



ANASTASIA LABS

Anastasia Labs X Maestro - Plug 'n Play 2.0

Project Close-Out Report

Project Number	1100025
Project manager	Maestro Team & Anastasia Labs
Date Started	May 23, 2024
Date Completed	April 21, 2025

Contents

List of KPIs	1
Challenge KPIs	1
Project KPIs	2
Key achievements	3
Key learnings:	4
Next steps	5
Final thoughts/comments:	6
Resources	6
Project	6
Smart Contracts	6
APIs	6
Closeout Video	6

Plug n' Play 2.0 : Project Closeout Report

URL: [Catalyst Proposal](#)

List of KPIs

Challenge KPIs

1. Enabling Upgradable Multisig Management:

We built and deployed an **Aiken Upgradable Multisignature Smart Contract** that allows dynamic **addition/removal** of signers and **threshold adjustments** without redeployment, enhancing on-chain governance flexibility.

2. Seamless Subscription Payments:

We designed and implement a **Subscription Payment Smart Contract** leveraging wallet-owned UTXOs to automate recurring payments trustlessly, improving user experience and on-chain billing reliability.

3. No-Code Deployment Platform:

By doing the above, we extend the Plug-and-play Smart Contracts platform to support both Upgradable Multisig and Subscription flows via **simple API calls**, reducing integration complexity for developers unfamiliar with Aiken.

Project KPIs

1. **Contract Development & Testing:**

Designed, Author and implemented rigorous unit-test, property based and integration tests for two core Aiken contracts (upgradable-multisig and subscription-payments) with $\geq 95\%$ code coverage.

2. **Offchain SDKs & API Routes:**

Deliver fully documented SDKs in TypeScript and Aiken, plus RESTful API endpoints on Maestro, with end-to-end integration tests passing in CI.

3. **Documentation Quality & Adoption:**

Publish comprehensive developer guides and tutorials; achieve ≥ 200 unique GitHub stars and ≥ 50 SDK downloads within first month of release.

4. **Increased Developer Adoption:**

Monitor API call volume, unique API consumers, and community feedback; target $\geq 1,000$ API calls and ≥ 20 integrated projects by completion.

Key achievements

1. **Milestone Acceptance & Evidence:**

All milestone 1–4 acceptance criteria (secure asset spending, threshold updates, dynamic signer management, and recurring payment flows) were met as documented in the repository’s dedicated sections. Comprehensive evidence—including unit tests, coverage reports, design specs, and live transaction logs—is available on GitHub.

2. **Upgradable Multisig Contract:**

Designed and deployed an Aiken-based multisignature contract; validated secure execution by authorized members, seamless signer threshold adjustments, and dynamic addition/removal workflows in on-chain tests.

3. **Subscription Payments Contract:**

Delivered an Aiken payment subscription contract enabling users to set up, manage, and cancel recurring payments directly from wallets; demonstrated UTxO chaining and fee bumping in unit tests and integration scenarios.

4. **Offchain SDKs & API Integration:**

Released TypeScript and Aiken SDKs per design specifications; all endpoints implemented on Maestro with CI-integrated end-to-end tests covering normal, edge, and error cases.

5. **Integration with Maestro:**

- We successfully integrated our smart contracts and SDKs with Maestro, enabling developers to interact with our contracts through Maestro’s platform. This integration provides a streamlined interface for deploying and managing smart contracts, further lowering the barrier to entry for developers.

6. **Extensive documentation and tutorials:**

- Published clear, high-level guides, design documents, and code samples; integration tutorials and video walkthroughs complete the developer onboarding experience.

Key learnings:

- **Importance of Modular Design and Security:**

Our project reaffirmed the value of modular, reusable smart contracts for streamlining development and enhancing security. We learned that building our contracts and SDKs as composable, modular components allowed us to reuse common patterns across features, accelerating development and simplifying maintenance.

- **Strength of Aiken for On-Chain Logic:**

We found that Aiken's high-level syntax improved code clarity and testability compared to lower-level languages, streamlining our debugging process and increasing our confidence in code coverage.

- **Comprehensive Testing Paradigms:**

By implementing both unit tests for individual contract functions and integration tests for API endpoints, we exposed critical edge cases early—underscoring the value of having end-to-end test automation integrated directly into our CI pipelines.

- **Importance of Comprehensive Documentation and Monitoring:**

Creating thorough documentation and linking user-friendly tutorial videos is vital for effective developer onboarding. Additionally, implementing monitoring tools for key metrics prepares us to make informed, data-driven improvements, helping guide future development based on real-world usage patterns.

Next steps

1. **Expand Smart Contract Library:** Continue developing and integrating additional smart contracts based on developer feedback and ecosystem needs. This will ensure the platform remains relevant and valuable to the growing Cardano developer community.
2. **Enhance API Documentation:** Improve the API documentation with more examples, use cases, and best practices to make it even easier for developers to integrate and utilize the smart contract services.
3. **Optimize Performance and Scalability:** Monitor the performance of the smart contract APIs as usage grows and make necessary optimizations to ensure high availability and responsiveness, even under heavy load.
4. **Collaborate with Ecosystem Partners:** Engage with other Cardano ecosystem projects and partners to explore opportunities for integrating the smart contract APIs into their platforms and services, further expanding the reach and adoption of the solution.
5. **Continuously Gather Feedback:** Maintain an open dialogue with the developer community, regularly soliciting feedback and suggestions for improvement. Use this input to guide future development and ensure the platform remains aligned with the evolving needs of Cardano builders.

Final thoughts/comments:

We are proud to have introduced a flexible, secure, and modular smart contract library to the Cardano community. Our team's commitment to best practices and developer experience has laid a strong foundation for future growth.

We believe that these contracts and APIs will empower builders to focus on innovation rather than boilerplate. We look forward to seeing the next generation of dApps leverage Plug n' Play 2.0 and driving further ecosystem expansion.

Moving forward, we remain dedicated to supporting the developer community and adapting our offerings to meet their needs, ensuring our library contributes meaningfully to the Cardano ecosystem.

Resources

Project	Smart Contracts	APIs
Catalyst Proposal	Single Asset Staking	Single Asset Staking API
Maestro Platform	Linear Vesting	Linear Vesting API
Main Github Repo	Direct Offer	Direct Offer API

Closeout Video

[Plug-and-play-01 API -Closeout Video](#)