Mohammad Ausaf

AI Infrastructure and Agent Systems Engineer | Bengaluru, India
ausaf9922@gmail.com | +91-7007388740 | linkedin.com/in/mohammad-ausaf | github.com/AusafMo | ausafmo.com

Experience

Galleri5 (Collective Artists Network) — Bengaluru, India
Manager - Software Engineering — *May 2026 – Present*
Software Engineer (AI) — *Dec 2023 – May 2026*

Hands-on technical lead: co-own backend architecture and code review across AI Studio; led a team of up to 3 engineers; partner on product direction. G5 AI Studio is the platform behind *Mahabharat: Ek Dharmayudh* (JioHotstar) and *Chiranjeevi Hanuman: The Eternal* (24M+ first-look views), built on Azure in partnership with Microsoft.

Agent systems

- **Creative agent (MIRO).** Built AI Studio's creative agent to plan, generate and edit image, video and audio work from a brief across 50+ tools (300K+ tool calls/month). It builds interactive apps inside the chat (review boards, production trackers), checks what it rendered before answering, tracks approvals so editing an approved step flags everything built on it, and routes requests to the right playbook.
- **Long-running turns.** Designed turns that survive real use: people redirect a running turn without stopping it, through a Mongo-backed inbox any replica can read; the agent resumes work on its own but never authorises spend; an overflowing turn compacts instead of stalling; a failed approved run wakes the agent. Turn completion went from about 95% to 99% (Oct to Nov).
- **Evals and observability.** Every turn is traced end to end, including the reason for every failed tool call, with LLM judges labelling outcome and failure cause (OpenTelemetry, prompt versions); designed so tracing can never break a turn. Halved content-safety false positives (about 8% to under 4%, catching 98%+ of unsafe cases) on a 600+ case eval suite that gates every push.
- **Sandboxed code execution.** Built the sandbox service on Daytona from scratch: 80K+ runs a month, a fresh isolated sandbox per request, egress blocked and failing closed, SSRF checks, and prompt-injection-resistant abuse screening that passed an internal red-team suite. Prototyped a lighter runtime (unpacked rootfs under bubblewrap, minimal capabilities) instead of a Docker restore, cutting cold start from 37–48 s to 17 s.

AI infrastructure

- **Inference gateway.** Built the request-routing layer every AI Studio job takes, from an empty repo: 190K+ jobs/month, 990K+ since launch, to 170+ model endpoints (fal, Replicate, Google, self-hosted H100s, others).
- **Fair multi-tenant scheduling.** One tenant's burst or one provider's rate limit shouldn't starve everyone else. Implemented deficit weighted round robin in atomic Redis Lua plus rate limiting, concurrency caps at four levels, and TTL slot leases so a crashed worker gives its capacity back on its own; resolved 20+ race conditions.
- **Crash-safe intake.** Durable execution, so a restart never drops an accepted job: write-ahead log, idempotent submits, leader-elected reconcilers. It grew out of a GPU dispatcher for H100 servers that lost 0 of 10,000 jobs under load test.
- **Billing and failure policy.** Usage metering that matches real cost: atomic credit deduction, settlement on actual output and compute time, refunds. Core rule: never re-dispatch a job that may still be running, to avoid paying twice for one expensive generation; new failure rules run in shadow before they enforce.
- **Reliability and velocity.** Fixed platform-wide 8–25 s stalls; made deploys drain in-flight agent turns instead of killing them; analytics rollup over 1.3M+ docs, 135 s to 1.5 s; cut deploy builds from 6.5–10.5 min to 2 min. Built the control plane ops uses to drain or cap failing providers. Owned design docs, incident RCAs and on-call.

Earlier: bulk episode and shot generation pipeline with QC for *Mahabharat*; video embeddings and ingest hardening for the Azure asset platform, working with Microsoft's engineering team (ISE); CLIP semantic search over millions of SKUs for Myntra, Ajio, H&M (Pinecone).

Projects

- **cfgit** (open source, PyPI). Version control for live database config in MongoDB and PostgreSQL: history, diff, rollback, blast-radius analysis; refuses to overwrite out-of-band edits; ships an MCP server so agents can change config safely. github.com/AusafMo/cfgit
- **Searchy** (open source, macOS). On-device photo search: SigLIP/CLIP and OCR results merged with weighted rank fusion, face clustering, Metal acceleration; Homebrew. ausafmo.com/searchy

Skills

- **Agents and LLMs:** agent runtimes, tool calling, MCP, Hermes Agent, LLM-as-judge evals, LangWatch, OpenTelemetry, Claude, Gemini, OpenAI, CLIP/SigLIP, PyTorch
- **Infrastructure:** distributed systems, Linux, Python, FastAPI, asyncio, Redis (Lua), MongoDB, PostgreSQL, Firestore, Daytona, Docker, Azure, GCP, AWS, GitHub Actions; also SQL, Swift, C++

Education

KIET Group of Institutions, Ghaziabad, India — 2024
Bachelor of Technology, Computer Science