# Treasury Withdrawal - Intersect metadata - Informatik Mündler - OpShin - Python Smart Contracts for Cardano

## **Title**

Withdraw ₳199,911for OpShin - Python Smart Contracts for Cardano

## **Abstract**

This treasury withdrawal funds **OpShin - Python Smart Contracts for Cardano** which will provide the following services:

This is OpShin: a toolchain that lets developers construct Cardano smart contracts in Python. The toolchain comprises several projects that aim to facilitate the development of Smart Contracts and dApps on Cardano. They are largely based on Python, or integrate well with it for maximal accessibility. On top of accessibility, OpShin ensures low transaction cost compared to PlutusTx.

At the time of writing, Python is the second-most used language on GitHub, with 14.75% of the active userbase working with Python, and enjoys a 22.5% year-over-year increase in users; this is driven in part by its utility in data science and machine learning.

Everyone who knows Python can leverage the toolstack available for Python development to build on Cardano; and that is a lot of people, with a lot of tools.

OpShin enforces strict typing on the high-level programming: it implements a type system on top of Python type hints. However, all traces of typing are erased during compilation, for a maximum efficiency program. OpShin beats many programming languages when taken to the test for real use-cases.

This Treasury Withdrawal is submitted by Intersect on behalf of the vendor. The following sections; Abstract, Motivation, Rationale and Vendor Profile have been sourced from the approved proposal submitted by the Vendor as part of the Intersect budget process.

This treasury withdrawal funds one of 39 proposals to give effect to the approved budget info action for ₳275,269,340, administered by Intersect via gov\_action1u9x73kwufaxa70lfy59g4ynwyrcsaxdcd0gxzzmh67s9fxq4j8hqqk2phgh. The information provided herein is intended to fulfill the spirit of the constitutional requirement for a treasury withdrawal info action by also providing the details of the proposed solution, alignment to the budget, and amount to be withdrawn from the Cardano Treasury.

## **Motivation**

This proposal aims to solve the following problem:

Developing Smart Contracts on Cardano is difficult to get started with. Historically, the functional programming paradigm of Haskell and Plutus has scared away new developers. Aiken, Plu-ts and similar alternatives are also based on functional programming. Moreover all of these languages are rare, some specifically designed for Cardano Contracts. Thus, if developers want to start building for Cardano, they need to learn a completely new language.

## **Rationale**

### Project Solution

We propose OpShin, a Python-based smart contract language for Cardano. Python is one of the most popular programming languages known. On top of this we provide the security guarantees that are the reason for the Cardano community to embrace functional programming - thus offering an accessible and easy to start with alternative to current languages.

OpShin is aimed at developers that want to build on Cardano. It drastically simplifies onboarding when experience in Python is given, and even without is easy to read and access.

By making development of Smart Contracts more accessible, OpShin is a key catalyst for a growing ecosystem of DApps on Cardano. The benefit to the community is immense.

The project is under ongoing development. The language itself is already completed and production ready, but the Smart Contract Ecosystem is always evolving (i.e. new Plutus versions, standards, CIPs). This evolution requires continuous adaptations. The proposal is enough to fund further development for one year. There are no fixed deliverables as it is hard to predict the changes in the ecosystem.

### Vendor Profile

The main proposer, Niels, has already bootstrapped the main part of the project. In its current form the tool is able to be used and already being used. The proposer is completely doxed and well-known in the Community.

He also has a strong network of competent and experienced developers interested and capable in supporting the project.

He is supported on a per-demand basis by contributors to OpShin such as Julius Frost and S.C. Musson, who have redeemed themselves by contributing valuable improvements.

The main goal of this proposal is to improve usability and developer experience with respect to Smart Contracts on Cardano. With the given funds, the continued development of the toolchain is secured as developers can be hired to tackle basic issues in the toolchain whereas the main proposer is able to allocate more time of the day towards developing OpShin himself.

### Contract Management

A written off-chain Legal Contract will be created between the Vendor and the Cardano Development Holdings (CDH), as mandated by the constitution, and will be administered by Intersect. This will include details of the project delivery schedule and dispute resolution.

### Project Delivery

All milestones, acceptance criteria, payment amounts and expected delivery dates will be agreed between the Vendor and Intersect, acting on behalf of the CDH. The vendor will deliver according to the agreed-upon project schedule within the Legal Contract, of which the necessary information will be made public via the budget management platform via transaction metadata.

Defined by the milestones within a Legal Contract, the vendor will submit and attest milestone acceptance to the community, Intersect or 3rd Party Assurer.

Project progress will be monitored via Intersect’s delivery assurance function which will be communicated to the community.

Acceptance of the above work is expected to be supported by a 3rd Party Assurer, who will be responsible for reviewing and signing off the work completed at each project milestone against the corresponding milestone deliverables detailed within the Legal Contract. This work is funded from a portion of this treasury withdrawal.

### Budget Management Tooling

To administrate treasury funds on-chain, Intersect will utilize the treasury management smart contract framework developed by Sundae Labs. The smart contracts have been [extensively tested](https://github.com/SundaeSwap-finance/treasury-contracts/tree/main/offchain/tests) including audits from TxPipe and MLabs. Examples of the usage of these contracts can be seen across mainnet described across Intersect authored [Blog 1](https://www.intersectmbo.org/news/smart-contract-mainnet-demo-a-step-toward-on-chain-treasury-withdrawals), [Blog 2](https://www.intersectmbo.org/news/smart-contract-mainnet-demo-day-two-update) and [Blog 3](https://www.intersectmbo.org/news/smart-contract-mainnet-demo-day-three-update).

Final mainnet validation test can be seen via the Disburse action within transaction: 0f591dc544ae14102dbb4a74d5311a6acffc1772b163d8b7a9656b9525950b17

With the confirmed treasury reserve contract address being: stake17xzc8pt7fgf0lc0x7eq6z7z6puhsxmzktna7dluahrj6g6ghh5qjr

#### Specifics

Intersect will utilize a single Treasury Reserve Smart Contract (TRSC), with many Project-Specific Smart Contracts (PSSC), managed by Intersect. Intersect’s management consists of three ‘admin’ and two Intersect ‘leadership’ roles. An Oversight Committee consisting of five external, independent third-party entities will provide checks and balances on Intersect, and safeguard against errors and unilateral control. The administration of both TRSC and PSSCs will be managed by Intersect, with external oversight on certain actions from the Oversight Committee.

The Oversight Committee consists of Sundae Labs, Cardano Foundation, Dquadrant, Xerberus and NMKR. Their role is to independently verify key administrative actions using on-chain logic, ensuring accuracy and consistency without exercising discretion over governance decisions.

For all details on Intersect’s configuration please see the [**Smart Contract Guide**](https://docs.intersectmbo.org/cardano-facilitation-services/cardano-budget/intersect-administration-services/smart-contracts-as-part-of-our-administration) on the knowledgebase.

The high level permissions are as follows:

* TRSC Fund and PSSC Modify
  + Two of the three Intersect admins, two of the five trusted entities and one of the two Intersect leadership sign-off must authorize
* TRSC Disperse
  + Two of three Intersect admins, three of five trusted entities and two of two Intersect leadership sign-off must authorize
* TRSC Pause and Resume
  + Two of three Intersect admins, and one of two Intersect leadership sign-off must authorize
* TRSC Sweep
  + One of three Intersect admins, and one of two Intersect leadership sign-off must authorize
* TRSC Reorganize
  + Two of three Intersect admins and three of five trusted entities must authorize

#### Processes

Upon enactment of this governance action, funding for this project will be directed into the TRSC’s stake account. All instances of TRSC and PSSC can not be staked with a SPO and will be delegated to the auto-abstain predefined DRep. From here funds will be withdrawn into a UTxO remaining at the TRSC.

When the Legal contract is prepared and the vendor is ready, funding for this project will be transferred using the Fund action to a PSSC. All milestones will be outlined within the metadata.

A dashboard will be available for the community to audit the TRSC or PSSC and track metrics related to this withdrawn ada as well as being immutably verifiable on chain.

The subsections; Contract Management, Project Delivery, and Budget Management Tooling described above cover the constitutional requirements specified in Article IV section 4 and 5.

## **References**

OpShin Homepage

* <https://opshin.dev/>

OpShin GitHub Page

* <https://github.com/OpShin/opshin>

OpShin Pioneer program introduction

* <https://www.youtube.com/watch?v=vMTi1TS8jec>

OpShin Book - Documentation for the Smart Contract Language

* <https://book.opshin.dev/>

Project Proposal In Ekklesia

* <https://2025budget.intersectmbo.org/ballots/680d1b63565577986442d123/proposals/680d1b64565577986442d26c>

Approved Budget Info Action submitted by Intersect via GovTool

* https://gov.tools/outcomes/governance\_actions/e14de8d9dc4f4ddf3fe9250a8a926e20f10e99b86bd0610b77d7a054981591ee#0

Details of all successful proposals (CSV)

* ipfs://bafybeicwrop4q7xvnyjdd5drumbe56sqtm5lbe2ul3c262zt4hgguzdycm

Automating Accountability: Cardano’s Smart Contract Framework Blog

* ipfs://bafybeihqx4ae72z7suqfnxrpqpqithp43cai7o2uuewnqtezgaoyc3ptyq

Sundae Labs Budget Management Smart Contracts Github Repository

* https://github.com/SundaeSwap-finance/treasury-contracts

Budget Management Smart Contracts TxPipe Audit Report

* ipfs://

Budget Management Smart Contracts MLabs Audit Report

* ipfs://

## **Authors**

* Intersect