**ABSTRACT**

Assistance for mail is a service that allows users to access and manage their email messages through voice commands. Instead of reading or typing emails, users can listen to their messages and respond using spoken instructions. This can be particularly helpful for individuals who may have difficulty reading or typing, or when hands-free communication is preferred. The system typically converts voice messages into text and vice versa, facilitating a seamless interaction between the user and their email. It often involves features like voice recognition technology to understand spoken commands and convert them into actionable tasks, such as reading emails, composing replies, or organizing messages. This type of service aims to enhance accessibility and convenience, providing an alternative means of email management for users who prefer or require voice-based communication. It can be beneficial in various scenarios, including while driving, multitasking, or for people with visual or motor impairments.

The main features of this project are:

• Minimal typing

• Implementing voice command

• Useful for blind users

• Ease of use

• Faster writing

• Multi-tasking

• Helpful to make meeting transcripts

• Easier to take notes

**REFERENCES**

**Published in:**[2021 5th International Conference on Intelligent Computing and Control System (ICICCS)](https://ieeexplore.ieee.org/xpl/conhome/9432068/proceeding)

**Date of Conference:**06-08 May 2021

**DOI:**[10.1109/ICICCS51141.2021.9432206](https://doi.org/10.1109/ICICCS51141.2021.9432206)

**Publisher:**IEEE

**Conference Location:**Madurai, India

**CHAPTER 1**

**INTRODUCTION**

|  |
| --- |
| **1.1 GENERAL INTRODUCTION** |
| Nowadays everybody is connected to the internet. It is an inseparable part of our life. It contains all the information of individuals and day to day history. Communication and interaction are possible mainly through the internet. Out of many technologies Email is the most common way of communication primarily for business and educational perspective. Although not all use net and have access. This is due to lack of facilities, knowledge and money. The users should have vision to see and read the screen. For the physically and visually challenged people net is like a useless and unfamiliar thing. However, there are technologies like TTS (text-to-speech), and ASR (automated-speech recognition) screen readers, but they do not provide full accuracy and efficiency to the impaired people to use the internet. As communication is needed in everyone’s life, net facilities should be available for everyone. |
| **1.2 GOAL OF THE PROJECT** |
| This project is based on voice email which will help impaired people to communicate. Even the naïve users can access and send emails to anyone. It is completely based on the voice response. There is no prior knowledge required to use this. Everything is automatically prompting the only thing is to give the responses of the voices to perform the actions. |

**CHAPTER 2**

**LITERATURE SURVEY**

|  |
| --- |
| **2.1 STUDY OF SIMILAR WORK** |
| There are lots of research works and prototypes available for voice assistant e-mail services but there is an unavailability for providing such services as commercial application and mostly done through by using the accessibility feature of corresponding operating system. |
| **2.1.1 EXISTING SYSTEM** |
| According to Email Statistics Report, 2014-2018 by a technology market research firm Palo Alto, CA, USA, there are a total of 4.1 billion email accounts created until 2014 to over 5.2 billion accounts in end of 2018 and making it one of the most used form of communication. The research, by the Vision Loss Expert Group (VLEG), shows that worldwide 253 million people are either blind or visually challenged that is, around 253 million people are not aware of how to use Internet or E-mail. Existing systems of today are basically applications that provide accessing of emails benefits to its users via web facilities. Making email widely used communication form. The existing systems do not support any voice commands or audio facilities and therefore it is not suitable for visually challenged people. |
| **2.1.2 DRAWBACK OF EXISTING SYSTEM** |
| * Lack of a good Graphical User interface. * Only applications are available. * There is an unavailability of generalized applications. |

**CHAPTER 3**

**OVERALL DESCRIPTION**

|  |
| --- |
| **3.1 PROPOSED SYSTEM** |
| There can be many proposed systems for the above problems. The present proposed system is completely based on the user’s accessibility and easiness of the email system. It is completely useful for both types of people whether they are able or disable. The current system is not available for both types of people in the market. This system is focused on the user’s behaviour and their perspective view. It is accessible to all types of people including illiterate people and even new users. The system uses IVR (interactive voice response) in order to interact with the users. It makes the system realistic and natural way to impart the messages and feelings. |
| **3.2 FEATURES OF PROPOSED SYSTEM** |
| The feature of proposed system comes when the users interact with the system it will automatically generate the voices to do the actions. There is a step by step process to perform the actions. The users have to hear the voices and respond for the desired actions. The main advantage of the system is that there is no application of multiple keywords, only tab button operation is required. Once the system is started every step is voiced based the users have to wait and respond for the desired actions. The users do not have to worry to have the single key operations. Every functionality is voiced based if one action is performed it conveys the message of completion to the users. |
| **3.3 FUNCTIONS OF PROPOSED SYSTEM** |
| 1. Real Time Voice Assistant For Composing Mail and Reading Inbox 2. Sending Email 3. Reading Inbox mails. 4. To Confirm The Intended Mail Credentials With The Help Of Voice Assistant |
| **3.4 REQUIREMENTS SPECIFICATION** |
| **1. Accuracy**  The proposed system should be accurate on generating results based on given inputs.  **2. Speed**  The proposed system should be in real time for generating results.  **3. Flexible**  The proposed system should be flexible to new updates and patches in near future.  **4. Good Interface**  The proposed system should maintain good interface even after upgradations. |
| **3.5 FEASIBILITY STUDY** |
| Feasibility Study in Software Engineering is a study to evaluate feasibility of proposed project or system. Feasibility study is one of stage among important four stages of Software Project Management Process. As name suggests feasibility study is the feasibility analysis or it is a measure of the software product in terms of how much beneficial product development will be for the organization in a practical point of view. Feasibility study is carried out based on many purposes to analyse whether software product will be right in terms of development, implantation, contribution of project to the organization etc. In our proposed system the product is feasibility can be achieved in all four aspects Technical Operational, Economical and Behavioural. |
| **3.5.1 TECHNICAL FEASIBILITY** |
| In Technical Feasibility current resources both hardware software along with required technology are analysed/assessed to develop project. This technical feasibility study gives report whether there exists correct required resources and technologies which will be used for project development. Along with this, feasibility study also analyses technical skills and capabilities of technical team, existing technology can be used or not, maintenance and upgradation is easy or not for chosen technology etc. In this proposed system technical feasibility is achieved according to above criteria. |
| **3.5.2 OPERATIONAL FEASIBILTIY** |
| In Operational Feasibility degree of providing service to requirements is analysed along with how much easy product will be to operate and maintenance after deployment. Along with this other operational scopes are determining usability of product, determining suggested solution by software development team is acceptable or not etc. The Operational feasibility can be ensured by the proposed system. |
| **3.5.3 ECONOMICAL FEASIBILITY** |
| Economic feasibility the most important and frequently used method for evaluating the effectiveness of the proposed system. It is very essential because the main goal of the proposed system is to have economically better results along with increased efficiency. Cost benefit analysis is usually performed for the expected from the proposed system. Since the organization is well equipped with the required hardware, the project was found to be economically feasible and the users who possess a device supports Windows operating system can easily use it. |
| **3.5.4 BEHAVIOURAL FEASIBILITY** |
| The proposed system satisfies behavioural feasibility because the system is providing with good and minimalistic GUI which can easily be understand for any end users and it’s encapsulates the conversion procedure from the users. Hence it’s easier to operate the system with ease. |

**CHAPTER 4**

**OPERATING ENVIRONMENT**

|  |
| --- |
| **4.1 HARDWARE REQUIREMENTS** |
| **1. Processor:** Dual Core 1.60 GHz or higher  **2. Hard disk:** 500 GB  **3. RAM:** 4GB  **4. Monitor:** 17” Colour Monitor  **5. Mouse:** Microsoft  **6. Keyboard:** Microsoft multimedia keyboard |
| **4.2 SOFTWARE REQUIREMENTS** |
| **1. Operating System:** Windows 8.1 Pro or higher  **2. Framework:** Microsoft .Net Framework  **3. Environment:** PyCharm Community Edition 2022.2  **4. Language:** Python 3.10  **5. Documentation:** Microsoft Word 2010 or higher |
| **4.3 TOOLS AND PLATFORMS** |
| **4.3.1 PYCHARM**  PyCharm is a dedicated Python Integrated Development Environment (IDE) providing a  wide range of essential tools for Python developers, tightly integrated to create a convenient environment for productive Python, web, and data science development.  **4.3.2 PYTHON 3.10**  Python is an interpreted, object-oriented, high-level programming language with dynamic  semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together.  Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance.  Python supports modules and packages, which encourages program modularity and code  reuse. The Python interpreter and the extensive standard library are available in source or  binary form without charge for all major platforms, and can be freely distributed.  **4.3.3 SPEECH RECOGNITION**  Speech recognition is a machine's ability to listen to spoken words and identify them. You can then use speech recognition in [Python](https://www.simplilearn.com/learn-the-basics-of-python-article) to convert the spoken words into text, make a query or give a reply. You can even program some devices to respond to these spoken words. You can do speech recognition in python with the help of computer programs that take in input from the microphone, process it, and convert it into a suitable form.  Speech recognition seems highly futuristic, but it is present all around you. Automated phone calls allow you to speak out your query or the query you wish to be assisted on; your virtual assistants like Siri or Alexa also use speech recognition to talk to you seamlessly.    **4.3.4 PYTTSX3**  pyttsx3 is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline and is compatible with both Python 2 and 3. An application invokes the pyttsx3.init() factory function to get a reference to a pyttsx3. Engine instance. it is a very easy to use tool which converts  the entered text into speech. The pyttsx3 module supports two voices first is female and the second is male which is provided by “sapi5” for windows. It supports three TTS engines :   * *sapi5* – SAPI5 on Windows * *nsss* – NSSpeechSynthesizer on Mac OS X * *espeak* – eSpeak on every other platform   **4.3.5 SMTP**  Simple Mail Transfer Protocol (SMTP) is a protocol, which handles sending e-mail and routing e-mail between mail servers.  Python provides *smtplib* module, which defines an SMTP client session object that can be used to send mail to any Internet machine with an SMTP or ESMTP listener daemon.    **4.3.6 EMAIL**  The [email](https://docs.python.org/3/library/email.html#module-email) package is a library for managing email messages. It is specifically not designed to do any sending of email messages to SMTP ([RFC 2821](https://datatracker.ietf.org/doc/html/rfc2821.html)), NNTP, or other servers; those are functions of modules such as [smtplib](https://docs.python.org/3/library/smtplib.html" \l "module-smtplib" \o "smtplib: SMTP protocol client (requires sockets).) and [nntplib](https://docs.python.org/3/library/nntplib.html" \l "module-nntplib" \o "nntplib: NNTP protocol client (requires sockets). (deprecated)). The [email](https://docs.python.org/3/library/email.html#module-email) package attempts to be as RFC-compliant as possible, supporting [RFC 5322](https://datatracker.ietf.org/doc/html/rfc5322.html) and [RFC 6532](https://datatracker.ietf.org/doc/html/rfc6532.html), as well as such MIME-related RFCs as [RFC 2045](https://datatracker.ietf.org/doc/html/rfc2045.html), [RFC 2046](https://datatracker.ietf.org/doc/html/rfc2046.html), [RFC 2047](https://datatracker.ietf.org/doc/html/rfc2047.html), [RFC 2183](https://datatracker.ietf.org/doc/html/rfc2183.html), and [RFC 2231](https://datatracker.ietf.org/doc/html/rfc2231.html).  **4.3.7 EASYIMAP**  Simple IMAP Wrapper. It contains builds that are only available via PyPM. |

**CHAPTER 5**

**DESIGN**

|  |
| --- |
| **5.1 SYSTEM DESIGN** |
| System Design involves translating system requirements and conceptual design into technical specifications and general flow of processing. After the system requirements have been identified, information has been gathered to verify the problem and after evaluating the existing system, a new system is proposed.  System Design is the process of planning of new system or to replace or complement an existing system .It must be thoroughly understood about the old system and determine how computers can be used to make its operations more effective. System design sits at technical the kernel of system development. Once system requirements have been analysed and specified system design is the first of the technical activities-design, code generation and test- that required build and verifying the software. System design is the most creative and challenging phases of the system life cycle. The term design describes the final system and the process by which it is to be developed.  System design is the high-level strategy for solving the problem and building a solution. System design includes decisions about the organization of the system into subsystems, the allocation of subsystems to hardware and software components and major conceptual and policy decision that forms the framework for detailed design. |
| **5.2 PROCESS FLOW DIAGRAM** |
|  |

Email & password

User Input Microphone

Google Web Speech API

3. Speech Recognition

System Speaking

Reading Mail

2. Google Text To Speech

1. Login

4. Validate Mail List

Pre Known Email address

5. SMTP Protocol

SMTPLIB API

Sending Mail

6. IMAP Protocol

Fetch Inbox Mail

**CHAPTER 6**

**FUNCTIONAL AND NON\_FUNCTIONAL**

**REQUIREMENTS**

|  |
| --- |
| **6.1 FUNCTIONAL REQUIREMENTS** |
| The functional requirements represent the intended behaviour of the system. The proposed system consists of   * Microphone access * Voice recognition * Converting Text to Voice * Sending Mail * Converting Voice to Text * Reading Mail from Inbox |
| **6.2 NON-FUNCTIONAL REQUIREMENTS** |
| * **Efficiency** * **Reliability** * **Accuracy** |

**CHAPTER 7**

**TESTING**

|  |
| --- |
| **7.1 SYSTEM TESTING** |
| System Testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently as expected before live operation commences. It certifies that the whole set of program hang together. System testing requires a test plan that consists of several keys, activities and steps to run program, string, system and user acceptance testing. The implementation of newly designed package is important in adopting a successful new system.  **Testing Objectives**   * Testing is the process of correcting a program with intend of finding an error. * A good test is one that has a high probability of finding a yet undiscovered error. * A successful test is one that uncovers an undiscovered error.   There are different types of testing methods available: |
| **7.2 UNIT TESTING** |
| In this testing we test each module individually and integrate the overall system. Unit testing focuses verification efforts on the smaller unit of software design in the module. This is also known as ‘module’ testing. The modules of the system are tested separately. The testing is carried out during programming stage itself. In this testing step each module is found to work satisfactory as regard to the expected output from the module. There are some validation checks for verifying the data input given by the user. It is very easy to find error and debug the system. |
| **7.3 INTEGRATION TESTING** |
| Data can be lost across an interface; one module can have an adverse effect on the other sub functions when combined by May not produce the desired major functions. Integrated testing is the systematic testing for constructing the uncover errors within the interface. This testing was done with sample data. The need for integrated test is to find the overall system performance. |
| **7.4 BLACK BOX TESTING** |
| This testing attempts to find errors in the following areas or categories: Incorrect or missing functions, interface errors, errors in data structures, external database access, performance errors and initialization and termination errors. |
| **7.5 VALIDATION TESTING** |
| At the culmination of Black Box testing, software is completely assembled as a package, interface errors have been uncovered and corrected and final series of software tests, validation tests begins. Validation testing can be defined in many ways but a simple definition is that validation succeeds when the software functions in a manner that can be reasonably accepted by the customer.  After validation test has been conducted one of the two possible conditions exists.   * The function or performance characteristics confirm to specification and are accepted. * A deviation from specification is uncovered and a deficiency list is created. |
| **7.6 OUTPUT TESTING** |
| After performing the validation testing, the next step is output testing of the proposed system since no system could be useful if it doesn’t produce the required data in the specific format. The output displayed or generated by the system under consideration is tested by, asking the user about the format displayed. The output format on the screen is found to be correct as the format was designed in the system according to the user needs. Hence the output testing doesn’t result in any correction of the system. |
| **7.7 USER ACCEPTANCE TESTING** |
| User acceptance of the system is the key factor for the success of the system. The system under consideration is tested for user acceptance by constantly keeping in touch with prospective system at the time of developing and making change wherever required. This is done with regard to the following points:   * Output Screen design. * Input Screen design. * Menu driven system. |
| **7.8 WHITE BOX TESTING** |
| White box testing is a testing case design method that uses the control structure of the procedural design to derive the test cases. The entire independent path in a module is exercised at least once. All the logical decisions are exercised at least once. Executing all the loops at boundaries and within their operational bounds exercise internal data structure to ensure their validity.  In our project testing was conducted at every step. Initially each module was tested separately to check whether they gave the desired output for the given input. The forms used to enter data by user were validated and appropriate error messages were displayed if incorrect data was entered. Once the data was entered correctly, the processing was done and testing was done to check whether the correct output was obtained. Once the test cases were conducted successfully for each module, the modules were integrated together as a single system. After integration, the test cases were again applied to check whether the entire system as a whole produced the desired output. At times, the test cases failed and the shortcomings were noted down and appropriate corrections were done. Once the integration testing was performed correctly, output testing was done and it did not result in any change or correction in the system. Black box testing and white box testing was also conducted successfully. All the loops, decisions, relations were executed at least once before giving it to the users for testing. In black box testing, it was checked whether the data in the proper format was stored in the database or not. Also, it was checked whether the interfaces were working properly or not. On successful completion of these tests, the system was then given to undergo user acceptance testing where the users entered test data to check whether the correct output was obtained. The users were satisfied with the output and thus the testing phase was completed successfully. |
| **7.9 TEST DATA AND RESULTS** |
| The primary goal of software implementation is the production of source code that is easy to read and understand. Clarification of source code helps in easier debugging, testing and modification. Source code clarification is enhanced by structural coding techniques, by good coding style, by appropriate supporting documents, by good internal comments and by the features provided in the modern programming language.  In our implementation phase, source code contains both global and formal variables. It contains predefined functions as well as the user defined functions. The result of the new system is compared with old system and supposes if the result is wrong the error must be debugged.  After the acceptance of the system by the user, the existing system should be replaced by this system. Any user handles this package very easily. It does not require any intensive training for the user. Procedures and functions involved in this system are very simple that anyone can understand and correspondingly act to the system with no difficulty. |
| **7.10 TEST CASES** |
|  |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **TEST CASE ID** | **TEST CASE NAME** | **EXPECTED OUTPUT** | **STATUS** | **DEFECTS** | | 1 | Speech Recognition | To recognize user’s voice. | Yes | Nil | | 2 | Sending Mail | Receive Mail from User using voice interaction and sends to corresponding Receiver. | Yes | Nil | | 3 | Reading Inbox | To read the unread mails from user’s inbox. | Yes | Nil | | 4 | Speech Recognition under heavy noise | To recognize user’s voice. | No | System does not recognizes user’s voice under heavy noise. | |

**CHAPTER 8**

**RESULTS AND DISCUSSIONS**

|  |
| --- |
| **8.1 RESULTS** |
|  |
| **8.2 SCREENSHOTS** |
|  |
|  |
|  |
|  |
|  |
|  |

**CHAPTER 9**

**CONCLUSION**

|  |
| --- |
| **9.1 SYSTEM IMPLEMENTATION** |
| Implementation is an activity that is contained throughout the development phase. It is the process of bringing a developed system into operational use and turning it over to the user. The new system and its components are to be tested in a structured and planned manner. A successful system should be delivered and users should have the confidence that the system should have work efficiently and effectively. The more complex system being implemented, the more will be the system analysis and design effort required just for implementation.  Implementation is the stage of the system when the theoretical design is turned into working system. The plan contains an overview of the system, a brief description of the major tasks involved in the implementation, the overall resources needed to support the implementation effort, and any site implementation requirements. The plan is developed during the design phase and is updated during the Development phase. The outline shows the implementation plan.  There are three types of implementations:   1. Implementation of a computer system for replacing the manual system. The problem encountered are converting files, training users, create accurate files. 2. Implementation of new computer system to replacing an existing one. This is usually a difficult conversion. If not properly planned, there can be many problems. Some larger computer systems have taken as long as a year to convert. 3. Implementation of modified application to replace an existing one using the same computer. This type of conversion relatively easy to handle, provided there are no major changes in file.   **Implementation Plan Preparation**  The implementation plan begins with preparing a plan for the implementation of the system. In this plan, discussion has been made regarding the equipment, resources and how to test the activities. The following information is acquired.   * What the task will accomplish? * Resources required to accomplish the task. * Key person(s) responsible for the task.   Examples of the major tasks are the following.   * Providing overall planning and coordination for the implementation. * Providing appropriate training for the personnel. * Ensuring that all the manual are applicable to the implementation. * Providing all the technical requirements. * Performing site surveys before implementation. * Ensuring that all the prerequisites have been fulfilled before the implementation date. * Providing personnel for the implementation team. * Acquiring special software or hardware. * Performing data conversion before loading data into the system. * Preparing site facilities for implementation.   **Implementation of Proposed System**  After having user acceptance for the system developed, the implementation phase begins. Implementation is the stage of project during which theory is tuned into practice. During this phase, all the programs of the system are loaded into the user’s computer. After loading the system training of the user starts. Such as type of training includes:   1. How to execute the package? 2. How to enter the data? 3. How to process the data (processing details)? 4. How to takeout the report?   The following two strategies are followed for running the system.  **Parallel Run:** In such run for a certain defined period, both the systems thereafter computerized and manual are executed in parallel. This strategy is helpful because of the following:   1. Manual result can be compared with the result of computerized system. For the care of demonstration of the success of this system, it was implemented with successfully running; manual systems and results are verified. 2. Failure of a computerized system at an early stage, do not affect the work of the organization, because the manual system continues to work as it used to do.   **Pilot Run:** In this type of run, some parts of the new system are installed first and executed successfully for the considerable time period. When the results are found satisfactory, only then the other parts are implemented. This strategy builds the confidence and errors are traced easily. |
| **9.2 FUTURE ENHANCEMENT** |
| The major drawbacks of the application can be used as the future enhancements for this project. There are two major drawbacks in this application i.e.., the exact voice recognition and the image or document attachment. So in the future enhancement, we can add the image or document attachment for the sender and also to create |
| **9.3 CONCLUSION** |
| This project is the best applicable for physically challenged people. One who is unable to read and write for the communication. The system is based on IVR voices and there is no use of keywords. Only mouse click operations are performed to initiate the operations. Any users are able to use the system as it does not require any prior knowledge. The only thing to keep in mind is to follow the instructions to do the right actions. The system is quite efficient and accessible to utilize for the communication for the blind people. There is no need to write or type the messages in the inbox of the email only through your voice’s communication is possible. The system is completely made for physically challenged people for easy communication. It enhances the path of communication in a fast and interesting way. Even the normal users can utilize the system smoothly. This system will help the blinds to overcome difficulties in accessing the emails. |

**CHAPTER 10**

**BIBLIOGRAPHY**

|  |
| --- |
| **10.1 BOOKS** |
| [1]. Speech And Language Processing: International Edition, by Daniel Jurafsky, James H. Martin [2]. Theory And Applications Of Digital Speech Processing, by Lawrence |
| **10.2 WEBSITES** |
|  |
| **10.3 JOURNALS AND PUBLICATIONS** |
|  |
|  |
|  |

1. [**SPEECH AND LANGUAGE PROCESSING: INTERNATIONAL EDITION**](http://catalogue.pearsoned.co.uk/educator/product/Speech-and-Language-Processing-International-Edition/9780135041963.page)

*by Daniel Jurafsky (Author),‎ James H. Martin (Author)*