Synapsis

AI Engineer Challenge

PT Synapsis Sinergi Digital

A. Scenario

A coal mining company aims to optimize its mining operations using production data. Your task is to design and implement a data pipeline that collects, transforms, and loads coal production data from various sources into a data warehouse. Additionally, you will validate the data and create a dashboard to visualize key production metrics.

B. Data Sources

Please you can download the datasets file here, that includes :

1. **SQL Database:** A table production_logs with columns date, mine_id, shift, tons_extracted, quality_grade.
2. **IoT Sensors:** A CSV file equipment_sensors.csv with columns timestamp, equipment_id, status, fuel_consumption, maintenance_alert.
3. **API:** Provides daily weather data for Berau, Kalimantan, Indonesia (latitude: 2.0167° N, longitude: 117.3000° E) via the Open-Meteo API endpoint https://api.open-meteo.com/v1/forecast?latitude=2.0167&longitude=117.3000&daily=temperature_2m_mean,precipitation_sum&timezone=Asia/Jakarta&past_days=0&start_date=**{date}**&end_date=**{date}** the date is formatted with YYYY-MM-DD and returning

   {"latitude":2,"longitude":117.25,"generationtime_ms":4.305839538574219,"utc_offset_seconds":25200,"timezone":"Asia/Jakarta","timezone_abbreviation":"GMT+7","elevation":44,"daily_units":{"time":"iso8601","temperature_2m_mean":"°C","precipitation_sum":"mm"},"daily":{"time":["2025-06-01"],"temperature_2m_mean":[26.2],"precipitation_sum":[3.4]}}

C. Challenge Task

1. Design the Data Pipeline

You can use analytical databases like clickhouse or apache doris to do this challenge. Make sure that everything is run on docker so we can replicate it.

a. **Extraction:**

- Retrieve daily production data from the production_logs table using an SQL query.
- Read the equipment_sensors.csv file for mining equipment sensor data.
- Call the weather API to fetch daily weather data.

b. **Transformation :** Generate the following metrics :

- total_production_daily: Total tons of coal mined per day.
- average_quality_grade: Average coal quality per day
- equipment_utilization: Percentage of time equipment is operational (status "active") per day.
- fuel_efficiency: Average fuel consumption per ton of coal mined.
- weather_impact: Correlation between rainfall and daily production (e.g., production on rainy vs. non-rainy days).

c. **Handling Missing or Invalid Data:**

- If tons_extracted is negative, replace it with 0 or flag it as an anomaly.
- If sensor data is missing for an equipment, use the previous day's average or mark as "unknown".

2. Implement the ETL Script

a. Write a Python script to extract, transform, and load the data into a data warehouse table named daily_production_metrics.

b. Use SQL queries for database operations where necessary.

3. Validate the Data

a. Implement checks:

- Ensure total_production_daily is not negative.
- Verify equipment_utilization is between 0 and 100%.

- Confirm weather data is complete for each production day.

b. Handle anomalies: Log errors to a separate file or send notifications for further analysis.

4. **Create a Dashboard**

   a. Use Metabase (or a tool like Power BI/Superset) to create a dashboard with at least three visualizations:

   - **Line Chart:** Daily production trends (total_production_daily) over one month.
   - **Bar Chart:** Comparison of average_quality_grade across mines (mine_id).
   - **Scatter Plot:** Relationship between rainfall (rainfall_mm) and daily production (total_production_daily).

5. **Document and Version Control**

   a. Write a brief report (1–2 pages) explaining the pipeline design, ETL process, and validation steps.

   b. Use Git for versioning the code and provide a link to the repository.

6. **Bonus Point**

   Use collected data to make a production data prediction model. Do a time series forecasting model to predict next day production data. Only do this bonus part if you have done all the challenges.

D. Rules

1. The challenge must be submitted in **4 days** after you get this challenge.
2. If you can complete the challenge test before the deadline, it will be taken into consideration by us.

E. Expected Deliverables

All deliverables **must be included in the GitHub repository**, including documents, images, and Python code. Make sure to invite **mufidmove@gmail.com** a collaborator & reviewer on the github repository.

The following deliverables must be submitted :

1. Data pipeline design document (in PDF format).
2. Python script and SQL queries (if applicable).
3. Dockerfile and docker config file (if any).
4. Dashboard (screenshot or link).
5. Documentation report.

– Do Your Best –