

Tech Stack for Relief AI Inc. (Naia® Architecture)

The following architecture outlines the core components of the Naia® tech stack, an AI-powered mental health solution. This stack integrates advanced AI models, secure communication protocols, and cutting-edge backend technologies to provide an efficient and secure user experience.

Introduction

The Naia® architecture is designed for high performance and security. It is built on a scalable microservice architecture, supported by advanced technologies for AI models, data privacy, and interactivity. The stack's components ensure a reliable, responsive application that interacts in real-time with speech and video, while adhering to strict security standards.

Technical Description of Architecture Components

1. **React for Application Frontend**

React is used to build a dynamic and interactive frontend that powers the user interface of Naia®. React's component-based structure allows for a reactive, performant, and scalable user experience.

2. **NodeJS Backend**

The backend of Naia® is built with Node.js, enabling efficient request processing and quick handling of real-time operations. Node.js is well-suited for scalable applications that require high throughput and low latency.

3. **AI Agent & Workflow Orchestration**

This component ensures that Naia®'s AI agents follow the correct logic and workflow for all requests and interactions. The orchestration manages complex processes and AI interactions that must occur in real-time.

4. **Vector DBs with Embeddings for RAG**

For efficient storage and retrieval of data, Naia® uses a Vector Database with Embeddings. This is particularly effective for Retrieval Augmented Generation (RAG), enabling fast extraction of relevant information to power AI interactions.

5. **PostgreSQL with REST API**

PostgreSQL serves as the relational database for storing persistent data such as user information, interaction logs, and other application data. The connection to the database is made via a REST API, which facilitates structured communication between the frontend and backend.

6. **JWT-based Authentication**

User authentication is managed through JSON Web Tokens (JWT), ensuring that only authorized users can access specific features and data. JWTs provide secure transmission and validation of identities.

7. **Role-Based Access Control (RBAC)**

Naia® uses Role-Based Access Control (RBAC) to manage users with different permissions. Each role has specific rights that restrict access to data and features within the application.

8. **DAC for Tenants and Users**

Discretionary Access Control (DAC) allows tenants and users to manage access to their own data. This

provides flexibility and customization in permission management.

9. **Hardware Encryption for AI Inference with OODA AI and TEE**

All AI inferences in Naia® are secured by hardware-based encryption, working in conjunction with OODA AI and Trusted Execution Environments (TEE). This security architecture protects sensitive user data during processing and ensures high levels of privacy.

10. **Blockchain-based Proof-of-Privacy**

The blockchain-based Proof-of-Privacy solution ensures that all data processing operations remain transparent and verifiable. Through the blockchain, users can view the privacy status of their data in real-time. More details are available on the [OODA AI Explorer](#).

11. **Real-time Speech Text-to-Speech (TTS) AI Models**

Naia® utilizes advanced Text-to-Speech (TTS) models, enabling the AI to speak in real-time and engage in human-like conversations. These models are essential for the interactive audio communication of Naia®.

12. **Real-time Video Text-to-Video (TTV) AI Models**

In addition to speech synthesis, Naia® uses Text-to-Video (TTV) models, allowing the platform to generate real-time visual responses. This enhances user experience by mimicking human interactions and delivering visual feedback to user inputs.

13. **AI LLMs with Llama 3, 4, and Gemma Models**

For generative text processing and intelligent conversation management, Naia® employs powerful Language Models (LLMs) such as Llama 3, Llama 4, and Gemma. These models enable Naia® to generate context-aware, complex, and empathetic responses, elevating the quality of interactions.