SEO Kit — everything crawlers & AI read

@lacspace/seo

SEO Autopilot & CI gate

v1.6.1 · zero dependencies · generated 2026-08-25

Overview

SEO Autopilot: defineSite() once, then site.meta()/article()/product()/faq() auto-fill Metadata AND JSON-LD — title template, canonical, OpenGraph, Twitter, auto description and dynamic OG image. Plus 19 schema.org builders, a @graph composer, content helpers (excerpt, reading time), and a built-in auditor with a CI sitemap crawler — `npx @lacspace/seo crawl <site> --min-grade A` fails your build the moment SEO regresses. Stop writing SEO by hand.

> **NOTE**
> When to use — Any Next.js (or framework-agnostic) site that needs correct metadata, Open Graph, Twitter cards and schema.org JSON-LD without hand-writing fragile objects on every page.

Package: @lacspace/seo · v1.6.1

Kit: SEO Kit — everything crawlers & AI read

Runtime: zero dependencies

Install

```
npm i @lacspace/seo
```

Quick example

```
site.meta({ title: "Pricing", path: "/pricing" })
```

Returns: **full Metadata + OG + Twitter**

Key API

defineSite(config) The Autopilot engine — returns a `site` with meta()/article()/product()/faq()/…

seoMetadata(input) Framework-agnostic -> Next.js `Metadata`.

auditHtml(html) Grade a page's on-page SEO 0–100 with per-check results.

CLI: audit / crawl `npx @lacspace/seo audit <url>` · `crawl <site> --min-grade A`.

Tips & warnings

> **TIP**
> Call `defineSite()` once in `lib/site.ts`; then `site.meta({ title, path })` returns a full Next.js `Metadata` object with canonical, OG and Twitter already filled.

> **TIP**
> Use `site.article()`, `site.product()`, `site.faq()` etc. to get metadata AND a matching JSON-LD `@graph` in one call — great for rich results.

TIP
Wire the CI gate: `npx @lacspace/seo crawl https://yoursite.com --min-grade A` fails your build if any page's on-page SEO regresses.

TIP
Zero-dependency and isomorphic — safe to import on the server, in the browser, on the edge and in React Native. It tree-shakes, so you only ship what you import.

WARNING
Set `NEXT_PUBLIC_SITE_URL` in production, or canonical/OG URLs fall back to your placeholder domain.

WARNING
Don't hardcode a canonical in the root layout — every child page would inherit it. Let each page set its own `path`.

Links & full reference

- npm: https://www.npmjs.com/package/@lacspace/seo
- Source & full README: https://github.com/lacspace/npm-packages/tree/main/seo
- Docs: lacspace.com/docs/seo

The npm page and GitHub README carry the complete API reference and more examples.