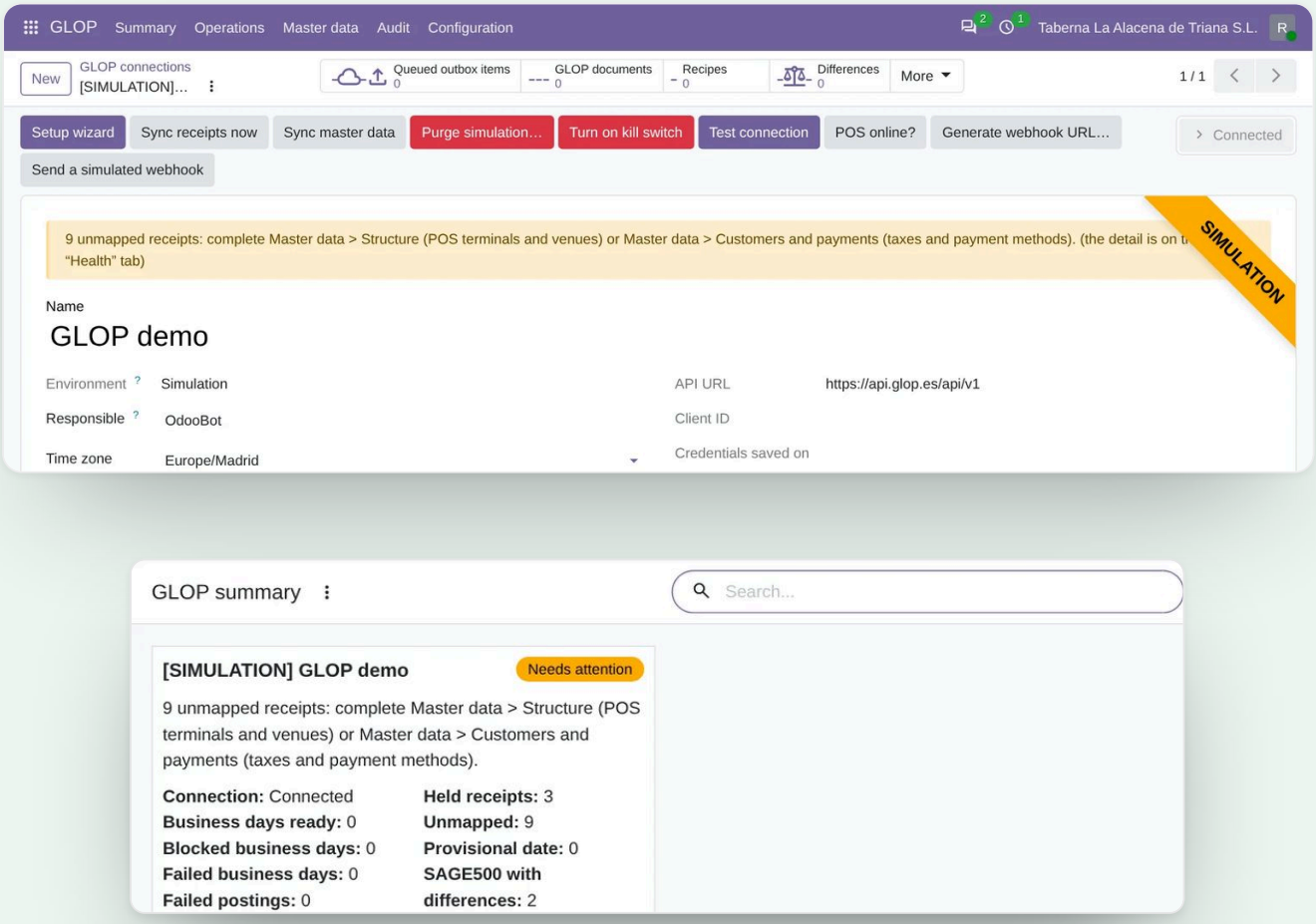
CONNECTOR GUIDE · POINT OF SALE AND HOSPITALITY

GLOP point-of-sale connector for Odoo

Sales from your GLOP point of sale reach Odoo accounting per terminal and business day, with VAT exactly as GLOP recorded it and without issuing a second invoice.

Odoo 17 · 18 · 19 · 20 | 14-day trial, no card | Edition of 20-11-2026



Real Odoo 20 screenshot. The [SIMULATION] GLOP demo connection with its access details and warnings, and the summary card for that same connection.

Contents

On the right, the page where each chapter starts. Menu and button names are those of Odoo in English.

1	Who it is for and what it solves	3
2	The connector in figures	4
3	Where you see it in your Odoo	5
4	What you need before you start	6
5	The FlexigoTech Cloud trial	7
6	Connecting your GLOP account step by step	8
7	From GLOP receipts to accounting: business days, entries and VAT	11
8	Payment methods, customers and credit sales	16
9	Master data: venues, terminals and catalogue	20
10	Purchasing and vendors with GLOP	23
11	Stock, recipes and counts	26
12	Orders from Odoo to GLOP tables and delivery	29
13	Reconciliation and accounting controls	32
14	Setup, simulation and monitoring	35
15	The data that goes out and comes back	39
16	What runs automatically	40
17	The connection options and their defaults	41
18	Security and personal data	44
19	Several connections and several companies	45
20	If something fails	45
21	Other ways of doing it	47
22	Plans and pricing	47
23	Frequently asked questions	48
24	Glossary	51
25	Checklist and contact	52

This guide describes version 20.0.1.2.1 of the module (Odoo branch 20.0, commit e888b59be). Everything in it comes from its published code, the platform's official documentation or real screenshots.

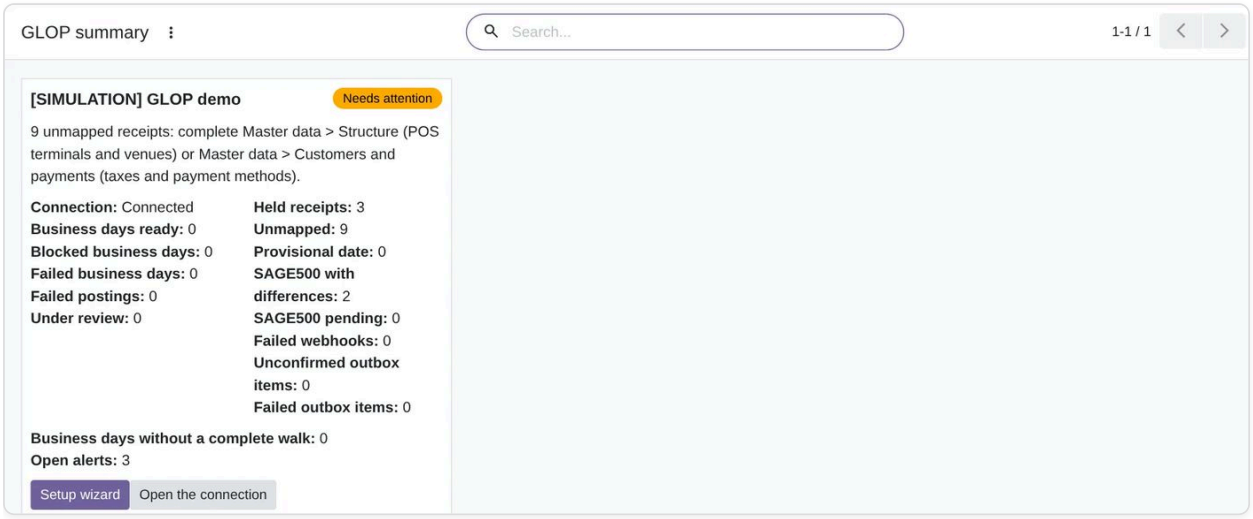
The screenshots come from a test Odoo 20 database, with a made-up company and data and no real connection to GLOP. The module also has a version for Odoo 17, 18 and 19.

1 Who it is for and what it solves

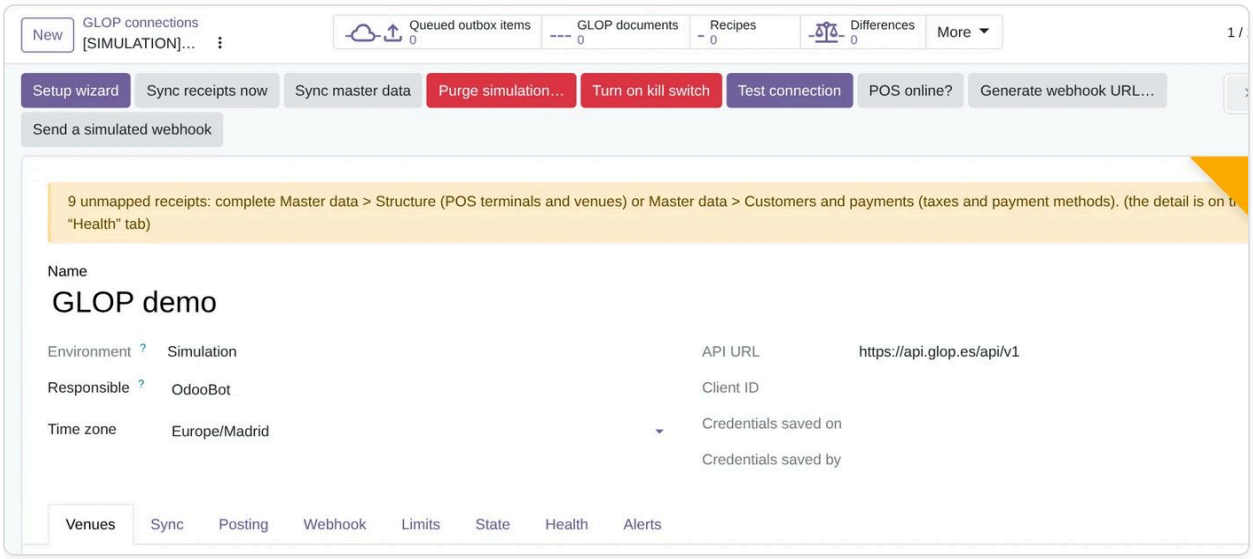
> The connector requires a FlexigoTech Cloud subscription, with a 14-day trial and no card needed. Without the trial or the subscription, the connector does not talk to GLOP.

GLOP tracks your sales by business day and Odoo keeps your accounts. Without the connector, someone has to manually enter each day's totals into Odoo, split the base and VAT by rate, note which sales were left on account and which were paid later, and check that no receipt is missing.

The connector reads GLOP's business days and generates one journal entry per terminal and business day, not per calendar day. Base and VAT come from GLOP's figures by document and VAT rate, without recalculating them. Payments go to the account that matches each payment method.



- **Real Odoo 20 screenshot.** The **GLOP summary** for the simulation connection, showing business days, unmapped receipts and open alerts.



- **Real Odoo 20 screenshot.** The simulation connection, with the unmapped receipts warning and the **Sync receipts now** and **Test connection** buttons.

What changes day to day

- Each business day reaches Odoo as one entry per terminal, with base and VAT exactly as GLOP recorded them.

- Credit sales leave the outstanding amount on the customer account and are reconciled when GLOP records the payment.
- If GLOP changes a receipt that has already been posted, the connector proposes an adjustment entry for a person to review.
- A receipt whose amounts do not add up is held with a code and an explanation, and the rest of its business day is still posted.
- In the **GLOP summary** you see, for each connection, the status, the business days that are ready, blocked or in error, the held or unmapped receipts and the open alerts.

What it does not do

- It does not create invoices, till receipts or point-of-sale orders in Odoo: GLOP still issues every receipt and invoice, and sends the VERI*FACTU records.
- It does not close or modify any GLOP tax document. It does not touch pre-receipts, does not close tables and does not add charges to a room.
- It does not show the VERI*FACTU status, the hash or the QR code, because the GLOP API does not provide them.
- It does not include the till closing or the Z report.
- It only supports mainland Spanish VAT and euros. It does not support equivalence surcharge, IGIC or IPSI.
- With periodic valuation stock, it does not post cost of sales.
- It does not decide accounts or taxes: it proposes, flags what is missing and confirms nothing on your behalf.

CONNECTOR GUIDE · GLOP

2 The connector in figures

4
Odoo versions: 17, 18, 19 and 20

10
scheduled tasks, all active on installation

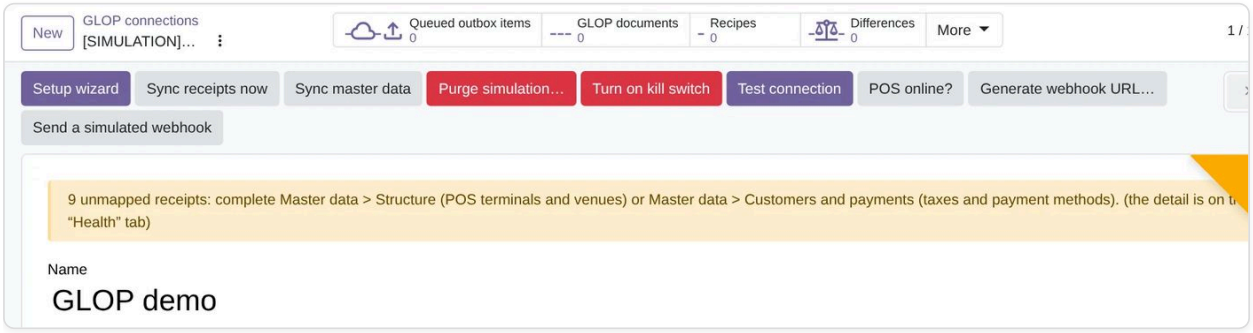
3
accounts on the Standard plan

10
accounts on the Pro plan

662
test functions across 55 files

The connector depends on Invoicing, Inventory, Purchase, Messaging, Analytic, the Spanish localisation and FlexigoTech Cloud, all from Odoo Community, so it works on both Community and Enterprise. It needs an Odoo installation where third-party modules can be added. In September 2026, the version for Odoo 19, still without FlexigoTech Cloud, read six paid documents from a real GLOP point of sale connected to GLOP's test cloud, and posted them in an Odoo 19 Enterprise: two entries and no invoices. On 14 November, version 1.2.0 connected to GLOP's real API in Odoo 17, 18, 19 and 20 through FlexigoTech Cloud's test environment. The only failure was reading families and price lists, which was fixed in the published 1.2.1; that test has not yet been repeated with 1.2.1. What Odoo writes to GLOP is switched off and has only been tested with the module's simulation and with isolated calls to a test account in GLOP's cloud, with no point of sale behind it. The screens in this guide come from that simulation.

3 Where you see it in your Odoo



- **Real Odoo 20 screenshot.** The [SIMULATION] GLOP demo connection in **GLOP connections**, with its buttons and counters.

GLOP › Summary	One card per connection with its status and anything that needs attention.
GLOP › Operations	**Day to day** with **Postings**, **Business days**, **Receipts** and **Receipt lines**. **Documents** with **GLOP tables**, **Orders to GLOP** and **GLOP documents**. **Outgoing to GLOP** with **Outbox items to GLOP**, **SAGE500 reconciliation** and **Send stock count**.
GLOP › Master data	**Structure**: **Venues**, **POS terminals** and **GLOP warehouses**. **Catalogue**: **Product families**, **Products**, **Pricelists**, **Sale formats**, **Recipes** and **Sizes and colours**. **Purchases**: **GLOP vendors**, **Purchase families** and **Purchase packages**. **Customers and payments**: **Taxes**, **Payment methods** and **Customers**.
GLOP › Audit	**Sync**: **Runs**, **Cursors** and **Received webhooks**. **Issues**: **API calls (errors)** and **Stock comparison**.
GLOP › Configuration	**GLOP connections**, **Settings** and **Create a simulation**.

Outside its own app, the connector adds its parts to Odoo's purchase orders, delivery orders, entries and journals, and to the general settings.

4 What you need before you start

- A GLOP account, with the Client ID and the secret GLOP gives you for its API.
- The complete dump of your point of sale to the GLOP cloud. The GLOP cloud only receives what is created or changed since you activate it: VAT rates and items created before that only arrive with this dump ("Envío de datos" › "Enviar todo", and "Enviar ventas" for the receipt history). GLOP support or your distributor does it, only at go-live and after warning GLOP beforehand. It cannot be launched from the API.
- The FlexigoTech Cloud subscription, or its 14-day trial with no card needed. Without it, the connector does not talk to GLOP.
- Odoo 17, 18, 19 or 20, on Odoo.sh or on your own server. It does not work on Odoo Online, which does not allow third-party modules to be installed.
- An Odoo administrator user, who is the one who accepts the cloud consent and saves the credentials.

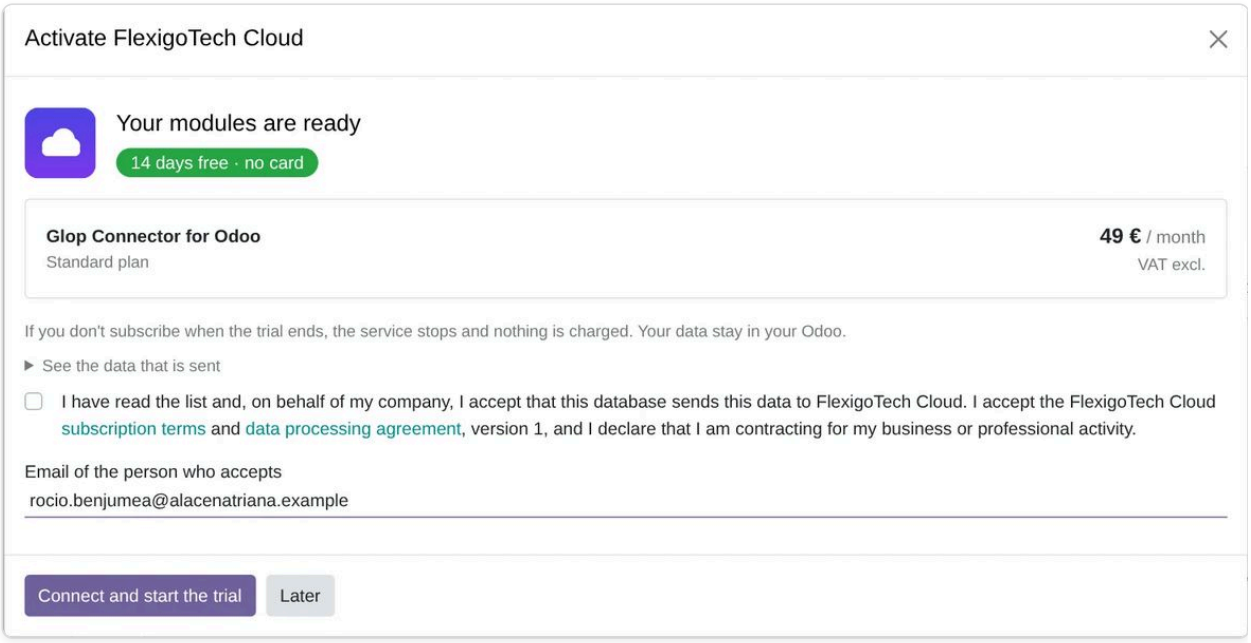
> The connector does not set up your GLOP account: it does not give you the Client ID or the secret and does not touch the point-of-sale configuration. It also does not register terminals or venues in GLOP: it reads them from GLOP. And it does not post on its own while **Post automatically** is off, which is how it comes.

It is installed like any third-party Odoo module, on Odoo.sh or on your server. Odoo installs `flexigotech_cloud`, the connection to FlexigoTech Cloud, together with the apps it depends on: Invoicing, Inventory and Purchase.

5 The FlexigoTech Cloud trial

1 **Press Activate in the top bar**

It appears as soon as you install the module. It opens the **Activate FlexigoTech Cloud** window, with the Standard plan and its 14 free days, no card needed.



- **Real Odoo 20 screenshot.** The activation window: read the list of data, tick the box and enter your email.

2 **Accept and enter your email**

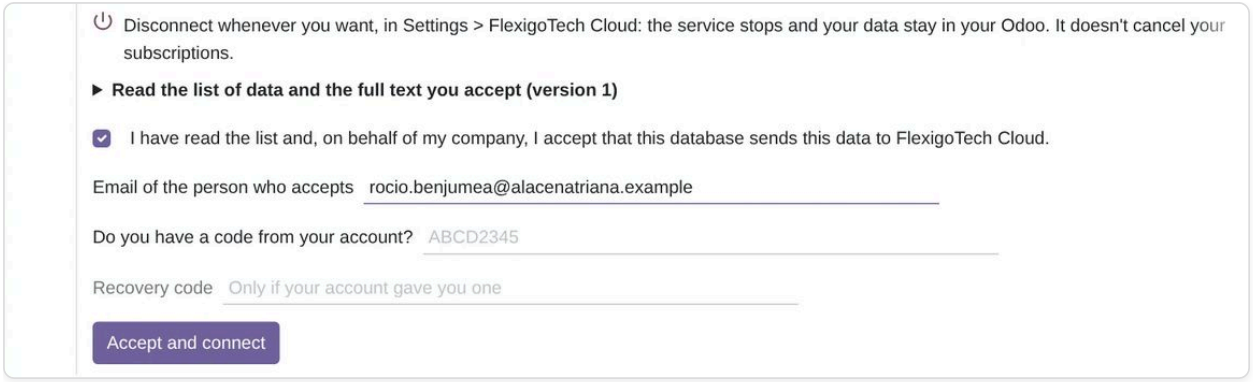
Tick "I have read the list and, on behalf of my company..." (it includes the subscription terms and the data processing agreement) and enter the **Email of the person who accepts**.

3 **Press Connect and start the trial**

Odoo registers the database with FlexigoTech Cloud and starts the module's trial. If you want to leave it for later, **Later** closes the window.

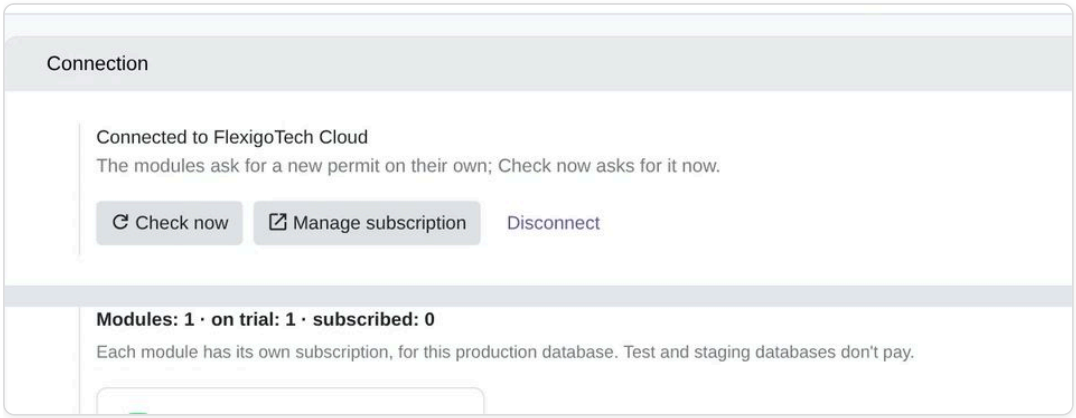
The same, from Settings

Settings › FlexigoTech Cloud has the full list of what the database sends. Tick the box, enter the email and press **Accept and connect**. If you already have an account at app.flexigotech.com, the field **Do you have a code from your account?** links the database to it (the code is valid for 10 minutes).



- **Real Odoo 20 screenshot.** Settings › FlexigoTech Cloud, before connecting: the box ticked, the email and **Accept and connect**.

With the database connected, the **Modules** block shows each module with its status: **On trial** and the date it ends, **Subscribed**, etc. From there you choose the plan (Standard or Pro, monthly or yearly) and press **Subscribe**: payment is made on Stripe's page, outside Odoo.



- **Real Odoo 20 screenshot.** Settings › FlexigoTech Cloud with the database connected and the connector on trial.

Prices exclude VAT. If you do not subscribe when the trial ends, the service stops and nothing is charged; your data stays in your Odoo. **Check now** requests permission immediately (otherwise the module renews it automatically every 6 hours).

> On a neutralised copy (staging) or on a database with demo data, a trial cannot be started: Settings says "This database is a neutralised copy or has demo data, so trials can't be started here."

CONNECTOR GUIDE · GLOP

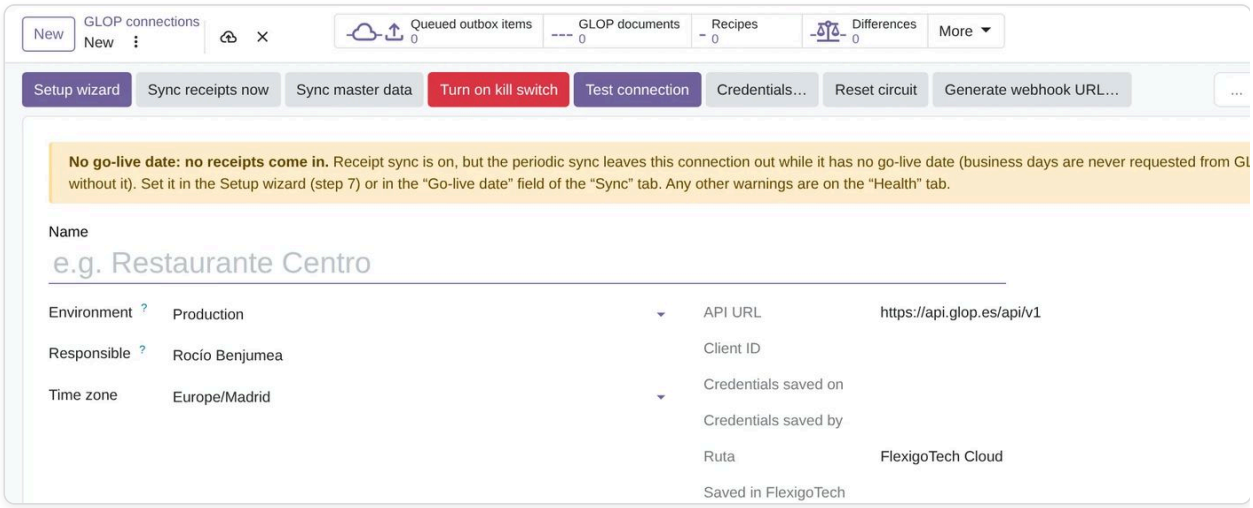
6 Connecting your GLOP account step by step

1. **Open the connections menu**

   Go to **GLOP › Configuration › GLOP connections**. The connections that already exist are listed there.

2. **Press New**

   The form for an empty connection opens. Nothing is connected until you save it.



- **Real Odoo 20 screenshot.** A new connection, with the **Test connection** and **Reset circuit** buttons in the header.

3 Fill in the name and the environment

Enter a **Name** and choose the **Environment**: **Production** or **Simulation**. The environment cannot be changed once the connection already has venues or receipts.

4 Check the API address and the time zone

The **API URL** is pre-filled with GLOP's. If there is no reason to change it, leave it as it is. Also choose the **Time zone**.

5 Press Credentials...

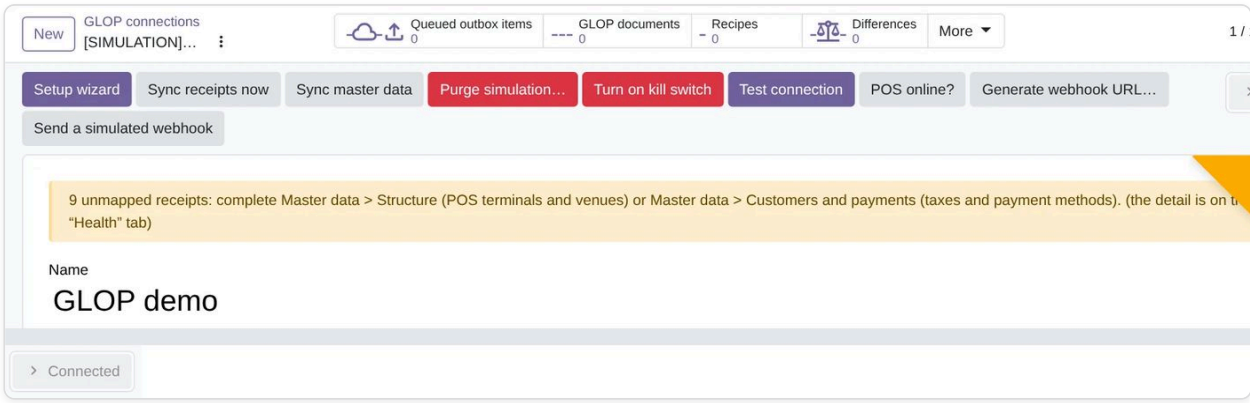
It is in the header and is the only place where the **Client ID** is entered or changed. It does not appear on a simulation connection.

6 Enter the credentials

Enter the **Client ID** and the **Client secret** that GLOP gave you. If the connection goes through FlexigoTech Cloud, also tick "I accept that the GLOP credentials and the calls to GLOP go through FlexigoTech Cloud": the Client ID and the secret travel encrypted just once to FlexigoTech Cloud, which stores them and calls GLOP with them, and they are not kept in Odoo. An Odoo administrator saves them. Without the tick you get "Before saving the secret, an administrator has to accept that the GLOP credentials and the calls to GLOP go through FlexigoTech Cloud. Nothing was sent." For security, the saved secret is never shown again.

7 Press Save and test the connection

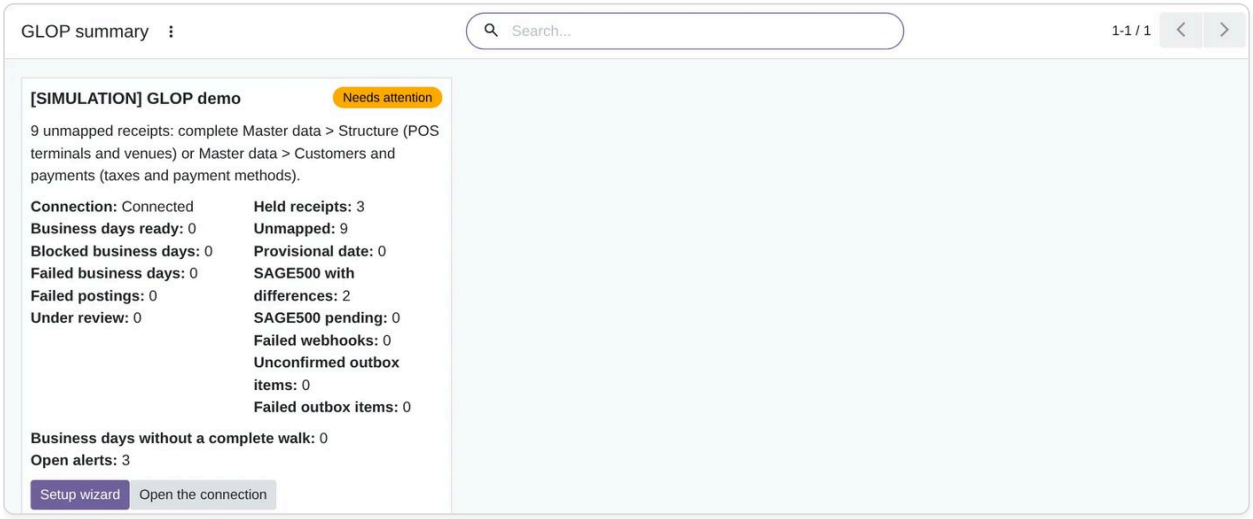
Odoo asks GLOP, through FlexigoTech Cloud, for the list of point-of-sale terminals. If all goes well, you get "Connection is fine: valid token and reachable point-of-sale terminals." with the point-of-sale terminals GLOP can see, and the connection changes to **Connected**. If GLOP does not have any point-of-sale terminal yet, it warns you that there will be no receipts to import until the point of sale sends data to its cloud. You can repeat the test whenever you like with **Test connection**.



- **Real Odoo 20 screenshot.** The simulation connection in **Connected** status, with the **Test connection** and **Sync receipts now** buttons. In simulation, the test warns that it has not called api.glop.es.

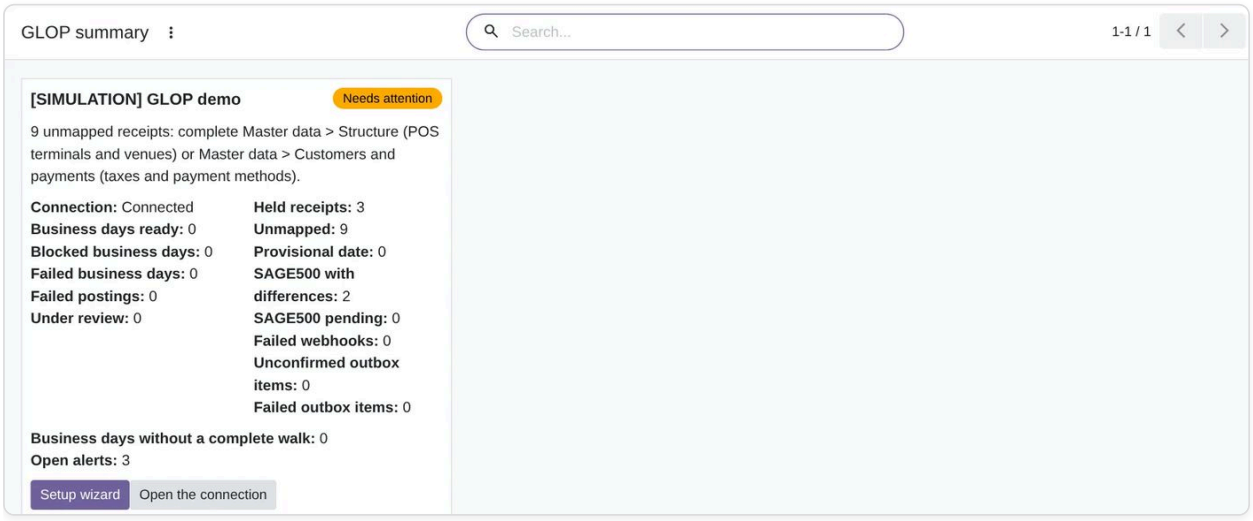
> The GLOP credentials are saved by an Odoo administrator, who is the one who accepts that they pass through FlexigoTech Cloud. The GLOP administrator permission is enough to press **Test connection**, **Credentials...** and the sync buttons. **Check in FlexigoTech Cloud** and **Delete from FlexigoTech Cloud** are only visible to the Odoo administrator.

After that, data arrives on its own. The scheduled task **GLOP: sync receipts by business day** reads receipts every 10 minutes, and **GLOP: master data** fetches terminals, taxes, payment methods and products once a day. You can also request it manually with **Sync receipts now** and **Sync master data**. If the connection has the kill switch on, **Sync master data** does not appear. Receipts are only requested from the **Go-live date**: without it none come in, and the connection warns you about this.



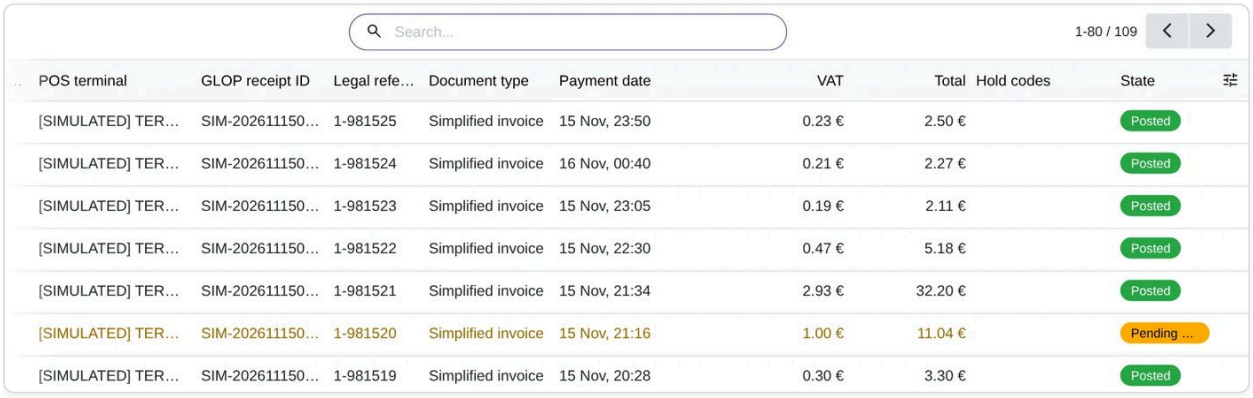
- **Real Odoo 20 screenshot.** The simulation's **GLOP summary**, with a card per connection and its status.

7 From GLOP receipts to accounting: business days, entries and VAT



- **Real Odoo 20 screenshot.** The simulation's **GLOP summary**, with the business days that are ready, blocked or in error.

Receipts come from GLOP through the scheduled task **GLOP: sync receipts by business day**, every 10 minutes. The task **GLOP: business days and posting**, every 15 minutes, evaluates each business day and leaves it ready to post. Receipts are in **GLOP › Operations › Day to day › Receipts** and business days in **GLOP › Operations › Day to day › Business days**.

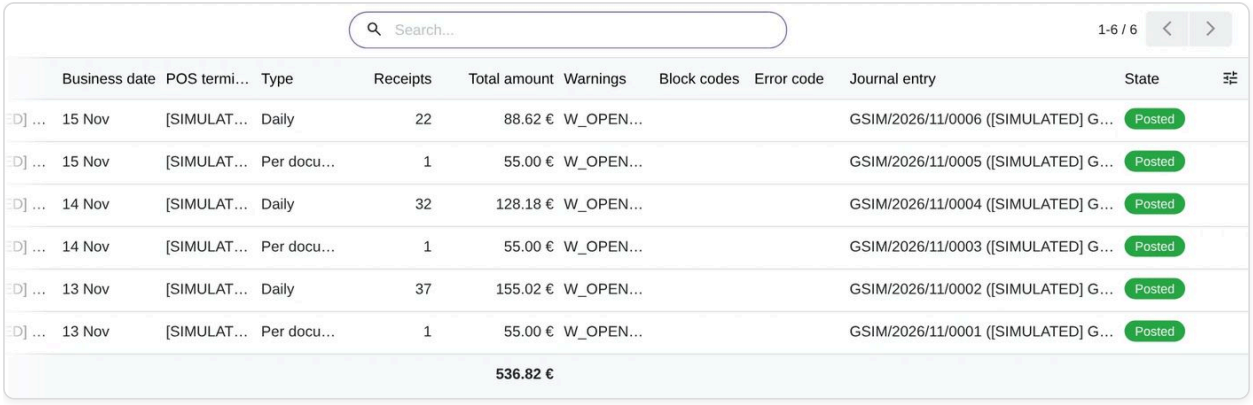


- **Real Odoo 20 screenshot.** The simulation's **Receipts** list, with the document type, payment date, VAT, total and status of each one.



- **Real Odoo 20 screenshot.** The simulation's **Business days**: three days in **Posted with held receipts**, with their receipts, the held ones and the difference.

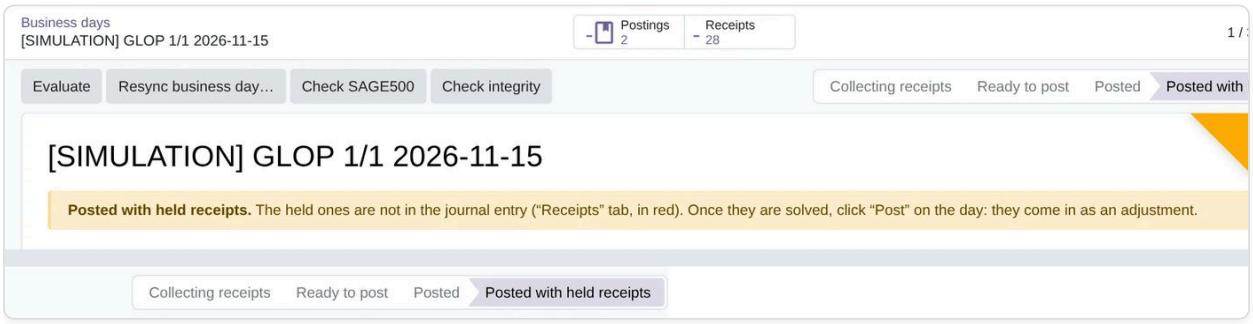
Each business day for a terminal gives rise to one entry with the base and VAT exactly as GLOP recorded them. The entries appear in **GLOP › Operations › Day to day › Postings**.



- **Real Odoo 20 screenshot.** The simulation's **Postings**, daily and per document, with their entry and status.

Business day buttons

Post	Posts the business day. It only appears if the day is **Ready to post** or **Blocked**.
Preview	Shows what would be posted without creating any entry. It only appears if there are receipts waiting to be posted.
Evaluate	Checks again now whether the business day is ready.
Resync business day…	Schedules a new read of the business day from GLOP.
Check SAGE500	Hidden while the day is still in **Collecting receipts**.
Check integrity	Only appears if the day already has postings.

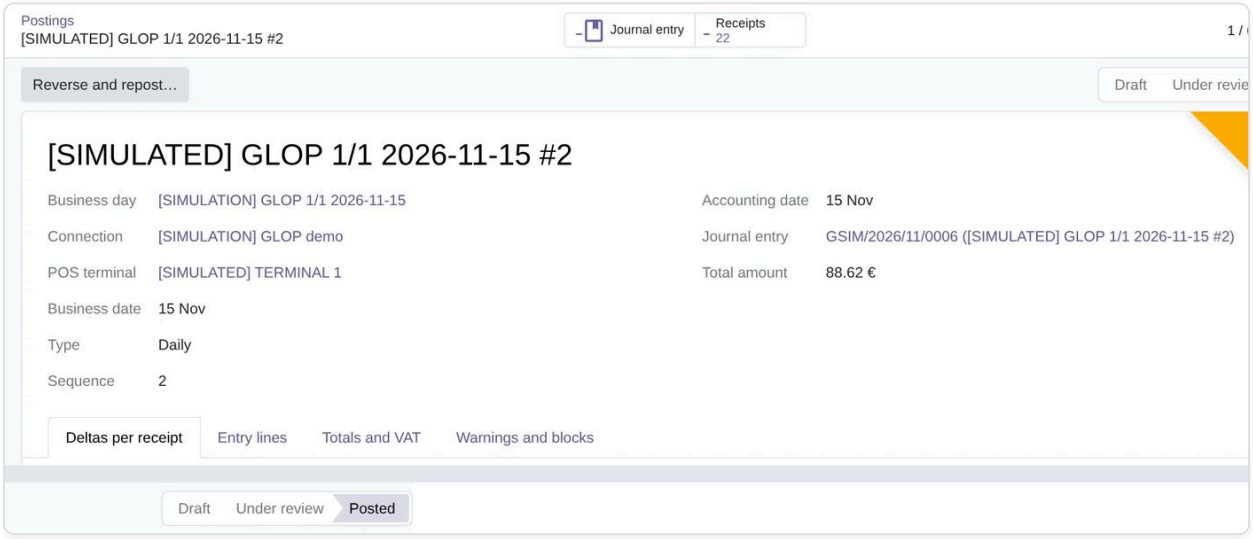


- **Real Odoo 20 screenshot.** A simulation business day in **Posted with held receipts**, with **Evaluate**, **Resync business day…**, **Check SAGE500** and **Check integrity**, and the warning that the held receipts will come in as an adjustment.

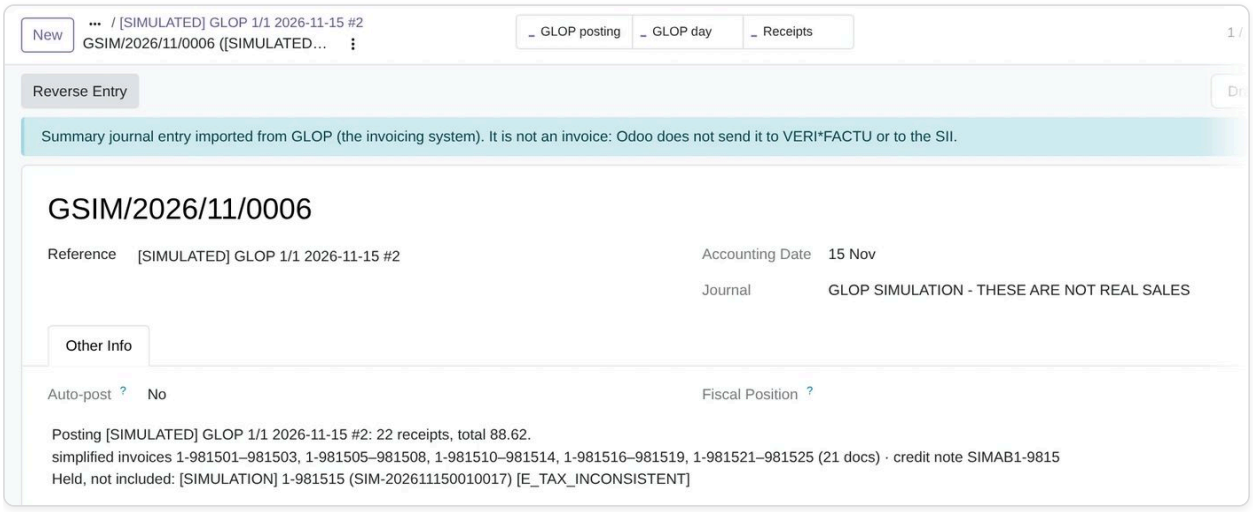
Posting buttons

Post	Posts a posting in **Draft** or **Blocked**.
Retry	Repeats a posting in **Error**.
Approve review	Approves a posting that is **Under review**.
Reject review…	Rejects the review and enter the reason.
Post on another date…	For a posting **Blocked** by the closing of a period.
Preview	Shows the entry lines it would create.
Reverse and repost…	Reverses the journal entry of a **Posted** posting.

Journal entry	Opens the journal entry, if one exists.



- **Real Odoo 20 screenshot.** A daily posting from the simulation, **Posted**, with **Reverse and repost...** and the tabs **Deltas per receipt**, **Entry lines**, **Totals and VAT** and **Warnings and blocks**.



- **Real Odoo 20 screenshot.** Its journal entry, in the simulation journal, with the notice that it is not an invoice and that Odoo does not send it to VERI*FACTU or to the SII.

Business day states

Collecting receipts, **Ready to post**, **Posted**, **Posted with held receipts**, **Blocked** and **Error**.

Posting states

Draft, **Under review**, **Posted**, **Reversed**, **Blocked** and **Error**.

1. **Open the business day**
   Go to **GLOP › Operations › Day to day › Business days** and choose the business day of the terminal.

2. **Click Preview**
   Check what would be posted.

3. **Click Post**
   The posting is created with its journal entry.

4 **Correct if needed**

If something fails, use **Retry**, **Approve review** or **Reverse and repost...**.

> A posted journal entry cannot be edited. To change it, reverse it and repost. The versions of a receipt cannot be modified either.

Business day states and what they mean

STATE	MEANING
Collecting receipts	The business day is still open: its **Entry time** has not yet arrived or the full set of receipts has not been collected.
Ready to post	It can now be posted.
Posted	Everything in the business day is in the journal.
Posted with held receipts	It has been posted without the held receipts.
Blocked	Something prevents posting. The reasons are in **Block codes** and **Detail of the block**.
Error	The process failed and remains visible.

Receipt states

Open, **Before go-live**, **Provisional date**, **Pending mapping**, **Held**, **To post**, **Posted**, **To adjust** and **Excluded**. To adjust is a receipt already posted whose amount changed in GLOP afterwards.

Which document each receipt is

The connector tells documents apart by the rule GLOP provided: with an invoice number it is a full invoice; with a receipt number, a simplified invoice; and with the refund flag, a corrective invoice. The series of corrective invoices is always "AB" plus the terminal number, for example AB1-1. A receipt without a payment date stays **Open**, and one without a fiscal number is held with the code E_NO_FISCAL_NUMBER.

Rules and conditions

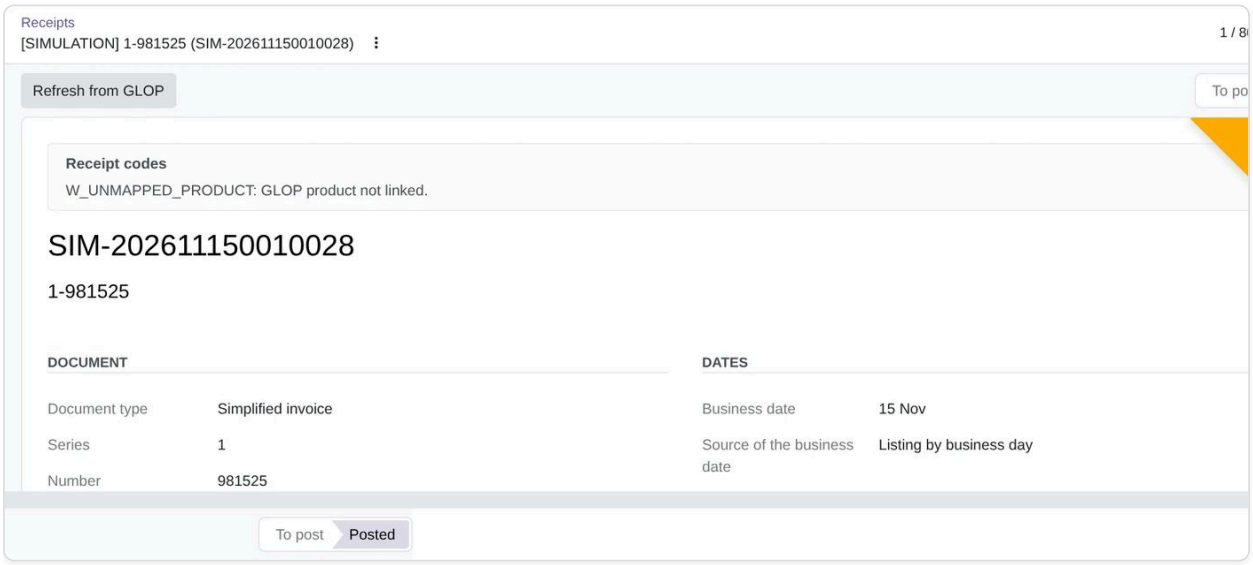
- **Accept the recalculation** only shows on a receipt with an inconsistent VAT waiting to be accepted. Otherwise it says "GLOP receipt ... has no inconsistent VAT waiting to be accepted."
- **Release** only appears with a receipt that is **Held**, **Pending mapping** or **Excluded**.
- **Exclude...** is hidden on excluded or posted receipts. The reason is required: "Enter the reason for excluding GLOP receipts." A receipt that is already posted says "GLOP receipt %s is already posted: use “Reverse and repost”."
- The business day's **Post** button does not retry postings with errors: each one has its own **Retry**.
- **Post on another date...** only appears if the posting is blocked by the closing of a period and is used by whoever administers the accounting. The reason is mandatory and the date cannot be in the future.
- **Reverse and repost...** requires a reason: "The reason for the reversal is required." With **Repost now** the day is posted again with the current mappings.

Record fields and tabs

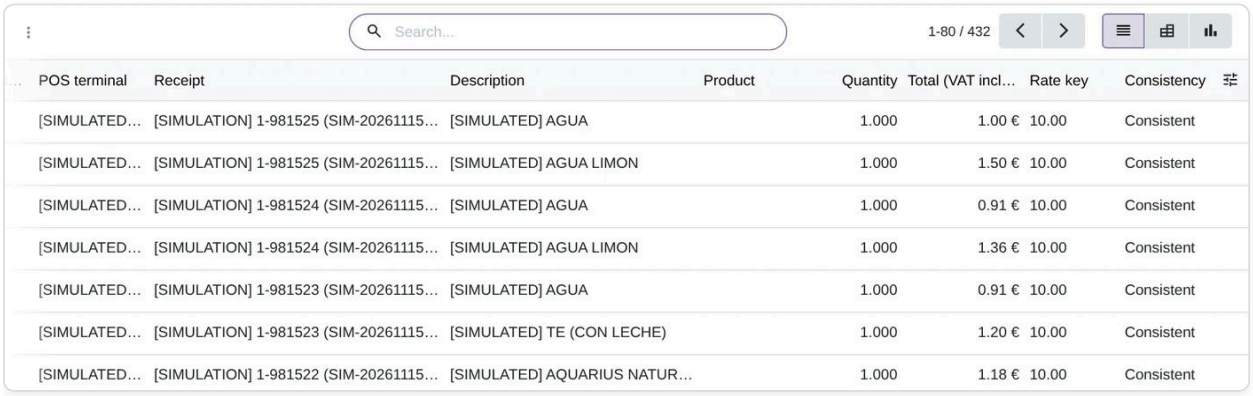
The business day has the groups **Business day**, **Preparation**, **Receipts of the day** and **Amounts**, with counters for receipts, **Unpaid**, **Held** and **Pending mapping**, and the **Difference** between the GLOP total and the journal. Its tabs are **Receipts**, **Postings**, **GLOP and posted totals** and **Detail of the block**.

The posting has **Type** (**Daily**, **Adjustment**, **Per document**, **Reversal**), **Accounting date**, **State of the stock** and the tabs **Deltas per receipt**, **Entry lines**, **Totals and VAT** and **Warnings and blocks**.

The receipt shows **Document**, **Dates** and **Amounts**, with the tabs **Codes**, **Lines**, **Payments**, **Versions**, **Postings** and **Payload**.



- **Real Odoo 20 screenshot.** A receipt from the simulation, **Posted**, with its document type, its series and its number, and the warning of a GLOP product not linked.

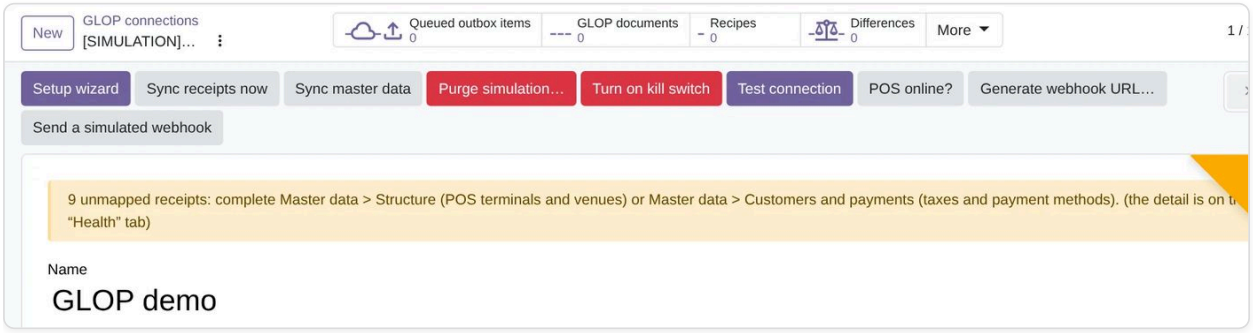


- **Real Odoo 20 screenshot.** The **Receipt lines** from the simulation, with the quantity, the total with VAT, the type key and the consistency of each line.

End-to-end example

You have the simulation connection "[SIMULATION] GLOP demo". Its **GLOP summary** shows 3 held receipts, 9 not mapped and 3 open alerts. In **GLOP › Operations › Day to day › Business days**, its three days are in **Posted with held receipts**: each already has its journal entry, without the held receipts. When you resolve a held receipt, you click **Post** on the day and it goes in as an adjustment. Receipts not mapped stay out; once you complete their mappings they are reviewed again automatically, and **Release** reruns them by hand.

8 Payment methods, customers and credit sales

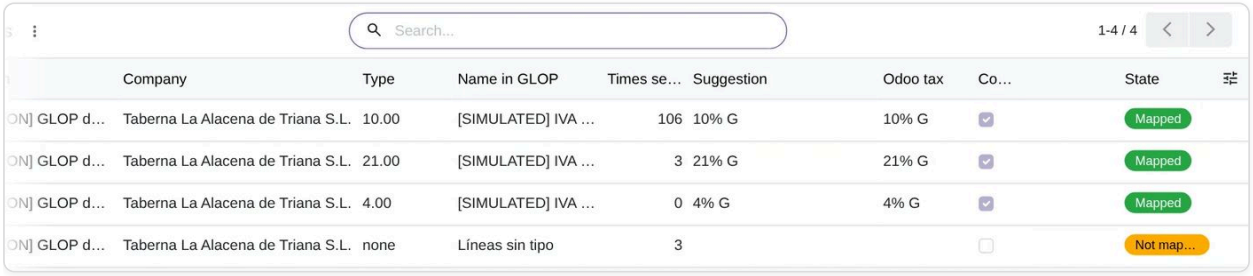


- **Real Odoo 20 screenshot.** The simulation connection, with the warning of 9 receipts not mapped that tells you to complete **Customers and payments**.

These lists are in **GLOP › Master data › Customers and payments**, with the menus **Taxes**, **Payment methods** and **Customers**. The rows for taxes and payment methods are created automatically when GLOP sends receipts that use them. Customers are imported by the scheduled task **GLOP: master data**, once a day. Customers are only sent to GLOP if you switch on **Send customers to GLOP** in the connection, which is off by default. Then the one you send with **Send to GLOP** goes out through the task **GLOP: sending customers**, within 15 minutes at most. That sending has only been tested with the module's own simulation and with direct, standalone calls to a test account in the GLOP cloud with no point of sale, on 18 September 2026. It has not been tested through FlexigoTech Cloud or with a real point of sale.



- **Real Odoo 20 screenshot.** The **Payment methods** of the simulation: cash and card, **Mapped**; the other two, **Not mapped**.



- **Real Odoo 20 screenshot.** The **Taxes** of the simulation: three VAT rates **Mapped** and the lines without a rate, **Not mapped**.



- **Real Odoo 20 screenshot.** The GLOP **Customers** of the simulation with their tax ID; one already has an Odoo contact and the **Send to GLOP** button.

Use suggestion	On a tax, copies into **Odoo tax** the one proposed in **Suggestion**. It only appears if there is a suggestion and you have not yet chosen a tax.
Confirm	On a tax, approves the chosen tax. It is hidden if it is already confirmed or if you have not chosen one.
Send to GLOP	On a customer, sends its **Contact** details to GLOP. It only appears if the customer already has a contact.

States

Not mapped	For payment methods and taxes: they still need to be related to Odoo.
Mapped	It already has its account or its tax.
Unconfirmed	For taxes: a tax has been chosen, but nobody has confirmed it.
Not valid	For taxes: the chosen tax is not valid for that VAT rate.

Steps

1. **Open the list**

   Go to **Payment methods** or **Taxes** and filter by **Not mapped** or **Pending**.

2. **Fill in the row**

   For a payment method, choose **Type** and **Account**. For a tax, choose **Odoo tax**.

3. **Confirm**

   For a tax, click **Confirm**. The receipts that were waiting for this mapping are reviewed again.

> A tax is not confirmed without choosing the Odoo tax first: you get the message "Choose an Odoo tax before confirming GLOP rate ...." If you change the tax on a row, it stops being confirmed and has to be confirmed again.

Payment methods

The GLOP API does not publish a catalogue of payment methods, so each row is learned from the receipts and identified by its **Key**, which can be the identifier, the payment text or the slot number. **Times seen** and **Seen on** show how much it is used. Nothing comes with default values.

Cash	Cash register payment.
Card	Card payment.
Other payment	Any other payment.
Expense (invitation / internal consumption)	Invitations and internal consumption.
Customer account (credit)	Credit sale, linked to a **Contact**.

The form separates two groups, **GLOP** and **Odoo**. The filters are **Not mapped**, **Mapped**, **Type** and **Company**. When you create or save a row with complete mapping, the affected receipts are reviewed again.

Rules for each type

CASE	MESSAGE
The key does not have a valid format	"Invalid GLOP payment method key “%s”."
Customer account without Contact	"GLOP payment method “%s”: a customer account needs a contact."
Cash or card with receivable or payable account	"GLOP payment method “%(label)s”: a cash or card payment cannot go to a receivable or payable account (%(account)s)."
Cash or card with account that does not allow reconciliation	"GLOP payment method “%(label)s”: bridge account %(account)s must allow reconciliation."
Expense with account that is not an expense account	"GLOP payment method “%(label)s”: an invitation or internal consumption needs an expense account (%(account)s is not one)."
Customer account with account that is not receivable	"GLOP payment method “%(label)s”: a customer account needs a receivable account (%(account)s is not one)."
Account or contact from another company	"GLOP payment method “%(label)s”: account %(account)s belongs to another company." and "GLOP payment method “%s”: the contact belongs to another company."

Taxes

Rows are identified by the VAT rate, because ticket lines carry the percentage and not a tax identifier. The rate is written with two decimals, for example "10.00", or "none" for lines without a percentage. Otherwise you get "Invalid GLOP VAT rate “%s”: use '10.00' or 'none'." The **GLOP tax ID**, **Name in GLOP** and **GLOP equivalence surcharge** are also stored.

Not mapped	No Odoo tax has been chosen.
Unconfirmed	A tax has been chosen, but **Confirm** is missing.
Mapped	The tax is chosen and confirmed.
Not valid	The chosen tax cannot map that GLOP rate for %(company)s: "Tax “%(tax)s” cannot map GLOP rate %(rate)s of %(company)s: %(problems)s."

Suggestion proposes the company's sales tax that meets all the rules. It is never confirmed on its own, and rates 0.00 and none get no suggestion. Only the accounting profile can **Confirm**, which records **Confirmed by** and **Confirmed on**.

Customers

GLOP customers only create links when master data is fetched. An Odoo contact is created only when a ticket needs it. First the contact is searched by tax ID, trying the spelling with and without the country prefix. If Odoo does not accept the tax ID, the contact is created without one, the note goes into the reference, and a warning appears on the ticket.

Some contacts are never matched with a GLOP tax ID: the company contacts, the customer for credit sales with no customer at each venue, the internal users and the work contacts of employees. Empty contact fields of the company are filled with GLOP data, without overwriting those it already has. When sending to GLOP, the tax ID is sent without the country prefix, unless GLOP already has the same tax ID in another spelling, which is respected.

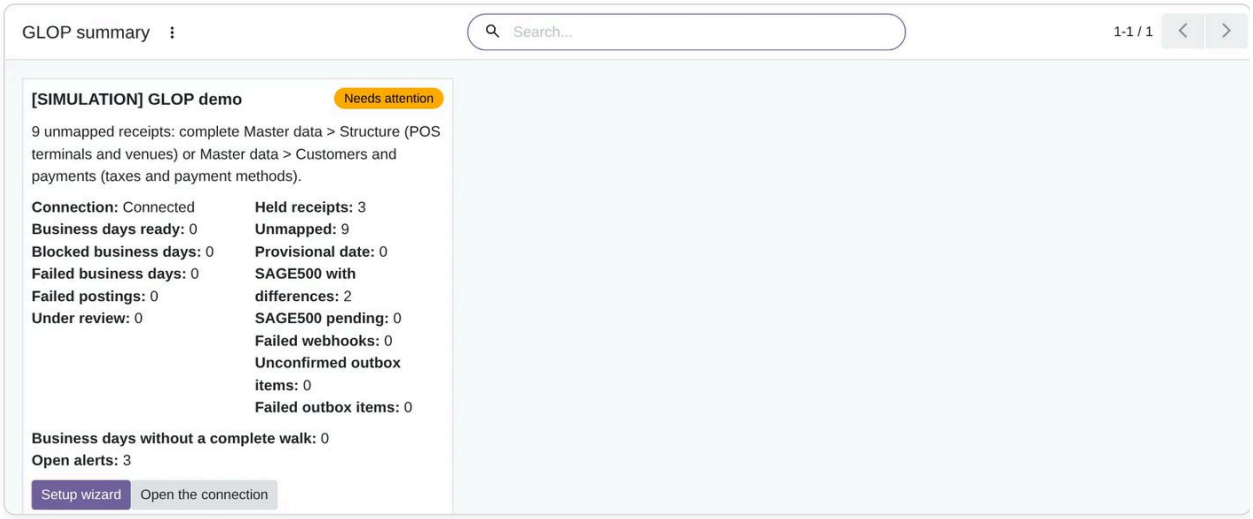
GLOP customer ID	Identifier in GLOP.
Tax ID in GLOP and **Normalised tax ID**	The tax ID as GLOP stores it and its normalised form.
Contact	The matched Odoo contact.
Seen in GLOP on and **Absences in GLOP**	When it was last seen and how many times it did not appear.

The filters are **No contact** and **With a contact**.

End-to-end example

1 **Tickets arrive**

The [SIMULATION] GLOP demo connection warns of 9 unmapped tickets and of missing taxes and payment methods in **Customers and payments**.



- **Real Odoo 20 screenshot.** The **GLOP summary** of the simulation, with "Unmapped: 9".

2 **Map the payment methods**

A credit payment method is set as **Customer account (credit)**, with its **Contact** and a receivable account.

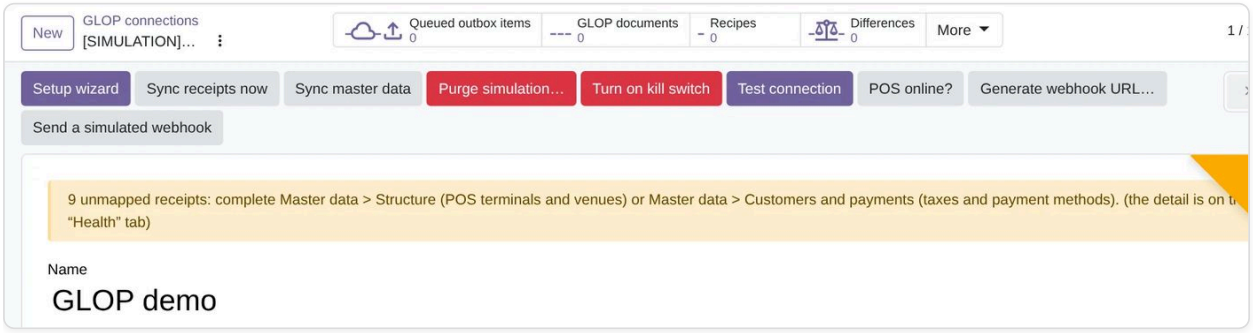
3 **Map the taxes**

Press **Use suggestion** and then **Confirm**.

4 **The tickets are reviewed**

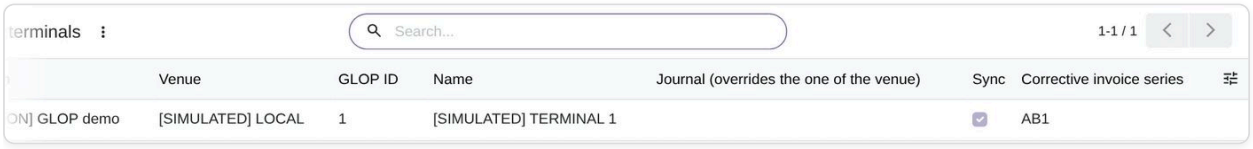
On confirming, the tickets that were waiting for the mapping are reviewed again.

9 Master data: venues, terminals and catalogue



- **Real Odoo 20 screenshot.** The simulation connection, with the **Sync master data** button and the warning about unmapped tickets.

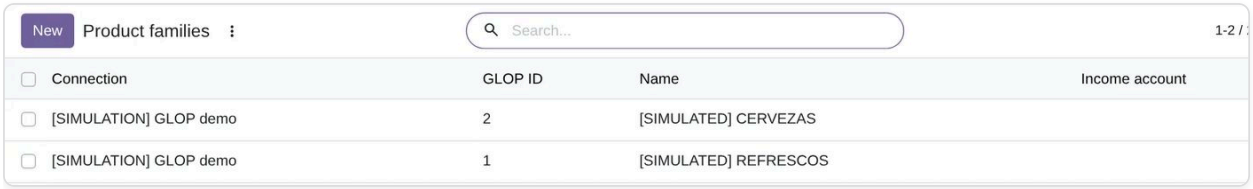
The scheduled task **GLOP: master data** runs every day and goes through all active connections that have master data enabled. You can also run it by hand from the connection with **Sync master data**. It reads venues, terminals, tables, warehouses, product families, products, price lists, sales formats and sizes and colours. The data arrives in **GLOP › Master data › Structure** and **GLOP › Master data › Catalogue**.



- **Real Odoo 20 screenshot.** The terminal of the simulation, with its venue and its **Corrective invoice series**, AB1, which the connector calculates on its own.



- **Real Odoo 20 screenshot.** The GLOP **Products** of the simulation, still **Not linked**, with the **Ignore** button.



- **Real Odoo 20 screenshot.** The GLOP **Product families** of the simulation, still without an **Income account**.

Sync master data	Starts the GLOP read now, without waiting for the daily task. When it finishes it shows what it read, what came back empty and what needs attention.
Ignore	On a GLOP product, leaves it out of the warnings. Visible while the product is not ignored.
Stop ignoring	Returns the product to the normal list. Only visible if it is ignored.
Use the suggestion	On a warehouse, a price list or a sales format, copies GLOP's suggestion into your decision. Only visible if there is a suggestion and it does not yet match the choice.
Fetch the photo from GLOP	On a product, manually downloads the photo GLOP has for it.

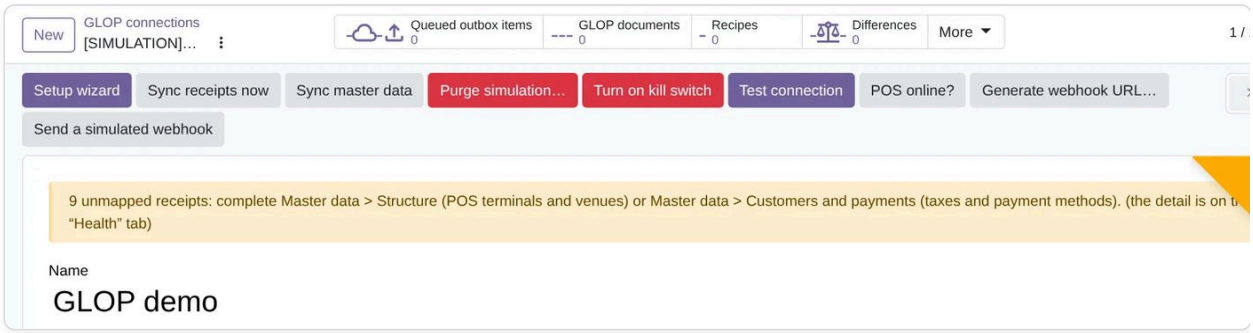
Set the photo on the product	Copies that photo onto the Odoo product. Only visible if a photo has been fetched and the product is linked.

Product states

- **Not linked**: GLOP has the product and Odoo does not yet know which product it corresponds to.
- **Linked**: the product is matched with an Odoo one, by **Barcode**, by **Internal reference** or by hand.
- **Ignored**: you have deliberately left it out.

1 Sync

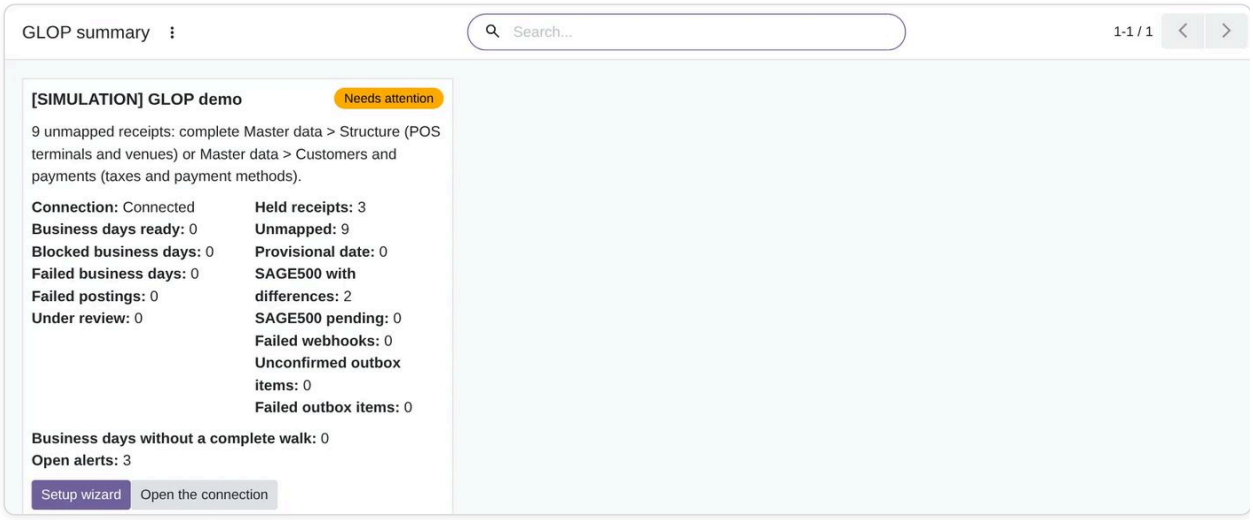
Press **Sync master data** on the connection.



- **Real Odoo 20 screenshot.** The **Sync master data** button on the simulation connection.

2 Review the products

Open **GLOP › Master data › Catalogue › Products** and link those that are **Not linked**.



- **Real Odoo 20 screenshot.** The **GLOP summary** of the simulation, with 9 unmapped tickets.

3 Complete the structure

Go to **GLOP › Master data › Structure › Venues** and set the journal and accounts for each venue.

> The sync never creates products or contacts in Odoo, never assigns a warehouse or a price list on its own, and never archives or deletes anything. New venues and terminals arrive with **Sync** unticked: nothing is accounted for until someone configures and activates them.

Venues: states and deduced

A venue is the point of sale whose accounting receives the tickets. It arrives with **Sync** unticked, and while it stays that way nothing from it is accounted for. **Deduced from a POS terminal** marks those that GLOP does not publish but names a terminal for: Odoo creates them with a provisional name and leaves a note on the connection. If GLOP publishes it later, the sync completes it and removes the mark. In the list you can filter by **Syncing**, **No journal**, **Missing in GLOP**, **Deduced from a POS terminal** and **Archived**.

Missing in GLOP

When a record stops coming back after a complete read, **Absences in GLOP** goes up by one. On reaching 3, the connection receives a note. Nothing is ever archived or deleted automatically: you decide.

Rules and errors

MESSAGE OR SITUATION	WHEN IT APPEARS
"GLOP venue “%(name)s”: account %(account)s (%(field)s) does not belong to company “%(company)s”."	You set an account that belongs to another company.
"GLOP venue “%(name)s”: a simulation connection can only post into a company marked as demo."	The venue belongs to a simulation connection and its company is not a demonstration one.
"GLOP point-of-sale terminal “%s”: the venue belongs to another connection."	The terminal and its venue are not from the same connection.
"GLOP product “%s”: a link with no product cannot be “Linked”."	You mark the product as linked without having chosen an Odoo product.
"GLOP product family “%(name)s”: account %(account)s does not belong to %(company)s."	The family's income account belongs to another company.

If you change the journal, the accounts or **Sync** of a venue, or link a product or set its account, the tickets waiting for that are re-evaluated automatically.

Venue fields

Journal	General journal, required to account.
Default income account	Income account used when neither the product nor the family has its own.
Discount account, **Rounding account (gain)** and **Rounding account (loss)**	Accounts for discounts and roundings.
Customer for credit sales with no customer	Contact to which credit sales arriving with no customer are charged.
Warehouse, **Outgoing type** and **Return type**	The warehouse and the stock operation types of that venue. **Warehouse suggested by GLOP** is proposed by GLOP from the name, but is never assigned automatically.
Time zone	The venue's own.

The record has the groups **GLOP**, **Accounting** and **Stock** and a **POS terminals** tab. Each point-of-sale terminal can override the journal and the outgoing type of the venue with **Journal (overrides the one of the venue)** and **Outgoing type (overrides)**. **Corrective invoice series** is calculated automatically and is not configured.

Products, product families, pricelists and formats

On each GLOP product, **Income account (overrides)** takes precedence over the one of the family. The **Product family** stores its **Income account**, and you can filter out those without one with **No income account**. In the pricelists, **Odoo pricelist** is your decision and **Suggested pricelist** is proposed by GLOP based on the name. In the sale formats, **Measure** says how much the format deducts and **Odoo unit of measure** is informational only: the connector does not convert quantities using it. In the warehouses, **Odoo warehouse** decides which warehouse stock is compared against.

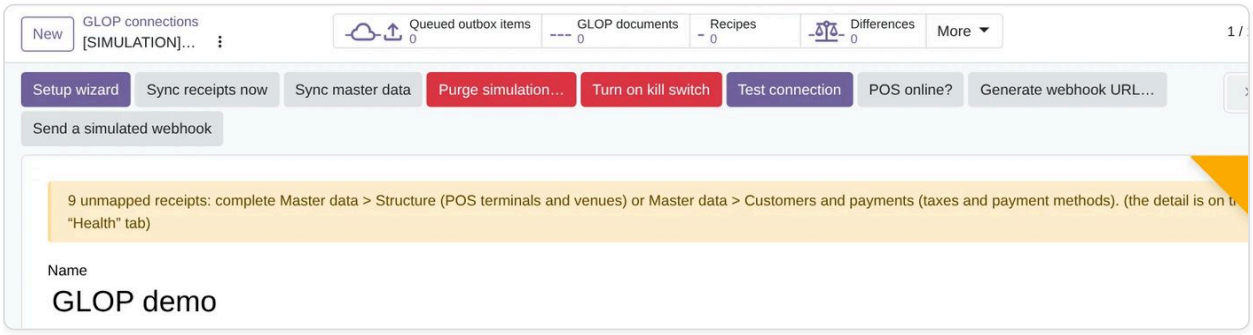
End-to-end example

With the connection [SIMULATION] GLOP demo, the **GLOP summary** warns of 9 unmapped tickets and sends you to complete the structure and the customers and payments. You press **Sync master data**, open **GLOP › Master data › Structure › Venues**, set the journal and income account on the venue, tick **Sync**, and in **GLOP › Master data › Catalogue › Products** you link the products that are **Not linked**. The tickets that were waiting for that data are re-evaluated automatically.

CONNECTOR GUIDE · GLOP

10 Purchasing and vendors with GLOP

> Everything in this section is off by default. **Sync vendors and purchases**, **Send purchase orders to GLOP**, **Turn receipts of goods into GLOP delivery notes** and **Send stock counts to GLOP** are switched on in the **Sync** tab of the connection. What Odoo writes to GLOP (purchase orders, conversions into delivery notes and the deletion of stock documents) has only been tested with the module's own simulation and with direct, standalone calls to a test account in the GLOP cloud without any point of sale, on 18 September 2026. It has not been tested through FlexigoTech Cloud or with a real point of sale.



- **Real Odoo 20 screenshot.** The simulation connection, with the **GLOP documents** counter at the top.

Where each piece of data comes from

- The pending orders and delivery notes are fetched by the scheduled task **GLOP: pending purchase documents**, every hour.
- The vendors and the purchase families and packaging are read by the scheduled task **GLOP: master data**, every day, or with the button **Sync master data** on the connection.
- You fetch a specific document yourself by its number with the wizard **Fetch a document from GLOP**.

Record buttons

Fetch from GLOP (document)	Reads that document again and saves what GLOP has now. It is not visible if the document has no **Number in GLOP**.
Turn into a delivery note (document)	Queues the conversion of the order with the quantities of its lines. It is only visible on a purchase document **In GLOP** whose type in GLOP is 1. GLOP treats the order as fully converted, even if only part of it has been received, and it does so afterwards: its response only confirms that the order was sent to the point of sale.
Delete in GLOP (document)	Queues the deletion of a stock document. It is only visible if it is **In GLOP**.
Purchase order and Goods receipt (document)	Open the linked Odoo purchase order or goods receipt. They are not visible if there is none.
Send the order to GLOP (purchase order)	Queues the creation of the order in GLOP. Only products and quantities are sent: GLOP sets its own prices and taxes. It is only visible with **Send purchase orders to GLOP** switched on and while the order has not been sent.
Fetch template from GLOP (GLOP vendor)	Reads the items that GLOP suggests ordering from that vendor.
Use the suggestion (purchase family or packaging)	Copies Odoo's suggestion into your decision. It is not visible if there is no suggestion or it already matches.

Document statuses

- **Draft**
- **Queued for GLOP**
- **In GLOP**
- **No longer in GLOP**
- **Error**

Linking a vendor

1. **Open the list**
   Go to **GLOP › Master data › Purchases › GLOP vendors**.
2. **Find the pending ones**
   Use the filter **No Odoo vendor**.
3. **Choose the contact**
   Fill in **Vendor** with the Odoo contact.

> The connector never creates a vendor, a category or a unit on its own: each link is your decision or a tax ID match that is unambiguous. To send an order to GLOP, the vendor must be linked.

Document statuses and what they mean

Draft	Document created in Odoo that has not yet been queued.

Queued for GLOP	Odoo has it ready and it does not yet appear in GLOP. The periodic read does not touch it.
In GLOP	GLOP has it pending delivery or partially delivered.
No longer in GLOP	GLOP no longer returns it: it was delivered or turned into a delivery note. Odoo keeps the trace and does not delete it.
Error	The sending failed. The reason appears in **Last error**.

Served and status in GLOP

Served can be **Not served**, **Partially served** or **Served**. **Status in GLOP** reflects whether GLOP has it pending delivery, partially delivered or delivered. The periodic read only brings back the first two cases, so what has already been served is kept as **No longer in GLOP**. The filter **Already served** groups them together.

Rules and conditions

- **Turn into a delivery note** only appears on a purchase document that is **In GLOP** and is of type 1. GLOP closes it entirely.
- **Delete in GLOP** only exists for stock documents **In GLOP**. It is the only write in this block that GLOP can undo.
- A document must already exist in GLOP for any operation except creating it.
- When sending an order, the connection must have sending purchase orders enabled, and the vendor must be linked to a GLOP vendor.
- Each line must match a GLOP product, and its unit must be the product's own unit. Otherwise the sending is rejected rather than dropping the line or converting quantities.
- If there are several connections, or several terminals, choose the connection and the terminal on the purchase order. If there is only one connection that sends orders, that one is used.

The GLOP vendor record

Connection and **GLOP vendor ID**	The connection it comes from and the vendor identifier in GLOP.
Vendor	The Odoo contact. It is assigned automatically only when the tax ID matches exactly one on each side; otherwise you choose it.
Tax ID in GLOP and **Normalised tax ID**	The tax ID as it arrives and its normalised version.
Central purchasing and **POS terminals in GLOP**	Whether it is a central purchasing unit and the terminals GLOP associates with it.
Accounting account in GLOP	Informational only: the connector does not create or modify accounts with it.
Seen in GLOP on and **Absences in GLOP**	When it was last seen and how many times it has been missing.
Order template tab	The items GLOP suggests ordering. **Template read on** indicates when they were read.

The template is only read with **Fetch template from GLOP**, not in the daily read. The filters **No Odoo vendor**, **Central purchasing** and **Do not show up in GLOP** help to sort the list.

Purchase families and packaging

Each family or packaging leaves an Odoo suggestion with a matching name: **Suggested category** or **Suggested unit**. If you accept it with **Use the suggestion**, it is copied to **Odoo category** or **Odoo unit of measure**. The unit is informational only: purchase documents are sent in GLOP's units.

End-to-end example

1. **Sync**
   With **Sync vendors and purchases** and **Send purchase orders to GLOP** switched on in the connection, press **Sync master data**.

2. **Link**
   In **GLOP vendors**, fill in **Vendor** where it is missing.

3. **Read the template**
   Press **Fetch template from GLOP**.

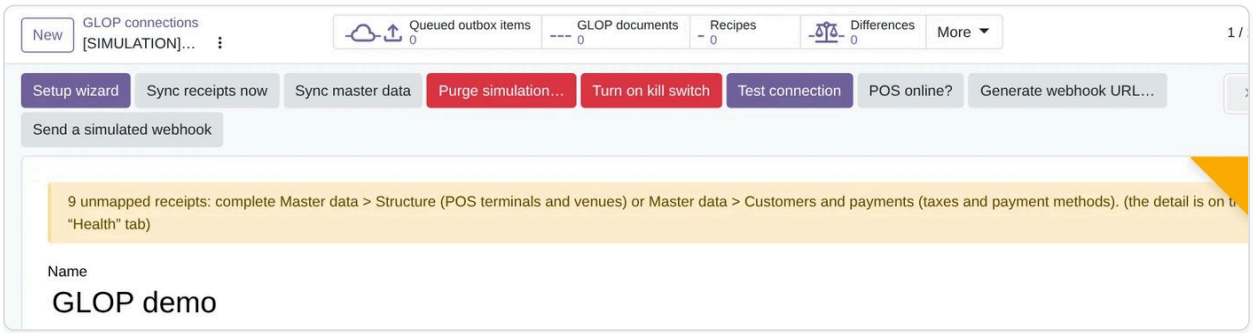
4. **Send**
   On the purchase order, press **Send the order to GLOP**. The document goes through **Queued for GLOP** and then **In GLOP**.

5. **Follow the result**
   In **GLOP documents** check the status, and in the **Outbox items to GLOP** tab the detail of each sending.

CONNECTOR GUIDE · GLOP

11 Stock, recipes and counts

> Everything in this section is off by default: **Stock** is set to **No stock**, and **Sync sale formats and recipes**, **Compare the stock with GLOP**, **Send recipes to GLOP** and **Send stock counts to GLOP** are unticked. What Odoo writes to GLOP (recipes and counts) has only been tested with the module's own simulation and with direct, standalone calls to a test account in the GLOP cloud without any point of sale, on 18 September 2026. It has not been tested through FlexigoTech Cloud or with a real point of sale.



- **Real Odoo 20 screenshot.** The simulation connection, with its buttons and the **GLOP documents**, **Recipes** and **Differences** counters.

Where each piece of data comes from

- With **Stock** set to **Products sold**, the connector generates the stock movements for what was sold from each posting. They come out as a delivery note and, if there are returns, as a return delivery note. On the delivery note you see the **GLOP direction** (**Outgoing** or **Return**) and the **GLOP posting**.
- With **Sync sale formats and recipes** switched on, the recipes arrive with the scheduled task **GLOP: master data**, every day, or with the button **Fetch from GLOP** on each recipe.
- GLOP's stock, to compare against Odoo's, is read from GLOP and stored in **Stock comparison**. The comparison only reads: it writes no quantities either in Odoo or in GLOP.

Recipe buttons

Recipes are in **Recipes**, under **GLOP › Master data › Catalogue**.

Fetch from GLOP	Reads the recipes of that sale format again and updates what GLOP says now.
Create a bill of materials	Turns the recipe into an Odoo kit-type bill of materials.
Fetch from the bill of materials	Fills in the lines with the product's kit bill of materials in Odoo.
Send to GLOP	Queues the whole recipe of that product and then reads it again to check it, because GLOP replies that it went fine even when the product does not exist.
Send this line	Sends the change of quantity or cost of a single ingredient.
Outbox items	Open the outbox items for that recipe.

Stock comparison statuses

Each row of the comparison has one of these values in **Comparison**: **They match**, **They differ**, **Not linked** or **No warehouse chosen**. The row stores **Compared on** and **Read from GLOP on**. The **Recalculate** button reads the Odoo stock again without asking GLOP.

Sending a count to GLOP

1. **Open** Send stock count
   It is under **GLOP › Operations › Outgoing to GLOP**. Choose the connection and the **GLOP POS terminal**; its warehouse is the one being adjusted.
2. **Press** Load the count
   The GLOP stock is read at that moment and placed next to the Odoo stock.
3. **Review the lines and press** Send to GLOP
   A single inventory adjustment is queued with the lines that change: GLOP applies each quantity as a difference on its stock at that moment. The **Send to GLOP** button does not appear until the count is loaded.

> With GLOP as the stock master, a product with a kit bill of materials does not move in Odoo: the move fails and names the product, because the ingredients would be deducted twice. In that case the recipe cannot be converted into a bill of materials. The link of a delivery note with its GLOP posting cannot be changed, and a second count cannot be sent while another one for the same warehouse is unresolved.

Who deducts the ingredients

The connection has a single owner of the ingredient stock, chosen under **Owner of the ingredient stock** in the **Posting** tab. By default it is GLOP: the point of sale has already deducted the ingredients of a recipe, and Odoo moves the product as GLOP sold it, without breaking down any kit. If the linked product has a kit bill of materials, the stock step fails with a message naming the product. The posting stays done and the stock can be retried later. If the master is Odoo, Odoo does break down the kit when the move is confirmed. This way the ingredients leave only once.

How the move is calculated

Storable products are moved. Each posted line contributes its stock difference, and credit notes count negatively. Products are linked the same way as ticket lines. If a product is not linked at the time of posting, it is allocated when it is linked, because the calculation is redone while the posting awaits review. The result goes to an outgoing or return delivery note, which is confirmed and validated automatically.

Statuses of each comparison row

- **They match**: the GLOP and Odoo stock are equal.
- **They differ**: the quantities do not match. The column **Difference (GLOP − Odoo)** shows by how much.
- **Not linked**: the GLOP product has no Odoo product. It is linked from the **Link** button.
- **No warehouse chosen**: the row has no warehouse assigned.

Comparison fields

Source	**Stock** or **Stock by size and colour**.
Product in GLOP and **Format/variant in GLOP**	The product and its variant. Without a variant, the value is "0".
Stock in GLOP and **Stock in Odoo**	The two quantities being compared.
Warehouse	The GLOP warehouse the row belongs to.
Product and **Link**	Open the Odoo product and the link with the GLOP product.

The list is filtered by **Differences only**, **Not linked**, **Sizes and colours**, **Warehouse** and **Comparison**.

The recipe: fields and tabs

Composed product in GLOP and **Sale format in GLOP**	The product that is broken down and its format. If GLOP does not send a format, the value is "0".
Ingredients	Tab with the components.
Odoo bill of materials ID	The kit-type bill of materials this recipe created.
Sent to GLOP on and **Last error**	When it was sent and what failed.
Seen in GLOP on and **Absences in GLOP**	When it was last seen and how many times it has been missing.

The filters of the recipe are **No Odoo product**, **With a bill of materials**, **Sent to GLOP**, **Failed** and **Sale format**. Each line shows its origin: **Read from GLOP** or **Fetched from the bill of materials**.

Count rules

- Before sending you have to click **Load the count**. Without that you get "Load the count first: without it, what GLOP has is unknown."
- If no line changes, the message is "The count changes nothing in GLOP: there is no difference to send."
- Since GLOP applies the count as a difference, a second count for the same warehouse cannot be sent while another is still queued, being sent or unconfirmed: the difference would be applied twice.
- If the Odoo warehouse is empty, the company's total stock is used.

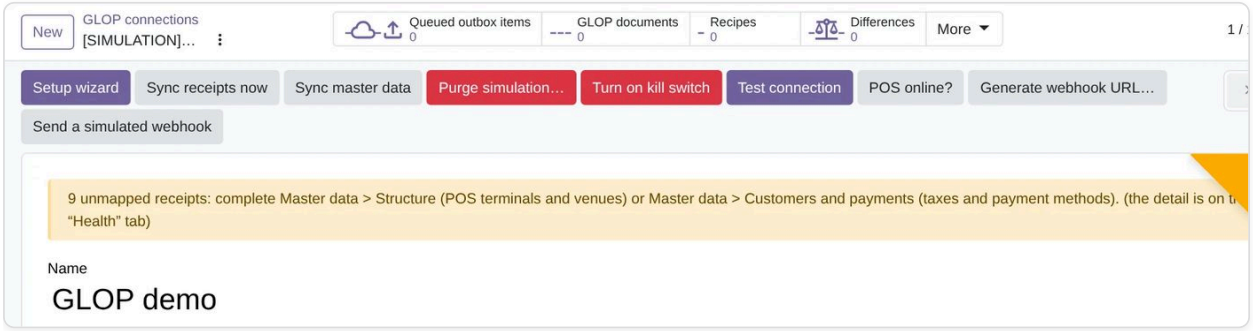
An end-to-end example

With **Send stock counts to GLOP** switched on, you open **Send stock count** and choose the terminal. You press **Load the count** and see, for each product, **In GLOP now**, **Counted in Odoo** and **Actual quantity**. **Will be sent (difference)** is calculated on the lines that change. You press **Send to GLOP**, a single adjustment is created in the queue, and you can follow it in **Outbox items to GLOP**.

CONNECTOR GUIDE · GLOP

12 Orders from Odoo to GLOP tables and delivery

> It is off by default: **Send table and delivery orders** is switched on in the **Sync** tab of the connection. It has only been tested with the module's own simulation and with a test order, direct, to a test account in the GLOP cloud with no point of sale, created and cancelled on 18 September 2026. It has not been tested through FlexigoTech Cloud or with a real point of sale.



- **Real Odoo 20 screenshot.** The simulation connection, with the **Setup wizard** button and the counters **Queued outbox items** and **GLOP documents**.

The orders are under **GLOP › Operations › Documents › Orders to GLOP**. You create each order yourself. The sending goes out when you press **Send to GLOP**: it enters the send queue and the scheduled task **GLOP: sending customers** dispatches it, every 15 minutes. That same task handles the rest of the connection's outgoing items. You can follow it under **GLOP › Operations › Outgoing to GLOP › Outbox items to GLOP**.

If you enter a **Table**, GLOP adds the lines to that table's open ticket. If you leave it empty, the order is parked as a delivery order.

Send to GLOP	Puts the order in the send queue. Only visible when the order is in **Draft**.

It did reach the POS	Marks the order as confirmed by hand. Only visible when the order is in **Sent, unconfirmed**.
It did not reach the POS	Closes the order as **Error**. Nothing is resent on its own. Only visible when the order is in **Sent, unconfirmed**.
Void in GLOP	Voids in GLOP an order already sent that has not yet been paid. Only visible when the order is in **Sent, unconfirmed**.
GLOP receipt	Opens the receipt that proves the lines arrived. Visible when the order has that receipt.

Statuses

Draft	The order is not yet sent.
Queued	The sending waits for its turn.
Sent, unconfirmed	The sending has gone out and the receipt that proves it is still missing.
Confirmed in GLOP	The receipt for this order has arrived, or you have confirmed it yourself.
Error	The order did not arrive.
Cancelled	The order has been cancelled.

1. **Create the order**
   Choose the terminal, order type and lines, and the table if it goes to a table.
2. **Press** Send to GLOP
   The order moves to **Queued** and the sending task dispatches it.
3. **Wait for confirmation**
   When the receipt arrives, the order moves to **Confirmed in GLOP**. If it does not arrive, use **It did reach the POS** or **It did not reach the POS**.

> A sent order is never sent again: resending would duplicate the lines on the table's ticket. Only a sent order that has not yet been paid can be cancelled.

Order statuses and what each one means

Draft	This is the initial status. Only here is **Send to GLOP** visible.
Queued	The sending is in the connection's queue and has not yet gone out.
Sent, unconfirmed	The sending went out to GLOP and the receipt that proves it is still missing. Meanwhile no extra request is made: the order can wait as long as needed. Here you see **It did reach the POS**, **It did not reach the POS** and **Void in GLOP**.
Confirmed in GLOP	The GLOP receipt whose source order is this one has arrived. The **Confirmed by** field says whether it was confirmed by **The GLOP receipt** or **One person**.
Error	The order did not arrive, or you marked it with **It did not reach the POS**. The reason is kept in **Last error**.

Cancelled	The order has been cancelled.

A receipt that arrives still open also serves as proof. In the point of sale the order appears as an ongoing ticket, and that ticket links to the order.

Rules and conditions

- A sent order is not queued again.
- Void in GLOP is only for a sent order that has not yet been paid.
- You cannot give a verdict by hand while a cancellation is pending.
- **It did not reach the POS** closes the order as **Error** and nothing is resent automatically.
- The send, cancel and verdict buttons belong to accounting. To open the receipt, the GLOP user permission is enough.

Each order carries its own identifier, created when it is sent and never reused. You see it in the field **orderId sent to GLOP**. The amount is sent in whole cents, and the **Total in cents** is what GLOP receives. If the ticket total does not match the one sent, the connector detects it.

Fields and tabs of the record

POS terminal	The GLOP terminal that receives the order. It is mandatory.
Table	If you enter one, the lines go to that table's open receipt. If empty, the order goes to delivery.
Order type	**In-store pickup**, **Home delivery**, **Eat in** or **Curbside pickup**. It comes as **Eat in**.
Channel	The sales channel GLOP will see. It comes as odoo.
Order note	The GLOP API requires a note on every order; it comes with the text Pedido de Odoo. With more than 15 characters, you need to adjust the "Nota pedido" receipt style in GLOP so that it prints in full.
Customer	It comes as MOSTRADOR.
Already paid	Marks the order as paid in GLOP.
Location	It is calculated and cannot be edited.
Outbox items	Counts the writes for this order sent to GLOP.
Sent on and **Confirmed on**	The dates of the send and of the confirmation.
Simulated	Indicates the order comes from a simulation connection. It is never sent to the real GLOP: it is answered by the simulated point of sale.

The form groups the data into **GLOP**, **Order**, **Customer** and **Delivery address**, with phone, email, address, number, postcode and town. The **Lines** tab holds the detail of what is sent. Each line is sent with the GLOP item and the price in cents.

In the list you can filter by **Unconfirmed**, **Drafts** and **Confirmed**, and group by **State** and **POS terminal**.

The send in Outbox to GLOP

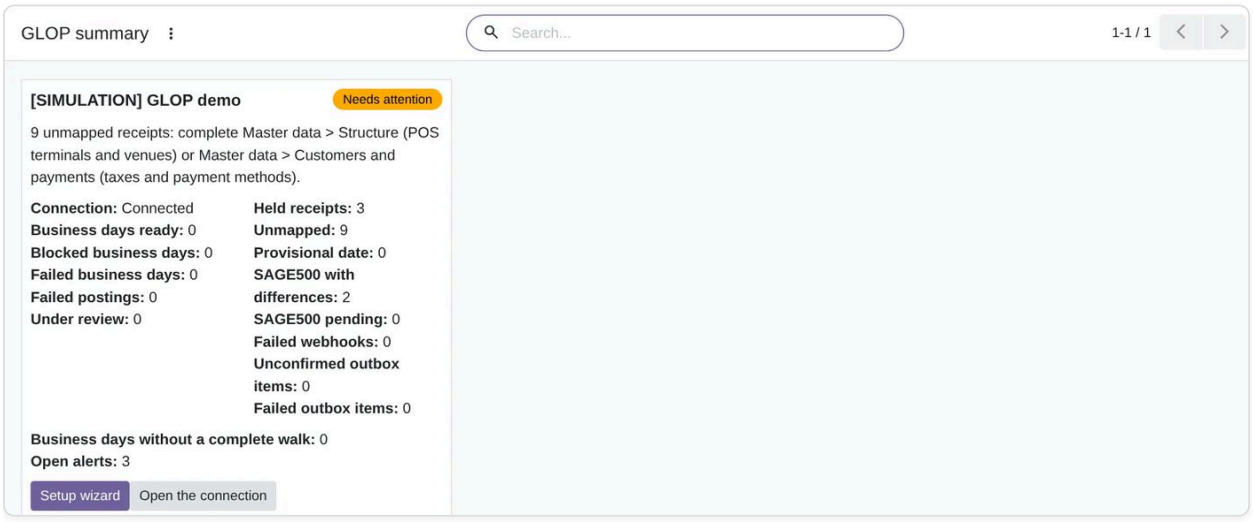
The order generates a send with the operation **Send a table or delivery order**. If it is voided, it generates **Void a table or delivery order**. The send has its own states: **Pending**, **Sending**, **Unconfirmed**, **Confirmed**, **Error** and **Cancelled**. With **Retry**, a send in error goes back to the queue, and the button is not shown in **Confirmed** or **Sending**. **Cancel** is only available when the send is queued or in error. An unconfirmed send is not cancelled blindly, because first you have to check in GLOP whether it was recorded. The content of a send is not edited: it is cancelled and sent again.

End-to-end example

With **Send table and delivery orders** switched on, you create an order for a table and press **Send to GLOP**. The order moves to **Queued**. When the outbox task sends it, it is in **Sent, unconfirmed** and the moment is shown in **Sent on**. When the receipt whose source order is this one arrives, the order moves to **Confirmed in GLOP** and **Confirmed by** shows **The GLOP receipt**. If the receipt does not arrive, you confirm it yourself with **It did reach the POS**. On a simulation connection, the module's simulated point-of-sale answers, without going out to GLOP.

CONNECTOR GUIDE · GLOP

13 Reconciliation and accounting controls

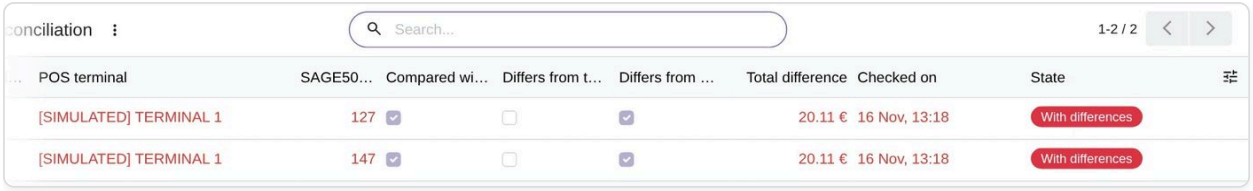


- **Real Odoo 20 screenshot.** The GLOP summary of the [SIMULATION] GLOP demo connection, with the open alerts.

Where the data comes from

- The scheduled task **GLOP: SAGE500 reconciliation** runs every day and checks each terminal and business date.
- The scheduled task **GLOP: integrity of the journal entries** runs every day and does not call GLOP.
- The **Check again** button repeats a check by hand.

The reconciliations are in **GLOP › Operations › Outgoing to GLOP › SAGE500 reconciliation**.



- **Real Odoo 20 screenshot.** The **SAGE500 reconciliation** for the simulation: two days **With differences**, with the total difference of each.

Button and record data

Check again	Repeats that day's check even if the SAGE500 check is switched off. Only the accounting group can see it.
Total difference	SAGE500 total minus what was posted, or minus the receipts if there is no entry yet.
Compared with what was posted	Ticked if the day had posted entries when it was checked.
Differs from the receipts and **Differs from what was posted**	Checkboxes showing which side SAGE500 does not match.
Days in a row without SAGE500	How many days in a row the export has not been available.

States

Pending	Not yet checked, or a response is missing.
It balances	SAGE500 matches what was compared.
With differences	There are differences.
Not available	The merchant cannot obtain the SAGE500 export.
Off	The check is switched off on the connection.

How to review a day with differences

1. **Filter the list**
   Use the **With differences** filter.
2. **Open the record**
   Look at the **By VAT rate** and **By sale type** tabs.
3. **Repeat the check**
   Press **Check again** once you have fixed the cause.

> Reconciliation records are read-only: they cannot be created or changed by hand, only the connector generates them.

What the control compares

Each check cross-matches three sides of a terminal and a business date: the GLOP SAGE500 export, the completed receipts for the day, and what has already been posted. The receipts include the held and excluded ones, because they are what GLOP recorded. The export is rounded per document and per VAT rate.

States and what each means

Pending	Initial value. It is checked again if a response was missing or if an entry was written since the last verification.
It balances	SAGE500 matches what was compared.
With differences	SAGE500 does not match the receipts or what was posted.
Not available	The merchant cannot obtain the export, or GLOP did not respond after the retries. Consecutive days are counted.
Off	The check is switched off on the connection. The scheduled task does not call GLOP, but the button does.

Rules and conditions

- A check already answered whose data has not changed does not call GLOP again, unless it is forced.
- The comparison with what was posted uses the day's posted and reversed entries. If the refund mode is net, refunds are folded into the sales.
- With the lock enabled, a day with nothing posted is not posted if its check has differences or could not be done. With automatic posting, the day is posted again once the check releases it.
- The integrity check is read-only. If it fails, the day moves to the error state with the detail; when it comes back clean, it returns to the availability review.

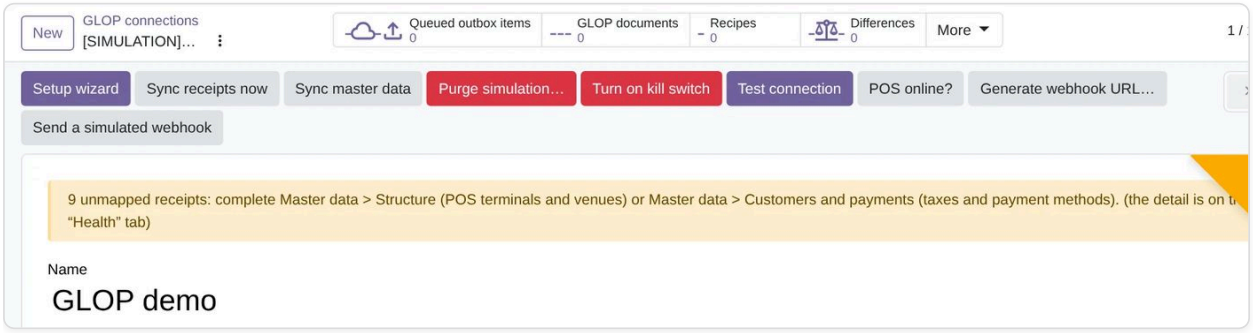
Fields and tabs of the record

Business day and **Check**	The two groups on the form.
By VAT rate and **By sale type**	Tabs with the breakdown by VAT rate and flow, and by SAGE500 type and VAT rate.
SAGE500 response	Tab with the text of the response.
Notes	Tab with the notes and the **Unavailability reason**, if the export could not be obtained.
SAGE500 rows and **SAGE500 documents**	How many rows and documents the export returned.
Checked on and **Run**	When and in which run it was checked.

An example

In the **GLOP summary** of the [SIMULATION] GLOP demo connection there are 2 reconciliations with differences and none pending. Open **SAGE500 reconciliation**, apply the **With differences** filter, open one of the two and check in the **By VAT rate** tab which key differs. Once it is fixed, press **Check again**.

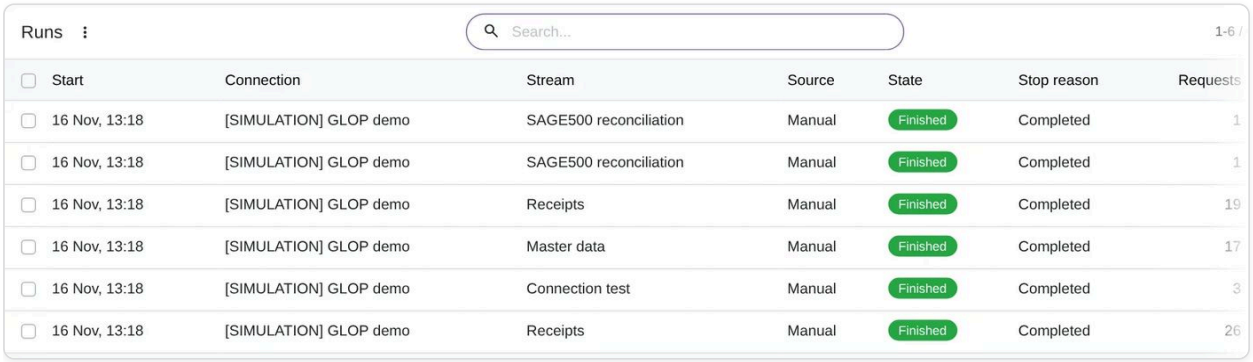
14 Setup, simulation and monitoring



- **Real Odoo 20 screenshot.** The [SIMULATION] GLOP demo connection with the **Setup wizard** button.

Where each piece of data comes from

DATA	WHERE IT COMES FROM
Runs	One per pass of a flow. Created by the scheduled tasks and the connection's buttons. Visible in **GLOP › Audit › Sync › Runs**.
Cursors	Updated automatically as each flow is processed. Visible in **GLOP › Audit › Sync › Cursors**.
Calls to GLOP	Saved on each call. Visible in **GLOP › Audit › Issues › API calls (errors)**.
Received webhooks	They arrive when GLOP sends a notice. The scheduled task **GLOP: process received webhooks** processes them every 5 minutes.



- **Real Odoo 20 screenshot.** The **Runs** of the simulation, with the flow, the origin, the status and the **Stop reason** of each one.

Wizard buttons

Test connection	Checks the connection and leaves the result inside the wizard.
Sync master data	Brings in venues and terminals, which arrive switched off.
Activate POS terminals	Switches on the point-of-sale terminals of the chosen venue.
Run a test without posting	Imports the last 3 business days of each activated terminal, without posting.
Save policies	Saves on the connection the policies chosen in step 7.
Start	Closes the setup in step 9.

Create a simulation connection...	Opens the simulation wizard.

Run statuses

- **In progress**
- **Finished**
- **Partial**
- **Failed**
- **Skipped**

Steps

1. **Open the connection**
   Go to **GLOP › Configuration › GLOP connections** and press **Setup wizard**.

2. **Walk through the steps**
   Use **Next** and **Previous**. There are nine steps: step 6 is the test without posting, and step 9 is the go-live.

3. **Check the monitoring**
   Review **Runs**, **Cursors** and **Received webhooks** in **GLOP › Audit › Sync**.

> While there is no **Go-live date**, no ticket comes in, and business days before that date are never requested. Set it in step 7 of the wizard or in the **Sync** tab of the connection.

The nine steps of the wizard

STEP	WHAT IT DOES
Step 1: credentials or simulation	Opens the credentials of the connection or creates a simulation connection.
Step 2: test the connection	Checks that GLOP responds.
Step 3: master data	Brings in venues and terminals, which arrive switched off.
Step 4: venues	Creates the journal and the bridge accounts that are missing, and activates the terminals.
Step 5: taxes	Fills in the suggested tax for each GLOP rate, without confirming anything.
Step 6: test without posting	Imports the last 3 business days of each activated terminal without posting.
Step 7: policies	Saves full invoices, returns, stock, SAGE500 control and the go-live date.
Step 8: webhook	Generates the webhook URL and activates it.
Step 9: go-live	Leaves ticket sync switched on and automatic posting switched off.

Rules and conditions

- **Credentials...** is not visible on a simulation connection.
- **Create the GLOP journal** only shows while the venue has no journal, and it does not touch an existing one.
- **Open POS terminals** needs a venue selected.
- **Use suggestions** only shows if there are rates with a suggestion. It fills in the tax without confirming it: you confirm them yourself from **Open taxes and confirm them**.
- **Activate webhook** is hidden if the webhook is already active or if you have not generated the URL yet.
- While the webhook is off, the controller responds 404.
- Nothing is imported when activating terminals: the import starts at step 6 or with the scheduled task.
- With **Start**, automatic posting stays off and the parallel period is noted in the chatter.
- The wizard warns about the conflict in **Owner of the ingredient stock**: if the GLOP point of sale and the Odoo bills of materials both break down the same sale, the stock would be deducted twice.
- Under **Returns**, the **Net (check the 303 first)** option asks you to check the 303 return first.

How to check the state of a connection

State of the connection	The state of the connection, copied from the connection.
Credentials saved	Wizard checkbox: whether credentials already exist.
Unmapped rates and **Unconfirmed rates**	How many tax rates are still to be mapped or confirmed.
Unmapped payment methods	How many payment methods are still to be mapped.
Test run	Checkbox: whether the test without posting has already been done.
What this venue is missing	List of what is missing in the chosen venue.
URL sent to GLOP	Checkbox: whether you have already told GLOP support about the URL.

Run statuses, stop reasons and sources

A run starts **In progress** and ends **Finished**, **Partial**, **Failed** or **Skipped**. The **Source** can be **Scheduled**, **Manual** or **Webhook**.

The **Stop reason** says why it stopped: **Completed**, **Request limit**, **Time limit**, **Circuit open**, **Kill switch**, **Authentication failed**, **Wrong credentials**, **Configuration error**, **GLOP asks to wait (429)**, **Pagination ignored**, **Another run in progress**, **Error**, **Nothing to do** and **Paused: FlexigoTech Cloud**.

Cursor statuses

Each cursor stores how far a flow has been read: **Pending**, **Walking**, **Complete**, **Unstable**, **Error** or **Off**. **Walk again** forgets what was read and starts over; the button does not show on a **Pending** cursor.

Received webhook statuses

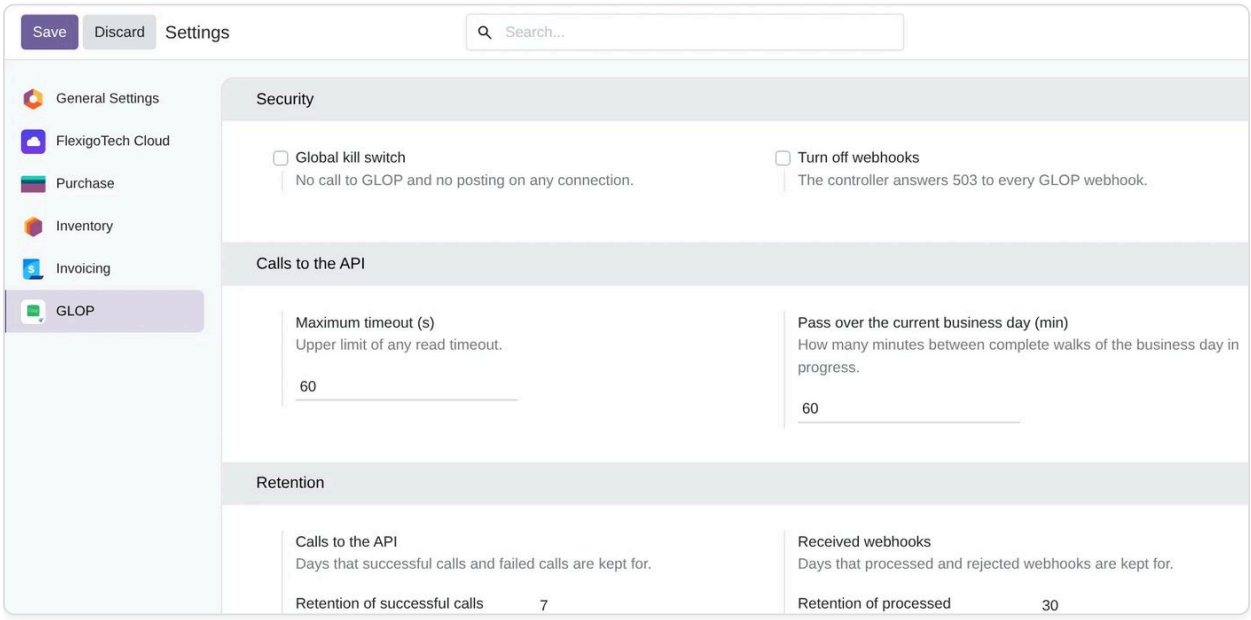
A notice starts **Pending**, moves to **Processing** and ends **Processed**. It can also end as **Ignored: open receipt**, **Ignored: does not exist in GLOP**, **Ignored: unknown type**, **Rejected: invalid** or **Error**. **Retry** only shows on **Error** items, or on **Ignored: does not exist in GLOP** items that have an **ID in GLOP**.

API calls

Each call stores its **Classification**: **Fine**, **Token rejected**, **Request rejected**, **Not found**, **Transient error**, **Timed out**, **Connection error**, **Unexpected response** and those blocked before leaving. The **Redacted excerpt** holds no secrets or personal data. The **Blocked before leaving** filter groups them.

General settings

Global kill switch	No call to GLOP and no posting on any connection.
Turn off webhooks	The controller responds 503 to every GLOP webhook.
Maximum timeout (s)	Upper limit for any read timeout.
Pass over the current business day (min)	How often, in minutes, the current business day is walked in full.
Retention of successful calls (days) and **Retention of failed calls (days)**	Days that successful calls (7 by default) and failed calls (90) are kept. In the same block, **Retention**, are the periods for received webhooks: 30 days for processed ones and 90 for rejected ones.

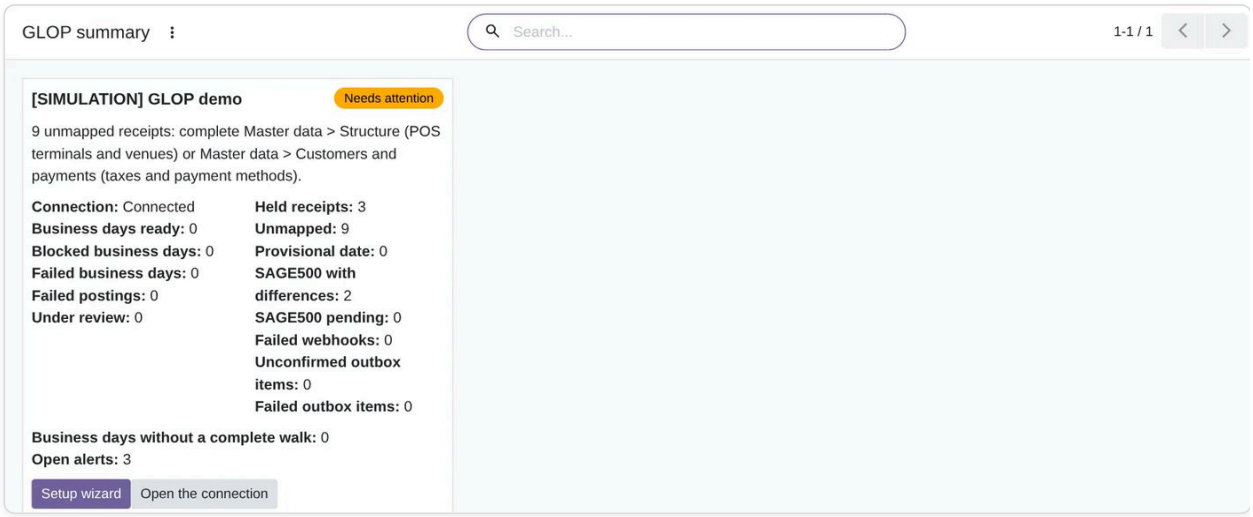


- **Real Odoo 20 screenshot.** GLOP settings with the **Global kill switch** and **Turn off webhooks**.

End-to-end example

1. **Check the connection notice**

   The connection [SIMULATION] GLOP demo is **Connected** and warns of 9 unmapped tickets.



- **Real Odoo 20 screenshot.** The **GLOP summary** of the simulation, with 9 unmapped tickets.

2 Complete what is missing

The notice says to complete **Master data** under **Structure** and under **Customers and payments**.

3 Review the monitoring

Open **Runs** and **Cursors**. Failed calls appear under **API calls (errors)**.

> To delete simulated data you must type DEMO in **Type DEMO to confirm**. **Purge** deletes the entries, tickets and business days of the simulation. It does not do so if a simulated delivery note is linked by hand, because it could be a real delivery.

CONNECTOR GUIDE · GLOP

15 The data that goes out and comes back

DATA	DIRECTION	WHEN
Receipts for each business day	From GLOP to Odoo	**GLOP: sync receipts by business day**, every 10 minutes, or the **Sync receipts now** button
GLOP notifications (webhooks)	From GLOP to Odoo	On arrival, if GLOP has registered them. **GLOP: process received webhooks** processes them every 5 minutes
Master data: venues, terminals, taxes, families, products and customers	From GLOP to Odoo	**GLOP: master data**, every day, or the **Sync master data** button
Pending purchase documents to be fulfilled	From GLOP to Odoo	**GLOP: pending purchase documents**, every hour
SAGE500 export	From GLOP to Odoo	**GLOP: SAGE500 reconciliation**, every day
Customers	From Odoo to GLOP	With **Send customers to GLOP** switched on (it comes switched off): the one you send with **Send to GLOP** goes out with **GLOP: sending customers**, every 15 minutes

DATA	DIRECTION	WHEN
Purchase orders, delivery notes, stock counts, recipes and table or delivery orders	From Odoo to GLOP	Only when you press their button and with their checkbox switched on in the connection; all of them come switched off. They go out with the same task, **GLOP: sending customers**

API calls to GLOP go through FlexigoTech Cloud. There, your GLOP API credentials are stored encrypted; they are not kept in Odoo, and the service alerts you when a call fails. A FlexigoTech Cloud subscription is required, with a 14-day trial and no card needed. GLOP notifications do not go through the cloud: they arrive directly at the address generated by **Generate webhook URL...** on the connection, so your Odoo must be reachable from the internet. GLOP registers them from its panel. They are not signed and are not retried: if one is lost, the per-business-day read every 10 minutes still brings the receipt.

> If you activate the connection's kill switch or the global one, no call is made to GLOP.

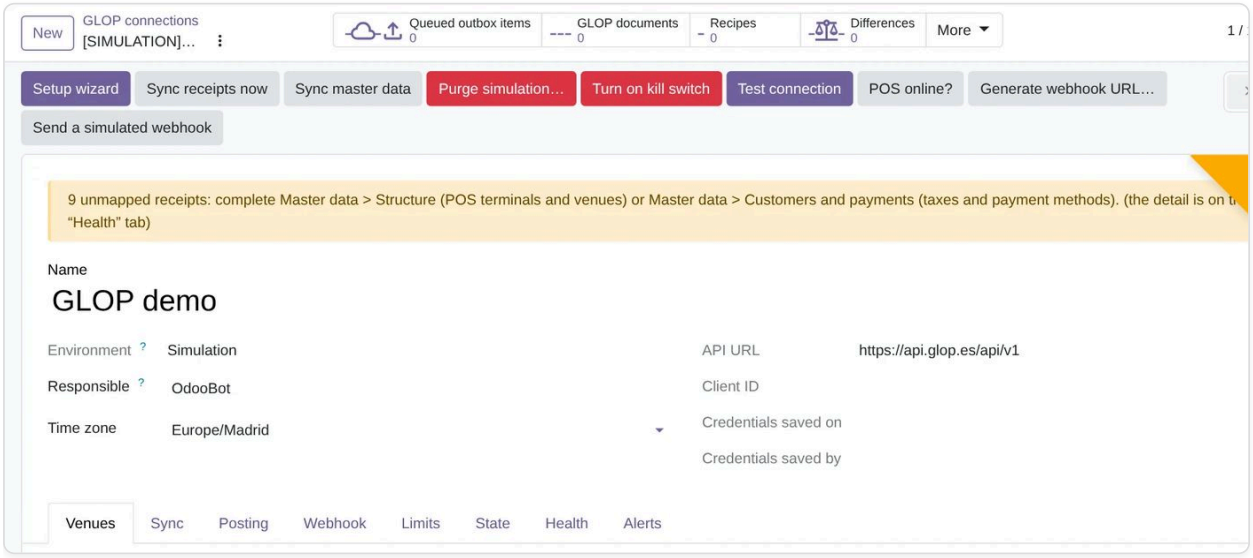
CONNECTOR GUIDE · GLOP

16 What runs automatically

TASK	FREQUENCY	WHAT IT DOES
GLOP: sync receipts by business day	Every 10 minutes	Brings in the receipts for each business day.
GLOP: process received webhooks	Every 5 minutes	Processes the GLOP notifications that have arrived.
GLOP: business days and posting	Every 15 minutes	Evaluates the business days and their posting.
GLOP: sending customers	Every 15 minutes	Sends to GLOP what is queued: customers, purchase and stock documents, recipes and table or delivery orders. It only works with connections that have at least one sending checkbox switched on.
GLOP: pending purchase documents	Every hour	Brings in the purchase documents pending fulfilment.
GLOP: master data	Every day	Updates the master data.
GLOP: SAGE500 reconciliation	Every day	Compares the receipts and entries with the SAGE500 export.
GLOP: integrity of the journal entries	Every day	Checks the posted journal entries.
GLOP: deep pass over old business days	Every day	Goes over old business days to detect missing receipts.
GLOP: night maintenance	Every day	Checks the health of the connections and raises or closes alerts.

Frequencies are changed in **Settings › Technical › Automation › Scheduled Actions**, with developer mode activated.

17 The connection options and their defaults



- **Real Odoo 20 screenshot.** The simulation connection form, with its header buttons and its tabs.

OPTION	WHEN CREATED	WHAT IT DOES
Environment	**Production**	Choose between **Production** and **Simulation**. It cannot be changed once the connection already has venues or receipts.
Sync receipts	Switched on	Allows reading receipts from GLOP.
Sync master data	Switched on	Allows bringing structure, catalogue and tax data from GLOP.
Sync customers	Switched on	Allows bringing customers from GLOP.
Sync vendors and purchases	Switched off	Reads from GLOP vendors, purchase families, packaging and pending purchase documents. It writes nothing to GLOP.
Sync sale formats and recipes	Switched off	Reads from GLOP the sale formats and recipes. It writes nothing to GLOP.
Compare the stock with GLOP	Switched off	Reads stock levels from GLOP and places them next to those in Odoo. It only shows the difference, it does not correct stock.
Send customers to GLOP	Switched off	Lets Odoo create and update customers in GLOP.
Send purchase orders to GLOP	Switched off	Allows sending Odoo purchase orders to GLOP.
Turn receipts of goods into GLOP delivery notes	Switched off	Allows turning the order of a validated receipt of goods into a GLOP delivery note.
Send stock counts to GLOP	Switched off	Allows sending inventory counts to GLOP and deleting stock documents.
Send recipes to GLOP	Switched off	Allows writing a product's recipe to GLOP.
Send table and delivery orders	Switched off	Allows sending orders to the point of sale, to a table or to delivery, and cancelling them.
Go-live date	Empty	Date from which business days are requested. Business days earlier than this are never requested.

OPTION	WHEN CREATED	WHAT IT DOES
Pass window (days)	3	Number of days looked back over.
Deep pass (days)	35	Number of days the deep review covers.
Hours until the business day is settled	30	Hours that pass before the business day is considered settled.
Maximum wait for open receipts (h)	48	Maximum hours an open receipt may wait.
Cut-off hour (provisional date)	6	Hour of the day used as the cut-off for the provisional date.
Post automatically	Off	With it off, each business day is posted when you press **Post**.
Full invoices	**Per document**	It can be changed to **Journal**.
Returns	**Correction labels**	The other option is **Net (check the 303 first)**.
Inconsistent VAT	**Hold**	The other option is **Recalculate**.
Extras	**Automatic**	You can choose **Add up to** or **Included**.
Credit note with positive amounts	**Hold**	The other option is **Invert sign**.
Editing after posting	**Review**	The other option is **Publish adjustment**.
SAGE500 check	**Warn**	You can set it to **Off** or **Block**.
Rounding tolerance	0.01	Rounding difference that is accepted.
Maximum rounding per journal entry	0.10	Maximum rounding accepted in a journal entry.
Stock	**No stock**	The other option is **Products sold**: Odoo moves the stock of what GLOP sells.
Owner of the ingredient stock	GLOP	Who deducts the ingredients of a compound product: the GLOP point of sale or Odoo's bills of materials.
Webhook active	Off	Turns on the reception of notices from GLOP.

The checkboxes for what is synced and sent, and the business day timings, are on the **Sync** tab; the documents, the checks and the stock, on **Posting**. **Environment** is at the top, next to the name, and **Webhook active**, on the **Webhook** tab. The checkboxes for sending to GLOP start off and are turned on one by one. While one is off, the connector rejects that send before it leaves the network. These sends have only been tested with the module's simulation and with one-off calls to a GLOP test account in the cloud, not through FlexigoTech Cloud.

What is done by hand

WHERE	BUTTONS
Header	**Setup wizard**, **Sync receipts now**, **Sync master data**, **Test connection**, **Credentials...**, **Turn on kill switch** or **Turn off kill switch**, **Reset circuit** (with the circuit open), **POS online?** and **Generate webhook URL...** or **New webhook URL...**. On a simulation connection, **Purge simulation...** and **Send a simulated webhook** also appear, and **Credentials...** does not. These belong to the GLOP administrator; the accountant can also open the wizard.
Header, depending on the checkboxes on Sync	**Fetch purchase documents** and **Fetch a document by number**, with **Sync vendors and purchases**; **Fetch recipes from GLOP**, with **Sync sale formats and recipes**; **Compare the stock with GLOP** and **Differences report**, with the checkbox of the same name; **Fetch tables from GLOP**, with **Send table and delivery orders**.
Next to the credentials	**Check in FlexigoTech Cloud** and **Delete from FlexigoTech Cloud**, only for the Odoo administrator and with the connection through FlexigoTech Cloud.
Webhook tab	**View received webhooks**, **Invalidate the previous URL** and **Revoke webhook URL**
Health tab	**Run maintenance now**

Limits	Stores **Maximum requests per run** (600), **Maximum seconds per run** (240), **Minimum interval between requests (ms)** (150), **Read timeout (s)** (30) and **Maximum tokens per hour** (4). They come with these values.
Webhook	Stores **Webhook active**, **Allowed networks (CIDR)** (empty: any origin), **Webhook limit every 10 min** (1200) and **Group webhooks (s)** (120).
State	Shows **Connection**, **Circuit**, **Last attempt**, **Last success**, **Last receipt seen**, **Last webhook**, **API version** and the **Last error** group.
Health	Shows the counts of what is pending: business days, held or unmapped receipts, postings, sends and alerts.
Alerts	Choose in **Alert users** who receives the connection's alerts, which arrive as activities.

18 Security and personal data

- The GLOP connection secret is stored separately, in a table locked to almost all users. It does not appear in views, exports, logs or the chatter.
- The GLOP API credentials are stored encrypted on FlexigoTech Cloud servers and do not stay in your Odoo. Calls to GLOP go through FlexigoTech Cloud, which alerts you when one fails.
- Whoever activates FlexigoTech Cloud accepts, on behalf of their company, that the database sends the data listed in **See the data that is sent**. Tick the box, accept the subscription terms and the data processing agreement, and enter the email address of the person accepting. The company name, that email, the date and the version of the accepted text are stored.
- The module has three permission levels: user, accountant and GLOP administrator. The user views. The accountant decides: opens the **Setup wizard**, posts, approves or rejects reviews, retries sends and fetches documents. The buttons **Sync receipts now**, **Sync master data**, **Turn on kill switch** and **Turn off kill switch** belong to the GLOP administrator. Only the Odoo administrator sees **Check in FlexigoTech Cloud** and **Delete from FlexigoTech Cloud**.
- In **Settings**, in the GLOP block, there are two global switches: **Global kill switch**, which blocks every call to GLOP and every posting on any connection, and **Turn off webhooks**, which makes the server reply 503 to every GLOP notice. Each connection also has its own kill switch.
- GLOP notices are not signed and are not retried. They are protected by a long key within the notice address and, if you fill it in, by the **Allowed networks (CIDR)** list on the connection. Only a digest of that key and its last 4 characters are stored, and the full address is shown only once.
- Personal data (email, tax ID, phone, address and customer name) is hidden in call logs. In **Settings**, in the **Retention** block, you choose how much is kept. By default, successful calls are kept for 7 days and failed ones for 90; processed webhooks for 30 days, and rejected ones for 90. What GLOP sent for each receipt is kept in full unless you set a period in months. The **GLOP: night maintenance** task deletes each day whatever is older than those periods.
- The connector creates only journal entries, never invoices or receipts. A journal entry posted from GLOP cannot be edited or deleted: to change it, it must be reversed and posted again.
- If you do not subscribe when the trial ends, the service stops and nothing is charged. Your data stays in your Odoo.

What gets recorded

Each task run and each call to GLOP is stored as a record, with a code that says why it failed. A finished receipt keeps its versions read-only, and a human decision such as accepting a recalculation is kept in that history. Runs are in **GLOP › Audit › Sync › Runs** and failed calls in **GLOP › Audit › Issues › API calls (errors)**. Open alerts appear on the connection and in the **GLOP summary**.

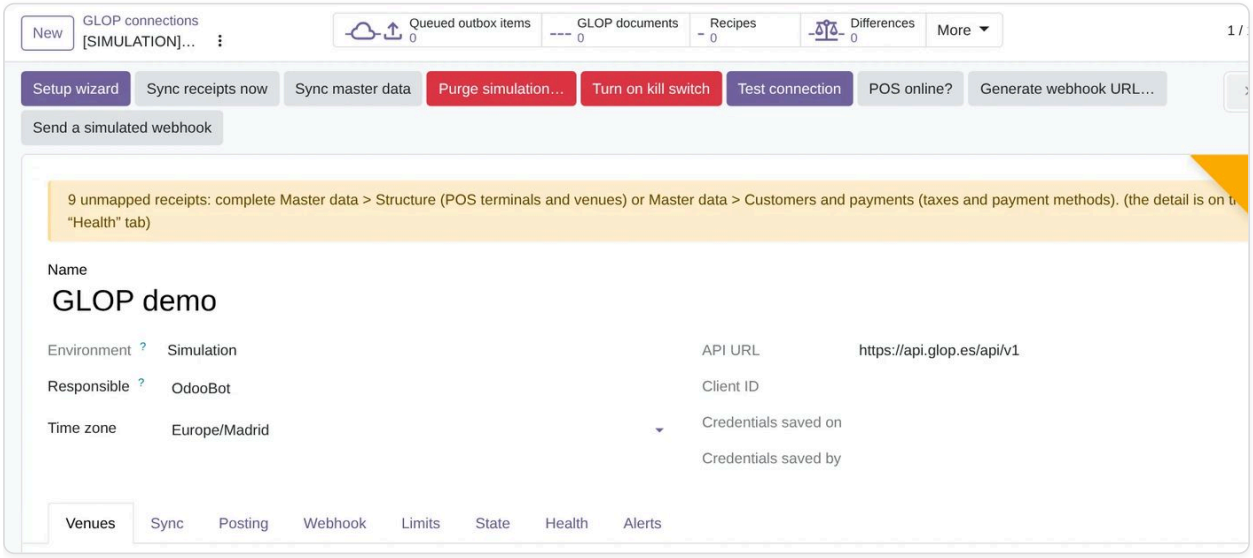
19 Several connections and several companies

Each connection in **GLOP › Configuration › GLOP connections** has its own company, time zone, API address and options. The posting switches, the sends to GLOP and the sync are turned on or off connection by connection, and each has its own kill switch.

Locations, terminals, taxes and payment methods are linked to the connection's company. Each button also checks that the record belongs to a company you have access to. A GLOP user from another company cannot press this company's buttons.

A simulation connection can only be in a company marked as demo. This way simulated data never touches a real company.

On FlexigoTech Cloud, each module has its own subscription for the production database. Each plan allows a number of GLOP connections: up to 3 with Standard and up to 10 with Pro. Test and pre-production databases do not pay.



- **Real Odoo 20 screenshot.** The simulation connection form, with **Sync receipts now** and **Test connection** in the header.

20 If something fails

The connection buttons do not call GLOP	Without the FlexigoTech Cloud trial or subscription, the connector does not talk to GLOP. Check the status in **Settings**, in the FlexigoTech Cloud block, and press **Check now** if you have just activated it.
The credentials do not work	Open **Credentials...** on the connection and re-enter the Client ID and the secret, without spaces. For security Odoo never shows the stored secret. Then press **Test connection**.
An action is missing due to permissions	Some actions require a specific group, and whoever handles the accounting is the one who posts or retries. If Odoo shows "This action requires the "%(group)s" group.", ask an administrator for that permission.
A posting ended with an error	Press **Retry** on the posting: it runs the process again from the pre-check. If the error is of the "LEGAL_GUARD" type, only a GLOP administrator can retry it, and its cause must be fixed first.

An order sent to GLOP did not arrive	If GLOP did not receive it, the order is closed as an error with **It did not reach the POS**. The connector does not resend it on its own.
You want to see what GLOP replied	Check **GLOP › Audit › Issues › API calls (errors)** and **GLOP › Audit › Sync › Received webhooks**. A webhook with an error is processed again with **Retry**; API calls are only a record. You decide how long they are kept in **Settings**, in the **Retention** block.

If the database has no FlexigoTech Cloud trial or subscription, or the connection does not yet have its credentials stored there, the connector does not call GLOP: each run ends with the reason **Paused: FlexigoTech Cloud**, which you see in **Runs**. As soon as the service and credentials are available, it calls again without you doing anything; to avoid waiting for the module to renew the permission, press **Check now** in **Settings**. In a neutralised copy of the database, such as an Odoo.sh staging branch, all connections are left with the kill switch on and the webhook off: the copy does not call GLOP or receive its notices.

Other notices you may see

"A connection test is already running for this connection."	Wait for the running test to finish and press **Test connection** again.
"The connection has no venues: sync the master data first."	Click **Sync master data** on the connection and repeat the action.
"A simulation connection has no GLOP credentials."	A simulation connection does not use credentials: there is nothing to enter.
"Enter the GLOP Client ID (no spaces)."	Enter the Client ID again with no spaces at the start, at the end or inside it.
"Enter the GLOP secret (no spaces). For security the saved one is never shown."	Enter the full secret; you cannot see the one that was already saved.
"Before saving the secret, an administrator has to accept that the GLOP credentials and the calls to GLOP go through FlexigoTech Cloud. Nothing was sent."	An administrator must accept sending data to FlexigoTech Cloud before the secret can be saved.
"Only a failed posting is retried (... is in state ...)."	Retry only works for postings in **Error** state. Check the state of the one you have opened.
"Enter the reason for excluding GLOP receipts."	Enter the reason before excluding the receipts.
"Select at least one GLOP receipt."	Tick one or more receipts in the list before running the action.

21 Other ways of doing it

METHOD	WHAT IT INVOLVES
By hand from the GLOP panel	Someone reviews each business day in GLOP and records the sales, payments and VAT in Odoo. This is what happens when there is no connection between the two tools.
Odoo alone	Odoo 20, Community and Enterprise, do not ship any module with "glop" in the name, so the connection to GLOP does not come as standard. Checked in the Odoo 20 code on 16-11-2026.
This connector	Reads GLOP receipts by business day and posts them in Odoo with their VAT. It brings in the master data and, if you enable them, recipes and purchase documents. It requires a FlexigoTech Cloud subscription, with a 14-day trial and no card required.

If you already record sales with another tool and do not want to load the past into Odoo, set the **Go-live date** on the **Sync** tab of the connection, or at step 7 of the **Setup wizard**. Earlier business days are never requested from GLOP, and no receipt comes in until you set it.

22 Plans and pricing

€9

License per Odoo version.

+ FlexigoTech Cloud

STANDARD

€49/month

or €490 a year

First response within 1 working day

PRO

€98/month

or €980 a year

First response within 4 working hours

Video call if needed

14-day trial, no card **Prices exclude VAT**

The module license is per Odoo version. FlexigoTech Cloud is paid monthly or yearly, and the 14-day trial requires no card. With Standard, up to 3 connections; with Pro, up to 10.

23 Frequently asked questions

Does it work without FlexigoTech Cloud?

No.

What does FlexigoTech Cloud do in this connector?

It sends the calls to the GLOP API and stores your GLOP API credentials encrypted on its servers, so they are not kept in your Odoo. It also alerts you when a call fails.

Is there a free trial?

Yes, 14 days free and no card required. If you do not subscribe at the end, the service stops and nothing is charged.

How much does it cost?

The module license costs €9 per Odoo version. FlexigoTech Cloud is paid separately: the Standard plan costs €49 a month or €490 a year, and Pro costs €98 a month or €980 a year, excluding VAT.

How many connections does each plan allow?

Each plan allows a number of connections with GLOP: up to 3 with Standard and up to 10 with Pro.

What is the difference in support between the plans?

The first response arrives within 1 working day with Standard and within 4 working hours with Pro, which includes a video call if needed.

Does it work on Odoo Online?

No. Odoo Online does not allow installing third-party modules.

Which Odoo versions is it published for?

Odoo 17, 18, 19 and 20. It is installed on Odoo.sh or on your own server.

Does it need Odoo Enterprise?

No. Everything it needs is in Odoo Community: Invoicing, Inventory, Purchase, Discuss, Analytic and the Spanish localisation, plus FlexigoTech Cloud. The test with a real GLOP point of sale was run on Odoo 19 Enterprise.

What do I need from GLOP?

A GLOP account with its API credentials (Client ID and secret), which the connector stores encrypted in FlexigoTech Cloud and not in your Odoo. In addition, during setup, a full dump of your point of sale to GLOP's cloud, which GLOP support or your distributor carries out.

Who can connect it?

An Odoo administrator, who is the one who accepts the FlexigoTech Cloud consent and saves the credentials.

How often do receipts arrive?

The **GLOP: sync receipts by business day** task reads them every 10 minutes.

Can I request receipts manually?

Yes, with the **Sync receipts now** button on the connection.

Are GLOP webhooks required?

No. The reading by business day every 10 minutes is the one that governs and brings in all receipts. If GLOP registers the webhooks from its panel, using the address that Odoo generates on the connection, the connector is notified sooner. They are not signed and are not retried: a lost notice does not lose the receipt.

What if GLOP resends receipts that were already there?

They are not duplicated. A receipt that is already posted and comes back with the same content creates no version, no revision and no journal entry. This happens, for example, when GLOP support uses "Enviar ventas" at go-live.

How often does master data arrive?

Once a day, with the **GLOP: master data** task.

How often are business days evaluated and posted?

The **GLOP: business days and posting** task evaluates each business day every 15 minutes. While **Post automatically** is switched off, which is the default, you decide when to post.

What entry does it generate?

One entry per terminal and business day, not per calendar day. Base and VAT come from GLOP's figures per document and VAT rate, without recalculating them.

Where do payments go?

Each GLOP payment method goes where it belongs: a cash or card clearing account, another receipt, an expense (invitations and internal consumption) or a customer account.

What happens to what was there before it was installed?

Business days earlier than the **Go-live date** are never requested. Until you fill it in, the connection receives no receipts.

What happens with credit sales?

The amount stays open on the customer's account and is reconciled when GLOP records the payment.

What happens if GLOP changes a receipt that was already posted?

The connector calculates the difference and proposes it as an adjustment entry for a person to review, indicating what changed.

What happens with a receipt that does not balance?

It is held back with a code and an explanation, and the rest of its business day is posted anyway.

Does the connector decide the accounts and taxes?

No. It proposes, flags what is missing and confirms nothing on your behalf: those assignments are your accountant's decision.

Does it create invoices, receipts or point-of-sale orders in Odoo?

No. It also does not close or modify any GLOP tax document, does not touch pre-tickets, does not charge tables and does not add charges to rooms.

Does it show the VERI*FACTU status or the QR code?

No, because the GLOP API does not provide them.

Does it include the cash close or the Z report?

No.

Which taxes and currencies does it support?

Only mainland Spain VAT and euros. It does not support equivalence surcharge, IGIC or IPSI.

And with periodic stock valuation?

With periodic stock valuation, the connector does not post cost of sales.

Can it send things to GLOP?

Yes: table or delivery orders, customers, purchase orders, receipts converted into delivery notes, stock counts and recipes. Each send has its own checkbox on the connection, switched off when created, and while it is off the attempt is rejected before any network call goes out. These sends have only been tested with the module's simulation and with ad hoc calls to a test account in GLOP's cloud, not through FlexigoTech Cloud.

What happens to the stock of what was sold?

If you set **Stock** under **Products sold** (it comes as **No stock**), the connector generates stock moves from each posting. By default, GLOP owns the ingredient stock: Odoo moves the product exactly as GLOP sold it, without breaking down kits.

Can I test it without touching a real company?

Yes, with a simulation connection. It can only sit in a company marked as demo, and it never talks to the real GLOP.

Can I have several connections or several companies?

Yes. Each connection has its own company, time zone, API address, its own options and its own kill switch.

Can I stop everything at once?

Yes. If you enable the global kill switch or the one on a connection, no call is made to GLOP.

What happens if FlexigoTech Cloud is not available?

While it cannot make calls, the connector neither reads from nor sends anything to GLOP. When it is back, the business-day reading reviews the previous days (**Pass window (days)**) and brings in whatever is missing. If what is missing is the trial or the subscription, each run ends with the reason **Paused: FlexigoTech Cloud**.

What about test databases?

Each module has its own subscription for the production database. Test and staging databases do not pay.

24 Glossary

Connection	Each link between an Odoo company and a GLOP account. It has its own options and is created in **GLOP › Configuration › GLOP connections**.
Environment	Connection field that distinguishes **Production** from **Simulation**. It cannot be changed once the connection already has establishments or receipts.
Simulation	Test connection that never talks to the real GLOP. It can only sit in a company marked as demo.
Establishment	Point of sale whose accounting receives the receipts. It arrives with **Sync** unticked and posts nothing until someone configures it.
Terminal	A point-of-sale till or terminal at an establishment. Each trading day on a terminal gives rise to one journal entry.
Trading day	A sales period on a terminal. The journal entry is made per terminal and trading day, not per calendar day.
Receipt	A sales document received from GLOP. It can be **Held**, **Pending mapping**, **To post** or **Posted**, among other statuses.
Held receipt	A receipt whose amounts do not reconcile. It is set aside with a code and an explanation, and the rest of its trading day is still posted.
Mapping	The link between a GLOP payment method or VAT rate and its Odoo account or tax. Without it, receipts stay unmapped.
Posting	The record that groups a trading day's journal entry, with its lines, totals and VAT, and its status.
Master data	Establishments, terminals, taxes, families, products and customers that the connector pulls from GLOP once a day or when you press **Sync master data**.
Go-live date	The date from which receipts are requested from GLOP. Without it, no receipts come in.
Recipe (bill of materials)	The recipe of a composite product: its ingredients with quantity and cost. It can be converted into a kit-type bill of materials.
Webhook	A notice that GLOP sends to your Odoo when something changes, if you have registered it from its dashboard. It carries no signature and is not retried, and it is processed every 5 minutes.
Kill switch	A switch that cuts every call to GLOP and every posting. It exists per connection and also globally.
Circuit	The connection status that opens after repeated failures so that calls to GLOP stop. It can be reset with **Reset circuit**.
SAGE500	A GLOP export that the connector compares with receipts and with what has been posted in **SAGE500 reconciliation**.
FlexigoTech Cloud	The online service through which calls to GLOP are routed. It requires a subscription, with a 14-day trial and no card needed.
Credit sale	A sale whose amount stays outstanding on the customer's account until GLOP records the payment.

25 Checklist and contact

- [ ] You have a GLOP account with its Client ID and secret, and GLOP or your distributor has done the full dump of your till to the GLOP cloud.
- [ ] Your Odoo is version 17, 18, 19 or 20 and is not Odoo Online, which does not support third-party modules.
- [ ] The module is installed, together with Invoicing, Inventory, Purchase and the Spanish localisation.
- [ ] In **Settings** you have activated FlexigoTech Cloud with the 14-day trial, and the status reads "Connected to FlexigoTech Cloud".
- [ ] You have created the connection in **GLOP › Configuration › GLOP connections** with its **Environment** and saved the credentials with **Credentials...**.
- [ ] **Test connection** returns "Connection successful" and the connection is **Connected**.
- [ ] You have pressed **Sync master data** and set the journal and accounts on each establishment before ticking **Sync**.
- [ ] You have mapped the payment methods and taxes, and confirmed the latter.
- [ ] You have set the **Go-live date**, either on the **Sync** tab or in step 7 of the **Setup wizard**: without it, no receipt comes in.
- [ ] You have checked the connection's options: **Post automatically** and the sending to GLOP are switched off, and you switch them on yourself.
- [ ] A test trading day appears in **GLOP › Operations › Day to day › Business days** and **Preview** shows the expected entry.

If you would like us to set it up with you, write to us. With the Pro plan, if needed, we can go through it by video call.

Email

To install it, set it up or for any question.
comercial@flexigotech.com

On the web

The connector page, with the guide and the prices.
flexigotech.com/en/glop-connector-odoo.html

On Odoo Apps

The module page on the Odoo store.
apps.odoo.com

GLOP is a trademark of its owner. FlexigoTech publishes this connector on its own, with no endorsement or sponsorship from the trademark owner. FlexigoTech is a brand of Flexibles y Accesorios Gobe, S.L. (Barcelona, Spain).