Cat Herder

End of Period Report

https://github.com/yearn/budget/issues/213

Plan

Ongoing Tasks

- Manage the funding request repository on GitHub.
- Oversee the handling of incoming requests from receipt to decision.
- Administer the internal yBudget processes, including team coordination, voting procedures, status updates, timely communications, and team incentives.

Request Period Specific Tasks

- [ ] Refine the yBudget process as necessary.
- [ ] Investigate workflow automation options (e.g., Telegram bots) once a stable process is established.
- [ ] Lead the development of a veYFI rewards program for yTeams.
- [ ] If time permits, improve the "blue dot" flow.

Topics

- yBudget2: process & epoch overview
- robokitty: progress report
- yTeam rewards program: progress report
- proposed next steps

yBudget2

Process

Started with taking the original high level idea and putting it into practice

Randomness from Google Sheets

Easy to get up and running

Not reproducible - how do we know Cat is not cheating?

Refreshes on load - data is lost

Randomness from ETH

Seed draws with source of randomness from beacon chain's RANDAO

Reproducible randomness, not known in advance

```python
def generate_random_score_from_seed(randomness: str, index: int) -> float:
        # Step 1: Combine the randomness and index. This ensures unique input for each ticket
        combined_seed = f"{randomness}_{index}"
        # Step 2: Create a SHA-256 hash object
        hasher = hashlib.sha256()
        # Step 3: Update the hash object with the combined seed
        hasher.update(combined_seed.encode('utf-8'))
        # Step 4: Get the final hash value
        result = hasher.digest()
        # Step 5: Convert the first 8 bytes of the hash to an unsigned 64-bit integer
        hash_num = struct.unpack('>Q', result[:8])[0]
        # Step 6: Calculate the maximum possible value for a 64-bit unsigned integer
        max_num = 2**64 - 1  # This is equivalent to 18446744073709551615
        # Step 7: Generate a float between 0 and 1 by dividing hash_num by max_num
        score = hash_num / max_num
        return score
```

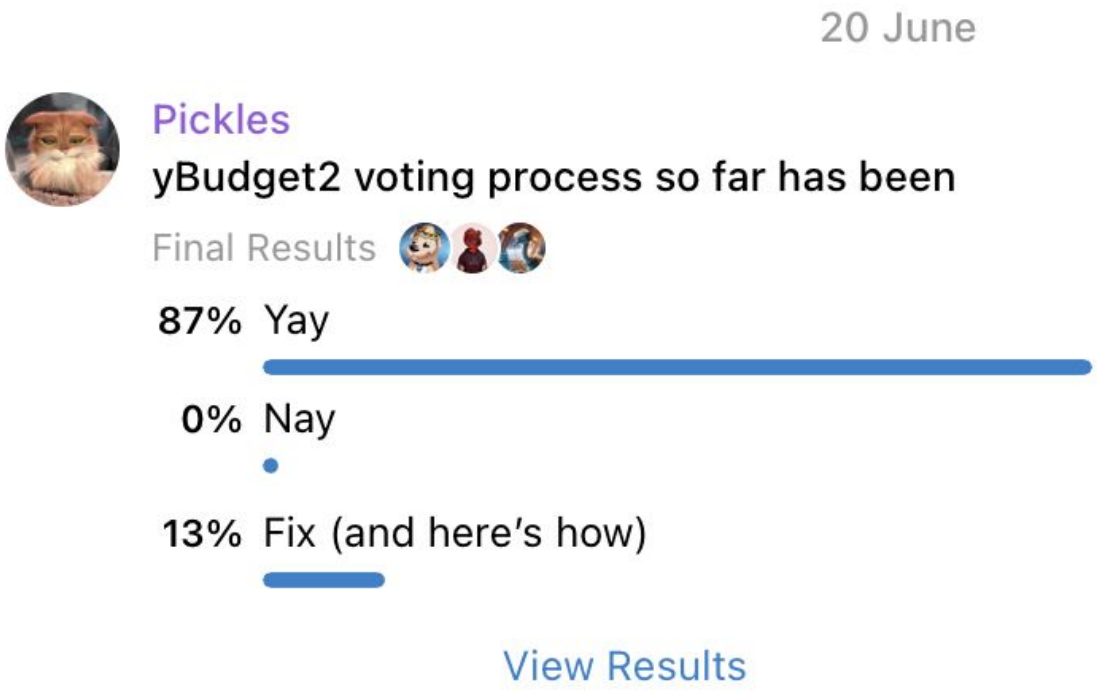
Started iterating using a Google Sheet

Easy proof of concept

Reproducible and auditable

Figure out what is needed

Feedback on process was positive

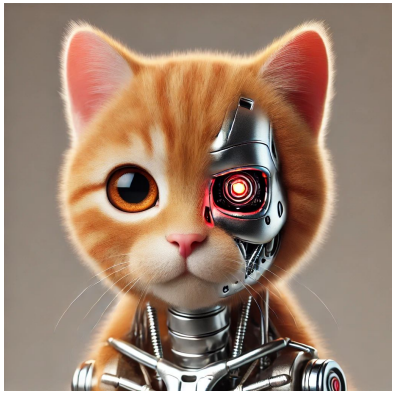


But not very scalable

Many many tabs

Bug and error prone

Lots of manual operations

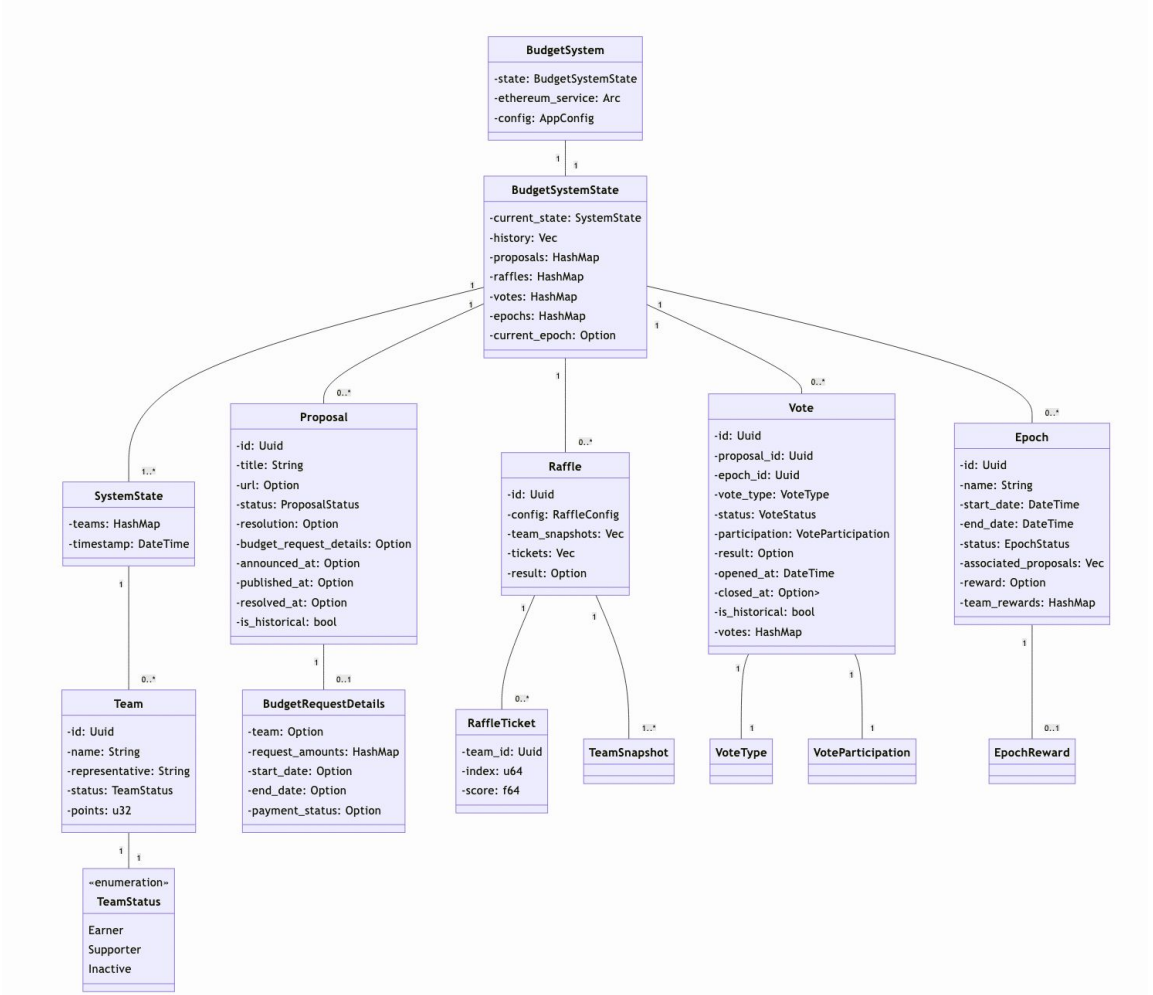


robokitty

robokitty alpha

- Bionic cat herder written in Rust
- Relies on BudgetSystem, with state managed as a JSON file
- Interface via Telegram and JSON script commands
- Core Features
  - Epoch management
  - Team and proposal handling
  - Raffle and voting mechanisms
  - Reward distribution

Class diagram

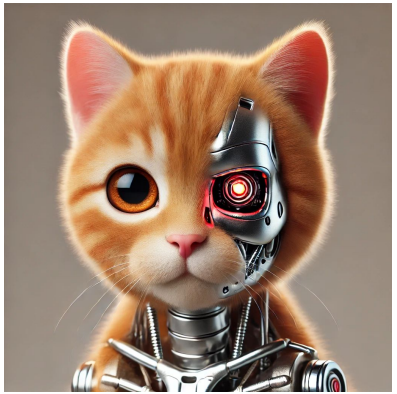


demo: proposal reports

demo: telegram

what’s next for robokitty?

- Build out telegram commands, make sure usability is good, add basic access controls
  - Team mgmt
  - Create raffles
  - Create votes
  - Finalize and close proposals automatically
- New features:
  - Payments: status & urls, integrate into yRoboTreasury scripts
- Modularize & refactor the codebase
- Consider transitioning to a database
- Build out a test suite to ensure data integrity
- Open source



yTeam Rewards

Motivation: Why more rewards?

- Distribute veYFI voting power to teams to prepare for eventual on-chain gov
- Combine a backward looking component to reward historical participation, with a forward looking component to incentivize behavior and increase cross-team collaboration along common objectives
- Ensure incentives are positive, not negative for Yearn

Alternatives considered & open discussion

- Allow the option to lock YFI with liquid lockers, or force naked veYFI only?
  - Allowing is in line with the last individual vests
  - Part of the motivation behind creating 1UP, to ensure a neutral locking place
  - Enables teams to earn yield from their veYFI allocation
  - Allows active teams to control a greater share of veYFI vote power

- We’ve not made a decision on this, as Epoch 1 ends today for yBudget 2, we will need to figure this out anyway

Alternatives considered & open discussion

- Do we even need a veYFI rewards program right now?
  - Is there a program that teams can get behind as ‘fair’?
  - Perhaps better focus on continuing with yBudget2
  - Pick up at a later time if needed

proposed next steps

proposed next steps

1. One more developer mode request period for cat herder to make robokitty capable to run the show (and then maintenance mode)
2. Delegate to yBudget2 to figure out whether to allow liquid lockers for the veYFI, it will be in their interest to figure out as they need to get their rewards tokens
3. IF we want to proceed with team rewards, appoint a trusted 5-7 person council with wide representation from various yTeams to have decision making power *(I am happy to do required work, and be off the actual council)*

thank you