Claudine Linda Wa Nciko

Student number: 169375

CAT1: Theory on Big Data in Finance and Banking

1. Introduction to Big Data (2 Marks)

Why does HakiLend's scenario qualify as a big data challenge? HakiLend's modernization project faces a Big Data challenge due to the following:

a. Volume:

- 20 years of legacy transactional data stored on mainframe COBOL systems.
- 20 million transactions per day, driven by mobile banking expansion.

b. Velocity:

- Need for real-time fraud detection of cross-border credit card transactions.
- Regulatory audits require timely access to historical records.

c. Variety:

- Structured data: transaction logs, loan records, customer profiles.
- Semi-structured data: credit risk reports from external agencies.
- Unstructured data: potential document uploads (ID scans, application PDFs).

d. Other Big Data Characteristics:

- Veracity: ensuring fraud detection models and compliance reporting are based on accurate data.
- Value: extracting insights for credit risk analytics and customer behavior prediction using machine learning.

HakiLend requires a scalable, flexible, and cost-effective Big Data solution rather than traditional databases.

References: ((Ngugi, 2025), (Zubenko, 2023), (Baker, 2025))

2. **Big data architecture & components (13 marks)**

- **Proposed high-level end-to-end architecture**: HakiLend's Big Data system must handle batch processing (historical data) and streaming (real-time fraud detection). Below is a layered architecture suitable for its modernization needs:

A. Data ingestion layer (collecting data from various sources)

- Batch Ingestion (Historical Data Integration)
  - Apache Sqoop: Extracts legacy COBOL data from mainframes into Hadoop.
  - AWS Glue: Processes structured data from relational databases.
- Streaming Ingestion (Real-Time Data Processing)
  - Apache Kafka: Captures live mobile transactions for fraud detection.
  - AWS Kinesis: Streams transaction logs and third-party credit data.

B. Storage layer (handling large data volumes efficiently)

- Data Lake (Raw Storage for Scalability & Variety)
  - Amazon S3 / Azure Data Lake: Stores structured, semi-structured, and unstructured data.
  - Hadoop HDFS: Provides distributed fault-tolerant storage for large-scale analytics.
- Data Warehouse (Optimized for Regulatory Reports & Historical Queries)
  - Snowflake / Google BigQuery: Enables fast SQL queries for regulatory compliance.
  - Apache Hive: Manages structured and semi-structured data for batch analytics.
- NoSQL Database (Credit Risk Integration & Low-Latency Access)
  - MongoDB / Apache Cassandra: Stores third-party credit rating data for risk assessment.

C. Processing layer (handling batch & streaming data separately)

- Batch Processing (Legacy Data Migration & Machine Learning Training)
  - Apache Spark: Runs batch ETL (Extract, Transform, Load) for credit risk model training.
  - MapReduce: Efficiently processes legacy transactional data.
- Streaming Processing (Fraud Detection & Instant Risk Assessment)

    - Apache Flink / Spark Streaming: Monitors transactions in real-time for fraud detection.
    - Elasticsearch: Stores indexed transactional logs for real-time search & investigation.

D. Analytics & Machine Learning Layer

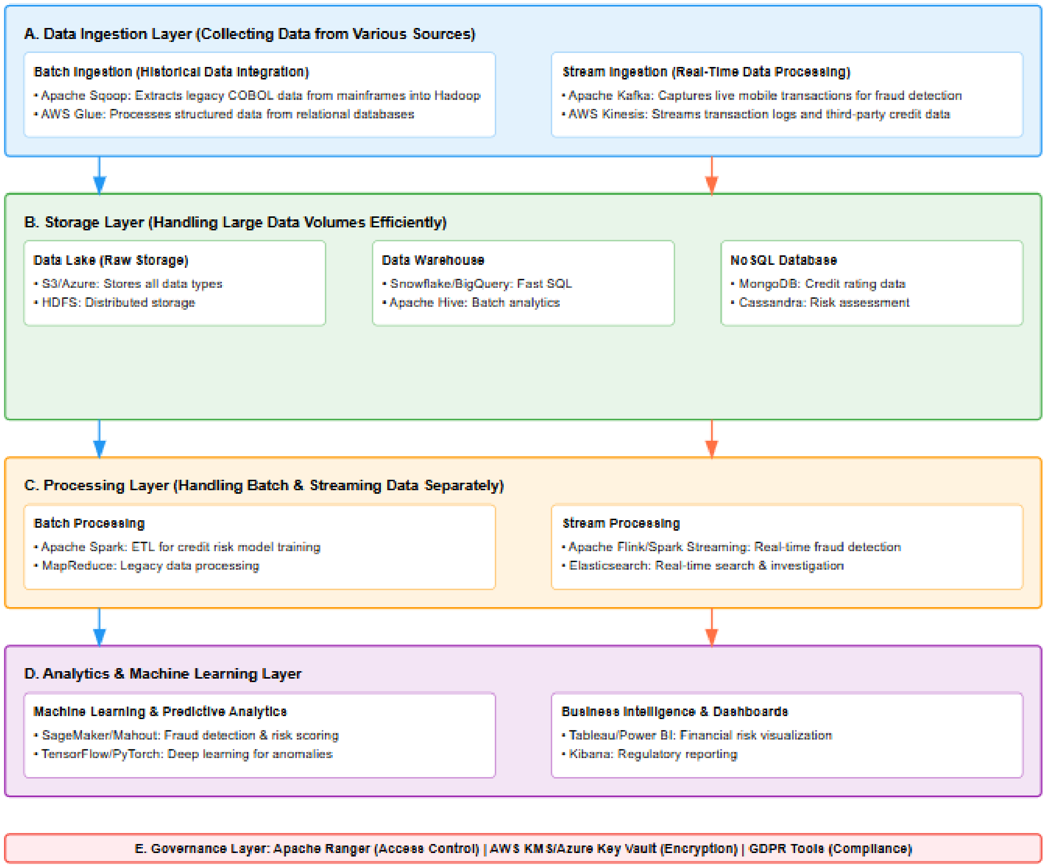
- Machine Learning & Predictive Analytics
    - AWS SageMaker / Apache Mahout: Trains fraud detection models & credit risk scoring models.
    - TensorFlow / PyTorch: Enables deep learning for anomaly detection in transactions.
- Business Intelligence & Dashboards
    - Tableau / Power BI / Kibana: Visualizes financial risk trends and generates regulatory reports.

E. **Governance, Security, & Compliance**

- Data Security & Access Control
    - Apache Ranger: Ensures role-based access control for audit compliance.
    - AWS KMS / Azure Key Vault: Encrypts sensitive data (to comply with local data laws).
- Regulatory Compliance & Audit Trails
    - GDPR & Local Compliance Tools: Stores audit logs in tamper-proof databases for regulatory checks.

- **Diagram labeling key components**

HakiLend End-to-End Big Data Architecture



- **How Each Component Addresses HakiLend's Needs**

Requirement	Solution
Legacy Integration	Apache Sqoop & AWS Glue for extracting mainframe data into a modern ecosystem.
Real-Time Fraud Detection	Apache Kafka & Spark Streaming monitor live transactions for anomalies.
Data Governance & Compliance	Apache Ranger ensures audit logs & controlled data access.
Scalability & Performance	Cloud storage (S3, BigQuery) handles massive data volumes cost-effectively.

References: ((Ngugi, 2025), (Dorlikar & Mohod, 2024), (Azzabi, Alfughi, & Ouda, 2024), (Hanae, Abdellah, Saida, & Youssef, 2023))

3. **Common Big Data Challenges in Banking (3 Marks)**

Challenge 1: regulatory compliance & data sovereignty

- Problem: HakiLend must comply with GDPR and in-country data sovereignty laws.
- Solution: use regional cloud zones (AWS Outposts, Azure Stack) to store sensitive customer data in-country.

Challenge 2: limited in-house big data team

- Problem: HakiLend's team lacks expertise in managing complex architectures.
- Solution: use managed cloud services like AWS Glue (ETL) & Databricks (ML) to reduce operational overhead.

References: ((Ngugi, 2025), (AWS-Documentation, 2024), (N-iX, 2023))

4. **HakiLend's Justification (2 Marks)**

Why traditional databases are insufficient

- Cannot handle high-velocity fraud detection (real-time analytics is needed).
- Do not support semi-structured or unstructured data (credit risk reports, logs).

Why big data solutions are necessary

- Machine learning-driven credit scoring improves lending decisions.
- Scalability in storage & processing ensures cost-effectiveness

References: (Finworks, 2023)

5. **YARN & Resource Management (2 Marks)**

How yarn allocates cluster resources among different applications

In HakiLend's case, where fraud analytics, risk modeling, and marketing teams need to run Hadoop jobs on a shared cluster, YARN (Yet Another Resource Negotiator) ensures efficient resource allocation by:

- Centralized resource management: YARN dynamically assigns CPU & memory to different teams based on job priority and resource availability.
- Multi-tenant scheduling: it allows multiple teams to submit Hadoop jobs concurrently without conflicts.

- containerized execution: jobs from fraud detection, credit risk, and marketing are isolated to avoid resource contention.

Key features ensuring balanced resource usage

- Capacity scheduler: ensures that fraud detection (real-time processing) gets higher priority, while batch processing jobs (risk modeling) are scheduled when resources free up.
- fair scheduler: distributes resources equitably so no single team monopolizes the cluster.
- Pre-emption: if fraud analytics needs urgent resources, YARN pre-empts lower-priority marketing jobs to ensure critical workloads run first.

References: ((Hadoop YARN Architecture, 2023), (Finworks, 2023), (Ngugi, 2025))

6. **Hadoop Ecosystem Tools (2 Marks)**

- Apache Kafka (Use: real-time fraud detection): streams mobile transactions & detects anomalies in real-time.
- Apache Airflow (Use: automated ETL pipelines): schedules daily regulatory reports & historical data integration.

References: ((Ngugi, 2025), (Gill, 2024), (Apache Hive))

7. **RDDs, DataFrames, and Datasets (3 Marks)**

Feature	RDDs	DataFrames	Datasets
Optimization	No	Yes	Yes
Performance	Slow	Faster	Fastest
Schema Enforcement	No	Yes	Yes

Best Choice for HakiLend: use DataFrames for ETL on structured financial data (faster SQL-like queries).

8. **Tool Selection & Complexity (2 Marks)**

To simplify Big Data management:

- Focus on cloud-managed services (BigQuery, AWS Glue) to reduce infrastructure burden.
- Minimize unnecessary tools (use Spark for both batch & streaming, rather than separate Flink).
- Leverage open-source tools to cut licensing costs (Kafka, Airflow).

References

Apache Hive. (n.d.). *Databricks Inc.* From https://www.databricks.com/glossary/apache-hive

AWS-Documentation. (2024). Data protection in AWS Outposts. From https://docs.aws.amazon.com/outposts/latest/userguide/data-protection.html?utm_source=chatgpt.com

Azzabi, S., Alfughi, Z., & Ouda, A. (2024, July 22). Data Lakes: A Survey of Concepts and Architectures. *MDPI*. From https://www.mdpi.com/2073-431X/13/7/183

Baker, D. (2025). 5 Challenges for Financial Institutions to Overcome When it Comes to Big Data. *Vericast*. From https://www.vericast.com/insights/report/5-challenges-for-financial-institutions-to-overcome-when-it-comes-to-big-data/

Dorlikar, R., & Mohod, D. S. (2024, June). Fraud Detection and Prevention in Financial Services Using Big Data Analytics. *International Journal of Scientific Research in Science Engineering and Technology*. From https://www.researchgate.net/publication/381263670_Fraud_Detection_and_Prevention_in_Financial_Services_Using_Big_Data_Analytics

Finworks. (2023, August 18). Future of Big Data in Financial Services. *Finworks*. From https://finworks.com/blogs/future-of-big-data-in-financial-services

Gill, N. S. (2024, August 29). Batch and Real Time Data Ingestion with Apache NiFi for Data Lake . *XenonStack*. From https://www.xenonstack.com/blog/real-time-data-ingestion

Hadoop YARN Architecture. (2023, April 24). *GeeksforGeeks*. From https://www.geeksforgeeks.org/hadoop-yarn-architecture/

Hanae, A., Abdellah, B., Saida, E., & Youssef, G. (2023). End-to-End Real-time Architecture for Fraud Detection in Online Digital Transactions. *(IJACSA) International Journal of Advanced Computer Science and Applications, 14*. From https://thesai.org/Downloads/Volume14No6/Paper_80-End-to-End%20Real-time%20Architecture%20for%20Fraud%20Detection.pdf#:~:text=In%20this%20article%2C%20we%20provide%20a%20real-time%20architecture,unsupervised%20machine%20learning%20%28ML%29%20algorithm%20n

Ngugi, J. (2025). Big Data Architecture. From file:///C:/Users/user/Desktop/MASTER/MODULE5-2025/BDFB/Big%20Data%20Architecture%20Case%20Study%20(1).pdf

N-iX. (2023, February 28). Big Data for financial services: benefits, challenges, and use cases. *N-iX*. From https://www.n-ix.com/big-data-for-financial-services/

Shalimov, A. (2023, August 23). Big Data in the Banking Industry: The Main Challenges and Use Cases. *Eastern Peak*. From https://easternpeak.com/blog/big-data-in-banking-and-financial-services/

ZAHARIA, M., XIN, R. S., WENDELL, P., DAS, T., ARMBRUST, M., DAVE, A., . . . STOICA, I. (2016, October 28). Apache Spark: a unified engine for big data processing. From https://people.eecs.berkeley.edu/~matei/papers/2016/cacm_apache_spark.pdf

Zubenko, V. (2023). Unlocking the potential of big data in modern banking: a comprehensive guide. From https://www.avenga.com/magazine/how-big-data-changes-banking/#:~:text=The%20future%20of%20big%20data,in%20an%20increasingly%20competitive%20landscape.