Project Assignment: Building a Scalable Meeting Intelligence Data Pipeline using the MeetingBank Dataset

1. Project Overview

This project forms the practical and graded component of the *Data Engineering* module. In teams of four, students will design and implement a **complete data engineering pipeline** based on the *MeetingBank* dataset: a large-scale, multimodal collection of city council meeting data from six major U.S. cities (Hu et al., ACL 2023).

The dataset contains over 1,300 meetings, including transcripts, summaries, metadata, agenda documents, and links to multimedia content.
The project's main objective is to transform this complex, semi-structured data into a **structured, queryable, and analyzable system**, simulating a real-world data engineering workflow.

The assignment encourages the integration of theoretical and practical competencies acquired during the course, including data ingestion, transformation, storage, querying, and ethical reflection, while fostering teamwork and professional communication.

2. Learning Objectives

Upon successful completion, students will be able to:

- Design and document an **end-to-end data engineering pipeline**.
- Ingest, clean, and transform structured and unstructured data.
- Apply **relational (SQL)** and **non-relational (NoSQL)** database systems.
- Implement efficient data storage, querying, and retrieval mechanisms.
- Demonstrate familiarity with **cloud and distributed systems** concepts.
- Reflect on **data ethics**, transparency, and responsible data management.
- Communicate project findings effectively in both written and oral formats.

3. Project Tasks

Core Project Tasks (all groups)

All teams will complete the following shared core tasks:

1. **Data Ingestion and Cleaning**
   - Access the MeetingBank dataset via the HuggingFace API or Zenodo repositories.
   - Use a *representative subset* of this dataset.
   - Parse meeting transcripts, summaries, and metadata (city, date, agenda, participants).
   - Handle JSON and CSV structures; clean and normalize raw data.
2. **Data Transformation and Modeling**
   - Create a structured schema suitable for both analytical and operational purposes.
   - Derive additional attributes (e.g., word counts, meeting durations, number of speakers, topic occurrences).
   - Apply data validation, consistency checks, and transformations for downstream querying.
3. **Data Storage**
   - Store structured data in a **relational database** (e.g., PostgreSQL or MySQL).
   - Store unstructured data (transcripts, summaries) in a **NoSQL database** (e.g., MongoDB or Elasticsearch).
   - Document schema design, data types, and indexing strategy.
4. **Querying and Analysis**
   - Implement analytical queries that demonstrate the usability of the data model.
   - Provide SQL examples (aggregations, joins, window functions) and NoSQL retrieval queries.
   - Visualize at least one meaningful analysis result (e.g., city-wise trends, topic frequencies).
5. **Documentation, Ethics, and Presentation**
   - Document the full data engineering workflow, from design to implementation.
   - Reflect on ethical aspects of using public meeting data and ensuring transparency.
   - Present project findings and a live demo of the pipeline in class.

4. Group-Specific Extension Tasks

Each of the four project teams will implement **one unique extension task** in addition to the core pipeline.
This allows exploration of specialized areas within data engineering while maintaining comparability across teams.

Group	Focus Area	Extension Task	Description
Group 1 – Cloud Pipeline Engineers	**Cloud & Big Data Systems**	Deploy part of the pipeline (e.g., storage or analytics layer) to a **cloud platform** such as AWS, Azure, or GCP.	Demonstrate cloud integration using services like S3, Redshift, or BigQuery. Include configuration details and scalability reflections.
Group 2 – Data Orchestrators	**Automation & Scheduling**	Integrate or simulate workflow orchestration with **Apache Airflow** or **Prefect**.	Create DAGs or scripted pipelines for automated ingestion, transformation, and loading. Document error handling and task dependencies.
Group 3 – Advanced Database Architects	**Query Optimization & Schema Design**	Implement and benchmark **query optimization techniques** and **advanced schema design**.	Apply indexing, CTEs, and normalization strategies. Present performance improvements and design rationales.
Group 4 – NLP & Text Engineers	**Text Processing & Analytics**	Extend the pipeline with **basic NLP-based analytics** on meeting transcripts.	Use libraries such as spaCy or HuggingFace to extract named entities, sentiment, or topics, and visualize results by city or meeting type.

5. Deliverables

Phase	Description	Deliverables
1. Proposal (until December 03, 11:55 PM via Moodle)	Definition of project goals, data sources, planned tools, and architecture sketch.	2-page project proposal
2. Architecture & Design (until December 05, 11:55 PM via Moodle)	Conceptual and logical design of the full pipeline (data flow, schema diagrams).	Data architecture diagram + rationale
3. Implementation	Construction of the working ETL/ELT pipeline, including database integration and data queries.	GitHub repository with code, data samples, and documentation
4. Presentation (December 19)	In-class presentation and live demonstration of the pipeline and results.	15-minute presentation (team-based)

5. Final Report (until December 23, 11:55 PM via Moodle)	Comprehensive technical documentation, project-reflection (e.g., work allocation within the team, project management tool used etc. → agile way preferred) and critical reflection.	Report (max. 10 pages)

6. Evaluation Criteria

Criterion	Description	Weight
Core Pipeline Implementation	Correctness, robustness, and completeness of the ETL process	40%
Extension Task	Quality and integration of the group-specific enhancement	20%
Querying & Analysis	Effectiveness of SQL/NoSQL queries and analytical insight	15%
Documentation & Reflection	Clarity, depth, and ethical awareness	15%
Presentation & Teamwork	Quality of communication, professionalism, and collaboration	10%

7. Technical Guidelines

- **Programming Language:** Python (pandas, json, SQLAlchemy, PySpark)
- **Databases:** PostgreSQL (or MySQL) + MongoDB (or Elasticsearch)
- **Cloud/Distributed Systems:** AWS, GCP, or Azure
- **Orchestration Tools:** Apache Airflow, Prefect (for Group 2)
- **Text Processing Libraries:** spaCy, NLTK, HuggingFace (for Group 4)
- **Visualization Tools:** Tableau, Power BI, Streamlit, or Matplotlib

All teams should ensure their code is modular, reproducible, and properly documented.

8. Ethical Considerations

Although the MeetingBank dataset is publicly available, students must:

- Cite the dataset creators and associated publication (Hu et al., ACL 2023).
- Avoid re-identifying individuals or extracting sensitive content.
- Reflect on data transparency, consent, and the implications of applying AI or analytics to public governance data.

9. Submission Format

- **Repository:** GitHub (with README, documentation, and working scripts).
- **Report:** PDF, max. 10 pages (including figures and references).
- **Presentation:** 15 minutes per group + 5 minutes Q&A on December 19
- **Deadline (Report + Repository):** December 23, 11:55 PM
  (submission via Moodle).

10. References

Hu, Y., Ganter, T., Deilamsalehy, H., Dernoncourt, F., Foroosh, H., & Liu, F. (2023). *MeetingBank: A Benchmark Dataset for Meeting Summarization*. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, Toronto, Canada.
https://meetingbank.github.io