

LMS Usability Testing – Affinity Map

Created by Claude on 8/23/2026

Status:  Complete — 5 of 5 participants logged **Participants logged:** P1 (Torey McNeil), P2 (Lenicha Bell), P3 (Edward Song), P4 (Kathleen Manning), P5 (Jennifer Maxwell) **Legend for completion tier:** ① Unassisted | ② Completed w/ difficulty (unassisted) | ③ Completed w/ assistance (prompted) | ④ Failed

Participant context (for interpreting findings):

- **P1 — Torey McNeil:** Internal TROA stakeholder; thinks in terms of training pipeline, compliance, and reporting needs downstream of course creation.
- **P2 — Lenicha Bell:** Director of HR at TROA; prior LMS experience with Canvas, Blackboard, and custom platforms. Key needs: robust reporting/ analytics, customizable courses, a library of "canned" content. Flagged a prior Canvas frustration — its calendar required manually selecting every course to see assignments, which caused missed deadlines (relevant context for any future scheduling/dashboard features, not this prototype directly).
- **P3 — Edward Song:** Thinks structurally/systemically — consistently reasons from the tree/hierarchy rather than surface labels, and is the first to raise performance and technical-architecture implications (studio load time, autosave cost) alongside interaction feedback. Also the first to raise AI-assisted course authoring as a near-future use case.
- **P4 — Kathleen Manning:** Prior experience is primarily with pre-built training material (PowerPoint/documents created outside the LMS) uploaded and assigned by *role* rather than by individual name; approval typically happened outside the LMS before upload. This background shapes her expectations around assignment, notifications, and draft/approval flows more than around content-block authoring itself.
- **P5 — Jennifer Maxwell:** Executive Operations Coordinator at TROA and Executive Assistant at Beacon Mental Health; prior LMS is Relias (a nationwide community mental health training platform) — single-screen scrolling pages, in-lesson knowledge checks, and a final scored test (80% pass) taken *outside* Relias with the score linked back for certification. Courses come pre-built from the state or from Beacon itself, taken quarterly/annually — she has never authored a course before, only taken them. This makes her

the closest proxy in your sample for a true novice course-creator, and her think-aloud is unusually explicit about *when* her mental model breaks vs. resolves.

Cluster 1: Visibility & Draft Status Model

●●●●● Pattern confirmed — 5 of 5 participants. Your #1 finding. Theme: The mental model for who can see a course, and at what stage, doesn't match the system's actual states.

Observation	Participant	Severity
"Restricted" label doesn't clarify <i>who</i> can view — Torey expected to specify department names	P1	Major
"Unlisted" was interpreted as "can't be found" — ambiguous vs. intended meaning	P1	Major
"All TROA" interpreted correctly as everyone at TROA	P1	Positive
No way to limit visibility until ready to share — expected a draft mode	P1	Critical
Unclear whether a course-in-progress is draft or already visible to LMS users	P1	Critical
"Unlisted" independently guessed as draft mode — "no one can see it but me while building"	P2	Major
"Restricted" — unclear how restriction is configured (department? internal vs. external?) and no visible way to actually choose people	P2	Major
Clicking "+ Add course" made her expect draft status by default, partly <i>because</i> of the plus-sign affordance and being early in the process	P2	Critical
"Restricted" flagged as vague again — no way to define who can access; suggested a pop-up or dedicated page to set access by department or by division (a division being a broader grouping than department)	P3	Major
Confirmed system's actual definitions: All TROA = everyone with TROA domain email, Unlisted = no one will see, Restricted = gated access to certain people — but the <i>mechanism</i> for Restricted is still missing from the UI	P3	Major
Noted the Create-a-Course page currently doubles as something resembling a home page (doesn't show other courses) — felt it should be a page dedicated only to course creation	P3	Minor

Publish button enablement rule: a course needs modules, lessons, pages, <i>and</i> content blocks all present before it can be published — all node types are required	P3	Major (new — publish readiness logic)
Independently expected to "pick and choose who gets to see" — proposed tagging by department, and separately tagging by individual within a department	P4	Major
Suggested restriction may need to live in a setting <i>outside</i> course creation entirely (i.e., a reusable access-group setting, not a one-off per-course config)	P4	Major (new — architecture suggestion)
Proposed two separate levels of restriction: one for narrow, sensitive audiences (e.g., Board of Directors only) and a separate, broader mechanism for department + individual-within-department restriction	P4	Major (new — two-tier access model)
Confirmed draft mode is needed specifically to support getting approvals on content correctness before publishing	P4	Critical
Noted the approval process can happen either inside or outside the LMS — when course content is authored outside the LMS, approval typically also happens outside, before upload	P4	Major (new — approval flexibility, echoes P1)
Deliberately chose "Unlisted" <i>as a workaround</i> to hide a course while seeking approval and preparing to publish — i.e., successfully repurposed an existing option to simulate the draft state everyone else expected to exist natively	P5	Critical (new — confirms the workaround, not the fix)
Guessed "Restricted" means IT-only, or wanted a toggle letting any course creator choose who views it — a 5th distinct wrong guess at the same label	P5	Major (5th confirmation)
Reported that clicking "+ Course" immediately makes the course visible to All TROA — i.e., the system's actual default is public-on-creation, not draft	P5	Critical (new — confirms the root cause behind every other participant's expectation mismatch)

Underlying need (consistent across all five participants):

- Draft should be a distinct, default state — five participants independently expected a new course to start as draft, and P5's report of the actual default (public to All TROA immediately) confirms *why* every participant has been surprised: the system's real behavior is the opposite of what everyone expects
- "Restricted" needs an actual mechanism for choosing who — five participants offered five different guesses (department, IT-only, board-level,

individual, toggle), meaning the label currently communicates nothing reliably

- Draft → approval flow (approver reviews) → publish (P1, P4, P5); publish gated on structural completeness (P3)
- System should support approval happening either inside or outside the LMS depending on where content originated (P1, P4)

Design implication: This is confirmed by every single participant tested and is unambiguously your #1 finding. P5's observation is the capstone: it identifies the actual root cause (public-by-default on creation) that explains the other four participants' surprise and improvised workarounds. Recommend leading your case study with this exact narrative arc: default behavior (public-on-creation) → user expectation (draft-by-default) → the gap this creates (5/5 participants confused or improvising) → the solution users converged on unprompted (draft state, tiered restricted access, approval flow, publish gated on completeness).

Cluster 2: Navigation / Findability of Key Actions

 **"No way back up a level" confirmed — 3 of 5 participants (P2, P4, P5)**

Theme: Primary action links are getting lost against surrounding UI, causing users to take the wrong path.

Observation	Participant	Severity	Completion Tier
"Save & add lessons" link not visible enough — Torey missed it entirely and started typing page content into the module Summary field instead	P1	Critical	② (recovered on own, but via wrong path first)
"Cancel edit" link not prominent enough — harder to exit edit mode than to enter it	P1	Major	②
Delete and Edit links themselves were easy to find and use	P1	—	① (Positive)
From the page level, there is no way to get back up to the module level or create a new module/lesson — navigation is one-directional	P2	Critical	④ (not resolved — structural gap, not just missed link)

" + Save and add another module" button visually highlighted, which led her to build out the entire course table of contents first rather than the flow the task intended	P2	Major	—
Cannot go back to course settings to edit the course name, summary, or visibility once past that step	P4	Critical	④
Cannot go back to a module or lesson level to add another module/lesson when already in a lower level — independently proposed adding a button inside the tree structure for this	P4	Critical	④ (matches P2's finding directly)
Being forced to click "Save & add lesson" <i>inside</i> the module in order to add a lesson felt unintuitive as the only path	P4	Major (echoes P1's missed-link finding from the opposite angle — she found it, but disliked being funneled through it)	
Explicitly named "no back button" as an issue in the creation flow	P5	Critical (3rd confirmation)	

Design implication: Now confirmed by 3 of 5 participants: creation flow only moves forward, never back, at any level (course settings, module, lesson, page). This is your second-strongest structural finding after Cluster 1. Combined with the tree-view suggestions (Cluster 10, now also implicitly requested by P5's need to reference the tree to understand structure), the recommendation is clear: a persistent, always-visible tree/breadcrumb structure that allows jumping to any level at any time, replacing the current one-directional flow.

Cluster 3: Content Structure Doesn't Match Real Training Content

 **Quiz question-type gap confirmed — 3 of 5 participants; single-content-block-per-page limitation confirmed — 2 of 5** **Theme:** The prototype's content model is simpler than what TROA actually needs to deliver.

Observation	Participant	Severity
Only a single page with multiple content blocks is needed — current structure may be more complex than necessary	P1	Major
Missing content block types: audio (for role-play evaluations), tables (for complex data)	P1	Critical

Quiz missing image-based question type (visual recognition training)	P1	Critical
Quiz type is set globally per quiz, not per question — no mixed question types	P1	Major
Quiz builder called "unusable" — same root issue: no per-question type selection (she's used to mixing multiple-choice and true/false within one quiz)	P2	Critical
Quiz title treated as optional — only needed if there's a meaningful title to add	P2	Minor
Image upload without accompanying text field felt incomplete/not useful on its own	P2	Minor
Independently requested selecting a question type per question rather than one type for the whole quiz	P3	Critical
Independently flagged the need to add tables into text content blocks, or a dedicated audio content block — matches P1's exact request	P3	Critical
Flagged that uploading an outside document to auto-generate a quiz is a need that should be scoped before further UI adjustments	P3	Major (new — content authoring workflow)
Bug: when adding a quiz question without selecting a correct answer, no validation/error message appears — should block or warn before saving	P4	Major (new — missing validation)
Explicitly wanted to combine text with an image or video on the same page, but each page only supports one content block — directly matches P1's ask for multi-block pages	P5	Critical (2nd confirmation)
Prior experience (Beacon/Relias) used only multiple-choice questions — no true/false or short answer — specifically because it's easier to grade at scale	P5	Note — counterpoint, see below

Design implication: Per-question type selection remains confirmed by 3 of 5 participants. The single-content-block-per-page limitation is now confirmed by 2 of 5 (P1, P5) with near-identical language ("combine things together" / "multiple blocks"). **Worth flagging a genuine tension in your findings:** P1, P2, and P3 all want *more* quiz flexibility (mixed question types), while P5's prior-tool experience biased her toward valuing simplicity (multiple-choice only, for easier grading at scale). This isn't a contradiction to resolve so much as a segmentation insight — flexibility matters more for power users building varied training, while grading simplicity matters more for high-volume compliance content. Worth a sentence in your report rather than smoothing it over.

Cluster 4: System Feedback & Trust Signals

●● Missing save-in-edit-mode confirmed — 3 of 5 (P3, P4, P5); "Save" word not load-bearing confirmed — 3 of 5 (P1 disagrees, P2/P5 say word doesn't matter) Theme: Users want ongoing confirmation that their work is captured correctly, and the system under-communicates this.

Observation	Participant	Severity
"Preview" feature important to periodically check what's been created so far	P1	Major
Word "Save" must stay in button labels (e.g. "+ Save & add another module") — removing it reads as "won't be saved"	P1	Major
Quiz acknowledgment only shows on correct answer — no acknowledgment/feedback path for other cases	P1	Minor–Major
Independently, when asked the same question, said she'd still expect the current module to be saved even <i>without</i> the word "Save" in the label	P2	Major
After "+ Save quiz," dropped back to the previous page with no choice of where to go next — then confused when clicking "+ Save and another page" duplicated the preloaded page's content into a new page	P2	Critical
Wants clear confirmation of <i>where a button will take her</i> before clicking — explicitly said she avoids clicking "+ Add quiz" because she can't predict the destination	P2	Major
Proposed a single master "Save" button instead of multiple Save CTAs scattered per screen, motivated by real performance concerns (see below)	P3	Major (new — architectural proposal)
Specifically flagged that autosave would worsen performance, since it would constantly check for changes to persist	P3	Major (technical constraint, not just preference)
During node editing, recommended a "Save"-only option (no bulk-create alongside it), to keep the editing action unambiguous	P3	Minor
After adding a quiz, was dropped back to the page she was on before — called this flow "not intuitive," directly matching P2's identical finding	P4	Critical (confirmed 2/5 — P2 + P4)
Found no way to save changes at all when editing an existing node — a save action she expected wasn't present	P4	Critical (new — matches P3's request for an explicit edit-mode Save button)
When editing a node, wanted to save changes but found no Save button — described resorting to a workaround (delete and re-add an item) rather than a real save action, calling it "a lot of work"	P5	Critical (3rd confirmation — P3, P4, P5 all confirm missing edit-mode save)

Independently confirmed that even without the word "Save" in the button, she'd still expect the module to be saved, since nodes "should be saved somehow" — but said having the word "feels better"	P5	Major (3rd confirmation the word itself isn't load-bearing, though it's reassuring)
---	----	---

Pattern to note: Missing save-in-edit-mode is now confirmed by 3 of 5 participants (P3, P4, P5) — this has graduated from "nice to have" to a clear, well-evidenced bug: **editing an existing node has no explicit save action**, forcing at least one participant into a delete-and-recreate workaround. On the "Save" wording question, 3 of 5 (P2, P4 implicitly, P5) say the word itself doesn't change their expectation that saving happens automatically — only P1 wants the word kept for reassurance. Recommend framing your finding as: the missing save action is the real bug; the word "Save" is a minor reassurance signal, not a functional requirement.

Cluster 5: Placement Causing Mental Model Confusion

●●●●● Quiz-attachment-level confusion confirmed — 5 of 5 participants.
Your #2 finding. Theme: Where controls are placed implies a different structure than intended.

Observation	Participant	Severity	Completion Tier
"Add quiz" button placed under the page — implied quiz attaches to each page, though Torey knew (correctly) it should attach to a module	P1	Major	—
Unsure if a quiz attaches to a lesson or a page — assumed lesson since it appears after page creation, but suspected it might be flexible	P2	Major	—
Clicking "+ Add quiz" actually added another <i>page</i> , not a quiz at the expected level — directly contradicted her expectation	P2	Critical	④ (task outcome didn't match intent — flag as failed sub-task)

Only after adding multiple lessons did she work out that quizzes attach at the module level — not obvious at first	P2	Major	②
Knew from the tree structure that a quiz is meant to attach to a module, but because he created the quiz from a page, it visually appeared to attach to the page instead — UI contradicted his own correct prior knowledge	P3	Critical	② (right answer, UI still confusing)
Bug: "Add quiz" is only available the <i>first</i> time a page is created — editing an existing page removes the option to add a quiz, which he found strange/inconsistent	P3	Major (new — inconsistent availability)	
Assumed the quiz was added to a lesson because she added it from a page — but stated that in reality, it should attach to a module, with the option for small checkup questions at the page level too	P4	Critical (4th confirmation)	②
Explicitly proposed quizzes should be addable at page, lesson, <i>or</i> module level, rather than one fixed level — a flexible-attachment model	P4	Major (new — proposed solution)	
Assumed the quiz was added to the lesson she was currently in; was "confused and disoriented" clicking "+ Add quiz" not knowing where she'd be taken; only after checking the tree structure did she discover the quiz actually attached to the whole module (all lessons) — not obvious at first glance	P5	Critical (5th confirmation)	② (resolved only by consulting the tree)

Design implication: Confirmed by all 5 of 5 participants — the single most consistent finding in your entire study. Every participant misjudged or was confused by quiz attachment level, and at least three (P3, P4, P5) only resolved the confusion by manually consulting the tree structure after the fact — meaning the tree is currently functioning as an ad hoc recovery mechanism rather than a preventive one. This strongly supports pairing this finding with Cluster 10's tree-visibility recommendation: make the tree persistently visible *during* creation, not just something users check afterward to figure out what happened. P4's proposal (flexible attachment at page/lesson/module level) remains the most complete solution offered and is worth featuring as your primary recommendation.

Cluster 6: Scale & Bulk Operations

Theme: Manual, one-at-a-time content entry doesn't hold up against TROA's real content volume — and separately, the bulk-create feature has a real field-parity gap.

Observation	Participant	Severity	Completion Tier
Bulk upload needed — 50-page counselor workbook, 350+ hrs of training content make manual entry impractical	P1	Critical	—
Expectation: bulk upload from Google Sheets for both course and quiz content	P1	Critical	—
Content changes annually (or bi-annually for TROA) — bulk <i>update</i> , not just bulk <i>create</i> , is needed	P1	Major	—
After bulk-creating modules, was dropped into the newly created module (module 2) and accidentally bulk-created <i>more modules</i> there instead of lessons — no bulk-create option for lessons was visible on that screen	P2	Major	②
Eventually found her way to the lesson screen and successfully bulk-created lessons, without moderator assistance	P2	—	② (Positive recovery)
Liked the bulk-create feature overall — found it easy enough to use, and noted he'll likely use AI to generate courses going forward	P3	Positive	①
Bug: when bulk-creating additional modules from an existing module, the input field labels default to "Module 1, Module 2, Module 3" even though modules with those names already exist — confusing/risks accidental duplication or overwrite	P3	Major (new — naming collision bug)	
Recommended bulk-create should <i>not</i> be offered while editing an existing node — keep bulk-create and single-node editing as separate, unambiguous modes	P3	Major (new — mode separation)	

Got stuck when asked to bulk-create lessons and add them to a module — directly tied to being funneled through the "Save & add lesson" flow inside the module (see Cluster 2)	P4	Major	③ (needed moderat or prompt to complet e)
Wasn't sure what bulk-create did at first — guessed "bulk-create modules" meant bulk-creating lesson <i>content</i> to learn, rather than creating empty module containers	P5	Major (new — purpose/ labeling confusio n)	
Field-parity gap: Single-Create includes summary and progression-requirement fields; Bulk-Create only offers a title field — meaning anyone using bulk-create must revisit every module individually afterward to fill in summary and progression requirement	P5	Critical (new — concrete, well-specified gap)	
Self-identified as "a completionist" who prefers finishing one item fully before moving on — this personal working style made the bulk-create field gap especially frustrating, and made her prefer single-create despite the extra manual effort	P5	Major (personal -style context)	
Explicitly said she'd be more confident using bulk-create if summary and progression requirement could be entered there directly	P5	Major (propose d fix)	

Note: Bulk-create sentiment is mixed and role/style-dependent: P3 completed it unassisted and liked it, while P2 and P4 hit level-navigation friction, and P5 avoided it altogether due to a genuine field-parity gap rather than confusion. **P5's finding is your most concrete, fixable bulk-create recommendation of the whole study:** bring Bulk-Create's field set to parity with Single-Create (at minimum, summary and progression requirement), which would directly address her stated blocker and likely increase adoption for completionist-style users generally.

Cluster 7: Reporting & Progress Rules (Post-Creation Needs)

Theme: Course creation is only half the workflow — participants are already thinking about what happens after a learner starts.

Observation	Participant	Severity
Needs a dashboard/report view for training pipeline status	P1	Major
Needs visibility into who completed what training, when, and whether they're cleared for live contacts	P1	Critical
Needs a "Progress Requirement" rule allowing learners to move ahead while awaiting supervisor review of open-response answers	P1	Major
Expects email notifications sent to individuals assigned a course, reminding them of a completion due date	P4	Major (new)
Expects an admin-facing view of who has completed a course, to follow up individually with anyone who missed the due date	P4	Critical (echoes P1's completion-tracking need almost exactly)
Course completion can be gated by time-on-page, ensuring learners actually engage with material rather than skipping through — the "next page" button stays disabled until a set time has elapsed	P4	Major (echoes P3's understanding of the existing Time rule)

Note: These remain arguably out of scope for the course-*creation* task you tested, but P1 and P4 independently converge on the same core need (admin visibility into who's completed what, by when) from different roles — training lead and HR. Flag as "future iteration" findings rather than folding into your primary hypothesis scoring, so they don't dilute your core task-completion metrics, but the convergence is worth a sentence in your report as evidence of a real, recurring need beyond this prototype's scope.

Cluster 8: Hierarchy & Naming Depth Mismatch

●● "No explanation of hierarchy/structure" confirmed — 2 of 5 explicitly (P3, P5), with P2 and P5 both showing real behavioral consequences of it

Theme: TROA's Course → Module → Lesson → Page hierarchy is deeper than what participants' prior tools trained them to expect, and it shows up as friction at multiple points — including simply not knowing the hierarchy exists.

Observation	Participant	Severity
Prior platform structure was Course > Module > Lesson, with lessons holding multiple <i>sections</i> — not pages/content blocks the way this prototype does	P2	Major
Felt there were "so many titles to think about" — module, lesson, <i>and</i> page titles — vs. her prior tool's simpler main-title/optional-subtitle pattern	P2	Major
Interpreted "page" as equivalent to a "section" from her prior tool, since a page also comes with a content block	P2	Minor (self-resolved via analogy)
Uncertain whether the course is meant to be a slide-format or a horizontal scrolling page — affects how she'd chunk content (bite-sized vs. longer-form)	P2	Major
No explanation anywhere of what a course <i>consists of</i> (Course > Module > Lesson > Page > Content Block) — made it hard to predict what comes next at each step	P3	Critical (root cause identified)
Explicitly conflated Module with Lesson in her own mental model — "In her mind, Module is Lesson," leaving her unsure what came after Module at all	P5	Critical (most severe hierarchy breakdown in the study)
Said directly her mental model didn't align with the LMS, and specifically questioned why lessons were needed if she already thought of a module as the equivalent of a lesson	P5	Critical
Entered content into the Summary field when creating a node, believing the summary was the node's content — did not realize nodes are empty containers and content lives elsewhere (in pages)	P5	Critical (new — this is a materially different and more severe misunderstanding than "confusing labels"; she was about to consider the course "finished" without ever adding real content)
Wanted a "finish" button at the end of module creation, revealing she expected module-creation to be a complete, terminal task rather than one step in a larger nested structure	P5	Major (new)
Stated a need for some kind of built-in explanation or short training on how module/lesson/page relate to each other	P5	Major (echoes P3's root-cause finding directly)

Eventually self-resolved the structure by reasoning aloud: <i>"If I was thinking through it... modules go to course and the module has lessons. But at first glance, what is this?"</i> — successful but effortful recovery	P5	—
Independently asked "What is pages?" mid-session, then resolved it herself: <i>"Now I understand what pages are. Pages are what's in the lesson"</i> — documents, pictures, videos that go into a lesson	P5	—

Design implication: P5's session is your clearest evidence that this is a genuine onboarding gap, not just a labeling quibble — she is the closest proxy in your sample to a true first-time course creator (her only LMS experience is as a learner, not an author), and she nearly finished "creating a course" without adding any real content, because she mistook a container's summary field for its content. This single observation may be your most important novice-user finding in the whole study: **the system currently has no mechanism to prevent someone from believing they're done when they've only built empty structure.** Recommend pairing this with Cluster 10's tree-visibility request and P3's root-cause diagnosis (no upfront hierarchy explanation) into one unified recommendation: a brief structural explainer or guided first-run experience, plus a persistent tree view, plus content-completeness validation before allowing publish (which also ties back to P3's publish-readiness rule in Cluster 1).

Cluster 9: Predictability of Button Destinations (New — P2)

Theme: Beyond individual mislabeled buttons, there's a broader pattern of participants hesitating to act because they can't predict the outcome of a click.

Observation	Participant	Severity
Avoided clicking "+ Add quiz" specifically because she couldn't predict where it would take her — wanted assurance it would ask what to do next rather than jumping straight into an unknown flow	P2	Critical

"Progression Requirement" setting direction was ambiguous — unclear if the rule governs progressing <i>into</i> or <i>out of</i> a module; she used instinct ("progress to this current module") to guess, and chose "no rule" partly to avoid the ambiguity	P2	Major
Explicitly requested clearer directional cues before clicking, and suggested tutorial videos, pop-overs, or tooltips as a mitigation	P2	Major

Design implication: This is the throughline connecting Clusters 2, 4, and 5 — Torey and Lenicha's individual complaints about specific buttons/labels are symptoms of one root issue: **the system doesn't preview consequences before a destructive or navigational action**. Consider naming this as your #1 or #2 top-line finding in the case study, with the individual button issues as supporting evidence rather than standalone findings.

Cluster 10: Structure/Tree Visibility (New — P3)

Theme: Once nodes are created, users lose track of the overall course structure — there's no persistent view of what's been built.

Observation	Participant	Severity
Created nodes are difficult to see after the fact — suggested a visible tree structure so users can see the full course hierarchy at a glance	P3	Critical (new)
Proposed the "Add node" action live <i>within</i> the tree view itself, so structure and creation are the same interface	P3	Major
If a tree view is added, suggested pushing it to the top of the page to use the full available height, rather than starting below the "Create a Course" heading	P3	Minor (layout detail)

Design implication: Connects directly to Cluster 8 — a persistent tree view would solve both "I can't see what I've built" and "I don't understand the hierarchy" in one component. This is a strong, concrete design recommendation to carry into your synthesis, since it's a single solution addressing two separate pain clusters.

Cluster 11: Progression Requirement Logic (P3, P2, P4)

● **"No rule for first module" pattern confirmed — 2 of 4 participants (P2 chose it instinctively, P4 stated it as an explicit rule) Theme:** The rules governing how learners move between modules are conceptually muddled, for more than one reason.

Observation	Participant	Severity
Progression Requirement: unclear whether the rule applies to a page, lesson, or module — or optionally all three	P3	Major
"Must pass quiz" was clear and not confusing	P3	Positive
"No rule" and "Must complete previous module" <i>felt like the same option</i> to him, since he assumes users shouldn't be able to jump around modules anyway (sequential order expected by default) — recommended combining the two	P3	Major (new — redundant options)
"Time" requirement (minimum time spent per page) exists in the current implementation and was understood	P3	Positive
Stated explicitly and unprompted: the first module should not have any progression rule	P4	Major (confirms P2's instinctive choice as a stated principle)

Cross-reference — P2 (Cluster 9): Lenicha separately found "Progression Requirement" ambiguous in *direction* (progressing into vs. out of a module) and picked "no rule" partly to sidestep that ambiguity — but her actual choice ("no rule" for the first module) turns out to match what P4 states as an explicit, confident rule. **Combined, three participants (P2, P3, P4) all touch this setting, converging on: first module = no rule by default, and "no rule" vs. "must complete previous" may not need to be separate options at all.** This is a strong, near-complete spec for redesigning the setting — worth writing up as a finding with a concrete recommendation rather than an open question.

Cluster 12: Studio Performance & Save Architecture (New — P3)

Theme: A real backend performance constraint is shaping (and should shape) save-interaction design, not just user preference.

Observation	Participant	Severity
Studio area of the LMS loads in 5–8 seconds, versus ~0.2 seconds for every other page — a significant, noticeable outlier	P3	Critical (technical finding)
Recommended a single master "Save" button rather than multiple scattered Save CTAs, specifically because of this performance gap	P3	Major
Warned that autosave would make performance worse, since it would require constantly checking for unsaved changes	P3	Major

Note: This isn't a typical usability finding — it's a technical/performance observation from a participant reasoning like a practitioner. Recommend flagging this separately in your report (e.g., as an "engineering consideration" callout) since it should inform the *feasibility* of any "add more inline save feedback" recommendation coming out of Cluster 4 — you don't want to recommend more autosave/inline confirmation without noting this constraint.

Cluster 13: Course Assignment Model (New — P4)

Theme: Assigning training isn't just about who can *see* a course (Cluster 1) — it's a separate, related need around who a course is actively *assigned to*.

Observation	Participant	Severity
Prior workflow never required knowing individuals by name — training was assigned by <i>role</i> , not by person	P4	Major
Expects to be able to assign a course either by specific name or by role — both should be supported	P4	Major (new — proposed model)
When a person logs into the LMS, they see their assigned training — this is the expected end-to-end outcome of assignment	P4	Major (context, not a bug)

Design implication: This is conceptually distinct from Cluster 1's visibility/restriction model — visibility controls who *can* see a course, assignment controls who's *expected* to complete it, and the two may need to be separate settings entirely (a course could be visible to a department but only *assigned* to specific roles within it). Worth a dedicated line in your report distinguishing "access" from "assignment," since P4 is the only participant to surface this distinction but it has real implications for the settings model in Cluster 1.

Cluster 14: Delete Modal Clarity (New — P4)

Theme: Destructive-action confirmation language doesn't match what's actually being described.

Observation	Participant	Severity
Multi-node delete modal has mismatched verbiage — the title states the module will be deleted, while the description states the module's children will be deleted, without being clear these are the same cascading action	P4	Major (new)

Design implication: A small, concrete, easily fixable copy bug — good candidate for a "quick win" recommendation in your report, distinct from the larger structural findings.



Final Synthesis Summary — All 5 Participants Complete

Findings ranked by frequency (strongest evidence first):

Finding	Frequency	Cluster
Draft-by-default expectation + "Restricted" visibility ambiguity	5/5	Cluster 1
Quiz attachment level (module/lesson/page) misjudged or confusing	5/5	Cluster 5
Per-question quiz type selection needed (not one type per whole quiz)	3/5	Cluster 3
Missing Save action when editing an existing node	3/5	Cluster 4
No way to navigate back up a level once past it	3/5	Cluster 2
"Save" word itself isn't load-bearing for trust (system should save reliably regardless)	3/5	Cluster 4
Progression Requirement setting is conceptually muddled	3/5	Cluster 11

No upfront explanation of the Course→Module→Lesson→Page hierarchy	2/5 (explicit) + downstream effects in a 3rd	Cluster 8
Single content block per page is too restrictive	2/5	Cluster 3
Bulk-create field parity gap / level confusion	Mixed across 3/5, in different ways	Cluster 6

Two findings should anchor your case study (5/5, unanimous):

1. **Visibility/Draft model** — the system defaults courses to public (All TROA) on creation, while every participant expected a private draft state; "Restricted" was interpreted five different ways by five different people, meaning it currently communicates nothing reliably.
2. **Quiz attachment level** — every participant misjudged which level (page/ lesson/module) a quiz attaches to, including one participant (P3) who had the *correct* answer from reading the tree structure beforehand and was still visually contradicted by the UI — ruling out "just a training issue" as the full explanation.

A genuinely novel insight from P5 worth featuring: she nearly completed the entire course-creation task believing she was "done," because she entered content into a node's Summary field without realizing nodes are empty containers and real content lives in pages. This is a more severe failure mode than simple confusion — it's a case where the system let a user reach a false sense of completion. Worth a dedicated callout, since none of the other 4 participants hit this exact failure, but it's high-severity when it happens.

One tension worth naming rather than smoothing over: P1, P2, and P3 want more quiz flexibility (mixed question types); P5's prior background biased her toward valuing grading simplicity (multiple-choice only). This is a segmentation insight, not a contradiction — different course types may call for different quiz complexity.

All 5 participant sessions are now synthesized into this map. Ready to move into hypothesis scoring (using the completion tiers established earlier) or drafting the case study's key findings section — let me know which you'd like to tackle next.