

Release Notes Template

All 11 sections, with a worked example filled in end to end.

This is the same template that sits on the page, in print form. The table lists the 11 sections and what goes in each one. The pages that follow show every section filled in for one release, so you can see a real value in each field before you write your own. Delete the sections your release does not touch.

#	SECTION	WHAT GOES IN IT
1	Header line	Product name, version number, release date, and the platform or environment the release applies to.
2	Summary of changes also called "Purpose"	One or two sentences on what this release addresses and who it affects.
3	New features	Each one named, with the user-facing capability rather than the internal build.
4	Improvements and enhancements also called "Performance Improvements"	Changes to things that already existed, described from the user's side.
5	Bug fixes	Plain-language description of what the user saw before and what happens now.
6	Breaking changes and migration steps	What stops working and what the reader has to change.
7	Known issues and workarounds also called "Ongoing Issues" or "Known issues or limitations"	Unresolved problems, affected platforms, and a timeline where one exists.
8	Upgrade or installation steps	What the reader must do to get on the new version.

#	SECTION	WHAT GOES IN IT
		also called "Upgrade Steps", "Upgrade notes" or "Installation Notes"
9	Deprecations	Features, integrations or APIs being sunsetted, with the date.
10	Security notes	Security-relevant changes, stated without exposing the vulnerability.
11	Where to get help and give feedback	Support contact, docs link, or feedback channel.

HEADER LINE FIELDS

Where your team tracks releases in a project tool, the header line can also carry: version number, release date, platform, assignee, impact level, documentation status, and feedback channel.

Filled-in example: Fieldpost 4.2.0

The same template, filled in for a fictional dispatch product shipping a feature release.

1 · HEADER LINE

Fieldpost 4.2.0 · **Released:** March 12, 2026 · **Platform:** Web and iOS

2 · SUMMARY OF CHANGES

Exports now finish on large accounts, Canadian pricing is correct, and self-hosted admins have one migration step.

3 · NEW FEATURES

- **Scheduled exports.** You can now set a report to run every Monday and land in your inbox.
- **Bulk status updates.** Select up to 500 jobs and change their status in one action.

4 · IMPROVEMENTS AND ENHANCEMENTS

- **Faster dispatch board.** The board loads in about two seconds on accounts with 10,000 open jobs, down from nine.

5 · BUG FIXES

- Fixed: exports over 50,000 rows timed out and returned an empty file.
- Fixed: prices showed in USD for Canadian accounts.

6 · BREAKING CHANGES AND MIGRATION STEPS

- Version 4.2.0 removes the v1 /exports endpoint. Point integrations at /v2/exports, which returns a job ID. Self-hosted installs run `fieldpost migrate --v2-exports` before starting 4.2.0.

7 · KNOWN ISSUES AND WORKAROUNDS

- On iOS, exports scheduled for Sunday run on Monday instead. Use the web app until 4.2.1 ships on March 26.

8 · UPGRADE OR INSTALLATION STEPS

- Cloud accounts are already on 4.2.0. iOS users update from the App Store by March 31.

9 · DEPRECATIONS

- Fieldpost deprecates the CSV-only report format on September 1, 2026. Switch saved reports to XLSX.

10 · SECURITY NOTES

- Session tokens now expire after 12 hours, and admins can end another user's sessions.

11 · WHERE TO GET HELP AND GIVE FEEDBACK

Docs: docs.fieldpost.example · Support: support@fieldpost.example · Feedback: the Feedback tab in the app.

Section checklist

Work down the list before you publish. Fill the section in, or delete it because this release does not touch it.

☐ Header line — product name, version number, release date, platform or environment

☐ Summary of changes — what this release addresses and who it affects

☐ New features — the user-facing capability, not the internal build

☐ Improvements and enhancements — what already existed, and what is better about it

☐ Bug fixes — what the user saw before, and what happens now

☐ Breaking changes and migration steps — what stops working, and what to change

☐ Known issues and workarounds — problem, affected platform, workaround, fix timeline

☐ Upgrade or installation steps — what the reader must do to get on the new version

☐ Deprecations — the feature, integration or API, and the date it sunsets

☐ Security notes — the change, stated without describing the vulnerability

☐ Where to get help and give feedback — docs link, support contact, feedback channel

KEEP THE SHAPE

Use the same 11 sections release after release, and delete rather than reorder the ones a given release does not need. A patch release drops new features, improvements, deprecations and upgrade steps, and leads with the fix, the impact, and whether the reader has to do anything.