



REAL-TIME E-VOICE: BRIDGING THE COMMUNICATION THROUGH SPEECH FOR VISUALLY IMPAIRED INDIVIDUALS

¹Nandana Rajesh, ²Neha Binduraj, ³Nischitha N Shetty, ⁴Sangeerth S, ⁵Dr. Sandeep Bhat

¹Final year B.E. Student, ² Final year B.E. Student, ³Final year B.E. Student, ⁴Final year B.E. Student, ⁵ Associate Professor, CSE

¹Department of Computer Science & Engineering,
¹Srinivas Institute of Technology, Mangaluru, India.

Abstract: Traditional email management challenges users with disabilities and multitaskers due to reliance on visual cues and manual input. Our Voice-Controlled Email Management System enables complete email management—composing, reading, and more—entirely through voice commands. It leverages speech recognition to accurately convert voice into actions and uses speech synthesis for auditory feedback. Designed for accessibility, the intuitive interface uses simple prompts, with features like dynamic speech rate control enhancing comprehension. This ensures an efficient, hands-free email experience, particularly for visually impaired users and those with limited mobility. It provides a smooth, accessible solution for managing emails.

Keywords: Voice Control, Email Management, Accessibility, Speech Recognition, Speech Synthesis, Assistive Technology.

I. INTRODUCTION

Email management is essential in modern communication, yet traditional interfaces pose significant barriers for individuals with disabilities, such as visual impairments and limited mobility, as well as for multitasking professionals. These users often struggle with systems reliant on visual cues and manual input, leading to frustration and reduced productivity. Current email platforms often lack robust, voice-controlled solutions tailored to these needs. This paper introduces "REAL-TIME E-VOICE," a project designed to create an innovative voice-based email system. By leveraging speech-to-text (STT) and text-to-speech (TTS) technologies, this system allows users to perform all essential email operations—including reading, composing, and sending emails—entirely through voice commands, thus offering a seamless and accessible communication experience without visual interfaces.

The core of the system is designed to interpret natural language commands, provide auditory feedback, and manage email interactions efficiently. The goal is to provide an efficient, user-friendly, and independent solution for email management, particularly empowering visually impaired individuals.

II. LITERATURE REVIEW

Existing research highlights numerous attempts to make email more accessible. Studies by Ingle et al. (2016) and Jain et al. (2021) explored voice-based email systems using STT, TTS, and IVR, though some faced limitations like requiring highly sensitive microphones or partial reliance on GUI elements. Other works (Jethani et al., 2018) introduced multi-lingual capabilities but still involved mouse clicks. Systems reviewed often focused on specific aspects like voice-based search (Paper [2] in section 2.3 of report) or integrated biometric authentication for security (as discussed in section 2.5 of report). While many systems aimed to reduce reliance on keyboards and mice, a fully integrated, intuitive, and comprehensive voice-only email management system with features like dynamic speech rate control for visually impaired users has been an area for further development. "Real-Time E-Voice" builds upon this body of work by focusing on a highly accessible user interface, robust error handling, and seamless integration of advanced speech technologies to provide a complete hands-free email management experience.

Several studies detailed specific architectural choices. For instance, various systems (as reviewed in Section 2.2 of the source report) utilized speech recognition, IVR, and even mouse click events for navigation, often relying on IMAP for message retrieval and STT for command input. Brintha et al. The use of algorithms like Viterbi to improve word recognition accuracy was explored, though efficiency could decrease with error rates (Paper [5], Section 2.2 of source report). Some systems, like that described in Paper [6], aimed to eliminate IVR, relying purely on STT and TTS.

The broader landscape of accessibility technologies also informs this domain. Research into voice-based search engines, voice-enabled social robots with gesture input, and the application of Principal Component Analysis (PCA) for enhancing speech signal clarity (as discussed in Section 2.3 of the source report) highlights the ongoing evolution of voice interaction. The

importance of user-friendly interfaces, secure authentication (including biometrics), and even web browsing capabilities through voice commands has been underscored (Section 2.5 of source report), pointing towards a demand for comprehensive digital inclusion.

More recent systems have leveraged contemporary development tools and APIs. Kavi Prakash N et al. (2022) proposed a Python-based system using gTTS and Pyaudio, incorporating STT/TTS for email operations including attachments, and face recognition for security, with plans for multi-language support. Similarly, Aditi Sharma et al. (2022) focused on Gmail accessibility using Python, Speech Recognition, and pyttsx3 libraries. Sherly Noel et al. (2020) developed an HCI-based smart voice email using the Google WebKit API, emphasizing noise elimination and voice sample authentication. Other Android-based assistive applications (Kuei-Chun Liu et al., 2015) have integrated TTS and Optical Character Recognition (OCR) for broader functionalities beyond email.

While these studies demonstrate significant progress, "Real-Time E-Voice" builds upon this foundation by specifically leveraging the Web Speech API for client-side speech processing within a standard web browser environment. This approach aims to provide a highly accessible, platform-independent solution that focuses on a seamless user experience through intuitive voice prompts, dynamic speech rate control for enhanced comprehension, and robust error handling, distinguishing it from systems reliant on specific desktop software, hardware, or more complex backend speech processing infrastructures.

III. PROPOSED SYSTEM: REAL-TIME E-VOICE ARCHITECTURE

The "Real-Time E-Voice" system employs a multi-component architecture designed for intuitive voice-controlled email management. The core components include:

- **User Device (Web Browser/Client):** This is the standard web browser on the user's computer or mobile device. It acts as the primary interaction medium. The browser must support the Web Speech API and have access to a microphone for voice input and speakers (or headphones) for auditory output. No special software installation is required on the user's end, enhancing accessibility.
- **Frontend:** Developed using HTML, CSS, and JavaScript, this component utilizes the Web Speech API for both speech recognition (converting spoken commands to text) and speech synthesis (providing voice responses). It processes user commands and displays a transcript and status updates.
- **Backend:** A Node.js server with Express.js handles the logic for email operations. It receives requests from the frontend (e.g., to send or fetch emails) and interacts with email services. * **Email Service:** Utilizes IMAP (Internet Message Access Protocol) for retrieving emails and SMTP (Simple Mail Transfer Protocol) for sending emails, interacting with standard email servers (e.g., Gmail).

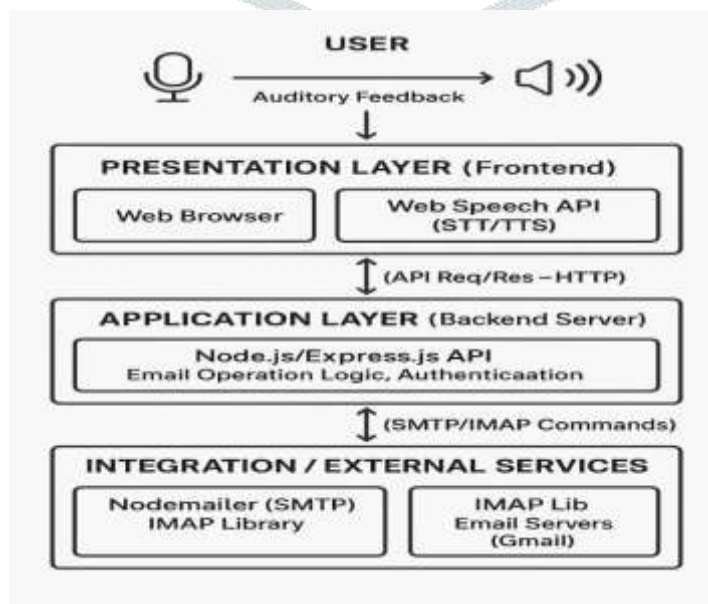


Fig.1: Architecture of a Voice-Controlled Email Application

Figure 1 illustrates the high-level layered architecture of the "Real-Time E-Voice" system. At the top, the User Layer represents the end-user interacting with the system via voice commands captured by a microphone and receiving auditory feedback through speakers. Voice commands flow into the Presentation Layer (Frontend), which operates within the user's web browser. This layer leverages the Web Speech API for Speech-to-Text (STT) conversion of voice commands and Text-to-Speech (TTS) synthesis for auditory responses, with UI logic managed by JavaScript to process user intent and manage interactions. For operations requiring email server access, the Presentation Layer communicates with the Application Layer (Backend Server) via HTTP API requests. The Backend Server, built with Node.js and Express.js, houses the core application logic, including API endpoint handling, email operation processing (such as composing or fetching emails), and user authentication with the email service.

This layer, in turn, interacts with the Integration / External Services Layer, utilizing libraries like Nodemailer for SMTP (sending emails) and an IMAP library for IMAP (retrieving emails) to communicate with external Email Servers (e.g., Gmail). Data and responses flow back up through these layers, culminating in auditory feedback delivered to the user, thus completing the voice-

controlled interaction loop. This layered approach ensures a clear separation of concerns, enhancing modularity and maintainability of the system.

IV. IMPLEMENTATION DETAILS AND TECHNOLOGIES

The implementation of the "Real-Time E-Voice" project relies on a carefully selected stack of modern web technologies to deliver its core functionalities. Central to the frontend is the Web Speech API, a browser-native API that provides robust speech recognition capabilities for understanding user commands and speech synthesis for delivering auditory feedback, including the crucial feature of dynamic speech rate control for enhanced user comprehension. Standard web technologies, namely HTML, CSS, and JavaScript, form the backbone of the user interface and client-side logic; specifically, an app.js file orchestrates the core voice command recognition and initial processing on the client side. For the backend, a Node.js runtime environment is utilized, with the Express.js framework enabling the efficient creation of API endpoints that manage email operations like sending and retrieving messages. Email dispatch via SMTP is facilitated by Nodemailer, a popular Node.js module, while corresponding IMAP functionalities for email retrieval and management are handled through other relevant Node.js libraries. The entire system communicates with email servers using standard Email Protocols, IMAP and SMTP, ensuring broad compatibility.

The development process focused on creating a clean, though visually minimalist, user interface optimized for voice-first interaction. Key implementation efforts involved integrating precise voice recognition to capture a range of commands, such as "compose email," "read emails," "search for [keyword]," "slow down," and "speed up." Upon receiving these commands, the backend processes the requests, securely interacts with the user's configured email account, and then transmits the results or status back to the frontend. The frontend subsequently vocalizes this information to the user. A significant aspect of the implementation was the incorporation of robust error handling mechanisms to gracefully manage scenarios like microphone access denial, unrecognized voice commands, and failures in email operations, thereby ensuring a smoother and more reliable user experience.

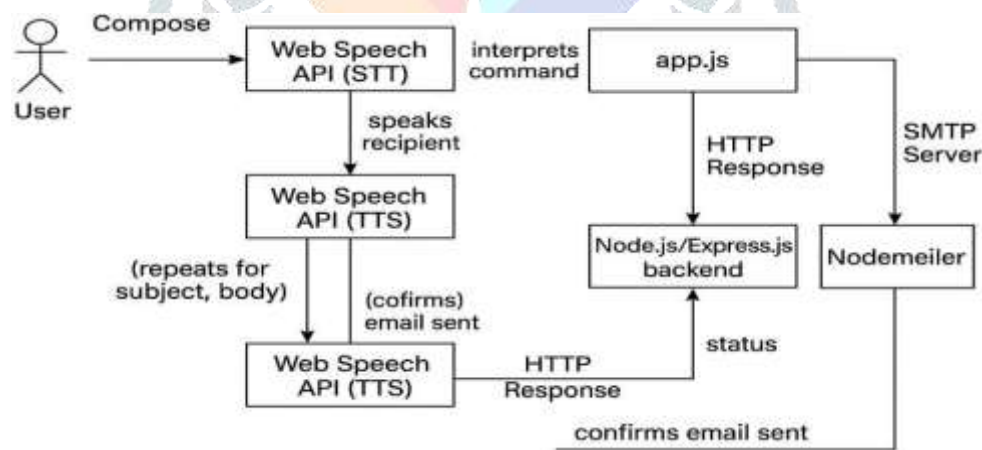


Fig.2: Voice-Driven Email Composition Process Flow

Figure 2 illustrates the step-by-step process of composing an email using voice commands in a voice-enabled email application. The process begins with the user issuing a voice command such as "Compose," which is captured and converted to text using the Web Speech API's Speech-to-Text (STT) functionality. The interpreted command is processed by the frontend logic (app.js), which then prompts the user via Text-to-Speech (TTS) to provide additional details such as the recipient, subject, and message body. Each user response is again captured via STT, processed, and passed along to the backend through an HTTP request. On the server side, the Node.js/Express.js backend receives the request, processes the email data, and utilizes Nodemailer to send the email through an SMTP server. Once the email is successfully sent, a status response is returned to the backend, which is then forwarded to the frontend. Finally, the Web Speech API's TTS confirms to the user that the email has been sent. This entire interaction flow demonstrates the seamless integration of speech recognition, backend processing, and email delivery technologies to create a hands-free email experience.

V. RESULTS AND DISCUSSION

The "Real-Time E-Voice" system successfully demonstrated its core functionalities, establishing an accessible and effective method for voice-controlled email management. During testing, key features yielded positive outcomes. The Compose Command functionality allowed users to initiate email creation by voice; the system then sequentially prompted for the recipient's email address, subject, and body entirely through auditory interaction, culminating in successful email delivery to specified recipients upon confirmation. Similarly, the Read Command effectively triggered the system to fetch and audibly present the latest unread emails or a predefined number of recent messages from the user's inbox, clearly vocalizing sender details, subject lines, and email content. For targeted retrieval, the Search Command enabled users to specify a keyword via voice, prompting the system to search the inbox and read out emails containing that term. A crucial accessibility feature, Dynamic Speech Rate Control, was also successfully implemented, allowing users to issue "slow down" or "speed up" commands to adjust the pace of synthesized speech, catering to varying comprehension needs and preferences. Finally, an Exit Command provided a straightforward way for users to gracefully terminate their session with the application.

Throughout these operations, the system consistently provided clear auditory feedback for all actions performed and any errors encountered, ensuring users remained continuously informed of the system's status. The user interface, deliberately designed to be visually minimal, proved effective for a voice-first interaction model. Furthermore, the integrated Web Speech API demonstrated reliable command recognition capabilities, particularly in environments with relatively low ambient noise,

underscoring its suitability for this application. These results confirm the viability of the system in providing a more inclusive email management experience.

VI. CONCLUSION

The "REAL-TIME E-VOICE" project successfully presents an innovative and effective solution for enhancing email accessibility, particularly for visually impaired individuals and those with limited mobility. By integrating advanced speech recognition and synthesis technologies, the system enables users to manage their emails entirely through voice commands, offering a user-friendly, hands-free experience. Key functionalities like composing, reading, searching emails, and dynamic speech rate control have been effectively implemented. This project demonstrates the significant potential of modern web technologies combined with accessibility-focused design to address real-world challenges. Future enhancements could include multi-language support, AI-driven contextual understanding, integration with smart devices for real-time notifications, and biometric authentication for enhanced security, further expanding its utility and reach.

VII. REFERENCES

- [1] Ingle, P., et al. (2016). "VOICE BASED E-MAIL FOR BLIND."
- [2] Jain, V., et al. (2021). "Voice based E-mail for the Visually Impaired."
- [3] Jethani, D., et al. (2018). "Voice-based system for the visually blind with multi-lingual facility."
- [4] Kumar, B., Singh, A. V., & Agarwal, P. (2022). A Novel Approach for Speech to Text Recognition System Using Hidden Markov Model. *International Journal on Recent and Innovation Trends in Computing and Communication*, 10(12), 181 –190. <https://doi.org/10.17762/ijritcc.v10i12.5934>
- [5] K. Prakash, A. RP, T. Monlam, P. V. Reddy, R. Singh, and R. Jain, "Voice based E-Mail with Attachment for Blind," *Proceedings of the Sixth International Conference on Trends in Electronics and Informatics (ICOEI 2022)*, pp. 274-280, 2022, doi: 10.1109/ICOEI53556.2022.9776840.