# Elile Architecture Documentation

## 1. System Overview

Elile is a real-time, voice-only conversational AI designed to provide culturally sensitive mental health support in the Omani Arabic dialect.  
The system uses a decoupled, microservices-inspired architecture, separating the frontend (Streamlit) from the backend (FastAPI). This ensures scalability and maintainability.  
  
Key Characteristics:  
- Voice-only interaction from input to output.  
- Emotion-aware and culturally sensitive responses.  
- Designed for real-time usage with a sub-20s round-trip latency.  
- Modular, maintainable, and easily extensible architecture.

### Core Technologies

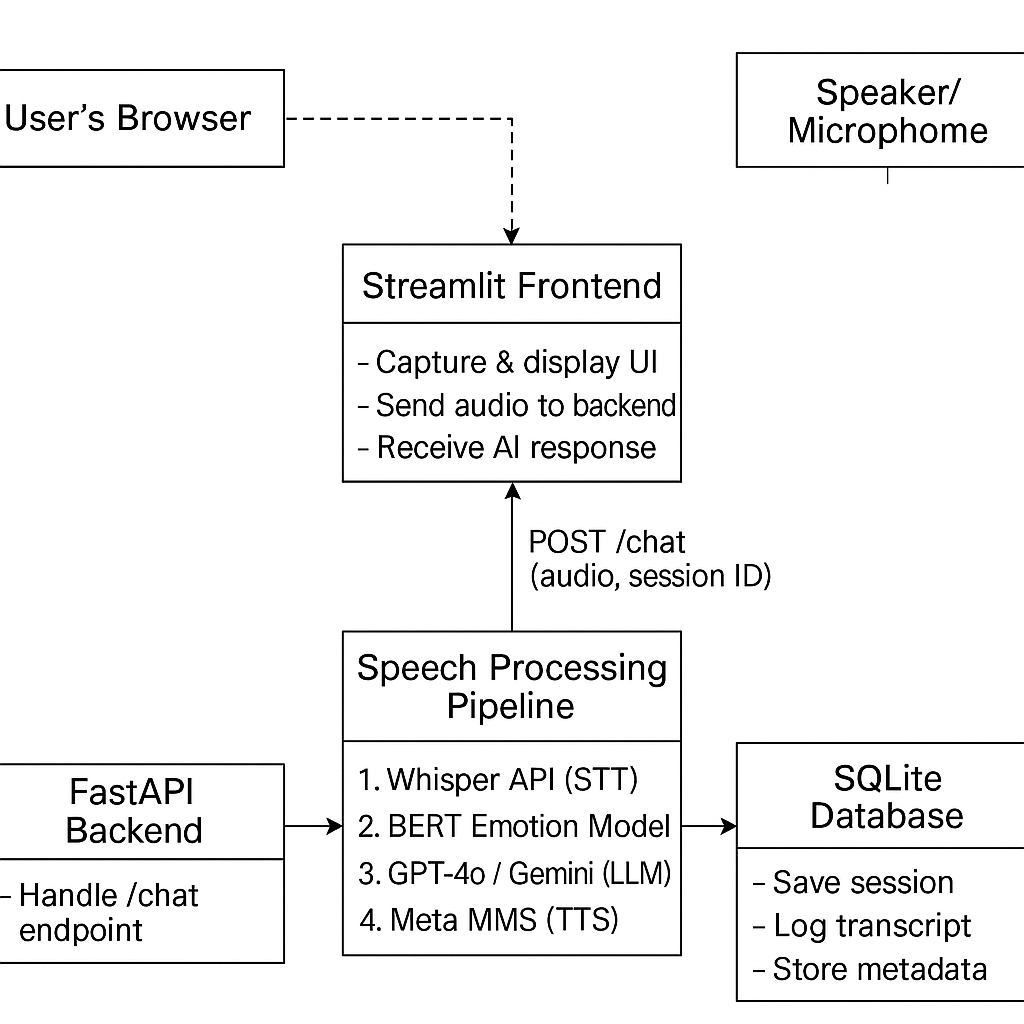
|  |  |
| --- | --- |
| Component | Technology |
| Frontend | Streamlit |
| Backend | FastAPI (Python 3.10) |
| AI Models | Whisper, BERT, GPT-4o/Gemini, MMS TTS |
| Database | SQLite |
| Deployment | Render.com |

## 2. System Architecture

Architecture Style:  
Decoupled architecture with microservices inspiration. Frontend and backend communicate via FAST APIs.

Main Components:

- Streamlit Frontend: Captures and plays audio, interacts with FastAPI backend, displays textual output  
- FastAPI Backend: Receives audio via POST /chat, orchestrates full pipeline, returns audio + logs to frontend  
- Speech Processing Pipeline:  
 1. STT – Whisper API for transcription  
 2. Emotion Detection – BERT model (HuggingFace)  
 3. LLM – GPT-4o primary, Gemini Flash fallback  
 4. TTS – Facebook MMS Arabic TTS (local)  
- Database (SQLite): Logs conversation metadata, supports simple analytics or future visualization



## 3. Data Flow

User → Streamlit UI → FastAPI Backend → AI Pipeline → SQLite → Streamlit

Steps:  
1. Voice Input: Audio is recorded in the Streamlit UI.  
2. API Request: Audio and session ID sent to FastAPI via /chat.  
3. STT Processing: Whisper API transcribes Arabic dialect audio to text.  
4. Emotion Classification: BERT model analyzes emotion (e.g., sadness, anger).  
5. LLM Response Generation: GPT-4o generates response; Gemini Flash used as fallback.  
6. TTS Synthesis: Response is converted to Arabic audio with MMS TTS.  
7. Response Playback & Logging: Audio is streamed back; metadata is saved.

## 4. Model Integration

|  |  |  |
| --- | --- | --- |
| Step | Model | Description / Integration Details |
| STT | Whisper API | OpenAI SDK. Prompt-enhanced. Stateless. Dialect-aware tuning for Omani Arabic. |
| Emotion | BERT | Hugging Face ‘bert-base-go-emotion’. Loaded on server startup. GPU-enabled. |
| LLM | GPT-4o / Gemini | GPT-4o (OpenAI) is default. Gemini Flash (Google) is fallback. Handled via SDKs. |
| TTS | MMS Arabic TTS | Facebook ‘mms-tts-arb’. Local HuggingFace pipeline for low-latency synthesis. |

## 5. Deployment Strategy

- Platform: Render  
- Backend: Dockerized FastAPI service  
- Frontend: Hosted Streamlit app  
- Model Hosting: Whisper via OpenAI Cloud API, BERT & MMS TTS hosted locally on Render instance with persistent storage

## 6. Future Enhancements

- Add multilingual support (e.g., English, Swahili)  
- Extend to female voice synthesis via voice cloning  
- Integrate real-time sentiment graphing  
- Enable offline mode for remote clinics  
- Introduce clinician portal for monitoring and triage