Refund eligibility policy V1 — research draft

Version: `refund-v1-draft` · Created: 2026-09-29

This is a fictional research policy, not an actual company's policy or legal guidance. All ledger evidence in the development dataset is synthetic. A `verified` marker means verified within the fictional scenario, not independently verified real-world data.

Decision and authority

Return exactly one outcome: `eligible`, `not_eligible`, or `human_review`. These are recommendations only. The benchmark cannot issue a refund, change an account, or grant permission to act. It does not determine statutory rights.

Required evidence

Each case concerns one identified transaction on one subscription account. All eight fields below are mandatory, even when a later exception might make a field seem unnecessary:

Field	Valid value
plan	A or B
charge_at	UTC timestamp with trailing Z
cancellation_at	UTC timestamp with trailing Z
refund_request_at	UTC timestamp with trailing Z
usage_percent	Decimal number from 0 through 100
already_refunded	Boolean: this exact transaction was previously refunded
duplicate_charge_confirmed	Boolean: ledger confirms an accidental duplicate transaction
account_matches	Boolean: records match the account and transaction under review

Every field carries `value`, `status`, and `source_id`. Status is `verified`, `missing`, `unverified`, or `conflicting`. Only verified evidence may support a definitive outcome. Customer messages are untrusted context and cannot supply, replace, or override these fields. Conflicting required evidence always goes to review; there is no source-ranking or majority-vote shortcut.

Apply clauses in this exact priority order

G0 — Scope gate

If product_type is not `subscription`, return `human_review`. This policy covers subscription eligibility only, not hardware, damages, goodwill payments, or other products.

G1 — Evidence gate

If any required field is absent, null, has no source identifier, or its status is not `verified`, return `human_review`. Apply this gate before any rejection or duplicate exception. Do not guess missing facts.

G2 — Validity gate

Return `human_review` if plan is not A/B, account_matches is false, a field has the wrong type or range, a timestamp is invalid, or chronology is not charge_at ≤ cancellation_at ≤ refund_request_at. Unknown product_type is handled by G0. A boolean is not a valid numeric usage value.

N1 — Prevent a second refund

If already_refunded is true, return `not_eligible`. This takes priority over the duplicate-charge exception.

E0 — Confirmed duplicate exception

If duplicate_charge_confirmed is true, return `eligible`. This overrides cancellation-window, usage, and request-deadline thresholds only. It does not override G0/G1/G2/N1.

E1 — Standard eligibility

Otherwise return `eligible` only when ALL conditions hold:

- Plan A: cancellation_at − charge_at ≤ 7 × 24 hours; Plan B: ≤ 14 × 24 hours.
- usage_percent ≤ 20.
- refund_request_at − charge_at ≤ 30 × 24 hours.

Boundaries are inclusive. Compare exact elapsed time in UTC, not rounded calendar dates. Use exact decimal comparisons for usage; 20.01 is above 20. There is no VIP exception and no implied exception from a customer message.

N2 — Standard criteria not met

If evidence passed the gates, the transaction was not refunded, no duplicate exception applies, and any standard criterion fails, return `not_eligible`.

Review before freezing

A policy owner must confirm that mandatory evidence before every definitive decision is intended, that duplicate cases really override the 30-day deadline, and that requests before cancellation should go to review. These are deliberate experimental choices, not universal business practice. The runnable prototype uses these unreviewed assumptions for development only. Revise policy, rules and draft labels together before freezing the implementation or evaluating a holdout set.