



**ANASTASIA LABS**

## **Project Closout Report Script**

<b>Project Number</b>	1000013
<b>Project manager</b>	Jonathan Rodriguez
<b>Date Started</b>	2023-10-08
<b>Date Completed</b>	2024-03-31

## Contents

<b>Introduction .....</b>	<b>1</b>
Project Context and Importance .....	1
<b>Project Objectives .....</b>	<b>1</b>
<b>Execution and Milestones .....</b>	<b>2</b>
Phase 1: Design & Development .....	2
Phase 2: Thorough Testing .....	3
Phase 3: Comprehensive Documentation .....	3
<b>Achievements and Outcomes .....</b>	<b>4</b>
<b>Key Learnings and Challenges .....</b>	<b>5</b>
<b>Future Prospects and Community Impact .....</b>	<b>5</b>
<b>Conclusion: .....</b>	<b>6</b>

# **“The Trifecta of Data Structures: Merkle Trees, Tries, and Linked Lists for Cutting-Edge Contracts”**

## **Slide 1**

### **Introduction**

Hello, Cardano community! I’m Mladen Lamesevic from Anastasia Labs.

I’m excited to present the closeout report for our project, “The Trifecta of Data Structures: Merkle Trees, Tries, and Linked Lists for Cutting-Edge Contracts.”

In this presentation, I’ll cover why we embarked on this project, how we executed it, and the outcomes we achieved. Let’s dive in!

## **Slide 2**

### **Project Context and Importance**

First, a bit of context.

We entered the Open Source Dev Ecosystem Challenge because the Cardano ecosystem faces scalability challenges due to constraints like the single UTXO model and the 16kb transaction size limit.

Additionally, there is a noticeable absence of shared design patterns as well as limited practical examples and educational resources for scaling solutions in the Cardano community.

To address these challenges, we focused on implementing robust on-chain data structures —Merkle Trees, Tries, and Linked Lists— in two of Cardano’s scripting languages, Aiken and Plutarch.

These structures are essential for enhancing the scalability and efficiency of DApps on Cardano and provide efficient solutions within the blockchain’s constraints.

## **Slide 3**

### **Project Objectives**

Our primary objectives were:

- To design and develop efficient data structures—Merkle Trees, Tries, and Linked Lists— for the Cardano blockchain in Aiken and Plutarch.

## Slide 4

- To ensure these data structures are thoroughly tested and production-ready.
- To provide detailed, easy-to-follow documentation to help developers implement these structures.

## Slide 5

### Execution and Milestones

We approached the challenge by doing research on data structures and developing those that could have the biggest impact. We then divided our project into three phases: design and development, testing, and documentation.

## Slide 6

### Phase 1: Design & Development

We successfully designed and developed the following data structures in both Plutarch and Aiken, each with its unique purpose and benefits:

## Slide 7

We chose to go with Aiken and Plutarch as opposed to the other Cardano development languages, such as PlutusTx because;

- Aiken, is particularly known for its expressive syntax, strong typing system and is rapidly gaining community adoption whereas,
- Plutarch is known for its ability to manipulate outcomes at a lower level (close to UPLC), which enables more control over the execution of smart contracts, leading to more efficient and compact scripts.

## Slide 8

- **Merkle Tree:** The Merkle Tree is valuable in proving the presence of arbitrary data within the tree structure. By carrying only the root hash in the script, an efficient and space-saving proof can be generated, ensuring data integrity and validity.

## Slide 9

- **Trie (Stick Breaking Set):** Developed with comprehensive functionality for Aiken and Plutarch, the Trie is particularly useful in facilitating mutable data storage in scripts by leveraging the sharing of common prefixes. This approach optimizes storage efficiency and enables more extensive data manipulation within the constrained on-chain environment.

## Slide 10

- **Linked List:** Created to enhance scalability and throughput in smart contracts, the Linked List leverages the EUTXO model to significantly enhance scalability and throughput. By linking multiple UTXOs together through a series of minting policies and validators, it improves the user experience interacting with smart contracts concurrently.

With this outlined approach of the execution of these technical solutions, we received a total funding of ₳238,374 from Catalyst Fund 10. To date, we have received ₳105,000, with ₳133,374 remaining to be distributed upon the completion of all milestones.

Each design pattern was tagged with a release version, enabling the community to track progress and updates.

## Slide 11

### Phase 2: Thorough Testing

We conducted rigorous testing for all three data structures to ensure their robustness and efficiency. Each data structure was subjected to code reviews and thorough unit tests, with results integrated into a CI/CD pipeline. We ensured that all design patterns passed the unit testing phase successfully.

## Slide 12

### Phase 3: Comprehensive Documentation

We focused on creating comprehensive, user-friendly, and visually engaging documentation for each data structure. Our aim was to ensure that developers of varying skill levels could understand, implement, and leverage these data structures effectively. Our documentation includes:

## Slide 13

Here's an example execution demo of a Merkle Tree

## Slide 14

- **Merkle Tree:** Detailed guides and practical examples which can be found at.
  - Aiken: <https://github.com/Anastasia-Labs/aiken-merkle-tree>
  - Plutarch: <https://github.com/Anastasia-Labs/plutarch-merkle-tree>

## Slide 15

Here's an example execution demo of a Trie

## Slide 16

- **Trie:** Step-by-step instructions and tutorials can be found here;
  - Aiken: <https://github.com/Anastasia-Labs/aiken-trie>
  - Plutarch: <https://github.com/Anastasia-Labs/plutarch-trie>

## Slide 17

Here's an example execution demo of a Linked Lists

## Slide 18

- **Linked List:** Clear diagrams and use cases can be found at;
  - Aiken: <https://github.com/Anastasia-Labs/aiken-linked-list>.
  - Plutarch: <https://github.com/Anastasia-Labs/plutarch-linked-list>

All documentation is well-organized, well-structured, and accessible to developers of varying skill levels and can be found in our Main Github Repo: <https://github.com/Anastasia-Labs/data-structures>

## Slide 19

### Achievements and Outcomes

We are thrilled to share the remarkable outcomes of our project, which we are hopeful will significantly contribute to the Cardano ecosystem:

- **Implementation of Advanced Data Structures:** Our team has pioneered the development and implementation of advanced data structures (Merkle Trees, Tries, and Linked Lists) in Aiken and Plutarch, enhancing the scalability of DApps on Cardano by allowing for more efficient use of the limited 16kb transaction size and the single UTXO model.
- **Extensive Documentation and Tutorials:** As you have seen, we are proud to have enriched the Cardano community with educational resources by providing comprehensive documentation and tutorials to help developers understand and utilize these data structures in their projects, covering the gap of limited scaling solution resources.

## Slide 20

- **Collaboration with Industry Leaders:** Some of our data structures have already garnered interest from industry leaders such as Sundae Labs. They successfully integrated our LinkedList implementation into their Single Asset Staking library and Taste Test mechanism, showcasing the real-world applicability of our work.
- **Quality Assurance:** Through rigorous code reviews and comprehensive unit tests, we have ensured that our implementations are not only reliable but also efficient, setting a benchmark for future developments in the community.

## Slide 21

### Key Learnings and Challenges

Throughout the project, we encountered several challenges and gained valuable insights:

- **Efficient Data Integrity with Merkle Trees:** We gained a deep understanding of constructing and verifying Merkle proofs, which enhanced our ability to implement secure and efficient data validation processes.
- **Flexibility and Efficiency with Linked Lists:** Leveraging Plutarch's functional programming paradigm and Aiken's robust type system to create Linked Lists that are both efficient and easy to maintain highlighted the importance of designing for efficient insertion and deletion operations.
- **Managing Mutable Data with Tries:** We learned to optimize storage by sharing common prefixes and implementing fast update and lookup operations.
- **Further Experience with Plutarch and Aiken:** We got to appreciate Plutarch's ability to provide more efficient and precise control over data structures, facilitating the construction of intricate data structures and optimizing the performance of smart contracts on the Cardano blockchain.

Aiken's expressive syntax and strong typing system also contributed to writing more clear and maintainable.

## Slide 22

### Future Prospects and Community Impact

These data structures hold immense promise for Cardano's ecosystem.

As Cardano continues to grow and evolve, developers will seek efficient solutions to overcome scalability challenges. Our data structures provide precisely that—scalability within the blockchain's constraints.

Imagine DApps that can handle more users, transactions, and complex smart contracts seamlessly. We anticipate early adopters will soon recognize their value, paving the way for broader adoption.

- **Sparkling Interest and Collaboration:** We've already generated interest and discussions within the community. By inviting developers to collaborate, participate in our community and twitter spaces, we foster a vibrant ecosystem and accelerate adoption.
- **Continuous Improvement:** Feedback from the community ensures ongoing enhancements, making our data structures even stronger and more reliable.
- **Comprehensive Documentation:** We are committed to improving our documentation with tutorials, real-world examples, and troubleshooting guides, empowering developers of all skill levels.
- **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 23

### Conclusion:

In conclusion, our project has successfully bridged a critical gap in the Cardano ecosystem by providing essential data structures and comprehensive documentation. We believe these tools will significantly enhance the scalability and functionality of Cardano-based applications.

Thank you for your support and interest in our work.

## Slide 24

For more information, visit our GitHub repository at:

<https://github.com/Anastasia-Labs/data-structures>.

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

- Our website at <https://anastasiabooks.com/>
- Follow us on twitter at <https://x.com/AnastasiaLabs>
- Join our discord community: <https://discord.com/invite/8TYSgwthVy>

See you next time!

Goodbye!