

# Alchemy Road to Web3 HACKATHON (week 3)

## GOVERNANCE

May 21, 2022

# Hackathon Objective:

The Project should implement:

- A governance token (ERC20, ERC721, etc.)
- A voting system using your governance token
- A UI that allows users to vote

# Details:

## DAO Model:

On-Chain Governance

## Based on:

- OpenZeppelin Governance  
(<https://docs.openzeppelin.com/contracts/4.x/api/governance>)
- Chainlink Hackathon 2022 Sessions  
(<https://chain.link/hackathon>)

## Frameworks/Tools used

### BackEnd:

- HardHat / Solidity (TS for deployment and interaction scripts)
- OpenZeppelin Wizard and libraries
- SCs deployed on Polygon Mumbai

### FrontEnd:

- Figma for prototyping
- ReactJS + Tailwindcss

# Smart Contracts deployed:

## GovernanceToken

Vote Power for  
each member.  
RTW3GT  
Supply: 10.000  
Allows Delegation

## Governor

Logic and  
Mechanics  
  
(Propose, Vote,  
Execute  
proposals, etc.)

## TimeLock

Queue to enforce  
Time Gap.  
Add a delay for  
governance  
decisions to be  
executed

## Box

SC with Store  
method to use as  
Custom Action  
for a proposal.

# Voting Time-related concepts:

## votingDelay

Delay, in number of block, between the proposal is created and the vote starts. This is used to enforce a delay after a proposal is published to leave time for users to buy voting power, of delegate it, before the voting of a proposal starts.

## votingPeriod

Delay, in number of blocks, (starts after votingDelay) between the vote start and vote ends.  
This is the period of time when the voting is Open.

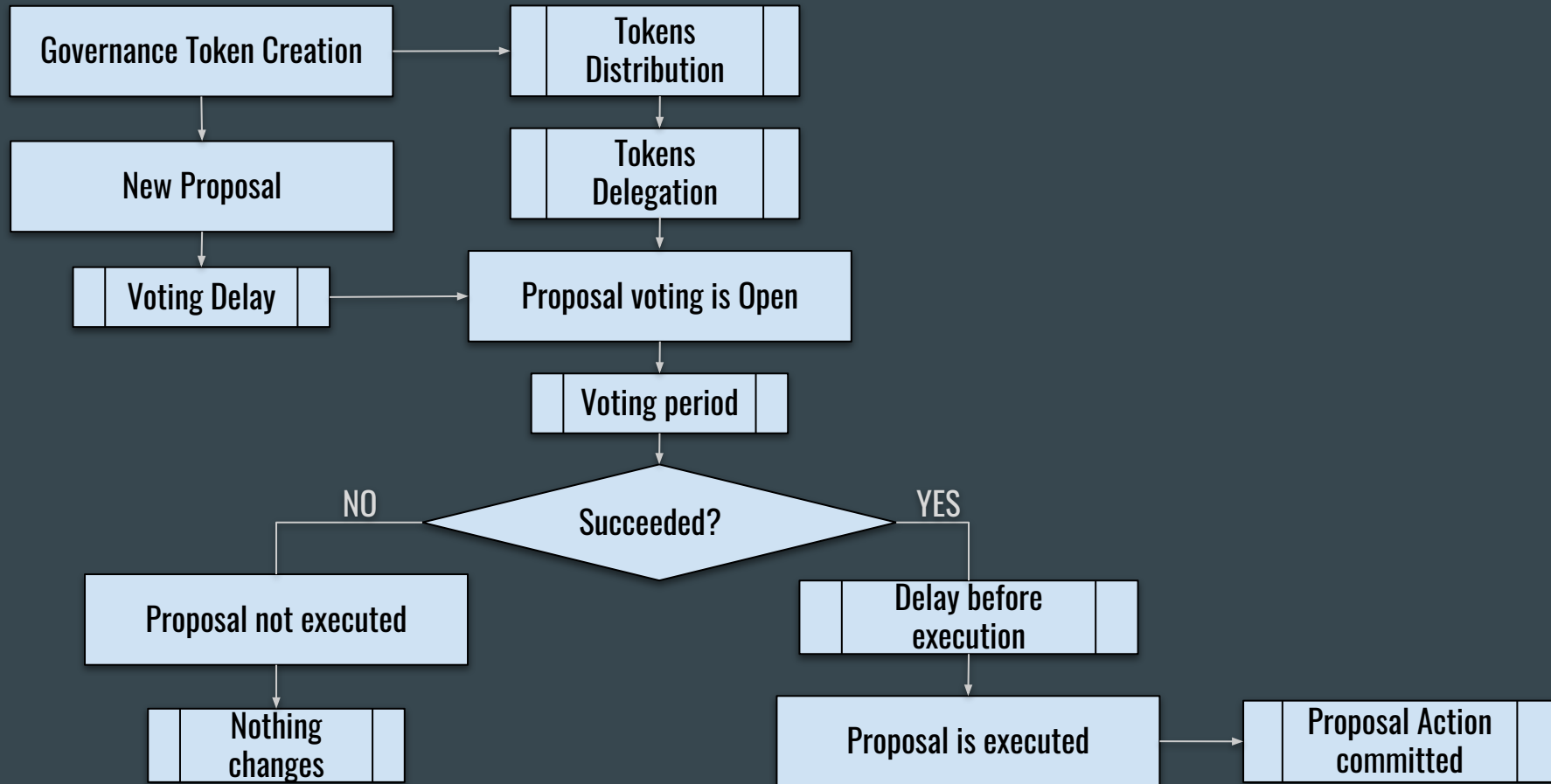
## minDelay

Delay period before a passed proposal is executed. The proposal is “queued”. This delay will give the users the opportunity to exit DAO if they disagree with the proposal.

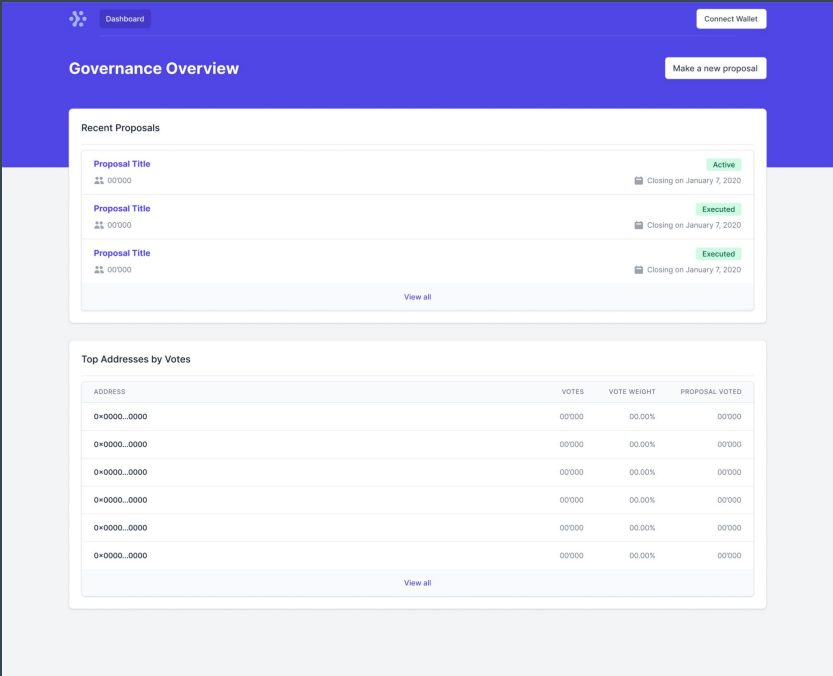
# Voting details:

- ERC20Votes (extension of ERC20) used as Governance Token to allow **Delegation** method.
- ERC20Votes also will keep track of historical balances so that voting power is retrieved from **past snapshots** (block when the proposal was launched) rather than current balance (to prevents double voting and flash-loans).
- To cast a vote the address must have Vote Power (RTW3GT Tokens)
- Vote power can be delegated by calling the delegate method
- Only one vote per address is allowed
- Initially each DAO member will have the same amount of Tokens (same Vote Power)
- castVoteWithReason method used to allow Voters to add a comment/reason to their votes
- Quorum = minimum number of cast voted required for a proposal to be successful.
- Used support=bravo (options 0 = Against, 1 = For, 2 = Abstain)
- Used quorum=bravo (only For votes are counted towards quorum)
- Used QUORUM\_PERCENTAGE = 5 (need 5% of voters to pass)

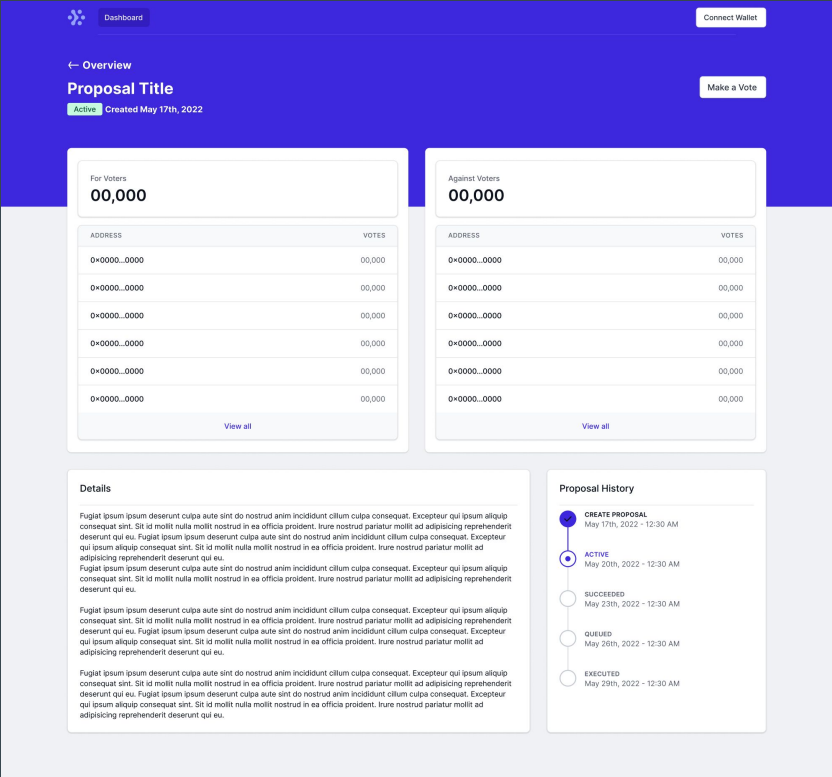
# Voting Flow:



# Frontend (2 interfaces):



Governance Overview



Proposal Overview



# Frontend details (Governance Overview):

Connect Wallet

Click on to connect to your Metamask Wallet (Select the Polygon Testnet).

Make a new proposal

Click on to open the modals for making a new Proposal (not working yet).

Recent Proposals		
Proposal 1	00'000	Active Closing on May 23, 2022
Proposal 2	00'000	Active Closing on May 23, 2022
Proposal 3	00'000	Active Closing on May 23, 2022
<a href="#">View all</a>		

List the 3 Proposals we have made, tried to list the most recent but it only display it on the console and not on the UI.

Click on one of the proposal to open the Proposal Overview from this particular Proposal.

# Frontend details (Proposal Overview):

Make a Vote

Click on to open the modals for making a vote for a particular Proposal. Calling the method “castVote” from Governor Smart-Contract.

For Voters	
00,000	
ADDRESS	VOTES
0x0000...0000	00'000
0x0000...0000	00'000
0x0000...0000	00'000
0x0000...0000	00'000
0x0000...0000	00'000
0x0000...0000	00'000
0x0000...0000	00'000
View all	

Against Voters	
00,000	
ADDRESS	VOTES
0x0000...0000	00'000
0x0000...0000	00'000
0x0000...0000	00'000
0x0000...0000	00'000
0x0000...0000	00'000
0x0000...0000	00'000
0x0000...0000	00'000
View all	

List the voters for and against, tried to list the best voters but it only display it on the console and not on the UI.

# Proposed Improvements

Governance Token  
mint

- Create a mechanism to deliver/mint the GovTokens to each DAO participants.

Deliverable 2

- List dynamically the for/against voters for a particular proposals.

Deliverable 3

- List dynamically the different proposals and the top voters for the entire DAO.

Deliverable 4

- Create a dynamic “roadmap” to track the state of a Proposal (Creation, Queued, Executed).

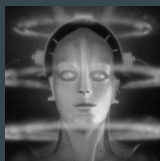
# Buglogic Team

Killiane



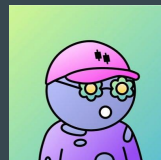
 [@KillianeMenand](https://twitter.com/@KillianeMenand)

GdG



 [@gabrieldegiuli](https://twitter.com/@gabrieldegiuli)

Toto



 [@togido](https://twitter.com/@togido)

Jesse



 [@Jesserc](https://twitter.com/@Jesserc)