Integrating Audio for Aya Vision

Bridging Audio and Vision - Team #16

The Problem:

Current voice-based AI assistants struggle with ***natural, fluid conversation***.

→ "Most systems lack **multilingual capabilities** and context awareness"

Goal: Empower Aya Vision with human-like conversation.

- **Real-time** interaction grounded in **multilingual** and **context-aware** understanding.

🔧 Our Approach:

- Build an **open-source** audio Software Development Kit (**SDK)** for developers.
- Integrate **state-of-the-art** TTS/STT models with Aya Vision.
- Designed for **modular, low-latency, and cross-language** support.

Our Solution Pt. I - Software Development Kit (SDK)

What We Built 🧠

- Modular STT/TSS SDK integrated with Aya Vision API
- Support for multimodal inputs (texts, images, audio inputs)
- Conversation history management

This is fully **open-source**: We want this to be a research **catalyst**—whether you're integrating new models, testing Aya Vision, or creating new projects.

STT Model Support: faster-whisper, whisper, wav2vec2, NeMo, Seamless
TTS Model Support: Google TTS, Groq TTS, Groq ASR

Technical Details ⚙️

- Flask-based web service with both **HTTP** endpoints and **WebSocket** support
- Multiple speech recognition models with dynamic loading
- **Real-time audio streaming** processing capability

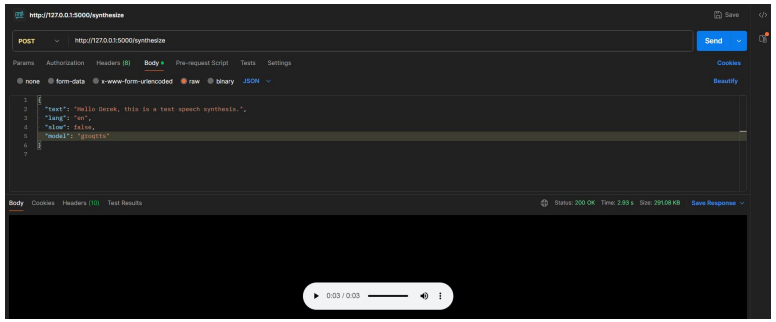


Figure 1: TTS synthesization endpoint response via Postman

Endpoints: /set-model, /available-models, /transcribe, /stream-audio, /synthesize, /audio/<filename>, /aya-response, /aya-response-tts

Our Solution Pt. II - Web App

Overview 🗂️

- Full-stack integration of Audio Integration Aya Vision SDK
- AI Chat App with multilingual support, real-time stt, tts, transcriptions & responses
- STT/TTS enabling/disabling, model selection, and customization

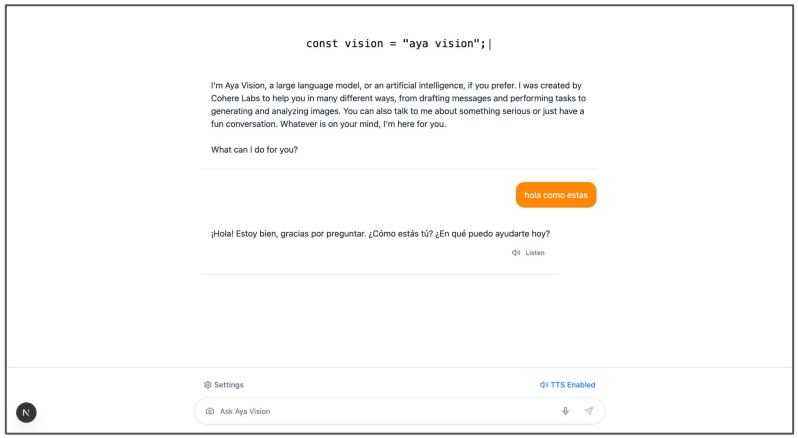


Figure 2: Web app UI/interface, spanish conversation

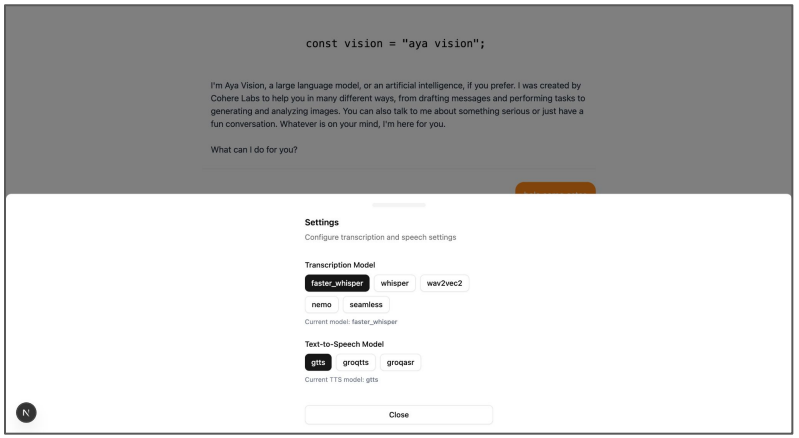


Figure 3: STT/TTS model selection

Figure 4: Real-time audio transcription, audio speech visualizer, image update, TTS enabler/disabler

Click here to watch the demo video:

Demo Video

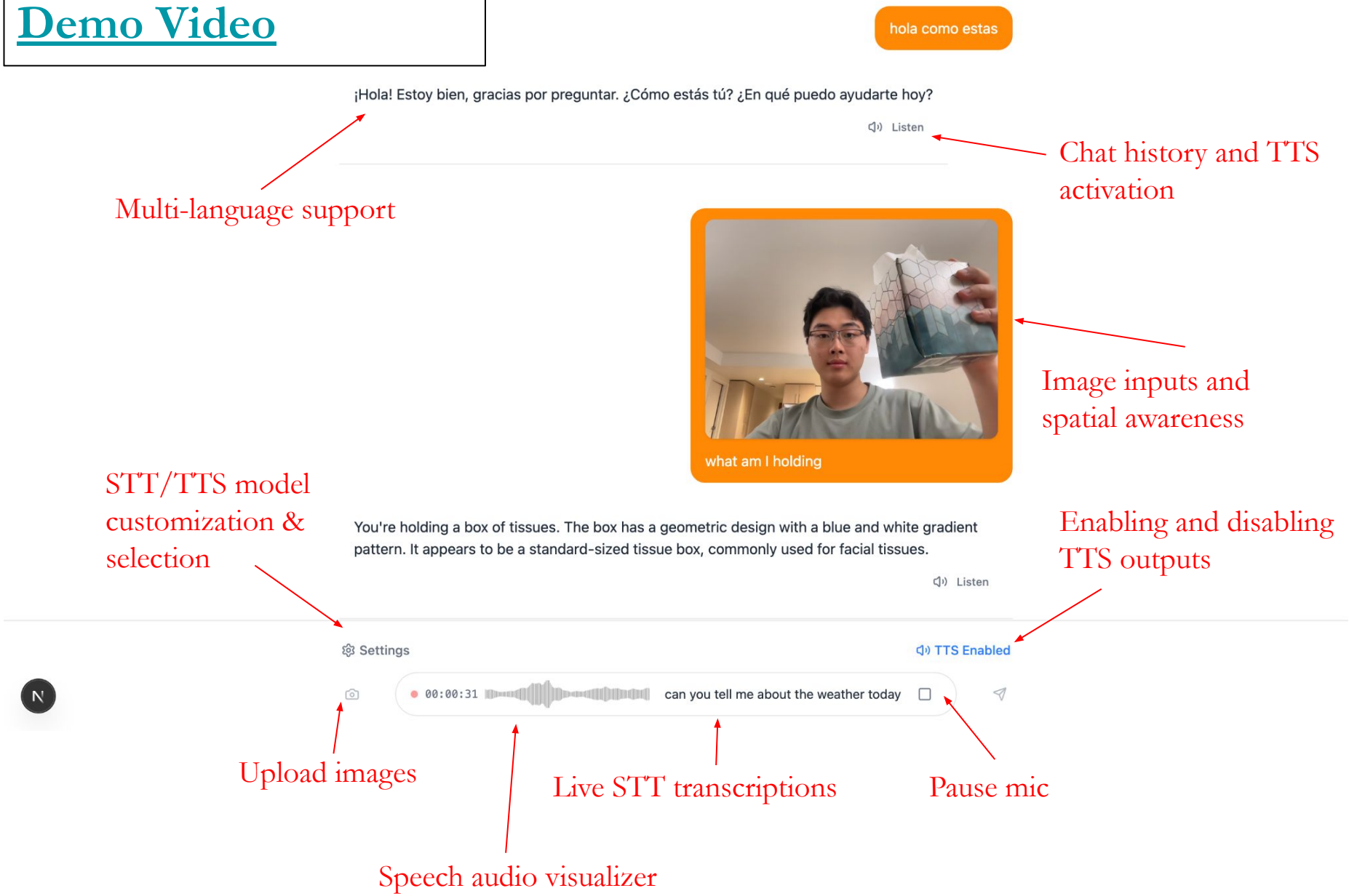


Figure 5: Web app usage example

System Architecture

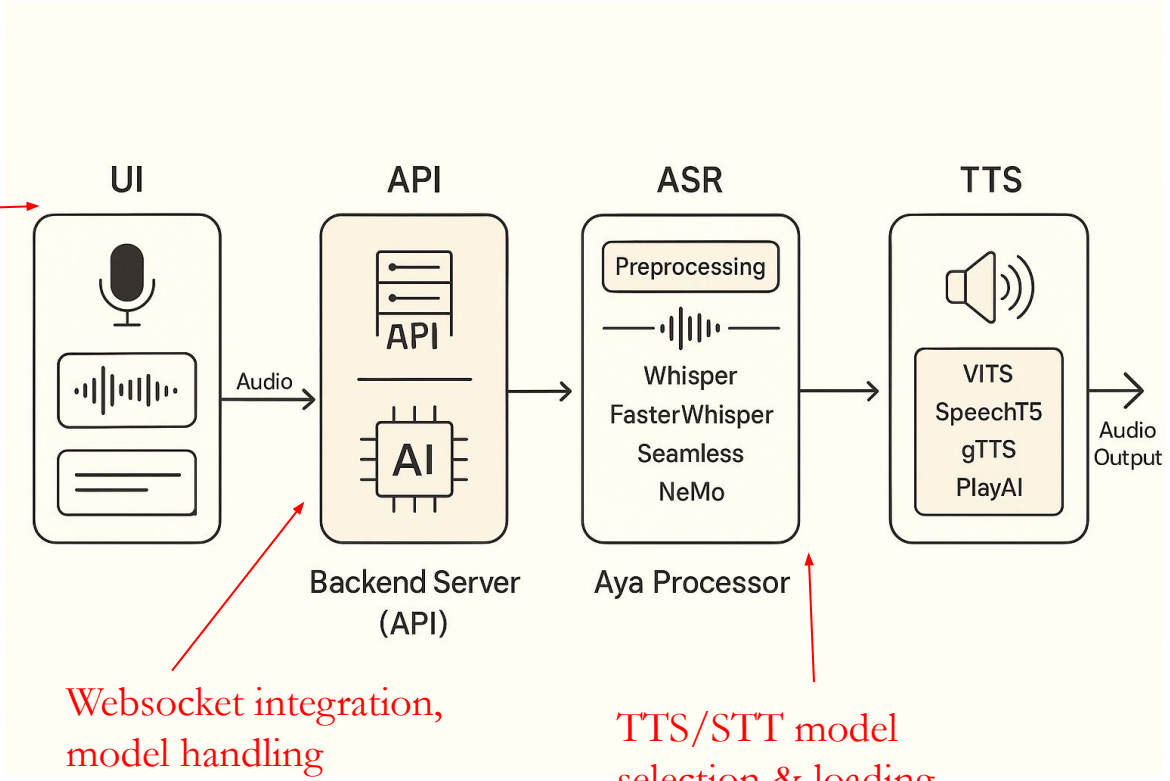
General Overview 📚

User Audio
→ Audio Preprocessing
→ STT model
→ Aya Vision
→ TTS Audio Output

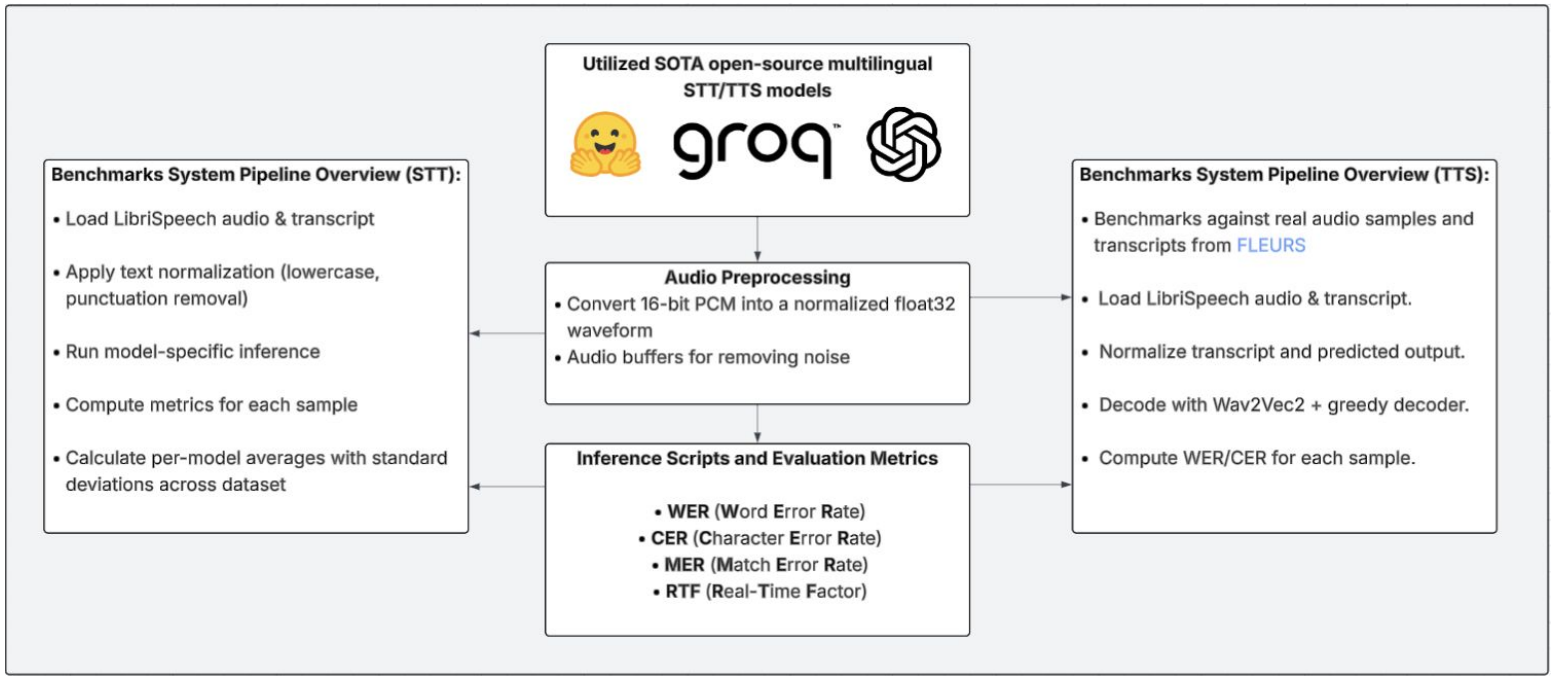
Frontend
+ Flask
backend

Built-in
preprocessing and
model toggle
system.

Multimedia capabilities: text, voice, audio, camera integration



Evaluation & Testing



Benchmarks System Pipeline Overview (STT):

1. Loaded LibriSpeech audio & transcript
2. Applied text normalization (lowercase, punctuation removal)
3. Ran model-specific inference
4. Computed metrics for each sample
5. Calculated per-model averages across dataset

STT Benchmarking Results

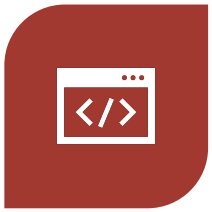
Model*	WER	CER	MER	RTF
Whisper (medium)	0.0653	0.0215	0.0627	0.6597
Wav2Vec2 (base)	0.0376	0.0108	0.0373	0.4331
Nemo (medium)	0.0319	0.0088	0.0312	0.0348
Seamless	0.1556	0.0921	0.1186	0.3244

* Showcasing the best performing configuration for each model
Tested on a sample of 100 Librispeech clean audio files, extensive benchmarking to be done soon

Key Innovations

MODEL-AGNOSTIC PIPELINE

REAL-TIME STT/TSS INTERACTION



LIGHTWEIGHT TESTING UI

FOUNDATIONS FOR MULTILINGUAL SUPPORT

Challenges

- **Latency & Sync:** Maintaining low-latency alignment between audio (STT) and visual (Aya Vision) inputs.
- **Noise Robustness:** Handling background noise and speech clarity issues.
- **Integration Complexity:** Ensuring compatibility between STT/TTS pipelines and Aya Vision's inference flow.

Next Steps

Optimize and Quantize Models for speed and reduced latency

Add more TTS/STT models for tests, benchmarks, integration

Create more sample apps using Aya Audio Integration SDK

Creating a **personal assistant** like Alexa that activates audio preprocessing and STT using a keyword name like ***"Hey Aya Vision"***.

Thank you

Thank you to Aya Expedition staff, mentors, and team!

Team:

- Derek Sheen, *Team Lead*
- Sanjana Kaza, *STT Sub- Lead*
- Oreoluwa Babatunde, *TTS Sub-Lead*
- Andrew Kim
- Ashay Srivastava
- Victor Olufemi, *Team Co-lead*
- Abhay Joshi, *Web Sub-lead*
- Ali Abdelkader
- Daniyal Ahmed

GitHub Repo