JSS COLLEGE OF ARTS, COMMERCE & SCIENCE

An Autonomous college Affiliated to university of Mysore Re-

Accredited By NACC with 'A' Grade,

Ooty Road, Mysuru-570025

A PROJECT REPORT ON

"Personal Chat Bot Assistant"

Submitted for partial fulfillment for the award of the degree of

Bachelor of Computer Application

Submitted By

Arun Nag Pateel G N

UNDER THE GUIDANCE OF

Ms. Megha S

Assistant Professor

Dept of Computer Science

JSSCACS

2022-2023

JSS COLLEGE OF ARTS, COMMERCE & SCIENCE

An Autonomous college Affiliated to university of Mysore Re-

Accredited By NACC with 'A' Grade,

Ooty Road, Mysuru-570025

CERTIFICATE

This is to certify that the Final Year Project report entitled **"Personal Chat Bot Assistant"** is a Bonafide work carried out by **Arun Nag Pateel G N (JUA20009)** Student of **6th Semester, Bachelor of Computer Application, JSS College of Arts Commerce and Science, Mysore** in Partial Fulfilment for the Award of the Degree of, Bachelor of Computer Application in JSS College of Arts Commerce and Science Mysore, During the Academic year **2022-2023**. It is Certified that all the Suggestions and Corrections indicated for the internal assessment have been incorporated in the report deposited in the department library. The report has been approved as it satisfies the requirements in respect of technical internship work prescribed for the said degree.

Signature of the Guide **Signature of HOD**

Date of examination:

VALUED

Name and Signature of examiner

DECLARATION

I am **Arun Nag Pateel G N** hereby declare that this term work entitled "**PERSONAL CHAT BOT ASSISTANT**" submitted to JSS College of Arts, Commerce and Science, Ooty road Mysore (Affiliated to University of Mysore) is a record of original work done by me and my project partner under the supervision and guidance of **Ms. Megha S**, Assistant Professor Department of Computer Science.

Candidate Name : Arun Nag Pateel G N (JUA20009)

Signature of the Student :

Place : Mysuru

Date :

ACKNOWLEDGEMENT

The Final Year Project would not have been possible without the guidance, assistance and suggestions of many individuals. We would like to express our deep sense of gratitude and indebtedness to each and every one who has helped us to make this project a success.

We heartily thank our beloved Principal, Prof. Vijayendra Kumar M P for his wholehearted support and for his kind permission to complete the Final Year Project.

We wish to express our deepest gratitude to Prof. Vidyashankar, Head of Department, JSS College of Arts Commerce and Science, JSSCACS for their constant encouragement and inspiration in completing this project.

We gratefully thank our project guide, Miss Megha, Department of Computer Application, for his encouragement and advice throughout the course of the Final Year Project.

In the end, we extend our gratitude towards our family members and friends for their valuable suggestions and encouragement.

ABSTRACT

A chat bot also known as a talk Bot, IM bot, chatterbot, Bot, interactive agent, or an Artificial Conversational Entity. Chat bot is a computer program that helps you to communicate via messages. They are formulated to mimic human behavior. They can reproduce exactly how a human would act as a conversational partner. In today’s era almost all tasks are digitalized. We have Smartphone in hands, and it is nothing less than having world at your fingertips.

Table of Contents

CHAPTER 1
INTRODUCTION

1.1 INTRODUCTION

Artificial Intelligence when used with machines, it shows us the capability of thinking like humans. In this, a computer system is designed in such a way that typically requires interaction from human. As we know Python is an emerging language, so it becomes easy to write a script for Voice Assistant in Python. The instructions for the assistant can be handled as per the requirement of user. Speech recognition is the Alexa, Siri, etc. In Python there is an API called Speech Recognition which allows us to convert speech into text. It was an interesting task to make my own assistant. It became easier to send WhatsApp messages without typing any word, searching on Google without opening the browser, and performing many other daily tasks like playing music with the help of a single voice command. In the current scenario, advancement in technologies is such that they can perform any task with same effectiveness or can say more effectively than us. By making this application, it realized that the concept of AI in every field is decreasing human effort and saving time.

As the voice assistant is using Artificial Intelligence hence the result that it is providing are highly accurate and efficient. The assistant can help to reduce human effort and consumes time while performing any task, they removed the concept of typing completely and behave as another individual to whom we are talking and asking to perform task. The assistant is no less than a human assistant, but the user can say that this is more effective and efficient to perform any task. The libraries and packages used to make this assistant focuses on the time complexities and reduces time.

The functionalities include, it can read PDF, it can send text on WhatsApp, it can open command prompt, your favorite IDE, notepad etc., It can play music, it can do Wikipedia searches for you, it can open websites like Google, YouTube, etc., in a web browser, it can give weather forecast, it can give desktop reminders of your choice. It can have some basic conversation.

Tools and technologies used are PyCharm IDE for making this application, and created all py files in PyCharm. Along with this the following modules and libraries are used in the proposed application. pyttsx3, SpeechRecognition, Datetime, Wikipedia, pywhatkit, pyjokes, pyPDF2, pyautogui, pyQt etc. And created a live GUI for interacting with the chatbot as it gives a design and interesting look while having the conversation.

1.1 OBJECTIVES

Main objective of building e-assistant software (a virtual assistant) is using semantic data sources available on the web, user generated content and providing knowledge from knowledge databases. The main purpose of an intelligent virtual assistant is to answer questions that users may have.

1.2 EXISTING SYSTEM

Users are familiar with many existing voice assistants like Alexa, Siri, Google Assistant, Cortana which uses concept of language processing, and voice recognition. They listen the command given by the user as per their requirements and performs that specific function in a very efficient and effective manner.

As these voice assistants are using Artificial Intelligence hence the result that they are providing are highly accurate and efficient. These assistants can help to reduce human effort and consumes time while performing any task, they removed the concept of typing completely and behave as another individual to whom we are talking and asking to perform task. These assistants are no less than a human assistant but user can say that they are more effective and efficient to perform any task. The algorithm used to make these assistant focuses on the time complexities and reduces time.

But for using these assistants one should have an account (like Google account for Google assistant, Microsoft account for Cortana) and can use it with internet connection only because these assistants are going to work with internet connectivity to login. They are integrated with many devices like, phones, laptops, and speakers etc.

1.2 PROPOSED SYSTEM

It was an interesting task to make our own assistant. It became easier to send whatsapp messages without typing any word, take screenshots, control volume settings, searching on Google without opening the browser, and performing many other daily tasks like playing music, opening your favorite IDE with the help of a single voice command. The proposed application is different from other traditional voice assistants in terms that it is specific to desktop and user does not need to make account to use this, it does not require any internet connection while logging in to the application.

The IDE used in this project is PyCharm. All the python files were created in PyCharm and all the necessary packages were easily installable in this IDE. For this application following modules and libraries were used i.e. pyttsx3, SpeechRecognition, Datetime, wikipedia, pywhatkit, pyjokes, pyPDF2, pyautogui, pyQt etc. And have created a live GUI for interacting with the chatbot as it gives a design and interesting look while having the conversation.

With the advancement this application can perform any task with same effectiveness or can say more effectively than us. By making this application, I realized that the concept of AI in every field is decreasing human effort and saving time. Functionalities of this project include, it can read PDF, it can send text on WhatsApp, it can open command prompt, your favorite IDE, notepad etc., It can play music, it can do Wikipedia searches for you, it can open websites like Google, YouTube, etc., in a web browser, it can give weather forecast, it can give desktop reminders of your choice. It can have some basic conversation.

1.6 ADVANTAGES OF PROPOSED SYSTEM

- Chatbot can gather information efficiently and accurately.
- Saves Time, assigning repeated tasks to virtual voice assistants frees up the human time and resources. They also perform the tasks more efficiently than the humans
- Chatbot have 24/7 Availability, this means that chatbot can answer user queries whenever user have queries.
- These digital voice-enabled assistants also make the operations hands-free. You don't need to touch any screen or type in anything.

CHAPTER 2
LITERATURE SURVEY

2.1. CHATBOT AS A E-ASSISTANT

Gayatri Nair, Soumya Johnson, V. Sathya (2018)

Abstract: Chatbots usually work as an optimizer of customer service. Chatbots are designed in such a way that the users are made to believe that they are talking to an actual human being but rather they are talking to a machine. The main advantage of this explicit characteristic of a chatbot is that it can be given a virtual personality of its own just like a specific person of a particular profession. This paper proposes an idea of an artificially intelligent chatbot as a personal assistant that can serve as your resume. Also, this chatbot helps the user in fixing meetings and adding them in the calendar. Our chatbot is basically a resumeBot. It is a resume in the form of a chatbot. A resume in the form of chatbot is the new generation in the field of job applications. A chatbot will help the recruiter to know your personality along with all your qualifications and personal details. The chatbot will communicate with the user with the help of natural language processing and AIML files.

Methodology

- Pattern Matching
- Artificial Neural network

Limitation

- Easy to steal information as it is used as resume bot

2.2. CHATBOT SYSTEM FOR HEALTHCARE USING ARTIFICIAL INTELLIGENCE

Aishwarya Kedar, Jyoti Dahale, Khushboo Patel, Shivani Lahamage, Prof. S. G. Chordiya (September 2020)

Abstract: Chatbot can be described as software that can chat with people using artificial intelligence.

These software are used to perform tasks such as quickly responding to users, informing them, helping to purchase products and providing better service to customers. In this paper, we present the general working principle and the basic concepts of artificial intelligence based chatbots and related concepts as well as their applications in various sectors such as telecommunication, banking, health, customer call centers and e-commerce. Additionally, the results of an example chabbot for donation service developed for telecommunication service provider are presented using the proposed architecture. We are using it for educational purpose to solve the quires of users. Chatbots are programs that mimic human conversation using Artificial Intelligence (AI). It is designed to be the ultimate virtual assistant, entertainment purpose, helping one to complete tasks ranging from answering questions, getting driving directions, turning up the thermostat in smart home, to playing one's favorite tunes etc. Chatbot has become more popular in business groups right now as they can reduce customer service cost and handles multiple users at a time. But yet to accomplish many tasks there is need to make chatbots as efficient as possible. in this system we provide the design of a chatbot, which provides an efficient and accurate answer for any query based on the dataset of FAQs using Artificial Intelligence Markup Language (AIML) and Latent Semantic Analysis (LSA). Template based and general questions like welcome/ greetings and general questions will be responded using AIML and other service based questions uses LSA to provide responses at any time that will serve user satisfaction.

Methodology

- AES
- Hashing and mapping

Limitation

- One of the key concerns in the use of AI chatbots in healthcare is user privacy

2.3 CHATBOTS AT THE DIGITAL WORKPLACE

Raphael Meyer von Wolff, Sebastian Hobert, Matthias Schumann (2019)

Abstract: Chatbots become quite hyped in recent times as they can provide an intuitive and easy-to-use natural language human-computer interface. Nevertheless, they are not yet widespread in enterprises. Corresponding application areas for collaboration at digital work-places are lacking and prior research contributions on this topic are limited. In this research paper, we aim at surveying the state of the art as well as showing future research topics. Thus, we conducted a structured literature review and showed that only few first research contributions exist. We also outline current potentials and objectives of chatbot applications. In the discussion of the results of our structured literature review, we show that research gaps are present. To tackle the research gaps, we derive open research questions.

Methodology

- Natural Language Processing

Limitation

- Pre-Defined Scope and Function: chatbots cannot improvise or undertake a task

2.4. DONNA INTERACTIVE CHAT-BOT ACTING AS A E-ASSISTANT

Namita Mhatre, Karan Motani, Maitri Shah, Swati Mali (April 2016)

Abstract: Chat-bots are computer programs coded to have a textual or verbal conversation which is logical or intelligent. Chat-bots are designed to make humans believe that they are talking to a human but instead they are in fact talking to a machine.

Taking advantage of this transparency property of chat-bot, an artificial character and personality can be given to a chat-bot which acts like a person of a specific profession. This paper describes an approach to the idea of implementing web-based artificially intelligent chatbot as a personal assistant of the user, which stimulates setting and initiating meetings of user with his clients. The exchange of information happens through email conversations whereas its evaluation happens through natural language procession and natural language generation and AIML files.

Methodology

- Google APIs
- Calendar API
- Gmail API
- Pattern matching

Limitation

- Efficiency decreases with the increase in the size of emails.

CHAPTER 3
SYSTEM REQUIREMENT SPECIFICATION

Software Requirement Specification (SRS) is a complete description of the behavior of the system to be developed. It is a fundamental document which forms the foundation of the software development process. It not only lists the requirements of a system but also has a description of its major feature. An SRS is basically an organizations understanding of a customer or potential client's system requirements prior to any actual design or development work. Use cases denote the functional requirement of the system. In addition to use cases, the SRS also contains non-functional requirements. Non-functional requirements impose constraints on the design or implementation. The SRS is often referred to as the "parent" document because all subsequent project management documents such as design specifications, statements of work, software architecture specifications, testing and validation plans and documentation plans are related to it.

A software requirement specification (SRS) is a description of a software system to be developed, laying out functional requirements and may include a set of use cases that describe interactions the users will have with the software.

SRS specifies the general factors like the external interfaces, evolution or growth path of the system, functional requirements, user characteristics and the assumptions, dependencies and risks associated with the system.

Some of the features of SRS are

- It sets a rigorous assessment of requirements before design can begin.
- It sets the basis for software design, test, deployment, training etc. It also sets pre-requisite for a good design though it is not enough.
- It sets basis for software enhancement and maintenance.
- It sets basis for project plans like scheduling and estimation.

3.1 FUNCTIONAL REQUIREMENT SPECIFICATION

A function is described as a set of inputs, the behavior, and outputs. Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish.

Functionalities of this project include

- User can login through Face Unlock
- Search anything from Wikipedia, google maps, etc.
- Play video from YouTube
- Search COVID related information
- Search Weather

- Jokes
- News
- Capture screenshots
- Math Calculations
- Timer
- In-built search image display
- Smart Dictionary Search
- OS Info, Battery Info
- Window, Tab Operations
- Opening Websites
- File Operations (Creating Files)
- Web Automation (HTML Project)
- Coin Toss, Roll Dice
- Translator
- To-Do List
- Volume Control

3.2 NON FUNCTIONAL REQUIREMENT

These are the requirements which are not directly concerned with the specific function delivered by the system. Non-Functional requirements may relate to emergent system properties such as reliability, response time and store occupancy. Non-functional requirements are also called the qualities of a system. These qualities can be divided into execution quality and evolution quality. Execution qualities are security and usability of the system which are observed during run time whereas evolution quality involves testability, maintainability, extensibility or scalability.

Alternatively, they may define constraints on the system such as the capability of the input/ output devices and the data representations used in system interfaces. Many non-functional requirements relate to the system as whole rather than to individual system features. This means they are often critical than the individual functional requirements. Non-functional requirements are constraints on the services or functions offered by the system.

The non-functional requirements of the system include

- The system ensures safety, security, and usability, which are observable during run time.
- The system is adaptable to different situations.
- The project has good and compact UI using TKinter with responsive interface. The project is light on resources.

3.3 SYSTEM SPECIFICATION

HARDWARE AND SOFTWARE REQUIREMENTS

The software is designed to be light-weighted so that it doesn't be a burden on the machine running it. This system is being build keeping in mind the generally available hardware and software compatibility. Here are the minimum hardware and software requirement for virtual assistant.

Hardware Requirements

- Processor : Intel core i5 8th Gen .
- RAM : 8 GB.
- Hard Disk : 1 TB.

Software Requirements

- Operating system : Windows 10 pro.
- IDE : Pycharm 3.2
- Editor : Visual studio Code.
- Programming language : Python 3.9

CHAPTER 4
SYSTEM DESIGN

4.1 ARCHITECTURE DIAGRAM

An architectural diagram is a diagram of a system that is used to abstract the overall outline of the software system and the relationships, constraints, and boundaries between components.

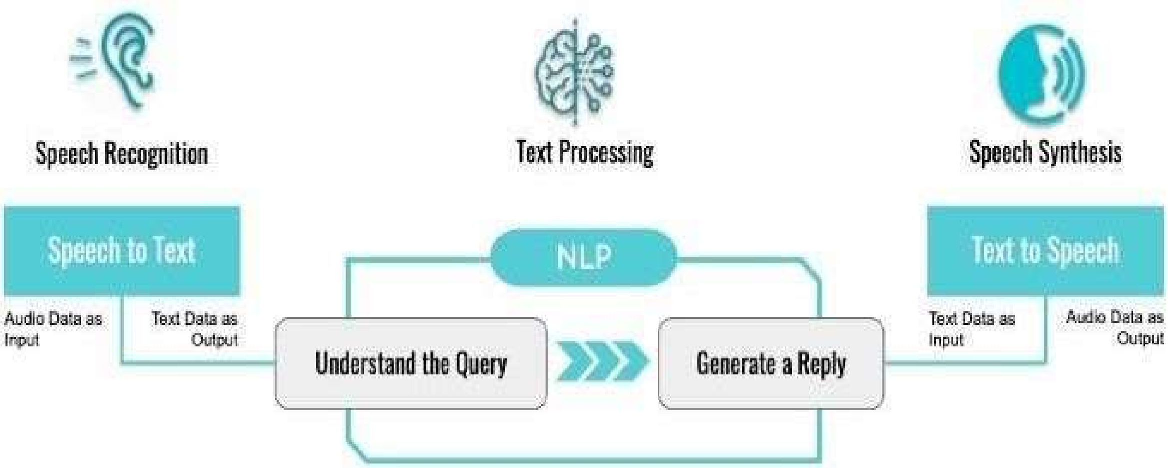


Figure 4.1: System Architecture Diagram

Natural Language Processing: Once the user query is converted to text, the bot will then process the identified text using the NLP package. NLP based chatbot can understand the user query written in their natural language and answer them immediately. NLP making chatbots effective in doing the majority of conversations easily without human assistance.

4.2 USE CASE DIAGRAM

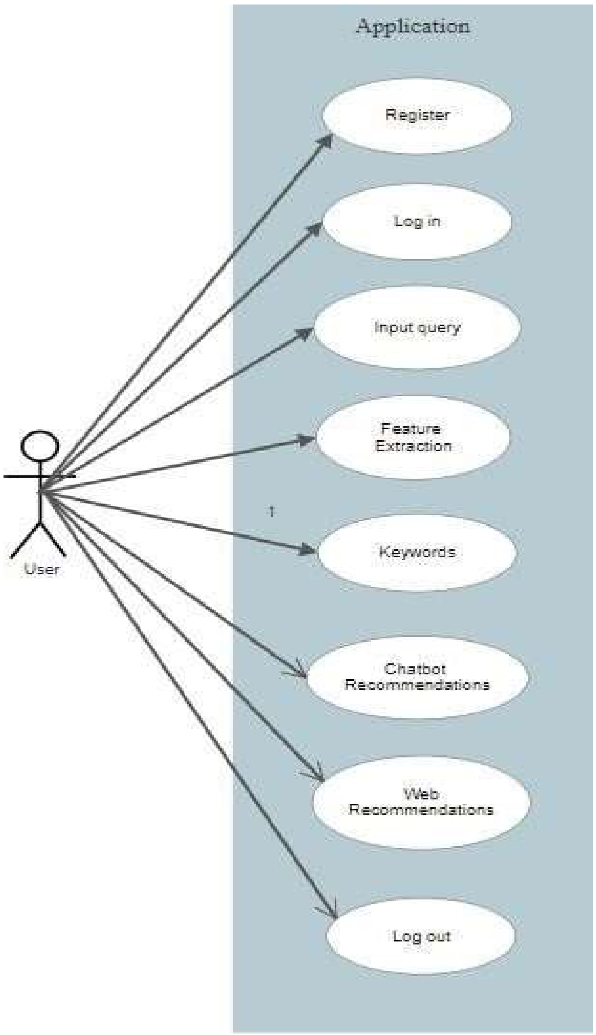


Figure 4.2: User Use Case diagram

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses. In this project there is only one user. The user queries command to the system. System then interprets it and fetches answer. The response is sent back to the user.

4.3 ACTIVITY DIAGRAM

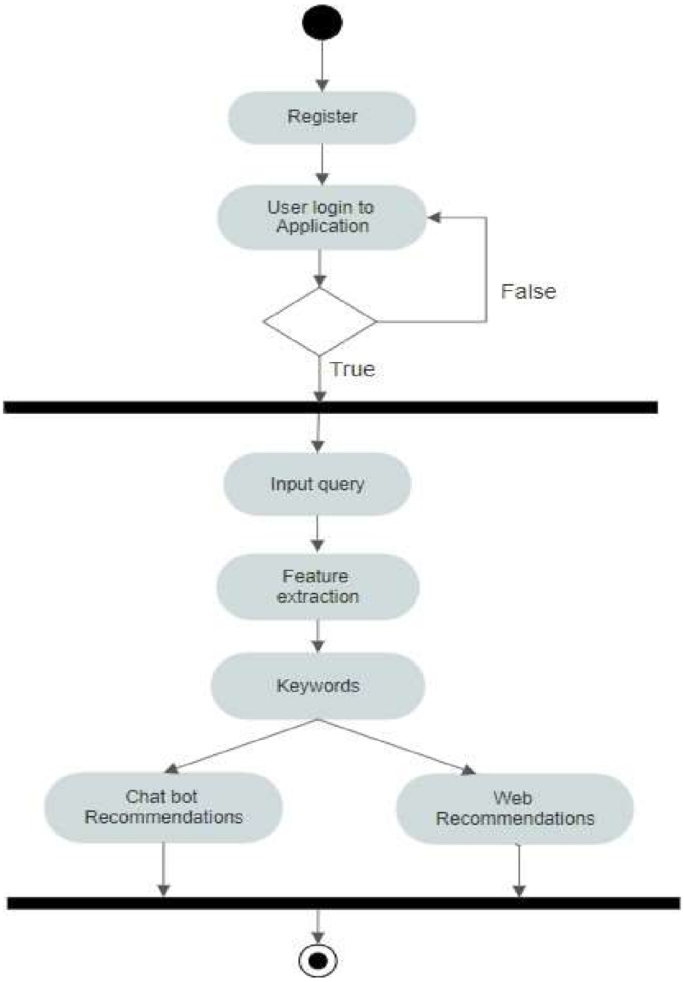


Figure 4.3: User Activity diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. Different symbols are used to represent activities. As the system is started, it first authenticate the authorized user, then voice assistant is on running in the background listening for available voice commands; once the user gives a command, based on the conditions provided to the voice assistant, the voice assistant gives the necessary output extracting the information from within application or web recommendations. This output is sent to the Speech Recognition which converts the speech into machine-readable form. Based on the input received the E-assistant chatbot then performs the desired task.

4.4 SEQUENCE DIAGRAM

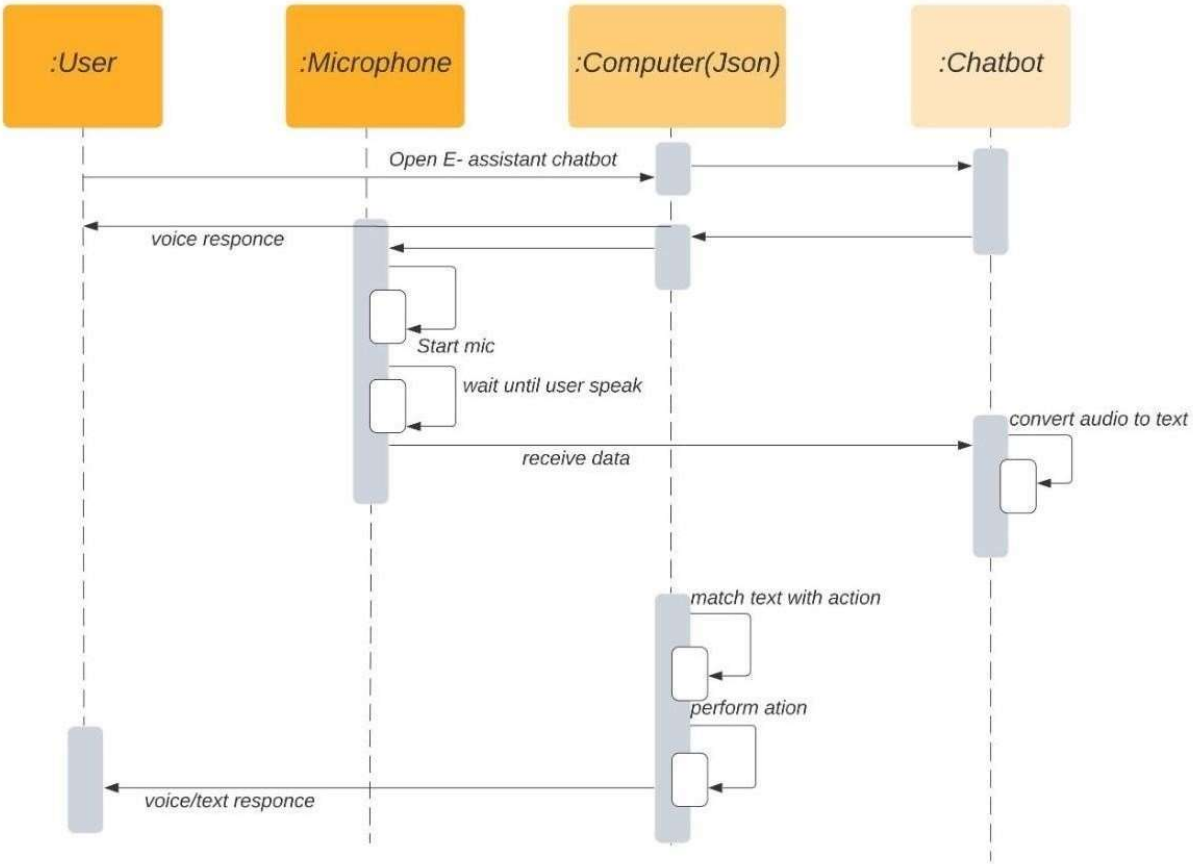


Figure 4.4: System sequence diagram

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are sometimes called event diagrams or event scenarios. The above sequence diagram shows how an answer asked by the user is being fetched from internet. The audio query is interpreted and sent to Web scraper. The web scraper searches and finds the answer. It is then sent back to speaker, where it speaks the answer to user.

4.5 CLASS DIAGRAM

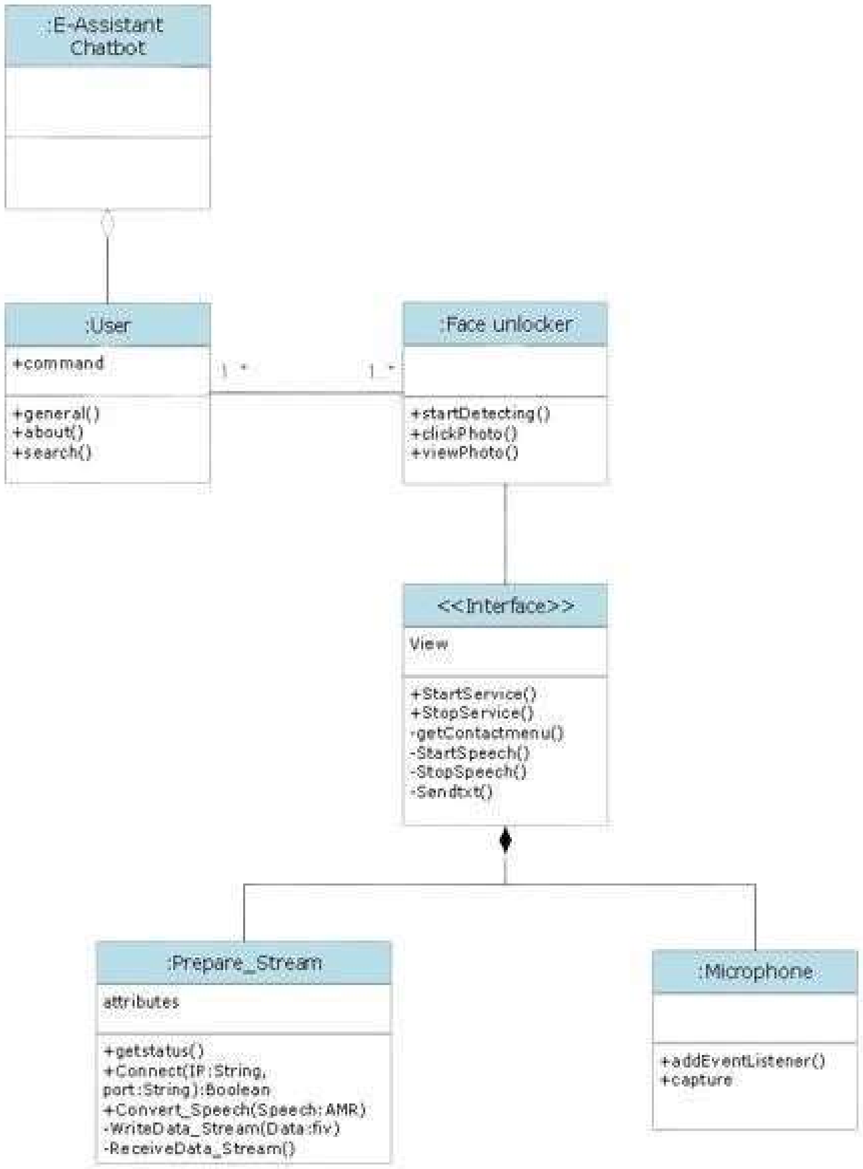


Figure 4.5: System class diagram

A Class is a Set of object that share a common structure and common behavior. Class is an abstraction of real-world items. An object is an instance or occurrence of a class. A Class describes a group of relationship and semantics. Object in a class have the same attribute and forma of behavior. Most objects derive their individuality from difference in their attribute values and specific relationships to other objects. A Class describe a group of objects with the same properties, behavior, kind of relationship and semantics. Class Diagram provides a Graphic notation for modelling and their relationships, thereby describe the possible objects.
The standard class diagram is composed of three sections Upper section: Contains the name of the class.

This section is always required, whether you are talking about the classifier or an object. Middle section: Contains the attributes of the class. Use this section to describe the qualities of the class. This is only required when describing a specific instance of a class. Bottom section: Includes class operations (methods). Displayed in list format, each operation takes up its own line. The operations describe how a class interacts with data.

CHAPTER 5
SYSTEM IMPLEMENTATION

E-assistant chat bot is a voice assistant that can perform many daily tasks of desktop like playing video from YouTube, sending WhatsApp messages, etc with the help of a single voice command. E-Assistant chatbot is different from other traditional voice assistants in terms that it is specific to desktop and user does not need to make account to use this, it does not require any internet connection to register face and login to the application.

5.1 REAL LIFE APPLICATION

- **Saves time:** Personal assistant is a desktop voice assistant which works on the voice command offered to it, it can do voice searching, voice-activated device control and can let us complete a set of tasks.
- **Conversational interaction:** It makes easier to complete any task as it automatically does it by using the essential module or libraries of Python, in a conversational interaction way. Hence any user when instruct any task to it, they feel like giving task to a human assistant because of the conversational interaction for giving input and getting the desired output in the form of task done.
- **Reactive nature:** The desktop assistant is reactive which means it know human language very well and understand the context that is provided by the user and gives response in the same way, i.e., human understandable language, English. So, user finds its reaction in an informed and smart way.
- **Multitasking:** The main application of it can be its multitasking ability. It can ask for continuous instruction one after other until the user "QUIT" it.
- **No Trigger phase:** It asks for the instruction and listen the response that is given by user without needing any trigger phase and then only executes the task.

5.2 DATA IMPLEMENTATION AND PROGRAM EXECUTION

As the first step, install all the necessary packages and libraries. The command used to install the libraries is "pip install" and then import it. The necessary packages included are as follows:

5.2.1 LIBRARIES AND PACKAGES

- **pyttsx3:** It is a python library which converts text to speech.
- **Speechrecognition:** It is a python module which converts speech to text.
- **playsound:** Single function module with no dependencies for playing sounds.
- **pillow:** Pillow is a Python Imaging Library (PIL), which adds support for opening, manipulating, and saving images

- **pyscreenshot:** The module can be used to copy the contents of the screen to a Pillow image memory using various back-ends.
- **pynput:** This library allows you to control and monitor input devices.
- **psutil:** psutil (process and system utilities) is a cross-platform library for retrieving information on running processes and system utilization (CPU, memory, disks, network, sensors) in Python.
- **opencv-python:** OpenCV (open source computer vision) is a very powerful library for image processing and machine learning tasks which also supports Tensorflow, Torch/Pytorch and Caffe
- **Wikipedia:** Wikipedia is a Python library that makes it easy to access and parse data from Wikipedia.
- **bs4:** Beautiful Soup is a library that makes it easy to scrape information from web pages.
- **youtube_search: S**earch for YouTube videos, channels & playlists. get video & playlist information using link. Get search suggestions.
- **wmi:** Windows Management Instrumentation (WMI) is Microsoft's implementation of Web-Based Enterprise Management (WBEM), an industry initiative to provide a Common Information Model (CIM) for pretty much any information about a computer system.
- **geopy:** geopy is a Python client for several popular geocoding web services. geopy helps to locate the coordinates of addresses, cities, countries, and landmarks across the globe using third-party geocoders and other data sources.
- **googletrans:** Googletrans is a **free** and **unlimited** python library that implemented Google Translate API. This uses the Google Translate Ajax API to make calls to such methods as detect and translate.
- **pyaudio:** PyAudio provides Python bindings for PortAudio, the cross-platform audio I/O library.

5.3 MODULES USED IN THE IMPLEMENTATION

- **appControl:** This module is used to perform system tasks, tab operations, window operations, and volume operations etc.
- **dictionary:** This module is used to get the definition and the meaning for user given words.
- **fileHandler:** This module is used to create files such as html, python, java, power point, excel, word.
- **game:** This module is used to play games.
- **math_function:** This module is used to do basic operations (add, sub, power, etc.), bitwise operations, and conversion methods

- **webScrapping:** This is the function which contains all the necessary task execution definition like getting COVID-19 information, Wikipedia, weather, latest news, maps, get directions, jokes, YouTube, google search, send WhatsApp messages, etc.

5.4 LOGIN MODULE FOR E-ASSISTANT CHATBOT

Hardware as a Rental cascade Algorithm (Haar cascade) : Haar Cascades are machine learning object detection algorithms that are used to identify faces in an image or a real-time video. The Haar Cascade algorithm uses edge or line detection features that are proposed by Viola and Jones within their research paper named “Rapid Object Detection employing Boosted Cascade of Simple Features”.

STEP 1: Import OpenCV

Install and set up OpenCV to python. OpenCV provides a real-time optimized Computer Vision library, tools, and hardware that is implemented in our project.
In the command prompt type command “pip install opencv-python”.

Create a new directory and create a new file named ‘Main.py’

#Importing OpenCV import
cv2

STEP 2: Import XML file

#Importing HAAR CASCADE XML file

face_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')

STEP 3: Import the test image. imread() is a method of OpenCV to read the input. Similarly, imshow() is a method to display the processed input in the form of output.
#Uploading test image

img = cv2.imread('Test.jpg')

STEP 4: Converting the image to grey Scale.

The application works on images that are in greyscale and hence we convert the image to greyscale for ease of face detection. #Converting to grey scale
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

STEP 5: Detecting Multi-scale faces.

This function allows to detect objects of different sizes in the input image and hence an image with multi people with different sizes of the face can also be detected. Parameters of the functions are – detectMultiScale (InputArray image, double scaleFactor=1.1, int minNeighbors=6)

STEP 6: Mentioning sides of the rectangle for face detection.

This function helps us to mention the dimension thickness and color of the rectangle that will be visible during the face detection.

```
cv2.rectangle(image, start_point, end_point, color, thickness)
```

```
#Allowing multiple scale(Multiple size) detection faces =
face_cascade.detectMultiScale(gray, 1.1, 6)
#Creating Rectangle around face for(x,
y, w, h) in faces:
    cv2.rectangle(img, (x, y), (x+w, y+h), (0, 0, 250), 2)
```

STEP 7: Displaying the detected image.

To display the image that has been detected. Also add the feature to close the image tab only when a key is pressed. #Displaying the image cv2.imshow('Detected Face Image', img) #Waiting for escape key for image to close cv2.waitKey()

5.5. FLOWCHART

A flow chart is a graphical representation of algorithms. Different symbols are used to represent flow chart.

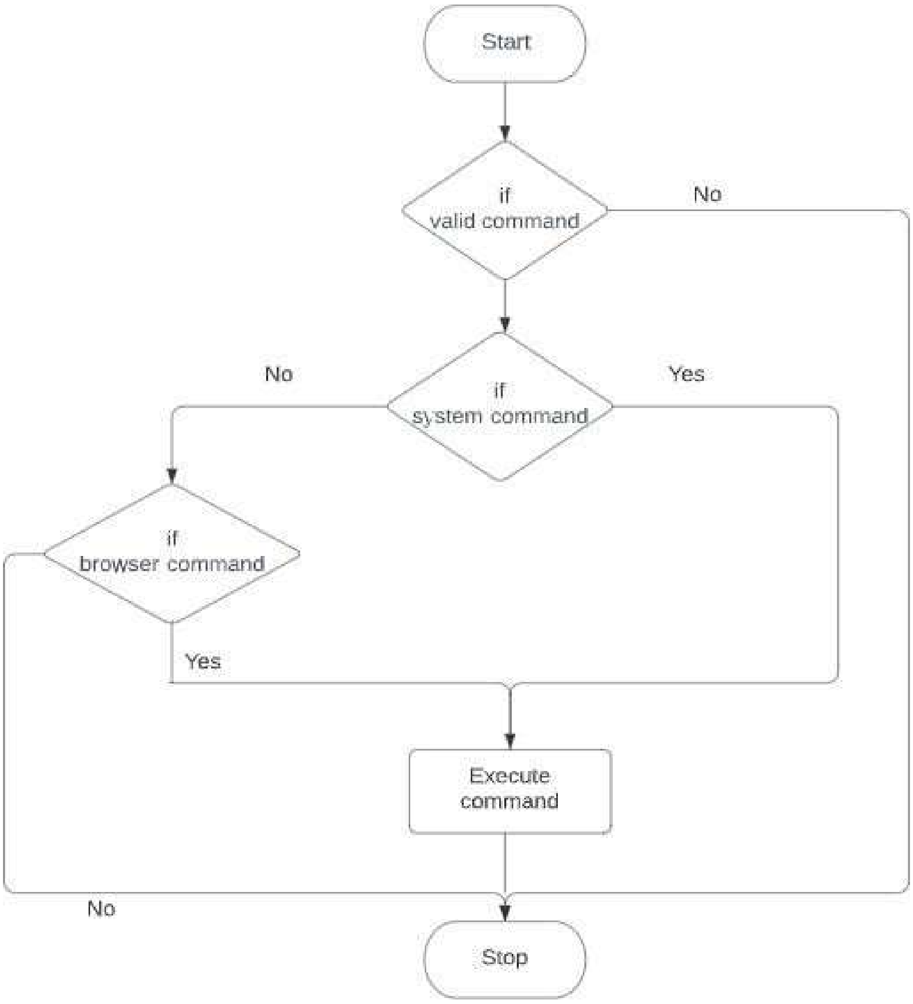


Figure 5.5: Flowchart of E-Assistant chat bot

Flow chart is the graphical representation of algorithms. Different symbols are used to represent flow chart. As the system is started, it first authenticates the authorized user, then voice assistant is on running in the background listening for available voice commands; once the user gives a command, based on the conditions provided to the voice assistant, the voice assistant gives the necessary output. This output is sent to the Speech Recognition which is convert the speech into machine-readable form.

Based on the input received the personal voice assistant then performs the desired task.

Chapter 6

SYSTEM TESTING

6.1 INTRODUCTION

System Testing is a process of verification and validation of software application or product to check whether it reaches the requirements. Software Testing is the process of executing a program or system with the intent of finding errors. Software testing is any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results. Software testing is an empirical investigation conducted to provide stakeholders with information about the quality of the product or service under test, with respect to the context in which it is intended to operate. This includes the process of executing a program or application with the intent of finding software bugs. The scope of software testing often includes examination of code as well as execution of that code in various environments and conditions as well as examining the aspects of code: does it do what it is supposed to do and do what it needs to do. In the current culture of software development, a testing organization may be separate from the development team. Testing is usually performed for the following purposes

- To improve quality□
- For Verification and validation□
- For reliability estimation□

Purpose of Testing

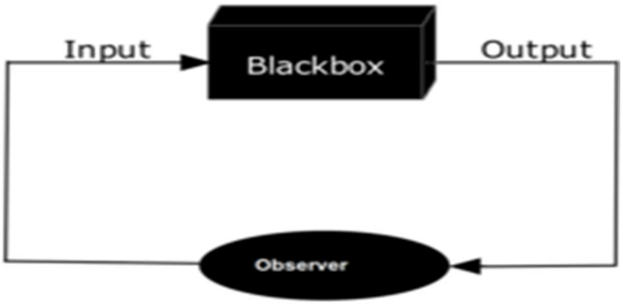
Testing accomplishes a variety of things, but most importantly it measures the quality of the software we are developing. This view presupposes there are defects in the software waiting to be discovered and this view is rarely disproved or even disputed. Several factors contribute to the importance of making testing a high priority of any software development effort. These include

- Reducing the cost of developing the program.□
- Ensuring that the application behaves exactly as we explain to the user for the vast majority of programs, unpredictability is the least desirable consequences of using an application.□
- Reducing the total cost of ownership. By providing software that looks and behaves as shown in the documentation, the customers require fewer hours of training and less support from product experts.□
- Developing customer loyalty and word-of-mouth market share.□

6.2 SOFTWARE TESTING STRATEGIES

6.2.1 BLACK-BOX TESTING

Black box testing is a method of software testing that examines the functionality of an application without peering into its internal structures or workings. This method of test can be applied to virtually every level of software testing: unit, integration, system and acceptance. It typically comprises most if not all higher level testing, but can also dominate unit testing as well.



Typical black-box test design techniques include:

- Decision table testing.▯
- All-pairs testing.▯
- Equivalence partitioning.▯
- Boundary value analysis.▯
- Cause–effect graph.▯
- Error guessing.▯
- State transition testing.▯ ▯ Use case testing.▯

Test cases are built around specifications and requirements, i.e., what the application is supposed to do. Test cases are generally derived from external descriptions of the software, including specifications, requirements and design parameters. Although the tests used are primarily functional in nature, non-functional tests may also be used.

6.2.2 WHITE-BOX TESTING

White box testing (also known as **clear box testing**, **glass box testing**, **transparent box testing**, and **structural testing**) is a method of testing software that tests internal structures or workings of an application, as opposed to its functionality (black-box testing). In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases. The tester chooses inputs to exercise paths through the code and determine the appropriate outputs. White-box testing can be applied at the unit, integration and system levels of the software testing process. Although traditional testers tended to think of white-box testing as being done at the unit level, it is used for integration and system testing more frequently today. It can test paths within a unit, paths between units during integration, and between subsystems during a system–level test. Though this method of test design can uncover many errors or problems, it has the potential to miss unimplemented parts of the specification or missing requirements.

Basic Procedure

White-box testing's basic procedures involves the tester having a deep level of understanding of the source code being tested. The programmer must have a deep understanding of the application to know what kinds of test cases to create so that every visible path is exercised for testing. Once the source code is understood then the source code can be analyzed for test cases to be created. These are the three basic steps that white-box testing takes in order to create test cases:

1. Input involves different types of requirements, functional specifications, detailed designing of documents, proper source code, security specifications. This is the preparation stage of white-box testing to layout all of the basic information.
2. Processing involves performing risk analysis to guide whole testing process, proper test plan, execute test cases and communicate results. This is the phase of building test cases to make sure they thoroughly test the application the given results are recorded accordingly.
3. Output involves preparing final report that encompasses all of the above preparations and results.

6.3 TEST CASES

Test Case Number	Testing Scenario	Expected result	Result
	Registration Testing		
TC – 01	Clicking Add face without entering details	Alert "Please fill all the details"	Pass
TC – 02	Clicking Add face without entering name	Alert " Please fill the details "	Pass
TC – 03	Clicking Add face without entering gender	Alert " Please fill the details "	Pass
TC – 04	Clicking Add face without checking I agree to use my face for security	Alert "Check the condition"	Pass
TC – 05	Clicking Submit without choosing your avatar	Alert "Choose your avatar"	Pass
	Login Testing		
TC – 06	Attempt to login without registered face	Alert "Face not found"	Pass
TC – 07	Attempt to login without connected to network	Alert "Your system is offline"	Pass
TC – 08	Attempt to send whatsapp message without entering valid phone number	Alert "Phone number is invalid"	Pass

TC – 09	Attempt to control volume by using keyboard	Alert "Volume settings changed"	Pass
TC – 10	Attempt to capture screenshot by using keyboard	Alert "Screenshot taken"	Pass
TC – 11	Input query to set timer	Alert "Alarm rings to set time"	
TC – 12	Clicking clear facial data	Alert "Your face has been cleared"	Pass
TC – 13	Closing the application using shutdown command	Alert "Shutting down the system"	Pass

Table 6.3. Test Cases for E-Assistant Chatbot

CHAPTER 7
CONCLUSION

The E-Assistant chatbot is built using available open-source software modules with visual studio code community backing which can accommodate any updates in future. The modular approach used in our E-Assistant chatbot makes it more flexible and easier to integrate additional modules and features without disturbing the current system functionaries. It not only works on human commands but also it is designed for give responses to the user based on query being asked or the words spoken by the user such as opening tasks and operations.

E-Assistant chatbot can be used to not only to chat and get information but it takes artificial intelligence in a whole new light. The main advantage of a chatbot is that it can provide services at any given rate. They respond immediately to the user’s demands with precisely relevant information. This enhances the rate of the communication operation. In simple terms, chatbots make our services faster and as a personal assistant it is the best. The proposed application is used to instantly take screenshots, control volume levels, adding items to list for reminders, language translations and various other reasons.

CHAPTER 8

FUTURE ENHANCEMENT

The proposed Intelligent voice assistant chatbot has an enormous and limitless scope in the future like Siri, Google now and Cortana most popular virtual voice assistants. The E-Assistant chatbot will easily able to integrate with devices near future for a connected Home using Internet of Things, voice command system and computer vision and also making chat bots to learn more on its own and develop a new skill in it. Chat bot android app can also be developed for making more chat bot voice terminals.

REFERENCE

[1] Sharma, Rakesh Kumar, and Manoj Joshi. "An Analytical Study and Review of open Source Chatbot framework, RASA." International Journal of Engineering Research and, vol.9, no.06 (2020).

[2] Thosani, Parth, et al. "A Self Learning Chat-Bot from User Interactions and

Preferences." 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS). IEEE, pp. 224-229, 2020.

[3] Carlene Lebeuf, Margaret-Anne Storey, and Alexey Zagalsky, "Software Bots" in IEEE Software, www.computer.org/software.

[4] "How Do Chatbots Work? A Guide to the Chatbot Architecture", https://www.marutitech.com/chatbotswork-guide-chatbot-architecture/.

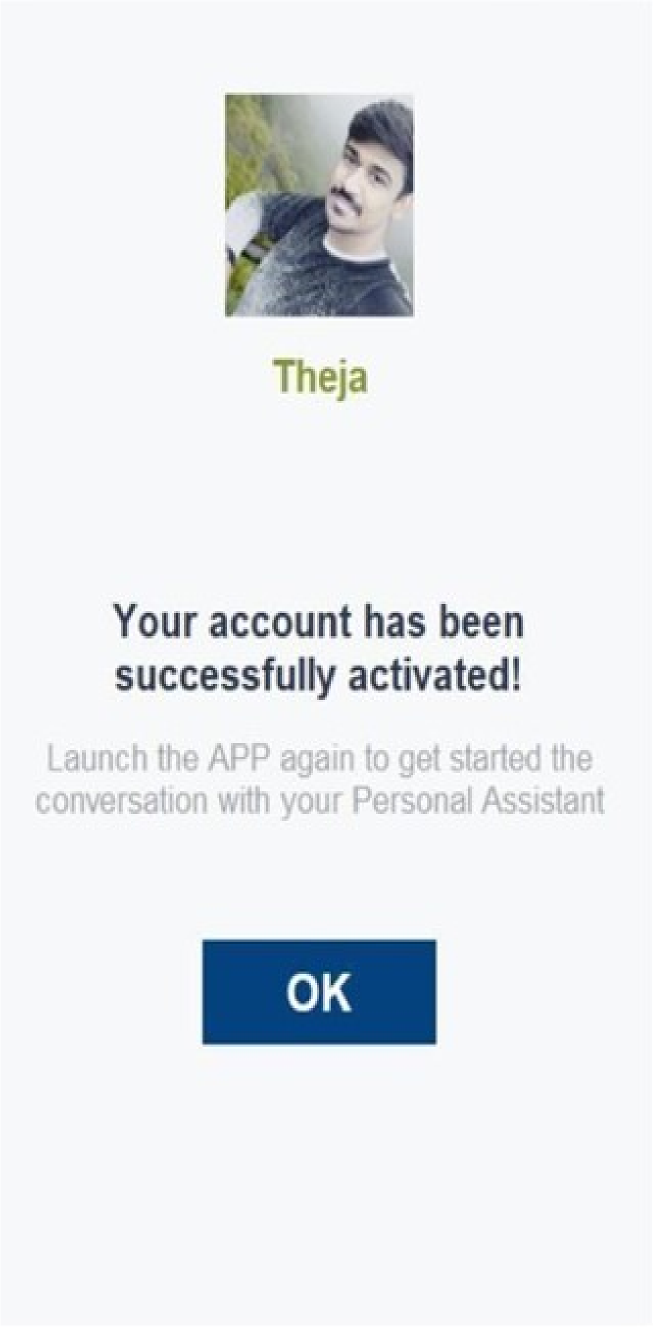
[5] Gk_, "Soul of the Machine: How Chatbots Work.", https://medium.com/@gk_/howchat-bots-workdfff656a35e2.

WEBSITES

- https://www.slideshare.net/VaishnaviKhandelwal6/final-presentation-on-chatbot
- https://www.researchgate.net/publication/351228837_Design_and_Development_of_CHATBOT_A_Review
- https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html
- https://blog.vsoftconsulting.com/blog/understanding-the-architecture-of- conversational-chatbot
- Building Chatbots with Python: Using Natural Language Processing and Machine Learning.

SNAP SHOTS

1. MAIN PAGE

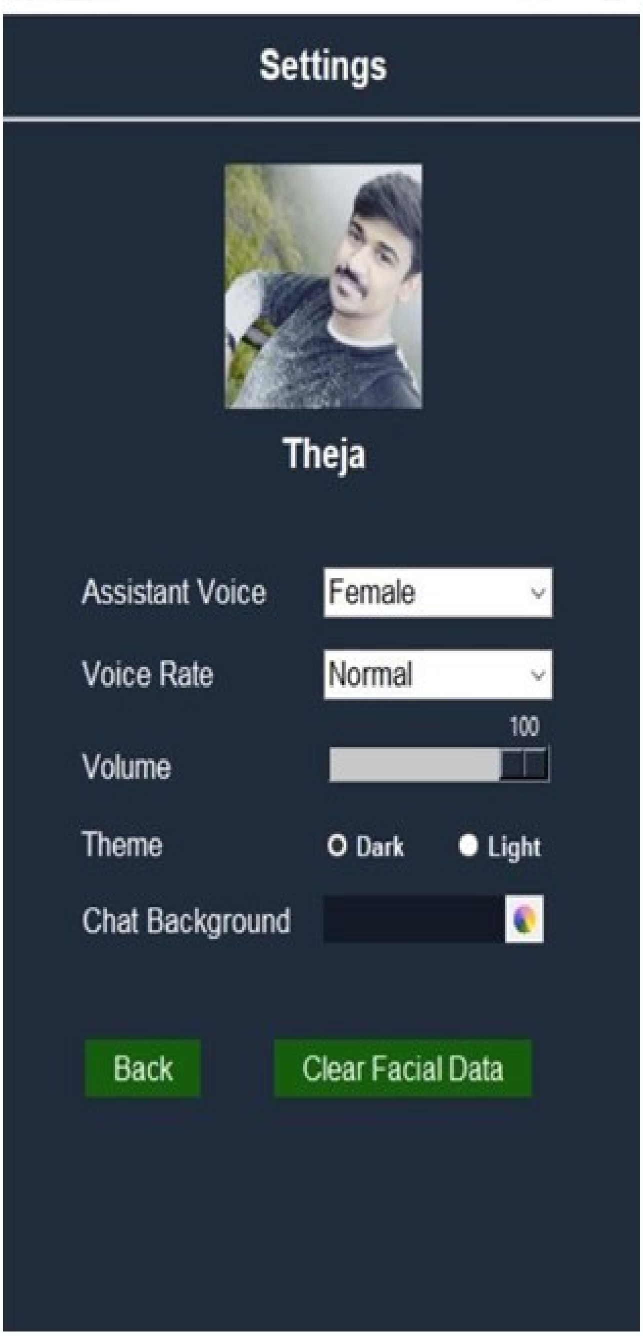


Theja

Your account has been successfully activated!

Launch the APP again to get started the conversation with your Personal Assistant

2. REGISTRATION PAGE



Settings

Theja

Assistant Voice: Female

Voice Rate: Normal

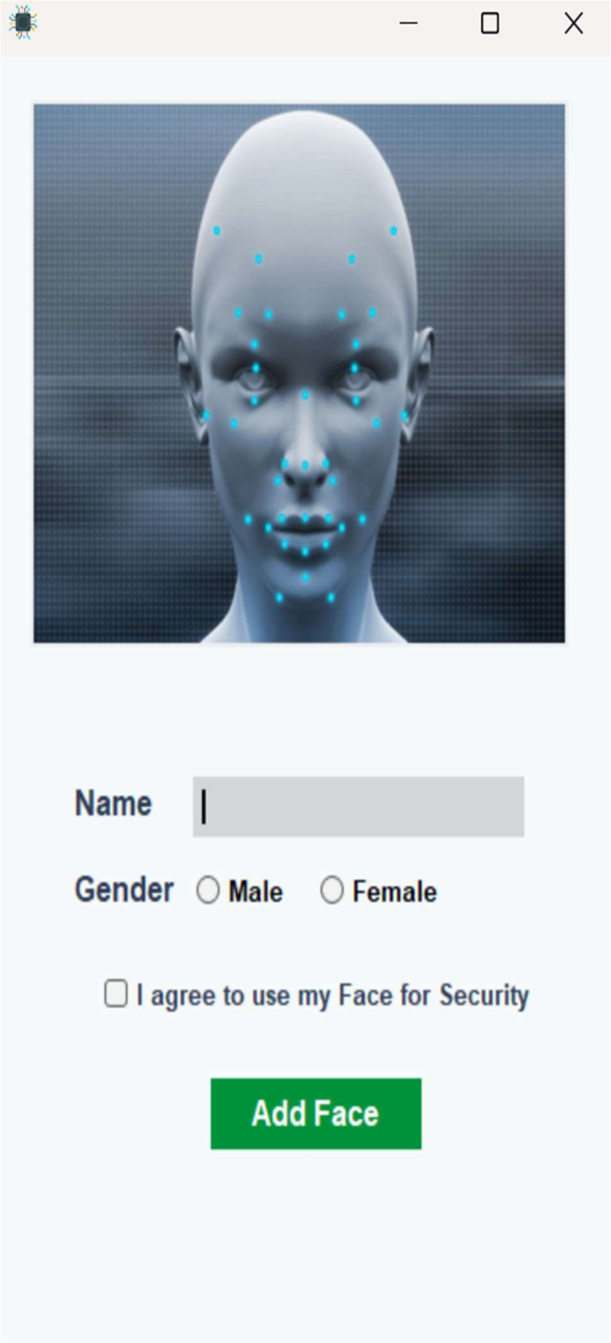
Volume: 100

Theme: ○ Dark ● Light

Chat Background

3. ACTIVATION PAGE

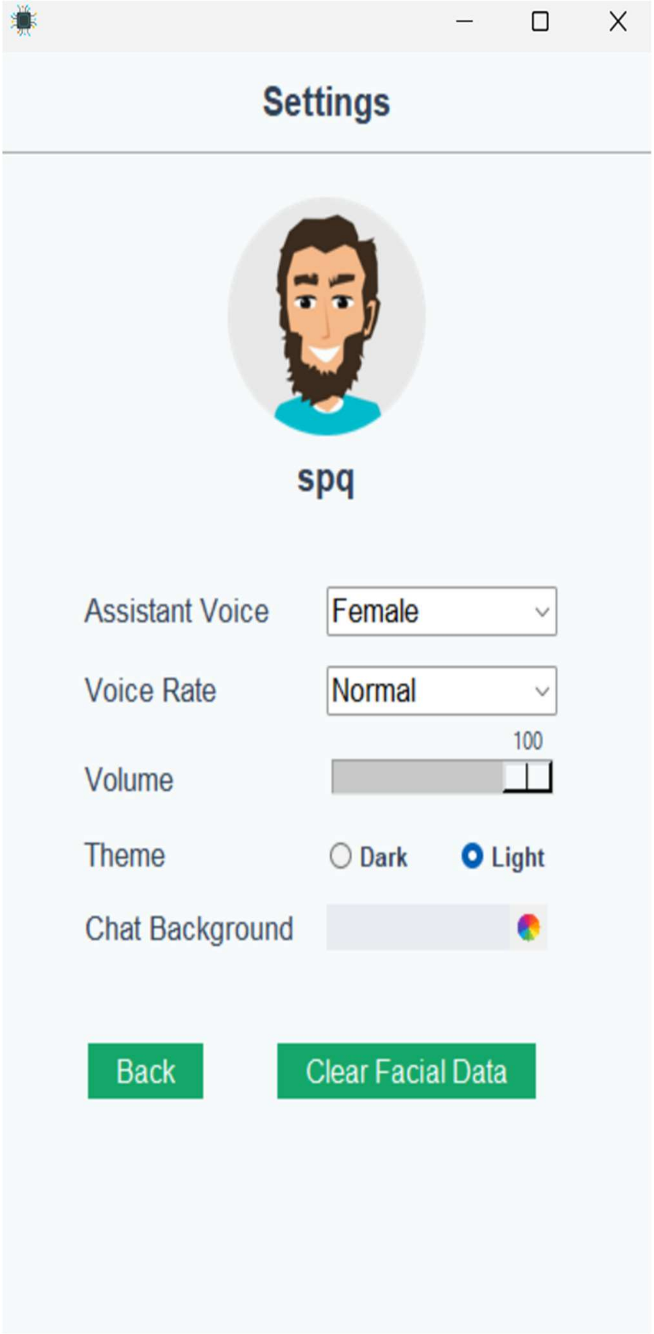
4. SETTINGS PAGE



Name |

Gender ○ Male ○ Female

□ I agree to use my Face for Security



Settings

spq

Assistant Voice: Female

Voice Rate: Normal

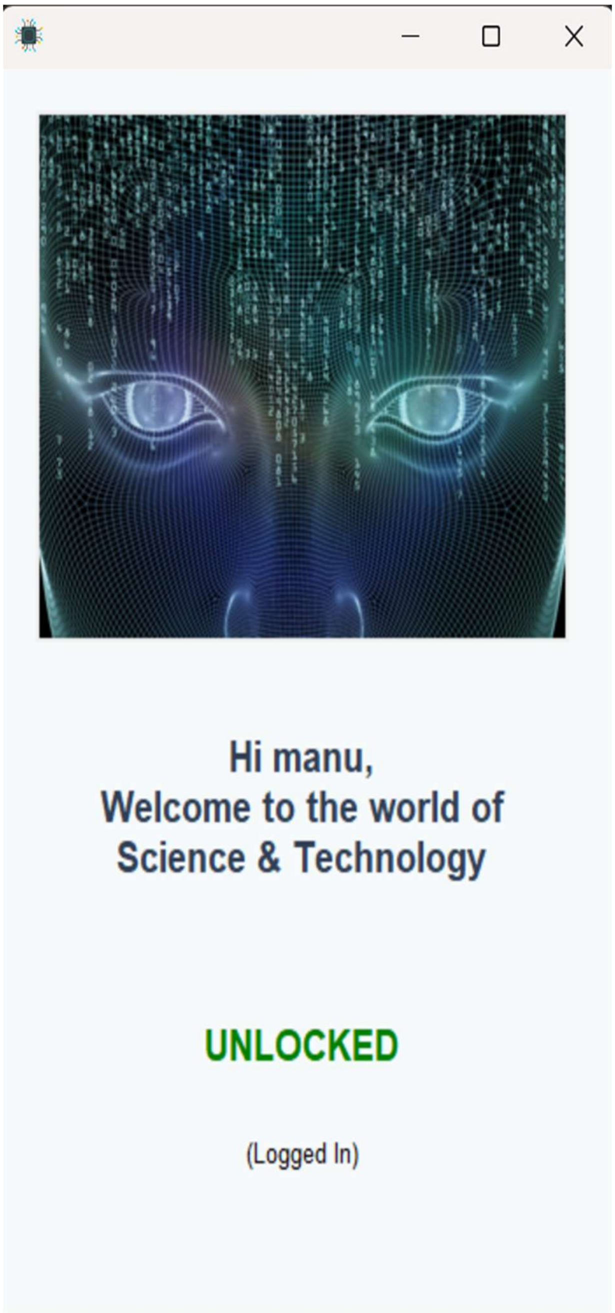
Volume: 100

Theme: ○ Dark ● Light

Chat Background

5. LOGIN PAGE

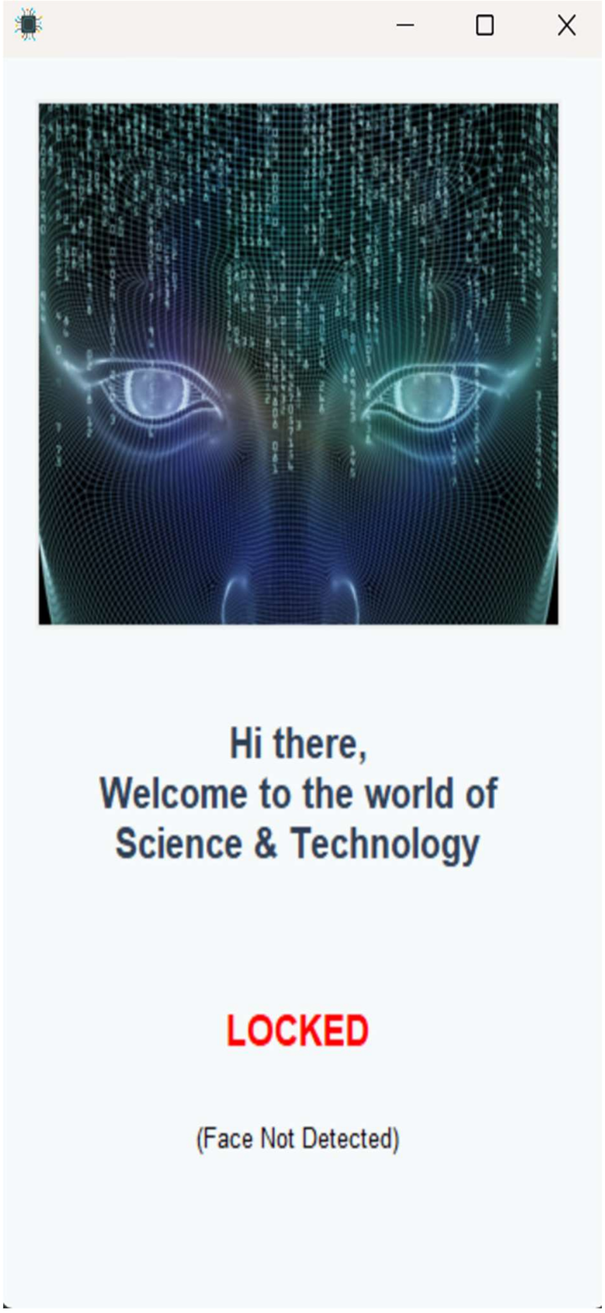
6. LOGOUT PAGE



Hi manu,
Welcome to the world of
Science & Technology

UNLOCKED

(Logged In)

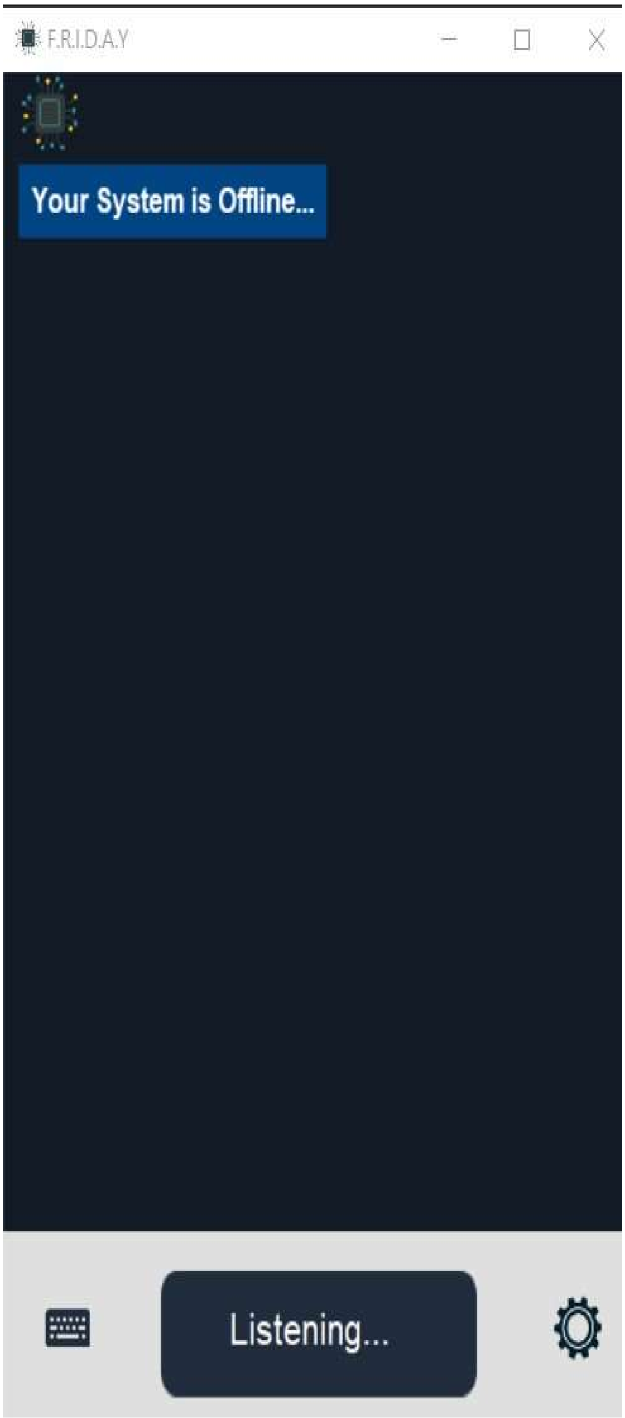


Hi there,
Welcome to the world of
Science & Technology

LOCKED

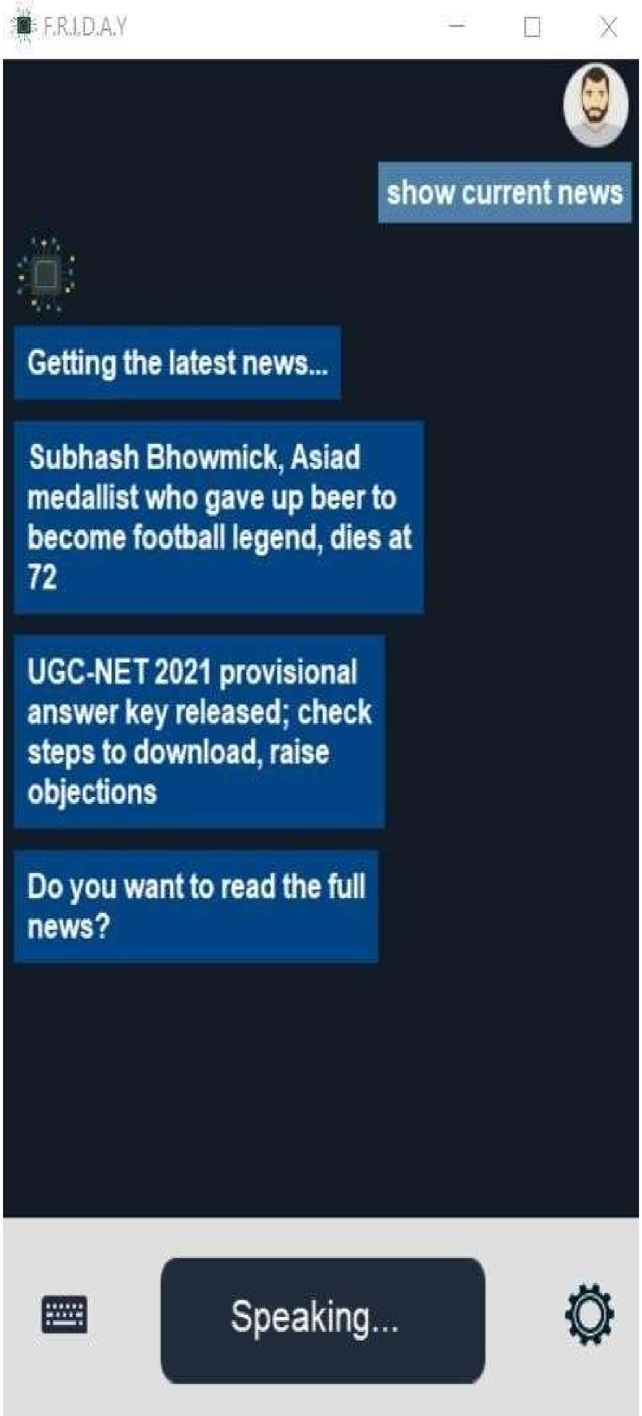
(Face Not Detected)

7. OFFLINE PAGE



Your System is Offline...

8. NEWS PAGE



show current news

Getting the latest news...

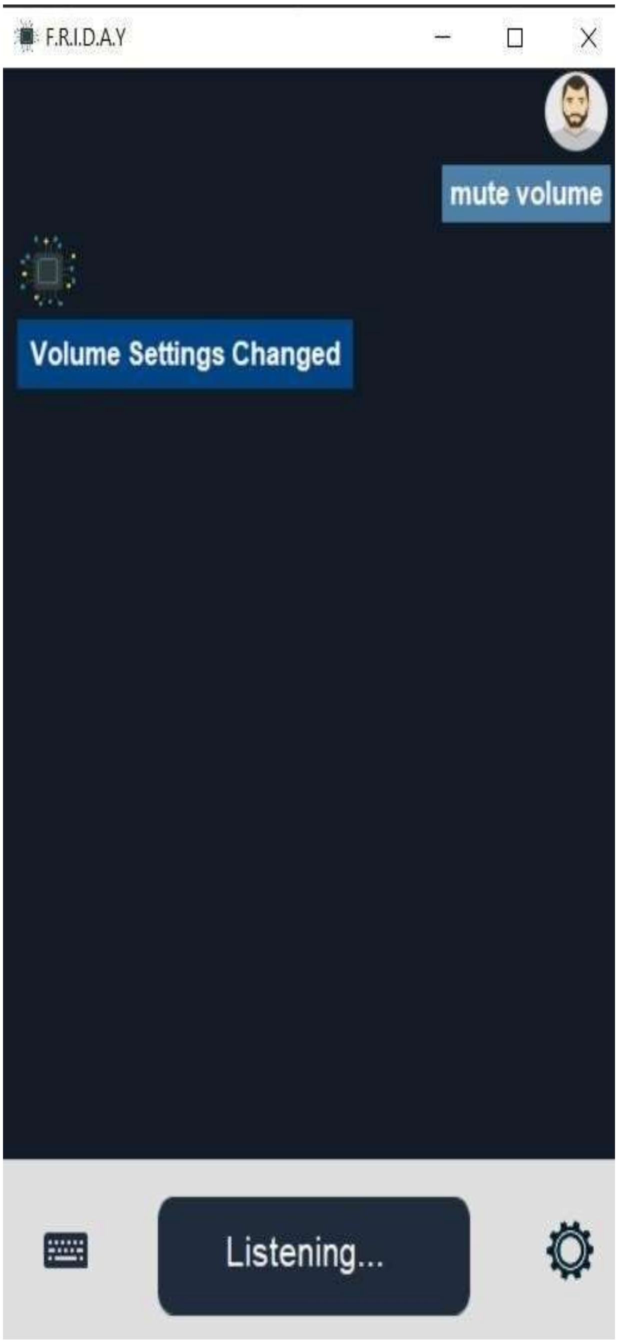
Subhash Bhowmick, Asiad medallist who gave up beer to become football legend, dies at 72

UGC-NET 2021 provisional answer key released; check steps to download, raise objections

Do you want to read the full news?

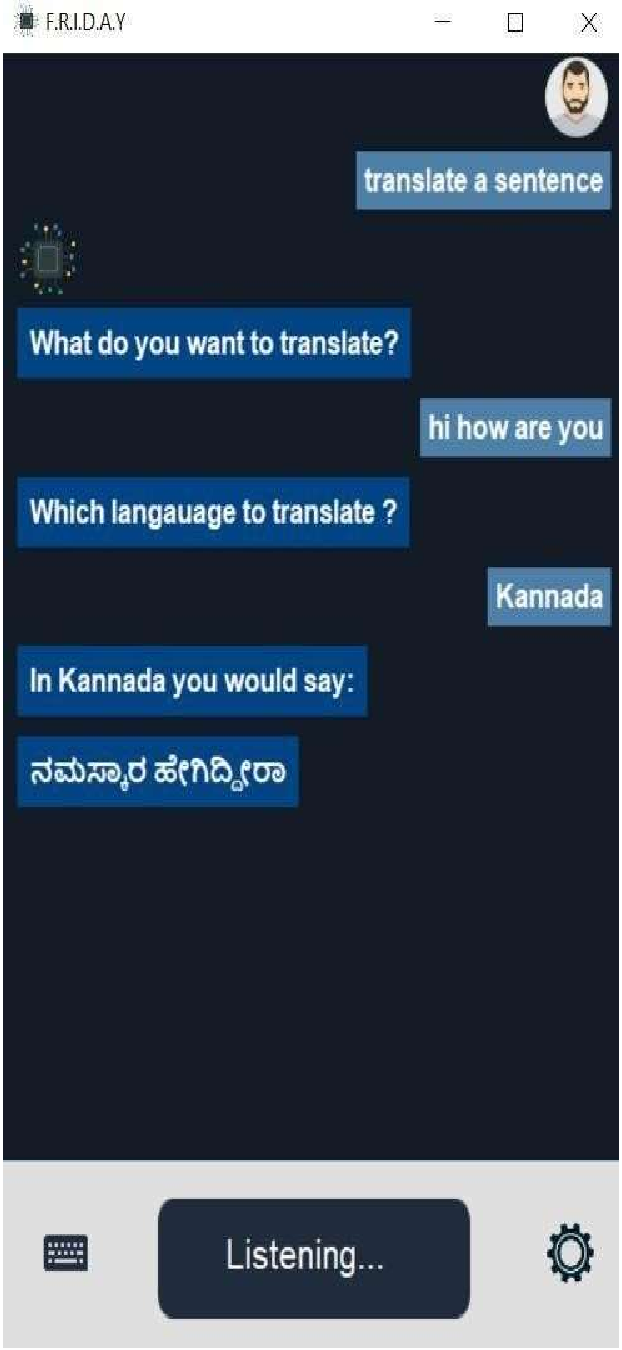
9. VOLUME CONTROL PAGE

10. TRANSLATION PAGE



mute volume

Volume Settings Changed



translate a sentence

What do you want to translate?

hi how are you

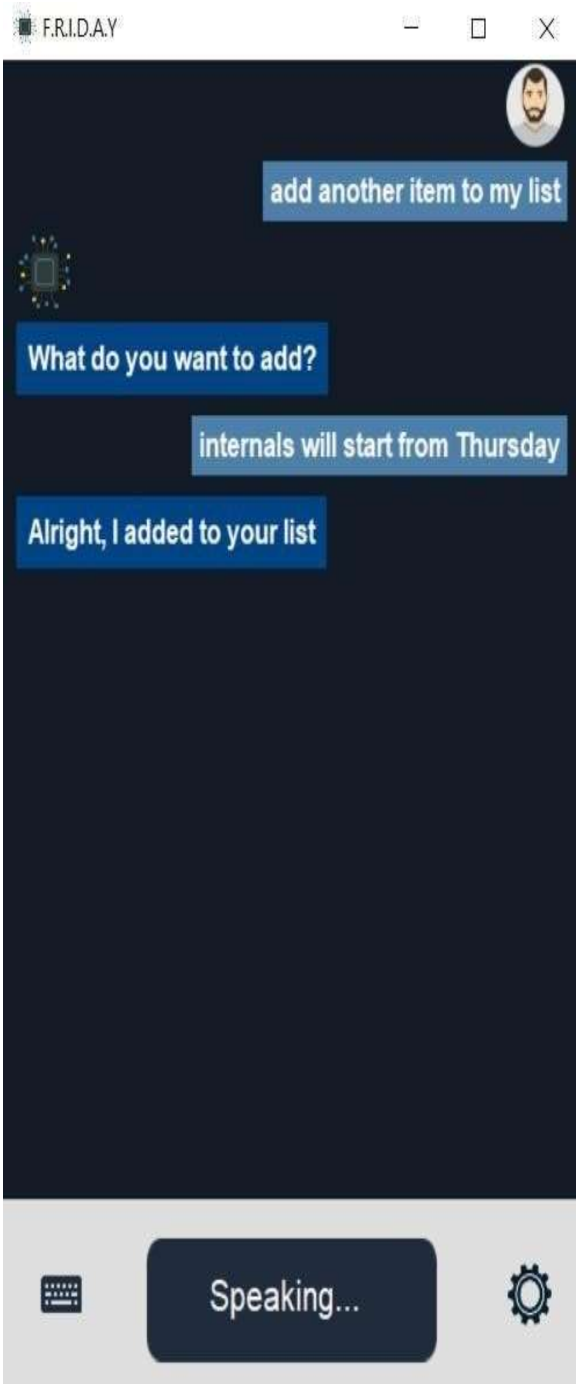
Which langauage to translate ?

Kannada

In Kannada you would say:

ನಮಸ್ಕಾರ ಹೇಗಿದ್ದೀರಾ

11. TO-DO LIST PAGE



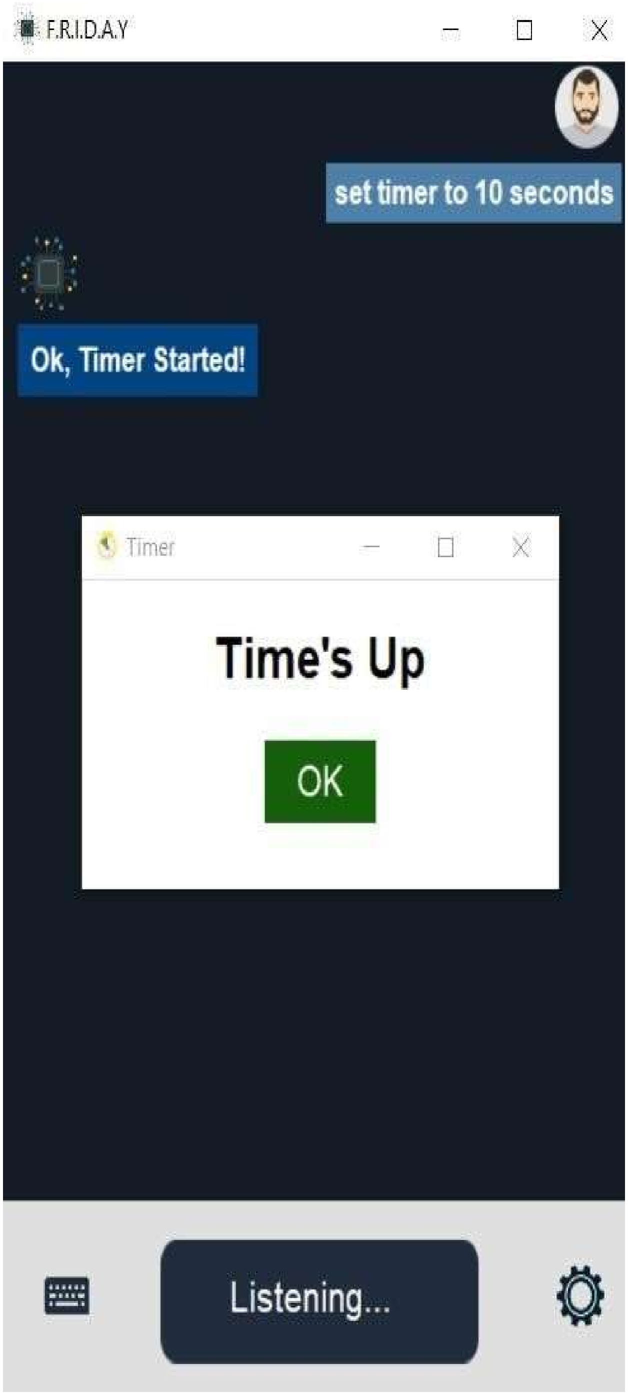
add another item to my list

What do you want to add?

internals will start from Thursday

Alright, I added to your list

12. SET TIMER PAGE



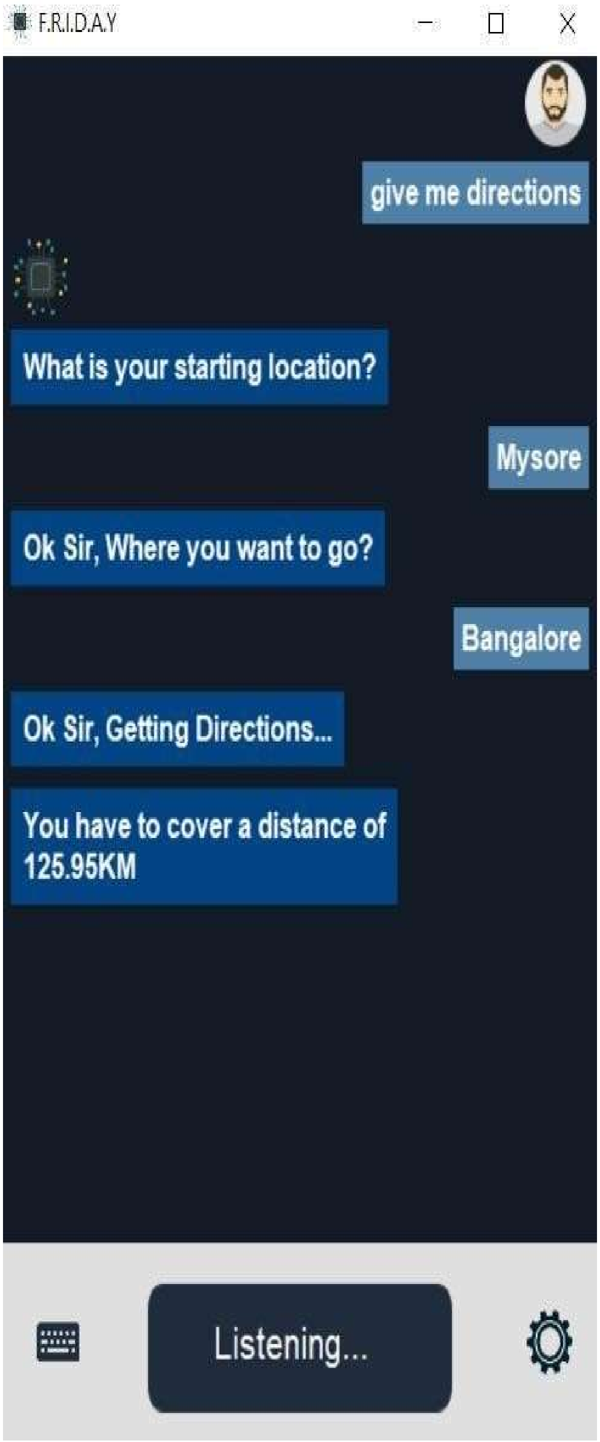
set timer to 10 seconds

Ok, Timer Started!

Timer

Time's Up

13. DIRECTIONS



give me directions

What is your starting location?

Mysore

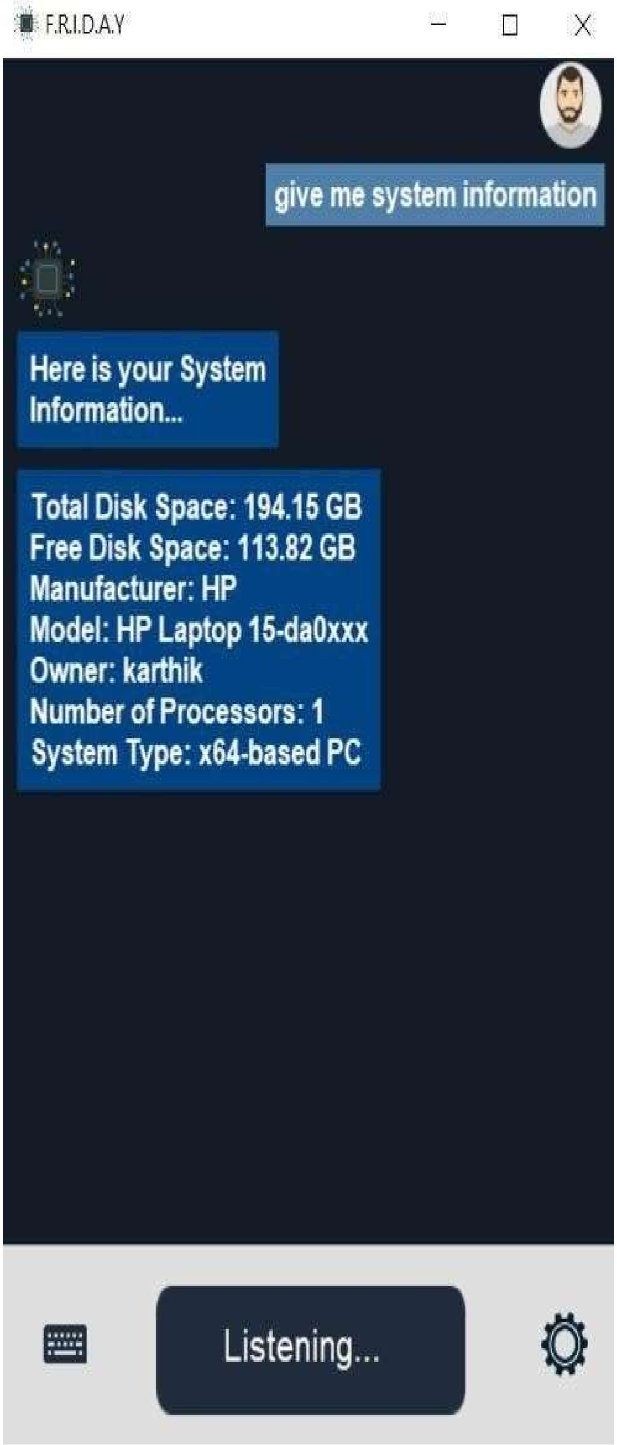
Ok Sir, Where you want to go?

Bangalore

Ok Sir, Getting Directions...

You have to cover a distance of 125.95KM

14. SYSTEM INFO PAGE



give me system information

Here is your System Information...

Total Disk Space: 194.15 GB
Free Disk Space: 113.82 GB
Manufacturer: HP
Model: HP Laptop 15-da0xxx
Owner: karthik
Number of Processors: 1
System Type: x64-based PC