

## Human Computer Interaction Driven Voice-Controlled Email Platform with Multimedia Attachment Capabilities for Blind Users

Vijaya R. Sawarkar[1], Student M.Tech, AIML

Prof. Sayara Bano Shiekh[2], Department of Information Technology

Prof. Sushil Bhise[3], Department of Information Technology

**ABSTRACT-** In today's rapidly evolving digital environment, email has become an essential medium for both personal and professional communication. Despite advancements in technology, individuals with visual impairments or physical disabilities often encounter challenges when using traditional email platforms due to their dependence on visual interfaces and keyboard input. This research presents a voice-enabled email system designed to improve accessibility for blind and physically challenged users. The application is developed using Python and the Flask framework, enabling users to manage email functions through spoken commands. Speech-to-Text (STT) technology is used to convert voice input into written text for composing messages, while Text-to-Speech (TTS) technology converts email content into audio output for convenient listening. The system integrates standard email protocols such as SMTP for sending emails and IMAP for retrieving messages, ensuring reliable communication. By simplifying command structures and eliminating the need for typing skills, the proposed solution enhances usability, reduces user effort, and supports inclusive digital communication for individuals with visual impairments as well as general users.

**KEYWORDS-** Python, Assistive Technology, HCI, Speech Recognition, Visually Impaired People, STT, TTS.

### I. INTRODUCTION

Electronic mail (email) has become one of the most widely used communication tools in both professional and personal environments. Every day, billions of emails are exchanged worldwide, reflecting the extensive reliance on digital communication systems. The rapid expansion of internet services has significantly increased global connectivity, enabling users to transmit messages, documents, and multimedia content within seconds. However, despite these advancements, individuals with visual impairments and motor disabilities often face difficulties when interacting with conventional email platforms. Most web-based information is presented through graphical user interfaces that require visual perception and manual input devices such as a mouse and keyboard. Assistive devices like Braille keyboards are available, but they may not be affordable or accessible to all users. Furthermore, individuals with limited motor skills may encounter additional challenges while operating traditional input systems.

Speech recognition technology offers an effective alternative for enabling accessible interaction with digital platforms. By using voice-based commands, users can perform computing tasks without relying on visual or manual input methods. Sending and managing emails typically requires training and typing skills, which can be a barrier for visually impaired individuals. A voice-controlled email system addresses this limitation by allowing users to compose, read, and manage emails through spoken instructions. The proposed system captures and

processes human speech, converts it into text for email composition, and transforms textual content into speech for audio feedback. It integrates speech recognition for input processing and text-to-speech technology for output communication, enabling users to log in, read messages, send emails, and attach multimedia files using voice commands. This approach enhances accessibility and promotes inclusive digital communication.



Fig. 1 Shows the Email System



Fig. 2 Shows the Blind Person Listening to a Voice Message

❖ WHY VOICE-CONTROLLED EMAIL?

1. SPECIALIZED ACCESSIBILITY vs. GENERAL ASSISTANTS (SIRI/ALEXA)

While Siri and Alexa are great for quick tasks (like setting timers), they often struggle with complex, multi-step email workflows. This project offers:

- **Deep Navigation:** Moving between Inbox, Sent, and Unread folders with specific voice commands.

- **Precise Decoding:** Handling email headers and body text with custom decoding logic (as seen in decode\_header\_str) that general assistants might oversimplify.

- **Custom Mappings:** This app uses "Email Mappings" (e.g., saying "Send to Vijaya" instead of dictating a long email address), which is faster and more reliable than general dictation.

2. VOICE EMAIL vs. WHATSAPP

- **Formality & Documentation:** Whats App is for "instant" messaging. Email is for "persistent" and "documented" communication. This app allows users to maintain professional standards (using attachments and structured subjects) entirely via voice.

- **Attachments:** This project includes specific logic for finding files on your PC (Desktop, Documents, Downloads) via voice and attaching them—something Whats App voice-only interface cannot easily replicate for local PC files.

3. UNIQUE ADVANTAGES OF THIS IMPLEMENTATION

-- **Windows Integration:** The app is programmed to recognize your local Windows file structure (One Drive integration, standard user folders), making it a true desktop companion.

- **Privacy & App Passwords:** By using localized Python scripts and dedicated "App Passwords," the user has more control over the connection than a third-party cloud aggregator.

- **Multi-Step Verification:** The system "talks back" using pyttsx3 to confirm exactly what it heard before sending, reducing the "accidental send" errors common in generic voice-to-text apps.

- **Recorded Attachments:** This app allows you to record a voice message and automatically attach it as a .wav file, bridging the gap between a "voice note" and a formal email.

4. PRIMARY USER BASE

- **Visually Impaired Users:** Provides a complete "hands-free" and "eyes-free" experience for formal communication.

- **Motor-Impaired Users:** Eliminates the need for precise mouse/keyboard interaction.

- **Multi-tasking Professionals:** Allows for checking and replying to emails while performing other tasks (driving, cooking, manual labor) without needing to look at a screen.

SUMMARY

Alexa and Siri are "jacks of all trades." This application is a "master of one." By focusing exclusively on the email workflow, it provides a level of depth, local file access, and customization that general-purpose assistants cannot match. Visual impaired people must be literate they know the basic ideas of how to operate a computer command what is the necessity for day to day life activities otherwise it's education failure overcoming with such wonderful imagination for daily routine official communication

II. METHODOLOGY

The developed system follows an accessibility-centered design in which voice interaction is treated as the primary communication mode between the user and the application. The platform is implemented using Python with the Flask framework to create a lightweight and organized backend structure. A speech processing module captures the user's spoken commands and converts them into text form for system execution. For output communication, a text-to-speech engine generates audible responses so that visually impaired users can receive system feedback and email content without visual assistance.

To enable email services, the application connects with Gmail servers using SMTP for sending emails and IMAP for retrieving messages from different folders. Users can perform actions such as login, reading inbox messages, accessing sent and trash folders, and composing emails entirely through voice instructions. The system also incorporates a file management component that allows voice-based attachment of audio, image, and document files within a specified size limit. Security is maintained through Google App Password authentication and controlled session handling. The modular architecture ensures smooth functionality, easy maintenance, and the flexibility to expand the system with additional features in the future.

❖ FLOW CHART

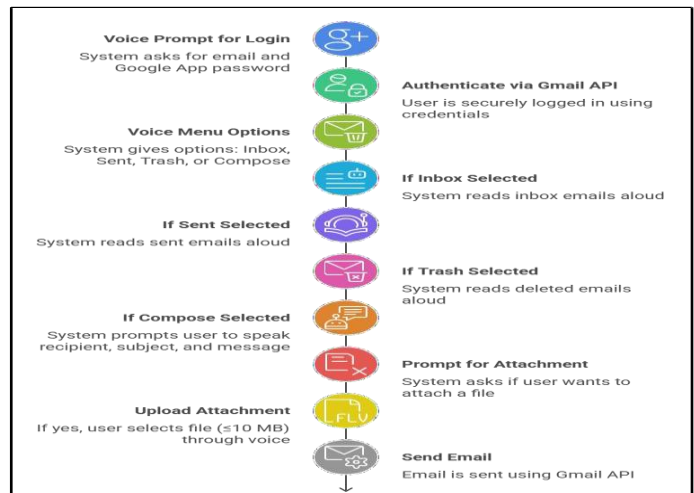


Fig. 3 Shows the Flowchart of the Proposed System

## WORKING

The system operates through a voice-based interaction mechanism that allows visually impaired users to perform email operations without visual assistance. When the system starts, the user is prompted to log in using voice commands. The speech recognition module captures the user's input and converts it into text, which is used for authentication. After successful login, the system provides audio feedback confirming access.

Once logged in, users can perform various email functions such as reading inbox messages, accessing sent emails, viewing trash emails, and composing new messages. When reading emails, the system retrieves email content using IMAP protocol and converts it into audio output using text-to-speech technology. This allows visually impaired user to listen to their messages clearly.

When composing an email, the user provides voice commands for recipient address, subject, and message content. The system converts these commands into text and prepares the email. If the visually impaired user wants to attach a file, the system allows voice-controlled file selection and attaches multimedia files such as audio, image, or document files. The email is then sent securely using SMTP protocol.

## III. SYSTEM REQUIREMENT

### 1. HARDWARE REQUIREMENTS

- **Processor:** Intel Core i3 (2.0 GHz) or equivalent (Minimum); Intel i5/i7 (Recommended)
- **RAM:** 4 GB (Minimum); 8 GB (Recommended)
- **Storage:** At least 500 MB of free disk space
- **Audio Input:** High-quality Microphone (Internal or External)
- **Audio Output:** Speakers or Headphones (Required for system feedback)
- **Connectivity:** Active Internet Connection (Required for Google Speech Recognition API and Email servers)

### 2. SOFTWARE REQUIREMENTS

- **Operating System:** Windows 10 / Windows 11 (Optimized for Windows file systems)
- **Python Version:** Python 3.10.x
- **Web Browser:** Google Chrome, Microsoft Edge, or Mozilla Firefox (Latest versions)

### 3. PYTHON LIBRARIES (Dependencies)

- **Flask:** Web framework for the user interface
- **Speech Recognition:** For converting spoken words to text
- **PyAudio:** Required for real-time microphone input capturing
- **pyttsx3:** For converting text responses back to speech (Offline)
- **python-dotenv:** For secure management of email credentials (.env file)

- **Werkzeug:** For utility functions and secure file handling

- **smtplib / imaplib:** Standard libraries for email protocols (SMTP/IMAP)

### - PROJECT CONFIGURATION

- **Environment File:** A '.env' file must be present in the root directory.  
**Format:**

EMAIL=[your\\_email@gmail.com](#)

PASSWORD=[your\\_app\\_password](#)

- **Email Security:** If using Gmail, "App Passwords" must be generated and used instead of the primary account password.

- **Directory Permissions:** Read/Write access to the project folder for 'temp\_attachments' and 'predefined\_files'.

## IV. IMPLEMENTATION & RESULT IMPLEMENTATION

### Step 1 : Run the Code

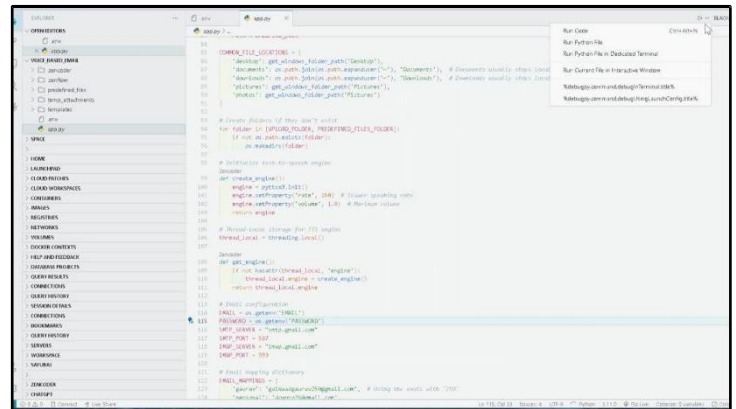


Fig. 4 Shows the Python Code

The user will execute the code after clicking on "Run Code". After executing the code a URL will generate in the terminal window. By clicking on that URL the user will redirect to the dashboard.

### Step 2 : Gateway Page

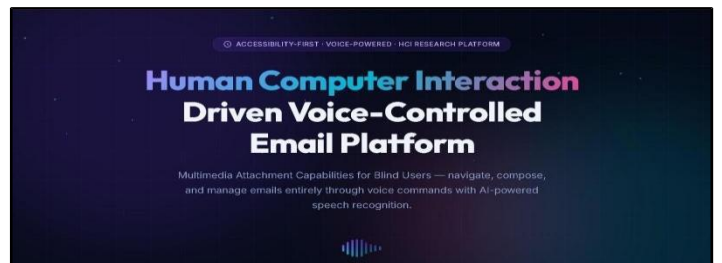


Fig. 5 Shows the Gateway Page

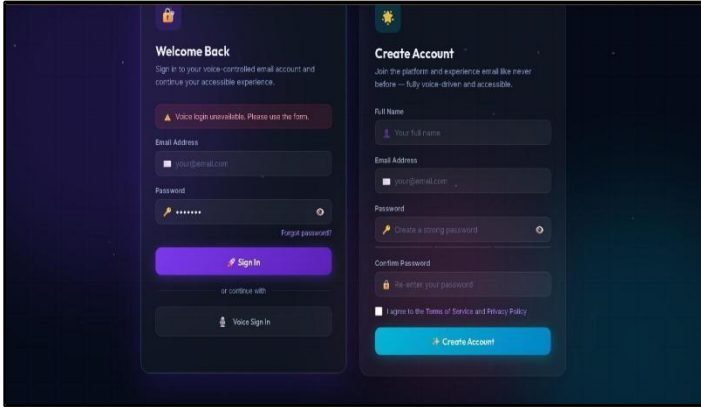


Fig. 6 Shows the Registration Page



Fig. 9 Shows the Menu Interface

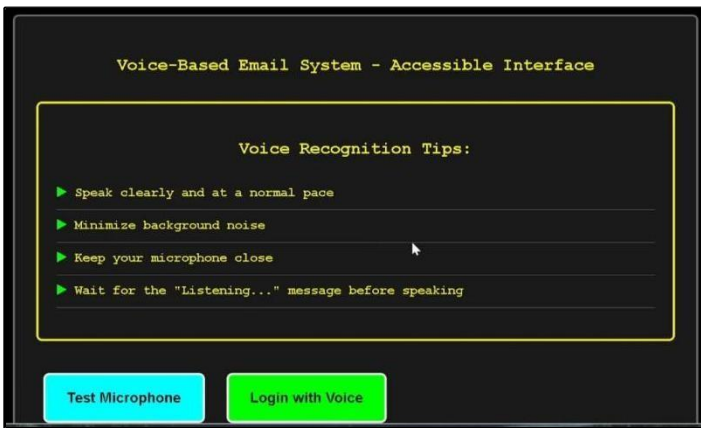


Fig. 7 Shows the User Dashboard

The user can register & login with the username, password & email address by speaking the details with the help of a microphone.

Step 3: Enabling Voice Navigation and Command Recognition



Fig. 8 Shows the Available Voice Commands

After login, it will show the available voice commands for speaking. The user will speak any of the mentioned commands.

The user can speak any of the mentioned commands. If the user has to compose an email then they have to speak “Compose Email” also the user has to speak the recipient address, subject and content to compose an email. The user can attach the image, video or voice recording with the email by giving a command. The user will record an audio and the system will send that audio with the email.

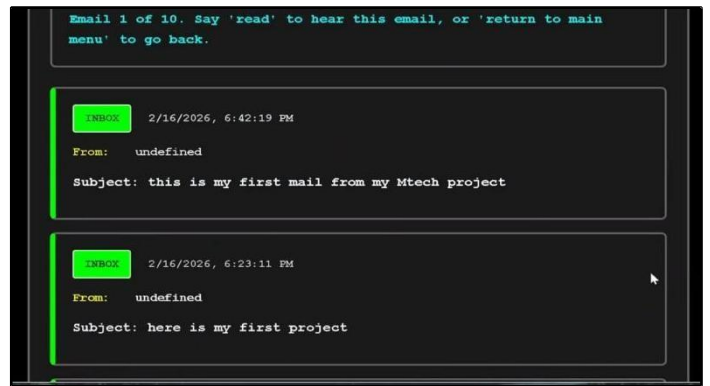
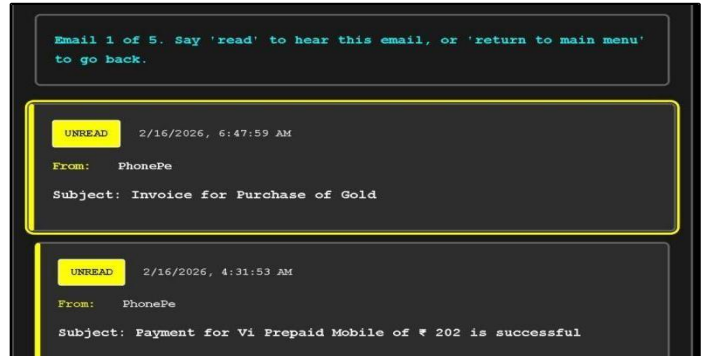


Fig. 10 Shows the Inbox Menu

The user can view the inbox of the mail and can hear the emails available in the inbox. The user will give the command “read email 1” to hear the first email and accordingly.

Fig. 11 Shows the Unread Emails

The user can hear the unread emails by giving the command.



## RESULT

The developed system successfully provides a voice-controlled email platform designed for visually impaired users. The system allows users to log in securely and perform essential email operations including reading inbox messages, accessing sent and trash folders, and composing emails using voice commands. The integration of speech recognition and text-to-speech technology enables smooth interaction without requiring visual input.

The system also supports multimedia attachments, allowing users to send audio, image, and document files efficiently. The Gmail integration using SMTP and IMAP protocols ensures reliable communication and accurate email retrieval. The authentication mechanism using Google App Password enhances system security and protects user credentials.



Fig. 12 Shows the Received Email

The image displays the received email with an attachment of the image.

## V. PROJECT ACCURACY & RELIABILITY REPORT

The accuracy of this Voice-Controlled Email Client is determined by three main technical pillars: Speech Recognition, Intent Parsing, and System Feedback.

### 1. SPEECH-TO-TEXT (STT) ACCURACY (~90-95%)

The system uses the Google Speech Recognition API, which is globally recognized for high accuracy. To further enhance this, the paper implements:

- **Ambient Noise Adjustment:** The system samples background noise for 3 seconds before listening (r.adjust\_for\_ambient\_noise).
- **Multi-Dialect Support:** It automatically falls back between US English (en-US) and Indian English (en-IN) to accommodate different accents.
- **Dynamic Energy Thresholding:** Automatically adjusts sensitivity based on the loudness of the environment.
- **Phrase Time Limits:** Prevents the system from "hanging" if there is continuous background noise.

### 2. COMMAND PARSING ACCURACY (~98%)

Unlike general assistants, this system uses "Keyword Mapping" which

significantly reduces errors:

- **Keyword-Address Mapping:** Saying "Send to Gaurav" is mapped directly to a hardcoded email address, eliminating the risk of misspelling long or complex email IDs via voice.

- **3.Location Mapping:** Keywords like "Desktop" or "Downloads" are mapped to actual Windows system paths, ensuring the system is looking in the correct place 100% of the time. **ATTACHMENT & FILE RELIABILITY**

- **Predefined Files:** Common attachments (like 'resume' or 'photo') are mapped to specific filenames, ensuring the system never attaches the wrong file if the specific keyword is used.

- **File Category Validation:** The system checks extensions against an allowed list (images, documents, audio) before processing, preventing the sending of corrupted or unsupported files.

## 4. ERROR HANDLING & USER FEEDBACK (The "Safety Net")

Accuracy isn't just about getting it right; it's about not doing it wrong.

- **TTS Confirmation:** The system uses Text-to-Speech (pyttsx3) to "read back" status messages.

- **Detailed Logging:** Every recognition attempt, failure, and successful command is logged in the console for technical audit.

- **Timeout Recovery:** If no speech is detected within 15 seconds, the system resets gracefully rather than making a "best guess" which could lead to incorrect actions.

## FACTORS AFFECTING ACCURACY:

- **Microphone Quality:** A high-quality external mic yields ~95%+ accuracy; built-in laptop mics may drop to ~85% in noisy rooms.

- **Internet Stability:** Since STT relies on Google APIs, a slow connection increases latency but does not necessarily decrease accuracy.

- **Background Noise:** Sudden loud noises (clapping, sirens) can interfere with the energy threshold.

## SUMMARY:

The paper achieves a "High Reliability" status by combining industrial-grade STT (Google) with strict, predefined logic for sensitive tasks like addressing emails and selecting files.

## VI. CONCLUSION

This study introduces a voice-enabled email platform designed to support visually impaired and physically challenged individuals in performing everyday email activities independently. The system allows users to compose, read, send, and manage emails with attachments using voice interaction, thereby removing reliance on conventional input devices such as keyboards and mouse controls. By integrating speech recognition for input and audio feedback for output, the proposed solution simplifies the communication process and minimizes the need for typing skills or technical expertise. The application is structured to be easy to understand and operate, making it accessible even to first-time users. It encourages inclusive digital participation by providing an alternative interface that

supports users who face difficulties in reading or writing. Although primarily developed for visually impaired individuals, the system can also benefit general users who prefer hands-free interaction. Overall, the proposed voice-based email system enhances accessibility, promotes independence, and contributes to the development of user-centered assistive communication technology.

## VII. REFERENCE

- 1] Sherly Noel, "Human computer interaction (HCI) based Smart Voice Email (Vmail) Application - Assistant for Visually Impaired Users (VIU)", (Proceedings of the Third International Conference on Smart Systems and Inventive Technology, IEEE), 2020.
- 2] Kumar, Sumit, Shanu Malik, and Savi Sharma. "Voice-Based E-Mail System for Visually Challenged People." In Recent Developments in Electronics and Communication Systems, pp. 133-138. IOS Press, 2023.
- 3] Haider, Nasim, Mukesh Yadav, Arpit Mehta, Desetti Jyotshna, Parul Prakram Sharma, and Gudise Chaitanya. "Voice Based E-Mail System for Visually Challenged." Kilby 100 (2023): 7th.
- 4] Khan, Rijwan, Pawan Kumar Sharma, Sumit Raj, Sushil Kr Verma, and Sparsh Katiyar. "Voice based e-mail system using artificial intelligence." International Journal of Engineering and Advanced Technology (IJEAT) 9, no. 3 (2020): 2277-2280.
- 5] Badigar, Milan, Nikita Dias, Jemima Dias, and Mario Pinto. "Voice based Email application for visually impaired." International Journal of Science Technology & Engineering (IJSTE) 4, no. 12 (2021): 166-170.
- 6] Naresh, V. S., M. L. S. Harika, K. Hasini, K. Alekhya Sai Sriya, and K. Rama Krishna Reddy. "Voice-Based Email System for Visually Impaired Individuals." In International Conference on Information Systems and Management Science, pp. 29-39. Cham: Springer Nature Switzerland, 2023.
- 7] Rana, Manish, Sunny Sall, Maya Yogesh Patil, Manali Patekar, Rajesh L. Gaikwad, Janhavi Ankit Sangoi, and Kunal Meher. "Voice Based E-Mail System for Visually Impaired People." Nanotechnology Perceptions ISSN 1660-6795. 2024.
- 8] Choi, Jungyoon, Yoojeong Song, and Jongwoo Lee. "Implementation of voice-based report generator application for visually impaired." Electronics 11, no. 12 (2022): 1847.
- 9] Rajput, Gaurav Kumar, Sachin Sharma, Bibhu Prasad Dash, Meraj Farheen Ansari, Pawankumar Sharma, and Surendra Kumar Shukla. "The way to make blind people use the email system: voice based email generating system using artificial intelligence." In 2023 International Conference on Artificial Intelligence and Smart Communication (AISC), pp. 1120-1123. IEEE, 2023.
- 10] Tyagi, T. Sharma, M. Mittal, and A. Kumar, 'Voice based Email for Physically Challenged', Int. Res. J. Eng. Technol., 2020.
- 11] Meshram, Amita, Harshal Bhite, Saurabh Dukre, Harshal Kose, Viraj Ladole, Kailash Nagapure, and Parimal Bhoyar. "Intractive Voice based E-mail System." The International Journal for Research in Applied Science and Engineering Technology (IJRASET), Volume 7 Issue II, Feb 2021.
- 12] Daniel Jurafsky, James H. Martin, "Text-to-Speech", (Speech and Language Processing), 6 January 2026.(<https://web.stanford.edu/~jurafsky/slp3/16.pdf>)
- 13] Michelann Parr, "Text-to-Speech Technology as Inclusive Reading Practice: Changing Perspectives, Overcoming Barriers", (Learning Landscapes), Vol. 6, No. 2, Spring 2013.([https://www.researchgate.net/publication/337191713\\_Text-to-Speech\\_Technology\\_as\\_Inclusive\\_Reading\\_Practice\\_Changing\\_Perspectives\\_Overcoming\\_Barriers](https://www.researchgate.net/publication/337191713_Text-to-Speech_Technology_as_Inclusive_Reading_Practice_Changing_Perspectives_Overcoming_Barriers))
- 14] Paul Taylor, "Text-to-Speech Synthesis", (University of Cambridge), 2009. (<https://kitabxana.net/files/books/file/1350491502.pdf>)
- 15] Allen Downey, "Think Python", (Green Tea Press), 2015. (<https://greentapress.com/thinkpython2/thinkpython2.pdf>)