WaoBiz

Shortlets Module

Short-Term Rental Management — User Guide

Channel management, unified guest inbox, dynamic pricing, automated check-in with smart-lock integration, turnover operations and owner payouts — a standalone WaoBiz module.

Version 2.0 · August 2026
WaoBiz POS Platform · WaoShortlet

Table of Contents

1. **Introduction**
   - What the Shortlets module does
   - A standalone module
   - Who it is for
2. **Getting Started**
   - Prerequisites
   - Permissions
   - Navigation
   - First-run setup checklist
3. **Units & Owners**
   - Adding a unit
   - Unit fields explained
   - Property owners
   - Commission precedence
   - Taking a booking
4. **Channel Manager**
   - How calendar sync works
   - Connecting a channel
   - Mapping a listing
   - The export feed
   - Double-booking prevention
   - Rate push
   - Reading the sync log
5. **Availability Calendar**
6. **Dynamic Pricing**
   - The three pricing layers
   - Rule types
   - Running a reprice
   - Price floors and ceilings
   - Manual overrides
7. **Guest Inbox**
   - How threads are grouped
   - Replying by channel
   - Thread status

8. **Automations & Check-In**
   - Triggers and offsets
   - Template variables
   - Smart locks & access codes
   - The message queue
9. **Cleaning & Operations**
   - Automatic turnover tasks
   - The task board
   - Team and auto-assignment
10. **Financial Reporting**
   - ADR, RevPAR and occupancy
   - Why revenue is split by night
   - Channel mix
11. **Owner Payouts**
   - Generating a statement
   - Statement lifecycle
   - Reading the breakdown
12. **Settings Reference**
13. **Scheduled Jobs**
14. **Troubleshooting**
15. **Current Limitations**

1. Introduction

1.1 What the Shortlets module does

The Shortlets module (WaoShortlet) turns WaoBiz into a short-term rental management system for serviced apartments, holiday lets and Airbnb-style properties. It covers six areas:

Feature	What it does
Channel Manager	Keeps calendars in step across Airbnb, Booking.com, Vrbo and your direct booking page so the same night is never sold twice.
Unified Inbox	Collects guest messages from every channel into one conversation list.
Dynamic Pricing	Adjusts nightly rates by occupancy, lead time, day of week and season — with an optional AI refinement pass.
Automated Check-In	Sends directions, house rules, WiFi and the door code on a schedule before arrival.
Cleaning & Operations	Creates and assigns a turnover clean the moment a guest checks out.
Financial Reporting	Tracks ADR, RevPAR, occupancy and margin, and produces owner payout statements.

1.2 A standalone module

Shortlets owns its own properties and reservations. It does **not** require the hotel module: you can license Shortlets on its own, HMS on its own, or both.

- A property is a **shortlet unit** with its own code, nightly rate and street address.
- A stay is a **shortlet booking** — created, priced, paid and checked out inside this module.
- Revenue still reaches Sales and Reports: the first payment on a booking posts a sale into your central ledger, exactly as the Car Wash and Laundry modules do. No separate reconciliation.

Unit + **Booking** → **Payment** → **Sales & Reports**

If you also run a hotel: HMS and Shortlets are separate inventories. A hotel room and a serviced apartment are set up independently, each with its own calendar and channel connections. HMS has its own channel manager for the same purpose — see the HMS guide.

Upgrading from an earlier version? Shortlets used to sit on top of HMS rooms. The migration converts every registered room into a shortlet unit and every HMS-backed stay into a shortlet booking, preserving the link to any revenue already posted. Nothing needs re-entering.

1.3 Who it is for

- **Operators** managing a handful to a few hundred apartments across multiple platforms.
- **Managing agents** who run properties on behalf of landlords and must issue owner statements.
- **Hotels adding apartments** — run both from one dashboard as separate inventories.

2. Getting Started

2.1 Prerequisites

Only one: **Shortlets enabled** on your subscription plan. Everything else — properties, rates, bookings — is created inside the module.

> **Set a nightly rate on every unit.** Dynamic pricing scales that rate, so a unit left at zero is skipped by the rate calendar entirely. The units list flags any unit in that state.

2.2 Permissions

Shortlets has ten permissions, assignable per role under **User Management** → **Roles** in the **Shortlets** group.

Permission	Grants access to
`shortlet_view`	Read-only access to every shortlet screen. Required for all other permissions to be useful.
`shortlet_manage_bookings`	Take bookings, check guests in and out, record payments, block dates.
`shortlet_manage_units`	Create and edit units and property owners.
`shortlet_manage_channels`	Connect channels, map listings, trigger syncs and rate pushes.
`shortlet_manage_pricing`	Create pricing rules, run repricing, set manual rate overrides.
`shortlet_manage_inbox`	Reply to guests and change thread status.
`shortlet_manage_automations`	Edit message templates, connect smart locks, issue and revoke door codes.
`shortlet_manage_tasks`	Manage cleaning tasks and the operations team.

`shortlet_manage_payouts`	Generate, approve and pay owner statements.
`shortlet_settings`	Change module-wide settings.

> **Suggested role split:** give cleaners `shortlet_view` + `shortlet_manage_tasks` only. Front-desk staff need `shortlet_manage_bookings`. Keep `shortlet_manage_payouts` restricted to finance — it controls money owed to landlords.

2.3 Navigation

All screens live under **Dashboard** → **Shortlets**:

Screen	Purpose
Dashboard	Live occupancy, month KPIs, today's arrivals and departures, channel health.
Bookings	Every stay. Take a direct booking, check in and out, record payments.
Units	Add apartments and set rate, address, owner, WiFi and lock details.
Owners	Landlords, their bank details and commission rates.
Calendar	Availability grid across all units and channels.
Channel Manager	OTA connections, listing mappings, imported stays, sync log.
Guest Inbox	All guest conversations.
Dynamic Pricing	Rate calendar and pricing rules.
Automations	Message templates, send queue, access codes.
Cleaning & Ops	Task board and team.
Owner Payouts	Statements and payment status.
Reports	ADR, RevPAR, occupancy, margin, channel mix.
Settings	Module defaults.

2.4 First-run setup checklist

1. Open **Settings** and set your brand name, unit label and currency.
2. Add your **property owners** with bank details and commission rates.
3. Add your **units** — each needs a unit code and a nightly rate; assign an owner and fill in the address, WiFi and directions.
4. Add your **cleaning team** under Cleaning & Ops so turnovers auto-assign.
5. On **Automations**, click *Install default journey* for a working five-message guest sequence.
6. On **Dynamic Pricing**, click *Add starter rules*, then run a reprice.
7. On **Channel Manager**, connect each OTA and map every unit to its listing.
8. Optionally, under **Settings** → **Smart locks**, connect your lock vendor so door codes reach the keypad without you keying them in.

3. Units & Owners

3.1 Adding a unit

Go to **Shortlets** → **Units** → **Add unit**.

1. Give it a **unit code** — a short identifier unique to your business, e.g. "LEK-01".
2. Set the **nightly rate**. This is what dynamic pricing scales; a unit at zero is skipped by the rate calendar.
3. Add a **display name** guests and staff will recognise, e.g. "Lekki 2-Bed Apartment".
4. Assign the **owner** (optional, but required before you can produce a payout statement).
5. Enter the **address** — each shortlet has its own, and it is used in guest directions and on cleaner task cards.
6. Fill in WiFi, directions, house rules and check-in notes. These feed the automated guest messages.
7. Choose the **lock provider**, then pick the specific lock from the list. See section 8.3.

> **Weekend and seasonal rates:** the nightly rate can carry per-weekday overrides, so a Friday can start higher than a Tuesday before any demand rules apply. Dynamic pricing then adjusts from there.

3.2 Unit fields explained

Unit code	Short identifier, unique per business. Shown wherever there is no display name.
Display name	Friendly name shown throughout the module in place of the unit code.

Property type	Free text ("Studio", "2-Bedroom", "Penthouse"). For your own grouping; nothing depends on it.
Nightly rate	The base price dynamic pricing scales. Supports per-weekday overrides.
Max guests	Occupancy cap shown on the booking form.
Min / max nights	Overrides the module default for this unit. Leave blank to inherit.
Cleaning fee	Becomes the default cost on auto-generated turnover tasks, and is deducted on the owner statement.
Security deposit	Recorded for reference on the unit profile.
Commission %	Your management fee for this unit. Overrides the owner's rate and the module default.
WiFi name / password	Injected into guest messages via template variables.
Smart lock	Manual (keypad) or a connected vendor. Picking a vendor lets you select the specific lock. See section 8.3.
Fixed door code	Leave blank to generate a unique code per stay. Fill it in only if the lock has one permanent code.
Directions	Free text used by the `{{directions}}` variable in pre-arrival messages.

> **Leave the fixed door code blank** unless the lock genuinely cannot change codes. Per-stay codes expire at checkout, so a past guest cannot re-enter.

3.3 Property owners

Owners are the landlords whose properties you manage. Add them under **Shortlets → Owners**. Beyond contact details, record their **bank name, account number and account name** — these print on the payout statement.

Owners are automatically linked into the central **Contacts** ledger, so they appear alongside your other customers and suppliers rather than living in a silo.

3.4 Commission precedence

When a payout is generated, the commission rate is resolved from the most specific setting available:

Unit rate → **Owner rate** → **Module default**

So a unit with 15% set overrides an owner set to 20%, which in turn overrides the 20% module default. Leave the unit and owner fields blank to simply use the default everywhere.

3.5 Taking a booking

Bookings live under **Shortlets** → **Bookings**. Most stays arrive on their own through the channel manager; this is for the ones you take directly — a phone enquiry, a repeat guest, a walk-in.

1. Click **New booking** and choose the unit and dates.
2. The total prices itself as you type — night by night, so weekend and seasonal rates are already applied. You see what the guest will actually pay before committing.
3. Add the guest's name and contact details. They are linked into your central Contacts ledger.
4. Optionally take a payment there and then; it posts straight into Sales.

> **The availability guard runs here too.** If those dates are already taken — by another booking, an imported OTA hold, or a maintenance block — the booking is refused and the message names what is blocking it. Same-day turnover is allowed: a guest departing on the 5th does not clash with one arriving on the 5th.

Through the stay

Action	What it does
Check in	Marks the guest in house.
Check out	Creates the turnover cleaning task, expires the door code and withdraws it from the lock.
Take payment	Records it against the booking. The first payment posts the sale into Sales and Reports; later ones update it.
Cancel	Stops queued guest messages and revokes the door code. The booking stays on record.

> **Moving a booking's dates re-prices it** and re-runs the availability check, so you cannot accidentally move a stay onto occupied nights or leave a stale total behind. Queued guest messages are rescheduled to match.

Blocking dates without a booking

For maintenance, an owner stay or a deep clean, add a **block** instead of a dummy booking. Blocks appear on the calendar, feed the availability guard, and go out on your channel feeds so the OTAs stop selling those nights. A block that would overlap a live booking is refused.

4. Channel Manager

4.1 How calendar sync works

Sync uses **iCal**, the calendar format Airbnb, Booking.com and Vrbo all publish. It needs no API approval and works with a standard host account. Sync runs in both directions:

Direction	What happens
Import (OTA → WaoBiz)	On a schedule, WaoBiz fetches each listing's iCal URL and records every reservation as a calendar hold. Those holds then block the dates everywhere else. Use *Sync now* for an immediate pull.
Export (WaoBiz → OTA)	WaoBiz publishes a feed URL per listing. The OTA polls it on their own schedule and blocks the dates it contains.

Guest books on Airbnb → WaoBiz imports the hold → Booking.com and your direct site both show those nights as taken

4.2 Connecting a channel

1. Go to **Channel Manager** → **Add channel**.
2. Select the platform and give the connection a display name, e.g. "Airbnb — main account".
3. Leave sync mode as **iCal** unless you have API credentials.
4. Set the **channel commission %** the platform charges you. This feeds the channel mix report.
5. Optionally set a **rate markup %** so the price pushed to that channel nets out to your direct rate after their commission.

6. Set the **sync interval**. Thirty minutes suits most operators.

4.3 Mapping a listing

A channel connection alone syncs nothing. You must map each unit to its listing on that channel.

1. Click **Map listing**.
2. Choose the channel and the unit.
3. Paste the **iCal import URL**, copied from the OTA's calendar export settings.
4. Save. WaoBiz immediately generates an export feed URL for this listing.

Where to find the import URL on each platform:

Platform	Location
Airbnb	Calendar → Availability → Connect to another website → Export calendar
Booking.com	Rates & Availability → Sync calendars → Export
Vrbo	Calendar → Import/Export → Export calendar

> **A mapping without an import URL is one-way.** The listing card shows a warning when this is the case. Without it, that channel's bookings will *not* block your calendar and you can be double-booked.

4.4 The export feed

Each mapped listing gets a feed URL of the form `https://your-domain/api/shortlets/ical/<token>`. Copy it from the listing card and paste it into the OTA's calendar *import* setting.

The feed includes:

- Your direct bookings for that unit
- Holds from **every other channel**, so each OTA sees the complete picture
- Manual maintenance and owner-stay blocks

> **On feed security:** the URL is deliberately unauthenticated, because OTAs fetch calendar feeds anonymously — the long random token in the URL *is* the credential. For that reason the feed carries only busy/free date ranges. It never contains guest names, contact details or rates. If a URL is ever exposed, use *Edit → regenerate token* to invalidate it, then re-paste the new URL at the OTA.

4.5 Double-booking prevention

Every attempt to create a booking — whether from the dashboard or your public booking page — checks three sources before it is allowed:

1. Existing **local bookings** for that unit
2. Imported **channel holds** from every connected OTA
3. Manual **unavailability blocks**

If any of the three overlaps the requested dates, the booking is rejected with a message naming the blocker.

> **Same-day turnover is allowed.** A guest departing on the 5th does not conflict with one arriving on the 5th. Only genuinely overlapping nights collide.

4.6 Rate push

Use **Push rates** on a channel to publish your computed nightly rates, with that channel's markup applied.

> **iCal carries no pricing.** For iCal channels, WaoBiz computes and records what the rates should be, but you must apply them in the platform's own extranet — the module tells you this plainly rather than pretending the push succeeded. Automatic rate publishing requires an API-mode channel with credentials.

4.7 Reading the sync log

The **Sync log** tab records every attempt. Columns read `+added / ~updated / −removed`.

Status	Meaning

success	All listings on the channel synced.
partial	Some listings synced, others failed. The message names them.
failed	No listing synced. Usually a bad or expired feed URL.

A **conflict count** means an imported reservation overlaps something you already had. The import still records it — the OTA booking exists in the real world whether or not it clashes — but you should resolve the clash immediately.

5. Availability Calendar

Shortlets → Calendar shows a 30-day grid: one row per unit, one column per night. Each booked night is colour-coded by its source, so you can see at a glance which platform is filling your calendar.

Colour	Source
Blue	Direct booking taken in WaoBiz
Rose	Airbnb
Indigo	Booking.com
Grey	Manually blocked (maintenance, owner stay)

Free nights display the current nightly rate in thousands, drawn from the dynamic pricing calendar. If those numbers are missing, run a reprice (section 6.3). Use the arrows to move a month at a time.

6. Dynamic Pricing

6.1 The three pricing layers

HMS base rate → **Demand rules** → **AI pass (optional)** → **Rate calendar**

Layer	Behaviour
Base rate	The unit's own nightly rate, including any per-weekday overrides set on it.
Demand rules	Deterministic and free. They stack multiplicatively, highest priority first. Always run.
AI pass	Optional. Reads the computed signals and refines the result against market patterns. Consumes AI credits.

Every night in the calendar stores its base rate, the rules that fired, and a plain-English reason, so you can always see why a price is what it is.

6.2 Rule types

Type	Triggers on	Typical use
Occupancy	Portfolio occupancy % for that date	+25% when above 80% booked
Lead time	Days between today and the stay	−12% inside 3 days to fill late gaps
Day of week	Selected weekdays	+20% Friday and Saturday
Season	A date range	+40% over the December holidays
Event	A date range	+60% during a nearby conference

Gap night	Orphan nights between two bookings	−20% on a single stranded night
Length of stay	Length of the free run available	Discount long empty stretches

Rules apply to all units by default, or you can scope one to a single unit. **Priority** controls order — higher runs first.

> **Start with the presets.** *Add starter rules* installs six sensible rules covering weekends, high and low demand, last-minute and early-bird discounts, and orphan nights. Tune from there rather than starting blank.

6.3 Running a reprice

Two buttons on the Dynamic Pricing screen:

- **Reprice (rules)** — free, instant, fully explainable. Use this routinely.
- **Reprice with AI** — adds the AI refinement. Requires AI pricing enabled in Settings and available credits.

Repricing covers your pricing horizon (90 days by default). Filter to one unit first to reprice just that unit.

> **AI never blocks pricing.** If credits run out or no AI provider key is configured, the module silently falls back to rules-only output. Prices are never left unset because an AI service was unavailable.

6.4 Price floors and ceilings

Settings define a **floor** and **ceiling** as multiples of the base rate (0.6× and 2.5× by default). No computed price can fall outside them — the bounds cap both the rules and the AI. This is your protection against a misconfigured rule discounting a unit to nothing.

6.5 Manual overrides

A price you set by hand becomes the **final rate** and always wins over the suggestion. Overridden nights are tagged `manual` in the calendar. Clear the override to return that night to automatic pricing.

7. Guest Inbox

7.1 How threads are grouped

Shortlets → Guest Inbox collects every guest conversation into one list, whatever channel it arrived on. Messages are matched to an existing thread in this order:

1. The booking they relate to
2. The platform's own thread ID
3. The guest's phone number or email

This is why a guest who messages you on WhatsApp and later books does not end up with two separate threads. Automated messages are mirrored into the thread too, so you always see the complete history with a guest.

7.2 Replying by channel

Thread source	Reply behaviour
WhatsApp / SMS / Direct	Sends over WhatsApp, falling back to SMS automatically.
Email	Sends by email using your configured mail transport.
Airbnb / Booking.com / Vrbo	Saved to the thread and flagged, because these platforms provide no outbound messaging API. Send the message in their extranet.

> **On OTA replies:** the module stores the message and clearly marks it undelivered rather than silently discarding it. Your record of the conversation stays complete even though the send must happen elsewhere.

7.3 Thread status

Threads are `open`, `pending` or `closed`. Opening a thread clears its unread badge. Close a thread once the guest's question is resolved; it can always be reopened.

8. Automations & Check-In

8.1 Triggers and offsets

An automation is a message template that fires at an offset from a point in the booking. A **negative offset means before**, positive means after.

Trigger	Anchor	Example
Booking confirmed	The moment the booking is created	Confirmation, sent immediately
Before check-in	Arrival date/time	−24h directions, −3h door code
Check-in day	Arrival date/time	Arrival-day welcome
Mid-stay	Midpoint of the stay	"Everything OK?" check-in
Before check-out	Departure date/time	−12h checkout reminder
After check-out	Departure date/time	+3h thank-you and review request

Messages are queued the moment a booking is created and sent as each falls due. Cancelling a booking cancels its pending messages automatically.

> **Install the default journey** to get a working five-message sequence covering confirmation, pre-arrival directions, the door code, a checkout reminder and a thank-you. Edit the wording to suit your brand.

8.2 Template variables

Wrap variables in double braces. The editor has a clickable list to insert them.

Variable	Resolves to

`{{guest_name}}`	The guest's name, or "there" if unknown
`{{unit_name}}`	The unit's display name
`{{address}}`	The unit's street address
`{{directions}}`	The unit's directions text
`{{house_rules}}`	The unit's house rules
`{{checkin_notes}}`	The unit's check-in notes
`{{access_code}}`	The door code — see 8.3
`{{wifi_name}}` / `{{wifi_password}}`	WiFi credentials
`{{check_in_date}}` / `{{check_out_date}}`	Formatted stay dates
`{{checkin_time}}` / `{{checkout_time}}`	Property check-in/out times from HMS settings
`{{nights}}`	Number of nights booked
`{{brand_name}}`	Your shortlet brand name

8.3 Smart locks & access codes

Turn on **Include the door code** for any template that needs to hand over access. Codes are issued per stay and expire at checkout, so a past guest cannot re-enter.

> **Codes are withheld until shortly before arrival.** Settings define an *access code lead time* (3 hours by default). Before that window, `{{access_code}}` renders as a placeholder instead of the real code — so a confirmation message written weeks ahead cannot leak the door code weeks ahead.

Connecting a lock vendor

Go to **Settings** → **Smart locks** and connect an account. Five platforms are supported:

Provider	Notes

TTLock	Needs a Client ID and secret from the TTLock Open Platform, plus the account's own username and password.
Nuki	A single Web API token, generated under Nuki Web → Account → API with keypad permissions.
igloohome	Client ID and secret from the igloodeveloper portal.
Seam	One API key. Seam brokers Yale, Schlage, August, SmartThings and many others — use it if your brand is not listed separately.
Tuya / Smart Life	Access ID, secret and the data centre your Tuya project was created in. Covers a long tail of locks sold under other brand names.

> **Credentials are verified before they are saved.** If a key is wrong you find out in Settings, not at a guest's arrival. They are encrypted at rest and never shown again after saving.

Once an account is connected, open a unit and pick its lock from the dropdown. Each unit points at one specific lock.

What happens on each stay

1. A code is generated for the booking, valid from six hours before arrival to four hours after checkout — so an early arrival or slow departure is not locked out.
2. The code is pushed to the lock.
3. It is sent to the guest at the time your automation specifies.
4. At checkout the code is withdrawn from the lock, not merely marked used.

> **The guest always gets a code.** Pushing it to the lock is a separate, best-effort step. If a vendor is unreachable, the code is still generated, stored and sent — the Access codes tab simply shows it did not reach the lock, so you can set it by hand. A booking never fails because a lock API is down.

The **On device** column on the Access codes tab tells you where each code stands:

Status	Meaning
on lock	The vendor accepted the code. Nothing to do.

manual	No integration for this unit — set the code on the keypad yourself. Not an error.
not sent	The vendor rejected it or was unreachable. Retried automatically up to five times; set it by hand if it keeps failing.

Two vendors set their own code. igloohome locks work offline and derive their PIN from an algorithm built into the lock, so igloohome's code is used rather than ours. Nuki requires six digits not starting with 1 or 2, so a non-compliant code is adjusted. In both cases the guest is sent the code that actually opens the door.

Fixed keypad code: if a property has one permanent code that cannot be changed, put it in the unit's *fixed door code* field. It will be reused for every stay instead of generating a new one — convenient, but it means a past guest's code still works.

8.4 The message queue

The **Message queue** tab shows every scheduled send and its outcome.

Status	Meaning
pending	Queued, not yet due.
sent	Delivered successfully.
failed	Failed three times. Usually a missing or invalid guest phone number.
skipped	Not sent because automated messaging is off in Settings.
cancelled	The booking was cancelled or the template deleted.

Send due now dispatches everything currently due without waiting for the scheduled job — useful when testing a new template.

9. Cleaning & Operations

9.1 Automatic turnover tasks

When a guest checks out, a turnover cleaning task is created automatically, with a seven-point checklist covering linen, bathroom, kitchen, floors, restocking, damage check and lock-up.

Guest checks out → Task created & assigned → Cleaner works the checklist → Marked complete → Cost flows to the owner statement

The due time is set from your **turnover window** (4 hours after checkout by default), with one exception:

> **Same-day arrivals are handled automatically.** If another guest arrives at that unit the same day, the task is raised to `urgent` and its due time pulled back to *their* arrival — not the default window.

9.2 The task board

Four columns track work: Unassigned, Assigned, In progress, Completed.

- Overdue tasks are outlined in red.
- Auto-created tasks carry an `auto` tag.
- Tasks already charged on a payout show `billed` and cannot be deleted.
- Cleaners tick checklist items directly on the card as they work.

Besides cleaning, you can raise **maintenance**, **inspection**, **restock** and **other** tasks manually at any time.

9.3 Team and auto-assignment

Add cleaners and maintenance staff on the **Team** tab. New turnover tasks are assigned to whichever active cleaner has the **fewest open jobs**, spreading work evenly without manual

dispatch.

If no cleaner is configured, tasks are still created but left unassigned. Removing a team member returns their open tasks to the unassigned pool rather than deleting them.

> **Task cost feeds the owner statement.** A task's cost defaults to the unit's cleaning fee. Completed task costs are deducted from the owner's payout for that period, so keep them accurate.

10. Financial Reporting

10.1 ADR, RevPAR and occupancy

Metric	Formula	What it tells you
ADR Average Daily Rate	Room revenue ÷ nights *sold*	What you earn per night actually booked. Measures pricing.
RevPAR Revenue Per Available Night	Room revenue ÷ nights *available*	Earnings across your whole portfolio, occupied or not. Measures overall performance.
Occupancy	Nights sold ÷ nights available	How full you are.
Net margin	Gross revenue − operating costs	What remains after cleaning and maintenance.

> **ADR vs RevPAR:** a high ADR with low RevPAR means you charge well but sit empty — consider discounting. RevPAR is the single best measure of portfolio health because it combines rate and occupancy.

10.2 Why revenue is split by night

Revenue is attributed to the nights actually stayed, **not to the date the booking was made**. A stay from 28 June to 3 July contributes its June nights to June and its July nights to July.

This is what keeps ADR, RevPAR and occupancy comparable month to month. Attributing the whole booking to one month would inflate that month and leave a hole in the next.

10.3 Channel mix

The **Channel mix** tab breaks revenue down by source, showing gross revenue, the platform's commission and your net. Bookings traced to an OTA show that platform; everything else counts

as Direct.

> **Use this to justify your direct channel.** Comparing net revenue per booking across channels usually shows direct bookings are worth materially more — useful when deciding how much to invest in your own booking page.

The **By unit** tab ranks every apartment by revenue with its own ADR, RevPAR, occupancy and margin, and exports to CSV. The **6-month trend** tab charts revenue, margin and occupancy over time.

11. Owner Payouts

11.1 Generating a statement

1. Go to **Shortlets → Owner Payouts → Generate statement**.
2. Choose the owner and the period. It defaults to last month.
3. Click **Generate**. The statement is created as a `draft`.

The calculation is:

```
Gross booking revenue   (pro-rated to nights within the period)
  − Management commission (unit rate → owner rate → default)
  − Cleaning costs        (completed tasks in the period)
  − Maintenance costs
  − Other costs
  ± Adjustments
  ──────────────────────────
  = Net payable to owner
```

11.2 Statement lifecycle

Status	Meaning	Can it change?
`draft`	Calculated but not agreed. Regenerating replaces its lines.	Yes, freely
`approved`	Agreed. Its task costs are locked as billed.	Locked
`paid`	Money sent. Cannot be deleted, only cancelled.	Locked
`cancelled`	Voided. Its task costs are released back to unbilled.	—

> **Why costs are locked on approval:** once a cleaning cost has been charged on one statement it is marked billed, so the next period cannot charge it again. Cancelling a statement releases those costs so they can be picked up correctly elsewhere.

11.3 Reading the breakdown

View statement opens the full document: header, the owner's bank details, and every backing line — each booking with its nights and pro-rated revenue, each commission charge, each cleaning and maintenance cost.

It shows nights sold, booking count and the period ADR, and prints cleanly for sending to the landlord. Because every figure traces to a specific booking or task, an owner query can always be answered precisely.

12. Settings Reference

Setting	Effect
Brand name	Title shown across shortlet screens and in guest messages via `{{brand_name}}`.
Unit label	Replaces the word "Room" with your term, e.g. "Apartment".
Check-in / check-out time	Your property-wide clock, used by the `{{checkin_time}}` and `{{checkout_time}}` template variables.
Currency symbol	Symbol used in rates, reports and statements.
Default min / max nights	Stay limits when a unit does not set its own.
Auto-create cleaning	Master switch for automatic turnover tasks at checkout.
Turnover due within	Minutes after checkout that a turnover clean is due. Overridden by a same-day arrival.
Send automated messages	Master switch for the guest journey. Off leaves messages queued but undelivered.
Pre-arrival lead	Default hours before arrival for pre-arrival messaging.
Release door code	Hours before arrival that `{{access_code}}` resolves to a real code.
Smart locks	Connect a lock vendor account so codes reach the keypad automatically. See section 8.3.
Enable AI pricing	Turns on the AI refinement pass. Rules run regardless.
Apply rates automatically	When on, suggested rates become live without manual confirmation.
Price floor / ceiling	Multiples of the base rate that bound every computed price.
Pricing horizon	How many days ahead repricing covers.
Default commission %	Fallback management fee when neither unit nor owner sets one.

Payout cadence	Your statement rhythm: weekly, fortnightly, monthly or quarterly.

13. Scheduled Jobs

Three background jobs keep the module current. They run automatically on the hosted platform.

Job	Schedule	What it does
Channel sync	Daily, 02:00	Imports every active channel calendar, respecting each channel's own sync interval.
Guest messages	Daily, 05:00	Delivers automated messages that have come due, and retries any door code that failed to reach its lock.
Turnover sweeper	Daily, 04:30	Creates turnover tasks for recent checkouts that do not have one — a safety net for bookings closed outside the app.

Safe to re-run. All three jobs are idempotent: repeated runs never duplicate a reservation, send a message twice, or create two cleaning tasks for one checkout.

Sync frequency and double-booking risk. These schedules are once-daily because Vercel's Hobby plan only permits daily cron jobs. That is fine for turnover tasks and guest messages, but it means an OTA booking may not be imported for up to 24 hours — a window in which the same night could be sold again on another channel. If you sell the same units across several platforms, either upgrade the hosting plan and raise these to hourly (change the three entries in `vercel.json`), or press *Sync now* on the Channel Manager before confirming a direct booking.

Each has a manual equivalent in the interface — *Sync now*, *Send due now*, and creating a turnover from a booking — so you never have to wait for the schedule when testing.

14. Troubleshooting

A unit does not appear in the rate calendar

Its nightly rate is zero. Dynamic pricing scales that rate, so with nothing to scale the unit is skipped. Set a rate on the unit and reprice. The units list flags any unit in this state.

A door code did not reach the lock

Check the **On device** column on **Automations** → **Access codes**:

- `manual` — no vendor is connected for that unit, or no lock was picked on the unit. Expected; set the code on the keypad.
- `not sent` — the vendor rejected it or was unreachable. It retries automatically up to five times. If it keeps failing, test the connection under **Settings** → **Smart locks**; a lock that is offline or out of battery is the usual cause.

Either way the guest still received a code — only the lock is out of step, so setting it by hand resolves it.

An OTA booking did not block my calendar

Check in order:

1. Is the unit mapped to a listing on that channel?
2. Does the mapping have an **iCal import URL**? The listing card warns when it does not.
3. Does the sync log show a recent successful import? A `failed` status usually means an expired feed URL — re-copy it from the OTA.
4. Some platforms only refresh their exported feed every few hours. Use *Import now* to force a pull.

Guests are not receiving messages

1. Confirm **Send automated messages** is on in Settings — otherwise sends are `skipped`.
2. Check the message queue for the failure reason. Most failures are a missing guest phone number on the booking.
3. Confirm WhatsApp or SMS credentials are configured on the platform.

The door code shows as a placeholder

Expected behaviour outside the release window. The code resolves only within *access code lead time* of arrival. Raise that value in Settings if you want codes sent earlier.

"Room is blocked by a <channel> booking"

The double-booking guard is doing its job — an OTA already holds those dates. Check the **Imported stays** tab. If the OTA booking was cancelled, run *Import now* to clear the stale hold.

An owner statement shows less revenue than expected

Revenue is pro-rated to nights *within* the period. A stay spanning the period boundary contributes only its in-period nights — the remainder appears on the adjacent statement. Also confirm every relevant unit is assigned to that owner.

AI repricing did nothing

1. Is **Enable AI pricing** on in Settings?
2. Does the business have AI credits remaining?
3. Is an AI provider key configured on the platform?

If any is missing, the module falls back to rules-only output and reports this in the result message.

15. Current Limitations

Stated here rather than left to be discovered.

Smart lock coverage

Five platforms are supported: TTLock, Nuki, igloohome, Seam and Tuya. Between them they cover most locks on the market — Seam alone brokers Yale, Schlage, August and others, and Tuya is the white-label behind many unbranded locks. A lock outside all five still works, you just set the code on the keypad yourself.

Two behaviours are worth knowing. **igloohome** locks are offline by design: they derive their PIN from an algorithm built into the lock, so igloohome's code is used rather than the one WaoBiz generated, and there is nothing to revoke — the PIN simply stops working at the end of its window. **Nuki** requires six digits not beginning with 1 or 2, so a non-compliant code is adjusted. In both cases the guest receives the code that actually opens the door.

Sync frequency

The background jobs run once daily, because the hosting plan permits daily schedules only. That is fine for turnover tasks and post-stay messages, but it means an OTA booking may go unimported for up to 24 hours, and a door code scheduled for three hours before arrival may be dispatched earlier in the day than intended.

If you sell the same units across several platforms, either raise the schedules on a paid hosting plan, or press *Sync now* on the Channel Manager before confirming a direct booking.

Booking concurrency

The availability check reads current bookings and then writes the new one. Two people booking the same unit for the same dates in the same instant could in principle both succeed.

Practical effect: very unlikely in normal operation, and the window is narrow. Cross-channel protection against OTA bookings is unaffected — that is the far more common risk and it is fully covered. A database-level constraint is the planned fix.

OTA outbound messaging

Airbnb, Booking.com and Vrbo do not expose outbound messaging APIs to third-party tools. Replies to those threads are recorded in WaoBiz but must be sent from the platform's own extranet. This is a platform restriction, not a WaoBiz gap.