Software by Design
Nextcloud apps

Maintenance · Visits · Proof packs

MaintenanceCheck

User manual

End-user documentation for the web app

Due board, work orders and register on your Nextcloud.

App version	Language	Date
1.3.4	English	25 October 2026

Contents

Introduction i
1 Roles and permissions 1
2 Managing the team 2
3 Getting started 3
4 Due board and visits 4
5 Work orders 6
6 Planning: dispatch, tours, exceptions, KPI 8
7 Register and catalogs 10
8 Bulk import and export 12
9 The mobile companion 13
10 Settings and integrations 14
11 When access is denied 16
12 Troubleshooting and FAQ 17
13 Accessibility 19
14 Glossary 20
15 Surface index 21
Shortcuts by role 22
About Software by Design 23
Support & contributing 23
Index 23

Introduction

MaintenanceCheck is the Nextcloud app for recurring maintenance on your instance: customers and equipment in the field register, due dates and visits, work orders with checklist, photos and signature, plus service-report or inspection-proof PDFs. Web free (AGPL); the optional companion needs a mobile licence (MN2) and seats — bordered here only.

As of: App version **1.3.4** (commit `5dce86e`), Companion **1.2.5**, 3 November 2026, English (professional).

Atlas farm POLICY 3.5.17 `exit_met`; for this edition screenshots were taken on 1.3.3–1.3.4 and wording aligned with the locale catalogs; verified 25 Oct against `941a98f` (backend-only delta: upgrade backup, reminder-job lock contract, quick-start header), 30 Oct against `c5f52b7` (css-only word-wrap rule inside the hidden Help menu), 2 Nov against `2e14b390` (pack/mark l10n wrong-sense fixes — quoted UI strings re-cited byte-exact, affected shots re-captured) and 3 Nov against `5dce86e` (error-ink required marks, feedback-footer re-render, backend services — docs-invisible).

Audience: Technicians · Office/dispatch · App/Nextcloud administrators.

What you need: MaintenanceCheck access · browser with JavaScript · app-admin for settings · office role for register/planning.

Chapters: Ch. 1 roles · 2 team · 3 getting started · 4–5 daily work · 6 planning · 7 register/catalogs · 8 bulk import · 9 companion · 10 admin · 11 access denied · 12 errors/FAQ · 13 accessibility · 14 glossary · 15 index.

Register: Body text uses professional second-person English. UI labels in quotes, e.g. **Due board**, **Work orders**, **Planning** and **No access to MaintenanceCheck**.
Quick start (office): Catalogs → customers/equipment/plans → work order from due → assign/tour.
Quick start (technician): **Due board** or **Work orders** → checklist/photos → complete visit → **My tour**.

1 Roles and permissions

MaintenanceCheck does not invent its own user accounts. Who you are comes from Nextcloud, and what you may do is decided by three layered signals: whether you are allowed through the app's access door at all, whether the office list names you, and whether you are a Nextcloud or app administrator. Understanding these three signals explains every screen the app will ever show you — and every one it refuses to show.

1.1 The three roles

Administrator A Nextcloud system administrator, or a user listed as app administrator under **Settings** → **Access**. Administrators see and change everything, including the access restriction, the office list, work policies and the mobile licence. In the demo instance this is Lena Wartung; her badge reads **Administrator**.

Office Dispatchers, service coordinators and back office staff. Membership comes from the **Roles** settings page, where the administrator names users or whole Nextcloud groups — system administrators count as office automatically. Office members write the register, dispatch jobs, plan tours, read KPIs and clear exceptions. Olaf Neumann carries this role in the demo data.

Technician The default for everyone else who is allowed into the app. Technicians see their assigned jobs first, can claim pool work, run checklists, attach photos and complete or skip visits. They never see the **Planning** group or **Settings**. Tim Nolte works in this role in the demo data.

1.2 The access door

On top of the three roles sits an optional door: the administrator can restrict the whole app to a named allow-list of users and groups (Chapter 10). The door *fails closed*: a user who is not on the list does not merely lose write access — the MaintenanceCheck entry disappears from the Nextcloud navigation, and a bookmarked link lands on a dedicated page that says only **No access to MaintenanceCheck** (Chapter 11). Dorothea Nolte demonstrates this state in the demo captures.

1.3 Who may do what

Capability	Admin	Office	Technician
Open the app	yes*	yes*	yes*
Read due board, visits, work orders	yes	yes	yes
Dispatch, day tours, exceptions, KPI	yes	yes	no
Own day tour (**My tour**)	yes	yes	yes
Read the register (customers, equipment, catalogs)	yes	yes	yes
Write the register	yes	yes	no
Complete or skip an open visit	yes	yes	yes
Reschedule, assign or cancel a visit	yes	yes	no
Run a work order (checklist, photos, signature)	yes	yes	assigned, helper or pool†
Settings and licence	yes	no	no

* With the restriction on, only allow-listed users plus administrators. † A technician opens a work order when they are the assigned person, a listed helper, or the job is still unassigned in the pool.

1.4 Rules that surprise new users

Three product rules are worth learning on day one, because they are deliberately generous where other apps are strict. First, the register is readable by *every* app user — technicians look up sites, access notes and equipment details in the field. Second, any app user may *complete or skip* any open visit, not only their own: a visit is a scheduled obligation, and a colleague covering a shift must not be blocked by ownership. Third, work orders are stricter than visits — they belong to the assigned person, the helpers and the office, until they sit unassigned in the pool.

2 Managing the team

Roles decide what a person *may* do (chapter 1); the team layer decides who *should* do it. This chapter gathers the operational tasks around people: keeping skills accurate, staffing orders well, and handing out mobile seats deliberately.

2.1 Skills: matching work to qualifications

A work order can demand a skill code from the skills catalog — an electrical qualification, a lift-inspection license, a high-voltage authorization. The dispatcher never guesses: when you assign an order that requires skills, the app compares the requirement against the assignee's recorded skills and answers visibly.

1. Open the skills catalog under **Catalogs** and keep one code per real qualification, named the way the certificate reads.
2. When a technician earns a new qualification, record it there before dispatching the matching work — the check at assignment time reads this list, nothing else.
3. At assignment, read the warning: under a warn-level policy **Assign anyway** documents a conscious exception; under a strict policy the app refuses (chapter 10).

Skills are deliberately informational where the policy allows it: a warning that the office acknowledges beats a hard block when a senior supervises a junior on-site.

2.2 Helpers: two names on one job

Some work is simply a two-person job. **Helpers** adds extra people to an order alongside the primary technician without splitting ownership: the order still belongs to its assignee, but helpers share the same open gates — they can execute the checklist, attach photos and close the visit. Use helpers for the heavy lift or the trainee shadowing a senior, not for the colleague who only drove the van.

2.3 The office list and mobile seats

Two further memberships live outside the role matrix. The office list under **Roles** promotes users or groups to dispatch rights — named explicitly, so a departing coordinator is removed in one place (chapter 10). Mobile seats under **License & mobile** are a different currency entirely: each named seat is a license unit for the phone companion, and the page warns you before a full license rejects the next assignment (chapter 9).

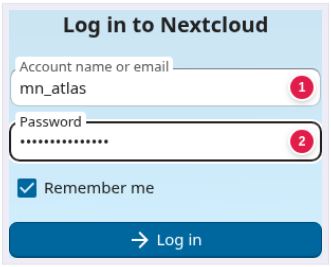
The fixture shows the intended shape: Lena Wartung carries administration, Olaf Neumann sits on the office list, and Tim Nolte works as a technician with his own day tour — three people, three memberships, no overlap required.

3 Getting started

This chapter walks you from the sign-in form to your first finished piece of work. Read the part that matches your role: office staff set the register up once, technicians can skip ahead to their first visit.

3.1 Sign in and find the app

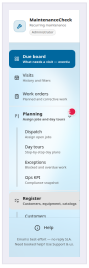
1. Open your Nextcloud address in the browser and sign in with your username (1) and password (2).
2. In the Nextcloud app navigation choose **MaintenanceCheck**. The app opens on the **Due board**.
3. If the entry is missing, you are not on the access list — ask an administrator (Chapter 11 explains the door).



Nextcloud sign-in: marker 1 username, marker 2 password.

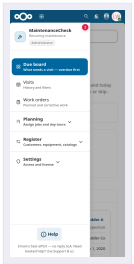
3.2 Navigation by role

The left sidebar is your map, and it already filters itself by role. Everyone sees **Due board**, **Visits** and **Work orders** plus the **Register** group with **Customers**, **Equipment** and **Catalogs**. Office members and administrators additionally get the **Planning** group (**Dispatch**, **Day tours**, **Exceptions**, **Ops KPI**); technicians see a flat **My tour** entry instead. Only administrators see **Settings**.



The administrator's sidebar; marker 1 marks a group toggle that opens its sub-entries.

On narrow screens below 1024 px the sidebar steps aside: tap **Menu** above the content to open the same navigation as a drawer (marker 1 in the shot below).



Narrow view: **Menu** opens the navigation drawer.

3.3 First run for the office

The register is the foundation — visits and work orders hang on customers, equipment and plans. Do this once, in this order:

1. Open **Catalogs** and check that at least one equipment type and one maintenance type exist; create them if the list is empty (Chapter 7).
2. Open **Customers** and create your first customer, for example *Demo RheinMain Facility Service GmbH*.
3. Open the customer and add a *site* such as *Zentrale Opernplatz* when the customer has more than one address.
4. Open **Equipment**, create a unit like *Klima Zentralgerät K-01*, and link it to the customer and site.
5. On the equipment detail page add a maintenance plan — interval plus first due date, or a meter threshold.
6. The first visit appears on the **Due board** automatically; no extra scheduling step is needed.
7. Optionally create a work order from the visit, assign Tim Nolte as the technician and place the job on today's day tour (Chapter 6).

3.4 First visit for a technician

1. Open the **Due board**; your assigned visits and jobs sort before the pool, and overdue items stay on top.
2. Open the visit or the linked work order and work the checklist, adding photos where evidence matters (Chapter 5).
3. Finish with **Complete visit** — or **Skip with reason** when the site was not reachable. The next due date rolls forward automatically.
4. Check **My tour** for the stops the office planned for you today.

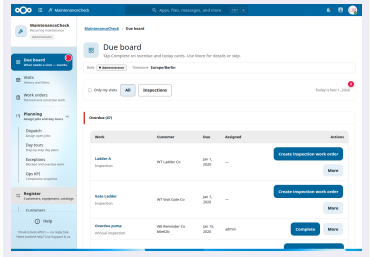
The board teaches you in place: when it is empty it says *Nothing due in the next 30 days*, and new visits announce themselves as *New visits appear here as soon as a maintenance plan makes them due.*

4 Due board and visits

A maintenance plan attached to a unit of equipment creates visits — scheduled obligations with a due date. The **Due board** is the front door of the app: it collects everything that needs a visit, overdue first, so the morning question "what burns today?" answers itself. Visits are deliberately easy to finish: any app user may complete or skip any open visit, because a colleague covering a shift must never be stuck behind an ownership check.

4.1 Reading the board

The board groups open visits into urgency lanes — **Overdue** on top, then **Due today** and the later windows. Each card names the equipment, the customer site and the plan that produced the visit, so a card like *Brandmeldezentrale BMZ-2 at Zentrale Opernplatz* reads as a concrete errand, not an abstract row. Marker 1 in the shot below shows the active board entry in the sidebar; marker 2 marks the board's own filter row.



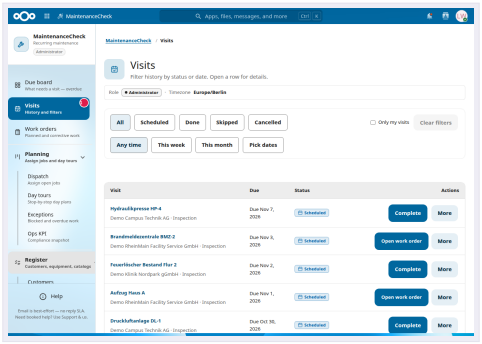
Due board as administrator: marker 1 the active nav entry, marker 2 the filter row.

Use the board filter when the day is crowded: *Focus the board on visits assigned to you.* narrows the lanes to your own work.

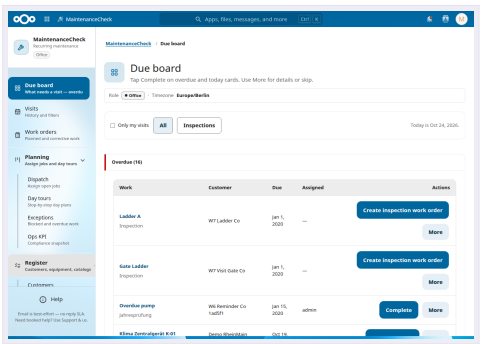
4.2 Completing and skipping a visit

1. Open the visit card on the board or in the **Visits** list.
2. Choose **Complete visit** when the work happened — the app stamps the completion and rolls the plan forward: *Visit completed — next due {date}*.
3. Choose **Skip** or **Skip with reason** when the site was not reachable; a skip with a reason of at least ten characters (*At least 10 characters when skipping.*) keeps the audit trail honest.
4. Skipped visits still roll the plan: *Visit skipped — next due {date}*.

Completed, skipped and cancelled visits never disappear — they stay in the **Visits** list for audit, filterable by status, due-date range and ownership (*Narrow visits by status, due date, or ownership.*). The list reacts immediately; there is no apply button.



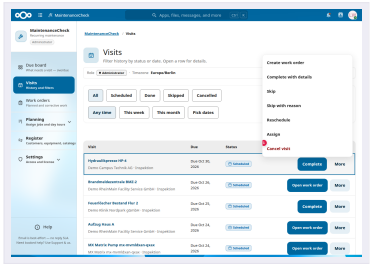
The **Visits** list with filters; marker 1 the active entry.



The same board as office member Olaf Neumann sees it.

4.3 The More menu on a visit

Every visit card bundles its secondary actions under **More**: **Complete with details**, **Skip** and **Skip with reason**. When the visit already carries a work order you also see the deep links **Log hours** into ProjectCheck and **Record time** into ArbeitszeitCheck — both only where those apps are installed. Office members and administrators additionally get **Reschedule**, **Assign** and the red-inked **Cancel visit** (marker 1).



Visit card with the **More** menu open; marker 1 marks **Cancel visit** in danger ink.

The menu opens as an overlay on the card; near the lower screen edge it flips upward and scrolls, so the danger action is never clipped away.

4.4 Office actions on a visit

Reschedule visit moves the due date. The dialog previews the new date (*New due date would be {date}.*) and rejects nonsense: *The due date must be between 2000-01-01 and ten years from today.*

Assign visit names a technician; the card then reads *Visit assigned to {user}.* and lands in that person's own list.

Cancel visit ends the visit *without a result*. The confirm dialog spells out the consequence — *This ends the visit without a result. No follow-up visit is created — you can schedule one manually on the plan later.* — and you either confirm with **Cancel visit** or back out with **Keep visit**. When you still want the cycle to continue, schedule a fresh visit on the plan (Chapter 7).

4.5 Edge cases and limits

Already closed Repeating an action on a finished visit is refused: *This visit was already closed.*

One work order per visit When a visit already carries a job, the app says *This visit already has a work order.* or *This visit already has an open work order.* — open the existing one instead of creating a duplicate.

No open visit A plan whose visit was cancelled shows *No open visit — schedule one to continue the cycle.* on the equipment page; **Schedule visit** starts the next occurrence manually.

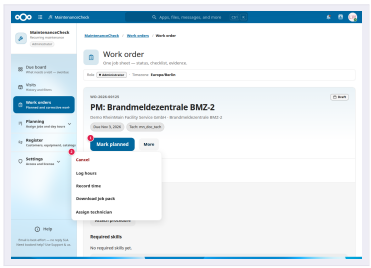
Empty states *Nothing due in the next 30 days*, *No visits yet* and *No visits match these filters* are informational, not errors.

5 Work orders

A visit is an obligation; a work order is the execution. The work order bundles everything a job needs on site — the checklist from its procedure, the kit to pack, photos and a signature as evidence, comments for the record and the PDFs that prove the work afterwards. Three kinds exist: **Preventive work order** (from a maintenance plan), **Corrective work order** (a repair, like the demo order *Klima K-01 — ungewöhnliche Vibration*) and **Inspection work order** (a formal check with a result).

5.1 Status chain

Jobs walk a straight line: **Draft** → **Planned** → **Ready** → **In progress** → **Done**, with **Blocked** and **Cancelled** as side paths. The header always offers *one* large primary action for the next legal step (marker 1 below — **Start work**, **Mark ready** or **Complete**, depending on where the job stands). Every other transition and helper sits under **More** (marker 2), including the red-inked **Cancel**.



Work-order header: marker 1 the primary action, marker 2 the open **More** menu.

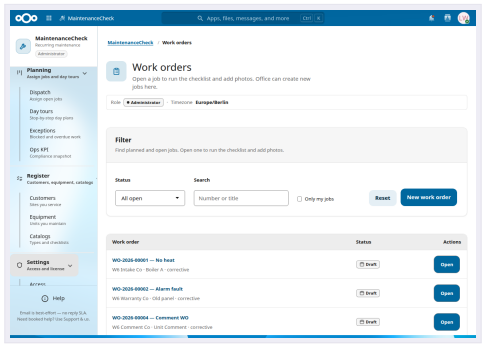
Once a job is **Done** the menu changes character: the transitions disappear and **Download service report** — and for inspections **Download inspection evidence** — take their place. **Download job pack** is available while the job is still open: *Job pack PDF* is the field sheet with status, checklist and evidence space.

5.2 Assignment, helpers and the pool

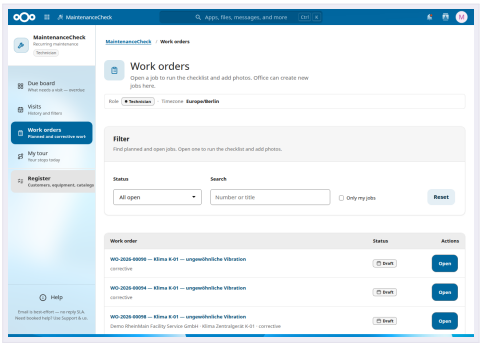
The office assigns one primary technician (**Assign technician**) and any number of **Helpers**. A job nobody owns sits in the pool, where any technician may claim it — that is why the technician's **Work orders** list always sorts assigned work before pool work. Jobs assigned to someone else stay locked for other technicians; the office sees everything.

Two guards watch the assignment. If the job demands skills the person lacks, the app warns *The assignee is missing required skills. Confirm to assign anyway.* — or blocks outright when the policy

says so (Chapter 10). If the day is already full, **Over capacity** fires the same way, based on the minutes-per-day budget. **Assign anyway** is the deliberate override, never a silent slip.



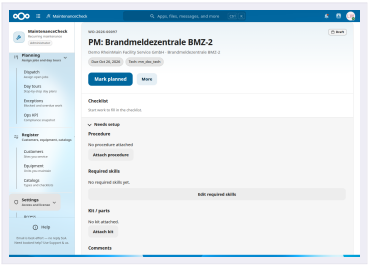
The work-order list for office and administrator.



Tim Nolte's technician list: assigned first, pool after.

5.3 Running the job

1. Open the work order and press the primary action — **Start work** moves a ready job to **In progress**.
2. Work the checklist top to bottom; required items must pass before **Done** is accepted.
3. Pack the kit if one is attached: the job shows *Packed {packed} of {required}* until every line is checked off.
4. Attach photos with **Add photo** — *Evidence for this job — clear and close-up.* Mark damage shots as **Defect photo** where the checklist asks for them.
5. Let the customer or yourself sign with **Draw signature** and **Save signature**; saving over an existing signature replaces it (*A signature is already on file. Saving replaces it.*).
6. Record what the next reader needs under **Comments** with **Post comment** — comments are append-only (*Append-only — comments cannot be edited or deleted.*), so write for the audit, not for the moment.
7. Press **Complete** to finish.



Work-order detail: checklist, kit, photos and actions on one page.

5.4 When Done is refused

The app protects the record: a job with open required checklist items answers *Checklist blocks completion: {items}* and names the blockers. An unpacked kit reports *The kit is not fully packed yet.* When the office knows better — the missing screw is documented on the photo, say — an office member may force the finish through **Force-close work order** with a mandatory reason of at least twenty characters (*The force-close reason must be at least 20 characters.*); the reason stays attached to the job as a visible note, so the exception is auditable instead of silent.

5.5 Inspection orders

Inspection work orders add a formal **Inspection result**; when the policy *Require inspection result on Done* is on, finishing without one is refused with *Please choose an inspection result.* A failed result

can spawn a corrective follow-up job automatically (*Inspection failed — open a corrective work order for the defects.*), so a deficiency never dies as a note in a PDF — it becomes scheduled work.

5.6 Time recording

Where the suite is installed together, the **More** menu offers **Log hours** (ProjectCheck) and **Record time** (ArbeitszeitCheck): both open the time apps with the work-order number already in the note. The link is soft — deleting the booking never touches the maintenance record.

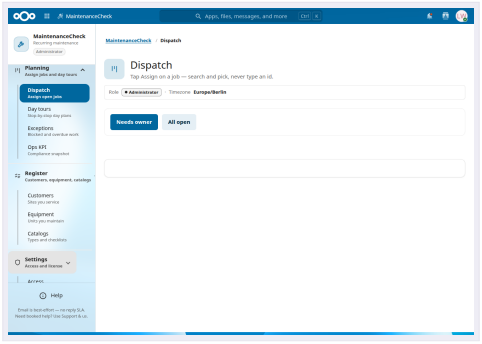
6 Planning: dispatch, tours, exceptions, KPI

The **Planning** group is the office's cockpit. It answers four daily questions: who takes which open job (**Dispatch**), in which order does each technician drive (**Day tours**), what is stuck and why (**Exceptions**), and are we keeping up (**Ops KPI**). Technicians do not see the group — a technician who opens a planning URL lands back on the due board, and their own day plan lives under **My tour**.

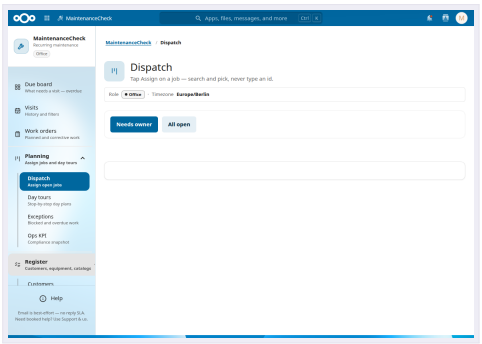
6.1 Dispatch: giving work an owner

Dispatch lists open work orders that still need a decision. The board's own words are the manual: *Jobs without an owner — tap Assign, pick a person, done.*

1. Filter first: **Dispatch filters** narrow by status, assignee or bottleneck; an all-assigned day reports *Every open job already has a technician — switch to All open to review.*
2. Tap **Assign** on a job and pick a person from the search — never type an identifier by hand.
3. Confirm with **Save assignment**; skills and capacity warnings appear before the commit, not after (Chapter 5).
4. Removing the owner returns the job to the pool: *Leave empty to remove the assignment.*



Dispatch with open jobs waiting for an owner.



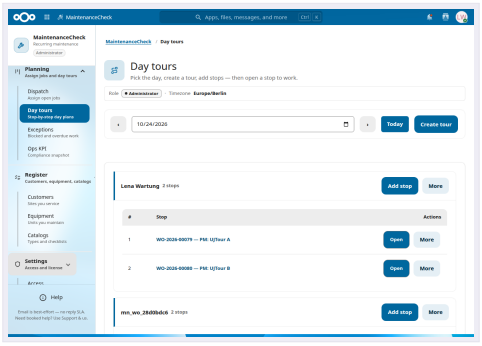
The same board in Olaf Neumann's office view.

6.2 Day tours: order for the road

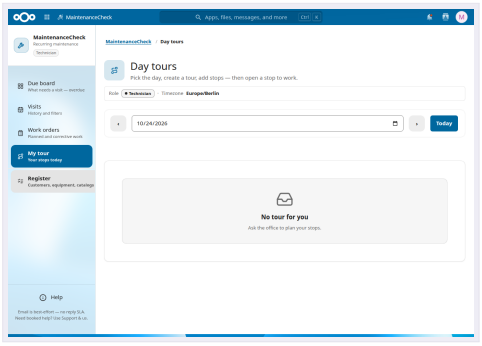
A day tour is one technician's ordered list of stops for one date — *One tour per technician per day — open a stop to do the work.*

1. On **Day tours** pick the date and **Create tour** for a technician: *Pick a technician. Then add work orders as stops.*
2. **Add stop** attaches open work orders to the tour; **Remove stop** releases one again.
3. **Suggest order** proposes a driving order (*Review the suggested stop order, then apply it.*); sanity check it against real traffic — *Suggested order — verify drive times* — then **Apply suggested order**.

4. **Lock order** freezes the sequence for the day; changing it later answers *This tour is locked. Unlock it to change the order.*
5. The technician now sees the route under **My tour** and works each stop top to bottom.



Day tours in the office view.



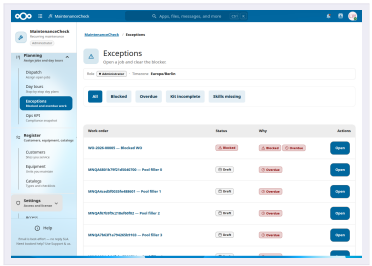
Tim Nolte's **My tour**: only his own stops.

Empty states speak plainly — *No tour for you*, *No tours on this day*, *No stops yet — add a work order.* Asking a technician for their plan returns *Ask the office to plan your stops.*

6.3 Exceptions: what is stuck and why

Exceptions is the negative space of the plan: every job that cannot proceed, with a translated reason badge instead of a raw code. The board fills itself — *When a job is blocked, overdue, or missing kit/skills, it appears here.* Reasons read as plain words: **Blocked**, overdue, **Kit incomplete**, **Skills missing**.

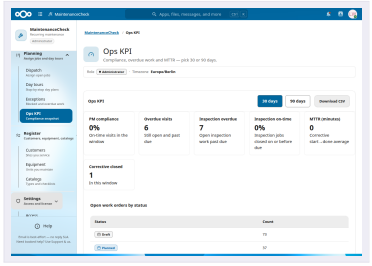
The point is not the list but the fix: each card tells you to **Open a job and clear the blocker.** — pack the kit, grant the skill, unblock or reschedule the job. When nothing is wrong the board says so: *No exceptions right now*, *No blocked jobs*, *No incomplete kits*, *No missing skills*.



The exceptions board with its reason badges.

6.4 Ops KPI: the honest scoreboard

Ops KPI condenses the window into a few numbers: on-time preventive visits, inspection punctuality (**Inspection on-time** vs. **Inspection overdue**, **Open inspection work past due**) and the overdue backlog. Pick the window — 30 or 90 days, as the subtitle *Compliance, overdue work and MTTR — pick 30 or 90 days.* says — and **Download KPI as CSV** when you need the same figures in a report. The metric definitions sit with each tile: *Inspection jobs closed on or before due* for inspection punctuality, *On-time visits in the window* for the headline rate.



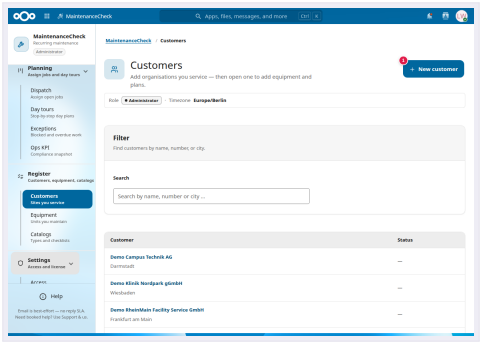
Ops KPI with the day-window chips.

7 Register and catalogs

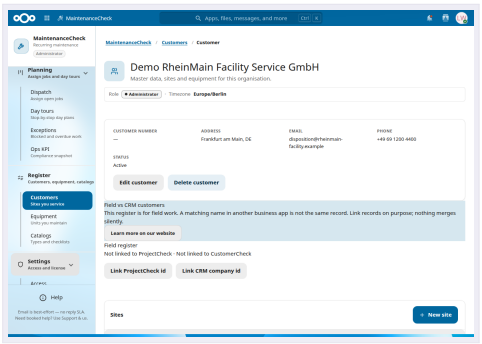
The **Register** group holds the master data everything else stands on: **Customers** with their sites, **Equipment** with plans, meters, inspections and visit history, and the shared **Catalogs** that keep types and checklists consistent. Reading is open to every app user — a technician on site needs the gate code as much as the office does. Writing belongs to office and administrators.

7.1 Customers and sites

1. Open **Customers** and use the create action (marker 1) to file an organisation like *Demo Klinik Nordpark gGmbH*.
2. Open the row — *Click a row to add equipment, sites and plans for that organisation.* — and add a **Site** for each address you service, such as *Haus B Technik*.
3. Keep **Access notes** current on the site — *Gate codes, parking, contact on site — copied to new work orders.* — because they travel automatically into every job.
4. Retire a customer you no longer serve with **Deactivate** instead of deleting: *Deactivate customers you no longer service — history stays readable.* Deleting a site with linked equipment is refused until the units move (*Delete “{name}”? Equipment linked to this site must be moved first.*).



Customer list; marker 1 the create action.

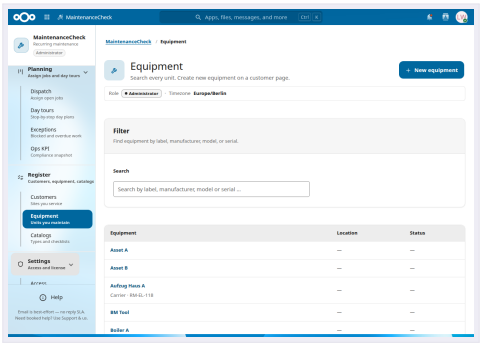


Customer detail with its sites.

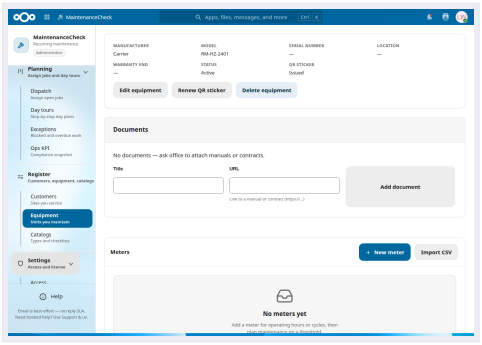
7.2 Equipment and QR stickers

Create a unit under **Equipment** with type, label, serial number and site — *Notstromaggregat NSG-3* at the Nordpark clinic is a typical row. **Create QR sticker** mints a sticker that opens this unit’s page when scanned; printing needs the pop-up allowed (*Pop-up blocked — allow pop-ups to print the sticker.*). Re-creating the sticker rotates the token: *Creates a new sticker. The previous sticker stops working.* — old stickers die immediately, which is exactly what you want when a sticker walks off.

Deleting a unit that carries history is refused in favour of deactivation: *This equipment has plans or visits. Deactivate it instead of deleting.*



The equipment register.



Unit detail: QR, plans, meters and history — *Plans, meters, inspections and visit history live on the detail page.*

7.3 Maintenance plans: the calendar engine

A plan on a unit says *when* the next visit exists. Two triggers:

Interval plans repeat on a calendar rhythm — weeks, months, years — starting from **First visit due on**. The app previews the roll (*The next visit will be due on {date}.*) and keeps intervals sane (*Interval out of range (max 3650 days, 520 weeks, 120 months or 10 years).*).

Meter plans fire on usage instead of time: *When a meter reaches a threshold*. They need a threshold (*A meter-triggered plan needs a threshold.*) and show *Visit opens when the reading reaches this value.* — no calendar visit is created meanwhile.

One open visit per plan is the rule — a second request answers *This plan already has an open visit.* After a cancellation, **Schedule visit** restarts the cycle manually (*Creates a new open visit for this plan — use this after a cancellation.*), and **Recalculate the open visit's due date** recomputes it from today. **Deactivate plan** pauses everything: *Plan inactive — no follow-up visit will be created.*

7.4 Meters and readings

Units with counters — hours, kilometres, cycles — get **Meters** on the detail page (**New meter**, **Edit meter**). Readings arrive two ways: typed into **Add reading** / **Save reading**, or bulk-loaded through **Import meter readings** as a CSV file — the app reports *Imported {n} readings — {t} plans became due.* as soon as thresholds trip. A counting-up guard (*Only allow readings that count up*) refuses a lower value with *This meter only counts up — the value is lower than the last reading.* A deactivated meter keeps its history but stops counting (*This meter is deactivated.*).

7.5 Inspection obligations

Legal or contractual checks live as **Inspection obligations** on the unit: *Add an obligation from an equipment class to schedule the next inspection.* Each obligation produces inspection work orders on its own rhythm; a failed inspection can block the next-due roll (*Failed inspection blocks next-due roll*) and spawn a corrective job (Chapter 5). Units without obligations say *No inspection obligations yet* or *No recurring inspections are set for this unit.*

7.6 Catalogs: shared lists

Catalogs is the toolbox behind jobs — *Equipment types, maintenance types, procedures, kits and skills.*

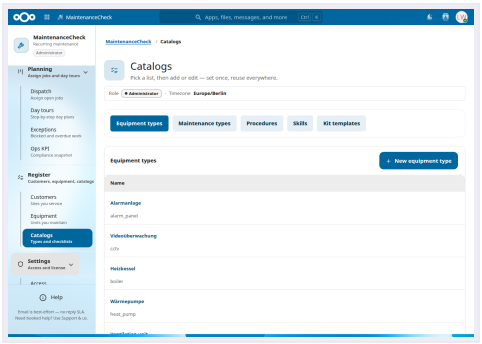
Equipment & maintenance types must exist before units and plans; empty catalogs announce *No types yet*. Deactivated types stay readable as *{name} (deactivated)*.

Procedures are reusable checklist templates — *Checklist templates for work orders. Import and export packs from the actions.* A procedure in use is forked before edits (*This procedure is*

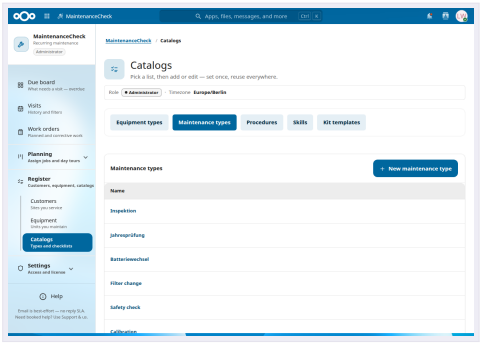
used by work orders. Fork it to make changes.), so history keeps the old wording. **Import pack** ingests `mn_procedure_pack_v1` JSON, **Export pack** shares one back out; overwriting a known code asks for the exact pack code first (*Type the exact pack code to overwrite.*).

Skills are codes jobs may require — *Skill codes for work orders. Use Grant skills to assign them to technicians.* With enforcement on, *Technicians need these skills when enforcement is on.*

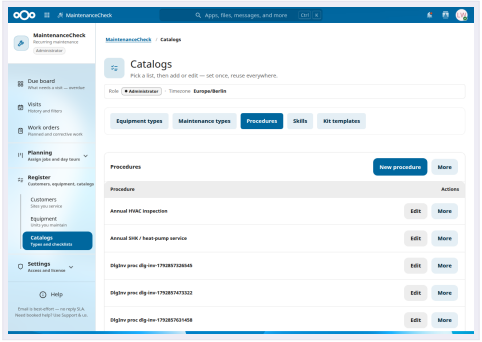
Kit templates are *Reusable parts lists for packing vans before a job.* — attach one to a job and the **Packed {packed} of {required}** counter starts watching.



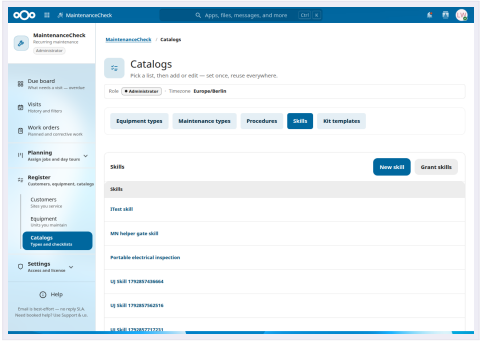
The catalogs hub.



Maintenance types: inspection, annual check, battery change.



Procedures and packs.



Skills catalog.

Empty lists hint at the next step — *No skills yet*, *No kit templates yet*, *Create a checklist template or import a pack.*

8 Bulk import and export

MaintenanceCheck keeps most data editable row by row, but two flows move data in bulk: meter readings arrive as CSV, and procedure checklists travel between installations as JSON packs. Both are deliberately conservative — the app validates loudly and writes atomically, so a bad file fails in full rather than poisoning half your register.

8.1 Importing meter readings

1. Open the equipment unit and its **Meters** section; note the meter codes on the unit's detail page — the file must reference them exactly.
2. Prepare a CSV with the columns the dialog documents: meter code, value, optional reading date (Y-m-d), optional note. A header row is allowed; the import caps at 500 rows per file.
3. Choose **Import meter readings**, select the file and confirm.
4. Read the result line carefully: it reports how many readings were accepted and how many plans rolled to due.

Three guards protect the ledger. Values must be numeric — a stray unit suffix rejects the row. Readings may not carry a future date. And the counts-up rule applies inside the file: a row lower than the last

stored reading is rejected with *This meter only counts up — the value is lower than the last reading.* Rows that pass land as ordinary readings, so meter plans trigger exactly as if you had typed each value on site.

8.2 Importing and exporting procedure packs

Procedures are reusable checklist templates; a *pack* moves them between Nextcloud installations — from a staging instance into production, or between sister companies.

1. On the source instance open **Catalogs** and use **Export pack**; the app writes a JSON file of type `mn_procedure_pack_v1`.
2. On the target instance open **Catalogs**, choose **Import pack** and select the file.
3. If a pack with the same code was already imported, the importer refuses instead of merging silently — type the *pack* code to confirm the overwrite (*A pack with this code was already imported. Type the pack code to confirm overwrite.*).
4. After import, review the procedures once: checklists travel as text, so site-specific wording stays editable per installation.

Inside an accepted pack, collisions resolve per procedure: an existing procedure code that nothing references is replaced outright, but a template that work orders still use is never rewritten in place — the old version is moved aside under a forked code (`CODE-v2`, `CODE-v3`, …) and deactivated while the incoming checklist takes the original code. A deactivated procedure whose code ends in `-v` in the catalog afterwards is that safety fork, not a duplicate.

The importer validates the pack format strictly: files that are not valid `mn_procedure_pack_v1` JSON, exceed the size limit, or contain no procedures are rejected before anything is written.

9 The mobile companion

MaintenanceCheck on the web is free software (AGPL) — every screen in this manual works without a licence. The optional phone and tablet companion is the commercial piece: it talks to the same Nextcloud from the field and needs a mobile licence key (prefix **MN2**) plus a named seat per device user, managed under **Settings** → **License & mobile** (Chapter 10).

9.1 What the companion does

The companion is built for the technician's day, not for administration. It mirrors the field workflow you know from Chapters 4 and 5:

- due visits and assigned work orders on the phone, with the same checklist, kit, photos and signature flow;
- the day tour as a stop-by-stop route — the office plans it on the web, the technician walks it on site;
- equipment lookup by QR scan, straight to the unit's plans and meters, plus meter readings on the spot;
- an app lock (PIN or biometrics) so a shared or lost phone does not open customer data;
- an offline queue: completions, photos and readings captured without a signal stay on the device and sync when the network returns — field work never waits for coverage.

9.2 Seats and limits

A seat is a named entitlement — *Named seats for the official apps — a software licence on invoice.* Users without a seat are told **You do not have a mobile seat assigned.**; seats beyond the

licensed count report **Your mobile seat is above the licensed limit.** and an expired key reads **The mobile license has expired.** None of that ever touches the web app: *No license key stored. The web app stays fully functional without one.*

This handbook documents the web application. A dedicated companion handbook is planned separately; the seat management side is covered in Chapter 10.

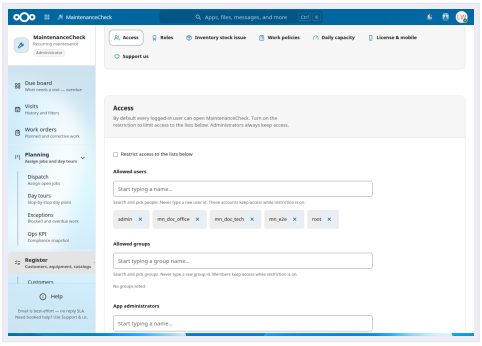
10 Settings and integrations

Settings is the administrators' room — office members get no menu entry and a blocked page instead of the forms (the shot at the end of this chapter), technicians the same. Seven sections live here, and each one controls a different kind of risk: who gets in (**Access**), who dispatches (**Roles**), what leaving a job does to stock (**Inventory stock issue**), how strictly the rules bite (**Work policies**, **Daily capacity**) and what the mobile licence covers (**License & mobile**).

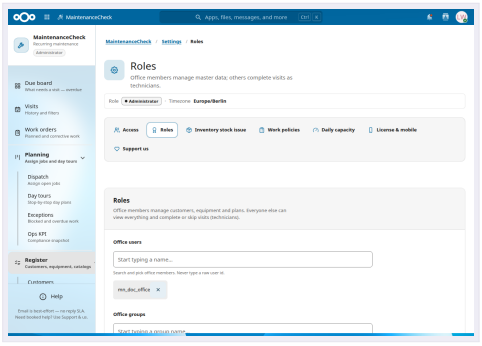
10.1 Access: the fail-closed door

Access decides who may open the app at all — *Who may open MaintenanceCheck — restriction, allow-lists, and app admins.* The restriction is off by default; enabling it means **Restrict access to the lists below**. Two allow-lists hold **Allowed users** (*Nextcloud user IDs with access while the restriction is on.*) and **Allowed groups** (*Members of these Nextcloud groups get access while the restriction is on.*); the same page names the app administrators.

The mechanics protect you in both directions. People off the list lose even the menu entry — the door fails closed, it never fails open. And Nextcloud system administrators can never lock themselves out: they always pass. Save the lists *before* enabling the switch — *Access settings saved.* is the confirmation.



Access: restriction, allow-lists, app administrators.



Roles: the office list by user and group.

10.2 Roles: who is office

Roles (*Office vs technicians*) is where administrators pick office members from the user and group pickers — *Office members manage master data; others complete visits as technicians.* Search and select, never type a raw user id: the picker guarantees the person exists, and deleted users cannot stay behind as ghost members. Everyone with app access who is not office is a technician automatically.

10.3 Work policies: how strictly rules bite

Work policies — *How strictly checklists, skills and daily capacity apply.* — tunes each guard to off, warn or enforce:

Checklist on done all required items must pass (**All required checklist items must pass**), a minimum percent threshold (*Used when the checklist policy is "minimum percent".*), or advisory only (**Checklist is advisory only**).

Skills on assign missing skills warn or block (*Missing skills on assign*).

Capacity on assign over-budget days warn or block (*Over capacity on assign*).

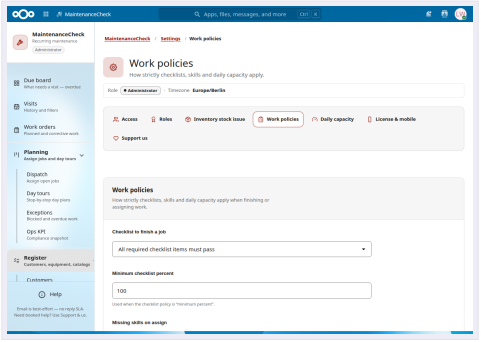
Equipment required **Require equipment on every work order**.

Failure code *Required for corrective jobs when policy says so.*

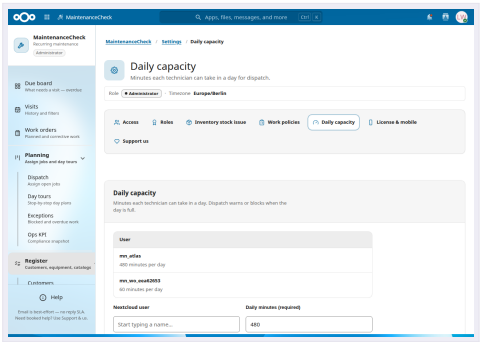
Defect follow-up **Defect follow-up after inspection fail** — *Off, warn in the UI, or automatically open a corrective work order.*

Inspection result **Require inspection result on Done**; **Failed inspection blocks next-due roll** stops the calendar from quietly rolling past a failed check.

Location policy **Location policy** governs how much site detail technicians see.



Work policies: checklist, skills, capacity and inspection guards.

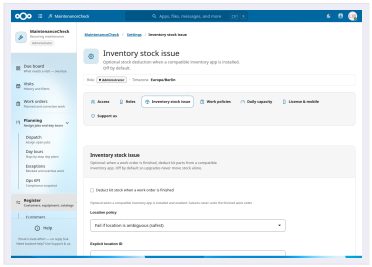


Daily capacity: the minutes-per-day budget per technician.

Daily capacity sets the dispatch budget — *Minutes each technician can take in a day for dispatch.* — defaulting to 480, with a **Capacity warn ratio** that starts warning before the day overflows. Save each section separately (*Work policies saved.*, *Capacity saved.*).

10.4 Inventory stock issue

Inventory stock issue wires the optional InventoryCheck bridge: **Deduct kit stock when a work order is finished** posts packed kit lines as stock issues when a job closes. The link is deliberately soft — if InventoryCheck is absent or the posting fails, the work order still completes; the job shows **Stock sync: {status}** (**Stock deducted**, **Stock not deducted**, **Stock sync is off**) so the stock state is visible instead of assumed.



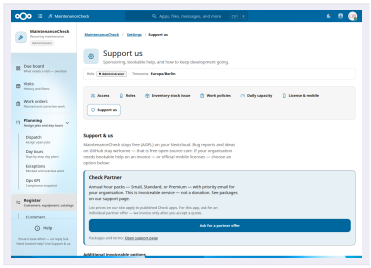
Inventory stock issue: the soft bridge to InventoryCheck.

10.5 License & mobile

License & mobile (*Keys and seats*) manages the companion licence: *Apply a mobile license key and assign named seats.* Apply the MN2 key (*License key applied.*), then hand named users a seat (**Assign seat** / **Assigned mobile seats**; **Remove seat for {user}** frees one). Full licenses refuse new seats with **All licensed seats are assigned. Remove a seat or upgrade the license.**, bad keys with **This license key is not valid: %s**, and **Remove license key** returns to the licence-free web app. **Request mobile seats by e-mail** and **View mobile license options** lead to purchasing.

10.6 Support us

Support us (*Sponsoring and help*) holds no app settings — it lists sponsoring options and bookable help on invoice. It is the only section that changes nothing in the database.

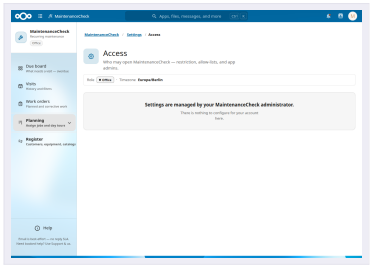


Support us: sponsoring and bookable help.

10.7 Soft links to the suite

MaintenanceCheck keeps its neighbours on soft leashes: InventoryCheck for stock on finish (above), ArbeitszeitCheck and ProjectCheck for the **Record time**/**Log hours** deep links in the **More** menus, and TourCheck, whose field-reported defects file corrective work orders here in status **Draft** — office dispatchers stay in charge of whether they become real work. None of the links synchronise data silently; each one hands over a reference, not ownership.

What office sees instead of these forms:



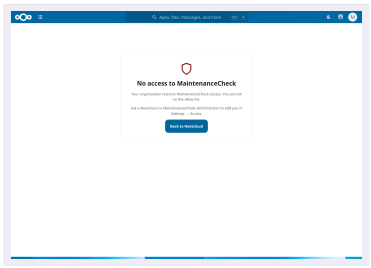
The blocked page an office member gets under **Settings**.

11 When access is denied

Denial in MaintenanceCheck is a page, not an error storm. Three different doors exist, and each closes quietly for a different reason — knowing which door you hit tells you whom to ask.

11.1 The app door: allow-list

When the administrator switches on the access restriction, the app refuses entry to anyone not listed — and it refuses *before* any data loads. The MaintenanceCheck entry vanishes from the Nextcloud navigation; a bookmark or a shared link opens a plain page reading **No access to MaintenanceCheck**. Nothing is half-shown and nothing leaks: fail closed means fail closed.



The denied page Dorothea Nolte sees: **No access to MaintenanceCheck**.

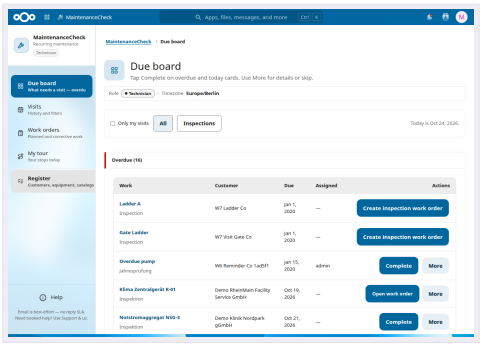
The fix is never technical on your side — ask the administrator to add your user or group under **Settings** → **Access** (Chapter 10).

11.2 The role doors inside the app

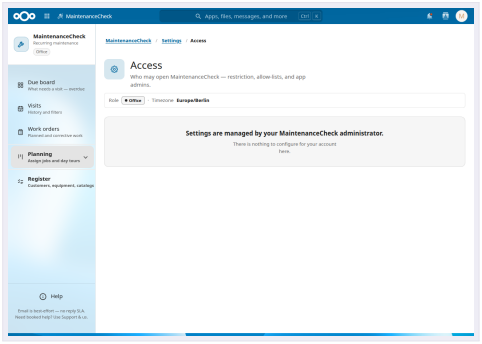
Once inside, two subtler doors shape what you see:

Planning hidden Technicians never see the **Planning** group — their day plan is **My tour**. A technician who opens a dispatch link directly is redirected to the due board instead of an error.

Settings blocked Office members and technicians who reach a settings URL get a blocked page, not the forms — the same **You do not have permission for this action.** the API returns. The menu simply does not offer the entry.



A technician opening **Dispatch** lands back on the due board.



Office without **Settings**: blocked, not broken.

11.3 The job door

Individual work orders carry their own door: a job that belongs to a colleague is locked for you unless you are the assignee, a helper, or office — or the job is still in the pool. That is also the answer to most “permission” surprises: check the assignment first, then the role.

12 Troubleshooting and FAQ

This chapter lists the messages the app can actually emit, grouped by where they appear. Each entry gives the cause and the way out. When a message is not an error but an empty state, it says so — an empty board is good news, not a failure.

12.1 Access and permissions

No access to MaintenanceCheck You are not on the allow-list while the restriction is on. The menu entry is hidden on purpose; ask an administrator (Chapter 11).

You do not have permission for this action. The action needs a role you do not have — typically a technician writing the register, opening **Settings**, or touching a work order that belongs to a colleague outside the pool (Chapter 1).

Blocked from phone UI This surface is office work; it does not exist on the companion by design.

12.2 Visits and plans

This visit was already closed. Someone finished it first — the list is current; look at the history instead of repeating the action.

This visit already has a work order. / This visit already has an open work order. Open the existing job; one visit carries one order.

This plan already has an open visit. Finish or cancel the open visit before scheduling the next.

This plan is inactive. Activate it before scheduling a visit. **Activate** the plan on the equipment page first.

No open visit — schedule one to continue the cycle. Not an error: the cycle paused after a cancel; **Schedule visit** restarts it.

The due date is not valid. / The due date must be between 2000-01-01 and ten years from today. Pick a real date inside the bounds.

The completion date is not valid. / The completion date must be a real date between 2000-01-01 a Completion dates may not lie in the future.

At least 10 characters when skipping. Give the skip a reason the next reader understands.

12.3 Work orders

Checklist blocks completion: {items} Required items are still open — finish them, or have the office force-close with a reason (Chapter 5).

Required checklist items are still open. Finish them or confirm the exception. Same guard in the confirm step.

The kit is not fully packed yet. / The kit is not fully packed yet. Office can override with a reason. Pack the remaining lines or ask the office.

The force-close reason must be at least 20 characters. The audit demands a real explanation, not a shrug.

Checklist is locked for this status. The checklist follows the job's status; move the job forward first.

The checklist already has results. Clear them before changing the procedure. Results are protected; clear them before swapping templates.

This work order is already part of a tour. Remove the stop before adding it elsewhere (Chapter 6).

The service report is available once the work order is done. Finish the job, then download the PDF.

Please choose an inspection result. The inspection policy requires a result on **Done**.

Please select the equipment this work order is for. The policy requires a unit on every job.

That person is already the technician. The assignment is already what you asked for.

The assignee does not have all required skills. / The assignee is missing required skills. Confirm to Skills guard — grant the skill or confirm the override if the policy allows.

This assignment exceeds the technician's daily capacity limit. The minutes budget for that day is full; pick another day or person, or override where permitted.

This tour is locked. Unlock it to change the order. **Unlock order** first, then reorder stops.

12.4 Register, meters and packs

This equipment has plans or visits. Deactivate it instead of deleting. History protects the unit; deactivate.

Delete "{name}"? Equipment linked to this site must be moved first. Move the units to another site, then delete.

"{name}" still has {children}. Deleting the customer removes all of it permanently. Same history rule on the customer side — confirm the cascade deliberately or deactivate instead.

An active plan is triggered by this meter. Change the plan first. Detach the plan's meter trigger before removing the meter.

A meter-triggered plan needs a threshold. / Enter a threshold for this meter plan. Meter plans fire on a number — set it.

The meter value is not valid. Numbers only.

This meter only counts up — the value is lower than the last reading. The count-up guard refuses suspicious readings; check the counter on site.

This meter is deactivated. Reactivate the meter before posting readings.

The pack file is not a valid procedure pack. / The pack file is too large. / Pack code is required. / Pa The imported JSON is not an `mn_procedure_pack_v1` pack — check the file.

A pack with this code was already imported. Enable overwrite to replace it. / Type the exact pack Code collision; confirm deliberately.

This code is already in use. Pick a different code for the catalog entry.

This procedure is used by work orders. Fork it to make changes. Forking keeps history intact.

Pop-up blocked — allow pop-ups to print the sticker. Allow the pop-up for the QR print view.

12.5 Photos, signature, licence

The photo is too large. / The photo limit for this work order is reached. Resize or use fewer shots per job.

Draw a signature first. / The signature image is not valid. / The signature image is too large. Re-draw inside the pad or check the image.

This license key is not valid: %s / The mobile license has expired. / All licensed seats are assigned. Companion licence states — web work is unaffected (Chapter 9).

12.6 Load failures

Could not load customers., **Could not load KPI.**, **Could not load exceptions.**, **Could not load comments.** and their siblings mean the page could not reach its data — reload first, then check whether you still have app access, and only then suspect the server.

12.7 How do I...?

…give a new technician access? Allow-list under **Access** if the restriction is on; otherwise just share the link — Chapters 11, 10.

…make someone a dispatcher? Add them under **Roles** — Chapter 10.

…set up recurring maintenance? Equipment → plan with interval or meter — Chapter 7.

…plan someone's day? **Day tours** → **Create tour** → stops — Chapter 6.

…finish a job that refuses Done? Checklist, kit, inspection result — Chapter 5.

…prove the work afterwards? **Download service report** / **Download inspection evidence** — Chapter 5.

…import meter readings in bulk? **Import meter readings** CSV on the unit — Chapter 7.

…let the phone app in? MN2 key plus a seat under **License & mobile** — Chapters 9, 10.

13 Accessibility

MaintenanceCheck is built to WCAG 2.1 AA inside its own canvas — the app content area, not the Nextcloud chrome around it. The acceptance runs cover the board at 320 px and 1280 px, keyboard-only flows and contrast; this chapter is the practical guide for users who rely on those properties.

13.1 Keyboard operation

- Every page lands focus in the app content (`#mn-main-content`); the tab order follows the reading order — navigation, page actions, then cards and lists.
- The current page marks itself with `aria-current="page"` in the sidebar, so position is announced, not only coloured.
- Menus and drawers behave like menus: **Menu** opens the mobile navigation drawer, Escape closes overlays, and the **More** menus on cards and job headers are fully keyboard-reachable — the danger item sits at the end, never on a hover-only corner.
- Dialogs (cancel visit, force-close, reschedule) trap focus while open and return it to the trigger on close.

13.2 Contrast, zoom and motion

Text and controls hold AA contrast on the suite's canvas in light and dark Nextcloud themes; red ink on danger actions such as **Cancel visit** is paired with wording, so colour is never the only signal. Reason badges on **Exceptions** are translated words, not colour codes. The layout is built mobile-first and works at 320/375/768/1024/1440 px without horizontal scrolling; browser zoom to 200 % reflows instead of clipping. Animations respect the system's reduced-motion setting, and forced-colors mode keeps borders and focus rings visible.

13.3 Screen readers

Interactive elements carry accessible names — icon-only controls announce their purpose (**Open navigation menu**, **Close navigation menu**); the drawer toggle states expanded/collapsed. List updates (filters, completed actions) and toast feedback use live regions, so a screen reader hears *Visit completed — next due {date}* without hunting for the change. Field errors are tied to their inputs instead of living in a detached banner.

13.4 Known limits

The Nextcloud header and app navigation outside the app canvas follow Nextcloud's own accessibility layer. PDF documents (job pack, service report, inspection evidence) are generated for print first; on-screen reading of the same data is always available on the work-order page.

14 Glossary

Domain terms a new user meets — not button labels, but the concepts behind them.

Due board The home lane of open visits, sorted by urgency, overdue first. It answers "what burns today?"

Visit One scheduled maintenance occurrence for a unit — born from a plan, finished by completing, skipping or cancelling.

Maintenance plan The recurrence rule on a unit: a calendar interval or a meter threshold that keeps producing visits.

Meter(-triggered plan) A plan fired by usage (hours, kilometres) instead of dates; a threshold reading creates the visit.

Work order The executable job around a visit or a defect — checklist, kit, photos, signature, PDFs. Kinds: preventive, corrective, inspection.

Procedure A reusable checklist template for work orders; packs share them between installations.

Kit The parts-and-tools list attached to a job; packing progress counts **Packed {packed} of {required}**.

Skill A capability code a job may require and a technician may be granted; policy decides whether missing skills warn or block.

Pool The set of unassigned jobs any technician may claim — listed after their own work.

Helper An extra person attached to a job beside the primary technician, with the same doors open.

Dispatch The office act of giving open jobs an owner.

Day tour / stop One technician's ordered route for one date; each work order on it is a stop.

Exception A job that cannot proceed — blocked, overdue, kit incomplete or skills missing — surfaced on its own board with a readable reason.

Inspection obligation A recurring formal check on a unit, tied to an equipment class; produces inspection work orders and can block the next-due roll on failure.

Equipment class A compliance category (e.g. pressure, lifts, fire alarm) obligations attach to.

Failure code A catalogued defect reason corrective jobs may have to carry when policy says so.

Force-close The office override that finishes a job with open required items — always with a reason that stays on the record.

Allow-list The named set of users and groups that may open the app while the access restriction is on.

App administrator A user trusted with app settings without being a Nextcloud system administrator.

MN2 The licence-key prefix of the paid mobile companion; web use never needs one.

Soft link A cross-app bridge (time recording, stock, field defects) that degrades gracefully instead of breaking the host action.

Ops KPI / MTTR The planning scoreboard: compliance, overdue work and mean time to repair over 30- or 90-day windows.

15 Surface index

Every surface of the app and where this manual covers it — the lookup table when you stand in front of a screen and want its chapter.

Surface	Who sees it	Chapter
Due board (home)	all roles	4
Visits list, filters, **More** menu	all roles	4
Work orders list and detail	all roles (jobs by identity)	5
Dispatch	office, admin	6
Day tours / **My tour**	office, admin / technician	6
Exceptions	office, admin	6
Ops KPI + CSV	office, admin	6
Customers and sites	read all, write office+	7
Equipment, QR, plans, meters	read all, write office+	7
Catalogs (types, procedures, skills, kits)	read all, write office+	7
Access settings	admin	10
Roles (office list)	admin	10
Inventory stock issue	admin	10
Work policies	admin	10
Daily capacity	admin	10
License & mobile (MN2)	admin	10, 9
Support us	admin	10
No access to MaintenanceCheck page	denied users	11
Mobile **Menu** drawer (<1024 px)	all roles	3
Companion phone app	licensed seats	9

15.1 Fastest path per role

Administrator **Access**/**Roles** first, then catalogs with the office, policies and capacity, optional stock issue and MN2 seats.

Office Customers → equipment and plans → jobs from the due board → assign and tours → exceptions and KPI. No settings.

Technician Due board or work orders (assigned before pool) → checklist, photos, kit → complete the visit → **My tour**. The register stays read-only.

15.2 Documents the app produces

Job pack the field sheet while a job is open — status, checklist, room for evidence.

Service report the customer-facing PDF once a job is **Done**.

Inspection evidence report the proof pack for inspections — *This creates an operational evidence pack — not a certificate.*

Download KPI as CSV the planning numbers for your own reports.

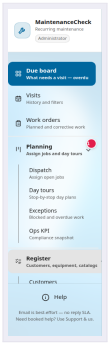
Meter CSV bulk readings in, due plans out (Chapter 7).

Shortcuts by role

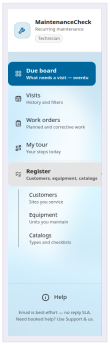
Administrator **Access**/**Roles** → catalogs/register with office → policies/capacity → optional stock issue and MN2 seats → dispatch/tours/KPI.

Office Customers → equipment/plans → job from due → assign → **Planning** (tours, exceptions, Ops KPI). No Settings.

Technician Due or jobs (assigned before pool) → checklist/photos/kit → complete visit → **My tour**. Register read-only.



Admin navigation including Planning and Settings.



Technician navigation without Planning.

About Software by Design

Software by Design builds Nextcloud apps for self-hosted data — time, projects, stock, fleet, CRM and maintenance. 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

MaintenanceCheck 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, 14
Access denied, 16
Access restriction, 1, 14
Accessibility, 19
Administrator, 1
Allow-list, 1, 16
Assignment, 6, 8
Catalogs, 10
Checklist, 6
Comments, 7
Companion app, 13
CSV, 12
CSV import, 11
Customers, 10
Daily capacity, 14
Day tours, 8
Dispatch, 8
Due board, 4
Equipment, 10
Errors, 17
Exceptions, 8, 9
Export, 12
FAQ, 17
Focus order, 20
Getting started, 3
Glossary, 20
Helpers, 2, 6
How do I, 19
Import, 12
Inspection, 7
Inspection obligations, 11
Integrations, 16
Interval, 11
Inventory stock issue, 14
InventoryCheck, 15
Keyboard, 19
Kit, 6
Kit templates, 11
KPI, 9
License & mobile, 14
Log hours, 8
Maintenance plan, 11
Meter readings, 12
Meters, 11
MN2, 13, 15
Mobile seats, 2, 13
More (action menu), 5
Navigation map, 21
No access, 16
Office, 1
Office list, 2, 14
Ops KPI, 8
PDF documents, 22
Permission matrix, 1
Permissions, 1
Photos, 7
Planning, 8
Policies, 14
Pool, 6
Procedure packs, 13
Procedures, 11
QR sticker, 10
Record time, 8
Register, 10
Role paths, 22
Roles, 1, 14
Screen reader, 19
Seats, 15
Settings, 14
Sign in, 3
Signature, 7
Sites, 10
Skill assignment, 2
Skills, 2, 11
Status, 6
Stock issue, 15
Stops, 8
Support, 16
Surface index, 21
Team, 2
Technician, 1
Tour, 8
Troubleshooting, 17
Visits, 4
Work orders, 6
Work policies, 14