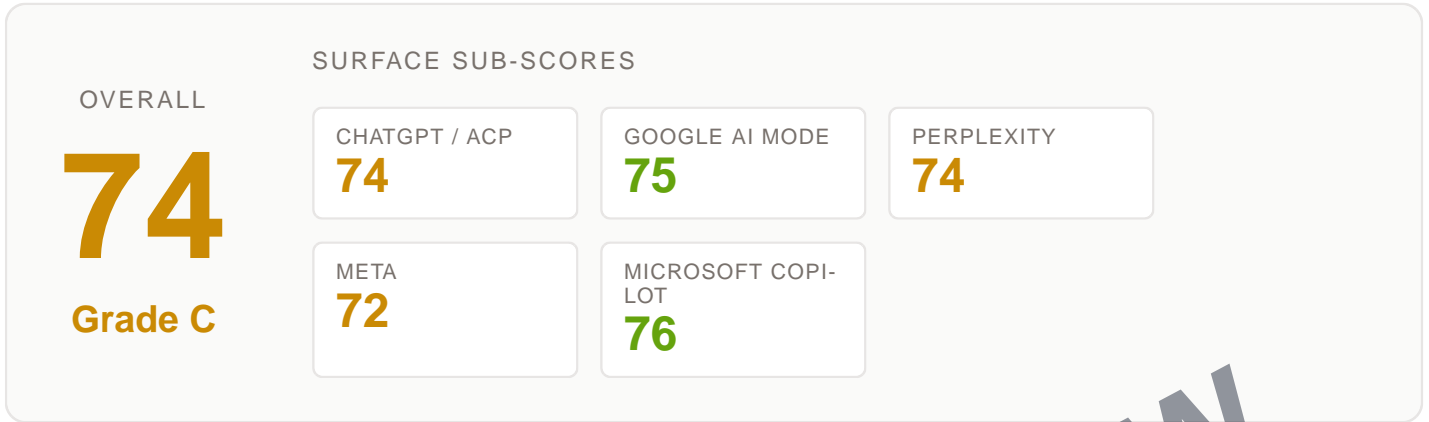


AI AGENT-READINESS REPORT

# vivo.co.nz

Ran 38 of 81 checks



## Findings - 12 need attention

**NEEDS WORK** **CRITICAL** **HTTPS enforced sitewide + HSTS (>= 6-month max-age)**

Enforce HTTPS sitewide and ship a Strict-Transport-Security header with max-age >= 6 months

**WHAT WE FOUND**

Confirmed the homepage is HTTPS (status 200), probed <http://vivo.co.nz/> for redirect behaviour, and parsed the Strict-Transport-Security header (value: "max-age=300").

How we checked: URL scheme + homepage status check, an <http://host/> redirect probe, and a Strict-Transport-Security max-age parse (>= 180-day threshold).

- HSTS max-age is below the 6-month minimum

Checked: [vivo.co.nz](http://vivo.co.nz)

**FAILED** **HIGH** **Product pages discoverable without JavaScript**

Make product pages discoverable without JavaScript

**WHAT WE FOUND**

Counted product pages discovered by the non-JavaScript crawl. None were found — JS-only storefront, products missing from sitemap, or the crawl was blocked.

How we checked: Count the product pages a non-JavaScript crawl could discover via the sitemap or initial HTML (no JS execution). The fetcher already attempted discovery; we read `ctx.pdpSample`.

- No product pages discoverable from a non-JavaScript crawl

Checked: [vivo.co.nz/sitemap.xml](http://vivo.co.nz/sitemap.xml), [vivo.co.nz](http://vivo.co.nz)



**FAILED HIGH Products are machine-discoverable**

Publish a product feed or a crawlable product sitemap

**WHAT WE FOUND**

Ran the discovery cascade (feed -> platform catalog -> typed sitemap -> content-verified crawl). Method: `none`; verified 0 product pages.

How we checked: Read the product-discovery cascade result from `ctx.discovery`. Score by discovery method (feed / platform\_api / sitemap\_typed -> pass when `verifiedProductCount >= MIN_CONFIDENT_PRODUCTS`; `content_verified -> partial`; `none` or `under-threshold -> fail`).

- No reliable way for agents to discover your products

Checked: `vivo.co.nz`

**FAILED HIGH UCP profile carries all four required top-level keys**

Add every required top-level key to the UCP profile

**WHAT WE FOUND**

Profile is missing required key(s): `signing_keys`.

How we checked: Read the profile root (or top-level `ucp` wrapper) and verify the presence of `version`, `services`, `capabilities`, and `signing_keys` keys.`

- Required top-level key `signing_keys` is missing`

Checked: `vivo.co.nz/.well-known/ucp`

**FAILED HIGH Each service satisfies the transport conditional field requirements**

Populate the conditional fields required by each service's transport

**WHAT WE FOUND**

Validated 1 services with recognised transports (0 satisfy their transport's required fields).

How we checked: For each `services[]` entry with a recognised transport, require the transport-conditional fields: `rest/mcp -> endpoint+schema`; `a2a -> endpoint`; `embedded -> schema`.

- Service is missing transport-conditional field(s)

Checked: `vivo.co.nz/.well-known/ucp`

**FAILED MEDIUM HSTS policy carries the includeSubDomains directive**

Add `includeSubDomains` to your Strict-Transport-Security header`

**WHAT WE FOUND**

Inspected the homepage Strict-Transport-Security header ("max-age=300") and the `includeSubDomains` directive is absent.`

How we checked: Parse the homepage `Strict-Transport-Security` header for the `includeSubDomains` directive (§6.1.2).`

- HSTS header is missing the `includeSubDomains` directive`

Checked: `vivo.co.nz`

**FAILED MEDIUM Organization/OnlineStore JSON-LD with contactPoint on homepage**

Add an Organization (or OnlineStore) JSON-LD block to your homepage with a `contactPoint`

**WHAT WE FOUND**

Parsed the homepage JSON-LD looking for an Organization/OnlineStore node with a `contactPoint`, but no Organization-class node is present.

How we checked: Parse homepage `<script type="application/ld+json">` blocks, flatten `@graph` , and look for an Organization/OnlineStore/Store node with a contactPoint carrying `email` or `telephone`.`

- No Organization/OnlineStore JSON-LD on homepage

Checked: `vivo.co.nz`

**NEEDS WORK** **MEDIUM** Sitemap declared in robots.txt

Add a `Sitemap:` line to robots.txt

**WHAT WE FOUND**

Read robots.txt and looked for a Sitemap: directive; none were declared.

How we checked: Read parsed Sitemap: directives from robots.txt (sitemaps.org / implementation note).

- robots.txt has no Sitemap: directive

Checked: vivo.co.nz/robots.txt

**FAILED** **MEDIUM** Sitemap resolvable and includes at least one product URL

Publish a sitemap containing product URLs

**WHAT WE FOUND**

Tried to resolve an XML sitemap from robots.txt (Sitemap: directives) or /sitemap.xml. No entries were returned.

How we checked: Parse ``<loc>`` entries from the resolved sitemap (or sitemap index) and classify each against product-URL patterns (`/products/...`, `/product/...`, `/p/<id>`, etc.).

- No XML sitemap was reachable, or it contained no ``<loc>` entries

Checked: vivo.co.nz/sitemap.xml

**FAILED** **LOW** HSTS policy carries the preload directive

Add `preload` to your Strict-Transport-Security header and submit to hstspreload.org

**WHAT WE FOUND**

Inspected the homepage Strict-Transport-Security header ("`max-age=300`") and the preload directive is absent.

How we checked: Parse the homepage `Strict-Transport-Security` header for the `preload` directive (hstspreload.org vendor extension to).

- HSTS header is missing the preload directive

Checked: vivo.co.nz

**FAILED** **INFO** Apple Pay markers detected (informational)

Enable Apple Pay through your payment processor (informational only)

**WHAT WE FOUND**

Scanned the homepage and 0 sampled PDPs for Apple Pay markers; none matched.

How we checked: Substring match on known Apple Pay SDK/markup signatures (ApplePaySession, apple-pay-button, /apple-developer-merchantid-domain-association) across the homepage and every sampled PDP HTML.

- No Apple Pay markers detected on the homepage or PDPs

Checked: vivo.co.nz

**FAILED** **INFO** Google Pay markers detected (informational)

Enable Google Pay through your payment processor (informational only)

**WHAT WE FOUND**

Scanned the homepage and 0 sampled PDPs for Google Pay markers; none matched.

How we checked: Substring match on known Google Pay SDK/markup signatures (pay.google.com/gp/p/js/pay.js, google.payments.api, <google-pay-button) across the homepage and every sampled PDP HTML.

- No Google Pay markers detected on the homepage or PDPs

Checked: vivo.co.nz

**PASSING** **CRITICAL** Googlebot allowed on product paths

Allow Googlebot on product paths

**WHAT WE FOUND**

Checked robots.txt for Googlebot at the representative product path `/`. The active group resolves to Allow.

How we checked: group match on `User-agent: Googlebot` rules at the representative product path `/` (via the parsed robots.txt isAllowed predicate).

Checked: vivo.co.nz/robots.txt, /



**NOT APPLICABLE** **CRITICAL** Offer price + priceCurrency valid

Set price as a number and priceCurrency as an ISO 4217 code

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**PASSING** **CRITICAL** OAI-SearchBot allowed

Allow OAI-SearchBot in robots.txt

**WHAT WE FOUND**

Checked robots.txt rules for OAI-SearchBot (OpenAI's ChatGPT search/discovery crawler) at path /. The active group resolves to Allow.

How we checked: group match on `User-agent: OAI-SearchBot` rules at path `/` (via the parsed robots.txt isAllowed predicate).

Checked: vivo.co.nz/robots.txt

**PASSING** **CRITICAL** No global wildcard root disallow

Remove the wildcard `Disallow: /` from robots.txt

**WHAT WE FOUND**

Scanned the robots.txt wildcard `User-agent: \*` group for a root Disallow (wildcard group present; root Disallow absent).

How we checked: Line-by-line scan of robots.txt; track membership of the `User-agent: \*` group (stacked UA lines combine into one group per section 2.2.1) and flag the file when a root `Disallow: /` appears in that group with no offsetting `Allow: /`.

Checked: vivo.co.nz/robots.txt

**PASSING** **HIGH** Bingbot allowed

Allow Bingbot in robots.txt

**WHAT WE FOUND**

Checked robots.txt rules for Bingbot (Microsoft's web crawler — also the source for Copilot Shopping's index) at path /. The active group resolves to Allow.

How we checked: group match on `User-agent: Bingbot` rules at path `/` (via the parsed robots.txt isAllowed predicate).

Checked: vivo.co.nz/robots.txt

**PASSING** **HIGH** Contact page exposes email or phone

Add a `mailto:` email link or `tel:` phone link to your contact page

**WHAT WE FOUND**

Reached the contact page and found a contact method: mailto link (mailto:info@vivo.co.nz), tel link (tel:0800), email (info@vivo.co.nz), phone (1631493591).

How we checked: URL probe of contact paths; the first 2xx body is scanned for `mailto:` / `tel:` hrefs, plain emails (placeholder hosts excluded), and phone-shaped numbers.

Checked: vivo.co.nz/contact-us

**NOT APPLICABLE** **HIGH** MerchantReturnPolicy finite-window has positive merchantReturnDays

Add a positive `merchantReturnDays` to finite-window return policies

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **HIGH** MerchantReturnPolicy satisfies Option A (country+category) or B (returnLink)

Make every MerchantReturnPolicy node satisfy Option A or Option B

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a





**NOT APPLICABLE** **HIGH** MerchantReturnPolicy node present on Product or Offer

Emit `hasMerchantReturnPolicy` on Product or Offer JSON-LD

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **HIGH** Offer `availability` is a Schema.org URL

Use a canonical Schema.org availability IRI on every Offer

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **HIGH** Offer JSON-LD carries shippingDetails (OfferShippingDetails)

Emit shippingDetails (OfferShippingDetails) on Offer JSON-LD

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **HIGH** Sampled PDPs are not gated behind a login wall (401 / 403)

Open PDPs to anonymous fetches

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **HIGH** No sampled PDP returns a noindex directive

Remove the noindex directive from every PDP

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **HIGH** Each PDP carries at most one Product JSON-LD node

Emit a single Product JSON-LD node per PDP

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**PASSING** **HIGH** PerplexityBot allowed

Allow PerplexityBot in robots.txt

**WHAT WE FOUND**

Checked robots.txt rules for PerplexityBot (Perplexity's shopping index crawler) at path /. The active group resolves to Allow.

How we checked: group match on `User-agent: PerplexityBot` rules at path `/ (via the parsed robots.txt isAllowed predicate).

Checked: vivo.co.nz/robots.txt

**PASSING** **HIGH** **Privacy policy page reachable**

Publish a privacy policy page and link it from your site nav/footer

**WHAT WE FOUND**

Found a privacy policy page at <https://vivo.co.nz/privacy-policy> (18837 chars of stripped body text).

How we checked: Discover candidate URLs by scoring homepage nav/footer anchors for privacy/gdpr/cookie keywords, then append platform-conventional paths; probe each with politeFetch and pass on the first 2xx with  $\geq 200$  stripped-body chars.

Checked: `vivo.co.nz/privacy-policy`

**NOT APPLICABLE** **HIGH** **Brand attribution on PDPs**

Surface brand attribution on every PDP

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **HIGH** **GTIN coverage on PDPs**

Populate ``gtin`` on every branded Product node

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **HIGH** **Product ``image`` populated**

Add a resolvable image URL to every Product node

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **HIGH** **Product JSON-LD present on PDPs**

Publish a Product JSON-LD block on every PDP

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **HIGH** **Product ``name`` populated**

Populate ``name`` on every Product JSON-LD node

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **HIGH** **Product JSON-LD includes ``offers``**

Add an ``offers`` object to every Product node

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**PASSING** **HIGH** robots.txt present at root

Publish a non-empty robots.txt at the site root

**WHAT WE FOUND**

Reached a non-empty /robots.txt at the site root (2129 bytes, 0 declared Sitemap lines).

How we checked: Check whether the fetcher reached a non-empty /robots.txt at the site root (§2.2.3 access method).

Checked: vivo.co.nz/robots.txt

**PASSING** **HIGH** Terms of service page reachable

Publish a terms of service page and link it from your site nav/footer

**WHAT WE FOUND**

Found a terms-of-service page at https://vivo.co.nz/terms-and-conditions (20531 chars of stripped body text).

How we checked: Discover candidate URLs by scoring homepage nav/footer anchors for terms/tos/legal/conditions keywords, then append platform-conventional paths; probe each with politeFetch and pass on the first 2xx with >=200 stripped-body chars.

Checked: vivo.co.nz/terms-and-conditions

**PASSING** **HIGH** UCP profile Cache-Control is shared-cacheable with max-age >= 60s

Serve /.well-known/ucp` with `Cache-Control: public, max-age=...`

**WHAT WE FOUND**

UCP profile Cache-Control is "public, max-age=300" — public with max-age=300s.

How we checked: Parse the `Cache-Control` header on the `/.well-known/ucp` response; require `public`, `max-age >= 60`, and no `no-store` / `no-cache` / `private`.

Checked: vivo.co.nz/.well-known/ucp

**PASSING** **HIGH** /.well-known/ucp response Content-Type is application/json

Serve /.well-known/ucp with `Content-Type: application/json`

**WHAT WE FOUND**

UCP profile Content-Type is "application/json; charset=utf-8".

How we checked: Check that the `Content-Type` header on /.well-known/ucp starts with `application/json` (optionally with a charset parameter).

Checked: vivo.co.nz/.well-known/ucp

**PASSING** **HIGH** /.well-known/ucp is publicly fetchable with no auth

Allow unauthenticated access to /.well-known/ucp

**WHAT WE FOUND**

Unauthenticated GET /.well-known/ucp returned 200.

How we checked: Confirm an unauthenticated GET to /.well-known/ucp returns a 2xx status.

Checked: vivo.co.nz/.well-known/ucp

**PASSING** **HIGH** /.well-known/ucp returns 200 directly with no redirects

Serve /.well-known/ucp directly with a 200 response

**WHAT WE FOUND**

GET /.well-known/ucp returned 200 directly with no redirect chain.

How we checked: Inspect the final HTTP status of GET /.well-known/ucp and whether any 3xx redirect was followed to reach it.

Checked: vivo.co.nz/.well-known/ucp

**PASSING** **HIGH** **/.well-known/ucp profile is present with a `version` field**

Publish `/.well-known/ucp` with at minimum a `version` field

**WHAT WE FOUND**

Found `/.well-known/ucp` with version "2026-04-08".

How we checked: Confirm `ctx.wellKnownUcp` is non-null and carries a non-empty `version` string (the only universally-required UCP profile field).

Checked: `vivo.co.nz/.well-known/ucp`

**PASSING** **HIGH** **Each service `transport` is rest, mcp, a2a, or embedded**

Set `transport` to one of rest, mcp, a2a, or embedded

**WHAT WE FOUND**

Validated 1 service `transport` (1 match the UCP enum).

How we checked: For each `services[]` entry, require `transport` to be one of: rest, mcp, a2a, embedded.

Checked: `vivo.co.nz/.well-known/ucp`

**PASSING** **HIGH** **UCP profile declares a valid shopping service entry**

Declare a shopping service entry with a recognised transport and an HTTPS endpoint

**WHAT WE FOUND**

Found 1 valid shopping service entry (transport + HTTPS endpoint).

How we checked: List every `services[]` entry whose namespace is `shopping` (or contains `shopping`) and require at least one with `transport` {rest,mcp,a2a,embedded} AND a syntactically valid `https://` endpoint.

Checked: `vivo.co.nz/.well-known/ucp`

**NOT APPLICABLE** **HIGH** **Every signing\_keys[] entry is a valid JWK**

Make every `signing_keys[]` entry a JWK with `ktypes` + `ktyspecificparams`

**WHAT WE FOUND**

Profile declares no `signing_keys`; JWK validation has no entries to evaluate.

How we checked: Walk `signing_keys[]` and validate each entry per §4.1 (kty required) + §6 (kty-specific required parameters). `kid` is OPTIONAL per §4.5 and not enforced here.

Checked: `vivo.co.nz/.well-known/ucp`

**NOT APPLICABLE** **MEDIUM** **MerchantReturnPolicy merchantReturnLink URL is reachable**

Repair every `merchantReturnLink` URL

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **MEDIUM** **MerchantReturnPolicy applicableCountry uses ISO 3166-1 alpha-2 codes**

Use ISO 3166-1 alpha-2 country codes in `applicableCountry`

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **MEDIUM** **MerchantReturnPolicy returnPolicyCategory uses valid Schema.org enum**

Use a valid Schema.org `returnPolicyCategory` enum value

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **MEDIUM** OfferShippingDetails shippingDestination is a valid DefinedRegion

Emit shippingDestination as a DefinedRegion with ISO addressCountry

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **MEDIUM** OfferShippingDetails shippingRate is a valid MonetaryAmount

Emit shippingRate as a valid MonetaryAmount

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **MEDIUM** Product `brand` is a string or Brand/Organization object

Emit `brand` as either a string or a typed Brand object on every Product

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **MEDIUM** Product `description` present

Populate `description` on every Product JSON-LD node

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **MEDIUM** Product `sku` populated

Populate `sku` on every Product JSON-LD node

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **MEDIUM** Product title not a placeholder

Replace placeholder and slug-shape titles with real product names

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**PASSING** **MEDIUM** Returns/refund policy page reachable

Publish a returns policy page and link it from your site nav/footer

**WHAT WE FOUND**

Found a returns/refund policy page at <https://vivo.co.nz/return-policy> (9844 chars of stripped body text).

How we checked: Discover candidate URLs by scoring homepage nav/footer anchors for return/refund/exchange keywords, then append platform-conventional paths; probe each with politeFetch and pass on the first 2xx with  $\geq 200$  stripped-body chars.

Checked: vivo.co.nz/return-policy

**PASSING** **MEDIUM** **Third-party review-platform integration detected**

Install a third-party review platform so agents see syndicated reviews on your storefront

**WHAT WE FOUND**

Found a third-party review-platform integration (reviews.io, bazaarvoice) on 1 of 1 scanned source.

How we checked: Substring scan of homepage and sampled PDP HTML for known review-platform asset fingerprints (judge.me, yotpo, stamped.io, reviews.io, okendo, loox, trustpilot, bazaarvoice).

Checked: vivo.co.nz

**PASSING** **MEDIUM** **Shipping policy page reachable**

Publish a shipping policy page and link it from your site nav/footer

**WHAT WE FOUND**

Found a shipping policy page at <https://vivo.co.nz/shipping> (8512 chars of stripped body text).

How we checked: Discover candidate URLs by scoring homepage nav/footer anchors for shipping/delivery/dispatch keywords, then append platform-conventional paths; probe each with politeFetch and pass on the first 2xx with  $\geq 200$  stripped-body chars.

Checked: vivo.co.nz/shipping

**NOT APPLICABLE** **MEDIUM** **Sitemap <loc> entries are entity-escaped**

Entity-escape `&`, `<`, `>` in every <loc>

**WHAT WE FOUND**

Wanted to inspect sampled <loc> entries for entity escaping, but the runner did not surface any sitemap resources.

How we checked: Sample the first 100 <loc> entries per sitemap document and check for raw &, <, or > (sitemaps.org entity escaping rules).

- Transport metadata not available — runner update pending

Checked: vivo.co.nz/sitemap.xml

**NOT APPLICABLE** **MEDIUM** **Sitemap entries share the host of the containing sitemap**

Keep every sitemap entry on the sitemap's own host

**WHAT WE FOUND**

Wanted to compare sitemap entry hosts against the containing sitemap, but the runner did not surface any sitemap resources.

How we checked: For each resolved sitemap resource, parse the sitemap URL's host and compare it against every parsed <loc> URL's host.

- Transport metadata not available — runner update pending

Checked: vivo.co.nz/sitemap.xml

**NOT APPLICABLE** **MEDIUM** **Sitemap root declares the sitemaps.org 0.9 namespace**

Add the sitemaps.org 0.9 xmlns to the root element

**WHAT WE FOUND**

Wanted to check the xmlns declaration on every resolved sitemap document, but the runner did not surface any sitemap resources.

How we checked: Substring-match `xmlns="http://www.sitemaps.org/schemas/sitemap/0.9"` against the raw XML of every resolved sitemap document.

- Transport metadata not available — runner update pending

Checked: vivo.co.nz/sitemap.xml

**NOT APPLICABLE** **MEDIUM** **Each capability has version + spec + schema**

Populate version, spec, and schema on every capabilities[] entry

**WHAT WE FOUND**

Profile declares no capabilities; required-field checks have nothing to evaluate.

How we checked: For each capabilities[] entry, require non-empty string values for `version`, `spec`, and `schema`.

Checked: vivo.co.nz/.well-known/ucp

**NOT APPLICABLE** **MEDIUM** Each service's `spec` URL origin matches its namespace authority

Point each service `spec` URL at the canonical UCP authority

**WHAT WE FOUND**

No services declared a `spec` URL; origin matching has nothing to evaluate.

How we checked: For each service with a `spec` URL, require the URL origin to be a canonical UCP authority OR the host/path to include the namespace token.

Checked: vivo.co.nz/.well-known/ucp

**PASSING** **MEDIUM** Every service `version` matches YYYY-MM-DD

Use ISO-date `version` strings on every service

**WHAT WE FOUND**

Validated 1 service version string (1 match YYYY-MM-DD).

How we checked: For each services[] entry, require `version` to be a string matching `/^d{4}-d{2}-d{2}$/`.

Checked: vivo.co.nz/.well-known/ucp

**PASSING** **LOW** About page reachable with substantive copy

Publish a substantive About page at a standard URL

**WHAT WE FOUND**

Found a substantive About page at <https://vivo.co.nz/about-us/> (44446 chars of stripped body text, threshold 200).

How we checked: URL probe of platform-specific about-page paths via polite Fetch; the first 2xx response whose HTML-stripped body length is  $\geq 200$  chars counts as a pass.

Checked: vivo.co.nz/about-us

**NOT APPLICABLE** **LOW** BreadcrumbList present on PDPs

Add a BreadcrumbList JSON-LD block to every PDP

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**PASSING** **LOW** ChatGPT-User allowed

Allow ChatGPT-User in robots.txt (advisory)

**WHAT WE FOUND**

Checked robots.txt rules for ChatGPT-User (OpenAI's user-initiated live fetcher (advisory)) at path /. The active group resolves to Allow.

How we checked: group match on `User-agent: ChatGPT-User` rules at path ` /` (via the parsed robots.txt isAllowed predicate).

Checked: vivo.co.nz/robots.txt

**NOT APPLICABLE** **LOW** Alt text on at least 80% of PDP images

Add descriptive alt text to product images (WCAG 2.x SC 1.1.1)

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **LOW** Product images meet Google's 50,000-pixel area threshold

Upload higher-resolution product images (area  $\geq 50,000$  pixels)

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **LOW** **MerchantReturnPolicy enrichment enums use valid Schema.org values**

Use Schema.org enum values for returnFees / returnMethod / refundType

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **LOW** **Offer `itemCondition` is canonical when present**

Either omit `itemCondition` (defaults to NewCondition) or set it to a canonical IRI

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **LOW** **OfferShippingDetails deliveryTime is a valid ShippingDeliveryTime**

Emit a ShippingDeliveryTime with handlingTime and/or transitTime populated

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**PASSING** **LOW** **Perplexity-User allowed**

Allow Perplexity-User in robots.txt (advisory)

**WHAT WE FOUND**

Checked robots.txt rules for Perplexity-User (Perplexity's live user initiated fetcher (advisory)) at path /. The active group resolves to Allow.

How we checked: group match on `User-agent: Perplexity-User` rules at path / (via the parsed robots.txt isAllowed predicate).

Checked: vivo.co.nz/robots.txt

**NOT APPLICABLE** **LOW** **Product `aggregateRating` present**

Add an AggregateRating to Product nodes when you have real reviews

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**NOT APPLICABLE** **LOW** **Product title quality (present, not all-caps)**

Use sentence-case product titles

**WHAT WE FOUND**

Couldn't confidently identify product pages (found 0), so product-level checks aren't applicable.

How we checked: n/a

**PASSING** **LOW** **/robots.txt is served as text/plain**

Send Content-Type: text/plain on /robots.txt

**WHAT WE FOUND**

Inspected /robots.txt Content-Type ("text/plain; charset=UTF-8"); recognised as text/plain.

How we checked: Inspect the /robots.txt response `Content-Type` header for a `text/plain` media type per §2.3.

Checked: vivo.co.nz/robots.txt



FREE PREVIEW

**PASSING** **LOW** /robots.txt is under 500 KiB (RFC 9309 §2.5 parser cap)

Trim /robots.txt below 500 KiB

**WHAT WE FOUND**

Measured /robots.txt at 2,130 bytes; within the 512,000-byte (500 KiB) cap.

How we checked: Measure the raw byte size of the /robots.txt body and compare against the §2.5 parser cap ( $\geq 500$  KiB).

Checked: vivo.co.nz/robots.txt

**PASSING** **LOW** /robots.txt is served as UTF-8

Serve /robots.txt as UTF-8

**WHAT WE FOUND**

Inspected the /robots.txt byte stream (2130 bytes); decodes as UTF-8.

How we checked: Inspect the raw byte stream of /robots.txt for UTF-8 decodability per §2.3.

Checked: vivo.co.nz/robots.txt

**NOT APPLICABLE** **LOW** Every sitemap <loc> URL is under 2048 characters

Keep every <loc> URL under 2,048 characters

**WHAT WE FOUND**

Wanted to check <loc> URL lengths across every resolved sitemap document, but the runner did not surface any sitemap resources.

How we checked: Iterate every parsed <loc> URL across all resolved sitemap resources and check length against the 2,048-character cap.

- Transport metadata not available — runner update pending

Checked: vivo.co.nz/sitemap.xml

**NOT APPLICABLE** **LOW** Sitemap respects 50 MiB / 50,000-URL caps per document

Split over-cap sitemaps into a sitemap index

**WHAT WE FOUND**

Wanted to verify each sitemap's byte size and entry count against sitemaps.org caps, but the runner did not surface any sitemap resources.

How we checked: Check raw byte size ( $\leq 52,428,800$  B) and entry count ( $\leq 50,000$ ) for every resolved sitemap resource.

- Transport metadata not available — runner update pending

Checked: vivo.co.nz/sitemap.xml

**NOT APPLICABLE** **LOW** Sitemap is served as UTF-8

Serve every sitemap document as UTF-8

**WHAT WE FOUND**

Wanted to check the encoding of every resolved sitemap document, but the runner did not surface any sitemap resources with transport metadata.

How we checked: Inspect every resolved sitemap document's raw byte stream for UTF-8 decodability (sitemaps.org encoding requirement).

- Transport metadata not available — runner update pending

Checked: vivo.co.nz/sitemap.xml

**NOT APPLICABLE** **LOW** UCP MCP-transport entries have valid HTTPS endpoints

Make every declared MCP transport endpoint an absolute HTTPS URL

**WHAT WE FOUND**

Walked services[] for `transport: "mcp"` entries; none advertised.

How we checked: Filter services[] to entries where transport=mcp and validate that `endpoint` is an absolute https:// URL.

Checked: vivo.co.nz/.well-known/ucp

**NOT APPLICABLE** **INFO** **llms.txt present (informational)**

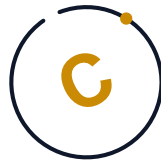
Publish an /llms.txt manifest (optional)

**WHAT WE FOUND**

Looked for /llms.txt at the site root; the fetcher returned no file.

How we checked: Check whether the fetcher reached an /llms.txt at the site root. Informational only — no failure path per llmstxt.org being a voluntary community convention.

Checked: vivo.co.nz/llms.txt



**FREE PREVIEW**