**GEMINI AI Image Analysis App Documentation**

This Streamlit app allows users to input a text prompt and upload an image. It uses the GEMINI API to generate a response based on the provided input and image. The response is then displayed on the app.

**Dependencies**

* streamlit: For creating the web app.
* PIL (Pillow): For image processing.
* google.generativeai: For interacting with the GEMINI API.
* os: For environment variable management.

**Environment Variables**

* GEMINI\_API\_KEY: The API key for accessing the GEMINI API. Set this in your environment before running the app.

**Code Explanation**

1. **Imports and Environment Setup**

import streamlit as st

import pathlib

import textwrap

from PIL import Image

import os

os.environ['GEMINI\_API\_KEY']='YOUR\_API\_KEY\_HERE'

* Imports necessary libraries.
* Sets the GEMINI API key in the environment variable.

1. **API Configuration**

import google.generativeai as genai

genai.configure(api\_key=os.environ['GEMINI\_API\_KEY'])

* Configures the GEMINI API with the API key.

1. **Function Definition**

def get\_gemini\_response(input, image):

model=genai.GenerativeModel('gemini-1.5-flash')

if input != "":

response = model.generate\_content([input, image])

else:

response = model.generate\_content(image)

return response.text

* get\_gemini\_response(input, image): Function that interacts with the GEMINI API to generate a response. It uses the gemini-1.5-flash model to generate content based on the provided input text and image.

1. **Streamlit App Setup**

st.set\_page\_config(page\_title='Image Creation')

* Sets the title of the Streamlit app.

1. **App Layout**

st.header('GEMINI AI IMAGE APP ANALYSIS')

input = st.text\_input('Input Prompt :', key='input')

upload\_file = st.file\_uploader('Choose an image', type=['jpg', 'jpeg', 'png'])

* Displays a header and creates an input field for text prompts and an uploader for images.

1. **Image Display**

image = ""

if upload\_file is not None:

image = Image.open(upload\_file)

st.image(image, caption='Upload Image', use\_column\_width=True)

* Displays the uploaded image with a caption if an image file is provided.

1. **Submit Button and Response Display**

submit = st.button('Explain brief about image')

if submit:

response = get\_gemini\_response(input, image)

st.subheader('The response is ')

st.write(response)

* Creates a submit button. When clicked, it calls get\_gemini\_response() with the input text and image, then displays the generated response.

**Running the App**

To run the app, execute the script using Streamlit:

streamlit run google vision.py

**Usage**

**1.**  **Interact with the App:**

* Open the app in your web browser (Streamlit will provide a local URL).
* Enter a text prompt in the "Input Prompt" field.
* Upload an image by selecting a file through the file uploader.
* Click the "Explain brief about image" button to get a response from the GEMINI AI based on the text prompt and uploaded image.

**2. View Results:**

* The app will display the uploaded image and show the generated response below it.