

By: Editorial team | **Updated:** October 15, 2026 | **Questions?** Contact the editorial team through the site's contact page.

A page can look complete to a visitor and still communicate very little to a search system. Key details might sit inside tabs, accordions, pop-ups, carousels, or a JavaScript interface that only works after a click. The result: a person can find the information, but the page's content visibility, accessibility, and crawl interpretation are weaker than they need to be.

This matters beyond traditional search rankings. AI-powered search tools and large language models may draw on crawled pages, indexed documents, or other available sources to answer a question. They do not all process pages the same way, and no layout can guarantee inclusion. But if important facts are absent from the page's accessible, readable content—or are difficult to interpret—your odds of being understood get worse.

The fix is not “remove every tab.” It's to make sure the information customers need is available, clear, and consistently represented in the page structure, whether someone reads the page, uses assistive technology, or encounters it through a search system.

What “hidden” content actually means

Content is often called hidden when it is not visible immediately on page load. That can describe several different situations, and they do not all carry the same risk.

- **Accessible, labeled tabs or accordions:** The content is present in the page and can be opened with a mouse, keyboard, or assistive technology.
- **Content loaded after interaction:** A click triggers a request or script that fetches the content. A search crawler may or may not trigger that action.
- **Content rendered only in the browser:** The page depends on JavaScript to create or display the text.
- **Content blocked or missing:** The information is unavailable to users, crawlers, or both because of errors, permissions, or broken controls.

“It's on the page somewhere” is not a useful test. Ask instead: Is the content present in the rendered page? Can someone reach and understand it without a mouse? Does each section have a clear label? Can a crawler retrieve the text reliably? And does the page make the information's meaning clear without relying on visual placement alone?

Why AI-powered search changes the question

Traditional SEO is often discussed in terms of rankings: which page appears for a query and where it appears. AI-powered search adds another concern: whether a system can identify a useful passage, connect it to a specific question, and represent it accurately in a response.

That is not a separate shortcut around technical SEO. Search products vary in how they crawl, index, retrieve, and summarize sources. Large language models may also be used in different ways across products, and a model's general training is not the same thing as live web retrieval. So avoid claims that a particular tab will “guarantee visibility.” Instead, make the content easy to retrieve and hard to misread.

Three practical signals help:

- **Relevance:** The page answers the question a customer actually has, using specific language and useful detail.

- **Credibility:** The information has a clear source, a qualified author or reviewer where appropriate, and enough evidence to support its claims.
- **Consistency:** Important facts agree across the page, headings, structured data, and related pages. Conflicting prices, eligibility terms, or definitions create doubt.

Good crawl interpretation supports all three. A bot that can retrieve a paragraph still needs to understand which product it describes, who qualifies, and whether a price is current. Clear headings, plain language, and visible context help both people and machines.

The common mistake: no pricing information in the scraped article body

A frequent failure in financial-services content is putting pricing only inside a quote widget, a dynamic result, or a separate application flow. The article may discuss business insurance at length, but the scraped article body contains no pricing information at all—not even an indicative range, the factors that affect cost, or a direct explanation that a quote is required.

That leaves a basic customer question unanswered: “What might this cost, and what changes the price?” It also makes the page less useful when someone compares options or asks an AI-powered search tool for a cost overview. Don’t invent a price just to fill the gap. If prices vary by business, say so plainly, explain the main pricing factors, and link to a real quote path. If you can publish a representative range, state its date, assumptions, and limitations.



For example, a small consultancy cover page might explain that premiums depend on the type of work, turnover, number of staff, claims history, and selected limits. If no representative price is available, tell readers how to get a quote and what application documents they may need. Put that explanation in readable page content—not only in a widget that appears after several steps.

Start with the customer’s question

Before changing the interface, check what visitors are trying to do. A running list of questions from support tickets, sales calls, and search queries is often more useful than a list of keywords. Group those questions by task, then

make sure the page answers the highest-priority ones in a place people can find.

Questions might include:

- What does business interruption insurance cover, and what does it exclude?
- What information do I need before applying for commercial lending?
- Does business insurance cover contractors or only employees?
- What affects the cost of insurance for a small consultancy?
- Which application documents should I prepare?

Use the wording customers use, then answer with precise definitions and qualifications. If the answer changes by product, region, or applicant type, make that distinction explicit. A vague “Learn more” tab label does little for either accessibility or crawl interpretation; “Eligibility and required documents” tells people what they will find.

Audit the page before redesigning it

Choose a page that matters to customers or revenue and inspect it in several ways. Do not rely on a screenshot alone. A polished visual layout can conceal missing text, a broken control, or content that only appears for some users.

1. **Read the page with JavaScript disabled.** This is a diagnostic, not a perfect imitation of every crawler. Note which important text disappears.
2. **Inspect the rendered DOM.** Check whether tab and accordion content exists in the rendered page, whether it is added only after interaction, and whether labels are connected to their panels.
3. **Test keyboard access.** Use Tab, arrow keys where appropriate, Enter, and Space. Confirm a visible focus state and a sensible reading order.
4. **Use a screen reader or accessibility checker.** Verify that controls announce their role, name, and state, and that content is not skipped or read in a confusing sequence.
5. **Check crawler access.** Review server responses, robots rules, rendered HTML, and relevant search-console tools. Confirm that the page and its resources can be fetched.
6. **Compare the page with customer questions.** Look for missing answers, including pricing context, exclusions, eligibility, and next steps.

Record what is visible before interaction, what appears after interaction, and what is available in the rendered source. That distinction helps developers identify whether the problem is design, accessibility, rendering, or content—not just “SEO.”

Choose the right fix for the interface

Interface pattern Common risk Useful improvement
Tabs Vague labels; panels difficult to reach by keyboard; content fetched only after a click Use descriptive labels, accessible tab behavior, and reliably rendered panel content. Consider showing key facts outside the tabs.
Accordions Important answers start closed or controls do not announce expanded state Use real buttons, clear question labels, and an accurate expanded/collapsed state. Keep core answers in the page.
Carousels Later slides are missed; essential details depend on auto-rotation Provide controls, pause behavior, accessible labels, and a non-carousel way to reach key information.
Quote or application widgets Pricing, eligibility, or document requirements appear only after data entry Explain the process and pricing factors in article text; link to the tool for personalized results.
Modal or pop-up Content blocks the

page or cannot be dismissed with a keyboard. Use sparingly, manage focus correctly, and avoid putting the only copy of essential information inside a pop-up.

These patterns can be implemented accessibly, but “accessible” does not automatically mean “easy for a search system to interpret.” Nor does visible text guarantee that a crawler will index it. Test the actual implementation instead of assuming the component’s name or framework settles the question.

Make the content understandable on its own

Each section should make sense when encountered out of context. That matters when a search system retrieves one passage rather than the whole page, and it helps people scan quickly.

Write headings that describe the answer

Prefer “What affects the cost of business interruption insurance?” over “Costs” or “More information.” Keep a logical heading order, and avoid using headings only to style text.

Put the answer before the caveat maze

Start with a direct, qualified answer, then explain the conditions. For instance: “Cover may help startnewswire.com replace lost income after an insured interruption, subject to the policy’s terms, limits, and exclusions.” Follow with examples of covered events and exclusions, and direct readers to the policy wording. Do not imply that all disruptions are covered.

Make next steps concrete

“Apply now” is a call to action, not an explanation. Tell readers what happens next, how long it may take if you can substantiate that claim, and what application documents to prepare. For commercial lending, distinguish general information from lender-specific requirements. For small consultancy cover, explain whether the reader needs a quote, a policy review, or both.

Build credibility into the page

First-hand experience and author credibility are not decorative extras. They help readers judge why they should trust an explanation, particularly in financial services, where terms and eligibility can have real consequences.

- Name the author and, where relevant, the qualified reviewer. Link to useful biographies that explain their experience.
- Identify the company responsible for the page and provide a real contact path, not just a generic promise to “get in touch.”
- Show when the page was published and reviewed or updated. Keep the date honest; do not change it without a substantive review.
- Link to authoritative policy wording, application guidance, or other primary sources where appropriate.
- Distinguish general education from personalized advice or a quote. State important limits and exclusions near the claims they qualify.

First-hand detail should be specific and supportable. A product team might explain why customers commonly misunderstand an exclusion or which documents applicants often need to find. Avoid presenting anecdote as universal policy. If an example is illustrative, label it that way.

Keep facts consistent across formats

Once the main page is fixed, compare it with product pages, FAQs, structured data, and application flows. The same eligibility rule should not be described differently in a tab and in the main article. Keep prices, dates, coverage descriptions, and contact details aligned. Structured data can help describe a page, but it does not replace visible, accurate content or repair a broken interface.

For a financial services company, a useful content set might include a plain-language business insurance overview, a separate commercial lending guide, a detailed business interruption insurance explainer, a small consultancy cover page, and an application documents checklist. Link between them where the next step genuinely helps the reader. Do not repeat the same keyword in every paragraph; explain each product in the language customers use.

Measure whether the fix worked

After release, test the page again with a keyboard, assistive technology, and your crawl or rendering tools. Track practical outcomes: whether key text appears in rendered output, whether important controls work, whether users reach the intended next step, and whether support or sales teams hear fewer versions of the same unanswered question.

Search impressions and referrals can be useful, but they are not proof that a page is clear or accessible. Nor is inclusion in an AI-powered answer something you can promise. Review performance over time and investigate changes alongside technical updates, content changes, and demand.



A simple standard for content visibility

Before calling the work done, ask whether a customer can find the answer, whether a keyboard or screen-reader user can reach it, and whether a crawler can retrieve and interpret it reliably. Then check that the answer is relevant, credible, and consistent with the rest of the site.

Tabs and accordions are not automatically bad. They become a problem when they hide essential information from the people and systems that need to understand it. Put the customer's question first, make the interface work for everyone, and keep the substance available in clear, dependable page content.