

# USER MANUAL



# GENERATOR



TEAM3D  
Hello@team3d.io

# TABLE OF *CONTENTS*

C:\team3d\generator.exe		
<b>OVERVIEW</b>	<b>3</b>	<b>POWER-UP VESTING</b>
The Prime Directive		Compounding a Vested Bonus
The TL;DR		Depositing Additional Funds to a Vesting Stake
<b>THE GENERATOR MODEL</b>	<b>4</b>	Player has left the game (Early Withdrawal Penalty)
		Stoker Rewards
<b>TERMINOLOGY</b>	<b>5</b>	<b>GENERATOR MAINTENANCE</b>
		Admin Authority
<b>COLLECTING FEES</b>	<b>6</b>	Beyond the Generator
		Monkey Wrench & User Agreement
		<b>CONTRACT FUNCTIONS</b>
Fee Allocation		Vault Contract
Staking Fees		Teller Contract
Rewards		
<b>STAKING IN THE GENERATOR</b>	<b>7</b>	<b>USER INTERFACE GUIDE</b>
		Selecting a Teller
Insert Coin LP Staking		Approving LP Tokens
Staking as a Service		Depositing LP Tokens
		Committing and Claiming
		Compound and Chill



# OVERVIEW

The goal of the Generator is to deliver value through decentralized gaming and finance. This value may be distributed to those who act per the community's goals and project at large by staking and holding their tokens. The longer the commitment and larger the stake, the greater the potential rewards. This mechanism increases stakers' and holders' potential profitability by reducing sell pressure on exchanges, incentivizing token lockups, and distributing surplus value in the ecosystem to those with positions in the Generator.

**The Generator represents one-third of the Vidya ecosystem: Earn.**



## The Prime Directive

- ✓ To reward stakers with VIDYA tokens collected on the VIDYA platform through the staking of LP pairs designed to help the VIDYA economy
- ✓ To provide projects with the ability to easily integrate LP token staking
- ✓ To incentivize activities that align with the goals of the community

## The TL;DR

- 1 Generator collects fees from user activity in the VIDYA market and ecosystem
- 2 Users stake LP tokens in the generator to support the VIDYA economy
- 3 Those staking in the generator receive fees in the form of VIDYA tokens

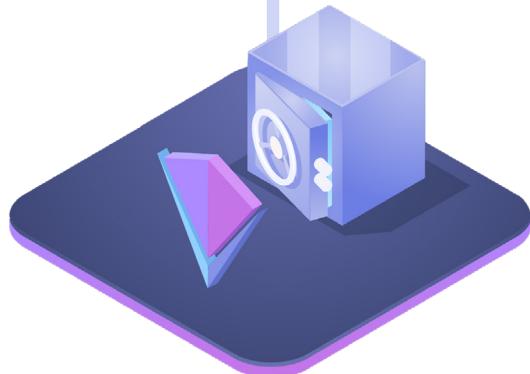
\*APY will not be advertised as fee distributions are based on player and market engagement with the VIDYA ecosystem. Reward distribution totals will be calculated as frequently as once per week. These numbers will inform stakers of potential rewards for staking in the Generator. Team3D and VIDYA make no guarantee that you will receive returns through staking on the VIDYA platform. Cryptocurrencies and decentralized applications are inherently risky. You can potentially lose your crypto through market forces. Please DYOR.

# THE GENERATOR MODEL

TRADING FEES ARE GENERATED FROM THE PLATFORM AND DECENTRALIZED EXCHANGES



THE GENERATOR VAULT ACCUMULATES GENERATED TRADING FEES IN VIDYA TOKENS



TELLERS DISTRIBUTE COLLECTED FEES TO ACTIVE STAKES



# TERMINOLOGY

## LP token

A liquidity pool token representing your stake in a pair of ERC-20 tokens on an exchange. These tokens support liquidity of the token pair and act as “market makers” on decentralized exchanges.

## VIDYA Rewards

The VIDYA distributed to stakers in the Generator.

## Vault

The contract responsible for collecting fees and holding fees in the Vidya ecosystem.

## Teller

The pooled contract responsible for maintaining staker LP deposits, adjusting LP weights within the pool, and receiving priority rewards from the Vault.

## Staking Fees

Any fee associated with staking tokens in a Teller.

## Vesting

A time-lock on stakes where a user is incentivized **not** to withdraw. The incentive to time-lock rewards the staker with a boost. A penalty is incurred if a time-locked stake is withdrawn before becoming **fully vested**.

## Fully Vested

A stake amount that has completed its time-lock. A penalty will no longer be incurred on this stake if withdrawn.

## Boost

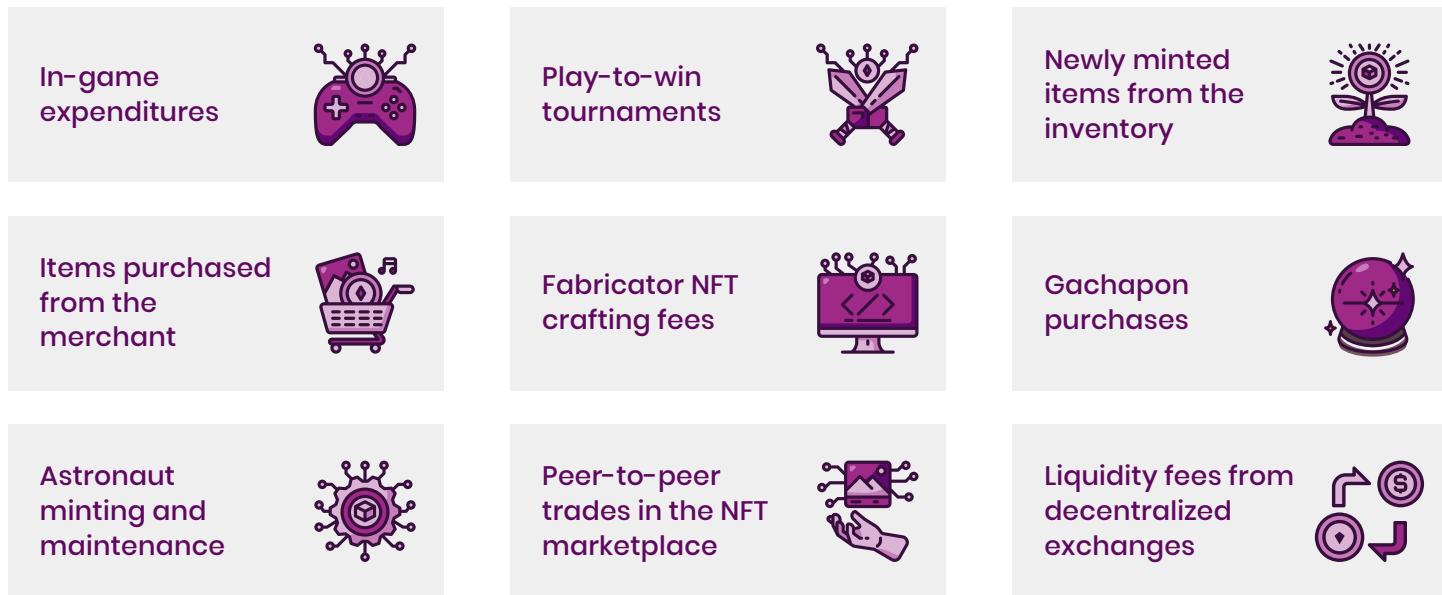
An optional selection to increase a staker’s LP weight within a Teller. This will increase the staker’s rewards per block.

## Early Withdrawal Penalty

Removing vested funds before they are fully vested will incur a penalty equal to 50% of the vested bonus percentage charged to the vested amount.

# COLLECTING FEES

The Generator collects fees from a variety of player and developer sources in the Vidya ecosystem:



## Fee Allocation

As fees are collected in the Generator, they are deposited into an emissions Vault for distribution to stakers. Larger stakes with longer commitments receive a proportional weighted bonus.

## Staking Fees

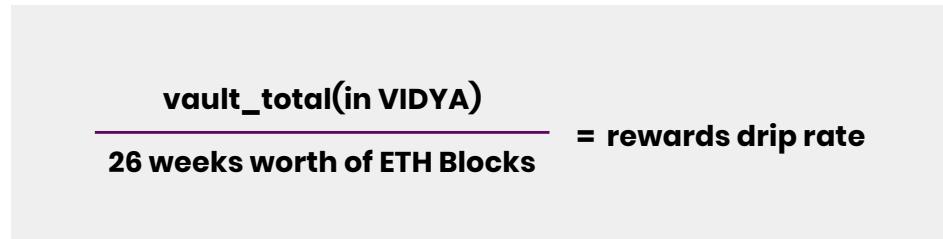
There is no entry or exit fee for standard, uncommitted stakes. There is a substantial exit fee that reduces over time for vested positions if withdrawn prior to becoming fully vested.

## Rewards

The distribution rate is calculated as frequently as once per week and payouts are determined by a period of six months. This ensures rewards are never over-allocated and will always pay stakers.

## Understanding the rewards calculation

The Vault total grows as fees are collected. When the calculation function is unlocked after a set timeline (~7 days) and a user interacts with the Generator, the calculation is made.



Rewards are paid to stakers in Tellers at the current **rewards drip rate** until the calculation function is called again, ~7 days. A staker's specific VIDYA rate is proportional to the priority of their Teller and their proportion of staked LP within the Teller, similar to yield farm distributions.

# STAKING IN THE GENERATOR



## Insert Coin LP Staking

Once the Generator is live, stakers will be able to stake an LP token in one of the Tellers. The reward for staking LP tokens is VIDYA. Stakers will need to press the claim rewards button to send them to their wallet. Rewards are not automatically distributed to staker wallets and must be claimed before exiting a commitment.

### LP EXAMPLES:

- ✓ VIDYA-DAI
- ✓ VIDYA-ETH
- ✓ DARK-ETH

## Staking as a Service

Team3D can add new liquidity pairs and Tellers to support and promote partnered projects. The capability of adding additional Tellers creates a mutually beneficial avenue for the communities and owners of other projects to implement a hassle-free staking system for their tokens. Having a supported Teller within the Generator increases and diversifies the liquidity of VIDYA and the paired or partnered token, benefiting holders of both tokens by reducing the number of liquid tokens held that could potentially result in selling pressure and incentivizing those who deposit or commit their tokens with VIDYA rewards over time.

Earning tokens by storing them instead of selling them means users no longer have to liquidate their token holdings to capitalize on their positions. Compounding a stake permits the application of long-term strategies to increase a user's weight in the platform in exchange for loyalty.

To request the addition of a partnership Teller within the Generator, project owners must contact [hello@team3d.io](mailto:hello@team3d.io) with all relevant project details. To be considered, projects must have undergone at least one professional audit. Following an internal vetting process, Team3D may decide whether or not to add the requested Teller at their discretion, regardless of the fulfillment of initial requirements.

# POWER-UP VESTING

In addition to the standard staking feature, stakers can time-lock their LP tokens for a selected *vesting period*. During this time, boosted stakes will receive a higher weight within a Teller based on a vesting schedule. Meaning \$1 unboosted will receive less rewards than \$1 boosted. As there is a substantial gain to time-locking LP tokens, there is also a substantial loss if LP tokens are removed before they are fully-vested. This penalty is described below under “Early Withdrawal Penalty”.

VESTING SCHEDULE		
Vesting Period	Earnings Boost	Early Withdrawal Penalty
		Earning Boost ..... 2
7 Days	0.40%	0.20%
14 Days	1.00%	0.50%
30 Days	2.90%	1.45%
90 Days	9.90%	4.95%
180 Days	22.0%	11.0%
365 Days	50.0%	25.0%

## EXAMPLE

Assume there are only 2 individuals with equal LP staked in a Teller that distributes 10 VIDYA per block.

Staker A is boosted.

Staker B is not boosted.

Normally the emissions would be split 5 and 5 between the stakers.

Since Staker A is boosted, A will receive higher than 5 VIDYA and the non-boosted staker will receive less than 5 VIDYA.

## Compounding a Vested Bonus

At the end of a vesting period stakers are able to withdraw LPs without penalty. Withdrawing this LP will remove the earned bonus percentage. Stakers may also choose to maintain the vested position with the earned bonus percentage. Depositing new LP tokens into an already vested position will compound the effects of the commitment bonus. Stakers can vest for a period of their choosing and compound their bonuses.

## EXAMPLE

After vesting for 1 year, a staker chooses to vest for 1 more year.

Their bonus is now 125%  
(1.5 Multiplier x 1.5 Multiplier  
= 2.25 Multiplier)

## Depositing Additional Funds to a Vesting Stake

If an individual should add additional funds to a position that's *not fully vested*, then the Generator will reset their vesting timer on a weighted average basis. Take the average time remaining of each deposit based on the weight of each deposit. This means of vesting timer calculation prevents long-term abuse where individuals stake small numbers for long periods early, then dump significant funds towards the end of their vesting schedule.

## Player has left the game (Early Withdrawal Penalty)

Withdrawing before the vesting period is complete will incur penalties. An early exit fee of **half** your boost percentage will be charged to stakers from their initial LP tokens staked.

A **minimum** of 90% of this penalty will be distributed to current stakers within the Teller. Team3D can decide to allocate the remaining 10% for other purposes or distribute it to current stakers.

**Caution:** Withdrawing from a vested position before completing the vesting period will remove all of the staker's LP tokens from the Teller. Any VIDYA not collected prior to early withdrawal will not be collected by the staker, they will be sent back to the Generator. **Users must collect rewards prior to early ending of commitment and withdrawal.**

### EXAMPLE

A staker vesting 0.10 LP for one year receives a 50% boost on the VIDYA rate.

After 3 months, the staker decides to remove the LP for other financial goals.

The initial 0.10 LP incurs a 25% (0.025 LP) fee for early withdrawal.  
(Earnings Boost:  $50 / 2 = 25\%$ ,  $0.10 - 0.025 = 0.075$ )

The staker will be able to withdraw 0.075 LP and any boosted VIDYA rewards collected during those 3 months.

Crypto markets are volatile. It may be in the staker's best interest to withdraw early from a vested deposit. The Generator will penalize appropriately. If this number is too steep for you, think twice before selecting a vested boost.

## Staker Rewards

Staker rewards are calculated at time of claim based on the following formula:

$$\frac{\text{User \%} \times \text{Priority} \times \text{Current VIDYA rate} \times (\text{now} - \text{Last Claim})}{\text{Total Priority}}$$

$$\text{User \%} = (\text{User Weight} \div \text{Total Weight})$$

$$\text{User Weight} = (\text{Deposit} \times \text{Commit Bonus}) \div (\text{Total Weight} \times \text{Teller LP})$$

$$\text{Total Weight} = \text{Sum of Users Weight}$$

$$\text{Current VIDYA rate} = (\text{Vault VIDYA} \div 6 \text{ months})$$

Fees are collated in the Vault from decentralized exchanges and applications across the platform for distribution to active stakes. While users aren't required to commit their deposits in order to earn rewards, commitment bonuses for locked stakes increase earning rates exponentially according to the amount staked and duration of the selected vesting period.

Each Teller represents a unique liquidity pair in the Generator and is assigned a priority according to the popularity and importance of its corresponding pair. The rate of reward is calculated using the above equation – taking into account a user's weight against the total weight via their deposit against the total amount deposited by all active stakes, their commitment bonus (if applicable), the amount of VIDYA tokens held in the Vault divided by the 6 month distribution schedule and the priority of the Teller users are staking within.

The Generator collects trading fees from decentralized exchanges but may also reap the benefits of future projects, with volume in Ethereum or other tokens used for buybacks of VIDYA, which can then be deposited in the Vault for distribution. Volume in VIDYA from the usage of decentralized applications on the overall Vidya platform can be deposited into the Vault as-is. A small fee is also applied to all sales of Vidya NFTs on Opensea, which is then deposited into the Vault upon reaching a reasonable threshold for distribution.

# GENERATOR MAINTENANCE

## Admin Authority

Team3D can add LP pairs as opportunities grow and remove LP pairs when they outlive their utility. These are defined as Tellers. In the event of a Teller being removed, boosted stakers can withdraw their deposit without penalty. Team3D can also set a “purpose” for 10% of the penalty fees, which can be allocated to a developer wallet for overhead, continued development, easter eggs, and community rewards. If a purpose is set, the remaining 90% of LP token penalty fees are distributed among active stakers. The default is no purpose set, with 100% of penalty fees accrued to active stakers.

## Beyond the Generator

After building a foundation of VIDYA LPs, we will launch LPs across multiple decentralized exchanges. As the project expands, we will explore options to implement cross-chain accessibility to the VIDYA platform. Dedicated stakers with significant commitments may also be considered for exclusive item drops, whitelists, launchpads, and other rewarding opportunities that arise in the future.

The Vidya platform is built upon interdependencies of decentralized applications, exchanges, and games, which may utilize the Generator and user positions within it in unique ways aside from allocating token volume to stakers.

## Monkey Wrench & User Agreement

If ever a bug of substantial severity is discovered by an audit or otherwise, it may necessitate the deployment of fixed Generator contracts and cancellation of previous Tellers within the old contract to prevent the influx of any additional users during this period. As outlined above, if a Teller is removed and the integrity of any held LP tokens remains intact, users may withdraw any outstanding deposits without incurring any penalties if such an event were to occur.

By using the Generator, you are in agreement that Team3D and Vidya cannot be held liable for any losses accrued notwithstanding any misuse or exploitation of the Vault or Teller contract code and/or any logical and mathematical precepts within them. Cryptocurrencies and decentralized applications are inherently volatile and the open source software they operate upon carries an inherent risk of exploitation which can result in the loss of some or all of the funds they manage. User discretion is advised when interacting with any and all products of this nature. Software audits and certifications serve as a guideline to user security and are by no means to be taken as a guarantee or assurance of safety and/or security.

The revolution will not be notarized.

# CONTRACT FUNCTIONS

## Vault Contract

### OWNER/ADMIN FUNCTIONS:

---

#### **addTeller(address \_teller, uint256 \_priority)**

- ✓ Grants access to the Teller to distribute funds to Teller's providers
  - ✓ \_priority states the ratio of this Teller in comparison to others
  - ✓ Emits NewTellerAdded(address, uint256)
- 

#### **changePriority(address \_teller, uint256 \_priority)**

- ✓ Changes the priority of the Teller in the Vault
  - ✓ Emits TellerPriorityChanged(address, uint256)
- 

#### **payProvider(address \_provider, uint256 \_providerTimeWeight, uint256 \_totalWeight)**

- ✓ Calculates and sends the appropriate amount to the provider
  - ✓ Emits TellerPriorityChanged(address, uint256)
- 

### INTERNAL/PRIVATE:

---

#### **calculateRate()**

- ✓ Calculates the amount of Vidya to disperse each block upto 6 months worth of blocks
  - ✓ Called automatically once a week
  - ✓ Emits VidyaRateCalculated(uint256)
- 

### PUBLIC:

---

#### **calculateRateExternal()**

- ✓ Allows anyone to calculate the dispersion rate if it has already been a week.

## Vault Contract

### OWNER/ADMIN FUNCTIONS:

---

#### toggleTeller()

- ✓ Determines if the Teller is active or inactive.
  - ✓ Will set a TellerClosedTime for the user to claim up to.
  - ✓ Emits TellerToggled(address, bool)
- 

#### addCommitment(uint256 \_bonus, uint256 \_days, uint256 \_penalty, uint256 \_deciAdjustment)

- ✓ Sets a new commitment struct to be used for commitments by providers.
  - ✓ Emits NewCommitmentAdded(uint256, uint256, uint256, uint256)
- 

#### toggleCommitment(uint256 \_index)

- ✓ Turns a commitment on or off for future use.
  - ✓ Emits CommitmentToggled(uint256, bool)
- 

#### setPurpose(address \_address)

- ✓ Sets the address to allocate 10% of future penalty fees to
  - ✓ Emits PurposeSet(address)
- 

### PUBLIC/PROVIDER:

---

#### depositLP(uint256 \_amount)

- ✓ As long as Teller is open user can deposit any amount of the targeted LP
  - ✓ Runs the claim function if the user has any amount to be claimed
  - ✓ Checks to see if user commitment time is over
  - ✓ Emits LpDeposited(address, uint256)
- 

#### withdraw(uint256 \_amount)

- ✓ Checks to ensure the user doesn't have enough tokens to withdrawal
- ✓ Runs the claim function
- ✓ Adjusts the user's weight and other attributes accordingly.
- ✓ Checks to see if user commitment time is over
- ✓ Emits Withdrew(address, \_amount)

### **commit(uint256 \_amount, uint256 \_commitmentIndex)**

- ✓ Checks to see if user has the amount of tokens to commit
  - ✓ If user is committing more tokens to an already committed amount calls function calculateNewEndTime
  - ✓ Checks to see if user commitment time is over
  - ✓ Emits Committed(address, uint256)
- 

### **breakCommitment()**

- ✓ Will break the commitment transferring the penalty to other stakers
- ✓ User will receive all of the deposited LPs minus penalty fee
- ✓ If purpose is set then 10% will be sent to the devAddress
- ✓ Emits CommitmentBroke(address, uint256)

## **INTERNAL:**

---

### **claim()**

- ✓ Calls the Vault, payProvider
  - ✓ Emits Claimed(address, true)
- 

### **commitBonus(uint256 \_commitmentIndex, uint256 \_amount)**

- ✓ Calculates the bonus received
  - ✓ Sends it to the commit function
- 

### **calculateNewEndTime(uint256 \_oldAmount, uint256 \_extraAmount, uint256 \_oldEndTime, uint256 commitmentIndex)**

- ✓ Uses the weighted mean average to allow users to continue to add to their commitments
  - ✓ Returns the new end for the commitment
- 

## **VIEW:**

---

### **userLPTokens()**

- ✓ Returns the amount of total tokens the user has in the pool

## STRUCTS:

---

**Provider{uint256 LPDepositedRatio, uint256 userWeight, uint256 commitmentIndex, uint256 committedAmount, commitmentEndTime}**

- ✓ LPDepositedRatio is used to ensure the provider earns from others penalties and that they will receive what they put in.
  - ✓ userWeight is equal to the amount deposited until user starts to commit amount then giving bonus to the userWeight
  - ✓ commitmentIndex which commitment chosen
  - ✓ committedAmount equal to the amount of tokens committed
  - ✓ commitmentEndTime when the user can claim without penalty
- 

**Commitment {uint256 bonus, uint256 duration uint256 penalty, uint256 deciAdjustment, bool isActive}**

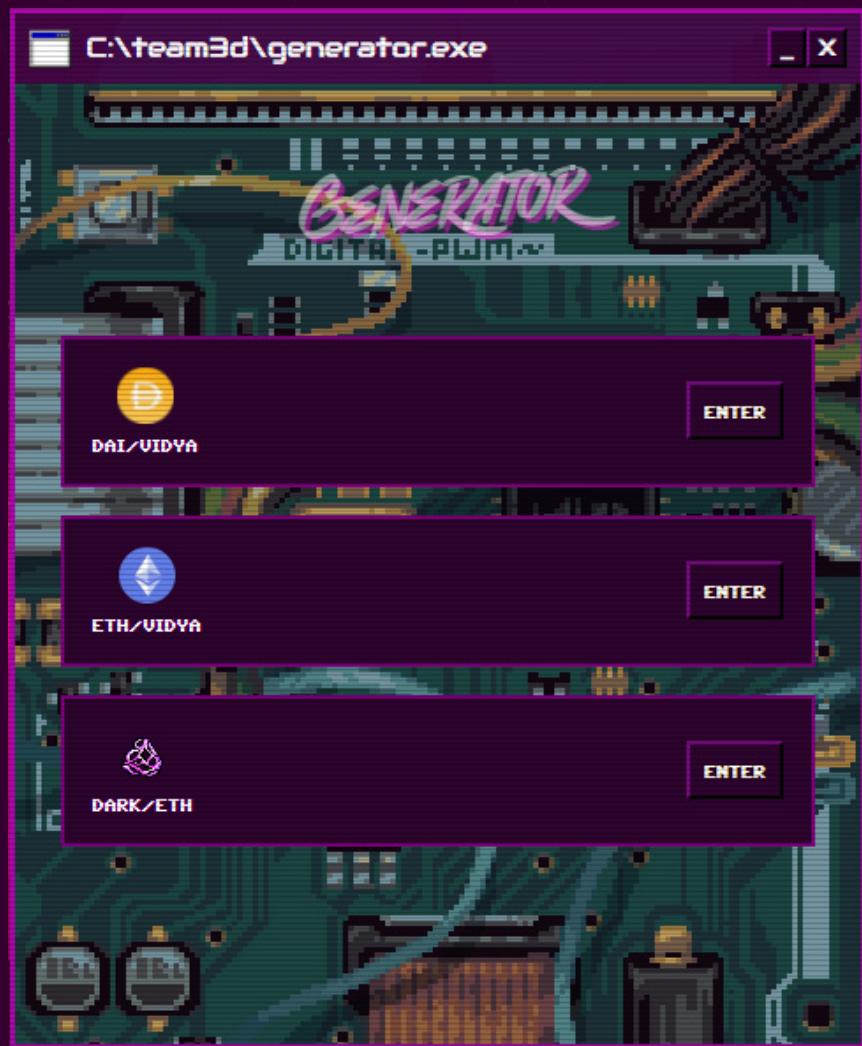
- ✓ bonus is the amount given as a bonus prior to being deciAdjustment
  - ✓ duration calculated in days, the length the user committed amount will be locked for
  - ✓ penalty, the % of tokens to be deducted from the user if they fail to reach the end of the commitment
  - ✓ deciAdjustment used to calculate how many decimal places are in the bonus and penalty.  
i.e. Bonus 500, Penalty 250, DeciAdjument 1000, is 50% bonus, with 25% penalty.
  - ✓ isActive determines if the current commitment isActive.
- 

**PROVIDERS CAN ONLY HAVE ONE TYPE OF COMMITMENT AT A TIME.**



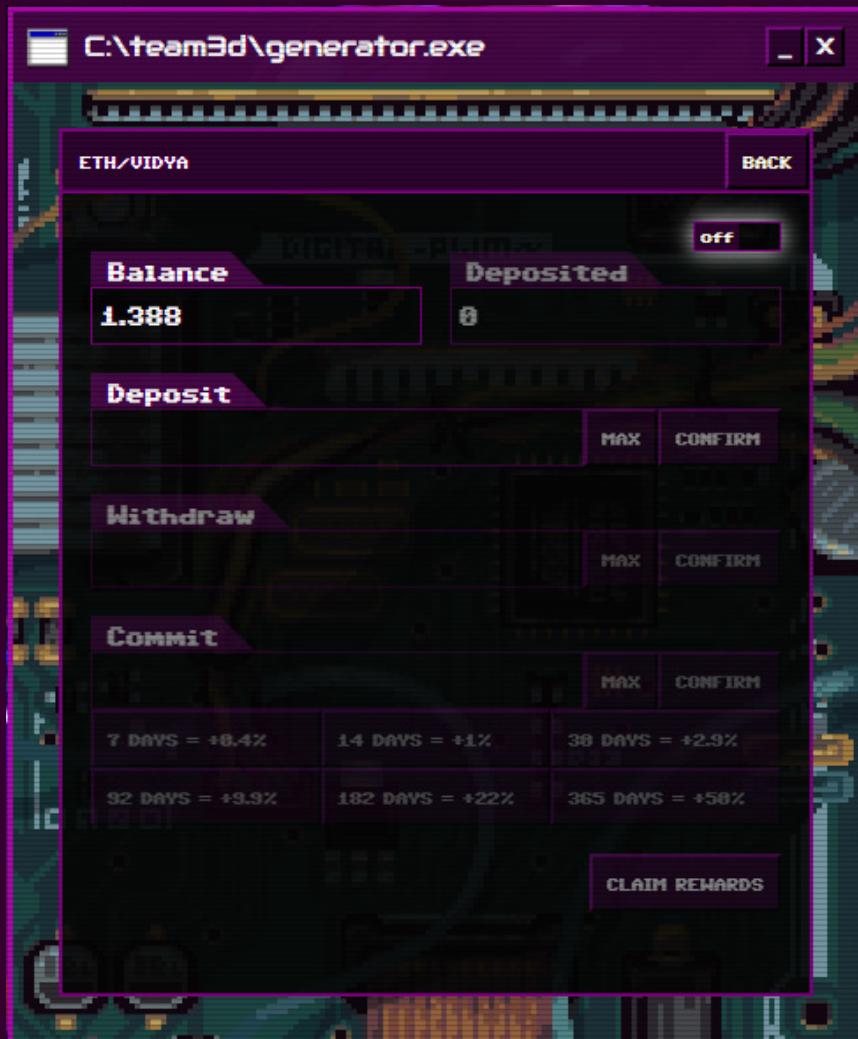
# USER INTERFACE *GUIDE*

# SELECTING A TELLER



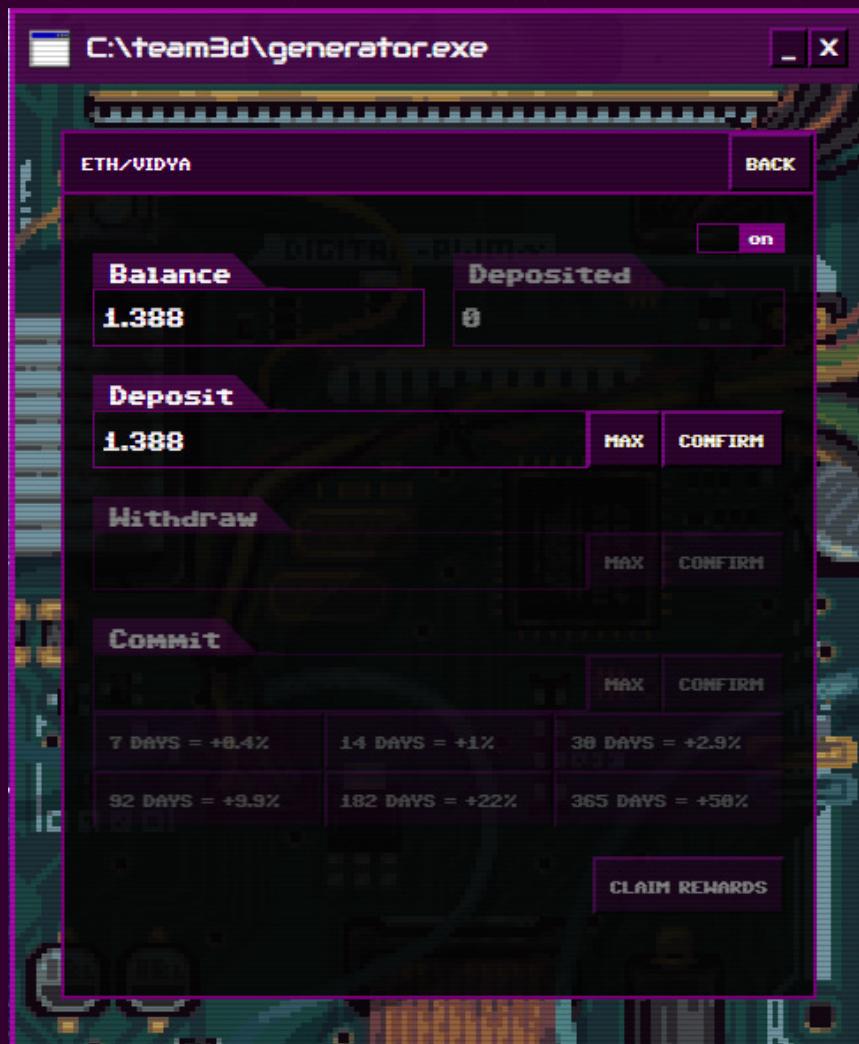
In order to use the Generator, first select your Teller from the list of pairs. This must correspond with the pair you have added liquidity for, as only LP tokens for that pair can be staked in their corresponding Teller.

# APPROVING LP TOKENS



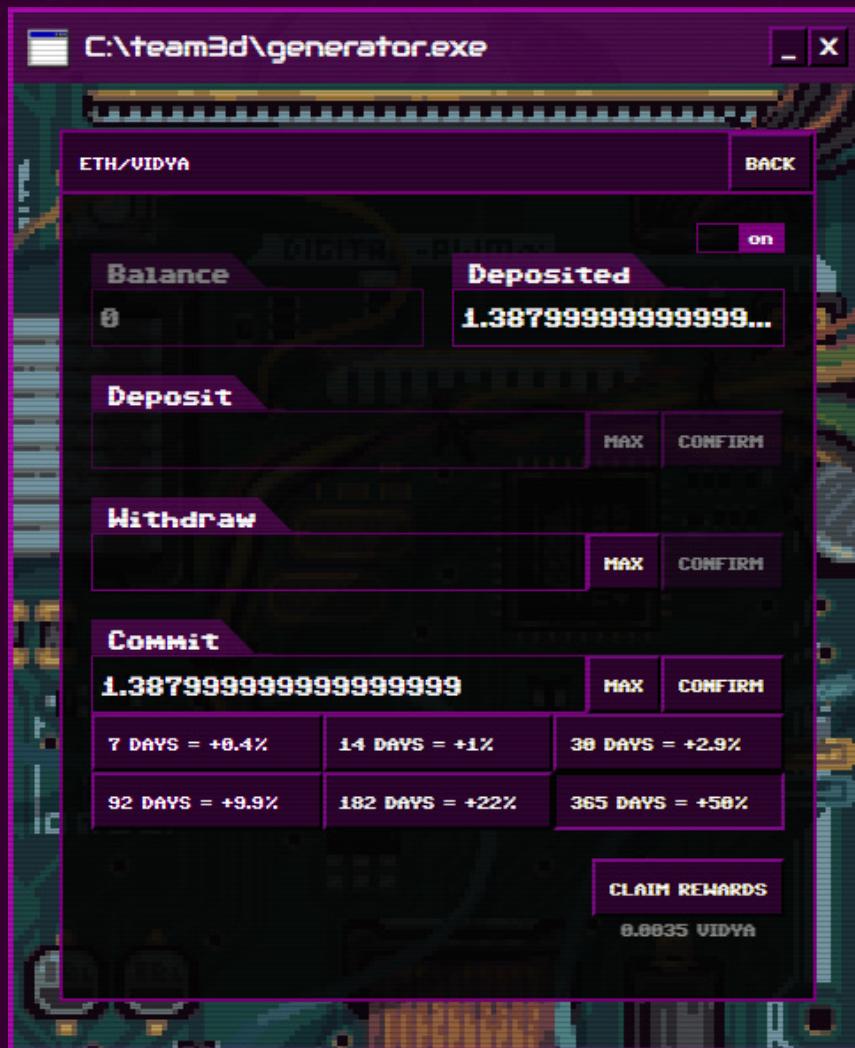
Once inside the Teller for your desired pair, approve spend for your LP tokens by clicking the on/off switch on the upper righthand corner of the staking window and confirming the transaction.

# DEPOSITING LP TOKENS



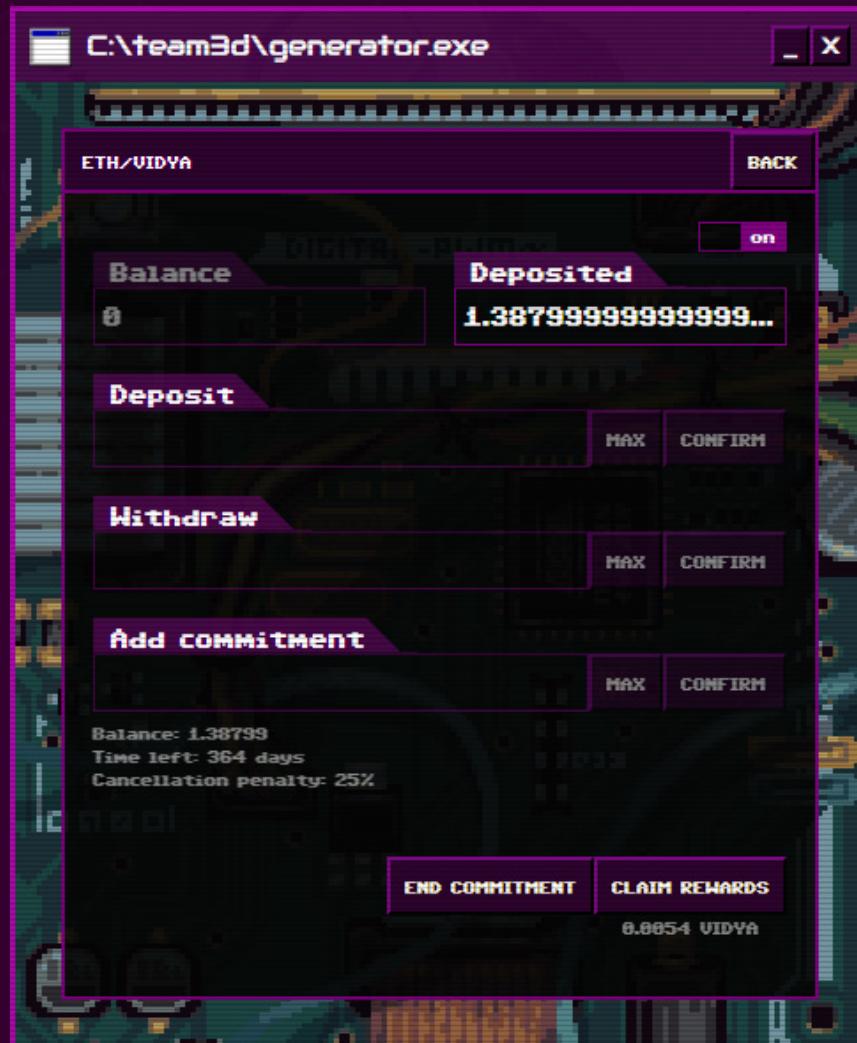
Now that your tokens are approved, the switch will read “on” and you can now input a desired deposit amount and click “confirm” to begin staking.

# COMMITTING AND CLAIMING



Once your deposit is processed, you can now claim rewards for your active stake. You may decide to commit your deposit for a duration of time, receiving a bonus depending on the selected duration of your stake. This bonus is applied to the weight of your stake as opposed to a flat percentage bonus on your initial deposit. In order to commit, input the desired amount to commit, select your desired duration and click “confirm” beside the commit input window.

# COMPOUND AND CHILL



If you've committed your tokens, information regarding your committed balance, remaining commitment duration and cancellation penalty will be made visible to you. Commitments can be compounded by adding additional LP tokens, but can only be committed for the duration selected in your previous commitment. Claim your accumulated rewards as you please and remain weary of ending your commitment early, as your penalty LP will be distributed among active stakers.



END  
OF DOCUMENT

GENERATOR

TEAM3D  
[Hello@team3d.io](mailto>Hello@team3d.io)