YASH GUPTA

Senior AI Engineer | Tech Lead | Agentic Systems & LLM Applications

Amsterdam, NL • +31 685 713 834 • yashgpt2894@gmail.com

yashgupta.fyi • linkedin/yashgpt2894 • github/yashgpt2894

SUMMARY

Senior software engineer with 9+ years building production Python systems, now focused on shipping agentic AI products end-to-end. Model-agnostic across Anthropic Claude, Google Gemini (Vertex AI), OpenAI GPT, and AWS Bedrock-hosted models — picks the right model per task on cost, latency, and capability. Designs agentic architectures with the Claude Agent SDK, LangChain, and MCP, covering context engineering, tool use, multi-step orchestration, evaluation, guardrails, and cost optimization. Track record of taking AI features from prototype to production across 7 European markets and millions of users. Featured AI work: **Backoffice Agent** (autonomous Jira-to-PR agent) and **Backoffice Brain** (Claude plugin over a 254-service codebase), plus a GenAI collateral service live on Google Cloud — details below.

CORE COMPETENCIES

Models (model-agnostic): Anthropic Claude (Opus / Sonnet / Haiku), Google Gemini (Pro / Flash) on Vertex AI, OpenAI GPT-4 / GPT-4o, AWS Bedrock-hosted models (Llama, Mistral); chooses model per task on cost, latency, and capability

Agentic AI & LLMs: Claude Agent SDK, LangChain, MCP (Model Context Protocol), Anthropic API, OpenAI API, context engineering, tool use & function calling, multi-step orchestration, RAG, knowledge-graph retrieval, prompt engineering, LLM evaluation (offline/online metrics, gold sets, A/B), guardrails & prompt-injection defenses

LLMOps & Observability: Granular tracing, token/cost accounting, latency & throughput optimization, hard cost ceilings, human-in-the-loop review, regression tests, incident response

Languages & Backend: Python (expert), Perl, SQL, Shell; Django, Django REST Framework, FastAPI, Flask, Next.js, REST APIs, microservices, OOP, Pytest

Data & Infrastructure: Apache Airflow, Kafka, PostgreSQL, MySQL, BigQuery; Docker, Kubernetes, Terraform; AWS (Bedrock, EC2, S3, Lambda), GCP (Cloud Run, Vertex AI, Firestore, Pub/Sub, Cloud Build); Jenkins, Git, Grafana, Prometheus, Linux

Leadership: Tech lead across cross-functional teams, HLD/LLD ownership, engineering standards, mentoring, stakeholder alignment

EXPERIENCE

Technology Lead / Senior AI Engineer — Infosys Limited, Amsterdam *Nov 2021 – Present*

- Lead design and delivery of AI-powered product features across **7 European markets**, serving millions of users; own architecture and reliability end-to-end.
- Architected agentic systems that turn Jira tickets into reviewed Bitbucket PRs (see **Backoffice Agent** below), with human gates and hard cost ceilings enforced in code.
- Built knowledge-graph retrieval over a large enterprise codebase so AI assistants cite source by file:line and refuse to answer when context is missing (see **Backoffice Brain**).
- Operate Apache Airflow pipelines handling **300–500+ DAG runs/day** across batch and real-time AI workloads; productionized inference APIs for low-latency consumption.
- Shipped video intelligence features (e.g., skip intro/outro) across **millions of playback sessions**; designed scalable microservices on Docker and Kubernetes.
- Own observability with SLIs/SLOs, granular tracing, and automated validation (Grafana, Kibana, Prometheus); routinely tune latency, throughput, and token spend.

Technologies: *Python, Claude Agent SDK, LangChain, MCP, AWS Bedrock, FastAPI, Next.js, Apache Airflow, Kafka, Docker, Kubernetes, AWS/GCP, PostgreSQL*

Senior Systems Engineer — Infosys Limited, New Delhi *Jan 2018 – Oct 2021*

- Built and maintained backend systems and RESTful APIs for data-intensive applications in Python (Django) and Perl (Catalyst).
- Developed a traffic simulation framework for resilience and load testing under high network traffic.
- Modernized and modularized legacy code bases, reducing operating cost and improving long-term maintainability.
- Contributed to distributed system design, performance, and scalability initiatives across the full release cycle.

Technologies: *Python, Perl, Django, Catalyst, REST APIs, MySQL, Docker, Jenkins, Git*

Systems Engineer — Infosys Limited, Bengaluru *Jun 2016 – Jan 2018*

- Built automation scripts and data pipelines in Python and Shell, reducing manual workload and accelerating analytics workflows.
- Evaluated technical feasibility of new system designs and proposed performance optimizations adopted into production.

Technologies: *Python, Perl, Shell, Linux, MySQL*

SELECTED AI PROJECTS

Backoffice Agent — *Claude Agent SDK, AWS Bedrock, FastAPI, Next.js, Python*

- Built an autonomous Jira-ticket-to-PR agent: plans, implements in an isolated **git worktree**, verifies (tests / mypy / linters per repo), pushes a branch, opens the PR, and comments back on the ticket — inference on AWS Bedrock.
- 5-stage workflow with human gates at **Plan / Diff / Push** and hard per-run cost ceilings (**$1–25 budget, 100-turn cap**); shrinks a 2–4 hour ticket loop to ~10 min of review.

Backoffice Brain — *Claude plugin, knowledge tree, MCP, Python*

- Shipped a Claude plugin packaging a **~254-service enterprise back-office codebase** into 12 invokable skills (plan-ticket, find-service, implement, verify, prepare-pr, …) backed by a ~280-file knowledge tree with per-service leaves.
- Skills cite source with **file:line** and refuse to invent facts when knowledge is missing, so engineers drive ticket work end-to-end without re-reading the codebase from scratch each conversation.

GenAI Collateral Engine — *Vertex AI (Gemini), Cloud Run, FastAPI, Firestore, Pub/Sub, Python · github.com/yashgpt2894/genai-marketing-collateral*

- Built and deployed a **production GenAI service live on Google Cloud** that turns two companies' PDFs into factually-grounded, layout-ready B2B marketing collateral: multimodal parsing → source-tagged company briefs → grounded, cited generation (Gemini) → deterministic validate-and-repair loop enforcing word limits and layout in code, with claim-level faithfulness checks and abstention over invention.
- **Cloud Run** (scale-to-zero) with Pub/Sub + DLQ async parsing, Firestore + GCS (CMEK, EU residency), Terraform-provisioned infra, Cloud Build CI/CD; prompt-injection hard gate (fail-closed), 51 offline tests, golden-set eval harness; measured cost **~$0.01 per article**, token + USD stamped on every generation.

Sensor Breakdown Diagnostics — *semi-supervised ML case study; scikit-learn, Python*

- Classified 1,600 machine breakdowns from only **40 expert labels**: proved the raw 20-sensor data has no cluster structure, isolated the failure signal to a single sensor (ANOVA F ≈ 25), and propagated labels via Label Spreading validated with leak-free nested CV and permutation tests.
- Calibrated confidence with an **"unknown" bucket** routing novel failure modes to human experts; designed the production path on Vertex AI Pipelines + BigQuery with active-learning retraining.

EDUCATION & CERTIFICATIONS

B.Tech, Computer Science & Engineering, SRM Institute of Science and Technology, India *2012 – 2016*

HackerRank: 5-star in Problem Solving and Python