­Project: Jarvis AI - Phase 1 (Core Assistant)

# 1. Project Description

Jarvis AI is a smart voice-controlled virtual assistant built using Python. This core version focuses on providing essential assistant features such as voice recognition, text-to-speech responses, web search, application launching, and system operations. This modular and scalable version will later be extended to IoT integration, mobile apps, and portable builds.

# 2. Tech Stack & Tools (Core)

• Language: Python

• Libraries: speech\_recognition, pyttsx3, pywhatkit, wikipedia, pyjokes, os, webbrowser, datetime

• IDE: VS Code / PyCharm

• Tools: Git, Mic, Speaker, Terminal/Console

• Optional (Future): Flask, OpenAI, paho-mqtt, PyInstaller, Firebase

# 3. Folder Structure

Jarvis-AI/  
│  
├── main.py # Entry point  
├── core/ # Core functionalities  
│ ├── speech\_engine.py # Handles text-to-speech  
│ ├── recognizer.py # Handles speech input  
│ └── command\_parser.py # Detects intent  
├── commands/ # Handles specific commands  
│ ├── web\_tasks.py # Google, YouTube, Wikipedia  
│ └── system\_control.py # Open apps, get time/date, shutdown  
├── ui/ # GUI (to be added later)  
│ └── main\_window.py  
├── config/ # Settings, API keys  
│ └── settings.json  
├── assets/ # Audio, icons  
└── requirements.txt # Project dependencies

# 4. Module-wise Plan

• Speech Recognition – Listen to voice and convert it to text.

• TTS Engine – Jarvis replies using pyttsx3.

• Command Parser – Match commands like 'open YouTube', 'what time is it' etc.

• Web Tasks – Search Google, play YouTube, get Wikipedia summaries.

• System Tasks – Open applications, get date/time, shutdown system.

• GUI Module – To be integrated later using Tkinter or PyQt.

• Portable Build – To be added later using PyInstaller or AppImage.

• IoT & Mobile Integration – Planned in Phase 2.

# 5. Python Libraries to be Used (Initial Phase)

• speech\_recognition – For converting speech to text

• pyttsx3 – For text-to-speech output

• pywhatkit – For YouTube, Google, WhatsApp tasks

• wikipedia – To fetch topic summaries

• pyjokes – For entertainment/jokes

• datetime – To fetch date and time

• os, webbrowser – For system and web tasks