**Assignment Overview**

I have made a google cloud photo sharing app using python as main language in the backend and on front end I have used HTML CSS and JavaScript . The font end of google app is connected with cloud bucket. The cloud application has a login and logout service user will require to create account to get access. user can create directories and sub directories where he can store images or files. user can delete gallery or images from cloud buckets . user has ability to upload and delete images while in the gallery as well as user will be able to upload only JPG or PNG files. all the uploaded images and folders have their thumbnails and user can enlarge the images when he will click on the image thumbnail. all the images are displayed in the grid. the images total size will be displayed in the app once you upload an image and user can see the remaining memory. the user can see the memory limit which is 50Mb. The classes and methods are explained below.

Main.py

def createUserInfo()

This methods takes out the user information like email from the datastore.

def retrieveUserInfo()

This method take out the user detail from the datastore by using the email address as a key.

def blobList()

It uses the local constant file to associate with the bucket.

def addDirectory()

This method adds a new directory to the bucket, by taking the name of the directory from the user.

def addInFile()

Add a file to the user’s folder name. Files are saved in user’s directory only.

def addFile()

Files are added to the bucket but not to a particular folder.

def downloadBlob()

This enables to download the file from the bucket.

def showimage()

Show images generates a URL to display the images. The images are only display for an hour (3600 s)

def deleteBlob()

It takes the user’s folder name and delete the file that need to be deleted.

def deleteDirectBlob()

It takes the directory name from the user and delete the folder.

def getSize()

It gets the total storage size from the user’s folder.

def deleteDir()

This method is a file handler method to delete the required directory from user’s directory.

def addDirectoryHandler()

This methods handles the addDirectory part. It first verifies the token and add it to the user object which has login.

def uploadFileHander()

This method handles the upload file. It uses token and add the file in the user object. It also check the storage size if it exceeds 50MB.

def downloadFile()

It uses the token to download a particular file.

def showGallery()

It uses the user’s directory to open a new page with only gallery pictures. It loops over the user’s directory to retrieve all files within it.

def root()

This method uses token and if it’s a new user, it creates an object user, if it’s an object already it check its files and directories. It then uses the blob\_list to loop over the files and directories. The storage size is also send to the index.html to output storage.

local\_constants.py

In this file there is PROJECT\_NAME which is photoshare and there is PROJECT\_STORAGE\_BUCKET id from google cloud bucket where all data is stored.

**Login Page :**

**Graphical user interface, application

Description automatically generated**This is the main login page where user can user can sign in or login with email and password. if the user has the data inside there he will get same data after login if user does not have any data he will get empty folder to upload files or images .

**Upload Files :**

This button will allow user to choose a file or image from the local device and he can upload to the cloud folder .

Graphical user interface, application

Description automatically generated

**Making a directory :**

This helps user to create a new directory where he can upload files or photos in this directory .

Graphical user interface, application

Description automatically generated

**Photo Display :**

Graphical user interface, application

Description automatically generatedwhen user will upload images it will look this in the cloud storage .

**Folder Display :**

Graphical user interface, diagram

Description automatically generatedThe name of the Folders are display in horizontal form. and user can see the folder name written under the folder logo .

**Storage :**

When user will upload the image he can see the total image size and user can see the remain memory in the storage .

Graphical user interface, text, application

Description automatically generated

**Name display and Sign out :**

When a user will sign in he can see his name and sign out option on the top left corner if user will click the sign out button he will have to login again .

Graphical user interface

Description automatically generated

**Open Gallery :**

Under the images there is gallery button if user click that button if will bring to photo library where he can see all uploaded images .

Graphical user interface

Description automatically generated

**Gallery Page :**

Graphical user interface, application

Description automatically generatedUser can see all images in photo library and he can see total remain memory in the storage .

Index2.html

It has a navigation bar which contain the name of the user that has logged in and a signout button.

This file is the template for the front end. It has place to upload images, make directory, display images from user’s folder, display folders inside user’s folder. It contains the total storage used out of 50MB.

A gallery button is also placed at the bottom of the page that goes to the gallery page.

Gallery.html

This template displays the images from the user’s folders. It also displays the amount of storage used.

script.js

This file contain the listener to load the sign in page and sign out page. It displays the sign in page and verifies the user and token.

Structure

The image below shows the model of the app.

Diagram

Description automatically generated