
finance.vote

Christopher Smith, Nick Almond, Naved Ali

Oct 21, 2022

BACK ROOM:

| | | |
|-----------|--------------------------------------|-----------|
| 1 | Setting up | 3 |
| 1.1 | Frontend | 3 |
| 1.2 | Backend | 3 |
| 2 | Minting Identity | 5 |
| 2.1 | How does it work? | 5 |
| 3 | How To Vote | 7 |
| 4 | Proposal view | 9 |
| 4.1 | Tabs | 9 |
| 5 | How to add submission | 13 |
| 5.1 | Text submissions | 13 |
| 5.2 | Media submissions | 14 |
| 5.3 | Mixed submissions | 14 |
| 6 | Create new realm | 15 |
| 7 | Creating Proposal | 17 |
| 7.1 | Entering New Proposal Mode | 17 |
| 7.2 | Steps to Creating Proposal | 17 |
| 7.3 | Proposal elements | 18 |
| 8 | Strategies | 25 |
| 8.1 | Main params | 25 |
| 9 | Create new task | 27 |
| 9.1 | Ranking | 28 |
| 9.2 | Submissions limits | 28 |
| 10 | Markdown syntax | 29 |
| 10.1 | Text formatting | 29 |
| 10.2 | Headings | 29 |
| 10.3 | Links | 30 |
| 10.4 | Images | 30 |
| 10.5 | Lists | 30 |
| 10.6 | Tables | 31 |
| 10.7 | Line break | 31 |
| 11 | Indices and tables | 33 |

12 Our products **35**
 12.1 Back to main page 35

Influence is designed for token weighted decentralised content curation and rough consensus formation around anything from tuning monetary policy parameters to meme markets. This is where DAO members deliberate and decide.



SETTING UP

1.1 Frontend

Repository link: [Influence](#)

1. In the influence terminal:

```
git checkout develop
```

2. In the main influence folder create **.env.local** file and put in there settings copied from **.env.example**. If you'd like to use local backend, remember to comment this line from **env.local**: `REACT_APP_API_URL='https://test.influencebackend.xyz'`
3. Install dependencies:

```
npm install
```

4. Run application:

```
npm start
```

1.2 Backend

Repository link: [Influence hub](#)

1. In the influence-hub terminal:

```
git checkout develop
```

2. Install dependencies:

```
npm install
```

3. In the main influence-hub folder create **.env.local** file and put in there settings copied from **.env**.
4. Add your wallet address into `admin_init` file under `migration` folder. Make sure export line is uncommented.
5. Run application:

```
docker-compose up --build
```

6. If after the building there are no documents in database under `strategies` collection, make sure `strategies_init` file under `migration` folder has uncommented export line.

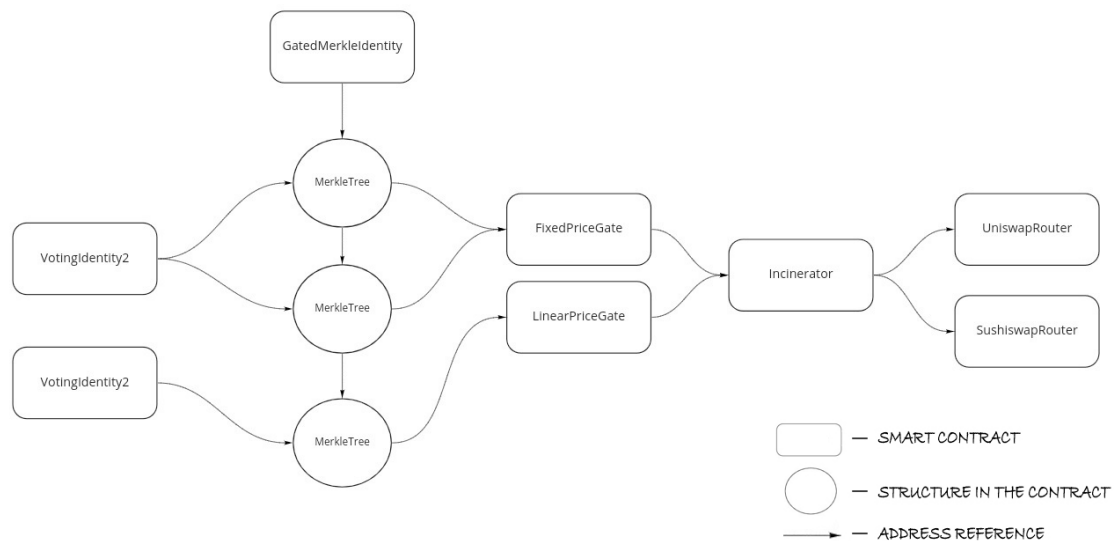
MINTING IDENTITY


Minting an identity is an anti-sybil protection.

Learn more about [minting here](#).

2.1 How does it work?

In purpose of voting, first thing to do is to mint an identity in cost of burning some **\$FVT** to gain voting power. And that is what it can be seen from user interface. But what is going on under the hood?



This scheme depicts mechanism standing behind minting an Identity within  **Influence.vote**.

Our starting point is **GatedMerkleIdentity** contract, which remembers all Merkle trees root hashes and verifies whether Merkle proof delivered by user is in any of the Merkle trees. More on [Merkle tree here](#).

Adding tree hashes to **GatedMerkleIdentity** requires passing NFT address and gate address, both of which could be get via contracts.

GatedMerkleIdentity connects to the contract of token, which is desired by user to get.

In Influence.vote the only available token's contract right now is **VotingIdentity2**, which creates an Identity and sets NFT addresses for them, also it transfers NFT from one user to another and shows minter's NFT balance. More on [Identities here](#).

Price of minting an Identity is dynamic and determined by amount of Identities already bought. That's why there are two price contracts. **FixedPriceGate** always returns basic price, while **LinearPriceGate** returns price increasing in time.

Incinerator takes a token address and some ether. It uses the ether to purchase the token and burn them (remove them from supply permanently).

The last two contracts indicated in the schema are used for trading from a smart contract.

UniswapRouter and **SushiswapRouter** are contracts that works nearly the same and are used to safely swap tokens to and from different assets.

Contents

- *How To Vote*
- *Proposal view*
 - *Tabs*

HOW TO VOTE


1. Connect your wallet.
2. Enter proposal and read its description.
3. If needed, make sure you're on proper chain.
4. Allocate your votes (quadratic strategies) or choose an option (binary strategies).
5. Hit **Vote** button.
6. Justify your vote (optional).
7. Hit **Submit vote** button.
8. Confirm voting in MetaMask.
9. Voila!

If you're still confused, checkout **a quick walkthrough video** down below and [Voting Explained chapter](#)

Voting notes

To submit a vote you need to click into vote field and type in a value or simply use arrows shown when you hover over a field. **Voting system is based on quadratic voting system, so every value typed in is squared.** This mechanism allows to vote more than once(allocate more than 1 vote) on a particular choice, but the stronger(more valuable) your votes get, the more it costs you. You can spend all of your vote power over one choice or split it over all available choices - it is up to you (but you have to distribute votes the way its total value doesn't exceed your vote power).

For some proposals (depends on strategy used), there is a possibility of **negative voting**. If you definitely don't want particular choice to win you can try to weaken its position. Typed in value is still squaring and counted out of your vote power.



influence.vote

← finance vote

Open

Technical Solution

What technical solution should be developed next?

- Kombucha - Chaining parts of the system
- Green Tea - Add user's profile customization
- Coffee - Forum for users and developers


Aggregated Votes My Vote Opinions Votes List **1**

MY INFLUENCE

INFLUENCE

45.00039275956284 \$!

ALLOCATION



VOTE SUBMITTED

| | |
|------------|---------------------------------|
| #Coffee | <input type="text" value="22"/> |
| #Green Tea | <input type="text" value="13"/> |
| #Kombucha | <input type="text" value="-4"/> |

NEGATIVE VOTES ARE POSSIBLE

DETAILS

| | |
|-------------|------------------|
| IPFS: | q9n1f8t5p... |
| Strategy: | Token weighted |
| Start Date: | 10/08/2021 08:00 |
| End Date: | 10/08/2021 10:00 |
| Snapshot: | 10,808,723 IS |
| Results: | JSON IS |

PROPOSAL VIEW

Main view of proposal contains a title and a description with all the information about ballot and ballot's choices. Details contain proposal hash (IPFS field), voting strategy, start and end voting date, block height (Snapshot field), countdown timer and voting result data in JSON and CSV.

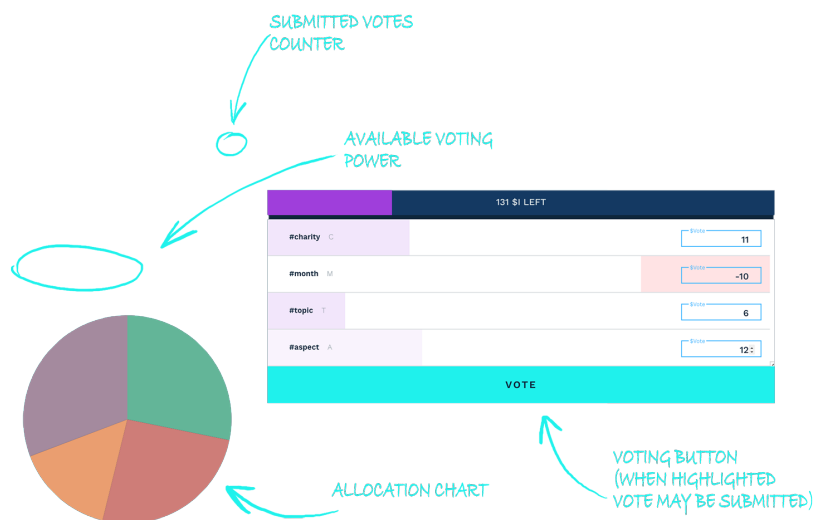
Below that, there are four tabs:

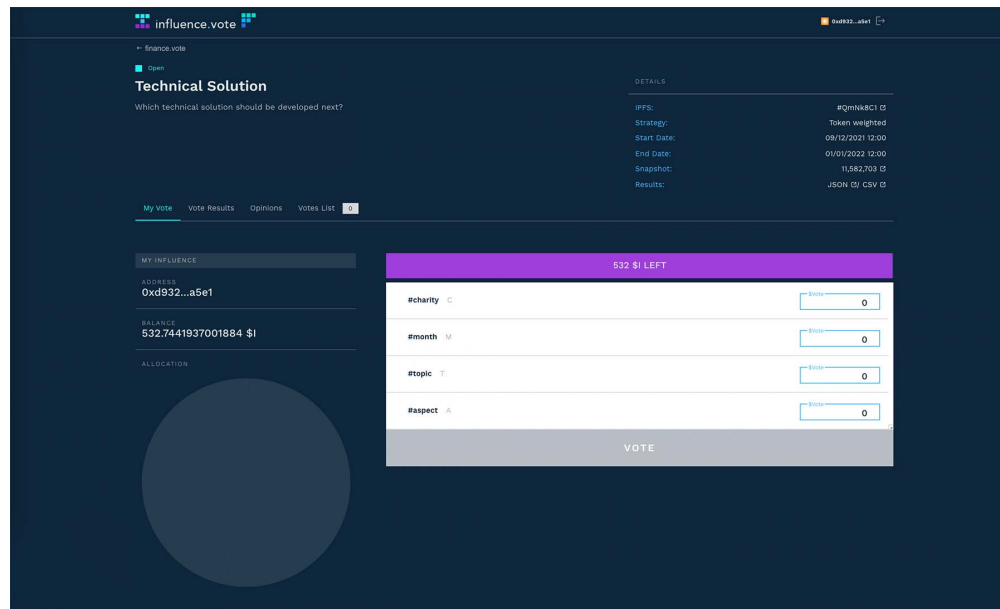
- My vote
- Vote Results
- Opinions
- Votes list (and submitted votes counter)

4.1 Tabs

4.1.1 My Vote

My Vote tab contains identity picker (only in multi identity strategies proposals), account available votes, distribution of votes (allocation field) and vote form.

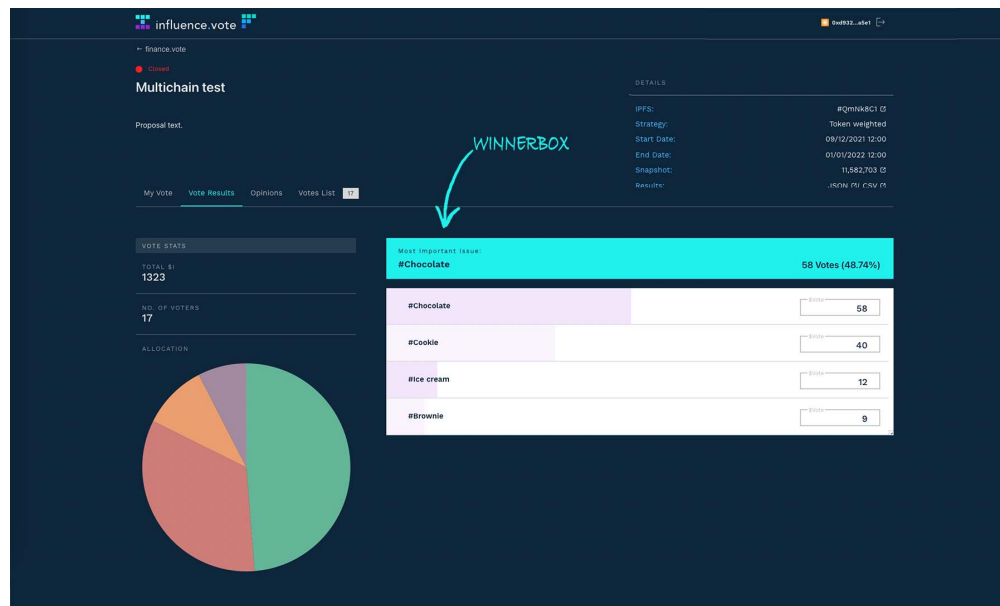




4.1.2 Vote Results

This tab shows current or final results of the ballot.

If a winnerbox were switched 'on' during proposal creating, it will be shown here as well.



4.1.3 Opinions

In the opinions tab you can see justifications for votes submitted by other users. Filtering by choices names is a simple way to filter out only interesting you choices.

Sorting options works either for all opinions or filtered out choices. You can sort by oldest or newest opinions and by \$I Power.

Technical Solution

What technical solution should be developed next?

1. Kombucha - Chaining parts of the system
2. Green Tea - Add user's profile customization
3. Coffee - Forum for users and developers

My Vote | Vote Results | **Opinions** | Votes List 4

Opinion 1: [0xd932...a5e1](#) / 4 months ago
#coff 25 \$I
 A good idea, this may have a positive effect on the productivity and efficiency of work.

Opinion 2: [0xefab...b186](#) / 4 months ago
#coff 289 \$I
 Coffee only

Sorting Options: SORT BY Old (dropdown arrow)
 TAGS: #COFF x (dropdown arrow)
 COFF TEA KOMB

Handwritten green text: "YOU CAN FILTER OPINIONS BY TAGS" and "SORTING OPTIONS" with arrows pointing to the filter and sort controls.

4.1.4 Votes list

Votes list tab informs how much voting power were spend by each token ID/wallet address (depending on strategy)

| My Vote | | Vote Results | | Opinions | | Votes List | 4 |
|---------|-------------------------------|--------------|--|----------|--|-------------------|---|
| ID NO. | | | | | | \$I POWER | |
| | 0xd932...a5e1 | | | | | 45 \$I | |
| | 0xefab...b186 | | | | | 289 \$I | |
| | 0x5c99...f61c | | | | | 101 \$I | |
| | 0xb9d9...444e | | | | | 121 \$I | |

HOW TO ADD SUBMISSION

Tasks are special proposals type, where users know only a question and their job is to propose an answer. Depend on task configuration, user may be able to add more than one submission per ID (token or NFT).

Task types:

1. text - you can add only text submissions
2. media - you can add only video, twitter or image submissions
3. mixed - you can add only text, only media or text and media at the same time

Steps:

1. Connect your wallet.
2. Enter task and read its description.
3. Hit **Create a submission** button.
4. Choose ID you want to send your proposal with.
5. Click **Next**.
6. Add **Title and content** (depend on task type).
7. Hit **Submit vote** button.
8. Confirm in MetaMask.
9. Voila!

5.1 Text submissions

Title and description are mandatory.

Description length depends on task configuration.

5.2 Media submissions

Influence supports YouTube, Vimeo, TikTok, Twitter urls and sending images (acceptable formats: **.jpeg, .jpg, .webp, .png, .gif**)

On submission creation, choose which media you would like to add and paste a media url or send an image.

5.3 Mixed submissions

This type leaves the most freedom. You can either add media or text, or both at the same time if you want. The only limitation is that you have to add at least something. And a title.

CREATE NEW REALM

Adding new realms is reserved for admins only.

To add a new realm choose add realm tile on the main influence site (if you don't see one, you don't have admin privileges).

Add new realm form basic elements:

1. Name field
2. Networks picker
3. Strategies picker
4. Members field
5. Logo height field
6. Logo width field
7. Realm logo uploader
8. Realm header uploader

All parameters set during creating a new realm are editable after the realm is created.

- You have to choose at least one network and one strategy to create a realm.
- Members field must be sent as a JSON array (at least one element):

```
["address1", "address2", "address3"]
```

- Recommended logo height and width is 150px. Graphics size shouldn't be higher than **1 MB**. Acceptable graphic formats: **.jpg, .jpeg, .png, .webp, .svg, .gif**
- There are additional fields, that are added to the form on chains adding. Those are [network name] contract address fields that you can leave empty.

Only the basic form elements, listed above, are mandatory to create a new realm

CREATING PROPOSAL

Contents

- *Creating Proposal*
 - *Entering New Proposal Mode*
 - *Steps to Creating Proposal*
 - *Proposal elements*
 - * *Proposal content*
 - * *Proposal voting choices*
 - * *Proposal details*
 - * *Draft mode*

7.1 Entering New Proposal Mode

1. On the main page choose interesting you realm.
2. Hit **New Proposal** button above the proposals list (only moderators can add new proposal).

7.2 Steps to Creating Proposal

A quick reminder on how to properly set up a proposal.

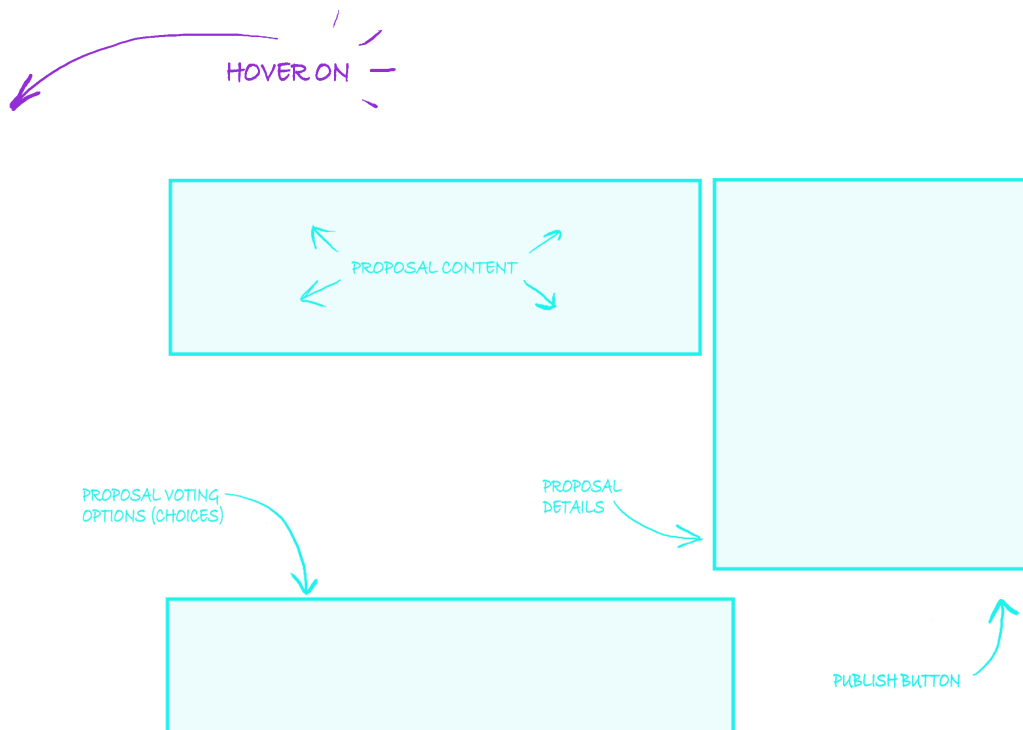
If you are not familiar with any (or some) of the points, we encourage you to read more in the further parts of this article.

1. Hit **New Proposal** button.
2. Fill in **title** (“Question” field) and **description** (“What is your question?”). You can use Markdown basic syntax to format your description (*read more*).
3. Set **start** and **end** dates.
4. Choose **strategy**.
5. Choose **proposal’s tag**.

6. **Optional:** Decide on proposal's options (or leave them as are):
 1. Which chains should be allowed
 2. Turn on winnerbox
 3. Restrict viewing results
 4. Decide who is eligible to vote (whitelist) or/and who is ineligible (blacklist)
7. Fill in voting options (**choices**). You can choose to fill in **choices descriptions** as well.
8. When everything is filled out we recommend to check everything because **once published there is no editing possibility**. You can decide to publish proposal in **draft mode**.
9. After making sure everything is correct you can finally hit **Publish** button. If all required fields are filled in properly, *loading screen and signature request from MetaMask will appear*.
10. **All done!** Proposal has been made and you've been sent to the main page. Created proposal should appear on the top of the proposals list.
11. If you've created proposal in a **draft mode**, decide on publishing it or deleting.

7.3 Proposal elements

When you first create a proposal You'll see this form:



7.3.1 Proposal content

Proposal content consists of the title (“Question”) and the description (“What is your proposal?”). You can use Markdown basic syntax to format your description ([read more](#)).

Below the question field there is a field reserved for a description of what your question concerns, try to clarify what you want to ask. You can also describe each choice you’ll put to the vote so everyone knows what it is about.

If you will need more space for the text area, use the expand handle.



Note: Remember to use less than 40 000 characters in your proposal description

7.3.2 Proposal voting choices

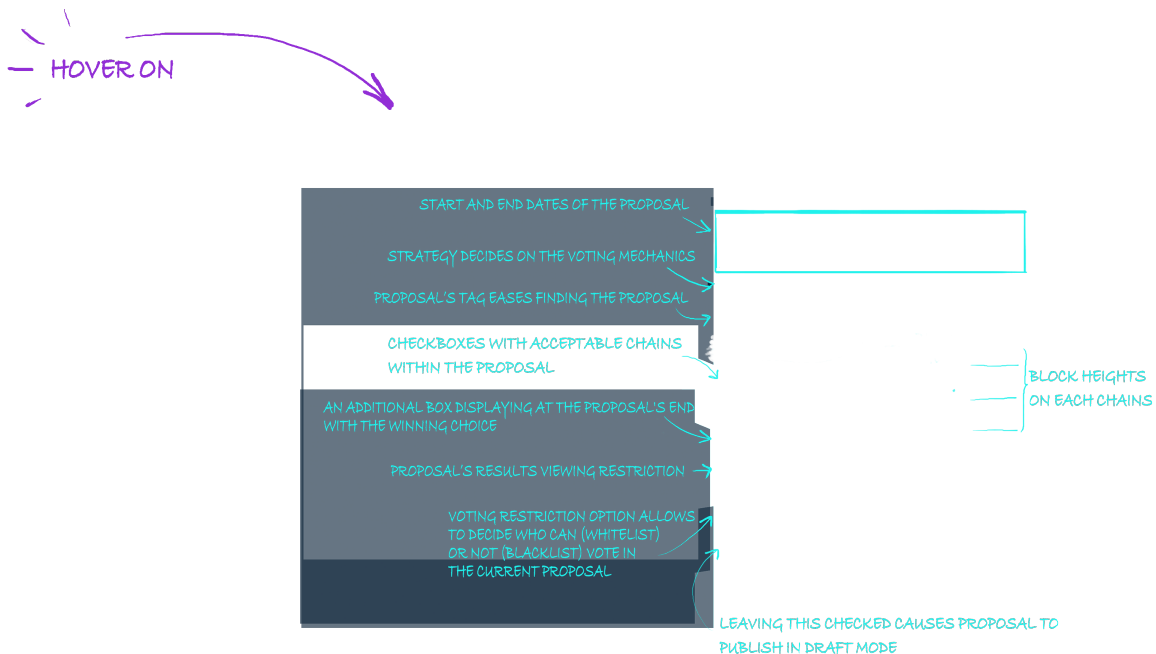
Filling in at least 2 choices is mandatory. They should relate to the proposal description.

To add more choices hit **Add choice** button, to remove choice use **X** mark next to the choice you want to remove.

Adding a choice description is optional. Descriptions are accessible through the voting component. Choices that have one are clickable and highlighting (check out the video for reference).

7.3.3 Proposal details

In the 'Action' field (Proposal's details) you decide when proposal starts and ends, on what kind of strategy it will run and what will be the proposal's tag. Below the graphic, you'll find a more specific description of each parameter.



Mandatory proposal's elements are:

- **start and end dates** (dates of voting start and end mustn't be the same)

- **strategy** - strategies are mechanics for proposals that determine the voting power of users and how they are allowed to cast their votes. Each strategy has different voting power calculations and relies on different sources (tokens, cryptocurrencies, NFTs). Each realm has a predefined set of strategies allowed to use in the proposals.
- **tag** - the proposal tag consists of up to 4 characters. Tags make it easier to find proposals in the list as well as categorize and organize them.
 - **You can choose from existing tags within dropdown menu** or create your own by hitting “Add new sign”. Choosing to add your own sign, a new window will pop up with name field and color picker.

- **snapshot** - it's the height of the block in the blockchain that was used to check account balance.

There are also a few additional setting options that could be used to control who can see and/or vote in the proposal and whether the proposal's winning option should be especially exposed after the end:

- **acceptable chains** - as influence is a multichain platform, those checkboxes allow user to decide which chains should or shouldn't be considered within the proposal. Some of the strategies sums up user balance within all chains, so those checkboxes help control it.
- **winnerbox** - it's a field that appears at the proposal's end and contains the name of the choice (or choices) that has won.

| Most Important Issue: | | |
|-----------------------|------|--------------------|
| #Green Tea | TEA | 12 Votes (100.00%) |
| #Green Tea | TEA | \$Vote 12 |
| #Kombucha | KOMB | \$Vote 0 |
| #Coffee | COFF | \$Vote 0 |

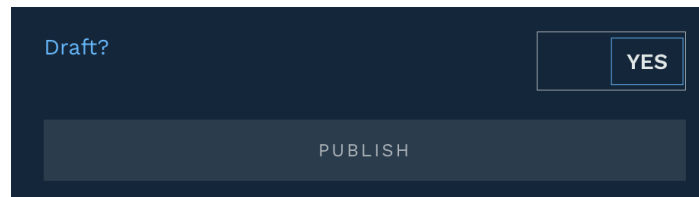
| 3 way tie: | | |
|------------------------------|---------------|------------------|
| #Coffee #Kombucha #Green Tea | COFF KOMB TEA | 3 Votes (33.33%) |
| #Coffee | COFF | \$Vote 3 |
| #Kombucha | KOMB | \$Vote 3 |
| #Green Tea | TEA | \$Vote 3 |

- **viewing restriction** - it's an option that restricts the preview of voting results for those who didn't meet the prerequisites (having a specific token or NFT in the wallet, determined by the strategy)
- **voting restrictions** are handled per chain (for example one address/token ID could be restricted only on mainnet but not on other chains). You can either type ID number/wallet address directly into the field or paste CSV formatted array:
 - **whitelist** - a list of IDs numbers or wallet addresses (**depending on strategy**) that are eligible to cast a vote:
 - **blacklist** - a list of IDs numbers or wallet addresses (**depending on strategy**) that are ineligible to cast a vote:

7.3.4 Draft mode

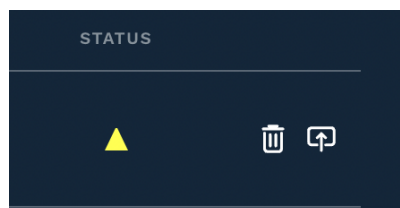
Before you publish your proposal you may want to share it with someone so they double check it. But since the proposal cannot be changed or deleted after publishing, you can publish a draft of it first.

All you have to do to achieve that mode is to simply leave the draft switch 'On' during proposal creating:

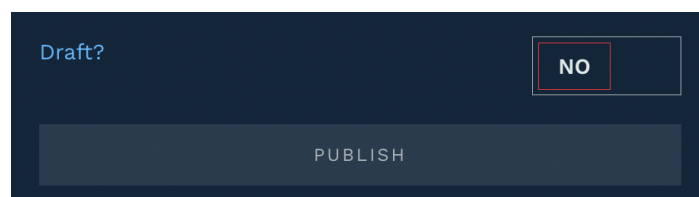


You may now share the draft with someone and be sure that no one will be able to vote while the proposal is in draft mode.

Once you'll decide that everything is correct, you can publish it (square with arrow icon) or if there were some mistakes, you can delete it (bin icon):



If you don't want to publish a proposal in a draft mode, remember to turn the draft switch 'Off' during proposal creation:



STRATEGIES

Influence supports various contracts, token standards and chains. Implemented different calculating methods called strategies help to handle such diversity and cover most user needs.

Strategies used in the app are built out of params. Those 6 of them (listed in the next section) create 6 basic strategies:

| Strategy | Base | Voting | Power Calculating |
|---------------------------------|-------------|-----------|-------------------|
| Token weighted quadratic voting | token-based | quadratic | not-aggregated |
| Token weighted binary voting | token-based | binary | not-aggregated |
| Quadratic voting | nft-based | quadratic | not-aggregated |
| Binary voting | nft-based | binary | not-aggregated |
| Aggregated quadratic voting | nft-based | quadratic | aggregated |
| Aggregated binary voting | nft-based | binary | aggregated |

Any other strategies are custom and require a separate approach to meet even the highest user requirements.

8.1 Main params

1. Base:

- **token-based**

takes user balance of ERC20 token and grants voting power based on that amount

- **nft-based**

grants voting power to those who own the expected NFT in their wallets

2. Voting method:

- **binary**

allows voting with the whole voting power at once on only one choice

- **quadratic**

allows voting on multiple choices while keeping the rule of quadratic voting at the same time (if you're not familiar with the idea, check it out [>here<](#))

3. Vote power calculating method:

- **aggregated**

it allows voting per wallet, meaning the user might vote only once, using concatenated assets at once

- **not-aggregated**

it allows voting per NFT ID, meaning each identity might vote differently

CREATE NEW TASK

The screenshot shows the 'CREATE NEW TASK' form in the finance.vote application. The interface is dark-themed. At the top, there's a navigation bar with 'VOTES' and 'SUBMISSIONS' links, a user profile '0xd932...a5e1' with a dropdown arrow, and a 'change account' button. The main form area is divided into two sections. On the left, under the heading 'Title', there's a text input field with the placeholder 'Add a description'. On the right, under the heading 'DETAILS', there's a form with several fields: 'Start Date:' with a 'Select start date' button, 'End Date:' with a 'Select end date' button, 'Ranking end date:' with a 'Select ranking end date' button, 'Max sub. length:' with a text input containing '10000', 'Task type:' with a 'Select...' dropdown, and 'Tag:' with a 'Tag' dropdown and a link 'Add new tag'. Below these, there are two checkboxes: 'ID holders' and 'Token holders'. Each checkbox has a 'Submissions per ID' or 'Submissions per address' label and a numeric input field set to '1'. At the bottom of the details section is a large 'PUBLISH' button. The footer of the page contains the text '2021 All rights reserved finance.vote v 1.5.3' and social media icons for Telegram, Twitter, and Medium.

Form fields:

1. Title & description
2. Start & end dates
3. Ranking end date - more on ranking further on
4. Max submissions length - how many symbols user may input into submission form
5. Task type - text, media, mixed (*more on task types* <<https://financevote.readthedocs.io/projects/influence/en/latest/content/submissions.html>> `_)
6. Tag - you can choose tag from the dropdown or create your own (*more on tags in the proposals view details*)
7. 'Open to' box fields - decide how many submissions can be added (*more on that further on*)

9.1 Ranking

The ranking is a special task mode that starts right after the task end and lasts till the ranking end date set on task creation is reached.

From task start till the ranking end, users may vote on the submitted submissions to decide which are the best, but after the task end cannot add any new ones. So ranking mode is basically created to give chance to those submissions that were added near the task end, so they can gain some votes too.

Voting on submissions:

9.2 Submissions limits

On task creation you can decide how many submissions the user may add to the task and what kind of assets are allowed - NFT, Token or both.

You cannot decide which NFT or which Tokens are allowed, you only decide on asset type. The system takes all assets that are allowed within that realm.

For some realms, multiple strategies are implemented, and some of them may be using different NFTs.

Since tasks are insensitive to strategies, there is no way to whitelist or blacklist any specific NFTs contract addresses.

MARKDOWN SYNTAX

While creating a new proposal, you will have to add a description that will be easy to read.

It could be tricky, especially when you have to explain it adequately.

So instead of loading people with long boring descriptions, you can use markdown syntax that helps you emphasize what's more important.

Check out markdowns supported by [influence.vote](#) down below.

10.1 Text formatting

- *italics text* - use single asterisk `*` around: `*italics*`
- **bold text** - use double asterisk `**` around: `**bold**`
- strikethrough - to get text strikethrough use double tilde `~~` around it: `~~strikethrough~~`
- *verbatim code* - use single backquote ``` around it: ``verbatim code``

10.2 Headings

To create a section or subsection within your description, you can use headings:

hint: the more hashes before the header, the smaller one.

- `# Header 1`
- `## Header 2`
- `### Header 3`
- `#### Header 4`
- `##### Header 5`
- `##### Header 6`

10.3 Links

Simply pasting in the source link works within influence.vote as desired, though there is a way of formatting them better.

Instead of long incomprehensible links, just put your link into brackets and write a short description before to format them nicely:

This link [Description](#) looks much better than <http://your.link>.

Usage: [Description] (<http://your.link>)

10.4 Images

Adding images into proposal is possible but limited. Markdown syntax allows to add graphics, but basics don't support size control.

Remember, this function is not dynamic and graphics may not look the same on every device.

To add an image use `![Caption in case it didn't show](http://example.graphic.com/elephant.png)`. Remember to put an alternative description into square brackets and image address into regular ones.

10.5 Lists

To create a beautifully look list with sub-items you just need to use symbols (*, +, -) and indentations for each level.

- **use * for main level**
 - use + for sub-items
 - and don't forget
 - **about indentations**
 - * sub-items of sub-items
 - * are possible
 - * just use – for them
 - * ...and of course a proper indentation
- you can always get back to the main level

Example:

```
* unordered list
+ sub-item 1
+ sub-item 2
  - sub-sub-item 1
```

10.6 Tables

To create a table you just need to use the following formatting. The first row content describes how column content will align, depending on using colons :

| Default | Left | Right | Center |
|---------|------|-------|--------|
| 12 | 12 | 12 | 12 |
| 123 | 123 | 123 | 123 |
| 1 | 1 | 1 | 1 |

10.7 Line break

To make a line break, simply type ---

It will do this:

We may not have covered all the possibilities.

If you need anything else than listed above, try it yourself. A preview is always available underneath the proposal description during proposal creation, so if you see the desired output, you can use it, and it will work!

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`

OUR PRODUCTS

-  launch
-  bank
-  markets
-  mint
-  yield

12.1 Back to main page

-  Home