Harika K

Senior Software Engineer

+1(940) 487 7844 | harika2506@outlook.com

PROFESSIONAL SUMMARY

Senior Software Engineer with over a decade of experience designing and scaling cloud-native platforms across insurance, finance, logistics, and e-commerce domains. Proven track record in building secure, high-throughput systems that power critical business operations through data-driven pipelines, modular microservices, and automated infrastructure.

Skilled in developing API-first backends using **Java (Spring Boot, WebFlux)** and **Python (FastAPI, Flask)**, secured with **GraphQL**, **OAuth2**, and **JWT** for robust service integration and governed access control. Delivered scalable ETL workflows and analytics platforms using **PySpark**, **Databricks**, **Delta Lake**, **Kafka**, **Airflow**, **AWS DMS**, and **Azure Data Factory**, supporting schema-evolving transformations, clause-level risk scoring, and audit-focused reporting.

Hands-on in deploying applications across **AWS** and **Azure** using **Docker**, **Kubernetes (EKS/AKS)**, **Terraform**, and **Vault** to enable repeatable, secure, and scalable deployments. Automated builds and releases through **GitHub Actions**, **GitLab**, and **Azure DevOps**, enforcing traceable, rollback-safe deployments across environments.

Designed systems with observability and compliance in mind, embedding **Prometheus**, **Grafana**, **OpenTelemetry**, and **ServiceNow** for real-time monitoring, alerting, and SLA visibility. Adept at turning complex business logic into production-grade services focused on traceability, resilience, and rapid iteration—whether supporting GenAI-driven clause workflows or risk-tiered ETL pipelines.

CERTIFICATIONS

- **Microsoft Certified: Azure AI Engineer Associate**
- **AWS Certified: Solutions Architect Professional**

SKILLS

GenAI & ML Engineering: OpenAI, Claude, Amazon Bedrock, Titan Embeddings, Hugging Face BART-MNLI, DistilBERT, NER, XGBoost, Random Forest, Isolation Forest, Keras, LangChain

ETL & Data Engineering: PySpark, Databricks, Delta Lake, Airflow, AWS DMS, Glue, Athena, Snowflake, ADF, SSIS, Sqoop, MapReduce, Hive, HiveQL, Ab Initio

Cloud & Infra: AWS (S3, DMS, Glue, Athena, MSK, EC2, IAM, CloudWatch), Azure (ADLS Gen2, ADF, Azure ML, Azure Monitor, Purview, App Services), Docker, Kubernetes (EKS, AKS), Terraform, Vault, GitHub Actions, Azure DevOps, GitLab

Databases & Storage: PostgreSQL, OracleDB, MongoDB, Redis, Amazon Redshift, Azure Data Explorer (Kusto), HDFS, Avro, Parquet

Backend Development: Java (Spring Boot, Spring WebFlux, Spring Security, CompletableFuture), Python (FastAPI, Flask), GraphQL, REST APIs, Node.js (CLI), JWT

Monitoring & Observability: Prometheus, Grafana, OpenTelemetry, Spring Actuator, Azure Monitor, ServiceNow, PagerDuty, Log4j

Reporting & Dashboards: Power BI, Tableau, Redash, SSRS

Version Control & Collaboration: Git, GitHub, GitLab, JIRA, Confluence

EXPERIENCE

Senior Software Engineer — Dec 2022 – Present

BCBS-M , Detroit, Michigan

Overview

An intelligent content processing platform built to extract structured signals from high-compliance documents and enable clause-level review automation. The system streamlined scoring, risk identification, and decision support by converting unstructured inputs into analyzable formats. It improved consistency, traceability, and turnaround times across legal and compliance workflows.

Responsibilities:

- Migrated 1M+ legacy contracts from **PostgreSQL** to **Amazon S3** using **AWS DMS** with Change Data Capture (CDC), ensuring near-real-time (<5 sec) data freshness.
- Streamed real-time document arrivals and scoring events with **AWS Kinesis**, enabling instant ingestion and dashboard updates.
- Orchestrated hybrid batch and event-driven pipelines using **Apache Airflow** and **AWS Step Functions** for clause extraction and scoring workflows.
- Processed and enriched nested JSON contract data in **PySpark**, combining signer metadata and contract context stored in **Parquet** on **S3**.
- Applied HIPAA-compliant masking and tokenization of sensitive fields via **PySpark**, securing PHI prior to analysis.
- Automated handling of schema drift using adaptive PySpark logic, minimizing ingestion downtime during document format changes.
- Implemented medallion architecture in **Delta Lake** on **S3**, organizing bronze, silver, and gold layers with version control and time travel.
- Leveraged **Delta Lake Change Data Feed (CDF)** to track incremental clause changes and enable efficient re-scoring pipelines.
- Registered structured datasets with the **AWS Glue Data Catalog** to facilitate schema discovery and querying via Athena.
- Developed modular **dbt** models for clause filtering, scoring logic, and reviewer transformation rules, iteratively refined through **Agile** sprints.
- Prototyped clause semantic similarity using **OpenAI GPT-4 embeddings** in Databricks; transitioned to **Titan embeddings** via **Amazon Bedrock** for production.
- Built retrieval-augmented generation (RAG) pipelines using **Titan Embeddings** and **Amazon OpenSearch**, grounding contract clause generation and reviewer summaries in retrieved precedents and structured metadata.

- Integrated **Claude via Amazon Bedrock** to generate clause-level summaries and risk rationales, adapting generation logic to reviewer personas (legal, compliance, audit) using prompt templates and metadata conditioning.
- Developed **agentic prompt orchestration logic** inspired by **LangChain**, enabling routing of generation tasks based on clause type, reviewer preferences, and feedback context through backend microservices.
- Deployed zero-shot classification models using **Hugging Face BART-MNLI** on **Amazon SageMaker** to tag clauses with renewal risk and compliance labels.
- Incorporated reviewer feedback loops into GenAI workflows, dynamically adjusting prompt structures, retrieval filters, and summarization focus areas to improve alignment and reduce manual tagging effort over time.
- Captured reviewer feedback via Spring Boot APIs; maintained intermediate states in **Redis** for performance and wrote finalized data back to Delta Lake.
- Exposed secure clause APIs via **Spring Boot (Java 17)** with **GraphQL** and **REST** endpoints backed by Delta Lake and OpenSearch.
- Enhanced API responsiveness with **Redis** caching for vector similarity lookups and scoring results.
- Created reviewer dashboards in **Power BI**, visualizing scoring metrics, turnaround times, and audit trails.
- Containerized scoring, enrichment, and feedback services with **Docker**; deployed on **EKS** with **Helm**-based blue/green rollouts.
- Provisioned and managed infrastructure with **Terraform**, maintaining reproducible, version-controlled deployments across EKS, S3, IAM, and VPC.
- Automated CI/CD pipelines using **GitHub Actions**, including multi-stage testing, container builds, and Helm validations.
- Instrumented microservices with **Spring Actuator** and **OpenTelemetry**, feeding metrics to **Prometheus** and **Grafana** for real-time observability.
- Integrated incident response tools (**PagerDuty**, **ServiceNow**) for auto-escalation of failures, SLA breaches, and pipeline retries to engineering and compliance teams.

***Key Technologies** :AWS DMS, S3, Delta Lake, PySpark, Databricks, Airflow, Step Functions, Glue, Athena,Titan Embeddings, Claude, OpenSearch, LangChain, BART-MNLI,Spring Boot, GraphQL, REST, Redis, Docker, EKS, Helm,Terraform, GitHub Actions, Prometheus, Grafana, OpenTelemetry, ServiceNow, PagerDuty, Power BI*

Senior Software Engineer — Dec 2019 – Nov 2022

Synchrony Financial - Stamford,CT

Overview

A platform modernization initiative to track platform usage, provider behavior, and repayment patterns across business systems. The solution enabled early risk detection, SLA breach visibility, and real-time anomaly insights, streamlining compliance reviews and partner performance monitoring through secure, audit-ready data pipelines and integrated scoring workflows.

Responsibilities:

- Built ADF ingestion workflows to consolidate **Kafka-based user interaction logs**, **batch provider visit files from SFTP**, and **repayment data from Oracle** into **ADLS Gen2**, standardizing inputs for risk modeling pipelines.
- Serialized incoming streams with **Apache Avro**, enforcing schema contracts via **Kafka Schema Registry** to maintain downstream consistency.

- Processed 3B+ repayment records in **Databricks** using **PySpark**, enriching data with provider IDs, visit schedules, and treatment categories to generate model features.
- Joined behavioral logs from **Salesforce** and claim summaries from **SAS**, enabling multi-source feature integration for risk analysis.
- Loaded curated features into **Snowflake**, then applied versioned risk classification using **Snowflake SQL procedures** and rule-based tagging.
- Integrated **Informatica** ETL pipelines with **Snowflake** to deliver analyst-curated risk features to compliance and audit systems.
- Scheduled scoring jobs with **ADF** and triggered **SSIS** workflows to merge SQL Server-based repayment logs into consolidated risk datasets on **ADLS Gen2**.
- Deployed **XGBoost** and **Random Forest** models in **Azure ML** to classify repayment risk tiers using engineered features like payment streaks, partial payments, and SLA breaches.
- Leveraged **Isolation Forest** models for unsupervised anomaly detection across provider repayment patterns, alerting early warning teams to unusual activity.
- Developed CLI-based scoring simulators in **Flask**, enabling business analysts to run scenario tests and receive model explanations for rapid feedback during Agile sprints.
- Developed asynchronous **Python FastAPI** endpoints to ingest reviewer annotations, scoring overrides, and model feedback, tightly integrated with ML pipelines for real-time risk recalibration.
- Applied deep NLP pipelines using **DistilBERT**, **Named Entity Recognition (NER)**, and keyphrase extraction models hosted in **Azure ML** to extract rationale, sentiment, and structured policy terms from free-text reviewer feedback and repayment notes.
- Transformed NLP outputs into structured features in **Databricks**, integrating them with scoring outputs to enable rationale-aware model recalibration and overrides.
- Indexed NLP-enriched feedback and audit logs in **Azure Data Explorer (ADX/Kusto)** for quick retrieval during compliance investigations and auditor queries.
- Delivered risk and sentiment dashboards in **Power BI**, enabling credit risk and compliance teams to monitor trends and reviewer rationale in near real-time.
- Enabled analyst-driven feature engineering using **Alteryx**, allowing rapid experimentation with **Snowflake** datasets and model inputs under **Agile** prototyping cycles.
- Enforced **RBAC** and dataset lineage tracking using **Azure Purview** and **Snowflake's information schema**, simplifying governance audits and enabling role-specific data access.
- Captured reviewer feedback and model override decisions through secure **Spring Boot** and **FastAPI** services, deployed via **Azure App Services** and integrated with real-time scoring pipelines.
- Managed CI/CD pipelines using **GitLab** and **Azure DevOps**, deploying **Flask**, **FastAPI**, and **ML** workloads with isolated QA and blue/green environments.
- Controlled access to all runtime services using **Vault**, scoped by Azure AD roles and environment-specific secrets provisioning.
- Maintained infrastructure using **Terraform**, provisioning secure **Azure ML** endpoints, **App Services**, **Snowflake** policies, and **Purview** metadata pipelines.
- Routed SLA violations, scoring drift alerts, and retry anomalies to **Azure Monitor** and **ServiceNow**, enabling proactive response across engineering and compliance teams.

***Key Technologies** :Azure Data Factory (ADF), Kafka, Apache Avro, Kafka Schema Registry, SFTP, Oracle, ADLS Gen2, Azure Purview, Databricks, PySpark, Snowflake, Informatica, SSIS, Azure ML, XGBoost, Random Forest, Isolation Forest, DistilBERT, NER, FastAPI, Flask, Spring Boot, MongoDB, Delta Lake, ADX (Kusto), Power BI, Alteryx, Terraform, Vault, Azure DevOps, GitLab, Azure App Services, Azure Monitor, ServiceNow*

Software Engineer **June 2016 – Dec 2019**

RedBus – Hyderabad,India

Overview

This initiative replaced a static, rules-based discounting model with a demand-aware decisioning platform designed to personalize promotional strategies in real time. The system integrated booking velocity, historical route patterns, and seasonal context to identify underperforming trips early and trigger targeted incentives. By aligning offers with predicted demand behavior, the platform helped increase conversion rates while minimizing unnecessary marketing expenditure.

Responsibilities:

- Extracted raw booking data from on-prem **OracleDB** using **PySpark JDBC**, landing route-level logs in **Amazon S3** as **Parquet** to support staged migration and fault-tolerant processing.
- Cleaned and enriched booking records using **PySpark**, resolving nulls, aligning timestamps, and normalizing pricing to prepare data for modeling in an **Agile** model iteration cycle.
- Migrated static route metadata and seat configurations into partitioned **Parquet** datasets on **S3**, enabling consistent lookup joins during model scoring.
- Scheduled ingestion and transformation flows via **CRON jobs**, and prototyped **Apache Airflow** DAGs for scalable orchestration of scoring pipelines.
- Transformed enriched data into **Redshift**-compatible tables using **PySpark**, applying schema mapping and loading via **Redshift COPY** for fast analytical querying.
- Designed, trained, and tuned **Gated Recurrent Unit (GRU)** networks using **Keras** on time-series booking data, capturing temporal dependencies in route demand and seasonality.
- Implemented hyperparameter tuning, dropout regularization, and early stopping to optimize model generalization and avoid overfitting.
- Achieved a 30% improvement in promotional offer efficiency by enabling targeted discounts based on model-predicted underperforming routes.
- Served promo eligibility scores via **Spring Boot** APIs, applying session-based access controls and persisting audit logs into **OracleDB**.
- Developed ETL modules in **Talend** to ingest vendor-curated pricing metadata and refresh seat availability indicators daily into **S3** and **Redshift**.
- Enabled batch and ad-hoc offer reports via **SSRS**, supporting audit trails and daily reviews of promo usage by campaign, region, and route.
- Packaged **Spring Boot** APIs and scoring services as fat JARs using **Maven Assembly Plugin**, deployed via scripted **Docker** containers on **Amazon EC2**.
- Managed multi-container test setups using **Docker Compose**, ensuring deployment parity between dev, QA, and staging environments.
- Optimized API latency by caching route-level scoring outputs in **Redis**, applying retry wrappers to handle **S3** and Redis read timeouts gracefully.
- Logged transformation checkpoints using **Log4j**, tracking promo effectiveness, scoring lag, and seat-fill conversion metrics.
- Visualized booking velocity and promo impact via **Tableau**, surfacing underperforming routes and identifying high-ROI discount patterns for planners.

Key Technologies:*OracleDB, PySpark, Amazon S3, Parquet, Amazon Redshift, Talend, SSRS, Spring Boot, Maven, Docker, Docker Compose, Redis, Log4j, Tableau, Apache Airflow, GRU (Keras), JWT, Resilience4j, AWS Glue Data Catalog, Athena, Prometheus, Grafana, Redash*

Hadoop Developer May 2014 – June 2016

BigBasket – Hyderabad,India

Overview

An enterprise data platform built to consolidate partner operations, transactional activity, return behavior, and SLA metrics across regional ecosystems. The system enabled centralized performance monitoring, trend analysis, and actionable insights for supply chain and partner management teams.

Responsibilities:

- Assisted in extracting daily sales and return records from **OracleDB** using **Sqoop**, landing data into **HDFS** for processing.
- Supported ingestion scheduling using **Oozie**, helping automate daily data transfers from regional systems into centralized storage.
- Helped clean and organize return data using basic **MapReduce** logic, focusing on removing nulls, aligning timestamps, and tagging missing values.
- Collaborated with senior engineers to join refund records with **SAP** delivery logs using **Hive**, allowing teams to trace issues back to fulfillment timelines.
- Created partitioned **Hive** tables to organize data by region and date, making it easier for analysts to run performance reports.
- Worked with BI teams to publish dashboards in **Tableau**, helping visualize trends in SLA breaches and return spikes.
- Collaborated with data integration team working on **Ab Initio** pipelines to align regional partner refund feeds with centralized ingestion workflows and ensure consistency across input formats.
- Supported downstream reporting teams by testing Hive outputs against **Teradata** summaries, helping validate SLA metrics across partners during weekly data reconciliation efforts.
- Participated in weekly **Agile** reviews to investigate data quality issues and iterate on schema design and Hive processing best practices.
- Documented key refund metrics and SLA definitions to help operational teams interpret reports more effectively.

Key Technologies : *OracleDB, Apache Sqoop, HDFS, Avro, Apache Oozie, CRON, MapReduce (Java), Hive, HiveQL, Hive SerDe, Hive UDF, ODBC, Tableau, Git, JIRA*