

# Deliverect Integration

Connects **Deliverect** with the Newo AI agent and **Food Order** so customers can browse the restaurant menu, hear an accurate total, choose an available payment method, place an order, check its status, and request cancellation.

***Time to first success:** about 30 minutes when the Deliverect location, channel link, menu, and connector access are already available. Deliverect channel certification and production activation may take longer.*

## Before You Start

Make sure you have:

- an active Deliverect account with a location and POS connection;
- a published menu assigned to the location;
- a Newo channel link for that location;
- access to the Deliverect account and location settings;
- access to Newo Builder with permission to edit settings and click **Publish All**;
- the Deliverect **Account ID**;
- an active Deliverect Pay online gateway linked to the location if customers should pay by payment link;
- an offline gateway linked to the location if customers should pay at pickup or delivery;
- agreement with Deliverect and your finance team on whether Newo reports calculated tax as remitted.

The integration is designed for one Deliverect location per Newo project. Use a separate project for another location.

## What the AI can do

- **Use the live Deliverect menu.** Food Order receives categories, products, prices, nested modifiers, availability, fulfillment types, currency, taxes, and configured fees.
- **Quote one final total.** The same Food Order total is used in the conversation, payment request, and submitted order.
- **Create unpaid orders.** Pay-at-pickup and pay-at-delivery orders are submitted immediately when that option is available for the selected fulfillment type.
- **Take payment before order creation.** For payment-link checkout, the agent sends a secure Deliverect Pay link and creates the real order only after payment authorization.
- **Keep payment and order correlated.** Newo stores the Deliverect Pay `paymentId` with the submitted order and adds it to the Channel Order note for support diagnostics.
- **Track current order status.** Deliverect status notifications update the customer's saved order, which the agent can explain in plain language.
- **Request cancellation.** The agent uses the customer's saved order and sends a cancellation request without asking the customer for internal identifiers.
- **Respect live availability.** Product snooze updates and busy-mode updates are applied to the current location.

## Menu, prices, taxes, and fees

Deliverect remains the source of truth for the menu. Newo converts each Deliverect menu into the Food Order catalog after channel registration and whenever Deliverect publishes a menu update.

| Deliverect data                    | What Food Order receives                                                   |
|------------------------------------|----------------------------------------------------------------------------|
| Menu and fulfillment configuration | Delivery, pickup, eat-in, or curbside order types supported by the menu    |
| Product and modifier prices        | Decimal currency amounts used when building the cart                       |
| Product tax rate                   | Tax rate for the product                                                   |
| Modifier tax rate                  | The modifier's own tax rate, even when it differs from the parent product  |
| Delivery fee and its tax           | One gross delivery fee                                                     |
| Service fee and its tax            | One gross service fee                                                      |
| Bag fee                            | Package fee                                                                |
| Currency and tax mode              | Currency, decimal precision, and whether listed prices already include tax |
| Snoozed products and busy mode     | Current operational restrictions for the registered location               |

Food Order calculates and speaks the final total. Deliverect validates that the selected products and modifiers still exist and are available, then uses the submitted Food Order total as the single final amount for payment and order creation. It does not recalculate the grand total or reject an order because of a second competing total.

The integration still calculates the individual item, modifier, tax, and fee rows required by Deliverect's order format. This breakdown is used only to describe the order correctly to Deliverect; it does not replace the Food Order total.

### When the menu refreshes

| When it happens                                                   | Observable result                                                                   |
|-------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| Channel registration becomes active                               | The location, menu, and payment configuration are loaded                            |
| Deliverect publishes a menu update                                | Food Order receives the refreshed catalog                                           |
| The operator clicks <b>Publish All</b>                            | Setup is refreshed and a menu sync is requested                                     |
| A customer submits an order after cached store data becomes stale | Current store and fee data are refreshed before validation                          |
| Food Order finishes rebuilding its catalog                        | Payment choices are published again so they are not lost during catalog replacement |

If a menu update is rejected, the last valid Food Order catalog remains available and a sanitized error is stored for support.

## Payment and order lifecycle

Food Order presents only the payment choices published for the selected fulfillment type.

| Customer choice                          | What happens                                                                                                                            |
|------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| Pay at pickup                            | The order is submitted immediately as unpaid                                                                                            |
| Pay at delivery                          | The order is submitted immediately as unpaid                                                                                            |
| Payment link                             | A draft is saved, a secure link is sent, and no restaurant order is created yet                                                         |
| Payment authorized                       | The saved draft is submitted as a paid order with the Deliverect payment reference                                                      |
| Payment refused                          | The draft remains unsubmitted and no restaurant order is created                                                                        |
| Paid order submitted successfully        | The temporary payment-to-customer mapping is deleted and the customer receives confirmation                                             |
| Paid order submission definitively fails | When automatic refunds are enabled, the integration requests a full refund by the saved paymentId; it does not submit a duplicate order |

The Channel Order API does not have a native field that links a Deliverect Pay payment to the order. Newo therefore owns the authoritative `channelOrderId ↔ paymentId` association. The payment ID is also appended to the order note, but Deliverect Pay screens and the downstream POS are not guaranteed to display it as a structured payment/order relationship.

### Cancellation and refund lifecycle

| Event                                              | Result                                                                                                       |
|----------------------------------------------------|--------------------------------------------------------------------------------------------------------------|
| Customer requests cancellation                     | Newo sends the same Channel Order ID with status 100; this requests a POS void but is not final cancellation |
| POS returns status 110                             | The order is finally canceled                                                                                |
| Status 110 for a paid order with a saved paymentId | Newo requests a full Deliverect Pay refund when automatic refunds are enabled                                |
| Status 110 for pay at pickup/delivery              | No refund is sent because Newo did not collect a payment                                                     |
| POS returns status 126                             | Cancellation failed; the order remains active and no refund is sent                                          |

Refunds are initiated only when Newo has the payment ID and original payment amount and **Enable Automatic Refunds** is on. Cancellation and payment correlation is stored in Newo because Channel Order does not expose a native Deliverect Pay reference field.

### Enable Automatic Refunds

This switch is on by default and controls only whether Newo may start a new, full Deliverect Pay refund.

| Switch | Confirmed cancellation of a paid order | Definitive failure after payment | Customer notification |
|--------|----------------------------------------|----------------------------------|-----------------------|
|--------|----------------------------------------|----------------------------------|-----------------------|

|     |                                                               |                                                                                |                                                                          |
|-----|---------------------------------------------------------------|--------------------------------------------------------------------------------|--------------------------------------------------------------------------|
| On  | Newo refunds the original charged amount after POS status 110 | Newo refunds the original charged amount instead of creating a duplicate order | SMS confirms refund success or explains that the automatic refund failed |
| Off | The order is canceled, but Newo does not refund the payment   | Newo preserves the payment reference but does not refund it                    | SMS asks the customer to contact support about the payment               |

The switch does not:

- disable order cancellation;
- affect pay-at-pickup or pay-at-delivery orders, because Newo collected no payment;
- refund when the POS rejects cancellation with status 126 ;
- refund an order submission whose outcome is uncertain;
- provide partial refunds;
- refund existing payments when the switch is changed; or
- stop a refund request that Newo has already sent.

Before turning the switch off, make sure support can find the Deliverect Pay payment and issue the full refund manually. Turning it on again affects only future refund decisions; a payment previously skipped still requires manual handling. Cancellation and refund webhooks are asynchronous, so their final results are sent by SMS and never cause a new spoken or chat response after the customer has left the conversation.

The integration prevents repeated customer messages or repeated callbacks from creating duplicate payment requests or duplicate orders.

### Payment choices published to Food Order

- Delivery menus receive **pay at delivery** only when an offline gateway is linked.
- Pickup, eat-in, and curbside menus receive **pay at pickup** only when an offline gateway is linked.
- **Payment link** is added only when Deliverect returns an online payment gateway profile for the registered channel.

The fulfillment type decides whether an offline gateway is presented as pay at pickup or pay at delivery. Without an offline gateway, the integration does not offer either method.

## One registration webhook

Only **Channel Registration Webhook URL** is entered manually in Deliverect.

When Deliverect calls that URL for a location, Newo returns the location's five callback URLs:

- payment updates;
- menu updates;
- order status updates;
- product snooze and unsnooze updates;
- busy-mode updates.

Deliverect stores those callback URLs inside the location configuration. Operators should not copy the five generated callback URLs into Deliverect one by one.

The same registration event stores the location, channel link, optional external location reference, and registration state in Newo. New payments and orders are blocked unless the registration state is active.

## Features at a glance

| Feature                                                   | Included                    |
|-----------------------------------------------------------|-----------------------------|
| Sync Deliverect menu to Food Order                        | ✓                           |
| Publish product and modifier prices                       | ✓                           |
| Publish product-specific and modifier-specific tax        | ✓                           |
| Publish delivery, service, and package fees               | ✓                           |
| Support nested modifier groups                            | ✓                           |
| Use Food Order total for conversation, payment, and order | ✓                           |
| Pay at pickup                                             | ✓ , with an offline gateway |
| Pay at delivery                                           | ✓ , with an offline gateway |
| Deliverect Pay redirect link                              | ✓ , with an online gateway  |
| Create the paid order after authorization                 | ✓                           |
| Store payment/order correlation in Newo                   | ✓                           |
| Add paymentId to the paid Channel Order note              | ✓                           |
| Refund after definitive paid-order creation failure       | ✓                           |
| Refund a paid order after POS confirms status 110         | ✓                           |
| Receive order status updates                              | ✓                           |
| Request order cancellation                                | ✓                           |
| Receive snooze and busy-mode updates                      | ✓                           |
| Multiple Deliverect locations in one Newo project         | ✗                           |
| Restaurant table reservations                             | ✗                           |
| Edit the Deliverect menu from Newo                        | ✗                           |
| Percentage-based service charges                          | ✗                           |
| Taxed bag fees                                            | ✗                           |
| Deliverect discounts in the submitted breakdown           | ✗                           |

## Setup

### 1. Prepare your Deliverect account

1. In Deliverect, confirm that the correct account, location, POS connection, and Newo channel link exist.

2. Publish a menu to that location and verify that its products are available for the required order types.
3. If payment links are required, open **Deliverect Pay → Configuration**, enable an online gateway, and link it to the location.
4. If payment at pickup or delivery is required, enable an offline gateway and link it to the location.
5. Copy the Deliverect **Account ID** for the customer account.
6. Keep the location's channel-registration configuration open. You will paste one Newo URL there after the first publish.

For Deliverect-side details, see the official [Channel API guides](#) and [Deliverect Pay guide](#).

**How to verify:** The location shows an active POS connection, a Newo channel link, and at least one published menu.

## 2. Get your Deliverect credentials

The integration uses the Deliverect connector provisioned for the Newo channel. It does not ask the operator to paste a Client ID, Client Secret, access token, or refresh token into the **Module - Deliverect** settings.

If the connector is not authorized, contact the Newo integration administrator responsible for the Deliverect partner credentials before continuing.

**How to verify:** In Newo, the Deliverect connector is present and authorized; there are no Client ID, Client Secret, or Access Token fields under **Module - Deliverect**.

## 3. Configure in Newo

1. Open the project in **Newo Builder** and find **Module - Deliverect**.
2. Set **Account ID** to the Deliverect customer account that contains the location.
3. Confirm **API Base URL** matches the environment. Keep staging until Deliverect certification is complete.
4. Confirm **Channel Scope** matches the scope assigned to the Newo channel.
5. Set **Payment Return URL** to the page customers should see after leaving checkout.
6. Set **Tax Remittance Policy** according to the approved commercial and legal setup.
7. Review the feature switches:

| Setting                                       | Purpose                                                                                                                                                     | Default |
|-----------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|
| <b>Enable Food Ordering</b>                   | Allows Food Order to submit unpaid orders or start payment-link checkout                                                                                    | On      |
| <b>Enable Order Status Lookup</b>             | Allows the agent to explain the latest saved Deliverect status                                                                                              | On      |
| <b>Enable Order Cancellation</b>              | Allows the agent to send cancellation requests                                                                                                              | On      |
| <b>Enable Automatic Refunds</b>               | Starts full Deliverect Pay refunds after confirmed cancellation or definitive paid-order creation failure; turn off only when support owns refunds manually | On      |
| <b>Publish Status and Cancellation Canvas</b> | Publishes Deliverect's two existing-order scenarios                                                                                                         | On      |

8. Click **Publish All**.

**How to verify:** **Channel Registration Webhook URL** is populated after **Publish All** and **Channel Registration Status** is still **unregistered** until Deliverect calls the URL.

#### 4. Hit Publish All

After the first **Publish All**:

1. Return to **Module - Deliverect**.
2. Copy **Channel Registration Webhook URL**.
3. Paste that URL into the Newo channel-registration configuration in Deliverect.
4. Save and activate the channel for the intended location.
5. Wait for **Channel Registration Status** to become **active**.
6. Click **Publish All** once more to refresh the menu, payment choices, and scenarios from the active registration.

Do not manually copy **Payment Update Webhook URL**, **Menu Update Webhook URL**, **Order Status Webhook URL**, **Snooze / Unsnooze Webhook URL**, or **Busy Mode Webhook URL** into Deliverect. They are returned automatically by the registration webhook.

#### How to verify setup

- **Channel Registration Status** is **active**.
- All six webhook URL settings are populated.
- Food Order shows the expected menu and order types.
- Food Order shows pay at pickup or pay at delivery only when an offline gateway is linked.
- Food Order also shows payment link when an online Deliverect Pay gateway is linked.
- The status and cancellation scenarios appear on the canvas when **Publish Status and Cancellation Canvas** is enabled.

### How to test that everything works

Use a staging location and a demo customer before enabling production traffic.

#### Test menu and total

1. Ask the agent to place an order containing a product and at least one modifier.
2. Confirm the displayed product, modifier, prices, and fees match Deliverect.
3. Confirm the total spoken by the agent is the same total shown before submission.
4. Change a product price or availability in Deliverect and publish the menu.
5. Start a new cart and confirm Food Order uses the updated catalog.

#### Test pay at pickup

1. Start a pickup order and select **pay at pickup**.
2. Confirm the order appears in Deliverect immediately.
3. Confirm it is marked unpaid and the payment amount equals the Food Order total.
4. Confirm no Deliverect Pay link is sent.

#### Test payment link

1. Confirm an online gateway is active and linked to the staging location.
2. Start an eligible order and select **payment link**.
3. Confirm the customer receives a secure payment link and no restaurant order exists yet.
4. Complete the test checkout.

5. Confirm the payment becomes authorized.
6. Confirm one paid order is then created in Deliverect with the same total and a note containing the `paymentId` .
7. Send or replay the same update again and confirm no duplicate order appears.

### Test status lookup

1. Create an order through the agent.
2. Change its status in the POS or Deliverect test environment.
3. Ask, "What is the status of my order?"
4. The agent should use the saved order and report the latest known restaurant status without asking for an internal order ID.

### Test cancellation

1. Ask to cancel the order that was just created.
2. Confirm the agent sends the cancellation request without asking for an internal order ID.
3. Confirm Deliverect receives the request.
4. Confirm the first result is **cancel requested**, not **canceled**.
5. Make the POS accept the cancellation and confirm Deliverect sends status `110` .
6. For a payment-link order, confirm a full refund request is sent for the saved `paymentId` ; for an unpaid order, confirm no refund is sent.
7. Repeat with POS status `126` and confirm the agent reports cancellation failure without sending a refund.

### Test operational updates

1. Snooze a product in Deliverect and confirm it can no longer be ordered.
2. Unsnuzzle it and confirm it becomes available again.
3. Turn busy mode on and off and confirm the operational state changes only for the registered location.

## Scenarios

### What gets added on Publish All

| Intent                          | Scenario                             |
|---------------------------------|--------------------------------------|
| Food Order Status Request       | Checking Food Order Status via Agent |
| Food Order Cancellation Request | Canceling Food Order via Agent       |

An introduction scenario routes new orders to Food Order and routes existing-order requests to these two Deliverect scenarios.

These scenarios are a reference implementation. You can edit them in the Newo canvas UI. Future integration updates do not overwrite edited copies. To receive a newer shipped version, delete the edited library copy and click **Publish All** again.

### Why trigger phrases matter

Each exact phrase below starts the related Deliverect action. You may change the surrounding conversation, but do not rewrite, split, or remove the phrase.

### Scenario 11 — Checking Food Order Status via Agent

Runs when the customer asks for the status of an existing food order.

1. Uses the customer's saved current orders as the source of truth.
2. Does not ask for an internal order ID or reference.
3. Says *"Let me check your order status right now."*
4. Waits for the latest saved restaurant state and explains it in plain language.
5. If there is no usable saved order or the lookup fails, transfers to a human during working hours or relays a message outside working hours.

Triggered by intent **Food Order Status Request**.

#### Trigger phrases (do not edit)

- *"Let me check your order status right now."*

#### What the agent reads to know the result

- **Current orders panel** — saved orders associated with the current customer.
- **Restaurant order delivery state panel** — loading, result, or error state from the latest status check.

#### Safe customization (edit in the canvas UI)

-  Change how the agent introduces the status or explains a successful result.
-  Change the working-hours fallback wording.
-  Do not change the trigger phrase.
-  Do not tell the agent to ask customers for internal order identifiers.

### Scenario 12 — Canceling Food Order via Agent

Runs when the customer asks to cancel an existing or just-submitted order.

1. Selects the order from the customer's saved current orders and conversation context.
2. Confirms only when the cancellation request is ambiguous.
3. Says *"I will ask the restaurant to cancel that order right now. Please give me a moment."*
4. Sends the cancellation request and explains its direct API result while the conversation is active. The later POS result is delivered by SMS.
5. If no usable saved order exists or the request fails, transfers to a human during working hours or relays a message outside working hours.

Triggered by intent **Food Order Cancellation Request**.

#### Trigger phrases (do not edit)

- *"I will ask the restaurant to cancel that order right now. Please give me a moment."*

#### What the agent reads to know the result

- **Current orders panel** — saved orders and the private references required for safe cancellation.
- **Restaurant order state panel** — loading, result, or error state from the cancellation request.

#### Safe customization (edit in the canvas UI)

-  Change the confirmation wording when the request is ambiguous.
-  Change how the agent explains that restaurant acceptance may still be required.
-  Do not change the trigger phrase.
-  Do not cancel using only a customer-facing order number.

## Trigger phrase reference

| Scenario                             | Exact phrase                                                                         | Trigger count |
|--------------------------------------|--------------------------------------------------------------------------------------|---------------|
| Checking Food Order Status via Agent | "Let me check your order status right now."                                          | 1             |
| Canceling Food Order via Agent       | "I will ask the restaurant to cancel that order right now. Please give me a moment." | 1             |

Keep these phrases intact when editing the scenarios. They trigger the corresponding action.

## Example dialogs

### Example 1 — Pay at pickup

**Customer:** I'll pick it up and pay there.

**Agent:** Your order has been submitted for pickup. Your total is \$8.72.

### Example 2 — Payment link

**Customer:** Send me a payment link.

**Agent:** Payment is required before your order can be placed. I've sent you a secure payment link.

**Customer:** I paid.

**Agent:** Payment received. Your order was submitted.

### Example 3 — Status

**Customer:** What is the status of my order?

**Agent:** Let me check your order status right now.

**Agent:** The restaurant has accepted your order and is preparing it now.

### Example 4 — Cancellation

**Customer:** Cancel the order I just placed.

**Agent:** I will ask the restaurant to cancel that order right now. Please give me a moment.

**Agent:** The cancellation request was sent. Final cancellation may depend on the restaurant accepting it.

**SMS:** Your order was canceled.

## Customizing the agent

- Turn **Enable Food Ordering**, **Enable Order Status Lookup**, **Enable Order Cancellation**, or **Enable Automatic Refunds** off to disable that capability.
- Turn **Publish Status and Cancellation Canvas** off if your project maintains custom existing-order scenarios.
- Edit the two Deliverect scenarios on the canvas, but keep their exact trigger phrases unchanged.

- Change **Payment Return URL** to your branded checkout-return page.
- Change **Tax Remittance Policy** only after confirming the commercial and legal requirement.
- Reduce **Store State Cache Window** only when support needs more frequent store and fee refreshes; shorter windows increase external requests.

Always preserve *"Let me check your order status right now."* and *"I will ask the restaurant to cancel that order right now. Please give me a moment."*

Integration updates do not overwrite your edited canvas copies. To pull newer defaults, delete the edited Deliverect library items and click **Publish All** again.

## FAQ

### Does Newo calculate a second order total?

No. Food Order supplies the final total used in the conversation, payment request, and submitted order. Deliverect calculates only the component rows required to describe the order.

### Why are delivery and service fees published with tax included?

Food Order currently accepts one value for each fee. The integration therefore publishes the fee plus its Deliverect fee tax as one gross amount.

### Can a modifier have a different tax rate from its product?

Yes. The integration publishes and submits the modifier's own Deliverect tax rate.

### Why do I see pay at pickup when Deliverect Pay shows only Stripe?

You should not. Stripe is an online gateway, so Food Order should offer only payment link. Pay at pickup requires a linked offline gateway. Click **Publish All** after changing gateway configuration.

### When is a payment-link order created?

Only after Deliverect reports the payment as authorized. Before authorization, Newo stores a draft and sends the secure link.

### What happens when connector credentials rotate?

The credentials are managed on the Deliverect connector, not in **Module - Deliverect**. Ask the integration administrator to reauthorize or update the connector, then click **Publish All** and run a staging menu and payment test.

### Can one Newo project serve multiple Deliverect locations?

No. Create a separate Newo project for each registered Deliverect location.

### Does a successful cancellation request guarantee cancellation?

No. The initial Channel Order response only confirms that status `100` was sent. Cancellation is final only after the POS returns status `110`. Status `126` means the POS could not cancel the order. The direct cancellation API result is spoken during the active conversation; the later `110` or `126` webhook result and any refund result are sent by SMS and do not trigger another agent reply.

### How are a Deliverect Pay payment and Channel Order linked?

Newo stores both `paymentId` and `channelOrderId` in the saved order record and adds the payment ID to the order note. Channel Order has no structured `paymentId` field, so Deliverect Pay screens or the POS may not show a native link between the two records.

Deliverect represents a cancellation as a second Channel Order with status `100` and replaces the displayed note with a cancellation message. Newo keeps the original note in its saved order and includes it in `cancellationReason`, so the customer comment and diagnostic `paymentId` remain visible. Legacy paid orders without a saved note fall back to the stored `paymentId`.

Incoming callbacks use compact AgentPersona mappings instead of creating an actor for every order. `payment_{paymentId}` routes authorization while payment is pending, `order_{channelOrderId}` routes normal status updates, and `cancellation_{channelOrderId}` routes the final cancellation/refund lifecycle. Temporary payment and cancellation mappings are removed when their async work finishes.

### When is a canceled order refunded?

Only after POS status `110`, and only when the saved order contains a Deliverect Pay `paymentId`. Pay-at-pickup and pay-at-delivery orders have no Newo-collected payment to refund. **Enable Automatic Refunds** must also be on. When it is off, Newo sends no refund request and directs the customer to support by SMS. Status `126` does not trigger a refund.

### What happens if I turn off automatic refunds?

New payment and order creation continue normally, and customers can still ask to cancel an order. Newo simply stops starting new automatic refunds. For a paid order that is later canceled or definitively fails to reach the restaurant, support must locate the Deliverect Pay payment and issue the full refund manually. Changing the switch does not refund older payments and does not stop a refund already in progress.

## Common errors and recovery

### First Publish All does not load Deliverect data

**Likely cause:** The Deliverect connector is not authorized, or **Account ID**, **API Base URL**, or **Channel Scope** is incorrect.

**How to recover:** Ask the integration administrator to verify the connector. Correct the three settings, then click **Publish All** again.

**How to verify:** **Channel Registration Webhook URL** is populated and the menu sync no longer reports an authorization error.

### Channel Registration Status stays unregistered or inactive

**Likely cause:** The registration URL was not saved in the intended Deliverect channel, Deliverect did not call it, or the channel link is inactive.

**How to recover:** Copy **Channel Registration Webhook URL** after **Publish All**, paste it into the correct Deliverect channel-registration setting, save, and activate the channel. Do not use one of the five callback URLs.

**How to verify:** **Channel Registration Status** becomes **active** and all five callback URL settings are populated.

### The menu is empty or stale

**Likely cause:** The wrong account or location is registered, the menu is not assigned to the location, or the last menu publication failed validation.

**How to recover:** Confirm **Account ID**, the registered location, channel link, and menu assignment. Publish the menu in Deliverect, then click **Publish All** in Newo.

**How to verify:** Food Order shows the current products, prices, modifiers, and order types. **Last Menu Sync Error** is empty.

### **Payment link is missing or the request fails**

**Likely cause:** No online gateway is linked to the registered location, the cached payment profile is stale, or the submitted customer and total data is incomplete.

**How to recover:** Link an active online gateway in **Deliverect Pay → Configuration**, click **Publish All**, and start a new staging cart with a customer name and a total greater than zero.

**How to verify:** Food Order offers **payment link**, checkout opens, and **Last Payment Methods Sync Error** is empty.

### **Payment is authorized but no order appears**

**Likely cause:** The paid order could not be submitted, the channel became inactive, or the result was uncertain.

**How to recover:** Do not ask the customer to pay again. A definitive order-creation failure triggers an automatic full refund when **Enable Automatic Refunds** is on. When it is off, support must handle the payment manually. If the submission result is uncertain, the integration preserves the payment and order state for support instead of refunding or resubmitting blindly.

**How to verify:** For a definitive failure, confirm the saved order attempt reaches `refunded`. For an uncertain result, support confirms whether Deliverect received the order before any manual replacement order is created.

### **Customer receives an SMS saying automatic refunds are disabled**

**Likely cause:** **Enable Automatic Refunds** was off when Newo reached a refund decision.

**How to recover:** Do not ask the customer to pay again. Find the saved Deliverect Pay payment and issue the full refund manually. Turn **Enable Automatic Refunds** on if future eligible payments should be refunded automatically.

**How to verify:** Confirm the manual refund in Deliverect Pay. Re-enabling the switch does not retroactively refund the payment that was skipped.

### **The scenario says the trigger phrase but no action starts**

**Likely cause:** The related feature switch is off, the canvas items were not published, or the exact phrase was changed.

**How to recover:** Turn on **Enable Order Status Lookup** or **Enable Order Cancellation**, restore the exact phrase from this README, and click **Publish All**.

**How to verify:** A staging conversation produces a loading state followed by a result or a clear error instead of continuing without action.

## Known limitations

- One Newo project supports one registered Deliverect location.
- New orders are handled by Food Order; Deliverect publishes only the status and cancellation canvas scenarios.
- Payment links require an online Deliverect Pay gateway profile.
- Pay-at-pickup and pay-at-delivery require an offline gateway profile; payment links require an online gateway profile.
- The integration accepts only authorized or refused payment updates for the payment-link lifecycle.
- A definitive paid-order creation failure triggers a full refund only when automatic refunds are enabled; an uncertain transport result is not silently retried or refunded.
- Percentage-based service charges, taxed bag fees, and discount rows are not represented in the current order breakdown.
- Cancellation status `100` is only a request. Status `110` confirms cancellation; status `126` means cancellation failed.
- Native Deliverect Pay-to-Channel-Order linking is unavailable in the Channel API. Newo stores the association, and the order note contains the payment ID for diagnostics.
- Commerce Basket/Checkout is intentionally out of scope for this Channel API integration.
- Status lookup and cancellation work only for orders saved to the same customer persona.
- The integration does not create or edit Deliverect menus, gateways, locations, or table reservations.

## Support handoff notes

When escalating, include:

- environment: staging or production;
- Deliverect account and location names, without pasting secrets;
- approximate time and timezone;
- customer session identifier;
- whether **Channel Registration Status** was active;
- whether the issue concerns menu, payment, order creation, status, cancellation, snooze, or busy mode;
- payment or order reference when available;
- the sanitized value of **Last Menu Sync Error** or **Last Payment Methods Sync Error**.

Never send connector credentials, access tokens, full webhook URLs, card data, or an unredacted screenshot.

## All settings reference

The **Name** column below is intended for administrators and support. Operators should use the UI labels shown in the descriptions.

| Name                                 | Description                                                                                | Required | Default                             |
|--------------------------------------|--------------------------------------------------------------------------------------------|----------|-------------------------------------|
| <code>deliverect_api_base_url</code> | <b>API Base URL</b><br>— Deliverect environment used for menu, payment, and order requests | Yes      | <code>https://api.staging.de</code> |

|                                |                                                                                              |           |              |
|--------------------------------|----------------------------------------------------------------------------------------------|-----------|--------------|
| deliverect_channel_name        | <b>Channel Scope</b> — lowercase channel scope assigned to Newo                              | Yes       | newoai       |
| deliverect_account_id          | <b>Account ID</b> — customer account containing the registered location                      | Yes       | Empty        |
| deliverect_channel_link_id     | <b>Registered Channel Link ID</b> — assigned during registration                             | Automatic | Empty        |
| deliverect_location_id         | <b>Registered Location ID</b> — assigned during registration                                 | Automatic | Empty        |
| deliverect_channel_location_id | <b>External Channel Location ID</b> — optional external location reference from registration | Automatic | Empty        |
| deliverect_registration_status | <b>Channel Registration Status</b> — unregistered, register, active, or inactive             | Automatic | unregistered |
| deliverect_enable_ordering     | <b>Enable Food Ordering</b> — allows payment and order submission                            | No        | True         |
| deliverect_enable_order_status | <b>Enable Order Status Lookup</b> — enables the                                              | No        | True         |

|                                |                                                                                                                                                                               |                   |                                               |
|--------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------|-----------------------------------------------|
|                                | status tool and scenario                                                                                                                                                      |                   |                                               |
| deliverect_enable_cancellation | <b>Enable Order Cancellation</b> — enables cancellation requests                                                                                                              | No                | True                                          |
| deliverect_enable_refunds      | <b>Enable Automatic Refunds</b> — permits new full Deliverect Pay refunds after paid order failures or POS-confirmed cancellation; disabling requires manual support handling | No                | True                                          |
| deliverect_payment_return_url  | <b>Payment Return URL</b> — page shown after checkout                                                                                                                         | For payment links | <a href="https://newo.ai">https://newo.ai</a> |
| deliverect_tax_remitted_policy | <b>Tax Remittance Policy</b> — reports no calculated tax or all calculated item tax as remitted                                                                               | Yes               | none                                          |
| deliverect_setup_scenarios     | <b>Publish Status and Cancellation Canvas</b> — publishes Deliverect-owned existing-order scenarios                                                                           | No                | True                                          |
| deliverect_store_state         | <b>Store State Cache</b> — current currency, tax                                                                                                                              | Automatic         | {}                                            |

|                                           |                                                                                              |             |       |
|-------------------------------------------|----------------------------------------------------------------------------------------------|-------------|-------|
|                                           | mode, timezone, fees, status, and fulfillment settings                                       |             |       |
| deliverect_operational_state              | <b>Webhook Operational State</b> — current busy mode and snoozed products                    | Automatic   | {}    |
| deliverect_store_state_updated_at         | <b>Store State Updated At</b> — timestamp of the last store refresh                          | Automatic   | 0     |
| deliverect_store_state_range_seconds      | <b>Store State Cache Window</b> — store and fee freshness window in seconds                  | No          | 60    |
| deliverect_canonical_menu_cache           | <b>Menu Runtime Index</b> — compact catalog used for validation and order construction       | Automatic   | {}    |
| deliverect_canonical_menu_catalog_version | <b>Canonical Menu Version</b> — revision used to keep checkout aligned with the current menu | Automatic   | Empty |
| deliverect_payment_gateway_profiles       | <b>Payment Gateway Profiles</b> — cached online and offline payment profiles for the         | For payment | []    |

|                                             |                                                                                                |           |       |
|---------------------------------------------|------------------------------------------------------------------------------------------------|-----------|-------|
|                                             | registered channel                                                                             |           |       |
| deliverect_last_menu_sync_error             | <b>Last Menu Sync Error</b> — sanitized menu synchronization diagnostic                        | No        | Empty |
| deliverect_last_payment_methods_sync_error  | <b>Last Payment Methods Sync Error</b> — sanitized payment-method synchronization diagnostic   | No        | Empty |
| deliverect_channel_registration_webhook_url | <b>Channel Registration Webhook URL</b> — the only URL manually configured in Deliverect       | Yes       | Empty |
| deliverect_payment_update_webhook_url       | <b>Payment Update Webhook URL</b> — generated payment callback returned during registration    | Automatic | Empty |
| deliverect_menu_update_webhook_url          | <b>Menu Update Webhook URL</b> — generated menu callback returned during registration          | Automatic | Empty |
| deliverect_order_status_webhook_url         | <b>Order Status Webhook URL</b> — generated order-status callback returned during registration | Automatic | Empty |
| deliverect_snooze_webhook_url               | <b>Snooze / Unsnnooze Webhook URL</b>                                                          | Automatic | Empty |

|                                               |                                                                                             |           |       |
|-----------------------------------------------|---------------------------------------------------------------------------------------------|-----------|-------|
|                                               | — generated availability callback returned during registration                              |           |       |
| <code>deliverect_busy_mode_webhook_url</code> | <b>Busy Mode Webhook URL</b><br>— generated busy-mode callback returned during registration | Automatic | Empty |

## Changelog

### Unreleased

- Add an operator-controlled automatic refund switch, enabled by default.
- Send asynchronous cancellation and refund outcomes by SMS without reopening the agent conversation.
- Document automatic-refund eligibility, manual handling, and non-retroactive behavior.

### 2.0.6

- Publish pay at pickup and pay at delivery only when an offline gateway is linked to the registered location.
- Publish payment link only when an online gateway is linked.
- Disable all payment choices when gateway synchronization fails instead of assuming cash is available.

### 2.0.5

- Republish Food Order payment choices after the Food Order catalog finishes rebuilding, preventing the menu refresh from clearing the payment map.
- Keep payment-link availability tied to the online Deliverect Pay gateway profile.

### 2.0.4

- Accept the current Food Order total formats and use one total for speech, payment, and submitted order.
- Publish each modifier's own Deliverect tax rate.
- Publish fulfillment-specific payment choices to Food Order.
- Publish delivery and service fees including their taxes, plus the package fee.

### 2.0.3

- Replace manual callback configuration with one channel-registration webhook that returns all location callback URLs.
- Create pay-at-fulfillment orders immediately.
- Create payment-link orders only after payment authorization and attach the payment reference.
- Remove temporary payment mappings after successful paid-order creation.
- Add order status, cancellation, snooze, and busy-mode handling for the registered location.