

AeroAspire - SDE Intern Training

JOHN NIKHIL G

Week2-Day 5-Oct4

Add search/filter UI; theme toggle (light/dark) if possible; polish

Questions:

1. Explain the full data flow: user action → state → props → UI update.

- a. When a user clicks or types, it triggers an event.
- b. The event changes the state using useState or similar hooks.
- c. When the state changes, React re-renders the part of the app that needs updating.
- d. Data is sent from a parent to child component using props.
- e. The new data shows up on the screen after React updates the UI.

2. What are common anti-patterns in React you noticed or want to avoid?

- a. Changing state directly instead of using setState or hooks.
- b. Keeping too many states in one component instead of splitting them.
- c. Forgetting to check how routing works between pages.
- d. Using hooks inside loops or conditions instead of at the top level.

3. If this app grows big, what architectural patterns would you use (e.g. component separation, state management tool)?

- a. I would break the app into smaller components for easier updates.
- b. Use a state manager like Redux or Context API for smoother data flow.
- c. Add login and permissions for users if needed.
- d. Allow saving some data offline for better performance.
- e. Keep testing and polishing the app to make it run better.