**🛡️ PASO 0 — ENTENDIMIENTO: ARQUITECTURA DE JARVIS WEB (CORTEX)**

**📈 Visión General**

"Jarvis Web" es una solución de **asistente conversacional local** que integra entrada de voz 🎤, procesamiento de lenguaje 🧠 y respuesta hablada 🔊, todo corriendo en **infraestructura propia** (sin depender de la nube).

**Componentes Principales:**

* **Frontend** (React): Interfaz web moderna tipo chat + micrófono.
* **Backend** (FastAPI): API que conecta audio, texto y Cortex para procesamiento.
* **Servidor de Modelos** (Cortex.so): Motor de lenguaje local (ej. LLaMA 3.2, Mistral, Phi-4).
* **Infraestructura**:
  + **PostgreSQL**: Usuarios y sesiones (opcional si se desea).
  + **Redis**: Gestión de contexto rápido (opcional, para sesiones de conversación).
  + **NGINX**: Reverse proxy + HTTPS (TLS automático con Let's Encrypt).
* **Docker Compose**: Orquesta todos los servicios fácilmente.

**🧠 Flujo de Trabajo Resumido**

scss

CopiarEditar

[Usuario habla 🎤]

↓

[Frontend (React + JS)]

↓ (Audio)

[Backend FastAPI]

↓ (Transcripción)

[Whisper / otro modelo]

↓ (Texto)

[Cortex.so - LLM local]

↓ (Respuesta de texto)

[Backend FastAPI]

↓ (Genera Audio)

[edge-tts]

↓ (Audio de respuesta)

[Frontend]

↓

[Usuario escucha 🔊]

**📦 Capas de la solución**

| **Capa** | **Tecnología** | **Rol Principal** |
| --- | --- | --- |
| **Frontend** | React + JS | Chat, micrófono, visualización de conversación |
| **Backend API** | FastAPI | Control de flujo, conexión Cortex, Whisper, edge-tts |
| **Motor IA** | Cortex.so | Generación de respuestas de lenguaje |
| **Infraestructura** | Docker, NGINX, PostgreSQL, Redis | Seguridad, almacenamiento, comunicación |
| **Despliegue** | Docker Compose | Levanta todo el ecosistema automáticamente |

**🧩 Esquema Visual Rápido:**

css

CopiarEditar

[ Navegador Web (Usuario) ]

↓

[ Frontend (React) ]

↓ API REST

[ Backend (FastAPI) ]

↙ ↘

[ Whisper (Voz->Texto)] [Cortex.so (Texto Inteligente)]

↘ ↙

[ edge-tts (Texto->Voz) ]

↓

[ Reproduce audio en navegador ]

**🎯 Objetivo de este Proyecto**

* 100% ejecución en infraestructura local 🏠
* No requiere servicios externos (salvo certificados SSL si quieres HTTPS)
* Modular, escalable, seguro
* Rápido y de baja latencia al no depender de OpenAI ni APIs remotas

**✅ Confirmación antes de avanzar**

Comandante, **¿está claro el PASO 0: ENTENDIMIENTO de la arquitectura de Jarvis Web (Cortex)?**

**Responde:**

* "✅ Entendimiento confirmado" para avanzar a **PASO 1.2**
* "🛠️ Quiero ajustar algo" si deseas modificar arquitectura, flujos o componentes