Kestrel API reference

A REST API over HTTPS with JSON bodies. Every object the dispatch board, the driver app, the CRM and the dispatch assistant can see is reachable here with the same permissions model. This page covers authentication, the two endpoints integrators ask for first, and the limits you will hit before anything else.

BASE URL	AUTH	FORMAT	STATUS
`https://api.kestrel-freight.example/v1`	`Bearer token (scoped API key)`	`application/json · UTF-8 · RFC 3339 times`	`99.96% last 90 days · status.kestrel`

Authentication #auth

Create a key in `Settings → API keys` on the dispatch board. Keys are scoped to a depot and a set of resources, for example `shipments:write loads:read`. Send it in the `Authorization` header. Keys issued to the dispatch assistant carry the `assistant` scope, which can read and propose but never publish a load or send an email.

> Keys are shown once. Rotating a key keeps the old one valid for 24 hours so a deploy can roll over without downtime. Revocation is immediate.

POST /shipments

Create a shipment for a shipper account

BODY PARAMETERS

`shipper_account_id` required	uuid	Account the shipment belongs to. Must be in the key's book.
`ref` required	string	Your reference. Unique per account; reuse returns the existing shipment (idempotent).
`pickup` required	object	Address, `window_from`, `window_to`. Geocoded on create.
`delivery` required	object	Same shape as `pickup`.
`pallets`	integer	1–33 for a standard trailer. Default `1`.

curl | Node | Python | Response 201

```
# request
curl -X POST $BASE/shipments \
  -H "Authorization: Bearer kf_live_…" \
  -H "Content-Type: application/json" \
  -d '{ "shipper_account_id": "9b1e…", "ref": "RTD-20481",
        "pickup": { "address": "Maasvlakte 2, Rotterdam",
          "window_from": "2026-05-07T06:00:00+02:00",
          "window_to":   "2026-05-07T09:00:00+02:00" },
        "delivery": { "address": "Hafenstr. 12, Duisburg",
          "window_from": "2026-05-07T13:00:00+02:00",
          "window_to":   "2026-05-07T17:00:00+02:00" },
        "pallets": 26, "weight_kg": 18400 }'

# response 201 Created
{
  "id": "shp_01J3Q8M7X4N2",
  "ref": "RTD-20481",
  "status": "booked",
  "pallets": 26, "weight_kg": 18400, "adr": false,
  "distance_km": 214.6,
  "created_at": "2026-05-06T15:12:40Z"
}
```

GET /loads?depot=venlo&date=2026-05-07

List planned truck-days for a depot and date

QUERY PARAMETERS

`depot` required	enum	`venlo` or `duisburg`.
`date` required	date	Plan date, `YYYY-MM-DD`, local depot time.
`status`	enum[]	`draft`, `review`, `published`, `in_progress`, `done`. Comma-separated.
`planned_by`	enum	`user` or `assistant`. Useful for audit exports.
`cursor`	string	Opaque cursor from `next_cursor`. Page size is fixed at 50.

Response 200 | curl | Node

```
{
  "data": [
    {
      "id": "ld_01J3QA2K9P7R",
      "depot": "venlo",
      "plan_date": "2026-05-07",
      "vehicle": { "plate": "VL-412-K", "trailer_type": "curtainside" },
      "driver":  { "id": "drv_0188", "adr": true },
      "status": "published",
      "planned_by": "assistant",
      "stops_count": 6,
      "hos_ok": true,
      "eta_last_stop": "2026-05-07T16:42:00+02:00"
    },
    … 49 more
  ],
  "next_cursor": "eyJwIjoyfQ",
  "total": 71
}
```

Limits apply per API key and are returned on every response in `X-RateLimit-Limit`, `X-RateLimit-Remaining` and `X-RateLimit-Reset`. Exceeding a limit returns `429` with a `Retry-After` header; clients should back off exponentially starting at 1 second.

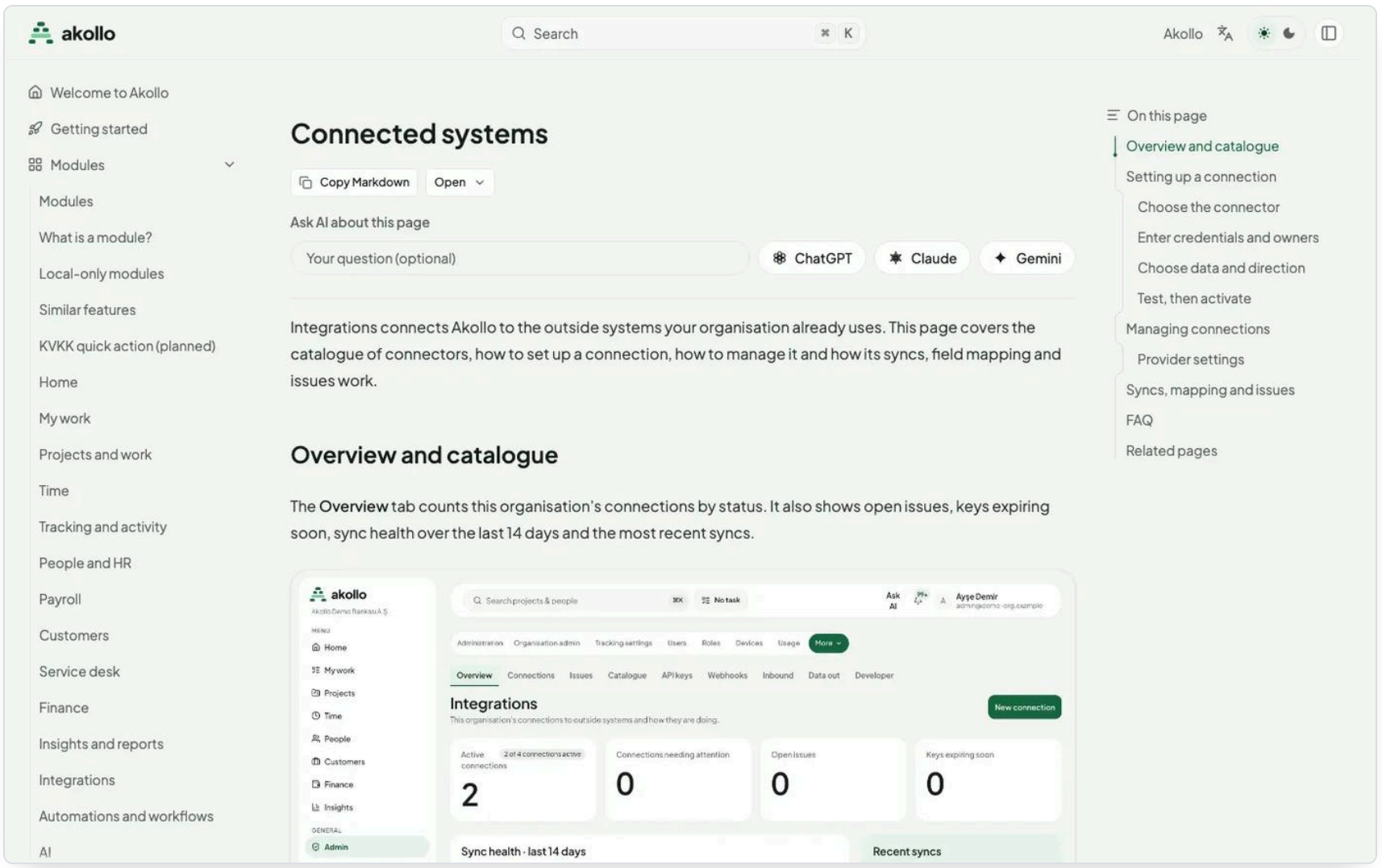
600 / min
Standard integration key, all read endpoints

120 / min
Write endpoints (POST, PATCH, DELETE)

20 / min
`POST /plans/propose`; each call plans one depot-day

REFERENCE · INTEGRATION DOCS ON A TECHSY DOCS SITE

The same reference is published next to the guides, so integrators find the auth rules and the connection steps in one place.



Akollo docs · connected systems. Example screen from a Techsy project, not Kestrel Freight's product.