

PROJECT BRIEF



PREPARED

July 2026

AUTHOR

Jordan Bee

Kaleo — Self-Hosted AI Assistant on WhatsApp

A personal, always-on AI assistant running entirely on my own servers that I message on WhatsApp (and Discord), which reads my email, calendar, and reminders, runs real tools and agentic tasks from a chat line, and proactively keeps me updated.

SIGNATURE CAPABILITY

I text it on WhatsApp like a person and it acts as my chief of staff — reading my email, calendar, and reminders to keep me updated and running multi-step agentic jobs, all on infrastructure I own.

OVERVIEW

I message it on WhatsApp exactly like texting a friend, in plain language or a voice note, and it acts: it reads my email, calendar, and reminders to brief me and chase what matters, runs skills, hits tools and APIs, executes multi-step agentic jobs, then replies in the same thread. It is multi-channel, answering on WhatsApp and Discord off the same brain, with persistent memory so it carries context between sessions. Voice notes are transcribed and routed to the model, with a daily spend cap as a guard. Everything runs on my own infrastructure, not a cloud SaaS, so I control the data, the numbers, and the model. A hard token gate and a file-based kill switch mean only authenticated commands ever execute and I can stop it sending within seconds.

TECHNOLOGY

Go / whatsmeow + Baileys WhatsApp bridge (REST on :8081), Python daemon, OpenClaw (Node/TypeScript) gateway, MiniMax-M2 via a self-hosted Pulse LLM router with fallbacks, Gmail / Google Calendar / Reminders access, systemd user services with linger, Docker Compose, Hostinger + Mac + Tailscale mesh, WhatsApp + Discord channels, SQLite session store.

HIGHLIGHTS

- ◆ **Reads my email, calendar, and reminders** to proactively brief me, surface what needs a reply, and keep me on schedule.
- ◆ Bridge architecture: a Go whatsapp/Baileys bridge holds the live WhatsApp Web session and exposes a local REST API; a Python daemon polls it, transcribes voice, routes to the model, and sends the reply.
- ◆ **Fully self-hosted** on my own boxes (Hostinger VPS + Mac/Linux server over Tailscale), with no third-party assistant cloud in the path.
- ◆ **Multi-number, multi-channel**: a Singapore number wired to the Hostinger instance and a US WhatsApp Business number to the Mac, plus a Discord bot off the same stack.
- ◆ **Survives reboots**: runs as a systemd user service with linger enabled, and the WhatsApp session migrates between hosts without re-scanning the QR.
- ◆ Safety rails: a **fail-closed token gate**, a touch-file **kill switch that halts sending within seconds**, and a daily transcription budget cap.

STATUS

Live and in daily use: the WhatsApp bridge runs as a persistent systemd service on the mini-server with the Discord bot active on Hostinger.



Jordan Bee
jordan.bee2012@gmail.com
+44 7532 722082