Defying Gravity in Testing

@AssumptionBuster To test state leaks, I'm keeping both panels (admin and normal user) visible and do reserve cancel then flip filters and sort. Identify what could cause false alarms or false confidence in my observations, then rewrite my test approach with corrections to reduce both.

- Logic and assumption elicitation:

    ○ The approach assumes two visible panels represent distinct, simultaneously valid role states.

    ○ It also assumes reserve-cancel and filter/sort flips expose cross-panel leaks.

- Adversarial critique — assumption weaknesses:

    ○ False alarms may stem from UI caching, asynchronous updates, DOM reuse, or CSS artifacts.

    ○ False confidence arises if server state, auth tokens, or concurrent sessions differ unseen.

- Persona-based debating:

    ○ From an attacker persona: could shared APIs or predictable identifiers allow spoofing of panel state?

    ○ From an accessibility user's view: might assistive technologies surface different timing artefacts misleading observations?

- Reflective challenge:

    ○ Which explicit signals tie visible UI changes to authoritative server state for this scenario?

    ○ Please clarify intended constraints and failure modes before a rewritten testing rationale is proposed.

Type here… (e.g., '@Brainstormer Find unseen exploration paths in the uploaded SRS') ➤