

# Raspberry pi based Hindi voice assistant system designed specifically to use it in the Village Panchayat office

## 1.Main.py

```
# main.py
import sounddevice as sd
import queue
import json
from vosk import Model, KaldiRecognizer
from tts_wrapper import speak_hindi
from command_logic import process_command

MODEL_PATH = "models/vosk-model-small-hi-0.22"
SAMPLE_RATE = 16000

WAKE_WORDS = ["नमस्ते", "नमस्कार"]
EXIT_COMMANDS = ["सिस्टम बंद", "बंद करो", "असिस्टेंट बंद"]

q = queue.Queue()

def callback(indata, frames, time, status):
    q.put(bytes(indata))

print(" ♦ Loading Hindi VOSK model...")
model = Model(MODEL_PATH)
recognizer = KaldiRecognizer(model, SAMPLE_RATE)

print("🎤 Wake word mode active (Say: नमस्ते)")
speak_hindi("नमस्कार। पंचायत सहायता प्रणाली तैयार है। कृपया नमस्ते कहकर सवाल पूछें।")

with sd.RawInputStream(
    samplerate=SAMPLE_RATE,
    blocksize=8000,
    dtype="int16",
    channels=1,
    callback=callback
):
    while True:
        data = q.get()
```

```

if recognizer.AcceptWaveform(data):
    result = json.loads(recognizer.Result())
    text = result.get("text", "").strip()

    if text:
        print("📝 आपने कहा:", text)

        # ✅ Wake word detection
        wake_detected = False
        for word in WAKE_WORDS:
            if word in text:
                wake_detected = True
                text = text.replace(word, "").strip()
                break

        # ❌ Ignore if wake word not present
        if not wake_detected:
            print("🔊 Wake word not detected — Ignoring")
            continue

        # ✅ If user only says wake word
        if not text:
            speak_hindi("जी, बोलिए।")
            continue

        # ✅ Exit check (SAFE)
        if any(cmd in text for cmd in EXIT_COMMANDS):
            speak_hindi("धन्यवाद। सिस्टम बंद किया जा रहा है।")
            break

        # ✅ Process Panchayat command
        response = process_command(text)

        print("🤖 जवाब:", response)
        speak_hindi(response)

```

## **2.Command logic.py**

```
# command_logic.py
```

```

def process_command(text):
    text = text.strip()

```

# 🏛️ Office opening time

if "पंचायत" in text and "खुले" in text:

return "पंचायत कार्यालय सुबह 9 बजे खुलता है।"

# 🏛️ Office closing time

if "पंचायत" in text and "बंद" in text:

return "पंचायत कार्यालय शाम 6 बजे बंद होता है।"

# 👤 Officer availability

if "अधिकारी" in text and "रोज" in text:

return "पंचायत अधिकारी रोज सुबह 9 बजे से शाम 6 बजे तक उपलब्ध रहते हैं।"

# 📄 Services available

if "कौन" in text and "काम" in text:

return (  
"यहाँ जन्म प्रमाण पत्र, मृत्यु प्रमाण पत्र, "  
"पेंशन, राशन कार्ड, सरकारी योजनाएं और शिकायत दर्ज की जाती हैं।"  
)

# 🍼 Birth certificate procedure

if "जन्म" in text and "प्रमाण" in text:

return (  
"काउंटर 1 से फॉर्म लें। "  
"माता पिता का आधार कार्ड लगाएं। "  
"जन्म पर्ची लगाएं। "  
"काउंटर 2 पर जमा करें। "  
"7 से 10 दिन में प्रमाण पत्र मिलेगा।"  
)

# 🧐 Pension scheme

if "पेंशन" in text:

return "आपकी उम्र 60 साल से ज्यादा है क्या?"

# ✅ Pension eligibility confirmation

if "हाँ" in text or "हां" in text:

return "आप पात्र हैं।"

# 🙌 Greeting

if "नमस्ते" in text or "नमस्कार" in text:

return "नमस्कार। पंचायत सहायता प्रणाली में आपका स्वागत है। कृपया अपना सवाल पूछें।"

# ❌ Exit

if "बंद" in text:

return "धन्यवाद। पंचायत सहायता प्रणाली का उपयोग करने के लिए।"

```
return "कृपया अपना सवाल दोबारा पूछें।"
```

### **3.tts wrapper.py**

```
# tts_wrapper.py  
import os
```

```
def speak_hindi(text):  
    os.system(f'espeak -v hi "{text}")
```