# Nutrition App Using Gemini Pro

This is a project on creating a nutrition app using gemini pro which is a comprehensive guide to healthy eating and well-being. Nutritionist AI is an innovative mobile application designed to provide personalized dietary recommendations and nutritional advice using the advanced capabilities of the Gemini Pro model. The app leverages artificial intelligence to analyze user data, dietary preferences, and health goals, delivering tailored meal plans, nutritional insights, and wellness tips. The primary aim of Nutritionist AI is to promote healthier eating habits and improve overall well-being through intelligent and data-driven recommendations.

Here's a review on how this app works:
- User interacts with the UI to enter the input.
- User input is collected from the UI and transmitted to the backend using the Google API key.
- The input is then forwarded to the Gemini Pro pre-trained model via an API call.
- The Gemini Pro pre-trained model processes the input and generates the output.
- The results are returned to the frontend for formatting and display.
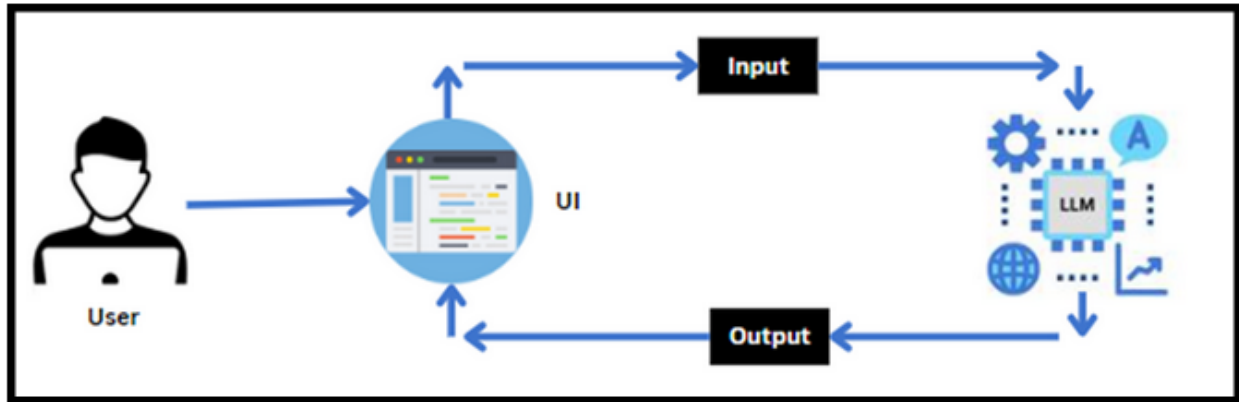
I have created a list of files in order to run this files and here is small algorithm to run this application:

- Requirements Specification
  - Create a requirements.txt file to list the required libraries.
  - Install the required libraries
- Initialization of Google API Key
  - Generate Google API Key
  - Initialize Google API Key
- Interfacing with Pre-trained Model
  - Load the Gemini Pro pre-trained model
  - Implement a function to get gemini response
  - Implement a function to read PDF content
  - Write a prompt for gemini model
- Model Deployment
  - Integrate with Web Framework
  - Host the Application

A small summary of the uses of each of these files:

- .env file: It securely stores the Google API key.
- app.py: It serves as the primary application file housing both the model and Streamlit UI code.
- requirements.txt: It enumerates the libraries necessary for installation to ensure proper functioning.

List of files I have created:

SB [CODESPACES: HIDDEN VAMPIRE]

> venv
- .env
- app.py
- README.md   ↓M
- requirements.txt
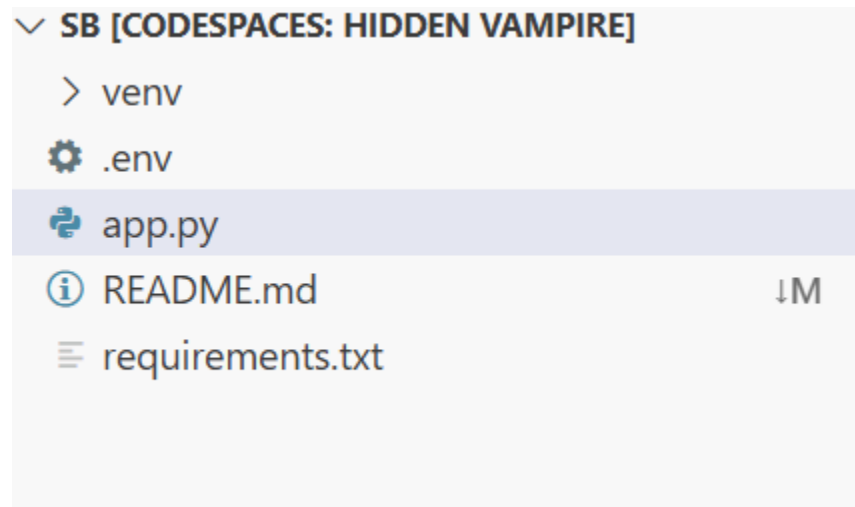
requirements.txt:

requirements.txt

```
1    streamlit
2    google-generativeai
3    python-dotenv
4
```

Initialization of Google API Key:

```
import pandas as pd

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

Importing libraries:

```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
import pandas as pd
```

Implementing get gemini response function:

```python
def get_gemini_repsonse(input_prompt,image_data):
    model=genai.GenerativeModel('gemini-1.5-flash')
    response=model.generate_content([input_prompt,image_data[0]])
    return response.text

def input image setup(uploaded file):
```

Implementing image reader function:

```python
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")
```

List of things gemini should respond with:

- Health status
- Nutritional breakdown
- Total calories
- Conclusion

Run the app using streamlit command:

```
(/home/codespace/sb/venv) @Anushree-R ➜~/sb (main) $ streamlit run app.py

Collecting usage statistics. To deactivate, set browser.gatherUsageStats to false.


  You can now view your Streamlit app in your browser.

  Local URL: http://localhost:8501
  Network URL: http://10.0.3.144:8501
  External URL: http://4.240.39.196:8501
```

ⓘ Your application running on port 8501 is available. See all
forwarded ports

Open in Browser   Make Public

Gemini Health App

Input Prompt:

Choose an image...

Drag and drop file here
Limit 200MB per file • JPG, JPEG, PNG   Browse files

Analyze Meal

Uploaded Image.

Analyze Meal

The provided output by gemini:

FitGen-Ai Analysis

The image contains a bowl of vegetable fried rice. It is difficult to determine the exact quantities of each ingredient, so the calorie and nutritional breakdown is an estimation based on a typical serving size of vegetable fried rice.

Item 1: Fried Rice - 350 calories Health Status: **Unhealthy** - While the vegetables in fried rice offer some nutrients, the high amount of oil and sodium used in the cooking process makes it an unhealthy choice. Nutritional Breakdown: Protein: 10% Fat: 40% Carbohydrates: 45% Fiber: 5%

Total Calories: 350 calories

Conclusion:

The fried rice is primarily composed of carbohydrates and fat, with a lower percentage of protein and fiber. This meal lacks a good balance of nutrients and is not recommended for regular consumption. The high fat and sodium content contribute to a less healthy dietary choice. For a healthier alternative, consider substituting white rice with brown rice, using less oil in the cooking process, and incorporating more protein sources like tofu or lean meat.

It provided the health status, nutritional breakdown, total calrories and conclusion as expected

FitGen-Ai Analysis

The image contains a bowl of vegetable fried rice. It is difficult to determine the exact quantities of each ingredient, so the calorie and nutritional breakdown is an estimation based on a typical serving size of vegetable fried rice.

Item 1: Fried Rice - 350 calories Health Status: **Unhealthy** - While the vegetables in fried rice offer some nutrients, the high amount of oil and sodium used in the cooking process makes it an unhealthy choice. Nutritional Breakdown: Protein: 10% Fat: 40% Carbohydrates: 45% Fiber: 5%

Total Calories: 350 calories

Conclusion:

The fried rice is primarily composed of carbohydrates and fat, with a lower percentage of protein and fiber. This meal lacks a good balance of nutrients and is not recommended for regular consumption. The high fat and sodium content contribute to a less healthy dietary choice. For a healthier alternative, consider substituting white rice with brown rice, using less oil in the cooking process, and incorporating more protein sources like tofu or lean meat.

Hence the app works as expected.