Blog-to-Podcast System Architecture

Frontend — Next.js (React / TypeScript)

Blog Input Form (URL / Text / Markdown)

Progress Tracker (Job Polling)

Script Preview Panel

Audio Player

Layer 1 — Input & Pre-processing

Scraper (httpx + newspaper3k / BS4 fallback)

Parser normalize_text() parse_markdown()

Chunker chunk_text() 800 tok / 100 overlap (tiktoken)

FastAPI /api/v1/

upload-blog
generate-script
generate-audio
generate-podcast
job/{id}
download/{id}

Layer 2 — Retrieval & Embedding

Embedding Service OpenAI text-embedding-3-small (via LiteLLM)

ChromaDB (persisted: ./data/chromadb) Metadata: chunk idx, title, URL

Layer 3 — Generation (RAG Agents via LangGraph)

Script Generator Agent (temp 0.7) Blog → Dialogue

Accuracy Agent (temp 0.3) Fact-check

Storytelling Agent (temp 0.6) Pacing & Cues

Engagement Agent (temp 0.6) Hooks & CTAs

Layer 4 — Audio Output

Dialogue Parser Split script → SpeakerSegments + cue extraction

Coqui XTTS v2 Voice Cloning Alex (HOST_A) Jordan (HOST_B)

Post-processing pydub normalize silence trim noisereduce

Assembler Merge segments + pauses MP3 192 kbps