

Yu Fong C.

Production AI Integration Engineer

Taipei, Taiwan · Remote (overlaps US / EU) · ro159147@gmail.com · yufongc.netlify.app · github.com/alpha-0619

I make AI work *in production*, not just in the demo. I take voice agents and LLM apps that fall apart once real users hit them and make them reliable: guardrails so they stop inventing answers, memory that holds across a conversation, tool and API error handling that fails safe, and escalation to a human when it should. The systems I ship handle real customers, real transactions, real money.

SELECTED WORK

High-volume customer messaging — built for a client

Production Python system I built for a business handling a heavy inbound load: intent classification (real inquiry vs. noise vs. needs-human), context-aware replies that stay in the brand's voice, a long-term memory layer (vector store + structured facts) so it remembers what each customer was told, anti-hallucination dedup, anti-loop defenses, and confidence-gated human escalation. **Handles hundreds of inbound messages a day; one operator now covers what would take a full support team.** *Python, multi-provider LLM routing, mem0 + Chroma.*

Voice assistant for air travelers — live in production

Voice agent that pulls live flight data over an MCP tool connection, **runs in English and Turkish**, holds context across a whole call, and has guardrails so it never invents a flight it doesn't have. Delivered and running in production. *Python, FastAPI, real-time voice, MCP.*

Telegram → helpdesk pipeline — Spanish-language marketplace

End-to-end customer-service integration: a Spanish Telegram bot intercepts customer DMs, routes them into a unified Crisp helpdesk via webhooks, with **two-way sync** so team replies post back to Telegram with thread fidelity. Demo-vs-production token isolation, bilingual handover docs. Delivered, paid, closed. *Python, python-telegram-bot, Crisp API, FastAPI, Docker.*

On-chain P2P escrow protocol — live on BSC Mainnet

Designed and shipped a trust-minimized escrow smart contract. **Solidity (~2,000 LOC), 68/68 tests passing** (reentrancy, ownership transfer, dispute paths), Next.js + wallet frontend, on-chain event monitoring with alerts. **Independent security audit: 8 findings, all resolved before mainnet. Live, securing real value, zero incidents.** *Solidity, Hardhat, Next.js.*

OPEN SOURCE

addr-screen github.com/alpha-0619/addr-screen

Multi-source Tron (TRC-20) address risk screening (OFAC + GoPlus + TRONSCAN). Cross-checks three independent sources, runs the sanctions list offline, no vendor lock-in, no API key for the default run. *Python, MIT.*

SKILLS

Languages / frameworks: Python, FastAPI, TypeScript, Next.js, Solidity

LLM / AI: Claude, OpenAI, Qwen, DeepSeek · RAG · guardrails & evals · MCP servers · prompt engineering · multi-agent

Voice: Vapi, Retell, Pipecat, Twilio · neural TTS

Integration: Telegram, WhatsApp, Crisp, Intercom · webhooks · CRM · Docker, PostgreSQL

Available for reliability audits, production hardening, and ongoing maintenance of AI / voice systems — and custom builds, built to survive production from day one.