Smeet Agrawal

+91 82006 81327 | smeetagrawal23@gmail.com | linkedin.com/in/sagrawal237 | github.com/Smeet23

Software Engineer at Big Circle shipping production systems for client products: payment rails, multi-tenant platforms, concurrency-safe data models, and evaluated RAG pipelines. Backend depth first, applied AI on top — Python and TypeScript on PostgreSQL, Redis and Azure.

Technical Skills

Languages: Python, TypeScript, JavaScript (Node.js), SQL, C++
Backend: FastAPI, Next.js, REST APIs, BullMQ, Celery, Prefect, Server-Sent Events, OAuth2 / mTLS, RBAC
AI & RAG: RAG pipelines, hybrid retrieval (BM25 + vector, RRF), reranking, LangGraph, LangChain, RAG evaluation (RAGAS, DeepEval), citation grounding, Langfuse
Databases: PostgreSQL (transactions, advisory locks, triggers), pgvector, Qdrant, Redis, Drizzle, Prisma, SQLAlchemy
Infrastructure: Docker, Azure (App Service, AKS, Blob, Key Vault), Kubernetes/KEDA, GitHub Actions, Auth0, Vitest, pytest

Professional Experience

Software Engineer Jun 2025 – Present
Big Circle
Backend and AI engineering across production client products – payments, contract accounting and document intelligence.

- **Payments integration** – integrated a Banking-as-a-Service provider for IBAN payouts on a rewards platform serving ∼5,000 users (OAuth2 over mutual TLS, HMAC-SHA256 request signing, idempotent webhook ingestion via an event ledger) and hardened the payout ledger against double payments and refunds with single-transaction debits, `SELECT …FOR UPDATE` status transitions and `EXCLUDE` constraints.
- **Concurrency correctness** – designed a per-contract Postgres advisory-lock protocol closing a check-then-write race that stranded billable hours on retired charge codes, backed by `BEFORE INSERT/UPDATE` triggers on eight tables so stale clients, imports and background recomputes cannot resurrect deleted state.
- **Silent-failure elimination** – diagnosed a unique-constraint violation that aborted an entire contract's cost recompute and was swallowed into an audit row, leaving finance dashboards serving hours-old figures with no visible error; fixed the projection's uniqueness scope and surfaced recompute failures to users.
- **Financial data model re-architecture** – on a government contract-accounting platform, collapsed a charge-code/CLIN mapping layer into a unified project-code model across 238 files and 14 coordinated migrations (a net removal of ∼4,800 lines), with preflight-gated migrations that abort rather than half-migrate ambiguous customer data.
- **Access control and test infrastructure** – implemented dynamic role-based access control (database-defined roles, platform invitations, field-level permissions) and stood up a Vitest and Postgres-backed integration harness with GitHub Actions quality gates and secret scanning, authoring 141 test files including cross-contract IDOR and per-permission suites.
- **Multi-tenant platform and async jobs** – on a document intelligence platform, built Auth0 role-based access with organization-level data isolation and audit logging; six BullMQ/Redis queues with retry and idempotency handling, streaming real-time progress over SSE backed by Redis Streams; deployed on Azure AKS with KEDA queue-based autoscaling and Key Vault secret injection.
- **Retrieval, evaluation and agents** – built hybrid retrieval over pgvector and Postgres BM25 fused with Reciprocal Rank Fusion and cross-encoder reranking, fed by Docling/TableFormer ingestion that extracts tables as first-class entities; a Postgres-checkpointed LangGraph research agent runs a six-stage subgraph (an on-demand findings store cut its context 10–20×), with RAGAS/DeepEval CI quality gates over faithfulness, citation accuracy and numeric grounding, and claim-level fact-checking resolved to word-level PDF highlights.

Projects

LexIntel – Legal Research Platform 2026
FastAPI, Celery, Qdrant, PostgreSQL, Redis, Next.js

- Built the platform solo and wrote its **answer-trust layer**: a format-aware citation parser that resolves every LLM citation back to its source chunk by line-range overlap, strips unresolvable citations from the answer text, and scores answers on a four-factor confidence model; plus per-matter Qdrant collections with deterministic IDs making reprocessing idempotent, and Redis pub/sub → SSE progress that survives a page refresh.

Open Source Contributions

- **Docling / docling-core** *(Python, IBM)* – Excel-to-Markdown serializer rendering sheet names as headings, landed across both repositories; PDF ligature-map and deeply-nested LaTeX macro fixes *(#3635, #587, #3249, #3254)*.
- **nginx** *(C)* – SSL: log `SSL_R_RECORD_LAYER_FAILURE` at info level, fixing TLS 1.3 error-stack misreporting that logged ordinary client disconnects as critical *(#1267)*.
- **Maxun** *(TypeScript)* – added missing i18n translation keys across all locales *(#1025)*. Further PRs under review in gitleaks and Kubernetes docs.

Education & Certifications

Dhirubhai Ambani Institute of ICT (DA-IICT) Jul 2025
Bachelor of Technology in Information & Communication Technology

Certifications: Anthropic Claude developer courses – Building with the Claude API, Introduction to Model Context Protocol, Introduction to Agent Skills, and Claude Code in Action. **Problem solving:** 500+ LeetCode, 150+ Codeforces.