



## SDG BLOCKCHAIN ACCELERATOR

### Debugging and Testing Report

## 1. Project Information

- **Project Name:** KarbonLedger – *Streamline CETP instance*
- **Challenge & UNDP Office:** CETP Compliance & Climate Credits – UNDP India (SDG Accelerator, Cohort 1)
- **Report Version:** v1.0 (Preview testnet prototype)
- **Repos:** <https://github.com/KonmaORG/karbonUmbrella>

<https://github.com/KonmaORG/carbonica-ledger>

UNDP Instance: <https://karbonledgerundpinstance.vercel.app/>

## 2. Testing Approach

### Unit testing (Aiken contracts)

- Contract suite (**karbonumbrella**) compiled with **Aiken v1.1.17**, Plutus v3; CI runs `aiken fmt`, `aiken check -D`, and `aiken build` on pushes.
- Contract structure: validators (`carbonica_validator.ak`, `cet_minter.ak`, `cet_user_script.ak`, `config_datum_holder.ak`, `crowdfunding.ak`, `identification_nft.ak`, `marketplace.ak`) with shared types/utilities; tested with `aiken check` locally and in CI.
- Example invariants verified in specs: single-mint identification NFT; royalty split in marketplace; multisig approval on project validation.

### Integration testing (Preview testnet & app stack)

- **Blockchain events:** Blockfrost webhook → Next.js API `POST /api/cot-actions` → signature verification → `tokenFlow` parsing → Supabase write. Exercised using Blockfrost's webhook simulator and preview testnet txs.
- **Token lifecycle logic:** tested mint/transfer/burn paths and quantity checks via `TokenQuantityMismatchError`.
- **App roles & data model:** validator voting → approved → webhook-triggered mint; schema and role controls validated with mock flows.

### Edge cases exercised

- Invalid/missing Blockfrost signature rejected; missing secret rejected.
- Quantity mismatch on non-mint/burn transactions throws and surfaces 400 error.
- Burn with no external outputs correctly marks status `retired`; mints appear when token in outputs but not inputs
- Verify if the wallet address corresponds to the sign-in credentials.

### 3. Error Logs

- **E-001 – Signature header missing**

*Symptom:* 400 Missing signature header from webhook route.

*Trace:* Request without blockfrost-signature returns early.

*Resolution:* Ensure Blockfrost webhook sends the signature; added check in test harness.

- **E-002 – Token quantity mismatch**

*Symptom:* 400 Token Quantity Mismatch during transfer path.

*Trace:* tokenFlow throws when input ≠ outputs on non-mint/burn; caught and returned as 400.

*Resolution:* Fixed sample tx construction in tests to mirror exact sum of outputs (incl. self-return change).

- **E-003 – Unexpected event type**

*Symptom:* Unexpected event type: epoch logged; ignored gracefully.

*Trace:* Switch guards to only process transaction.

*Resolution:* Left as is; correct behavior.

### 4. Resolved Issues

Issue ID	Description	Root Cause	Resolution	Status
R-01	Webhook authentication failing	Secret not loaded in env during local runs	Guard + explicit error and .env checklist	<input checked="" type="checkbox"/> Fixed
R-02	False “burn” on self-return outputs	Return quantity not added before comparison	Added return_qty aggregation and action selection	<input checked="" type="checkbox"/> Fixed
R-03	Retire events not emitted when no external outputs	Edge case not mapped to retired	Added explicit retired action when outputs.length === 0	<input checked="" type="checkbox"/> Fixed

## 5. Optimization Notes

- **Event parsing** consolidated: built input/output index maps once per tx to reduce repeated scans when deriving flows.
- **DB writes** batched: `UpdateDB(actions, payload)` accepts array of derived actions to minimize round-trips.
- **Role & RLS** verified to avoid server-side overfetch; sensitive writes use service role only.

## 6. Tools and Environments Used

- **Aiken**: v1.1.17 (compiler & CI setup).
- **Cardano**: Preview testnet (network\_id 41 configured in Aiken project).
- **Blockfrost**: Webhooks + API for tx events; signature verification via `@blockfrost/blockfrost-js`.
- **App stack**: Next.js + Tailwind + ShadCN; Lucid Evolution; Supabase for DB & RLS; Netlify for FE hosting (per repo stack).
- **CI**: GitHub Actions with Aiken toolchain.

## 7. Remaining Issues / Next Steps

- Expand **contract-level tests** for crowdfunding state transitions and config datum updates (multisig paths) based on documented validators & utilities.
- Add **end-to-end demo**: project submission → 3/5 validator approvals → COT mint webhook → registry entry; this follows the defined workflow.
- Record **Preview tx hashes** in Supabase for the report appendix and link them in the Credits/Registry UI (UI already displays hashes and metadata).