


Challenger settings

Challenge intensity 

☐ mild


☒ spicy

Agent Selector

Tip: You can also explicitly type an agent tag in your message, e.g.,

`@novelty_radar` or `@bug_logger` .

Preferred Agent (optional)

Auto (Smart Intent Detection) 



Upload Document

Requirements / Logs / Specification (txt / pdf)

Drag and drop file here

Limit 200MB per file • TXT, PDF

Browse files

 Requirements Specifica... 
38.6KB

> Preview of Uploaded Document
(first 1000 characters)

Defying Gravity in Testing



@AssumptionBuster I'm keeping both panels visible, switching users (admin and normal user) around and do cancel then flip filters and sort to see if reservation state leaks. What invariants should I check to confirm whether any stale state is persisting and how can I avoid false alarms?



- Logic and assumption elicitation:
 - The approach assumes cancelling under one active user immediately updates both visible panels.
 - It further assumes reservations are tied to the active User ID and UI honors FR-4, FR-8, FR-17, FR-19.
- Adversarial critique — assumption weaknesses:
 - Component-local state, memoization, or virtual DOM diffing may leave stale UI despite model updates.
 - Admin actions, reordering, or capacity overrides could mutate shared structures, breaking user isolation.
- Persona-based challenge:
 - As an attacker, could rapid user switches exploit race conditions to view other users' reservations?
- Reflective challenge:
 - Which explicit invariants will convincingly prove absence of stale cross-panel state?
 - How will expected rendering timing differences be distinguished from genuine data leaks?

Type here... (e.g., '@Brainstormer Find unseen
exploration paths in the unloaded GPC')

