Prepare a case for the team interview that deals with our future architecture: S3 -

Apache Iceberg - Spark & Dremio - Superset.

Example blog article: https://dev.to/alexmercedcoder/end-to-end-basic-data-engineering-

tutorial-spark-dremio-superset-2hgi

As a Data Analyst:

• pick only 2 tasks, we suggest having at least one analytics task with visualizations in

it (eg. Dremio with superset, or Jupyter on iceberg or dremio, or a reporting

implementation of your own with iceberg/dremio as storage).

As a Data Engineer:

• the superset part is optional

• For the API part, the goal is to have a python implementation ready that shows your

skills, feel free to shortcut if it suits to you

Preparation:

• Recreate the stack locally with Docker (as described in the blog article):

• This includes MinIO, Iceberg, Dremio, Jupyter Notebook with PySpark, and

Superset.

• Be prepared to present your setup:

• What did you notice?

• What worked well?

• What didn't work well?

Tasks:

1. Implement an API (using Python/Spark, PySpark)

• Use a data source such as Public APIs.

• Be prepared to present your implementation.

2. Load a larger dataset (>500 MB overall) from a JSON file into Iceberg:

• Hint: You can use the data from Task 1, but also something else/artificial.

• Present your implementation, show/explain the Source JSON and how it is

represented in Iceberg:

▪ How do you structure the data processing?

▪ What edge cases did you encounter?

▪ How did you define the table in Iceberg?

• Did you use partitioning? If so, what definition?

• What storage format did you use (Avro, Parquet, ORC) and

why?

• Please share your observations and thoughts.

3. Make the data available and accessible in Dremio:

• Provide at least one simple SQL query to show how the data appears in

Dremio.

• Create a "reflection" with an aggregation in Dremio.

▪ Share your observations and thoughts on reflections in Dremio.

4. Create at least one chart and a maximum of one dashboard in Superset and

present it.

5. Optional:

• Create at least one chart in Jupyter Notebook (using Pandas, Plotly/Dash,

etc.) and present it.