Project 29: Daily Productivity Planner

Description:

This smart planner helps you organize your day by turning your priorities, schedule, and goals into a focused daily plan. It includes time blocks, breaks, habits, and even motivating tips. Great for students, professionals, or anyone who wants to beat procrastination.

`daily_productivity_planner.py`

```python
import openai
import os
import gradio as gr

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

# Function to generate a daily productivity plan
def create_daily_plan(user_input):
    # System prompt that shapes the assistant into a productivity coach
    messages = [
        {
            "role": "system",
            "content": (
                "You are a personal productivity coach. Based on the user's ta
                "create a well-structured daily plan. Include time blocks, bre
                "Keep it realistic, encouraging, and clear."
            )
        },
        {
            "role": "user",
            "content": f"Here's what I need to get done today:\n{user_input}"
        }
    ]

    try:
        # Make request to OpenAI's API
        response = openai.ChatCompletion.create(
            model="gpt-3.5-turbo",  # GPT-4 works too
```

```python
            messages=messages
        )

        # Return the daily schedule
        return response['choices'][0]['message']['content'].strip()

    except Exception as e:
        # Return error if something goes wrong
        return f"Error: {str(e)}"


# Gradio interface for productivity planner
iface = gr.Interface(
    fn=create_daily_plan,                              # Function to run
    inputs=gr.Textbox(lines=4, placeholder="e.g. 3 meetings, write a report, 1
    outputs="text",                                    # Output the structur
    title="🧠 Daily Productivity Planner",              # App title
    description=(
        "Type your daily priorities or tasks, and I'll generate a focused, rea
        "Try: 'Study Python, walk dog, two meetings, and groceries' or 'Finish
    )
)


# Launch the planner
iface.launch()
```