

Mohammed Faizan Ahmed

info2faizanahmed@gmail.com | [+91-9391281076](tel:+91-9391281076) | github.com/AhmedFaizanDev | linkedin.com/in/mohdfaizanahmed | faizanahmed.in

Summary

Computer Science undergraduate (B.Tech, 2027) with two years of production experience across backend, full-stack and applied AI. Built a database view-maintenance engine benchmarked $\approx 3,600\times$ faster than recomputation and verified by 570 automated tests, a multi-tenant SaaS backend on Django and PostgreSQL, and an LLM routing system evaluated against measured baselines. Hackathon winner.

Education

Mahindra University (Mahindra École Centrale)
B.Tech, Computer Science and Engineering

2023 – 2027
Hyderabad, India

Technical Skills

Languages: Python, TypeScript, JavaScript, SQL, C/C++ (coursework)

Web & Backend: Django, Django REST Framework, Express.js, Node.js, REST API design, JWT authentication, OAuth 2.0, RBAC, OpenAPI

Databases: PostgreSQL, SQLite, Redis, schema design, indexing, query optimization

AI & Data: PyTorch, scikit-learn, XGBoost, pandas, NumPy, OpenCV, LLM integration, vector search

Tools & Infrastructure: Docker, Docker Compose, AWS EC2, Nginx, Git, GitHub Actions CI/CD, pytest, Linux/Unix, shell scripting, Postman

Experience

Artificial Intelligence Intern | *AI Linc Technologies Pvt. Ltd.*

Jul. 2026 – Aug. 2026

- Built generative-AI product features for client work, including automated video content generation and long-form document generation pipelines, integrated into backend services with input validation, retries, timeout handling and defined fallback behaviour on model failure.
- Cut per-request inference cost 35% by routing simpler requests to a smaller model and caching repeated sub-results in Redis, verified against an offline evaluation set so quality did not regress.

Full Stack Engineer | *AI Linc Technologies Pvt. Ltd.*

Jan. 2026 – Jun. 2026

- Built 20+ backend features end to end, owning each from database schema and migrations through API design to the contracts consumed by the frontend team.
- Integrated ML models into production backend services behind versioned REST endpoints, adding input validation and fallbacks so model failures degraded gracefully instead of breaking requests.

Full Stack Developer (Freelance) | *Catalyst Connect, Hyderabad*

Feb. 2026 – Mar. 2026

- Built the backend for a production exhibition and stall management platform delivered to a paying client, using Node.js, Express and PostgreSQL.
- Built the REST API with JWT authentication and role-based access, stall booking with conflict prevention on concurrent reservations, payment tracking and multi-exhibition support; deployed on Docker with Redis and a CI/CD pipeline, and used to run 3 exhibitions covering 200+ stalls.

Backend Developer | *YuVi Collab Technology Pvt. Ltd.*

Jun. 2024 – Sep. 2024

- Built and maintained 25+ Django REST API endpoints, authentication and background workflows over PostgreSQL for client-facing systems.
- Cut p95 latency on the heaviest list endpoint from 1.4s to 310ms by removing N+1 query patterns and adding composite indexes.

Head of Tech & Design | *Qubit Club, Mahindra University*

Jul. 2025 – Jun. 2026

- Led a 6-person technical team, owning the club website and technical delivery for 8 workshops and events reaching 400+ attendees.

Projects

ivm: Incremental Database View Engine | *Python, SQL, SQLite, pytest, GitHub Actions*

2026

- Built a library that keeps a declared SQL view up to date as its underlying tables change, by applying only what changed instead of re-running the whole query; this is the problem streaming databases solve with a dedicated server, packaged here as an importable library.
- Implemented incremental joins and **GROUP BY** aggregates that stay correct under deletes, including the cases that break naive versions, such as recovering the next **MIN/MAX** when the current one is removed.
- Because a wrong result here looks like a normal number, built the test suite around an independent recompute oracle: **570 tests** check the incremental result against a from-scratch recomputation after every single change, with CI running on Python 3.10–3.12.
- Benchmarked at **37,000 updates/sec** (p99 43µs) and $\approx 3,600\times$ faster than full recomputation on a 50,000-row dataset.

Blackgrid: Multi-Tenant Inventory Platform | *Django, DRF, PostgreSQL, Redis, Docker, Nginx*

2025

- Built a SaaS platform serving multiple independent businesses from one deployment, with data isolation enforced centrally in middleware (tenant IDs signed into the JWT and checked on every request), so one customer can never read another's data.
- Shipped 40+ REST endpoints with role-based access control (Owner/Admin/Viewer), CSV bulk import with validation, auto-generated OpenAPI documentation, and support for four industry verticals via a configuration adapter.

- Wrote 84 backend tests including cross-tenant access attempts that must return 403; deployed the full stack on Docker Compose with PostgreSQL, Redis and Nginx.

TinyRouter: Learned Routing Across Language Models | *Python, PyTorch, CMA-ES*

2026

- Built a tiny **10K-parameter** model that decides, for each question, which of three large language models should answer it, trained by an evolution strategy against a simple right/wrong reward, since the choice itself is not differentiable and gradient methods do not apply.
- Measured it properly on **120 held-out questions** with baselines averaged over 3 runs and compared against random routing as a control: **0.858 average accuracy vs. 0.835** for the best single model and 0.833 for random selection.
- Built a diagnostic proving how much gain was theoretically available, showing the remaining gap was a limitation of the router rather than of the model pool; tracked \$62 of API spend exactly from token ledgers.

PyTor: BitTorrent Client Built from the Protocol Spec | *Python, TCP Sockets, Multithreading*

2024

- Implemented a peer-to-peer file-sharing client directly from the BitTorrent specification: **.torrent** parsing, tracker communication, the peer handshake, message framing over raw TCP sockets, and multithreaded downloads from many peers at once.
- Verified each downloaded piece against its SHA-1 hash and re-requested corrupt pieces from other peers, sustaining 4.2 MB/s across 30 concurrent connections.

Fault-Tolerant LLM Content Pipeline | *TypeScript, Node.js, Express, Docker, AWS EC2*

2025

- Built a pipeline generating ≈ 250 -page structured documents from a single topic through **≈ 186 sequenced LLM calls**, with checkpoint/resume so a failure partway through a long run resumes without regenerating completed work.
- Held marginal cost to **$\approx \$0.12$ per document** through model tiering and context compression; deployed on AWS EC2 with Docker Compose.

Achievements

Winner: SUDHEE 2025 Hackathon | *CBIT, Hyderabad*

Mar. 2025

- Built a real-time speech-to-Indian-Sign-Language translator, placing first among 60+ competing teams; owned the streaming speech-to-text pipeline and gesture rendering, holding end-to-end latency under 1.2s so output tracked live speech.

3rd Place: AminoVerse Health-Tech Hackathon | *Mahindra University*

Apr. 2025

- Built the backend data-integration and vector search layer over four bioinformatics databases (PDB, UniProt, StringDB, DrugBank) for a gene-disease analysis assistant, delivered in a 48-hour build.

Certifications

Oracle Cloud Infrastructure 2025 Certified Generative AI Professional (Oct. 2025) | LLMs, RAG pipelines, semantic search, vector databases.

Red Hat Linux Administration III: Automation with Ansible (Udemy, Feb. 2025) | Linux system administration, shell scripting, Ansible automation.