**Deployment Instructions**

**Deployment Target:** Local (desktop) or cloud-based web app (Gradio-based demo)

**System Requirements**

**Hardware**

* At least 4 GB RAM
* Microphone-enabled device (for audio capture)
* Stable internet connection (for LLM API calls)

**Software**

* Python 3.9+
* Virtual Environment (recommended)
* ffmpeg installed (required by whisper)

**Python Dependencies**

* All the dependencies are in requirements.txt file

***pip install -r requirements.txt***

* Ensure the .env file is present with these variables set:

***OPENAI\_API\_KEY=your\_openai\_key\_here***

***CLAUDE\_API\_KEY=your\_claude\_key\_here***

**Project Structure**

audio/ record\_audio.py

data/ logs/

samples/

docs/ architecture.pdf

cultural adaptation.pdf

deployment instruction.pdf

future roadmap.pdf

safety protocol.pdf

llm/ generator.py

prompts.py

stt/ convert\_text.py

tests/ test\_integration.py

test\_llm.py

test\_stt.py

test\_tts.py

tts/ tts.py

utils.py

.env

gradio\_interface.py

logger.py

main.py

README.md

requirements.txt

**Running the Web Interface (Gradio)**

Launch the interface locally using:

***python gradio\_interface.py***

This starts a web server with microphone input, transcription, LLM generation, and audio output.

**Running the CLI-Based Flow (for batch testing)**

To test individual audio samples and log results:

***python -m llm.generator***

Ensure data/samples/ contains WAV or MP3 files for testing.

**Security & Privacy**

* Logs are stored in data/logs/ in JSON format, respecting user privacy.
* No audio data is permanently stored unless for debugging.
* All keys are managed through the .env file and never committed to Git.

**Optional Deployment (Cloud)**

If deploying to Hugging Face Spaces or Streamlit Cloud:

* Replace local paths with relative cloud paths.
* Handle audio using in-memory buffers.
* Mask API keys via secrets/environment config.