HelloHeart

Senior Data Engineer Home Assessment

Introduction

As part of our interview process for the Senior Data Engineer position, we invite you to complete a home assessment designed to showcase your skills in creating ETL jobs, scheduling tasks with Airflow, and demonstrating cloud deployment capabilities. This project is a key component of our evaluation process and will help us understand your technical capabilities, problem-solving approach, and documentation skills.

Project Objective

Your task is to design and implement an ETL (Extract, Transform, Load) process that meets the following criteria:

1. **Extract** data from a provided source.
2. **Transform** the data according to specified requirements.
3. **Load** the transformed data into a target database.
4. Schedule the ETL process using Airflow.
5. Design a production-ready deployment for running the job in an AWS environment, using either LocalStack for a practical implementation or through detailed architecture documentation.

Data Source

- For this assessment, please use the following publicly available dataset: Public Health Data
- You may choose any dataset within this repository that interests you, provided it is relevant to health data and sufficiently complex to demonstrate your skills.

Requirements

ETL Process

- **Extract:** Write a script to extract data from the chosen dataset.
- **Transform:** Perform data cleansing, manipulation, and aggregation operations to prepare the data for analysis. Include at least three transformation operations (e.g., filtering, grouping, aggregating).
- **Load:** Load the transformed data into a database of your choice (explain why you chose it).

Airflow Scheduling

- Define a DAG in Airflow to manage the ETL workflow.
- Schedule the workflow to run at a frequency of your choice, justifying your decision.
- Implement error handling and logging to ensure reliability and maintainability.

Cloud Deployment Alternatives

Given the potential complexity and costs associated with deploying applications to a cloud provider, we offer two alternatives for demonstrating your cloud deployment capabilities:

Option 1: LocalStack Deployment

- Utilize LocalStack to simulate cloud environments locally.
- Containerize your ETL job using Docker, ensuring it can run within the LocalStack environment.
- Provide a `docker-compose.yml` file that orchestrates your ETL job, Airflow scheduler, and any mock cloud services required for your solution (e.g., S3, RDS).
- Include instructions in the `README.md` for running your solution (from a clean machine to a running solution) with LocalStack, detailing how to set up and interact with the simulated cloud services.

Option 2: Cloud Deployment Architecture Documentation

- If you prefer not to use LocalStack, you may instead submit detailed architecture documentation that explains how your ETL job and Airflow scheduler would be deployed to AWS.
- Your documentation should include:
  - ○ A comprehensive architecture diagram showing all components of your solution and how they interact within the cloud environment.
  - ○ Descriptions of the cloud services used and the role of each service in your solution.
  - ○ An explanation of any managed services (e.g., managed databases, container orchestration services) and why they were chosen.
  - ○ Consideration of security, scalability, monitoring, and cost-efficiency in your architecture design.

Deliverables

Please adjust your submission to include one of the following based on your chosen deployment alternative:

1. For **LocalStack Deployment**:

    - A GitHub repository containing all source code, `Dockerfile`, `docker-compose.yml`, and a `README.md` with setup and execution instructions using LocalStack.
2. For **Cloud Deployment Architecture Documentation**:
    - A GitHub repository containing all source code and a `README.md` with your architecture documentation, including diagrams and descriptions as specified above.

Ensure your documentation or LocalStack setup clearly demonstrates your understanding of cloud concepts and your ability to deploy applications in a cloud environment.

Evaluation Criteria

Your submission will be evaluated based on the following criteria:

- **Code Quality:** Clarity, readability, and organization of code.
- **Data Handling:** Effectiveness of data transformations and loading strategies.
- **Error Handling and Logging:** Robustness of error handling in the ETL process and Airflow scheduling.
- **Deployment and Documentation:** Clarity of deployment instructions and overall documentation quality.
- **Innovation and Problem-Solving:** Creativity in approach and solutions to encountered challenges.

Submission Guidelines

- Complete the project within 1 week of receiving this assessment.
- Submit your GitHub repository link via email (you may respond to recruiting to share the link).
- Ensure your repository is public.

We look forward to reviewing your project. Should you have any questions or need clarification on any aspect of the assessment, please don't hesitate to reach out.

Best of luck!