❖ SPEECH RECOGNITION TECHNIQUES

➢ Speech-Based Data Entry System

Project-1

Submitted by,

Kaviya. S

Objective:

To build a system where users can input structured data (e.g., name, age, diagnosis) using voice commands, which the system recognizes, parses, and stores in a DataFrame or database, and optionally exports to PDF, Excel, or an HER format.

Project Workflow:

1. Voice Input
- ❖ Capture voice via microphone or audio file.
- ❖ Use a speech recognition engine (like OpenAI Whisper, Google Speech, or Python's speech_recognition).

2. Transcription
- ❖ Convert audio to text.
- ❖ Example: "Name John Smith Age 45 Diagnosis Hypertension"

3. NLP Parsing
- ❖ Extract key-value pairs using rules or NLP (e.g., regex, spaCy).
- ❖ Store in structured format: name, age, diagnosis, etc.

4. Store & Export

- ❖ Save to Pandas DataFramPDE
- ❖ Export to CSV, Excel, or PDF

- Optional: store in SQLite or other databases

Technologies:

1. Python
2. Whisper / SpeechRecognition
3. Pandas
4. FPDF or openpyxl
5. spaCy (optional for NLP parsing)
6. Streamlit (optional GUI)

Sample Code: Speech → Structured Entry → PDF

```
Import whisper

Import pandas as pd
```

```
Import re
From fpdf import FPDF

# Load Whisper model
Model = whisper.load_model(“base”)

# Transcribe audio
Result = model.transcribe(“entry1.wav”)
Text = result[“text”]
Print(“Transcript:”, text)

# Simple regex parsing (you can improve this with NLP)
Def parse_info(text):
  Info = {
    “Name”: re.search(r”(Name|name)\s+([A-Za-z ]+)”, text),
    “Age”: re.search(r”(Age|age)\s+(\d+)”, text),
    “Diagnosis”: re.search(r”(Diagnosis|diagnosis)\s+([A-Za-z ]+)”, text)
  }
  Return {
    “Name”: info[“Name”].group(2) if info[“Name”] else “”,
    “Age”: info[“Age”].group(2) if info[“Age”] else “”,
    “Diagnosis”: info[“Diagnosis”].group(2) if info[“Diagnosis”] else “”
  }

Parsed_data = parse_info(text)
```

```
# Store in DataFrame
Df = pd.DataFrame([parsed_data])
Print(df)

# Export to PDF
Pdf = FPDF()
Pdf.add_page()
Pdf.set_font("Arial", size=12)
Pdf.cell(200, 10, txt="Speech-Based Data Entry", ln=True, align='C')

For key, value in parsed_data.items():
   Pdf.cell(200, 10, txt=f"{key}: {value}", ln=True)

Pdf.output("entry_data.pdf")
```

Project Deliverables

1. Audio file samples
2. Transcription + structured data
3. PDF or CSV output
4. Optional UI (Streamlit or Tkinter)

Extensions

1. Real-time microphone input
2. Error correction or confirmation prompts
3. Use in healthcare, form entry, inventory, etc.
