

Productionising SonicSerenity & SpeechSmith

A Comparative Analysis of Monoliths vs. Cloud-Native Serverless Ecosystems - Anoushka Tandon

SonicSerenity: <https://sonic-serenity.com/>

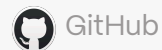
SpeechSmith: <https://www.speechsmith.net/>

Project Codes:

antandon/**SonicSerenity-Final**



1 Contributor
0 Issues
0 Stars
0 Forks



GitHub – antandon/SonicSerenity-Final

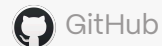
Contribute to antandon/SonicSerenity-Final development by creating an account on GitHub.



antandon/
SpeechSmith_1



1 Contributor
0 Issues
0 Stars
0 Forks



GitHub – antandon/SpeechSmith_1

Contribute to antandon/SpeechSmith_1 development by creating an account on GitHub.



Project Blueprint 1 — SonicSerenity

AI-Powered Personalised Music Therapy Platform

Problem

Mental health interventions are inaccessible, expensive, and generalised — traditional music therapy cannot adapt continuously to an individual's changing emotional state; existing wellness apps recommend playlists without physiological feedback.

Solution

AI-powered web platform delivering real-time personalised music therapy via computer vision, digital signal processing, and generative AI — co-designed with psychologists to align with established therapeutic principles.

Emotion Detection

Computer vision pipeline reads emotions in real-time and maps affect states to curated Solfeggio frequency tracks.

Live Dashboards

Real-time emotion graphs with therapist access portal; all session data persisted to user accounts.

GenAI Module — Lyria + GPT-4:

Conversational AI interface chains a GPT-4-powered reasoning engine with Google's Lyria model to generate novel therapeutic audio sequences based on user preferences — enabling fully adaptive, non-repetitive soundscapes beyond static playlist retrieval.

- PYTHON

FLASK

NGINX

AWS EC2

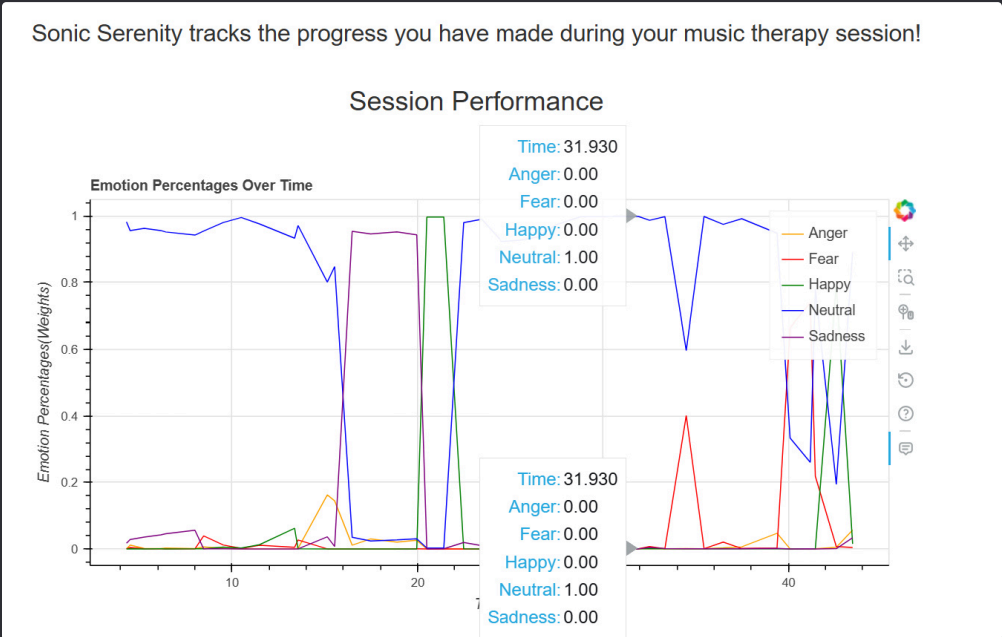
AWS REKOGNITION

FIREBASE FIRESTORE

MUSICGEN

LANGCHAIN

OPENCV



SonicSerenity

About UsOur ApproachOur ServicesTherapistSonicSoundsFAQContact Us

Welcome, Anoushka

Step 1: Choose Your Therapeutic Audio

Select from our curated collection of therapeutic sounds designed to promote emotional well-being.

Tibetan Singing Bowls

Ancient healing frequencies for deep relaxation

OM Chanting

Sacred mantras for spiritual connection

Crystal Bowls

Pure crystal frequencies for deep healing

Marimba Melodies

Gentle wooden tones for relaxation

Rain Sounds

Natural rainfall for peaceful meditation

Continue to Step 2

Experience SonicSounds AI

Discover the power of AI-driven sound therapy with SonicSounds. This advanced tool analyzes audio frequencies and provides therapeutic sound recommendations based on cutting-edge machine learning algorithms.

I'm glad you're ready to find some calm!

Track: Calming piano melodies for anxiety relief.

Here's your track! Use the Audio Controls below to adjust tempo, pitch, volume, and more.

Here's your track! Use the Audio Controls below to adjust tempo, pitch, volume, and more.

Message

Tell me how you're feeling or what kind of music you need...

Send

Recommended Track

0:00

1:20

Audio Controls

Adjust the generated music with real-time audio processing:

Apply effects instantly as you move sliders (may be slower)

Real-time Processing

Tempo

0.5=Half Speed, 1.0=Original, 2.0=Double Speed

1

Brightness

Negative=Darker, Positive=Brighter

0

Pitch Shift

Semitones (-12 = octave down, +12 = octave up)

0

Harmonic/Percussive Balance

0.0=All Percussive, 1.0=All Harmonic

0.5

Volume

Volume multiplier

1

Apply Effects

Reset

SonicSerenity: Host & Network Gateway

Compute, Proxy, and Architectural Paradigm

1 Compute

Single Amazon EC2 instance inside an isolated Python venv - prevents OS package corruption and ensures reproducible deployments.

2 Network Gateway

NGINX Reverse Proxy handles SSL/TLS on Port 443 HTTPS, shielding Flask from direct public exposure and DDoS vulnerabilities.

3 Why Not Direct Flask

Flask's built-in server is development-only — it cannot handle concurrent traffic securely at production scale.

4 Architectural Paradigm

Monolith (new_therapy.py) — eliminates inter-service latency, requires one server instance, and enables straightforward state sharing across cores.

Infrastructure View

The screenshot displays the AWS Management Console interface for an EC2 instance. The left sidebar shows navigation options like Dashboard, AWS Global View, Events, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Capacity Manager, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, and Key Pairs. The main content area shows the 'Instance summary' for 'i-0cb19abef5f9a972 (sonicserenity-app)'. It includes details such as Instance ID, IP address, Instance state (Running), Public IP address (35.154.76.165), Private IP address (172.31.41.12), Instance type (t2.micro), VPC ID, Subnet ID, and Instance ARN. Below the summary, there are tabs for Details, Status and alarms, Monitoring, Security, Networking, Storage, and Tags. The 'Details' tab is active, showing Instance details like AMI ID, AMI name, and Platform details.

```
[[{"sonicserenity.therapy@gmail.com": "1760441504", "marimba.wav"}], [{"patient_name": "Anoushka Tandon", "therapist_email": "yash.bhardwaj5040@gmail.com", "patient_email": "", "sessions": [{"sonicserenity.therapy@gmail.com": "1760441504", "marimba.wav"}]]
```

2026-07-26 15:02:04,485 INFO 127.0.0.1 - - [26/Jul/2026 15:02:04] "GET /therapists HTTP/1.1" 200 -
2026-07-26 15:02:04,739 INFO 127.0.0.1 - - [26/Jul/2026 15:02:04] "GET /static/css/therapists.css HTTP/1.1" 200 -
2026-07-26 15:02:04,755 INFO 127.0.0.1 - - [26/Jul/2026 15:02:04] "GET /static/js/script.js HTTP/1.1" 200 -
2026-07-26 15:02:04,781 INFO 127.0.0.1 - - [26/Jul/2026 15:02:04] "GET /static/css/style.css HTTP/1.1" 304 -
2026-07-26 15:02:04,784 INFO 127.0.0.1 - - [26/Jul/2026 15:02:04] "GET /static/images/headphones.png HTTP/1.1" 304 -
2026-07-26 15:02:04,786 INFO 127.0.0.1 - - [26/Jul/2026 15:02:04] "GET /static/images/Logo.png HTTP/1.1" 304 -
2026-07-26 15:02:04,785 INFO 127.0.0.1 - - [26/Jul/2026 15:02:04] "GET /static/images/logo_footer.png HTTP/1.1" 304 -
2026-07-26 15:02:04,911 INFO 127.0.0.1 - - [26/Jul/2026 15:02:04] "GET /static/images/1.png HTTP/1.1" 200 -
2026-07-26 15:02:05,550 INFO 127.0.0.1 - - [26/Jul/2026 15:02:05] "GET /sonic_sounds HTTP/1.1" 200 -
2026-07-26 15:02:05,699 INFO 127.0.0.1 - - [26/Jul/2026 15:02:05] "GET /static/css/style.css HTTP/1.1" 304 -
2026-07-26 15:02:05,705 INFO 127.0.0.1 - - [26/Jul/2026 15:02:05] "GET /static/css/therapists.css HTTP/1.1" 304 -
2026-07-26 15:02:05,706 INFO 127.0.0.1 - - [26/Jul/2026 15:02:05] "GET /static/images/Logo_footer.png HTTP/1.1" 304 -
2026-07-26 15:02:05,707 INFO 127.0.0.1 - - [26/Jul/2026 15:02:05] "GET /static/images/Logo.png HTTP/1.1" 304 -
2026-07-26 15:02:05,775 INFO 127.0.0.1 - - [26/Jul/2026 15:02:05] "GET /static/images/headphones.png HTTP/1.1" 304 -
2026-07-26 15:02:06,470 INFO 127.0.0.1 - - [26/Jul/2026 15:02:06] "GET /static/images/just_logo.png HTTP/1.1" 200 -
2026-07-26 15:02:38,125 INFO 127.0.0.1 - - [26/Jul/2026 15:02:38] "GET /sonic_sounds HTTP/1.1" 200 -
2026-07-26 15:02:38,256 INFO 127.0.0.1 - - [26/Jul/2026 15:02:38] "GET /static/css/style.css HTTP/1.1" 304 -
2026-07-26 15:02:38,264 INFO 127.0.0.1 - - [26/Jul/2026 15:02:38] "GET /static/css/therapists.css HTTP/1.1" 304 -
2026-07-26 15:02:38,267 INFO 127.0.0.1 - - [26/Jul/2026 15:02:38] "GET /static/images/Logo.png HTTP/1.1" 304 -
2026-07-26 15:02:38,269 INFO 127.0.0.1 - - [26/Jul/2026 15:02:38] "GET /static/images/logo_footer.png HTTP/1.1" 304 -
2026-07-26 15:02:38,271 INFO 127.0.0.1 - - [26/Jul/2026 15:02:38] "GET /static/images/headphones.png HTTP/1.1" 304 -
2026-07-26 15:02:38,321 INFO 127.0.0.1 - - [26/Jul/2026 15:02:38] "GET /static/images/1.png HTTP/1.1" 304 -
2026-07-26 15:02:38,886 INFO 127.0.0.1 - - [26/Jul/2026 15:02:38] "GET /static/images/just_logo.png HTTP/1.1" 304 -

SonicSerenity: Concurrency & Cloud Data Tiering

Task Parallelization

Python multiprocessing — **Process A** streams MJPEG to AWS Rekognition; **Process B** runs Librosa + SciPy for Solfeggio frequency analysis without blocking the main web thread.

Bandwidth Protection

Firebase Storage Signed URLs hand media delivery directly to Google's global CDN — preventing EC2 network exhaustion from large MP3 proxying.

Data Tiering

Firebase Firestore (NoSQL document store) captures unpredictable time-series emotion frames without requiring rigid schema migrations.

```
graph LR; A[Browser Client] --> B[NGINX Proxy]; B --> C[Flask Monolith]; C --> D[Firebase Data Tier];
```

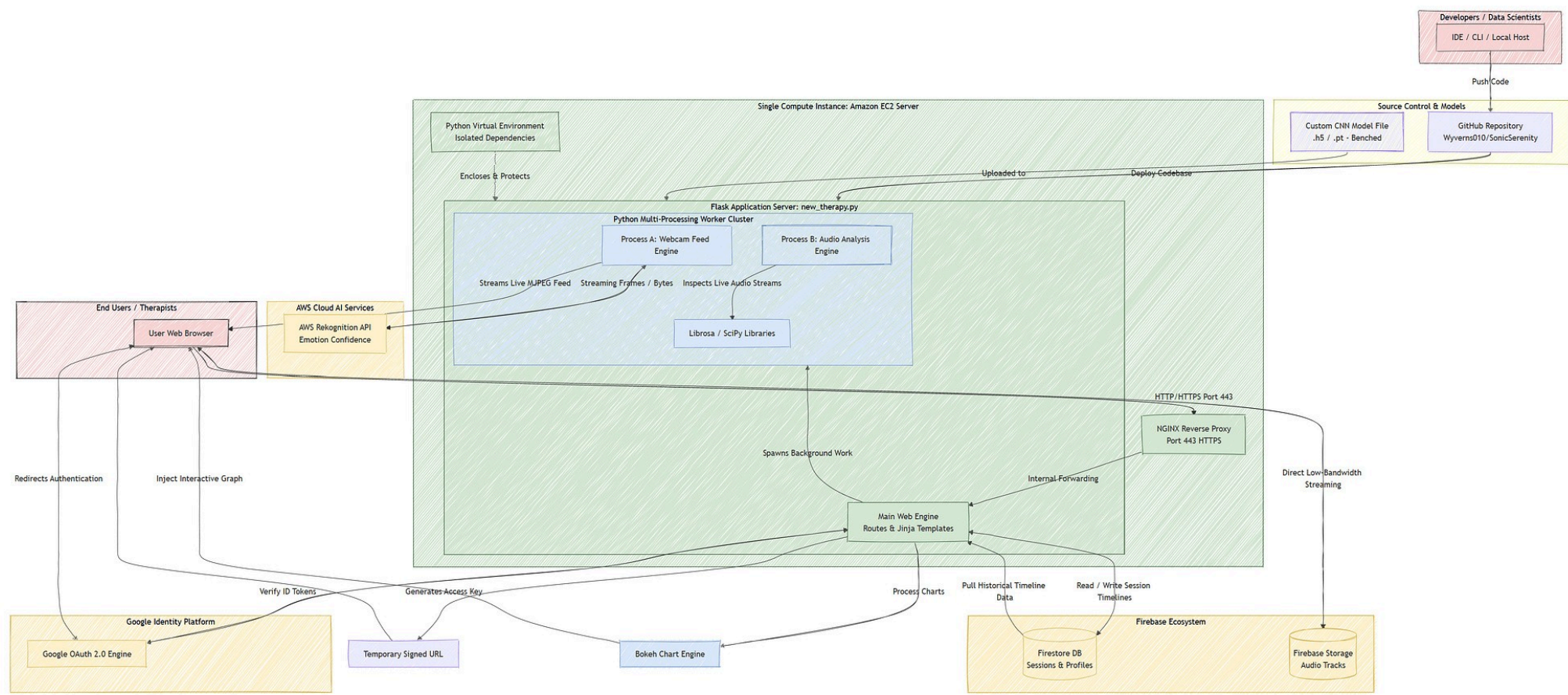
Browser Client

NGINX Proxy

Flask Monolith

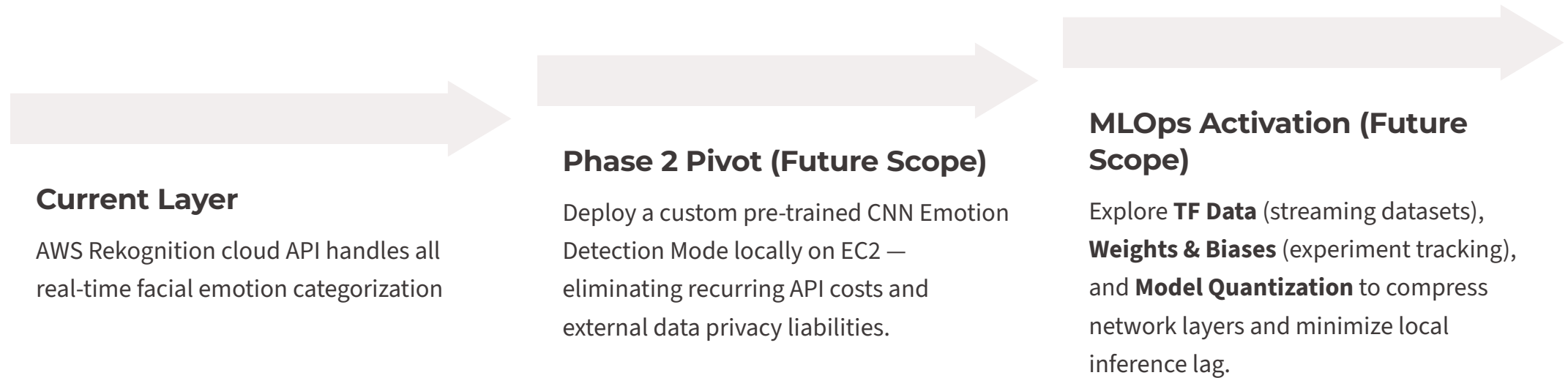
Firebase Data Tier

SonicSerenity Architecture Flowchart



SonicSerenity Forward Roadmap: Moving AI In-House

Future Scope (Implementing pending knowledge from Productionizing AI course)



Project Blueprint 2 — SpeechSmith

AI Communication Coach

Problem

Indian job market - highly competitive, heavily values English proficiency. Need to improve the English communication skills of India's youth to enhance their employment opportunities.

Solution

AI-powered communication coach delivering instant, personalised speech feedback on multiple linguistic dimensions — specifically adapted for first-generation English speakers and underserved communities as a scalable virtual public-speaking mentor. Exercises are adapted to unique use-cases: vocational centers (aviation institutions - customer interactions, paramedical institutes, interview-preparation), government schools

Speech Analysis

Real-time transcription pipeline; analyses pace, filler words, vocabulary richness, and clarity scores. Provides improvement recommendations and tracks progress history.

REACT 18

TYPESCRIPT

VITE

SHADCN/UI

SUPABASE POSTGRES

DENO EDGE FUNCTIONS

OPENAI WHISPER

GOOGLE GEMINI

ROW LEVEL SECURITY

Filler Word

00:12

Issues:

"um"

Solution:

Replace with a brief pause

Speech Feedback

80/100

Pronunciation Clarity

90/100

Clear and accurate pronunciation throughout

Speaking Pace

80/100

Maintained good speaking rhythm

Emotional Delivery

80/100

Good expression and engagement

Filler Words

70/100

Minimal filler words present

AI Analysis Complete

Analysis based on your actual recorded speech

Feedback generated using advanced AI speech analysis

Practice regularly to see improvement over time

Pronunciation Assessment

Overall Score

80/100

Issues Found

1

Pronunciation & filler words

Category Analysis

Pronunciation Clarity

90/100

Speaking Pace

80/100

Emotional Delivery

80/100

Filler Words Analysis

1 detected

Eliminate these words for more confident and professional delivery

"um" x1

Tip: Replace filler words with brief pauses or remove them entirely for clearer, more confident speech.

AI-Generated Recommendations

Pronunciation Clarity:

Clear and accurate pronunciation throughout

Speaking Pace:

Maintained good speaking rhythm

Emotional Delivery:

Good expression and engagement

Filler Words:

Minimal filler words present

Mood & Tone Analysis

Confident

Primary detected emotion

Confidence Level

80%

Energy Level

90%

Tone Characteristics

neutral tone

professional audience

Confident delivery

Audience Impact Analysis

Emotional Delivery:

Strong emotional connection with clear confidence

Audience Suitability:

Well-suited for professional presentations and formal settings

Energy Assessment:

High energy level that keeps audience engaged

AI Recommendations

Good expression and engagement

Filler Words Analysis

1

Total Filler Words

1

Unique Types

5/5

Fluency Score

Filler Word Cloud

um

Filler Word Breakdown

"um"

Low frequency

1 time

Improvement Suggestions

Practice your speech to reduce hesitation and filler word dependency

Practice Hub

Improve your speaking skills with our comprehensive collection of practice exercises, from basic storytelling to advanced job interview preparation.

All Exercises

Beginner

Intermediate

Advanced

Public Speaking

Intermediate

Build confidence in formal presentations and speeches

Duration: 10-15 min

Start Practice

Storytelling

Beginner

Master the art of engaging narrative delivery

Duration: 5-10 min

Start Practice

Debate Topics

Advanced

Practice persuasive speaking and argumentation

Duration: 15-20 min

Start Practice

Descriptive Narration

Beginner

Enhance your descriptive speaking abilities

Duration: 5-10 min

Start Practice

Creative Storytelling

Intermediate

Unleash creativity in impromptu storytelling

Duration: 10-15 min

Start Practice

Job Interviews

All Levels

Practice common interview scenarios and questions

Duration: 15-30 min

Start Practice

Track Your Progress

Every practice session is automatically analyzed and saved to your profile. Monitor your improvement over time and identify areas for focused practice.

Progress Tracking

Targeted Feedback

Achievement Goals

SpeechSmith's Serverless Paradigm

1 Frontend Stack

React 18 + TypeScript + Vite + shadcn/ui — Vite bundles optimized static assets that execute entirely client-side, shifting rendering load off the server entirely.

2 Compute

100% Serverless via Supabase Deno Edge Functions — dormant until triggered, waking in milliseconds; **pay-per-execution** vs. flat EC2 hourly rate.

3 Database + RLS

Supabase Postgres enforces relational structure; Row Level Security rules baked directly into database tables ensure User A can never access User B's rows — even if application code has a bug.

Comparative Production Engineering: SonicSerenity vs. SpeechSmith

Infrastructure Comparison

Engineering Vector	SonicSerenity — Monolith	SpeechSmith — Serverless
Compute Execution	Continuous (EC2 always awake)	On-Demand (Deno Edge Functions)
Database Blueprint	Unstructured NoSQL (Firestore)	Structured Relational (Postgres)
Security Enforcement	Application Code (Flask sessions)	Database Core (SQL RLS policies)
Network & SSL Layer	Manual NGINX Proxy Config	Fully Managed (Lovable/Supabase)
Isolation Mechanics	Python venv	Container-isolated Cloud Sandboxes
System Scalability	Manual (Auto Scaling Groups)	Native Cloud Elasticity
Fault Isolation	High Risk (crash freezes thread)	Absolute (edge crash is isolated)
Cost Basis	Flat Hourly Rate	Pay-Per-Millisecond

SpeechSmith: Fault-Tolerant AI & Storage Pipelines

Resilient Fallback Chain

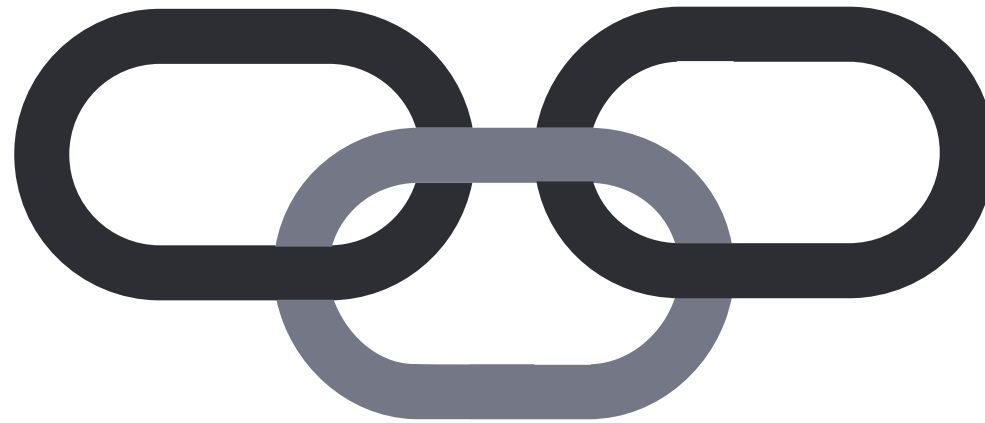
The `analyze-speech` edge function tries **OpenAI Whisper** → falls back to **Google Gemini** → falls back to a **canned demo string** — the user experiences zero interface degradation regardless of upstream API failure.

In-Browser Edge Processing

Audio captured as a binary Blob, converted to Base64, and visualized via Recharts + react-wordcloud entirely client-side — preserving serverless footprint and saving database compute.

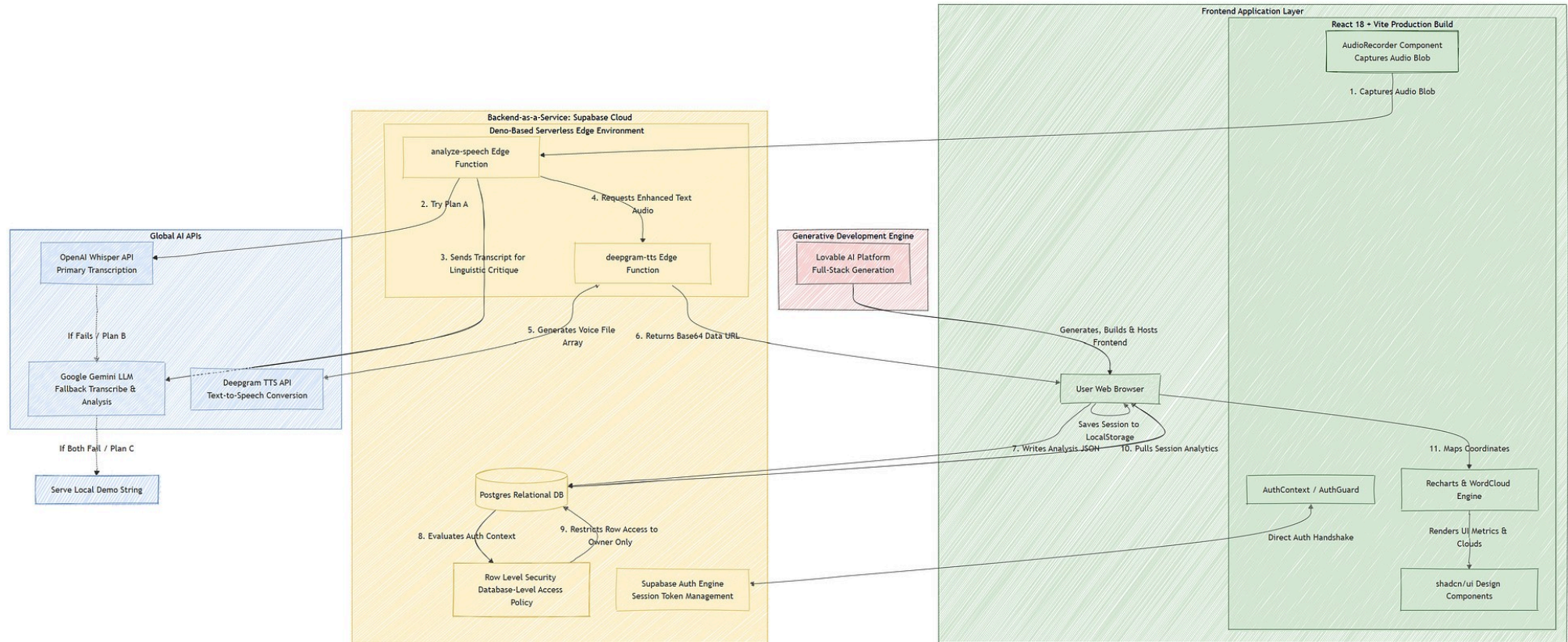
Primary: Whisper

Fallback: Demo



Fallback: Gemini

SpeechSmith Architecture Flowchart



System Awards & Academic Impact - SonicSerenity

	CREST Gold Award For strong project documentation and innovation
	JEI Publication Peer-reviewed paper on DFT and signal processing
	Springer FICTA Best Paper Conference award — GenAI in Music Therapy
	Fortis Hospitals MicroGrant Secured for active deployment
	Blue Ocean Top 100 Global innovation cohort selection
	Presented at IIT Mumbai National academic stage showcase
	23+ Psychologist Validations Endorsement across practitioners

Platform Impact at Scale - SpeechSmith

45K+

Users Reached

Users impacted across India and globally

18+

Institutions

Vocational centres and schools