HEMANTH KOPPAKA
SR. SOFTWARE ENGINEER
LinkedIn | +1(978)237-4932 | hemanthdpk@gmail.com \

PROFESSIONAL SUMMARY

- **Senior Software Engineer** with around **8 years** of experience delivering enterprise-scale solutions across **healthcare, finance, and credit risk domains**, driving modernization, compliance, and business impact.
- Experienced in **AI Engineering** of delivering enterprise-grade Generative AI, and data engineering solutions across healthcare, finance, and compliance domains, specializing in large-scale LLM, RAG, and ETL ecosystems on Azure and AWS.
- Designed and owned end-to-end data pipelines powering backend applications and AI services, building scalable ingestion and transformation workflows using **PySpark, Databricks, Kafka, and SQL** to process large healthcare and financial datasets, storing curated outputs in **Delta Lake, PostgreSQL, and S3**, and exposing them through **FastAPI/Django REST APIs** for real-time analytics, ML inference, and operational dashboards across cloud platforms (**Azure and AWS**).
- Hands-on with **LangChain, LangGraph, Azure OpenAI, Bedrock, and Hugging Face Transformers**, building multi-agent workflows, retrieval-augmented generation (RAG) pipelines, and governance frameworks ensuring explainability, consistency, and regulatory trust.
- Specialized in **Python full-stack development** with **FastAPI, Django REST Framework, Flask, Celery, Pandas, NumPy,** and **Angular, React.js (Redux, Recharts, Tailwind CSS), Next.js** for interactive, real-time applications.
- Hands-on with **databases (PostgreSQL, MySQL, MongoDB, Oracle, Oracle SQL, DynamoDB, Cloud SQL, Azure SQL)**, **caching (Redis, ElastiCache)**, and **event-driven streaming (RabbitMQ, Kafka, IBM MQ, AWS Kinesis, Azure Service Bus)** to power high-throughput systems.
- Strong background in multi-cloud architecture: **AWS** (EKS, RDS, API Gateway, QuickSight, OpenSearch), **Azure** (AKS, SQL, Service Bus, Power BI Embedded, Cognitive Search), **GCP** (GKE, Cloud SQL, Cloud Run, Cloud Build, Elastic Cloud/Elasticsearch) with **IaC** (Terraform, ARM, CloudFormation).
- Extensive experience in **CI/CD automation** with **Jenkins, Harness, Azure DevOps, GitHub Actions, AWS CodePipeline**, implementing blue/green & canary deployments with integrated security scans.
- Skilled in **security and observability** with **OAuth2, JWT, RBAC, HIPAA/financial compliance**, and enterprise monitoring stacks (**CloudWatch, Azure Monitor, GCP Logging, Grafana, Prometheus, Sentry, Langfuse**) ensuring performance, traceability, and resilience.

TECHNICAL SKILLS

Programming Languages	SQL, PL/SQL, Python 3.11, OOP, Java, Node.js
APIs & Web Services	REST APIs (Django REST, Flask), GraphQL (FastAPI), AWS Lambda APIs, Google Cloud Functions
Databases & Warehouses	PostgreSQL, MySQL, MongoDB, DynamoDB, Oracle DB, Amazon Redshift, Snowflake, Google BigQuery
AI Models / Frameworks	Open AI (GPT-3, GPT-4), Langchain, LangGraph, CrewAI, SageMaker, TensorFlow, Scikit-Learn, Pytorch
Version Control & Source Code Management	Git, GitHub, Bitbucket, AZURE REPOS
Python Libraries	NumPy, Pandas, matplotlib, Boto3, Pyspark
ETL / Data Pipelines	AWS Glue, Azure Data Factory, PySpark on AWS EMR, Azure Databricks, GCP Dataflow, Kafka (AWS MSK), Event Hubs, Lambda Triggers, Cloud Functions
Config management tools	Chef, Puppet, Ansible.
Security & Authentication	Okta SSO, AWS Cognito, JWT, RBAC, IAM Policies, HIPAA Compliance Implementation
IaC & Containerization / Orchestration	Docker, Kubernetes (AWS EKS, Azure AKS, GCP GKE), Terraform, AWS CloudFormation, Azure Resource Manager (ARM), GCP Deployment Manager

CI/CD & Build Tools	Jenkins, GitHub Actions, Harness, Gitlab, Azure Devops, Ant, Gradle, Maven, MS build
Cloud Platforms	AWS, Azure, Google Cloud Platform
Monitoring & Logging	AWS CloudWatch, Splunk, DataDog
Operating systems	Windows, Linux, Unix, MacOS
Methodologies	Agile, Scrum, Waterfall, Kanban
Tools & IDE	Visual Studio, Eclipse, PyTest, PyCharm, VIM editor, Cyme application modules

---

Professional Experience

Client: Cigna- Evernorth, Hartford, CT **July 2024 - Present**

Senior Software Engineer - AI

Responsibilities:

- Built **LangGraph-based multi-agent workflows** on Azure OpenAI and Azure AI Foundry to automate eligibility, claims validation, and clinical insight generation, streamlining provider and member workflows under HIPAA compliance.
- Developed an interactive **React.js dashboard** to visualize AI responses, retrieved context providing clinicians transparent and auditable decision insights.
- Implemented **FastAPI-based, async Python microservices** for eligibility verification, claim status, validation, provider search, and RAG orchestration, keeping clean domain boundaries and consistently low-latency API performance under load.
- Built an automated document summarization system using **Azure Form Recognizer and GPT-4 to parse forms**, reducing manual review time and saving over 1200 analyst hours annually.
- Migrated legacy chatbot systems to **GPT-4 and Pinecone-powered RAG architecture**, improving contextual awareness and retention in user interactions with measurable uplift in CSAT metrics.
- Integrated GPT-4 models into the enterprise chatbot for high-volume client queries, incorporating fallback logic using semantic vector search and dynamic intent-based routing for escalation.
- Implemented **Pydantic-based structured output validation** for LLM responses, enforcing strict JSON schemas for claims, eligibility, and clinical insight workflows to improve response reliability and downstream system compatibility.
- Managed API secrets and access control with Azure Key Vault and implemented role-based **IAM policies across GenAI deployments** to ensure enterprise compliance and credential safety.
- Constructed evaluation framework using BLEU, ROUGE, and human feedback to monitor prompt output quality and optimize LLM completions in production through continuous learning.
- Built Databricks ETL pipelines (Delta Lake , Spark) for claims and pharmacy feeds using Jobs/Workflows and Unity Catalog governance to support HIPAA-grade lineage, access control, and reproducible data products.
- Engineered Python and **PySpark ETL workflows** across **PostgreSQL, MSSQL, and ADLS Gen2**, integrating Kafka and Event Hubs streams to increase ingestion throughput by approximately 40%.
- Designed hybrid RAG pipelines using **LangChain, Azure Cognitive Search, FAISS, and Pinecone**, unifying retrieval across EOB, FHIR/HL7, PDFs, and SQL sources to improve contextual accuracy by 29%.
- Implemented adaptive chunking and dynamic **Top-K retrieval for RAG pipelines**, tightening context precision and minimizing GPT-4 hallucinations.
- Implemented **Model Context Protocol (MCP)** servers and **AI Foundry tool registries** to expose secure, schema-validated tools to **LangGraph agents**, enforcing strict access controls and preventing prompt-level data leakage.
- Implemented Redis-based caching and session management for high-frequency inference requests, authentication tokens, and metadata lookups, reducing API latency and improving throughput.
- Containerized microservices using **Docker** and deployed via Helm to **Azure Kubernetes Service (AKS)** with HPA autoscaling and blue-green rollout strategies for zero-downtime updates.

- Designed and governed API layers using **Azure API Management and API Gateway patterns**, implementing JWT authentication, rate limiting, routing, and versioning.
- Adopted event-driven architectures using **Azure Event Hubs and Service Bus** to enable resilient, asynchronous communication between GenAI agents and ETL pipelines.
- Automated CI/CD pipelines with **GitHub Actions and Azure DevOps,** embedding static analysis, continuous testing, compliance validation, Trivy scans, SBOM signing, and gated Helm releases.
- Implemented observability and monitoring using **Azure Monitor, App Insights, and Langfuse** to track latency, token cost, hallucination metrics, and API reliability.
- Built a React-based compliance and audit console to review prompts, retrieved context, tokens, and outputs, supporting HIPAA and HITRUST evidence for AI traceability.
- Delivered measurable business impact including 41% improvement in eligibility throughput, 33% reduction in claim backlog, 29% increase in contextual accuracy, and 45% reduction in manual chart-review time using Form Recognizer and GPT-4 and AI Foundry-managed model deployments**.**

Client: Hearst, New York, NY ***October 2023 – July 2024***
Senior Software Engineer
Responsibilities:

- Developed Django REST Framework (DRF) APIs to expose ratings, issuers, outlook changes, and portfolio metrics with strong serializer validation, consistent error contracts, and predictable pagination/filtering for large datasets.
- Designed and built Python FastAPI microservices to expose credit-risk model inference as low-latency REST endpoints, supporting real-time scoring, risk flagging, and model-driven decision workflows for analyst-facing applications.
- Developed responsive, component-based frontend modules using React.js and Vue.js to deliver seamless user experiences across financial dashboards and healthcare data portals, ensuring real-time interactivity and cross-browser compatibility.
- Migrated legacy ML pipeline execution into Databricks, integrating Oracle Autonomous Data Warehouse as a core source and refactoring jobs for compatibility with Databricks ML APIs to improve runtime and maintainability.
- Developed several Spark/Scala scripts for data extraction from various sources and provided data insights and reports as per need.
- Worked on extracting real-time data using Spark Streaming and Kafka, converted it to RDDs, processed it into DataFrames, and loaded the data into HBase.
- Maintained and configured the data using Zookeeper and tracked the nodes in the Kafka cluster.
- Built backend data ingestion and normalization flows using Apache NiFi to integrate structured and semi-structured sources such as CRM extracts, scanned-form outputs, and call logs, landing curated datasets into Delta tables for analytics and ML feature consumption.
- Added compute-heavy simulation endpoints as async FastAPI services for stress tests such as rate shocks and spread moves using NumPy and Pandas, optimized to return results fast enough for interactive dashboard use.
- Moved legacy data prep into Databricks and tuned the jobs so feature tables were ready faster for FastAPI-based real-time scoring.
- Set up incremental loads and simple validation checks to catch broken upstream files early, so bad data did not flow into PostgreSQL dashboards or model inference.
- Built ETL pipelines to bring in CRM data, scanned-form outputs, and call logs, cleaned and standardized them, and stored them in Delta tables for consistent model inputs.
- Wrote and optimized complex SQL using CTEs, window functions, joins, and aggregates to validate rating and portfolio metrics, model inputs, and troubleshoot production data issues tied to APIs.
- Used PySpark in Databricks to transform large structured and semi-structured datasets into Delta tables, creating reliable feature datasets for ML pipelines and real-time scoring services.

- Implemented event-driven processing for rating updates using Amazon Kinesis Data Streams and Kinesis Data Analytics, publishing transformed events into PostgreSQL so the platform could surface near real-time changes without manual refresh cycles.
- Optimized PostgreSQL persistence for high-volume financial workloads by improving schema design, writing efficient ORM query patterns, and applying indexing and migration strategies that reduced slow queries during peak usage windows.
- Introduced Redis-based caching for reference and lookup-heavy datasets such as issuer metadata, sector mappings, and instrument universes, cutting repeated DB round-trips and stabilizing API latency under bursty traffic.
- Integrated Elasticsearch into backend search APIs to support fast issuer and instrument discovery, maintaining index freshness through controlled ingestion and consistent query semantics for filters and typeahead use cases.
- Secured backend services with Amazon Cognito (OIDC) and JWT-based authorization, enforcing role-based access paths for analysts, investors, and regulator-style views while maintaining auditability and least-privilege controls.
- Containerized Python services with multi-stage Docker builds and deployed to **Amazon EKS** through GitHub Actions pipelines, standardizing release workflows and enabling repeatable environment promotion from dev to test to prod.
- Provisioned and managed core platform infrastructure such as Amazon EKS, Amazon Kinesis, PostgreSQL, and Redis using reusable Terraform and CloudFormation modules, keeping environments consistent and reducing drift across deployments.
- Instrumented backend services with OpenTelemetry for traces, metrics, and logs, and integrated SLO-style alerting into the team's operational workflow, improving incident triage speed and making performance regressions visible early.

TransUnion, Stamford, CT **Sep 2019 – July 2023**

Software Engineer

Responsibilities:

- Built Python backend APIs using Django REST Framework and Flask to support borrower onboarding, eligibility checks, KYC steps, and partner integrations.
- Designed clean REST endpoints for multi-step onboarding flows (save progress, resume later, status tracking), keeping the workflow reliable for end users and operations teams.
- Developed dynamic and responsive Single Page Applications (SPAs) using Angular, improving user experience and application performance through efficient state management with NgRx and Redux.
- Integrated Angular applications with RESTful APIs and WebSockets, enabling real-time data updates and enhancing interactivity for high-performance web applications.
- Designed and developed responsive web interfaces using HTML5, CSS3, and Bootstrap, ensuring cross-browser compatibility and mobile-first design for optimal user experience.
- Implemented async background processing for document verification and bureau callbacks using Celery/RQ and message queues, so the UI wasn't blocked by slow external systems.
- Used RabbitMQ (DLQ, retry, TTL) for event-driven workflows like KYC events, bureau responses, and partner notifications, making the system stable under spikes.
- Processed onboarding and credit verification events through Apache Kafka topics, enabling reliable asynchronous communication between microservices handling KYC validation, credit bureau callbacks, and partner notifications.
- Modeled and managed data with SQLAlchemy ORM, applying migrations, constraints, and tuned queries to support high transaction throughput.
- Built and optimized transactional persistence using Aurora PostgreSQL and Oracle where needed for reporting, including schema tuning, indexing, and efficient ORM query patterns.
- Created ETL-style jobs in Python and PySpark to move, transform, and reconcile onboarding and credit application data across systems, implementing validation checks to detect missing, delayed, or duplicate records early.
- Worked on big data (Hadoop) environment along with exposure to HIVE, Spark, Cassandra, SQL and ETL components.

- Developed PySpark data processing pipelines to aggregate onboarding events and bureau responses stored in S3 and relational databases, supporting downstream analytics and compliance reporting.
- Stored document metadata and verification results in DynamoDB when low-latency lookups were needed, and used OpenSearch for fast search across case records.
- Improved API performance with caching, connection pooling, and careful payload shaping, keeping common onboarding endpoints consistently responsive.
- Secured APIs using AWS Cognito/OAuth2 and JWT scopes, and managed secrets with SSM/Secrets Manager using least-privilege IAM policies.
- Containerized Python services with Docker and deployed them on ECS Fargate enabling repeatable releases and easier scaling.
- Set up CI/CD pipelines using GitHub Actions and Jenkins to run unit tests, contract tests, build container images, scan dependencies, and deploy with safe rollback options.
- Utilized Python libraries such as Pandas for data processing, Boto3 for AWS service integrations, and PyTest for automated testing of backend services.
- Worked extensively with AWS Cloud Services including Lambda, S3, Step Functions, Glue, EC2, ECS/Fargate, RDS, Redshift, and CloudWatch to build scalable backend systems and data pipelines.
- Implemented serverless event-driven workflows using AWS Lambda and Step Functions to orchestrate document verification steps, external bureau calls, and KYC validation tasks.
- Built and maintained data ingestion pipelines using AWS Glue and S3, enabling structured analytics datasets used by internal credit risk and compliance teams.

Arete IT Services, India ***2018 May – July 2019***

Software Developer

Responsibilities:

- Designed and developed backend modules using Python (Django REST Framework) and Java (Spring Boot), deploying scalable microservices on AWS EC2, Elastic Beanstalk, and Kubernetes (EKS).
- Architected end-to-end backend solutions ensuring optimized data flow across AWS infrastructure, including RDS, S3, and API Gateway.
- Built RESTful APIs with secure authentication (JWT/OAuth2), API versioning, and efficient JSON structures for cloud-native access**.**
- Engineered real-time features such as order management, workflow execution, and streaming analytics using advanced algorithms, multithreading, and event-driven Java modules.
- Integrated backend logic with AWS RDS (MySQL/PostgreSQL), applying schema normalization, indexing, and stored procedure optimization for transactional and analytics-heavy workloads.
- Implemented CI/CD pipelines with Jenkins and AWS CodeDeploy, managing PyTest/UnitTest executions, Docker image builds, and zero-downtime deployments.
- Followed TDD practices, proactively monitoring test results in Jenkins and collaborating with QA for regression-free releases.
- Developed predictive models with PyTorch and applied optimization algorithms (SciPy linear programming) for logistics forecasting and cost minimization.
- Leveraged OOP principles, design patterns, and reusable utility tools to maintain clean, modular codebases across Python, Java microservices, and AWS Lambda.
- Collaborated with cross-functional teams across finance, logistics, and retail domains, conducting sprint demos, and aligning backend implementations with enterprise cloud standards to ensure scalability, reliability, and visibility.

EDUCATION

Master's in Computer Science from University Of Massachusetts, USA

CERTIFICATIONS

- ***Microsoft Certified: Azure AI Fundamentals***
- ***Microsoft Certified: Azure Fundamentals***
- ***Databricks Generative AI Fundamentals***
- ***AWS Certified Solutions Architect - Associate***
- ***Multicloud Network Associate***