**Desktop Assistant Documentation**

**Introduction**

The desktop assistant is a voice-controlled application designed to perform various tasks, provide information, and entertain users through speech recognition and text-to-speech capabilities. The assistant is equipped with features such as time and date retrieval, web browsing, information fetching from wikipedia, weather updates, news headlines, and more.

**Features**

1. Greetings and time-related functions

Wishme()

Greets the user based on the current time of the day.

Example command: "hello assistant."

Greetme()

Bids farewell to the user.

Example command: "goodbye."

Time()

Retrieves and speaks the current time.

Example command: "what is the time?"

Date()

Retrieves and speaks the current date.

Example command: "what is today's date?"

2. Voice interaction

Speak(audio)

Converts text to speech using the pyttsx3 library.

Example command: "tell me a joke."

3. System actions

Screenshot()

Captures and saves a screenshot using pyautogui.

Example command: "take a screenshot."

Lock\_screen()

Locks the windows screen using the ctypes library.

Example command: "lock the screen."

Sleep\_system()

Puts the system to sleep using the ctypes library.

Example command: "put the system to sleep."

4. Speech recognition and user commands

Takecommand()

Captures user speech, recognizes it using google's speech recognition api, and returns the recognized text.

Example command: "what can you do for me?"

5. Information retrieval

Wikipedia integration

Allows the assistant to search and provide summaries from wikipedia.

Example command: "tell me about artificial intelligence."

Weather information

Fetches and speaks the current weather information based on the user's location using the openweathermap api.

Example command: "what's the weather like today?"

News headlines

Retrieves and speaks top news headlines using the news api.

Example command: "tell me the latest news."

6. Web browsing

Web browser opening

Opens specified websites (youtube, google, stack overflow) and performs searches on microsoft edge.

Example commands:

"open youtube."

"search for python tutorials on google."

Custom website opening

Opens a website specified by the user by extracting the website name from the command.

Example command: "search stack overflow."

7. Entertainment

Youtube song search

Searches and plays songs on youtube using pywhatkit.

Example command: "play despacito."

Programming jokes

Fetches and tells programming-related jokes using the official joke api.

Example command: "tell me a programming joke."

8. Reminder and note-taking

Reminder setting

Allows the user to set reminders, which are saved in a file (reminders.txt).

Example command: "remind me to call john tomorrow."

Reminder retrieval

Reads out reminders set by the user.

Example command: "do i have any reminders?"

Note-taking

Allows the user to jot down notes, which are saved in a file (notes.txt).

Example command: "note down my meeting at 2 pm."

Note retrieval

Reads out notes saved by the user.

Example command: "do i have any notes for today?"

9. Miscellaneous

Screen locking, system sleep, screenshot capture, shutdown

Provides functionality to lock the screen, put the system to sleep, capture screenshots, and shut down the assistant.

Example commands:

"lock the screen."

"shutdown the assistant."

10. Exception handling

Various try, except blocks are used to handle exceptions gracefully and provide informative responses to the user.

Example command: "tell me a joke about elephants."

11. Continuous interaction

The assistant listens for commands continuously and responds based on user inputs.

Example command: "what else can you do?"