Task – Image Generation

Day – 6

app.py

```
from flask import Flask, render_template, request

import requests

import os

import json

from datetime import datetime

from PIL import Image

from io import BytesIO

import base64

app = Flask(__name__)

# Directories

SAVE_DIR = "static/generated_images"

META_DIR = "metadata"

os.makedirs(SAVE_DIR, exist_ok=True)

os.makedirs(META_DIR, exist_ok=True)

# Your Hugging Face API key

HF_API_TOKEN = "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"

# Stable Diffusion model endpoint

HF_MODEL_URL = "https://api-inference.huggingface.co/models/black-forest-labs/FLUX.1-schnell"

# Headers

headers = {"Authorization": f"Bearer {HF_API_TOKEN}"}

def generate_image(prompt):

    HF_MODEL_URL = "https://api-inference.huggingface.co/models/black-forest-labs/FLUX.1-schnell"

    headers = {

        "Authorization": f"Bearer {HF_API_TOKEN}"
    }
    response = requests.post(
        HF_MODEL_URL,
        headers=headers,
        json={"inputs": prompt},
    )
    if response.status_code != 200:
        raise Exception(f"Failed to generate image: {response.status_code} {response.text}")
    # Save image
    image = Image.open(BytesIO(response.content))
    timestamp = datetime.now().strftime("%Y%m%d_%H%M%S")
    filename = f"{timestamp}.png"
    image_path = os.path.join(SAVE_DIR, filename)
    image.save(image_path)
    metadata = {
        "prompt": prompt,
        "filename": filename,
        "timestamp": timestamp,
        "image_path": image_path
    }
    with open(os.path.join(META_DIR, f"{timestamp}.json"), "w") as f:
        json.dump(metadata, f, indent=4)
    return image_path, metadata
def load_gallery():
    gallery = []
    for file in sorted(os.listdir(META_DIR), reverse=True):
        if file.endswith(".json"):
            with open(os.path.join(META_DIR, file)) as f:
                meta = json.load(f)
                gallery.append(meta)
    return gallery
@app.route("/", methods=["GET", "POST"])
def index():
    image_path = None
    metadata = None
    error = None
    if request.method == "POST":
        prompt = request.form.get("prompt")
        try:
            image_path, metadata = generate_image(prompt)
        except Exception as e:
            error = str(e)
    gallery = load_gallery()
    return render_template("index.html", image_path=image_path, metadata=metadata, error=error, gallery=gallery)
if __name__ == "__main__":
    app.run(debug=True)
```

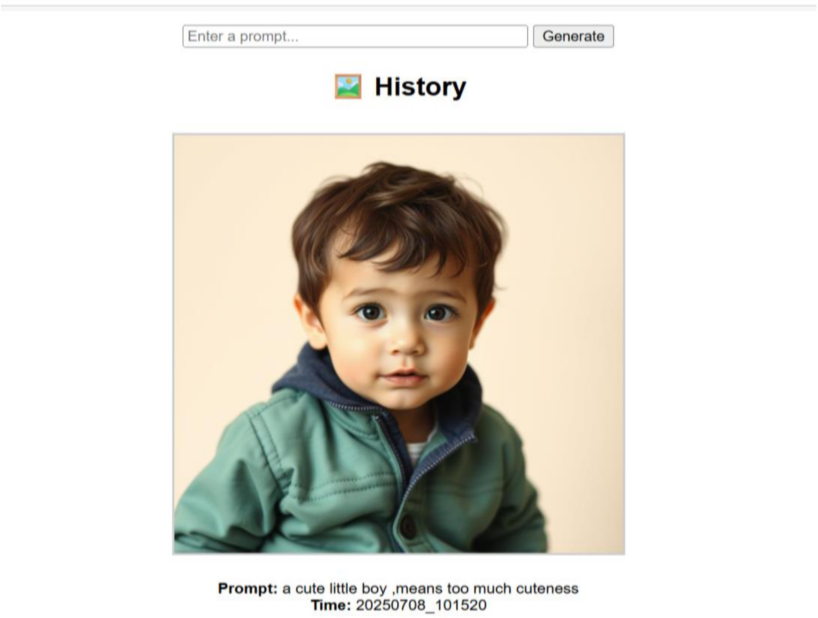
index.html –

```
<!DOCTYPE html>
<html>
<head>
    <title>AI Image Generator - Hugging Face</title>
    <style>
        body { font-family: Arial; text-align: center; }
        img { max-width: 400px; margin: 10px; border: 2px solid #ccc; }
        .meta { margin: 10px; font-size: 14px; }
    </style>
```

```
</head>
<body>
    <h1>🎨 Hugging Face AI Image Generator</h1>
    <form method="post">
        <input type="text" name="prompt" placeholder="Enter a prompt..." required style="width: 300px;">
        <button type="submit">Generate</button>
    </form>
    {% if error %}
        <p style="color: red;">{{ error }}</p>
    {% endif %}
    {% if image_path %}
        <h2>Generated Image</h2>
        <img src="{{ image_path }}" alt="Generated Image">
        <div class="meta">
            <strong>Prompt:</strong> {{ metadata.prompt }}<br>
            <strong>Time:</strong> {{ metadata.timestamp }}
        </div>
    {% endif %}
    <h2>🖼️ History</h2>
    {% for item in gallery %}
        <div>
            <img src="{{ item.image_path }}" alt="Image">
            <div class="meta">
                <strong>Prompt:</strong> {{ item.prompt }}<br>
                <strong>Time:</strong> {{ item.timestamp }}
            </div>
        </div>
    {% endfor %}
```

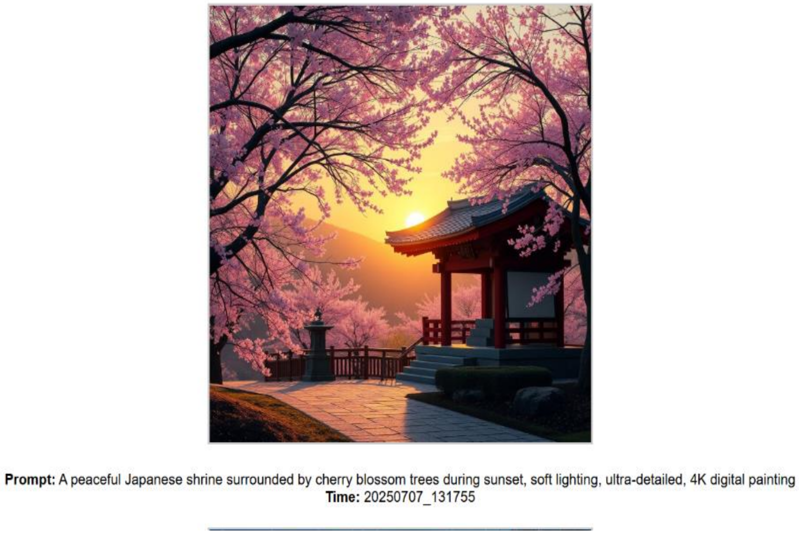
```
</body>
</html>
```

Output -



🖼️ History

Prompt: a cute little boy ,means too much cuteness
Time: 20250708_101520



Prompt: A peaceful Japanese shrine surrounded by cherry blossom trees during sunset, soft lighting, ultra-detailed, 4K digital painting
Time: 20250707_131755