WaoBiz

Shortlets Module

Short-Term Rental Management — User Guide

Channel management, unified guest inbox, dynamic pricing, automated check-in, turnover operations and owner payouts — built on the WaoBiz HMS engine.

Version 1.0 · August 2026
WaoBiz POS Platform · WaoShortlet

Table of Contents

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 How it relates to HMS

Shortlets is **an extension of HMS, not a separate system**. A shortlet booking *is* an HMS booking. This matters in practice:

- An apartment is an HMS **room**. Its property type is an HMS **room type**, which carries the nightly base rate.
- A stay is a normal sales transaction, so shortlet revenue appears in **Sales and Reports automatically** — no separate reconciliation.
- Shortlet tables only add what hotels do not need: a per-unit street address, an owner, WiFi, lock details, channel links.

HMS Room + Shortlet Unit profile = A bookable shortlet

> **Note:** Because Shortlets extends HMS, it is gated by the same `hms` module key. If HMS is on your plan, Shortlets is available — there is nothing extra to purchase.

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 system, since they share the same booking engine.

2. Getting Started

2.1 Prerequisites

Before registering your first shortlet unit you need, in this order:

1. **HMS enabled** on your subscription plan.
2. **At least one HMS room type** with a nightly base price set. Dynamic pricing scales this base — a room type with no base price is skipped entirely.
3. **At least one HMS room** created under that room type. This becomes your apartment.

> **Common first mistake:** Registering a unit whose room type has no base price. Everything else works, but the unit never appears in the rate calendar. Set the base price in **HMS → Room Types → Pricing** first.

2.2 Permissions

Shortlets has nine 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_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, 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. 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.
Units	Register apartments and set address, owner, WiFi and access 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. Register your **units**, assigning each to an owner and filling 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.

3. Units & Owners

3.1 Registering a unit

A shortlet unit is a profile layered on an existing HMS room. Go to **Shortlets** → **Units** → **Add unit**.

1. Pick the **HMS room**. Only rooms not already registered appear in the list.
2. Give it a **display name** guests and staff will recognise, e.g. "Lekki 2-Bed Apartment".
3. Assign the **owner** (optional, but required before you can produce a payout statement).
4. Enter the **address** — this is the field that most distinguishes a shortlet from a hotel room.
5. Fill in WiFi, directions, house rules and check-in notes. These feed the automated guest messages.
6. Set the lock provider and, if the property uses one fixed keypad code, the fixed door code.

> **Why the address matters:** hotel rooms share one building address, so HMS has no per-room address field. Shortlets are scattered across a city, so each unit carries its own. It is used in guest directions and on cleaner task cards.

3.2 Unit fields explained

Display name	Shown throughout the module in place of the bare room number.
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 an integrated provider. 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.

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 nightly price on the HMS room type, including any per-weekday pricing already configured there.
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 lock 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.

Provisioning depends on the unit's lock provider:

Provider	Behaviour
Manual / keypad	Uses the unit's fixed code, or generates a unique per-stay code you set on the keypad.

TTLock, Nuki, igloohome, Seam	The code is generated and recorded, and the record carries a note reminding you to set it on the lock. Direct lock-API provisioning is not yet connected — see section 15.

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".
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.
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.
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 HMS room type has no base price. Dynamic pricing scales the base rate, so with no base there is nothing to scale. Set it under **HMS** → **Room Types** → **Pricing** and reprice. The unit card also flags this.

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

Two capabilities are modelled but not fully connected. Both are stated plainly in the interface.

Smart lock provisioning

Access codes are generated, scheduled, delivered and expired correctly, and the lock provider is recorded per unit. However, WaoBiz does not yet call the lock vendors' APIs directly. For any provider other than *Manual*, the code record carries a note reminding you to set that code on the physical lock.

Practical effect: the guest journey works end to end; programming the lock is a manual step.

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.
