Technical Test: ETL and Data Pipeline with Airflow and Python

You have an existing ERP system that exports daily sales data in CSV format to an S3 bucket. Your task is to build a simplified ETL pipeline that extracts this data, transforms it (e.g., cleans and aggregates), and loads it into a data warehouse. The pipeline should be scheduled to run daily.

Task 1: Set Up Airflow

- Set up Airflow using docker.
- Create a DAG (Directed Acyclic Graph) in Airflow that runs daily at a specified time.

Task 2: Data Extraction

- Write a Python script to connect to the S3 bucket (or any storage), download the latest sales data file, and store it locally.
- Include basic error handling for missing or corrupted files.

Task 3: Data Transformation

- Perform basic data cleaning (e.g., handle missing values, standardize formats).
- Aggregate the sales data by product category.

Task 4: Data Loading

- Load the transformed data into a data warehouse (e.g., PostgreSQL).

Task 5: Data Quality Checks

- Implement a simple data quality check in the Airflow DAG, such as row count validation.

Task 6: Documentation and Code Review

- Document your code, explaining key design choices and assumptions.
- Include a README file with instructions on setting up and running the pipeline.
- Ensure all code and documentation are uploaded to a GitHub repository.

Task 7: Testing

- Write basic tests for the transformation logic.
- Focus on ensuring that the pipeline runs correctly end-to-end.

Submission Requirements

- **GitHub Repository:**
    - Create a public or private GitHub repository for this project.
    - Commit all code, scripts (including your docker build/docker compose), and documentation to the repository.
    - Share the repository link as part of your submission.