HAMZA ALI

AI Engineer (Full-Stack) — Python, FastAPI, React & LLM Systems

Lahore, Pakistan (open to remote) | +92 348 8819211 | hamza171324@gmail.com
linkedin.com/in/hamza-ali-b9b8b22a6 | github.com/hamza1713 | personalportfolio-theta-gules-56.vercel.app

SUMMARY

Full-stack engineer focused on AI systems. I build the whole thing: FastAPI backends, React frontends, retrieval pipelines, evals, and Docker deployment. FinSight, my final-year project, is an enterprise RAG and Text-to-SQL platform where each department's documents stay scoped to its own users. The AI Code Review Agent is a multi-agent PR review service with an MCP server. I spend the time on the unglamorous parts: access control, tests (90 and 690+), and measuring when the model is wrong. Looking for a forward deployed role where I work directly with client teams to get AI into real workflows.

TECHNICAL SKILLS

AI & LLM: LangChain, LangGraph, CrewAI Flows, MCP, RAG (Hybrid Retrieval, ChromaDB, Reranking), Text-to-SQL, Tool Calling, Prompt Engineering, Gemini API, Evaluation (RAGAS, DeepEval)

Backend & Full-Stack: Python, FastAPI, REST APIs, React, TypeScript, JavaScript, SQL, Redis, Pydantic, Webhooks, Job Queues, Streaming (NDJSON / SSE)

Cloud, DevOps & Security: Docker, Docker Compose, GitHub Actions (CI), Git, Microsoft Azure (Azure ML), Bash, JWT, RBAC, Role-Scoped Data Access, Rate Limiting

ML & Testing: PyTorch, TensorFlow / Keras, Scikit-learn, NumPy, Pandas, Pytest, Regression Testing, Benchmark Design

EXPERIENCE

Independent AI / Full-Stack Engineer — Self-Directed & Open-Source Projects *Nov 2024 – Present*

FinSight — Enterprise RAG + Text-to-SQL Platform with Role-Based Access Control GitHub *FYP, 2026*

Python, FastAPI, LangChain, Gemini, ChromaDB, DuckDB, Redis, React 19, TypeScript, Docker

- Built the retrieval pipeline: 3-way query expansion, ChromaDB dense search, SQLite FTS5 (BM25), rank fusion, and Cohere reranking. Answers cite their sources and only draw on documents the user's role can see.
- Enforced department-level access in three layers: JWT with per-request role re-validation, a guard that blocks cross-department queries before they reach the LLM, and ChromaDB metadata filtering. The RBAC security suite scored 0.93 with zero metadata-filter violations.
- Added an LLM query router (RAG, SQL, or greeting) with automatic SQL-to-RAG fallback. Model-generated SQL runs in a sandbox: DuckDB in memory, SELECT-only, external access disabled, 10 s timeout, 1,000-row cap.
- Brought repeat-query latency under 20 ms with a role-scoped Redis cache (in-memory fallback). Added tiered rate limiting and streamed answers to a React client over NDJSON.
- Measured quality with RAGAS and a quota-free metric pipeline (14 QA samples: 0.84 answer relevancy, 0.82 answer similarity). 90 automated tests cover the API, RBAC, security regressions, and resilience. Ships with Docker Compose.

AI Code Review Agent — Multi-Agent Pull Request Review Platform GitHub *2026*

Python, CrewAI Flows, FastAPI, SQLite, Semgrep, Bandit, Ruff, MCP, React 19, TypeScript, Docker, GitHub Actions

- Designed the pipeline to run deterministic checks first. Semgrep, Bandit, Ruff, a custom AST constant folder, and an AST call-graph index all run before any LLM call, at zero token cost, which limits what the agents can claim.
- Ran a 3-agent CrewAI Flows crew (Senior Developer, Security Engineer, Tech Lead) in parallel with provenance guardrails. A reconciler written in plain code deduplicates findings and scores confidence from a documented rubric instead of a second LLM pass.
- Built a durable SQLite WAL job queue behind a FastAPI gateway for webhook processing (HMAC-validated webhooks, fail-closed Bearer auth, orphaned job recovery). It posts PR reviews, Check Runs, and SARIF v2.1.0 reports through the GitHub API and a reusable GitHub Action.
- Shipped 15 PR slash commands with author-association RBAC, an MCP server exposing 8 review tools to Claude Code and Cursor, GitLab and Bitbucket adapters, and a YAML policy engine. Posting and auto-remediation are off by default.
- 690+ automated tests on a Python 3.10–3.13 CI matrix, plus a 14-case scanner regression suite that matches by file, exact line, and defect type and counts every unmatched hit as a false positive.

Data Science Intern — Advanced Telecom Services (ATS) AI Lab *Aug 2024 – Oct 2024*

- Ran EDA and feature engineering on about 1 million NOAA lightning-strike records (2009–2018), alongside Azure ML training.
- Built a Tableau heat map of US strike density, identified Texas and the southeastern US as the highest-risk zones, and presented mitigation recommendations.

EDUCATION

BS Computer Science — Abbottabad University of Science and Technology, Pakistan *2022 – Jul 2026*

Final Year Project: FinSight AI, An Enterprise RAG Chatbot with Role-Based Access Control

Peer teaching: regularly taught and guided fellow students on technical skills and project decisions

CERTIFICATIONS

Deep Learning Specialization, DeepLearning.AI (Nov 2025) • Machine Learning Specialization, Stanford Online (Oct 2024) • Retrieval Augmented Generation, DeepLearning.AI (Dec 2025) • Multi-Agent Systems with CrewAI, DeepLearning.AI (Apr 2026)