Training Day-26 report

22 July, 2025

Today, I continued developing the **Polling Application**, focusing on **real-time functionality** using sockets, and refining the app's **authorization and data-fetching strategies**.

Work Undertaken:

1. **Socket-Based Controllers:**
   • Developed a set of **controller functions** dedicated to **socket interactions**, enabling:
   – **Real-time vote updates**
   – **Live poll status tracking (e.g., active/inactive)**
   • These controllers help in dynamically updating the UI without needing manual page refreshes.

2. **Authorization Completed:**
   • Finalized the **authorization flow**, ensuring users must be authenticated to create or participate in polls.

3. **Architecture Enhancement Consideration:**
   • Currently, fetching polls relies on **client-initiated requests** via a traditional GET route.
   • Planned to **shift this logic to socket-based broadcasting**, allowing the system to **automatically send newly created polls to all connected clients in real time**.

Key-Learnings:
• Gained hands-on experience integrating **WebSocket-based communication** alongside traditional.REST-endpoints.
• Understood how to separate logic for real-time vs. standard HTTP-based functionality.
• Reinforced the need for **efficient data delivery** in interactive apps to enhance user experience.

Conclusion:
With real-time vote handling now functional and the UI framework in place, the next focus will be on completing poll creation and display through socket-based updates. This will ensure a smoother and more dynamic user experience.