

Muhammad Tayyab Tahir Qureshi

+92 335 4954174 | Lahore, Pakistan | m.tayyabtq@gmail.com
[linkedin.com/in/ttqureshi](https://www.linkedin.com/in/ttqureshi) | github.com/ttqureshi | [kaggle.com/mtayyabtahir](https://www.kaggle.com/mtayyabtahir)

SUMMARY

AI and Software Engineer with hands-on experience designing and deploying production-grade AI systems, including Agentic AI workflows, Retrieval-Augmented Generation (RAG) pipelines, and LLM orchestration. Proven track record building multi-agent systems using LangGraph and LangChain, containerized services with Docker, REST APIs with FastAPI and Django, and integrating cloud services on AWS. Currently pursuing an MS in Artificial Intelligence at LUMS while contributing to AI-first products at Arbisoft.

TECHNICAL SKILLS

Languages	Python
Agentic AI	LangGraph, LangChain, Multi-Agent Orchestration, AI Automation (n8n)
LLM & RAG	RAG Pipelines, LLM Orchestration, Prompt Engineering, Schema Design, VectorDB (ChromaDB, FAISS)
Backend	FastAPI, Django, Django REST Framework
AI / ML	Transformers, Neural Networks, CNNs, NLP, Computer Vision, Classical ML, Deep Learning
Libraries	PyTorch, Scikit-learn, NumPy, Pandas, OpenCV, PySpark, Hugging Face
Containerization	Docker, Multi-stage Dockerfiles, Docker Compose, Amazon ECR
Cloud & Infra	AWS (ECS, EC2, Lambda, S3, VPC), Amazon Bedrock, Terraform (IaC)
Tools	Git, GitHub, n8n, Taskiq (Redis), PostgreSQL, SQLAlchemy, Alembic
Engineering	System Design, Modular Architecture, Production Deployment

EXPERIENCE

Software Engineer (AI Focus) | *Arbisoft*

Jun 2024 – Present

Nightshift Agents — Autonomous Multi-Agent Automation Platform

- **Built an autonomous multi-agent system** that runs unattended overnight: agents ingest documents and emails, extract structured data, propose work, and report to a human operator via Slack — with sensitive actions gated behind human approval and every step fully audited.
- **Engineered the orchestration layer in n8n** — scheduled heartbeat workflows plus Google Drive and Gmail ingestion pipelines that feed a central data store through a secure gateway.
- **Designed a custom MCP (Model Context Protocol) gateway** exposing a tool API to the agents, deployed on **AWS ECS** behind an internal load balancer in a private VPC, with bearer-token auth, field-level encryption, and append-only (WORM) audit logging.
- **Implemented the agent reasoning loop on Amazon Bedrock (Claude)**, bridging MCP tools into the model's tool-use interface with a one-retry-then-escalate control flow and a propose-never-self-apply safety model.
- **Containerized services with Docker** — wrote multi-stage Dockerfiles for lean production images and a **docker-compose** setup for local multi-service development, then built and pushed images to **Amazon ECR** for deployment to ECS.
- **Provisioned cloud infrastructure as code** with **Terraform** (ECS, ALB, VPC, Secrets Manager), and PostgreSQL-backed persistence.

IT-Helpdesk Agent — LangGraph Multi-Agent System

- **Architected a production multi-agent system** using LangGraph with a classifier-driven orchestrator that fans work across dedicated subgraph lanes (RAG, integrations, escalation) and merges results via namespaced state, reducing ticket resolution complexity.
- **Designed a Chroma-backed RAG pipeline** with a retrieval step feeding an LLM tools loop; checkpointed runs on PostgreSQL for full conversation recovery across sessions.
- **Engineered a knowledge-base ingestion system** for document uploads (single and batch) with lifecycle management, presigned URL previews, and off-request processing via Taskiq workers to keep the request path non-blocking.
- **Built a Next.js + FastAPI full-stack interface** using an Nx monorepo with same-origin `/api` proxying and Vercel AI SDK-style streaming chat for real-time LLM responses.

BenchPrep — AI-Powered Interview Platform (FastAPI)

- **Developed async FastAPI REST APIs** with Pydantic validation, async SQLAlchemy on PostgreSQL, and Alembic migrations for a scalable interview management platform.
- **Integrated an OpenAI-powered interview evaluator** running off-request via Taskiq (Redis) to avoid blocking; implemented cron-based reminders and overdue detection as separate workers.
- **Implemented LiveKit-based real-time interview sessions** and S3 presigned URLs for secure resume uploads and private recording/report downloads.
- **Built a role-candidate matching pipeline** using FastAPI background tasks to asynchronously score and rank candidates against open roles.

Sama X VoC — Voice of Customer Platform

- **Delivered a call-review web application** for analysing customer call summaries, with long-running AI inference jobs offloaded to AWS Lambda for cost-efficient, event-driven scaling.
- Deployed and maintained the stack on AWS (EC2, Lambda, S3) using FastAPI and Python.

Open edX — Platform Modernisation

- **Led Python 3.12 enablement** and deprecation fixes across the Open edX stack, ensuring continued CI health across a large-scale open-source platform.
- Contributed to long-term platform modularisation, improving maintainability and reducing inter-component coupling.

OTHER PROJECTS

Wagtail AI Chatbot

- **Built a production-ready RAG plugin** for Wagtail/Django enabling semantic search and LLM-powered chat over site content using FAISS and ChromaDB.
- Designed a modular retrieval-and-generation pipeline with LangChain supporting persistent indexing, metadata filtering, and scalable vector retrieval.

Realtime Semantic Segmentation and Vision-Based Heading Controller [\[link\]](#)

- **End-to-end perception-control pipeline** from monocular video: **PIDNet** (PyTorch) for real-time 6-class semantic segmentation, fused with **dense optical flow** (OpenCV Farneback) and a **PD yaw controller** to produce heading and speed commands, with annotated video and per-frame CSV logging.

Document QA System — LangChain + ChromaDB

- Built an end-to-end document question-answering system: chunked PDFs, generated embeddings, stored vectors in ChromaDB, and retrieved context for LLM-generated answers.
- Tuned chunk size and retrieval parameters to improve answer relevance across varied document types.

Image Classification — CNN (PyTorch)

- Implemented and trained a CNN for image classification in PyTorch; applied preprocessing and normalisation to stabilise training and tracked loss/accuracy across epochs.

Spam Email Classifier [\[link\]](#)

- Built a TF-IDF + Logistic Regression spam classifier with full preprocessing (tokenisation, stopwords removal); evaluated with precision, recall, and confusion matrix.

Other: Canny Edge Detector ([from scratch](#)) | Image Segmentation | RISC-V Processor Pipeline | [Movie Genre Classification](#)

EDUCATION

MS Artificial Intelligence (weekend program)
Lahore University of Management Sciences (LUMS)

2025 – Present
CGPA: 3.88

BS Computer Engineering
University of Engineering & Technology, Lahore

2020 – 2024
CGPA: 3.827

CERTIFICATIONS

Generative AI — Analytix Camp | Supervised Machine Learning — DeepLearning.AI | Advanced Learning Algorithms — DeepLearning.AI | Intro to Machine Learning — Kaggle | Data Visualization — Kaggle