**Meerasa Shaik**

Wilmington, DE | +1(302)-568-9588 | [meeruchp088@gmail.com](mailto:meeruchp088@gmail.com)

**PROFESSIONAL SUMMARY**

Results-driven Python Developer with 5 years of experience building secure, high-performance backend applications using Django, Flask, and Fast API. Adept at designing RESTful APIs, integrating third-party services, and leveraging microservices architecture for maintainability and performance. Experienced in event-driven systems using Kafka, RabbitMQ, and AWS messaging services to support real-time, decoupled communication. Proficient in cloud-native development on AWS, including Lambda, DynamoDB, Cognito, and Fargate, with a focus on serverless design and automation. Demonstrated expertise in containerization with Docker and orchestration with Kubernetes, supported by robust CI/CD pipelines.

I recently built a **Personal AI Assistant on my local laptop**, capable of reading unread Gmail emails securely, summarizing them, answering deep technical and general questions, processing multiple document types for instant Q&A, and fetching real-time weather reports — showcasing expertise in Generative AI, Lang Chain, Streamlit, and OpenAI APIs. Reduced infrastructure costs by 30% through serverless migration, and decreased deployment time by 50% via CI/CD automation. Solid background in working with relational and NoSQL databases, including PostgreSQL, MySQL, MongoDB, and Redshift.

**SKILLS**

* **Programming Languages:** Python (Advanced), Java, Go and Bash
* **Python Frameworks:** Django, Flask, Fast API, Django REST Framework, Celery, Socket.IO, Uvicorn, Gunicorn
* **AI & Automation:** OpenAI GPT APIs, Lang Chain, Prompt Engineering, Streamlit, Gmail API, PyPDF2, python-docx
* **Data Analysis and Visualization**: Pandas, NumPy, Matplotlib, Seaborn, Plotly
* **Front End Technologies:** Html, CSS, JavaScript, TypeScript and ReactJS
* **Cloud Platforms**: AWS, Azure, GCP
* **Containerization & Orchestration**: Docker, Kubernetes, OpenShift, Rancher, EKS
* **CI/CD Tools**: Jenkins, GitHub Actions, GitLab CI, Bitbucket Pipelines, GitOps
* **Source Control & Build Tools:** Git, GitHub, GitLab, Bitbucket, Maven, ANT, Gradle, Bamboo
* **Infrastructure as Code & Configuration Management**: Terraform, Ansible, Chef, Puppet
* **Messaging:** Kafka, RabbitMQ
* **Monitoring & Logging Tools**: CloudWatch, Datadog, Grafana, Prometheus, Kibana, Splunk, ELK Stack, Nagios
* **Databases**: SQL, MySQL, PostgreSQL, Oracle, Amazon RDS, MongoDB, Redis, DynamoDB
* **Operating Systems & Virtualization:** Linux (RHEL, Ubuntu), Windows, VMware, vSphere, vCenter
* **Issue & Project Tracking Tools**: Jira, ServiceNow, Redmine, Confluence
* **Development Methodologies:** Agile (Scrum), SDL
* **Python IDEs & Editors:** PyCharm, PyScripter, Spyder, PyDev, Sublime Text, Visual Code
* **Application & Web Servers**: Nginx, Apache Tomcat, WebLogic, JBoss

**PROFESSIONAL EXPERIENCE**

**TD Bank June 2024 - Present**

**Senior Python Developer** Mount Laurel**,** NJ

* Developed and maintained scalable RESTful APIs using Django REST Framework and Fast API, enabling seamless integration between front-end applications and backend microservices with optimized response times.
* Developed custom middleware in FastAPI for centralized auditlogging, globalexception handling, and correlation ID tracking, seamlessly integrated with AWS CloudWatch for end-to-end visibility.
* Enabled automatic Swagger (OpenAPI) documentation for all API endpoints using Fast Api’s native support, facilitating developer onboarding and reducing external integration turnaround time.
* Developed and consumed secure RESTful APIs with OAuth2 and JWT-based authentication, integrating with third-party services and internal microservices to support modular architecture
* Integrated Redis for caching API responses and managing sessions, significantly improving API performance and reducing database read load by 60%.
* Designed and implemented robustdatabase schemas in PostgreSQLandMySQL**,** ensuring normalized data structures, efficient indexing, and high-performing query operations in production environments.
* Utilized popular Pythonlibraries such as Pandas, NumPy, and SQL Alchemy for data processing, ORM integration, and pipeline development in analytics-heavy applications.
* Integrated Apache Kafka into microservices architecture to enable high-throughput, fault-tolerant event streaming between decoupled systems, improving real-time data pipeline performance and scalability.
* Built serverless applications on AWS using AWS Lambda, API Gateway, and DynamoDB, reducing infrastructure overhead and improving response scalability by 40%.
* Integrated RabbitMQ into CI/CD pipelines for triggering deployment workflows and notifications, using durable queues and message acknowledgments to ensure fault tolerance.
* Implemented automated CI/CD pipelines using Jenkins, GitHub Actions and AWS Code Pipeline to streamline testing and deployment workflows, reducing release cycle times by 50%.
* Integrated Docker into CI/CD pipelines to automate the build, test, and deployment of containerized Python applications, significantly reducing deployment time and manual effort.
* Wrote and optimized Docker files for Python-based microservices, reducing image size and build time while enforcing best practices in container security and layer caching.
* Deployed and managed containerized Python applications on Kubernetes clusters using Helm charts and YAML manifests, ensuring high availability, scalability, and load balancing.
* Integrated Kubernetes with monitoring tools such as Prometheus and Grafana to track cluster health, application performance, and container metrics in real time.
* Actively participated in Agile development processes including daily stand-ups, sprint planning, and retrospectives while using Jira and Confluence for task tracking and documentation.

**Tools & Technologies**: Python, Flask, Django, Fast API, Pandas, NumPy, MySQL, PostgreSQL, AWS, Jenkins, GitHub, GIT, Docker, Kubernetes, Grafana, Prometheus.

**HCL Technologies March 2021 – Dec 2022**

**Python Developer** Pune India

* Led the design and development of a real-time data processing platform using Python**,** Fast API, and Kafka, enabling live customer behaviour tracking across 10M+ users with sub-second latency.
* Developed scalable RESTfulAPIs using Fast API and integrated OAuth2 authentication with JWT for secure, token-based access across mobile and web platforms.
* Designed and implemented RESTfulAPIs using FastAPI in Python to handle real-time financial transactions with comprehensive validation, structuredlogging**,** and modulardatapersistencelayers**.**
* Architected and developed complex ETL pipelines using Python (Pandas, NumPy) to clean, enrich, and aggregate transactional data from diverse sources into PostgreSQL and Amazon Redshift for analytics
* Built and maintained Kafka consumers and producers using confluent-Kafka to support streaming event ingestion and downstream microservices for fraud detection and recommendation engines.
* Integrated AWS S3 for cloud storage of user-generated content and large data exports, using boto3 for efficient upload/download workflows in Python.
* Configured CI/CD pipelines with GitHub Actions and AWS Code Pipeline to automate testing, Docker image builds, and blue-green deployments to EKS clusters.
* Implemented Docker containers for all microservices and configured DockerCompose for seamless local development and automated testing environments.
* Deployed services on Kubernetes (EKS**)** using Helm charts, managing auto-scaling, rolling updates, and resource quotas to maintain high availability under heavy traffic.
* Designed databaseschemas in PostgreSQL and optimized SQL queries and indexes, reducing query time by 60% for high-volume API endpoints.
* Used Redis as a caching layer for frequently accessed API responses, improving response time by over 70% and reducing load on primary databases.

**Tools & Technologies**: Python, Fast API, Kafka, Pandas, NumPy, SQL, PostgreSQL, AWS, CI/CD, Docker, Kubernetes

**Briq Sep 2019 – Mar 2021 Python Developer** Hyderabad, India

* Designed and implemented a RESTful API using Django and Django REST Framework to manage product inventory, order processing, and supplier integration across multiple retail location.
* Created user-friendly endpoints with CRUD operations, input validation, and pagination to support both internal staff tools and mobile applications for warehouse managers.
* Built backend logic to handle order tracking, stock alerts, and automatic restocking using Python scripting, ensuring efficient and accurate inventory control.
* Developed relational database schemas in MySQL, implemented stored procedures, and wrote optimized SQL queries for reporting and data consistency.
* Automated data ingestion from CSV and Excel files using Pandas, cleaning and transforming product data from vendors into a standardized format for system entry.
* Used Git for version control and managed source code with GitHub, following Git Flow for collaborative development with other team members.

**PROJECTS**

**Personal AI Assistant | Python, OpenAI GPT, Lang Chain, Streamlit, Gmail API**

* Built a local AI assistant to securely log into Gmail, read unread emails from real people, and summarize them, ignoring automated notifications.
* Integrated Generative AI to answer coding, technical, and general questions with context-rich outputs.
* Enabled document QA by processing multiple PDFs and DOCX files for instant question answering using retrieval-augmented generation.
* Designed an interactive Streamlit UI to support multi-functional AI workflows, including real-time weather queries.
* **Technologies Used:** Python, OpenAI GPT APIs, Lang Chain, Streamlit, PyPDF2, python-docx, Gmail API.

**Upload Multiple PDFs Chatbot | Python, OpenAI GPT, Lang Chain, FAISS, Streamlit**

* Developed an AI-powered chatbot that processes multiple PDF documents simultaneously, enabling users to ask natural language questions and receive accurate, context-aware answers instantly.
* Implemented retrieval-augmented generation (RAG) by chunking documents, creating vector embeddings with Lang Chain, and performing similarity search using FAISS.
* Designed a user-friendly Streamlit interface for seamless PDF uploads, chat-based Q&A, and real-time answer rendering.
* Enhanced productivity for document-heavy workflows by transforming static PDFs into interactive knowledge bases accessible via chat.
* **Technologies Used:** Python, OpenAI GPT APIs, Lang Chain, FAISS, PyPDF2, Streamlit.

**CERTIFICATION**

* AWS Certified Cloud Practitioner CLF-C02
* Academy Accreditation: Generative AI Fundamentals (Databricks)
* Introduction to Generative AI(Google)
* Generative AI Masterminds INTL (Out skill)

**EDUCATION**

**Wilmington University Apr 2024**

Master of Science in Information Technology Wilmington, DE