API Interface (Proposed)

API Interface Documentation (Proposed)

Generated: 2026-02-24

This document describes the **proposed** stable interface for the receipts-first architecture.
It is intentionally explicit: writes happen only through command endpoints.

Conventions
- All responses include `requestId`.
- Success:
  - `{"ok": true, "data": ..., "requestId": "..."}`
- Error:
  - `{"ok": false, "error": {"code":"...","message":"...","details":{}}, "requestId":"..."}`
- Idempotency:
  - All command endpoints accept `Idempotency-Key` header (UUID).
- Auth:
  - Session cookie / bearer token from better-auth.
  - Identity selection explicit via `X-Identity-Id` header (or body field).

Core resources

Identities
- `GET /v1/identities` — list identities for account
- `POST /v1/identities` — create a custodial identity (Vault Transit key)
- `GET /v1/identities/:id` — identity details (includes publicKey)
- `POST /v1/identities/:id/commands/revoke` — revoke (no more signing)

Pixbooks (bound to identity)
Queries
- `GET /v1/pixbooks` — list pixbooks for account
- `GET /v1/pixbooks/:id` — pixbook projection (fast read)
- `GET /v1/pixbooks/:id/receipts?after=<eventId>` — canonical receipts feed
- `GET /v1/pixbooks/:id/snapshot` — derived snapshot (optional)

Commands
- `POST /v1/pixbooks/commands/create` — creates pixbook + `PIXBOOK_CREATED` receipt
- `POST /v1/pixbooks/:id/commands/save` — explicit save receipt
- `POST /v1/pixbooks/:id/commands/add-card`
- `POST /v1/pixbooks/:id/commands/remove-card`

Trades (cards tradable, books remain owned)
- `POST /v1/trades/commands/propose`
- `POST /v1/trades/:id/commands/accept`
- `POST /v1/trades/:id/commands/reject`
- `GET /v1/trades/:id` — projection / status
- `GET /v1/trades/:id/receipts` — receipts stream (optional)

PixPax issuance (packs)
- `POST /v1/packs/commands/issue` — writes `PACK_ISSUED`
- `POST /v1/packs/:id/commands/open` — writes `PACK_OPENED`
- `GET /v1/packs/:id/receipt` — canonical receipt(s) for pack issuance/open

Compatibility
- Existing endpoints remain during refactor with deprecation headers.
- New explicit command/query endpoints can be introduced under `/v2` if you prefer.