K. Sri Rishikesh Varma

AI & ML Engineer | **Python** | **Agentic Systems, LLM Tooling & Evals**

Hyderabad, Telangana, India | +91 6301852709 | rishikeshvarma9854@gmail.com

linkedin.com/in/rishikeshvarma-kucharlapati | github.com/rishikeshvarma

Summary

Final-year B.Tech (CSE, AI & ML) student and AI engineer who builds agentic systems in Python: tool-using agents, local LLM runtimes, multi-agent orchestration (MCP, A2A), and the eval and tracing tooling used to test them. Co-inventor on a published Indian patent application and author of an IEEE Xplore paper. Projects include a self-healing LLM compiler with a CI regression gate, a local personal AI agent with an intent-routing memory layer, and a semantic firewall for MCP traffic. Uses AI coding agents (Claude Code, Antigravity) daily for building, debugging and code review.

Technical Skills

Languages: Python, Java (OOP), SQL, JavaScript

AI Agents & LLMs: Tool/Function Calling, Model Context Protocol (MCP), Multi-Agent Systems (A2A), Intent Routing, Agent Memory, Context Window Management, RAG, Prompt Engineering, LLM APIs (Claude, Gemini, Groq)

ML & Model Serving: PyTorch, Hugging Face Transformers, llama.cpp, vLLM, Ollama, ONNX Runtime, Model Quantization, Embeddings

LLM Evals & Testing: Eval Harnesses, LLM-as-Judge, Tracing, Regression Testing, Pytest, Pydantic

Backend, Data & DevOps: FastAPI, PostgreSQL (pgvector), SQLite, Supabase, Docker, GitHub Actions (CI/CD)

Tools: Git, GitHub, Linux (Ubuntu), Claude Code, Antigravity, n8n

Experience

Ties Verse Apr 2026 – Aug 2026

Data Engineer (Remote) Hyderabad, India

- Built automated Python data scrapers and API aggregation pipelines that collected and normalized news sources for tracking and community distribution.
- Conducted data research workshops on geopolitical analysis and global trends using structured intelligence-mapping datasets.

Projects

AppCompiler: Self-Healing LLM Compiler + Eval Harness github.com/rishikeshvarma/APP_Compiler

Tech: Python, Gemini API, FastAPI, Pydantic, Pytest, GitHub Actions

- Built a 4-stage LLM pipeline (Intent, Design, Schema, Validate/Repair) that compiles a plain-English request into UI layout JSON, REST endpoint specs and a PostgreSQL schema with immutable audit logs.
- Built an eval harness with deterministic UI-API-DB consistency checks, an LLM-as-judge limited to what code cannot decide, step-level traces (latency, tokens, error class) and a schema-versioned dataset.
- Added a baseline regression gate in GitHub Actions CI (52 tests); the harness exposed 4 bugs in the compiler, including a validator that silently skipped its main check by reading the wrong JSON path, each fixed and locked in with a regression test.

Agent-Dummie: Local Personal AI Agent Personal Project | In Progress

Tech: Python, llama.cpp, Qwen3.8 27B (GSQ-RCO quantized), Kubernetes, SQLite, Markdown Knowledge Vault, Git

- Building a privacy-first personal AI agent that runs fully on local hardware (RTX 5060 Ti, 16 GB VRAM), with tool calling, web search and persistent memory; core engine verified end to end, including recovery after a power-cut reboot.
- Serves Qwen3.8 27B through llama.cpp with GSQ-RCO quantization and a 225K-token context window, and runs two agents as Kubernetes pods sized to fit a single consumer GPU.
- Designed a dual memory layer (SQLite database + Markdown knowledge vault) and a 4-way intent router (Personal / Search / Mixed / Meta) that decides between memory retrieval and live web search, stopping public web data from being stored as personal facts.
- Diagnosed a multi-turn memory failure caused by letting the LLM decide when to write memory; rolled back to the last verified build via Git and redesigned writes around deterministic, gated triggers.
- Built with AI coding agents and reviewed their output critically, catching generated code that hardcoded answers for specific test inputs instead of general logic.

MCP-Shield: Semantic LLM Firewall for Model Context Protocol Final-Year Project (4-member team) | In Progress

Tech: Python, MCP, JSON-RPC 2.0, ONNX Runtime, mDeBERTa-v3 (86M), llama.cpp, Qwen3-8B (GGUF), FastAPI, SQLite

- Building a local inline security proxy (mcp-shield proxy, PyPI package mcp-shield-llm) that sits between MCP hosts such as Claude Desktop and Cursor and their MCP servers, inspecting both directions of JSON-RPC 2.0 traffic for indirect prompt injection, tool-description poisoning and command injection.
- Designed a 3-tier hot-path cascade (structural policy gate under 1 ms, cache and signature layer at 1–5 ms, fine-tuned 86M mDeBERTa-v3 encoder classifier on ONNX Runtime) targeting p95 latency under 50 ms with a GPU and 250 ms CPU-only, chosen over a generative SLM whose per-call review takes seconds.
- Designed a one-engine-per-user process model: thin per-server front-ends share one engine that holds the model, cache, taint store and SQLite event store, so the model loads once per machine; mcp-shield doctor picks the ONNX execution provider (CPU, CUDA, DirectML, CoreML) and event loop per OS.
- Specified cross-server taint tracking with winnowed fingerprints that guarantee detection of any reused 32-character run, SHA-256 capability pinning with first-contact quarantine, and Spotlighting markers on forwarded tool text.
- Designed the optional explainer (Qwen3-8B, Q4_K_M GGUF) on a pinned llama.cpp llama-server with JSON-schema-constrained output, chosen over Ollama to control the build, quantization, context and sampling; the same server runs the benchmark agent and synthetic data generation.
- Target: cut Attack Success Rate from a 52.8% baseline to under 10% on held-out AgentDojo and MCPSecBench suites while retaining at least 95% task utility, benchmarked against Prompt Guard 2.

PhishNinja: Agentic AI Security Framework Patent Application Published

Tech: Python, Supabase, pgvector, Groq, Llama 3.1 8B, n8n

- Engineered a custom multi-agent system from scratch with an intent-classifier routing core that isolates live DOM mutations for analysis.
- Integrated 384-dimensional embeddings (all-MiniLM-L6-v2) with Supabase pgvector (PostgreSQL) for semantic threat matching.
- Served Groq LPU inference (Llama 3.1 8B Instant) for real-time explainable-AI justifications within a 640 ms budget, keeping edge queues under 1% of cloud API rate limits.
- Built an out-of-band alerting ring that triggers webhooks to a Raspberry Pi cluster running n8n for mobile alerts; achieved 99.7% precision across 1,500 validation trials.

Patents & Publications

- **Patent Application (published, co-inventor):** PhishNinja: An NLP-Driven Framework for Phishing and Spam Detection Using Explainable AI. Application No. 202541111904 A, India. 98% accuracy with quantized on-device inference.
- **IEEE Publication:** Phishing Detection: A Comprehensive Survey of Methods and Trends. DELCON Conference, published in IEEE Xplore (Document 11400192).

Education

Keshav Memorial Institute of Technology Hyderabad, India
B.Tech in Computer Science and Engineering (AI & ML), CGPA: 7.5 2023 – 2027

Narayana Junior College Hyderabad, India
Class XII (Intermediate), 86.6% 2023

FIITJEE World School Hyderabad, India
Class X (SSC), GPA: 10/10 2021

Certifications & Achievements

Certifications: Generative AI / Deep Learning Specialization, IIT Hyderabad

Leadership & Activities: Tech Head, Riti Upcycling Club, KMIT; District-Level Tennis Runner-up; Learning German (A1)