

Project 15: Meeting Notes Summarizer

Description:

This tool takes raw meeting notes or transcriptions and summarizes them into key decisions, action items, and major discussion points. It's ideal for professionals who want clean summaries without reading through messy minutes or call logs.

meeting_notes_summarizer.py

```
import openai
import os
import gradio as gr

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

# Function to summarize raw meeting notes
def summarize_meeting_notes(notes_text):
    # System prompt that sets the AI's behavior
    messages = [
        {
            "role": "system",
            "content": (
                "You are an expert executive assistant. Read raw meeting notes  

                decisions made, and action items. Structure the output clearly  

                Be concise and formal, suitable for business documentation."
            )
        },
        {
            "role": "user",
            "content": f"Please summarize the following meeting notes:\n\n{notes_text}"
        }
    ]

    try:
        # Call OpenAI's API to generate the summary
        response = openai.ChatCompletion.create(
            model="gpt-3.5-turbo", # Swap to gpt-4 if desired
```

```

        messages=messages
    )

    # Return the formatted summary
    return response["choices"][0]["message"]["content"].strip()

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

# Gradio interface for pasting meeting notes
iface = gr.Interface(
    fn=summarize_meeting_notes,                # Summarization function
    inputs=gr.Textbox(lines=15, placeholder="Paste your meeting notes or transcript"),
    outputs="text",                            # Display summarized output
    title="📝 Meeting Notes Summarizer",        # App title
    description=(
        "Paste in raw meeting notes, messy bullet points, or a transcript. "
        "Get a clean summary with decisions, action items, and highlights."
    )
)

# Launch the app
iface.launch()

```