

Picsha Fulfillment API

Partner Integration Specification — Draft v1.0 (for review)

July 28, 2026 · Picsha, a Graphx company · Contact: Pete Traversy — petert@graphx.com

This document is a draft specification shared for technical review with platform partners. Endpoint names, fields, and payloads are stable in intent but open to feedback before the spec is frozen. Please direct comments to the contact above.

1. Overview

Picsha operates white-label AI fulfillment services for photo retailers:

- **Picsha Retouch** — AI photo restoration and retouching (dust and scratch removal, tear repair, fade and stain correction, colorization, red-eye).
- **Picsha Books** — AI-designed photo books delivered as print-ready PDFs.

The Fulfillment API gives platform partners (kiosk, web-to-print, and retail e-commerce systems) two integration surfaces:

Part A — Order Submission (partner → Picsha). Your platform submits photos and product options to Picsha; Picsha performs the AI work and returns finished, print-ready output (high-resolution images today; AI-designed book PDFs on the v1.1 roadmap). Use this to offer Picsha services as products inside your own ordering flow.

Part B — Fulfillment Feed (Picsha → partner). Orders placed on a retailer's branded Picsha storefront are pushed to your platform via webhook, with time-limited download URLs for the print-ready assets — so physical products (prints, framed prints, books) flow directly into the retailer's existing production queue instead of being downloaded manually.

A partner may implement either surface or both. Both share the same authentication, webhook mechanics, and conventions.

Design principles

- Plain REST + JSON over HTTPS. No SDKs required.
- Every order is scoped to a store (a retailer), matching Picsha's existing multi-tenant model.
- All processing is asynchronous: submit, then receive a webhook (or poll).
- Finished output is delivered as time-limited presigned URLs — no shared storage credentials.
- All monetary amounts, where present, are integer cents (USD).

2. Environments and versioning

Environment	Base URL
Sandbox	https://api-sandbox.picsha.com/fulfillment/v1
Production	https://api.picsha.com/fulfillment/v1

The version lives in the path. Additive changes (new optional fields, new event types) do not bump the version; breaking changes ship as /v2.

3. Authentication

- Each platform partner is issued an API key per environment: `Authorization: Bearer pfk_live_...` (or `pfk_test_...`).
- Every request is scoped to a store via `storeId` — the retailer's Picsha store identifier (the same slug as their storefront subdomain, e.g. `theprintrefineryky`). A partner key is entitled to an explicit list of stores; requests for other stores return 403.
- Webhook deliveries are signed with a per-partner secret (see §8).
- Store provisioning (creating a retailer's Picsha store, branding, catalog, pricing) remains a Picsha-side operation in v1; a provisioning API can follow if partners need it.

4. Core resources and statuses

Resource	Description
Store	A retailer. Identified by <code>storeId</code> .
Asset	An uploaded customer photo, referenced by <code>assetId</code> .
Order	A unit of fulfillment work, referenced by <code>orderId</code> .
Output	A finished, print-ready file (image or PDF) attached to a ready order.

Order lifecycle (both surfaces):

```
received → processing → ready
                        ↘ failed
```

`failed` is terminal and always carries a machine-readable `failureReason`.

5. Part A — Submitting orders to Picsha

5.1 Uploading photos

Option 1 — presigned upload (recommended).

```
POST /assets/presign
{ "storeId": "theprintrefineryky", "filename": "scan-001.jpg",
  "contentType": "image/jpeg" }

→ 200
{ "assetId": "ast_9f1c2ab4", "uploadUrl": "https://...",
  "expiresAt": "2026-07-28T18:00:00Z" }
```

PUT the file bytes to `uploadUrl`. Accepted: JPEG, PNG, TIFF, HEIC, up to 30 MB. The asset is usable in orders once the PUT completes.

Option 2 — fetch by URL.

```
POST /assets
{ "storeId": "theprintrefineryky",
  "sourceUrl": "https://partner.example.com/tmp/scan-001.jpg?sig=..." }

→ 201 { "assetId": "ast_9f1c2ab4" }
```

The URL must be HTTPS and reachable by Picsha at submission time (expiring signed URLs are fine). Assets are retained 30 days, then purged.

5.2 Retouch orders

```
POST /orders
Idempotency-Key: 7c1e4c9a-...

{
  "storeId": "theprintrefineryky",
  "product": "retouch",
  "externalRef": "PF-118742",
  "assets": ["ast_9f1c2ab4"],
  "options": { "addons": ["colorize", "redeye"] },
  "webhookUrl": "https://partner.example.com/hooks/picsha"
}

→ 202 { "orderId": "ord_6f39eb74", "status": "received" }
```

- **Baseline restoration is always applied:** dust and scratch removal, tear repair, fade correction, stain removal.
- `options.addons` (optional): colorize (black-and-white to color), redeye.
- `externalRef` is your platform's own order/line identifier; it is echoed on every status response and webhook.
- `webhookUrl` overrides the partner's default webhook endpoint for this order (optional).

Typical processing time is one to two minutes per photo. Output is a full-resolution retouched image (up to 4K on the long side, original aspect ratio preserved), delivered as JPEG by default.

5.3 Book orders (v1.1 roadmap)

Planned for v1.1: submit 20–200 photos, and Picsha's book engine analyzes the images, selects and sequences them, chooses a design collection, and returns a print-ready PDF plus a machine-readable plan (title, chapters, page layouts).

```
POST /orders
{ "storeId": "...", "product": "book", "externalRef": "...",
  "assets": ["ast_...", "ast_...", ...],
  "options": { "sizeId": "12x12", "style": "auto" } }
```

This mode requires Picsha's server-side book renderer, which is in development. It is included here so partners can plan for it; the v1 launch surface is Retouch submission (§5.2) plus the Fulfillment Feed (§6), which already carries finished book PDFs.

5.4 Order status and output

```
GET /orders/ord_6f39eb74

→ 200
{
```

```

"orderId": "ord_6f39eb74",
"externalRef": "PF-118742",
"storeId": "theprintrefineryky",
"product": "retouch",
"status": "ready",
"createdAt": "2026-07-28T17:02:11Z",
"completedAt": "2026-07-28T17:03:29Z",
"outputs": [
  { "type": "image", "format": "jpeg",
    "width": 3840, "height": 2887,
    "downloadUrl": "https://...presigned...",
    "urlExpiresAt": "2026-07-28T18:03:29Z" }
]
}

```

downloadUrl is a presigned URL valid for 60 minutes. Re-fetch the order at any time for a fresh URL; outputs are retained for 90 days. Polling is supported, but webhooks (§8) are strongly recommended.

6. Part B — Receiving Picsha storefront orders (Fulfillment Feed)

When a retailer's customers order physical products on the retailer's Picsha storefront, those orders can be routed to the platform that runs the retailer's production. Two mechanisms, usable together:

6.1 Webhook push

On payment and completion of AI processing, Picsha sends order.ready to the partner's registered endpoint:

```

{
  "event": "order.ready",
  "orderId": "ord_c81a02d4",
  "storeId": "theprintrefineryky",
  "source": "storefront",
  "product": "retouch",
  "createdAt": "2026-07-28T16:40:03Z",
  "customer": { "name": "J. Smith", "email": "customer@example.com" },
  "lines": [
    { "type": "print", "label": "8×10 Print", "minInches": 8, "qty": 2 },
    { "type": "frame", "label": "Walnut Frame", "widthInches": 1.25 },
    { "type": "mat", "label": "White Mat", "widthInches": 2 }
  ],
  "outputs": [
    { "type": "image", "format": "jpeg", "width": 3840, "height": 3072,
      "downloadUrl": "https://...presigned...", "urlExpiresAt": "..." }
  ]
}

```

For Picsha Books storefront orders, product is "book", lines carry the book size and page count, and the output is a print-ready PDF:

```

"lines": [
  { "type": "book", "label": "12×12 Photo Book", "sizeId": "12x12",
    "widthInches": 12, "heightInches": 12, "pageCount": 28 } ],
"outputs": [
  { "type": "pdf", "pages": 28, "downloadUrl": "https://...",
    "urlExpiresAt": "..." } ]

```

- Print sizes are specified by minimum (short-side) inches; the long side follows the photo's aspect ratio. The catalog of print sizes, mounts, frames, and mats is configured per store.
- Digital-only lines (customer downloads) are fulfilled by Picsha directly and are not pushed to the feed unless requested.

- Customer contact fields are included only where the retailer's production workflow requires them, per the data terms in §10.

6.2 Pull queue

```
GET /stores/theprintrefineryky/fulfillment-queue?status=ready
```

```
→ 200 { "orders": [ ...same shape as the webhook payload... ] }
```

Use this for reconciliation, recovery after webhook downtime, or as a simple polling integration if the partner platform cannot receive webhooks.

6.3 Status round-trip (optional)

Partners may report production progress back so the retailer and customer see status in one place:

```
POST /orders/ord_c81a02d4/fulfillment-status
{ "status": "in_production" } // in_production | shipped | completed | problem
```

Optional in v1; strongly encouraged.

7. Errors, idempotency, rate limits

- Errors use conventional HTTP codes with a consistent body: { "error": { "code": "invalid_asset", "message": "..." } }
- POST /orders accepts an Idempotency-Key header; retrying with the same key returns the original result rather than creating a duplicate order.
- Default rate limits: 60 requests/minute per key for order submission, 600/minute for status reads. Higher limits available on request.

8. Webhook mechanics

- Partner registers a default webhook URL per environment (per-order override available on submission).
- Every delivery includes X-Picsha-Signature: t=<unix-ts>,v1=<hex> — an HMAC-SHA256 of "<t>.<raw body>" using the partner's webhook secret. Reject deliveries with a bad signature or a timestamp older than 5 minutes.
- Events in v1: order.processing, order.ready, order.failed.
- Deliveries are retried with exponential backoff for up to 24 hours; a 2xx response acknowledges. The pull queue (§6.2) covers anything missed.

9. Commercial terms (outside the API)

Wholesale fulfillment pricing, settlement, and retailer billing are commercial terms agreed between Picsha, the retailer, and (where applicable) the platform partner; they are intentionally not encoded in the v1 API. Order payloads carry no prices in Part A; Part B payloads reflect what the customer purchased on the storefront. A fees echo can be added in a later revision if partners want it.

10. Data handling

- Customer photos and finished outputs are stored encrypted at rest in AWS (us-east-1); uploads are virus-scanned at ingest.
- Source assets are retained 30 days; finished outputs 90 days; both are then purged.
- Customer PII is limited to what the order requires (name/email where the retailer's workflow needs it) and is never used for anything other than fulfilling the order.

11. Draft status and next steps

This is a draft for technical review. We are gathering feedback from platform partners now; the spec will be frozen within two weeks of consolidated feedback, with sandbox credentials issued to committed partners approximately 30 days after spec freeze. Part B (Fulfillment Feed) and Part A Retouch submission are the launch surface; Part A Books follows in v1.1.

We are equally glad to integrate through a partner platform's existing fulfiller/lab program where one exists — whichever path is fastest for our mutual retailers.

Contact: Pete Traversy · petert@graphx.com · Picsha, a Graphx company

Appendix A — Enumerations

Field	Values
Order status	received, processing, ready, failed
Retouch addons	colorize, redevye
Baseline restoration (always on)	dust/scratch removal, tear repair, fade correction, stain removal
Line type (feed)	retouch, download, print, mount, frame, mat, book, pages, discount
Output type	image, pdf
Fulfillment status (round-trip)	in_production, shipped, completed, problem

Appendix B — Example book size catalog

Sizes are configured per store; a representative default catalog:

sizeId	Dimensions	Included pages	Extra pages
5x5	5 × 5 in	20	per-page price
6x6	6 × 6 in	20	per-page price
8x8	8 × 8 in	20	per-page price
8.5x11	8.5 × 11 in	20	per-page price
11x8.5	11 × 8.5 in (landscape)	20	per-page price
12x12	12 × 12 in	20	per-page price