

A.S Mustaffa

Sr. Python Full Stack Developer

asif.techdev@gmail.com | 203-534-9907

PROFESSIONAL SUMMARY

- **12+ years of hands-on Python development experience** building production-grade web applications, RESTful APIs, microservices, and data pipelines using **Python (FastAPI, Flask, Django)**, with strong frontend skills in **React, Next.js, Angular, and TypeScript**.
- Currently leading **RAG (Retrieval-Augmented Generation) platform development and AI engineering** — de- signing end-to-end RAG pipelines with document ingestion, chunking, embedding generation, vector storage (FAISS, pgvector), semantic retrieval, re-ranking, and LLM-powered response synthesis for enterprise knowledge systems.
- Proven expertise building **full-stack Python applications end-to-end** — from database schema design and backend API development to React/Next.js frontend interfaces, authentication layers, and deployment automation.
- Production experience with **Large Language Models (LLMs)** including OpenAI GPT-4/GPT-4o, Azure OpenAI, and Anthropic Claude — implementing prompt engineering, fine-tuning, multi-turn conversation memory, agentic workflows, and response validation pipelines.
- Built and deployed multiple **AI-powered Python services from scratch**: real-time speech-to-text (OpenAI Whisper, AWS Transcribe), multilingual translation (M2M100, Azure Translator), AI video summarization (GPT-4 + FFmpeg), and automated code review bots with RAG.
- Deep expertise in **Python ecosystem**: asyncio, Pydantic, SQL Alchemy, Celery, pytest, Alembic, Poetry, and standard data science libraries (Pandas, NumPy, Scikit-learn, PyTorch, TensorFlow).
- Experienced in **cloud-native Python deployments** on AWS (Lambda, EC2, S3, Bedrock, SageMaker) and Azure (OpenAI, Speech Services, App Services), with Docker/Kubernetes containerization and CI/CD automation.
- Skilled in **database design and optimization** across PostgreSQL, MySQL, MongoDB, Snowflake, Redis, and DynamoDB — writing complex queries, stored procedures, and building Python-based ETL/data pipelines.
- Strong **NLP/ML engineering** background: text generation, summarization, semantic search, named entity recognition, sentiment analysis, intent classification, and multi-modal AI processing (text + image + audio).
- Background in **investment banking and capital markets** with knowledge of Fixed Income products, building secure, compliant, high-performance Python applications for regulated financial environments.
- Known for **technical leadership** — mentoring engineers on Python best practices, AI architectures, and RAG design patterns while driving Agile execution and delivering measurable business impact.

TECHNICAL SKILLS

Python and Frameworks	Python 3.x, FastAPI, Flask, Django, Celery, Pydantic, SQLAlchemy, Alembic, pytest, Poetry, pip
AI/ML and GenAI	OpenAI GPT-4/GPT-4o, Azure OpenAI, Anthropic Claude, LangChain, LlamaIndex, RAG Pipelines, Prompt Engineering, Fine-Tuning, Agentic AI Workflows, Hugging Face Transformers, OpenAI Whisper (Faster-Whisper), Facebook M2M100
ML and Data Science	PyTorch, TensorFlow, Scikit-learn, Keras, Pandas, NumPy, Matplotlib, NLP, NER, Semantic Search
Vector Databases	FAISS, pgvector, Pinecone, ChromaDB, Embeddings (Bedrock Titan, OpenAI Ada-002)

Frontend	React, Next.js, Angular, TypeScript, JavaScript, HTML5, CSS3, Bootstrap, Tailwind CSS, jQuery
Databases	PostgreSQL, MySQL, MongoDB, Redis, Snowflake, DynamoDB, Cassandra, Oracle, SQL Server, SQLite
Cloud Platforms	AWS (EC2, S3, Lambda, Bedrock, SageMaker, DynamoDB, RDS, EKS, CloudFormation, Kinesis, SNS), Azure (OpenAI, Speech Services, Translator, AD, App Services, Blob Storage, Key Vault, Functions), GCP (AutoML, Vision AI, Apigee)
DevOps and MLOps	Docker, Kubernetes, Jenkins, GitHub Actions, Harness, Bamboo, Git, Terraform, CI/CD Pipelines, Model Versioning, A/B Testing
Data and Messaging	Apache Kafka, RabbitMQ, IBM MQ, Apache Spark, Hadoop, Hive
Backend and APIs	REST, GraphQL, gRPC, WebSocket, SOAP, Microservices, OAuth2/JWT, RBAC
Observability	Splunk, Grafana, CloudWatch, ELK Stack

PROFESSIONAL EXPERIENCE

Sr. Python Full Stack Developer and AI Engineer **GE Vernova, Boston, MA**

Apr 2023 – Present

Leading Python full-stack development and AI/RAG platform engineering — designing, building, and deploying production microservices, REST APIs, and Generative AI features including enterprise RAG pipelines and agentic LLM workflows.

- Led end-to-end **Python full-stack development** using FastAPI, Flask, and Django for backend services with React and Next.js (TypeScript) frontends — building RESTful APIs and microservices for energy analytics, trading workflows, and compliance platforms.
- Designed and delivered **secure Python backend services** for capital markets and trading analytics, with strong focus on performance, reliability, and regulatory compliance.
- Built **high-throughput Python REST APIs** and internal service integrations for financial platforms, improving inter-service communication and supporting low-latency business operations.
- Developed **Python ETL pipelines and data processing workflows** using Pandas, SQL Alchemy, and Celery for batch data ingestion, transformation, and loading into PostgreSQL and Snowflake.
- Implemented **OAuth/JWT-based authentication with Azure AD**, RBAC, and entitlement checks across Python API and React UI layers for regulated data access.
- Built **Python-based audit logging and observability** — centralized request tracing, access event capture, and Splunk dashboards for compliance reviews and production monitoring.
- Worked closely with Snowflake DB teams to **design schemas, optimize complex SQL queries**, and load large datasets for downstream reporting and analytics.
- Implemented **CI/CD pipelines** using Jenkins, GitHub Actions, and Harness to automate Python build, test (pytest), and deployment across Dev, QA, UAT, and Production.
- Mentored engineers on **Python best practices**, API design patterns, AI architectures.

RAG Platform and AI Engineering (Key Initiative):

- Architected and built a **production-grade RAG (Retrieval-Augmented Generation) platform in Python** for internal knowledge systems — enabling engineers to query technical docs, runbooks, and safety manuals using natural language.
- Designed the full **Python ingestion pipeline**: document parsing (PDF, DOCX, HTML) with PyPDF2/docx, intelligent chunking with overlap, embedding generation via Azure OpenAI Ada-002, and vector storage in pgvector (PostgreSQL) with metadata filtering.

- Implemented **semantic retrieval with re-ranking in Python** — combining dense vector similarity search with cross-encoder re-ranking to improve answer relevance by 35% over baseline retrieval.
- Built a **multi-turn conversational interface** with Python (FastAPI backend + React frontend) and conversation memory management using LangChain, enabling context-aware follow-up questions across sessions.
- Implemented **prompt engineering and guardrails** — structured system prompts with citation requirements, hallucination detection, and confidence scoring to ensure factual, traceable responses grounded in source documents.
- Developed **Python-based evaluation and observability framework** — tracking retrieval precision, answer faithfulness, latency, and token usage with Splunk dashboards for continuous model improvement.
- Built **agentic AI workflows in Python** — multi-step tool-calling agents for automated diagnostics, report generation, and data pipeline orchestration using LangChain and custom agent frameworks.
- Deployed on Azure with **Docker/Kubernetes containerized Python microservices**, Azure OpenAI for LLM inference, and CI/CD via GitHub Actions and Harness.

***Environment:** Python 3.11, FastAPI, Flask, Django, React, Next.js, TypeScript, Azure OpenAI (GPT-4, Ada-002), LangChain, pgvector, FAISS, Docker, Kubernetes, PostgreSQL, Snowflake, MongoDB, Redis, Celery, SQLAlchemy, Pandas, Splunk, Grafana, Jenkins, GitHub Actions, Harness, Azure AD, AWS (EC2, S3, Lambda)*

Sr. Python Full Stack Developer
Lululemon, Seattle, WA

Sep 2020 – Mar 2023

Led Python full-stack development and AI service integration for Lululemon's enterprise Digital Asset Management (DAM) platform — building Python-powered media processing services, automated publishing workflows, and intelligent asset discovery tools for global marketing and creative teams.

Python Full-Stack and Platform Development:

- Led **full-stack Python development** of the Nuxeo-based DAM platform using FastAPI and Flask for backend services with React/Next.js frontend interfaces, serving global marketing, creative, and e-commerce teams.
- Built **Python RESTful APIs and microservices** for asset ingestion, metadata enrichment, workflow automation, and bulk operations across the DAM ecosystem.
- Developed **Python ETL pipelines** for ingesting product and campaign data from upstream systems, transforming and mapping metadata, and loading enriched records into the DAM and Snowflake for analytics.
- Implemented **secure Python authentication layers** with OAuth2/JWT and role-based access controls across all DAM APIs and React UI layers.
- Deployed **Python services on AWS** (EC2, S3, Lambda) with Docker containerization and CI/CD automation via Jenkins and GitHub Actions.
- Created **Splunk dashboards** to monitor Python service health, asset processing pipelines, API performance, and publishing workflow completion rates.
- Developed a **Python-based automated publishing service** that pushes approved DAM assets to Adobe Scene 7 (Dynamic Media) for web delivery and CDN distribution.
- Built Python REST API integrations with Scene 7's upload/publish APIs — handling metadata mapping, rendition selection, and publish status tracking.
- Automated the full workflow: asset approval, metadata extraction, rendition selection, Scene 7 upload, and publish confirmation with callback. **Reduced manual publishing effort by 80%.**

***Environment:** Python 3.x, FastAPI, Flask, React, Next.js, TypeScript, Nuxeo Platform, Azure OpenAI (GPT-4), Azure Speech Services, Azure Translator, AWS Transcribe, AWS Translate, AWS Rekognition, OpenAI Whisper (Faster-Whisper), Adobe Scene 7 API, FFmpeg, FAISS, Docker, AWS (EC2, S3, Lambda), Snowflake, MongoDB, PostgreSQL, Splunk, Jenkins, GitHub Actions, Git*

Sr. Python Developer
Huntington Bank, Richmond, VA

Jun 2016 – Aug 2020

Designed, developed, and deployed **mission-critical Python applications** for fixed income trading, risk management, and compliance platforms in a regulated financial environment.

- Developed **Python backend services and RESTful APIs** using Flask and FastAPI for fixed income trading and risk platforms, with Node.js (Express.js) for supplementary services.
- Built and optimized **complex SQL queries, stored procedures, and indexing strategies** on MSSQL to reduce latency for real-time data access in high-frequency trading systems.
- Designed and implemented **Python NLP-powered data extraction pipelines** — parsing and classifying unstructured financial documents (trade confirmations, regulatory filings) for automated processing.
- Built **Python ETL pipelines** to ingest trade and reconciliation data from APIs and flat files, validate/transform records using Pandas, and load curated data into Snowflake for analytics and reporting.
- Integrated **IBM MQ and RabbitMQ with Python services** for reliable, asynchronous, event-driven communication between distributed trading applications.
- Deployed **Python applications on AWS and Azure** leveraging EC2, S3, Azure Blob Storage, Key Vault, and API Gateway for secure, scalable deployments.
- Implemented **secure Python authentication and authorization** using JWT, OAuth2, and centralized API management via Apigee.
- Collaborated with DevOps to implement **CI/CD pipelines for Python services** using Jenkins, GitHub Actions, Docker, and Bamboo across multiple release environments.
- Used Splunk to monitor batch job logs and API failures; created Grafana dashboards for application health, API latency, and batch job trends.
- **Optimized Snowflake queries** for reporting teams experiencing slow performance, working closely with DB teams on schema design.

Environment: *Python 3.x, FastAPI, Flask, Express.js, Node.js, MSSQL, Snowflake, MongoDB, Pandas, AWS (EC2, S3), Azure (Blob Storage, Key Vault), IBM MQ, RabbitMQ, Docker, Jenkins, GitHub Actions*

Python Developer
SunTech Technologies, India

Jul 2013 – Feb 2015

Contributed to **Python backend development** and data processing automation using Python and Django in a collaborative team environment.

- Built backend services using **Python and Django framework**, supporting frontend development with HTML, CSS, JavaScript, and AJAX.
- Designed **Python RESTful APIs** and deployed them on FastHTTP server within AWS infrastructure.
- Implemented interactive UI features using **PyQt** for data filtering and visualization.
- Wrote **Python scripts** for task automation, data processing, XML parsing, and database loading.
- Worked with **MySQL and PostgreSQL databases** and supported UI component integration with jQuery and AJAX.
- Gained hands-on experience with Git for version control and participated in Agile practices using JIRA.

Environment: *Python, Django, JavaScript, jQuery, AJAX, MySQL, PostgreSQL, HTML, CSS, Linux, Git, Oracle DB, Heroku*

AI PROJECTS AND SERVICES BUILT

Whisper Transcription Service

Python, FastAPI, OpenAI Faster-Whisper (large-v3)

Built a **high-accuracy Python speech-to-text API** using Faster-Whisper for automated media content

processing.

- Processes audio/video and generates accurate transcriptions, making media searchable and accessible.
- Runs fully on-premises — no data leaves the organization, ensuring data privacy compliance.
- Reduced transcription costs by eliminating third-party cloud service dependency.

AI Video Highlight Generator

Python, FastAPI, Azure OpenAI GPT-4, Azure Speech, Azure Translator, FFmpeg, Next.js

Built a **Python end-to-end AI video processing pipeline** that creates highlight reels from long-form video with multilingual subtitles.

- Pipeline: Audio extraction, transcription, GPT-4 highlight selection, multilingual subtitles, FFmpeg video assembly.
- Full-stack app: Next.js frontend with Python FastAPI backend and Celery background job processing.
- Deployed on AWS EC2 with Docker Compose and Caddy reverse proxy.

Real-Time Audio Live Translation

Python, FastAPI, Azure Speech Services, Azure Translator, Next.js, WebSocket

Built a **Python real-time multilingual communication platform** that transcribes and translates live speech.

- Python WebSocket backend integrating Azure Speech Services with Azure Translator for instant cross-language translation.
- Eliminates human interpreters in meetings, training sessions, and live events.