



International Journal of Research Publication and Reviews

Journal Homepage: www.ijrpr.com ISSN 2582-7421

Multivoice Assist-AI

Mr. B. Sanjay Krishna ^{*1}, Saravana Krishanan M ^{*2}, Sathish Kumar S ^{*3}, Shiva Surya J^{*4}, Soundarajan J ^{*5}, Vasanthanarayanan M S ^{*6}

^{*1} Department of Computer Science and Engineering, Bachelor of Engineering, Sri Shakthi Institute of Engineering and Technology Coimbatore-641062, Tamil Nadu, India.

^{*2} Department of Computer Science and Engineering, Bachelor of Engineering, Sri Shakthi Institute of Engineering and Technology Coimbatore-641062, Tamil Nadu, India

^{*3} Department of Computer Science and Engineering, Bachelor of Engineering, Sri Shakthi Institute of Engineering and Technology Coimbatore-641062, Tamil Nadu, India

^{*4} Department of Computer Science and Engineering, Bachelor of Engineering, Sri Shakthi Institute of Engineering and Technology Coimbatore-641062, Tamil Nadu, India

^{*5} Department of Computer Science and Engineering, Bachelor of Engineering, Sri Shakthi Institute of Engineering and Technology Coimbatore-641062, Tamil Nadu, India

^{*6} Department of Computer Science and Engineering, Bachelor of Engineering, Sri Shakthi Institute of Engineering and Technology Coimbatore-641062, Tamil Nadu, India

ABSTRACT:

In recent years, artificial intelligence has become widely used for answering user queries. However, many AI chatbots have limited multilingual support, and users may face difficulties communicating in different languages. In our project, "Multivoice Assist-AI," we aim to address this problem by developing a multilingual AI chatbot that allows users to interact through text and voice in multiple languages.

In this system, the user enters a query through text or voice input. The system automatically detects the language and converts speech into text when required. Natural Language Processing (NLP) is used to understand the user's query and context. An AI model generates a suitable response, which can be translated into the user's preferred language and delivered as text or voice output. This project mainly focuses on improving multilingual communication, providing convenient voice interaction, and making AI assistance more accessible to users from different language backgrounds.

Keywords: Artificial Intelligence, Multilingual Chatbot, Natural Language Processing, Language Detection, Speech-to-Text, Text-to-Speech, Voice Assistant, API Integration

INTRODUCTION:

Artificial Intelligence is widely used in many applications such as chatbots, virtual assistants, and recommendation systems. However, many AI chatbots have limited language support, which creates difficulties for users who prefer to communicate in their native languages. Some users also find typing difficult and prefer interacting through voice.

To overcome these issues, we developed Multivoice Assist-AI. The main idea of this project is to create a multilingual AI chatbot that understands user queries in different languages and provides suitable responses through text or voice. The system uses automatic language detection, Natural Language Processing (NLP), speech-to-text, and text-to-speech technologies to improve communication. This project aims to make AI assistance more accessible, convenient, and user-friendly for people from different language backgrounds.

OBJECTIVES:

- To develop an AI chatbot that supports multiple languages.
- To automatically detect the language of user queries.
- To understand user queries using Natural Language Processing (NLP).
- To provide text and voice-based interaction.

- To generate relevant responses in the user's preferred language.
- To improve accessibility and simplify communication with AI systems.

SYSTEM STRUCTURE:

The Multivoice Assist-AI system is divided into three main parts:

- **Frontend:** Developed using React Native and Expo to provide a user-friendly interface for text and voice interaction.
- **Backend:** Implemented using Node.js and Express.js to process user requests and integrate AI services.
- **Database:** Firebase/Firestore is used to store user information and conversation history.

WORKING OF THE SYSTEM:

The system works in the following steps:

1. The user enters a query through text or voice input.
2. The query is sent to the backend server.
3. The system detects the user's language.
4. If voice input is used, speech-to-text converts it into text.
5. NLP processes the query to understand its meaning and context.
6. The AI model generates a suitable response.
7. The response is translated into the user's language if required.
8. The system displays the response as text or provides voice output using text-to-speech.
9. The conversation is stored for future reference, if enabled.

Features:

- Supports multiple languages
- Automatic language detection
- Text and voice interaction
- AI-based response generation
- Speech-to-text and text-to-speech support
- Conversation history management

Advantages:

- Enables communication across different language backgrounds
- Provides convenient voice-based interaction
- Reduces the need for manual language selection
- Improves accessibility to AI assistance
- Supports more natural user interaction

Applications:

- Student learning assistance
- Multilingual customer support
- Voice-enabled virtual assistance

- Language learning
- Information retrieval
- Multilingual chatbot systems

Table:

Feature	Text-Based Chatbot	Voice Assistant	Multivoice Assist-AI
Language Support	Limited/Varies	Varies	Multiple languages
Input Method	Text	Voice	Text and voice
Language Detection	Varies	Varies	Automatic
Speech Processing	No	Yes	Yes
Response Translation	Varies	Varies	Supported
Conversation History	Varies	Varies	Supported

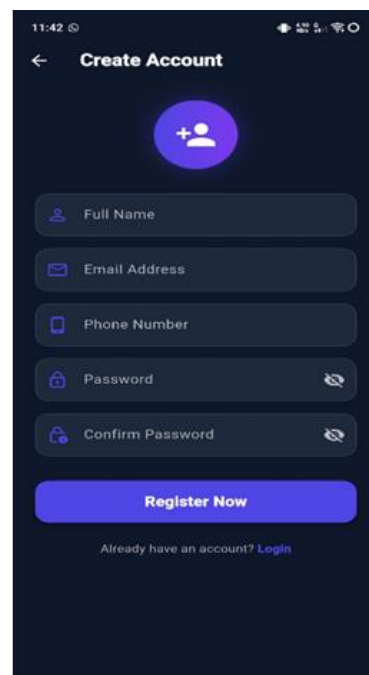
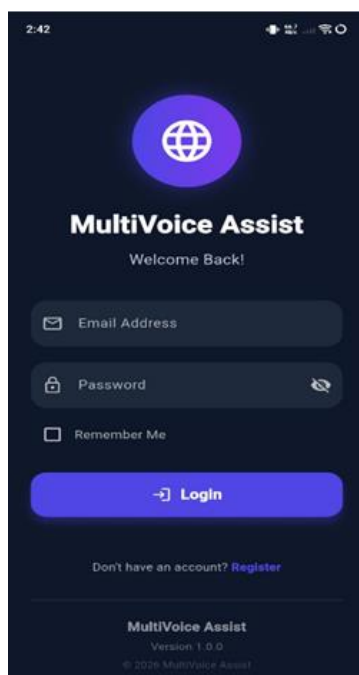
Implementation Details:

Multivoice Assist-AI is designed as a multilingual conversational system that supports text and voice-based communication. The frontend is developed using React Native and Expo, while Node.js and Express.js are used to manage backend requests and integrate AI services. Firebase/Firestore can be used to store user information and conversation history.

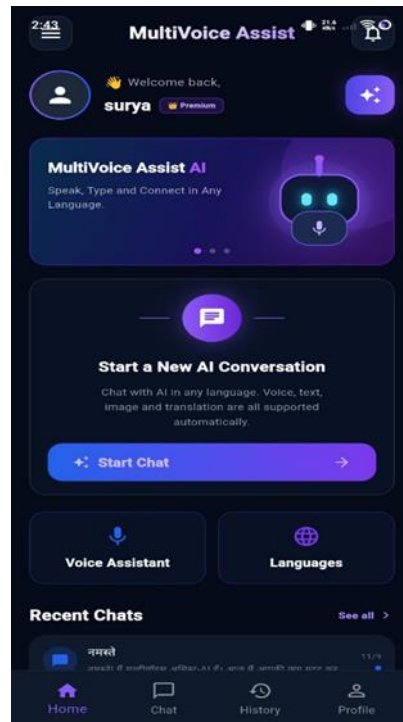
The system identifies the language of user input and uses speech-to-text processing when voice input is provided. Natural Language Processing (NLP) helps interpret the query, and an AI model generates a suitable response. When required, the response is translated into the user's language and delivered as text or speech using text-to-speech technology.

RESULTS AND DISCUSSION:

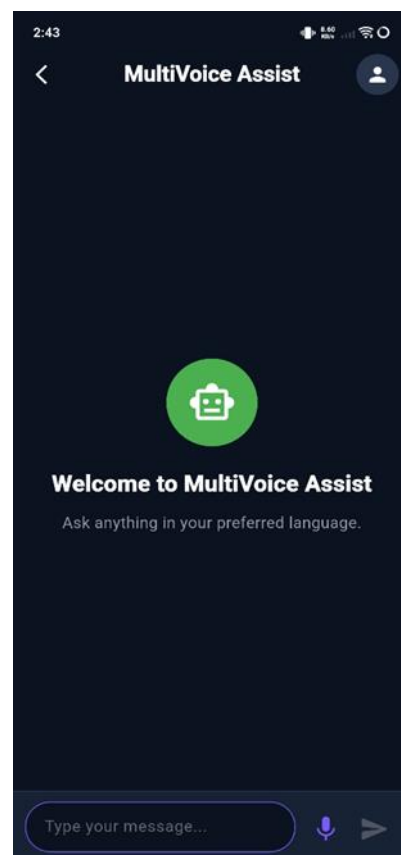
After developing and testing the Multivoice Assist-AI system, the application is designed to support multilingual communication through text and voice interaction. The system aims to identify the user's language, process queries using AI and Natural Language Processing (NLP), and provide suitable responses. Voice input can be converted into text, while generated responses can be delivered through text or speech. These features aim to make AI assistance more accessible and convenient for users who communicate in different languages.

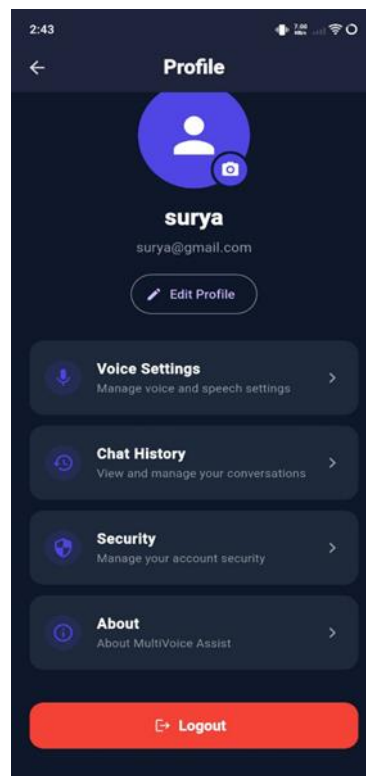
Sign In Page & Login Page:

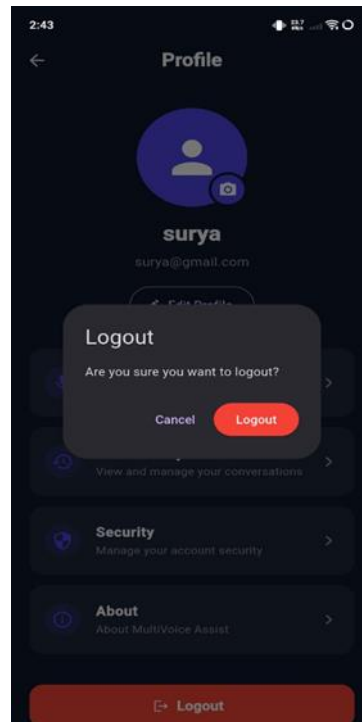
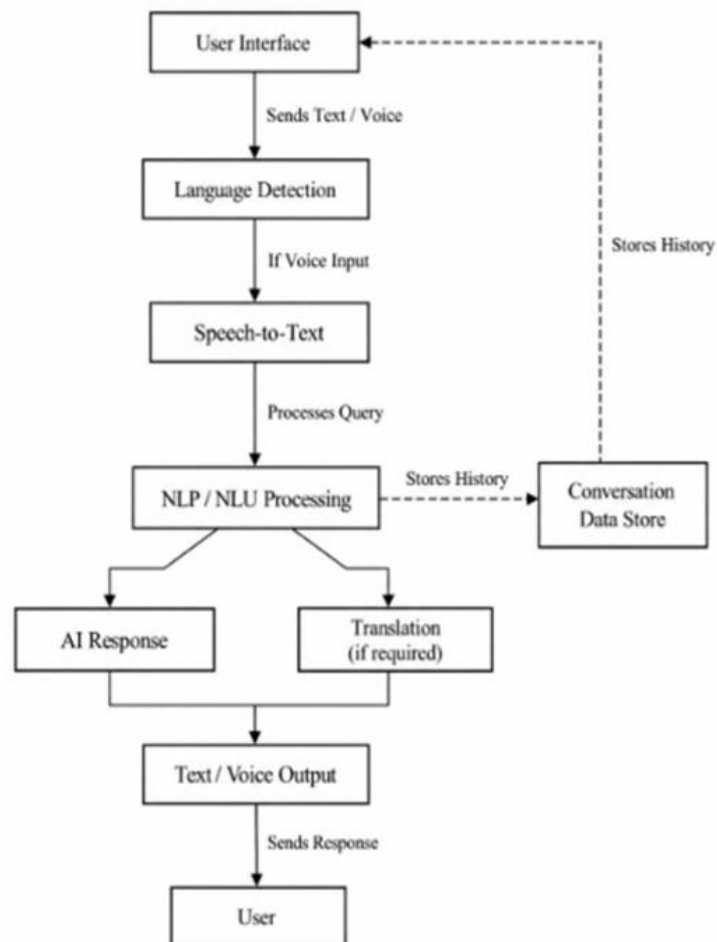
Dashboard page:



Chat page:



Chat History:**Profile page:**

Logout Page:**Workflow Diagram:**

Equations:

The response generation process is defined as:

$$(1) \quad (R = AI(Q,L,C))$$

Where:

- (R) is the generated AI response.
- (Q) is the user's query.
- (L) is the detected language.
- (C) is the conversation context.
- (AI) represents the AI model or API.

For voice input, speech is converted into text as follows:

$$(2) \quad (Q = STT(A))$$

Where:

- (A) is the user's audio input.
- (STT) represents the Speech-to-Text process.
- (Q) is the transcribed query.

For multilingual response generation, translation is defined as:

$$(3) \quad (R_L = T(R,L))$$

Where:

- (R_L) is the response translated into the user's language.
- (T) represents the translation process.
- (R) is the original generated response.
- (L) is the target language.

The final output is delivered as text or voice, depending on the user's preference.

ACKNOWLEDGEMENTS:

We would like to express our sincere gratitude to our project guide, **Mr. Sanjay Krishnan**, for his valuable support, guidance, and encouragement throughout the development of our project, Multivoice Assist-AI. We also thank our institution, Sri Shakthi Institute of Engineering and Technology, for providing the necessary support and resources to complete this project successfully.

CONCLUSION

Multivoice Assist-AI is a multilingual AI chatbot designed to support communication through text and voice in different languages. By combining automatic language detection, Natural Language Processing (NLP), AI-based response generation, and speech processing, the system aims to make AI assistance more accessible and user-friendly. It enables users to interact with the chatbot in their preferred language and receive responses through text or voice. In the future, we plan to improve language recognition, enhance response quality, support additional languages, and integrate more advanced voice and translation features.

REFERENCES:

1. Vaswani, A., et al. (2017). *Attention Is All You Need*. Advances in Neural Information Processing Systems.
2. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*.
3. Jurafsky, D., & Martin, J. H. *Speech and Language Processing*.
4. Hugging Face. *Transformers Documentation*.
5. OpenAI. *API Documentation*.

6. Mozilla. *Common Voice: Open Multilingual Speech Dataset*.
7. MASSIVE Dataset. *Multilingual Natural Language Understanding Dataset*.