

CH31 — Error States (global + per feature)

Doc ID: HZ-V1-CH31_Error_States_R1

Revision: R1 (2026-01-02)

Status: Draft

Depends on: CH04, CH06, CH07, CH08, CH12, CH15, CH17, CH18, CH19, CH20, CH21, CH22, CH23, CH24, CH25, CH28, CH29, CH30

Related: CH01, CH02, CH03, CH05, CH09, CH10, CH11, CH13, CH14, CH26, CH27, CH32, CH33, CH34, CH35, CH36, CH37, CH38

Supersedes: (none — first revision)

Owned Decisions (locked by this chapter): error taxonomy; severity levels; UI surfaces; message-copy rules; error IDs; per-surface fallback behavior; logging requirements for errors.

Open Questions / Placeholders:

- **PLACEHOLDER:** Localization scope for error copy • Owner: CH32 • Options: EN-only v1 / EN+ES / EN+Top-5 languages • Default: EN-only • Decide-by: before App Store submission
- **PLACEHOLDER:** Final brand tone for error copy (minimal vs friendly) • Owner: CH15 • Options: Minimal / Friendly / Coachy • Default: Minimal-friendly hybrid • Decide-by: CH15 lock
- **PLACEHOLDER:** Crash reporting vendor selection • Owner: CH28/CH33 • Options: Sentry / Firebase Crashlytics / other • Default: Firebase Crashlytics • Decide-by: R2 of CH28

1. Purpose and Scope

This chapter defines every error state in Handz V1: how errors are categorized, how they are surfaced in the UI, what exact copy appears, what actions are offered, how we log errors, and how we prevent error loops. CH31 does not redefine safety/abuse limits (owned by CH30) or offline/sync architecture (owned by CH28); it specifies the user-facing behavior when those systems fail or block.

Non-goal: inventing new product rules. If an error reveals missing product logic, create a PLACEHOLDER and reference the owning chapter.

2. Error Taxonomy

2.1 Severity levels (S0–S4)

- **S0 Info:** no user action required; optional toast/badge; never blocks a flow.
- **S1 Warning:** user can continue; warn about limits or partial failures; offer a safe default action.
- **S2 Recoverable Error:** action failed; user can retry or choose an alternative; show a clear next step.
- **S3 Blocking Error:** user cannot proceed on the current path; must resolve or exit; provide one primary action.
- **S4 Critical:** corrupted state / security block / crash risk; hard-stop and route to safe screen; capture diagnostics.

2.2 Error surfaces (where errors appear)

- **Toast** (2.5s): S0–S2; short message; optional single action (“Retry”).
- **Inline field error**: S1–S2 validation; appears under the field; clears on edit.
- **Inline empty-state panel**: “Nothing here yet” + primary CTA.
- **Banner** (persistent): S1–S3; sits under top bar; dismissible if S1–S2; non-dismissible if S3.
- **Modal**: S2–S4; requires acknowledgement; 1–2 actions max.
- **Full-screen fallback**: S3–S4; used when the entire screen cannot load or state is unsafe.

2.3 Copy rules (wording constraints)

- Be specific about *what happened* and *what to do next*. Avoid vague “Something went wrong.”
- Never blame the user. Use neutral language (“We couldn’t…”).
- One primary action. Optional secondary action is “Not now” or “Cancel.”
- When a restriction is due to plan state, say so once and link to the payroll or account screen (see CH25).
- When a restriction is due to safety/abuse controls, reference “Safety limits” and route to CH30-owned warning ladder UI (See: HZ-V1-CH30).
- Do not show raw server error strings. Log them; show user-facing copy.

2.4 Logging rules (analytics + debug)

- Every S2+ error must emit: `error_id`, `surface`, `screen_id`, `action_attempted`, `plan_state` (Guest/Free/Pro/Trial), `network_status`, `timestamp`, and a sanitized technical payload.
- S3+ must include a stable “`incident_id`” for support/debug correlation.
- Never log user-entered free text verbatim (privacy). Instead log length + field name.
- If a retry succeeds, emit a “recovered” event with the original `incident_id`.

3. Global Error Patterns (must be implemented once, reused everywhere)

3.1 Global error component set

- **HandzToast**: toast queue; max 1 visible; coalesce duplicates within 5s.
- **HandzBanner**: persistent banner region below top bar; supports single action button.
- **HandzEmptyState**: illustration slot + title + body + primary CTA + optional secondary link.
- **HandzModal**: standardized modal with title, body, primary CTA, secondary CTA.
- **HandzFullScreenFallback**: used when a screen cannot render safely; contains “Go back” and “Try again” where applicable.

3.2 Retry behavior standard

- If the failing call is idempotent (GET/read): allow unlimited retries, but add exponential backoff after 3 rapid retries.
- If the failing call is non-idempotent (POST/create): allow retry only if we have a client-generated request_id and server supports de-duplication (otherwise show “Try again later”).
- All retries must preserve user input; never clear forms on failure.

3.3 Degraded mode standard

- If network is offline: show a banner “Offline — some actions are unavailable” and disable network-dependent buttons with inline explanation.
- If entitlements cannot be verified: default to the safer side (Free behavior) but show banner “Can’t verify subscription — some Pro features may be locked until you reconnect.”
- If remote config fails: use built-in defaults; log S1 warning once per session.

4. Error Dictionary (authoritative list)

Each error has an ID (ERR-xxx). Implementers must map technical failures to these IDs. If a new failure is discovered that doesn't fit, add a new ERR entry in CH31 R2 (do not create ad-hoc copy in code).

4.1 Network and Service Availability

ERR-001 — No internet connection

Severity: S2 **Surface:** Banner (persistent) + Toast on attempted action

Trigger: Network status = offline OR request fails with offline indicator

User-facing copy:

Title: **Offline**

Body: You're not connected. You can keep viewing saved content, but actions that require internet won't work.

Primary action: OK (dismiss toast) / Banner has no CTA

Routing: Banner appears on all screens; dismiss only when network returns.

Logging: Emit on transition to offline and on first blocked action per screen.

ERR-002 — Request timed out

Severity: S2 **Surface:** Toast (with Retry)

Trigger: A request exceeds timeout threshold (client) or server times out

User-facing copy:

Title: **That took too long**

Body: Try again. If it keeps happening, check your connection.

Primary action: Retry

Secondary action: Not now

Routing: Retry repeats the same request; preserve user state.

Logging: Include endpoint_name and elapsed_ms (sanitized).

ERR-003 — Service temporarily unavailable

Severity: S3 **Surface:** Full-screen fallback

Trigger: Server returns 503/maintenance flag OR critical dependency down

User-facing copy:

Title: **Handz is down right now**

Body: We're having trouble on our side. Try again in a bit.

Primary action: Try again

Secondary action: Go back

Routing: Try again reloads the screen; Go back pops navigation stack; if root screen, routes to Dashboard.

Logging: Include status_code and request_id if present.

4.2 Authentication and Session

ERR-010 — Session expired

Severity: S3 **Surface:** Modal

Trigger: Auth token invalid/expired; server returns 401 on authenticated request

User-facing copy:

Title: **Session expired**

Body: Please log in again to continue.

Primary action: Log in

Secondary action: Not now

Routing: Primary routes to Welcome Gate / Login; on success, return user to the last attempted screen if safe.

Logging: Include last_screen_id and attempted_action.

ERR-011 — Apple/Google sign-in failed

Severity: S2 **Surface:** Modal

Trigger: 3rd-party auth returns error/cancel; or credential mismatch

User-facing copy:

Title: **Couldn't sign you in**

Body: Try again, or choose a different sign-in method.

Primary action: Try again

Secondary action: Use another method

Routing: Secondary routes to Welcome Gate with sign-in options expanded.

Logging: Do not log provider raw messages; log provider=apple/google and error_category.

ERR-012 — Email verification required

Severity: S3 **Surface:** Full-screen interstitial

Trigger: User signs up via email but has not verified; attempts to proceed

User-facing copy:

Title: **Verify your email**

Body: Check your inbox for a verification link, then come back here.

Primary action: Resend email

Secondary action: I already verified

Routing: Resend triggers verification email; "I already verified" re-checks status.

Logging: Log resend_count and cooldown.

4.3 Guest Restrictions (plan state gating)

ERR-020 — Guest cannot save flows

Severity: S3 **Surface:** Modal

Trigger: Guest attempts to save a flow OR create persistent content

User-facing copy:

Title: **Create an account to save**

Body: Guest mode lets you explore, but saving flows requires an account.

Primary action: Create account

Secondary action: Not now

Routing: Primary routes to Sign Up; returning user resumes editing (draft remains in memory until app closes).

Logging: Include action_attempted=save_flow and screen_id.

ERR-021 — Guest cannot customize move library

Severity: S2 **Surface:** Banner or Modal (when entering customization)

Trigger: Guest attempts to select disciplines/move packs, create custom moves, or edit defaults

User-facing copy:

Title: **Account required**

Body: To personalize your move library, create an account.

Primary action: Create account

Secondary action: Continue as guest

Routing: Continue as guest returns to previous screen without applying customization.

Logging: Track conversion_prompt_shown.

4.4 Free Plan Caps and Paywalls (See: HZ-V1-CH08, HZ-V1-CH25)

ERR-030 — Free saved flow limit reached

Severity: S3 **Surface:** Modal

Trigger: Free user already has 2 saved flows and attempts to save a new one or import to library

User-facing copy:

Title: **Flow limit reached**

Body: Free accounts can save up to 2 flows. Upgrade to save more, or delete one to continue.

Primary action: Upgrade

Secondary action: Manage flows

Routing: Upgrade opens paywall (CH25). Manage routes to Library (CH15) with multi-select enabled and helpful hint.

Logging: Include current_saved_flows_count.

ERR-031 — Practice is Pro (credits required)

Severity: S3 **Surface:** Paywall interstitial

Trigger: User taps Practice without entitlement OR no remaining practice credits (Free)

User-facing copy:

Title: **Practice Mode is Pro**

Body: Use Practice Mode to drill your paths and track progress. Start a 7-day trial or upgrade to Pro.

Primary action: Start trial / Upgrade

Secondary action: Not now

Routing: Primary opens subscription flow (StoreKit, CH25). Secondary returns to previous screen.

Logging: Include credits_remaining and source_screen.

ERR-032 — Free inbox cap reached

Severity: S2 **Surface:** Modal

Trigger: Free user inbox has 10 items and receives a new import

User-facing copy:

Title: **Inbox is full**

Body: Free accounts can hold up to 10 imports in Inbox. Delete one to make room.

Primary action: Open Inbox

Secondary action: Not now

Routing: Primary routes to Inbox (CH18) with bulk actions highlighted.

Logging: Include inbox_count.

4.5 Storage, Media, and Limits (See: HZ-V1-CH29)

ERR-040 — Upload size limit reached (2GB)

Severity: S3 **Surface:** Modal

Trigger: Pro user exceeds total upload cap OR upload attempt would exceed cap

User-facing copy:

Title: **Storage limit reached**

Body: You've hit your 2GB upload limit. Delete an upload to free space, or switch to links.

Primary action: Manage uploads

Secondary action: Use links instead

Routing: Manage uploads routes to Settings > Storage (CH29-defined). Use links opens “Add Link” flow.

Logging: Include used_bytes and attempted_bytes.

ERR-041 — Video upload not shareable

Severity: S1 **Surface:** Inline info + Toast

Trigger: User attaches local upload to a move/sequence and then attempts to share a flow

User-facing copy:

Title: **Uploads stay private**

Body: Uploaded videos don't travel with shared links. Use a link if you need others to see it.

Primary action: Got it

Routing: Show inline note near the video field and a one-time toast on first share attempt.

Logging: One-time per user per session; include share_attempted=true.

ERR-042 — Invalid video link

Severity: S2 **Surface:** Inline field error

Trigger: Link field fails validation (not a URL, unsupported host, or unreachable preview)

User-facing copy:

Title: **Check the link**

Body: That link doesn't look usable. Paste a direct URL to the video.

Primary action: OK

Routing: Inline error under link field; Save disabled until fixed.

Logging: Log host_domain only.

4.6 Flow Builder (See: HZ-V1-CH12–CH14)

ERR-050 — Cannot add branch (branch cap reached)

Severity: S2 **Surface:** Toast + disabled button tooltip

Trigger: Node already has 10 outgoing branches (cap from CH00 lock; owned behavior here)

User-facing copy:

Title: **Branch limit reached**

Body: This move already has 10 next options. Replace one or merge paths.

Primary action: OK

Routing: Disable “Add branch” on that node; long-press shows tooltip with this copy.

Logging: Include node_id and outgoing_count.

ERR-051 — Invalid connection

Severity: S2 **Surface:** Toast

Trigger: User tries to connect a node to itself OR creates a direct cycle disallowed by CH12 rules

User-facing copy:

Title: **Can’t connect that way**

Body: That connection would create an invalid loop. Try connecting to a different move.

Primary action: OK

Routing: Connection is not created; canvas remains unchanged.

Logging: Include source_node_id and target_node_id.

ERR-052 — Unsaved changes warning

Severity: S1 **Surface:** Modal

Trigger: User attempts to exit editor with dirty state

User-facing copy:

Title: **Discard changes?**

Body: You have unsaved changes. Do you want to keep editing or discard?

Primary action: Keep editing

Secondary action: Discard

Routing: Discard exits to previous screen; Keep editing closes modal.

Logging: Include time_in_editor_seconds.

4.7 Sharing, Inbox, Imports (See: HZ-V1-CH17–CH19)

ERR-060 — Share link creation failed

Severity: S2 **Surface:** Toast (Retry)

Trigger: User taps “Create share link” and backend fails or rate-limits

User-facing copy:

Title: **Couldn't create link**

Body: Try again. If it keeps failing, check your connection.

Primary action: Retry

Secondary action: Not now

Routing: Retry repeats link creation with request_id; preserve flow state.

Logging: Include link_type and rate_limit_bucket if present.

ERR-061 — Share link expired or revoked

Severity: S3 **Surface:** Full-screen fallback (link viewer)

Trigger: Viewer opens a link that is invalid/expired/revoked

User-facing copy:

Title: **Link not available**

Body: This share link has expired or was revoked.

Primary action: Go back

Routing: Go back returns to prior app screen; if opened from outside, routes to Welcome Gate.

Logging: Include share_id hash.

ERR-062 — Import contains missing moves

Severity: S2 **Surface:** Modal (resolver)

Trigger: Import payload references moves not in receiver's library

User-facing copy:

Title: **Some moves are missing**

Body: This flow includes moves you don't have yet. Choose how you want to handle them.

Primary action: Review & import

Secondary action: Cancel

Routing: Primary routes to Import Conflict Resolver (CH19). Cancel keeps item in Inbox.

Logging: Include missing_moves_count and has_custom_payload.

ERR-063 — Free user cannot practice inbox items

Severity: S2 **Surface:** Banner on Inbox item + disabled CTA

Trigger: Free user views an Inbox item and taps Practice

User-facing copy:

Title: **Practice requires saving**

Body: To practice this, save it to your library first. Practice works on saved flows only.

Primary action: Save to library

Routing: Primary triggers save-to-library; if saved-flow cap reached, show ERR-030.

Logging: Include inbox_item_id.

4.8 Practice Mode (See: HZ-V1-CH20–CH22)

ERR-070 — Practice session failed to start

Severity: S3 **Surface:** Full-screen fallback (Practice)

Trigger: Practice config invalid OR required data missing (e.g., selected path references deleted node)

User-facing copy:

Title: **Can't start practice**

Body: One of your selected paths is missing. Edit your selection and try again.

Primary action: Edit selection

Secondary action: Go back

Routing: Edit selection returns to Practice Setup with invalid paths pre-highlighted.

Logging: Include invalid_path_count.

ERR-071 — Timer desync detected

Severity: S2 **Surface:** Banner

Trigger: App backgrounded/resumed and timer drift exceeds tolerance

User-facing copy:

Title: **Timer adjusted**

Body: We corrected the timer based on actual time passed.

Primary action: OK

Routing: Banner auto-dismiss after 5s; log drift_seconds.

Logging: Include drift_seconds and app_state_transitions.

ERR-072 — Practice log save failed

Severity: S2 **Surface:** Modal

Trigger: Session ends but log cannot be saved (network/server issue)

User-facing copy:

Title: **Couldn't save your session**

Body: Your session is stored on this device for now. We'll try again when you're back online.

Primary action: OK

Routing: Store as pending_log; show small badge in History until synced (See: CH28).

Logging: Include pending_log_id and retry_schedule.

4.9 Mastery and Maintenance (See: HZ-V1-CH23–CH24)

ERR-080 — Maintenance overload

Severity: S1 **Surface:** Inline panel

Trigger: User selects too many paths for maintenance; schedule exceeds user-selected weekly capacity

User-facing copy:

Title: **That's a lot to maintain**

Body: At your current schedule, this plan may feel overwhelming. Consider narrowing to your top paths.

Primary action: Adjust plan

Secondary action: Keep anyway

Routing: Adjust opens maintenance plan editor. Keep anyway proceeds and marks plan as "High load".

Logging: Include planned_minutes_per_week and user_capacity_minutes.

ERR-081 — Mastery confidence mismatch

Severity: S1 **Surface:** Inline prompt

Trigger: User indicates mastery feels wrong (manual downgrade)

User-facing copy:

Title: **Adjust mastery**

Body: If this doesn't feel locked in, you can step it back. That updates your plan going forward.

Primary action: Edit mastery

Routing: Opens mastery edit screen (CH23).

Logging: Include prior_mastery_state and new_state.

4.10 Permissions (camera, photos, notifications)

ERR-090 — Photos permission denied

Severity: S3 **Surface:** Modal

Trigger: User tries to attach media but Photos permission is denied

User-facing copy:

Title: **Photos access needed**

Body: Allow Photos access in Settings to upload videos from your library.

Primary action: Open Settings

Secondary action: Not now

Routing: Open Settings deep-links to iOS Settings for Handz.

Logging: Include permission_status.

ERR-091 — Notifications disabled

Severity: S1 **Surface:** Banner

Trigger: User enables reminders in-app but OS notifications are disabled

User-facing copy:

Title: **Notifications are off**

Body: To receive reminders, turn on notifications in Settings.

Primary action: Open Settings

Secondary action: Later

Routing: Later dismisses; banner reappears when user revisits Notifications screen.

Logging: Include reminder_type.

5. Empty States (authoritative)

Empty states must feel like guidance, not dead ends. Each includes a title, one-sentence explanation, and a primary CTA.

EMPTY-01 — Library has no saved flows

Title: Start your first flow

Primary CTA: Create Flow

Behavior: Routes to Flow Builder with move picker open.

EMPTY-02 — Moves list is empty (non-guest)

Title: Add a move to start building flows

Primary CTA: Add Move

Behavior: Routes to Add Move screen (CH09–CH11).

EMPTY-03 — Inbox is empty

Title: No imports yet

Primary CTA: Learn how sharing works

Behavior: Opens a short explainer (CH15/CH17 copy).

EMPTY-04 — Practice history empty

Title: Your practice sessions will show up here

Primary CTA: Start a session

Behavior: If not entitled, routes to paywall; otherwise to Practice Setup.

EMPTY-05 — Maintenance page empty

Title: Pick a gameplan to maintain

Primary CTA: Create Gameplan

Behavior: Routes to gameplan builder (CH23).

6. Screen-Level Error Requirements (mapping)

This section lists required error handling per major screen group. Detailed behavior lives in the Error Dictionary above; screens must implement the relevant ERR IDs and empty states.

Welcome Gate / Auth screens (CH07)

- ERR-001
- ERR-010
- ERR-011
- ERR-012

Dashboard / Library (CH15)

- EMPTY-01
- ERR-001
- ERR-002
- ERR-003
- ERR-030

Moves (CH09–CH11)

- EMPTY-02
- ERR-001
- ERR-002
- ERR-042

Flow Builder (CH12–CH14)

- ERR-001
- ERR-002
- ERR-050
- ERR-051
- ERR-052

Flow Detail (CH16)

- ERR-001
- ERR-060
- ERR-041
- ERR-030
- ERR-031

Sharing / Link Viewer (CH17)

- ERR-001

- ERR-061
- ERR-003

Inbox (CH18)

- EMPTY-03
- ERR-032
- ERR-062
- ERR-063
- ERR-030

Practice Setup/Active/Complete (CH20–CH22)

- ERR-031
- ERR-070
- ERR-071
- ERR-072
- EMPTY-04

Gameplans/Mastery/Maintenance (CH23–CH24)

- ERR-080
- ERR-081
- EMPTY-05

Settings/Notifications (CH27)

- ERR-091
- ERR-090

Uploads/Storage (CH29)

- ERR-040
- ERR-090
- ERR-041

Safety/Abuse (CH30)

- See: HZ-V1-CH30 (Warning ladder defines messaging + escalation)

7. Acceptance Test Checklist (Given / When / Then)

Tests are written so a non-technical reviewer can validate behavior exactly.

Global toast behavior

- Given I trigger the same S2 error twice within 5 seconds, when the second error occurs, then I see only one toast and no duplicate stacking.
- Given a toast with a Retry action is shown, when I tap Retry, then the same request is attempted again and I remain on the same screen with my inputs intact.

Offline banner

- Given my device goes offline, when I am on any screen, then an “Offline” banner appears within 2 seconds.
- Given I am offline, when I tap a network-only action (Create share link), then I see ERR-001 toast and the action does not proceed.

Guest save gate

- Given I am in Guest mode, when I tap Save on a flow, then I see ERR-020 and I cannot save.
- Given I choose Create account from ERR-020, when I complete sign up, then I return to the editor with my draft still present (unless the app was killed).

Free saved flow cap

- Given I am Free and already have 2 saved flows, when I attempt to save another flow, then I see ERR-030 with actions Upgrade and Manage flows.
- Given I tap Manage flows, when I arrive in Library, then multi-select mode is enabled and I see a hint explaining I must delete or upgrade.

Upload share warning

- Given a flow contains an uploaded (private) video, when I open the Share screen, then I see ERR-041 info explaining uploads do not travel with links.

Practice log failure

- Given I complete a practice session while offline, when the session ends, then I see ERR-072 and a pending log appears in History with a “Syncing” indicator.
- Given the device reconnects, when the app is opened, then the pending log is retried and on success the “Syncing” indicator disappears.

Link revoked

- Given I open a revoked share link, when the link viewer loads, then I see ERR-061 full-screen message and no flow content.

8. Replit Build Prompt (CH31 only)

Copy/paste this into Replit Agent with CH00 + CH31 attached.

PROJECT: Handz iOS (portrait-only). Implement CH31 Error States exactly.

INPUTS: You have PDFs CH00 and CH31.

RULES:

- Do not invent product logic outside CH31. If missing, create a TODO with 'PRD PLACEHOLDER' and stop that feature.
- Implement error surfaces (Toast, Banner, Inline, Modal, Full-screen) as reusable components.
- Map network/auth/storage/practice/share failures to the ERR IDs defined in CH31.

BUILD TASKS (in order):

1) Create Error model:

- enum Severity { S0,S1,S2,S3,S4 }
- type HandzError { id, severity, title, body, primaryAction?, secondaryAction?, route?, meta? }

2) Implement UI components:

- HandzToastQueue (dedupe within 5s, max 1 visible)
- HandzBannerHost (persistent region, supports action)
- HandzModal (standard buttons, focus-safe)
- HandzFullScreenFallback (Try again / Go back variants)
- HandzEmptyState (title/body/CTA)

3) Implement global hooks:

- useNetworkStatus() => online/offline; emits ERR-001 banner
- useEntitlementStatus() => plan state; if unknown, degrade to Free + banner

4) Implement error mapping utilities:

- mapAxiosOrFetchErrorToHandzError() => ERR-001/002/003
- mapAuthErrorToHandzError() => ERR-010/011/012
- mapStorageErrorToHandzError() => ERR-040/042

5) Wire screen integrations:

- Add error handling to Share link create (ERR-060)
- Add guest save gate to Flow save (ERR-020)
- Add free cap modal to save/import (ERR-030)
- Add practice log pending flow (ERR-072) storing pending logs locally and retrying on reconnect

6) Add analytics logging:

- emit error_shown, error_action_clicked, error_recovered with incident_id for S3+
- do not log user free text

DELIVERABLES:

- Components + hooks + mapping utils
- Example integrations on at least 3 screens (Flow Builder, Share, Practice)
- Unit tests for mapping utilities
- Manual test checklist matching CH31 acceptance tests

9. Troubleshooting Notes (CH31)

Use this to diagnose issues during vibe coding and QA.

- **Duplicate toasts:** Ensure toast queue dedupes by (error_id + screen_id + action_attempted) for 5 seconds.
- **Endless retry loops:** Never auto-retry POST/create actions without request de-duplication; add manual Retry only.
- **Practice logs missing:** Confirm pending logs are persisted locally before showing ERR-072; verify retry triggers on reconnect/app open.
- **Paywall shown incorrectly:** If entitlement fetch fails, degraded Free mode should show a banner explaining verification issue; don't silently lock.
- **Link viewer blank screen:** Must route to ERR-061 or ERR-003 fallback; never render empty canvas without copy.
- **Field validation feels harsh:** Use inline errors; disable Save with explanation rather than modal spam.