IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

VOICE BASED EMAIL GENERATING SYSTEM FOR BLIND PEOPLE USING ARTIFICAL INTELLIGENCE.

[1]B.Balakrishna, [2] Dr.N.Jayalakshmi, [3] V. Lakshmi
[1] Assistant Professor, [2]Associate professor, [3] Assistant Professor
[1] Department Of Computer Applications,
Gayatri Vidya Parishad College Of Engineering (A) , Visakhapatnam, India-530048

Abstract: The communication among people is sharing of information. The Internet is widely used in almost all communication applications. E-mails are the most widely used and reliable way to communicate with each other. The usage of email is quite easy for regular users but when it comes to the visually impaired persons they are unable to access the system. for those visually impaired people, we are presenting the voice based email system under the principle of voice controlling. Therefore, a Speech completely messaging device is suggested using Py but instead Ai to save time for externally examined people. The system is totally based on voice reaction which will make it easy to use and effective to utilize. The people who are visually impaired persons are easy to access this voice based email system and can able to listen the all mails present in the inbox in form of voice. This will provide easy accessibility to the visually impaired persons in the world.

I. Introduction

1.1 Introduction:

Vision-impaired people cannot use the normal mail methods that the whole world can use. To make this program perfect for vision-impaired people, various technologies provided to them such as display reader, automatic speech viewer, text to text, text to speech etc. This technology is of little use to those people as they have failed to provide it the correct answer as usual. The main purpose or objective of creating a voice-based email for vision-impaired people is to help access emails easily and effectively with just a click of a mouse. This app is based on the use of STS converters & TTS converters which helps in controlling the email accounts by using only voice & users will be able to read, send the mail & many others useful things to anyone. This system will notify the user through voice to perform a specific task & users will give reply according to their needs.

1.2 TERMINOLOGY

The use of Speech-to-Text (STT) and Text-to-Speech (TTS) technologies is pivotal in enabling individuals with vision impairments to access and utilize email services effectively. STT technology converts spoken words into text, facilitating easy email composition. On the other hand, TTS technology provides voice output of emails received, reading out the subject and body to the user.

Contrary to the misconception that only visually capable individuals can use computers and the internet effectively, technologies employed on the internet can be immensely beneficial for vision-impaired individuals as well. One such technology involves a third-party assistance system, where a vision-impaired person dictates the email content to a third party. The third party then sends the composed message on behalf of the vision-impaired individual to the desired recipient. This system effectively bridges the accessibility gap for blind individuals, allowing them to send emails without hindrance.

This website serves as a solution to empower individuals with vision impairments to independently send emails through voice commands. Users simply need to articulate their intentions, and the system executes the corresponding actions, eliminating the need for physical input. The interface is designed to be user-friendly, requiring only mouse clicks for navigation, thus bypassing the need for users to memorize keyboard layouts or alphabets' locations on the computer.

By leveraging speech recognition and voice response technology, this website streamlines email usage for vision-impaired individuals, enabling tasks such as sending, reading, and managing emails with ease. The platform prioritizes simplicity and accessibility, ensuring that vision-impaired users can interact with email systems seamlessly and independently.

1.3 OBJECTIVES

A voice-based emailing system is being developed to empower visually impaired and illiterate individuals in utilizing everyday technologies such as sending and receiving emails. This system includes features like E-Mail Compose, Inbox Access, and reading/listening to mails. It incorporates a login system that eliminates the need for keyboards, relying instead on Speech recognition technology. Furthermore, the application has been adapted for use on a touch-enabled standalone system, featuring Blind-friendly GUI forms designed for easy accessibility. The "Blind-friendly Email System" prototype is thus being developed to enhance the digital experience for individuals with visual impairments and literacy challenges.

NEED OF THE STUDY.

Blind individuals encounter challenges when using standard email systems designed for visual interactions, rendering them inaccessible to visually impaired users. While technologies like screen readers and braille keyboards are available, they often fall short in providing a seamless user experience. This project aims to rectify this by developing a Voice-Based Email Generating System using Artificial Intelligence. The objective is to empower blind users with a voice-controlled interface, making email usage inclusive and granting independent access to vital communication tools. This initiative aims to narrow the digital divide, promoting equal participation for individuals with visual impairments in today's digital landscape.

2. RESEARCH METHODOLOGY

2.1 Research Design

The research follows an applied research methodology, focusing on the development and implementation of a voice-based email system for visually impaired individuals. The system is designed to integrate Speech-to-Text (STT) and Text-to-Speech (TTS) technologies to enhance accessibility. The methodology consists of requirement analysis, system design, implementation, testing, and evaluation.

2.2 Data Collection

Data collection was conducted through primary and secondary sources.

- **Primary Data:** Surveys and interviews were conducted with visually impaired individuals to understand their challenges in using traditional email systems. Feedback from accessibility experts was also gathered to refine the system design.
- **Secondary Data:** Literature reviews on existing assistive technologies, voice recognition systems, and user accessibility studies were analyzed to identify gaps and potential improvements.

2.3 System Development

The system was developed in multiple phases:

1. **Requirement Analysis:** Identifying key functionalities needed by visually impaired users, such as voice command email composition, inbox access, and email management.
2. **Technology Selection:** Choosing suitable STT and TTS tools such as Google Speech API, IBM Watson, or Microsoft Azure Cognitive Services for voice processing.
3. **System Design:** Developing a user-friendly interface with minimal visual dependency and seamless voice interaction.
4. **Implementation:** Coding and integrating the STT and TTS modules into a web-based platform or standalone application.
5. **Testing and Debugging:** Ensuring functionality through unit testing, system testing, and user testing.

2.4 Tools and Technologies Used

- **Programming Languages:** Python, JavaScript (for web-based applications)
- **Frameworks:** Flask/Django for backend development
- **APIs:** Google Speech API, IBM Watson, or Microsoft Azure for speech recognition
- **Database:** MySQL or Firebase for storing user emails and authentication details
- **User Interface:** HTML, CSS, JavaScript for front-end development with accessible design principles

2.5 User Testing and Evaluation

User testing was conducted in collaboration with visually impaired individuals to assess:

1. **Usability:** Ease of navigation and interaction with voice-based commands.
2. **Accuracy:** The precision of the STT and TTS functionalities in various environmental conditions.
3. **Efficiency:** Speed and effectiveness in composing, sending, and receiving emails.
4. **Satisfaction:** Overall user experience and satisfaction with the system.

2.6 Ethical Considerations

- Ensuring informed consent from visually impaired participants during testing.
- Maintaining data privacy and security for email content.
- Compliance with accessibility standards and guidelines such as WCAG (Web Content Accessibility Guidelines).

2.7 Limitations and Future Enhancements

- **Limitations:**
  - Potential inaccuracies in voice recognition due to varying accents or background noise.
  - Limited accessibility in areas with poor internet connectivity.
- **Future Enhancements:**
  - Integration of multilingual support for diverse user needs.
  - Development of a mobile application for increased accessibility.
  - AI-driven predictive text suggestions to enhance email composition efficiency.

By following this methodology, the study aims to develop a fully functional and accessible voice-based email system tailored to the needs of visually impaired users.

3. TECHNOLOGIES USED:

HTML(Hypertext Markup Language):
Latest Version: HTML5
HTML, serves as the fundamental markup language for creating structured web pages. It provides a set of elements and attributes to define the content and layout of web documents, including headings, paragraphs, images, links, forms, and more. HTML5 also introduces new features such as semantic elements (like `<header>`, `<nav>`, `<article>`, `<footer>`), multimedia support (audio and video), and enhanced form controls for improved user experience and accessibility on modern web browsers and devices.

CSS (Cascading Style Sheets):
Latest Version: CSS3
Style sheet language for web page styling.
CSS3, the latest iteration of Cascading Style Sheets, is crucial for styling HTML elements and controlling their appearance on web pages. It offers powerful styling capabilities such as color management, typography control, layout design (using Flexbox and Grid), animations, and responsive design techniques. With CSS3, developers can create visually appealing, responsive, and interactive web interfaces across different devices and screen sizes, enhancing the overall user experience.
c) JavaScript:
Latest Version: ECMAScript 2021 (ES12)
High-level programming language for web development.
JavaScript, in its latest ECMAScript 2021 version (ES12), remains a cornerstone of web development, providing essential functionalities for client-side scripting, interactivity, and dynamic content generation on web pages. ES2021 introduces new language features such as String.prototype.replaceAll(), logical assignment operators (&&=, ||=), and Promise.any() for asynchronous programming. JavaScript frameworks and libraries like React, Vue.js, and Node.js leverage ES2021 features to build modern web applications and server-side solutions efficiently.
d) Python:
Python is an ideal choice for your voice-based email system for blind users due to its simplicity, versatility, and extensive AI capabilities. Its clean syntax and rich libraries streamline development, enabling efficient voice input recognition, natural language understanding, and email management functionalities. Python's robustness ensures reliable performance, while its compatibility with various platforms ensures accessibility for users across devices. Additionally, Python's AI frameworks like Tensor Flow and PyTorch empower advanced features such as voice biometrics for authentication and personalized email generation based on user preferences. Leveraging Python in your project fosters rapid development, scalability, and integration with AI technologies critical for enhancing user experience.

Django Framework:

Django is a high-level Python web framework renowned for its simplicity, scalability, and versatility, making it an excellent choice for your voice-based email system for blind users. With Django, you can swiftly develop robust backend components that handle user authentication, email management, and AI integration seamlessly. Django's built-in features such as ORM (Object-Relational Mapping), admin interface, and templating engine simplify development tasks by facilitating efficient database operations, administrative tasks, and dynamic content rendering. Its adherence to the DRY (Don't Repeat Yourself) principle promotes code reusability and maintainability, ultimately reducing development time and effort. Moreover, Django's strong emphasis on security, including protection against common web vulnerabilities, ensures that sensitive user data remains safe. Its extensive documentation, active community support, and a vast ecosystem of plugins and libraries further enhance its appeal for building sophisticated AI-powered applications like your voice-based email system.

Speech Recognition:
In our AI-powered voice-based email system for the visually impaired, we leverage Google Speech Recognition API. This technology processes audio signals, converting spoken commands into text for email operations. It incorporates advanced algorithms to filter ambient noise, extract speech features like pitch and duration, and match them against a vast database for accurate transcription. The seamless integration of Google Speech Recognition API ensures precise and efficient speech-to-text conversion, empowering users to navigate and manage emails effortlessly through spoken interactions, thereby enhancing accessibility and usability in digital communication for visually impaired individuals.

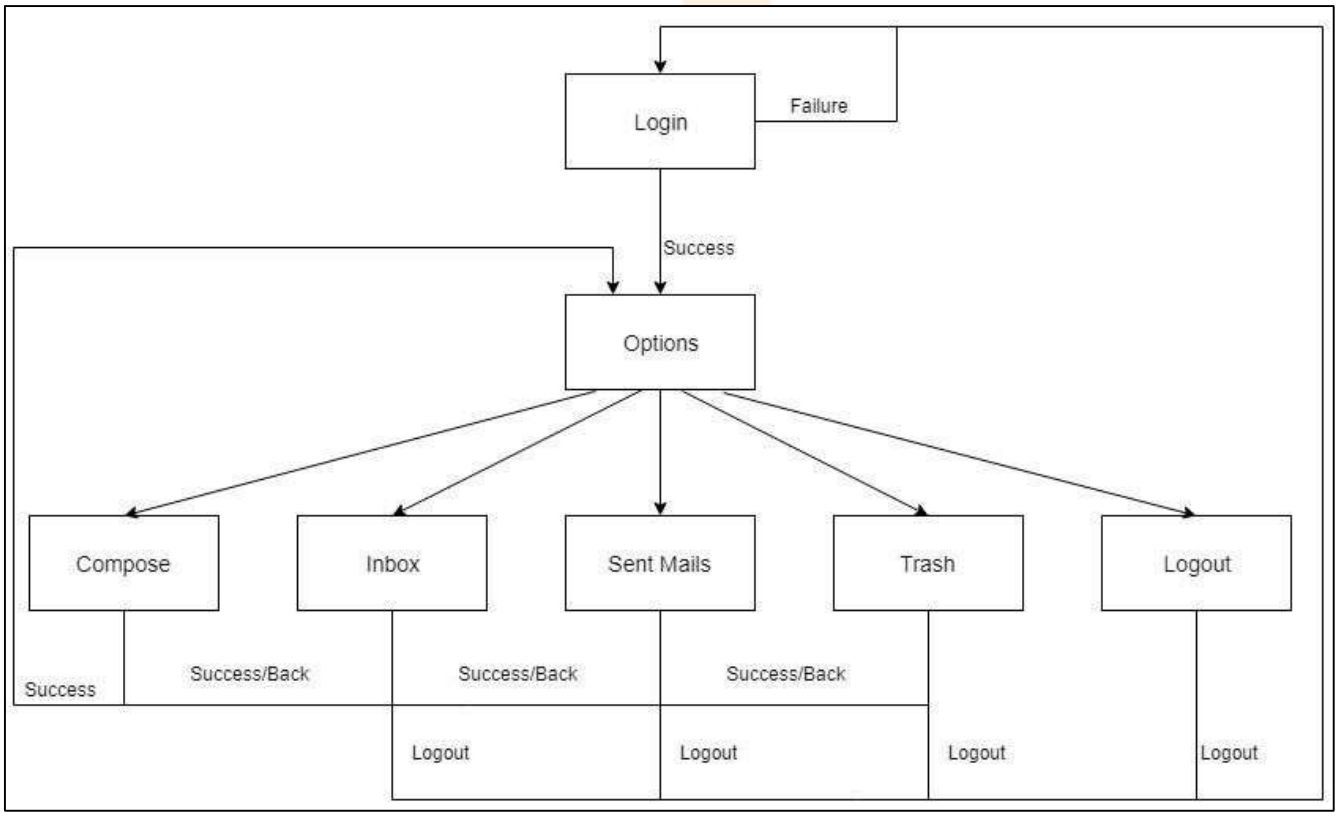
Natural Language Processing:
Our voice-based email system for the blind integrates Natural Language Processing (NLP) with Interactive Voice Response (IVR), powered by AI. NLP algorithms understand and interpret spoken commands, enabling intuitive email interactions. IVR guides users through voice prompts, facilitating email operations without visual cues. Through AI, NLP enhances comprehension of user input, ensuring accurate execution of email tasks. This fusion of NLP and IVR empowers blind users to effortlessly manage emails using spoken language, enhancing accessibility and usability in our innovative email system.

Text-to-Speech:
Text-to-Speech (TTS) technology is integral to our voice-based email system for blind users utilizing AI. This technology enables computers to transform written text into spoken language, allowing users to listen to their emails rather than reading them. The TTS process involves various stages, including text preprocessing, linguistic analysis, prosody modeling, and waveform generation. During prosody modeling, natural intonation, stress, and rhythm are added to the synthesized speech, making it sound more human-like. Waveform generation then converts the prosody model into an audio waveform, enabling playback of the synthesized speech as human-like speech. This TTS capability is crucial in our system, providing blind users with an accessible and efficient means of interacting with email content through spoken language.

SMTP and IMAP Protocols:
SMTP (Simple Mail Transfer Protocol) and IMAP (Internet Message Access Protocol) are two of the most commonly used email protocols. SMTP is a protocol used for sending email messages between servers over the Internet When a user sends an email message, it is transmitted from their email client to their email server using SMTP. The email server then uses SMTP to transmit the message to the recipient's email server, which in turn delivers it to the recipient's email client. SMTP is a reliable and efficient protocol for sending email messages over the Internet.
IMAP, on the other hand, is a protocol used for accessing and managing email messages stored on a remote server. When a user logs into their email account using an IMAP client, such as an email app, the client uses IMAP to connect to the email server and retrieve the user's email messages. IMAP allows users to manage their email messages directly on the server, enabling them to access their messages from multiple devices and clients.

SYSTEM IMPLEMENTATION

System Implementation Diagram



4. System Implementation Explanation:
Users must log in using their correct email ID and password. Upon successful authentication, they gain access to the menu page offering five options:
Compose: Allows users to create and send emails to any recipient.

Inbox: Enables users to check for unread messages and search for specific emails within the inbox.
Sent: Provides access to sent emails, allowing users to search and read the content of sent messages.
Trash: Navigates to the trash page, where users can view deleted emails.
Logout: Logs the user out of their Gmail account, returning them to the login page.
All functionalities are accessible solely through voice commands.

IMPLEMENTATION

We are using g-mail API (Application programming interface) and speech-to-text, and text-to-speech APIs. In this project, we are using the following algorithms

4.1 Speech-to-Text Conversion:

Speech-to-text converters analyze spoken sounds, filter them, and transform them into readable digital formats, often saving the recognized text in a file. This software, leveraging voice recognition, listens to audio inputs and provides a transcribed transcript that can be edited on a device.

Here's a detailed breakdown of how speech-to-text conversion works:

Sound Recognition: As someone speaks, their words create vibrations. Speech-totext technology detects these vibrations and converts them into digital signals using an analog-to-digital converter.

Analog-to-Digital Conversion: The converter processes audio files, meticulously measuring waveforms and filtering out irrelevant noise to isolate relevant sounds.

Segmentation and Phoneme Matching: The captured sounds are segmented into tiny time intervals (hundredths or thousandths of seconds) and matched to phonemes, which are sound units differentiating words in a language. For instance, English has about 40 phonemes.

Mathematical Modeling: A mathematical model processes these phonemes, comparing them against known sentences, words, and phrases stored in a network.

Text Generation: Based on the analysis and mathematical modeling, the software generates the most probable version of the spoken audio as text, which can then be presented in a textual format or used as input for computer-based commands.

This process involves a sophisticated machine-learning model that employs linguistic algorithms to decipher spoken words into text using Unicode characters. The intricate steps ensure accurate transcription, making speech-to-text conversion a vital tool in modern communication and accessibility solutions.

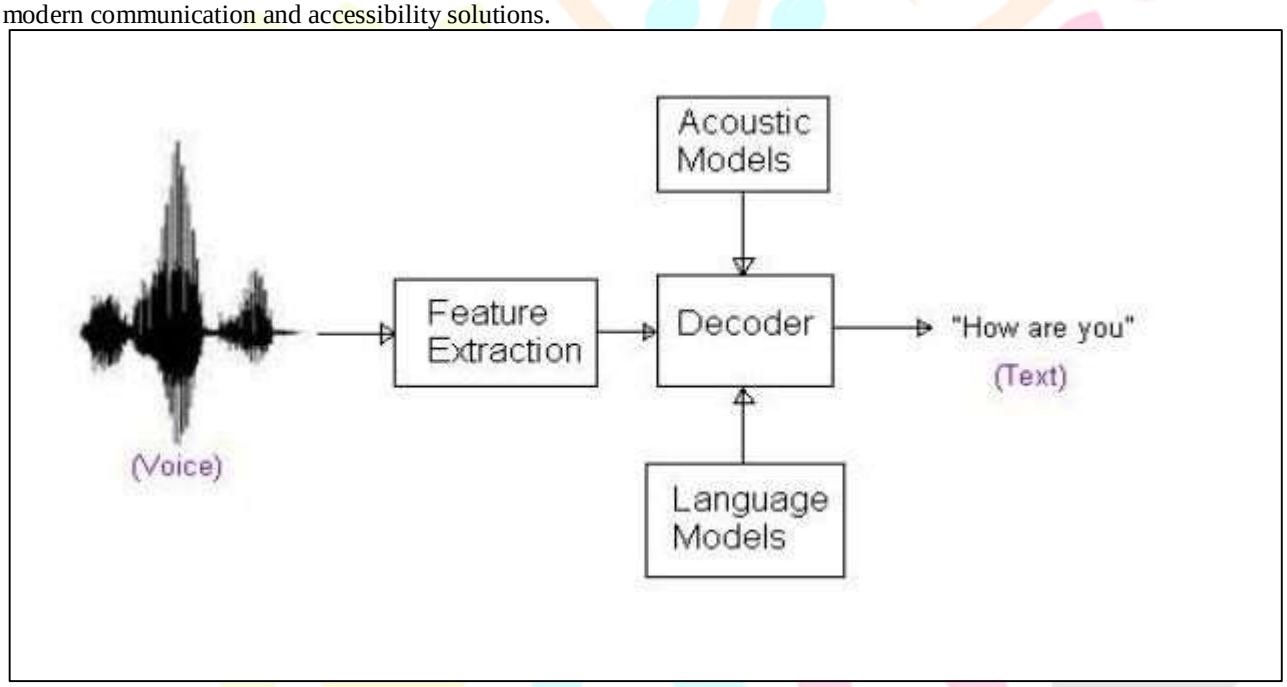


Fig 4.1 Speech -to -Text Diagram

Voice response analysis drives all commands in the email system, from logging in to composing and sending emails, and managing the inbox. Speech Recognition is the key tool for converting spoken words into text, enabling seamless interaction. This library simplifies audio input retrieval, eliminating the need for complex microphone access and audio file processing scripts. By acting as a wrapper for various speech APIs, including the Google Web Speech API, SpeechRecognition offers unparalleled flexibility. The inclusion of a default API key for the Google Web Speech API further streamlines the setup process, making speech-to-text conversion swift and efficient.

4.1.2 Text-to-Speech Conversion:

Text-to-speech (TTS) technology is a branch of computer science focused on designing systems that convert written text into spoken speech. It's an essential part of speech synthesis, which involves creating artificial speech from written text. TTS systems typically consist of two main modules: image processing and voice processing.

Image Processing Module

The image processing module involves analyzing and interpreting visual data, such as text on a screen or document. This module identifies and extracts the text that needs to be converted into speech. It handles tasks like character recognition, text segmentation, and formatting analysis.

Voice Processing Module

The voice processing module is responsible for generating the actual speech output based on the text input received from the image processing module. This module uses various techniques and algorithms to create natural-sounding speech, including phoneme synthesis, diaphone synthesis, and voice recording/playback.

Functions of Text-to-Speech Technology

Conversion of Written Text to Speech: The primary function of TTS technology is to convert written text into spoken speech. This enables computers to communicate audibly with users, enhancing accessibility for individuals with visual impairments and enabling voice based interactions in applications.

Synthesis Methods:

Simple Voice Recording: Basic recording and playback of pre-recorded voice segments.

Phoneme Synthesis: Breaks speech into phonemes (basic linguistic units) and reassembles them to generate fluent speech.

Diaphone Synthesis: Utilizes approximately 400 diaphones for splitting phrases and creating natural-sounding speech.

Man-Machine Interaction: TTS technology plays a crucial role in man-machine interaction, allowing users to interact with computers through speech. It supports interactive voice response (IVR) systems, voice assistants, and other applications where spoken output is required.

Accessibility: TTS technology promotes accessibility by providing auditory feedback for visually impaired users. It enables screen readers to convert text on a screen into spoken words, making digital content more accessible.

Automatic Speech Synthesis Advancements

Automatic speech synthesis is rapidly evolving, driven by advancements in speech science and engineering. It is considered the next major innovation in man-machine interaction after speech recognition (TTS). Developers are encouraged to explore the capabilities of TTS technology, including its ability to generate natural-sounding speech from plain or marked-up text inputs.

Conclusion

Text-to-speech technology is a fundamental component of speech synthesis, enabling computers to convert written text into spoken speech. Its applications range from accessibility features for visually impaired users to interactive voice response systems and voice-enabled applications. Understanding the capabilities and methods of speech synthesis is essential for developers working with speech technologies.

Fig 4.2 Text -to-Speech Diagram

It powers the email service to read aloud the text to the user in the form of audio. Throughout the project, this function is used to provide the user with voice commands to guide him to where on the website he is currently and what actions can he perform.

4.1.3 SIMPLE MAIL TRANSFER PROTOCOL(SMTP)

Email remains one of the most critical services on the internet today, facilitating communication across various web systems. SMTP (Simple Mail Transfer Protocol) is a core protocol used for transmitting emails, while POP (Post Office Protocol) or IMAP (Internet Message Access Protocol) are used for retrieving emails on the recipient's end.

SMTP operates at the application layer of the OSI model and is integral to the process of sending emails. When a user initiates an email, they establish a TCP (Transmission Control Protocol) connection with an SMTP server. This server, usually listening on port 587, handles the transmission of emails.

SMTP Process

Connection Establishment: The sender opens a TCP connection to the SMTP server.

Mail Transmission: Once the connection is established, the sender can send the email.

Request-Response Dialogue: A simple dialogue occurs between sender and receiver processes, where the sender provides email addresses (sender and recipient) and transmits the email message.

Message Format: Emails follow the RFC 822 format, including headers (e.g., From, To, Subject) and the message body. SMTP Models

SMTP employs two primary models for email transmission:

End-to-End Method: In this model, the SMTP client communicates directly with the destination host's SMTP server to deliver the email. The email is held until successfully copied to the recipient's SMTP server.

Store-and-Forward Method: Here, emails can pass through intermediary hosts before reaching the final destination. Each successful transmission signifies the email has been forwarded to the next host until it reaches the recipient's server.

SMTP plays a crucial role in email communication, ensuring reliable transmission of messages across networks. Its versatility in handling both end-to-end and store-and-forward methods makes it adaptable for various organizational and network configurations.

4.1.4 Sending Email using smtplib
The automation of sending emails in Python is facilitated by the smtplib module. This module provides the SMTP class, which allows us to connect with mail servers and send emails. The SMTP class represents an SMTP client session object that can be used to transmit messages to any internet-connected machine following the SMTP format. SMTP is commonly used to establish a connection with a mail server and send messages.
Key Concepts
SMTP Server Connection: You can connect to an SMTP server by passing the mail server's hostname and port to the constructor, or by using the connect () method explicitly.

host: Represents the hostname of the SMTP server. This can be an IP address or a domain name like gmail.com or outlook.com.
port: Specifies the port number where the SMTP server listens for requests. The default SMTP port is 25.
local_hostname: If the SMTP server is running on the local machine, you can use localhost for this argument.
SMTP.sendmail() Method: This method is used to send emails and takes the following parameters:
sender: Email ID of the sender.
receivers: Email IDs of the recipients.
message: The message formatted according to RFC822 standards.
python
Note: The message should be a properly formatted RFC822-compliant message. Headers like Subject, From, To, and Date must be included in the message content. Additional Headers: It's important to manually add headers like From (sender's email) and To (recipient's email) to the message content.

4.1.5 Reading Emails from Gmail using Python
Python offers great flexibility in automating tasks such as reading emails from Gmail. This functionality can be particularly useful for users who may have visual impairments, as they can rely on text-to-speech converters to listen to their emails.

To achieve this, we'll need to utilize a few key modules and functionalities: Mail Server Credentials: You'll need your mail server details (e.g., imap.gmail.com) along with your Gmail username and password.
Login to Gmail Account: We've already discussed how to log in using a Python script.
IMAP Server: For reading emails, we'll primarily work with imaplib, specifically the IMAP4 class.
Using imaplib Module - IMAP4 Protocol
The imaplib module provides classes and methods to interact with IMAP4 mail servers. Here are some key classes and methods within imaplib that we'll utilize:
IMAP4: The base class representing the IMAP4rev1 protocol. It contains methods for various IMAP commands, such as login, select, fetch, expunge, etc.

5. Accessing the webpage:

5 .1 User Login Page:

E-mail system login module is altogether voice based email generating system, requiring the user login with mail id and used password, Fig 7. At that point when a client talks about their mail id, password, the user login framework illuminates that and inquires as to whether they are right. Assuming they answer that they will be, they can utilize those subtleties to go on with their activities.

Assuming the client answers that anything is inaccurate, the user login framework says them again for their user email id and user password. The UI for the login page is developed using html,CSS , js and Ajax modules in python. and user email id and password

Fig 5.1 Login page

5.2 Menu page/option choosing page:

Once the login has been successfully completed the menu page will be open the user must click by mouse anywhere on "SCREEN" to perform the operation of mail. The email system prompts the user to choose whether they want to open the compose page, inbox page, sent page or trash page if none of this option users can logout the menu page giving voice command.

Fig 5.2 Menu page

5.3 Compose page:

This is among the primary services are given by email. The email prompts the user to compose an email after the option chosen by the user, (Fig.7 and Fig8.). Compose module is entirely voicebased, requiring RECIPIENT'S EMAIL ID to compose mail and also asks for more RECIPIENT'S EMAIL IDs. When a user speaks their RECIPIENT'S EMAIL ID , the system informs them and asks if they are correct. If the user responds that they are, they can use those details to continue with operations. When the user replies incorrect, the compose system says them once more of their RECIPIENT'S MAIL ID. Once this confirmed system prompts the user to speak SUBJECT and BODY messages. Once the user completes the subject and body message system informs the user to add attachment. Attachment includes audio record or file name with extension such as jpg, png or pdf. The recognized receiver email, subject, body content through speech is shown in corresponding text boxes.

Fig 5.3 Compose page

5.4 Inbox page:

Users can listen to unread emails in the Inbox page, respond to a particular email, and forward it to others by clicking with the mouse. They can also perform operations depicted in Figures 8 and 9. Initially, users are prompted to identify the precise user whose communication they want to read. The system then reads the subject contents of the message using the Google Text-to-Speech (gTTs) method. If they wish to read a specific email, they can choose the search option. Opting for the unread option makes the system read the subject content of the message. Subsequently, users are prompted to reply, forward, or return to the menu page. The back option allows them to navigate back to the menu page.

Fig5.4 Inbox page Inbox

Sent page:

This module is used to listen to sent mails. This module includes two options: search and back. Search option is used to search the sent mail by using a specific email id,our application takes input from the user according to the input application will do. Back option helps to go back to the menu page.

Fig 5.5 Sent page

Trash page:

This module is used to listen to the deleted mails, Fig.11. The deleted mails can be listened to by using a specific mail id

Fig 5.6 Trash page

CONCLUSION

The application supports visually impaired people by providing easy access to email. It helps in overcoming the disadvantages that visually impaired people have encountered in the past. Creating a voice-controlled email system for blind people using AI is a big step forward in making communication easier for everyone. With smart technology that understands and talks back, blind users can easily write, send, and organize emails just by speaking. This system is easy to use and works smoothly with the email services we already have, making blind users more independent and efficient in digital communication. As technology gets better, projects like this show how AI can make the world more inclusive and help people from different backgrounds connect and communicate better. Our voice-based email system for the blind includes enabling users to attach various types of files such as images, videos, documents, and more, enhancing communication and accessibility. Currently, due to Gmail restrictions, only one user can access the system using an App password; however, multiple users will be able to use the system in the future. To enhance security, we can incorporate facial authentication.

REFERENCES:

[1] Guillermo Arturo Hernández Tapia, Ana Lilia ReyesHerrera. "E-mail management system for blind people in Spanish language". In Interacción'17: XVIII International Conference on Human Computer Interaction Cancun Mexico September,2017.

[2] Amritha Suresh, Binny Paulose, Reshma Jagan and Joby George, "Voice Based Email for Blind". International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET) - Volume 2, Issue 3, 2016, pp. 93-97.

[3] Milan Badigar, Nikita Dias, Jemima Dias and Mario Pinto, "Voice Based Email Application For Visually Impaired. International Journal of Science Technology & Engineering (IJSTE) - Volume 4, Issue 12, June 2018, pp. 166-170.

[4] Pranjal Ingle, Harshada Kanade and Arti Lanke, "Voice Based email System for Blinds". International Journal of Research Studies in Computer Science and Engineering (IJRSCSE)- Volume 3, Issue 1, 2016, pp. 25-30.

[5] Bishal Kalita and Santosh Kumar Mahto, "Voice Based Email for Blind People". International Journal of Engineering Science and Computing (IJESC) - Volume 9, Issue 10, October-2019, pp. 2378923799.

[6] Saurabh Sawant, Amankumar Wani, Sangharsh Sagar, Rucha Vanjari and M R Dhage, "Speech Based Email System for Blind and Illiterate People".

[7] Dudhbale, P., Wankhede, J.S., Ghyar, C.J., and Narawade, P.S., "Voice Based System in Desktop and Mobile Devices for Blind People". International Journal of Scientific Research in Science and Technology, 4, 2018, pp. 188-193.