Korbin Schulz, Aarian Ahsan, Sisira Aarukapalli, Riyasat Rashid, Luigi Vectorelli, Edgar Sanchez

9/10/24

Use Case Team 6

Project Name: Data Pipeline System for CSV Generation and Storage

Project Overview:

- **Purpose**: The goal of this project is to design and implement a data pipeline that generates data and stores it in CSV format for analysis and processing.
- **Target Audience**: Data engineers, analysts, and developers who need efficient data storage solutions.
- **Scope**: This project will involve the generation, storage, and transfer of data using pipelines. It will exclude more advanced features like database sharding but may use basic data management techniques.

Functional Requirements:

1. The system will generate synthetic data based on a pre-defined schema.
2. The system will export the generated data into CSV files.
3. The system will transfer generated CSV files to a specified storage location (local or cloud).
4. The system will provide basic error logging and reporting capabilities.

Non-Functional Requirements:

- **Performance**: The system should generate and transfer 1 GB of data within 30 minutes.
- **Security**: Data transfer should occur over secure protocols such as HTTPS.
- **Usability**: The system should provide a simple command-line interface for easy operation.
- **Reliability**: The system must ensure no data loss during transfer with 99.9% uptime.
- **Maintainability**: The codebase should be modular and well-documented to ensure ease of future development and updates.

Assumptions and Dependencies:

- Data will be generated in CSV format only.
- External libraries for CSV handling (like Python's `csv` module) are allowed.
- The system assumes cloud storage integration for data transfer (e.g., AWS S3 or Google Cloud Storage).

Acceptance Criteria:

- Successful generation of the specified dataset in CSV format.
- Transfer of generated CSV files to a defined storage location without data corruption.
- Logging and reporting functionality for errors encountered during data generation or transfer.

Additional Considerations:

- Integration with cloud services like AWS S3 or Google Cloud will be optional but supported.
- The project will not handle extremely large datasets that would require sharding or distributed computing solutions.