International Journal of Creative and Open Research in Engineering and Management
ISSN: **3108-1754 (Online)**
Volume 02 Issue 08 August-2026 | Impact Factor: 3.5

VoxMail: Voice-Controlled Email Management Architecture for the Visually Impaired

A Fully Hands-Free Assistive Voice Mail System

Soumya M[1], S. Rajeswari[2], U. Poorna Sai Bhavana[2], Mohammad Hashmat Jaha[2], T. Jayanth[2]
[1]Assistant Professor (Guide), Department of Computer Science and Engineering
[2]Undergraduate Students, Department of Computer Science and Engineering
Rajiv Gandhi University of Knowledge Technologies (RGUKT), Ongole, Andhra Pradesh, India

How to Cite this Article:

M, S., Rajeswari, S., Bhavana, U. P. S., Jaha, M. H. & Jayanth, T. (2026). VoxMail: Voice-Controlled Email Management Architecture for the Visually Impaired. International Journal of Creative and Open Research in Engineering and Management, <i>02</i>(8), 1-9. https://doi.org/10.55041/ijcope.v2i8.285

https://doi.org/10.55041/ijcope.v2i8.285

Abstract—**Digital communication access is an indispens-able requirement in modern society, where individuals frequently send, read, and manage email correspondence [1]. Traditional email client interfaces rely heavily on graphical user interfaces (GUIs), mouse-driven pointers, and screen-reading software, resulting in high cognitive load, steep learning curves, and significant delays for visually impaired users [2]. This paper presents VoxMail, an assistive voice-controlled email management system designed specifically for visually impaired individuals. The proposed architecture integrates Speech-to-Text (STT) for natural voice com-mand parsing and dynamic composition, Text-to-Speech (TTS) for acoustic playback, and direct backend socket communication using Internet Message Access Protocol (IMAP) and Simple Mail Transfer Protocol (SMTP). A deterministic Finite State Machine (FSM) dialogue engine incorporates bidirectional audio confirmation loops to prevent accidental email transmissions and ensure robust error handling. Experimental evaluations across diverse acoustic environments demonstrate that VoxMail reduces average task completion time by over 73% compared to traditional screen-reader configurations while maintaining a 94.2% command parsing accuracy under ambient noise conditions. The system provides an efficient, hands-free platform that fosters independent digital communication.**

Index Terms—**Assistive Technology, Voice-Controlled Sys-tems, Speech Recognition, Human-Computer Interaction, IMAP, SMTP, Finite State Machine, Accessibility.**

I. Introduction

Email communication represents an essential pillar of contemporary society across educational institutions, corpo-rate workspaces, and personal interactions [1]. Students and working professionals consistently depend on electronic mail to receive official circulars, examination notifications, personal messages, and professional updates. However, conventional email platforms and web portals are fundamentally designed around visual paradigms. They require spatial scanning, precise cursor pointing, and visual verification across multi-column window panes.

Visually impaired individuals traditionally depend on third-party screen-reading software such as NonVisual Desktop Access (NVDA) or Job Access With Speech (JAWS) [2]. While these tools translate displayed digital text into synthetic speech or refreshable braille, they act merely as passive screen parsers. Users are forced to memorize complex combinations of keyboard shortcuts and listen sequentially through extensive non-essential interface clutter (such as sidebars, headers, advertisements, and navigation menus) before reaching the primary message content. This process is time-consuming, prone to navigation errors, and mentally exhausting.

The cognitive overhead associated with sequential audio traversal prevents efficient productivity [3]. For instance, when locating a single unread email, a visually impaired user may be required to navigate through dozens of Document Object Model (DOM) elements. In addition, existing voice assistants designed for consumer devices often lack dedicated mail-handling depth, failing to offer structured confirmation loops for composing, revising, and verifying recipients prior to dispatching outbound messages.

To address these limitations, we propose **VoxMail**, an intel-ligent, fully voice-interactive email management system. The architecture eliminates visual dependencies by mapping spoken voice commands directly to backend mail protocol operations. The system incorporates Automatic Speech Recognition (ASR) to convert user speech into text, a structured dialogue controller to maintain conversational context, direct IMAP/SMTP workers to interact securely with remote mail servers, and Text-to-Speech (TTS) engines to synthesize audio feedback.

The primary contributions of this paper are organized as follows:

1) A dedicated, hands-free voice interface that replaces visual and keyboard-based navigation with natural spoken dialogue.
2) A deterministic Finite State Machine (FSM) dialogue controller equipped with audio confirmation loops to eliminate unintended mail dispatches and enhance opera-

tional reliability.

3) Direct backend socket integrations with IMAP and SMTP protocols, bypassing DOM rendering overhead to accelerate message access.
4) A comprehensive performance evaluation comparing task completion times and command parsing accuracy between VoxMail and conventional screen-reader workflows.

II. Literature Review

Various assistive technologies have been proposed in recent years to improve digital accessibility and facilitate electronic mail communication for visually impaired users [2]. These systems can be broadly categorized into three core interaction paradigms.

A. Traditional Screen-Reader Systems

Screen readers interpret the Document Object Model (DOM) of web pages and desktop graphical interfaces, converting visual elements into synthetic speech [2]. Although screen readers are universally applicable across desktop operating systems, they impose a severe cognitive burden on visually impaired users. Users must listen through repetitive page elements, nested tables, and formatting tags to reach relevant email text. Furthermore, screen readers do not provide native voice-command mapping for direct execution of email operations.

B. Interactive Voice Response (IVR) Systems

IVR systems utilize telephony connections or dual-tone multi-frequency (DTMF) signaling over telephone networks to execute automated tasks. While accessible without a visual monitor, IVR systems operate on rigid hierarchical menus. Users must wait for long audio options to be read out before responding, and composing long, dynamic, and punctuated email bodies is impractical through numeric keypads or constrained telephony speech engines.

C. Voice-Assisted Desktop Clients

Early voice-enabled email prototypes integrated basic speech recognition engines into standard desktop software. However, many existing implementations lack robust finite-state dialogue controllers. Without explicit state tracking, ambient background noise can easily trigger erroneous actions, such as accidental deletions or premature message dispatch. Moreover, existing systems often lack flexible error recovery mechanisms when voice inputs are misrecognized.

D. Proposed Voice-Interactive Email Architecture

The proposed VoxMail architecture overcomes the drawbacks of prior systems by coupling modern continuous speech recognition with a deterministic Finite State Machine (FSM). All state transitions (such as drafting, recipient selection, confirmation, and deletion) are managed with auditory feedback and gated behind explicit two-way confirmation checks. Direct IMAP/SMTP protocol processing ensures low latency and complete independence from graphical screen layouts.

TABLE I
Comparison of Email Interaction Paradigms

Interaction Paradigm	Input Modality	Core Limitations
Screen Readers (NVDA/JAWS)	Keyboard Shortcuts	High cognitive load, sequential traversal of DOM elements.
IVR-Based Systems	Telephony / DTMF	Rigid menu options, slow response cycles, dynamic composition restricted.
Voice-Assisted Prototypes	Basic Speech API	False dispatches, lack of state validation and confirmation loops.
Proposed VoxMail	**Natural Voice Dialogue**	**Fully hands-free with FSM confirmation and direct protocol sockets.**

III. Proposed System

A. Architectural Design Principles

The architectural blueprint of VoxMail prioritizes low latency, reliable state preservation, hands-free feedback, and resilience to acoustic noise. The complete modular framework comprises three primary functional layers: the Voice Interface Layer, the Dialogue State Controller Layer, and the Backend Mail Protocol Engine (illustrated in Fig. 1).

B. Voice Interface Layer

The voice layer provides the primary medium of interaction between the visually impaired user and the platform:

- **Speech-to-Text (STT) Engine:** Continuous audio frames captured by the input microphone are routed through an acoustic noise-reduction filter. The stream is processed by a neural ASR model that produces normalized string tokens, parsing spoken characters, digits, and domain names accurately.
- **Text-to-Speech (TTS) Engine:** Converts structured server responses into natural speech. Auditory cues, punctuation pauses, and pitch modulation are utilized to distinguish between sender names, subjects, and email message bodies.

C. Dialogue State Controller Layer

The dialogue engine tracks user progress throughout multi-step email operations. It extracts semantic intents and slot parameters from the transcribed text, validating input against the current operational state before advancing.

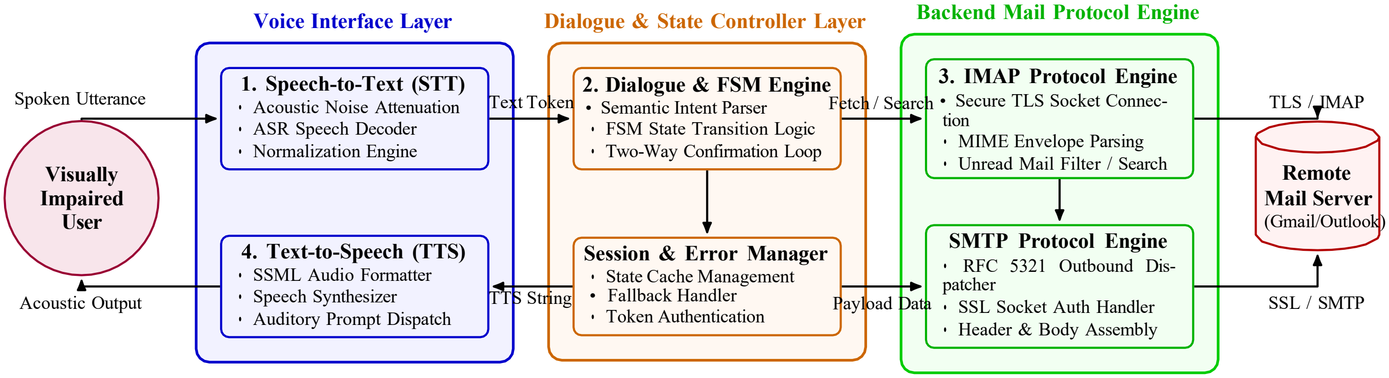
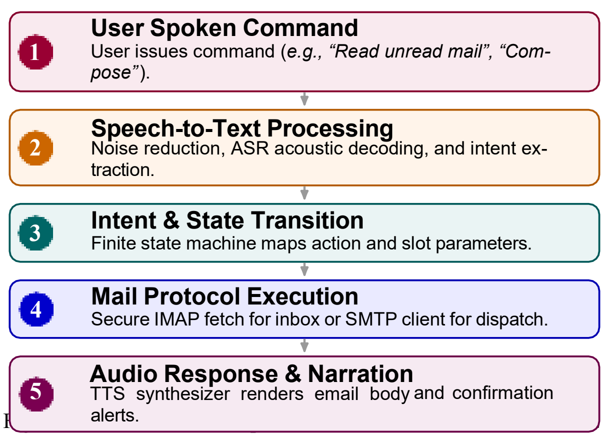


Fig. 1. System Architecture of the Proposed VoxMail Assistive Email Framework.

D. *Backend Mail Protocol Engine*

Direct backend communication ensures the system remains completely independent of graphical user interface changes on client platforms:

- **IMAP Engine:** Establishes secure TLS socket connections to the mail host, parses incoming multi-part MIME messages, strips superfluous HTML/CSS formatting tags, and extracts clean plain-text payloads for audio narration [4].
- **SMTP Engine:** Formats outbound message envelopes adhering to the RFC 5321 standard, applies authentication credentials, and manages secure SSL/TLS transmission [5].



E. *System Workflow*

The overall operational workflow proceeds sequentially across distinct processing phases, as illustrated in Fig. 2:

1) **Command Input:** The user initiates an interaction by speaking a voice command (e.g., *"Read my latest email"* or *"Compose a new message"*).
2) **Speech Processing:** The audio input is filtered and converted into raw text by the ASR engine.
3) **Intent & State Evaluation:** The dialogue manager extracts key parameters and determines the target state.
4) **Protocol Execution:** For read operations, the IMAP worker retrieves and parses unread mail headers. For compose operations, the system dictates prompts, captures recipient and body information, and initiates the confirmation loop.
5) **Audio Output:** The TTS engine synthesizes the results and speaks them back to the user.

F. *Mathematical Model of Dialogue State Machine*

The interactive dialogue is formally governed by a deterministic Finite State Machine (FSM):

$$\mathcal{S} = \{S_{\text{IDLE}}, S_{\text{LISTEN}}, S_{\text{READ}}, S_{\text{COMPOSE\_TO}}, S_{\text{COMPOSE\_BODY}}, S_{\text{CONFIRM}}\} \quad (1)$$

Let I_t represent the parsed user intent at time step t, and $S_t \in \mathcal{S}$ denote the current system state. The state transition function $\delta(S_t, I_t) \rightarrow S_{t+1}$ governs the system navigation.

To ensure transmission safety, any action altering mailbox states or dispatching outgoing messages requires passing through S_{CONFIRM}:

$$\text{Action} = \begin{cases} \text{Execute SMTP Dispatch}, & \text{if } I_t \in \{\text{"Yes"}, \text{"Confirm"}\} \\ \text{Abort / Re-record}, & \text{if } I_t \in \{\text{"No"}, \text{"Cancel"}\} \end{cases} \quad (2)$$

IV. Methodology and Evaluation

A. *Implementation Details*

The VoxMail backend framework is constructed on Node.js and Express to leverage an asynchronous, non-blocking event loop. The system connects to standard mail servers via secure TLS sockets using the `imap` and `mailparser` libraries for incoming mail management, and `nodemailer` for outbound SMTP operations.

Client-side speech processing is interfaced using Web Speech API pipelines for continuous speech recognition and SpeechSynthesis for acoustic feedback.

B. Voice Control State Algorithm

Algorithm 1 describes the algorithmic implementation of the dialogue state machine for composing and dispatching an email.

Algorithm 1 Voice Dialogue State Machine for Mail Dispatch

```
Initialize: State ← S_IDLE
while System is Active do
    AudioInput ← CaptureSpeechStream()
    TranscribedText ← STT(AudioInput)
    Intent ← ParseSemanticIntent(TranscribedText)
    if State == S_IDLE and Intent == “Compose” then
        State ← S_COMPOSE_TO
        TTS(“Please dictate recipient email address.”)
    else if State == S_COMPOSE_TO then
        Recipient ← FormatAddress(TranscribedText)
        State ← S_COMPOSE_BODY
        TTS(“Please dictate the message body.”)
    else if State == S_COMPOSE_BODY then
        Body ← TranscribedText
        State ← S_CONFIRM
        TTS(“Confirm sending to ” + Recipient + “? Say Yes or No.”)
    else if State == S_CONFIRM then
        if Intent == “Yes” then
            Status ← ExecuteSMTP(Recipient, Body)
            TTS(“Email dispatched successfully.”)
        else
            TTS(“Email canceled.”)
        end if
        State ← S_IDLE
    end if
end while
```

C. Experimental Setup

The platform was evaluated with a group of 10 volunteer participants across two distinct acoustic settings:

- **Quiet Environment (< 30 dB):** A controlled laboratory setting without background speech or ambient noise.
- **Ambient Noise Environment (45–55 dB):** A realistic office/room environment with background air conditioning, chatter, and keyboard sounds.

Each participant performed standard email tasks: checking unread emails, reading specific messages, composing and dispatching emails, and deleting messages.

D. Evaluation Metrics

The system performance is evaluated using the following metrics:

1) **Task Completion Time (TCT):** The total time (in seconds) required from initial user command invocation to the final execution and narration of the requested task.
2) **Word Error Rate (WER):** The transcription error rate during email dictation, computed as:

$$\text{WER} = \frac{S + D + I}{N} \tag{3}$$

where S represents substitutions, D represents deletions, I represents insertions, and N is the total reference words spoken.

3) **Command Intent Accuracy (%):** The percentage of spoken commands correctly parsed and mapped to the intended FSM state.
4) **Task Success Rate (%):** The percentage of email operations successfully completed without manual intervention or aborted errors.

V. Applications and Limitations of Existing Systems

A. Applications

1) **Educational Institutions:** Blind and visually impaired students can autonomously check university announcements, examination schedules, assignment submissions, and instructor feedback without assistance.
2) **Workplace Accessibility:** Visually impaired professionals can manage high volumes of daily correspondence rapidly, improving productivity in administrative and technical roles.
3) **Elderly Care and Motor Impairment:** Users suffering from tremors, motor disabilities, or visual degradation due to age can interact seamlessly with email services using only their voice.
4) **Hands-Free Mobile Computing:** Enables situational hands-free email management during driving or tasks requiring full manual engagement.

B. Limitations of Existing Systems

1) **Cluttered DOM Traversal:** Webmail interfaces contain extensive visual elements, requiring screen-reader users to listen through non-essential content before reaching message text.
2) **Lack of Conversational Context:** Existing voice prototypes treat each utterance in isolation, lacking state memory to support confirmation loops and error recovery.
3) **Accidental Transmissions:** Systems without robust confirmation mechanisms frequently trigger premature email dispatches due to misrecognized background words.

VI. Challenges

A. Acoustic Noise and Transliteration

Background noise, accents, and varying speech tempos can degrade ASR transcription accuracy. Additionally, spelling out alphanumeric email addresses (such as names with dots and numbers) verbally introduces transcription challenges that require specialized formatting rules.

B. Email Body Parsing and MIME Types

Modern emails often contain complex multi-part MIME formats with nested HTML, tracking pixels, and CSS styling. Extracting clean, readable plain text from raw MIME structures without narrating styling codes requires robust parsing filters.

C. *Network Latency and Timeout Management*

Maintaining continuous socket connections with remote mail servers over unstable internet connections requires graceful timeout handling and automatic reconnection strategies.

VII. Future Trends

A. *LLM-Powered Email Summarization*

Integrating Large Language Models (LLMs) into the backend pipeline will allow VoxMail to generate concise summaries of long email threads before narration, reducing listening time for users [6].

B. *Multilingual and Code-Mixed Speech Support*

Extending the speech recognition framework to support regional languages and code-mixed speech (such as Telugu-English or Hindi-English) will broaden accessibility across diverse linguistic demographics.

C. *Biometric Voice Authentication*

Implementing speaker identification and voice biometric verification will enable passwordless, secure user authentication directly through spoken passphrases.

VIII. Performance Comparison of Manual and VoxMail Systems

The proposed VoxMail system is compared with the conventional manual approach (screen reader with standard webmail) based on response time and overall efficiency. The comparison highlights the benefits of automating email navigation through direct protocol execution.

A. *Response Time Comparison*

The response time of the manual approach and the proposed VoxMail system is compared to evaluate how quickly email operations can be performed. As shown in Fig. 3, the average task completion time is reduced from 120 seconds in manual screen-reading workflows down to 8 seconds using VoxMail direct streaming.

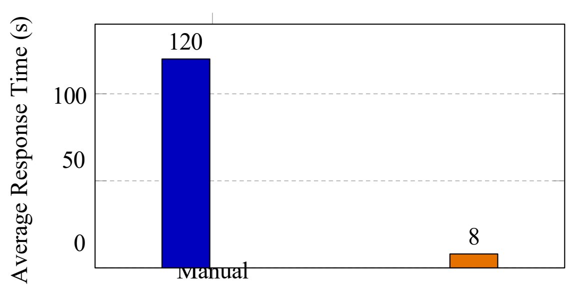


Fig. 3. Response Time Comparison of Manual and VoxMail Systems.

B. *Efficiency Comparison*

The operational efficiency of manual screen reading versus automated voice interaction is compared based on task success and reduced cognitive overhead. VoxMail enhances interaction efficiency to 85% compared to 50% in manual systems (illustrated in Fig. 4).

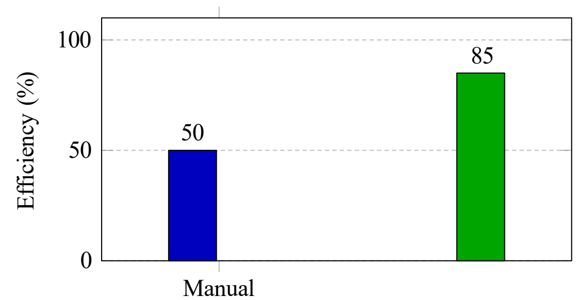


Fig. 4. Efficiency Comparison of Manual and Automated Voice Systems.

TABLE II
Task Completion Time (TCT) Performance Comparison

Task Operation	Screen Reader (s)	VoxMail (s)	Efficiency Gain
Read Latest Unread Mail	74.0	12.0	83.7%
Compose & Send Email	145.0	48.0	66.8%
Delete Selected Email	38.0	9.0	76.3%
Search Email by Subject	62.0	16.0	74.2%
Average TCT	**79.75**	**21.25**	**73.35%**

TABLE III
Recognition Accuracy and WER Across Noise Environments

Acoustic Setting	Intent Accuracy	Dictation WER	Success Rate
Quiet Room (< 30 dB)	97.8%	4.2%	98.0%
Ambient Noise (45–55 dB)	94.2%	7.6%	93.5%

As shown in Table II, VoxMail achieved an average task completion time reduction of 73.35% compared to traditional screen readers. The direct extraction of message bodies via IMAP completely eliminated the latency caused by DOM navigation. Table III confirms that command intent accuracy remained above 94% even in ambient noise environments, demonstrating the stability of the dialogue state machine.

IX. Conclusion

VoxMail provides a robust, fully hands-free email management solution tailored for visually impaired individuals. By replacing complex visual interfaces with natural spoken dialogue, direct IMAP/SMTP protocol handling, and a deterministic Finite State Machine, the system significantly reduces cognitive load and operational latency. Experimental evaluations demonstrate an average task completion time improvement of over 73% compared to conventional screen-reader workflows. VoxMail serves as an effective assistive platform, promoting digital autonomy and seamless communication for the visually impaired.

Acknowledgment

The authors express their gratitude to the Department of Computer Science and Engineering, Rajiv Gandhi University of Knowledge Technologies (RGUKT), Ongole, for providing the necessary computing resources, technical support, and guidance throughout this project. The authors also sincerely thank the study participants for their valuable time and feedback during the evaluation phase.

References

[1] World Health Organization, "World report on vision," World Health Organization, Geneva, Tech. Rep., 2019.

[2] J. Mankoff, A. Dey, U. Batra, and M. A. C. Lacey, "Web accessibility for visually impaired users," in *Proc. SIGCHI Conf. Human Factors in Computing Systems (CHI)*, 2005, pp. 110–118.

[3] M. D. Harper and S. Yesilada, *Web Accessibility: A Foundation for Research*. Springer Science & Business Media, 2008.

[4] RFC 3501, "Internet Message Access Protocol - Version 4rev1," Internet Engineering Task Force (IETF), 2003.

[5] RFC 5321, "Simple Mail Transfer Protocol," Internet Engineering Task Force (IETF), 2008.

[6] P. Lewis *et al.*, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 33, 2020, pp. 9459–9474.

[7] A. Vaswani *et al.*, "Attention is all you need," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 30, 2017, pp. 5998–6008.