

VAIBHAV LEHRA

Senior Python Engineer | FastAPI · OpenAI & Anthropic Integrations · AI Orchestration · RAG · Cloud-Native
Delhi NCR, India | vaibhavlehra@gmail.com | vaibhavlehra.com | github.com/vlehra | linkedin.com/in/vaibhavlehra

PROFESSIONAL SUMMARY

Python engineer with 6+ years building production-grade distributed microservices and REST APIs in FastAPI, Django and Flask across AdTech and EdTech, secured with OAuth 2.0. Builds AI workflows on OpenAI and Anthropic models — prompt engineering, LangChain/LangGraph multi-agent orchestration, RAG on pgvector, MCP, SSE streaming and guardrails. Designs for reliability with retries and backoff, structured logging and Prometheus/Grafana and ELK observability, on Docker/Kubernetes across AWS and GCP. Strong at root-causing production issues: 7.2M-product, 6.2 GB feeds now stream in 2 GB of RAM, and a render pipeline re-architected for 2.3x throughput. Mentors a 10-engineer team and uses AI tools daily in Agile delivery.

TECHNICAL SKILLS

Python & Backend: Python (async/await, type hints, OOP, data structures), SQL, TypeScript · FastAPI (async endpoints, Pydantic validation, SSE streaming), Django, DRF, Flask, Celery, microservices, REST / OpenAPI design

Data & Databases: PostgreSQL (pgvector), MySQL, Redis, MongoDB, Elasticsearch · Pandas, Polars, Jupyter, streaming and batch ingestion · query optimization, indexing, profiling, caching

Cloud & DevOps: AWS (EC2, S3, RDS, SQS), GCP · Docker (multi-stage builds), Kubernetes (health probes, ConfigMaps) · Jenkins, GitLab CI, SonarQube · Prometheus, Grafana, ELK / Kibana, structured logging · SQS, RabbitMQ · Terraform (working knowledge)

Generative AI: OpenAI and Anthropic APIs, prompt engineering, LangChain / LangGraph, multi-agent orchestration, RAG with pgvector, MCP servers and tools, SSE streaming, LLM input/output guardrails · Claude Code

Security & Frontend: OAuth 2.0, JWT, API keys / scoped tokens, DRF permissions, rate limiting, LLM guardrails · React.js, Redux, TypeScript

Ways of Working: Agile / Scrum, Jira, design and code reviews, POCs and pilots, mentoring, production support, cross-team collaboration

PROFESSIONAL EXPERIENCE

Software Development Engineer II | Tyroo Technologies, Gurugram

Feb 2023 – Present

- Own architecture and delivery of distributed Python microservices (FastAPI, Django, Flask) on two platforms — the Product Information Management (PIM) service ingesting multi-million-product catalog feeds, and the Dynamic Product Ads pipeline delivering video creatives to Facebook and Snapchat — from design docs through Dockerised Kubernetes deployment, CI with SonarQube gates, retries with backoff, structured logging, Prometheus/Grafana and ELK observability and production support; work with product, QA and ad-ops in Agile sprints.
- Mentor a 10-engineer team and lead its LLM-based development with Claude Code and agentic workflows; build MCP servers and tools, and integrate OpenAI and Anthropic APIs with SSE streaming and LLM input/output guardrails.
- AI QA Agents** — Architecting a multi-agent QA platform on LangChain/LangGraph: an orchestrator routes each Bitbucket PR to six single-purpose agent workers that design, run and triage tests. A RAG pipeline on pgvector retrieves the versioned Confluence PRD and linked Jira story to ground LLM test generation; agents file only evidence-backed bugs, retest every fix and run sanity and regression on an isolated production test brand, with human gates on test sets, UAT, release and rollback. Authored the architecture proposal, quality gates and phased pilot rollout.
- PIM** — Designed and piloted a replacement for a RAM-bound ingestion engine (whole feeds held at ~15x file size): a two-pass Polars streaming DataFrame pipeline at flat memory — 4.1 GB and 6.2 GB feeds (2.8M / 7.2M products) now process in 2 vCPU / 2 GB where the old engine exhausted 6.5 GB in seconds, retiring the S3 chunk-splitting workaround. De-risked the cutover with a field-level parity harness across both engines — byte-identical on real feeds, corrupted encodings and a 19-case hostile edge corpus.
- PIM** — Roughly doubled per-worker throughput of event-driven SQS workers via long-polling and cached clients/queue URLs; improved connection recovery by hardening MySQL and Redis reconnect handling; migrated the service from end-of-life Python 3.9 to 3.11.
- Renderer** — Re-architected the distributed video render pipeline: 65s to 28s per video (57% faster) and 2.3x per-worker throughput (~55 to ~128 videos/hour), halving renderer infrastructure at constant volume.
- Renderer** — Debugged and eliminated a 112 MB per-job memory leak causing OOM crashes and lost in-flight renders on 8 GB Kubernetes pods; cut peak memory 27% and DOM objects per render from 38,353 to under 1,000.
- Renderer** — Cut output video size 92% (25 MB to 2.1 MB) via codec-optimized MP4 encoding, reducing S3 and CDN egress proportionally; added Redis template caching (60s+ timeouts to sub-10ms) and a disk-backed brand-audio cache (1.5–2.5s to ~5ms on 99%+ of renders).

Backend Developer | Qricle, Gurugram

Oct 2022 – Dec 2022

Backend Developer | Myclassroom Learning Pvt. Ltd., Noida

Sept 2020 – Oct 2022

- Architected the PostgreSQL data model for a live-learning platform spanning live classes, test series, video lectures and student performance analytics.
- Built the DRF API layer — CRUD, filtering, pagination, authentication — powering React web and mobile clients, and standardized deployment across multiple EC2 instances on NGINX and Unicorn; built and led the engineering support function, mentoring engineers and establishing triage and escalation for production issues.

EDUCATION

B.Tech, Electrical and Electronics Engineering — Guru Tegh Bahadur Institute of Technology, GGSIPU, Delhi | 2018