



ANASTASIA LABS

Anastasia Labs X Maestro - Plug n' Play-02

Project Closout Report Script

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

Contents

Introduction	1
Project Context and Importance	1
Project Objectives	2
Execution and Milestones	2
Phase 1: Aiken Contract Development	2
Phase 2: Off-Chain SDKs Building	3
Phase 3: Integration Phase: Smart Contract API Integration	4
Phase 4: Smart Contract API Testing	5
Phase 5: Documentation and Community Engagement	5
Achievements and Outcomes	6
Key Learnings and Challenges	7
Future Prospects and Community Impact	8
Conclusion	8

Anastasia Labs X Maestro - Plug n' Play-02

Slide 1

Introduction

Hello, Cardano builders!

Welcome to the close-out report for our Catalyst Fund 11 project:

Slide 2

“Plug 'n Play 2.0, a collaboration between Anastasia Labs and Maestro.”

“I’m [Name], and today we’ll recap our journey: why we built Plug 'n Play 2.0, how we executed each phase, our key outcomes, lessons learned, and what comes next.”

Slide 3

Project Context and Importance

“The Cardano ecosystem boasts rapid growth, but deploying secure smart contracts often demands deep Plutus/ Haskell or Aiken expertise. That complexity slows innovation. Plug 'n Play 2.0 set out to change that, offering ready-made, secure and robust contracts with a no-code interface.”

Slide 4

Project Objectives

Our four objectives were:

- Architect upgradable multisig and subscription contracts in Aiken;
- Wrap them in intuitive TypeScript SDKs;
- Publish managed REST endpoints on Maestro;
- and back it all with top-notch docs, tests, and walkthrough videos.

Slide 5

Execution and Milestones

We structured the project into five phases.

- Phase 1 We tackled contract design and unit tests.
- Phase 2 produced SDKs for off-chain integration.
- In Phase 3 we wired everything into Maestro's REST APIs.
- Phase 4 validated end-to-end flows under all edge cases. Finally,
- Phase 5 where we are wrapping up with user guides and this project close-out film.

Slide 6

Phase 1: Aiken Contract Development

In Phase 1, we authored two on-chain contracts in Aiken.

- The multisig contract enforces multi-signature spending, supports live threshold changes, and dynamic signer add/removal—no redeploy needed.

- The subscription contract chains UTxOs for trustless, recurring payments, and even supports optional penalty-fees for early cancellations; every scenario covered by unit- and property-based tests.

Slide 7

Upgradable Multisig Smart Contract:

The Aiken Upgradable Multisig Smart Contract is a powerful tool for managing digital assets on the Cardano blockchain.

It enables authorized members to execute asset transactions and updates within predefined thresholds.

It allows for secure spending of assets, seamless adjustment of signer thresholds, and dynamic addition or removal of signers for enduring usability.

Slide 8

Payment Subscription Smart Contract:

The Subscription contract is designed to facilitate recurring payments on the Cardano blockchain. It allows users to set up automated, scheduled transactions, ensuring timely and consistent payments without manual intervention. This feature is particularly useful for subscription-based services, memberships, and any scenario requiring regular payments.

Slide 9

Phase 2: Off-Chain SDKs Building

Phase 2 focused on user-friendly SDKs, it exposed functions like `createService()` for merchants, `createAccount()` for users, `initiateSubscription()` to start, `unsubscribe()` to cancel with optional penalty, and `withdrawFees()` for merchants.

The multisig SDK provides `initiateMultisig()` to set up wallets, `validateSign()` for partial signatures, `validateUpdate()` for adding/removing signers, and `endMultiSig()` to close out. Each method is backed by clear TS types and unit tests.”

Slide 10

Phase 3: Integration Phase: Smart Contract API Integration

In Phase 3, we mapped SDK calls to Maestro's managed REST API.

For example,

merchants hit POST /subscription/create-service with service-fee details and interval length.

Subscribers call POST /subscription/initiate with service and account IDs.

For multisig, teams use POST /multisig/initiate and POST /multisig/sign.

We published interactive docs and Postman IDs for easy exploration.

Slide 11

Phase 4: Smart Contract API Testing

We conducted thorough testing for each smart contract in our library to ensure their robustness. This rigorous process involved comprehensive code reviews and unit tests, all integrated into our continuous integration/continuous deployment (CI/CD) pipeline. This testing was essential to validate the functionality, reliability, security and performance of our solutions.

Slide 12

Phase 5: Documentation and Community Engagement

We provided detailed and user-friendly, comprehensive documentation and tutorials to facilitate easy adoption and implementation of our smart contracts. The documentation was designed to cater to developers of all skill levels, ensuring a smooth onboarding process.

Slide 13

Here's an example execution demo of the Aiken Upgradable Multisig Contract

Slide 14

Detailed guides and practical examples of our Single Asset Staking Contract can be found in the following links:

- Plutarch Contract: <https://github.com/Anastasia-Labs/aiken-upgradable-multisig>
- SDK: <https://github.com/Anastasia-Labs/aiken-multisig-offchain>
- API: <https://www.gomaestro.org/smart-contracts>

Slide 15

Let's have a look at an execution demo of the Payment Subscription Contract

Slide 16

To follow up on more details, guides and practical examples of our Payment Subscription Contract can be found in the following links:

- Plutarch Contract: <https://github.com/Anastasia-Labs/payment-subscription>
- SDK: <https://github.com/Anastasia-Labs/payment-subscription-offchain>
- API: <https://www.gomaestro.org/smart-contracts>

Slide 17

Achievements and Outcomes

We're pleased to highlight the key achievements of our project, which provide a strong base for further innovation in the Cardano ecosystem:

- **Implementation of Secure and Modular APIs:** Our team delivered secure, modular, and reusable APIs for the Aiken Upgradable Multisig Smart and the Payment Subscription Contract. These APIs simplify the development process by reducing the time needed to build and deploy smart contracts, allowing developers to focus on more complex, application-specific logic.
- **Quality Assurance:** We prioritized the robustness and reliability of our smart contracts by conducting thorough code reviews and comprehensive unit tests. This extensive testing ensures our contracts are secure, reliable, and ready for real-world deployment.
- **Seamless Integration:** To further ease development, we successfully integrated our smart contracts with Maestro, providing a unified interface for contract deployment and management. This integration simplifies access and interaction with our contracts, making the development process more efficient.

Slide 18

- **Extensive Documentation and Tutorials:** As demonstrated, we are proud to have enriched the Cardano community with rich resources by providing comprehensive documentation and tutorials to help developers quickly understand and utilize our solutions.
- **Industry Collaboration and Community Engagement:** We are happy to have collaborated with Maestro, a Key player in the Cardano ecosystem who also deployed both Contracts on Maestro platform.

Slide 19

Key Learnings and Challenges

Throughout the project, we faced several challenges that provided us with valuable insights. Here are the key learnings we've gathered:

- **First, Modular Design and Security.** We've seen the effectiveness of modular, reusable smart contracts. These contracts not only simplify the development process but also enhance security. By providing pre-built, battle-tested contracts, we allow developers to focus more on innovation. Rigorous testing has ensured these contracts are robust against vulnerabilities.
- **Second, Simplifying Integration.** We've learned that simplifying the integration process through APIs is crucial. It lowers the barriers for developers, whether they're part of small teams or large enterprises. This simplicity is key to fostering ecosystem growth and encouraging more developers to build on Cardano.
- **Third, Balancing Customization with Standardization.** Striking a balance between offering customizable solutions and maintaining standardized interfaces is essential. This balance provides the necessary flexibility while ensuring consistency, which is critical to cater to a broad range of developer needs.
- **Lastly, Comprehensive Documentation and Monitoring.** Detailed documentation and user-friendly tutorials are vital for effective developer onboarding. Additionally, implementing monitoring tools for API usage helps us understand user needs and make data-driven improvements. This ensures we can continually refine our offerings to better serve the community.

Slide 20

Future Prospects and Community Impact

- **Expand Smart Contract Library:** We aim to develop and integrate additional smart contracts, driven by developer feedback and evolving ecosystem needs. This will ensure that our platform remains a valuable resource for the growing Cardano developer community.
- **Enhance API Documentation:** We will improve our API documentation by adding more examples, use cases, and best practices, making it even easier for developers to utilize our services effectively.
- **Optimize Performance and Scalability:** As the use of our smart contract APIs grows, we will monitor performance closely and optimize to ensure high availability and responsiveness, even under heavy load and as the Cardano ecosystem evolves.
- **Collaborate with Ecosystem Partners:** We plan to engage with other projects and partners within the Cardano ecosystem to explore integration opportunities, expanding the reach and adoption of our smart contract APIs.
- **Continuously Gather Feedback:** We will maintain an open dialogue with the developer community, regularly soliciting feedback and suggestions. This input will guide our future development, ensuring we continue to meet the needs of Cardano builders.
- **Long-Term Commitment:** We will keep our libraries up-to-date with the ever-evolving Cardano ecosystem, ensuring our data structures remain relevant. By maintaining the open-source approach and transparent development process, we believe will inspire confidence and trust within the community.

Slide 21

Conclusion

In conclusion, we're proud to have introduced the 'Plug n' Play 02' smart contract library to the Cardano community. Together with Maestro, we have managed to develop secure, modular, and reusable smart contracts, create comprehensive off-chain SDKs, integrated with API endpoints and provide extensive documentation, ultimately supporting the creation of no-code DApps, making it easier for a wider range of users to build on Cardano.

These achievements enhance the capabilities of DApps on Cardano, offering developers valuable tools to innovate and secure their applications. The no-code DApps feature,

in particular, opens up blockchain development to those without programming skills, democratizing access to this technology.

We believe our work will inspire further innovation and drive adoption within the Cardano ecosystem. We're committed to supporting the developer community and continually improving our offerings to meet their evolving needs, ensuring that our library remains a vital resource for the platform's growth.

Thank you for your support and interest in our work.

Slide 22

For more information on the project or Anastasia Labs, visit our GitHub repository at:

<https://github.com/Anastasia-Labs/plug-n-play-contracts>

If you want to know more about Anastasia Labs or contact us, you can visit

- Our website at <https://anastasialabs.com/>
- Maestro Platform <https://www.gomaestro.org/smart-contracts>
- Follow us on twitter at <https://x.com/AnastasiaLabs>
- Join our discord community: <https://discord.com/invite/8TYSgwthVy>

Slide 23

See you next time!

Thank You and Goodbye!