

# AlphaGrowth Staking Service Grant Application

## Introduction

*Tell us about your company/project, team experience, track record and vision.*

AlphaGrowth is a Web3 driven business operating from Barcelona, Spain. The company is a spin-off of FabArmadillo, a private fund created in 2017 to perform PoW mining operations as well as crypto-trading analysis.

As a company, we have experience in Web3 project development and consulting, such as ouche.art – a market place for native digital artist- and Dogstarcoin.com – a DeFi project build on Stellar, granted with \$78k by SDF-

In May 2022 we launched our firsts PoS nodes. At this time, we validate eight chains: Oasis Network, Fantom, Evmos, Near Protocol, Osmosis, Kava, RebusChain & Teritori. We leverage institutional customers interest within this networks to launch nodes and offer a non-custodial staking service to retail customers with minimum commission. The brand operating this business is alphanodes.io. We're a Staking Rewards VSP operator since January 2024 with an AA rate. TVL on non-custodial is ~\$2M

Also in May 2022, some of our institutional customers, required us to offer them a solution to become launch validators on Ethereum. Since then, we have been growing, up to the +500 validators that we currently manage. This solution is under Ethernodes.io's brand, and it's a custodial Eth staking solution for institutional investors and intermediaries. The platform was built under some customers' requirements: they had some extra eth liquidity that wasn't enough to set up a new validator and required us to pool surplus assets. Total TVL in Ethernodes.io is ~\$50M.

Alphagrowth and its brands are registered as a custodial service provider under Spanish administration – Banco de España –.

## The Product

*What are you developing? What is your USP?*

Ethernodes.io is a custodial staking solution Ethereum, **already in production since September '23, with +1.700 Eth deposited**. The platform aims to leverage DVT through SSV tech by automating its processes. The implementation of SSV as a DVT infrastructure and its automation is the specific request for the SSV grant.

Ethernodes receives deposits from customers and manage all the process for them: infra set up; validator launch & management; surplus asset pooling; rewards distribution; custody of funds. The USP is actually quite simple: earn real & transparent ETH yield by just depositing in our platform. Customers don't need to either have technical knowledge or time dedication to generate their yield. **The Beta version of Ethernodes.io is already up & running, with 1.700 Eth already deposited**. We've **deployed 53 validators leveraging SSV tech** so far, but a lot more has to come.

A key component of Ethernodes.io is transparency. The platform shows exactly how funds are managed. That means that anyone, at any time, can audit balances, deposits, rewards allocations, nodes performance and yield offered. We're now under **development of Ethernodes V2**, a version in which the customers will not only receive their staking yield but also **a portion of the revenues Ethernodes** is earning as an Ethereum Node Operator, validating for Liquid Staking protocols. So far, we've integrated Stader validators into Ethernodes V2, that we will be running on SSV.

Customers can add as many deposits as they wish to their user. Each deposit can have a different wallet address linked to it. That means that a customer that is an intermediary using our structure to offer Eth staking yields to its final users, can manage infinite deposits under one user, each one tagged to one different customer / wallet. As we're running Beta, both the process to deploy/undeploy validators and reward distribution are managed partially manually. In order to scale up, we need to properly automate processes to become more efficient. We've proved market fit since the platform attracted ~20% new deposits since inception (August 23) and the MVP has been validated. Now is time to go one step forward by automating infrastructure, yield sharing & applying SSV DVT tech to our backend.

Ethernodes.io V2 will work on two different layers:

- Automated deposits, Withdrawals & Reward distribution  
Managed by SmartContracts
- Deploy/undeploy of validators using SSV DVT tech  
Starting by leveraging on our own SSV operator cluster (1Q24). Using DVT tech is a must for Ethernodes, as this is a custodial solution that needs to rely on a solid structure that minimizes risks of downtime and slashing. Ethernodes will carefully choose each cluster of operators and periodically reevaluate their performance.

We want Ethernodes.io to become an agnostic player on Ethereum staking solutions. That means that we'd leverage DVT tech to not only offer the staking reward to our customer because of its deposits, but also perform a revenue sharing of commissions received as an operator. This way, we'll outperform traditional staking rewards for Ethereum and attract more deposits.

**Ethernodes is not a liquid staking solution but a revenue sharing solution.** We leverage our customer's deposits to deploy validators on different protocols, using DVT infra for resilience. Our customers get the APR from staking and an additional APR based on the amount of revenue that Ethernodes share with them. The first three protocols with validators to integrate within this model are Stride, Lido Simple Module & RocketPool.

## GTM

*What is your GTM strategy? How do you stack up against competitors?*

At Ethernodes.io we differentiate two types of customers:

- Institutional investor  
This customer understands blockchain & Ethereum's potential. He wants to be a player in the field but hasn't the technical knowledge or time to integrate teams or managers to know how to perform. They rather outsource the process to a company they can trust and talk with, that actually becomes a partner, to take the most out of their investment.  
We identified and understood this need, and decided to do something about it. Institutional investors can now yield their Ethereum by just depositing in Ethernodes.io. We take care of the rest.
- DeFi solutions intermediary  
Some players have emerged in the last 12 months on ecosystems. Those who understand how massive DeFi has become and how tricky might be for newbies. These players try to abstract DeFi complexity by creating platforms that go beyond a simple CEX. They offer access to curated DeFi options, with attractive & real yields. They usually work on a "custodial based service", which means that the platform itself is responsible for choosing its partners & protocols. Then, sometimes the customer can select the product; sometimes the product is already designed by the platform.

We offer Ethernodes.io as a product itself that they can offer to their final customers, or use it to create their products. As we can easily differentiate each deposit for one single user, reconciliation and distribution become super convenient.

Thus, our GTM strategy can be split up into three:

- Attract volume of LS protocols in which we act as an operator (increase yield by revenue sharing policy to our customers): Stader, RocketPool, Lido, etc
- Close partnership deals with institutional investors: VC's, FO, Asset managers.
- Place our product on third-party DeFi platforms: Criptan, Commonsense.finance, AllDefi.io, LunarXY, PDAX, MétodoBitcoin, etc

## Proposal Details

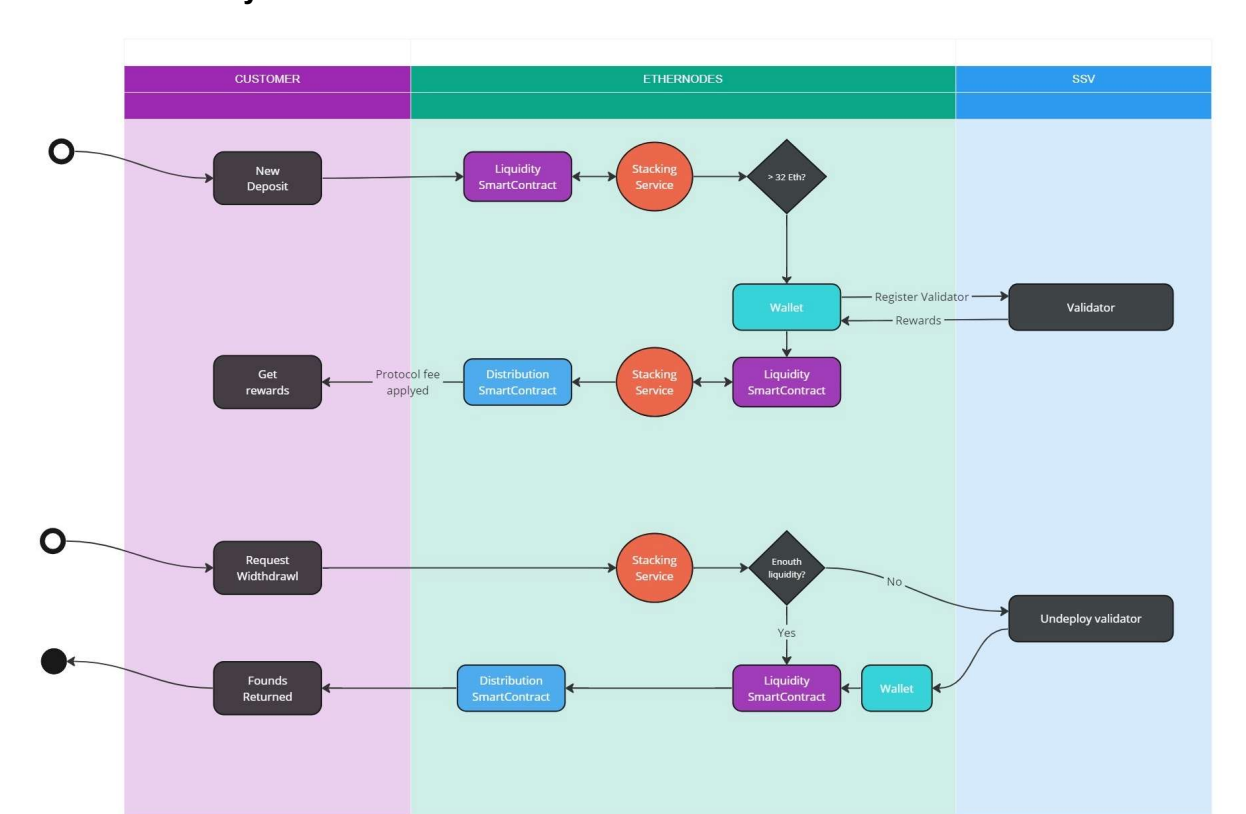
### Technical Overview

Present a high-level overview of how your product or service would integrate with the ssv.network through **user flows**, and provide an **system overview** of your main components (e.g. smart contracts, dapp, off-chain entities, etc..).

#### User Flow

The following diagrams cover Ethernodes' main functionalities. As we stated before, Ethernodes' goal is to abstract difficulty to the user when carrying out the staking Eth process.

#### High Level Functionality



Users' main actions are *deposit*, *get rewards* and *withdrawal*. The platform cares about the actions required to generate rewards: pooling a validator each 32 ETH, managing operators, analyzing risks, distributing rewards, etc.

- *New Deposits*

As a custodial regulated platform, only Customers previously KYCed will be able to deposit in Ethernodes. If a deposit is received from a non-whitelisted address, it will be returned to its original sender. Deposits can be performed by using a Web3 connection in the platform itself or directly sending ETH to the Liquidity SmartContract (LSC). If funds are sent directly to the LSC, a manual re-conciliation must be done. LSC acts as a liquidity distributor, being at the core of Ethernodes.

- *Rewards distribution*

Customers will accumulate rewards each day, based on the performance of the validators set up by Ethernodes & the performance of the revenues earned by Ethernodes on their duties as a NodeOperator. Rewards will be distributed proportionally to each customer's deposit and can be autocompounded daily. For those customers that don't want to autocompound rewards, a claim process will be developed. It will be a *pull over push* process for whitelisted addresses.

All rewards will be distributed using an isolated SC, the Distribution Smart Contract (DSC). DSC will be funded by LSC, based on the pending rewards to be claimed by users, which will be an off-chain information. Deposits with auto-compound functionality activated won't be taken into account for DSC balance requirements.

Once a customer claims a reward, DSC will check if the address is both whitelisted on the LSC and on the off-chain DB. It'll also get info regarding the amount to be transferred from the Off-Chain DB for security purposes.

- *Request Withdrawal*

When a user wants to withdraw a deposit, the request will be sent to the LSC. After checking if the address is whitelisted on both LSC & the off-chain database, the LSC will perform the withdrawal if there are enough funds. If that's not the case, a validator will be un-deployed to bring liquidity to the LSC.

Validators deployment or un-deployment will be done by a backend wallet, once 32 eth are reached at the LSC and funds are sent to the deployment wallet. LSC would only withdraw funds to whitelisted addresses, which are customer deposit addresses, the wallet used to deploy nodes and DSC addresses.

## On-Chain contracts

- *Liquidity Smart Contract (LSC)*

LSC is the central system of Ethernodes. **LSC is just a SC that collect deposits and execute withdrawals and rewards distribution events.** This allows us to manage funds efficiently, as less on-chain transactions are needed to perform all requests: deposits, withdrawal and rewards balances are well managed on an off-chain DB, while LSC holds liquidity for the protocol no matter the origin of the action. Ethernodes is over collateralized by 30 to 63 eth, which means that users deposits can be reconciled whether funds are in an active validator, the LSC, the DSC or the Wallet which deploys the validators.

LSC will have a whitelist of addresses. These addresses will be consumed by the DSC in the claim process and also to send funds to the multisig wallet, responsible for deploying new validators.

- *Distribution Smart Contract (DSC)*

DSC is an isolated SC whose only function is to distribute rewards when a user claims them. DSC will request LSC the fund needs to cover for pending claims, which will be an off-chain data. When a user

claims, DSC asks both LSC and off-chain DB if the address is legit (whitelisted). Both LSC and off-chain DB addresses have to agree about the address being whitelisted. Off-chain DB will be responsible for giving the input needed in terms of the amount that must be sent.

- Operator Wallet

A specific operator wallet will be used to set up new validators. Once the LSC reaches a threshold of 32 eth, it will send the eth to the operator wallet. Then, Ethernodes' backend will deploy a new validator leveraging SSVs cluster of operators.

### **Off-Chain DataBase & relation with LSC and DSC**

Ethernodes works as a custodial staking ethereum service. All customers have to go through a KYC process to use our services. Each user is linked to an e-mail address and can perform deposits using different wallets. Each deposit from each different wallet will be OFAC list checked.

Each deposit works as an isolated deposit in terms of database, and it's related to one withdrawal address. That means that one single user, identified by email address, can have different deposits linked to a different wallet address. This functionality is already up & running in the platform and give us the flexibility needed to onboard centralized platforms that want to consume the product.

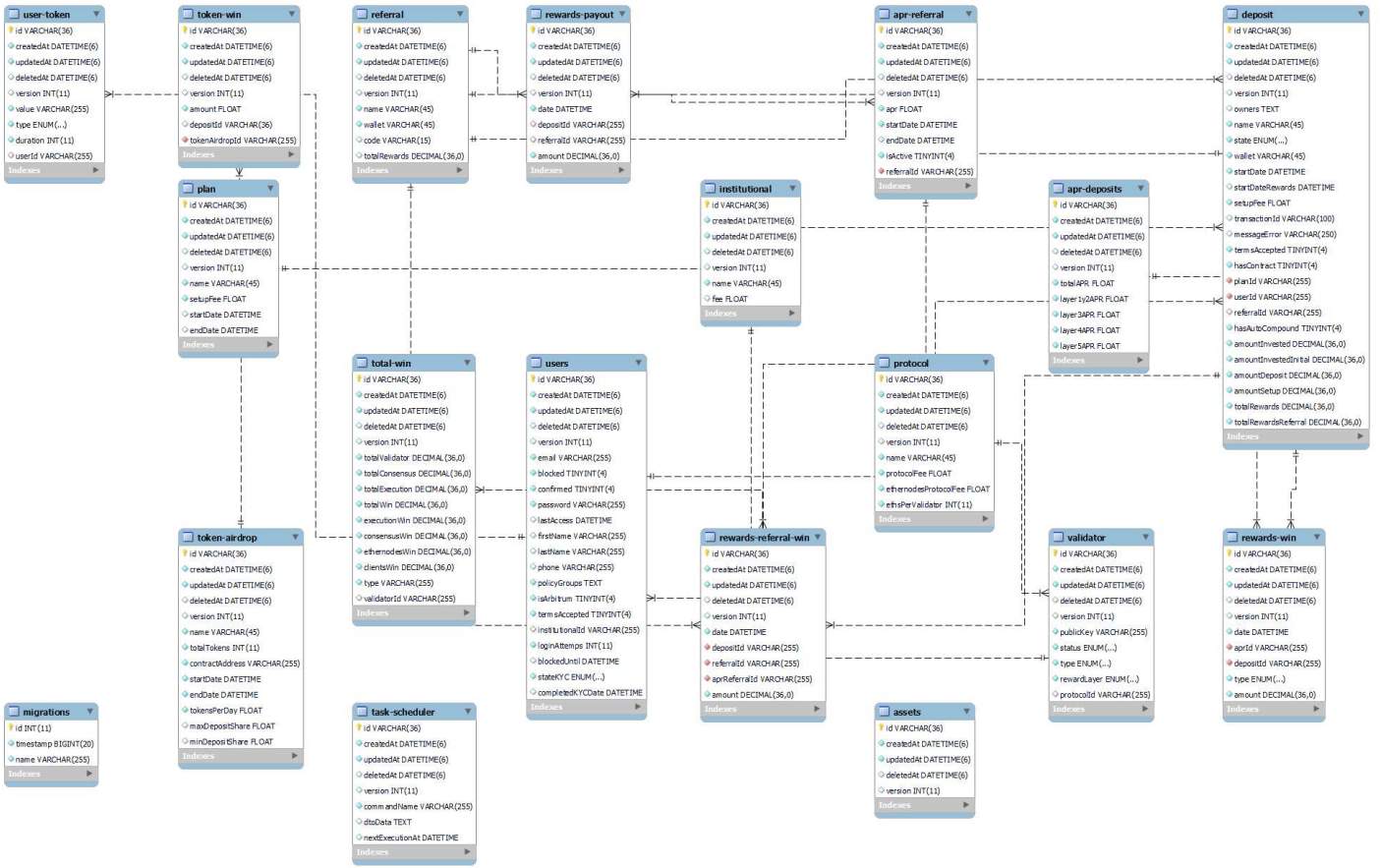
Off-chain DB lays at the center of Ethernodes information system as it held all relations between a user, its deposits and authorized wallets. **Rewards are calculated and distributed in a daily basis**, by obtaining data from validators active in the beaconchain linked to Ethernodes. Reward assignation is registered daily on a deposit level, assigning the amount rewarded based on the weight a deposit has over the total amount deposited. For example, if total rewards of the day are 1eth and there are two deposits in Ethernodes of 6 & 4 eth each one, this 1 eth of rewards is assigned as 0.6 to the 6-eth deposit and 0.4 eth to the 4-eth deposit.

When a customer requires a withdrawal or rewards have to be distributed to Ethernodes depositors, the systems would call the DSC to update the information regarding transfers to be made. As the DSC has no balance in it, the LSC is required to send enough funds to the DSC to perform the transactions needed. If no balance to be transferred is available at the LSC, a warning will be sent to Ethernodes admins to decide whether disable a validator or not.

Once a transfer is executed by the DSC, it will be marked as executed both at the DSC and off-chain Data Base. DSC can only execute transactions of non-marked transfers, and the LSC would be the only authorized SC to register new pending transactions in the DSC. The process will be called to execute one a day.

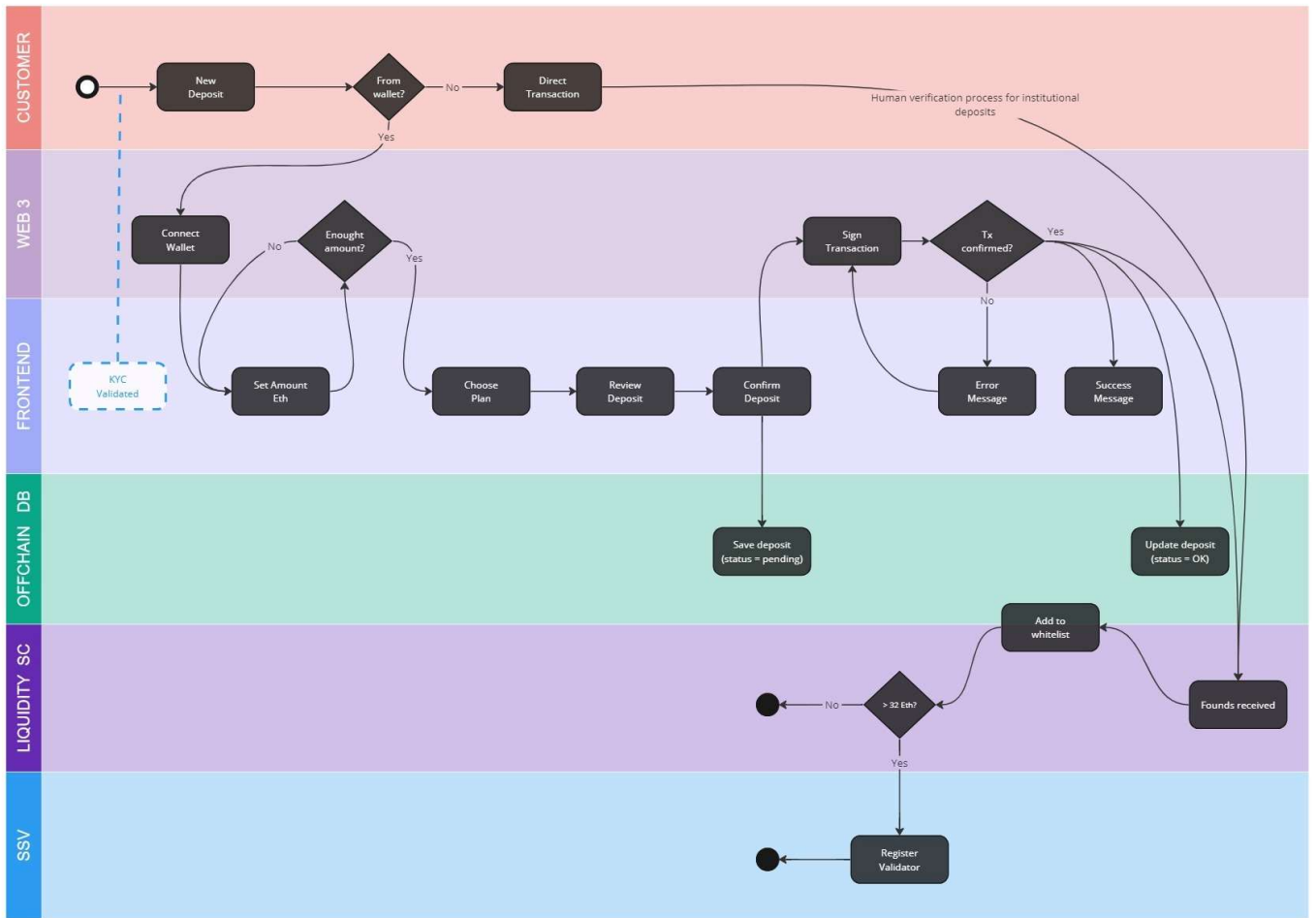
For security proposes, addresses linked to a deposit are whitelisted on-chain at the Distribution Smart Contract. This way, an additional check by calling the whitelisted addresses form the DSC and the off-chain data base can be made before executing any transaction.

Find the off-chain data base entity relation diagram of Ethernodes:

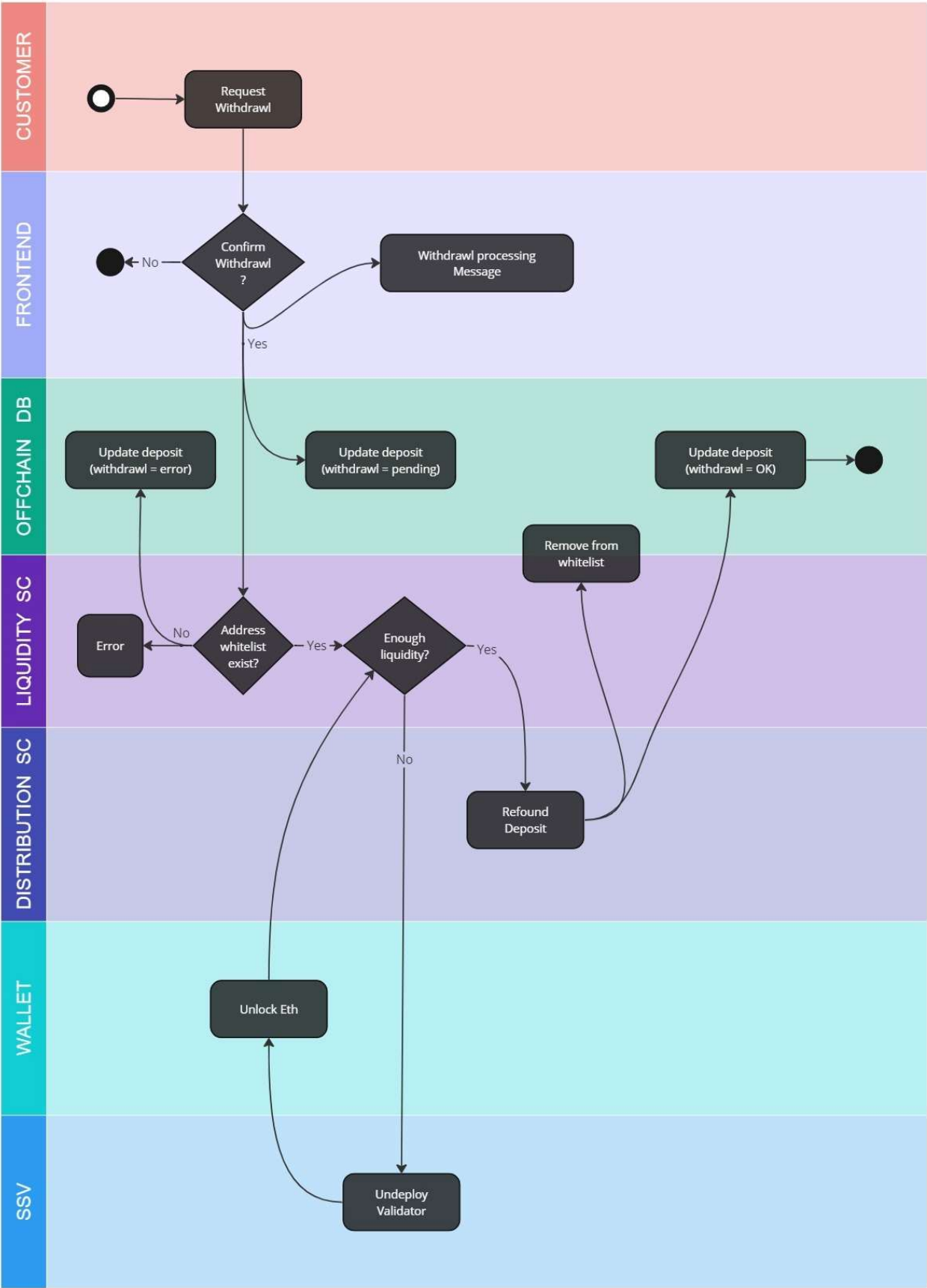


Find here detailed diagrams on how each functionality works:

## Deposit process

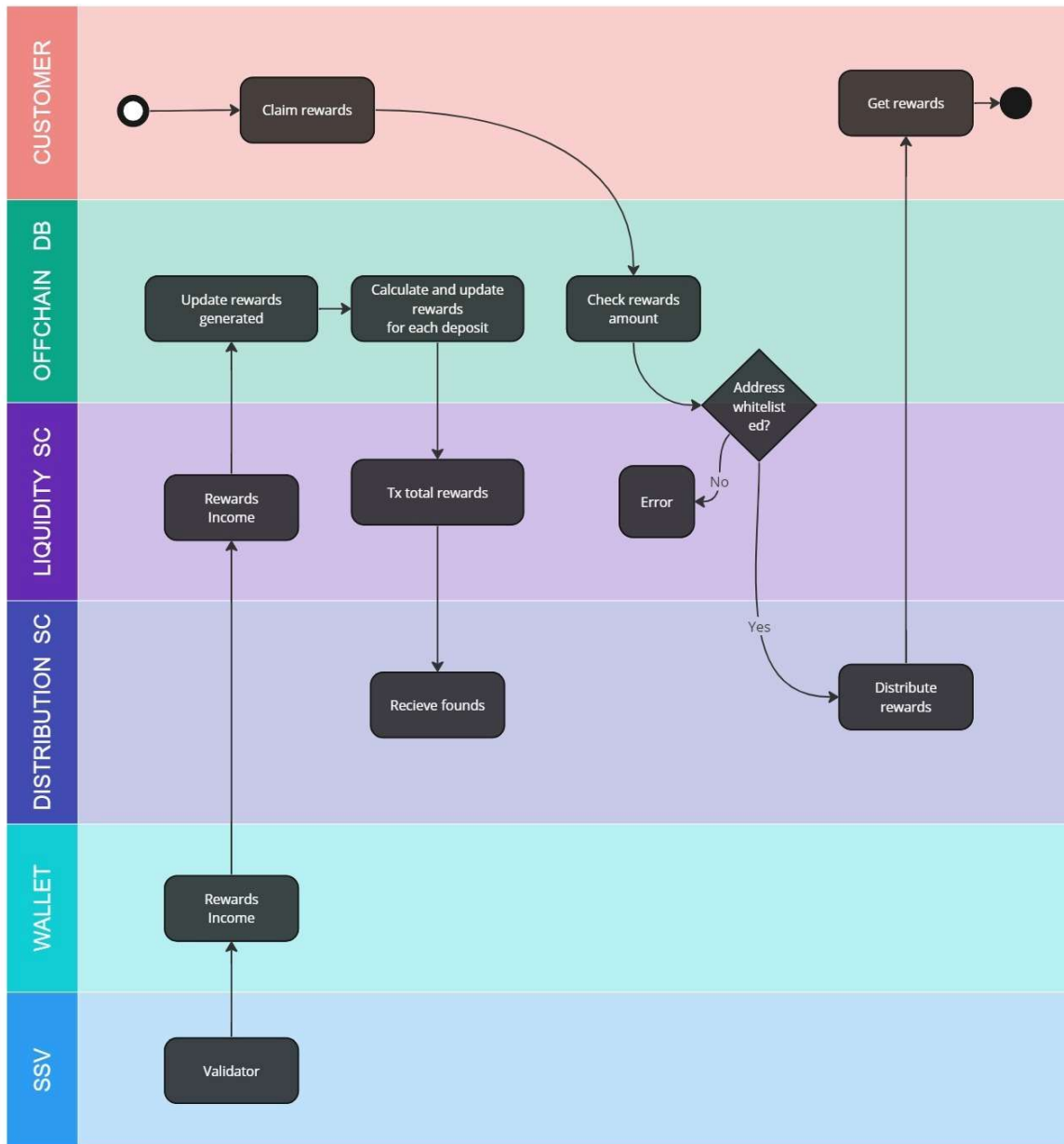


# Withdrawl process

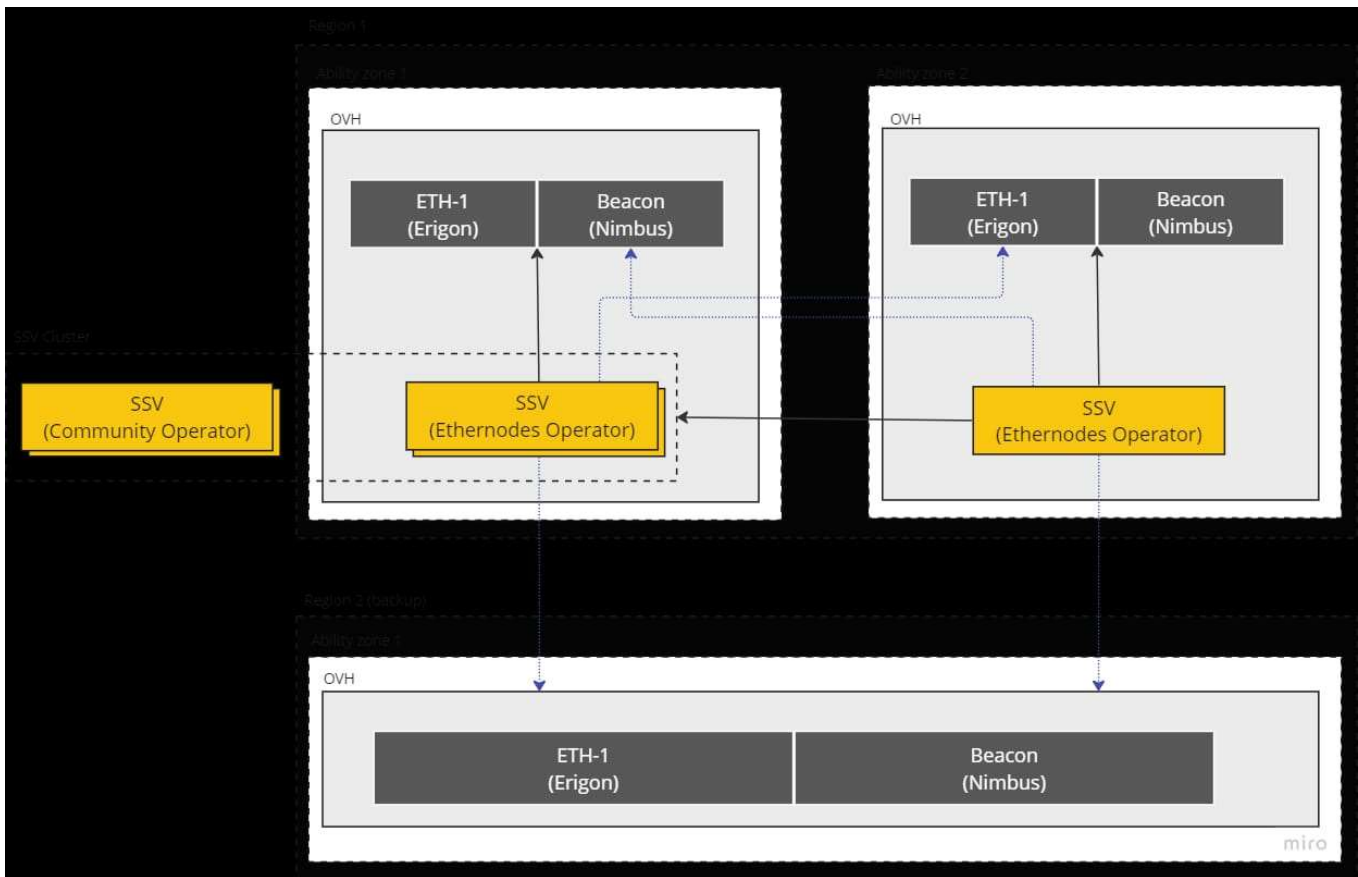




# Rewards process



The infrastructure integration with SSV will work as showed in the following diagram:



## Key Management

As a centralized platform, key management is a sensitive process. All validator keys, operator private keys and keys related to any wallet used by Ethernodes are stored off-chain in a cold storage with redundancy, encryption and specific protocol to access them.

The private key that manages operators is also needed at the backend to perform some critical operations. Ethernodes' operator private key will be stored on a machine with total access blocked, except for a whitelist of allowed IPs. Thus, the connection between Ethernodes' backend and key storage will be managed through a single IP path highly monitored. Being Ethernodes' machine the only one allowed to access the machine in which the operator keys are stored, in any case of unauthorized access to Ethernodes backend server will automatically shut down.

Keys stored on the isolated machine will be encrypted.

## Operator Selection

*Describe how you plan to select the operators to manage the service's validators.*

Ethernodes will work with a cluster of 4 operators with the following specifications:

- 1 own operator, bare metal, based in Spain
- 2 own operators, cloud, based in Europe
- 1 third party operator to leverage DVT and have strongest redundancy.

Third party operator will be selected based on the following criteria:

- Good performance & uptime
- Reasonable fees
- Owned by trusted and recognizable operators of the ecosystem

We might divide volume towards different third-party operators, based on volume managed and to avoid concentration risks.

As Ethernodes.io is a regulated custodial solution, we must ensure that operations are performed by a majority of operators managed by us.

## Withdrawals

*Describe how you plan to handle user withdrawals (pre and post withdrawals are enabled).*

When a user wants to withdraw a deposit, the request will be sent to the DSC. After checking if the address is whitelisted on both DSC & off-chain database, the DSC will perform the withdrawal if there are enough funds at the LSC to be provided to the DSC. If that's not the case, a warning will be sent to the admin for a validator to be un-deployed in order to bring liquidity to the LSC. The DSC would check periodically if it holds enough funds to process the withdrawal request.

## SSV Payments

*If relevant, outline how you plan to manage the facilitation of fees charged by the network and its operators.*

Fees will be managed by Ethernodes backend, ensuring each validator cluster has enough funds to perform operations for at least 6 months. A functionality called "SSV balance process" will be responsible for checking and topping up each SSV cluster balance periodically. The main functions of this process are:

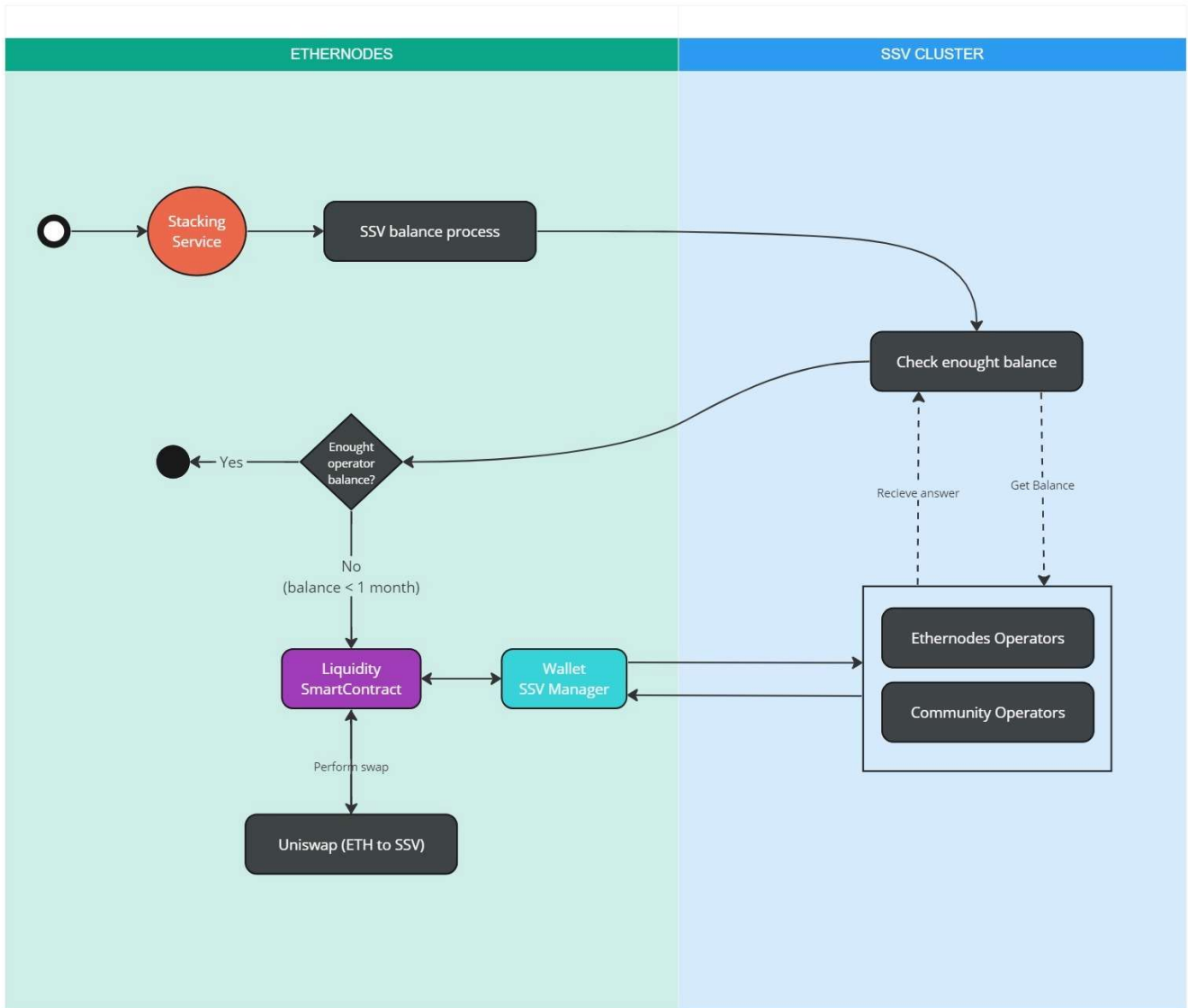
- **Check Balance:** once a week, a listener (SSV Scanner) will ask the cluster for both its current balance and the total SSV needed for six months of operations.  
If the answer is that the current balance is lower than the requested for one month of activity, a process in the LSC to acquire SSV and top up the cluster will be triggered.
- **Topping balance:** the LSC will receive the call to action from the "SSV balance process". The LSC will be rooted to Uniswap V3 to swap Eth for the amount of SSV equivalent to 6 months of operations –for each active cluster–. For security reasons, the swap process will require at list two signatures at the backend to be triggered.
- The LSC will send the SSV to each cluster to top up the balance requirements.

A monitoring tool will give an online status regarding cluster operations and a weekly status on SSV balances for each cluster.

Liquidity SmartContract events emitter for balance monitor:

- onWithdrawlPay
- onRewardsPay
- onNewDeposit
- onRegisterValidator

Those events will ensure that everything is smooth and there is always enough collateral for all validators.



## Project Plan

Outline your project milestones with a breakdown of objectives and deliverables - please include references to your integration phases of SSV **testnet** and **mainnet**.

*Please note that as milestones are required to be presented for evaluation by the Grants Committee, it is advised to structure their scope in a way to enable this.*

#	Milestone	Deliverables	Est. Effort
1	Title	<ul style="list-style-type: none"> <li>Description</li> <li></li> </ul>	2 weeks

<i>Predefined milestones by Grant committee (Please follow)</i>			
1	MVP Design + UX	<ul style="list-style-type: none"> <li>• Low-Level system specifications and diagram ✓</li> <li>• Contracts Liquidity and Distribution technical design</li> <li>• Frontend UI mockups and design ✓</li> </ul>	3 weeks
2	Testnet Integration	<ul style="list-style-type: none"> <li>• Frontend, Dapp and backend implementation ✓</li> <li>• Off chain development ✓</li> <li>• Contracts Liquidity and Distribution implementation and deployment on testnet</li> <li>• SSV testnet integration</li> <li>• Infrastructure architecture implementation on test environment ✓</li> <li>• Implement deposit, withdrawal and rewards processes</li> </ul>	6 weeks
3	Mainnet Integration	<ul style="list-style-type: none"> <li>• Smart contracts audit</li> <li>• Corporate web ✓</li> <li>• Deployments and testing on mainnet ✓</li> <li>• SSV Balance Process</li> <li>• Backend reconciliation (rewards calculation, distribution, etc) ✓</li> </ul>	6 weeks
4	TVL target reached 25 validators	<ul style="list-style-type: none"> <li>• Documentation ✓</li> <li>• SSV mainnet integration ✓</li> <li>• Beta v2 launch ✓</li> </ul>	5 weeks

## Payments

*State the total sum of funding you are requesting to develop this grant.*

Total estimated cost is \$45.000 (detail available if required)

## Terms

*Grants are paid 100% in SSV on a 7-day moving average.*

## Milestone Allocation

Outline payment allocation per each milestone.

*Please note that the mainnet milestone must represent at least 30% of the funding allocation.*

<b>Milestone</b>	<b>Amount</b>	<b>Percentage</b>
<i>Testnet Integration</i>	<i>\$10.000</i>	<i>22,22%</i>
<i>Mainnet Integration</i>	<i>\$15.000</i>	<i>33,33%</i>
<i>25 Validators reached</i>	<i>\$20.000</i>	<i>44,44%</i>
<b>Total</b>	<b>\$45.000</b>	<b>100%</b>

# Open Source

Indicate which components of your product will be open source and under which license (e.g. MIT).

*Please note that tooling and resources that benefit the entire developer's ecosystem of the SSV network are more favorable.*

Ethernodes.io will release the SC on Github on the 2H24.

## Q&A

### Questions raised by the SSV grants committee

- *We understand that your service is already in production, and you're applying for a grant to include DVT. If so, please make this clearer in the proposal*

Yes, ethernodes.io is already in production since September 2023. Since the conception of Ethernodes, the main objective has been to manage the validators using DVT-SSV technology, which is the main reason for the present grant.

Find here the owner of the first 53 validators owned by Ethernodes, deployed in SSV:  
<https://ssvscan.io/account/0x12c1ef3208de4805ba4ac3fea6a4f132f8d31243>

- *It is unclear if your service is a liquid staking solution given the term "Liquidity Smart Contract", which could be the contract for liquid staking or just to collect funds until 32 ETH is reached to launch a validator. Please provide clarity around this in the proposal.  
If it's a liquid staking solution, clearly state it as such, and provide more information for the token design (rebasing or reward-bearing).  
If it's just to collect funds, also clarify in the proposal.*

Ethernodes is not a Liquid Staking Protocol. The Liquidity Smart Contract is just a SC that aims to manage the liquidity requirements when receiving deposits, withdrawal requirements or distributing rewards

- *Diagrams in the proposal are barely readable and need higher resolution*

Updated for a better resolution

- *The off-chain information is one of the most important factors in a staking pool, and there is no mention of how this is fed to the contract. Please revise.*

The off-chain data base structure and high-level explanation has been included to the proposal. The off-chain data has to interact with the Distribution Smart Contract (DSC) for executing withdrawal requests and rewards distribution, if needed. At the same time, Liquidity Smart Contract (LSC) is responsible for feeding the DSC with enough balance to perform its duties daily. If there's not enough balance in the LSC, a warning will be sent to the admin to decide whereas to launch the process of disable on validator or not.

- *You state that Ethernodes is over-collateralised, but not by how much it is over-collateralised.*

Ethernodes is over collateralised by a max of 64 Eth (2 validators) and a minimum of 30 Eth. The amount is reviewable based on the total deposited, to stay always over a 2%. This 2% is enough to manage all fees needed in case of deprecating the platform.

- *You also state that a multisig wallet will be used to set up new validators. We're wondering what's the multisig for, if the backend manages it? Who holds the signatures? What's the threshold?*

A multisig was designed for better security over the operator wallet. Once the team deep dive into the requirements to have a multisig managing the operator address, this option has been dismissed. Thus, the operator wallet will be managed by the backend with different layers of accessing.

- *We're missing a description on key management: how do you handle validator keys in the backend? Do you custody keys yourself or do you use a custodian? If key management is currently centralised, do you consider to include DKG for decentralised key management?*

The private key that manages operators is also needed at the backend to perform some critical operations. Ethernodes' operator private key will be stored on a machine with total access blocked, except for a whitelist of allowed IPs. Thus, the connection between Ethernodes' backend and key storage will be managed through a single IP path highly monitored. Being Ethernodes' machine the only one allowed to access the machine in which the operator keys are stored, in any case of unauthorized access to Ethernodes backend server will automatically shut down. Keys stored on the isolated machine will be encrypted.