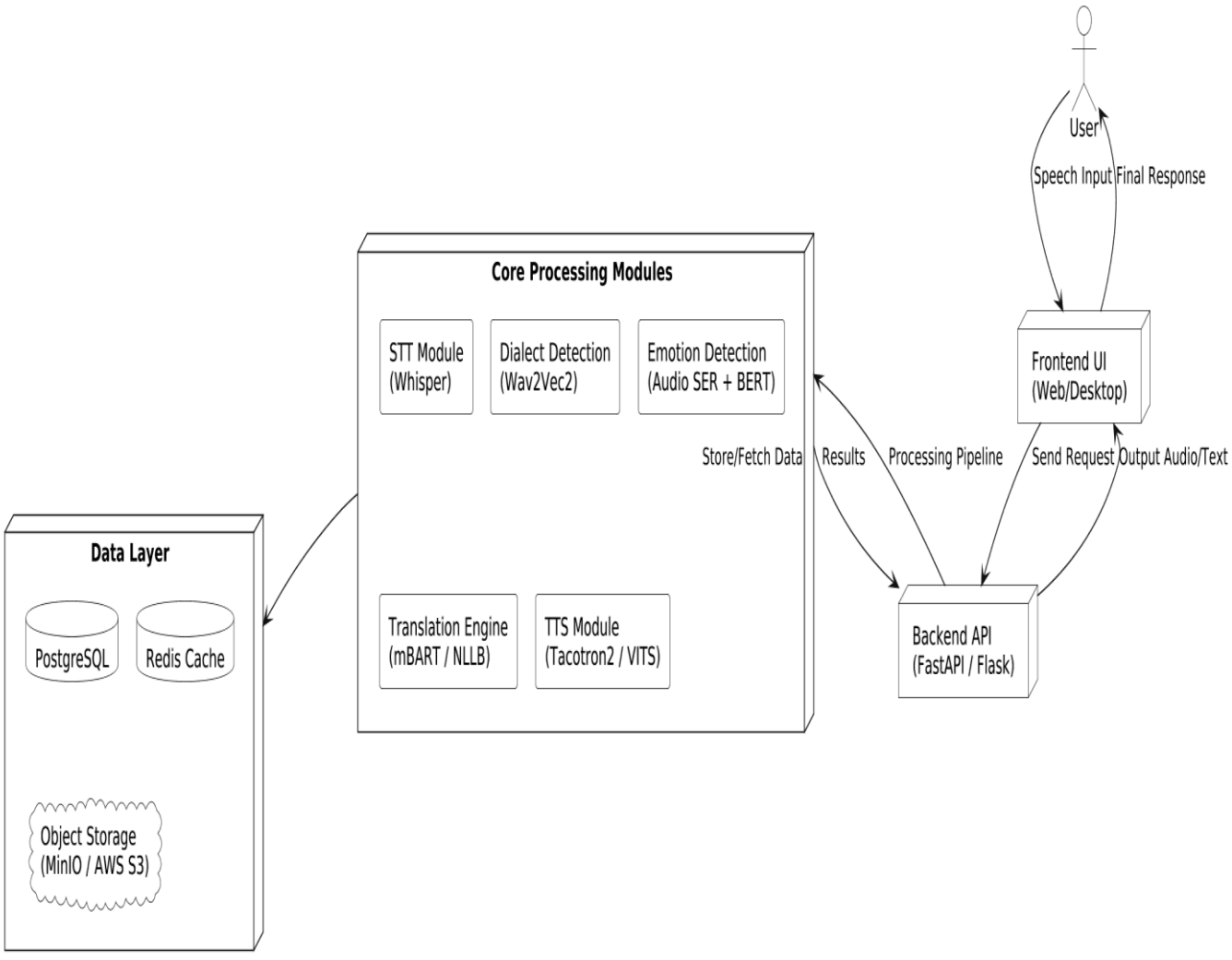

FIGURE 9 : SYSTEM ARCHITECTURE OF EPMSSTS

CONCLUSION

6. CONCLUSION

The Emotion-Preserving Multilingual Speech-to-Speech Translation System (EPMSSTS) addresses the major limitations of existing translation tools by focusing not only on linguistic accuracy but also on emotional tone, dialect variation, and contextual meaning. By integrating speech-to-text, dialect detection, multimodal emotion recognition, contextual translation, and expressive text-to-speech, the system ensures that the translated output remains natural, meaningful, and closely aligned with the speaker's original intent. This human-centric approach enhances clarity and emotional understanding in real-world communication scenarios.

Overall, the project demonstrates a robust, modular, and scalable prototype capable of supporting effective multilingual and emotion-aware communication. The system offers practical value in sensitive and interactive domains such as healthcare, education, customer support, and accessibility. With potential future enhancements like wider language support, real-time processing, and advanced emotional modeling, EPMSSTS lays a strong foundation for developing next-generation AI-based communication systems that bridge both linguistic and emotional gaps across diverse users.

BIBLIOGRAPHY

BIBLIOGRAPHY

1. Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerry-Ryan, R. J., Saurous, R. A., Agiomyrgiannakis, Y., & Wu, Y. (2018). Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions. In ICASSP 2018 – IEEE Int. Conf. on Acoustics, Speech and Signal Processing.
2. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of NAACL-HLT 2019 (pp. 4171–4186).
3. Liu, Y., Gu, J., Goyal, N., Li, X., Edunov, S., Ghazvininejad, M., Lewis, M., & Zettlemoyer, L. (2020). Multilingual denoising pre-training for neural machine translation. Transactions of the ACL, 8, 726–742.
4. Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2022). Robust speech recognition via large-scale weak supervision [White paper]. OpenAI.
5. Jia, Y., Weiss, R. J., Biadsy, F., Macherey, W., Johnson, M., Chen, Z., & Wu, Y. (2019). Direct speech-to-speech translation with a sequence-to-sequence model. In Proceedings of Interspeech 2019.
6. Duret, J., Esteve, Y., & Parcollet, T. (2023). Learning multilingual expressive speech representation for prosody prediction without parallel data. In 12th ISCA Speech Synthesis Workshop (SSW2023).
7. Baevski, A., Zhou, Y., Mohamed, A., & Auli, M. (2020). wav2vec 2.0: A framework for self-supervised learning of speech representations. In Advances in Neural Information Processing Systems 33 (NeurIPS 2020).
8. NLLB Team. (2022). No language left behind: Scaling human-centered machine translation. arXiv preprint arXiv:2207.04672.