Jumbotail Assignment- 2

Understanding From This Use case -

1) Based on the use case you provided, the problem can be categorized as a data engineering problem.
2) Specifically, it involves developing a system to track user journey in an e-commerce application and analyzing user behavior at various stages of the application funnel.
3) The main objective of the problem is to implement a solution that can efficiently handle a large volume of events generated by daily active users in the application.
4) The solution needs to capture and process these events, track the percentage of users at each stage of the user journey, and evaluate the performance of different cities based on user engagement. And get data to improve the performance.
5) This problem Includes the aspects of data processing, real-time analytics, and database management to create a system that will analyze and track user journeys with an e-commerce application.

Solution-

To summarize, here's the technology stack we'll use for the implementation:

1) Programming language- python
2) Web Framework: Flask
3) Database: Oracle
4) In-Memory Model: Apache Spark (Pyspark because we are using Python)

Now the instruction that how we should run the code from scratch-

1. Install Dependencies:

- First we need to install Python on the system.
- Install the required libraries by running pip install flask kafka-python pyspark cx_Oracle in your command prompt or terminal.

2. Set Up Oracle Database:

- Then we need to Install and configure Oracle Database on the system.
- Need to Create a new schema or tablespace to store the event data.
- Now the connection details (e.g., host, port, username, password) for accessing the Oracle database.

3. Set Up Apache Kafka:

- Install and configure Apache Kafka on the system.
- Need to Set up the necessary topics in Kafka to handle the event streams.

4. Implement the Components:

Jumbotail Assignment- 2

- Then Set up a Flask application to handle the webhook API.
- Implement the Event Producer class to simulate user journeys and generate events. Use multithreading to represent multiple users.
- Implement the in-memory queue using Kafka to receive and store the events.
- Implement the Queue Consumer to consume events from the Kafka queue, perform necessary calculations, and store the data in the Oracle database.
- Utilize PySpark to process the events, calculate percentages, and evaluate city performance. Ensure to have a Spark cluster set up.

5. Testing and Running the Solution:

- Test the solution by running the Flask application and simulating user journeys with the Event Producer class.
- Monitor the system performance, response times, and data consistency.
- Verifying that events are properly pushed to Kafka, consumed by the Queue Consumer, and stored in the Oracle database.
- Perform queries on the Oracle database to retrieve the necessary data for generating the desired output.

conclusion

A little summary of this problem-

i) We have developed a solution for tracking user journeys in an e-commerce application consisting of components such as the Event Producer, Webhook API, In-memory Queue (Apache Kafka), Queue Consumer, and Oracle Database

ii) The system utilizes Apache Kafka as an in-memory queue for receiving and storing event payloads. These events are then processed and stored in an Oracle database.

iii) The solution also includes a Webhook API implemented with Flask to handle incoming events.

iv) Additionally, we have used Pyspark for data processing, and the chosen database is Oracle. Overall, the solution allows for the tracking of user journeys, calculation of percentage metrics, and evaluation of city performance.