SDG BLOCKCHAIN ACCELERATOR

Debugging and Testing Report – Genius Tags

1. Project Information

- **Project Name:** ClimateAid

- **Challenge & UNDP Office:** Malawi

- **Report Version:** V2

2. Testing Approach

Unit Testing

- **Aiken Unit Tests**: Comprehensive test suite written in Aiken covering all validator logic
    - Test coverage includes organization management, project creation, user permissions, and transaction validation
    - 685 lines of test code in `comprehensive-tests.ak`
    - Tests cover edge cases for invalid datum/redeemer, unauthorized signatures, and permission validation
    - Test structure follows Aiken best practices with proper test data setup and validation

Integration Testing

- **Offline Contract Tests**: Validated contract compilation and script generation without blockchain interaction
- **Blockchain Integration Tests**: Prepared for Cardano Preview Testnet deployment (requires Blockfrost API key)
- **Mesh SDK Integration**: TypeScript tests using @meshsdk/core for blockchain interaction

Edge Cases Tested

- Transaction with missing redeemer validation
- Transaction with invalid datum structure
- Unauthorized user access attempts
- Insufficient permissions for operations
- Double spend prevention
- Invalid organization/user registration
- Project sharing permission validation

3. Error Logs

- **Error 1:** Insufficient funds for lovelace … Required: 2.5 ADA, Available: 0 ADA

- **Error 2 :** OutputTooSmall: The output value is less than the minimum required

- **Error 3:** Transaction hex is undefined before signing

4. Resolved Issues

(List issues found and how they were resolved. Use a table if needed.)

Error 1: Insufficient Funds
- **Error:** Insufficient funds for lovelace … Required: 2.5 ADA, Available: 0 ADA
- **Cause:** No UTXO selection before transaction completion
- **Resolution:** Added .selectUtxosFrom(await wallet.getUtxos())
- **Status:** ✅ Fixed

Error 2: Output Too Small
- **Error:** OutputTooSmall: The output value is less than the minimum required
- **Cause:** Manual UTXO selection not accounting for min ADA
- **Resolution:** Let Mesh SDK handle UTXO selection automatically
- **Status:** ✅ Fixed

Error 3: Undefined Transaction Hex
- **Error:** Transaction hex is undefined before signing
- **Cause:** Signing transaction hash instead of unsigned CBOR
- **Resolution:** Use txBuilder.complete() for unsigned hex, then sign
- **Status:** ✅ Fixed

5. Optimization Notes

Script Performance Optimizations

- **Script size**: 107 bytes (✅ Excellent - well under 16KB limit)
- **Script efficiency**: Optimized for minimal execution units
- **Validator structure**: Refactored for readability and maintainability

Code Quality Improvements

- **Type safety**: Strong typing with Aiken's type system

- **Error handling**: Comprehensive validation for all inputs
- **Documentation**: Extensive inline documentation and comments
- **Test coverage**: 685 lines of comprehensive test code

Performance Metrics

- **CPU usage**: Optimized for minimal execution cost
- **Memory usage**: Efficient data structures using Aiken's built-in collections
- **Script hash**: 970ec4df1e9f7bb2346d336575208db7ff5cb6e3e450d85c83869c27

6. Tools and Environments Used

Development Tools

- **Aiken CLI**: v1.1.19+e525483 (`aiken check`, `aiken build`)
- **Node.js**: Latest LTS version
- **TypeScript**: For blockchain integration tests
- **Mesh SDK**: @meshsdk/core v1.9.0-beta.2 for Cardano interaction

Testing Environments

- **Local Development**: macOS 24.6.0
- **Blockfrost**: For blockchain data access

Build Tools

- **tsx**: For TypeScript execution
- **dotenv**: For environment variable management
- **npm**: Package management

7. Remaining Issues / Next Steps

- *Critical Issues*

- ***Move Blockchain code to run on client***

- ***Impact***: *Better security by letting each client act with their own wallet*

- ***Solution***: *Refactor legacy frontend codebase to be able to use up to date Cardano tools.*

- *Pending Improvements*

- ***Blockchain Integration***: *Complete testnet deployment testing*

- ***Performance Optimization***: *Further script size optimization*

- ***Security Audit***: *Comprehensive security review*

- ***Next step***: *Professional security audit before mainnet deployment*

- *Planned Next Steps*

- *Security audit and mainnet preparation*

- *Production deployment and monitoring setup*