[AI-enabled Software Engineer] Home Task: AI Writing Assistant

🎯 Objective

Build a single-page application (SPA) that uses a Large Language Model (LLM) to rephrase user input into different writing styles. The app should provide a clean user experience and demonstrate your ability to work with full-stack technologies and AI APIs.

✅ Base Requirements

- Tech Stack:
  - Frontend: ReactJS or others
  - Backend: Python (FastAPI), Java, .NET, or Node.js
- Functionality:
  - The user enters text into a text field and clicks a "**Process**" button.
  - The backend sends the input to an LLM API (e.g., OpenAI, Claude).
  - Once the LLM response is received, the frontend displays rephrased versions of the input in different writing styles:
    - Professional
    - Casual
    - Polite
    - Social-media
  - While processing is ongoing:
    - The user cannot initiate another request on this page.
    - The user can click a "Cancel" button to stop the current process.

🌟 Bonus (Recommended) Features

- Streaming Output: Display the LLM's response in real-time as it arrives (e.g., word-by-word or sentence-by-sentence).
- Display each writing style in its own separate text area:
  Professional:
  [Rephrased text]

  Casual:

[Rephrased text]

...

- UI/UX is clean and resembles a production-grade enterprise app.
- Use modern features of the chosen frameworks/languages.
- Backend and frontend logic is well-structured and modular.
- Include unit and/or integration tests.
- Backend is containerized and runs in a Linux-based Docker container.

💡 Example

Input:

"Hey guys, let's huddle about AI."

Output Example:

Professional: Hello everyone, let's schedule a meeting to discuss AI.
Casual: Hey folks, let's catch up on AI stuff.
Polite: Hi all, would you be open to a quick meeting about AI?
Social-media: Yo team! Quick sync on AI? 👀

Bonus (streamed):

[wait] Professional:
[wait] Professional: Hello everyone
[wait] Professional: Hello everyone, let's schedule
[wait] Professional: Hello everyone, let's schedule a meeting to discuss AI.

📝 Submission Guidelines

- Provide a link to your GitHub repository or a downloadable archive. No binaries and packages please.
- Nice to have: a brief README with setup/run instructions and any assumptions made.

NOTE: Please do not use AI while implementing the Test Task, Client would like to check/see your engineering skills.