

Publikus API & ERP Webhook Integrációs Kézikönyv

Összeköttetés vállalatirányítási (ERP), CRM, TMS vagy saját fuvarszervező szoftverekkel. Automatikus adatkinyerés, ajánlatkérések, státuszfrissítések és kétirányú szinkronizáció.

Verzió: 2026.1 (Stabil) | Csomag: PRO & Enterprise | Formátum: REST JSON & Webhooks

1. Bevezetés és Hitelesítés (Authentication)

A Cargonity Public API lehetővé teszi, hogy a beérkező fuvarmegbízásokból kinyert strukturált adatokat közvetlenül beemeld a meglévő informatikai rendszereidbe (pl. SAP, Odoo, Navision, egyedi ERP), illetve automatizáld a válaszadást.

Hitelesítés API kulccsal (X-API-KEY):

Minden API kérés fejlécében kötelező megadni a fiókod egyedi API kulcsát. Ezt a Cargonity beállítások menüpontjában éred el.

HTTP Header: X-API-KEY: a1b2c3d4-e5f6-7a8b-9c0d-e1f2a3b4c5d6

Alapértelmezett Base URL: <https://cargonity.com/api>

Biztonsági figyelmeztetés: Az API kulcs teljes hozzáférést biztosít a fiókodhoz. Kizárólag szerveroldali hívásokban használd!

2. Szállítmányok Kezelése (Shipments API)

A beérkező és elemzett fuvarok lekérdezése, manuális rögzítése, státuszának módosítása és jóváhagyása.

GET

/api/shipments/

Szállítmányok listázása szűrőkkel

Lekérdezi a feldolgozott szállítmányokat dátum, státusz vagy ERP készenlét szerint.

Query Paraméterek / Fejlécek:

- `date_from` / `date_to`: Időszak szűrő YYYY-MM-DD (vagy 'today' / 'ma')
- `from_id`: Minimális szállítmány ID folyamatos szinkronhoz
- `erp_ready`: Logikai szűrő (true/false) — csak az ERP átadásra kész megbízások
- `needs_attention`: Logikai szűrő (true/false) — kézi beavatkozást igénylő fuvarmegbízások
- `status`: Közvetlen státusz szűrő (pl. `READY_FOR_ERP`, `NEEDS_ATTENTION`)

Válasz minta (200 OK JSON):

```
[
  {
    "id": 2309,
    "reference_number": "PO-4500123456",
    "position_number": "10",
    "loading_city": "Budapest",
    "loading_country_code": "HU",
    "loading_date": "2026-07-01",
    "loading_date_to": "2026-07-02",
    "unloading_city": "München",
    "unloading_country_code": "DE",
    "unloading_date": "2026-07-03",
    "goods_name": "Steel plates on pallets",
    "pallet_count": 4,
    "weight_kg": 1200.0,
    "price": 1450.0,
    "currency": "EUR",
    "status": "READY_FOR_ERP",
    "erp_ready": true,
    "confidence": 0.98
  }
]
```

GET

/api/shipments/{shipment_id}

Egyedi szállítmány adatlapja

Lekérdezi egy konkrét szállítmány teljes rekordját az ID alapján.

Query Paraméterek / Fejlécek:

- `shipment_id`: Numeric shipment ID

PATCH**/api/shipments/{shipment_id}****Szállítmány frissítése és ERP általi átvétele**

Részleges frissítés (PATCH). A külső ERP ezen keresztül közvetlenül jóváhagyhatja vagy javíthatja a megbízást!

Request Body (JSON):

```
{
  "position_number": "PO-987654",
  "status": "READY_FOR_ERP",
  "notes": "Approved in external ERP."
}
```

Válasz minta (200 OK JSON):

```
{
  "id": 2309,
  "status": "READY_FOR_ERP",
  "position_number": "PO-987654",
  "updated_at": "2026-07-01T10:15:00Z"
}
```

POST**/api/shipments/{shipment_id}/force-erp-draft****Kényszerített átadás ERP-be (Draft / Incomplete)**

Ha az ERP a hiányzó adatok ellenére is átveszi a megbízást, azonnal átállítja READY_FOR_ERP-re és kiküldi a webhookot.

Válasz minta (200 OK JSON):

```
{
  "status": "ok",
  "message": "Status set to READY_FOR_ERP.",
  "shipment_id": 2309,
  "erp_webhook_triggered": true
}
```

POST**/api/shipments/{shipment_id}/generate-reply****AI ajánlat / válasz e-mail generálása**

A szállítmány adatai és a diszpécseri döntés alapján AI levelet fogalmaz.

Request Body (JSON):

```
{
  "action": "counter_offer",
  "price": 1600.0
}
```

Válasz minta (200 OK JSON):

```
{
  "subject": "Re: Quote Request: Budapest - Munich freight",
  "body": "Dear Partner! Thank you for the request. We can fulfill the transport for 1600 EUR..."
}
```

3. E-mailek és Üzenetek Kezelése (Emails API)

A beérkező levelek, PDF és egyéb csatolmányok metaadatainak lekérdezése és kezelése.

GET /api/emails/

Beérkező levelek listázása és keresése

Lapozható lista a beérkezett e-mailekről keresési lehetőségekkel.

Query Paraméterek / Fejlécek:

- page / page_size: Oldalszám és méret (alapértelmezett: 1 / 25)
- q: Keresési kifejezés (tárgy, feladó, tartalom)
- date_from / date_to: Időszak szűrő (YYYY-MM-DD)

POST /api/emails/{email_id}/send-reply

Közvetlen válaszlevél küldése

Válasz e-mail kiküldése a feladónak a Cargonity rendszerén keresztül.

Request Body (JSON):

```
{
  "to_email": "partner@freight-forwarder.com",
  "subject": "Re: Quote: Budapest - Munich",
  "body": "Dear Partner! Offer accepted."
}
```

Válasz minta (200 OK JSON):

```
{
  "status": "sent",
  "email_id": 4821
}
```

4. Valós Idejű ERP Webhook Értesítések (Push Webhooks)

A Cargonity ahelyett, hogy pollingot igényelne, képes valós időben HTTP POST hívással értesíteni a vállalatirányítási rendszert. A Webhook URL a Beállítások menüpontban adható meg.

Támogatott Webhook Események:

- shipment_ready: Akkor váltódik ki, amikor egy szállítmányhoz minden kötelező mező maradéktalanul rendelkezésre áll.
- shipment_needs_attention: Akkor váltódik ki, ha a 2 körös automata adatbekérés után is hiányzik kötelező ERP adat. Az ERP így azonnal feladatot hozhat létre a diszpécsernek!

Webhook HTTP POST Payload minta (JSON):

```
{
  "event": "shipment_needs_attention",
  "status": "NEEDS_ATTENTION",
  "shipment_id": 2309,
  "thread_token": "C-2309",
  "missing_fields": ["position_number", "dimensions"],
  "reference_number": "PO-123456",
  "position_number": null,
  "sender_company": "Partner Logistics GmbH",
  "sender_name": "Hans Gruber",
  "sender_phone": "+49 89 123456",
  "original_sender_email": "partner@freight.de",
  "loading_city": "Székesfehérvár",
  "loading_date": "2026-09-22",
  "unloading_city": "München",
  "unloading_date": "2026-09-23",
  "weight_kg": 4000.0,
  "pallet_count": 13,
  "price": 1000.0,
  "currency": "EUR"
}
```

5. Integrációs Kódpéldák (Code Examples)

Python (Requests):

```
import requests

url = "https://cargonity.com/api/shipments/"
headers = {
    "X-API-KEY": "a1b2c3d4-e5f6-7a8b-9c0d-e1f2a3b4c5d6",
    "Accept": "application/json"
}

response = requests.get(url, headers=headers, params={"erp_ready": "true"})
if response.status_code == 200:
    for shipment in response.json():
        print(f"Order #{shipment['id']}: {shipment['loading_city']} -> {shipment['unloading_city']}")
```

JavaScript / Node.js (Fetch API):

```
const response = await fetch('https://cargonity.com/api/shipments/?erp_ready=true', {
  headers: {
    'X-API-KEY': 'a1b2c3d4-e5f6-7a8b-9c0d-e1f2a3b4c5d6',
    'Accept': 'application/json'
  }
});
const shipments = await response.json();
console.log('Ready shipments count:', shipments.length);
```