Partner API

Quote, book and track Nexroo deliveries across Sydney from your own systems.

Version	1.2.0
Date	6 November 2026
Base URL	https://nexroo.com.au/api/v1
Reference	https://nexroo.com.au/api/v1/openapi.json (OpenAPI 3.1)
Support	support@nexroo.com.au (quote the requestId from any error)

Nexroo delivers parcels across Sydney. This API lets a partner network quote, book and track Nexroo deliveries from its own systems. It is provided by Panagia Group Pty Ltd (ABN 63 686 497 405), trading as Nexroo.

Available now: account check, service levels, depots, quotes, bookings, cancellation, labels, tracking, proof of delivery, webhooks (ours or your own), recent events and a sandbox for test keys. No code? Nexroo is also on Zapier and Make.

Panagia Group Pty Ltd, ABN 63 686 497 405, trading as Nexroo, Sydney NSW. Terms of carriage: nexroo.com.au/terms. Privacy policy: nexroo.com.au/privacy.

Contents

1. Getting access
2. Terms and cover
3. Requests
4. Errors
5. Service levels
6. Depots and drop-offs
7. Quotes
8. Bookings
9. Tracking
10. Webhooks
11. Sandbox
12. Going live
13. Changes

Getting access

Your business needs an approved Nexroo business account (apply at nexroo.com.au/business). Nexroo then switches API access on. From then on the account owner makes keys in the nexbiz dashboard (nexroo.com.au/business/dashboard → Connect apps), or asks us to:

- **Live keys** (`nxk_live_…`) quote and book real deliveries, billed to your account on your usual invoice.
- **Test keys** (`nxk_test_…`) give real prices, but their bookings never become deliveries.

A key is shown once when it is made. Keep it on your server, never in a browser or app. If a key leaks, ask us to revoke it: requests with it stop within a minute.

Key access. Each key is made with one of two:

- **Full:** quotes, bookings, tracking and proof of delivery.
- **No proof of delivery:** the same, without the delivery photo, signature or recipient's name, and it cannot change the webhook secret. Use it for systems that only need to book and track.

Rotating keys. An account can have 10 keys at once. To rotate one, make a new key in the dashboard, switch your system over to it, then revoke the old one there. `GET /v1/account` shows the key in use, its access and when it was made. Keys made in the dashboard have full access; ask us for one without proof of delivery.

Terms and cover

- **Carriage:** consignments booked through the API are carried on our terms, nexroo.com.au/terms, and invoiced under your business account.
- **Cover:** standard cover is **$450 per consignment** (the lesser of the declared value and $450). Declaring `goodsValueCents` sets the signature rule, not cover. Extra cover is only by arrangement in writing before pickup.
- **Times:** collection windows and delivery times are targets, not guarantees. If we miss the window through our fault, the delivery fee for that consignment is refunded.
- **Recipients' details:** handled under our privacy policy, nexroo.com.au/privacy. Tell your recipients that their name, address and phone go to Nexroo for the delivery. What we give back is set out under Proof of delivery.

Requests

Send the key as a bearer token, and JSON with its content type:

```
curl https://nexroo.com.au/api/v1/account \
  -H "Authorization: Bearer nxk_live_…"
```

- Bodies are JSON (`Content-Type: application/json`), at most 256 KB.
- Money is in Australian **cents, GST inclusive**, with the GST shown beside it.
- Times are ISO 8601 with the Sydney offset, e.g. `2026-10-22T21:00:00+11:00`; dates are `YYYY-MM-DD` in Sydney.
- Every response carries an `X-Request-Id` header.
- **Rate limits:**
  - Each key may make about **120 requests a minute**.
  - Every answer carries `RateLimit-Limit`, `RateLimit-Remaining` and `RateLimit-Reset` (seconds). Slow down when `RateLimit-Remaining` runs low.
  - Over the limit the API answers `429` with a `Retry-After` header in seconds.
  - Twenty requests a minute with a bad key from one address are also answered `429`.

Errors

Errors always look like this:

```
{ "error": { "code": "validation_failed", "message": "Give a size, or the weight and dimensions.",
   "field": "parcels[0].size", "requestId": "req_PeWeQHtm6Bxx" } }
```

Status	Codes	Meaning
400	`invalid_json`, `invalid_request`	The body could not be read. Malformed JSON may come back as a plain 400.
401	`unauthorized`	No key, an unknown key, or a revoked key.
403	`forbidden`	The account is not active, or its API access is off.
404	`not_found`	No such endpoint or record.
405	`method_not_allowed`	Wrong method; the `Allow` header says which.
409	`conflict`, `request_in_progress`, `cancellation_closed`, `barcode_in_use`	The request clashes with the current state.
413, 415	`payload_too_large`, `unsupported_media_type`	Body over 256 KB, or not JSON.
422	`validation_failed`, `not_serviceable`, `address_not_found`, `date_unavailable`, `quote_expired`, `idempotency_key_reused`	The request is readable but cannot be done. `field` names the culprit.
429	`rate_limited`	Slow down; see `Retry-After`.
500, 503	`internal`, `unavailable`	Our fault. Retry later, and tell us the `requestId` if it persists.

Service levels

`GET /v1/services` lists the services your account can book.

Service	Book by	Collected	Delivered
`nextday`	6:30 pm on the collection day	3–5 pm on the collection day	6–9 pm the next business day
`sameday`	12 pm	the same day	by 6 pm the same day
`express`	3 pm	when ready	within 3 hours of being ready

All three run Monday to Friday. Public holidays are not yet excluded automatically.

These times are **targets, not guarantees**. Traffic, weather and access can change them. If we miss the window through our fault, the delivery fee for that consignment is refunded.

The collection day can be today until the cut-off, or any open day up to 30 days ahead.

Each service in the response gives its days, cut-off, windows and delivery target as data, e.g. `"promise": { "type": "next_business_day", "by": "21:00" }`. The field is called `promise`, but it is a target like the rest.

Depots and drop-offs

`GET /v1/depots` lists the Nexroo depots you can hand parcels over at. A consignment can start either way:

- **Pickup:** a Nexroo driver collects from your address.
- **Drop-off:** you bring the parcels to a depot.

Drop-offs follow the Next Day rules. The parcels must be at the depot by 6:30 pm, and they are delivered 6–9 pm the next business day. There is no pickup fee on any consignment.

Quotes

`POST /v1/quotes` prices a consignment at your account's rates.

```
curl https://nexroo.com.au/api/v1/quotes \
  -H "Authorization: Bearer nxk_live_…" \
  -H "Content-Type: application/json" \
  -d '{
    "pickup": { "postcode": "2150" },
    "delivery": { "postcode": "2000" },
    "parcels": [{ "quantity": 2, "weightKg": 3, "lengthCm": 40, "widthCm": 30, "heightCm": 20 }]
  }'
```

- **Pickup** is a postcode, or `{ "depotId": "…" }` for a drop-off.
- **Delivery** is the delivery postcode.
- **Service and date** are optional. Without them you get every service on its first open date.
- **Parcels:**
  - A line of identical parcels takes a `size`, `S`, `M`, `L` or `XL`.
  - Or it takes the weight and dimensions, and the size is the first one both fit. There is no cubic weight. The sizes are S to 3 kg (40 × 30 × 8 cm), M to 10 kg (45 × 35 × 25), L to 20 kg (60 × 45 × 40) and XL to 25 kg (80 × 60 × 50).
  - A parcel is at most 25 kg and 80 × 60 × 50 cm.
  - A consignment is at most 30 parcels.
- **Price:** set by the delivery postcode's zone and each parcel's size, then multiplied for faster services.
- **Handover (optional):** `signatureRequired`, `goodsValueCents` and `containsAlcohol`, as on a booking. A signature adds your account's signature fee to the price.

The answer has one option per service, available with its price and dates, or with the reason it is not:

```
{
  "quoteId": "q_CPTUSegsBTjZlY31iBmK",
  "expiresAt": "2026-10-20T19:24:05+11:00",
  "currency": "AUD",
  "options": [
    {
      "service": "nextday", "label": "Next Day", "available": true,
      "price": {
        "currency": "AUD", "incGstCents": 3520, "gstCents": 320, "exGstCents": 3200, "accountRate": false,
        "breakdown": {
          "zone": "zone2", "zoneLabel": "Zone 2",
          "parcels": [{ "size": "M", "quantity": 2, "unitIncGstCents": 1760, "chargeableKg": 3 }],
```

```
          "parcelsIncGstCents": 3520, "serviceMultiplier": 1, "pickupFeeIncGstCents": 0,
             "signatureFeeIncGstCents": 0
         }
       },
       "collectionDate": "2026-10-21", "collectionWindow": { "from": "15:00", "to": "17:00" },
       "deliverBy": "2026-10-22T21:00:00+11:00", "bookBy": "2026-10-21T18:30:00+11:00"
     },
     { "service": "sameday", "label": "Same Day", "available": false, "reason": "…" }
   ]
 }
```


A quote is held for 15 minutes. A postcode outside our area is refused with `422 not_serviceable`.

Bookings

`POST /v1/consignments` books a consignment on your account. It is priced at your rates and invoiced with your other orders.

```
curl https://nexroo.com.au/api/v1/consignments \
  -H "Authorization: Bearer nxk_live_…" \
  -H "Content-Type: application/json" \
  -H "Idempotency-Key: 6f1c2a8e-4b1d-4b7e-9a0c-2f4b6c1d9e11" \
  -d '{
    "service": "nextday",
    "reference": "PO-1001",
    "channel": "shopify",
    "pickup": {
      "type": "address",
      "address": { "line1": "10 Church St", "suburb": "Parramatta", "state": "NSW", "postcode": "2150" },
      "contact": { "name": "Warehouse", "phone": "02 9876 5432" },
      "instructions": "Dock 3"
    },
    "delivery": {
      "address": { "line1": "1 George St", "line2": "Level 5", "suburb": "Sydney", "state": "NSW",
         "postcode": "2000" },
      "contact": { "name": "Jane Citizen", "phone": "0412 345 678", "email": "jane@example.com" },
      "instructions": "Reception"
    },
    "parcels": [{ "quantity": 2, "weightKg": 3, "lengthCm": 40, "widthCm": 30, "heightCm": 20, "barcodes":
       ["PB0001", "PB0002"] }]
  }'
```

Where it starts:

- **Pickup:** `"pickup": { "type": "address", … }` with an address and a contact. The phone is required. `readyAt` (HH:mm) says when the parcels are ready; Express is due 3 hours after it.
- **Drop-off:** `"pickup": { "type": "depot", "depotId": "…" }`, a depot from `GET /v1/depots`. Drop-offs are Next Day only.

Other fields:

- **`reference`:** your own reference, up to 60 characters.
- **`barcodes`:** optional, one per parcel on the line. A parcel with your barcode can be scanned by it at our depot and in our vans; a barcode already on an open consignment is refused with `409 barcode_in_use`.

- `signatureRequired`: hand it only to someone who signs. Without it, the driver may leave the parcel in a safe place and photograph it.
- `goodsValueCents`: optional, the value of the goods in cents. Left out, it is taken as $99 or less. Over 9900 ($99), a signature is required. It does not change cover: see Terms and cover.
- `containsAlcohol`: `true` when the consignment contains alcohol. It is handed only to someone 18 or over who shows photo ID and signs, and never left. With no valid ID the attempt fails (`failed_attempt`, "No valid ID or under 18") and the parcel goes back to the depot.
- **Signature fee:** a consignment that needs a signature, asked for or required, is charged your account's signature fee once (`price.breakdown.signatureFeeIncGstCents`). A quote without a signature still books: the fee is added at booking.
- `quoteId`: books at a quote's price within its 15 minutes. It must match the booking: service, postcodes, depot and parcels. Leave it out to book at the current price.
- `date`: the collection (or drop-off) date. Omitted, the first open date is used.
- **Recipient email:** if your account has Nexroo's recipient emails switched on, a delivery contact's `email` gets our collected, out-for-delivery and delivered emails. They never show a price.

Addresses are checked on our map. One we cannot place is refused with `422 address_not_found` naming the field; check the street, suburb and postcode.

Retries are safe. Every booking needs an `Idempotency-Key` header, unique per consignment (a UUID works):

- **Same key, same body:** the same consignment comes back, with `Idempotent-Replayed: true`.
- **Same key, different body:** refused with `422 idempotency_key_reused`.
- **While the first request is still running:** `409 request_in_progress`; retry in a few seconds.
- **After a refusal (a 4xx):** the key is free again for the corrected request.
- **Lifetime:** a key is remembered for 48 hours.

The answer, `201` with a `Location` header:


```
{
  "consignmentId": "NX-AB4EKM", "reference": "PO-1001", "mode": "live",
  "status": "booked", "statusLabel": "Booking confirmed", "service": "nextday",
  "start": "pickup", "depotId": null, "collectionDate": "2026-10-21", "deliverBy": "2026-10-
     22T21:00:00+11:00",
  "pickup": { "suburb": "Parramatta", "postcode": "2150" },
  "delivery": { "name": "Jane Citizen", "suburb": "Sydney", "postcode": "2000" },
  "parcels": [
    { "piece": 1, "number": "0277951027014", "barcode": "PB0001" },
    { "piece": 2, "number": "0277951027022", "barcode": "PB0002" }
  ],
  "signatureRequired": false, "goodsValueCents": null, "containsAlcohol": false, "idCheckRequired": false,
  "price": { "currency": "AUD", "incGstCents": 3520, "gstCents": 320, "exGstCents": 3200, "accountRate":
     false, "breakdown": { … } },
  "trackingUrl": "https://nexroo.com.au/track/JBULfYAbU3u_58cVHELC7g",
  "labelsUrl": "https://nexroo.com.au/api/v1/consignments/NX-AB4EKM/labels",
  "createdAt": "2026-10-20T19:21:23+11:00"
}
```


`trackingUrl` is the public tracking page you can give the recipient. Each parcel's `number` is the 13-digit number on our label.

Where the order came from. `channel` (optional, up to 40 lowercase letters, digits, `-` or `_`) says which system booked it: `shopify`, `woocommerce`, `ebay`, `zapier`, `make`, or your own token. Our staff see it on the consignment ("Shopify via API"), and it comes back on the consignment as `channel`. It changes nothing else.

Your consignments

- `GET /v1/consignments/{consignmentId}`: one consignment as it stands.
- `GET /v1/consignments`: your API consignments, newest first, 50 a page. Pass `nextCursor` back as `cursor` for the next page; `?reference=PO-1001` finds one by your reference.
- `GET /v1/consignments/{consignmentId}/labels`: A6 labels as a PDF, one page per parcel. Print them if your parcels do not carry your own barcodes.

You only ever see your own account's consignments; anyone else's reference answers `404`.

Cancelling

`POST /v1/consignments/{consignmentId}/cancel`, optionally with `{ "reason": "…" }`.

When	Result
Before 6:30 pm the evening before collection (or the drop-off's delivery day)	Cancelled, free
Booked after that cut-off, and no driver assigned yet	Cancelled, free
Routes planned or a driver on the way	Refused with `409 cancellation_closed`. Email support@nexroo.com.au to cancel: a late-cancellation fee of **$16.50 incl. GST** ($15 + GST) applies
The driver is at the pickup address	Charged in full
Already cancelled	Answers `200`

Drop-offs at a depot

For a drop-off, the parcels must reach the depot by 6:30 pm on the `collectionDate`. They are delivered 6–9 pm the next business day. Our staff receive them by scanning our label, your barcode, or the consignment id.

Tracking

`GET /v1/tracking/{code}` finds one of your consignments by:

- our id (`NX-AB4EKM`, or `NX-AB4EKM/2` from a label),
- the 13-digit parcel number on our label (spaces allowed),
- or your own barcode, URL-encoded.

```
{
  "consignmentId": "NX-6USSAF", "reference": "PO-1010", "mode": "live",
  "status": "delivered", "statusLabel": "Delivered", "service": "nextday", "deliverBy": "2026-10-
     22T21:00:00+11:00",
  "origin": { "suburb": "Parramatta", "postcode": "2150" }, "destination": { "suburb": "Sydney",
     "postcode": "2000" },
  "parcels": [{ "piece": 1, "number": "…", "barcode": "PB0601" }],
  "milestones": [
    { "code": "booked", "at": "…", "description": "Booked" },
    { "code": "picked_up", "at": "…", "description": "Collected from Parramatta" },
    { "code": "out_for_delivery", "at": "…", "description": "Out for delivery to Sydney" },
    { "code": "delivered", "at": "…", "description": "Left in a safe place: Front door" },
    { "code": "delivery_photo", "at": "…", "description": "Photo of the delivery" }
  ],
  "proofOfDelivery": { "deliveredAt": "…", "leftSafe": true, "location": "Front door", "recipientName":
     null, "photo": true, "signature": false, "idCheck": null },
  "trackingUrl": "https://nexroo.com.au/track/…"
}
```


Milestone	When
`booked`	The consignment is booked
`picked_up`	A driver has collected it (pickups)
`received_at_depot`	It is at our depot: a drop-off received, or a pickup passing through
`out_for_delivery`	It is in the van for delivery
`delivered`	Delivered: left in a safe place, or signed for
`delivery_photo`	The delivery photo has been uploaded
`failed_attempt`	A delivery attempt failed (the description says why)
`returned_to_depot`	Back at the depot after a failed attempt
`returned_to_sender`	Returned to the sender
`cancelled`	Cancelled

Proof of delivery:

- **Who took it:** `recipientName` is the first name the driver recorded, never the full name.
- **Alcohol:** `proofOfDelivery.idCheck` is the type of photo ID the driver checked, e.g. `{ "idType": "drivers_licence" }`. The types are `drivers_licence`, `photo_card`, `proof_of_age` and `passport`. Its number is never given.
- **Photo:** `GET /v1/consignments/{consignmentId}/pod/photo` returns the delivery photo.
- **Signature:** `GET /v1/consignments/{consignmentId}/pod/signature` returns the signature.

Both return the image itself, behind your key. They answer `404` until the driver's phone has uploaded it, usually within minutes of delivery. `proofOfDelivery.photo` and `.signature` say when they are there.

Kept 12 months. Photos and signatures are deleted 12 months after delivery. After that, `photo` and `signature` are `false` and the downloads answer `404`. Download what you must keep longer. The consignment and its milestones stay, with the booking record.

A key made without proof-of-delivery access gets `recipientName: null`, `photo` and `signature` `false`, and `403 forbidden` from the downloads.

Webhooks

Nexroo can post each milestone to your system as it happens. There are two ways to get one:

- **Ask us.** Give us an HTTPS URL, one for live and optionally one for test. We set it up and give you its signing secret (`whsec_…`) once. You can choose all milestones or only some.
- **Make your own** with `POST /v1/webhooks` (see Your own webhooks): up to 10 per mode, each with its own secret. This is what Zapier and Make do for you.

Every webhook on the account hears every key's consignments of its mode.

Each event is a `POST` of JSON:

```
{
  "id": "evt_gMuTPm291qAeT3TSK0B__3_delivered",
  "type": "consignment.delivered",
  "createdAt": "2026-10-20T19:37:02+11:00",
  "mode": "live",
  "data": {
    "consignment": { "consignmentId": "NX-6USSAF", "reference": "PO-1010", "status": "delivered",
       "statusLabel": "Delivered", "barcodes": ["PB0601", "PB0602"] },
    "milestone": { "code": "delivered", "at": "2026-10-20T19:37:00+11:00", "description": "Left in a safe
       place: Front door" }
  }
}
```

Headers:

- `Nexroo-Signature: t=<unix seconds>,v1=<signature>`
- `Nexroo-Event-Id`
- `Nexroo-Event-Type`
- `Nexroo-Delivery-Attempt`

A test from us has the type `ping`.

Check the signature before trusting an event:

1. Compute HMAC-SHA256 of `"<t>.<raw body>"` with your secret.
2. Compare it with each `v1` in the header, in constant time. Accept the event if **any** matches: for 24 hours after the secret is changed there are two, one per secret.
3. Reject events whose `t` is more than 5 minutes from your clock, so a recorded request cannot be replayed. Each retry is signed afresh.

```
import crypto from 'node:crypto'

function verifyNexroo(rawBody, header, secret) {
  const parts = (header ?? '').split(',')
  const t = Number(parts.find((p) => p.startsWith('t='))?.slice(2))
  if (!t || Math.abs(Date.now() / 1000 - t) > 300) return false
  const expected = crypto.createHmac('sha256', secret).update(`${t}.${rawBody}`).digest()
  return parts.filter((p) => p.startsWith('v1=')).some((p) => {
    const given = Buffer.from(p.slice(3), 'hex')
    return given.length === expected.length && crypto.timingSafeEqual(expected, given)
  })
}
```

```
import hashlib, hmac, time

def verify_nexroo(raw_body: bytes, header: str, secret: str) -> bool:
    parts = [p.split("=", 1) for p in (header or "").split(",") if "=" in p]
    t = next((v for k, v in parts if k == "t"), "")
    if not t.isdigit() or abs(time.time() - int(t)) > 300:
        return False
    expected = hmac.new(secret.encode(), f"{t}.".encode() + raw_body, hashlib.sha256).hexdigest()
    return any(hmac.compare_digest(expected, v) for k, v in parts if k == "v1")
```

Answer quickly, with any `2xx`, within 10 seconds. Do slow work after you answer.

Retries:

- **Other answers or timeouts:** retried after 1, 5 and 15 minutes, then 1, 3, 6 and 12 hours.
- **A 4xx other than 408 or 429:** given two more tries only.
- **410 Gone:** switches the webhook off.
- **Failing for three days:** a webhook whose every delivery has failed for three days is switched off. Ask us to switch it back on.

Repeats and order:

- An event can arrive more than once. Ignore an `id` you have already handled.
- Events can arrive out of order. Use `milestone.at`, and `GET /v1/tracking` when in doubt.

Changing the signing secret. `POST /v1/webhooks/secret` (with a key of full access) makes a new secret for the webhook of that key's mode and returns it once. Our staff can do the same from Dispatch. For 24 hours every event is signed with both secrets, so you can switch over without losing any; then only the new one.

```
curl -X POST https://nexroo.com.au/api/v1/webhooks/secret \
  -H "Authorization: Bearer nxk_live_…"
```

```
{ "secret": "whsec_…", "previousSecretValidUntil": "2026-10-22T10:15:00+11:00" }
```

Webhooks belong to the account: every key's consignments are sent. Our staff can also send a test or resend a failed event from Dispatch.

Your own webhooks

A key with full access can make webhooks for its mode (live or test) without asking us. Each has its own signing secret, shown once when it is made; to change it, delete the webhook and make a new one. A new webhook hears events within a minute.

```
curl https://nexroo.com.au/api/v1/webhooks \
  -H "Authorization: Bearer nxk_live_…" \
  -H "Content-Type: application/json" \
  -d '{
    "url": "https://hooks.example.com/nexroo",
    "events": ["consignment.delivered", "consignment.failed_attempt"],
    "description": "Delivery updates to our order system"
  }'
```

```
{
  "id": "wh_AbCdEf0123456789",
  "url": "https://hooks.example.com/nexroo",
  "description": "Delivery updates to our order system",
  "events": ["consignment.delivered", "consignment.failed_attempt"],
  "enabled": true,
  "disabledReason": null,
  "failingSince": null,
  "secretHint": "k2Pq",
  "createdAt": "2026-11-06T09:12:00+11:00",
  "secret": "whsec_…"
}
```

- `events` left out or empty: every milestone. Both `consignment.delivered` and `delivered` are accepted.
- `url` must be HTTPS on the standard port, on the public internet. A URL one of your webhooks already posts to is refused with `409 conflict` and that webhook's id.
- `GET /v1/webhooks` lists yours (never the secret); `GET /v1/webhooks/{id}` shows one.
- `DELETE /v1/webhooks/{id}` removes it (`204`). Deliveries still queued to it are dropped.
- `POST /v1/webhooks/{id}/test` posts a `ping` now and answers `{ "delivered": true, "status": 200, "error": null }`, or what went wrong.
- The webhooks we set up for you are not listed here and cannot be deleted here; ask us.
- A webhook switched off (410 Gone, or three days of failures) stays listed with `enabled: false` and the reason. Delete it and make a new one.

Recent events

`GET /v1/events` answers with the latest events on your newest consignments (live or sandbox, by the key's mode), newest first, in exactly the shape your webhook receives them. Use it to see what an event looks like before any has been posted, or to load a sample into an integration.

```
curl "https://nexroo.com.au/api/v1/events?type=consignment.delivered&limit=3" \
  -H "Authorization: Bearer nxk_test_…"
```

```
{ "events": [ { "id": "evt_…", "type": "consignment.delivered", "createdAt": "…", "mode": "test", "data":
   { "consignment": { … }, "milestone": { … } } } ] }
```

- `type`: only these event types, comma-separated. Left out: all of them.
- `limit`: 1 to 10; 3 if left out. The events come from your last ten consignments.
- `createdAt` is the milestone's time, since the event is rebuilt rather than replayed.

Zapier and Make

Nexroo is an app on Zapier and on Make. Connect it with a key from the nexbiz dashboard, then build a workflow such as "new paid Shopify order → Nexroo: create consignment", or "Nexroo: consignment delivered → email the customer". Behind the scenes the apps use this API: bookings are `POST /v1/consignments` with `channel` set to the platform, and the triggers are webhooks made with `POST /v1/webhooks`, deleted again when the workflow is switched off. A test key runs the whole workflow in the sandbox.

Sandbox

Build and test with a **test key** (`nxk_test_…`). It uses the same base URL and endpoints.

- **Quotes** are real prices.
- **Bookings** are checked and priced exactly like live ones, but they are kept in a sandbox:
  - Their ids start `NT-`.
  - They never reach our drivers or your invoice.
  - Addresses are not looked up, so any address in a suburb we serve is accepted.
- **Tracking, cancelling and labels** work on test consignments:
  - Labels are marked TEST - DO NOT SHIP.
  - There is no public tracking page (`trackingUrl` is null), no 13-digit parcel numbers, and no proof-of-delivery images.
- **Key separation:** a test key never sees live consignments, and a live key never sees test ones.

Move a test consignment on yourself, as a driver or the depot would:

```
curl https://nexroo.com.au/api/v1/sandbox/consignments/NT-DXY78T/events \
  -H "Authorization: Bearer nxk_test_…" \
  -H "Content-Type: application/json" \
  -d '{ "code": "picked_up" }'
```

Start	Steps, in order
Pickup	`picked_up`, `out_for_delivery`, `delivered`
Drop-off	`received_at_depot`, `out_for_delivery`, `delivered`
A failed attempt (from `out_for_delivery`)	`failed_attempt`, `returned_to_depot`, then `out_for_delivery` again or `returned_to_sender`
Before collection	`cancelled`

Each step answers with the tracking and posts the event to your **test** webhook, signed with that webhook's own secret. A step that cannot happen yet answers `409`, saying what must come first.

Going live

1. Build against the sandbox with a test key and a test webhook.
2. Check your signature verification with a test event from us.
3. Make a live key in the nexbiz dashboard (or ask us), and a live webhook if you want pushes.
4. Book with the live key. Live consignments are real deliveries, invoiced to your account.

Changes

1.2.0 (6 Nov 2026)

- Keys are made and revoked by the account owner in the nexbiz dashboard; staff still can.
- `channel` on a booking says which system it came from, and comes back on the consignment.
- Your own webhooks: `POST /v1/webhooks`, `GET /v1/webhooks`, `GET /v1/webhooks/{id}`, `DELETE /v1/webhooks/{id}` and `POST /v1/webhooks/{id}/test`, up to 10 per mode, each with its own secret.
- `GET /v1/events`: the latest events in webhook shape, for samples.
- Nexroo on Zapier and Make.

1.1.1 (31 Oct 2026)

- A parcel given by weight and dimensions is sized by the first size both fit; cubic weight no longer applies. The size weights are now 3, 10, 20 and 25 kg.
- A booked pickup is free: `pickupFeeIncGstCents` is always 0.
- The standard signature fee is $3.30.

1.1.0 (21 Oct 2026)

- Proof of delivery gives the recipient's first name only, and for alcohol only the type of ID checked (no characters of its number).
- Delivery photos and signatures are kept 12 months after delivery.
- Keys can be made without proof-of-delivery access.
- `RateLimit-Limit`, `RateLimit-Remaining` and `RateLimit-Reset` on every answer.
- `POST /v1/webhooks/secret`; after any change of secret, the old one signs alongside for 24 hours.
- `GET /v1/account` shows the key's access and when it was made.
- Times are stated as targets; the cancellation fee and standard cover are stated.