

Authoring Standards

- [KB Article Template](#)
- [Status Legend](#)
- [Prompt Rules](#)
- [BookStack book and chapter map](#)
- [UI label verification checklist](#)
- [Article review checklist](#)
- [Screenshot placeholder rules](#)

KB Article Template

Purpose

Use this template for every LinkUp knowledge base page. It keeps the public help simple while giving us a small implementation check under the same article.

Page template

Section	What to write
Title	Use a short task-based title, such as Buy a ticket or Confirm an offline reservation .
Public help	Explain the task in plain language for the reader. Use short steps and avoid developer wording.
Expected LinkUp behavior	State what the product should do when the feature works correctly.
Implementation status	Mark the feature as Implemented, Partial, Legacy Only, Missing, Planned, or Needs Verification.
Verification checklist	Add simple checks that prove the feature works in the web app, mobile app, admin panel, or scanner app.
Gaps / notes	Record anything unclear, risky, or different from the intended behavior.

Writing rules

- Write for one audience at a time.
- Use second person: “you”.
- Keep each step to one action.
- Use **bold** for buttons, screens, and fields.
- Use `[VERIFY: ...]` when a screen, button, or workflow has not been confirmed.
- Do not describe planned features as live features.

Implementation check table

Expected behavior	Status	Surface	Manual check
Every KB page includes public help and internal implementation notes.	Implemented	BookStack	Create or review a page and confirm both sections exist.
Unconfirmed UI is marked before publishing.	Implemented	Authoring process	Search the article for [VERIFY:] markers before release.

Gaps / notes

This is the first local standard. Update it after the first five to ten articles if the structure feels too heavy or too light.

Status Legend

Purpose

Use this legend to label every LinkUp feature the same way. The label should describe the current truth, not the future plan.

Status labels

Status	Meaning	When to use it
Implemented	The feature exists and matches the intended workflow.	Use after the feature has been checked in the app, admin panel, API, or mobile app.
Partial	Some pieces exist, but the full workflow is not complete.	Use when web, mobile, admin, or backend support is uneven.
Legacy Only	The feature exists only in the old system or old screen.	Use when the feature has not moved into the current LinkUp surface.
Missing	The feature does not exist yet.	Use when no route, screen, admin tool, or mobile flow can be found.
Planned	The feature is intended but not live.	Use for roadmap items, proposed workflows, or future policy.
Needs Verification	The docs or code suggest the feature exists, but it has not been tested.	Use when a human needs to check the real screen before publishing.

Rule of thumb

If we are not sure, use **Needs Verification**. Do not turn uncertainty into a confident help article.

Implementation check table

Expected behavior	Status	Surface	Manual check
Every KB article uses one of the approved status labels.	Implemented	BookStack	Review each page before publishing.

Unverified workflows stay marked until tested.	Implemented	Authoring process	Search BookStack for <code>Needs Verification</code> and <code>[VERIFY:]</code> .
--	-------------	-------------------	---

Gaps / notes

This page is an authoring control. It does not prove whether a product feature works.

Prompt Rules

Purpose

Use these rules when asking Codex or Claude to draft a LinkUp KB article. The goal is simple public help plus clear implementation truth.

Required prompt inputs

Input	What it means	Example
Audience	Who the page is written for.	attendee, organizer, scanner, internal team
Topic	The task or question the page answers.	Buy a ticket
Doc type	The shape of the article.	how-to, troubleshooting, reference, policy
Feature area	The product area being documented.	payments, scanning, Discover, admin
Source context	The docs, code, screenshots, or notes the writer should rely on.	Route, screen, model, test plan, or audit note

Prompt rule

Ask for one page at a time. Include source context before asking for the final article. If the source is uncertain, ask the model to mark it with `[VERIFY: ...]`.

Banned habits

- Do not invent button names.
- Do not describe planned features as live.
- Do not use vague instructions like “click the button”.
- Do not bury implementation gaps in friendly wording.

Implementation check table

Expected behavior	Status	Surface	Manual check
The prompt produces a public article and internal implementation notes.	Implemented	Authoring process	Generate one test page and confirm both sections exist.
The prompt supports BookStack-ready output.	Implemented	BookStack	Paste the article into BookStack and confirm headings/tables render cleanly.

Gaps / notes

The reusable DOCX prompt should be updated after these first pages settle.

BookStack book and chapter map

Public help

Define where each type of LinkUp article belongs.

Rule

Keep authoring simple, consistent, and honest about uncertainty.

Implementation status

Expected behavior	Status	Surface	Manual check
Authoring standard is followed across pages.	Needs Verification	BookStack	Review a sample of pages after each batch.
Verification markers are resolved before public release.	Needs Verification	BookStack	Search for <code>[VERIFY:]</code> before publishing.

Verification checklist

- Review headings and structure.
- Check for unverified UI labels.
- Confirm status table exists.
- Confirm page is in the right book.

Gaps / notes

- These standards should evolve after the first review pass.

UI label verification checklist

Public help

Prevent wrong button, screen, or field names from being published.

Rule

Keep authoring simple, consistent, and honest about uncertainty.

Implementation status

Expected behavior	Status	Surface	Manual check
Authoring standard is followed across pages.	Needs Verification	BookStack	Review a sample of pages after each batch.
Verification markers are resolved before public release.	Needs Verification	BookStack	Search for <code>[VERIFY:]</code> before publishing.

Verification checklist

- Review headings and structure.
- Check for unverified UI labels.
- Confirm status table exists.
- Confirm page is in the right book.

Gaps / notes

- These standards should evolve after the first review pass.

Article review checklist

Public help

Review every page for clarity, status accuracy, and verification markers.

Rule

Keep authoring simple, consistent, and honest about uncertainty.

Implementation status

Expected behavior	Status	Surface	Manual check
Authoring standard is followed across pages.	Needs Verification	BookStack	Review a sample of pages after each batch.
Verification markers are resolved before public release.	Needs Verification	BookStack	Search for <code>[VERIFY:]</code> before publishing.

Verification checklist

- Review headings and structure.
- Check for unverified UI labels.
- Confirm status table exists.
- Confirm page is in the right book.

Gaps / notes

- These standards should evolve after the first review pass.

Screenshot placeholder rules

Public help

Explain when to add screenshots and when to leave a verification placeholder.

Rule

Keep authoring simple, consistent, and honest about uncertainty.

Implementation status

Expected behavior	Status	Surface	Manual check
Authoring standard is followed across pages.	Needs Verification	BookStack	Review a sample of pages after each batch.
Verification markers are resolved before public release.	Needs Verification	BookStack	Search for <code>[VERIFY:]</code> before publishing.

Verification checklist

- Review headings and structure.
- Check for unverified UI labels.
- Confirm status table exists.
- Confirm page is in the right book.

Gaps / notes

- These standards should evolve after the first review pass.