Software by Design
Nextcloud apps

Stock · Locations · Clear bookings

InventoryCheck

User manual

End-user documentation for the web app

Items, locations, movements and stocktake — step by step.

App version	Language	Date
1.5.1	English	25 October 2026

Contents

Introduction i
1 Roles and permissions 1
2 Getting started 2
3 Dashboard — the daily overview 4
4 Items — your master data 5
5 Locations — where the stock lives 7
6 Movements — the append-only journal 8
7 Reordering — seeing the need early 10
8 Order sheets & suppliers 12
9 Stocktake — counting against the books 14
10 Labels and CSV export 15
11 CSV import — bootstrapping the catalogue 16
12 Team, office and the access door 17
13 The mobile companion — where the licence line is 19
14 Settings — the administrator reference 20
15 Accessibility 23
16 Errors, messages and FAQ 24
17 Glossary 27
18 Surface catalogue 28
About Software by Design 29
Support & contribute 29
Index 30

Introduction

InventoryCheck is the Nextcloud app for stock, locations and auditable movements on your instance. You manage items with SKU and scan code, book receive, issue, transfer and adjust as an immutable journal, spot low stock early, and run stocktake campaigns from create through counting to close. The web UI is free (AGPL); the optional companion needs a mobile licence (IV2) and is only bordered here.

This manual is organised around what you are trying to do. Each task chapter explains the idea first — why the journal is append-only, why closing a stocktake writes adjustments — then walks the procedure step by step, then lists the edge cases that bite in practice. If you only remember one thing: nothing is silently overwritten. Mistakes are corrected by new, visible bookings, not by rewriting history.

As of: App version **1.5.1** (commit 9812943), Companion **1.0.4**, 1 November 2026, English (professional).

Atlas farm POLICY 3.5.17 dual ACCEPT (exit_met); this build adds the restock suite — reorder view, on-order markers, order sheets with approval, supplier and pack fields, lot expiry, kits, weekly digest and dashboard widget (Ch. 7–8); earlier in 1.5.0: issued materials can be pushed to the customer's InvoiceCheck draft invoice via **Connections** (Ch. 14); 1.4.2 added the optional **Online product lookup** (Ch. 4). **Audience:** Field · Office/store · App / Nextcloud administrators.

What you need: InventoryCheck access · browser with JavaScript · app-admin rights for settings.
Chapters: Ch. 1 roles · 2–6 daily work · 7–8 restock · 9 stocktake · 10–11 labels/CSV · 12 access/team · 13 companion · 14 settings · 15 accessibility · 16 errors/FAQ · 17 glossary · 18 catalogue.
Register: Body text in professional English. UI chrome in quotes, e.g. **Items**, **Receive stock**, **Stocktake** and **Access denied**.
Quick start: Create a location + item → **Receive stock** → issue/transfer in daily work → count and close a stocktake when needed.

1 Roles and permissions

InventoryCheck does not maintain its own user accounts. Everything you may do follows from three building blocks that already exist on your Nextcloud: your group memberships, the *office list* inside the app, and an optional *access restriction* that can gate the whole app. Read this chapter once before you touch stock — almost every "why can't I …" question in chapter 16 comes back to one of these three lists.

1.1 The three roles

Your role is shown as a coloured badge next to the page title. It is resolved in this order:

Administrator You are a Nextcloud system administrator, or you are listed under **Settings** → **Access** as a **Delegated app administrators** entry. Administrators see every surface including all settings pages and the licence section.

Office You are named under **Settings** → **Office** as an **Office users** entry, or you belong to an **Office groups** group. Office is the storekeeper role: master data, receive, adjust, stocktake campaigns, import and export.

Field Everyone else who may open the app. Field is the daily-work role: read items and locations, issue and transfer stock, and type counted quantities into a stocktake someone else started. Field cannot receive, adjust, import, or close a stocktake.

A Nextcloud system administrator always counts as Administrator, even when not listed anywhere in the app. A delegated *app* administrator is almost as powerful but cannot change the administrator list itself.

1.2 What each role may do

Capability	Admin	Office	Field
Open the app	yes*	yes*	yes*
Read items, locations, movements	yes	yes	yes†
Create / edit / deactivate master data	yes	yes	no
Receive stock and adjust stock	yes	yes	no
Issue stock, transfer stock	yes	yes	yes†
Create, start and close a stocktake	yes	yes	no
Enter counted quantities	yes	yes	yes†
Import CSV and export CSV	yes	yes	no
Print labels	yes	yes	yes
See prices, suppliers and spend fields	yes	yes	no
Create, send and cancel order sheets	yes	yes	no
Send an order above the approval threshold	yes	no	no
Send an order over the remaining monthly budget (inside the emergency allowance)	yes	no	no
Send an order over the emergency allowance	no	no	no
Change settings and licence	yes	no	no

* With an active access restriction, only the allow-list or an administrator can open the app at all. † With location access on, only locations granted to your user — everything else looks as if it does not exist.

1.3 Location access for field users

Office and administrator always see every location. For field users an administrator can additionally switch on **Limit field users to granted locations** (chapter 14). Then a field user only works with the locations that have been explicitly granted to their account — a warehouse picker who only works in “Handbook Main Warehouse” never sees “Handbook Van” balances, dialogs or stocktake lines. A denied location is indistinguishable from a location that does not exist; this is deliberate, so the ACL never leaks which other places your organisation keeps.

1.4 The access restriction (front door)

Independently of roles, an administrator can restrict *who may open InventoryCheck at all*: **Restrict access to allow-listed users and groups** plus the **Allowed users** and **Allowed groups** lists. Anyone outside the list lands on the **Access denied** page (chapter 12). Keep the restriction off until the allow-list is filled — the app itself warns you, because enabling an empty list would lock out everybody, including you.

1.5 Who should read what

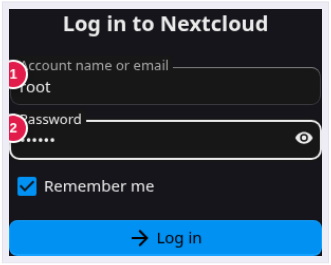
- **Field:** chapters 2–6 and the counting part of 9.
- **Office:** everything field reads, plus chapters 4, 5, 7–8, 10 and 11.
- **Administrator:** the whole manual; your chapters are 12, 13 and 14.

2 Getting started

This chapter takes you from the Nextcloud login page to your first completed booking. It assumes the app is installed; if you are the administrator setting it up for the first time, work through the setup path in section 2.3 before inviting users.

2.1 Sign in and open the app

1. Open your Nextcloud address in the browser.
2. Sign in with your usual username and password.
3. Open **InventoryCheck** from the app menu, or use the direct app URL your administrator gave you.



Nextcloud sign-in — username and password first, then the app menu.

If your organisation gates the app you may land on **Access denied** instead — that means the allow-list is active and your account is not on it (chapter 12).

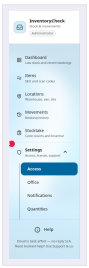
2.2 Find your way around

The sidebar on the left lists your work areas; which entries you see depends on your role. On phones and narrow windows the sidebar hides behind the **Menu** button above the content.

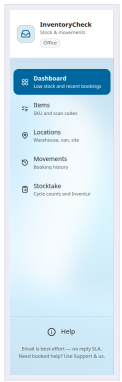
Administrator **Dashboard**, **Items**, **Locations**, **Movements**, **Stocktake**, then **Settings** with its own sub-pages.

Office The same work areas, without **Settings**.

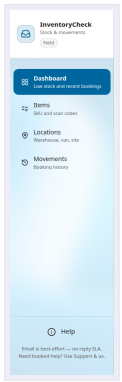
Field **Dashboard**, **Items**, **Locations**, **Movements** — no **Stocktake** menu entry, because starting and closing campaigns is an office task. You still reach an open campaign through its link.



The sidebar as an administrator sees it — work areas on top, settings below.



Office: the work areas, no settings.



Field: daily work only, no stocktake entry.

2.3 First-time setup for administrators

Do these steps in this order — each one builds on the previous:

1. Open **Settings** → **Office** and pick the people who run the store: add them under **Office users** or via an **Office groups** group. Until somebody is on this list, only administrators can receive stock.
2. Stay in **Settings** and open **Access**. Decide whether the app should be open to every Nextcloud user or gated. If you gate it, fill **Allowed users** or **Allowed groups** *first* — only then turn on **Restrict access to allow-listed users and groups**.
3. Open **Locations** and create your first place with **New location**, for example code `DOC-WH`, name “Handbook Main Warehouse”, kind **Warehouse**.
4. Open **Items** and create your first item with **New item**, for example SKU `DOC-FILTER`, name “Handbook Air Filter”.
5. Back on the **Dashboard**, press **Receive stock**, pick the new item and location, enter the quantity and confirm. The balance is now visible everywhere.

That is the whole bootstrap: place, item, first receive. Everything else — transfers, stocktake, exports — builds on these three records.

2.4 Your first complete task

Try the daily-work loop once end to end, so the model clicks: stock arrives at a place, stock leaves a place, and every change is a row in the journal.

1. On the **Dashboard**, choose **Receive stock**.
2. In the dialog, pick item “Handbook Air Filter” and location “Handbook Main Warehouse” — or scan/enter the code with **Find by code**.
3. Enter quantity 25 and confirm. Expected result: the dialog closes, the movement appears under **Movements**, and the dashboard stock figure rises by 25.
4. Now choose **Issue stock**, same item and location, quantity 3, and confirm. Expected result: a second journal row, and the on-hand figure drops to 22.

You have just done the two bookings you will use most. Notice that neither row can be edited afterwards — if the 3 should have been a 2, you fix it with **Reverse** or **Adjust stock**, never by rewriting history (chapter 6).

2.5 The tip cards

The first time you open a page, a short “How it works” card may sit on top — three steps that explain the page. Once you know the page, press **Hide tips** and it stays hidden for your account on every page. The cards do not come back unless a colleague re-enables them; they are per user, not per device.

3 Dashboard — the daily overview

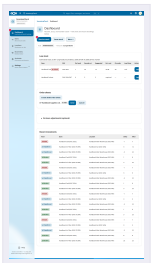
The **Dashboard** is where you land after opening the app. It answers the two questions you ask most often at the start of a shift: *what is running low?* and *what just happened?* — and it puts the action your role needs most directly under your fingers.

3.1 What the page shows

Low stock Every active item whose total on-hand quantity sits below its **Reorder level**. For office this is the full reorder view — suggested quantities, estimated cost, on-order markers, lead times and the order sheets built from it — covered in chapters 7 and 8. If your administrator enabled per-location hints, the same warning is also broken down per place, so you can see that “Handbook Main Warehouse” is fine while “Handbook Van” is about to run dry.

Recent movements The newest journal rows across the places you may see: receives, issues, transfers and adjusts, newest first. One glance tells you whether the last delivery was booked and who booked it.

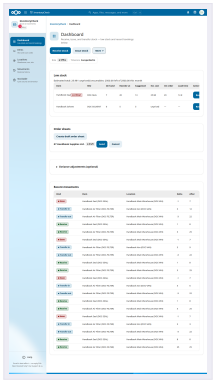
Quick actions Office and administrator get **Receive stock** up front (marker 2 in the screenshot); field users get **Issue stock** instead, with transfer and — where allowed — more actions behind **More**.



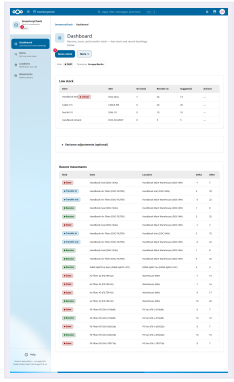
The dashboard as an administrator: marker 1 is the active navigation entry, marker 2 the primary quick action — **Receive stock** for this role.

3.2 The same page in other roles

The dashboard is role-aware. Office sees the storekeeper layout, field users see issue in front. Nothing is hidden that you would need — only reordered.



Office dashboard: marker 1 shows the **Office** badge.



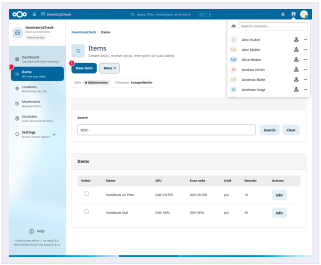
Field dashboard: marker 1 the **Field** badge, marker 2 the **Issue stock** action.

3.3 Negative numbers on purpose

If a balance ever goes negative you will see it flagged here. By default the app refuses issues that would drive a location below zero (*“Not enough stock at this location.”*); an administrator can allow negative stock under **Settings** → **Office** (chapter 14), in which case the dashboard is where the overshoot surfaces — treat it as a to-do list for corrections, not as normal stock.

4 Items — your master data

Under **Items** lives the catalogue of everything you keep in stock. An item is the master record: it carries the identifiers the rest of the app books against, but it does not carry a quantity — stock always belongs to an item *at a location*. All roles may read items; creating, editing and deactivating is office and administrator work.



The item list: marker 1 is the search field, marker 2 the **New item** action.

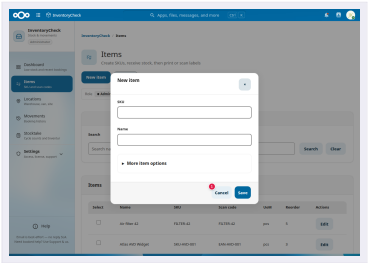
4.1 The list and how to find things

The list shows your items with SKU, name and unit; filters narrow it to active or inactive records. Type into the search field (marker 1) and it matches SKU, scan code and name at once — “DOC-FILTER” and “Handbook Air Filter” lead to the same row. Empty catalogues show *“No items yet”*, a filter with no hits shows *“No items match these filters”*. Opening a row brings you to the item detail with its balances per location, photo and actions.

4.2 Create an item

1. Press **New item** on the **Items** page.
2. Fill **SKU** and **Name** — these are the two fields you always need. The SKU is the human-facing article number; it must be unique.

3. Open **More item options** when you need more than the minimum: **Scan code (optional)** for the barcode on the physical label (may differ from the SKU, must also be unique), **Unit** (defaults to `pcs`), **Reorder level** for low-stock warnings, **Target stock (optional)** and **Default location (optional)**.
4. Choose a **Tracking mode** only if the item needs it: **No lot / serial tracking** for ordinary stock, **Lot / batch tracking** when every receive carries a lot number, **Serial number tracking** when every single piece has its own serial — serial items always move in quantity 1.
5. Office and administrator additionally see the commercial block field users never see: **Supplier note (optional)**, **Last price in cents (optional)**, and the buying fields the reorder engine reads — **Category (optional)**, **Supplier name (optional)**, **Supplier article no. (optional)**, **Lead time in days (optional)**, **Units per order pack (optional)**, **Pack price in cents (optional)** and **Criticality**. What they drive is covered in chapters 7–8; a category outside the configured list is refused with a field error.
6. Save. The item is immediately bookable; there is no separate “activate” step.

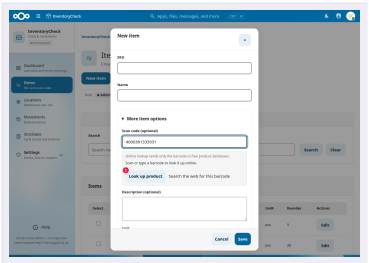


The **New item** dialog: marker 1 is the primary action that stores the record.

Why so few required fields? The app deliberately lets a catalogue grow bottom-up: create the item when the first box arrives, enrich it when someone actually needs reorder levels or a photo. A lean record today beats a perfect record that was never entered.

4.3 Online product lookup (optional)

When an administrator has turned on **Online product lookup** (chapter 14), the **New item** dialog gains a lookup panel under **More item options**, right below the **Scan code** field — office and administrator only. Type or scan a valid EAN/UPC barcode (8, 12, 13 or 14 digits including the check digit) and **Look up product** becomes active. The app then asks free product databases and offers a name and description suggestion.



Marker 1: the **Look up product** button lights up once the barcode is valid.

Suggestions only fill *empty* fields and never save by themselves — you stay the editor. If nothing is found, **Search the web for this barcode** opens a web search in a new tab. Privacy: only the barcode leaves your server, never names, stock or users.

4.4 Edit, deactivate, photo

- **Edit item** reopens the same dialog. Changing SKU or scan code does *not* reprint labels that are already stuck on shelves — print fresh ones after a code change (chapter 10).

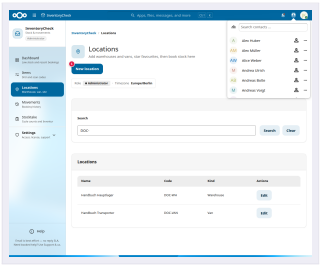
- **Active** is the off-switch, not a delete. Once an item has movement history it cannot be deleted at all — the journal would lose its subject — so you deactivate it instead. Deactivation is also blocked while the item has stock or sits on an open stocktake: clear balances or close the campaign first.
- **Upload photo** attaches a picture (JPEG, PNG or WebP, up to 2 MB) shown on the detail page — helpful when the picker knows the part by sight, not by SKU.

4.5 Edge cases

- *"This code is already in use."* — the SKU or scan code collides with another item. Check the inactive records too; codes stay reserved.
- *"The code format is not valid. Use letters, digits, and . _ / - only."* — the same message on locations; codes reject whitespace and other symbols so they stay printable and scannable.
- Lot- and serial-tracked items do not appear in stocktake campaigns (chapter 9) and always require their tracking data during receive.

5 Locations — where the stock lives

A location is any place that can hold stock: a building, a shelf row, a service van, a site box. Quantities never exist "in the app" — they exist at a location, and every movement names at least one. Creating and changing locations is office/administrator work; every role can read them, though field users may see only the places granted to them (chapter 1).



The location list: marker 1 is the **New location** action.

5.1 Create a location

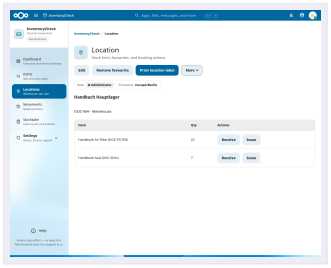
1. Press **New location** on the **Locations** page.
2. Enter a **Code** — the short, scannable identifier that goes on printed labels (for example `DOC-WH`) — and a **Name** people recognise ("Handbook Main Warehouse").
3. Pick a kind: **Warehouse**, **Shelf**, **Van**, **Site** or **Other**. The kind is a label for humans and reports; all kinds behave the same in bookings.
4. Leave **Active** on and save. The location can now receive stock.

5.2 Favourites keep daily work short

Each user can star up to **20** favourite locations (☆ **Add to favourites** on a row or detail page). Favourites sort first wherever you pick a place: in booking dialogs, when starting a stocktake, and on the companion app. Star the places you actually touch daily — a picker who works "Handbook Main Warehouse" and "Handbook Van" should not scroll past thirty site boxes every time. Beyond the limit you get *"You already have the maximum number of favourite locations."* — unstar something first.

5.3 The location detail page

Opening a row shows everything on hand at that place, the booking actions right where they belong (**Receive stock**, **Issue stock** on a balance line), favourites toggle and label printing. For a field user this is the natural work page: open your van, see what is in it, issue what the job consumed.



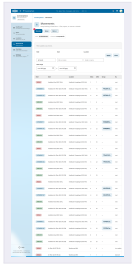
A location detail: stock lines with per-row actions, favourites and labels.

5.4 Deactivate instead of delete

A location with movement history cannot be deleted — the journal must keep naming where stock went. Instead it is deactivated: it stops being offered for new bookings but keeps its history and balances. Deactivation is also refused while the location still holds stock or has an open stocktake; empty it or close the campaign first. If the place truly never saw a booking, delete is offered and removes it outright.

6 Movements — the append-only journal

Every quantity change in InventoryCheck is a *movement*: a single immutable row that says what changed, where, by whom and when. **Movements** is therefore not a list you edit — it is the audit trail of your store. You never rewrite a row; you add the booking that sets things right. That is why there is no edit button anywhere in this chapter: nothing is silently overwritten, ever.



The journal: marker 1 marks the per-row **Reverse** action — the safe way to undo.

6.1 The four movement types

Receive stock Goods arrive: quantity goes up at one location. Office and administrator only — a receive states that something entered your responsibility.

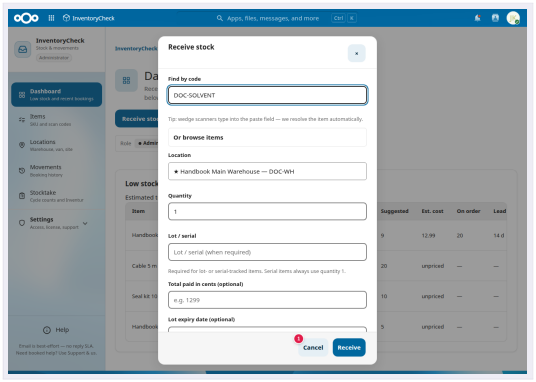
Issue stock Goods leave: quantity goes down at one location. Every role with location access may issue — this is the daily field action (“three air filters went onto the job”).

Transfer stock Goods move between two of your locations. One booking, two journal rows: **Transfer out** at the source, **Transfer in** at the destination, linked as one group. All roles with access to *both* places.

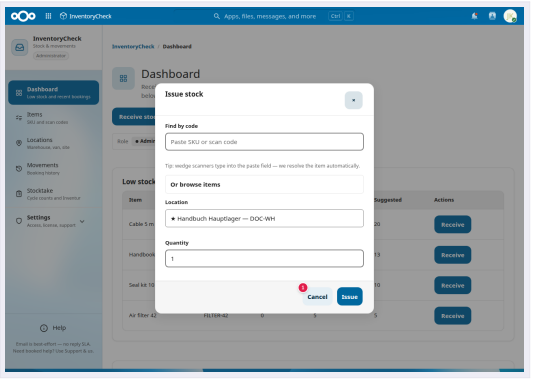
Adjust stock The shelf and the system disagree; office/administrator sets the **New quantity** explicitly. When the policy **Require a reason code on every stock adjust** is on, you must also pick a **Reason code** — adjustments answerable, audits painless.

6.2 Book a receive (office)

1. Open **Receive stock** from the dashboard or a location/item page.
2. Pick item and location, or scan the shelf label with **Find by code**: type or scan the code and the matching record is filled in. No hit gives *"No active item matches this code."*.
3. Enter the quantity (and lot/serial data when the item requires it). Office and administrator additionally may fill **Total paid in cents (optional)** — what the delivery cost, which feeds the last price and the reorder estimates (chapter 7) — and, for lot-tracked items, **Lot expiry date (optional)**. A lot code that was received before with a different expiry aborts the booking: correcting yesterday's paperwork is a reversal plus a re-booking, never a silent overwrite.
4. Confirm. The row lands in the journal and the balance updates immediately; when the item carried an on-order marker or sits on an open order sheet, the receive covers both in the same booking (chapters 7, 8).



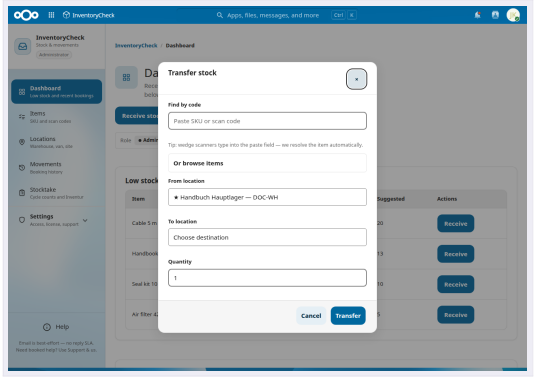
Marker 1: the **Receive stock** dialog on a lot-tracked item — code resolved, lot, spend and expiry fields visible.



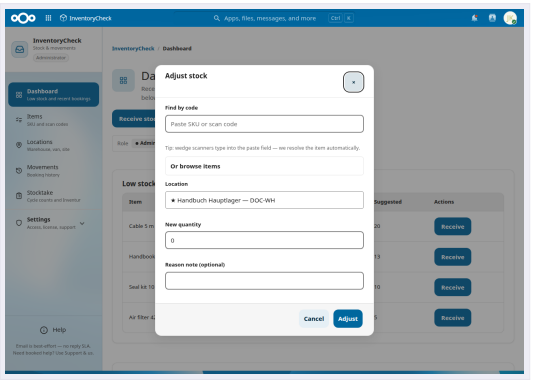
Marker 1: **Issue stock** — the field daily driver.

6.3 Issue and transfer (all roles)

1. Open **Issue stock**, pick item and location, enter the quantity, confirm. If the location holds less than you ask and negative stock is off, the app refuses with *"Not enough stock at this location."* — receive or transfer first, do not force it.
2. For **Transfer stock** additionally pick the destination; source and destination must differ (*"Transfer source and destination must be different."*). After confirming, the journal shows the pair **Transfer out**/**Transfer in**.



The **Transfer stock** dialog: source, destination, quantity.



Adjust stock: set the counted **New quantity**; with the policy on, a reason code is mandatory.

6.4 Undo safely: reverse, never edit

Booked 30 instead of 3? Open **Movements**, find the row, press **Reverse** on it. The app posts a *compensating booking* that mathematically undoes the first one — the journal now shows both, and

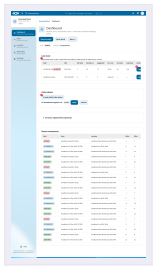
the balance is right. The app says it plainly: *“Reverse posts a compensating booking. History stays complete — nothing is deleted.”* A transfer is reversed from its **Transfer out** row only; the app writes the counter-pair itself. When a row cannot be reversed the action is not shown at all — a muted dash sits where the button would be. If the state moved on after you opened the page, the app still refuses instead of guessing (*“The action conflicts with the current state. Reload and try again.”*).

6.5 Edge cases and limits

- *“The quantity is not valid.”* — zero or a negative number where a positive quantity is required; for adjusts, also when the “new” quantity equals the old one.
- *“The resulting stock would be out of the allowed range.”* — the booking would exceed the storage limits; split it into sane steps.
- *“Please check the highlighted fields.”* — validation failed; the responsible fields are marked in the dialog.
- Lot and serial items always carry their tracking data, and serial items move one piece per booking.

7 Reordering — seeing the need early

The dashboard’s **Low stock** list is more than a red flag — it is the app’s purchase proposal. For every active item InventoryCheck weighs three signals: the static **Reorder level** you set on the item, the speed at which stock actually leaves, and what somebody has already ordered. From that it computes a **Suggested** quantity per row — enough to close the gap to your reorder level (or the optional **Optional target / order-up-to** ceiling), and rounded up to whole **Units per order pack (optional)** when the supplier only sells the item in packs. Administrators can go one step further and let the consumption window feed the suggestion directly: the **Suggestion mode for reorder quantities** select in the restock settings (chapter 14) switches from **Fill up to target stock or reorder level** — the default — to **Cover expected consumption plus lead time**. In coverage mode the app compares “refill to the level” with “what the next lead-time-plus-window days will burn” and proposes the larger amount; **Consumption window for reorder suggestions (days)** sets that window.



The reorder view as office sees it: marker 1 is the estimate line (total, unpriced rows, stale markers, budget state), marker 2 the per-row actions (**Receive**, **Mark ordered**), marker 3 the **Order sheets** block.

7.1 Reading the reorder view

Office and administrator see the full picture; field users get a slimmer list that carries quantities only — no prices, no suppliers, no order sheets.

Suggested How much to buy, after the rounding described above. The **Receive** action on the row opens the receive dialog with exactly this quantity prefilled — one tap less to mistype.

Est. cost What the suggestion is expected to cost, computed from the pack price, falling back to the last price you paid. Rows with no price at all show **unpriced** and are counted separately above the table — a missing price must never silently zero the estimate.

On order The quantity already marked as ordered (see below). A **stale** badge appears once the marker is older than the warning age from settings: the goods were promised but nothing arrived.

Lead time The supplier lead time stored on the item — long-lead items deserve earlier orders.

Badges **critical** pins a row to the top of the list (set **Criticality** on the item when a missing part stops the work); **usage spike** flags an item whose last week of issues tripled its own baseline — an invitation to look before you order, never a verdict.

Above the table a one-line summary keeps you honest: the **Estimated total** across all rows, how many rows are **unpriced**, how many order markers are **stale**, one muted line per budgeted category of the form *"{left} left of {total} this month"* — received deliveries have already eaten into what remains — and a budget badge when an estimate would overshoot the remainder: **Over budget**, or **Over budget — within emergency allowance** when only the emergency headroom still covers it (chapter 14). The estimate itself never blocks you — but the budget decides who may *send* an order sheet (chapter 8).

7.2 Mark ordered — the team memory

When you place an order outside the app — a phone call, a supplier portal — press **Mark ordered** on the row. The item then carries a visible marker: everyone sees a delivery is on its way, so nobody orders the same part twice. When the goods arrive the next receive shrinks or clears the marker automatically — **Clear ordered** removes it by hand when an order falls through. Markers age but never expire on their own; the stale badge is the app's way of asking "should that still be open?"

7.3 Where the warnings reach you

- **In the app:** the reorder view itself — full for office, compact for field.
- **Weekly digest:** a Nextcloud notification summarises low stock, open order markers and lots nearing their expiry and links straight back to this page — one bell, not an e-mail flood. It is on by default for office and administrators; an app administrator can silence it instance-wide via `restockDigestEnabled` on `POST /api/config/restock` — the flag is API-only, no settings field exists.
- **Nextcloud Dashboard:** the **Low stock** widget puts your top shortage items onto the Nextcloud start page; add it from the dashboard's customisation like any other widget.
- **Per location:** with the reorder hint enabled (chapter 14), a second block shows *which* place is short — the warehouse may be fine while the van runs dry.
- **Export:** **More** → **Export reorder CSV** hands the same list to purchasing or a spreadsheet.

7.4 Lots and expiry

Items on **Lot / batch tracking** carry an optional **Lot expiry date** on every receive (office only — chapter 6). The app counts lots that enter the expiry horizon from settings and reports the number in the weekly digest, so perishables surface before they become write-offs. A lot recall — which shelves and vans still hold a given lot — is answered by the companion's recall screen and the API; on the web, the movement rows for an item show each booking's lot code, so the **Movements** journal answers the same question by eye.

7.5 Edge cases

- Nothing short shows *"Nothing is below reorder level."* — the healthiest empty state in the app.

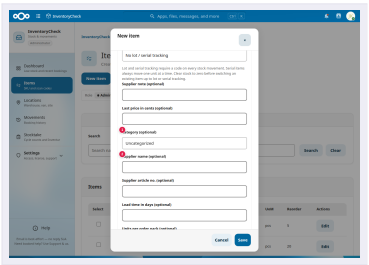
- An item without any price shows **unpriced** and stays out of the estimate total; fill **Pack price in cents (optional)** on the item or let a priced receive set the last price (chapter 6).
- **Mark ordered** proposes the pack-rounded quantity when the item has **Units per order pack (optional)** — edit the marker by clearing and re-marking.
- Ordering twice for the same item is possible but visible: the **On order** column tells you a marker exists before you reach for the phone.

8 Order sheets & suppliers

The reorder view tells you *what* is missing; the order sheet is the shopping list you hand to a supplier. An order sheet is a draft document: it gathers the lines the reorder view proposes, carries the supplier's name, and can be sent — which e-mails the sheet to the supplier's address and marks it *sent* — or cancelled. Once goods arrive, receives automatically cover the open lines, and a sheet whose lines are all received flips to *received* by itself. The lifecycle is therefore draft → sent → received, with *cancelled* as the only way back out.

8.1 Supplier data lives on the item

There is no separate supplier screen in the web app — the item record is where procurement detail belongs. Under **More item options**, office and administrator fill the buying fields (chapter 4): **Supplier name (optional)**, **Supplier article no. (optional)**, **Lead time in days (optional)**, **Units per order pack (optional)** and **Pack price in cents (optional)**. These five drive everything in this chapter: the suggestion rounds to packs, the estimate prices the packs, the lead time column warns you to order earlier, and the order sheet header can carry the supplier name. A **Category (optional)** groups items for the budgets — the list itself is maintained under **Settings** → **Restock** (chapter 14); choosing a value outside the configured list is refused. Field users never see any of this: money and supplier fields are office-side on purpose.



The buying fields under **More item options**: marker 1 **Category**, marker 2 **Supplier name (optional)**.

8.2 Create and send an order sheet

1. Open the dashboard and read the reorder view (chapter 7). If rows are short, office sees the **Order sheets** block below them.
2. Press **Create draft order sheet**. The app drafts one line per short row, quantity equal to the suggestion — the draft appears in the list with its number and supplier name, or **(no supplier)** when none is attached yet.
3. Fits the budget? If not, **Edit** the draft first — change quantities or remove lines, then **Save**; the total re-counts and the send check runs again.
4. Check the draft against the rows, then press **Send**. The sheet flips to *sent* and — when the supplier carries an e-mail address — the app mails a plain-text copy to the supplier. The mail is best-effort: the sheet is sent either way.
5. Wrong draft? **Cancel** retires it — cancelled sheets stay on record but leave the open list.

Three guards keep purchasing honest:

- **Approval threshold.** If an administrator set an **Order approval threshold in cents (empty = no approval)**, a sheet whose total estimate exceeds it can only be sent by an app administrator. Office pressing **Send** above the line gets *"This order needs app-admin approval before sending."* — deliberate, because the spend is material.
- **Monthly budget cycle.** Category budgets reset on the first of the month; deliveries already received have eaten part of them. On **Send** the app weighs the sheet's per-category estimate against what is left. Overrun? Office sees the same *"This order needs app-admin approval before sending."* — only an app administrator may send it, and only as far as the **Emergency budget in cents (empty = none)** allowance reaches; past that ceiling *nobody* can send the sheet. A line in a capped category that has no price counts as overrun too — an unpriced row must never slip past the budget. With no budgets configured, nothing is enforced.
- **Nothing double-books.** A receive covers the marker on the item *and* the open sheet lines in the same booking (chapter 7); a fully covered sheet closes itself as *received*.

8.3 Spend on receive — the price history

The office receive dialog carries two optional money fields (chapter 6): **Total paid in cents (optional)** books what the delivery actually cost, and the item's last price updates with it. That last price is what the reorder view uses when an item has no pack price — real spend data in, better estimates out. The cumulative numbers are available over the API as the valuation and movement-value reports for accounting exports.

8.4 Procedure kits — one booking, many parts

Some issues are always the same basket: a boiler service takes a specific seal, two gaskets and a litre of solvent. A *kit* bundles those lines once; issuing the kit books every line against one location atomically — all-or-nothing, no half-stocked solutions. Kits are created and issued through the API and the mobile companion's kit screen (chapter 13); the journal then shows one issue group, so the audit trail stays honest about which parts went where.

8.5 Beyond the click-UI

Three restock surfaces are deliberately API-only, meant for integrations rather than daily clicking: the *supplier registry* (names plus order e-mail addresses — an order sheet linked to a registry supplier gets mailed on send), the *stock-change webhook* (an app-admin posts a `webhookUrl` on the restock settings endpoint and every booking pushes a `stock.change` event — URLs are server-side validated: HTTPS only, no private or loopback addresses, no credentials, no redirects, best-effort delivery), and the *starter CSV template* endpoint that hands you a ready-made import file (chapter 11).

8.6 Edge cases

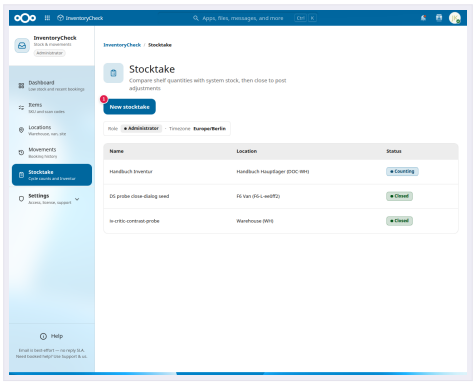
- **Send** with no supplier e-mail simply marks the sheet sent — print or copy it by hand.
- *"This order needs app-admin approval before sending."* — the estimate passed the approval threshold or a category's remaining monthly budget; an app administrator sends or lowers the order (chapter 14).
- A send refused *even for an app administrator* means the sheet overshoots the remaining budget by more than the **Emergency budget in cents (empty = none)** allows — shrink the order or raise the ceiling under **Settings** → **Restock**.
- Cancelling a sheet does not touch on-order markers on the items — clear those separately if the order is really dead (chapter 7).
- A receive that only partly covers a line leaves the rest open; the sheet stays *sent* until every line is covered.

9 Stocktake — counting against the books

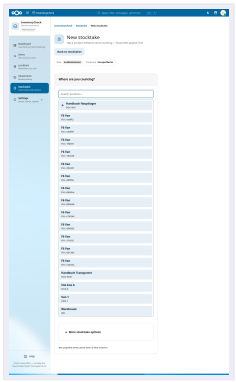
A stocktake answers one question: does the shelf still match the journal? The app freezes a *snapshot* of the system quantity for every active item at one location, people walk the shelves and type real quantities, and closing the campaign posts one **Adjust stock** per counted line so the books equal reality. Creating and closing is office/administrator work; typing counted quantities is open to anyone who may see the location — which is why field has no **Stocktake** menu entry yet can still count inside a campaign they were sent.

9.1 Create and start

1. Open **Stocktake** → **New stocktake**.
2. Pick the location (your favourites sort first) and give the campaign a name — the app suggests a dated name like “Stocktake 2026-11-01”.
3. Confirm. The app snapshots every active item at that location and the campaign opens in *counting* state, ready for quantities.



Marker 1: the stocktake list with the **New stocktake** action.

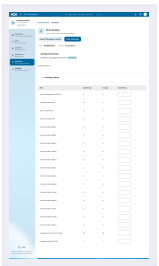


Starting a campaign: location, name, snapshot.

A location can only carry one open stocktake — start a second one and the app points you at the open campaign instead. Lot- and serial-tracked items are excluded by design: a serial unit is verified piece by piece, not counted into a sum, so they never appear as count lines. Very large catalogues have a cap; if the app says *“Too many active items to include in one stocktake. Deactivate unused items or split the count, then try again.”* — tidy the catalogue or count in two goes.

9.2 Count

The campaign page lists every included item with a quantity field. Type what is really on the shelf; each line autosaves when you leave the field, so several people can share the counting safely. With **Blind count — hide system quantities** the expected figure stays hidden — counters cannot “count to the number” and the result is honest. A progress bar shows counted versus open lines.



A campaign in counting state: per-line quantity fields with autosave, progress, and **Close stocktake** for the office role.

9.3 Close — the point of it all

1. When counting is done (or done enough), office/administrator presses **Close stocktake**.
2. Uncounted lines? The app does not silently treat them as zero — it asks. **Close anyway and leave uncounted items unchanged** posts adjustments only for the lines that were actually counted; the untouched lines keep their book value.
3. Confirm. One **Adjust stock** row per counted line lands in the journal — visible under **Movements**, like every other change.

9.4 When the world moved during counting

If stock moved *after* the snapshot — somebody issued five air filters while you were still counting the back shelf — closing with your counted quantity would silently overwrite that later booking. The app refuses to do that quietly: it flags the conflicted lines (*“Stock changed after this stocktake started. Review conflict lines, then close again and confirm you accept the counted quantities.”*) and only proceeds when you explicitly confirm that you reviewed the conflicts. That confirmation is a decision you take, not a speed bump the app forgot.

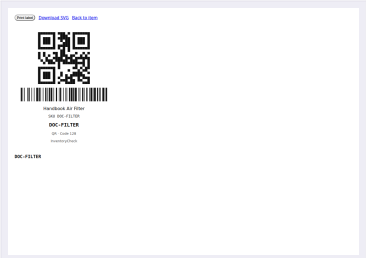
10 Labels and CSV export

Two ways data leaves the app: on paper, as scan labels for shelves and boxes; and as CSV, for spreadsheets, accounting and migration.

10.1 Print scan labels

Every location and every item can carry a printed code — that is what **Find by code** and the scanner companion read at the shelf.

1. For a single record open its detail page and press **Print label** — a print-ready sheet opens in the browser, sized for a standard label.
2. For several records tick the rows in the **Items** or **Locations** list and press **Print labels**: one sheet per record, batched in a single print job.



A label sheet in the browser’s print view — code plus scannable barcode.

Print after creating a place or item, and reprint after renaming a code — the paper never updates itself (chapter 4).

10.2 CSV exports

Exports are office/administrator work and always reflect the filters you see — a filtered movement list exports filtered rows. The **Export CSV** actions sit next to the data:

Items The full catalogue: sku, scan code, name, unit, reorder level, supplier note, price — the same columns the importer reads, so an export is a valid starting point for a round trip.

Locations Codes, names, kinds, active flags.

Balances Item × location quantities — the actual stock-on-hand snapshot for spreadsheets.

Movements The journal rows for the chosen filters (item, location, period): kind, quantity delta, resulting quantity, who, when.

Export DATEV-style CSV Movement quantities shaped for DATEV-flavoured bookkeeping pipelines.

Export reorder CSV The shopping list: every item under its reorder level with the suggested target quantity — hand it to purchasing as-is. Fed back through the importer, its `unit_price_minor` lands on `last_price_minor` while the computed columns are flagged **not imported** (chapter 11) — a round trip loses nothing and invents nothing.

Export variance CSV Counted versus book differences after a stocktake — the audit document for the campaign.

All exports are UTF-8. When your Nextcloud runs in German, headers are emitted in German as well — keep that in mind when you feed the files to tools expecting the English column names (the importer reads both, see chapter 11).

11 CSV import — bootstrapping the catalogue

Import exists for the day you adopt InventoryCheck with three hundred existing SKUs in a spreadsheet, and for bulk updates you would rather not click through. It creates items and can place opening stock — office/administrator only.

11.1 What the file looks like

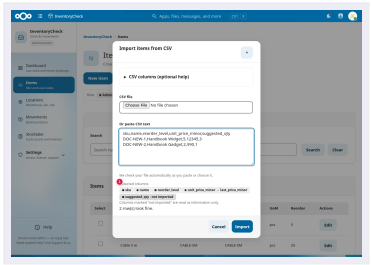
One header line, then one row per item. The recognised columns:

Column	Meaning
`sku`	Unique item number (required).
`scan_code`	Barcode/QR value on the label; unique too.
`name`	Display name (required).
`description`	Free text.
`uom`	Unit of measure, e.g. `pcs`.
`reorder_level`	Threshold for low-stock warnings.
`target_stock`	Order-up-to ceiling; suggestions fill to it instead of the reorder level.
`active`	`1`/`0`.
`supplier_note`	Commercial note, office-visible.
`last_price_minor`	Last price in minor units (cents).
`opening_location_code`	Code of the location for opening stock.
`opening_qty`	Quantity to receive once at that location.
`supplier_name`	Supplier display name.
`supplier_sku`	The supplier's own article number.
`supplier_lead_days`	Delivery lead time in days; feeds reorder suggestions.
`order_pack_qty`	Units per order pack.
`order_pack_price_minor`	Price per pack in minor units (cents).
`category`	Category label; budgets and filters group by it.
`criticality`	`none` or `critical` — pins the item on the low-stock list.

Separator semicolon or comma; encoding UTF-8. The importer also accepts the German header names, so an export made on a German Nextcloud round-trips without edits — and the app's own exports re-import cleanly too: the reorder export's `unit_price_minor` lands on `last_price_minor`, while its computed columns (`suggested_qty`, `est_line_minor`, `packs`, `on_hand`) are listed in **Detected columns** yet flagged **not imported** — read as information, never silently applied. Duplicate codes — two rows with the same SKU, or a SKU that already exists — are rejected, not merged.

11.2 Import step by step

1. Open **Items** → **Import CSV**.
2. Choose the file, or paste the CSV text into the box. You do not press “check”: the app dry-runs your input automatically the moment you paste or pick the file and reports *“ok row(s) look fine.”* plus a numbered list of line problems (*“Line line: message”*). Right above the verdict the wizard shows **Detected columns** as badges — how your headers were mapped, so a mismatched column name (`reorder_level` vs. `reorder`) surfaces before anything is written. Badges carrying a **not imported** flag name columns the app recognises but does not consume — the computed columns of a reorder export, for example.
3. Fix what it flags or leave it: on commit the importer *skips* bad rows and keeps the good ones (*“Import will skip rows with errors and keep the good ones.”*) — you always know in advance how many rows will land.
4. Confirm. Rows with `opening_location_code` + `opening_qty` create the item *and* post the opening receive in one go, so stock history starts correctly instead of with unexplained positive balances.



The **Import CSV** dialog after pasting: marker 1 is the **Detected columns** mapping the dry-run found — check it against your file.

A ready-made starting file exists for integrations and first runs: the dialog’s **CSV columns (optional help)** disclosure links **Download the starter CSV with all import columns** — a commented skeleton carrying every column the importer reads. The same file comes from the starter-template endpoint for scripted setups (chapter 18).

11.3 Practical limits

- *“Choose a file or paste CSV text first.”* — the confirm button stays honest until there is input.
- The importer hard-caps a file at 2,000 rows; anything larger is refused outright before the dry-run — split bigger catalogues into parts.
- The importer never deletes and never updates by stealth: a row that matches an existing SKU is a duplicate and is skipped, so re-running a half-failed file is safe.
- Import creates items and opening balances — not locations, not movements. Create locations first (chapter 5), then import.

12 Team, office and the access door

InventoryCheck knows three concentric circles of permission: the *front door* (who may open the app at all), the *office list* (who runs the store), and *location grants* (which places a field user may touch). All three are set under **Settings** and explained here; the pages themselves are listed field by field in chapter 14.

12.1 The office list — who runs the store

On **Settings** → **Office** you name the storekeepers: individual **Office users**, whole **Office groups**, or both. Membership changes take effect immediately — remove someone and their next **Receive**

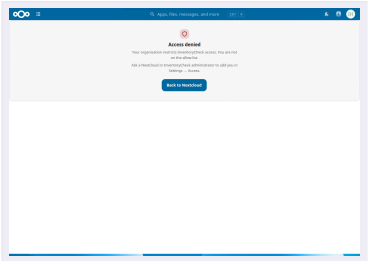
stock is refused with *“You do not have permission for this action.”*. Aim for a small group, not a crowd: every office member can adjust stock, which is the most powerful write in the app.

12.2 The front door — access restriction

On **Settings** → **Access**:

1. Fill **Allowed users** and/or **Allowed groups** with everyone who should reach the app — including yourself.
2. Only then enable **Restrict access to allow-listed users and groups**. The app warns about the empty-list trap (*“Always add at least one user or group to the allowlist before enabling the restriction toggle. Otherwise the app can lock itself — including you — until a system admin re-opens it.”*) and refuses to save an empty allow-list.
3. Optionally name **Delegated app administrators** — colleagues who may change app settings without being Nextcloud system administrators. Only a system admin can edit that list.

Everyone outside the allow-list lands on **Access denied** instead of the app. The page says exactly that (*“Your organisation restricts InventoryCheck access. You are not on the allow-list.”*), offers **Back to Nextcloud**, and deliberately shows no stock data — an outsider learns nothing from it. The fix is always the same: an administrator adds the person (or a group they belong to) under **Settings** → **Access** and the user reloads.

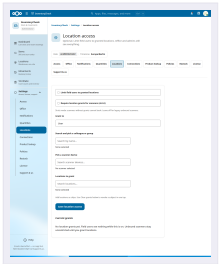


Access denied: polite, empty of data, with a way back.

12.3 Location grants — the inner circle

Office and administrators see every location; for field users the list can be narrowed. On **Settings** → **Locations** (page title **Location access**) an administrator enables **Limit field users to granted locations** and then grants specific locations to specific users. The effect is total for that user: ungranted places disappear from lists, dialogs, favourites and stocktake campaigns — a denied location looks exactly like one that does not exist, so the ACL never reveals what else is stored.

The same page carries **Require location grants for scanners (strict)**: shared scanner devices, which have their own grants, must additionally hold a grant for the location being booked — belt and braces for device fleets that roam between sites (chapter 13).



Location access: the limiter toggle and the grant list.

12.4 Good practice

- Restrict by *group*, not by individual user — a Nextcloud group like “warehouse” survives staff changes without settings edits.

- Keep the office list small and the field grants specific; review both after reorganisations.
- Turn the front-door restriction on only after the allow-list is filled — see step 2 above.

13 The mobile companion — where the licence line is

The web app documented in this manual is free (AGPL) and needs no licence. The optional phone and scanner companions are a separate, paid product — licences have the prefix *IV2* and are managed entirely inside the web app under **Settings** → **License**. This chapter borders the two so you know where the web manual ends.

13.1 What the licence buys

Mobile seats Named phone users of the official companion app. Each seat is assigned to one Nextcloud user; the number of seats comes from your IV2 key.

Scanner device slots Shared warehouse scanners that sign in with a one-time **Pairing code** instead of a personal password — right for devices handed around a shift.

All of it is administered on the web: paste the key, assign seats, create device slots, hand out pairing codes. A seat is freed with **Remove** (toast *"Seat removed."*) and can be re-assigned at any time; a device slot can be **Deactivate**d — which also drops its location grants — and brought back with a new pairing code. **Regenerate code** invalidates the current code immediately and shows the new one exactly once, so copy it while it is on screen.

13.2 Offline behaviour — the part web users see

The companions are built for basements and building sites: when a device loses its connection it keeps usable data locally and queues bookings that need the server. Those queued bookings sync when the connection returns. For web users this matters in exactly one place: the journal. A field booking may appear under **Movements** minutes after it happened on the phone — that is a sync delay, not a duplicate, and the compensating-booking rules in chapter 6 apply to synced rows like any other.

13.3 Scanner policy hooks

Two policies under **Settings** → **Policies** exist specifically for the companions: **Require scanning the location code when booking from the scan app** (the phone must scan the shelf label, not just pick a name) and the location-grant strictness for scanners described in chapter 12. Both aim at the same failure: a tired picker booking the wrong shelf.

13.4 Companion coverage of the restock suite

Three restock surfaces live primarily on the companion: the *kit issue* screen that books a whole basket atomically (chapter 8), the *lot recall* screen that answers where a given lot still sits, and shelf-scan item creation for stock that is not in the catalogue yet. The web API carries the same capabilities; the screens are simply where that work happens.

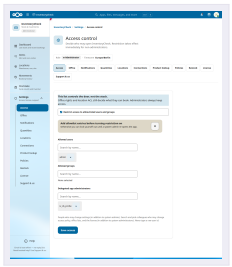
Everything else about the companions — installation, device setup, daily scanning — is outside this manual. Web stays fully usable without IV2; nothing in chapters 2–11 requires a licence.

14 Settings — the administrator reference

The **Settings** area is visible only to administrators (Nextcloud system admins and delegated app admins — see chapter 1). It is organised as one page per topic in the sidebar; the short navigation labels can be shorter than the page titles. This chapter walks the pages in order and explains what each switch *does*, not just what it is called.

14.1 Access

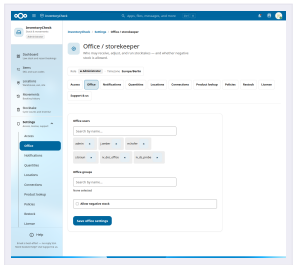
Nav **Access**, page title **Access control** — the front door described in chapter 12: **Restrict access to allow-listed users and groups** (off by default) plus **Allowed users**, **Allowed groups** and **Delegated app administrators**. Saving an empty allow-list with the restriction on is refused — the door would lock everyone out.



Access control: restriction toggle, allow-lists, delegated admins.

14.2 Office

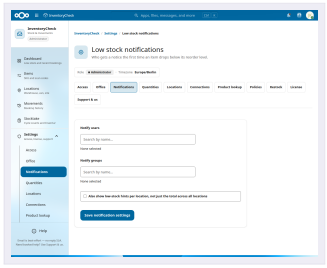
Nav **Office**, title **Office / storekeeper**: **Office users** and **Office groups** define the storekeeper role. Below sits **Allow negative stock** (off by default): on, and issues may drive a balance below zero — useful when goods physically leave before the paperwork is done, at the price of flagged negative lines on the dashboard; off, and the app refuses with *“Not enough stock at this location.”*. The app itself advises: *“Leave it off unless you intentionally book below zero.”*



Office assignment and the negative-stock switch.

14.3 Notifications

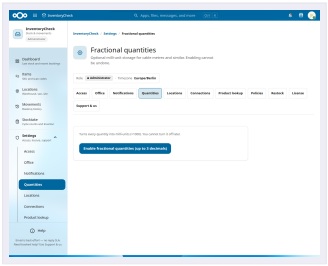
Notifications picks the recipients of low-stock mail. The optional **Also show low-stock hints per location, not just the total across all locations** breaks the warning down per place — the difference between “enough filters overall” and “the van is empty”.



Notification recipients and per-location low-stock hints.

14.4 Quantities

Quantities carries one consequential switch: **Enable fractional quantities (up to 3 decimals)**. Turning it on converts every stored quantity to milli-units, existing numbers are scaled, and the change is **one-way** — the confirm dialog says so verbatim: *"This permanently switches every quantity to milli-units (×1000). You cannot turn it off later. Continue?"* Turn it on before the first booking if your stock comes in metres or kilos; decide once.



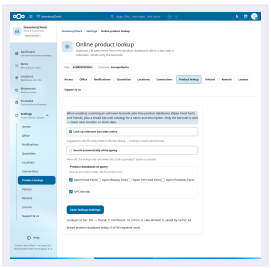
The one-way fractional-quantities switch.

14.5 Location access

Nav **Locations**, title **Location access** — **Limit field users to granted locations**, the grant list, and **Require location grants for scanners (strict)**. Explained in chapter 12; the screenshot is repeated there.

14.6 Product lookup

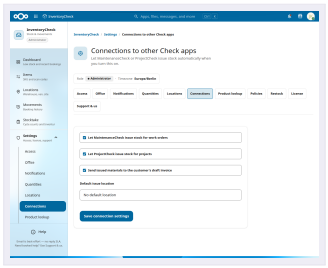
Nav **Product lookup**, title **Online product lookup** — the optional barcode lookup from chapter 4, off by default. **Look up unknown barcodes online** enables the panel; **Search automatically while typing** fires it as you type (otherwise only on the button); **Product databases to query** orders the free sources (Open Food Facts, Open Beauty Facts, Open Pet Food Facts, Open Products Facts, UPCitemdb) — first match wins. Only the barcode is sent, never names, stock or users. A stats line below shows hits, misses, errors, rate-limits and cache saves plus the daily budget of the broad catalogue. **Save lookup settings** applies.



The opt-in lookup page: sources order and the privacy note.

14.7 Connections

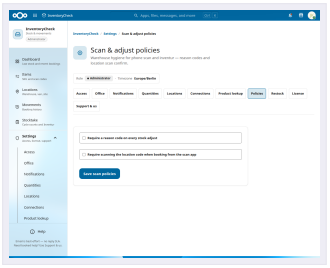
Connections wires InventoryCheck to sibling apps when they are installed: MaintenanceCheck and ProjectCheck may consume stock. **Send issued materials to the customer's draft invoice** mirrors project issues onto the matching InvoiceCheck draft — pushes are queued durably, retried by a background job, and the page shows **Pending invoice pushes** with **Retry now** while any are open. A default location for these automatic issues is chosen here too.



Suite connections, including the InvoiceCheck push.

14.8 Policies

Policies holds the hygiene switches: **Require a reason code on every stock adjust** and **Require scanning the location code when booking from the scan app**. Policies apply to the next booking, never retroactively.

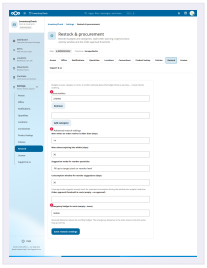


Reason-code and scan policies.

14.9 Restock

Nav **Restock**, title **Restock & procurement** — the procurement knobs from chapters 7–8, editable by app administrators only (everyone else sees the page read-only with *"Only app administrators can change restock settings."*). The page puts the common path first: the categories and their budgets. Each entry feeds the item form's **Category** picker and carries a *monthly* budget in cents — every delivery received this month has already consumed part of it, so the reorder view's estimate line shows what is left per category (*"{left} left of {total} this month"*) and raises the **Over budget** badge when an estimate overshoots the rest, or **Over budget — within emergency allowance** when only the emergency headroom still covers it. The badge warns on the view; the budget itself decides who may *send* an order sheet (chapter 8).

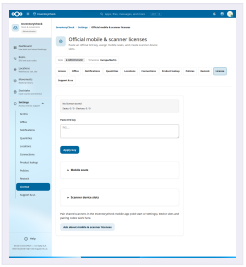
The tuning knobs live one click deeper under **Advanced restock settings**: **Warn when an order marker is older than (days)** sets when a **stale** badge appears on on-order markers, **Warn about expiring lots within (days)** the horizon the weekly digest counts lots against, **Suggestion mode for reorder quantities** whether a suggestion simply refills to the reorder level or also covers the consumption expected during the window plus supplier lead time, **Consumption window for reorder suggestions (days)** that window, **Order approval threshold in cents (empty = no approval)** the spend above which only an app administrator may send an order sheet, and **Emergency budget in cents (empty = none)** the extra an app administrator may still authorize once a category's monthly budget is spent — beyond it nobody sends (chapter 8). **Save restock settings** applies.



The **Restock & procurement** page: marker 1 a category's monthly budget, marker 2 the **Advanced restock settings** fold, marker 3 the **Emergency budget in cents (empty = none)** field inside it.

14.10 License

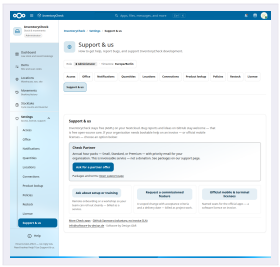
Nav **License**, title **Official mobile & scanner licenses**: paste the IV2 key, then manage **Mobile seats** (**Remove** frees a seat instantly) and **Scanner device slots** (**Regenerate code**, **Deactivate** — both ask first; Escape, × or a click beside the dialog cancels). Covered in chapter 13.



The IV2 licence page: seats and device slots.

14.11 Support & us

Support & us: partner options, sponsoring and commissioned features — the place for invoiceable help and how to fund the app.



Support & us.

15 Accessibility

The web UI targets WCAG 2.1 AA: sufficient contrast against the Nextcloud theme, complete keyboard operation, and semantic markup a screen reader can navigate. This chapter collects what that means in daily use — and what to do when something falls short.

15.1 Skip links and landmarks

Every app page starts with two skip links: **Skip to app navigation** and **Skip to main content**. They are the first things Tab reaches, so a keyboard or screen-reader user can bypass the Nextcloud chrome and the app sidebar. The sidebar carries `role="navigation"` and marks the current page with `aria-current`; the work area lives in `<main id="iv-main-content">`. The **Access denied** page has its own skip link to the denied main area.

15.2 Keyboard operation

Everything in this manual can be done with the keyboard alone:

- Tab and Shift+Tab move between controls in DOM order; the focused element shows a visible `:focus-visible` ring.
- Buttons, links and list-row actions activate with Enter or Space; dialogs open as real `<dialog>` elements, trap focus while open, and close with Escape — focus returns to the control that opened them.
- Touch targets are sized for motor impairments too: at least about 44 px, 48 px for primary and danger actions.

There are deliberately no invented app shortcuts (no “Ctrl+S”): saving and posting always go through visible, labelled buttons, so nothing depends on knowing key combos.

15.3 Live regions and feedback

Async feedback is announced, not just painted: toasts and inline messages use `aria-live` regions (errors assertively, status politely), empty states carry `role="status"`, and the stocktake progress uses `role="progressbar"` with the current count. Validation errors land on the fields themselves — `aria-invalid` plus an inline message per named control — so a screen reader announces *which* field failed and why, matching the *“Please check the highlighted fields.”* summary.

15.4 Contrast, zoom, reduced motion

- Colours come from the Nextcloud theme; the app never hardcodes a hue that breaks in dark or high-contrast themes. `prefers-reduced-motion` and forced-colors media queries are respected in the styles.
- Browser zoom to 200 % keeps every page usable; layouts are mobile-first and reflow instead of scrolling sideways at narrow widths.

15.5 Report a barrier

If you hit something that blocks you — a dialog that steals focus, a state only colour encodes, an unlabelled control — write to `dev@software-by-design.de` with the page and what happened. Accessibility bugs are treated as defects, not suggestions.

16 Errors, messages and FAQ

Every message the app can put in front of you, grouped by where it appears — what it means, why you are seeing it, and what actually fixes it. After that, a “how do I …” index into the rest of the manual.

16.1 Permission and access messages

“You do not have permission for this action.” Your role does not cover what you tried — typically a field user pressing receive or adjust, or someone removed from the office list whose session still shows the button. The role matrix in chapter 1 tells who may do what.

“You are not allowed to use InventoryCheck.” / ***“You are not allowed to use InventoryCheck right now.”*** App access denied at the door — see the next entry.

Access denied / ***“Your organisation restricts InventoryCheck access. You are not on the allow-list.”*** The access restriction is on and your account is not allowed. Nothing is broken; an administrator adds you under **Settings** → **Access** (chapter 12).

An item or location seems to be missing With location access on, ungranted places look exactly like non-existent ones — no error, no trace. Ask an administrator whether a grant is missing before assuming data loss (chapter 12).

16.2 Quantity and stock messages

"Not enough stock at this location." / ***"Not enough stock at this location (available: %s)."*** Issue or transfer exceeds the on-hand quantity and negative stock is off. Book the missing receive or transfer first (chapter 6); office can allow negatives under **Settings** → **Office**.

"The quantity is not valid." Zero, negative, or a no-op adjust (new quantity equals the old one).

"The resulting stock would be out of the allowed range." The booking would exceed storage limits — split it.

"Enter a quantity of zero or more." A count or quantity field was left empty or negative.

16.3 Codes and master data

"This code is already in use." SKU, scan code or location code collides — inactive records still hold their codes.

"No active item matches this code." **Find by code** or a scan found nothing — check the code, check the item is active.

"The code format is not valid. Use letters, digits, and . _ / - only." Illegal characters in a code; keep codes printable.

"Please check the highlighted fields." Validation — the fields that failed are marked individually.

"This item has movement history and cannot be deleted. Deactivate it instead." / ***"This item is on an ope*** Deletion rules protecting the journal (chapter 4); same idea for locations (chapter 5).

16.4 Stocktake messages

"Only office users and app admins can start a stocktake." By design — field counts inside a campaign via its link.

"Stock changed after this stocktake started. …" Conflicting bookings after the snapshot; review and confirm (chapter 9).

"Too many active items to include in one stocktake. …" Catalogue cap — deactivate or split.

"An item on this stocktake was switched to lot or serial tracking. …" Tracked items are excluded from counts; remove the line or revert the tracking mode.

16.5 Empty states and connectivity

"No items yet" / ***"No movements yet"*** / ***"No stocktakes yet"*** / ***"No locations yet"*** Not errors — the empty-state text of a fresh install or a filter with no hits (*"No items match these filters"*).

"No internet connection or the server cannot be reached." The load-error panel: check network and server, reload — nothing was half-saved.

"The action conflicts with the current state. Reload and try again." Someone else changed the same record first; reload and re-apply. The app fails closed rather than overwriting silently.

"You already have the maximum number of favourite locations." Twenty per user — unstar one first.

"This order needs app-admin approval before sending." The sheet's estimate passed the approval threshold or a category's remaining monthly budget; an app administrator sends it or the order shrinks (chapter 8).

Order-sheet send refused for everyone The sheet overshoots the remaining monthly budget by more than the **Emergency budget in cents (empty = none)** allowance — that ceiling binds even app administrators. Shrink the order, or raise the category budget or the allowance under **Settings** → **Restock** (chapter 8).

"Something went wrong. Please try again." The safety net: a failure carried no message of its own, so the app caught it and said so instead of dying silently. Retry once; if it repeats, note what you were doing and report it via **Help** (chapter 16).

"Only app administrators can change restock settings." Expected — budgets, warning windows and the approval threshold are admin-only (chapter 14).

"No open order sheets." Not an error — the order-sheet block's empty state; create a draft first (chapter 8).

"Nothing is below reorder level." Not an error — every item is stocked above its level (chapter 7).

16.6 How do I ...?

...receive my first delivery? chapter 2 (first task) and 6.

...fix a wrong booking? **Reverse** on the journal row — chapter 6.

...move stock to a van? **Transfer stock** — chapter 6.

...count the shelf? **Stocktake** — chapter 9; field users count via the campaign link.

...find out why an item won't deactivate? chapter 4 (stock, history, open stocktake).

...give a colleague app access? allow-list and office list — chapter 12.

...limit a field user to their van? location grants — chapter 12.

...get the SKU list into a spreadsheet? chapter 11 in, chapter 10 out.

...understand mobile seats? chapter 13.

...reorder what is running out? the reorder view — chapter 7.

...record that I already ordered? **Mark ordered** — chapter 7.

...send the shopping list to a supplier? order sheets — chapter 8.

...keep big orders under control? the approval threshold — chapters 8 and 14.

16.7 Frequently asked questions

Why is there no edit button on a movement? The journal is append-only so history can be trusted; **Reverse** writes the truth forward (chapter 6).

Why does field not see Stocktake? Campaigns are office-run; counters get a link. Roles: chapter 1.

Why can't I switch fractional quantities off? The storage format changes permanently — the warning is literal (chapter 14).

Why does a denied location show no "forbidden" hint? ACL denial deliberately looks like "does not exist" (chapter 12).

Do I need a licence for the web app? No — IV2 covers companion seats and scanners only (chapter 13).

Why did my import skip rows? Bad lines are reported and skipped, good ones kept — the dry-run told you beforehand (chapter 11).

17 Glossary

Domain terms a new user meets in the app and in this manual — *not* button labels (those are always shown the way the UI spells them, in quotes).

Allow-list The list of users and groups who may open the app when the access restriction is on (chapter 12).

Approval threshold The spend above which only an app administrator may send an order sheet — the four-eyes rule for purchasing (chapter 8).

Balance The current on-hand quantity of one item at one location. Stock never exists "globally" — always per place.

Blind count A stocktake mode that hides the expected system quantity so counters report what they see, not what they expect.

Compensating booking The journal row **Reverse** writes to undo an earlier row — the audit-safe "delete".

Budget A monthly per-category spending limit for reorder estimates; received deliveries consume it, overshooting what is left warns on the reorder view and gates who may send an order sheet.

Emergency allowance The extra amount (**Emergency budget in cents (empty = none)**) an app administrator may authorize on top of a spent monthly budget — beyond it nobody sends.

Criticality The item flag that pins a shortage to the top of the reorder view — for parts whose absence stops the work.

Journal The append-only movement history: every receive, issue, transfer, adjust and reversal, forever.

Kit A named bundle of items issued together in one atomic booking — the fixed basket for a recurring job.

Location grant Permission for a field user (or scanner) to work with one specific location.

Lot / batch A tracking mode where every receive carries a lot identifier — for goods made or bought in batches.

Milli-units The storage format after enabling fractional quantities: every quantity times 1000, so 1.5 l and 250 g are exact.

Movement One immutable journal row: kind, item, location, quantity delta, resulting quantity, actor, time.

Negative stock A balance below zero — allowed only if the office setting permits it; otherwise blocked at booking time.

On-order marker The flag set by **Mark ordered** so the team sees a purchase is running; receiving the goods clears it.

Opening balance The first receive of an imported item — created by the CSV import when `opening_location_code` and `opening_qty` are set.

Pairing code The one-time secret a shared scanner uses to sign in to its device slot — no personal password involved.

Order sheet The draft shopping list built from low stock: sent to a supplier, covered line by line as goods arrive.

Reorder level The per-item threshold; below it the dashboard warns and the reorder CSV lists the item.

Scan code The barcode/QR value printed on labels; identifies an item or location to **Find by code** and the scanner companion. May differ from the SKU.

Seat A named mobile licence slot for one phone user of the companion app.

Serial number A tracking mode where every single piece is identified; serial items always move in quantity 1.

SKU “Stock keeping unit” — your article number, unique across the catalogue.

Snapshot The frozen system quantities a stocktake compares reality against; conflicts arise when bookings post after the snapshot.

Stocktake / cycle count The campaign that counts a location’s shelf against the books and closes with adjustments.

Suggested quantity The amount the reorder view proposes — the gap to the reorder level or target stock, pack-rounded.

Usage spike The badge on a row whose last week of issues tripled its baseline — a hint to look before ordering.

Velocity / consumption window The number of past days the consumption-based suggestion looks at.

Transfer pair The two linked journal rows a single transfer writes: **Transfer out** at the source, **Transfer in** at the destination.

18 Surface catalogue

Every surface in the app, mapped to the chapter that explains it — the lookup table when you met a screen this manual does not show.

Surface / function	Chapter
Nextcloud sign-in	2
App menu, sidebar, **Menu** on phones	2
Role badges Administrator / Office / Field	1
Dashboard: low stock, recent movements, quick actions	3
Tip cards, **Hide tips**	2, 3
Items list, search, filters	4
New item dialog incl. tracking modes	4
Online product lookup panel	4, 14
Item detail: balances, photo, edit, deactivate	4
Locations list and kinds	5
Favourites (max. 20 per user)	5
Location detail: stock lines, actions	5
Receive stock dialog	6, 2
Issue stock dialog	6
Transfer stock dialog	6
Adjust stock dialog, reason codes	6, 14
Find by code in booking dialogs	6
Movements journal and filters	6
Reverse / compensating booking	6
Stocktake list and **New stocktake**	9
Campaign counting page, blind mode, autosave	9
Close stocktake, uncounted lines, conflicts	9
Print label / **Print labels**	10
CSV exports (items, locations, balances, movements)	10

Surface / function	Chapter
DATEV, reorder, variance exports	10
Import CSV dialog, dry-run report	11
Settings → **Access** (allow-list, delegated admins)	12, 14
Settings → **Office** (users, groups, negative stock)	12, 14
Settings → **Notifications**	14
Settings → **Quantities** (fractional, one-way)	14
Settings → **Locations** (field grants, scanner strict)	12, 14
Settings → **Product lookup**	14
Settings → **Connections** (suite, invoice push)	14
Settings → **Policies**	14
Reorder view (suggested, est. cost, on-order, lead time, badges)	7
Mark ordered / **Clear ordered** markers	7
Order sheets: draft, send, cancel	8
Supplier and pack fields on items	4, 8
Spend and lot-expiry fields on receive	6, 8
Weekly digest notification, **Low stock** widget	7
Kits, supplier registry, webhook, reports (API)	8
Settings → **Restock** (categories + monthly budgets; **Advanced restock settings**: windows, suggestion mode, threshold, emergency budget)	14, 7
Settings → **License** (IV2, seats, slots)	13, 14
Settings → **Support & us**	14
Access denied page	12
Empty states and error messages	16
Keyboard, screen reader, WCAG	15

Mobile companion screens are intentionally outside this manual — chapter 13 borders what the web administrator manages and where the companion's own documentation begins.

About Software by Design

Software by Design builds Nextcloud apps for self-hosted data — time, projects, stock, fleet, CRM. Web UIs stay free (AGPL) on your instance.

Website: https://nextcloud.software-by-design.de/ **Partner:** mailto:info@software-by-design.de **Bugs/ideas:** mailto:dev@software-by-design.de

Support & contribute

InventoryCheck stays AGPL-free. Under **Settings**→**Support & us**: partner offer, setup, commissioned features, sponsoring.

Check Partner: https://nextcloud.software-by-design.de/support/ **Sponsors:** https://github.com/sponsors/aSoftwareByDesignRepository

No legal/tax/compliance advice. Third-party marks belong to their owners.

Index

Access, 17
Access denied, 17
Access restriction, 2
Accessibility, 23
Adjust, 8
Administrator, 1
Allow-list, 18
API, 13
Approval, 12
Approval threshold, 22
aria-live, 24
Blind count, 14
Budget, 22
Catalogue, 28
Clear ordered, 11
Close stocktake, 15
Companion, 19
Compensating booking, 9
CSV columns, 16
CSV export, 15
CSV import, 16
Cycle count, 14
Dashboard, 4
Deactivate item, 6
Deactivate location, 8
Errors, 24
Export, 15
FAQ, 24
Favourites, 7
Field, 1
First steps, 2
Focus, 24
Fractional quantities, 21
Glossary, 27
How do I, 26
Import, 16
Initial setup, 3
InvoiceCheck, 22
Issue, 8
Item photo, 6
Items, 5
IV2, 19, 23
Journal, 8
Keyboard, 23
Kit, 13, 19
Labels, 15
Location access, 2, 18
Location scan policy, 19
Locations, 7
Lot expiry, 11
Lot recall, 19
Low stock, 10
Mark ordered, 11
Mobile licence, 19
Mobile seats, 19
Movements, 8
Navigation, 3
Negative stock, 20
New item, 5
New location, 7
New stocktake, 14
Office, 1
Offline queue, 19
On-order marker, 10
Online product lookup, 6
Order sheet, 12
Overview, 4
Permission matrix, 1
Permissions, 1
Print label, 15
Receive, 8
Reorder, 10
Reorder view, 10
Reverse, 9
Roles, 1
Scan code, 5
Scan label, 15
Scanner strict, 18
Settings, 20
  Access, 20
  Connections, 22
  License, 23
  Location access, 21
  Notifications, 20
  Office, 20
  Policies, 22
  Product lookup, 21
  Restock, 22
  Support, 23
Sign-in, 2
Skip link, 23
SKU, 5
Spend, 13
Stocktake, 14
Stocktake conflicts, 15
Supplier, 12
Supplier fields, 12
Team, 17
Transfer, 8
Troubleshooting, 24
WCAG, 23
Webhook, 13