Rex AI Assistant – System Blueprint (v1.0)

(Conceptual Overview and Technical Architecture)

Rex is a local-first, voice-driven AI assistant that integrates natural conversation, reasoning, and smart-home control while maintaining privacy and modularity.

Core Architecture:
1. Wake Word Detection: ESP32 + INMP441 microphone (custom “Hey Rex” model)
2. Speech-to-Text: Faster-Whisper (GPU accelerated)
3. Reasoning Core: Dynamic model switching (MythoMax 13B, LLaMA 3 8B, Mistral 7B)
4. Memory System: Short-term context buffer and long-term JSON storage per user
5. Action Layer: Integrations with Home Assistant, IFTTT, Plex, and n8n automations
6. Text-to-Speech: Coqui XTTS v2 with custom voices (Jensen, Majel)
7. Output Layer: Local speaker playback, browser UI (askrex.app), and smart displays

Core Principles:
• Privacy-first local processing
• Modular, extensible design
• Contextual awareness with memory
• Natural conversational timing and voice
• Resilient offline operation

Communication Protocols: Flask API, WebSockets, MQTT, JSON schemas
Security: HTTPS via Cloudflare Tunnel, encrypted data at rest, environment variable secrets
Future Modules: Emotion-aware voice, vision input, task scheduling, mobile companion app

Blueprint Paths:
C:\AI\Assistant\ - Core code and models
C:\AI\Assistant\Memory\ - User memory data
C:\AI\StartupScripts\ - Autostart Rex
C:\AI\Assistant\ha_entity_server.py - Smart home API bridge

Rex is designed as a neural city — each district handling sensory or cognitive tasks, all connected by shared infrastructure that adapts and evolves.