

PRIYABRAT DALBEHERA

AI Engineer | LLM Systems | Multi-Agent Orchestration | RAG Pipelines | Production GenAI | 4 Years End-to-End Ownership

+91-7749086836 | ipriyabrat689@gmail.com | Bengaluru, India | [LinkedIn](#) | [GitHub](#) | [Portfolio](#) | Open to Remote / Bengaluru

4 Yrs Production AI	300K+ Rows / Real-time	2 Live SaaS Products	~80% Manual Work Cut	40% Latency Reduced	6-Layer Guardrail System
------------------------	---------------------------	-------------------------	-------------------------	------------------------	-----------------------------

SUMMARY

AI Engineer with 4 years of end-to-end production ownership — designed, built, and deployed multi-agent LLM systems that are in use today. Shipped a **3-agent LangGraph NL-to-SQL pipeline** (300K+ rows, real-time) that eliminated SQL hallucinations for C-suite reporting via a custom 6-layer guardrail framework; cut manual analyst work by **~80%** and p95 latency by **40%**. Solo-shipped **2 live AI SaaS products** with paying users — full stack, from architecture to CI/CD. Obsessed with system reliability, LLM cost control, and shipping fast without breaking things. **Looking to join a product-engineering team** where AI is the core product, not a side feature.

TECHNICAL SKILLS

AI / Agents	LangGraph, CrewAI, AutoGen, Multi-Agent Orchestration, Tool-Use Agents, MCP Protocol, A2A
LLMs	OpenAI GPT-4o, Groq, Claude API, Gemini, Prompt Engineering, LoRA / QLoRA Fine-tuning
RAG / Vector DB	FAISS, ChromaDB, pgvector, BM25, Hybrid Search, Re-ranking, RAGAS Evaluation
Backend	FastAPI, Python (Advanced), SQL (Advanced), Pydantic, SSE Streaming, REST APIs, WebSocket
Data / Analytics	DuckDB, Pandas, NumPy, NL-to-SQL, ETL Pipelines, SQL Server
DevOps / Cloud	Docker, Docker Compose, AWS (EC2, S3, ECR, ECS Fargate), GitHub Actions CI/CD
Observability	LangSmith, RAGAS, Cost & Latency Tracking (P50/P95/P99), Hallucination Scoring
Databases	PostgreSQL, pgvector, ChromaDB, Redis (caching), SQL Server

EXPERIENCE

Senior Software Engineer — AI & GenAI Systems

Aug 2023 – Present

Anxilla Technologies Pvt. Ltd. · Bengaluru, India

- Architected and owns a **3-agent LangGraph NL-to-SQL pipeline** (query rewriting → SQL generation → AI summarization) converting plain-English business questions into DuckDB SQL for C-suite KPI dashboards — Revenue, Load Factor, Yield — across **300K+ rows of live airline operations data**; used daily by leadership for pricing decisions
- Engineered a **6-layer guardrail framework** (input sanitization → schema-aware SQL validation → output filtering → PII masking → confidence scoring → anomaly alerting) that drove SQL error rate to **near-zero** and eliminated hallucinations in executive-facing reports — zero escalations since launch
- Built **autonomous self-correction loops**: agents detect SQL failures, auto-rewrite ambiguous queries with schema context, and gate output on confidence threshold before surfacing results; **cut manual analyst intervention by ~80%**, freeing 2 FTEs for higher-value work
- Optimized pipeline with **async streaming + agent state caching**, reducing p95 response latency from **8s → under 5s (40% improvement)**; maintained SLA under concurrent C-suite query load
- Deployed on **AWS EC2 + Docker** with full GitHub Actions CI/CD (lint → test → Docker build → deploy); zero-downtime rolling releases across **1,000+ agent runs/day** with no production incidents
- Instrumented **LangSmith observability**: cost per query (USD), token usage, P50/P95/P99 latency, and RAGAS hallucination scores — enabled data-driven model selection (Haiku vs Sonnet) that **cut per-query LLM cost by ~30%**

ML Engineer — Forecasting & Analytics

Jun 2022 – Aug 2023

Revermax (now Anxilla Technologies Pvt. Ltd.) · Bengaluru, India

- Built **ML-driven Pax and fare forecasting models** (Python, Pandas, SQL Server) predicting passenger demand across airline routes; forecasts directly informed pricing strategy for revenue management teams
- Engineered **4 KPI analytics pipelines** processing 300K+ rows (Revenue, Passengers, Load Factor, Yield) with Current vs Last Year vs Target benchmarking — replaced 100% of legacy Excel reporting
- Automated ETL pipelines with preprocessing and anomaly detection, **cutting data prep time by ~50%** and saving **~10 analyst-hours/week**; zero manual data-cleaning steps remaining in the workflow

LIVE AI PRODUCTS (SHIPPED & RUNNING)

Ask Priya — 24/7 Multilingual AI Voice Assistant **[LIVE]** [aiwithpriyabrat.com](#)

Stack: Python · LLM APIs · BM25 RAG · SSE Streaming · Telegram Bot · GitHub Actions CI/CD

- Production AI agent — **multilingual, 24/7 uptime**, auto-syncs **11+ GitHub repos hourly** for real-time project awareness; self-improving memory via visitor feedback loops with zero human maintenance
- Real-time **Telegram recruiter alert system** with geolocation + one-click Google Calendar booking; fully automated CI/CD — every git push triggers test, build, and deploy in under 3 minutes

SmartChatX — Multi-Tenant AI SaaS Chat Platform **[LIVE]** [smartchatx.aiwithpriyabrat.com](#)

Stack: FastAPI · Python · pgvector · Docker Compose · JWT Auth · OpenAI / Groq / Claude

- Production **multi-tenant AI SaaS** — per-tenant pgvector store isolation, pluggable LLM providers (OpenAI, Groq, Claude), customizable RAG knowledge bases; built for zero cross-tenant data leakage by design
- **Solo-built from scratch to production**: JWT auth, usage analytics dashboard, Docker Compose containerization, one-command deploy — full product lifecycle owned by one engineer

KEY ENGINEERING PROJECTS

AgentOps-Deploy — Production LangGraph Agent on AWS ECS Fargate [GitHub](#)

Stack: LangGraph · Docker · AWS ECR · ECS Fargate · ALB · GitHub Actions CI/CD

- **Production-grade CI/CD pipeline** for LangGraph multi-agent systems: lint → test → Docker build → ECR push → ECS Fargate deploy, triggered automatically on every merge to main — 0 manual deploy steps
- Zero-downtime rolling updates, container health checks, ALB routing, and semantic agent versioning — reference architecture for deploying agentic workloads at scale

multi-agent-research-analyst — 6-Agent LangGraph Research Pipeline [GitHub](#)

Stack: LangGraph · OpenAI · FAISS · FastAPI · SSE Streaming · Streamlit

- **6-agent graph pipeline** (planner → researcher → critic → writer → reviewer → publisher) generating analyst-grade research reports grounded in FAISS RAG — each agent has a distinct role persona and can challenge upstream outputs
- Real-time SSE streaming to Streamlit UI; full production-quality agentic orchestration with graph state persistence across agent hops

LLMWatch — LLM Observability & Evaluation Platform [GitHub](#)

Stack: LangChain · LangGraph · RAGAS · LangSmith · Streamlit · PostgreSQL · GitHub Actions

- Observability layer tracking **cost (USD/query)**, **latency (P50/P95/P99)**, **token usage**, and **RAGAS hallucination scores** across LangChain/LangGraph agent runs — all metrics persisted in PostgreSQL for trend analysis
- Live Streamlit dashboard + **GitHub Actions CI quality gate**: build auto-fails if RAGAS score drops below threshold — continuous LLM quality enforcement baked into the release pipeline

VisionRAG — Multimodal Document Intelligence [GitHub](#)

Stack: GPT-4o Vision · FastAPI · pgvector · OpenAI Embeddings · Docker

- **Multimodal RAG pipeline** — GPT-4o Vision extracts text, tables, and key-value pairs from PDFs, scanned images, and invoices; pgvector similarity search returns source-cited answers with page-level attribution
- Production-ready async FastAPI backend, full Docker Compose stack — one-command deploy; handles mixed document types in a single unified pipeline

finance-ai-agent — Autonomous 4-Agent Investment Analyst [GitHub](#)

Stack: LangGraph · Groq · Financial APIs · Python

- **4-agent debate system** — Bull, Bear, Risk Analyst, and CIO agents each build independent investment theses, then challenge each other via LangGraph state machine before converging on a risk-adjusted recommendation
- Demonstrates **adversarial multi-agent reasoning**: no single agent can override the others — consensus is required, mimicking a real investment committee decision process

EDUCATION

M.Tech — Computer Science	Odisha University of Technology and Research (OUTR), Bhubaneswar	2019 – 2021
B.Tech — Engineering	Parala Maharaja Engineering College (PMEC)	2014 – 2018

WHAT I BRING TO A PRODUCT TEAM

Ownership	Solo-shipped 2 live AI SaaS products and 5+ production systems — design → deploy → monitor
System Design	Multi-agent orchestration, guardrail architecture, multi-tenant isolation, CI/CD pipelines
Cost Engineering	LLM cost routing (Haiku vs Sonnet), semantic caching, token budgeting → ~30% cost reduction
Observability	RAGAS eval, LangSmith tracing, P50/P95/P99 dashboards, CI quality gates on hallucination scores
Velocity	NL-to-SQL multi-agent pipeline: designed, built, and deployed to production in 3 weeks
Communication	Owns C-suite AI dashboards; translates LLM system design into business outcomes for leadership