

// MANUAL

SerpControl & SerpControl Ultra

**Eight on-page SEO tools.
In your browser tab.**

A field guide to every tool in the SerpControl extension. What each tool does, how to use it, and how to read the results.

// CONTENTS

- 01 Install and open
- 02 How tools run
- 03 Meta Inspector
- 04 Schema Checker
- 05 Headings
- 06 Link Map
- 07 Word Cloud
- 08 Page Audit
- 09 AI Bot Shield
- 10 LLM Visibility
- 11 Plan, auth, support

// INSTALL AND OPEN

Install and open

SerpControl is a Chrome, Firefox, and Safari extension. Pin it once, then run any of the eight tools with two clicks on the active tab. Free tools send nothing to a server. They read the live DOM and render an overlay.

// STEPS

- Install from the Chrome Web Store, Firefox Add-ons, or Safari Extensions Gallery.
- Pin the SerpControl icon to your toolbar. One click away.
- Open any page you want to audit. Click the icon and the tool palette opens.
- Pick a tool. The overlay renders directly on the page. Press **Esc** or **CLOSE** to dismiss.

// KEYBOARD

```
alt + shift + s → open the SerpControl popup
click + click   → pick a tool
esc / CLOSE     → remove the overlay
```

// HOW TOOLS RUN

How tools run

Each tool injects a single script into the active tab. The script reads the current document, computes its result locally, and renders an overlay anchored at z-index 2147483647. No data leaves the browser for free tools.

Ultra tools work the same way. The only difference is an entitlement check against app.serpctrl.com before the script is injected. Once verified, your entitlement is cached for an hour, so repeat runs are instant.

// TOOL_01

FREE

Meta Inspector

Reads every metadata signal a search engine looks at on the current page and shows it in one panel. The first thing to run on any URL. Before you open DevTools. Before you run a crawler. If the meta is wrong, nothing else matters.

// WHAT IT EXTRACTS

<code>title</code>	Contents of <title>. Warning if longer than 60 characters.
<code>description</code>	Contents of meta name="description". Warning if missing or over 160.
<code>canonical</code>	link rel="canonical". Flags a self-canonical mismatch.
<code>hreflang</code>	All link rel="alternate" hreflang entries. Language and href.
<code>robots</code>	Page-level meta robots and googlebot directives. noindex and nofollow show up in red.
<code>og: + tw:</code>	Full Open Graph and Twitter Card stack. Title, description, image, url, type.
<code>robots.txt</code>	Fetches /robots.txt and checks whether the current URL is allowed for Googlebot. Cached per host for 1h.

// SIGNALS YOU'LL SEE

- **Self-canonical OK.** The canonical URL matches the current URL. Green pill.
- **Cross-canonical.** Canonical points elsewhere. Yellow pill, with the target shown.
- **OG and canonical mismatch.** og:url doesn't match canonical. Red pill.
- **noindex and nofollow.** Page-level directives surface as red badges.
- **Length warnings.** Title or description over the recommended length.

// HOW TO USE IT

- Open any page and click **Meta Inspector**.
- The overlay slides in from the right. Scroll through each section.
- Hover any value to see the full text. Long titles get truncated.
- Click **CLOSE** or press **Esc** to dismiss.

// WHEN TO RUN IT

Every audit. Every new template. Every time you change next-seo config. Two seconds, and it catches 80% of indexing problems.

// TOOL_02

FREE

Schema Checker

Finds every JSON-LD block on the page, parses it, and validates it against schema.org rules. Both the universal ones and a type-specific rule set. Each issue ships with a one-line fix you can paste into your code.

// WHAT GETS CHECKED

- **Required keys.** @context and @type must be present and valid.
- **Type casing.** @type must be TitleCase (e.g. Product, not product).
- **Empty values.** null or empty-string properties are flagged for removal.
- **ISO formats.** Dates and durations validated against ISO 8601.
- **HTML in strings.** Raw HTML inside JSON-LD properties is flagged. Google strips it anyway.
- **Relative URLs.** Bare paths or relative URLs in URL fields get caught.
- **Bare filenames.** logo.png instead of a full URL gets a fix recommendation.
- **Type-specific rules.** Article, Product, Recipe, Event, FAQPage, BreadcrumbList, LocalBusiness, Organization, Person, VideoObject and more each get bespoke checks.

// RENDER SOURCE

Each block is tagged **SSR** or **CSR** based on whether the script existed in the initial HTML or was injected by JavaScript at runtime. CSR-injected JSON-LD is invisible to most AI crawlers. GPTBot, ClaudeBot, PerplexityBot, CCBot. Move it into the SSR document head and the problem goes away.

// SEVERITY LEVELS

Critical	Required field missing or wrong type. Google will reject the rich result entirely.
Warning	Field is present but suboptimal. Empty value, wrong format, or relative URL.

// HOW TO USE IT

- Run on any page that has structured data. Each JSON-LD block is listed individually.
- Scan the issue list per block. Each one shows three rows: Property, Message, Fix.
- Click the **Raw JSON** tab to inspect the full parsed tree.

// TOOL_03

FREE

Headings

Every H1 through H6 on the page in one panel. Indented by level and color-coded. One glance tells you the hierarchy and shows you the page the way a screen reader or a search engine sees it.

// WHAT YOU'LL SEE

<code>count</code>	Total heading count at the top of the panel.
<code>colors</code>	H1 green, H2 white, H3 and below get progressively darker grey.
<code>indent</code>	Each level indents 12px deeper. The whole panel reads as a tree.
<code>(empty)</code>	Headings with no text are surfaced explicitly. Easy to miss in DevTools.

// CLICK BEHAVIOUR

- Click any heading in the panel. The panel closes.
- The page scrolls smoothly to that heading and highlights it for ~2 seconds with a color-matched outline.
- Run the tool again to reopen the panel. Clicking is one-shot navigation.

// WHAT TO LOOK FOR

- **Multiple H1s.** Flat hierarchy. Pick one primary H1, demote the rest.
- **Skipped levels.** H2 to H4 with no H3. Bad for accessibility, mediocre for SEO.
- **Empty headings.** Usually a CSS-as-decoration mistake. Replace with a div.
- **Keyword-stuffed H1.** A short scan tells you instantly.

// TOOL_04

FREE

Link Map

Every `<a href>` on the page, split into internal and external. Duplicates are merged with a count of how many times each URL appears. One click runs a broken-link pass, firing HEAD requests against every link.

// PANEL ANATOMY

<code>INTERNAL (n)</code>	Links whose hostname matches the current page. n is the number of unique URLs.
<code>EXTERNAL (n)</code>	Everything else. Outbound links, subdomains, third-party scripts you embedded.
<code>count badge</code>	How many anchors on the page point to this URL. ×3 means three anchors share the URL.
<code>text</code>	First 80 characters of the anchor text. Empty anchors fall back to the href.

// BROKEN LINK CHECK

- Click **CHECK LINKS** in the top bar.
- HEAD requests fire in parallel against every URL on the active tab.
- Each row gets a status pill. **OK**, **BROKEN**, or **CHECKING**.
- Same-origin checks are reliable. Cross-origin URLs depend on CORS and may report **CHECKING** indefinitely.

// SORT AND FILTER

- Sort by count (most-linked first), URL A to Z, or URL Z to A.
- Toggle internal and external tabs to focus on what matters for the audit.

// WHEN TO RUN IT

Internal-link audits, anchor-text reviews, and quick sanity checks before publishing. Pair with **Page Audit** to see nofollow usage in context.

// TOOL_05

FREE

Word Cloud

Top 100 words on the page by frequency, sized by share of body text. Stop-words are stripped, tokens shorter than three characters get dropped. A 30-second sanity check on what the page is actually about, versus what you think it's about.

// HOW IT TOKENIZES

- Reads document.body.innerText. Visible text only. Hidden divs are excluded.
- Lowercases everything, splits on Unicode letter boundaries (`\p{L}{3,}`).
- Removes a built-in English stop-word list (the, and, is, you, this, etc.).
- Counts the remaining tokens, sorts by frequency, takes the top 100.

// WHAT YOU'LL SEE

<code>size</code>	Bigger word means higher frequency. The biggest word maps to your highest count, the smallest to your lowest of the 100.
<code>colors</code>	Eight-color rotation for legibility. No semantic meaning. Just readability.
<code>counter</code>	Top of the panel: top N words for transparency.

// WHAT TO LOOK FOR

- Your **target keyword** in the top 5. If it's not, the page isn't about what you think.
- **Branded terms** dominating an informational page. Usually a sign of over-promotion.
- **Unintended dominance.** A CMS template injecting boilerplate that drowns out the body.

// TOOL_06

ULTRA

Page Audit

Nine-section on-page audit, scored 0 to 100. Critical, Warning, Info classification. Every issue ships with a one-line fix. The single tool that replaces five paid extensions during a manual on-page review.

// THE NINE SECTIONS

1. Meta	Title, description, canonical, robots, hreflang, OG, Twitter Card.
2. Headings	Hierarchy, H1 count, empty headings, level skips, keyword presence.
3. Content	Word count, text-to-code ratio, thin-content warnings, top keywords.
4. Images	Alt-text coverage, oversize files, missing dimensions, lazy-loading hints.
5. Links	Internal and external balance, nofollow patterns, anchor-text diversity.
6. Schema	JSON-LD validation, render source (SSR or CSR), type-specific rules.
7. robots.txt	Directive parsing, conflicts, allow and disallow rules for the current URL.
8. Performance	Document size, script count, CSS count. Quick proxies before Lighthouse.
9. Indexability	The final verdict. Is this page indexable, and if not, what's blocking it.

// HOW THE SCORE WORKS

Each section starts at 100 and loses points per issue. **Minus 10** for Critical, **minus 5** for Warning, **minus 1** for Info. The page score is the average across all nine sections, rounded to the nearest integer.

Score colours. **80 or higher is green**, publish-ready. **50 to 79 is amber**, fix the criticals first. **Under 50 is red**, rewrite the page rather than patch it.

// USING THE FIX LIST

- Each issue: **Check, Finding, Recommendation**. The recommendation is one sentence you can paste into a ticket or directly into your code.
- Filter by severity to skip the noise and ship the criticals first.
- Click any section header to collapse it. Useful when one section dominates.

// REQUIRES

An active SerpControl Ultra license. Sign in once at app.serpctrl.com. Your entitlement is cached for an hour, then re-verified silently in the background.

// TOOL_07

ULTRA

AI Bot Shield

Twenty AI-search, AI-training, and SEO crawlers checked against your robots.txt. Tells you which bots can read your content, which are training on it, and gives you a copy-paste robots.txt snippet that matches the recommended posture.

// BOT GROUPS

AI Search	Bots that fetch live pages to cite in AI answers. OAI-SearchBot , ChatGPT-User , PerplexityBot , Applebot . Keep them visible . These drive AI Overview citations.
AI Training	Bots that scrape pages for LLM training data. GPTBot , ClaudeBot , Google-Extended , Meta-ExternalAgent , Bytespider , CCBot , and others. Block unless you want your content used for training .
SEO	Crawlers from SEO tools. AhrefsBot , MJ12bot , DotBot , PetalBot , Yandex . Allow the ones whose data you actually use. Block the rest to save crawl budget.

// STATUS COLUMN

- **BLOCKED**. robots.txt explicitly disallows the bot at root.
- **RATE-LIMITED**. Crawl-delay is set, but the bot is not blocked.
- **UNPROTECTED**. The bot can crawl your entire site freely.

// RECOMMENDATION COLUMN

Each bot has a recommended posture. **block** for training crawlers, **delay:1 to 3** for AI search and useful SEO bots. The state badge tells you whether your current setup matches. **IDEAL** matches the recommendation, **ACCEPTABLE** is close, **WRONG** needs fixing.

// COPY-PASTE SNIPPET

Click **COPY ROBOTS.TXT**. The snippet sets the recommended posture for all 20 bots. Rate-limit AI search fetchers, block training crawlers, allow useful SEO bots with delays. Paste it into /robots.txt at your site root, deploy, then re-run the tool to verify.

```
User-agent: OAI-SearchBot
Crawl-delay: 1

User-agent: GPTBot
Crawl-delay: 3

User-agent: ClaudeBot
Disallow: /

User-agent: Google-Extended
Disallow: /
# 16 more rules in the full snippet
```

Your robots.txt is fetched once per host and cached for 1 hour. Click **RECHECK** to refetch.

// TOOL_08

ULTRA

LLM Visibility

Differs the page's raw HTML against its rendered DOM and tells you what an AI crawler that doesn't execute JavaScript actually sees. Most LLMs (**GPTBot**, **ClaudeBot**, **PerplexityBot**, **CCBot**) don't render JS. If your content lives in useEffect, they read a blank page.

// FIVE TABS

Checks	Pass and fail issues by severity (Critical, Warning, Info). Each one with Why and Fix notes.
Stack	Detected framework. Next.js, Nuxt, React SPA, Vue, Angular, Svelte, Gatsby, Astro, Remix, WordPress, Shopify, Webflow, Squarespace, Wix, Drupal.
Metadata	Title, description, canonical, viewport, robots, OG tags, hreflang count, JSON-LD block count. Raw versus rendered. A mismatch means JS is hydrating tags after first paint.
Content	Word count, H1 and H2 count, image alt coverage. Raw versus rendered. The JS-only words list shows content that exists only after hydration.
Raw	The exact HTML the bot received. Search it. If your headline isn't there, you have a client-side rendering problem.

// THE SCORE

0 to 100 score based on how much of the page is visible without JS. **80 or higher** is AI-bot-safe. **50 to 79** is partial. Your meta is server-rendered, but key content isn't. **Under 50** is client-only render. Bots see an empty shell.

// WHAT IT CATCHES

- **JSON-LD injected by JS.** Present in the rendered DOM, missing from raw HTML. AI crawlers won't parse it. Move it to SSR `<script type="application/ld+json">`.
- **Empty roots.** `#root`, `#__next`, or `#app` with no children in raw HTML. The page is fully client-rendered.
- **Paywall signatures.** Piano, Tinypass, Zephr, Pico, or `isAccessibleForFree:false`. Bots may be blocked even if humans aren't.
- **Cloaking risk.** Meta or content differs significantly between raw and rendered. Could be unintentional, but Google's anti-cloaking heuristics may flag it.
- **OG tag drift.** OG values present in the rendered DOM but not in raw HTML. Social cards and AI Overview citations won't render correctly.

// WHEN TO RUN IT

Any page on a JS framework. Any page with paywall logic. Any page that should be cited by ChatGPT, Perplexity, or Claude. If the score is below 80, fix the SSR before chasing rankings.

// PLAN, AUTH, SUPPORT

Plan, auth, support

The five free tools work without an account. The three Ultra tools cost €10 right now, paid once at app.serpctrl.com. Subscription pricing comes later.

// PRICING

Free	Meta Inspector, Schema Checker, Headings, Link Map, Word Cloud. No account.
Ultra	Adds Page Audit, AI Bot Shield, LLM Visibility. €10 flat, one-time . Subscription pricing comes later.
Comparison	Ahrefs Lite €1,450 a year. Moz Pro €1,200 a year. SerpControl Ultra €10, paid once.

// HOW AUTH WORKS

- Click an Ultra tool. The popup hands off to app.serpctrl.com with a short-lived code.
- If you have an active session there, you're recognised instantly and the tool runs.
- If you're signed out, you're routed to the app sign-in page, then back to the extension.
- Verified entitlement is cached for 1 hour. A background alarm refreshes it silently.
- Free tools never wait on auth. They run immediately.

// BROWSERS SUPPORTED

Chrome	Manifest V3. Chrome Web Store.
Firefox	Manifest V2. Firefox Add-ons.
Safari	Manifest V2. Safari Extensions Gallery.

// PRIVACY

Free tools run entirely in your browser tab. No network calls beyond the page itself. Plus a one-time fetch of `/robots.txt` for Meta Inspector and AI Bot Shield, cached for an hour. Ultra tools verify your subscription state with app.serpctrl.com. Nothing about the audited page is sent to our servers.

// SUPPORT

Email	support@serpctrl.lv
Web	serpctrl.com · serpctrl.lv
Publisher	SIA «Cyber Unicorn», reg. 40203002129, Jūrmala, Latvia

// END OF MANUAL

Other tools tell you what's broken. SerpControl tells you how to fix it.