Designing 100x Microblogging Platform

Objective:

The objective of this assignment is to create a Low-Level Design (LLD) for the frontend of a microblogging platform, ensuring a user-friendly, responsive, and scalable user interface. By the end of this assignment, you should understand the principles of designing the frontend of a large-scale application, ensuring it can scale with the growth of users while providing a seamless user experience.

Tasks:

1. **Understanding Requirements:**
    - Define the core features of the microblogging platform from a user interface perspective such as user registration, posting microblogs, following users, timeline generation, and searching.
    - Identify the requirements for mobile responsiveness and accessibility.
2. **Component Breakdown:**
    - Break down the user interface into smaller components (e.g., Header, Footer, Sidebar, Post Component, Profile Component, etc.)
    - Define the responsibility and functionality of each component.
3. **State Management:**
    - Choose a suitable state management solution (Context API, etc.)
    - Design the state structure and actions for managing the application state.
4. **Routing:**
    - Design the client-side routing structure for navigating through different pages/components.
    - Ensure the routing strategy aligns with the features of the microblogging platform.

5. **Error Handling and User Feedback:**
    - Design a user-friendly error handling strategy to display error messages and other feedback to users.
    - Implement loading states to give users feedback when data is being fetched.
6. **Optimization:**
    - Propose optimization strategies for improving the performance of the frontend, such as lazy loading, code splitting, and optimizing images.
    - Discuss how these optimizations contribute to the scalability of the platform.
7. **Style Guide and Coding Conventions:**
    - Create a style guide and set of coding conventions to ensure consistency and readability of the code.
    - Discuss the importance of a unified coding standard within a development team.

Deliverables:

- A detailed Low-Level Design document explaining your design choices.
- A component hierarchy diagram.
- Code snippets showcasing key aspects of your design (e.g., state management, component structure, etc.)

Resources:

- System Design Basic
- Frontend System Design
- RADIO framework
- Frontend LLD
- Twitter Advance System Design