ZATTOO

Hiring Assignment

This assignments consists of two problems

Problem 1) Programming task :

There is a sample dataset provided with this assignment. This dataset has a sample of events captured from the web app. It has following fields :
event_name: The name of the event tracked by the application.
user_id: Unique identifier of a user.
app_version: The version of the app used by the user.
event_time: Timestamp of the event. device_id: Unique
identifier of the client device. client: The name of the
client. (Only web in this sample)

Setup

You may choose whichever setup you are more comfortable with:

- colab
- local setup (Jupyter notebook etc)
- Docker

Use case:

As a product owner of this application I would like to extract specific insights from the user interactions with the application. Provide a solution that answers the below questions while demonstrating industry best practices. Make necessary assumptions based on the data and document them with the solution.

Questions

1. On which day did we observe the most activity?
2. How would you define a session?
3. What is the highest number of events observed in a session?
4. What is the average session duration?

Problem 2) In this problem, You may choose to solve any of the given two options ((a) streaming task or (b) batch-processing task), as per your preference

a) Imagine you are watching Netflix / listening to music on Spotify or similar apps. The app generates an event every 4 seconds and is captured by a component (Database or Apache Kafka or something similar) .
The event consists of the following information:

a. User identifier (user_id).
b. program identifier (program_id) . Whatever user is watching / listening to.
c. Program start time (start_time). The time when the user started watching/listening to this particular program .
d. A Unique session identifier (session_id) . A unique identifier for the duration of the session i.e. as long as the user is interacting with the device and watching / listening to something. (Assume session_id is same across all interactions with the app for a user)
e. Device (device) . The device identifier of the client e.g. apple tv / iphone etc
f. Event time (event_time). Timestamp of the current event.

How would you design the architecture of the solution which listens to the source frequently (Kafka / database(something similar)), and checks whether the stream is still ongoing. If yes it keeps the state of the watch request till it's no longer on going. Once the watch request stream is over (user is no longer watching / listening) it calculates the duration of the stream and writes the record to a destination component (database / file / kafka etc). The stream is assumed to be over when there is no record for a particular session_id for 60 sec after the last captured event.

Following fields need to be written to the file. user_id, program_id,
start_time, session_id, device, stream_duration

Describe the solution in detail. What tools/frameworks would you prefer and why. How would you ensure the component is scalable to handle peak hours.
Please use diagrams , & solution description/ algorithm to explain. There is no need for a concrete implementation. Just an architecture with description is expected.

Or

b) This exercise simulates a data pipeline for processing video session data for a streaming app

Scenario:

- Imagine you are watching Netflix / listening to music on Spotify or similar apps. The app generates an event every 4 seconds and writes to a kafka topic

    The event consists of the following information:

    - User identifier (user_id).
    - program identifier (program_id) . Whatever user is watching / listening to.
    - Program start time (start_time). The time when the user started watching/listening to this particular program .
    - A Unique session identifier (session_id) . A unique identifier for the duration of the session i.e. as long as the user is interacting with the device and watching / listening to something. (Assume session_id is same across all interactions with the app for a user)
    - Device (device) . The device identifier of the client e.g. apple tv / iphone etc
    - Event time (event_time). Timestamp of the current event.

    Say, the data is serialized in json format

- You need to build an Airflow DAG (Directed Acyclic Graph) to ingest this data, transform it, and load it into BigQuery(preferably)/Snowflake/Redshift for analysis.

Tasks:

1. *Data Ingestion:*
    - Create an Airflow task to consume data from the Kafka topic.
    - Define how the data will be deserialized from the Kafka message format
2. *Data Transformation:*
    - Create an Airflow task to process the ingested data. This could involve:
        - Extracting relevant fields (e.g., start_time, user ID, program_id).
        - Deriving new metrics (number of sessions per user per day). This also involves to define what is a session
        - Performing data cleaning (e.g., handling missing values, invalid data).
    - You can use any Python library(like Pandas) for data manipulation.
3. *Data Modeling:*
    - Define the schema for the BigQuery/Snowflake/Redshift table that will store the transformed data.
    - This includes specifying data types for each field.

4. *Data Loading:*

    - Create an Airflow task to load the transformed data into the BigQuery table. Define how the data will be loaded (e.g., append to existing table, overwrite)

Deliverables:

- Python code for the Airflow DAG with defined tasks. You do not need to write the whole DAG, just include the processing part along with all the diagrams
- BigQuery table schema definition.