DOCUMENTAȚIE TEHNICĂ · V1.1

Integrare parteneri API

Sistemul tău vorbește direct cu marketplace-ul MommyHAI: trimiți produse și stocuri, primești comenzi. Fără introducere manuală, fără fișiere trimise pe email.

Pentru
Echipa tehnică a partenerului

Contact
hai@mommyhai.com

Noiembrie 2026
Ediția a doua

01 Cum funcționează

PE SCURT

Astăzi, ca să vinzi prin MommyHAI, cineva ar trebui să adauge fiecare produs cu mâna, într-un panou. La un catalog de sute de produse, e imposibil de ținut la zi — mai ales stocul.

Cu integrarea prin API, sistemul tău ne trimite datele singur. Tu rămâi stăpân pe catalog, noi ne ocupăm de vânzare și de client.

1 · TRIMIȚI

Îți urci produsele printr-un apel, ori de câte ori se schimbă ceva.

2 · PUBLICĂM

Produsele apar automat în magazinul și în aplicația MommyHAI.

3 · COMANDĂ

Clientul cumpără și plătește cu cardul. Te anunțăm la prima verificare, din 5 în 5 minute.

4 · CONFIRMI

Livrezi cu logistica ta și marchezi comanda finalizată.

Alege modul de integrare

Există patru feluri în care putem lucra împreună. Diferă prin cine face efortul tehnic și cât de proaspete sunt datele. Alege-l pe cel care ți se potrivește — se poate schimba oricând.

MODUL A

Ne trimiți tu, prin API

Sistemul tău apelează API-ul nostru ori de câte ori se schimbă ceva.

Pentru cine: catalog mare, stoc care se mișcă des, ai echipă tehnică.

DISPONIBIL ACUM

MODUL B

Citim noi de la tine

Ne dai adresa feed-ului tău, iar noi îl citim periodic. Tu nu implementezi nimic.

Pentru cine: ai deja magazin online sau un export de produse.

DISPONIBIL ACUM

MODUL C

Trimiți un fișier

Ne trimiți un CSV cu produsele, îl importăm noi.

Pentru cine: catalog care se schimbă rar, fără resurse tehnice.

DISPONIBIL ACUM

MODUL D

Din panou, manual

Primești cont și adaugi produsele din browser.

Pentru cine: sub 30 de produse, actualizări rare.

DISPONIBIL ACUM

CUM ALEGI RAPID

Dacă ai deja un magazin online pe o platformă cunoscută, întreabă-ne de **modul B** — de obicei e cel mai puțin de muncă pentru tine. Dacă ai peste 30 de produse și stoc care se schimbă în timpul zilei, **modul A** e singurul care ține pasul.

Comenzile ajung la tine în trei feluri

Indiferent de modul ales mai sus, poți primi comenzile prin oricare dintre canalele de mai jos — sau prin toate deodată.

CANAL	CUM E	STARE
Email	Un email pentru fiecare comandă nouă, cu tot ce trebuie ca să o pregătești. Zero implementare.	disponibil
Notificare automată (webhook)	Trimitem comanda direct în sistemul tău, semnată criptografic. Cel mai rapid.	disponibil
Întrebi tu (API)	Sistemul tău întreabă periodic dacă sunt comenzi noi.	disponibil

Piesa centrală: SKU-ul

Toată integrarea se sprijină pe **codul tău de produs (SKU)**. El este numele comun prin care ambele sisteme știu despre ce produs vorbesc. Trimite același SKU de două ori și produsul se actualizează — nu apare de două ori în magazin.

CONSECINȚĂ PRACTICĂ

SKU-ul nu trebuie să se schimbe niciodată pentru un produs existent. Dacă îl schimbi, pentru noi devine un produs nou, iar cel vechi rămâne în magazin.

Ce primești de la noi

ELEMENT	LA CE FOLOSEȘTE
Cheie API `mhk_…`	Identifică apelurile tale. Se trimite la fiecare cerere.
Secret webhook	Cu el verifici că notificările de comandă vin într-adevăr de la noi.
Cont în platformă	Opțional: intri și vezi vizual produsele și comenzile.

SECURITATE

Cheia și secretul se transmit o singură dată, pe canal securizat. Nu le pune în cod public sau în repository. Dacă bănuiești o scurgere, anunță-ne și îți emitem altele în câteva minute — produsele tale rămân legate de contul tău, nu de cheie, deci nu se pierde și nu se dublează nimic.

02 Modul C · Trimiți un fișier cu produse

PE SCURT

Cea mai simplă variantă: ne trimiți un fișier CSV cu produsele tale, iar noi îl încărcăm. Nu ai nevoie de programator, doar de un export din sistemul tău sau de un tabel completat corect.

Potrivit dacă produsele și prețurile nu se schimbă zilnic. Pentru stoc care se mișcă în timpul zilei, treci pe modul A.

Cum arată fișierul

CSV, salvat cu **UTF-8** și separator virgulă. Prima linie e antetul cu numele coloanelor. Un rând = un produs.

```
sku,name,description,price,stock_quantity,featured_image_url,ingredients,weight
IAURT-125,Iaurt 125g,Iaurt cremos,7.50,120,https://exemplu.ro/iaurt.jpg,lapte,0.125
SUC-250,Suc de mere 250ml,Suc presat,9.90,80,https://exemplu.ro/suc.jpg,mere 100%,0.250
```

COLOANĂ		OBSERVAȚII
`sku`	**obligatoriu**	Codul tău unic; pe el se face potrivirea la reimport
`name`	**obligatoriu**	—
`description`	opțional	Dacă are virgule, pune textul între ghilimele. Fără el, produsul intră oricum, dar pagina lui publică rămâne fără text descriptiv (primești un `warning` în răspuns).
`price`	**obligatoriu**	Punct zecimal: `7.50`. Fără simbolul monedei
`stock_quantity`	**obligatoriu**	Număr întreg
`featured_image_url`	opțional	Adresă publică (HTTPS recomandat)
`gallery_image_urls`	opțional	Mai multe adrese separate prin `|`
`ingredients` `weight`	opțional	La alimente, în practică obligatorii
`sale_price` `stock_status`	opțional	Vezi capitolul 09 pentru toate câmpurile

Coloanele pe care nu le recunoaștem sunt ignorate, nu dau eroare — poți trimite exportul tău așa cum e, chiar dacă are coloane în plus.

CUM NI-L TRIMIȚI

Cel mai bine: îl pui la o adresă stabilă (link de download, Drive partajat) și ne dai adresa. Atunci îl putem reciti periodic, fără să ni-l retrimiți de fiecare dată. Alternativ, îl atașezi pe email.

CE SE ÎNTÂMPLĂ LA REIMPORT

Potrivirea se face după `sku` : produsele existente se actualizează, cele noi se adaugă. **Un produs scos din fișier nu se șterge automat** din magazin — ne spui explicit ce retragi, ca să nu dispară ceva din greșeală.

03 Modul B · Citim noi din sistemul tău

PE SCURT

Dacă ai deja un magazin online sau un export automat de produse, nu mai are rost să implementezi nimic. Ne dai adresa de unde citim și un acces, iar noi venim periodic să luăm datele.

E varianta cu cel mai puțin efort de partea ta — de obicei o configurăm într-o zi.

Ce ne trebuie de la tine

CERINȚĂ	DETALII
O adresă stabilă	Un URL care întoarce lista de produse și nu se schimbă în timp.
Format	JSON, XML sau CSV. Nu contează structura exactă — ne adaptăm noi la a ta, atâta timp cât e consecventă.
Acces	Cheie API, user și parolă, sau token în adresă. Ne e suficient un acces **doar de citire** — nu avem nevoie să scriem nimic la tine.
Câmpurile	Cel puțin: cod produs, denumire, descriere, preț, stoc, imagine. Vezi lista completă la capitolul 09.
Paginare	Dacă ai peste ~500 de produse, spune-ne cum se cere pagina următoare.
Frecvență permisă	Cât de des putem interoga fără să te deranjăm. Implicit: catalogul o dată pe zi, stocul la fiecare 30 de minute.

Platforme pe care le citim direct

Dacă magazinul tău rulează pe una dintre acestea, nu trebuie să pregătești nimic — ne dai doar un acces de citire:

PLATFORMĂ	CE NE DAI
WooCommerce	O pereche consumer key / consumer secret, cu drept de citire
Shopify	Un token de acces la Admin API sau Storefront API
Magento, PrestaShop, OpenCart	Acces la API-ul standard al platformei
Altceva sau sistem propriu	Orice adresă care întoarce produsele — discutăm 15 minute și ne adaptăm

Cum decurge

PAS	CINE	DURATĂ
Ne dai adresa și accesul de citire	tu	10 minute
Citim feed-ul și potrivim câmpurile pe ale noastre	noi	în aceeași zi
Îți arătăm primele produse citite, să confirmi că sunt corecte	împreună	15 minute
Pornim sincronizarea automată	noi	—

Pentru WooCommerce și Shopify potrivirea e deja făcută — mergem direct la confirmare.

> **CE FACEM NOI ÎN SCHIMB**
> Ne ocupăm de traducerea datelor tale în formatul nostru, de descărcarea imaginilor, de maparea categoriilor și de monitorizarea sincronizării. Dacă feed-ul tău se strică sau își schimbă structura, **te anunțăm** — nu descoperi din vânzări pierdute.

> **CÂT DE DES CITIM**
> Implicit, catalogul complet o dată pe zi. Dacă stocul tău se schimbă des, putem citi mai rar sau mai frecvent — ne spui tu ce suportă sistemul tău, ca să nu îl încărcăm.

04 Modul A · Autentificare API

> **PE SCURT**
> O singură cheie secretă, trimisă la fiecare apel. Nu există login, sesiuni sau parole de reînnoit.

```
Authorization: Bearer mhk_cheia_ta
Accept: application/json
Content-Type: application/json
```

Trimite `Accept: application/json` **la toate cererile.** Recomandat: răspundem JSON și fără el, pe toate rutele de partener. Dacă generezi un client din fișierul OpenAPI, setează antetul ca implicit global.

Adresa de bază a tuturor apelurilor:

```
https://web.mommyhai.com/api/v1/partner
```

Primul apel, ca să confirmi cheia (un cont fără produse răspunde `{"data":[]}`):

```
curl https://web.mommyhai.com/api/v1/partner/products \
  -H "Authorization: Bearer mhk_cheia_ta" \
  -H "Accept: application/json"
```

Toate cererile merg pe HTTPS. Limita este de **60 de cereri pe minut**, socotită pe cheia ta — nu pe adresa IP, deci nu contează de pe câte servere apelezi.

Cât durează un apel

OPERAȚIUNE	DURATĂ TIPICĂ	DE CE
Produs nou	2–4 secunde	Îl publicăm și în magazin, nu doar la noi
Actualizare produs	~1 secundă	Produsul există deja
Stoc sau preț	~1 secundă	Modificare punctuală

> **RECOMANDARE DE IMPLEMENTARE**
>
> Pune un timeout de cel puțin **300 de secunde** pe `PUT /products` (produsele se procesează unul după altul) și trimite loturi de cel mult 25 la primul import. Dacă întrerupi o cerere pe la jumătate, produsele deja procesate rămân salvate — la reluare vor fi recunoscute după SKU și actualizate, nu duplicate.

05 Produse

> **PE SCURT**
>
> Trimiți o listă de produse. Noi le creăm pe cele noi și le actualizăm pe cele existente, potrivindu-le după SKU. Dacă un produs din listă are o problemă, restul intră normal — nu pierzi tot lotul pentru o greșeală.

> **ÎNAINTE DE PRIMUL APEL**
>
> Nu există mediu de test separat, iar un produs nu se șterge prin API. Fă prima probă cu un SKU `TEST-…` și `"is_active": false`: produsul se salvează, dar nu apare în magazinul public. Exemplele de mai jos au `is_active: false` tocmai de aceea; scoate-l când vrei să publici.

`PUT` `/products`

```
curl -X PUT https://web.mommyhai.com/api/v1/partner/products \
  -H "Authorization: Bearer mhk_cheia_ta" \
  -H "Accept: application/json" \
  -H "Content-Type: application/json" \
  -d '{
    "products": [{
      "sku": "IAURT-125",
      "is_active": false,
      "name": "Iaurt cu căpșuni 125g",
      "description": "Iaurt cremos cu bucăți de căpșuni, fără conservanți.",
      "price": 7.50,
      "stock_quantity": 120,
      "featured_image_url": "https://exemplu.ro/iaurt.jpg",
      "ingredients": "lapte, căpșuni 12%, zahăr, culturi lactice vii",
      "weight": 0.125
    }]
  }'
```

Atenție: un `200` sau un `created` / `updated` nu garantează că produsul a ajuns și în magazinul public (sincronizarea poate eșua după salvare, fără să fie semnalată). După fiecare lot deschide câteva produse în magazin; dacă un produs lipsește sau are stoc sau preț vechi, scrie-ne. `GET /products` confirmă ce am înregistrat noi (SKU, `updated_at`, alergeni, nutriție), nu stocul sau prețul din vitrină.

Răspunsul îți spune verdictul pentru fiecare produs în parte:

```
{
  "summary": { "total": 1, "ok": 1, "failed": 0 },
  "results": [{
    "index": 0,
    "sku": "IAURT-125",
    "result": "created",
    "product_id": "8a7a3c59-9d73-4381-a6ea-70585194e6bc"
  }]
}
```

Valorile posibile: `created`, `updated` sau `error`. La eroare primești câmpurile exacte de reparat. Maximum **100 de produse** per cerere; pentru a rămâne sub timeout, trimite cel mult 50 de produse noi pe cerere. Dacă niciun produs din lot nu a intrat, răspunsul e `422` cu același corp (`summary` / `results`). Erorile `errors.upstream` (cu „timed out” sau „Backend error (5xx)”), `errors.conflict` și „Could not verify allergen codes” sunt temporare: reia peste câteva minute. „Backend error (4xx)” și „Shop ownership rejected… Check shop_id” nu sunt temporare: verifică datele și `shop_id`.

Imaginile

Ne dai adresa publică a imaginii, iar noi o descărcăm și o găzduim. Cerințe: adresă publică (HTTPS recomandat), format imagine, sub 5 MB, recomandat minim 800×800 px.

> **COMPORTAMENT LA EROARE**
>
> Dacă o imagine nu poate fi descărcată, **produsul se salvează oricum**, iar în răspuns apare un avertisment la `warnings`. Nu pierzi un produs bun pentru o poză stricată.

Ce avem înregistrat

`GET` `/products`

Îți întoarce SKU-urile tale și momentul ultimului `PUT` (`updated_at`; un `PATCH` de stoc sau preț nu îl schimbă). Util pentru o reconciliere periodică între catalogul tău și al nostru.

06 Stoc și preț

> **PE SCURT**
>
> Pentru schimbări dese, nu retrimite produsul întreg. Există un apel scurt, doar pentru stoc și preț — mai rapid și mai ieftin pentru amândoi.

`PATCH` `/products/{sku}`

```
curl -X PATCH https://web.mommyhai.com/api/v1/partner/products/IAURT-125 \
  -H "Authorization: Bearer mhk_cheia_ta" \
  -H "Accept: application/json" \
  -H "Content-Type: application/json" \
  -d '{ "stock_quantity": 0 }'
```

```
{
  "data": {
    "sku": "IAURT-125",
    "stock_quantity": 0,
    "stock_status": "outofstock"
  }
}
```

Poți trimite oricare dintre `stock_quantity`, `stock_status`, `price`, `sale_price`, `is_active`, `allergens`, `allergen_traces`, `nutrition` (vezi capitolul 10). Când stocul ajunge la zero, produsul devine automat indisponibil și nu mai poate fi comandat.

CÂT DE DES

Ideal, la fiecare schimbare reală de stoc. Minim acceptabil: o dată la 15–60 de minute. La produse proaspete, cu stoc mic, un stoc vechi înseamnă comenzi pe care nu le poți onora — iar clientul dă vina pe amândoi.

Limita de 60 de cereri pe minut e pe cheie și se împarte cu citirea comenzilor. La volum mare, pune modificările de stoc într-o coadă și trimite-le fără să depășești limita. La `PUT /products` pe un produs existent, un `description` lipsă sau gol lasă descrierea neschimbată.

07 Comenzi

PE SCURT

Vezi doar comenzile care conțin produsele tale — niciodată ale altui partener. Le poți citi oricând, iar când ai livrat, marchezi comanda drept finalizată.

GET `/orders`

```
curl "https://web.mommyhai.com/api/v1/partner/orders?status=processing" \
  -H "Authorization: Bearer mhk_cheia_ta" \
  -H "Accept: application/json"
```

Cere întotdeauna `status=processing` pentru comenzile de livrat. Fără `status`, lista include și comenzi neplătite sau eșuate (`pending` , `failed`) și cele anulate sau rambursate.

```
{
  "data": [{
    "order_id": 5441,
    "order_number": "5441",
    "status": "processing",
    "subtotal": "50.00", "shipping_total": "7.00", "discount_total": "0.00", "total": "50.00",
    "delivery_date": "2026-11-14", "delivery_slot": "07:00-09:00",
    "billing": { "first_name": "…", "phone": "…", "city": "…" },
    "address_components": { "zone": "București", "street": "…", "number": "…" },
    "awb": null, "tracking_url": null, "invoice_url": null,
    "line_items": [ … ]
  }],
  "pagination": { "total": 1, "page": 1, "per_page": 50 }
}
```

Filtre disponibile: `status` , `after` , `before` (dată ISO 8601), `page` , `per_page` (maxim 100). Identificatorul comenzii este `order_id` (număr întreg la `GET /orders` ; text, de exemplu `"5441"` , în răspunsurile `PATCH /orders` și de la factură). O comandă apare câte o dată pentru fiecare magazin al tău, doar cu liniile acelui magazin. `total` e valoarea produselor tale, **fără transport**; de încasat pentru rând: `total` + `shipping_total` . Datele cu oră sunt în UTC; `delivery_date` e în ora României. `after` și `before` se trimit în UTC, cu `Z` . Lista are cele mai recente comenzi primele, iar prin API pot fi citite individual sau modificate doar cele mai recente ~1.000 de rânduri-comandă: pentru o comandă mai veche,

`GET /orders/{id}`, `PATCH` și factura răspund `404`, deci închide comenzile la timp. În `pagination`, `total` și `total_pages` descriu pagina curentă, nu toate comenzile: citește paginile până primești `data: []`. Dacă ai mai multe magazine la noi, o pagină poate omite rânduri; filtrează cu `after` / `before` pe intervale scurte, cu cel mult `per_page` rânduri (implicit 50; folosește `per_page=100`). `invoice_url` e un link semnat, valabil o oră.

Varianta fără cod: comanda pe email

Dacă nu vrei să implementezi nimic pentru comenzi, îți trimitem un email la fiecare comandă nouă. Conține tot ce trebuie ca să o pregătești și să o livrezi. Așa arată:

De la MommyHAI <comenzi@mommyhai.com>
Către comenzi@partenerul-tau.ro
Subiect [MommyHAI] Comandă nouă #5441 — 39,90 lei

Ai o comandă nouă

Comanda **#5441** a fost plasată și **plătită cu cardul** pe 29 iulie 2026, ora 12:01.

PRODUS	COD	CANT.	PREȚ
Iaurt cu căpșuni 125g	IAURT-125	2	15,00 lei
Suc natural de mere 250ml	SUC-250	1	9,90 lei

Produse 24,90 lei · transport 15,00 lei · Total 39,90 lei

Livrare către

Maria Ionescu · 0722 000 000
Str. Exemplu nr. 10, ap. 4, sector 2, București, 020123

Mențiune de la client: „Vă rog sunați înainte de livrare.”

Marchează drept livrată

Exemplu ilustrativ: în email „Total” include transportul, iar „Preț” e valoarea liniei; în API, `total` e fără transport, iar `price` e prețul unitar. Butonul deschide pagina comenzii în contul tău MommyHAI. Poți schimba statusul și de acolo, sau prin API.

> **UNDE AJUNGE**
>
> La adresa de comenzi pe care ne-o dai la început. Poate fi o adresă de grup, ca să-l primească toată echipa. Emailul e o notificare — răspunsul la el nu ajunge automat în sistem; confirmarea comenzii se face din cont sau prin API.

Confirmarea comenzii

`PATCH` `/orders/{order_id}`

```
curl -X PATCH https://web.mommyhai.com/api/v1/partner/orders/5441 \
  -H "Authorization: Bearer mhk_cheia_ta" \
  -H "Accept: application/json" \
  -H "Content-Type: application/json" \
  -d '{ "status": "completed" }'
```

Același apel primește și `awb` (numărul de expediere al coletului, de la curier) și `tracking_url` (se trimit împreună). Poți trimite doar statusul, doar AWB + link, sau pe toate trei. La `completed` pentru produse fizice, AWB-ul și linkul sunt obligatorii; clientul primește emailul de urmărire când se schimbă.

```
# doar AWB + link de urmărire
curl -X PATCH https://web.mommyhai.com/api/v1/partner/orders/5441 \
  -H "Authorization: Bearer mhk_cheia_ta" \
  -H "Accept: application/json" \
  -H "Content-Type: application/json" \
  -d '{ "awb": "1234567890",
        "tracking_url": "https://curier.exemplu.ro/awb/1234567890" }'

# livrată, cu AWB + link în același apel
curl -X PATCH https://web.mommyhai.com/api/v1/partner/orders/5441 \
  -H "Authorization: Bearer mhk_cheia_ta" \
  -H "Accept: application/json" \
  -H "Content-Type: application/json" \
  -d '{ "status": "completed", "awb": "1234567890",
        "tracking_url": "https://curier.exemplu.ro/awb/1234567890" }'
```

`awb` : 3-40 de caractere (litere, cifre, spații sau `- _ . /`), începând cu literă sau cifră; fără `tracking_url` sau invers, răspunsul e 422. La `completed` nu mai ești obligat să trimiți AWB dacă magazinul tău are deja unul salvat pe comandă. `tracking_url` : adresă http(s), maxim 500 de caractere. Cel puțin unul dintre `status` , `awb` , `tracking_url` e obligatoriu, altfel răspunsul e 422. Dacă lipsește AWB-ul la `completed` , 422 cu mesajul de la platformă. Dacă schimbarea de status nu ajunge în magazin, răspunsul e `502` cu `status_not_applied` (AWB-ul rămâne salvat). `awb` și `tracking_url` goale, trimise împreună, șterg AWB-ul. Dacă aceeași comandă are produse din mai multe magazine ale tale, trimite și `shop_id` (numeric, ca la factură), ca să spui al cărui colet e AWB-ul.

STATUS	CÂND ÎL FOLOSEȘTI
`processing`	Ai preluat comanda și o pregătești
`in-curs-livrare`	Comanda a plecat spre client
`on-hold`	Ceva o blochează temporar
`completed`	Comanda a fost livrată
`cancelled`	Comanda nu poate fi onorată

Statusurile `pending`, `refunded` și `failed` țin de plată și rămân în grija platformei.

Statusul e al comenzii întregi, nu doar al rândului tău. Dacă pe comandă sunt și produse de la alt furnizor, un `completed`, `cancelled`, `on-hold` sau `in-curs-livrare` pus de tine îl schimbă și pentru el (el primește `order.cancelled` sau `order.completed`). Pe comenzile cu mai mulți furnizori nu seta `cancelled` fără să ne scrii. O comandă `pending`, `failed` sau `refunded` nu poate fi schimbată prin API (`422`, `invalid_status_transition`), iar una `cancelled` nu poate fi readusă în `processing`, `in-curs-livrare` sau `completed`. O comandă pusă `on-hold` de noi pentru o plată neconfirmată nu trebuie mutată în `processing`: dacă ai dubii, scrie-ne.

Atașarea facturii

Încasarea o face MommyHAI; factura o emiți tu și o atașezi la comandă.

POST `/orders/{order_id}/invoice`

```
curl -X POST https://web.mommyhai.com/api/v1/partner/orders/5441/invoice \
  -H "Authorization: Bearer mhk_cheia_ta" \
  -H "Accept: application/json" \
  -F "file=@factura-5441.pdf"
```

```
HTTP/1.1 201 Created
{
  "data": {
    "order_id": "5441",
    "shop_id": 1614,
    "filename": "1794650400000-factura-5441.pdf",
    "uploaded_at": "2026-11-14T10:00:00+00:00"
  }
}
```

Fișierul	PDF, trimis ca `multipart/form-data` în câmpul `file`. Verificăm conținutul, nu numele: fișierul trebuie să înceapă cu `%PDF-`. Maximum 5 MB.
Ce comandă	Doar o comandă care conține produse de-ale tale, adică una pe care o vezi la `GET /orders`.
Mai multe magazine	Dacă aceeași comandă are produse din mai multe magazine ale tale, trimite și `shop_id` (ca text în același formular) ca să spui al cui e factura. Fără el primești `422 shop_id_required` cu lista magazinelor.
A doua oară	Pe aceeași comandă și același magazin se atașează o singură factură. A doua încercare primește `409 invoice_exists`, iar factura existentă rămâne. Dacă reîncerci după un timeout și primești 409, factura s-a atașat. Pentru o corecție emite storno și o factură nouă și scrie-ne.

RĂSPUNS	CE ÎNSEAMNĂ
`404 not_found`	Comanda nu există sau nu are produse de-ale tale (ori `shop_id` nu e pe ea).
`422 file_required`	Lipsește câmpul `file`.
`422 invalid_file`	Fișierul nu e PDF.
`422 file_too_large`	Fișierul are peste 5 MB.
`422 shop_id_required`	Comanda are produse din mai multe magazine ale tale; trimite `shop_id`.
`409 invoice_exists`	Comanda are deja factura ta.

08 Notificări de comandă (webhook)

> **PE SCURT**
>
> Nu trebuie să ne întrebi periodic dacă a venit vreo comandă. Dacă ne dai o adresă de webhook, te sunăm noi — trimitem un mesaj către sistemul tău la prima verificare de după eveniment (din 5 în 5 minute; cu reîncercări, până la ~10 minute).
>
> Fiecare mesaj e semnat criptografic, ca să poți verifica sigur că vine de la noi și nu de la altcineva care ți-a aflat adresa.

```
POST adresa-ta-de-webhook
X-MommyHAI-Event: order.processing
X-MommyHAI-Timestamp: 1785326740
X-MommyHAI-Signature: t=1785326740,v1=3f7a…

{
  "event": "order.processing",
  "order": {
    "order_id": 5441, "status": "processing",
    "subtotal": "50.00", "shipping_total": "7.00", "discount_total": "0.00", "total": "50.00",
    "delivery_date": "2026-11-14", "delivery_slot": "07:00-09:00",
    "shipping_lines": [ { "method_title": "Livrare, sâmbătă, 14 nov., 07:00-09:00", "total":
"7.00" } ],
    "line_items": [ … ]
  },
  "sent_at": "2026-07-29T12:05:40+00:00"
}
```

Verificarea semnăturii

Vector de test pentru implementarea ta: secretul `secret_de_test`, timestamp-ul `1785326740` și corpul brut `{"event":"order.processing","order":{"order_id":5441}}` dau `v1=2d1a465293ef528f6f8ce0741ebc31e735af4cd72a91277a633bb9d8c49373d5`. Vectorul are un timestamp vechi: verifică semnătura separat de regula de vechime.

Calculezi HMAC-SHA256 peste `timestamp + "." + corpul brut`, folosind secretul tău, și compari cu valoarea `v1` din antet.

```
// Node.js
const crypto = require("crypto");

function esteValid(corpBrut, anteteBrute, secret, acum = Date.now() / 1000) {
  // antete cu litere mici
  const antete = Object.fromEntries(
    Object.entries(anteteBrute).map(([k, v]) => [k.toLowerCase(), v]));
  const ts = antete["x-mommyhai-timestamp"];
  const semnatura = antete["x-mommyhai-signature"] || "";
  if (!ts || !Number.isFinite(Number(ts))) return false;
  // >5 min: respinge (test: acum = 1785326740)
  if (Math.abs(acum - Number(ts)) > 300) return false;
  const asteptat = crypto
    .createHmac("sha256", secret)
    .update(ts + "." + corpBrut)
    .digest("hex");
  const primit = (semnatura.match(/v1=([0-9a-f]+)/) || [])[1] || "";
  const a = Buffer.from(asteptat), b = Buffer.from(primit);
  return a.length === b.length && crypto.timingSafeEqual(a, b);
}
```

```
// PHP
function esteValid(string $corpBrut, array $antete, string $secret, ?int $acum = null): bool {
    $acum ??= time();
    // getallheaders() poate întoarce numele cu litere mici
    $antete    = array_change_key_case($antete, CASE_LOWER);
    $ts        = $antete['x-mommyhai-timestamp'] ?? '';
    $semnatura = $antete['x-mommyhai-signature'] ?? '';
    if ($ts === '' || !ctype_digit($ts)) return false;
    // mesaj mai vechi de 5 minute (la testul cu vectorul, pune $acum = 1785326740)
    if (abs($acum - (int) $ts) > 300) return false;
    $asteptat  = hash_hmac('sha256', $ts . '.' . $corpBrut, $secret);

    // extrage partea v1= din "t=...,v1=..."
    $pos = strpos($semnatura, 'v1=');
    if ($pos === false) return false;
    $primit = substr($semnatura, $pos + 3);

    return hash_equals($asteptat, $primit);
}
```

> **TREI CAPCANE DE EVITAT**
>
> **1.** Folosește **corpul brut** al cererii, nu unul serializat din nou din obiect — orice diferență de spațiu sau ordine strică semnătura.
>
> **2.** Respinge mesajele mai vechi de câteva minute, ca protecție împotriva reluării unui mesaj interceptat.
>
> **3.** Răspunde cu `200` imediat și procesează după. Dacă răspunzi cu eroare sau nu răspunzi în 15 secunde, reîncercăm, până la cinci încercări în total.

Evenimentul se numește `order.<status>`: `order.processing`, `order.in-curs-livrare`, `order.on-hold`, `order.completed`, `order.cancelled`, `order.refunded`. `order.pending` și `order.failed` nu se trimit, dar `order.cancelled` **poate veni și pentru o comandă care nu a fost niciodată plătită: livrează doar la** `order.processing`. Include și schimbările de status făcute de tine prin API.

Pentru aceeași comandă, același status se trimite **o singură dată**: dacă o comandă revine la un status prin care a mai trecut, nu primești notificare. Un AWB nou sau o factură atașată nu declanșează notificare. Ca plasă de siguranță, poți oricând interoga `GET /orders/{order_id}`.

- Se verifică doar comenzile **create în ultimele 48 de ore** (cel mult 100 de rânduri), la fiecare 5 minute, și se vede doar statusul de la verificare: două schimbări rapide se pot reduce la ultima. O schimbare de status la o comandă mai veche de 48 de ore nu se anunță: citește `GET /orders/{order_id}`.
- Pe o comandă cu produse din mai multe magazine ale tale primești o singură notificare, pentru primul rând; celelalte se citesc din `GET /orders`.
- La o reîncercare, după un eșec, corpul poate avea altă formă (comanda întreagă, cu `id` în loc de `order_id` și cu liniile tuturor furnizorilor; câmpurile interne sunt scoase). Decide după antetul `X-MommyHAI-Event` și după `order_id` sau `id` din corp, și confirmă cu `GET /orders/{order_id}`.
- Deduplică pe partea ta după `(order_id sau id din corp, antetul X-MommyHAI-Event)`: o reîncercare după un răspuns pierdut poate livra aceeași notificare de două ori. Citește corpul brut (în PHP, `file_get_contents('php://input')`).
- Timestamp-ul din antet se recalculează la fiecare încercare, deci o reîncercare trece de regula de vechime. Adresa webhook-ului trebuie să răspundă 2xx **direct, fără redirect** (nu urmărim redirectele: un `3xx`, de exemplu `http` → `https`, e socotit eșec și se reîncearcă). Un endpoint căzut mai mult de ~10 minute pierde notificarea (cel mult 5 încercări, aproximativ două la fiecare verificare).

`delivery_date` și `delivery_slot` apar doar pentru partenerii cu program de livrare (ziua și intervalul alese de client). `shipping_total` este transportul încasat de la client pentru produsele tale; `subtotal` este valoarea produselor, iar `total` este valoarea produselor tale după reduceri, fără transport.

09 Câmpurile unui produs

CÂMP		DESCRIERE
`sku`	**obligatoriu**	Codul tău unic, maximum 100 de caractere. Nu se schimbă niciodată. Evită `/`: un SKU cu `/` nu poate fi adresat la `PATCH /products/{sku}`.
`name`	**obligatoriu**	Denumirea afișată clientului.
`description`	opțional	Text sau HTML simplu. Dacă lipsește, produsul se salvează fără descriere și primești un avertisment (`warnings`) în răspuns.
`price`	**obligatoriu**	Preț final în lei, cu TVA inclus. Punct zecimal: `7.50`.
`stock_quantity`	**obligatoriu**	Număr întreg de bucăți.
`featured_image_url`	opțional	Imaginea principală, adresă publică (HTTPS recomandat). La actualizare, o imagine schimbată nu e garantat propagată în magazinul web (se vede în aplicație); verifică în magazin.
`gallery_image_urls`	opțional	Până la 10 imagini suplimentare. La actualizare se **adaugă** la galeria existentă, nu o înlocuiesc: nu o retrimite dacă nu s-a schimbat.
`short_description`	opțional	Text scurt pentru liste.
`sale_price`	opțional	Preț promoțional, mai mic decât `price` (altfel eroare pentru produs); cu `sale_from` și `sale_to`.
`stock_status`	opțional	`instock`, `outofstock`, `onbackorder`. Se deduce din stoc dacă lipsește.
`is_active`	opțional	`false` = produsul intră ca **nepublicat** (draft) și nu apare în magazin. Dacă lipsește, produsul nou apare **publicat**. La actualizarea unui produs existent, absența câmpului lasă starea neschimbată.
`weight`	opțional	Kilograme. În practică obligatoriu la alimente.
`length width height`	opțional	Dimensiuni pentru livrare.
`ingredients`	opțional	Obligatoriu legal la produse alimentare.
`allergens`	opțional	Listă de coduri: alergenii pe care produsul îi **conține** ca ingredient. Vezi capitolul 10.
`allergen_traces`	opțional	Listă de coduri: „poate conține urme de” (contaminare încrucișată).
`nutrition`	opțional	Obiect cu valorile la 100 g/ml. Vezi capitolul 10.

Ai mai multe magazine la noi? Trimite și `shop_id` la nivelul cererii, ca să știm unde intră produsele. E identificatorul magazinului din contul tău MommyHAI (un UUID, pe care ți-l dăm noi) și nu are legătură cu `shop_id` numeric din comenzi și de la factură.

Erori

COD	CE ÎNSEAMNĂ	CE FACI
`401`	Cheie lipsă, greșită sau revocată	Verifică antetul `Authorization`
`404`	SKU sau comandă inexistentă pentru contul tău	Creează întâi produsul; verifică `order_id`
`422`	Date invalide	Răspunsul indică exact câmpul. Corectează și retrimite
`429`	Prea multe cereri	Așteaptă și reia; maximum 60 pe minut
`409`	Factură deja atașată (`invoice_exists`)	Factura existentă rămâne; pentru corecție, storno + factură nouă
`422`	Schimbare de status nepermisă (`invalid_status_transition`)	Vezi statusurile de la capitolul 07
`502` `503`	Problemă temporară la noi (inclusiv timeout: `upstream_timeout`; `status_not_applied` la un `PATCH` de status)	Reia peste câteva minute. După un `PATCH` cu eroare sau timeout, verifică starea cu `GET /orders/{order_id}`. Nu pierzi date

Erorile au trei forme: `{"error","message"}` (autentificare, resursă lipsă, reguli de business), `{"message","errors"}` (validarea cererii) și `{"summary","results"}` (`PUT /products` , rezultat per produs). Detaliile, cu toate câmpurile comenzii și exemple, sunt în fișierul OpenAPI / Swagger (`mommyhai-partner-api.html` și `.yaml`) pe care îl primești odată cu acest ghid.

10 Alergeni și valori nutriționale

Pentru produse alimentare, Regulamentul (UE) nr. 1169/2011 cere declararea alergenilor și a valorilor nutriționale. Le trimiți în câmpuri dedicate — nu în descriere, nu în alte câmpuri improvizate — iar noi le afișăm pe pagina produsului, în magazin și în aplicație.

Cele două liste sunt diferite

`allergens` = ce **conține** produsul ca ingredient. `allergen_traces` = ce **poate conține ca urme**, din contaminare încrucișată. Le afișăm separat, exact cum cere legea, iar filtrele „fără gluten” / „fără lactoză” exclud produsul dacă alergenul apare în *oricare* dintre cele două liste.

Același cod nu poate apărea în ambele liste și nu se repetă în aceeași listă: altfel setul poate rămâne gol sau incomplet, fără eroare. După trimitere, verifică cu `GET /products` .

Codurile acceptate

Cei 14 alergeni din Anexa II a regulamentului. Trimiți codul, nu textul:

COD	ALERGEN	COD	ALERGEN
`gluten`	Cereale care conțin gluten	`telina`	Țelină
`crustacee`	Crustacee	`mustar`	Muștar
`oua`	Ouă	`susan`	Semințe de susan
`peste`	Pește	`sulfiti`	Dioxid de sulf și sulfiți
`arahide`	Arahide	`lupin`	Lupin
`soia`	Soia	`moluste`	Moluște
`lapte`	Lapte (inclusiv lactoză)	`nuci`	Fructe cu coajă lemnoasă

Un cod pe care nu îl recunoaștem oprește doar produsul respectiv, nu tot lotul, și primești în răspuns exact ce cod e greșit și lista celor valide.

Valorile nutriționale

Obiectul `nutrition`, cu valorile raportate **la 100 g sau 100 ml** (ca pe etichetă). Toate câmpurile sunt opționale — trimiți doar ce ai:

CÂMP	UNITATE	CÂMP	UNITATE
`energy_kcal`	kcal	`sugars_g`	g
`energy_kj`	kJ	`fibre_g`	g
`fat_g`	g	`protein_g`	g
`saturated_fat_g`	g	`salt_g`	g
`carbohydrates_g`	g	`serving_size`	text, ex. „porție 385 g”

Exemplu complet

Atenție: exemplul de mai jos publică un produs real (nu există ștergere prin API). Pentru probă pune `"is_active": false` și un SKU `TEST-…`. Cererea din testul de integrare, cu SKU și `is_active` de probă, și răspunsul primit:

```
PUT /api/v1/partner/products
Authorization: Bearer mhk_cheia_ta
Accept: application/json
Content-Type: application/json

{
  "products": [
    {
      "sku": "TEST-SANDWICH-PUI",
      "is_active": false,
      "name": "Sandwich cu pui",
      "description": "Sandwich cu piept de pui, salată și roșii, pe pâine integrală.",
      "price": 20,
      "stock_quantity": 30,
      "stock_status": "instock",
      "weight": 0.210,
      "ingredients": "Pâine, piept de pui, salată, roșii, maioneză, unt.",
      "allergens": ["gluten", "oua", "lapte", "mustar"],
      "allergen_traces": ["soia", "susan"],
      "nutrition": {
        "energy_kcal": 214, "fat_g": 8.9, "saturated_fat_g": 2.6,
        "carbohydrates_g": 21.4, "sugars_g": 2.8, "fibre_g": 2.9,
        "protein_g": 12.6, "salt_g": 1.05, "serving_size": "porție 210 g"
      }
    }
  ]
}
```

```
200 OK
{"summary":{"total":1,"ok":1,"failed":0},
 "results":[{"index":0,"sku":"TEST-SANDWICH-PUI","result":"created",
             "product_id":"801eefd3-e358-4c31-976e-4e6656b9f729"}]}
```

Cum arată pentru client

Produsul apare în magazin și în aplicația mobilă, cu alergenii afișați în două blocuri distincte: „Conține” (declarația de rețetă) și „Poate conține urme de” (avertismentul de contaminare încrucișată). Cu `is_active: false`, produsul nu apare până nu îl publici.

La `GET /api/v1/partner/products` primești înapoi, pentru fiecare SKU, `ingredients`, `allergens`, `allergen_traces` și `nutrition` — ca să poți verifica ce am înregistrat noi.

Actualizare parțială

Regula: câmpul **trimis** înlocuiește complet ce era înainte; câmpul **netrimis** rămâne neatins; o listă goală `[]` șterge alergenii. Excepții: `nutrition` se înlocuiește **întreg** (valorile pe care nu le retrimiți se pierd), iar galeria (`gallery_image_urls`) se **adaugă** la cea existentă, nu o înlocuiește. Pentru restul câmpurilor, `null` e ignorat (excepție: la `PUT` , `allergens: null` și `allergen_traces: null` șterg setul, ca `[]` ; nu trimite `null`): nu poți șterge prin API un preț redus, datele promoției, ingredientele sau dimensiunile (scrie-ne). Un produs nu se șterge prin API; îl retragi cu `is_active: false` sau cu stoc 0. Deci poți trimite doar `{"allergens": ["lapte"]}` pe `PATCH /api/v1/partner/products/{sku}` fără să retrimiți tot produsul.

11 Modul D · Din panou, manual

PE SCURT

Dacă ai puține produse și se schimbă rar, nu ai nevoie de nimic tehnic. Primești un cont, intri din browser și adaugi produsele ca într-un magazin online obișnuit.

Poți începe așa și trece oricând pe unul dintre celelalte moduri — produsele rămân aceleași.

Ce poți face din cont

SECȚIUNE	CE FACI ACOLO
Produse	Adaugi și editezi: denumire, cod, descriere, preț, preț redus, stoc, categorie, imagine principală și galerie, gramaj și dimensiuni, ingrediente.
Comenzi	Vezi comenzile cu produsele tale: client, telefon, oraș, total, status, data plății. Poți atașa factura în PDF.
Compania ta	Date de facturare, email pentru comenzi, telefon, regulile de transport ale magazinului.

CE VEZI ȘI CE NU

Contul îți arată **doar produsele și comenzile tale**. Ceilalți parteneri din marketplace nu îți văd datele, iar tu nu le vezi pe ale lor — separarea e impusă de sistem, nu ține de disciplina cuiva.

CÂND SĂ TRECI MAI DEPARTE

Peste ~30 de produse, sau când stocul începe să se schimbe zilnic, munca manuală devine sursă de erori: prețuri uitate, stoc neactualizat, comenzi pe produse epuizate. Atunci merită modul A sau C.

12 Checklist de integrare

PE SCURT

Parcurgem împreună fazele de mai jos. Fiecare punct este ceva ce se verifică efectiv, cu un apel real — nu se presupune. Trecem la faza următoare doar când cea curentă e bifată integral.

Faza 0 · Pregătire

- [ ] Contract semnat și profil de partener creat
- [ ] Magazinul tău este legat de marketplace (o facem noi — fără asta nu îți vei vedea comenzile)
- [ ] Ai desemnat un contact tehnic și unul operațional
- [ ] Ai primit cheia API; secretul de webhook ți-l configurăm după ce ne dai adresa `https` a webhook-ului

Faza 1 · Produse

- [ ] Prima probă: un `GET /products` (răspunde `{"data":[]}` și confirmă cheia), apoi un produs cu SKU `TEST-…` și `"is_active": false`. Nu există mediu de test separat, iar un produs nu se șterge prin API: fără `is_active: false`, produsul apare în magazinul public → răspuns `created`
- [ ] Același SKU retrimis cu alt preț → răspuns `updated`, fără dublură în magazin
- [ ] Un produs cu un câmp lipsă → `result: error` pentru acel produs, cu câmpurile de reparat (HTTP 200 dacă restul lotului a intrat)
- [ ] Imaginile apar corect pe produs
- [ ] Set pilot de 10–20 SKU încărcat și validat vizual în magazin

Faza 2 · Stoc

- [ ] Modificare de stoc → se vede în magazin
- [ ] Stoc pus pe zero → produsul nu mai poate fi comandat
- [ ] Frecvența de actualizare agreată în scris

Faza 3 · Comenzi

- [ ] Comandă de test plasată de noi cu produsele pilot
- [ ] Notificarea ajunge la webhook-ul tău
- [ ] Semnătura validată de sistemul tău
- [ ] Comanda apare și la `GET /orders`
- [ ] Schimbi statusul în `completed` → se vede la noi
- [ ] Anulare agreată: `cancelled` nu returnează banii; rambursarea o facem noi, deci ne scrii înainte să anulezi o comandă plătită

Faza 4 · Go live

- [ ] Cheia API păstrată în siguranță (nu în cod public) și secretul de webhook configurat
- [ ] Catalog complet încărcat, număr de produse confirmat de ambele părți
- [ ] O comandă reală urmărită cap-coadă în prima zi
- [ ] Contacte de suport schimbate în ambele sensuri
- [ ] Monitorizare atentă în primele două săptămâni

Ce ne trimiți ca să pornim

INFORMAȚIE	EXEMPLU
Contact tehnic	nume, email
Contact operațional pentru comenzi	nume, telefon
Adresa de webhook	`https://…/mommyhai/comenzi`
Email de rezervă pentru comenzi	—
Cât de des se schimbă stocul	„la fiecare vânzare” / „de 2 ori pe zi”
Orașele în care livrezi	național / listă
Termen de confirmare și livrare	—
Set pilot	10–20 SKU

Întrebări tehnice: hai@mommyhai.com — răspundem în aceeași zi lucrătoare.