**MCPX‑KendoBridge — Error Catalog & Envelope Standard**

**Document:** docs/error\_catalog.docx  
**Project Code:** MCPX‑KendoBridge  
**Version:** 2.0.0 (Kendo Migration)  
**Last Updated:** 2025‑09‑27  
**Owner:** DoSE (Accountable) — DocFactory (Author/Responsible) — SRE/QA/SecLead/T‑Arch (Consulted)

**Purpose.** Define a **canonical error envelope** and a **stable taxonomy of error codes** for the MCPX‑KendoBridge proxy and Admin Portal. This catalog ensures consistent behavior across **API**, **UI**, **tests**, **monitoring**, **runbooks**, and **OpenAPI 3.1**. It aligns with Technijian guardrails: GitHub‑first SDLC, four environments (**Alpha → Beta → RTM → Prod**), **No‑Hard‑Coding**, **Stored‑procedure‑only**, **add‑only** DB changes, and secrets only in environment stores.

**Compliance banner.** Dynamic behavior (timeouts, heartbeat cadence, origin allow‑list, child command/args/cwd, feature flags) is **DB‑sourced via SPs** (sp\_Config\_\*, sp\_Feature\_IsEnabled, sp\_Lookup\_Get) and **never hard‑coded**; secrets (SQL connection strings, Telerik license) are **never** stored in code/DB/logs and live only in **GitHub Environments**. This catalog does not introduce secrets in examples or logs.

**1) Scope & References**

* **Applies to:** API endpoints (/mcp, /healthz, /ready, /config/effective, legacy /messages, /sse) and Admin Portal user‑visible messages.
* **OpenAPI 3.1 of record:** api/openapi/mcp-proxy.yaml (includes examples for the envelope and codes).
* **Runbooks:** deploy, rollback, incident, scale‑out (map each error to actions).
* **Monitoring:** SLI/SLOs for Availability, JSON latency, **SSE TTFB**, heartbeat cadence (error codes power panels/alerts).

**2) Canonical Error Envelope (all endpoints)**

**Shape (required on any non‑2xx):**

{ "code": "<kebab\_case\_code>", "message": "<human-readable>", "requestId": "req-<opaque>" }

**Rules**

1. **Stable code** string; never change or remove once released.
2. **message** is safe for logs and end‑users (no secrets, no payload echoes).
3. **requestId** is included when available and MUST be present in server logs; never include tokens or PII.
4. Envelope is returned with the appropriate **HTTP status**; envelope is **never HTML**.
5. UI surfaces the message and a short help text; detailed troubleshooting goes to runbooks.
6. **Logging redaction:** No request bodies or secrets; structured logs include requestId, sessionId, childPid, status, path, latency\_ms, mode=json|sse. (See Monitoring doc.)

**3) Error Code Taxonomy (authoritative list)**

| **Code** | **HTTP** | **Category** | **Typical Cause** | **Client Guidance** | **Server Action** | **Runbook / Alert** |
| --- | --- | --- | --- | --- | --- | --- |
| origin\_forbidden | 403 | Policy | Origin not in DB allow‑list | Do not retry; ensure origin is allow‑listed | None; log policy hit | Incident/Compliance; alert on surge |
| missing\_session\_id | 400 | Client | GET /mcp without Mcp-Session-Id | Retry with header | None | Tests cover; low alert |
| feature\_disabled | 403 | Policy/Flag | Legacy endpoints with flag off | Switch to /mcp or contact ops | None | Low; track regressions |
| timeout | 408 | Network/Perf | Exceeded Network:RequestTimeoutSeconds | Retry with backoff | Check perf/ingress | P2 if sustained; perf smoke |
| not\_ready | 503 | Startup/Deps | DB/SP unreachable or child probe failing | Retry later | Fix deps; gate rollout | P1 if prod; readiness down |
| spawn\_failed | 500 | Child/Process | Kendo MCP child couldn’t start | Retry session; if persists, incident | Inspect child cmd/args (DB) | P1/P2 based on scope |
| bad\_gateway\_child\_unavailable | 502 | Child/Process | Child crashed or unreachable | Retry session | Restart child; inspect logs | P1 if wave‑wide |
| unauthorized | 401 | Auth | Missing/invalid bearer (if enforced) | Re‑auth | Check gateway/IdP | Security runbook |
| bad\_request | 400 | Client | Malformed JSON‑RPC, invalid schema | Fix request | None | Tests cover |
| rate\_limited *(future)* | 429 | Throttle | Rate limit exceeded | Retry after delay | Tune quotas | Optional alerts |
| internal\_error | 500 | Unknown | Unhandled exception | Retry with backoff | Triage; add specific code | P1 if spike |

**Notes**

* Codes above are **enumerated in OpenAPI** and **valid across all environments**.
* New codes require ADR or catalog update; see §9.
* “Future” codes may appear in OpenAPI for planning but **must** be gated behind feature flags if not active.

**4) Detailed Definitions & Examples**

**4.1 origin\_forbidden (403)**

* **When:** Origin header fails allow‑list (Security:AllowedOrigins) sourced from DB.
* **Envelope example:**
* { "code": "origin\_forbidden", "message": "Origin not allowed", "requestId": "req-abc123" }
* **Monitoring:** Track errors\_total{code="origin\_forbidden"}; alert on abnormal spikes (possible misconfig or attack).
* **Runbook:** Check DB config via /config/effective, then **Config Rollback** if mis‑set.

**4.2 missing\_session\_id (400)**

* **When:** GET /mcp or legacy GET /sse missing Mcp-Session-Id.
* **Client:** Re‑issue with the header (server issues header on first POST /mcp).
* **Tests:** 01\_session\_establish and unit tests for header validator.

**4.3 feature\_disabled (403)**

* **When:** Call to /messages or /sse while EnableLegacyHttpSse=false.
* **Ops:** Prefer /mcp endpoints; only toggle the flag with justification and Evidence entry.

**4.4 timeout (408)**

* **When:** Server exceeded DB‑sourced request timeout.
* **Monitoring:** Observe latency histograms and **SSE TTFB** trends; investigate ingress buffering.

**4.5 not\_ready (503)**

* **When:** Readiness probe fails (DB/SP unreachable; optional child spawn probe fail).
* **RTM:** Always validates against **Prod DB (read‑only)** before Prod promotion.

**4.6 spawn\_failed (500)**

* **When:** Failed to start the child MCP process using DB‑configured Mcp:Child\*.
* **Ops:** Verify Mcp:ChildCommand, Mcp:ChildArgs, Mcp:ChildCwd via /config/effective; check file system/permissions.

**4.7 bad\_gateway\_child\_unavailable (502)**

* **When:** Child crashes mid‑stream or is unreachable.
* **Ops:** Check child\_restart\_count, CPU pressure, session spikes; consider **scale‑out**.

**4.8 unauthorized (401)**

* **When:** Missing/invalid bearer when gateway enforcement is enabled.
* **Logs:** Never log token contents; redact headers.

**4.9 bad\_request (400)**

* **When:** JSON‑RPC payload malformed or violates schema.
* **Developer:** Validate params before POST; rely on OpenAPI‑generated clients.

**4.10 rate\_limited (429, future)**

* **When:** Rate‑limit controls enabled.
* **Headers:** Include standard Retry-After if/when introduced.

**4.11 internal\_error (500)**

* **When:** Unexpected server error.
* **Ops:** Triage, capture requestId, convert to specific codes where possible.

**5) Mapping to Monitoring, Alerts & Evidence**

* **Metrics:** errors\_total{code="…"}, http\_requests\_total, latency histograms, **sse\_ttfb\_ms\_bucket**, heartbeat gap.
* **Dashboards:** Group by code; highlight origin\_forbidden, not\_ready, spawn\_failed, bad\_gateway\_child\_unavailable.
* **Alerts:**
  + P1: not\_ready sustained, internal\_error spike, **SSE TTFB p95 > 200 ms** (15 min).
  + P2: origin\_forbidden surge, timeout sustained.
* **Evidence:** Include error breakdown screenshots and post‑release 24‑h snapshots in the Evidence Pack per release (retain ≥ 1 year).

**6) UI Presentation (Admin Portal)**

* Display the **message** with a neutral tone; avoid technical leakage.
* Provide **next steps** links to runbooks or help docs (no secrets or IDs).
* For streaming views, show **connection state** and **heartbeat age**; on feature\_disabled suggest using /mcp.
* Accessibility: ensure aria‑live polite announcements for new error banners (WCAG 2.2 AA).

**7) Server Implementation Notes (non‑secret)**

**Envelope helper (C# sketch):**

public sealed record ErrorEnvelope(string code, string message, string? requestId = null);

IResult ToError(int status, string code, string message, HttpContext ctx)

{

var reqId = ctx.TraceIdentifier;

ctx.Response.StatusCode = status;

var env = new ErrorEnvelope(code, message, reqId);

// Log structured (no payload bodies, no secrets)

\_logger.LogWarning("err code={Code} status={Status} requestId={ReqId} path={Path}",

code, status, reqId, ctx.Request.Path);

return Results.Json(env);

}

**SSE path:** on error after some frames, send a final JSON event: message with error envelope semantics if applicable, then close gracefully (avoid partial frames).

**8) Test Coverage (what must be proved)**

* **Unit:** envelope shape; header validation; origin policy; timeout translation; mapping from exceptions to codes.
* **Contract tests:** OpenAPI #/components/responses/Error examples for each code.
* **E2E (Gherkin):**
  + 04\_origin\_denied → origin\_forbidden 403
  + Missing session header → missing\_session\_id 400
  + Legacy disabled → feature\_disabled 403
  + Readiness down → not\_ready 503
* **Perf:** ensure error handling does not inflate **SSE TTFB** beyond budget.
* **Evidence:** include failing/passing samples and the OpenAPI lint/diff output.

**9) Governance for New/Changed Codes**

1. **Naming:** kebab\_case verbs/nouns; concise and expressive.
2. **Back‑compat:** Adding new codes is allowed; **changing/removing existing codes is not** without a major version bump and migration plan.
3. **Process:** Propose via PR updating this catalog + OpenAPI + tests; include a short ADR entry.
4. **Docs:** Update Monitoring (panels/alerts), Runbooks (triggers/actions), and UI copy if user‑visible.
5. **Evidence:** Record the change in the Release with lint/diff output and test updates.

**10) Quick Reference — Codes & HTTP**

400: missing\_session\_id | bad\_request

401: unauthorized

403: origin\_forbidden | feature\_disabled

408: timeout

500: spawn\_failed | internal\_error

502: bad\_gateway\_child\_unavailable

503: not\_ready

429: rate\_limited (future)

**11) Cross‑References**

* **OpenAPI 3.1:** api/openapi/mcp-proxy.yaml — error examples & headers.
* **FR/NFR:** transport, sessioning, readiness, streaming budgets.
* **Monitoring & SLOs:** thresholds and dashboards; TTFB & heartbeat SLIs.
* **Compliance:** secrets policy, DB/SP rules, No‑Hard‑Coding.
* **Runbooks:** deploy, rollback, incident, scale‑out — actions tied to codes.

**12) Assumptions**

* Ingress **does not buffer** text/event-stream; read/idle timeouts allow heartbeats.
* RTM validates against **Prod DB (read‑only)**; errors in RTM must mirror Prod semantics.
* App logs contain **requestId** but **never** payload bodies or secrets.

**13) Next Steps**

* Wire errors\_total{code} metrics; add panels & alerts per §5.
* Ensure OpenAPI and tests reference this catalog as the **single source of truth**.
* Review codes quarterly; add specific codes to replace internal\_error buckets where patterns emerge.

**Footer (optional for Word header/footer):**  
*MCPX‑KendoBridge • Error Catalog & Envelope • v2.0.0 • 2025‑09‑27 • Confidential — Technijian Internal*