Sandbox Operator's Manual

Funding controls, failure response and recovery runbook

Development-stage documentation. Payments run in test mode only; no real money moves. Live charges, live AI and real-money refunds are not enabled. This document describes approved policies and required handling, not a claim that all financial or operational controls are deployed.

Contents

Administrator sign-in security
Fee Schedule — ACH Direct Debit processing
Scope review, explicit acceptance and capped cost recovery
ACH deposits: bank-receipt funding lock
Read this before activating any paid test
Common incident response procedure
Recording and reviewing a fee assessment
1. All-inclusive Estimator, missing rates and spending limits
2. Access, identity and organization permissions
3. Payment authorization, processing and payout delays
4. Failed, cancelled, reversed or disputed payments
5. Duplicate or out-of-order payment notifications
6. Unavailable, changed or expired estimates
7. Insufficient balance, simultaneous requests and quota exhaustion
8. Top-up failure or missing consent
9. Network timeout, lost response and repeated submission
10. Provider outage, rate limit or partial model response
11. BLOCK, DENY, SUSPENDED or no consensus
12. Execution uncertainty, consumed permission or missing receipt
13. Cancellation, closing a test or account suspension
14. Pending, failed or delayed refunds
15. Fees, taxes, rounding and currency mismatches
16. Restart, database failure or integrity mismatch
17. Security incident, leaked credentials or malicious workload
18. Test/live mode, configuration or merchant-account mismatch
19. Unavailable features, email failures and insufficient capacity
Refund deadline and unresolved financial obligations
Release evidence and documentation maintenance

Administrator sign-in security

Administrator access requires an additional authenticator code. Open /console/mfa, generate a setup key, add it manually in Authy as Code of Trust, and confirm the current six-digit code. Save the single-use recovery codes securely. Verification lasts 15 minutes and is specific to your session. Used authenticator codes cannot be replayed; wait for the next code when needed. Replacing the authenticator requires a current code or unused recovery code and invalidates all existing MFA verifications. There is no email-only MFA bypass.

MFA factors are encrypted using the server SESSION_SECRET and stored with the Console authentication database. Preserve both securely during backup and recovery; changing the key without a migration blocks access. The current local SQLite store is not a verified durable, shared production

store. Do not claim production-wide MFA readiness until persistence across publishing, instance replacement and scaling has been verified. These controls do not configure MFA for hosting, source control, email or financial-provider dashboards, and do not establish that production Plaid Link is available.

Fee Schedule — ACH Direct Debit processing

Standard settlement: a per-transaction processing fee, calculated as a percentage of the transaction amount and capped at a maximum dollar amount. Two-day settlement uses a different processing rate where available. These are processing options, not a guarantee of receipt in the operating bank account.

Instant bank-account verification: a separate per-verification fee applies when instant bank-account verification is used.

Failed payments: a flat fee applies to a failed payment, including a returned debit, where assessed.

Disputed payments: a flat fee applies to a disputed payment where assessed. A dispute does not by itself establish misuse.

Rates, caps and flat amounts are pending confirmation. These categories are disclosures of processing costs, not authorization to collect an unspecified fee. Applicable amounts and conditions must be shown in the Estimator and Fee Schedule and accepted with the exact quote and Terms before payment.

Applicable processing costs must be included in the all-inclusive estimate; they must not be added again as hidden charges. Conditional failure or dispute costs must have their treatment disclosed before acceptance. Listing a cost does not authorize an automatic extra charge or a deduction from unused refundable funds. Existing documented-cost, review and no-double-recovery safeguards remain in force. Live payments remain disabled.

Scope review, explicit acceptance and capped cost recovery

Before payment, review the test scope, counts, token and concurrency limits, duration, expected price, maximum authorized spend, contingency reserve, cost-recovery caps and exact Terms version. Accept the specific quote and Terms together. Changes require a new review and acceptance. Unavailable prices are not zero-dollar quotes; no payment may begin without a valid priced quote and finalized accepted Terms.

The cost-recovery currency is USD. Cancellation recovery is limited to documented, nonrecoverable costs, capped at the lesser of $25 or 5% of the accepted base quote. Cancellation caused by the operator or a provider carries no cancellation fee. No incurred eligible costs means no recovery fee.

Verified misuse recovery is limited to documented incremental costs, capped at the lesser of $100 or 10% of the accepted base quote. Misuse means an evidenced violation of the agreed scope or access rules, not a legitimate DENY, BLOCK or HELD result. The tester must receive notice and an opportunity for operator review before an assessment.

Cancellation and misuse recovery together cannot exceed the lesser of $100 or 10% of the accepted base quote. The base excludes contingency reserves, taxes and other fees. Percentage caps round down to whole cents. Actual recovery cannot exceed eligible documented costs or the available

refundable balance. Never count the same cost twice, including costs already billed as usage.

These are disclosed cost-recovery limits, not automatic punitive charges or a forfeiture of the deposit. No negative balance or separate automatic penalty payment is authorized. A reserve is not automatically earned revenue. Itemize evidence and any approved deduction, and submit the remaining unused refundable funds within seven working days of closure or cancellation. An unresolved claim must not indefinitely extend that deadline.

The capped policy is approved for the proposed workflow; legal enforceability and finalized agreement wording still require review before paid launch. No actual fee collection is enabled. These terms cannot be applied retroactively to earlier accepted plans.

ACH deposits: bank-receipt funding lock

Stripe access now uses a direct server-side test API key, not the Replit Stripe connector. Use the same test environment as the registered webhook. Live API keys and provider write requests are rejected. Keep the signing secret separate from the API key; neither belongs in browser code or chat. Direct access, publication and signed test-event delivery must be verified before describing the receiver as connected end to end.

ACH preparation is test-only. No ACH checkout, real debit, bank connection or live payment has been enabled by this release. Bank debit authorization through Stripe is separate from accepting the Sandbox Terms; neither action unlocks testing.

Chargeable testing must remain locked while an ACH deposit is processing, settled only to Stripe, awaiting payout, or reported paid by Stripe without verified bank receipt. The specific deposit must be reconciled to funds actually received in the operator's operating bank account and sufficient available funding before work begins.

A payout can contain multiple deposits, fees and adjustments. Never assign the entire payout to every tester. Match the Stripe account, test/live mode, payment, balance transaction, payout, currency and net deposit allocation to independent bank-receipt evidence. Partial receipts or a shortfall after fees must not unlock a larger test reservation.

Duplicate or out-of-order events cannot create more credit or erase a later failure. Missing, stale, inconsistent, reversed, failed or disputed funding remains locked. A browser redirect, screenshot, manually entered status or payout.paid notification alone is not proof of bank receipt.

ACH can still be returned or disputed after funds reach a bank account. Bank receipt is not an irrevocability guarantee. Stop new affected work on reversal or dispute and reconcile existing reservations and refunds without duplicate reimbursement. Bank and payout delays must be displayed without promising an exact start date.

A test-only Stripe notification receiver is implemented. It requires a configured signing secret, verifies the event against the connected Stripe account, and records a minimal event receipt durably before acknowledgment. It never grants balance or marks funds settled. Real delivery remains unverified until endpoint configuration and deployment are completed. Before live launch, implement durable allocation and reconciliation, an independently verified bank-receipt source, and refund/return handling. Automated checks use synthetic evidence only; they are not a real ACH transaction or bank-integration test.

Read this before activating any paid test

Current working features are access review, organization roles, scoped keys, complimentary deterministic TEST fixtures and usage-plan persistence. Financial status helpers and documentation are not a substitute for an implemented verified payment pipeline.

Never unlock paid testing from a checkout redirect, captured payment, pending payout or manually asserted balance. Fully processed funding, actual availability and sufficient unreserved coverage are all required. A higher administrative privilege cannot waive the Gate.

Before launch, verify backups and restoration, complete the reviewed durable storage/authentication cutover, approve the rate card and finalized terms, test payment/refund reconciliation and deploy the isolated live-test environment. Do not publish to bypass these prerequisites.

Common incident response procedure

1. Contain: stop new affected work without erasing existing records. Use organization suspension or revoke compromised credentials where appropriate.

2. Identify: record the organization, plan, operation ID, event time, observed error and evidence location. Never include API tokens, payment credentials or personal payloads in an incident report.

3. Reconcile: compare durable app state with authoritative provider/target records. Preserve original idempotency keys, payment identifiers and execution receipts. Unknown is not successful or failed.

4. Communicate: provide the affected tester an accurate status and next step. Do not promise bank receipt timing, a resolution time or execution that cannot be verified.

5. Resolve and verify: apply the authorized correction, rerun integrity/access checks, record the outcome and resume only when safety gates pass. Preserve the original audit history.

Recording and reviewing a fee assessment

In the development Sandbox, open Plan & Fund, then Fee assessments beside the plan's Refund status. A Console operator can record the cancellation cause or independent misuse violation and select only trusted server cost records. Close a cancelled plan first; preserve its original closure date. No eligible incurred costs means waive or reject, never apply a flat fee.

The draft preserves immutable copies of the cost evidence and historical accepted fee package. Publish a portal notice that explains the itemization. This does not send email or prove external delivery. A different operator must record the review outcome and reason after notice. Missing acceptance, changed evidence, already-billed usage, and duplicate source costs cannot authorize recovery.

Organization members can dispute a noticed assessment, including a review already marked approved. A dispute blocks approval. An operator other than the proposer or disputant must record an upheld, rejected or waived resolution with a reason. Preserve all notices, decisions and disputes; no edit or delete action replaces their history.

Approval is a review outcome, not a deduction authorization. There is no trusted evidence-import or

financial-settlement adapter enabled here; browser amounts, uploads and administrator assertions do not certify costs or funds. Do not insert real cost records manually to bypass this gate. All money movement remains disabled.

Use the assessment ID and its linked organization/plan when reconciling the existing refund obligation or incident. Assessments do not create a second refund record, close an incident, reserve money or reset the seven-working-day refund deadline. Retain unresolved assessments alongside refund status for operator attention.

1. All-inclusive Estimator, missing rates and spending limits

Tester-facing rule: The Estimator is the quote for the selected Test or Live scope, starting at USD 1,500. It must include AI, Replit infrastructure, platform costs, payment processing, applicable taxes and at least 20% for EARTHX after included costs. Unknown costs are not zero. Until verified commercial rates exist, monetary quotes and payments remain unavailable. Complimentary TEST fixtures are separate. A starting price is not unlimited usage; no separate automatic system or overage charges are authorized. Optional refundable reserves are not earned revenue.

Operator response: Validate every required rate, its source, currency and validity before issuing a payable estimate. Include every Council seat, round and retry, plus fixed infrastructure and operating allocations. Round conservatively to whole cents and verify EARTHX retains at least 20% after processing and included costs. Never use refundable reserves as margin. Missing, expired or incomplete rates must block pricing. Bind accepted quotes to exact scope, Terms and limits; reserve in-flight costs atomically before paid work. Re-estimate and obtain fresh acceptance before additional funded usage. Planning controls do not enforce a live runner in this release.

2. Access, identity and organization permissions

Tester-facing rule: An invitation, payment or positive balance does not replace access approval. Expired sessions, revoked credentials, insufficient privileges and requests for another organization's records must be refused. Do not share credentials or evade limits using multiple identities.

Operator response: Confirm the authenticated identity, organization membership, current role and credential expiration. Revoke compromised keys, preserve the audit history and restore access only through the approved identity process. Never disclose another organization's data to diagnose a request.

3. Payment authorization, processing and payout delays

Tester-facing rule: Funds must be fully processed and available to cover costs before chargeable testing can begin. Checkout success, authorization, capture, a pending payment or a pending payout is insufficient. No unfunded overages are permitted.

Operator response: Keep testing locked until trusted server-side evidence confirms both fully processed funding and available funds. Do not accept screenshots, browser redirects, tester assertions or an administrator-entered balance as payment evidence. A missing verification signal means remain locked, not assume success.

4. Failed, cancelled, reversed or disputed payments

Tester-facing rule: Failed, cancelled, reversed or disputed funding cannot authorize new tests. A dispute must not result in a duplicate reimbursement. Applicable payment-provider and legal requirements still apply; these terms do not create an automatic forfeiture of unused funds.

Operator response: Stop new reservations against affected funding, reconcile the original payment and any refund/dispute, and preserve transaction identifiers. Follow the provider's dispute workflow. Do not issue another refund without checking whether the customer has already been reimbursed.

5. Duplicate or out-of-order payment notifications

Tester-facing rule: Repeated payment notifications must not grant duplicate credit. Unverified or inconsistent notifications must not unlock testing. A displayed success page is not proof of a settled deposit.

Operator response: Before paid launch, verify webhook signatures, merchant account, mode, amount, currency and immutable checkout intent. Apply events transactionally with unique event and payment identifiers. Reconcile out-of-order events against authoritative provider state; never treat their arrival order as financial truth.

6. Unavailable, changed or expired estimates

Tester-facing rule: Usage quantities are estimates until an approved rate card produces a valid quote. No price is implied by an unavailable or blank monetary field. Plan changes or expired quotes require recalculation and approval before funding or execution. Do not silently raise an accepted spending ceiling.

Operator response: Check quote version, expiry, model/rate snapshot, currency and configuration. Reject stale or conflicting updates. Validate all counts and monetary calculations on the server, including integer bounds, and flag a test window too short for the requested rate.

7. Insufficient balance, simultaneous requests and quota exhaustion

Tester-facing rule: New chargeable work must pause if available funding cannot cover its bounded reservation, or if an applicable count, token, time, rate or concurrency quota is exhausted. A prepaid contingency reserve is extra funded capacity, not permission for a negative balance.

Operator response: Use shared atomic reservations and quota checks across all instances. Already-funded work may finish only within its reserved limits. Reject unfunded starts, keep reserved amounts unavailable to other requests, and release only verified unused reservations. Do not assume a per-process counter is a shared limit.

8. Top-up failure or missing consent

Tester-facing rule: No automatic top-up may occur without separate explicit consent and defined limits. A failed top-up must not create credit or allow unfunded usage. Automatic top-ups are not enabled in the current release.

Operator response: When implemented, verify consent, amount, trigger, ceiling and funding status for each top-up. Pause new work when the top-up cannot fund it. Never fall back to an operator-funded charge or retry a payment indefinitely.

9. Network timeout, lost response and repeated submission

Tester-facing rule: A timeout or closed browser does not establish whether a test, charge or refund completed. Retrying the same operation must reuse its identifier. Starting a new operation can create a separate charge.

Operator response: Recover the saved operation and authoritative result before retrying. Keep uncertain operations visible. Use the same idempotency key and unchanged body; reject conflicting reuse. Do not silently abandon an unresolved request on refresh or create a replacement payment/refund.

10. Provider outage, rate limit or partial model response

Tester-facing rule: Provider outages, rate limits, incomplete responses and timeouts can prevent a result. A failed test must not be represented as ALLOW or proof of execution. Charging treatment for billable partial provider work requires a disclosed approved policy before paid launch.

Operator response: Bound time, tokens, calls and retries. Preserve which work occurred and which charges are evidenced. Do not settle an estimated provider charge as an actual charge or silently substitute models. Fail closed when required judgments or evidence are unavailable.

11. BLOCK, DENY, SUSPENDED or no consensus

Tester-facing rule: A blocked or suspended result is a governance outcome, not permission to bypass the Gate. An ALLOW judgment alone is not proof of execution. Current deterministic TEST fixtures cost one complimentary TEST credit per accepted run, regardless of verdict; paid-result pricing is not yet approved.

Operator response: Retain recorded reasons and permitted remediation. Distinguish correctable proposals from immutable restrictions. Requester, reviewer and administrator privileges do not waive immutable boundaries. Never label a fixed fixture as a live independent Council determination.

12. Execution uncertainty, consumed permission or missing receipt

Tester-facing rule: Authorization and execution are separate. A consumed permission without a verified receipt is an unresolved outcome, not permission to execute again. Only the action supported by verified evidence may be described as executed.

Operator response: Retrieve the original receipt and target-side result. Do not reissue or replay a one-use authorization automatically. The existing local test tool can consume permission before its separate record write; reconcile that failure explicitly. External execution remains unavailable in this Sandbox.

13. Cancellation, closing a test or account suspension

Tester-facing rule: Closure or cancellation stops new work. Reconcile funded work already in progress and preserve the unused refundable balance. Suspension does not silently forfeit unused funds. Refund submission is due within seven working days after closure or cancellation; reconciliation must not indefinitely postpone that deadline.

Operator response: Freeze new work, retain the closure timestamp and compute the submission deadline immediately. Identify in-flight reservations, reconcile evidenced usage and escalate unresolved items before the deadline. Do not delete the plan or its financial history to cancel it.

14. Pending, failed or delayed refunds

Tester-facing rule: Unused refundable funds must be submitted for refund to the original payment method within seven working days of closure or cancellation. Bank processing time is additional. Pending, rejected or failed refunds are not completed refunds. Actual refund processing is not enabled in this release.

Operator response: Track amount, original payment, submission deadline, submission identifier and provider status separately. Keep adequate refundable funds available. If balance is insufficient, the payment method is closed, a refund fails, or provider processing is delayed, retain the obligation and escalate. Do not issue a second reimbursement without reconciliation.

15. Fees, taxes, rounding and currency mismatches

Tester-facing rule: Prices, currency, fee treatment, taxes and credit-expiration terms require approval and disclosure before paid checkout. Do not silently deduct processing fees from promised unused-funds refunds. TEST credits have no cash value and cannot be converted into purchased funds.

Operator response: Use integer minor-unit accounting and the correct currency precision. Verify charge/refund currency and total against the accepted quote. Reconcile processor fees and taxes separately. Unapproved fees, currency conversion or rounding rules are release blockers, not discretionary adjustments.

16. Restart, database failure or integrity mismatch

Tester-facing rule: A service restart or storage failure must not reset balances, revive consumed permissions, grant access or turn an uncertain operation into a successful one. When authoritative state cannot be verified, chargeable operations remain blocked.

Operator response: Pause execution, preserve evidence and restore only from verified consistent backups. Reconcile accepted transactions after the backup before rollback. Check ledger hashes, ownership, reservations and payment/refund identifiers. An empty database must not bootstrap an unrelated visitor as administrator.

17. Security incident, leaked credentials or malicious workload

Tester-facing rule: Testing is limited to the approved sandbox, scope and capacity. Do not access other organizations, attack production, exfiltrate data or bypass controls. Credential compromise or harmful traffic may require suspension; payment does not authorize unrestricted activity.

Operator response: Suspend affected access, revoke credentials, stop new work and preserve relevant evidence. Avoid recording secrets or sensitive payloads in incident logs. Assess exposure and required notices with qualified personnel; restore access only after the incident is contained.

18. Test/live mode, configuration or merchant-account mismatch

Tester-facing rule: Test transactions and complimentary balances cannot fund live testing. A payment made in the wrong environment or merchant account must not activate a plan. Misconfigured integrations must stop affected work rather than fall back to another credential or billing source.

Operator response: Verify deployment mode, intended merchant account, provider credentials, routing and rate configuration before enabling a financial operation. Keep test and live identifiers and balances separate. Restore the reviewed configuration and reconcile any affected operations before resuming; never copy test credit into a paid balance.

19. Unavailable features, email failures and insufficient capacity

Tester-facing rule: This development release does not provide live AI testing, real payments, refund submission, automatic email delivery or an isolated production stress environment. Saving a plan does not book capacity or fund a test. No uptime or capacity guarantee is established by these terms.

Operator response: Use persisted access requests as the source of truth; do not assume notification delivery. Confirm feature readiness and an approved test window before access is activated. Measure capacity in isolation and stop tests at agreed thresholds rather than risking production availability.

Refund deadline and unresolved financial obligations

Compute the deadline from the original closure/cancellation timestamp in America/New_York. Exclude weekends and observed US federal holidays and apply approved additional closures. The deadline is the end of the seventh working date; bank receipt time is additional.

Record pending, submitted, settled and failed statuses distinctly. Check approaching deadlines and unresolved obligations each working day. Payment or accounting uncertainty does not justify silently deleting a refund obligation or extending its deadline.

In staging, Console administrators can use the Sandbox Administrator tab's incident queue to record an organization, plan and original payment references, then acknowledge, escalate or record a failure with an immutable note. Pending obligations are labelled unresolved. The queue flags submission deadlines within two working dates and after the due date, refreshing every 60 seconds while the browser tab is visible. Review it each working day; no email or external notification is sent.

A refund report requires a closed plan. The saved original closure supplies its deadline, even when the report is filed late. Operator-entered payment references remain unverified reports, not proof of funds or provider submission. Check the existing payment incident before creating another; the same provider, merchant account and payment cannot have a second refund obligation.

Acknowledgement, escalation and reported failure do not discharge a refund obligation or reset its deadline. Record the responsible operator or team when escalating, and contact them separately. Preserve the same request key when retrying an uncertain operation. History cannot be edited or deleted; add a correction note instead.

Before refund processing is enabled, implement and test operational alerts, provider-confirmed submission, duplicate prevention, sufficient refundable liquidity and a documented escalation path. This manual does not create those integrations.

Release evidence and documentation maintenance

Maintain evidence for each failure case: access/tenant isolation; processed-funds gating; concurrent reservation limits; idempotency; provider failures; cancellation and refund deadlines; compromised credentials; restarts and restoration. Label synthetic tests separately from real provider and production verification.

Both this manual and the Terms & Conditions are generated from the same failure-case inventory for their web pages and PDF downloads. Update that source when behavior changes. The existing Testing Manual remains the tester-facing how-to and links to these documents.

Direct the authorized purchaser to /console/sandbox/accept-terms before checkout. Each purchaser and organization needs its own current finalized acceptance. Missing, stale or draft-only acceptance blocks payment, including direct API calls. Preserve prior snapshots; do not replace acceptance with an administrator assertion or treat consent as payment authorization.