Cozy Reef Season One

ImpGuard Minesweeper Zoogie*

Version 1.0
25 November 2021

Contents

* This author has affiliation with Chainlink Labs and has financial interest in the success of Chainlink technology.

1 Introduction

Cozy Reef Labs created the Cozy Reef NFT project to explore how blockchain technologies can be used to create unique community-based gaming experiences. We leverage a variety of technologies in the blockchain ecosystem to provide players with an unprecedented level of visibility into the inner workings of each game and foster a high-stakes competitive playing field with a play-to-earn ethos.

2 Design Goals

Our vision for Cozy Reef extends beyond any single game or technology. We're building a platform where continued success in competition unlocks the opportunity to join the founders (the Cozy Reef Killers) and directly impact the future of the project.

Blockchain technology allows us to provide transparency, security, and fairness in a competitive environment. In order to create a sense of high stakes and progression, we purposefully limit the number of participants for our games to owners of a Cozy Reef NFT. Each game will act as a funnel—only winners continue to the next game in a given season. Games early in the season are forgiving and encourage collaboration, progressing towards more competitive games for individuals and teams. Players who clear all the games in a season receive a CRK Mask NFT, which they can sell or use to mint a Cozy Reef Killer NFT, giving them voting power for future games and the direction of the Cozy Reef project as a whole.

In order to build this system, there are several challenges to address. First, we need to ensure that our contracts are secure and free from modification, even from contract owners. Every winner can be audited and traced back to the series of transactions they made in order to win, ensuring that game outcomes are fair and tamper-proof. Second, the cost of transactions has increased significantly with market interest. Ethereum mainnet has prohibitively high gas prices that we aim to circumvent for our players and ourselves to reduce costs and allow us to build more interesting games. Third, since blockchains are public by nature we must design games that are fun and interesting even when players have access to the entire game state.

The rest of this document describes our high level strategies to solve these core challenges.

3 Contract Security

We require more than just Blockchain technology alone to verify that project owners will act in good faith. With clear game contracts and transparency throughout the course of the project,

external parties can validate that the project operators have acted in good faith so far. Our goal is to develop our contracts and games such that anyone can verify that expectations match reality. We continue to pursue games and experiences that are decentralized and require no maintenance, but recognize that we will not fully achieve this ideal in season one.

We incorporate a consistent set of best practices when creating our NFTs and game contracts to ensure that outcomes are fair and provable.

3.1 NFT Contract [1][2][3]

1. The NFT will launch with a provenance value representing the hash of all the metadata files and images that will be revealed after sale. This guarantees that once all the images are revealed, anyone can hash the image values together to verify that there were no nefarious actions taken to change the outcome.
2. The core NFT project leverages Chainlink VRF technology to generate a “provably-fair and verifiable” random index for NFT token IDs. This helps guarantee that buyers have the same opportunities to receive NFTs at varying rarities [4].
3. Our NFTs will feature a locking mechanism to completely lock further modification. Once the NFT has been revealed (either through a full mint or community vote) we will lock the contract. This provides a guarantee that an NFT is permanent within the blockchain ecosystem.

3.2 Game Contract

1. All game logic is codified in the contract itself to ensure that the game is public and immutable once launched. This transparency has an impact on game balance and is a core consideration when we design each game.
2. All game state is easily queryable from the contract. This is similar to the above requirement, but also ensures that external parties can query and validate the on-chain game state that players see in the UI.
3. The games will be played on the Polygon sidechain, and all winners and game results will be stored on the Ethereum Mainnet. Similar to our NFT contracts, these game results will be locked and immutable after the correct winners are confirmed by the project owners(which can be validated by anyone).

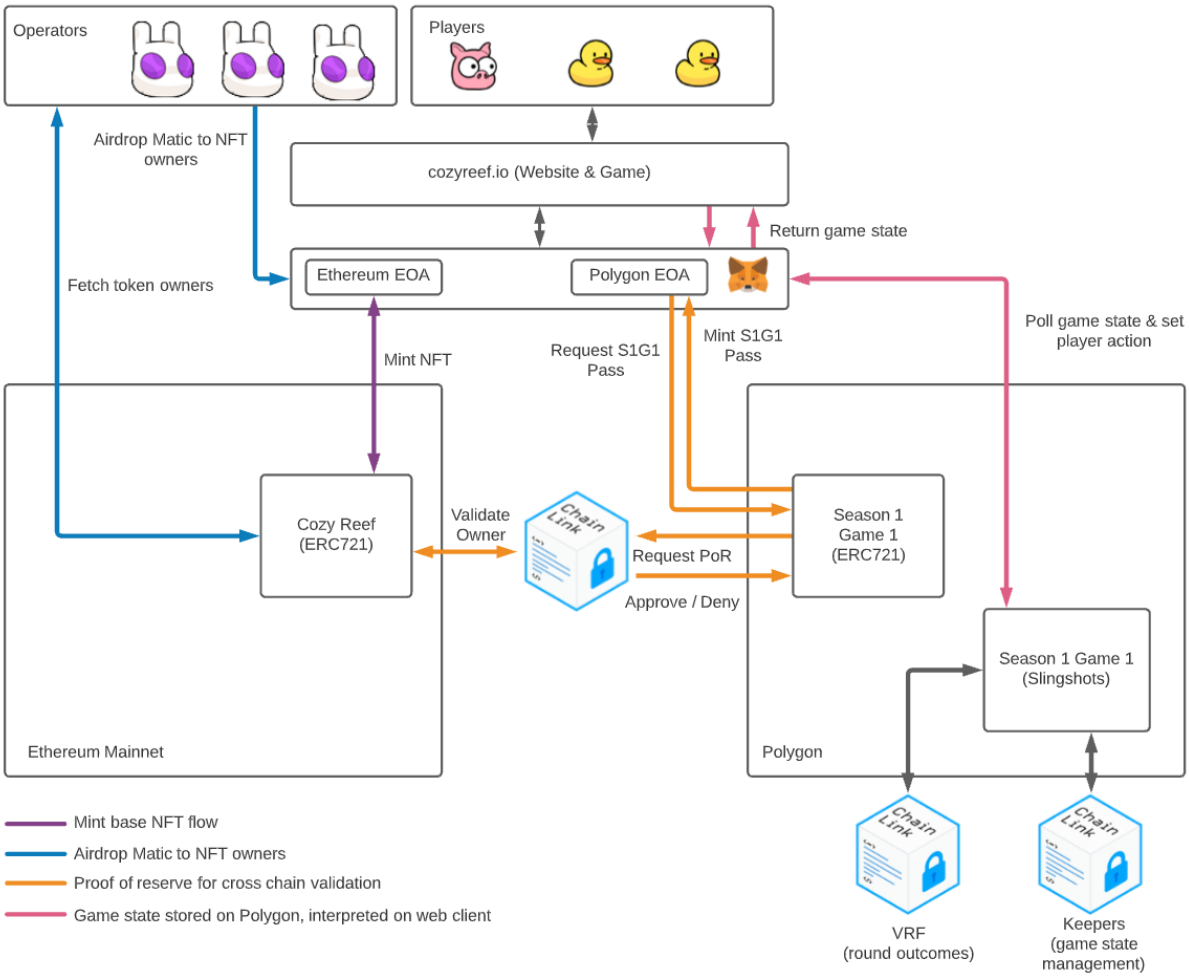

Figure 1. High level architecture of the Cozy Reef ecosystem

4 Cross Chain

The primary motivation to deploy and run our games on the Polygon side-chain is to drastically reduce costs. Figure 2 shows price estimations for the USD cost of a single transaction a player makes if the first game (Slingshots) is played on Ethereum mainnet versus Polygon side-chain.

Network	Gas	Gas Price (gwei)	Txn Cost (gwei)	USD / Token	USD / Txn	USD / Game (200 moves)
Ethereum mainnet	132000	100	13200000	4200.00	$ 55.44	**$ 11,088.00**
Polygon	112000	40	4480000	1.69	$ 0.0076	**$ 1.51**

Figure 2. Game 1 (Slingshots) transaction cost estimator (USD prices estimated as of November 2021)

Running games on Polygon reduces cost by roughly **7500x**. We believe this side-chain is more than sufficient to support the needs of **one time ephemeral game instances** in speed and

reliability, and the cost savings allows us to build games with generous amounts of transactions and compute, creating more opportunities for fun and interesting game experiences.

Operating Cozy Reef games on Polygon means players will use a new token (Matic) on a new chain. With funding generated from the initial sale of the project paired with the low cost of game operations, we believe we can remove this complexity for our players by airdropping tokens to each player to play the full game loop (see Figure 1).

5 NFT Proof of Reserves

The Cozy Reef games will operate on Polygon primarily for cost savings to both players and game operators. However, the original Cozy Reef ERC721 tokens will be deployed to Ethereum mainnet due to security, robustness, and value retention. Players can only play Cozy Reef games if they own a Cozy Reef token at the time that a game starts. Since the game instances are ephemeral and the cost of transactions on Polygon is low, we're approaching ownership validation through a proof of reserves model. Cozy Reef owners will be able to mint a game access token on Polygon through a validation workflow illustrated in Figure 3.

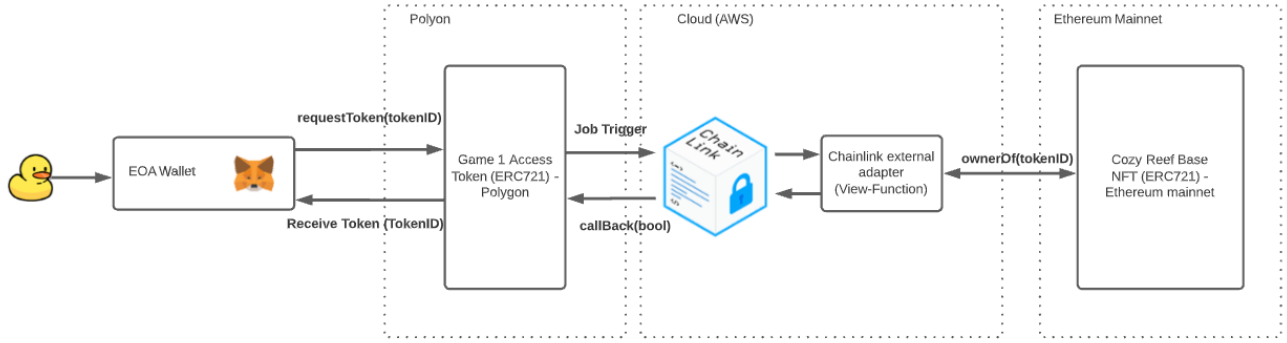

Figure 3. Workflow for proof of reserve for game pass granting on Polygon

The workflow for minting a game token is described as follows:

1. A player will call the requestToken method within the Polygon sidechain to issue an event to obtain a seasonal ticket, passing the tokenID of the Cozy Reef token that they own on Ethereum mainnet for validation.
2. If the requested tokenID has not already been claimed for the game instance, an event requesting the Chainlink job run is emitted. An off-chain oracle listening for the requestToken job event kicks off to verify token ownership on Ethereum mainnet.
3. The Chainlink oracle calls the external adapter service, which calls the ownerOf method on the Cozy Reef token contract, passing in the tokenID to retrieve the owner address [5].
4. Validation of the token occurs by verifying that the address of the tokenID owner matches the requester address. If this criteria is met, the game access token is minted and released to the requester.

We opt-ed for a simple validation model that has both technical and product tradeoff implications. To address the potential consequences:

Additional process for multi-token owners - The current validation model is only suited for single token validation. Multi token owners will have to repeat the validation process for each token they want to verify and claim on Polygon. The alternative approach is to validate and mint all tokens the requester owns. We opted for the first approach because this model only has transaction costs on Polygon, circumventing the need for any expensive transactions on Ethereum mainnet. Multi-token owners also have the option to play with only one token, allowing them to keep their other tokens' playing eligibility for prospective buyers.

Ownership discrepancies across the chain - We do not take custody of the original token in order to mint the game token, meaning a Cozy Reef owner can mint a game token on Polygon and then sell the original token. Since the Polygon game token loses all value after the conclusion of the game instance, we accept this potential discrepancy as there are no negative impacts to the outcome of the game. The benefit with this model is that we do not assume the risk or the costs associated with managing a token bridge.The trade off is that buyers may purchase a Cozy Reef token that has lost eligibility to play in the current season. We will provide tooling for prospective buyers to check token eligibility.

Single oracle node deployment - The oracle deployment for the first game of the season will be centrally managed by Cozy Reef Labs. We assessed that this as an acceptable risk after considering the following factors:

- Game one is low stakes with intended high survival rates.
- All token mints are auditable for nefarious actions.
- Financial impact of the centralized component becoming compromised is minimal. In the event that the oracle crashes we can redeploy the infrastructure and reprocess pending requests. The worst case outcome for the validator misbehaving is that we redeploy all the game contracts and supporting infrastructure.
- As Cozy Reef progresses further into the season and game results have higher financial implications, we intend to migrate the oracle network to decentralized nodes managed by third party node operators.

6 Game Design

In the pursuit of designing fun blockchain games we identified several constraints inherent to the nature of today's blockchain technology:

1. All information is public. As a result, game state can be visible to all players at all times and the core game loop cannot rely on hidden information.

2. All game actions must be done as a single user-verified transaction, meaning more real time based games are not currently feasible.
3. Front running, or more well-off individuals paying higher gas prices to get their transactions in earlier than others (commonly assisted by software), is a pervasive issue. Games that rely on timing and reaction speeds will be tough to balance.
4. Transaction costs are expensive. Updating all player states every game loop is cost prohibitive, so workarounds must be made to optimize for reducing transaction costs as much as possible.

6.1 Public Audibility

The two tools that we want to leverage when building games in this public environment are randomness via decentralized oracles—such as VRFs from providers like Chainlink—and cryptographic tricks to hide limited information on the chain.

Randomness allows us to introduce some element of variety that a rigid contract typically would not. However, this tool must be used sparingly. We want to maintain player agency and avoid games that feel like coin flips.

Separately, cryptographic tricks allow us to hide small bits of information while maintaining the integrity of the chain. An example would be requiring a user to hash an action with a self-selected nonce, and then revealing their action later by once again submitting the action and the nonce used to generate the hash.

6.2 Front Running

Working around user actions as transactions to a distributed ledger is the least challenging complication. Turn based games exist in all sorts of mediums so we have many existing games to take inspiration from.

However, the issue of front running is a significant problem. Our key design philosophy here is to make front running either impossible or not essential to success. By ensuring that speed does not guarantee better results, our games take front running out of the equation.

6.3 Cost

Dealing with high gas prices is likely the most difficult issue to engineer around. Optimizing for a low number of state changes each transaction will reduce our costs, but restricts the complexity of our game mechanics. In games, storing information about each player is quite common (for example, keeping track of player scores over time).

On the blockchain, one transaction that would need to update thousands of player scores would be very cost prohibitive. Instead, we opt to not store the player score at all, leveraging lazy evaluation patterns to compute player score on demand. In doing so, we store and update less data on the chain while maintaining consistency overall.

6.4 Fun

Above all, our core mission is to bring fun and engaging game experiences to our players. We define this to mean building games with high agency, skill, and player to player interactions.

Games that are overly reliant on randomization do not meet our criteria for fun. Limiting player interactivity will reduce transaction costs and state management, but will also diminish the fun factor. Product and technical creativity will be required to overcome these limitations to develop experiences that embrace the advantages of blockchain while still focusing on player agency and community.

6.5 Reward

Play-to-earn mechanisms are a core tenant of the Cozy Reef platform. Players rightfully expect rewards for the time they spend playing games and unlocking achievements. The right rewards will create a high-stakes competitive environment with long-lasting satisfaction of accomplishment, while the wrong rewards may create short sighted incentives and misaligned motivators that degrade the overall game experience.

7 Case Study: Slingshots

The first game Cozy Reef Labs is building takes inspiration from the children's game red light, green light. We want to capture the feeling of taking calculated risks to cover distance faster than your competitors.

From the outset, red light, green light avoids some of the problems inherent to a blockchain game. It’s transactional and has a simple set of actions the user can take: stop and go. Public information does not pose issues since players are essentially playing the game as individuals against a game controller.

Red light, green light, however, has some fundamental issues when built on the blockchain. The game is based on timing, and those who wait until the last moment to stop moving will net the most distance possible. The game also does not meet our criteria for fun. While there are elements of skill involved, the player still effectively competes against a coin flip. A real game of red light, green light typically involves a metagame of watching others and changing your behaviour based on that. Online, with a game controller swapping lights back and forth, this nuance will be difficult to capture.

Slingshots is a game inspired by red light, green light with key changes to address these fundamental issues. The rules are as follows:

- There are 5 slingshots that launch players across the map.
- Players are allowed to pick a slingshot to put themselves in.
- Every hour, a Chainlink Keeper will trigger a Chainlink VRF call to determine if each slingshot fires, misfires, or does nothing.
- If a slingshot fires, all players in that slingshot are flung forward and gain distance.
- If a slingshot misfires, all players will be stunned for a brief period of time.
- If a slingshot does nothing, the player does not gain distance nor is penalized. The probability that the slingshot ends up firing (or misfiring) increases during the next round.

Two additional rules add variability and encourage player interaction:

- The more players in a slingshot, the farther it will fire, but the higher chance it has to misfire.
- The more often a slingshot has successfully fired, the larger its distance multipliers become, encouraging players to get on slingshots that have high multipliers.

7.1 Public Audibility

Similar to playing red light, green light in person, public information adds interesting elements to the gameplay. Players can view what slings other players have selected. Players know all the probabilities and all the multipliers available, and must decide on their risk and reward profile. Players can choose the slingshot with a high multiplier but a high chance to misfire or take a conservative approach. We expose on-chain information that could impact a decision in the game UI so players who interact with the contract directly have no advantage.

We believe that the truth-based model of the blockchain enhances this game experience further. All information is publicly available and verifiable. When a player wins, we are able to view every transaction state and player decision. A traditional version of the game would require a lot of developer effort to expose this information, whereas the blockchain guarantees and requires that visibility.

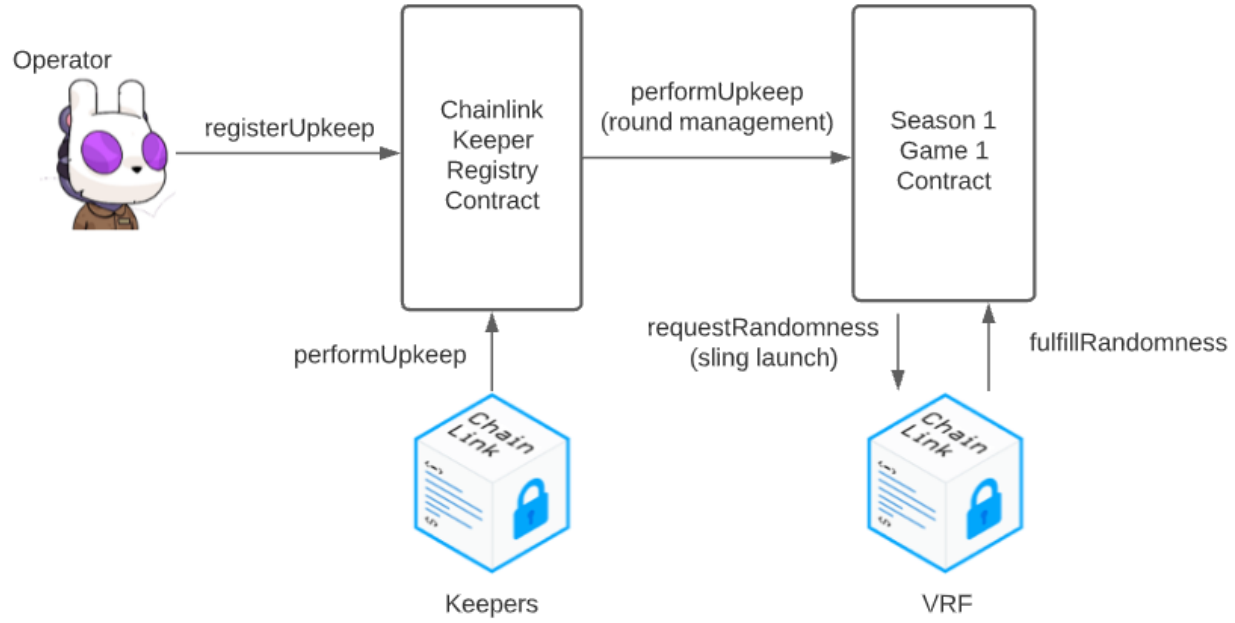

Figure 4. Round management performed by Chainlink Keepers and randomness to determine sling outcomes determined by Chainlink VRF

The usage of Chainlink oracles to generate provably random numbers and decentralized upkeep offers further audibility that the game is operating as intended without tampering. Chainlink VRF generates the random values that determine game state outcomes - whether the sling launches or snaps, and for what distances. Chainlink Keepers act as a decentralized game runner to reliably facilitate the game loop (Figure 4).

7.2 Front Running

Transactions for Slingshots are simple. Players select which slingshot they want to be on. Front running poses two issues: (1) waiting until the last moment can give players a slight advantage as they know the latest game state, and (2) if a player is able to submit their transaction after the VRF call generates a random number, they can choose the slingshot with the best outcome.

We believe that the game does not provide a significant benefit for players that wait until the last moment. Slingshots have a random chance to fire or not and there are five to choose from. Committing earlier or later in a given cycle does not have a significant impact on your ability to succeed overall compared to your ability to manage risk effectively over the course of the entire game.

In order to prevent players trying to get ahead of our game transactions, we implemented a lockout period before each launch to prevent further user actions while we generate our random numbers and resolve the game state.

7.3 Cost

For slingshots, we split out our game state into two pieces: game state and player state. The game state is a single structure, while player states are created for each individual player. To ensure no single transaction becomes prohibitively expensive, we follow two guiding practices:

1. Player actions are only allowed to read and write a player's own player state as well as the (singleton) game state.
2. Game actions are only allowed to read and write the game state.

For the game to be functional, we must be able to compute the total distance traveled by any particular player upon request. Therefore the game state, player state, and all actions are designed to ensure this computation is possible, without violating the two rules above. The player state stores a list of moves, rather than a current location. The game state stores the complete launch history for each slingshot, rather than just the most recent launch. Combining these values allows us to compute which launches a player was part of throughout the game and their accumulated distance on demand. Values such as "number of players in a slingshot" that are not critical to the distance calculation are stored in the game state and only reflect the current state to reduce storage costs.

Example: For a single slingshot that fires and impacts the state of all players in it, mutating the data stored at an existing storage location would cost roughly 5,000 gas per storage block modified. Assuming player state is pre-existing on the contract before a slingshot fires and tracking player state by storing the total distance traveled for 1000 players, the contract call would cost roughly 5,000,000 gas. Alternatively, storing the launch data (distance, backfire, etc…) each time a launch occurs and lazily computing the distance traveled by a player on demand, the contract only allocates 5 blocks of storage for 20,000 gas each, resulting in a fixed cost of 100,000 gas independent of total player count[6].

7.4 Fun

Ultimately the players of Cozy Reef will decide if the games are fun. However, there are objective criteria that we believe are foundational for a meaningful game experience.

Player agency is high - Players can choose to load into one of five slings to try to maximize their distance. Each sling has a different risk and reward profile, and players decide which sling to take.

Requires skill to win - Players must make strategic decisions to choose the right risk reward slingshot profiles in order to get to the finish line before the Cozy Reef Killer catches up to the player.

Player to player interaction - As more players enter a sling, the chances of the sling breaking and stunning the player increases. However, the distance multiplier for the sling also increases. Players must make decisions based on how other players in the game compose themselves across the five slings.

7.5 Reward

Cozy Reef games will play out over seasons. Only players who survive each game are eligible to play the next, and players who survive all the games in a season will receive a Cozy Reef Killer mask, which they can use with their base NFT to mint a Cozy Reef Killer NFT. Cozy Reef Killers will have voting rights for the future direction of the project amongst other to-be-decided benefits.

Players who survive each seasonal game will also earn “Cozy Coins” (ERC20 tokens) relative to their final score in addition to eligibility to continue competing in the season. Tokenomics and utility design are in progress and will be reviewed in a future document, but will likely include the ability to redeem the coins for competitive game passes and purchasing additional NFTs to engage with in Cozy Reef. This will balance out the core reward mechanism (Cozy Reef Killers mask) that only a select few players each season will receive with progressive rewards that a larger percentage of the player base earn for their achievements.

8 Future Work

Future work primarily includes developing new games that continue to leverage the capabilities unique to blockchain to create interesting game experiences. Current aspirations include global Rock Paper Scissors and faction-based Risk amongst others.

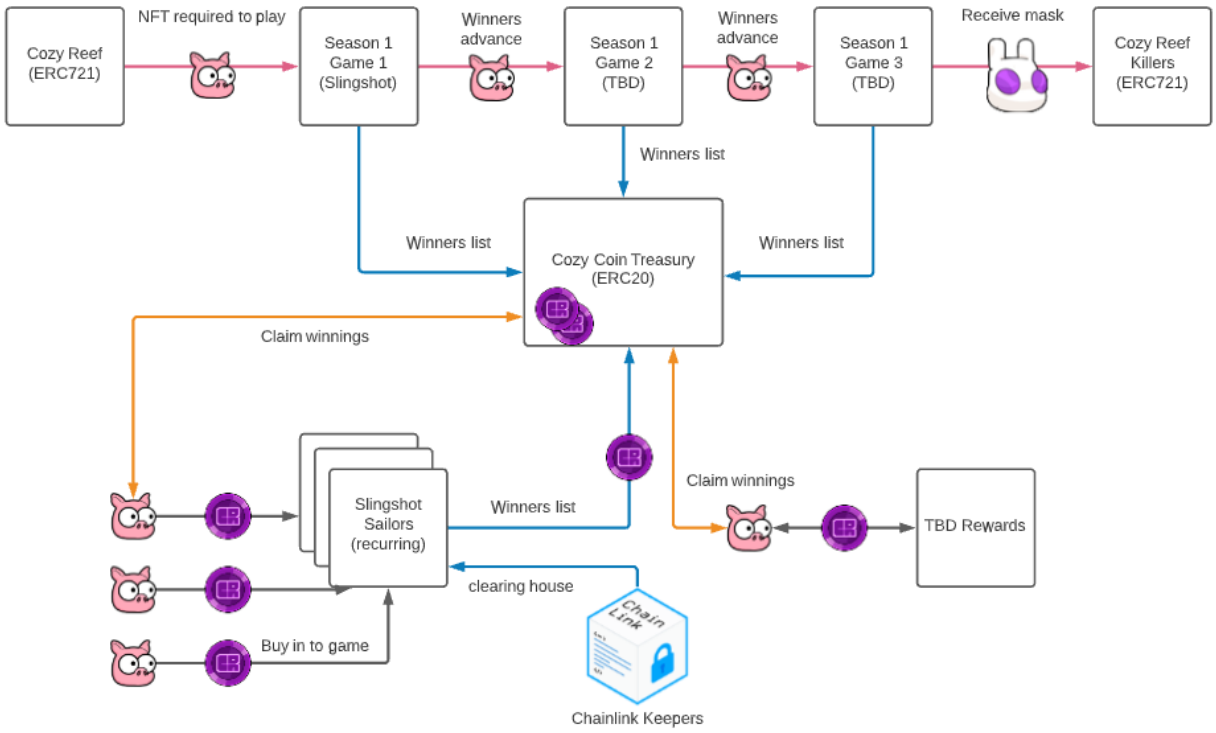

Figure 5. Example workflow on future incorporation of Cozy Coins and tokenomics

We will also build a platform that creates engaging reward incentives and tokenomics through the ERC20 Cozy Coin. In the example shown in Figure 5, players earn tokens by surviving the seasonal games. They are then able to claim their winnings from the treasury, and spend tokens to replay games and purchase other rewards. The team currently lacks expertise on tokenomics and will rely on experienced advisors to solidify the design.

Initial playtests of the first Cozy Reef game showed technical limitations from Metamask that caused lag and degraded the game experience. Future work will include exploration on optimizing an EVM wallet for our gaming requirements.

9 Conclusion

We aim to leverage blockchain technology and smart contracts to develop uniquely transparent and competitive games with satisfying rewards and community interaction on the Cozy Reef platform. We're excited to deploy the first season of games and welcome a select few victorious players to the Cozy Reef Killers, where they'll influence the future direction of the project. While we're currently focusing on the development of the NFT project and the first season of games, we have identified several opportunities to evolve Cozy Reef into a long-term gaming platform alongside the community.

References

[1] guyo13. “ARE NFT Projects Doing Starting Index Randomization and Provenance Wrong or Is It Just Me?” *OpenZeppelin Community*, 18 Aug. 2021, https://forum.openzeppelin.com/t/are-nft-projects-doing-starting-index-randomization-and-provenance-wrong-or-is-it-just-me/14147.
[2] “Loopy Donuts Contract.” *Etherscan*, https://etherscan.io/address/0x2106c00ac7da0a3430ae667879139e832307aeaa#code.
[3] “ BoredApeYachtClub Contract.” *Etherscan*, https://etherscan.io/address/0x60e4d786628fea6478f785a6d7e704777c86a7c6.
[4] “Introduction to Chainlink VRF.” *Chainlink*, https://docs.chain.link/docs/chainlink-vrf/.
[5] Smartcontractkit. “Chainlink External Adapter for View-Function.” *GitHub*, https://github.com/smartcontractkit/external-adapters-js/tree/develop/packages/sources/view-function.
[6] “protocol_params.Go.” *GitHub - Go-Ethereum*, 18 June 2021, https://github.com/ethereum/go-ethereum/blob/master/params/protocol_params.go.