Voice Controlled Assistant - Project Report

Introduction:

The Voice Controlled Assistant project demonstrates the integration of speech recognition and text-to-speech technology to build an interactive AI-based assistant. The system listens to user commands, interprets them, and responds with appropriate actions such as telling the time, opening Google, narrating jokes, reading news, and saving notes

Abstract:

This project provides a hands-free way to interact with computers using natural voice commands. By combining Python libraries for speech recognition, text-to-speech, and web integration, the assistant can perform everyday tasks. The project highlights the practical application of Artificial Intelligence and Natural Language Processing concepts. This assistant not only improves accessibility for differently-abled users but also demonstrates how automation can make daily tasks easier.

Tools:

Used The following tools and libraries were used to build the assistant:-
Python 3
- speech_recognition (for voice-to-text conversion)
- pyttsx3 (for text-to-speech output)
- webbrowser (to open websites)
- datetime (to tell the current time)
- pyjokes (to generate jokes)
- feedparser (to fetch and read news headlines)
- Microphone and Speakers (as input/output devices)
- Google Speech Recognition API (for accurate speech-to-text)

Steps Involved in Building the Project:

1. Capturing voice input using the microphone.
2. Converting speech into text using Google Speech Recognition API.
3. Mapping recognized text to predefined commands (Google search, time, joke, news, notes).
4. Providing spoken feedback using pyttsx3.
5. Implementing additional modules for jokes, news, and note-taking.
6. Running the program in a continuous loop until the user says 'exit'.
7. Testing the system with different accents and background noise to improve reliability.
8. Saving user notes in a local file for future reference

Conclusion:

The Voice Controlled Assistant successfully demonstrates how speech recognition and text-to-speech can be combined to create an intelligent and interactive system. The project is scalable and can be extended with features like weather updates, music control, or integration with AI

chat models. It provides a foundation for developing more advanced personal assistant applications.

In conclusion, the project enhances human-computer interaction by enabling natural communication. The assistant shows how even beginner-friendly Python libraries can be used to build practical and useful real-world applications