

Postally API Reference

Version 1.71 | May 1, 2026 | <https://api.postally.ca>

1. Introduction

Postally is a direct mail platform that lets you send postcards through a REST API. Letters and self-mailers are coming soon. Design your mail pieces with HTML templates or uploaded PDFs, manage recipient lists, and track delivery status — all programmatically.

The API is organized around REST principles. All requests and responses use JSON. All endpoints are prefixed with `/v1/` for API versioning.

1.1 Base URL

Production: `https://api.postally.ca`

Development: `http://localhost:3000`

1.2 Quick Start

1. Obtain your API key from the Postally dashboard (Settings > API Keys).
2. Use your test key (`dm_test_...`) for development — no real mail is sent and no charges apply.
3. Switch to your live key (`dm_live_...`) when ready for production.

1.3 Supported Mail Types

Field	Type	Required	Description
<code>postcard</code>	4x6, 5x7, 6x9, 6x11	—	Single or double-sided postcards (v2 full-bleed spec — see section 22.3)
<code>letter</code>	8.5x11 standard	—	Letters with optional envelopes, buckslips, return envelopes
<code>self_mailer</code>	6x18, 7x10, 8.5x11, 9x12	—	Tri-fold or bi-fold self-mailers (direct send or via campaigns). Currently coming soon in production.

2. Authentication

Every API request (except public endpoints) must include an API key. Postally accepts two authentication schemes — both work identically. Bearer is the canonical form documented throughout the rest of this reference; HTTP Basic Auth is supported for legacy CLI tooling that uses `curl -u key:` shorthand.

2.1 Bearer Token (canonical)

Pass the key in the Authorization header. This is the form used in every example throughout this reference:

```
curl -H "Authorization: Bearer dm_live_abc123" https://api.postally.ca/v1/users/me
```

2.2 HTTP Basic Auth (legacy)

For legacy tooling, you may pass the API key as the username with an empty password. The request is converted internally to a Bearer-equivalent — there is no functional difference.

```
curl -u "dm_live_abc123:" https://api.postally.ca/v1/users/me
```

2.3 API Key Environments

Field	Type	Required	Description
dm_test_*	Test	—	No real mail sent, no charges. Use for development and testing.
dm_live_*	Live	—	Real mail is printed and delivered. Credits are charged.

2.4 Dashboard Session Tokens

Dashboard users authenticate via POST `/v1/auth/login` which returns a 64-character hex session token. Pass it as a Bearer token for subsequent requests.

2.5 Permissions & PHI Access

API keys carry a permissions array (e.g., ["*"] for full access or specific permissions like ["recipients.read", "send.postcard"]). Accounts flagged as PHI-regulated also enforce PHI access levels:

Field	Type	Required	Description
phi_access_none	Level 0	—	No access to protected health information
phi_access_address	Level 1	—	Can access recipient addresses only
phi_access_content	Level 2	—	Can access mail content containing PHI
phi_access_full	Level 3	—	Full PHI access including audit trails

2.6 Public Endpoints (No Auth Required)

The following endpoints do not require API key authentication:

GET /health, /health/live	— Health checks
GET /qr/:code	— QR scan redirect
POST /v1/orders/quote ce mail-class policy)	— Price quote (advisory – does not enforce mail-class policy)
POST /v1/pricing/quote -class policy)	— Campaign price estimate (enforces mail-class policy)
GET /v1/policy/mail-class ation text	— Canada Post policy constants + attestation text
POST /webhooks/stripe	— Stripe webhook (signature-verified)
POST /v1/triggers/inbound/:id	— Trigger webhook (secret-verified)
POST /v1/auth/login, /signup, /forgot-password, /reset-password	
GET /docs, /v1/docs	— Documentation

GET /v1/auth/verify-email

– Email verification callback

Note: GET /v1/specs and GET /v1/specs/templates DO require authentication. Use GET /v1/policy/mail-class for the policy constants that dashboards and integrators need without an API key.

2.7 Email Verification

New accounts must verify their email address before sending mail. On signup, Postally validates the email domain (MX record check) and sends a verification link that expires in 24 hours.

What is blocked until verified:

Creating mailpieces (postcards, letters, self-mailers, checks)

Creating or approving campaigns

Creating orders

Firing webhook triggers

What is allowed while unverified:

Login, browse dashboard, create templates, manage recipients, view pricing

To resend the verification email: POST /v1/auth/resend-verification (authenticated, max 3 per hour).

3. Conventions & Common Patterns

3.1 Data Formats

Field	Type	Required	Description
IDs	UUID	—	All resource identifiers are UUID v4 strings
Timestamps	ISO 8601	—	All timestamps are UTC with timezone (TIMESTAMPTZ)
Money	Integer cents	—	All monetary values in cents. Currency is CAD.
Strings	UTF-8	—	All text is UTF-8 encoded

3.2 Pagination

List endpoints support pagination via query parameters:

Field	Type	Required	Description
limit	integer	No	Results per page. Min 1, max 100, default 20.
offset	integer	No	Number of results to skip. Default 0.

Paginated responses include a pagination object:

```
{ "data": [...], "pagination": { "total": 150, "limit": 20, "offset": 0 } }
```

3.3 Rate Limiting

The API enforces a 4-tier rate limiting system per API key. The tier is determined by the endpoint:

Tier	Limit	Required	Description
READ	120 req / 60s	—	GET endpoints (list, get, balance, etc.)
STANDARD	60 req / 60s	—	Default tier for most endpoints
WRITE	30 req / 60s	—	POST create/update endpoints (recipients, templates, sends, etc.)
EXPENSIVE	10 req / 60s	—	Batch operations, campaign approvals, bulk imports

Rate limit status is returned in response headers:

Field	Type	Required	Description
x-ratelimit-limit	integer	—	Maximum requests allowed in window
x-ratelimit-remaining	integer	—	Requests remaining in current window
retry-after	integer	—	Seconds to wait before retrying (only on 429)

3.4 Idempotency

To safely retry POST requests without creating duplicates, include an Idempotency-Key header:

```
curl -H "Idempotency-Key: unique-request-id-123" -X POST ...
```

Keys are cached for 24 hours. Reusing a key with a different request body returns HTTP 422. Cached responses include the header Idempotent-Replayed: true.

3.5 Request ID

Every response includes an X-Request-Id header. You can also send your own for correlation. Useful for debugging with Postally support.

3.6 Test Mode

Use a test API key (dm_test_...) or pass test=true in the request body. In test mode: no real mail is sent, no credits are charged, mailpieces are created with is_test=true.

4. Error Handling

Errors return a JSON object with an error code, human-readable message, and optional details array:

```
{ "error": "validation_error", "message": "Request validation failed",  
  "details": [{ "path": ["address_line1"], "message": "Required" }] }
```

4.1 Error Codes

Code	Error	Required	Description
400	validation_error	—	Request body or query params failed Zod validation. Check details array.
400	mail_class_policy_violation	—	Canada Post Personalized Mail policy rejection: piece count below the 125-piece floor, or non-promotional content_category. Response includes suggested_mail_class = 'first_class' as the safe fallback.
400	attestation_required	—	mail_class='standard' was requested but attestation_accepted was not set to true. Customer must confirm the Personalized Mail attestation (GET /v1/policy/mail-class returns the canonical text).

Code	Error	Required	Description
400	baa_required	—	BAA must be on file before creating PHI recipients on PHI-regulated accounts.
400	empty_group	—	Campaign group contains no recipients.
401	authentication_required	—	No API key provided in request.
401	invalid_api_key	—	API key not found, revoked, or expired.
402	insufficient_credits	—	Prepaid credit balance too low. Top up first.
403	insufficient_permissions	—	API key lacks the required permission scope.
403	insufficient_phi_access	—	PHI access level too low for this operation.
403	account_inactive	—	Account has been suspended or deactivated.
404	not_found	—	Resource does not exist or belongs to a different account.
409	cancel_window_expired	—	Mailpiece is past the cancellation deadline.
409	conflict	—	Duplicate resource or invalid state transition.
422	idempotency_mismatch	—	Same Idempotency-Key used with a different request body.
429	rate_limit_exceeded	—	Too many requests. Check retry-after header.
503	service_unavailable	—	An external service (S3, SES) is

Code	Error	Required	Description
			temporarily unavailable.

5. Recipients

Recipients represent the people or organizations receiving your mail. All personally identifiable information (PII) is encrypted at rest using AES-256-GCM.

5.1 Create Recipient

POST /v1/recipients/new

Note: Validation: required address fields (address_line1, city, postal_code) cannot be whitespace-only. retention_expires_at, when provided, must be a future ISO datetime. tags array is capped at 50 entries. Same rules apply to /v1/recipients/update and /v1/recipients/import.

Field	Type	Required	Description
first_name	string	No	Max 255 characters
last_name	string	No	Max 255 characters
company	string	No	Max 255 characters
address_line1	string	Yes	Max 500 characters
address_line2	string	No	Max 500 characters
city	string	Yes	Max 255 characters
state	string	No	Max 100 characters (province/state code)
postal_code	string	Yes	Max 20 characters
country	string	No	2-3 character ISO code. Default: "CA"
email	string	No	Valid email address, max 255 characters
phone	string	No	Max 30 characters
return_first_name	string	No	Optional sender first name for return address on this recipient's mailpieces. Max 255 characters.

Field	Type	Required	Description
return_last_name	string	No	Optional sender last name for return address. Max 255 characters.
return_company	string	No	Optional sender company for return address. Max 255 characters.
return_address_line 1	string	No	Return address line 1. Max 500 characters. If set (with the required return fields below), this return address is used instead of the account default.
return_address_line 2	string	No	Return address line 2. Max 500 characters.
return_address_line 3	string	No	Return address line 3. Max 500 characters.
return_city	string	No	Return city. Max 255 characters. Required if any return_* field is set.
return_state	string	No	Return province/state. Max 100 characters.
return_postal_code	string	No	Return postal code. Max 20 characters. Required if any return_* field is set.
return_country	string	No	Return country, 2- or 3-char ISO code (e.g. CA, USA). Required if any return_* field is set. Self-mailers require return_country=CA.

Field	Type	Required	Description
data_region	enum	No	us ca eu uk au other. Default: "ca". Controls data residency region for GDPR/PIPEDA compliance.
lawful_basis	enum	No	consent contract legal_obligation vital_interests public_task legitimate_interests. Default: "contract"
is_phi	boolean	No	Mark as Protected Health Information (PHI). Default: false
metadata	object	No	Arbitrary key-value pairs for your use
tags	string[]	No	Array of tag strings
retention_expires_at	datetime	No	ISO 8601. Auto-delete recipient after this date.

Note: Return address resolution: at dispatch time, Postally uses the recipient's return_* fields if a complete return address is present (return_address_line1, return_city, return_postal_code, return_country all set). Otherwise, the account-level return address from Settings is used as a transitional fallback. This per-recipient return address is the source of truth going forward — CSV imports should map return_* fields directly, and bulk campaigns with mixed return addresses are fully supported.

Example Request:

```
curl -H "Authorization: Bearer dm_test_abc123" -X POST https://api.postally.ca/v1/recipient/new \
```

```
-H "Content-Type: application/json" \
```

```
-d '{"first_name":"Jane","last_name":"Smith","address_line1":"100 King St W",
```

```
"city":"Toronto","state":"ON","postal_code":"M5X 1A9","country":"CA"}
```

5.2 Get Recipient

GET /v1/recipients/get/:id

Returns the full recipient object with decrypted PII fields.

5.3 List Recipients

GET /v1/recipients/list

Field	Type	Required	Description
limit	integer	No	1-100, default 20
offset	integer	No	Default 0
group_id	uuid	No	Filter by recipient group
suppressed	boolean	No	Filter by suppression status

5.4 Update Recipient

POST /v1/recipients/update

Partial update. Include id and only the fields you want to change.

5.5 Delete Recipient

POST /v1/recipients/delete

Soft delete. Body: { "id": "uuid" }. Returns: { "id": "uuid", "deleted": true }

5.6 Suppress Recipient

POST /v1/recipients/suppress

Marks a recipient as suppressed (Do Not Contact). Suppressed recipients cannot receive mail.

Body: { "id": "uuid", "reason": "unsubscribed" }

5.7 Bulk Import

POST /v1/recipients/import

Import up to 10,000 recipients in a single request. Optionally assign all to a group.

```
Body: { "recipients": [{...}, ...], "group_id": "uuid" }
```

```
Response: { "created": 95, "failed": 5, "errors": [{ "index": 3, "issues": [...]}] }
```

Each recipient row supports all fields from §5.1 Create Recipient, including the return_* fields. When importing from CSV, map columns like return_street, return_zip, sender_company etc. to the return_* recipient fields — this lets you ship the same campaign with different return addresses per recipient (useful for multi-brand merchants or agency-managed accounts).

6. Recipient Groups

Groups organize recipients for campaigns and bulk operations.

6.1 Create Group

POST /v1/groups/create

Field	Type	Required	Description
name	string	Yes	1-255 characters
description	string	No	Max 1000 characters
metadata	object	No	Arbitrary key-value pairs

6.2 List Groups

GET /v1/groups/list

Returns groups with recipient_count for each.

6.3 Get Group

GET /v1/groups/get/:id

6.4 Delete Group

POST /v1/groups/delete

Body: { "id": "uuid" }. Does not delete the recipients in the group.

Note: Returns 404 not_found for group ids that don't exist OR belong to another account. 409 group_in_use only fires for caller-owned groups; foreign-account references are never disclosed.

7. Templates

Templates define the design and content of your mail pieces. Use HTML with merge variables or upload PDF files.

7.1 Create Template

POST /v1/templates/create

Note: In production-like environments (NODE_ENV=production or staging), formats other than postcard_* return 400 type_disabled at create time — letters, self-mailers, and checks are coming soon. The same gate also fires on /v1/templates/update (when format is patched) and /v1/designs/save.

Note: Field caps: content is limited to 256 KB. Larger HTML content is rejected with 400 validation_error 'content exceeds 256 KB cap'.

Field	Type	Required	Description
name	string	Yes	1-255 characters
type	enum	Yes	postcard letter self_mailer check. Only postcards are active; others are coming soon.
format	enum	Yes	postcard_4x6 postcard_5x7 postcard_6x9 postcard_6x11 letter_standard self_mailer_6x18 self_mailer_8.5x11 self_mailer_7x10 self_mailer_9x12 check_standard
content	string	No	HTML content with {{variable}} merge syntax

Field	Type	Required	Description
front_pdf_url	url	No	URL to front PDF (from /v1/uploads/presign)
back_pdf_url	url	No	URL to back PDF
variables	string[]	No	List of merge variable names used in content
metadata	object	No	Arbitrary key-value pairs

7.2 Merge Variables

Use double curly braces in template content: `{{first_name}}`, `{{company}}`, `{{custom_field}}`. At send time, pass values via `merge_data` to replace variables.

7.3 List / Get / Update / Publish / Unpublish / Delete

GET /v1/templates/list?type=postcard

GET /v1/templates/get/:id

GET /v1/templates/:id

Note: REST-style alias for /v1/templates/get/:id — both return identical responses.

POST /v1/templates/update

POST /v1/templates/publish

Publish: { "id": "uuid", "published": true }

Note: Returns 422 design_required when published=true on a template that has neither a front_pdf_url nor any saved builder design — campaigns against an empty template would fail at render time.

POST /v1/templates/unpublish

Unpublish: { "id": "uuid" }

Note: Returns 409 template_in_use if any non-terminal campaign still references the template. Equivalent to POST /v1/templates/publish with published=false.

Note: Returns 404 not_found if the template id doesn't exist OR is owned by another account. The existence of foreign-account assets is not disclosed (the same 404 covers both cases).

POST /v1/templates/delete

Note: Returns 404 not_found for foreign-account template ids. Delete responses never include reference counts for foreign-account assets.

7.4 Preview Render

POST /v1/designs/preview-render

Render a design against a specific recipient WITHOUT creating a mailpiece. Returns a PNG preview image that shows exactly what would ship, including merge variables substituted from the recipient, address block, return address block, and postage indicia positioned per the piece configuration. Use this to inspect how edge-case recipients (long names, long addresses, missing return address) will actually render before sending.

Field	Type	Required	Description
template_id	uuid	Conditional	Template to render. Either template_id or design_id is required.
design_id	uuid	Conditional	Specific design (front or back) from the template builder. Either template_id or design_id is required.
recipient_id	uuid	Conditional	Saved recipient to render against. Either recipient_id or recipient is required.
recipient	object	Conditional	Inline recipient (same shape as POST /v1/postcards/create recipient). Either recipient_id or recipient is required.
piece_config	object	No	Override piece-level config: {

Field	Type	Required	Description
			return_address_mode, postal_zone_background, back_barcode_background, paper_finish }. Defaults to template preview defaults.

Response (200):

Returns image/png body with the rendered preview. Output is capped at 2 megapixels (~2000x1000) to bound bandwidth.

Error responses:

Status	Meaning
401	Missing or invalid API key
403	PHI access required (recipient is PHI-flagged and caller lacks phi_access_full)
404	Template or recipient not found
422	Content moderation failed (design contains rejected content)
429	Rate limit exceeded (10 preview-renders per minute per account)
503	Content moderation unavailable (Rekognition down) — preview is not rendered without moderation

Note: Compliance checks are NOT run on preview renders. Use this endpoint for visual QA only; actual proof generation and dispatch still run all 13 PDF compliance checks. Every preview call is audit-logged with action=design.preview_render.

7.5 Save Design (Builder)

POST /v1/designs/save

Used by the dashboard template builder to save canvas-edited designs. Accepts an optional template_id to attach the design to a specific template.

Note: Returns 422 type_format_mismatch when template_id is provided AND the design's type or format does not match the linked template's. The error message includes both pairs (design's vs template's) so callers can self-correct.

8. Envelopes

POST /v1/envelopes/create

Field	Type	Required	Description
name	string	Yes	1-255 characters
type	enum	Yes	outer return
size	enum	Yes	number_10_standard number_10_window number_9_reply 6x9_booklet 9x12_flat
front_pdf_url	url	Yes	Front design PDF
back_pdf_url	url	No	Back design PDF
has_window	boolean	No	Default: false
window_position	object	No	{ x, y, width, height } in points
return_address_printed	boolean	No	Default: true
postage_type	enum	No	none prepaid_first_class business_reply. Default: none
metadata	object	No	Arbitrary key-value pairs

GET /v1/envelopes/list

GET /v1/envelopes/get/:id

POST /v1/envelopes/delete

Note: Returns 404 not_found if the envelope id doesn't exist OR is owned by another account. 409 envelope_in_use only fires for assets the caller owns; foreign-account ids never expose conflict counts.

9. Buckslips

Buckslips are small insert cards included in letter envelopes.

POST /v1/buckslips/create

Field	Type	Required	Description
name	string	Yes	1-255 characters
size	enum	No	3.5x8.5 4x6 5x7 3.5x6. Default: 3.5x8.5
front_pdf_url	url	Yes	Front design PDF
back_pdf_url	url	No	Back design PDF
metadata	object	No	Arbitrary key-value pairs

GET /v1/buckslips/list

GET /v1/buckslips/get/:id

POST /v1/buckslips/delete

Note: Returns 404 not_found for bucksliip ids that don't exist OR belong to another account. 409 bucksliip_in_use only fires for caller-owned assets.

10. Address Verification

Postally uses Smarty for address validation and autocomplete. Addresses are automatically verified before mail is sent.

10.1 Validate Address

POST /v1/addresses/validate

Field	Type	Required	Description
address_line1	string	Yes	Max 500 characters
address_line2	string	No	Max 500 characters
city	string	Yes	Max 255 characters
state	string	No	Max 100 characters
postal_code	string	Yes	Max 20 characters
country	string	No	2-3 char ISO code. Default: "CA"

Returns: { "valid": true, "address": { normalized fields }, "zip_code": "...", "latitude": ..., "longitude": ... }

10.2 Autocomplete

GET /v1/addresses/autocomplete?q=100+King&country=CA

Query parameter q must be at least 3 characters. Returns an array of address suggestions.

Note: When the upstream provider is unavailable, returns { "suggestions": [], "error": "service_unavailable", "message": "Address autocomplete is temporarily unavailable." } instead of an HTTP error. The raw upstream status is never leaked in the response body.

10.3 Verify Stored Recipient

POST /v1/addresses/verify-recipient

Body: { "recipient_id": "uuid" }. Verifies and updates the stored address for the recipient.

10.4 Data Quality Check (pre-campaign verification)

POST /v1/data-verification/check

Runs 14 data quality checks against a list of recipients (or a saved group) BEFORE creating a campaign. Returns per-check severity (pass/warning/blocker), count of affected recipients, sample recipient IDs for inline preview rendering, and a summary with blockers/warnings counts plus can_proceed boolean.

Field	Type	Required	Description
type	enum	No	postcard letter self_mailer check. Affects which checks apply (notably: self_mailer makes missing/non-CA return address a BLOCKER instead of a warning).
group_id	uuid	Conditional	Saved group to verify. Either group_id or recipients is required.
recipients	array	Conditional	Inline recipient list. Either group_id or recipients is required.

Response (200):

```
{ "total_recipients": 500,  
  "summary": { "blockers": 0, "warnings": 3, "passed": 11, "can_proceed": true },  
  "checks": [  
    { "id": "missing_address", "severity": "pass", "passed": true, "count": 0,  
      "sample_ids": [], "affected_indices": [] },  
    { "id": "long_full_name", "severity": "warning", "passed": false, "count":  
1,  
      "sample_ids": ["rcp-uuid"], "affected_indices": [42],
```

```

    "details": "1 recipient with name over 40 chars (longest: 50 chars ...)"
  },
  { "id": "missing_return_address", "severity": "warning", "passed": false,
    ... }
]
}

```

All 14 checks:

Check ID	Severity
missing_address	Blocker
missing_city	Blocker
missing_postal_code	Blocker
invalid_country	Blocker
duplicate_addresses	Warning
long_full_name	Warning
long_address	Warning
missing_first_name	Warning
missing_last_name	Warning
missing_company	Warning
piece_count	Informational
missing_return_address (postcard/letter)	Warning
missing_return_address (self_mailer)	Blocker
non_ca_return_on_self_mailer	Blocker

Note: The sample_ids/affected_indices on each failing check let you render the problematic recipient via POST /v1/designs/preview-render to see exactly how the piece will look before deciding to exclude or fix the row.

11. Sending Mail

The send endpoints create individual mail pieces. Each piece goes through address verification, cost calculation, PDF rendering, and enters the production queue.

11.1 Send Postcard

POST /v1/postcards/create

Field	Type	Required	Description
size	enum	No	postcard_4x6 postcard_5x7 postcard_6x9 postcard_6x11. Default: postcard_4x6. The field name `format` is accepted as an alias (same values) for callers migrating from the campaigns or orders API.
template_id	uuid	Yes (unless test=true)	Reference to a published template. Must match the request size and type. Required for all non-test sends — the platform has no bring-your-own-PDF path. Test mode (test: true) is exempt.
recipient_id	uuid	No	Reference to a saved recipient (use this OR inline recipient)
recipient	object	No	Inline recipient: { address_line1, city, postal_code, ... }

Field	Type	Required	Description
merge_data	object	No	Key-value pairs for template {{variables}}
mail_class	enum	No	first_class (Lettermail) standard (Personalized Mail). Auto-selected if omitted: Lettermail is the safe default. Single-send postcards are ALWAYS 1 piece, so 'standard' is rejected with mail_class_policy_violation — Personalized Mail requires the 125-piece floor enforced at campaign or batch level.
content_category	enum	No	promotional transactional general. Default: promotional. Only relevant for bulk/campaign sends — ignored on single-send postcards which are always Lettermail regardless of category.
qr_code_enabled	boolean	No	Generate trackable QR code. Default: false
qr_target_url	url	No	QR code destination URL. Required if qr_code_enabled.
qr_zone	object	No	QR placement coordinates for customer-uploaded PDFs: { x, y, width, height, side }. Inches from top-left. Min

Field	Type	Required	Description
			0.5in for scannability. Side: front (default) or back.
return_address_mode	enum	No	full text none. Default: full. Controls return address rendering: full = white background + text, text = text only on your design, none = hidden (max design space, undeliverable mail not returned).
return_address	object	No	Per-send return address override: { return_first_name, return_last_name, return_company, return_address_line1, return_address_line2, return_address_line3, return_city, return_state, return_postal_code, return_country }. When present, these fields are written to the recipient row and used at render time instead of the recipient or account default. Useful for API single-sends that need a different return address per call (e.g. multi-tenant SaaS shipping mail on behalf of end-customers).

Field	Type	Required	Description
postal_zone_background	enum	No	white transparent. Default: white. Controls front-side postal panel and barcode zone fill. Transparent lets your light design background show through. Fails compliance if background luminance is too dark.
back_barcode_background	enum	No	white transparent. Default: white. Controls back-side barcode clear zone fill independently from the front. Same luminance compliance check applies.
paper_finish	enum	No	matte glossy. Default: matte. Paper coating for postcards. Both use 100lb C2S cardstock.
billing_group	string	No	Cost grouping label, max 255 chars. Unrelated to bundle_key — this is a fulfillment tag that carries through to your invoice.
bundle_key	string	No	Opt-in grouping key. When the auto_bundle_api_campaigns feature is enabled on your account, multiple /v1/postcards/create calls (or single-send

Field	Type	Required	Description
			and batch combined) with the same bundle_key inside a 24-hour window attach to a single auto-generated campaign. Makes the send visible in your /campaigns list and fires campaign.approved/.completed/.failed webhooks. Max 255 chars. Do NOT use PII (customer emails, names) — bundle_key is stored unencrypted and visible in admin logs; the API rejects values containing '@'.
campaign_name	string	No	Human-readable label for the auto-generated campaign. Only applies when bundle_key is set (or on /batch). Max 255 chars. Defaults to 'API bundle: <bundle_key>' or 'API batch <timestamp>' when omitted.
skip_campaign	boolean	No	Explicit opt-out from auto-campaign bundling. When true, the piece is created with campaign_id=NULL (legacy API behavior) regardless of bundle_key or the

Field	Type	Required	Description
			account feature flag. Default: false.
tags	string[]	No	Array of tag strings
test	boolean	No	Test mode. Default: false. Test pieces are never auto-bundled into campaigns.
metadata	object	No	Arbitrary key-value pairs

Example Request:

```
curl -H "Authorization: Bearer dm_live_abc123" -X POST https://api.postally.ca/v1/postcards/create \
```

```
-H "Content-Type: application/json" \
```

```
-H "Idempotency-Key: order-12345-postcard" \
```

```
-d '{"size":"postcard_4x6","template_id":"tpl-uuid","recipient_id":"rcp-uuid",
"merge_data":{"first_name":"Jane"},"mail_class":"standard"}'
```

Response (201):

```
{ "id": "mpc-uuid", "type": "postcard", "format": "postcard_4x6",
"status": "created", "is_test": false, "mail_class": "standard",
"cost": { "base": 69, "postage": 56, "envelope": 0,
"return_envelope": 0, "bucksip": 0, "total": 125 },
"cancel_by": "2026-03-17T06:59:59.999Z", "created_at": "2026-03-16T12:00:00Z"
}
```

Note: Cancellation window: any time before 23:59:59 Vancouver-time (America/Vancouver) on the day the piece was created. The `cancel_by` field is the exact UTC instant of the deadline. Pieces submitted late in the evening have a correspondingly shorter window. The daily print-house dispatch runs at 02:00 Vancouver-time, leaving a 2-hour buffer after the cancellation cutoff.

11.2 Send Letter

Important: Letters are coming soon. This endpoint currently returns 400 type_disabled in production. Available in test mode for integration development.

POST /v1/letters/create

Includes all postcard fields plus:

Field	Type	Required	Description
mail_class	enum	No	first_class (Lettermail) standard (Personalized Mail). Default: first_class. Single sends (always 1 piece) cannot use 'standard' and will be rejected with mail_class_policy_violation.
content_category	enum	No	promotional transactional general. Default: promotional. Only applies when bulk-sending via batch or campaign. Single sends ignore it (they're always Lettermail).
duplex	boolean	No	Double-sided printing. Default: false
page_count	integer	No	1-20 pages. Default: 1
envelope_id	uuid	No	Custom outer envelope
return_envelope_id	uuid	No	Return envelope to include
buckslip_id	uuid	No	Buckslip insert to include

11.3 Send Self-Mailer

Important: Self-mailers are coming soon. This endpoint returns 400 type_disabled in production and requires the self_mailers_enabled feature flag. Available in test mode.

POST /v1/self-mailers/create

Send a self-mailer directly (without a campaign). Accepts the same common fields as postcards plus:

Field	Type	Required	Description
size	enum	No	self_mailer_6x18 (tri-fold) self_mailer_8.5x11 (bi-fold) self_mailer_7x10 (bi-fold) self_mailer_9x12 (bi-fold). Default: self_mailer_6x18
template_id	uuid	Yes	Reference to a published self-mailer template
duplex	boolean	No	Double-sided printing. Default: false
paper_finish	enum	No	matte glossy. Default: matte. Paper coating (80lb stock).

11.4 Send Check

Important: Checks are coming soon. This endpoint returns 400 type_disabled in production.

POST /v1/checks/create

Send a check payment via mail. Requires payee_name, amount, and recipient address. Currently in development.

11.5 Batch Send

Send up to 500 postcards or letters in a single request:

POST /v1/postcards/batch

POST /v1/letters/batch

Body: { "items": [{...}, ...] }. Each element in the items array uses the same schema as the corresponding single-send endpoint. Max 500 items per batch.

Auto-campaign fields (bundle_key, campaign_name, skip_campaign) go at the TOP LEVEL of the body, not on individual items. One batch call = one auto-campaign. Per-item auto-campaign fields are ignored by the batch handler.

Response:

```
{ "batch_id": "uuid", "campaign_id": "uuid-or-null", "total": 500,  
  "created": 498, "failed": 2,  
  "results": [{ "index": 0, "id": "uuid", "status": "queued" }, ...],  
  "errors": [{ "index": 42, "error": "validation_error", "message": "..." }] }
```

campaign_id is populated when the auto_bundle_api_campaigns feature is enabled on your account and the batch was not sent with skip_campaign=true or test=true. When set, every piece in the batch is linked to that campaign and appears in the /v1/campaigns/list response and campaign-level webhooks.

11.6 Auto-campaigns for API sends

Per-batch and bundle_key-grouped API sends can be automatically bundled into campaign rows so that API customers get the same campaign-level visibility (admin Campaign Overview, customer /campaigns page, campaign webhooks, future per-campaign QBO sync) as dashboard customers. The feature is off by default on each account and is enabled via the auto_bundle_api_campaigns feature flag.

Behavior when enabled:

POST /v1/postcards/batch (and /v1/letters/batch) auto-creates one campaign per call with source='api_batch'. Every mailpiece in the batch is linked to that campaign. Response body includes campaign_id.

POST /v1/postcards/create (and /v1/letters/create) with a bundle_key parameter attaches to (or creates) a campaign with source='api_bundled'. Multiple /create calls with the same bundle_key inside a 24-hour window — and the same type + format — share one campaign. bundle_key is case-sensitive. Without bundle_key, single-send /create is unaffected (campaign_id=NULL).

Auto-campaigns jump straight to status='processing' — no proof approval flow. Proofs are a dashboard/marketing feature; transactional API sends bypass them entirely.

campaign.approved, campaign.completed, and campaign.failed webhook events fire as they do for dashboard campaigns. The payload gains an optional source field ('api_batch' | 'api_bundled' | 'dashboard') and bundle_key (string | null) so subscribers can distinguish origin if they care.

Different (type, format) combinations under the same bundle_key fork into separate campaigns — campaigns.format is NOT NULL. A bundle_key used across postcard_4x6 and postcard_6x9 produces two campaigns with the same bundle_key label.

Billing is unchanged. API sends still debit prepaid credits per piece. The campaign row is a visibility/reporting artifact; it does not carry pricing columns.

Behavior when disabled (or when skip_campaign=true or test=true):

Pieces are created with campaign_id=NULL, matching the legacy API behavior. Nothing else changes.

Note: Retry semantics: /batch and /create are safe to retry with the same Idempotency-Key header. A retried call returns the original response (including campaign_id) without creating a duplicate campaign or double-counting pieces, even if the first attempt crashed before the idempotency cache was written.

Note: bundle_key is stored unencrypted and visible in admin logs. Do not put customer PII (emails, names, account numbers) in bundle_key. The API rejects values containing '@' as a defensive guard.

11.7 Cost Breakdown

Every send response includes a cost object with per-component pricing in cents (CAD):

Field	Type	Required	Description
base	integer	—	Base printing cost
postage	integer	—	Mailing/postage cost
envelope	integer	—	Custom envelope surcharge (0 if standard)
return_envelope	integer	—	Return envelope cost
buckslip	integer	—	Buckslip insert cost
total	integer	—	Sum of all components

12. Mailpieces

12.1 Get Mailpiece

GET /v1/mailpieces/get/:id

Returns the full mailpiece with cost breakdown, tracking info, and QR scan count. The response includes flat cost fields (`cost_base_cents`, `cost_postage_cents`, `cost_envelope_cents`, `cost_return_envelope_cents`, `cost_buckslip_cents`, `cost_total_cents`, `tax_cents`) as well as a legacy nested cost object for backward compatibility.

When status is 'failed' or 'returned', the response also includes a top-level `failure_reason` string (human-readable) and a whitelisted metadata object containing either `compliance` (if the piece was rejected by post-render PDF compliance) or `render_error` (if rendering crashed before compliance ran). The compliance payload has the shape `{ passed, checked_at, checks: [{ check, passed, detail }] }`. The `render_error` payload has the shape `{ error, path, failed_at }`.

12.2 List Mailpieces

GET /v1/mailpieces/list

Field	Type	Required	Description
<code>limit</code>	integer	No	1-100, default 20
<code>offset</code>	integer	No	Default 0
<code>status</code>	string	No	Filter by status
<code>type</code>	string	No	Filter by type (postcard, letter, self_mailer)
<code>campaign_id</code>	uuid	No	Filter by campaign
<code>billing_group</code>	string	No	Filter by billing group

12.3 Cancel Mailpiece

POST /v1/mailpieces/cancel

Body: `{ "id": "uuid" }`. Must be before the cancellation deadline — 23:59:59 Vancouver-time (America/Vancouver) on the day the piece was created. The exact deadline for a given piece is

returned as `cancel_by` in the create response and on GET `/v1/mailpieces/get/:id`. Returns 409 `cancel_window_expired` if the deadline has passed.

12.4 Status Lifecycle

Mailpieces progress through the following statuses:

```
created -> queued -> processing -> printing -> dispatched -> inducted -> in_transit -> in_local_area -> delivered
```

Note: For postcards (Personalized Mail), statuses after 'inducted' are ESTIMATES based on Canada Post service standards. There is no per-piece tracking for postcards. 'inducted' means Canada Post accepted the batch.

A mailpiece can also transition to returned (undeliverable) or failed (production error) from various stages. Cancellation is only possible from created or queued status within the cancel window.

13. QR Code Tracking

Enable QR codes on any mail piece to track recipient engagement. Each scan is logged with timestamp, IP address, and geolocation.

13.1 Enable QR Codes

Set `qr_code_enabled: true` and `qr_target_url` on any send request. The QR code is automatically rendered on the mail piece.

13.2 Scan Redirect

```
GET /qr/:code
```

Public endpoint (no auth). When a recipient scans the QR code, they are redirected (302) to the target URL. The scan is logged automatically.

13.3 Scan Analytics

```
GET /v1/analytics/qr/:mailpiece_id
```

Returns total scan count and individual scan records with `scanned_at`, `ip_address`, `geo_country`, and `geo_city`.

14. Campaigns

Campaigns enable batch sending to an entire recipient group. Create a campaign, review the estimated cost, then approve to trigger async mailpiece creation.

14.1 Create Campaign

POST /v1/campaigns/create

Field	Type	Required	Description
name	string	Yes	1-255 characters
type	enum	Yes	postcard letter self_mailer
format	enum	Yes	Mailpiece format (e.g., postcard_4x6, letter_standard)
template_id	uuid	Conditional	Template to use for all pieces. Either template_id or design_id is required.
design_id	uuid	Conditional	Design from the template builder. Either template_id or design_id is required.
group_id	uuid	Yes	Recipient group to send to
mail_class	enum	No	first_class (Lettermail) standard (Personalized Mail). Auto-suggested from piece_count + content_category: Lettermail for <125 pieces or non-promotional content, Personalized Mail for 125+ promotional. Explicit 'standard'

Field	Type	Required	Description
			below the 125-piece floor is rejected with mail_class_policy_violation.
content_category	enum	No	promotional transactional general. Default: promotional. Only 'promotional' is eligible for Personalized Mail ('standard' mail_class) per Canada Post rules. Transactional and general correspondence must use Lettermail regardless of piece count.
attestation_accepted	boolean	Conditional	Required when mail_class='standard'. Customer attests the mailing is addressed marketing/promotional content and complies with Canada Post Personalized Mail eligibility. See GET /v1/policy/mail-class for the canonical attestation text. Captured on campaigns.attestation_accepted_at and audited via campaign.attestation_accepted.
envelope_id	uuid	No	Custom envelope (letters only)
return_envelope_id	uuid	No	Return envelope (letters only)

Field	Type	Required	Description
buckslip_id	uuid	No	Buckslip insert (letters only)
qr_code_enabled	boolean	No	Default: false
duplex	boolean	No	Default: false
billing_method	enum	No	prepaid_credits pay_per_job. Default: prepaid_credits
return_address_mode	enum	No	full text none. Default: full. Controls return address rendering on postcards.
postal_zone_background	enum	No	white transparent. Default: white. Front-side postal panel and barcode zone fill.
back_barcode_background	enum	No	white transparent. Default: white. Back-side barcode clear zone fill.
paper_finish	enum	No	matte glossy. Default: matte. Paper coating for postcards.
scheduled_at	timestamp	No	ISO 8601 datetime for scheduled send. Campaign enters 'scheduled' status after proof approval.
schedule_type	enum	No	immediate scheduled. Default: immediate.
production_start_date	date	No	Preferred production date (YYYY-MM-DD). Used for planning with print house.

Field	Type	Required	Description
sample_email	string	No	Email address to receive a digital sample before production.
sample_physical	boolean	No	Request a physical sample mailpiece before full production. Default: false.
billing_group	string	No	Label to group mailpieces for billing/reporting (e.g., department, client name).
metadata	object	No	Arbitrary key-value pairs

Response includes `piece_count` and `estimated_cost_cents` based on the group size.

14.2 Approve Campaign

POST `/v1/campaigns/approve`

Two-phase approval: First call (draft) generates a proof and emails it to you. Second call (awaiting_proof) with the `proof_token` starts production. If pricing changed since creation, returns a `price_changed` error requiring `accept_price_change: true` to proceed.

14.3 Update Draft Campaign

PATCH `/v1/campaigns/:id`

Update a draft campaign. Only fields in the whitelist below can be changed, and the campaign must be in 'draft' status. Attempts to patch a campaign in any other state return 409 conflict — use `/v1/campaigns/:id/revise` first to move it back to draft.

Field	Type	Required	Description
name	string	No	1-255 characters
mail_class	enum	No	first_class standard. Re-runs policy validation. Changing to

Field	Type	Required	Description
			'standard' below 125 pieces or on non-promotional content fails with mail_class_policy_violation.
content_category	enum	No	promotional transactional general. Re-runs policy validation. Emits a campaign.content_category_changed audit action.
attestation_accepted	boolean	No	Accept the Canada Post Personalized Mail attestation. Required when switching mail_class to 'standard'. Emits campaign.attestation_accepted audit action.
group_id	uuid	No	Change recipient group. Piece count and pricing are re-computed.
return_address_mode	enum	No	full text none
postal_zone_background	enum	No	white transparent
back_barcode_background	enum	No	white transparent
paper_finish	enum	No	matte glossy

Response is the updated campaign row plus a fresh pricing_estimate block so the dashboard can refresh the cost summary. Mail-class changes additionally emit a campaign.mail_class_changed audit log entry with before/after values.

14.4 Campaign Lifecycle

draft -> generating_proof -> awaiting_proof -> processing -> completed

| | |

draft (revision) failed cancelled

Campaigns require proof approval before production. Scheduled campaigns transition from awaiting_proof to scheduled, then processing at the scheduled time. A campaign transitions to 'failed' automatically when every mailpiece reaches a terminal-failure state (post-render compliance rejection or render crash) with nothing ever dispatched — in that case, credits for each failed piece are automatically refunded.

14.5 List / Get

GET /v1/campaigns/list

GET /v1/campaigns/get/:id

The Get response includes a pieces_by_status object derived in real time from the mailpieces table: { created, queued, processing, printing, dispatched, inducted, in_transit, in_local_area, delivered, returned, failed, cancelled }. Always prefer this map over the legacy pieces_sent / pieces_failed aggregate columns — those counters are bumped at batch-completion logging time and can be stale if post-render compliance rejects pieces afterwards.

When the campaign is in status='failed', the response also includes failure_reason (short machine code, e.g. 'all_inserts_failed' or 'approve_batch_exception') and failure_detail (human-readable sentence). Both fields are null on non-failed campaigns. Use failure_reason for programmatic dispatch and failure_detail for UI display.

14.6 Resend Proof

POST /v1/campaigns/:id/resend-proof

Re-sends the proof email for a campaign in awaiting_proof status. Optionally send to a different email address (e.g., for client review). Rate limited to 3 resends per campaign.

Field	Type	Required	Description
email	string	No	Send proof to this email instead of account owner. Useful for client review.

```
curl -H "Authorization: Bearer dm_live_abc123" -X POST https://api.postally.ca/v1/campaigns/{id}/resend-proof \
```

```
-H "Content-Type: application/json" \
```

```
-d '{"email": "client@example.com"}'
```

```
Response: { "success": true, "message": "Proof email sent to client@example.com" }
```

14.7 Revise Campaign

POST /v1/campaigns/:id/revise

Request a revision for a campaign in awaiting_proof status. Transitions the campaign back to draft for changes. Can be used by customers to self-fix issues or report bugs with the proof.

Field	Type	Required	Description
proof_token	string	Yes	Proof token from the proof email (validates the requester saw the proof)
revision_note	string	Yes	Description of what needs to change (10-5000 characters)
revision_type	enum	Yes	self_fix (customer will fix it themselves) bug_report (issue with the platform)
affected_area	string	No	Which part of the design is affected (e.g., front, back, address zone)

14.8 Cancel Campaign

POST /v1/campaigns/cancel

Cancel a campaign. Only works for campaigns in draft, generating_proof, awaiting_proof, or scheduled status. Campaigns in processing or completed status cannot be cancelled.

Field	Type	Required	Description
id	uuid	Yes	Campaign ID to cancel

15. Billing: Prepaid Credits

API users typically use prepaid credits. Buy credit bundles upfront, and costs are automatically deducted when you send mail.

15.1 Check Balance

GET /v1/account/balance

Returns: { "balance_cents": 50000, "balance_display": "\$500.00", "currency": "CAD" }

15.2 Top Up Credits

POST /v1/account/topup

Body: { "amount_cents": 10000 }. Minimum \$20 (2000 cents), maximum \$5,000 (500000 cents). Creates a Stripe Checkout session and returns a checkout_url.

If a pending top-up already exists for this account with the same amount, the same checkout_url is returned along with pending_session_exists: true (HTTP 200). If a pending top-up exists for a DIFFERENT amount, the request is rejected with HTTP 409 pending_topup_exists and the response includes pending_amount_cents + pending_checkout_url so the caller can either complete or cancel the pending session first. The guard is cleared automatically when Stripe reports the session as completed or expired.

Note: Volume bonuses are not currently active. All top-ups receive 1:1 credit value. Contact sales for volume pricing.

15.3 Credit History

GET /v1/account/credits/history

Returns a paginated list of credit ledger entries with type (credit/debit), source, description, and running balance.

15.4 Automatic Refunds on Compliance or Render Failure

If a mailpiece fails post-render PDF compliance (e.g., its QR code overlaps the Canada Post barcode clear zone, or its back-side design violates a spec) or the rendering pipeline crashes before compliance runs, the mailpiece is marked 'failed' and the full charge (cost_total_cents +

tax_cents) is automatically refunded to your credit balance. You do not need to take action — the refund is idempotent and visible in the credit history with source=mailpiece_refund.

You can see the reason for the failure on the mailpiece detail endpoint (GET /v1/mailpieces/get/:id) via the top-level failure_reason field and the whitelisted metadata.compliance or metadata.render_error objects. Campaigns whose pieces all fail transition to the 'failed' status automatically.

16. Billing: Pay-Per-Job Orders

For one-off jobs, Postally supports pay-per-job billing via Stripe Checkout. No prepaid credits needed.

16.1 Get Price Quote

POST /v1/orders/quote

Public endpoint (no authentication required). Returns pricing comparison between pay-per-job and prepaid credits.

Field	Type	Required	Description
mailpiece_format	enum	Yes	e.g., postcard_4x6, postcard_5x7, postcard_6x9, postcard_6x11, letter_standard
mail_class	enum	No	first_class (Lettermail) standard (Personalized Mail). Default: standard. Personalized Mail requires piece_count >= 125 AND content_category='promotional' at order create time.
piece_count	integer	Yes	1 to 100,000
content_category	enum	No	promotional transactional general. Default: promotional. Only 'promotional' is eligible for Personalized Mail.
custom_envelope	boolean	No	Default: false
return_envelope	boolean	No	Default: false
buckslip	boolean	No	Default: false

Field	Type	Required	Description
qr_code	boolean	No	Default: false
duplex	boolean	No	Default: false
tax_rate_percent	number	No	0-30. Default: 0

Important: Quotes are ADVISORY. `/v1/orders/quote` does NOT enforce the Canada Post mail-class policy — it returns the cheapest valid price with a fallback to Lettermail if the requested `mail_class` is ineligible. The create-time validator at `POST /v1/orders/create` IS strict and will reject a standard mailing below the 125-piece floor with `mail_class_policy_violation`. Check the response's 'policy' block if present, or call `POST /v1/pricing/quote` which DOES return a policy verdict with `allowed/reason/suggested_mail_class`.

16.2 Create Order

POST `/v1/orders/create`

Creates a draft order with reserved pricing (valid for 24 hours). Returns full pricing breakdown and line items.

16.3 Checkout

POST `/v1/orders/checkout`

Body: `{ "order_id": "uuid" }`. Returns a Stripe Checkout URL. The customer completes payment on Stripe, and the webhook automatically marks the order as paid and begins production.

Response:

```
{ "order_id": "uuid", "checkout_url": "https://checkout.stripe.com/pay/...",
  "total_cents": 15000, "total_display": "$150.00", "expires_in_seconds": 3600 }
```

16.4 Order Lifecycle

`draft -> awaiting_payment -> paid -> processing -> completed`

Orders can also reach refunded (charge reversed), expired (quote timed out), or cancelled (user cancelled before payment).

16.5 List / Get / Cancel

GET /v1/orders/list

GET /v1/orders/get/:id

POST /v1/orders/cancel

Only draft or awaiting_payment orders can be cancelled.

17. Outbound Webhooks

Receive real-time notifications when events occur. Postally sends HMAC-signed HTTP POST requests to your URL.

17.1 Create Webhook

POST /v1/webhooks/create

```
Body: { "url": "https://yourapp.com/webhook", "events": ["mailpiece.delivered", "mailpiece.failed"] }
```

Response includes a secret field (shown only once). Store it securely for signature verification.

17.2 Event Types

Event	Category	Required	Description
mailpiece.created	Mailpiece	—	New mailpiece created
mailpiece.processing	Mailpiece	—	Entered production queue
mailpiece.printing	Mailpiece	—	On press, being printed
mailpiece.printed	Mailpiece	—	Print run complete, awaiting dispatch
mailpiece.dispatched	Mailpiece	—	Dispatched to print house for production
mailpiece.inducted	Mailpiece	—	Canada Post accepted batch (Statement of Mailing)
mailpiece.in_transit	Mailpiece	—	In carrier network
mailpiece.in_local_area	Mailpiece	—	At local delivery facility
mailpiece.delivered	Mailpiece	—	Successfully delivered
mailpiece.returned	Mailpiece	—	Returned to sender

Event	Category	Required	Description
mailpiece.failed	Mailpiece	—	Production or delivery failure
mailpiece.cancelled	Mailpiece	—	Cancelled within window
mailpiece.qr_scanned	Mailpiece	—	QR code scanned by recipient
campaign.approved	Campaign	—	Campaign approved, processing started
campaign.completed	Campaign	—	All pieces sent
campaign.failed	Campaign	—	Campaign processing failed
recipient.suppressed	Recipient	—	Recipient marked as suppressed

17.3 Signature Verification

Every webhook request includes an X-Webhook-Signature header containing HMAC-SHA256(request_body, secret). Verify it to ensure the request is from Postally.

Node.js Example:

```
const crypto = require('crypto');

const expected = crypto.createHmac('sha256', webhookSecret)
  .update(rawBody).digest('hex');

const valid = crypto.timingSafeEqual(
  Buffer.from(signature), Buffer.from(expected));
```

17.4 Retry Behavior

Failed deliveries (non-2xx response) are retried with exponential backoff and jitter, up to 3 retries over 24 hours.

17.5 List / Delete

```
GET /v1/webhooks/list
```

Note: Paginated. Accepts ?limit (default 25, max 100), ?offset, or ?page. Response shape: { data: [...], pagination: { total, limit, offset } }.

POST /v1/webhooks/delete

18. Inbound Triggers

Triggers let external services automatically send mail by posting data to a unique URL. Configure field mapping to extract recipient information from any JSON payload.

18.1 Create Trigger

POST /v1/triggers/create

Field	Type	Required	Description
name	string	Yes	1-255 characters
template_id	uuid	Yes	Template to use for the trigger mailpiece
mailpiece_type	enum	No	postcard letter self_mailer. Default: postcard
format	enum	No	Mailpiece format
envelope_id	uuid	No	Envelope (letters only)
return_envelope_id	uuid	No	Return envelope
buckslip_id	uuid	No	Buckslip insert
field_mapping	object	Yes	Maps recipient fields to JSON paths (dot notation). For return address support, map payload paths to return_first_name, return_company, return_address_line1, return_city, return_state, return_postal_code, return_country, etc.
default_return_address	object	Conditional	Fallback return address used when the inbound webhook payload doesn't carry

Field	Type	Required	Description
			<p>return_* fields via field_mapping.</p> <p>REQUIRED for self_mailer triggers (Canada Post compliance — self-mailers must always have a Canadian return address).</p> <p>Shape: {</p> <ul style="list-style-type: none"> return_first_name, return_last_name, return_company, return_address_line1, return_address_line2, return_city, return_state, return_postal_code, return_country }

Example: Shopify Order Trigger

```

{ "name": "Shopify Thank You Card",
  "template_id": "tpl-uuid",
  "mailpiece_type": "postcard",
  "field_mapping": {
    "first_name": "customer.first_name",
    "address_line1": "shipping_address.address1",
    "city": "shipping_address.city",
    "postal_code": "shipping_address.zip",
    "country": "shipping_address.country_code"
  }
}

```

```
}
```

Response includes `inbound_url` and `secret` (shown only once). Configure Shopify to POST to the inbound URL with the secret as the X-Trigger-Secret header.

18.2 Fire Trigger

```
POST /v1/triggers/inbound/:id
```

Authenticate via X-Trigger-Secret header. Query parameter authentication is not supported (secrets in URLs leak in server logs). Send any JSON body — the `field_mapping` extracts recipient data automatically.

18.3 List / Delete

```
GET /v1/triggers/list
```

```
POST /v1/triggers/delete
```

19. File Uploads

Upload PDF and image files for templates, envelopes, and buckslips using S3 presigned URLs.

19.1 Get Presigned URL

POST /v1/uploads/presign

Field	Type	Required	Description
filename	string	Yes	1-255 characters
content_type	enum	Yes	application/pdf image/png image/jpeg
purpose	enum	Yes	template envelope buckslip
file_size	integer	No	Max 20 MB. Recommended: keep under 5 MB per format for optimal rendering performance.

19.2 Upload Flow

1. Call POST /v1/uploads/presign to get upload_url and file_url.
2. PUT your file directly to upload_url with the correct Content-Type header.
3. Use file_url as front_pdf_url or back_pdf_url when creating templates, envelopes, or buckslips.

20. Zapier & Make.com Integration

Connect Postally to 5,000+ apps through Zapier or Make.com (formerly Integromat).

20.1 Authentication Test

GET /v1/integrations/auth/test

Validates your API key and returns account details.

20.2 Triggers (Polling)

Zapier polls these endpoints for new data:

Endpoint	Method	Required	Description
/triggers/new-mailpiece	GET	—	New mailpieces created
/triggers/status-change	GET	—	Mailpiece status updates
/triggers/new-recipient	GET	—	New recipients added
/triggers/qr-scan	GET	—	QR code scans
/triggers/new-order	GET	—	New orders created

All triggers support ?since=ISO_DATE&limit=N query parameters.

20.3 Webhook Subscriptions (Make.com)

POST /v1/integrations/hooks/subscribe

POST /v1/integrations/hooks/unsubscribe

20.4 Actions

Endpoint	Method	Required	Description
/actions/send-postcard	POST	—	Send a postcard

Endpoint	Method	Required	Description
/actions/send-letter	POST	—	Send a letter
/actions/create-recipient	POST	—	Create a recipient
/actions/price-quote	POST	—	Get a price quote
/actions/balance	GET	—	Check credit balance

All action endpoints accept a test: true boolean. When set, no real mail is sent and no credits are charged — the mailpiece is created with is_test=true. The format field is validated against the same enum as /v1/postcards/create and /v1/letters/create; invalid formats return 400 validation_error without reaching the pricing engine.

21. Compliance Endpoints

21.1 GDPR: Right to Erasure

POST /v1/gdpr/erasure

Body: { "recipient_id": "uuid" }. Permanently destroys all personal data. Cascades to mailpiece snapshots, S3 PDFs, and QR analytics. Returns the count of S3 objects deleted.

Note: Recipients with active PHI retention periods cannot be erased (returns 409 retention_conflict).

21.2 GDPR: Right to Access

GET /v1/gdpr/access?recipient_id=uuid

Returns all stored data including mail history, group memberships, sub-processors, and processing purposes.

21.3 GDPR: Data Export

GET /v1/gdpr/export?recipient_id=uuid

Returns a machine-readable JSON file (Content-Disposition: attachment) per GDPR Article 20.

21.4 PHI Disclosure Accounting

GET /v1/hipaa/disclosure-accounting?recipient_id=uuid

Returns all PHI disclosures (to print house, Stripe, address verification). Only available for accounts flagged as PHI-regulated. The endpoint path retains the historical /hipaa/ prefix for backward compatibility; Postally itself is not HIPAA-certified.

22. Account & Specs Endpoints

22.1 Audit Log

GET /v1/account/audit-log

Returns a customer-facing audit trail of account activity. Supports pagination via limit and offset query parameters.

22.2 Platform Specs

Read-only endpoints returning platform configuration and capabilities. All /v1/specs/* endpoints REQUIRE authentication (will return 401 without a valid API key or session token).

GET /v1/specs

Returns supported formats, viewport dimensions, bleed + safe-zone geometry per format, supported countries for address verification, test mode behavior, and versioning info. Includes a spec_versions.v2_postcard_spec block with the full v2 postcard geometry (trim, full, address_box, text_safe_area, barcode_clear per format). Does NOT return pricing tiers — use GET /admin/pricing/public-summary (public, read-only) or POST /v1/orders/quote for pricing.

GET /v1/specs/templates

Returns supported template types, formats, and merge variable syntax.

GET /v1/specs/test-mode

Returns test mode behavior and limitations.

GET /v1/specs/versioning

Returns API versioning policy and changelog.

GET /v1/specs/address-verification

Returns supported address verification countries and provider details.

22.3 Canada Post Mail Class Policy

GET /v1/policy/mail-class

PUBLIC endpoint (no authentication required). Returns the single source of truth for Canada Post Personalized Mail eligibility rules. Dashboards and integrators SHOULD fetch this once

and cache, rather than hardcoding constants.

Response (200):

```
{  
  
  "personalized_mail_min_pieces": 125,  
  
  "content_categories": {  
  
    "promotional": { "label": "Promotional / Marketing", "eligibleForPersonalizedMail": true, ... },  
  
    "transactional": { "label": "Transactional", "eligibleForPersonalizedMail": false, ... },  
  
    "general": { "label": "General correspondence", "eligibleForPersonalizedMail": false, ... }  
  
  },  
  
  "attestation_text": "I confirm this mailing is addressed marketing/promotional content..."  
  
}
```

Policy enforcement summary:

Single sends (POST /v1/postcards/create, POST /v1/letters/create) are always 1 piece and therefore cannot use mail_class='standard' — attempts return 400 mail_class_policy_violation. Campaigns and batches must have piece_count >= 125 AND content_category='promotional' to use mail_class='standard'. Below the floor or with non-promotional content, the validator returns 400 mail_class_policy_violation with a suggested_mail_class='first_class' fallback. POST /v1/pricing/quote surfaces the policy verdict in a 'policy' block ({allowed, suggested_mail_class, reason}). POST /v1/orders/quote is ADVISORY and does NOT run the validator — check the policy block if present, or expect rejection at /v1/orders/create time. Attestation is required on every campaign/order/batch submitting mail_class='standard'. See the attestation_accepted field on POST /v1/campaigns/create and POST /v1/orders/create. Full rationale (phantom fee, Canada Post 100-piece minimum charge, Postally 25-piece buffer) is covered in the Acceptable Use Policy PDF under 'Canada Post Personalized Mail Eligibility'.

22.4 Postcard Spec Versions (v1 vs v2)

Postcards have two coexisting geometry versions identified by spec_version on template_designs, campaigns (as proof_spec_version), and mailpieces. All NEW postcards on production use v2.

Field	Type	Required	Description
v1 (legacy)	deprecated	—	trim = full = nominal card size (6x4, 7x5, 9x6, 11x6). Safe zone 0.25" inset from card edge. Address panel on the right half. Single unified white postal panel. No longer available for new templates as of April 2026.
v2 (current)	active	—	trim = nominal, full = trim + 0.125" bleed per side (e.g., 4x6 → 6.25"x4.25" full). Safe zone 0.125" inside trim. Address box at bottom-right of full-bleed canvas. Indicia (22x40mm) and return address (22x40mm) nested inside the address box at 5mm insets. Barcode clear strip flush to trim bottom.

Key implications for customer-uploaded PDFs and templates:

Authoring canvas MUST be full-bleed dimensions, not trim. A 4x6 postcard design must be 6.25"x4.25" at 300-312 DPI.

Bleed area (outer 0.125") will be trimmed off during finishing — do not place critical content there.

Safe zone is an additional 0.125" inside trim — recipient-critical content must stay inside the safe zone.

Address box location is fixed by Canada Post geometry. Do not place design content where the recipient address renders — see GET /v1/specs response `spec_versions.v2_postcard_spec.formats.{format}.address_box` for the exact coordinates. The Postcard Design Guide PDFs (postcard-design-guide-4x6, -5x7, -6x9, -6x11) available under GET /v1/docs contain the canonical CAD diagrams with all zone coordinates in inches and millimeters.

23. API Key Management

Manage your API keys programmatically — list existing keys, generate new ones, revoke compromised keys, and update key properties.

23.1 List API Keys

GET /v1/api-keys/list

Returns API keys for your account, including revoked keys. Paginated.

Note: Accepts ?limit (default 25, max 100), ?offset, or ?page. Response shape: { data: [...], pagination: { total, limit, offset } }.

Response fields per key:

Field	Type	Required	Description
id	UUID	—	Unique key identifier
key_prefix	string	—	First 12 characters of the key (e.g., dm_live_abc1)
name	string	—	Human-readable key name
environment	string	—	"test" or "live"
permissions	array	—	Permission scopes (e.g., ["*"] for full access)
phi_access_level	string	—	PHI access tier: phi_access_none, phi_access_address, phi_access_content, phi_access_full
last_used_at	timestamp	—	When the key was last used to authenticate
expires_at	timestamp	—	Optional expiry date (null = never expires)

Field	Type	Required	Description
is_revoked	boolean	—	Whether the key has been revoked

23.2 Create API Key

POST /v1/api-keys/create

Generate a new API key. The full key string is returned only once — store it securely.

Field	Type	Required	Description
name	string	Yes	Human-readable name (1-255 characters)
environment	string	No	"test" or "live" (default: "test")
permissions	array	No	Permission scopes (default: ["*"])
phi_access_level	string	No	PHI access tier (default: "phi_access_none")
expires_at	timestamp	No	Optional expiry date

Example:

```
curl -H "Authorization: Bearer dm_live_abc123" -X POST https://api.postally.ca/v1/api-keys/create \
```

```
-H "Content-Type: application/json" \
```

```
-d '{"name": "Production Key", "environment": "live", "permissions": ["*"]}'
```

Important: The full API key is returned only at creation time. Store it immediately — it cannot be retrieved later.

23.3 Revoke API Key

POST /v1/api-keys/revoke

Revoke an API key. Revoked keys cannot be used to authenticate. Use the reactivate endpoint to restore a revoked key.

Field	Type	Required	Description
id	UUID	Yes	ID of the API key to revoke

23.4 Reactivate API Key

POST /v1/api-keys/reactivate

Restore a previously revoked API key. The key becomes active again with its original permissions.

Field	Type	Required	Description
id	UUID	Yes	ID of the revoked API key to reactivate

23.5 Update API Key

POST /v1/api-keys/update

Update a key's name, permissions, PHI access level, or expiry date. Cannot update revoked keys.

Field	Type	Required	Description
id	UUID	Yes	ID of the API key to update
name	string	No	New name (1-255 characters)
permissions	array	No	New permission scopes
phi_access_level	string	No	New PHI access tier
expires_at	timestamp	No	New expiry date (null to remove expiry)

24. Test Mode

Use test mode during development to validate your integration without sending real mail or incurring charges.

Feature	Details
Activation	Use a test API key (dm_test_...) or pass test=true in request body
Mail delivery	No real mail is printed or delivered
Billing	No credits charged, no Stripe charges
Data	Test mailpieces created with is_test=true
Recipients	Shared between test and live modes
Endpoints	All endpoints behave identically except billing and production

25. PDF Compliance & Rendering

Every rendered postcard undergoes 18 automated compliance checks before entering the print queue. If more than 5% of pieces in a campaign fail compliance, the campaign is cancelled and credits are refunded.

Compliance Checks:

#	Check	Required	Description
1	File size	Yes	Minimum 5KB (detects blank renders)
2	Page count	Yes	1-2 pages (front, optional back)
3	Dimensions	Yes	Within 0.15in of target format
4	Orientation	Yes	Landscape for postcards
5	Content coverage	Yes	Minimum 20KB (real content, not blank)

#	Check	Required	Description
6	Merge variables	Yes	No raw {{variable}} leaked in output
7	Address present	Yes	Recipient name + street + city + postal
8	Address zone	Yes	Address within Canada Post postal zone
9	Back barcode zone	Yes	No content in back-side barcode area
10	Address truncation	Warn	Flags addresses truncated to fit zone
11	Image failures	Yes	Rejects gray-box renders from 404 images
12	QR scannability	Yes	Module size min 0.03in for reliable scan
13	Text contrast	Yes	WCAG 2.0 — text 4.5:1, QR 3:1 minimum
14	Return address mode	Warn	Warns when mode=none (mail not returned) or mode=text (verify readability)
15	Postal zone luminance	Yes	When postal_zone_background=transparent, background must have luminance >= 0.70
16	Back barcode luminance	Yes	When back_barcode_background=transparent, same luminance check
17	Barcode bg mismatch	Warn	Warns when front and back barcode zone settings differ

#	Check	Required	Description
18	Content moderation	Yes	Blocks designs flagged for NSFW/offensive content

Postal Panel (Front Side):

The front (mailing) side includes a postal panel covering the right portion of the card. By default this is a white rectangle containing recipient address, postage indicia, and quiet zone per Canada Post spec. Set `postal_zone_background` to transparent to let your light-coloured design show through instead (compliance check verifies background luminance).

Return Address Mode:

Controls how the return address renders on the front side. `full` (default): white background panel + black text. `text`: black text only, rendered directly on your design. `none`: no return address (maximum design space, but undeliverable mail cannot be returned).

Barcode Clear Zone:

A 19mm strip at the bottom of both front and back sides, full card width, reserved for Canada Post barcode printing. By default this is white. The back side can be made transparent independently via `back_barcode_background`. Content placed in this area will be covered when using white mode.

Content Moderation:

All uploaded images and design captures are automatically scanned by AWS Rekognition for NSFW/offensive content. Content detected at 50% or higher confidence (nudity, violence, hate symbols, drugs, etc.) is blocked immediately and cannot be printed or mailed.

QR Zone for Customer PDFs:

When using customer-uploaded PDFs (not the design builder), specify `qr_zone` on `POST /v1/postcards/create` to place a tracked QR code at exact coordinates. Without `qr_zone`, QR codes are auto-placed in the bottom-left safe area if `qr_code_enabled` is true.

```
{ "qr_zone": { "x": 0.5, "y": 2.5, "width": 0.75, "height": 0.75, "side": "front" } }
```

Coordinates in inches from top-left. Minimum 0.5in for scannability.

Text Rendering:

Address text auto-shrinks from default font size to minimum 8pt to fit the zone. Non-Latin characters (CJK, Arabic, emoji) are replaced with ? using the LiberationSans font fallback. Rotated text elements are supported via PDF coordinate transforms.

26. Permission Enforcement

API keys with specific permissions (not wildcard *) are automatically checked against the HTTP method and URL path. Write operations (POST, PUT, PATCH, DELETE) require resource.write permission; read operations (GET) require resource.read.

Permission Format:

```
resource.action – e.g., campaigns.write, recipients.read, templates.*
```

Examples:

Key Permissions	Request	Required	Result
[*]	POST /v1/campaigns/create	campaigns.write	Allowed (wildcard)
[recipients.read, campaigns.read]	GET /v1/recipients/list	recipients.read	Allowed
[recipients.read, campaigns.read]	POST /v1/campaigns/create	campaigns.write	403 Forbidden

Note: Dashboard sessions always have wildcard permissions. Permission enforcement only applies to API key access.

Input Sanitization:

All name and title fields are automatically sanitized to strip HTML tags. This prevents XSS when API responses are rendered in HTML contexts. Text content is preserved: 'Hello' becomes 'Hello'.

27. Recommended Retry Strategy

For robust integrations, implement retry logic for transient failures:

Code	Type	Required	Description
429	Rate Limited	—	Wait for the retry-after header value, then retry
5xx	Server Error	—	Exponential backoff: 1s, 2s, 4s. Max 3 retries.
Network error	Connection failed	—	Retry with backoff. Always use Idempotency-Key.

Always include an Idempotency-Key header on POST requests so retries are safe and won't create duplicates.