

zapier.georgelambert.org · Markdown indexes

VERAE TIME × ZAPIER

Billing and API keys

site/docs/01-product/billing-and-keys.md

Billing and API keys

Implementation: `packages/zappier`. Manuals: `packages/zappier/docs/`.

Keys (feature a)

- Customer signup at `/portal` issues an API key immediately (Free plan).
- Header: `x-api-key`.
- Dashboard can regenerate (invalidates the old key).
- Admin can provision customers.

Zapier connection test should use this key, not middleware `zmw_`.

Pricing (custom)

`openapi.yaml` `operationId` is the rate-card key. List prices are runtime-editable in `/admin`.

Billed cents = `round(list × tier multiplier)`; usage up to `monthlyCreditCents` is free. Per-customer `multiplierOverride` allowed.

Seed tiers: `free` (1.0, 100¢), `pro` (0.5, 1000¢), `business` (0.25, 10000¢).

After composition, rate-card keys become Verae operations (`timestamp`, `timestamp-wait`, `verify`, `status`, ...), not demo `transform` / `storage`.

After limit (feature b)

- Unlisted endpoint on free (no defaultRule) → 403.
- Credit/meter continues; Stripe job reports **delta** above monthly credit (`zapier.api_cents`).
- Portal **Reload balance** (\$1–\$10,000) prepaid drawdown on invoice issue.
- Admin **Invoices** tab for purchase-order customers.

Zapier must surface 402/403 with a link to `/portal` billing (same idea as middleware `upgradeUrl`).

Invoices and receipts (k, l)

Portal invoice list + print-ready HTML. Admin generate/issue/paid. These are **payment** documents.

Certified timestamp retrieval receipts (feature i) are a different object on the Verae adapter.

Payment method (j)

Portal Stripe reload is implemented. Card-on-file / Stripe Subscription objects for (m) are an open question; tiers today are customer types in SQLite.

Two invoices

Customers may pay Zapier for **tasks** and Verae/zapier for **timestamp usage**. See [zapier-billing-research.md](#). Do not meter Verae usage in Zapier task units.