Code Assessment Document (v1.0)

By

Chitwan Humad

Contents

Assessment Details

Problem Statement:

- Create an ELT pipeline that ingests a CSV dataset (choose any sufficiently dense source eg. https://www.kaggle.com/datasets/abdullah0a/telecom-customer-churn-insights-for-analysis).
- Load up the dataset into a staging database of your choice.
- Design a transformation layer to process the input dataset for missing values (use defaults) and anonymising PII.
- The destination for the processed data should be a database ideal for generating reports.
- Establish an orchestration workflow for this pipeline to accept a feed every hour (should be configurable).
- Integrate any open-source reporting tool to generate statistics about the flow.
- Ensure the entire setup is available through composable container definition(s).

Tech Stack:

- Language/frameworks/solutions of your choice. Please just ensure, the solution is easy to run on a laptop.
- Please use open-source solutions wherever possible.

Delivery:

- Please share the entire source code as a public Github repository.
- Do add relevant instructions to run the code.
- Please also ensure it stays accessible for the duration of the discussions with HGI.

ETA:

- Please ensure the assignment is completed in about 16-20 hours (can be split over days if practical schedules demand).

Assessment – In scope:

- Bronze Layer (Data Lake): Read and writing csv data via a pipeline to store in the database table – csv data will be pulled up from Kaggle
- Silver Layer (Transformed Data layer): Data transformation and stored the refined data into database tables – up to 3 use cases
- Gold Layer (Reporting Layer): Pre-aggregated data for reporting purposes – up to 3 use cases

Assessment – Out of scope:

- Containerization of the solution

Assessment – Use Case Success Criteria:

The solution should be considered as successful if the following use cases are achieve during the user acceptance testing:

1. Tech stack selection: should be Open source as far as possible
2. Each run should have internal runid to track pipeline runs
3. CSV file(s) should be able read from <>/in/<name>.csv folder and load into the database without any change in the data in the raw_customer table of bronze_db database
4. Solution to enable hourly to ingest a new file hourly
5. The processes file should be moved/archived in the processed file into <>/processed/<name>_runid _datetimeid _done.csv
6. Solution should follow 3 use cases to conduct data transformations:
    a. Check for NaN or missing values for a few fields (field names – TBD)
    b. Check valid values for Age – should be a positive integer only

   c. Check for valid values from the data dictionary for ContractType field as Month-to-Month, One-Year, Two-Year

7. Bad data rows based on the above should be saved into the <>/baddata/<name>_runid _datetimeid _done.csv
8. Read bronze_db.raw_customer table data and perform following transformation to make presentable reports:
   a. Transform InternetService missing values to None
   b. Round off TotalCharges values to 2 decimals
   c. Define new dimension as Tenure_Range for each 10 blocks, e.g. 1-10, 11-20 so on
   d. Define Age_band dimension 20-25, 36-30 so on every 5 years
   e. Drop Age field to preserve PII information
   f. Define new dimension Category High/Medium/Low for MonthlyCharges < 50 Low, 51-100 medium and > 100 high
9. The transformed data should be stored into silver_db.customer table
10. Produce a aggregated data models to generate various reports in the gold_db.<table_names>, like:
    a. Count of customers by Categories (i.e. High/Medium/Low)
    b. Aggregated revenue (TotalCharges) by Contract Types
    c. Aggregated revenue (TotalCharges) by InternetService
    d. Customer demographic Presentation who availed technical support facility by Age_band and gender
11. A run and log table to record runs

Assessment – Tech Stack Selection: Tentative

- OS – Windows laptop
- Prefect for data pipeline – Open source
- Superset or Birt – Open source
- Database – Sqlserver Express using sa credentials

Assessment – Deliverables:

- Git repo url https://github.com/chitwanhumad/hg_datapipeline

(Kindly confirm you can access the url)

Assessment – Completion Date: Tentative

22-Aug-2025