

# Web3 Builders DAO-Rules



**1.0.0 version, by Luftjunkie**

## 🧠 First of all, what the fuck is a DAO?

A DAO (Decentralized Autonomous Organization) is a group of people on the internet who work together like a team or club — but instead of one boss making all the decisions, everyone gets a say.

They use special computer programs (called smart contracts) on the blockchain to make decisions, vote on ideas, and manage shared money — all in a way that's open, fair, and can't be secretly changed by just one person.

It's like a digital co-op, where the rules are written in code, and the members decide what happens together. 💬💰🤝

# Why implement the DAO into our community ?

Having an impact on the surrounding environment is definitely a desired thing. However in most of the discord servers, decisions are centralized mostly in the hands of few people and it even can lead to collapse of the server, so people will no longer be willing to contribute there. One way of solving it is via direct-democracy, where the users have equal votes and are eligible to decide about the changes that should have been performed. This solution might seem cool, but it's still flawed.

Why ? Because each vote is equal, there is no other thing taken into consideration as the single entity.

That's why introducing DAO would be a great solution, because the vote-power can be dependent on other variables than the amount of entities.

For non-technical users

Users could gain impact on changes, by owning a deserved amount of governance-tokens, depending on their contribution, problem solving in the community and also for the progress they could have made, that they could showcase on the discord-server.

It all can be greatly algorithmized, so that each individual will have the vote-power he deserves. Who will be able to cast the vote ? What is the voting actually for in our community ? Who can join the DAO ? What will be the vote-power of an individual ? All that has been covered in this file.

## Overview:

1. Who Can Join the DAO ?
2. Who Can Propose ?
3. How Voting Works ?
4. What Can Be Proposed ?
5. Penalties or Limits
6. Token Distribution Rules
7. Admin Role Powers
8. Tokenomics
9. Security

## 10. Development (Roadmap + Essentials of the development) + Roadmap

### 1. Who can join the DAO ?

Joining and contributing to the DAO is reserved only for those people, who:

- Will go to the PWA, will authenticate and they will be verified as the Web3 Builders Community Member and claim their governance-tokens to their wallet, so they can then participate in proposals and create their own ones.
- Will be the members of the WEB3 Builders community on discord.

### 2. Who can propose in the DAO ?

Every proposal is worth considering, but to increase the engagement of the members that are willing to change something in the server, there will be special rules and ranges for proposing. But in general the proposers are the founders and members, who:

- Possess  $\geq 5\%$  tokens of the voting power.
- Are longer than a month in the community.
- Are contributing actively to the community.

Those 3 requirements have to be fulfilled in order to be able to imply some change on the server.

### 3. How Voting Works ?

#### **Proposal Duration:**

2 hours - 14 days (depending on the set time).

The power of the vote of a certain user will depend on the amount of the tokens the user possesses, so that vote of user A  $\neq$  vote of user B.

Democracy fails and making everyone equal here only demotivates the members, because they cannot change anything, they basically do not have influence. When they know what is good for the community and they are indeed experienced in the field, they have a chance to have a bigger impact on the final decision.

### **Given:**

We have 20 users and one of the users is our Bob, who is an experienced blockchain developer and is contributing highly on a daily basis to the community. He owns about 12% of the entire-voting power, so essentially his vote can be the deciding one, whether something is going to be approved or not. It might sound like plutocracy, but in fact, Bob has deserved his voting power by daily contributing to the community with valuable content so he shouldn't be punished for his contributions and engagement.

Why should a programming newbie have the same voting-power as an experienced developer ? It simply does not make sense and can lead to tensions. And so the power will be possible to be obtained by the activity of the users, contributions and sending reports of what they did.

The minimum quorum will depend on the urgency of the voting that will be proposed, let me exemplify it.

### **Not-Urgent change on the server: min 40% Quorum**

E.g.

Adding a new Discord channel for off-topic discussions.

Updating the DAO's visual theme or PWA color scheme.

Naming internal squads or working groups.

Small UX enhancements to the DAO dashboard.

### **Medium-Urgent Change on the server: 60% Quorum**

E.g.

Approving a new partnership or collaboration with another community.  
(That would promote them on our community server)

Introducing a new community role (e.g., “Mentor”, “Quality Reviewer”).

Making a budget allocation for a community event or hackathon prize pool.

Changing the rules for how proposals are submitted or voted on.

Approving token rewards for a group of contributors.

### **Urgent Change on the server: 90% Quorum**

E.g.

Freezing a compromised smart contract due to an active exploit.

Slashing tokens of a high-ranking user caught abusing their position.

Emergency removal of a moderator/admin due to proven malicious behavior.

Reversing a proposal that caused unintended consequences (e.g. accidental mint).

Enacting a short-term pause on new members due to spam or attack.

---

**\*\* The proposal in order to be passed, has to obtain 60% of the quorum.**

**Vote Periods:** Will there be multiple voting periods per year (quarterly, monthly) ?

I suppose the voting period will depend on the vote-power of the proposal-initiator, the lower the voting power is, the shorter period he can choose, so that his proposal would be even voted and considered by others. Another is the urgency and also on the proposal type (whether it needs for others to be processed a bit more time to make the right decision, for instance financial boost of the server, adding a new member etc. or it is just a quick decision proposal).

## **4. What Can Be Proposed ?**

The proposal should only pertain to matters related to growth of the community in case of activity, direction of the server, relationships between each other and building a great rapport and kind of brotherhood. However, there also can occur cases, where not everyone will be willing to grow together, and suddenly will act detrimentally on our community either by not being active, cursing others' families or any other case that is unacceptable in our community. Although I personally (Luftie) think everyone can say what they want, I do not accept any red-neck behaviour, which is only to discourage others from work. However not every case is distinct for everyone, even for admins, that's why we will put to the voting cases like:

**Moderation-related (ban, warn, reward)**

**Feature requests (in the community)**




**New member acceptance**

**Allowing new form of activity on our chats**

**Teaming-proposals (As we do consider some team-work in the future)**

Will there be different proposal categories or departments for governance (e.g., financial, user management, product development) ?

### **1. Category / Department (functional area)**

-  Financial (e.g., budget allocation, payments, grants)
-  User Management / Community (e.g., bans, role assignment, permissions)
-  Server Development (e.g. Changing the arrangement of the servers, new form of engagement added, new bot that will notify about something)

- 📢 Communication / Branding (e.g., public announcements, design updates)
- 👥 Teaming up for the coding-challenge between groups to practice working together

## 2. Urgency Tier

- 🟢 **Not-Urgent (minor improvements)**
- 🟡 **Medium-Urgent (important, not critical)**
- 🔴 **Urgent (time-sensitive, high-impact)**

How will the DAO decide if a proposal falls under an accepted tier ?

**Community Review:** DAO members review the proposal and the selected urgency tier.

**Flagging System:** If members feel the selected tier is inappropriate, they can flag the proposal for review.

- If the community agrees that the tier is correct, the proposal moves forward.
- If the tier is found to be incorrect, the proposal is either reclassified by the proposer after this mark, or if not reclassified then the proposal will be rejected.

### **Penalty for Misuse:**

- If the proposer is found to have misused the tier selection (e.g., intentionally misclassifying an urgency), a penalty of 10% slash of the coins he owns or temporary (4-7 days) ban may be applied.

## 5. Penalties or Limits

Max number of proposals per user: x per week?

The amount of the proposals per week will depend on the users voting-power:

X = 0 times - If user's vote is less than 5% of the entire voting-power

X = 2 times - If the user contribution to the community is 5%, but less than 10%

X = 7 times - If user's impact is 10% and higher on the community

**Invalid spam proposals: 5% Slash tokens? Temporary ban?**

For the invalid spam proposals, the user should be temporarily banned and deprived of a considerable amount of tokens like 5%-15%.

**Proposal abuse: If someone is proposing malicious or irrelevant things repeatedly, will they be penalized ?**

First, the administration of the server is responsible to talk to the guy and tell him that something is not right, that will be the warning. If he repeats it again, he will get a 70% slash of their tokens and as said before temporarily banned, if that doesn't work, he will be deprived of all his tokens and kicked out of the community.

**Proposal spam threshold: How will you define a "spam" proposal, and what are the penalties for spamming ?**

Any unsolicited, excessive, disruptive, or irrelevant content that negatively affects the flow, quality, or safety of community interaction.

 **Spam Examples:**



- 🚨 **Message Flooding:** Sending multiple messages rapidly to spam channels or DMs (e.g., 10+ messages in under a minute with no meaningful content on the inappropriate channel).
- 🔗 **Unsolicited Links:** Promoting external links (websites, Discords, NFTs, projects) without permission.
- 📢 **Mass Tagging:** @everyone or @here without approval, or tagging users repeatedly.
- 💰 **Scam Promotions:** Fake airdrops, giveaways, or phishing links.
- 🤖 **Bot Activity:** Unauthorized bots that send messages, react to messages automatically, or mass-DM users.
- 💬 **Off-topic in Core Channels:** Repeatedly posting unrelated content in designated serious channels (e.g., #proposals or #product-feedback).

Workflow for those cases ? If the situation is intense and after the moderator or even the members request for terminating the action of spamming, the user will be muted and there will be an arranged the voting on what to do, after we will have the conversation with the spammer.

- First warning (auto-deleted + notice),
- Second offense = muted,
- Third = muted + proposal for ban (maybe urgent).

*\*\* In highly intense situations, the founders/moderation/admins reserve the right to act immediately. Perhaps that it's some attack of the bots or something similar.*

## 📌 6. Token Distribution Rules

As I've said, inequality is the biggest factor of development and progress. That's why the distribution rule will do everything to provide this inequality in life. The initial token distribution meaning the tokens you will get primarily will depend on your:

- Seniority in programming (how many years have you been coding already ? So that it was quite consistent)
- Do you already have job software-developer experience or not ?
- Technologies known for (Frontend, Backend, Blockchain Development, CI/CD, CLI, Devops) and your advancement in it.
- Interest in web3 ? Since how long into web3 in case of real interest, like really diving deep into it, exploring this world ?
- Knowledge Verification Test (About 10-12 Crypto + Programming questions) Result
- What rule does the user have in the discord server ?

On what rules will the tokens be assigned to the user ?

Based on the user provided information (We suppose everyone will be honest with each other), if the user will not be honest and the lie will come out, then there will be consequences as it comes out. The initial token distribution is for as it says distribute tokens.

However moving forward to the explanation of the formula, responsible for rewarding the user with initial tokens

Initial tokens =  $IRTB + [(PSR * IRTB) + (JEXS * IRTB) + (TKL * IRTB) + (WI * IRTB) + (KVTR * IRTB) + (DSR * IRTB)] + (RN * IRTB)$ ;

IRTB- Initial Received Token Base, which in our case  $IRTB = 1000$

PSR - Seniority Level in programming

< 1 year - 0.015

1-2 years - 0.04

3-5 years - 0.075

5+ years - 0.15

## JEXS - Job Experience Seniority (as Software Developer)

> 1 year - 0.03  
1-2 years - 0.05  
3-5 years - 0.065  
5+ years - 0.09

## TKL - Technologies Knowledge Level

For each technology applies the same rule:

Not Selected - 0.0  
Low Knowledge - 0.0005  
Higher-low Knowledge - 0.00075  
Medium Knowledge - 0.002  
Good Knowledge - 0.005  
Expert - 0.01

The amount of technologies will be gathered then together and this amount of tokens will be applied to the user's Initial Tokens.

Suppose:

User who knows 4 given technologies good level, 2 on higher-low and one on the expert level.

$4 * 0.05 + 2 * 0.00075 + 0.01 = 0.2 + 0.015 + 0.1 = 0.45$ , so the user receives an additional 45 Tokens for the knowledge of the technologies.

\*\* Because  $0.45 * 100 = 45$ .

## WI - Web3 Interest

> 1 year - 0.00075  
1-2 years - 0.001  
3-5 years - 0.005  
5+ years - 0.015

## KVTR - Knowledge Verification Test Rate

0 - 40%: 0.0

41 - 59% - 0.00025  
60 - 75% - 0.0005  
75 - 85% - 0.001  
86 - 90% - 0.005  
91 - 95% - 0.0065  
96 - 99% - 0.0075  
100% - 0.01

DSR - Discord Server Role

Admin / Moderator / Founder - 0.35

(Other rules with rates not defined yet)

Others - 0.05

RN - Random number

This will basically be a value between 0.0 and 0.25, to give the user a chance to gain some additional tokens, and for not everyone having a fully deterministic amount of tokens

### **\*\* IMPORTANT NOTE !**

**The algorithm is for now set primitively and is done for the purpose of the Phase 1 version and is certain it's gonna be changed ! It has to be discussed with members and external advisors.**

---

## **7. Admin Role Powers**

**Admins typically have the highest level of control, so their permissions should be well-defined:**

## **Approve/reject new members?**

Yes they can accept new members, but only after presenting the person to the voters in DAO, and only afterwards they either click accept or reject. They are also in charge of following the rules and act appropriately to what the rules are saying.

## **Modify proposal rules ?**

Yes, BUT if and only if the process will seem to be flawed after a certain period of time and there will be the agreement of the users to change the rules, which should then be voted on.

## **Mint tokens ?**

No, moderation by themselves cannot mint additional tokens for their own advantage. If the max supply is hit, there are no more tokens minted. The increase of the tokens is defined in the **Tokenomics section**.

## **Freeze contracts?**

In case of malicious action that occurred to the contract, the DAO will be frozen and the smart-contract will be put to be tested and checked for vulnerabilities and then audited. By the time DAO will be paused, there will unfortunately be rules of democracy temporarily introduced, until the contracts are ready to be put back to work.

*\*\* Freezing Contracts - Temporarily pausing or disabling the functionality of a smart contract to prevent further damage or exploitation when suspicious or malicious activity is detected.*

## **Adjust token supply?**

No, token supply is fixed and it is fucking untouchable.

## **Notes:**

- Founders will be obliged to act rightfully with the rules written down, they cannot take any arbitrary actions.
- 

## 8. Tokenomics

**Token Name:** (e.g. DevDAO Token or BUILD)

**Symbol:** BUILD

**Total Supply:** 19,000,000 BUILD

**Type:** ERC-20 with DAO governance rights

**Use Cases:**

- Vote on proposals (server rules, treasury use, direction)
- Earned through activity, not bought
- Used in periodic governance rounds, bonus distributions, and reputation weighting

**Earnings Model:** How tokens will be earned (e.g., rewards based on activity, governance participation) ?

Besides the tokens that will be acquired by entering the DAO

**Usage:** How tokens will be used (voting, staking, gaining access to specific features) ?

The tokens will be used to make decisions regarding any matter related to direction, members, visual change or some feature of the community, to make it better and better. The tokens will be used for the governance for now as it's a community-governance token, so it'll be used for gaining more vote power in next votings. Users also will gain the features like being able to set a proposal, when fulfilling conditions of the DAO.

## **Supply and Minting Logic: Token caps, minting strategies, etc.**

### **Fixed supply?**

✓  $1.9 * 10^7$  Tokens as the max supply

**How will the tokens be distributed on a monthly basis ?**

**$\text{MATs} = \text{DRS} * 7 + \text{SP} * 20 + (\text{DMAA} * \text{CRA}) * 5 + (\text{VCP}) * 6.5;$**

**MAT - Month-Activity Token**

**DRS - Daily Reports Sent**

**SP - Solved problems, issues in coding**

**VCP - Voice Chat Participation**

**DMAA - Daily Messages Average Amount**

**CRA - Content Rate Average (low, medium, high)-quality content provided on the server.**

**E.g.**

**Talking about weather / shilling coins or generally no-productive information - low quality**

**Recommending videos / Sharing views on the market and future of crypto / Discussions about technology - medium quality**

**Helping others / documenting progress / sharing newly learned technology / knowledge-full posts - high-quality**

**Token faucet or admin-based minting for initial test distribution ?**

It's still to be defined, I have actually considered 2 solutions.

1. An initial airdrop

This would be a contract that would have set a specific amount of tokens for a certain address.

2. Making the step-verification / validation and mint the tokens to the address based on the provided information.

**Inflation control ?**

The protocol foresees both inflationary and deflationary movements.

Inflationary movements:

- Rewarding users with the tokens (Inflationary de facto)
- Punishing the user for malicious actions (Deflationary)
- Burning the tokens (Deflationary)

**Will there be staking rewards or liquidity mining ?**

For now it's not considered. The tokens are just made for pure voting, governance and community engagement purposes.



## **Are you planning on any incentive programs (e.g., rewards for long-term holders) ?**

There is a plan to add such milestones for the users, that would reward users for instance:

- 50 times help with issues, which has been confirmed with solved problems + 250 Tokens to the wallet more.
- Actively participated in voice chats 10 times in a row + 50 Tokens.
- Has done a major work in the contest-project delivery. + 100 Tokens

And many more, but it has yet to be established.

## **Will tokens have utility beyond governance (e.g., discounts, access to features, exclusive rights)?**

For now only governance utility. What will the future bring, you never know. It's not considered yet, I need some feedback.

## **Are you planning on introducing staking mechanisms or liquidity incentives to boost token value?**

This would be interesting, but for current purposes, it's only for governance.

---

## **9. Security**

- Smart contract audits: Are you planning for both manual and automated audits?

I consider giving my smart-contract to the auditor but only as I will decide to deploy my contract on the mainnet.

---

## 10. Development / Deployment Process + Roadmap

- **What testnet will the smart-contract be deployed on ?**

First it will be deployed on Sepolia Testnet or any other testnet, where the deployment will be cheap enough. I do not plan for now deployments on mainnet tbh. I'm too poor, so don't play **Jew** and give me **money**, lmao. Or you know what no, no. Fuck your fiat, give me **crypto immediately !** I accept everything besides shitcoins and worldcoin. Mostly welcome **Bitcoin**, **Ether**, **Monero**, **Solana** and **Chainlink**.

- **Security: Make sure your contracts are thoroughly tested and audited.**

Of course I will test the DAO thoroughly, but the audit case is still open. Personally, I don't see much need to do it at this stage.



## ROADMAP + TIMELINE



### Tech Stack:

**Frontend:** Next.js + TypeScript

**Backend (Discord bot, cron jobs, recurring logic):** Node.js

**Smart Contracts:** Foundry (Solidity)

**Others:** Docker, Redis, GitHub Actions, GraphQL






---

### Phase 1: Prototype Development (April 21 – May 22)

---






## **PRE-WEEK — April 14–20: Laying the Foundation**

### Backend & Data Structure Refresh

-  Deep dive into DAOs, smart contracts & tokenomics
-  Explore DAO giants: Compound, NounsDAO, OpenZeppelin
-  Map out system architecture and workflow





---

## **WEEK 1 — April 21–27: Project Setup & Smart Contract Base**

-  Convert project to Monorepo (frontend, backend, bot, contracts)
-  Automate checks with CI/CD (linting, type checks, contract testing)
-  Start coding the governance token & DAO logic
-  Frontend scaffolding: layout, wallet connection
-  Backend setup: API structure, database, Redis caching






---

## **WEEK 2 — April 28–May 4: Voting Logic & Discord Bot MVP**

-  Finish core voting system for DAO proposals
-  Build Discord bot to track voice chat activity
-  Create REST + GraphQL APIs for proposal & voting data
-  Frontend: proposal list, voting interactions, profile base



---



## **WEEK 3 — May 5–11: Rewards System & Cron Jobs**

-  Setup monthly automated reward triggers
-  Reward contributors based on Discord activity
-  Backend logic to distribute tokens to active members
-  Frontend: claim reward button, vote status, comments
-  CI/CD: auto-deploy contracts, run tests on every PR





---

## **WEEK 4 — May 12–18: Profiles, Token Burn & Notifications**





-  Profile page: token stats, activity logs, burn option
-  Integrate Service Workers & Web Push for alerts

-  Containerize backend & bot with Docker
  -  Keep the bot alive with PM2 (restart on crash)
- 





## ✅ **WEEK 5 — May 19–25: QA, Docs & Governance**

-  Bug-fix session: reward logs, double voting, offline issues
  -  Add DAO terms and rules page
  -  Clean up codebase, write internal docs & wiki
  -  Finalize deployment pipelines for frontend & backend
- 





## ✅ **WEEK 6 — May 26–June 1: Ready for Production**

-  Refactor GraphQL & backend logic
  -  Write more tests
  -  Finalize Docker builds & optionally orchestrate with Docker Compose
  -  Include README & architecture diagram
- 

## ✅ **WEEK 7+8 — June 2–15: Polishing & Stretch Goals**

-  UI polishing, remaining bug fixes
  -  Activity leaderboard
  -  Admin dashboard
  -  Proposal editing window (time-limited)
- 

## **FINAL WEEK — June 16–22: DAO LAUNCH**

-  Release MVP to the public
  -  Open to testers & early contributors
  -  Share with the Web3 community
  -  Celebrate the launch together!
-

⚠️ **Phase 1 Timeline Flex Buffer:** If the complexity demands more time, deployment might be extended up to **June 22nd**, but ideally, May 22 is the target.

---

## 🔧 **Phase 2 (June 23 – Oct): Refinement + User Feedback Loop**

🎯 Goal: Educate users, collect usage data, fix bugs, and refine flow 🔄  
Optimize contracts, UI, bot commands, backend logic 📈 Scale up GraphQL queries and backend caching (Redis tuning) 📖 Add analytics and documentation for contributors 🎨 UX polish for smoother onboarding and reward claiming

---

## 🧠 **Phase 3 (Oct 20 – Dec 31): Finalization + AI Upgrade (POTENTIAL)**

🧬 Focus on AI-enhanced features, final contract refinement, and advanced bot logic

🚀 Add AI scoring model to evaluate Discord participation value  
🔍 Tweak voting eligibility & auto-grant systems  
📱 Enhance PWA notifications & UI 📢 Extend bot commands for advanced governance (e.g., proposals, vote summaries)

👤 All development will be streamed on **Luftie's DLive** – follow to stay updated!

---

## 🚧 **ROADMAP SUMMARY**

You're watching a wild developer build a DAO in real-time. Join the journey, drop by on **DLive**, and maybe if you're a bit more wealthier than an average beggar, that you meet up as you go to the shop, please don't be a **Jew** and sprinkle some support on Luftie 💰. Let's make the DAO a reality 🔥

**XOXO, Luftie** 🤗