

Cloud Computing - CSE 5333-001

Project Proposal

Priyank Gupta - 1001629364

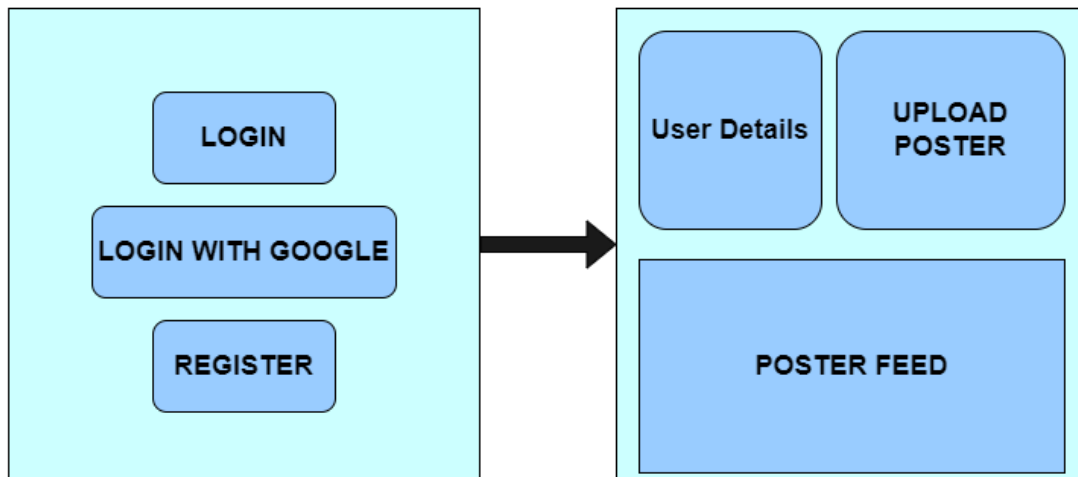
Ravi Rajpurohit - 1002079916

Vedanti Ambulkar - 1001829121

Service/Application Overview -

- Description of how the user will interact with the service

There will be an interactive user interface through a web application enabling users to interact with the service. A login page will be there for authentication to allow only the privileged users. We will use OAuth to let users sign in using a google account. After login, the user will be provided with an interface to upload a photo. This photo can be submitted to the application and stored in a database.



- What does the service enable?

This service will enable us to store the authorized user's data in a database for easy retrieval and store it in a data warehouse for further analysis. The service can be used as a photo library for ongoing or upcoming events. For example - this service can be used to create a library of ongoing or upcoming research studies taking place in a university; creating a digital record of the notice board. Users with specific privileges can upload images or retrieve already uploaded library of images.

Components Used -

- Which components will be used

We are planning to use the following components of Google Cloud Platform -

1. Compute Engine
2. Cloud SQL
3. Cloud Pub/Sub
4. Dataflow
5. Google Storage

- Purpose of each component

The purpose of each component planned to use is mentioned below -

Compute Engine: A compute engine virtual machine instance will be used to set up docker and run the interactive web application for users. A linux image will be used for computer operations. This is the initial and important part of the service as it will enable hosting the application and let users interact to upload or retrieve data.

Cloud SQL: This service will be used to create a relational database to store the user authentication data. The information provided by user while registering will be saved in this database which will further be used to authenticate user privileges.

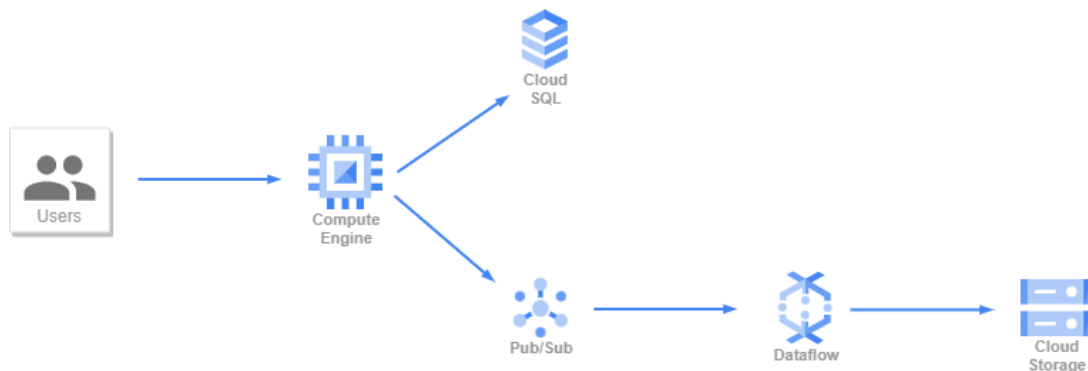
Cloud Pub/Sub: This service will be used to keep track of changes in the storage bucket in the form of messages, for example - to track the added or deleted objects. A notification keeps track of the entire description including the event triggered and the object that was changed. This can be used to notify user about new uploads.

Dataflow: This service will be used to subscribe to the Pub/Sub topic and write to the Cloud Storage bucket. It can also make transformations to the message received by the subscribed topic.

Architecture

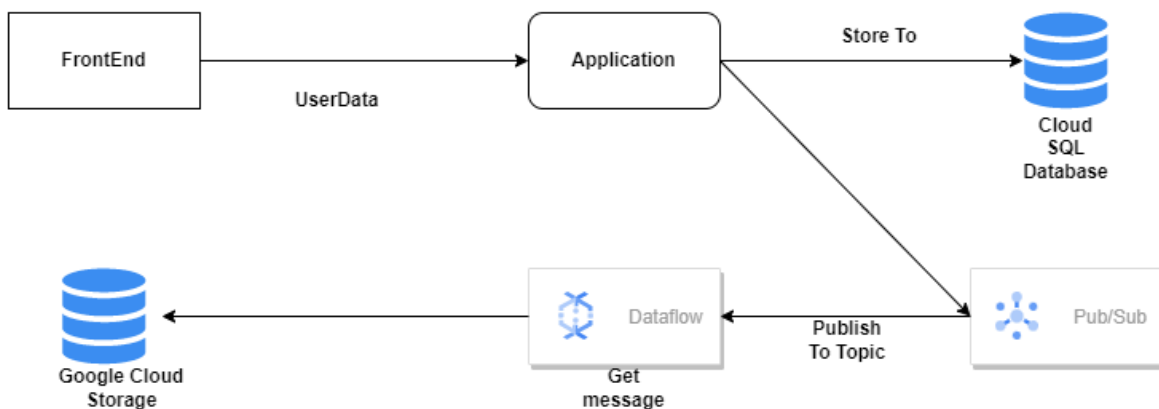
- How is the data flowing through the service?

The architecture of the entire service outlining the data flow as well as showing how each component is connected is -



As shown in the above figure, the user will submit data to the web application running on docker inside compute engine instance. Following are the further steps -

1. User data is stored in the Cloud SQL database
2. Media gets uploaded into Cloud Storage bucket using a Dataflow pipeline which will be triggered whenever a message is published onto the Pub/Sub topic



- How will components connect to each other?

The computer engine instance will be accessible to the users as a service over the internet. It will do the task of inserting data to Cloud SQL database as part of its operations. This will result in publishing a topic to Cloud Pub/Sub where data will be fed to the Dataflow engine. Dataflow will transform the data and upload to Cloud Storage bucket.

Design

- What language, framework, etc. are you using?

We will be using the following language/ framework:

1. Flask - to create web application
2. Python - to code the logic of web application
3. SQL - to insert data in Cloud SQL
4. Python/JavaScript - depending on the Dataflow template

- What are the major code processes/workflows that will control functionality?

The main code processes and workflows for the service are mentioned below -

1. The compute engine instance will host a docker image of the web application enabling us to store user data to the backend database as well as move the data to Pub/Sub topic
2. The Dataflow pipeline will perform the heavy lifting of extracting the data from the pub/sub topic. At last, push the data to a Cloud Storage bucket

- A description of major objects in the design

1. **Compute Engine:** The initiating point of this service is a compute engine instance hosting the web application running inside a docker. The application is accessible over internet allowing users to submit the relevant data. This lets our data flow in two ways - one, to a backend database created using Cloud SQL and another, to a Pub/Sub topic where data gets published
 2. **Cloud SQL:** This will be used to create a backend database to store the profile data submitted by users using our web application.
 3. **Dataflow:** This enables us to perform the required Extract, Transform and Load operation. Dataflow will be used to extract the data from the Pub/Sub topic and then load it to a Cloud Storage bucket for later use
-

Implementation Plan

- What parts of the application and design do you plan to implement in what order?

1. Create the web application and deploy it on the Compute engine VM
2. Create the datastore in Cloud SQL
3. Create a Pub/Sub topic
4. Create a Cloud Storage bucket
5. Create a dataflow template

- Who is responsible for which parts of implementation?

Ravi Rajpurohit: Develop the web application, store data in Cloud SQL table, maintain the documents and assist where ever required

Priyank Gupta: Develop the web application, store data in Cloud SQL table, create Pub/Sub topic, implement Google OAuth if possible

Vedanti Ambulkar: Front end development, handle dataflow, enabling application on google cloud platform and documentation

- How will you know you are on schedule for finishing and a planned milestones timeline?
The plan is to work on a weekly basis. We will divide the project into small tasks and complete small problems each week which will help us to continue and be mindful of the progress. This will enable us to utilize time properly. We will discuss among the team members about the progress and roadblocks, on a weekly or need basis, while defining the goals for next week.

Test Plan

- How will you verify that the service game is working correctly?
 1. Check if the service is accessible by a user
 2. Check if the data gets stored in the Cloud SQL database
 3. Check if the data gets uploaded to Cloud Storage bucket

Declaration

I, Ravi Rajpurohit, agree with the given proposal.

I, Priyank Gupta, agree with the given proposal.

I, Vedanti Ambulkar, agree with the given proposal.