CLOUD COMPUTING | CSE:5333 | FALL 2023
GROUP NO. 08
PROJECT PROPOSAL

DataPrep

Ashish Shah, Manan Arora, Nikhil Nandkumar Mhatre

Application Overview:

DataPrep is a one-stop comprehensive cloud-native platform for data pre-processing and doing basic statistics hosted on Google Cloud. It facilitates users to upload, pre-process, and analyze large datasets for various machine learning and data science tasks. Whether you need to clean data, handle NaN values, normalize numerical data, encode categorical data, detect outliers, or perform other data preprocessing steps, DataPrep offers a user-friendly and scalable solution.

Components Used:

1. Google Kubernetes Engine (GKE)
DataPrep will be a containerized app. We will host the app on the Google Kubernetes engine which internally uses compute engine instances grouped to form a cluster. GKE provides cluster orchestration powered by Kubernetes, it provides the mechanism through which interaction with the cluster is done. The application codebase along with its dependencies will be packaged and will be deployed on the cluster.
There are other options available from other cloud providers like Amazon Elastic Container Service, Azure Kubernetes Service, OpenShift Container platform from Red Hat, etc.

2. Google Cloud Storage
Google Cloud storage is a fully managed storage solution provided in the GCP suite. It will allow DataPrep to store the raw data that is uploaded by the user on the frontend. Also, after the processing is done the processed data will be stored in the Google Cloud Storage until the user requests for data purge. There are other options available from other cloud service providers like Amazon simple storage service(s3), and Azure blob storage for achieving similar functionality.

3. Dataproc/ Dataflow
DataPrep will use either the Dataproc cluster or the Dataflow service from the GCP suite to perform operations on the User's data.

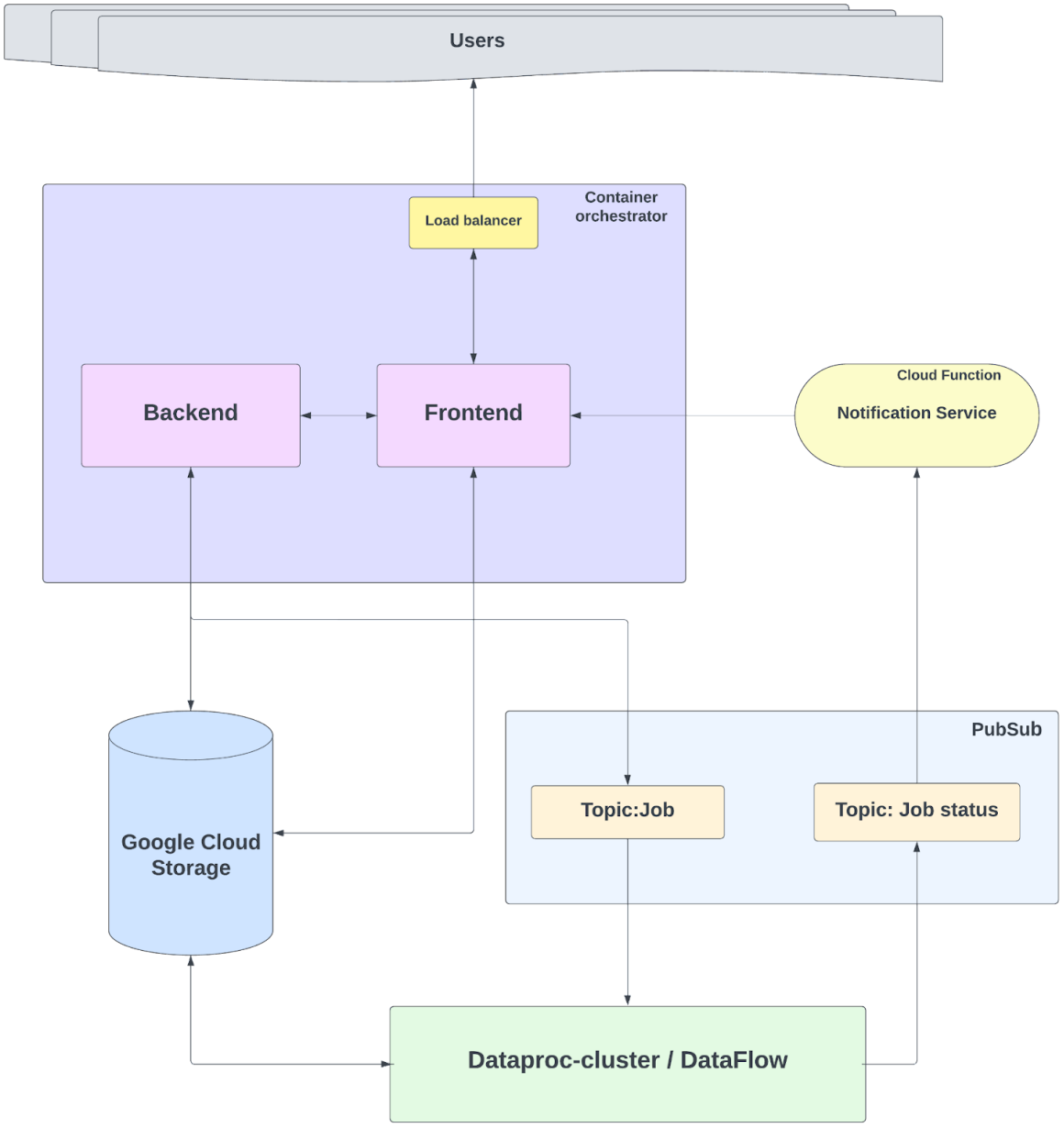
4. Cloud Functions
Google Cloud Function will be used to send notifications to users on the DataPrep and on email. The dedicated Function will subscribe to the Job status topic of the PubSub system to get real-time status updates of completed jobs from the Dataproc/Dataflow

5. Pub/sub

PubSub is a fully managed messaging service offered by Google Cloud. DataPrep uses PubSub for communication between different components of the system.

There are other messaging services like Confluent Kafka which can be used to achieve similar functionality

Architecture:



DataFlow :

1. The User uploads their dataset directly to a fixed location in the Google Cloud Storage.
2. The Data processing unit (Dataproc/ Dataflow) reads the raw data from the storage and performs the user-specified operations.
3. The processed data is again stored in the Google Cloud storage to be accessed by the user.

Communication between components:

1. **User-Frontend:** Using GUI/ website.
2. **Frontend-Backend:** Using Rest API calls.
3. **Backend-Dataproc/dataflow-Notification service:** Using PubSub. Components post and consume messages to and from the topics.

Implementation Details:

Technology / Language / Tools:
• Frontend: HTML, CSS, and Vue.js
• Backend: Python, Django
• Security: Google OAuth2
• Testing: Unit Testing followed by Integration Testing
• Version Control: GitHub
• Compute Engine: GKE, Dataproc, Cloud Functions
• Project Management: JIRA

Frontend-User Interaction:
When a user accesses the front end, they are authenticated using Google Authentication. After authentication Users can interact with the front end hosted on GKE. They can upload their datasets and choose the processing steps they want to apply.

Frontend-Backend Interaction:
The front end communicates with the backend hosted on GKE for various operations using REST API calls.

Backend Processing:
The backend component is responsible for User authentication, authentication token verification, and decoding tokens to extract user information. The backend also processes the users' data processing requests. It creates jobs for the Dataproc-cluster/dataflow. It publishes the job messages to a Pub/Sub topic called 'jobs'. This message includes all the necessary information regarding the user, the user's data (which is hosted on Google Cloud storage), and the type and sequence of operations/transformations that need to be performed on the user's dataset.

Data Processing:
Data processing takes place in an event-driven manner based on the configuration specified by the user in the front end. The Dataproc cluster/ Dataflow subscribes to the PubSub topic jobs to get job details from the backend and performs operations accordingly.

Notifications: DataPrep will facilitate users to get realtime notifications informing them when the processing of the raw data is completed so that they stay informed about the status of their data processing request.

All of the team members are new to the cloud and data processing so we all will be sharing the responsibility to work on all the components.

Timeline:

Phase1: By October 28th 2023
We plan to implement a simple website to perform a single data transformation on the data provided by the user.

Phase2: By December 2nd 2023
We will implement additional functionality and UI allowing users to specify multiple data processing/transformation steps at once, receive notifications on completion of processing tasks, and purge their data from the cloud storage. Finish Testing and bug fixes.

Test Plan:

Unit Testing:
Test individual components like frontend, backend, data processing, and functions and make sure that they are operating as expected.

Integration Testing:
Test the complete system after integration between the frontend, backend, data processing, and other functions components and make sure that they are interacting securely without failure.

End-to-End Testing:
Test the entire workflow from file upload to user receiving the processed data along with the notification system and data transformations.

User Acceptance Testing (UAT):
Users will test the website to ensure the service meets their needs and any logical and flow errors will be detected and fixed here.

Agreement:

We have discussed and agreed upon the specified attributes and ownership as mentioned above in the project proposal of CSE 5333- Cloud Computing.

Ashish Shah – 1002076818 (axs6820@mavs.uta.edu)
Manan Arora – 1002143328 (mxa3328@mavs.uta.edu)
Nikhil Nandkumar Mhatre - 1002122555 (nxm2555@mavs.uta.edu)