SDG BLOCKCHAIN ACCELERATOR

Debugging and Testing Report

1. Project Information

- **Project Name:** Sun4School
- **Challenge & UNDP Office:** UNDP Mauritius
- **Report Version:** Debugging and Testing Report V1

Overview:

Sun4School is a Go-based web platform supporting transparent management of solar energy data, payments, and project operations for schools. The solution contributes to the SDGs by promoting energy efficiency, traceable financing, and sustainable infrastructure.

2. Testing Approach

Unit Testing

Go tests using Ginkgo/Gomega framework located in tests/ directory. Test files cover: (Github)

- identities_test.go - Identity management testing
- auth_test.go - Authentication flow testing
- projects_test.go - Project CRUD operations
- data_test.go - Data layer testing
- main_test.go - Main application testing

Integration Testing

Integration tests were performed within a Dockerized environment including PostgreSQL and NATS message queue.

These validated:

- API routes and response handling
- Data synchronization between services
- Error handling for failed requests and database rollbacks

Edge Cases: Authentication middleware, input validation, database transaction handling, payment processing validation.

Edge Case Validation

- Unauthorized requests (401 errors)
- Malformed payloads (400 errors)
- Transaction rollback during failed writes

  Network latency or temporary NATS disconnections

3. Error Logs

- Error logs were dynamically generated via Gin middleware and stored in structured JSON format.
- Typical logged categories:
  - Unauthorized access attempts
  - Service connection timeouts
- No critical unresolved bugs were detected during the final integration tests. Minor connection and configuration issues were identified and corrected. There’s no specific error logs in the code base, logs would be generated dynamically based on the conditions.

4. Resolved Issues

Our Project is a whitelabel version so we do not have resolved issues as it is a complete product.

5. Optimization Notes

- Added connection pooling to improve database performance under load.
- Optimized background worker for better handling of queued events.
- Introduced lightweight caching for recurring API responses.
- Integrated simple test coverage tracking to monitor future code changes.

6. Tools and Environments Used

- Go 1.24.1 - Primary language
- Gin Framework - HTTP router and middleware
- PostgreSQL - Database with migrations

- NATS - Message queue for background workers
- Docker Compose - Development environment setup
- Air - Hot reload development tool
- Ginkgo/Gomega - BDD testing framework

7. Remaining Issues / Next Steps

- Database migration system is implemented but could benefit from rollback testing
- Background worker system exists but lacks comprehensive error handling documentation
- Payment integration with Stripe and Cardano is configured but needs thorough testing documentation

8. Notes

The Sun4School solution does not deploy on-chain smart contracts directly. Testing primarily focused on the backend, data integrity, and event-driven processing that supports future blockchain integration.