



Resources » The Ultimate Guide to Knowledge Base Management

# Knowledge Base Maintenance: A Practical Framework

Mathew Patterson • January 14, 2025



- The Ultimate Guide to Creating a Customer Knowledge Base
- The 10 Best Knowledge Base Software in 2026
- Knowledge Base Design Tips for Better Self-Service Support
- 5 Simple Knowledge Base SEO Tips Anyone Can Follow
- Formatting Tips to Improve Your Knowledge Base Articles
- Tips & Templates for Writing Great Knowledge Base Articles
- Creating Knowledge Base Videos: Tips, Tools, and Examples
- Knowledge Base Maintenance: A Practical Framework**
- 10 Actionable Knowledge Base Metrics to Start Tracking Today
- How to Revamp Your Knowledge Base Architecture
- 18 Knowledge Base Examples That Get It Right

What do mechanics, doctors, gardeners, dentists, and support teams all have in common? There are at least two things:

- 1.) They could all have a “House, M.D.”-style show in which they diagnose and solve mysterious issues, and I would watch it.
- 2.) They all spend a lot of their time trying to convince their various clients that a little maintenance now is better than dealing with a much larger problem later.

In reality, maintenance is not glamorous enough for a TV show. Nobody is getting a special slide in the company town hall meeting celebrating maintenance work. It often goes entirely unrecognized — unless it’s not done.

Knowledge base maintenance is absolutely critical to quality customer support, but it is too often under-resourced, neglected, or entirely forgotten. However, with a little thought, planning, and ongoing effort, we can build a sustainable system to address maintenance. Here’s how to get it done.

## Elements of knowledge base maintenance

Whether you call it documentation or your wiki or help content, a knowledge base should do what it says on the tin: store knowledge. It contains information intended to be useful to the people (and, these days, the machine learning AI algorithms) that consume it.

Although the initial writing of those documents can be a huge effort, at that point the work is far from over. Maintaining a knowledge base includes:

- Regularly reviewing documents for accuracy and completeness.
- Updating documents to reflect changes to the subject of the documentation, such as new features or changed behaviors.
- Improvements and updates to navigation and knowledge base structure to make them more findable and accessible.
- Removal of information that is no longer relevant or correct.
- Reviewing customer and support team feedback and adjusting documentation accordingly.
- Expansion and addition of help documents to provide better coverage of the topic area.

Just reading that list might feel overwhelming. Take a breath. This is doable; you simply need a plan and an understanding of why the outcome of maintenance is worth the effort.

## Why knowledge base maintenance matters

A knowledge base should not be a dusty old history book. It should be a living set of documents with genuine practical benefits, and that requires maintenance. The primary reason knowledge bases matter is because they are the engine of customer self-service. A customer who doesn’t have to wait for a reply from the support team is on their way more quickly and has removed some of the support team’s workload.

That only works when the self-service resources are accurate, helpful, and easily found and understood. Keeping self-service documentation in that state is what knowledge base maintenance is all about.

If your customers try to self-serve but find that the documentation doesn’t answer their questions or is inaccessible or inaccurate, they’ll soon learn not to bother. Similarly, your own support team will not rely on the knowledge base to answer their own questions.

Very quickly that will result in a knowledge base actively decaying, as people avoid using or adding to it. Just like a garden, your knowledge base needs to be regularly pruned to encourage new growth.

Additionally, today’s knowledge bases have a new audience: artificial intelligence. AI tools are slurping up documentation and using it to answer customers directly or to provide drafts for support teams to work with. Whereas customers and support staff will usually notice an abandoned knowledge base and just avoid using it, AI can’t typically make that judgment. It will happily pump out old and inaccurate information to your customers.

Maintenance clearly matters. So why is it so often not done?

## Why we fail to maintain knowledge bases

Maintenance is often left to linger at the bottom of the support team’s To Do list. Each team might have their own conscious or unconscious reasons, but some of the most common include the following:

- **Maintaining a knowledge base is hard.** It requires a lot of knowledge just to realize when an update is needed, then crafting an update takes time, effort, and skill.
- **You don’t know where to start.** This is especially true if it has been left so long that the task feels overwhelming to even consider.
- **There’s a lack of visibility.** You may not know which help pages even exist, let alone which ones need to be fixed. Missing metric and reporting systems are a common issue.
- **There’s no clear owner.** If nobody has ownership of the knowledge base, it’s easy for everyone to avoid action.
- **It’s never quite urgent enough.** Unlike the support queue, which might be currently burning down, an out-of-date knowledge base isn’t immediately causing obvious problems. The downstream problems only show up later.
- **The tools are hard to use.** Software tools to both identify pages needing help and to update those pages can add significant friction to the process.

Your list may vary, of course, but ultimately the next steps are the same. We need to find a way to make maintenance a normal part of our workload.

## How to (consistently) maintain a knowledge base

There is no one-size-fits-all process for knowledge base maintenance (nor, for that matter, do one-size-fits-all clothes actually “fit all”). Instead, look through the following suggestions and apply the ones that best target your specific company and team situation.

### Identify an owner

It doesn’t have to be a full-time documentarian; you just need somebody who can be the decision-maker on which documents need updating, in what order and how it will be done. Merely identifying an owner prevents maintenance work from stalling because nobody knows who should take action.

### Create the time and space for maintenance

Maintenance work demands time and energy, and unless that time is budgeted for and protected, it cannot happen. Team leaders, directors, and VPs — this is your job. If maintenance matters to you (and it *must*), you need to build it into your financial and headcount planning.

### Make the costs and benefits visible

How many customers are you seeing in the queue because there was no knowledge base article (or they couldn’t find it or it was inaccurate)? Your software may record failed searches, but even if not, a little manual review of common questions will generate data.

If the costs of poor maintenance are visible to others, it will be easier to push for the time and attention it needs.

### Create a clear knowledge base maintenance process

Often the biggest roadblock to timely maintenance is just people not knowing how to get a page updated. Considering your specific technology stack, the capabilities of your team, and the scale of your knowledge base, make a plan to:

- **Identify what needs fixing.** Anyone in the company should be able to flag a document that’s in need of review. Could you set up a help desk tag, Slack emoji, or email address to ping things to? Publicize them internally and reward people for contributing. Similarly, make it easy for customers to report issues directly from your knowledge base. Integrated feedback forms and support forms that track the recently visited knowledge base pages can help.
- **Turn an idea for improvement into completed maintenance.** What steps need to happen, and who needs to take those steps? Your maintenance process should clearly outline who can make updates, who needs to review them, and where those changes take place.

Publish your maintenance plan internally, and make sure everyone knows about it.

### Schedule regular reviews

Outside of individual page updates, build in a larger regular review process where you can sweep through the whole knowledge base and note any obvious issues. A good spring clean sets you up well for the rest of the year.

### Build documentation into your release processes

If you’re documenting software or other regularly updated products, then you’ll need to be consistently adding to your documentation. Work with the product teams to embed documentation as a step in their own release processes so you’re less likely to be surprised by sudden changes and scrambling to keep up.

### Spread the workload

Look for ways to get the whole team involved. Perhaps your team members don’t feel comfortable writing a public doc, but could they tag a private answer to a customer as “should be documented” and provide a draft for your tech writer to work from? A custom field or reporting tag in your help desk might be perfect for this situation.

If your writer can spend less time trying to identify what needs fixing and more time actually crafting the documents, your maintenance will happen much more quickly.

### Design for maintainability

Wherever possible, avoid having multiple documents containing the same information. More “atomic” documents covering smaller chunks of your system or products will be simpler to maintain than giant documents covering multiple areas.

Learn from the [digital garden approach](#) by prioritizing cross-linked snippets of information and clearly defined individual notes. When you come to update them later, it will be easier to spot the topics each doc covers.

### Reduce the friction

If every maintenance job seems massive and intractable, it won’t happen. Instead, optimize your processes for small, frequent changes that keep everything up to date without a giant project. Reduce the number of steps between noticing a document needs updating and publishing that update as much as you can.

Having a good publishing tool will help here; make that a consideration when you’re next reviewing your software options.

## Make that stitch in time

You know maintenance is good for your team, your customers, and yourself. You understand it needs to be done and deserves your time and attention. All that remains is getting it done.

Make your own life as easy as possible by creating an environment conducive to knowledge base maintenance. Schedule it, report on it, and make it visible. You’ll feel better when it’s done.

Oh, and don’t forget to floss your teeth. I know you’ve been lying to the dentist about that.

Previous Chapter

**Creating Knowledge Base Videos: Tips, Tools, and Examples**

Next Chapter

**10 Actionable Knowledge Base Metrics to Start Tracking Today**

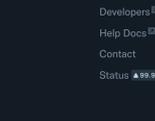
# START FOR FREE

Learn the *power user* in less than an hour.

Become a *power user* in less than a day.

- Start for Free

Book a Demo



<p><b>Product</b></p> <ul style="list-style-type: none"> <li>Inbox</li> <li>AI Chatbot</li> <li>Messages</li> <li>Insights &amp; Analytics</li> <li>Apps &amp; Integrations</li> <li>Mobile</li> <li>Product updates</li> </ul>	<p><b>Solutions</b></p> <ul style="list-style-type: none"> <li>Education</li> <li>SaaS</li> <li>Travel &amp; Hospitality</li> <li>Real Estate</li> <li>Property Management</li> <li>Logistics</li> <li>Manufacturing</li> <li>Financial Services</li> <li>Professional Services</li> </ul>	<p><b>Compare</b></p> <ul style="list-style-type: none"> <li>Zendesk</li> <li>Intercom</li> <li>Freshdesk</li> <li>HubSpot</li> <li>Gorgias</li> <li>Front</li> </ul>	<p><b>Company</b></p> <ul style="list-style-type: none"> <li>About</li> <li>Customers</li> <li>Careers</li> <li>Newsletters</li> <li>Terms &amp; Privacy</li> <li>Accessibility</li> <li>Do Not Sell</li> </ul>	<p><b>Support</b></p> <ul style="list-style-type: none"> <li>Developers</li> <li>Help Docs</li> <li>Contact</li> <li>Status <span style="color: green;">▲ 99.99%</span></li> </ul>
---	--	---	---	--

