# SOMNATH GHOSH

**Engineering Manager · Distributed Systems · Applied AI & Agentic Platforms**

Bangalore, India · +91 9232661324 · somnath.ghosh2010@gmail.com

linkedin.com/in/somnathghosh007 · github.com/somnath-habbit

## SUMMARY

Engineering Manager with 10+ years building production backend platforms and **7 years in direct line management** — 15 engineers today, 31 at peak. I lead modernization at scale: a message-broker re-architecture with zero data loss, critical incidents down 83%, 64 Lambda runtime upgrades at zero downtime, 34 projects delivered in 2025.

I also still build. Independently I architect and operate agentic AI systems in production: a 34-stage pipeline on AWS Step Functions with 12 LLM providers behind capability-split fallback chains, a four-gate eval harness for output that has no single correct answer, MCP servers consumed by other agents, a multi-agent SDLC orchestrator whose trust boundaries are architectural rather than prompt-based, and a zero-knowledge vault that brokers scoped, revocable credentials to automated callers. I treat the model as an untrusted principal and design the blast radius accordingly.

## CORE SKILLS

**Applied AI & LLM Systems** — Multi-provider routing with capability-split fallback chains and circuit breaking · Agent architecture and autonomous loops · MCP server development · RAG with explicit context budgeting · Eval harness design for non-deterministic output · Vision-model QA · LLM-as-judge with anti-sycophancy controls · Prompt-injection and data-leakage guardrails · Secret management for machine principals · Token and cost economics · Grounding against hallucination · Local inference

**Engineering Leadership** — Team building to 31 · Hiring and interview design · Performance management and PIPs · Mentoring (30+ advisees, 6+ appraisal cycles) · Technical roadmapping · Incident and on-call process · SAFe 6.0 · AI-assisted development standards

**Architecture & Cloud** — Microservices · Event-driven and serverless · Step Functions orchestration · Message queues (SQS, RabbitMQ, Celery) · API and data modeling · AWS (Lambda, Step Functions, ECS/Fargate, S3, DynamoDB, Aurora, SQS/SNS, CloudFormation, SAM, CloudFront, Cognito, WAF, VPC) · GCP (Pub/Sub, GKE) · Docker · Terraform · systemd · nginx

**Backend, Data & Operations** — Python (Django, Flask, FastAPI, Celery, Spark) · Node.js · TypeScript · MySQL, PostgreSQL · MongoDB, DynamoDB · Redis · ChromaDB · DataDog · Alerting design and runbooks · FinOps and per-call cost telemetry

## PROFESSIONAL EXPERIENCE

### Clarivate Analytics — Associate Manager, Software Engineering

**Apr 2023 – Present · Bangalore**

Own the Alexander Street Back-Office platform (under ProQuest): 8+ applications spanning content automation, metadata management, CDN operations, and video processing. Built the India team from 2 to 15 engineers. **34 projects delivered in 2025.**

- Migrated the content-automation build queue from RabbitMQ to SQS as a **parallel broker** with full Celery compatibility, dual-writing across 5–10 services — chosen over a big-bang cutover on a system already causing recurring production pain. **Zero data loss; zero queue-related incidents since.**
- Cut critical incidents **83%** (one per 15 days → one per quarter) with **zero customer-breaking incidents**. DataDog observability across three core systems dropped issue detection from ~3 weeks to **2–3 days**, moving most defects into the same sprint as the code that caused them.
- Delivered MySQL 5.x→8.x, Apache, and Aurora migrations plus **64 Lambda runtime upgrades** at 100% success and **zero unscheduled downtime**. Reclaimed **66 GB** via archival (163.3 → 97.6 GB, −60%).
- Built an end-to-end video chapterization pipeline integrating a third-party AI service into a Lambda architecture; live since Jul 2025 at 2–3 videos/day. Automated royalty and audio reporting, removing 2–3 developer tickets monthly.

- Hired **5 roles in 3 months**, designing the interview framework and evaluation criteria myself rather than delegating the design; separately won the business case for 2026 headcount.
- Held the quality bar through **a full year with no dedicated QA function** via mandatory peer review and developer cross-testing — the incident numbers above were achieved under that constraint, not despite it.
- Established primary/secondary service ownership, sustainable on-call with defined escalation, and runbooks; cross-trained the team off single-point-of-failure risk. Sprint velocity rose from 12–13 to **21–29 points**.
- Directed AI-assisted development across the team — code review, refactoring, documentation, incident analysis — with review standards for model-generated code.
- Ran a performance improvement process through proper channels; mentored an engineer through a full re-architecture to independent end-to-end ownership. Authored a 4-year CMS consolidation roadmap within 4 months of joining, and the 2026 technology roadmap with quarterly capacity-matched assignment.

### Trica (Let's Venture) — Engineering Manager · Jul 2022 – Mar 2023 · Bangalore

Managed **31 engineers** across 3 squads with 3 tech leads on a B2B equity-management platform (ESOP/SAR/RSU, cap table) serving 900+ enterprise customers. Designed containerized microservices on ECS/Fargate with multi-cluster MongoDB Atlas for HA; adopted micro-frontends to unblock parallel squad delivery. Sustained delivery, appraisals, and morale through a company-wide cost reduction.

### Hashedin by Deloitte — Senior Engineering Management Specialist · Jul 2019 – Jul 2022 · Bangalore

Led delivery teams of up to 25+ engineers; career advisor to 30+ engineers across 6+ appraisal cycles. Unified Pidilite's mobile and web field apps onto one backend with real-time Salesforce sync, leading a 25-person cross-functional team (*Certificate of Appreciation, Pidilite CTO*). Built Mahindra's agri-equipment lending platform on hybrid AWS/GCP with Pub/Sub, and a serverless healthcare vendor-management system on Lambda with fine-grained ACL. 3 Spot Awards.

### Healthplix Technologies — Senior Software Developer · Sep 2018 – Jul 2019 · Bangalore

Physician-facing EMR used at point of care across 12+ states. Built an **R/x recommendation engine on AWS SageMaker**, trained on prescription and diagnosis data captured through an OCR + NLP pipeline. Built Spark ETL from MySQL to MongoDB; restructured the application to a micro-kernel architecture.

### Nreach Online Services (Xoxoday) — Software Engineer → Senior Software Engineer · Sep 2015 – Sep 2018 · Bangalore

One of 4 founding engineers on a corporate rewards platform serving 1,000+ global clients; built from scratch in 5 months. Led evolution through monolith → MVC → HMVC → microservices; built 50+ modules including ACL, payments, and multi-tenant onboarding. Progressed to technical lead of 5.

## APPLIED AI SYSTEMS — BUILT AND OPERATED INDEPENDENTLY

Sole architect, engineer, and operator. ~130,000 LOC across 15+ services, 85 AWS Lambdas, ~3,100 test functions. AWS compute and storage for the entire estate runs at **$0.20–0.50/month** against a hard ₹2,000/month alarm.

**Autonomous Video Production Pipeline** — *production on AWS ap-south-1 · 47 services · 67,490 Python LOC · 1,296 tests · 274 architecture documents*

- **Orchestration.** 34-stage DAG on Step Functions: stateless orchestrator Lambda, 71 Lambdas, Fargate media workers, SQS+DLQ, 11 DynamoDB tables per environment. The state machine is **6 states** — the dependency graph lives in code, so adding a stage edits a dictionary instead of rewiring infrastructure. Dependency resolution is a 64-line pure function with zero I/O, making the most failure-prone component trivially testable.
- **Model routing.** 12 providers behind 6 capability-split chains, resolved per environment from a 650-line declarative registry. Cost and safety policy live in configuration, not convention: retry-heavy verify, classifier, and heal stages are pinned off paid chains — *if both free options are down, the stage fails or skips rather than silently billing* — and dev chains structurally exclude paid keys, so a dev run cannot bill. Five models were eliminated by real testing rather than reputation, including **an 11B vision model that beat its own 90B sibling at 1.3s vs 24s with identical accuracy**.

- **Evals.** Four copy-left gates catch defects at the cheapest point they surface — free JSON checks, per-asset images, per-page screenshots, then a 159-check final battery, tiered by complexity, with deterministic checks gating the paid vision model. Verdicts are structured and region-aware, returning a fix hint fed back into the regeneration prompt *so retries converge instead of re-rolling blindly*. Audio sync is verified by re-transcribing the final muxed audio into a drift curve — best measured **0.951 across 263 words**.
- **Agents and MCP.** Four MCP servers: pipeline coordinator, headless-browser visual QA, diagram rendering, avatar OAuth. A headless producer agent on a systemd timer runs a video end to end unsupervised — cost gates before spending, prerequisite-gated topic traversal, and **LLM-as-judge at three approval gates under an explicit anti-sycophancy rule** (*never blanket-approve to keep the run moving; a rubber-stamped gate is worse than a stalled one*). Authority is bounded by construction: it opens PRs but never merges, never publishes, never touches infrastructure, and runs against a Bash allowlist. On its first full cloud run it produced a 13.9-minute 1080p video, passed vision QA over 211 deduplicated frames, then **failed its own publish-readiness gate**, escalated, alerted, and published nothing. Correct escalation over a false pass is the design goal.
- **Cost.** ₹40–60/video measured all-in, reconciled against a design-time estimate that proved ~10× optimistic. Per-call telemetry to a daily log and each video's manifest; NAT Gateway deliberately excluded (~$384/yr) in favour of egress-only security groups.

**Maestro — Multi-Agent SDLC Orchestrator** — *~22,700 Python + 7,631 TS LOC · 837 tests · 43 endpoints*

A planning agent emits a task DAG; a deterministic orchestrator — not a model — owns sequencing and merging. **Six role-scoped agents run under least privilege enforced structurally, not requested in a prompt**: reviewer has no write tool, devops touches only infrastructure files, developer is confined to its task branch. Agents cannot commit to main — every task gets its own git worktree and the orchestrator owns the merge, so a confused agent structurally cannot damage the trunk. Turn limits bound every session; failed branches are preserved for forensics. Verification hooks (file existence, content match, exit code, HTTP check) act as a post-task eval layer triggering retry. Human approval is a first-class DAG node type; models are cost-tiered by role.

**Content Engine — RAG editorial pipeline** — *~11,747 LOC incl. 3,721 LOC of tests · 23 endpoints · FastAPI under systemd*

RAG over two ChromaDB collections with **explicit character-budgeted context assembly** per source and per chunk rather than concatenate-and-hope; the embedding provider was chosen on quota shape, not benchmark score — per-minute limits instead of a per-day quota already consumed by sibling services. Triage returns an 18-field structured verdict including `moat_risk` and `withhold`, making **information-disclosure control a model-enforced part of the output**, backed by a non-model regex validator with an auto-redaction table. The scanner may only touch explicitly registered repositories. Every agent task declares a timeout, token budget, and hard cost cap, with a watchdog enforcing timeout, 30s inactivity, and budget overrun through SIGTERM → grace → SIGKILL. **Its defining postmortem:** one AI CLI was hardcoded across five call sites; the subscription rotated and every call failed *silently* — 30 scans, 0 drafts, ~2 months undetected. The fix was a provider-abstraction layer that auto-detects the authenticated CLI, normalizes three JSONL event-stream formats, and preserves the original return contract field for field, so **not one call site changed**. Success-shaped output that is actually nothing is the failure class that makes AI systems hard to operate.

**Guptam — Zero-Knowledge Credential Vault** — *4,705 Python LOC · 105 tests · built and tested, pre-launch*

The server never holds plaintext or a decryption key. Keys derive from the master password via **PBKDF2 at 600,000 iterations** (the OWASP minimum, cited as such in source) under **AES-256** authenticated encryption, so full database compromise yields ciphertext and nothing else. Storage sits behind an abstract repository with encrypted SQLite and DynamoDB implementations, so the local build has no cloud dependency to audit. Built for the agent era: **service accounts distinct from human sessions**, API tokens hashed at rest with a short identifying prefix retained so a token can be recognized and revoked without ever being recoverable, last-used timestamps for detecting stale credentials, and middleware separating a browser session from a programmatic caller. An automated system requests the credential it needs at the moment it needs it against a scoped token, instead of every agent carrying a permanent copy of every secret in its environment.

**Cross-cutting practice** — A version-controlled agent-governance hierarchy: one canonical operating document every per-repo document resolves upward to, encoding a data-classification policy of what may never be published, locked architecture decisions, and hard cost ceilings. A standing token-economics rule forbids agents from reading images into their own context — visual QA is delegated to a vision model over MCP that returns text, keeping agent context for reasoning rather than pixels. A **live-run verification gate** separates "tests pass" from "verified": fixes stay quarantined until confirmed by a real production run, and a gaps register tracks built, trigger-proven, and demonstrated as three separate columns. Further MCP work includes a publishing server exposing 8 tools with learned style transfer, and a spreadsheet server exposing 7.

## EDUCATION, CERTIFICATION & AWARDS

**B.Tech, Computer Science & Engineering** — Calcutta Institute of Technology (WBUT), 2011–2015

**SAFe 6.0 Certified Scrum Master** — Scaled Agile, 2025

Certificate of Appreciation, Pidilite CTO · 3 Leadership Spot Awards, Hashedin by Deloitte · Shining Star Award, Xoxoday (2017)