**MCPX‑KendoBridge — Admin Portal Ops Supplement (KendoReact Fluent v12, Read‑Only)**

**Document:** docs/08b\_ui\_ops\_supplement.docx  
**Project Code:** MCPX‑KendoBridge  
**Version:** 2.0.0 (Tech‑stack migration finalized)  
**Last Updated:** 2025‑09‑27  
**Owner:** UI Lead (Responsible) — DoSE (Accountable) — DocFactory (Author) — SRE, SecLead, Dev Lead (Consulted)

**Purpose.** Operational specification for the **read‑only Admin Portal** that supports day‑to‑day operations and release verification of MCPX‑KendoBridge. It details **routes, components, SSE health instrumentation, CSP/egress policy, a11y/perf budgets, build & license handling, and evidence hooks**. Aligns with Technijian guardrails: **GitHub‑first**, **Alpha → Beta → RTM → Prod**, **Add‑only schema**, **Stored‑procedure‑only DAL**, **No‑Hard‑Coding** (all dynamic values via DB SPs), and **secrets only in environment stores**.

**Compliance banner (UI applies these rules everywhere):**  
**No‑Hard‑Coding** of dynamic values (timeouts, heartbeat cadence, allowed origins, child command/args/cwd); the UI reads **non‑secret** runtime values from /config/effective, which the API sources via **SPs** (sp\_Config\_\*, sp\_Feature\_IsEnabled, sp\_Lookup\_Get). UI is **read‑only**; **no** DB writes; **no secrets** rendered or logged.

**1) Scope & Non‑Goals**

* **In scope:** Read‑only dashboards for **Health/Ready**, **Streaming/Sessions**, **Config (non‑secret)**, **Access/Policy** (Origin allow‑list visibility), release‑ops widgets (snapshots & quick probes).
* **Out of scope:** Any configuration editing; token handling UI; license display; developer tooling.

**2) Routes, Pages & Component Map (KendoReact Fluent v12)**

| **Route** | **Purpose** | **Kendo Components** | **Data Sources** |
| --- | --- | --- | --- |
| / (Dashboard) | Ops landing: uptime, session count, child processes, readiness | AppBar, TileLayout, Card, Skeleton | GET /healthz, GET /ready |
| /sessions | Stream quality: **SSE TTFB**, heartbeat age, active streams, fan‑out | Grid, Badge, ProgressBar, Chart(optional), Tooltip | POST /mcp (SSE probe), GET /mcp (SSE sub) |
| /config | **Non‑secret** config visibility | Grid, SearchBox, PanelBar | GET /config/effective |
| /access | Policy visibility: **Origin allow‑list**; legacy flag state | ListView, Badge, Notification | GET /config/effective |
| /about | Build & version metadata (no secrets) | Card, Grid | Static from image labels/env at render |

**Theming SOP.** Figma Make → ThemeBuilder → **Kendo Fluent v12** overrides; import base @progress/kendo-theme-fluent then ThemeBuilder design‑token package; verify parity and **WCAG 2.2 AA**. (This supplements docs/08\_ui\_ux.docx.)

**3) Streaming Instrumentation (TTFB & Heartbeats)**

**Goals.** Make streaming health **visible** and **testable** from the UI without exposing secrets.

**3.1 SSE probe (client‑initiated, read‑only)**

* **TTFB p95 ≤ 200 ms** (intra‑VPC budget) is displayed and color‑coded.
* **Heartbeat cadence** equals Network:SseKeepAliveSeconds (±1 s), calculated as the time delta between successive : comment lines.
* **Monotonic id** check for event: message frames.

**React sketch (browser SSE via fetch + ReadableStream):**

// /web/src/ops/sseProbe.ts

export async function sseProbe(baseUrl: string, sessionId: string, body: unknown) {

const started = performance.now();

const res = await fetch(`${baseUrl}/mcp`, {

method: 'POST',

headers: {

'Accept': 'text/event-stream',

'Content-Type': 'application/json',

'Mcp-Session-Id': sessionId

},

body: JSON.stringify(body)

});

const reader = (res.body as ReadableStream<Uint8Array>).getReader();

let ttfbMs = 0, lastBeat = 0, beats: number[] = [], lastId = 0;

const dec = new TextDecoder();

let buf = '';

// First chunk → compute TTFB

const first = await reader.read();

if (!first.done) {

ttfbMs = Math.round(performance.now() - started);

buf += dec.decode(first.value, { stream: true });

}

// Continue streaming

while (true) {

const { value, done } = await reader.read();

if (done) break;

buf += dec.decode(value, { stream: true });

// Parse SSE lines

let line;

while ((line = buf.substring(0, buf.indexOf('\n') + 1)) && line.length > 1) {

buf = buf.slice(line.length);

const trimmed = line.trim();

if (trimmed.startsWith(':')) { // heartbeat

const now = performance.now();

if (lastBeat) beats.push(Math.round(now - lastBeat));

lastBeat = now;

} else if (trimmed.startsWith('id:')) {

const id = Number(trimmed.slice(3).trim());

if (!Number.isNaN(id)) lastId = Math.max(lastId, id);

}

}

}

return { ttfbMs, heartbeatGapsMs: beats, maxId: lastId };

}

**UI usage:** Run probe on demand (Beta/Prod) and render **TTFB** and **heartbeat gap** distributions; do **not** persist results. Targets and thresholds mirror **NFR** and **Monitoring** docs.

**4) Data Sources & Read‑Only Contracts**

* /healthz, /ready — liveness/readiness displays; **no secrets**, JSON only.
* /config/effective — **non‑secret** key/values (child cmd/args/cwd, allowed origins, timeouts).
* /mcp — SSE probe and session stream view (read‑only).

**Never** render or log secrets; **never** issue DB writes from UI. All runtime values ultimately come from SQL via **SPs**; the UI simply consumes the API surfaces.

**5) Security & CSP (front‑end)**

**CSP (baseline; tighten per env):**

default-src 'none';

style-src 'self';

script-src 'self';

img-src 'self';

font-src 'self';

connect-src 'self' https://alpha.example.com https://beta.example.com https://rtm.example.com https://prod.example.com;

base-uri 'none';

frame-ancestors 'none';

* **Same‑origin** only; **no** external images/CDNs.
* Enforce security headers: X-Content-Type-Options, Referrer-Policy, X-Frame-Options or CSP frame-ancestors, COOP/CORP, HSTS.
* **Origin policy** is enforced **server‑side**; UI merely **displays** the allow‑list from /config/effective.

**6) Accessibility & UX States (WCAG 2.2 AA)**

* **Axe smoke** on /, /sessions, /config, /access.
* **ARIA live**: Streaming status banners (Connected, Reconnecting, Disconnected, Draining) announced politely.
* **Focus management:** Return focus to first actionable element after route changes; visible outline consistent with Fluent token.
* **Color contrast:** ≥ 4.5:1 for text; ≥ 3:1 for non‑text UI; use ThemeBuilder tokens.
* **Keyboard support:** All popovers/dialogs/drawers are tabbable with ESC to close.

**7) Performance Budgets & Telemetry (client‑side)**

* **JSON mode latency:** display p50/p95 from ops probes; target **≤ 300 ms / ≤ 800 ms**.
* **SSE TTFB:** display p50/p95; **≤ 200 ms** target; warn at 180–200 ms band.
* **Heartbeat gap:** target equal to Network:SseKeepAliveSeconds (±1 s).
* **No secret logging:** browser console **must not** print headers, tokens, or payloads.
* **Synthetic checks:** optionally run lightweight probes every 5–10 min (non‑authenticated) in Beta/Prod. Results are **not** persisted by UI; external monitoring stores them.

**8) Build & Licensing (KendoReact) — build‑time only**

* CI injects **Telerik license** at build time using TELERIK\_LICENSE / TELERIK\_LICENSE\_PATH secrets (GitHub Environments); never stored in code, images, or UI bundles.
* UI build step masks and shreds any temp license file; see CI workflow and **Rotate License** runbook.

**9) Evidence Hooks (what the UI helps capture)**

* One‑click **snapshots** (client‑side) of /ready, /healthz, /config/effective to assist release Evidence; **not** stored by UI (operators download JSON).
* Inline **perf probe** results can be downloaded as JSON for Evidence (non‑secret).
* Display **RTM parity** banners if /config/effective differs from expected Prod values (hash comparison fed by CI).

**10) Error Presentation (maps to Error Catalog)**

Render the canonical envelope { code, message, requestId? } with friendly guidance and zero payload echo. Code → help:

| **Code** | **UI Copy (concise)** | **Action** |
| --- | --- | --- |
| origin\_forbidden | “This origin isn’t allowed.” | Check **Access** page; contact ops |
| missing\_session\_id | “Session header required.” | UI auto‑retries with header |
| feature\_disabled | “Legacy endpoint disabled.” | Use /mcp |
| timeout | “Request timed out.” | Retry; see **Sessions** probe |
| not\_ready | “Service not ready.” | Check **Dashboard**; escalate if Prod |
| spawn\_failed/bad\_gateway\_child\_unavailable | “Tool process unavailable.” | See **Sessions**; capture evidence |
| unauthorized | “Please sign in.” | (If gateway enforces bearer) |

See docs/error\_catalog.docx and OpenAPI for full list/examples.

**11) Acceptance Criteria (UI Ops)**

1. **Routes render** with zero external egress; CSP headers effective.
2. **SSE probe** shows TTFB p95 ≤ 200 ms and heartbeat cadence ≈ config value.
3. **Config page** lists **only non‑secret** keys; copy‑to‑clipboard available.
4. **Access page** shows **Origin allow‑list** and legacy flag state.
5. **Axe smoke:** no **critical** violations on all pages.
6. **Evidence actions** present to download snapshots (no server storage).
7. **RTM banner** appears when parity mismatch is detected.
8. **No secrets** in logs, UI, or network tab.

**12) RACI (Admin Portal Ops)**

| **Activity** | **A** | **R** | **C** | **I** |
| --- | --- | --- | --- | --- |
| UI ops spec & upkeep | DoSE | UI Lead | SRE, Dev Lead | QA |
| SSE probes & dashboards | DoSE | UI Lead | SRE | QA |
| A11y & CSP compliance | DoSE | UI Lead | SecLead | QA |
| Evidence capture helpers | DoSE | UI Lead | SRE, DocFactory | All |

**13) Risks & Mitigations**

| **Risk** | **Impact** | **Mitigation** |
| --- | --- | --- |
| Ingress buffers SSE | Probes show false negatives/TTFB spikes | Enforce ingress settings; validate with CI conformance tests |
| UI accidentally logs secrets | Compliance breach | Lint rule & review; disable verbose logging; QA scan bundles |
| External asset inclusion | CSP violation | Block external CDNs; bundle fonts/assets locally |
| Drift between RTM and Prod | Prod failure risk | RTM parity banners; block promotion per CI/CD plan |

**14) Implementation Notes (non‑secret)**

**Route shell (React):**

<AppBar>…</AppBar>

<Drawer>…</Drawer>

<Routes>

<Route path="/" element={<Dashboard />} />

<Route path="/sessions" element={<Sessions />} />

<Route path="/config" element={<Config />} />

<Route path="/access" element={<Access />} />

<Route path="/about" element={<About />} />

</Routes>

**Config grid (read‑only):**

// GET /config/effective → { [key:string]: string }

const rows = Object.entries(config).map(([k,v]) => ({ key:k, value:v }));

// Render with KendoReact Grid (no editing)

**Stream status banner:** show state machine: idle → connecting → connected → heartbeat-ok | heartbeat-drift → draining → closed.

**15) Cross‑References**

* **UI/UX SOP:** docs/08\_ui\_ux.docx (Figma Make → ThemeBuilder → Fluent v12; component map; a11y).
* **FR/NFR:** docs/05\_fr.docx, docs/06\_nfr.docx (SSE budgets, read‑only UI).
* **Monitoring:** docs/11\_monitoring.docx (SLIs/SLOs, alerts).
* **CI/CD:** docs/10\_ci\_cd.docx (license injection; gates).
* **Evidence Pack:** docs/12\_evidence\_pack.docx (snapshots & retention).
* **Compliance:** docs/13\_compliance.docx (CSP/egress, secrets).
* **Error Catalog:** docs/error\_catalog.docx (codes & guidance).

**16) Assumptions**

1. Admin Portal is **same‑origin** with the API; bearer enforcement (if any) is handled at gateway.
2. Ingress supports **text/event-stream** pass‑through with read/idle timeouts that allow heartbeats.
3. **RTM** uses **Prod DB (read‑only)**; UI parity banners are informative only (no blocking from UI).

**17) Next Steps**

* Land SSE probe and heartbeat visualization; bind thresholds to **NFR/Monitoring** budgets.
* Add a CI **ingress conformance** check (assert no buffering for SSE).
* Bake evidence snapshot buttons and JSON download prompts into the Beta/Prod builds.

**Footer (optional for Word header/footer):**  
*MCPX‑KendoBridge • Admin Portal Ops Supplement • v2.0.0 • 2025‑09‑27 • Confidential — Technijian Internal*