Technical Report – Chatbot-Oriented Project Management Application

1 Overview

The project is a **Project Management Chatbot Application** built using React, styled-components, and a **LLM (Large Language Model)** for intelligent responses. The LLM chosen for this application is **LLaMA 3**, which will be fine-tuned for project management-specific use cases. The project includes the following key components:

- **Taskboard**: A Kanban-style board for managing tasks.
- **Chatbot**: A conversational agent to assist with project management, powered by **LLaMA 3**.
- **Sidebar**: For navigation and quick access to important tools.
- **Monitoring & Feedback**: A dedicated component for evaluating and improving chatbot performance.

2 Tech Stack

- **Frontend**: React, styled-components, react-beautiful-dnd
- **Backend**: RESTful API for interaction with **LLaMA 3**.
- **Local Storage**: Persistence for task management on the client-side.
- **API Layer**: RESTful API connecting the frontend to the LLaMA 3 backend.

3 Technical Decisions by Component

3.1 Taskboard Component

The **Taskboard** allows users to manage tasks visually across different columns (e.g., *To Do*, *In Progress*, *Done*). Key decisions include:

- **Component Structure**: The taskboard consists of columns (each managed as a Column component) and tasks (Task component). React's component system allows for modularity and easy maintenance.
- **Drag-and-Drop**: Implemented using the **react-beautiful-dnd** library, which provides smooth task reordering across columns.
- **State Management**: The **useState** hook manages internal state, while **useEffect** persists this state in **localStorage**, ensuring that tasks persist across page reloads.
- **UI/UX**: **Styled-components** provide modular, reusable CSS that is responsive and readable across devices.
- **Persistence of Data**: Task data is stored in the browser's **localStorage**, ensuring tasks are retained across sessions.

3.2 Chatbot Component: Powered by LLaMA 3

The **Chatbot** is driven by **LLaMA 3**, designed for conversational assistance in project management scenarios.

- **Frontend Integration**: The chatbot sends user inputs to the **LLaMA 3 API**, receives responses, and displays them in a chat interface built in React, enabling real-time interaction.
- **Contextual Conversation Handling**: The chatbot maintains conversation context using a session-based approach. History is stored in the frontend state and sent to the API to ensure response relevance.
- **User Interface**: The chatbot UI includes a chatbox for user queries and a history panel displaying the conversation. It dynamically updates with new messages.
- **State Management**: **useState** and **useEffect** manage conversation history and state across interactions.
- **LLM Backend (LLaMA 3)**: The chatbot queries a **LLaMA 3** model hosted on a backend, fine-tuned for project management tasks like task assignments, deadlines, and milestones.

3.3 Monitoring & Feedback Component

A dedicated **Monitoring and Feedback** component tracks and evaluates chatbot performance for continuous improvement.

3.3.1 Key Features

- **User Feedback Collection**: Each chatbot response can be rated, and the feedback is stored for analysis, aiding fine-tuning of the LLaMA 3 model.
- **Interaction Logging**: Logs user-chatbot interactions for performance analysis, tracking accuracy, relevance, and response time.
- **Performance Metrics**: Evaluates response relevance, user satisfaction, and contextual accuracy.

3.3.2 Technical Implementation

- **Frontend**: Feedback buttons allow users to rate chatbot responses, captured in React state and sent to the backend via API calls.
- **Backend**: Feedback and logs are stored in a database for analysis. Feedback helps retrain and fine-tune the LLaMA 3 model.
- **UI/UX**: The feedback interface is simple and non-intrusive, allowing users to provide feedback seamlessly.

3.4 Sidebar Component

The **Sidebar** serves as the navigation hub, enabling users to switch between components such as the Taskboard, Chatbot, and Monitoring & Feedback.

Key Features:

- **Navigation**: Provides quick access to different components.
- **Responsiveness**: The sidebar collapses on smaller screens for mobile and tablet users.

Technical Implementation:

- **React Router**: Navigation between sections is handled using **React Router** for smooth transitions without page reloads.
- **Styled-components**: The Sidebar is styled for responsiveness with hover effects for a smooth user experience.

4 AI Training Process for the Chatbot (LLaMA 3)

4.1 Model Selection: LLaMA 3

LLaMA 3 was selected for its performance in natural language understanding and handling multi-turn conversations. It is ideal for project management tasks where contextual understanding is crucial.

4.2 Dataset for Fine-Tuning

LLaMA 3 is fine-tuned using a dataset curated from:

- **Project Management Documentation**: Task definitions, project timelines, milestones, and coordination strategies.
- **Real-World Conversations**: Transcripts from project management scenarios.
- **Tools-Specific FAQs**: Common issues faced by users of tools like Jira, Trello, and Asana.

4.3 Training Pipeline

1. **Data Preprocessing**: The text is cleaned and tokenized. Conversation logs are structured as question-response pairs.
2. **Fine-Tuning**: The **LLaMA 3** model is fine-tuned using transfer learning with carefully tuned hyperparameters.
3. **Evaluation and Testing**: Metrics such as response relevance and accuracy are used to evaluate the model. Rigorous testing simulates real-world conversations.

5 Continuous Model Improvement

The chatbot follows a **continuous improvement cycle**:

- **User Feedback Integration**: Feedback is used to retrain the model.
- **Fine-Tuning Schedule**: Regular fine-tuning based on new data from real-world interactions.
- **Error Handling**: The chatbot includes fallback responses for unrecognized inputs, prompting users for clarification.

6 Conclusion

This **Project Management Chatbot Application** is built using modern web development practices (React) and cutting-edge AI technology (**LLaMA 3**). The integration of a **Taskboard**, **Chatbot**, and **Monitoring & Feedback** ensures continuous improvement based on user interactions. The choice of **LLaMA 3** provides a robust model for natural language understanding, which can be fine-tuned over time for project management needs.