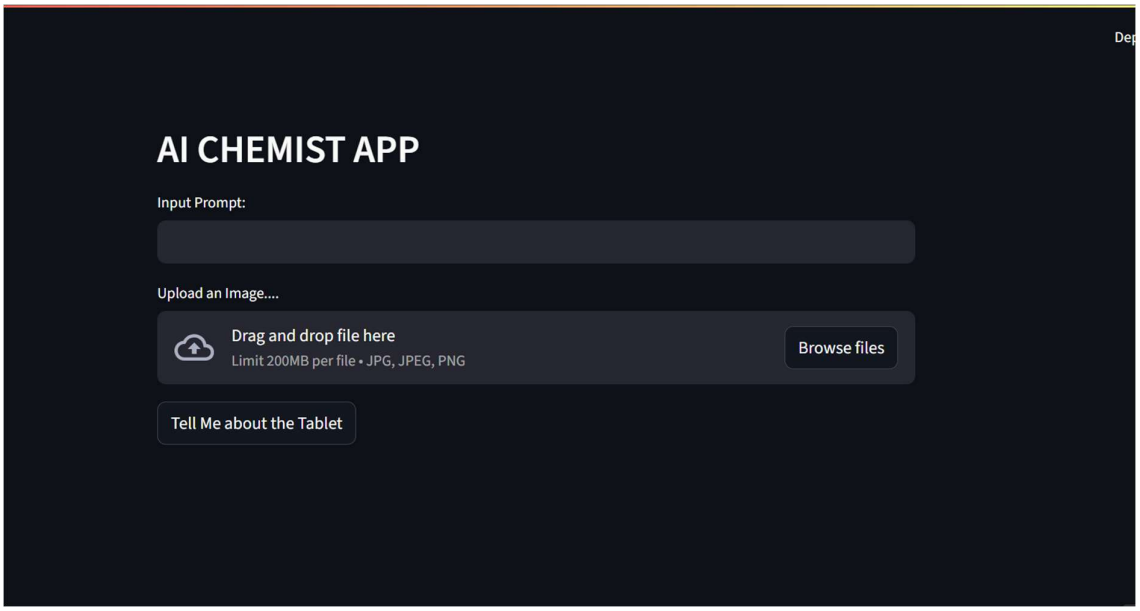
Title: AI Chemist: Pioneering the Future of Chemical Science with Gemini Vision Pro

Abstract : AI Chemist is a state-of-the-art mobile application designed to provide experimental recommendations and customized chemical solutions using the advanced Gemini Pro model. This program provides personalized experiment designs, chemical synthesis methods, and intelligent data analysis by utilizing artificial intelligence to assess user input, laboratory circumstances, and research objectives. Through intelligent, data-driven advice and support, AI Chemist aims to improve the efficiency and innovation of chemistry research.

AI Chemist is a mobile application that leverages artificial intelligence to provide personalized chemical experiment recommendations, synthesis routes, and data analysis. The system integrates a large language model (Gemini Pro) through API calls to understand user objectives and lab constraints, offering customized experimental designs. This report discusses the motivation, architecture, implementation, and potential impact of AI Chemist in research efficiency and innovation.

Introduction: Chemical solutions that are clever and adaptable are becoming more and more necessary in both industrial and academic labs. By providing an interactive mobile platform that can provide personalized chemical procedures and recommendations, AI Chemist aims to close this gap.



Proposed Methodology : Pharmacists, chemists, and other medical personnel frequently need to be able to visually recognize tablets or medications in addition to knowing their uses. Manual identification is laborious and prone to mistakes. Using multimodal AI that interprets visuals and textual cues to provide medicinal insights, this program seeks to automate this procedure.

- **Tools and Technologies Used :**

Streamlit	**Building the web-based UI**
Dotenv	**Loading API keys securely from .env**
PIL(image)	**Handling image uploads**
Google.generativeai	**Interfacing with Gemini multimodal LLM**
Gemini Pro Vision	**Analyzing image + prompt to generate a detailed response**
.env	**Storing sensitive environment variables (API keys)**

- **Working Principle :**

The application accepts:

- A **textual prompt**
- An **image upload** (e.g., tablets or medicines)

It then uses **Gemini Flash 2.0 (multimodal)** to interpret both and generate contextual pharmaceutical information.

- **Python Code :**

app.py > ...

```python
### Health Management APP
from dotenv import load_dotenv

load_dotenv() ## load all the environment variables
import streamlit as st
import os
import google.generativeai as genai
from PIL import Image

genai.configure(api_key=os.getenv("GOOGLE_API_KEY"))

## Function to load Google Gemini Pro Vision API And get respor
```

Load the variables in the environment.

Build the user interface using streamlit.

To access the generative multimodal model, utilize Gemini's SDK. PIL

manages the loading and displaying of images.

```
## Function to load Google Gemini Pro Vision API And get response

def get_gemini_repsonse(input,image,prompt):
    model=genai.GenerativeModel('gemini-pro-vision')
    response=model.generate_content([input,image[0],prompt])
    return response.text
```

- Initializes a Gemini model instance.
- Accepts a prompt and image data.
- Sends them to the model for content generation.
- Returns the textual response from Gemini

```
def input_image_setup(uploaded_file):
    # Check if a file has been uploaded
    if uploaded_file is not None:
        # Read the file into bytes
        bytes_data = uploaded_file.getvalue()

        image_parts = [
            {
                "mime_type": uploaded_file.type,  # Get the mime type of the uploaded file
                "data": bytes_data
            }
        ]
        return image_parts
    else:
        raise FileNotFoundError("No file uploaded")

##initialize our streamlit app
```

- Converts the uploaded image to byte data compatible with Gemini.
- Wraps it with MIME type for the Gemini SDK.
- Returns a format suitable for the generate_content() method.

```
##initialize our streamlit app
input_prompt = """
you are an expert pharamaceutical/chemist where you need to see the tablets from
            the input image and ,  also provide the details of every drug / tablets items with  below format

            1.Examine the image carefully and identify the tablets depicted.
            2.Describe the uses and functionalities of each tablet shown in the image.
            3.Provide information on the intended purposes,features and typical applications of the tablets.
            4.If possible , include any notable specifications or distinguishing characteristics of each tablet.
            5.Ensure clarity and conciseness in your descriptions, focusing on key details and distinguishing facts
            ----
"""
```

- The system prompt instructs Gemini to act like an expert chemist.
- Gives clear, structured tasks for image analysis:
    - Identify each tablet
    - Describe uses

- Provide distinguishing features

```
##initialize our streamlit app

st.set_page_config(page_title="AI Chemist App")

st.header("AI Chemist App")
input=st.text_input("Input Prompt: ",key="input")
uploaded_file = st.file_uploader("Choose an image...", type=["jpg", "jpeg", "png"])
image=""
if uploaded_file is not None:
    image = Image.open(uploaded_file)
    st.image(image, caption="Uploaded Image.", use_column_width=True)


submit=st.button("Tell me")
```

The code image provided demonstrates how to build up the AI Chemist App's user interface with Streamlit. After giving the app page a unique title, it displays a header with the words "AI Chemist App." A text input field on the interface allows users to optionally submit a prompt to direct the analysis. Users can choose an image file (limited to jpg, jpeg, or png formats) via a file uploader widget; this file usually shows a picture of tablets or medications. The code ensures that the image adapts responsively to the column width for improved viewing by using the PIL library to open and show it immediately within the application if an image is uploaded. Lastly, users can submit their information (text and image) for additional processing by clicking the "Tell me" button. The AI Chemist workflow's data collecting and preview stages are successfully formed by this section of the code, which also produces an easy-to-use user interface for interactive pharmacological analysis.

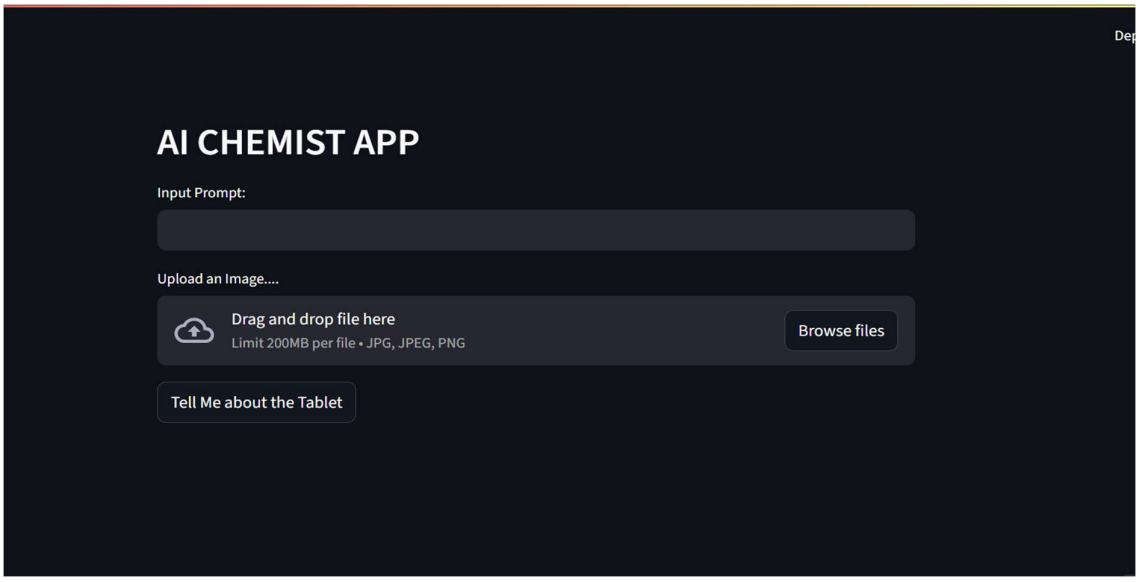
```
## If submit button is clicked

if submit:
    image_data=input_image_setup(uploaded_file)
    response=get_gemini_repsonse(input_prompt,image_data,input)
    st.subheader("The Response is")
    st.write(response)
```

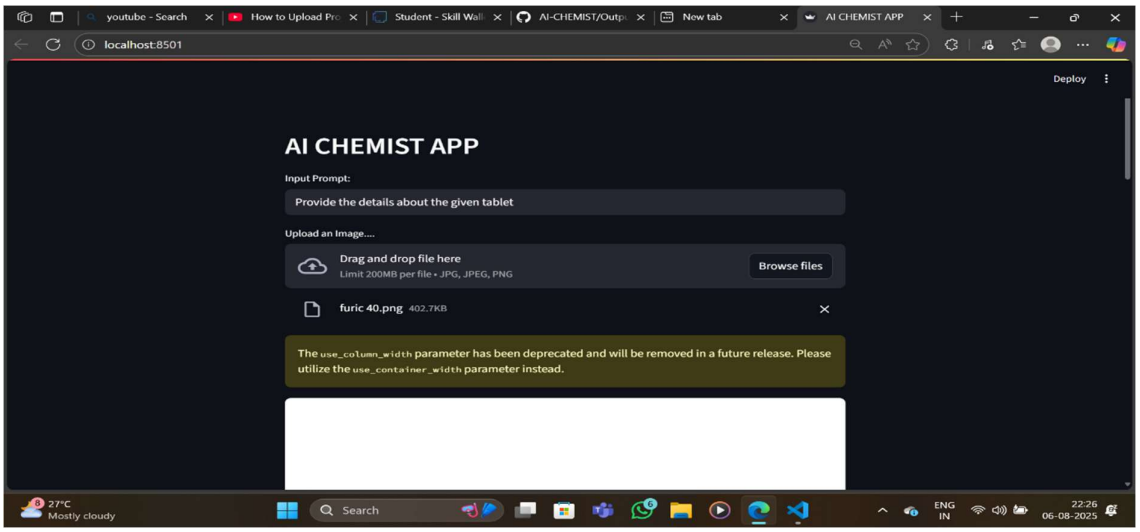
This code initializes a Streamlit application titled "AI Nutritionist App" by setting the page title and creating the app's header. It includes a text input field for users to enter a custom prompt and a file uploader for users to upload an image in JPG, JPEG, or PNG format. If an image is uploaded, it is opened using the PIL library and displayed within the app with a caption. A button labeled "Tell me the total calories" is also provided, which users can click to trigger the application's

functionality for analyzing the uploaded image to calculate and display the total calorie content of the food items depicted.

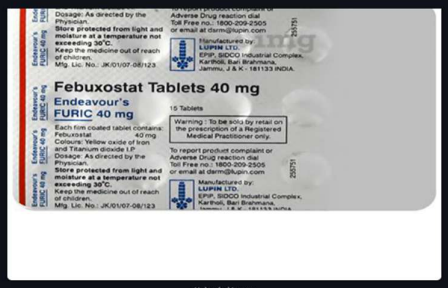
Output :



The picture displays the AI Chemist App's main user interface, which was created using Streamlit. It has an image uploader that enables drag-and-drop or file searching for adding tablet photos (limited to JPG, JPEG, and PNG formats) and a simple, dark-themed interface with a "Input Prompt" text box for optional user instructions. An obvious "Tell Me about the Tablet" button beneath the uploader starts the AI analysis procedure. The user interface is easy to use, intuitive, and made to seamlessly lead users from entering data to activating the AI model.



localhost:8501



Uploaded Image

Tell Me about the Tablet

The Response is

Here's a detailed description of the medication presented in the image:

1. Tablet Identification:

- The tablets are identified as **Endeavour's FURIC 40 mg**. The active ingredient is **Febuxostat**.

2. Uses and Functionalities:

- **Febuxostat** is a xanthine oxidase inhibitor. It is used to lower uric acid levels in the blood.
- The primary use of Febuxostat is in the **treatment of gout**. It helps to prevent gout attacks by reducing the formation of uric acid crystals in the joints.
- Febuxostat is typically prescribed for individuals with hyperuricemia (high uric acid levels) when other treatments, like allopurinol, are not suitable or effective.

3. Intended Purposes, Features, and Applications:

- **Intended Purpose:** To manage hyperuricemia and prevent/reduce the frequency of gout flares.
- **Features:** Film-coated tablets, containing 40 mg of Febuxostat.

youtube - Search | How to Upload Pro | Student - Skill Wall | AI-CHEMIST/Outpu | New tab | AI CHEMIST APP

localhost:8501

Tell Me about the Tablet

The Response is

Here's a detailed description of the medication presented in the image:

1. Tablet Identification:

- The tablets are identified as **Endeavour's FURIC 40 mg**. The active ingredient is **Febuxostat**.

2. Uses and Functionalities:

- **Febuxostat** is a xanthine oxidase inhibitor. It is used to lower uric acid levels in the blood.
- The primary use of Febuxostat is in the **treatment of gout**. It helps to prevent gout attacks by reducing the formation of uric acid crystals in the joints.
- Febuxostat is typically prescribed for individuals with hyperuricemia (high uric acid levels) when other treatments, like allopurinol, are not suitable or effective.

3. Intended Purposes, Features, and Applications:

- **Intended Purpose:** To manage hyperuricemia and prevent/reduce the frequency of gout flares.
- **Features:** Film-coated tablets, containing 40 mg of Febuxostat.
- **Typical Application:** Long-term management of gout. It is usually taken once daily. It is important to continue taking Febuxostat even when you are not experiencing a gout flare.

4. Specifications and Distinguishing Characteristics:

- **Dosage:** 40 mg per tablet. The dosage is as directed by the physician.
- **Appearance:** Tablets contain "Yellow oxide of Iron and Titanium dioxide I.P" as colour ingredients.
- **Manufacturer:** Lupin Ltd.
- **Storage:** The packaging specifies storage protected from light and moisture at a temperature not exceeding 30°C.
- **Prescription Only:** The packaging clearly states "To be sold by retail on the prescription of a Registered Medical Practitioner only."

5. Important Considerations (Based on Packaging):

- **Dosage:** Must be taken as directed by the physician.
- **Storage:** Proper storage is essential to maintain the drug's efficacy.
- **Keep out of Reach of Children:** This is a standard safety precaution for all medications.
- **Adverse Reactions:** The packaging provides contact information (toll-free number and email) to report product complaints or adverse drug reactions.