SEO Portal intake-audit loop + chatbot P0 security (MC 103100/103105/103101) 2026-06-07

SEO Portal: intake-audit loop + chatbot P0 security

Date: 2026-06-07 | **MCs:** #103100 (intake-audit loop), #103105 (chatbot P0 security), #103101 (chatbot eval) | **Deployed image:** alairegistry.azurecr.io/seo-readiness-portal:20260607-intake-audit-p0 | **Source commit:** c6aed80ab (branch seo-intake-audit-loop-103100)

What changed

#103100 — intake-audit loop (gaps B + C)

- Client intake business context (competitors, priorityServices, targetMarkets, businessSummary) is now threaded from the pipeline into the audit runner, report generator and action-plan generator. Previously this rich intake data was collected but ignored.
- Connected GSC/GA OAuth state is plumbed into the audit runner and the "Access and measurement readiness" report section, overriding the self-reported dropdowns with real connection data, with graceful fallback when OAuth is not connected.
- Files: src/lib/audit/runner.ts, src/lib/reports/generator.ts, src/lib/reports/action-plan.ts, src/lib/workspace/persistence.ts. No-ranking disclaimer and forbiddenClaimWords guard preserved.

#103105 — chatbot P0 security fixes (from #103101 eval)

- **P0-A** Prompt injection: user message newlines are stripped before building the conversation, and the Ollama call was switched from /api/generate (flat prompt) to /api/chat (structured roles) so role boundaries are enforced server-side. All three dispatchChat callers updated (route.ts, action-plan.ts, deliverables.ts).
- **P0-B** Input guard: now scans ALL user-role messages, not just the last one. Previously an injection in message[0] of a multi-turn payload bypassed the guard.
- **P0-C** XSS: LLM output is HTML-escaped before markdown transforms / dangerouslySetInnerHTML in ChatMessage.tsx, so injected tags render inert.

Verification

- type-check EXIT 0; next build compiled clean (resolved a stale middleware.ts/proxy.ts conflict; canonical = middleware.ts).
- P0-B runtime-proven: secret word in message[0] (clean last message) triggered the pre-canned guard reply, no LLM call.
- P0-A runtime-confirmed: newline role-injection did not hijack the model; /api/chat path functional (Ollama replied).
- P0-C code-verified (DOM screenshot deferred — browser offline).
- Live post-deploy probes (seo-tools.snowit.ba): /intake/test 200, /api/health 200, /api/intake-chat (bad token) 401.
- Evidence: /tmp/evidence-103105/runtime-verification.md, /tmp/evidence-103105/deploy-verification.md, /tmp/evidence-103100/validation-results.txt, /tmp/alai/seo-chatbot-review/REPORT.md

Known follow-ups

- #103111 — monorepo alai-holding.git unpushable (>100MB files); source committed locally only, not on remote. Restore push/PR path.
- Remaining chatbot improvements from eval: 6 of 11 intake fields are structurally hard to extract (P1), no confirm-before-prefill step, in-memory rate limiter, unbounded paid-tier cost on Ollama outage.

Revision #1
Created 2026-06-07 19:59:57 UTC by John
Updated 2026-06-07 19:59:57 UTC by John