# ADITYA SHARMA

**Software Engineer | Backend & AI Systems**
Bhopal, MP | adityasharma00070@gmail.com | (+91) 8535082365
LinkedIn | GitHub | LeetCode (Guardian) | Codeforces (Specialist)

## SUMMARY

Backend & AI Systems Engineer specializing in multi-agent LLM workflows (LangGraph, MCP), scalable RAG pipelines, and CUDA-accelerated systems. Skilled in LLM post-training (RLHF, RLVR) and rigorous LLM evaluation (RAGAS, DeepEval). Elite competitive programmer (Top 0.09% LeetCode Guardian, Codeforces Specialist) with a proven track record of optimizing complex algorithms and high-performance applications.

## EDUCATION

**Indian Institute Of Information Technology (IIIT), Bhopal** *2023 – 2027*
B.Tech in Information Technology **CGPA: 9.02 / 10.0**
***Coursework:*** *Distributed Systems, Data Structures & Algorithms, Linear Algebra, Probability & Statistics, Operating Systems.*

## TECHNICAL SKILLS

- **Languages:** Python (High-Performance & Concurrent), C, CUDA, Java, C++, SQL, Bash.
- **Deep Learning & Model Training:** PyTorch, Core Deep Learning, Neural Networks, LLM Fine-Tuning (LoRA/QLoRA), Post-Training, vLLM, LLM Evaluation (RAGAS, DeepEval), GPU Kernel Optimization (Nsight Compute/Systems).
- **AI Agents & Tooling:** LangGraph, Model Context Protocol (MCP), LLM APIs (OpenAI, Groq, Anthropic), Vector Databases (ChromaDB, Qdrant).
- **Infrastructure & Backend:** RESTful APIs, FastAPI, Redis, Celery, PostgreSQL, Docker, Linux/Unix, GitHub Actions.

## ACHIEVEMENTS & OPEN SOURCE CONTRIBUTIONS

- **Open Source:** Contributed a merged PR to **LeetGPU's JAX compiler**, fixing a bug in the kernel-compilation path for one of the platform's GPU-programming challenge problems.
- **Competitive Programming: LeetCode Guardian** (2665, Top 0.09% globally), **CodeChef 4-Star** (1856, Global Rank 168/28k+ in Starters), **Codeforces Specialist** (1468).

## EXPERIENCE

**Freelance Systems Evaluation Engineer** | **Airdawg Labs** *May 2026 – Aug 2026*

- **Engineered** deterministic evaluation suites in Python, Go, and Rust across 1,200+ test cases within isolated Docker environments, benchmarking frontier LLM coding agents and reducing CI/CD pipeline validation bottlenecks by 35%.
- **Calibrated** task difficulty for GPT and Claude-class models, identifying critical failure modes and driving worst-model pass rates below 20% to ensure rigorous, enterprise-grade model evaluation.

**AI Engineer Intern** | **Get Set Skilled™ Pvt Ltd** *July 2026*

- **Integrated** LLMs and built Retrieval-Augmented Generation (RAG) pipelines for the Margdarshak AI platform, improving API response times by 15% and ensuring highly relevant context retrieval.
- **Collaborated** with the engineering team to resolve complex system bugs and deploy software enhancements, achieving 100% compliance with enterprise security and coding standards.

## PROJECTS

**CUDA Reranker** | *CUDA, PyTorch, HuggingFace Transformers, Nsight Compute/Systems* | Code

- **Built** a RAG reranking pipeline (SciFact/BEIR, MiniLM cross-encoder), profiled it with Nsight Compute to find the true attention bottleneck, then hand-wrote a fused CUDA kernel via register caching, tiling, vectorization, and an online-softmax redesign, cutting latency 286ms $\rightarrow$ 10.5ms ($\sim$29x).
- **Added** padding-mask support and a tile-skip optimization guided by Nsight stall analysis, reaching 7.70ms vs. PyTorch SDPA's 10.27ms masked latency ($\sim$25% faster).
- **Integrated** the kernel into an unmodified HuggingFace BERT model via `transformers`' `AttentionInterface`, validating near-machine-precision logits (1.9e-6 error) and zero ranking regression (20/20 top-10 match).

**repo-trace** | *Python, LangGraph, ChromaDB, RAGAS, Gemini* | Code

- **Architected** a Corrective RAG (CRAG) assistant as an 8-node LangGraph `StateGraph` that self-grades retrieval quality (correct/ambiguous/incorrect) and routes to query refinement, live web fallback, or reranking before generation, preventing hallucinated answers from weak retrieval.
- **Engineered** a custom asymmetric MMR retriever and a structured-output relevance filter, root-causing and replacing a production LLMChainFilter failure caused by Gemini's multi-block response format.
- **Built** a citation-validity guardrail with two independent retry-loop counters (retrieval vs. generation) and validated the full pipeline via RAGAS evaluation (0.884 Faithfulness, 0.839 Context Precision) sliced per graph execution path.