

MUHAMMAD ANAS

Munich, Germany | +49 170 9055176 | anashabib139@gmail.com

[linkedin.com/in/anas-baqai-bo21](https://www.linkedin.com/in/anas-baqai-bo21) | github.com/AnasBaqai | English (C1), German (A1)

SUMMARY

Full-Stack and AI Engineer with 2+ years of experience shipping production web applications. Comfortable across the stack, from **TypeScript** and **Python** backends to **React** and **Next.js** frontends and the **AWS** infrastructure underneath. Most recent work has been putting **large language models** to practical use: **AI agents** with **tool calling**, **RAG**, and document extraction pipelines that take manual work off people's desks. Currently an MSc Computer Science student in Germany and a published **IEEE** researcher. **Work authorization:** student visa, eligible for EU Blue Card.

EDUCATION

University of Passau | MSc in Computer Science | Passau, Germany | Oct 2024 - Present

National University of Computer and Emerging Sciences (FAST-NUCES) | BS in Software Engineering | Karachi, Pakistan | Aug 2020 - Jun 2024

WORK EXPERIENCE

Full-Stack AI Engineer | **Redseven Entertainment GmbH** (ProSiebenSat.1 Group) | Munich, Germany | Mar 2026 - Present

- Shipped **three internal web applications** in **NestJS**, **Next.js**, and **PostgreSQL** that replaced Excel and email based workflows for TV production teams, saving them **10+ hours a week** of manual data entry
- Designed "**Nellie**," an **AI assistant** that lets rights and finance staff ask licensing cost questions in plain English. A fast routing model reads each question and hands it to the specialist agent that can answer it, built on the **OpenAI Responses API**
- Wrote the **tool-calling loop** behind those agents, giving them **12 typed tools** for cost breakdowns and missing licensing data, and stored conversations in **PostgreSQL** so users resume days later
- Hardened the assistant for finance use: agents **propose** changes instead of writing them, and each approval is re-checked, applied in one **database transaction**, and logged. **Role-based permissions** block unscoped bulk edits
- Developed **LLM extraction pipelines** turning HR emails and scanned receipts into **schema-validated** records, flagging fields the model is unsure about instead of guessing at them
- Implemented the pricing engine reproducing legacy Excel cost rules across **111 licensors** with live exchange rates, and integrated **Microsoft Graph** (Outlook, **Entra ID SSO**).
- Deployed all three to **AWS** on **ECS Fargate** with **Aurora Postgres** and **S3**, provisioned in **Terraform**, with **Docker** builds and **GitLab CI/CD** releasing on merge

AI Engineer | **Arcpeak** | Munich, Germany | Aug 2025 - Feb 2026

- Built an **AI business-analysis tool** in **Python** on the **OpenAI GPT APIs** that reads a company's spend data and points out where enterprise clients are losing money
- Created the conversational side with the **OpenAI Agents SDK**, giving the assistant **tools** it could call, guardrails on what it would answer, and **Redis Streams** for resumable sessions. Users reached an answer **60% faster** than with the old report-based flow
- Replaced manual deploys by defining the **AWS** setup in **Terraform** (**ECS Fargate**, **RDS**, **ElastiCache**, **ALB**) and wiring **GitHub Actions** to build and release automatically, cutting deployment time by **80%**
- Added **JWT** and **OAuth 2.0** login and **Stripe** subscription billing, enabling the company to onboard its first paying customers
- Set up automated testing with **Pytest**, wired into the CI pipeline so tests run before anything ships
- Delivered the **React** and **TypeScript** dashboard where clients read those insights and track their own metrics

Backend Engineer (Freelance) | **Boardd** – Enterprise Business Platform | Remote | May 2025 - Present

- Built the backend for a business-management platform in **Node.js** and **Express** on **MongoDB**, growing it to over **200 REST endpoints** covering projects, team collaboration, and client billing

- Handled the money side with **Stripe Connect** so the platform could pay several parties at once, including Treasury accounts, virtual card issuing, recurring invoices, and onboarding for connected accounts
- Made the app collaborative in real time over **Socket.IO**, with live task editing, Kanban boards, drag-and-drop backed by optimistic locking, and presence indicators showing who else is in a project
- Secured it with **JWT** auth, rotating refresh tokens, **Redis**-backed sessions, and a role system with **40+ permissions**
- Connected messaging and storage services: **Twilio** SMS, **SendGrid** email, **Firebase** push, and file sync across **AWS S3**, Google Drive, Dropbox, and OneDrive

Full-Stack Software Engineer | **WorkSpin** | Karachi, Pakistan | Jul 2023 - May 2024

- Developed the backend for an event-discovery app in **Node.js**, using **Socket.IO** for live updates and adding **OAuth/JWT** login and **Stripe** checkout
- Cut query latency by **70%** and peak database load by **50%** by reshaping the **MongoDB** schemas and adding **Redis** caching. Also delivered a gym-management system with **AWS S3** media storage, live streaming, and **Firebase** alerts

PROJECTS

- **InsightQL – AI Database Assistant:** A **Next.js** and **NestJS** tool that lets non-technical users query a database by typing a question in plain English, using **LangChain's** SQL agent over **OpenAI GPT**. Gets people an answer roughly **3x faster** than writing the SQL themselves
- **bugSage – AI Debugging Assistant:** A **FastAPI** chatbot that pulls relevant docs and past issues out of a **Pinecone** vector database (**RAG**) before answering, so its fixes for Express.js bugs match the code you are actually running
- **CLI Assistant – Agentic Terminal Tool:** A **Python** assistant that runs entirely offline on a local model via **Ollama**, with an agentic loop that chains five tools together through **function calling**

SKILLS

Languages: Python, TypeScript, JavaScript, SQL, Java

AI & LLM: OpenAI API, OpenAI Agents SDK, LangChain, AI Agents, Tool/Function Calling, RAG, Vector Databases (pgvector, Pinecone), Prompt Engineering, PyTorch

Backend: Node.js, NestJS, Express, FastAPI, REST APIs, GraphQL, WebSockets (Socket.IO), Prisma, TypeORM

Frontend: React, Next.js, Redux, Tailwind CSS, HTML/CSS

Databases: PostgreSQL, MongoDB, Redis

Cloud & DevOps: AWS (ECS Fargate, Aurora/RDS, S3, Secrets Manager), Docker, Terraform, Kubernetes, GitLab CI/CD, GitHub Actions, Git

Practices: Agile/Scrum, Code Review, Unit Testing (Jest, Pytest), CI/CD, Microservices

PUBLICATIONS

Deep Learning for User Mobility Prediction in RIS-Assisted 6G THz Networks – IEEE

Benchmarked deep learning models for predicting user movement in next-generation (6G) mobile networks, to keep connections stable as users move.