Project 64: Time Blocking Planner

Description:

This tool turns your to-do list into a time-blocked daily schedule. It helps you plan your day hour-by-hour, ensuring focused work sessions, buffer breaks, and smart energy use. Great for deep work, students, or busy professionals.

time_blocking_planner.py

```
import openai
import os
import gradio as gr

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

# Function to generate time-blocked plan
def generate_time_blocks(tasks, start_hour, end_hour):
    # Prompt AI to act like a time management coach
    messages = [
        {
            "role": "system",
            "content": (
                "You are a smart scheduling assistant. Take the list of tasks
                f"{start_hour}:00 and {end_hour}:00. Each task should be match
                "Include 5–10 min buffer breaks. Return the result in a clear
            )
        },
        {
            "role": "user",
            "content": f"My tasks: {tasks}"
        }
    ]

    try:
        # Call OpenAI API
        response = openai.ChatCompletion.create(
            model="gpt-3.5-turbo",
            messages=messages
        )

        # Return time-blocked schedule
        return response["choices"][0]["message"]["content"].strip()

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

# Gradio UI for input/output
iface = gr.Interface(
    fn=generate_time_blocks,
    inputs=[
        gr.Textbox(label="Tasks with durations (e.g. Write report - 1h, Check
        gr.Slider(minimum=5, maximum=12, step=1, label="Start Hour (24hr forma
        gr.Slider(minimum=12, maximum=22, step=1, label="End Hour (24hr format
    ],
    outputs="text",
    title="⏰ Time Blocking Planner",
    description="Enter your tasks and available hours. I'll break your day int
)

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