

Abkus Support - Manual & FAQ

Onboard and help in one PDF. Generated from the same source as the online help.

Index

- Getting started
- Building the knowledge base
- Import knowledge from a URL or a PDF
- Inbox and tickets
- Working a ticket: macros, status, SLA and CSAT
- Embed the widget
- Billing and subscriptions
- Developer integration
- Public widget limits and safety

Manual & FAQ

Getting started

Abkus Support answers your customers from your own knowledge, and hands the conversation to a human whenever it is not sure.

1. Sign in with your Abkus account (Hub SSO - no separate password).
2. Your workspace is created for you on first sign-in - your knowledge base and widget key live there.
3. Add knowledge: paste text, import a page by URL, or upload a PDF.
4. Send a test message and check the answer cites your knowledge.

Building the knowledge base

Every AI answer is grounded in your knowledge. No grounding means no answer - the ticket escalates instead.

1. Add short, factual documents (one topic each) - pasted, imported by URL, or uploaded as PDF.
2. Keep them current: an outdated policy becomes a wrong answer.
3. Re-index after a bulk change so retrieval sees the new text.
4. Verify with a question whose answer you can check by hand.

Import knowledge from a URL or a PDF

Abkus Support can take knowledge from your public help pages and from PDF manuals, and the imported text lands in the same knowledge base - chunked and embedded like a pasted document.

1. Paste text directly when the content is short.
2. Import a public page by URL: the dashboard fetches it, keeps the readable text, and refuses private, internal or unreachable hosts.
3. Upload a PDF manual: the text is extracted, and the file is refused if it is scanned (no text layer) or larger than 10 MB.
4. Check the imported document's title and chunk count, then ask a question whose answer you can verify by hand.

Inbox and tickets

Each conversation is a ticket. The AI answers first; you take over when it escalates.

1. Open the dashboard to list tickets and their status.
2. Open a ticket to read the full thread with citations.
3. Reply as an agent - replying reopens an escalated ticket.
4. Assign the ticket to the person who owns it.

The search box scans the newest 2 000 tickets and keeps the filters you set. A query containing @ also matches the requester's e-mail (that column is encrypted, which is why the scan is bounded); the answer says when it reached the limit. More than 30 searches per minute are paused for a moment - wait and search again.

Working a ticket: macros, status, SLA and CSAT

Day-to-day work happens on the ticket: reply, set the right status and priority, and close the loop with the

customer.

1. Reply as an agent - a reply reopens an escalated or resolved ticket.
2. Change status and priority in the ticket header; "Mark as resolved" closes it and lets the visitor rate the support.
3. Insert a reply macro (a canned answer) in the reply box - {subject}, {requester} and {workspace} are filled with this ticket's data.
4. An escalated ticket shows WHY it reached a human (no grounding, low confidence or a safety case) - replying takes the handoff and reopens it.
5. Watch the first-response SLA badge: on time, due soon or late, measured against the team's target for that priority.
6. Read the CSAT: the visitor rates the support once (1 to 5) after the ticket is resolved, and the dashboard shows the workspace average.
7. Find tickets with the inbox filters (status, priority, assignee, subject or requester e-mail); filtering never changes the workspace numbers.
8. Take the ticket with Assign to me so the rest of the team sees it is yours, and release it with Unassign. Replies are capped at 8 000 characters.

Embed the widget

Any Abkus app can show the support widget without shipping its own chat UI.

1. In the dashboard, turn the widget on to get its public widget key.
2. List the origins allowed to embed it (your site's origins).
3. Paste the loader into the app page, setting data-key to that widget key.
4. Publish, open the app, and confirm a message becomes a ticket.

Billing and subscriptions

Subscriptions and invoices are handled by the Abkus Hub, not by this app.

1. Open your account from the user menu.
2. Review your plan and invoices under subscriptions.
3. Change plan or payment method there.
4. If a payment fails, access continues during the recovery window.

Plans and prices are listed on /pricing, read from the Hub catalog, and the service health is published on /status.

Developer integration

Post a message with your Hub session and it becomes a ticket in YOUR workspace.

1. Call POST /api/widget with your workspace's appSlug and body.
2. The engine answers from that workspace's knowledge, or escalates.
3. Continue the thread by sending the ticketId back.
4. All endpoints require a Hub session.

Public widget limits and safety

The public widget is closed by default: the workspace lists the origins allowed to EMBED it, writing needs a short-lived token, and posting is rate-limited.

1. Only origins listed for the workspace may embed the widget; an unlisted site cannot.
2. Writing needs a short-lived token the page obtains first - the public key alone cannot post, and that token is what keeps a visitor on their own thread until it expires.
3. Posting is rate-limited per IP and per widget key; a burst is refused, not queued.
4. Messages are limited to 4000 characters and must be JSON; there are no attachments.
5. A security incident, a legal threat, or a refund dispute always goes to a human.

Frequently asked questions

Which app can Abkus Support answer for?

Any app that plugs the support widget or posts messages to its API. Each app gets its own workspace and knowledge base.

Does the AI answer on its own?

It answers when it has grounded knowledge; otherwise it escalates to a human with the full conversation.

Where does billing happen?

Subscriptions are handled centrally by the Abkus Hub, like every Abkus app.