



Tenant User Manual

Every screen and every action available to staff at a business using PayGO — organised by role, so you can go straight to what your own seat can do.

COVERS

Every tenant role

VERSION

2026-09-14

UPDATED WITH

docs/REPORTING_PLAN.md — Reports tab, sections, device-expiry

Who this is for

A business using PayGO has up to eight kinds of seat, each seeing a different set of screens. This manual covers all of them — find your own role below and use it as your quick reference; the screen-by-screen sections further down go into the detail of each one.

ROLE	WHAT THE SEAT IS FOR	MAY ISSUE	BATCH?
Tenant administrator	Runs the business inside PayGO: staff, prices, plan, branding, and everything every other seat can do	Add/set time, lock, permanent unlock, counter repair	Yes
Commercial officer	Payment contracts and the customers behind them — confirms money, approves prices, releases paid-off kits	Add/set time, lock, permanent unlock	No
Token officer	Issues tokens, including two-hundred-device runs a technician or agent has asked for and a commercial officer has approved	Add/set time, lock, counter repair	Yes
Sales agent	Registers households and takes money at the door	Add/set time	No
Technician	Fits kit, gives the first token, corrects a device in the field	Add/set time	No
Support agent	Front-line help — can top up, cannot set an absolute balance or unlock	Add time only	No
Partner integration	A 3rd-party system, authenticating with an API key rather than a password — same limits as a sales agent	Add/set time	No

ROLE	WHAT THE SEAT IS FOR	MAY ISSUE	BATCH?
Viewer	Read-only — checks what has happened, issues nothing	—	No

A **recovery agent** is not a ninth role — it is a permission (named by your administrator) that lets a person reset a colleague's password from the People screen, on top of whatever role they already hold.

Signing in

1. Open the link your administrator gave you and enter the username and password you were issued.
2. The **first time** you sign in with a password somebody else chose, you are asked to replace it before anything else works — this is not optional, and there is no way to postpone it.
3. A session lasts 12 hours. If your account is deactivated or your role changes while you are signed in, your next action is refused and you are asked to sign in again.
4. Eight wrong password attempts locks the account for 15 minutes — this protects the account, it is not a fault to report unless it happens to someone who was not the one typing.

Finding your way around

The navigation on the left (or, on a phone, along the bottom and behind **More**) is built from your own role — a screen you cannot use simply does not appear, rather than appearing and refusing you. The sections below are in the order they appear in the navigation.

Any list on any screen — tokens, households, a report's own tables — shows 5 rows at a time by default. The box beside it lets you choose 10, 25, 50, 100, 200, 500 or All, filter by typing, and sort by clicking a column heading; **Download CSV** saves every row currently matching your filter, not only the page on screen.

In this manual

- | | |
|--------------|------------------|
| 1. Dashboard | 10. Reports |
| 2. Issue | 11. Prices |
| 3. Inventory | 12. Messages |
| 4. Devices | 13. Business |
| 5. Customers | 14. Places |
| 6. Record | 15. Device Types |
| 7. Approvals | 16. Connect apps |
| 8. Payments | 17. Prospect |
| 9. Accounts | 18. People |

Dashboard

Every tenant seat

The one screen answering a question you have before you know which household or which kit you mean — where the business stands right now.

- **Period** — Today, This week, This month, This year, or a custom date range. Narrows every figure on this screen that is about *activity* — what was registered, added or issued a token inside it. Stock and register totals are always as of right now and are never narrowed by the period, since there is no past snapshot to filter.
- **Households** — how many are registered, and where the funnel stands (asked about, registered, converted, stalled), plus how many were registered in the period chosen.
- **Kit** — devices by state (in stock, running, expired, locked, unlocked), and how many were added to the fleet in the period chosen.
- **Devices issued tokens** — every device that received a token in the period, newest first, with the count, days granted and whether it was confirmed entered. Downloadable as a CSV like any other table.
- **Where they are** — a map of every household with a known location; tap one to open it.
- **By place** — the same numbers broken down by region, district, ward and village.

Issue

Every non-vendor seat

One device, one household, one token — the screen this app is built around.

1. Find the device by serial (or scan/search it) or the household it belongs to.
2. The screen shows the kit's own estimated runtime and what your seat is permitted to give it — **add time** (top up what is left) or **set time** (replace the balance outright). Both are shown; which one you want is never hidden behind an "advanced" option.
3. Choose the amount and press **Issue**. The nine-digit token appears immediately — read it to the customer or key it into the device yourself.
4. If the device was tokenised very recently, a second token within the cooling-off window needs another seat's approval (see Approvals) unless the issue is tied to a payment your seat just recorded, which is exempt.

Token officers, sales agents and technicians name the payment. Before issuing time by hand, search by the payment's reference (or the payer's phone number) and pick the matching unspent payment for this device — picking it is what uses it up, so the same payment can never fund two tokens. If there honestly is no payment (a goodwill extension after an outage, say), press **Ask for an exception for today** on the refusal: a commercial officer decides how many tokens, for you, for today only. Commercial officers and administrators are never asked.

The token type is always on screen. Setting time on a kit already in service can take away days the household already paid for — the screen says so before you press Issue, it does not let you find out afterwards.

The device serial field locks — and visibly shrinks — the moment a token goes out, so a stray tap or paste cannot send the next token to the wrong kit by accident. Press **Change device** to unlock it and move on to a different serial.

Inventory

Tenant administrator

Commercial officer

Token officer

Sales agent

Technician

Support agent

The shelf, kit-first — for handing a specific device to a household rather than looking the household up first (that is Customers).

1. Pick a device from what is on the shelf.
2. Attach it to a household — an existing one, or register a new one on the spot.
3. Issue the opening token in the same screen.

"Your devices" lists everything issued through your own seat, for a quick look back.

Devices

Tenant administrator

Token officer

Loading a manufacturer's file, and fixing a device already on record.

1. **Load devices from a file** — upload the manufacturer's export. The screen previews exactly what will change before anything is written; nothing is applied until you confirm. A key, a starting code, a counter, or an `is_locked` flag already on record is never overwritten by an import.
2. **Find and correct a device** — search by serial to fix a mistyped model, location, or customer link. This never moves a counter and never touches the secret key.

Customers

Every non-vendor seat — cards inside vary by role

Households first, then the kit — the mirror of Inventory.

- **Around you / the map** — nearby households, with distance, if your browser shares your position.
- **Potential customers** — leads: asked about a kit but not yet registered. Sales/field seats only.
- **Money taken, the books, agents and their float, payments to confirm** — the field-money queue: what an agent is carrying, and the commercial officer's confirmation queue. Confirming money is never the same seat that took it.
- **Load many households / load customers from a file** — bulk registration. A row with a serial is a customer (the kit attaches); a row without one is a lead. Name plus all four of region/district/ward/village are required on every row, and a row with a serial also needs an identification type and number — the kit is fitted, so who holds it is no longer optional. Hamlet (sub-village), birth year, household size, profession and prior electricity source may all be added on a row too, and are never required. Money on a sheet becomes a *pending* receipt, never credited automatically.
- **Customer register** — every household, searchable, with which kit (if any) they hold and how long it has left.
- **Kits in customers' hands** — the fleet, viewed by who holds it.
- **Agreement to keep their details** — on the registration form, choose how the household agreed: *their phone number is on file* or *they signed, and I have a photo*. It is never required, so a form is never held up; a household without one is listed as still needing it. Once recorded it cannot be changed.
- **Photos from Kobo** — a household brought in through a connected Kobo form may carry a signature photo and a photo taken at the work site, alongside the identification photo. Shown read-only on the household's own edit screen when either is on file; there is no camera capture for these two in PayGO itself, only through Kobo.
- **Households with no agreement** (commercial officer) — a bar above the register counts them. Filter to the households you have checked with whoever registered them and press **Confirm the ones shown**; your name and the date are recorded against each (200 at a time).
- **Download their data** (administrator, commercial officer) — on each household: a zip with everything held about them, for a request from the household themselves. Give it only to them or to someone they have authorised.

Record

Every non-vendor seat, including Viewer

What has actually been given out — readable by everyone with a tenant, because the person who most needs to check is not always the person who may issue.

- **What has been issued** — every token, with totals (tokens, days granted, devices touched, confirmed vs. not).
- **By product / by person** — the same totals broken down.
- **Tokens** — the full searchable list; filter by date, person, product, device or confirmation state, and download as CSV.
- **Notifications** — what an automatic payment tried to tell a household (a token arriving, a kit settling, a device-expiry warning) — never the message text itself, only whether it went.

Approvals

Any seat with something to ask or decide

Two queues on one screen. Which cards you see depends on whether your seat asks or decides.

- **Ask for tokens** — any seat that may issue can ask for a device to be tokenised again inside its cooling-off window, or for a commercial officer to approve a run of many devices at once.
- **Approvals** — a commercial officer or token officer (or a tenant administrator, standing in) decides a repeat-token request; a commercial officer decides a run.
- **Days waiting / token runs** — once a run is approved, a token officer generates it under their own allowance, against the exact list frozen at approval time.
- **Unlocks waiting** — a kit paid off in full waits here for a commercial officer or administrator to release it permanently. This cannot be undone.
- **Tokens with no payment behind them** — a token officer, sales agent or technician who has no payment to name asks here; a commercial officer or administrator says how many tokens, for that person, for today only. It ends at midnight, covers time tokens only, and never a lock or a permanent unlock.

Payments

Tenant administrator

Commercial officer

Token officer

Sales agent

Support agent

Every payment received, who paid, and which token it funded — the detail behind the single count on The Books.

- **State** — *Unspent* is money taken that has not yet become days; *Spent* funded a token (the token and who issued it are on the row).
- **Method, From, To** narrow the list; the box above the table searches everything on a row — a reference, a phone number, a name, a serial.
- **Download CSV** saves exactly the rows left after filtering.

Payments to split

Most payments are for one device and are recorded on that device's account as usual. When one handover of money covers **more than one device**, record it under **Payments to split** instead (+ *Record a payment to split* — amount, how it arrived, reference, who paid and their phone). It is then **held**: it buys no days and issues no token, and nothing is credited to any device.

- **Commercial officer** presses **Split**, types each device's serial and its share, and the window shows how much is still to place. The shares must add up to the payment **exactly**. Each share is credited like a payment recorded on that device's account — a share that meets the deposit issues the opening token. Read each token from its device's account.
- A payment can be split **once**. If one device's share is refused (for instance its account is closed) nothing already credited is taken back; fix the cause and press Split again to finish the rest — the shares already chosen stay as they were.
- **Void** is for a payment recorded by mistake before anything was credited from it, with a reason; its reference is freed so the right payment can be recorded.
- Only money not yet credited can be split. Money already credited to a device stays there.

Payer names and phone numbers are personal data, so this screen is for seats that already reconcile or record money.

Accounts

Every non-vendor seat

One agreement per kit — what is owed, what has been paid, and what it would cost to close today.

1. **Sell a kit** — pick a device, a household, and a product; the price is frozen at the moment of sale and a later price change never reaches it.
2. Recording a payment against an open account credits it and, if the opening period is still owed, issues the token in the same step.
3. **Release a kit that has not been paid for** — an early permanent unlock, for the seats that may (commercial officer, administrator) — reversible only by a new agreement, not by a token.

Reports

Every seat with a tenant — the "Reports you receive" and "Devices nearing end of runtime" cards are

Tenant administrator only

What happened, on demand — and, for the administrator, what arrives automatically.

1. **Reports** — pick Daily, Weekly or Monthly and press View. This is always readable, whatever the automatic settings below say. Each list (new devices, activations, top-ups, unlocks, by product, by person) is its own table, not a wall of text — filter, sort, choose up to 500 rows or All, and Download CSV for the full list whatever you're currently filtered to. The emailed/PDF copy still keeps its own short version of each list, for a message that stays readable on a phone.
2. **Reports you receive** (administrator only) — tick Daily, Weekly and/or Monthly to have the report emailed automatically, and an SMS alert for same-day issues. Two optional sections can be added to the report body: **Revenue & Sales** (real payments, by method) and a **Regional breakdown** (devices in service, by region) — both off until turned on, and turning one on never switches the other off.
3. **Devices nearing end of runtime** (administrator only) — turn this on and set how many days of power counts as "soon." Once on, the household gets an SMS warning and every active commercial officer, sales agent and technician gets an email naming which devices newly need attention — once per shortfall; paying again and later running low resets the warning.

Prices

Every non-vendor seat reads — publishing is Tenant administrator / Commercial officer

Every seat may read a price, because an agent at the door has to quote one. Only the seats that own commercial terms may publish a new one.

- **Products** — the device types this business sells or leases.
- **Publish a price** — a new version of a product's price book. A sale already made keeps the price it was sold at; only new sales use the new price.

Messages

Tenant administrator Commercial officer Sales agent

What an automatic payment message says, and sending one message to many people at once.

- **Automatic message content** — which language every household is written to (administrator's choice) and which optional pieces each message kind includes — a greeting, a balance (commercial officer's choice). A message's mandatory content — the token itself, the amount — can never be switched off.
- **Send a message to many people** — a bulk SMS to a chosen list of households. Before sending, fill in **Consent**: a short sentence saying how these people agreed to receive messages from you (for example, "customers who gave their phone number at registration"). The send is refused without it, and the sentence is recorded with the send. Bulk messages cannot be recalled.

Business

Tenant administrator

Who this business is, and its household numbering.

- **Organisation information** — name, contact, TIN, location, the SMS sender name a household sees, and the language every automatic message goes out in.
- **Customer codes** — the letters households are numbered under, and how much room is left in the current one.
- **Your logo and colour** — branding your staff see when they sign in.
- **Connected phones & apps** — the keys that let an AutoSMS phone or a partner's system talk to PayGO for this business.

Connect an AutoSMS phone

1. Press **Connect an AutoSMS phone** and give the phone a name you will recognise later (for example *shop-phone-1*).
2. PayGO makes two keys and shows them **once**, with the steps and the service address to type on the phone: a *payments* key (reports money, reads no token) and an *SMS relay* key (sends PayGO's messages from the phone's SIM, reports no money).
3. On the phone: AutoSMS → Settings → Integration → PayGO Token Service. Paste both keys and set the claim number.
4. Send a small test payment and watch it arrive.

A phone lost or replaced: press **Revoke** on both of its keys. Nothing else stops working. If a key is lost before it was typed in, revoke it and connect again — a key is never shown twice.

Create key makes a single key for another system; tick only what it needs.

Places

Tenant administrator

Where the business works — region, district, ward, village, and an optional hamlet / sub-village — kept as a maintained list, never a gate on registration.

1. Add one place at a time on the form, or load a file — one row is a whole path (region, district, ward, village, hamlet), so a single row can add up to five places at once.
2. A short row is fine (a region alone, or a region and district); a gap in the middle of a row is refused rather than half-loaded. Hamlet is always optional, on the form and in a file.
3. Loading the same file twice adds nothing — it is not a special check, places already on the list are simply skipped.

A registration is never refused for naming a place off this list — the list is a picker, and an import reports an unlisted place rather than rejecting the row.

Device Types

Every tenant seat reads

Tenant administrator adds and edits

The physical components a kit is built from, named once — any device whose model string matches a type's name or one of its aliases reports the same components, so getting one right fixes every device built from it at once. This is what a report elsewhere on the platform reads to know a kit's rated solar wattage, battery capacity, system voltage and rated power.

1. **+ Add a device type** — a name, a manufacturer (who actually made the hardware, not the business reselling it), and a list of components. Each component is tagged with what it represents (solar panel, battery, system voltage, rated power, or "other" for anything not needed elsewhere) plus a name, a size and how many.
2. **Also known as** — alternate model strings already typed onto real devices. Any device whose own model text matches the type's name or one of these is treated as this type — nothing on the device record itself changes.
3. A type's name cannot be changed once added, since it is what existing devices are matched against — retire it and add a new one instead of trying to rename it.
4. **Retire** stops a type being offered for new devices without deleting it; devices already matching it keep matching it. **Restore** brings it back.

Connect apps

Tenant administrator

Plug PayGO into other systems without any technical knowledge: bring households in from a form, or send installed devices out to another company. Choose the system, answer the two or three questions only you can, and PayGO works out the rest.

GreenLeaf Technology

conrad
Tenant administrator

Sign out

TOKENS

- Dashboard
- Issue
- Inventory
- Devices
- Customers
- Record
- Approvals
- Payments
- Accounts
- Reports

SETTINGS

- Prices
- Messages
- Business
- Places
- Device Types
- Connect apps**

Connect apps

Bring households in from a form, or send installed devices out to another system. Choose the system below and follow the steps — nothing here needs any technical knowledge, and **nothing is sent out until you press Send**.

Connected

Nothing is connected yet. Choose a system below.

Connect something new

KoboToolbox (household registration)

Brings data into PayGO — A deployed Kobo form pushing one submission per household to this tenant (lib/kobolntake.js). One connection is one form, one tenant.

Connect

Prospect (Tanzania) — send installed devices

Sends data out of PayGO — Sends the devices you have queued, with their household and specifications, to Prospect. Nothing is sent until you press Send.

Connect

A spreadsheet of households

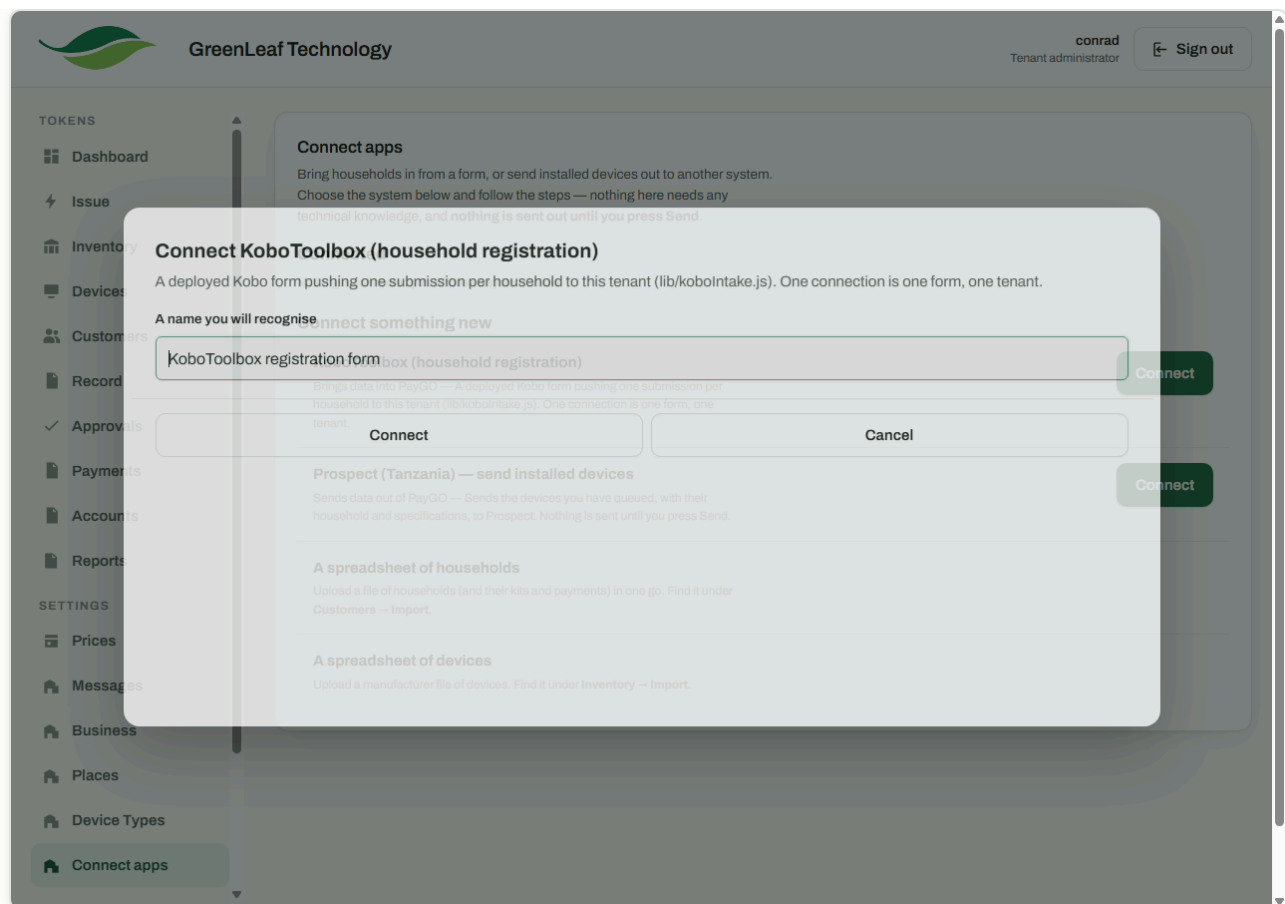
Upload a file of households (and their kits and payments) in one go. Find it under **Customers** → **Import**.

A spreadsheet of devices

Upload a manufacturer file of devices. Find it under **Inventory** → **Import**.

Connect apps, before anything is connected — the two systems on offer, and where the spreadsheet importers already live.

1. **Connect something new** lists what can be connected. Press **Connect**, give it a name, and confirm with the 6-digit code from your authenticator app — opening a path to another system always asks for it.

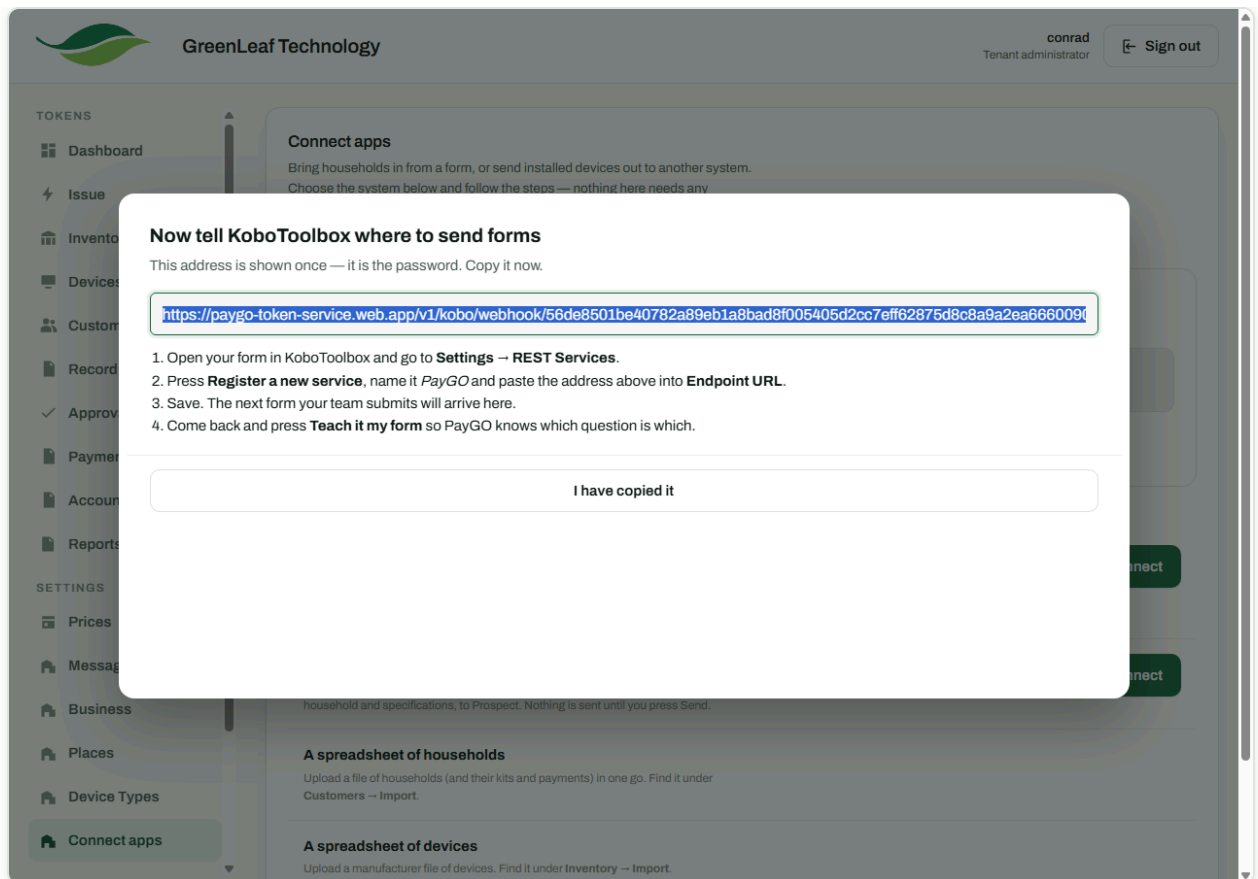


Connecting a Kobo form only ever asks for one thing: a name you will recognise it by.

2. **KoboToolbox form** — brings households into PayGO from a Kobo form, three ways: a live push every time somebody submits, a pull PayGO does itself (on a timer, or on demand) that also catches a submission that was *edited* after it was first sent in, and a one-time load of a backlog collected before any of this existed.

Part 1 — connect the form (do this once)

1. On the Connect apps screen, under **KoboToolbox form**, press **Connect**. Give it a name you will recognise later (for example, "Field registration form") and confirm with the 6-digit code from your authenticator app.
2. PayGO now shows you an address — a long one, starting `https://.../v1/kobo/webhook/...`. This is shown *once only*; copy it before closing the window, because PayGO cannot show it to you again (if you lose it, use **New address** on the card to get a fresh one).



Shown once, right after connecting — this address is the password. Copy it into KoboToolbox's REST Services before closing the window.

3. In KoboToolbox, open the form itself, go to **Settings** → **REST Services**, add a new service, and paste the address you just copied in as its endpoint URL. From this point on, every new submission on that form is sent to PayGO the moment somebody submits it.
4. Back in PayGO, the card for this connection now exists.

GreenLeaf Technology

conrad
Tenant administrator [Sign out](#)

TOKENS

- Dashboard
- Issue
- Inventory
- Devices
- Customers
- Record
- Approvals
- Payments
- Accounts
- Reports

SETTINGS

- Prices
- Messages
- Business
- Places
- Device Types
- Connect apps**

Connect apps

Bring households in from a form, or send installed devices out to another system. Choose the system below and follow the steps — nothing here needs any technical knowledge, and **nothing is sent out until you press Send**.

Connected

Village registration form on

KoboToolbox (household registration) · brings data in

Last form received **none yet**

Form questions set up **built-in form**

Teach it my form Test a form What arrived New address Pause

Connect something new

KoboToolbox (household registration) Connect

Brings data into PayGO — A deployed Kobo form pushing one submission per household to this tenant (lib/kobointake.js). One connection is one form, one tenant.

Prospect (Tanzania) — send installed devices Connect

Sends data out of PayGO — Sends the devices you have queued, with their household and specifications, to Prospect. Nothing is sent until you press Send.

A spreadsheet of households

Upload a file of households (and their kits and payments) in one go. Find it under Customers → Import.

A spreadsheet of devices

Upload a manufacturer file of devices. Find it under Inventory → Import.

Once connected, the card shows when a form last arrived and offers buttons to test it, see what has arrived, load or pull its data, and change its settings.

Part 2 — teach PayGO your form's questions (skip this if your form already uses PayGO's standard question wording)

1. On the Kobo form, submit one real test entry (any made-up name and details are fine).
2. Back on the card in PayGO, press **Teach it my form**, then **Use the form that just arrived**. PayGO reads that test submission and proposes which of its questions is the name, the phone number, the village, and so on.

GreenLeaf Technology

conrad
Tenant administrator

Sign out

Check each question

PayGO has made its best guess. Change anything that is wrong — a question you do not need can stay "not used".

QUESTION ON YOUR FORM	IS PAYGO'S...
Jina la kwanza e.g. Amani	First name
Jina la mwisho e.g. Juma Bakari	Last name
Jinsia e.g. Mwanaume	Gender
Namba ya simu ya mteja e.g. 0712345678	Phone number
Wilaya e.g. Mkuranga	District
Kata e.g. Kisiji	Ward
Kijiji e.g. Koma	Village
Namba ya Kifaa e.g. 501000001	Device serial number

Save this Cancel

Connect apps

A spreadsheet of devices
Upload a manufacturer file of devices. Find it under Inventory → Import.

“Teach it my form”: PayGO proposes which question is which field from a real submission. Check each row and change anything wrong before saving.

3. Check every row. Where PayGO guessed wrong, change it to the right field using the dropdown next to it. When every row looks right, press **Save**.
4. **Test a form** lets you paste any submission and see exactly how it would be filed — without writing anything — any time you want to double-check the mapping later.

Part 3 — turn on pulling, so PayGO can also fetch data itself (recommended — this is what catches an edit made after the fact)

1. **Get a Kobo API token.** Sign in to KoboToolbox in your browser. Open your **Account Settings** and look for **API Access** (on some accounts, visiting <https://kf.kobotoolbox.org/token/?format=json> while signed in shows the token as plain text instead). Copy the long string of letters and numbers shown — that is your token.
2. **Find your form's id.** Open the form in KoboToolbox and look at the address bar. It reads something like <https://kf.kobotoolbox.org/#/forms/awPRkLydN5KU7VGesKAX4a/summary>. The form id is only the part between /forms/ and the next / — in that example, awPRkLydN5KU7VGesKAX4a. Copy just that part, nothing else.
3. On the Kobo card in PayGO, press **Edit**. Paste the token into the **Kobo API token** field, and the form id into the **Kobo form id** field.

4. If you want PayGO to check Kobo on its own, type a number of minutes into **Pull automatically every (minutes)** — 30 is a sensible choice for most forms. Leave this box empty if you would rather only pull by pressing a button yourself.
5. Press **Save** and confirm with your authenticator code again.
6. The **Pull now** button on the card only lights up once both the token and the form id are saved. Press it any time to fetch what is new or changed immediately — you do not have to wait for the automatic timer. PayGO tells you how many households were new, how many were updated, and how many were already up to date.

What a pull actually does, in plain terms:

- A submission PayGO has never seen before becomes a new household — exactly as if it had arrived through the live push.
- A submission that is already on file, and has since been *edited* in Kobo (say, somebody corrected a phone number), has that correction copied across — but **only into a field that is still blank** in PayGO. A field that already has a value — whether it came from Kobo the first time, or was typed in by somebody here at your business afterwards — is **never** overwritten by a later Kobo edit. This protects anything your own staff corrected from being silently undone by an old Kobo entry.
- A submission that has not changed at all since the last pull is simply left alone — nothing is rewritten, and it does not count as new or updated.

Other buttons on the card, for reference: **What arrived** lists everything received and, for anything a person needs to look at, why. **Load existing data** is for a backlog of households this form collected *before* any of the above was set up: choose or paste the form's own CSV export from KoboToolbox (its Data tab → Downloads — not a PayGO template), read what it would do, named row by row for anything it cannot use, then confirm. However a submission arrives — live push, a pull, or a loaded file — it is never written twice; PayGO always recognises one it has already filed.

3. **Prospect** — enter your Prospect key. The address is fixed to Prospect's own; nothing else can be typed. This connection is sold separately from your plan — if it is not switched on yet, the card shows a notice instead of a Connect button; ask your account manager. Devices are chosen on the Prospect screen, or press **Sync now** on this card to find and send every ready household that has not been sent yet in one step — no selecting required. **See what will be sent** shows the exact details (name and ID number included) before you press **Send** — with the code again. Each household is sent once, up to 50 at a time, and a device already sent cannot be released or sent again on its own; if its data changes afterwards, press **Resend** for that one device on the Prospect screen (also code-protected) to send the correction.

GreenLeaf Technology

conrad
Tenant administrator

Sign out

TOKENS

Dashboard

Issue

Inventory

Devices

Customers

Records

Approvals

Payments

Accounts

Reports

SETTINGS

Prices

Messages

Business

Places

Device types

Connect apps

Connect Prospect (Tanzania) — send installed devices

Sends the devices you have queued, with their household and specifications, to Prospect. Nothing is sent until you press Send.

A name you will recognise

Prospect

Connected

Prospect's address

Leave as it is unless Prospect told you a different address.

https://tz.prospect.energy

Where devices are sent

/v1/in/shs

Teach it my form

Test a form

What arrived

New address

Pause

Key goes in header

Authorization

Connect something new

Key is sent as

KoboToolbox (household registration)

Bearer, or "none" to send the key on its own.

Bearer

Your Prospect key

Prospect (Tanzania) — send installed devices

Sends data out of PlayGO — Sends the devices you have queued, with their household and specifications, to Prospect. Nothing is sent until you press Send.

Connect

A spreadsheet of households

Upload a spreadsheet of households (and their kits and payments) in one go. Find it under Customers → Import.

Connect

Cancel

A spreadsheet of devices

Upload a manufacturer file of devices. Find it under Inventory → Import.

Connecting Prospect: a name, the key Prospect gave you, and the technical defaults left as they are unless Prospect says otherwise.



TOKENS

Dashboard

Issue

Inventory

Devices

Customers

Record

Approvals

Payments

Accounts

Reports

SETTINGS

Prices

Messages

Business

Places

Device Types

Connect apps

Connect apps

Bring households in from a form, or send installed devices out to another system. Choose the system below and follow the steps — nothing here needs any technical knowledge, and **nothing is sent out until you press Send**.

Connected

Village registration form on

KoboToolbox (household registration) - brings data in

Last form received
none yet

Form questions set up
8 matched

Teach it my form

Test a form

What arrived

New address

Pause

Prospect on

Prospect (Tanzania) — send installed devices - sends data out

Waiting to send
1

Sent
0

Tried, not accepted
0

Send 1 waiting now

See what will be sent

Test the connection

Pause

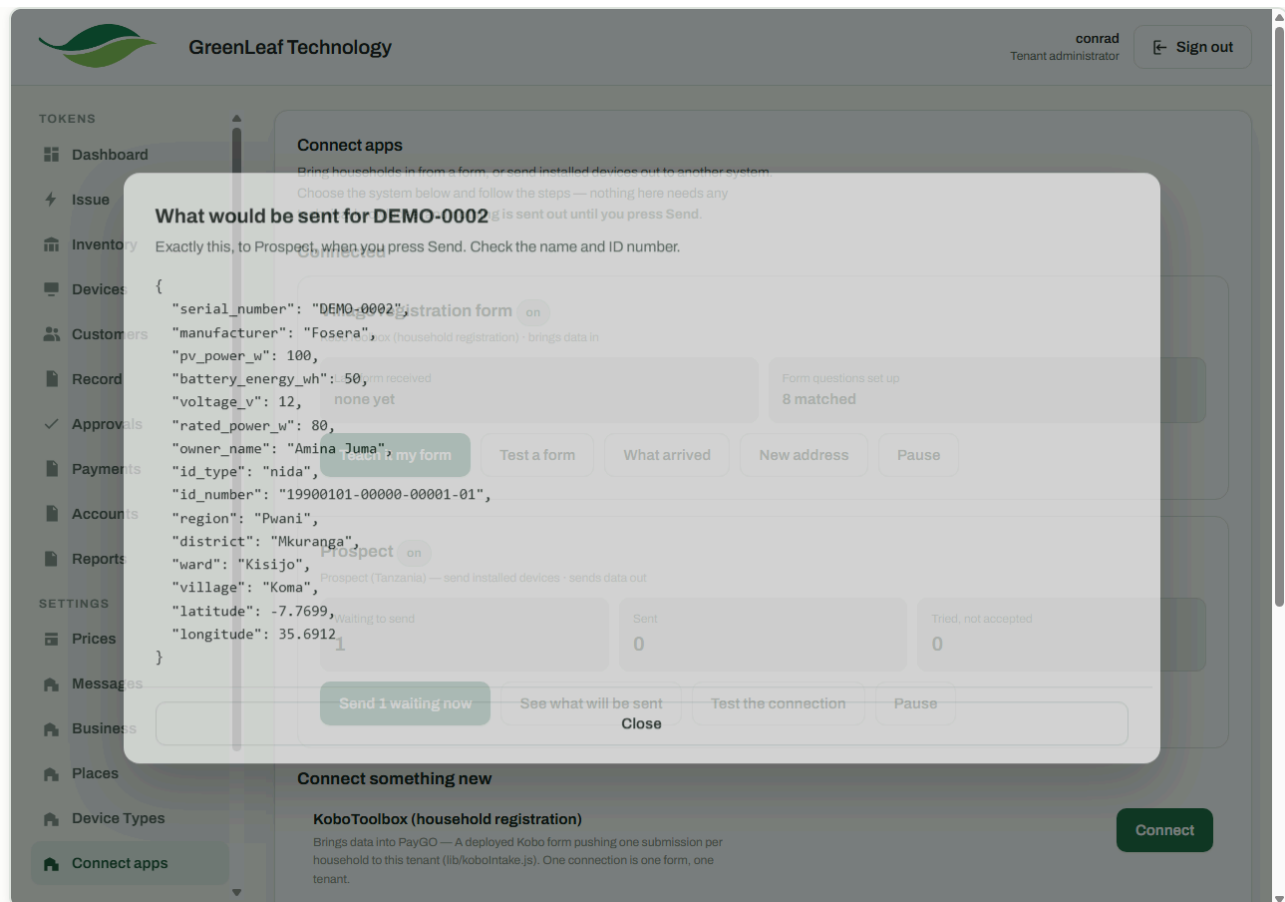
Connect something new

KoboToolbox (household registration)

Brings data into PayGO — A deployed Kobo form pushing one submission per household to this tenant (lib/kobointake.js). One connection is one form, one tenant.

Connect

The connected Prospect card — how many are waiting, sent, and tried-but-not-accepted, with Send, See what will be sent, Test and Pause.



"See what will be sent" — the exact JSON the next Send would submit for one queued device. Nothing is sent by opening this.

4. **Pause** stops a connection at once with no code; **Resume** needs one. **New address** replaces a form's address and keeps the old one working for a day while you change the form.
5. **Delete** (a code required) removes a connection completely — not just pausing it. Use it for one created by mistake or no longer wanted; anything it already received or sent stays on file, but the setup itself, and any address or key it generated, is gone for good. Prospect allows only one connection at a time — edit or delete the one you have rather than adding a second. A second Kobo form is different: that is a genuinely separate form, so Kobo has no such limit.

Nothing is ever sent out on its own. A spreadsheet of households or devices is uploaded from **Customers** → **Import** and **Inventory** → **Import** as before.

Prospect

Every seat that meets households

Getting a device's data ready for Prospect's own reporting. Nothing on this screen calls Prospect — sending is a separate, deliberate step on Connect apps; this screen selects which devices qualify, shows what each is still missing, and remembers what has already been chosen so the same device is never offered twice.

1. Every device holding a household is listed, marked **ready**, **missing** (with exactly what), or **queued**.
Type into the filter box — a serial, an identification number, a name, anything on the row — to narrow the list to what you are looking for.
2. **Queue** marks one device as selected. **Queue every ready device currently shown** does the same for everything the filter box currently narrows to, skipping anything already queued or still missing something.
3. **Release** un-queues a device. Safe until it has been sent: a device already sent to Prospect cannot be released (that would let it be sent twice). Sending itself is on Connect apps — either the usual **Send**, or **Sync now**, which finds and sends every ready household without you having to Queue each one by hand first.
4. **Resend** appears only for an administrator, only on a device already sent. Use it when that household's details changed afterwards — it does not resend on its own; it needs a fresh code, the same as Send.
5. **Fix** fills in one household's identification on the spot. For identification, GPS position, or both at once across many households, use **Fill in what's missing, in bulk** below: get the template, fill in only the cells that are blank on the real record, and upload it. A cell already on file is never overwritten, however the file states it.

People

Tenant administrator

Token officer

A named recovery agent (any role)

Staff accounts — how much of this screen you see depends on which of the three above you are.

- **People** — every seat, its role and whether it is active. An administrator may deactivate anyone but themselves, and may not create or edit a seat senior to their own.
- **Add someone** — a name, phone and email are required for a new seat (so it can be reached, once there is a reason to).
- **Where they work** (sales agent / technician only) — region, district, ward, village: optional, not checked against the Places list, and used when a report elsewhere on the platform asks who sold or installed a particular kit.
- A recovery agent may reset one colleague's password without being an administrator — nothing else on this screen opens for that permission alone.

Plan

Tenant administrator

Token officer (reads)

What this business has bought, what it has used this month, and what that comes to.

- **Your plan** — the plan itself, this month's usage against it, and the billing dates.
- **Ask to change plan** — request a move to a different plan; nothing changes until the vendor approves it.
- **Subscription / bill for a month** — the billing state and a reconstructed bill for any past month, derived from the token ledger rather than stored, so it can always be reproduced.
- **Limits by role** — this tenant's own per-role daily ceilings, where narrower than the plan's own default.

GreenLeaf Technology Solutions Co. Ltd — PayGO Token Service. This manual is a living document — see CLAUDE .md's own instruction that it is updated in the same change as any screen it describes. Questions about a specific seat's own limits: ask your tenant administrator.