

Project 54: Flashcard Generator

Description:

This smart assistant creates flashcards from topics you input—breaking down complex subjects into Q&A format. Ideal for spaced repetition, exam prep, or bite-sized learning across any subject: science, history, language, or coding.

flashcard_generator.py

```
import openai
import os
import gradio as gr

# Load the OpenAI API key
openai.api_key = os.getenv("OPENAI_API_KEY")

# Function to generate flashcards
def generate_flashcards(topic, number_of_cards):
    # Prompt AI to act like a flashcard creator
    messages = [
        {
            "role": "system",
            "content": (
                f"You are a flashcard generator. Create {number_of_cards} flashcards "
                f"on the topic: '{topic}'. Use simple and clear language. Each "
                "Return them as a numbered list."
            )
        },
        {
            "role": "user",
            "content": f"Topic: {topic}"
        }
    ]

    try:
        # Call OpenAI to get flashcards
        response = openai.ChatCompletion.create(
            model="gpt-3.5-turbo", # GPT-4 optional
```

```

        messages=messages
    )

    # Return the list of flashcards
    return response["choices"][0]["message"]["content"].strip()

except Exception as e:
    return f"Error: {str(e)}"

# Gradio interface for generating flashcards
iface = gr.Interface(
    fn=generate_flashcards,
    inputs=[
        gr.Textbox(label="Topic (e.g. Photosynthesis, Python Loops, World War
        gr.Slider(1, 10, step=1, label="Number of Flashcards")
    ],
    outputs="text",
    title="📁 Flashcard Generator",
    description="Enter a topic and number of cards, and I'll generate study fl
)

# Launch the flashcard builder
iface.launch()

```