Engineer Software

Ready for AI agents

Website and application reviews for crawler access, readable content, structured data and agent interfaces, with scoped implementation.

Make the information on your website easier for search engines and AI tools to access and interpret. We review crawl access, page content, structured data and the interfaces your application exposes, then implement the agreed changes. This site provides an example you can inspect. Inclusion in search results or an AI answer remains the provider's decision.

Turn the service into a result

Make "Ready for AI agents" serve a business outcome. We can connect the technical work to the users, operations and decisions it needs to support. A site or an app that the AI assistants your customers use can find, read and act on, checked on every release rather than promised once.

Explore our services Get in touch →

Examine the relevant work

Use the related work on this page to examine our approach. Ask about the tradeoffs and the conditions behind each result. Built a multilingual Hugo site in 3 languages — 108 template files serving three languages and four formats

Built a multilingual Hugo site in 3 languages Get in touch →

Discovery settings and indexing signals

We review crawler access through robots.txt, the sitemap and search-indexing settings. This site's Content Signals policy permits search, AI answers and training; your policy may differ. Pages intended for indexing belong in the sitemap with truthful modification dates. IndexNow announces changes to participating engines; receiving a notification does not guarantee that a page is crawled, indexed or ranked.

Read: the text, not the furniture

A page built for eyes is mostly navigation, script and layout, and an agent that pays for every word it reads pays for all of it to reach the paragraph it came for. Every page here has a Markdown twin at the same address: add `index.md`, or ask for `text/markdown` and the server answers in kind. `llms.txt` is a map of the whole site written for a language model, `llms-full.txt` is the site in one file, and every page carries structured data that names the company, its services and where the page sits among the others. Every home page answers with a `Link` header, `rel="describedby"`, that points at `llms.txt`, so an agent learns the map exists before it reads a word of the page.

Usable: able to act, not only to read

Many agents that work a browser read a page the way a screen reader does, through its accessibility tree. The work that makes a site usable without a mouse — real buttons, labelled forms, headings in order, landmarks — is the same work that lets an agent find the form and fill it in. This site answers every criterion of WCAG 2.2 AA in writing. For an application, usable means a described API: an OpenAPI document, an MCP server where agents should call it directly, and the OAuth metadata that lets an agent act for a user without a password changing hands.

Adding a file is **the easy half** — knowing which ones to leave out is the work.

A scanner reports everything it cannot find; somebody has to decide which of it is true for you.

A decision on every recommendation

New proposals for agents arrive faster than they settle, and a scanner will report every one of them missing. We classify each finding rather than obey it: what it does, who reads it, and what it costs to keep true. On this site that meant declining five. One robots rule for every agent rather than one per agent, because a rule written for one crawler silently replaces the general one for it. No API catalogue and no OAuth metadata, because there is no API. No agent card

and no commerce file, because there is nothing here for an agent to call or buy. A client gets the same thing: a list of what to add, and a list of what to leave alone, each line with its reason.

Check it from your own browser

Nothing on this page has to be taken on trust. Read robots.txt, llms.txt, agents.txt and ai.txt, or open this page as Markdown. The build reads every twin for leftover markup and every sitemap date against the page's history, and it refuses to publish when either is wrong — so the half of this site that agents read cannot quietly fall behind the half that people do.

How an engagement runs

It starts read-only. We look at what agents get from you today — the robots file, the sitemap, the text behind the layout, the structured data, the accessibility tree and, for an application, the API — and hand you the findings, each with a decision and its reason. The changes are then made in your repository, through your review, with checks that fail the build when a page stops being readable, so the work does not decay once we leave. It is bought like the rest of our work, on the terms in what it costs.

If you want agents to find you, read you right and get things done for the people who sent them, tell us what you run.

Your next customer **may send an agent first** — make sure it comes back with the right answer.

It starts with a read-only look at what agents get from you today, and every finding arrives with its reason.

Ask for a first look What it costs →

Scope the work before commissioning it

Describe the workflow, the existing interfaces and the users who need the change. We can define a testable delivery and a handover that lets your team maintain it.

See what it costs Get in touch →

Talk through your requirements

Tell us what you need from “Ready for AI agents”, including the deadline and the team that will maintain it. We can discuss a practical way forward.

Explore our services

welcome@engineer.company +45 50 10 61 84 Copenhagen, Denmark

Take this page with you

Contact card

Carefully Crafted With Love by D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – present · Engineer ApS · All rights reserved.

Printed from https://software.engineer.company/ai-ready/