MMU®

MULTIMEDIA UNIVERSITY

CIT6234 Advanced Database

ASSIGNMENT 2 (30%) & PRESENTATION (5%)

Objective:

The objective of this assignment is to enable students to design and implement big data pipelines using systems thinking, processing massive amounts of data using mainstream big data technologies such as HDFS, HBase, Hive, and Spark. These technologies are the cornerstone of many modern data architectures. This assignment ensures that students not only understand the theory but also gain practical experience in big data implementation by designing end-to-end big data architectures through simulating real-world enterprise systems. Students will learn real-time and batch processing models, implementing data acquisition, storage, processing, and analysis, and understanding how different big data technologies are integrated, transforming abstract concepts into concrete examples.

Important Milestones:-

Week 12 (26th Jan 2025) by 12.A.M via eBwise

Marks will be deducted by 10% for each day late. No submission will be accepted after being 3 days late.)

Presentation on Part B ONLY: Week 13 & Week 14 *(during tutorial section)*

Complete this assignment with the same group of 3 to 4 members formed for assignment. Each group should submit the report (softcopy) to your respective tutors.

Consider any solutions based on big data scenarios (offline batch processing, batch reporting and business intelligence, real-time retrieval, real-time random access, real-time processing and services, big data application development processes, and etc.). Design and implement a big data pipeline for any solution based on a big data scenario. Each group will choose a suitable case study (e.g., e-commerce: electronics retail, bookstores, fashion, etc.) and explain why it needs big data technology to implement recommendation functionality.
The following recommended open datasets are for reference only. You are welcome to propose other big data solutions using other open datasets; the only requirement is that these solutions must be based on the concepts or topics discussed in class.

Tasks [Total: 100 marks]

Task 1: Case Study Selection & System Justification [20 Marks]

1.1 Case Study Selection and Big Data Justification [4 marks]
Select a domain and an open dataset. Justify why this domain/scenario requires big data technologies.

1.2 Business Process & Scenario Identification [6 marks]
Identify the core business processes and the specific big data scenario you are addressing (e.g., Offline Batch Processing, Real-time Retrieval, Batch Reporting and BI).

1.3 System Architecture Diagram [10 marks]
Design and document the complete big data pipeline architecture using Draw.io or similar tool. The diagram must show:

- The data flow from source to final application/insight
- The integration and role of HDFS, HBase, Hive, and Spark
- The distinction between batch and real-time processing paths (if applicable)

Task 2: Data Modeling & Storage Strategy [25 Marks]

2.1 HDFS Data Storage Strategy [6 marks]

- Design the HDFS directory structure for raw, processed, and curated data.
- Justify your data partitioning strategy (e.g., by date, category, region).

2.2 HBase Schema Design for Real-time Access [6 marks]

- Design an HBase table to support low-latency, random read/write access for a specific use case (e.g., quick user profile lookups, real-time product stats).
- Define the row key, column families, and explain your design choices.

2.3 Hive Data Warehouse Design [7 marks]

- Create a Hive external table schema over the data in HDFS.
- Design a dimensional model (star or snowflake schema) for analytical processing, defining fact and dimension tables.

2.4 Spark Data Processing Plan [6 marks]

- Outline the role of Spark in your pipeline (e.g., for ETL, data cleaning, feature engineering, or machine learning).

- Describe the DataFrames/Transformations you will use.

Task 3: ETL & Data Processing Implementation [15 Marks]

3.1 Data Ingestion & Extraction [5 marks]
Develop scripts/code to ingest the chosen open dataset into HDFS.

3.2 Data Transformation with Spark [5 marks]
Implement a Spark application (using PySpark or Scala) to perform data cleaning, filtering, and transformation, preparing the data for Hive and HBase.

3.3 Data Loading & Quality Checks [5 marks]

- Load the transformed data into the Hive tables and HBase.
- Implement basic data quality checks (e.g., handling nulls, checking for duplicates).

Task 4: Analytics & Query Implementation [25 Marks]

4.1 Batch Analytics with Hive SQL [8 marks]
Write at least four complex Hive queries that demonstrate batch analytics on your data, such as:

- Top-N analysis (e.g., top 10 most viewed products)
- User segmentation and behavior analysis
- Aggregation using `GROUP BY` and `ROLLUP`/`CUBE`
- A query involving multiple joins and a `CASE` expression.

4.2 Real-time Querying with HBase [4 marks]
Design and execute HBase shell commands or Java/Python code to perform random access queries (e.g., `get` a specific user's recent activity).

4.3 Advanced Analytics with Spark [5 marks]
Implement a Spark SQL or MLlib application to perform a more advanced analytical task (e.g., calculating a key metric, building a simple recommendation model, or performing clustering).

4.4 Technology Comparison & Analysis [8 marks]
Compare the performance and use-case suitability of Hive, HBase, and Spark. Run a comparable query/operation on two technologies and analyze the results (e.g., a key-value lookup in HBase vs. Hive, or an aggregation in Hive vs. Spark SQL).

Task 5: Performance Optimization [10 Marks]

5.1 Hive & Spark Optimization [5 marks]

- Implement performance optimizations in Hive (partitioning, bucketing, using ORC/Parquet) and/or Spark (caching, partitioning).

- Show a performance comparison (e.g., query runtime) before and after optimization.

5.2 HBase Optimization [5 marks]

- Propose and, if possible, implement an optimization for your HBase table (e.g., Bloom filters, compression, pre-splitting).

- Justify how this optimization improves performance for your specific access pattern.

Task 6: Documentation & Reflection [5 Marks]

6.1 Comprehensive Documentation [5 marks]
Document the entire project, including:

- Architecture decisions and trade-offs.

- Challenges faced and how they were resolved.

- A reflection on how the different technologies (HDFS, HBase, Hive, Spark) integrated to form a complete big data solution.

Recommended Open Datasets

Students are encouraged to use one of the following datasets or propose their own:

1. **RetailRocket E-commerce Dataset**

   - **Link:** Kaggle: RetailRocket

   - **Content:** 2.7 million events (views, cart additions, transactions) from a real e-commerce website.

   - **Suitability:** Perfect for recommendation engines, user behavior analysis, and batch reporting.

2. **Amazon Product Data / Amazon Reviews**

   - **Link:** Stanford Network Analysis Project (SNAP)

   - **Content:** Product metadata and hundreds of millions of reviews.

   - **Suitability:** Excellent for product affinity analysis, review sentiment analysis, and large-scale batch processing.

3. **NYC Taxi Trip Data**

  - **Link:** NYC TLC Trip Record Data
  - **Content:** Detailed trip records for yellow and green taxis, including pick-up/drop-off dates/times, locations, fares, and passenger counts.
  - **Suitability:** Ideal for spatio-temporal analysis, calculating aggregates over time, and simulating a real-time dashboard for taxi availability.

[Presentation on Part B]
Presentation: [10 marks]

Criteria	**Description**	**Mark Allocated**	**Score Achieved**
Content Knowledge	Depth, accuracy, and understanding of the subject matter	3	
Organization	Clear structure (introduction, body, conclusion); logical flow of ideas	2	
Presentation Skills	Clarity, pace, engagement, voice modulation, eye contact, body language	2	
Visual Aids	Quality, relevance, and clarity of slides.	1	
Time Management	Sticking to the allotted time without rushing or dragging	1	
Response to Questions	Ability to handle questions thoughtfully and accurately	1	
TOTAL:			