Assuming this to be a production environment, the python script will be callable from airflow which runs on scheduled basis for example CRON jobs. Then the airflow runs the python scripts which eventually extracts the data from the API and then splits the data into multiple tables by performing a merge on the existing data in the BigQuery. A merge operation is required to avoid data duplication when the airflow runs twice with the same data. Once the data is available in BigQuery table, it can be transformed into meaningful data by building DBT models according to the stake holder's requirements. Data test can be performed using DBT tests and version control can be performed using Git. Then the transformed data will be transferred to BigQuery and can be used by data visualization tools. Custom tests can be written in DBT to check the uniqueness of data and not null of the fields like login_uuid, login_username, login_password for the given data. Freshness checks can also be perfromed on the data and the stale data can be alerted using slack.