



SDG BLOCKCHAIN ACCELERATOR

Prototype (PoC) Report – Template

1. Project Information

- **Project Name:** Blockchain-Enabled CRF Fund Disbursement System
- **Challenge & UNDP Office:** UNDP Bangladesh
- **Document Version:** V1

2. Project Overview

This prototype demonstrates a **hybrid cooperative loan management platform** that integrates **Cardano blockchain smart contracts** with **traditional finance systems** (mobile money/ fiat).

The solution enables:

- **Primary cooperative admins** → onboard beneficiaries, set loan terms, track repayments.
- **Secondary cooperative committees** → approve or review loan requests.
- **Beneficiaries** → apply for loans, receive disbursements (on-chain + fiat), and make repayments via mobile money.
- **Blockchain layer** → provides **transparency, immutability, and auditability** of loan agreements and repayments.
- **Fiat/MFS integration** → ensures end-users interact via familiar mobile wallets, while the cooperative backend syncs with blockchain records.

Main features:

- On-chain **loan contracts** and cooperative governance (Aiken validators).
- Off-chain **admin dashboard** for loan & user management.
- **Mobile money integration** for seamless disbursement and repayment.
- Role-based access: **Admin, Committee, Beneficiary**.
- DID support for **secure digital identities**.

3. Repository Structure

(Outline how your code and related files are organized.)

Suggested Structure:

```
/crf_app
```

```
/app
    /templates      → HTML frontend (loan dashboard, forms, status
pages)
    /blockchain_integration → Cardano/Plutus integration helpers
    /loan_manager.py   → Loan creation, disbursement, repayment logic
    /models.py        → Off-chain data models (members, loans,
repayments)
    /main.py          → Flask app entry point (API endpoints)
/deployed_contracts
addresses
/scripts           → Transaction build/sign/submit scripts
/docs             → README.md, deployment guide, architecture
notes
```

4. Build Instructions

```
# Install dependencies
pip install -r requirements.txt

# Set environment variables
export BLOCKFROST_PROJECT_ID=<your_blockfrost_key>
export NETWORK=preprod

# Run Flask backend
python main.py
```

5. Test Instructions

Ensure your virtual environment is activated. Then, run the main application file:

```
```bash
python app/main.py
```
```

You should see output indicating the Flask server is running, typically on [`http://127.0.0.1:5000/`](http://127.0.0.1:5000/).

Access the Frontend:

Open your web browser and navigate to `http://127.0.0.1:5000/`.

Expected outputs:

- Successful DID registration → returns DID string.
- Loan approval by committee → triggers on-chain contract validation.
- Repayment logging → hash stored on-chain, verified in DB

6. Deployment Instructions

Deploy contracts to Cardano testnet:

```
# Build transaction
cardano-cli transaction build \
--testnet-magic 1 \
--tx-in <UTXO> \
--tx-out <ADDRESS>+1000000 \
--change-address <CHANGE_ADDR> \
--out-file tx.raw

# Sign transaction
cardano-cli transaction sign \
--tx-body-file tx.raw \
--signing-key-file cooperative.skey \
--testnet-magic 1 \
--out-file tx.signed

# Submit transaction
cardano-cli transaction submit \
--testnet-magic 1 \
--tx-file tx.signed
```

Prerequisites:

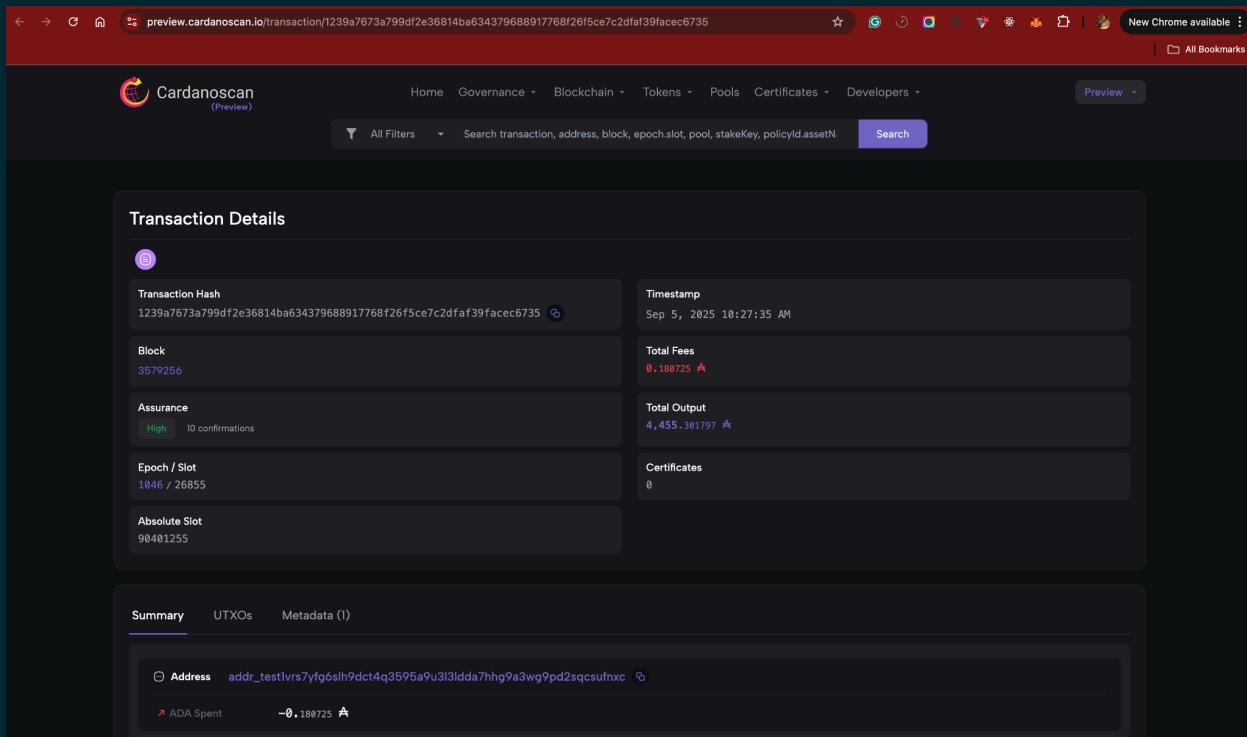
- Preprod ADA in funding wallet
- Running cardano-node synced with Preprod
- Blockfrost API key

7. Testnet / Emulator Results

| Transaction ID | Contract Action | Status | Notes | Transaction ID | Contract Action |
|----------------------|-----------------------|---------|--|----------------------|-----------------------|
| 1239a7673a799df2e... | Loan Disbursement | Success | Submitted via Flask → Blockfrost integration | 1239a7673a799df2e... | Loan Disbursement |
| c8736511fc51d3... | Loan Contract Created | Success | Generated conceptual Plutus script | c8736511fc51d3... | Loan Contract Created |

The screenshot shows a web browser window with the URL `127.0.0.1:8000`. The title bar reads "Cladfy - LoGIC CRF App" and "Demo of the Cardano Blockchain Interaction (v1)". The top right corner shows "Network Status" with a green dot indicating "Preprod (Epoch 1046)". The main content area has a purple header "Transactions". Below it is a table titled "Transaction History" with columns: TX HASH, TYPE, AMOUNT (ADA), STATUS, DATE, and ACTIONS. Two transactions are listed:

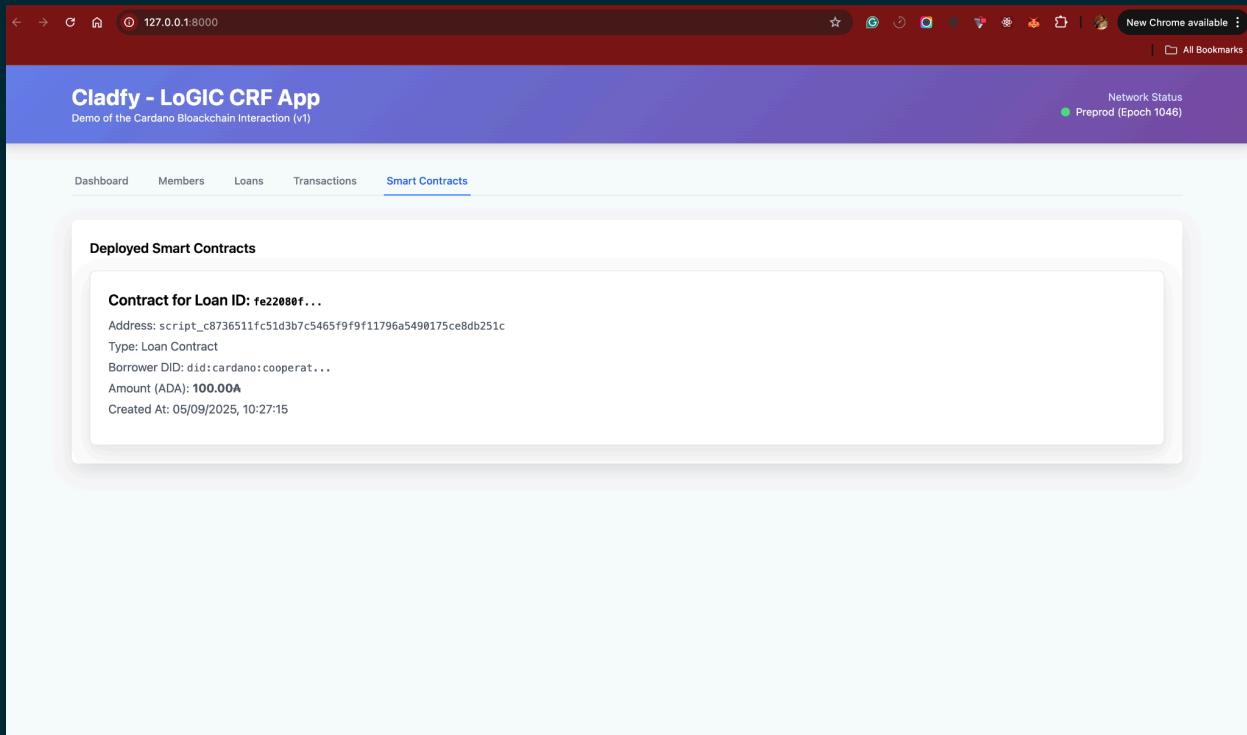
| TX HASH | TYPE | AMOUNT (ADA) | STATUS | DATE | ACTIONS |
|-----------------------------------|--------------|--------------|-----------|----------------------|--|
| 1239a7673a799df2e36814ba634379... | Disbursement | 100.000000A | Submitted | 05/09/2025, 10:27:24 | View on CardanoScan Refresh Status |
| 223961303234356539636263653639... | Repayment | 20.000000A | Submitted | 05/09/2025, 10:33:32 | View on CardanoScan Refresh Status |



The screenshot shows the CardanoScan transaction details page for a specific transaction hash. The page has a dark theme with a red header bar. At the top, there are navigation links for Home, Governance, Blockchain, Tokens, Pools, Certificates, Developers, and a Preview dropdown. Below the header is a search bar with a 'Search' button. The main content area is titled 'Transaction Details' and contains several sections with information:

- Transaction Hash:** 1239a7673a799df2e36814ba634379688917768f26f5ce7c2dfa39facec6735
- Block:** 3579256
- Assurance:** High (10 confirmations)
- Epoch / Slot:** 1046 / 26855
- Absolute Slot:** 90401255
- Timestamp:** Sep 5, 2025 10:27:35 AM
- Total Fees:** 0.180725 ADA
- Total Output:** 4,455,301797 ADA
- Certificates:** 0

Below this, there are tabs for Summary, UTXOs, and Metadata (i). The Summary tab is active, showing a single output address: addr_testlvr... with a spent amount of -0.180725 ADA.



The screenshot shows the Cladfy - LoGIC CRF App dashboard, running on a local host at 127.0.0.1:8000. The interface has a purple header bar with the app name and a 'Demo of the Cardano Blockchain Interaction (v1)' message. On the right, there is a 'Network Status' indicator showing 'Preprod (Epoch 1046)'. The main content area has a light gray background and includes the following elements:

- A navigation bar with links: Dashboard, Members, Loans, Transactions, Smart Contracts (which is the active tab).
- A section titled 'Deployed Smart Contracts' containing a card with the following details:
 - Contract for Loan ID:** fe22080f...
 - Address:** script_c8736511fc51d3b7c5465f9f11796a5490175ce8db251c
 - Type:** Loan Contract
 - Borrower DID:** did:cardano:cooperat...
 - Amount (ADA):** 100.00A
 - Created At:** 05/09/2025, 10:27:15

8. Dependencies & Environment

Dependencies:

```
Flask
blockfrost-python
pycardano
requests
cbor2
python-dotenv
```

Environment variables:

```
BLOCKFROST_PROJECT_ID="*****"
COOPERATIVE_TEST_ADDRESS="*****"
```

9. Demo / Walkthrough

- Register a new cooperative member → DID created
(`did:cardano:cooperative:....`)
 - Submit a loan application → conceptual Plutus contract generated
 - Approve and disburse → Cardano Preprod transaction hash returned
 - Dashboard shows loan status and repayment records
-
- Video Walkthrough: <https://www.awesomescreenshot.com/video/43887999>
 - See screenshots of the Beneficiaries' access web app below.
 - To access the beneficiaries' web app, visit <https://logic-crf.cladfy.app>

The screenshot shows the CRF-LoGIC application dashboard. On the left is a dark sidebar with navigation links: Dashboard, My Loans (with sub-links: My Loans, Apply New Loan, Loan Calculator), Transfer Money, Deposit Money, Withdraw Money, Transaction Requests, and Reports. The main content area has a header with a close button, language selection (English), user profile (John), and a bookmark section. The dashboard is divided into sections: 'Accounts Overview' (table with one row: Account Number MS12, Account Type Member Savings, Currency BDT, Balance b250,000.00, Loan Guarantee b0.00, Current Balance b250,000.00), 'Upcoming Loan Payment' (table with one row: Loan ID CCA2, Next Payment Date 05/Sep/2025, Status Upcoming, Amount to Pay b2,575.00, Action Pay Now), and 'Recent Transactions' (table with one row: Date 29/Aug/2025 02:41 PM, AC Number MS12 - Member Savings (BDT), Amount + b250,000.00, DR/CR CR, Type Deposit, Status Completed, Action View). At the bottom, it says 'Showing 1 to 1 of 1 Entries'.

10. Remaining Issues / Next Steps

- **Persistence:** Move from in-memory storage → PostgreSQL
- **Multi-Cooperative Support:** Add registry for multiple co-ops and committees
- **Smart Contracts:** Improve contract validators
- **Integrations:** Link with the current Multi-Cooperative management system
- **Governance:** Add DAO-style voting in Phase 2
- **Security Audit:** Implement HSM/multi-sig for cooperative key management.
- **Scaling:** Benchmark disbursement throughput and transaction confirmation delays.