Take Home Assignment: Real-time AI Poetry Generation with Emotion Visualization

Estimated Effort: This exercise should take 4 to 8+ hours, depending on individual proficiency.

Tech Stack:
Frontend: ReactJS, Figma or any Design tool
Backend: Python/Flask
Deployment: Docker, Azure (preferred) or AWS
Machine Learning: LLMs
Version Control: GitHub

Project Objective: Create a web application that takes a short user-provided prompt for a poem, uses an LLM of your choice to generate a poem, analyzes the emotions present in the poem, and visualizes these emotions using a dynamically loading graphic of your choice. The application should include intuitive UI for user interactions and delightful animations for emotions visualization.

Tasks:

1. **Design Diagram:** Before starting, plan tasks and document how you'll breakdown the assignment into milestones and allocated times. Prepare a simple design diagram of the app you plan to build. Here's a Sample Diagram for inspiration.
2. **Design a UI**: Use HTML, CSS, React, and a library like React Spring or Framer Motion to design an engaging, animated interface. The UI should include:
    a. An input field where users can enter the poem prompt.
    b. Upon submission, an animated transition should carry the user's input to a display on the screen.
    c. The response should be displayed in real-time as the poem is generated by the LLM (streamed token by token).
    d. Once the poem is complete, an animated visualization should load with analysis of all the emotions within the poem.
3. **Integration with Open AI LLM**: Set up a server-side process (server should be in Python/Flask) to send the user's prompt to the LLM and receive the generated poem.
4. **Emotion Analysis**: Implement or use emotion analysis function of your choice to analyze the generated poem. Emotions can be joy, surprise, disgust, sadness, fear, anger, etc..
5. **Socket Programming**: Implement web-socket communication for real-time text streaming between the client and server.
6. **Data Visualization**: Use a suitable React library to dynamically create and update an animated visualization based on the emotion analysis data received from the server.

Deliverables:

1. A working web app deployed on AWS.
2. Link to repos with source code of the project, including all frontend and backend code.
3. A brief doc with:
    a. Breakdown of milestones, timeline and diagram.
    b. Description of app design and how the app works
    c. Your thought process, design considerations, and justification of choices
    d. Any details or instructions on how to run the app if required
    e. List of Test Cases to verify functionality
    f. List of enhancements you can think of
    g. List of components or tasks you'd include if you were to make this an "Enterprise Grade" app