

Gaming on Sui

Gaming on Sui leverages blockchain technology to enhance in-game economies, ownership, and interactions. With features like dynamic NFTs, kiosks, soulbound assets, and on-chain randomness, Sui provides builders with the tools to create immersive, transparent, and fair gaming experiences. Players benefit from true asset ownership, decentralized marketplaces, and seamless Web3 integration without compromising usability.

This topic explores the key features of Sui for gaming, real-world use cases, and essential tools for builders. Whether you're building an RPG, a racing game, or a digital card game, Sui provides the Web3 infrastructure to power your next-generation gaming projects.

One of the aspects that defines a Web3 game is having objects within the game reside on a blockchain. Whether it's in-game currency, battle passes, skins, or any number of other objects, knowing how to create and integrate these items is the first step in building the desired user experience on Sui.

In-game currencies allow users to purchase game features like items, upgrades, and premium content. Before Web3, these currencies remained only in the game and their values were set by the game developers.

Using Sui, players can now have true ownership of their in-game currency. The currency exists as a token on the Sui blockchain, where players can conceivably buy, sell, or swap that currency for anything else that also lives on chain. If you create an ecosystem of games, players can purchase currency in one of your games but spend it in another one that also uses that same currency. The possibilities are vast, but you must first learn how to create the currency. The following topics can get you started.

Similar to in-game currencies, tokens provide a level of engagement and control integrated into the game experience that isn't possible without Web3. For example, you can create tokens associated with your game that reward loyalty amongst your user base. You can also produce regulated tokens that allow the bearer of a specific capability to control the addresses that have access to the token, facilitating gated access to special events, leagues, or other game features. The following topics provide more insight into tokens on Sui.

NFTs are able to represent many traditional game features. For example, in-game objects, battle passes, rewards, skins, game cards or keys, and loot boxes are just a few of the features that you can create using NFTs on Sui. The following topics and sections provide more details.

On Sui, everything is an object and all objects are NFTs. For the purpose of the gaming discussion, you can also think of these NFTs as game assets.

You can view an object on Sui as a key-value pair data structure. In Move, the smart contract language of Sui, objects are defined as a struct. As an example, consider the game board of a tic-tac-toe game:

The first key-value pair for any object is `id: address`, which is a unique value of type `UID`. Every object has a different address, which is why every object is an NFT because this mandatory key-value pair sets each object apart and makes it unique.

Smart contracts contain the functions that create objects. Using the previous example, the function (`new`) that creates the object represented (a digital tic-tac-toe gameboard) provides the values for each attribute. The Sui framework creates the `UID` that uniquely identifies this particular game.

This is an on-chain action, or transaction. In this case, the transaction is a request to the chain to create an object. You provide the necessary data to the function and pay the gas fee for the computation effort of the network validators. The result is the creation of a new NFT that exists on the Sui blockchain at the address (`id`).

On Sui, you can update an NFT asset using a separate transaction, provided the smart contract that defines the NFT allows it. Similar to creation, you provide data to the relevant update function and pay the gas fee, and the smart contract updates the object at the correct address with the new information.

Using the tic-tac-toe example, you might instruct the smart contract to update the gameboard object to place an x on an available square. Because the computation effort required for most updates is less than the effort to create the original object, the resulting gas fees are typically less, as well.

For more information on the computation of gas fees, see [Sui Gas Pricing](#).

On Sui, you can include one object inside of another (dynamic fields). For example, an object named `Parent` might contain objects of type `Child`. The smart contract can provide the necessary functions to add and remove the child objects from their parents.

The function to add child objects to a parent might resemble the following. In this case, `ofield` is an alias for the `sui::dynamic_object_field` package.

You can apply dynamic fields to many use cases. An object named `TicketBooth`, for example, could contain objects of type `ConcertTicket`. An object of type `Car` might have a field named `is_functioning` that is `false` until all necessary objects (`Engine`, `Tires`, `Body`, `Wheel`) are present, at which point it can automatically update its `is_functioning` field to `true`.

Composability is a really important feature of Sui. While it provides many options when developing an idea on Sui, it also allows building on top of existing projects, using third-party assets in your project. This can range from “Only users who own a particular NFT are eligible for a discount” to full collaboration between two distinct projects and everything in between on the smart contract level.

Sui enables more use cases with the transfer to object feature, which allows an owned object to be sent to another object (shared or owned). In this case, the sent object appears as owned by the parent object. In the above examples, the objects are wrapped inside other objects and this can be seen in the parent object's metadata when inspecting the parent object.

When an object is sent to another object, the metadata of the parent object remain the same, but its id has a new object. This feature can enable use cases such as on-chain wallets where a `Wallet` object is used to deposit other objects. Combining this with transfer to object can lead to complex and exciting use cases.

See [Transfer to Object](#) for more details on how to effectively transfer objects to other objects on Sui.

On Sui, you can delete an object if the smart contract allows the operation. If the correct smart contract function is present, then you can delete the object in a single transaction. This results in a gas-fee rebate, which happens whenever bytes are freed on chain. The transaction's gas payer receives a rebate to account for the future storage of the object no longer being necessary but having already been paid for. See the [Storage fund](#) section in Tokenomics to learn more about rebates.

A soulbound asset is an NFT that belongs to an address and cannot be transferred or deleted. On Sui, assets are usually freely transferable between addresses, but this is undesirable behavior in some cases.

Assets such as game season passes, loyalty accrual assets, avatars, identification assets for a product, and assets that grant certain privileges at the smart contract level are well-suited to be bound to an address without the option of transfer or deletion.

On Sui, it is straightforward to implement such an asset, and the infrastructure guarantees that the desired behavior always holds.

You define an NFT asset as soulbound at the contract level. After designated as soulbound, it cannot be converted to a normal transferable asset. Changing it would require creating a new asset with the same name and a migration strategy, such as deleting the soulbound asset and creating a normal one to take its place.

Soulbound assets are created by omitting the store ability. Without this ability, you cannot store soulbound assets inside other objects. Keep this behavior in mind when deciding the asset type. This means that Soulbound assets are not fully composable; they can store other assets but cannot be stored inside other assets.

The same pattern can be used to implement NFT assets that are transferable/burnable only under certain circumstances. You can define these circumstances at the smart contract level by making the asset soulbound and defining custom transfer and burn functions.

The Sui Object Display standard is a tool that helps define how objects appear in apps and interfaces. It works like a template where you can insert data from an object to control how it's displayed. Use it to manage how different types of data are shown, even if the data itself is stored on chain while the display happens off chain (in apps or websites).

The Sui Object Display standard provides several key benefits and components:

There are a few limitations to be aware of. First, the current structure of Sui Object Display is per type, limiting its scope. Secondly, its structure does not allow for nested attributes or enums.

At a high level, you implement this feature using the following steps:

A display is a map of keys and values, both strings. Values allow for string interpolation, meaning the value changes dynamically depending on the NFT being viewed.

A basic example is to create a Display for Asset objects:

The above Display as is defined, for the example Asset will become:

For another Asset, the id and expiration values change according to the new Asset's values.

An app can use the Display object, where any custom keys can be understood. By default, most third-party apps like explorers or wallets recognize the attributes described below.

See [Sui Object Display](#) for more details on the standard.

Designing and managing tokens and coins on the Sui blockchain is crucial to creating a viable game economy.

GameFi (Gaming Finance) combines gaming with blockchain-based financial incentives. It provides players with economic benefits through token rewards for in-game achievements. The recent rise of GameFi has led to significant growth in token launches to support gaming ecosystems.

The terms that follow are frequently used when discussing GameFi, so it's important to make sure your definition of the terms match the documentation.

Tokenomics refers to the model and design of the rules that govern tokens for a Web3 ecosystem. In the case of GameFi, it defines how a token is created, distributed, utilized, and maintained within your gaming platform. Some of the features that comprise a token's economics include supply details, distribution mechanisms, staking and vesting.

There are several types of token supply that are considered when discussing tokenomics for an on-chain ecosystem.

There is more than one type of distribution mechanics, but they are not necessarily all used.

Ongoing distribution refers to staking and vesting of GameFi tokens. Not all game economies include staking or vesting models. When deciding whether yours should, consider the benefits:

To learn more about token vesting strategies, see [Token Vesting Strategies](#).

When deciding how best to launch your tokens, it's important to consider some key points:

On Sui, owned objects are either freely transferable or non-transferable. To ensure royalties, Sui provides a standard called Kiosk. A kiosk is a shared object that restricts access to a single address or user. See [Sui Kiosk](#) for an in-depth look at the Kiosk standard.

The kiosk "owner" (although a shared object has no owner from the perspective of Sui, the smart contract ensures that only one address is permitted to access it) is allowed to:

Any other address is allowed to:

The adoption of Kiosks implies that marketplaces become aggregators of "listed" items inside different Kiosks.

An Asset that has undergone the "lock" operation cannot undergo the "take" operation anymore; it can only undergo the "list" operation.

The "buy" operation requires the use of another Object called Transfer Policy. This is usually a Shared Object and contains Rules that govern the "buy" operation, the most common Rule being "Royalties."

A "buy" operation cannot be completed for an Asset without a defined Transfer Policy.

An empty Transfer Policy, one that does not have Rules, means that the Asset is freely tradable. Since Kiosk only allows the "list" and "buy" pair of operations, a transfer is possible by setting the price to 0 SUI.

Rules can be anything programmable with Move. To use marketplaces, an Asset creator should use Rules defined in <https://github.com/MystenLabs/apps/tree/main/kiosk/sources/rules>

Common Rules include:

The combination of the Lock Rule and the Royalty Rule enforces royalties to be paid to the creator. The Lock Rule ensures an Asset cannot be "taken" out of a Kiosk (to be freely traded), while the Royalty Rule ensures that any Asset traded through Kiosk will have royalties deducted from the transaction.

Adding the Lock Rule is recommended when royalties are a strict requirement. As long as marketplaces support only Kiosk on Sui, even without the Lock Rule, users may not have other options. It is safe to assume that peer-to-peer trading is unsafe, and most, if not all, users will avoid it since there is no way to ensure the transaction will take place smoothly. In peer-to-peer transactions, someone has to initiate either the Asset transfer or the payment transfer, and there are no guarantees that the follow-up will take place.

The most important thing is during the initial airdrop or minting of the asset to ensure the Asset is put inside a Kiosk and not sent to an address directly.

There are a number of tools available in the Sui ecosystem to help you realize your Sui game vision.

Playtron GameOS is a Linux-based operating system that seeks to turn PCs, handhelds, and desktops into dedicated gaming consoles. It supports multiple game stores like Steam and Epic Games, offering a seamless gaming experience across devices such as Steam Deck, ROG Ally, and Lenovo Legion Go.

<https://www.playtron.one/playtron-os>

E4C: Ludus is a cross-platform gaming layer designed to unify Web2 and Web3 gaming experiences on a single platform. Leveraging the Sui blockchain, it offers developers access to dynamic NFTs and zkLogin, facilitating integration of blockchain features into games across various platforms. The native E4C token is its primary currency, enabling in-game purchases and transactions. E4C: Ludus also provides a unified frontend for players to access a range of games, with the aim to enhance user engagement and simplify the gaming experience. The platform is set to launch globally in 2025, with a demo currently available for users to explore.

<https://ludus.ambrus.studio/>

Sui Coins is the utility layer for tokens and NFTs on the Sui network, offering asset management tools that include token swaps, automated dollar-cost averaging, airdrops, an incinerator for deleting assets, zkSend for private transfers, and a merger tool to consolidate small balances. Sui Coins also features an open-source SuiCoins Terminal for integrating crypto swaps across platforms.

<https://www.suicoins.com/> <https://terminal.suicoins.com/>

Beamable is a development platform that helps you integrate live services and backend features into your games. It offers SDKs for both Unity and Unreal Engine, facilitating development and deployment of online game functionalities. You can incorporate features such as player authentication, inventory management, and microservices within the environments of your chosen game engines. Beamable provides support for the Sui blockchain, allowing for the integration of Web3 elements like NFTs and on-chain assets into games. The Beamable SDKs offer tools and sample projects to help you build on the Sui network.

<https://beamable.com/>

Forge is a platform that enables game developers to create custom loyalty programs, rewarding players for engaging in community activities and in-game challenges. Players earn loyalty points by completing actions you define, which can be redeemed for in-game items and digital content. Forge also offers analytics tools to help developers understand their audience and improve monetization strategies.

<https://forge.gg/>

Snag Solutions provides white-label loyalty and marketplace platforms to enhance community engagement and control your digital ecosystems. Their solutions enable you to track and reward user contributions, create customizable marketplaces, and integrate social features like peer-to-peer trading and user profiles. Snag Solutions offers customization options, APIs, and SDKs to align with your brand's identity.

<https://www.snagsolutions.io/>

Venly is a developer platform that aids blockchain integration for businesses through secure digital wallets, tokenization services, and payment solutions. It offers APIs and SDKs for management of digital assets. Venly enables you to create, trade, and manage NFTs, tokens, and payments securely while maintaining full ownership of your assets.

<https://www.venly.io/>

Cycle through the available videos using the thumbnails, then tap or click the video to play.

There aren't real-world implementations for the integrations described in this section. These examples are meant to be a thought exercise to showcase the possibilities for viable game integration on the Sui network.

ShadowQuest is a multiplayer game that combines fantasy with RPG battle mechanics. To enhance the gaming experience with Web3 technologies, ShadowQuest is integrating with Sui offering seamless blockchain interactions to players without compromising the overall gaming experience.

Seamless player onboarding and wallet integration

ShadowQuest wants to onboard players without adding complexity, especially those unfamiliar with Web3. By using [zkLogin](#), users can sign in using social platforms like Google, Facebook, Twitch, and Apple. This automatically creates a Sui wallet linked to their ShadowQuest account, making blockchain interactions seamless.

Simplified transaction handling

Players in ShadowQuest earn or use in-game assets such as NFTs or \$SHADOW tokens. To attract users unfamiliar with Web3, ShadowQuest manages game transactions to avoid Web3 friction, like wallets popping up for signing transactions. By sponsoring user transactions, the friction is minimized because ShadowQuest users do not directly pay for transaction costs and gas fees.

Enoki transactions can be signed without requiring confirmation from the user to approve the transaction.

ShadowQuest uses Enoki Gas Pool to sponsor transactions, covering gas fees for players. This ensures all in-game transactions are seamless and cost-free for players, providing a better user experience.

NFT marketplace and royalty enforcement

ShadowQuest allows players to buy, sell, or trade in-game items, such as weapons, armor, and cosmetics. The items should respect royalties to ensure that creators benefit from each transaction.

Kiosk provides a decentralized marketplace solution, ensuring royalties are enforced on all NFT trades. This helps both the game developers and creators maintain control over secondary sales, ensuring revenue generation throughout the asset's lifecycle.

NFT usage for game access

ShadowQuest uses NFTs as entry tickets for different game modes and events. Players can acquire or earn various Runes, which grant access to specific game challenges or seasonal competitions. These NFTs cannot be traded or transferred to other players.

Soulbound NFTs represent different Runes that are either earned through gameplay or purchased. These NFTs grant exclusive access to matches and seasonal challenges but cannot be traded after bound to a player.

Sui for Speed is a racing game set in the Sui ecosystem. Players pilot customizable vehicles through fantastical terrains, competing in races, time trials, and exploration challenges. By integrating with the Sui network's blockchain technology—including features such as Walrus, dynamic NFTs, SuiNS, and asset tokenization—the game offers players true ownership of their vehicles and in-game assets, along with a vibrant, player-driven economy.

Customizable vehicles with dynamic NFTs

In Sui for Speed, players own racing vehicles represented by dynamic NFTs that you can upgrade and customize with new parts, skins, and abilities. As players progress and win races, their vehicles evolve, reflecting their achievements and style.

Dynamic NFTs on the Sui network allow vehicles to securely update attributes and metadata over time. Every upgrade and customization is recorded on-chain, ensuring each vehicle's uniqueness and authenticity.

SuiNS: Personalized racer profiles and teams

Players can register unique names for their racer profiles and teams using SuiNS, like [speedster@suihorspeed.sui](#) or [dragonracer@suihorspeed.sui](#). This simplifies social interactions, team coordination, and improves the community aspect of the game.

SuiNS provides a decentralized domain naming system, allowing memorable and personalized names on the blockchain.

Tokenized circuits and earnings from circuit usage

Race circuits are tokenized as unique NFTs allowing players to own, design, and vote to enable track customization and drive better changes. When other players race on these circuits, the owners earn \$RALLY tokens as usage fees or royalties. This system incentivizes creativity and allows players to monetize their track designs.

Asset tokenization on the Sui network enables minting of circuits as unique assets with secure ownership. Smart contracts automatically distribute earnings to circuit owners when their tracks are used, enhancing the game's economy through player-driven content.

Decentralized storage for game data

The game world includes extensive data such as track designs and leaderboards. Leveraging decentralized and efficient storage,

combined with asset tokenization, enables true decentralization of these terrains and models.

Walrus offers scalable off-chain storage for large amounts of game data. This ensures high availability and security, protecting against data loss and enhancing player trust.

Competitive events and betting mechanisms

The game hosts regular competitive events and tournaments where players can participate individually or as teams. Additionally, players can place bets on race outcomes using \$RALLY tokens, adding an extra layer of excitement and engagement.

On-chain logic enables secure and transparent management of events and betting systems. The Sui network ensures fairness, with immutable records of bets and outcomes.

ArcaneBattles is a strategic, multiplayer card game inspired by classics like Hearthstone. Players collect, trade, and battle with a variety of magical cards representing spells, creatures, and heroes. By integrating with the Sui blockchain, ArcaneBattles aims to enrich the gameplay experience through decentralized features that promote true ownership, fairness, and a dynamic in-game economy.

Dynamic in-game economy with two closed loop tokens

ArcaneBattles implemented two stable in-game currencies to facilitate various transactions, enhancing player engagement and economic depth.

Closed loop token (CLT)

ArcaneBattles utilizes two Closed Loop Tokens within its ecosystem:

The dual-token system adds depth to the in-game economy, encouraging players to engage in various activities and make strategic decisions about resource allocation.

True ownership and NFT card rental

ArcaneBattles allows players to own their cards as NFTs and provide a rental marketplace for rare or powerful cards.

Each card in ArcaneBattles is represented as an NFT on the Sui blockchain, granting players true ownership of their digital assets. The NFT Rental feature allows players to rent out their rare or high-level cards to others for a fee. This creates a community-driven economy where new or casual players can access powerful cards temporarily, while owners earn passive income from their collections.

Fair and unpredictable gameplay

ArcaneBattles ensures randomness in card draws and in-game events to prevent manipulation and enhance fairness.

By leveraging Sui's on-chain randomness, ArcaneBattles introduces unpredictable elements such as random card draws, critical hit chances, and random effects from certain cards. This randomness is verifiable and secure, preventing manipulation by any party and maintaining fairness across all gameplay aspects.

Enhanced card visualization and dynamic wear mechanism

ArcaneBattles provides rich, dynamic displays of card information and introduces a wear-and-tear mechanic to simulate card degradation over time.

ArcaneBattles uses Sui's Display Standard to offer detailed metadata for each card NFT, including stats, abilities, and artwork. Beyond static information, the game introduces a dynamic display mechanism where cards visually show signs of wear as they are used in battles. Over time, frequently used cards may appear scratched, faded, or have other visual cues indicating wear. After extensive use, cards have a chance to be destroyed entirely.

This wear-and-tear mechanic simulates the experience of physical card games, where rare cards are often kept in pristine condition and used sparingly. It encourages players to strategically decide when to use their valuable cards and adds a layer of depth to the game's economy and strategy.

Players can mitigate or repair wear on their cards by using Mystic Dust to restore them or upgrade them to gold versions, which are more durable and feature enhanced visuals. This system adds a strategic resource management element, as players must balance the benefits of using powerful cards against the potential cost of their degradation.

By integrating these Sui blockchain features, ArcaneBattles not only enhances the gaming experience but also pioneers the next generation of digital card games. The dual-token economy adds complexity and depth to in-game transactions, encouraging strategic decision-making. The wear-and-tear mechanic introduces a novel layer of strategy and realism, as players must consider the longevity of their cards. Together, these features create a rich, engaging, and immersive experience that leverages the full potential of blockchain technology within a gaming context.

Representing in-game objects on chain

One of the aspects that defines a Web3 game is having objects within the game reside on a blockchain. Whether it's in-game currency, battle passes, skins, or any number of other objects, knowing how to create and integrate these items is the first step in building the desired user experience on Sui.

In-game currencies allow users to purchase game features like items, upgrades, and premium content. Before Web3, these currencies remained only in the game and their values were set by the game developers.

Using Sui, players can now have true ownership of their in-game currency. The currency exists as a token on the Sui blockchain, where players can conceivably buy, sell, or swap that currency for anything else that also lives on chain. If you create an ecosystem of games, players can purchase currency in one of your games but spend it in another one that also uses that same currency. The possibilities are vast, but you must first learn how to create the currency. The following topics can get you started.

Similar to in-game currencies, tokens provide a level of engagement and control integrated into the game experience that isn't possible without Web3. For example, you can create tokens associated with your game that reward loyalty amongst your user base. You can also produce regulated tokens that allow the bearer of a specific capability to control the addresses that have access to the token, facilitating gated access to special events, leagues, or other game features. The following topics provide more insight into tokens on Sui.

NFTs are able to represent many traditional game features. For example, in-game objects, battle passes, rewards, skins, game cards or keys, and loot boxes are just a few of the features that you can create using NFTs on Sui. The following topics and sections provide more details.

On Sui, everything is an object and all objects are NFTs. For the purpose of the gaming discussion, you can also think of these NFTs as game assets.

You can view an object on Sui as a key-value pair data structure. In Move, the smart contract language of Sui, objects are defined as a struct. As an example, consider the game board of a tic-tac-toe game:

The first key-value pair for any object is `id: address`, which is a unique value of type `UID`. Every object has a different address, which is why every object is an NFT because this mandatory key-value pair sets each object apart and makes it unique.

Smart contracts contain the functions that create objects. Using the previous example, the function (`new`) that creates the object represented (a digital tic-tac-toe gameboard) provides the values for each attribute. The Sui framework creates the `UID` that uniquely identifies this particular game.

This is an on-chain action, or transaction. In this case, the transaction is a request to the chain to create an object. You provide the necessary data to the function and pay the gas fee for the computation effort of the network validators. The result is the creation of a new NFT that exists on the Sui blockchain at the address (`id`).

On Sui, you can update an NFT asset using a separate transaction, provided the smart contract that defines the NFT allows it. Similar to creation, you provide data to the relevant update function and pay the gas fee, and the smart contract updates the object at the correct address with the new information.

Using the tic-tac-toe example, you might instruct the smart contract to update the gameboard object to place an x on an available square. Because the computation effort required for most updates is less than the effort to create the original object, the resulting gas fees are typically less, as well.

For more information on the computation of gas fees, see [Sui Gas Pricing](#).

On Sui, you can include one object inside of another (dynamic fields). For example, an object named `Parent` might contain objects of type `Child`. The smart contract can provide the necessary functions to add and remove the child objects from their parents.

The function to add child objects to a parent might resemble the following. In this case, `ofield` is an alias for the `sui::dynamic_object_field` package.

You can apply dynamic fields to many use cases. An object named `TicketBooth`, for example, could contain objects of type `ConcertTicket`. An object of type `Car` might have a field named `is_functioning` that is `false` until all necessary objects (`Engine`, `Tires`, `Body`, `Wheel`) are present, at which point it can automatically update its `is_functioning` field to `true`.

Composability is a really important feature of Sui. While it provides many options when developing an idea on Sui, it also allows building on top of existing projects, using third-party assets in your project. This can range from “Only users who own a particular NFT are eligible for a discount” to full collaboration between two distinct projects and everything in between on the smart contract level.

Sui enables more use cases with the transfer to object feature, which allows an owned object to be sent to another object (shared or owned). In this case, the sent object appears as owned by the parent object. In the above examples, the objects are wrapped inside other objects and this can be seen in the parent object's metadata when inspecting the parent object.

When an object is sent to another object, the metadata of the parent object remain the same, but its id has a new object. This feature can enable use cases such as on-chain wallets where a `Wallet` object is used to deposit other objects. Combining this with transfer to object can lead to complex and exciting use cases.

See [Transfer to Object](#) for more details on how to effectively transfer objects to other objects on Sui.

On Sui, you can delete an object if the smart contract allows the operation. If the correct smart contract function is present, then you can delete the object in a single transaction. This results in a gas-fee rebate, which happens whenever bytes are freed on chain. The transaction's gas payer receives a rebate to account for the future storage of the object no longer being necessary but having already been paid for. See the [Storage fund](#) section in Tokenomics to learn more about rebates.

A soulbound asset is an NFT that belongs to an address and cannot be transferred or deleted. On Sui, assets are usually freely transferable between addresses, but this is undesirable behavior in some cases.

Assets such as game season passes, loyalty accrual assets, avatars, identification assets for a product, and assets that grant certain privileges at the smart contract level are well-suited to be bound to an address without the option of transfer or deletion.

On Sui, it is straightforward to implement such an asset, and the infrastructure guarantees that the desired behavior always holds.

You define an NFT asset as soulbound at the contract level. After designated as soulbound, it cannot be converted to a normal transferable asset. Changing it would require creating a new asset with the same name and a migration strategy, such as deleting the soulbound asset and creating a normal one to take its place.

Soulbound assets are created by omitting the store ability. Without this ability, you cannot store soulbound assets inside other objects. Keep this behavior in mind when deciding the asset type. This means that Soulbound assets are not fully composable; they can store other assets but cannot be stored inside other assets.

The same pattern can be used to implement NFT assets that are transferable/burnable only under certain circumstances. You can define these circumstances at the smart contract level by making the asset soulbound and defining custom transfer and burn functions.

The Sui Object Display standard is a tool that helps define how objects appear in apps and interfaces. It works like a template where you can insert data from an object to control how it's displayed. Use it to manage how different types of data are shown, even if the data itself is stored on chain while the display happens off chain (in apps or websites).

The Sui Object Display standard provides several key benefits and components:

There are a few limitations to be aware of. First, the current structure of Sui Object Display is per type, limiting its scope. Secondly, its structure does not allow for nested attributes or enums.

At a high level, you implement this feature using the following steps:

A display is a map of keys and values, both strings. Values allow for string interpolation, meaning the value changes dynamically depending on the NFT being viewed.

A basic example is to create a Display for Asset objects:

The above Display as is defined, for the example Asset will become:

For another Asset, the id and expiration values change according to the new Asset's values.

An app can use the Display object, where any custom keys can be understood. By default, most third-party apps like explorers or wallets recognize the attributes described below.

See [Sui Object Display](#) for more details on the standard.

Designing and managing tokens and coins on the Sui blockchain is crucial to creating a viable game economy.

GameFi (Gaming Finance) combines gaming with blockchain-based financial incentives. It provides players with economic benefits through token rewards for in-game achievements. The recent rise of GameFi has led to significant growth in token launches to support gaming ecosystems.

The terms that follow are frequently used when discussing GameFi, so it's important to make sure your definition of the terms match the documentation.

Tokenomics refers to the model and design of the rules that govern tokens for a Web3 ecosystem. In the case of GameFi, it defines how a token is created, distributed, utilized, and maintained within your gaming platform. Some of the features that comprise a token's economics include supply details, distribution mechanisms, staking and vesting.

There are several types of token supply that are considered when discussing tokenomics for an on-chain ecosystem.

There is more than one type of distribution mechanics, but they are not necessarily all used.

Ongoing distribution refers to staking and vesting of GameFi tokens. Not all game economies include staking or vesting models. When deciding whether yours should, consider the benefits:

To learn more about token vesting strategies, see [Token Vesting Strategies](#).

When deciding how best to launch your tokens, it's important to consider some key points:

On Sui, owned objects are either freely transferable or non-transferable. To ensure royalties, Sui provides a standard called Kiosk. A kiosk is a shared object that restricts access to a single address or user. See [Sui Kiosk](#) for an in-depth look at the Kiosk standard.

The kiosk "owner" (although a shared object has no owner from the perspective of Sui, the smart contract ensures that only one address is permitted to access it) is allowed to:

Any other address is allowed to:

The adoption of Kiosks implies that marketplaces become aggregators of "listed" items inside different Kiosks.

An Asset that has undergone the "lock" operation cannot undergo the "take" operation anymore; it can only undergo the "list" operation.

The "buy" operation requires the use of another Object called Transfer Policy. This is usually a Shared Object and contains Rules that govern the "buy" operation, the most common Rule being "Royalties."

A "buy" operation cannot be completed for an Asset without a defined Transfer Policy.

An empty Transfer Policy, one that does not have Rules, means that the Asset is freely tradable. Since Kiosk only allows the "list" and "buy" pair of operations, a transfer is possible by setting the price to 0 SUI.

Rules can be anything programmable with Move. To use marketplaces, an Asset creator should use Rules defined in <https://github.com/MystenLabs/apps/tree/main/kiosk/sources/rules>

Common Rules include:

The combination of the Lock Rule and the Royalty Rule enforces royalties to be paid to the creator. The Lock Rule ensures an Asset cannot be "taken" out of a Kiosk (to be freely traded), while the Royalty Rule ensures that any Asset traded through Kiosk will have royalties deducted from the transaction.

Adding the Lock Rule is recommended when royalties are a strict requirement. As long as marketplaces support only Kiosk on Sui, even without the Lock Rule, users may not have other options. It is safe to assume that peer-to-peer trading is unsafe, and most, if not all, users will avoid it since there is no way to ensure the transaction will take place smoothly. In peer-to-peer transactions, someone has to initiate either the Asset transfer or the payment transfer, and there are no guarantees that the follow-up will take place.

The most important thing is during the initial airdrop or minting of the asset to ensure the Asset is put inside a Kiosk and not sent to an address directly.

There are a number of tools available in the Sui ecosystem to help you realize your Sui game vision.

Playtron GameOS is a Linux-based operating system that seeks to turn PCs, handhelds, and desktops into dedicated gaming consoles. It supports multiple game stores like Steam and Epic Games, offering a seamless gaming experience across devices such as Steam Deck, ROG Ally, and Lenovo Legion Go.

<https://www.playtron.one/playtron-os>

E4C: Ludus is a cross-platform gaming layer designed to unify Web2 and Web3 gaming experiences on a single platform. Leveraging the Sui blockchain, it offers developers access to dynamic NFTs and zkLogin, facilitating integration of blockchain features into games across various platforms. The native E4C token is its primary currency, enabling in-game purchases and transactions. E4C: Ludus also provides a unified frontend for players to access a range of games, with the aim to enhance user engagement and simplify the gaming experience. The platform is set to launch globally in 2025, with a demo currently available for users to explore.

<https://ludus.ambrus.studio/>

Sui Coins is the utility layer for tokens and NFTs on the Sui network, offering asset management tools that include token swaps, automated dollar-cost averaging, airdrops, an incinerator for deleting assets, zkSend for private transfers, and a merger tool to consolidate small balances. Sui Coins also features an open-source SuiCoins Terminal for integrating crypto swaps across platforms.

<https://www.suicoins.com/> <https://terminal.suicoins.com/>

Beamable is a development platform that helps you integrate live services and backend features into your games. It offers SDKs for both Unity and Unreal Engine, facilitating development and deployment of online game functionalities. You can incorporate features such as player authentication, inventory management, and microservices within the environments of your chosen game engines. Beamable provides support for the Sui blockchain, allowing for the integration of Web3 elements like NFTs and on-chain assets into games. The Beamable SDKs offer tools and sample projects to help you build on the Sui network.

<https://beamable.com/>

Forge is a platform that enables game developers to create custom loyalty programs, rewarding players for engaging in community activities and in-game challenges. Players earn loyalty points by completing actions you define, which can be redeemed for in-game items and digital content. Forge also offers analytics tools to help developers understand their audience and improve monetization strategies.

<https://forge.gg/>

Snag Solutions provides white-label loyalty and marketplace platforms to enhance community engagement and control your digital ecosystems. Their solutions enable you to track and reward user contributions, create customizable marketplaces, and integrate social features like peer-to-peer trading and user profiles. Snag Solutions offers customization options, APIs, and SDKs to align with your brand's identity.

<https://www.snagsolutions.io/>

Venly is a developer platform that aids blockchain integration for businesses through secure digital wallets, tokenization services, and payment solutions. It offers APIs and SDKs for management of digital assets. Venly enables you to create, trade, and manage NFTs, tokens, and payments securely while maintaining full ownership of your assets.

<https://www.venly.io/>

Cycle through the available videos using the thumbnails, then tap or click the video to play.

There aren't real-world implementations for the integrations described in this section. These examples are meant to be a thought exercise to showcase the possibilities for viable game integration on the Sui network.

ShadowQuest is a multiplayer game that combines fantasy with RPG battle mechanics. To enhance the gaming experience with Web3 technologies, ShadowQuest is integrating with Sui offering seamless blockchain interactions to players without compromising the overall gaming experience.

Seamless player onboarding and wallet integration

ShadowQuest wants to onboard players without adding complexity, especially those unfamiliar with Web3. By using [zkLogin](#), users can sign in using social platforms like Google, Facebook, Twitch, and Apple. This automatically creates a Sui wallet linked to their ShadowQuest account, making blockchain interactions seamless.

Simplified transaction handling

Players in ShadowQuest earn or use in-game assets such as NFTs or \$SHADOW tokens. To attract users unfamiliar with Web3, ShadowQuest manages game transactions to avoid Web3 friction, like wallets popping up for signing transactions. By sponsoring user transactions, the friction is minimized because ShadowQuest users do not directly pay for transaction costs and gas fees.

Enoki transactions can be signed without requiring confirmation from the user to approve the transaction.

ShadowQuest uses Enoki Gas Pool to sponsor transactions, covering gas fees for players. This ensures all in-game transactions are seamless and cost-free for players, providing a better user experience.

NFT marketplace and royalty enforcement

ShadowQuest allows players to buy, sell, or trade in-game items, such as weapons, armor, and cosmetics. The items should respect royalties to ensure that creators benefit from each transaction.

Kiosk provides a decentralized marketplace solution, ensuring royalties are enforced on all NFT trades. This helps both the game developers and creators maintain control over secondary sales, ensuring revenue generation throughout the asset's lifecycle.

NFT usage for game access

ShadowQuest uses NFTs as entry tickets for different game modes and events. Players can acquire or earn various Runes, which grant access to specific game challenges or seasonal competitions. These NFTs cannot be traded or transferred to other players.

Soulbound NFTs represent different Runes that are either earned through gameplay or purchased. These NFTs grant exclusive access to matches and seasonal challenges but cannot be traded after bound to a player.

Sui for Speed is a racing game set in the Sui ecosystem. Players pilot customizable vehicles through fantastical terrains, competing in races, time trials, and exploration challenges. By integrating with the Sui network's blockchain technology—including features such as Walrus, dynamic NFTs, SuiNS, and asset tokenization—the game offers players true ownership of their vehicles and in-game assets, along with a vibrant, player-driven economy.

Customizable vehicles with dynamic NFTs

In Sui for Speed, players own racing vehicles represented by dynamic NFTs that you can upgrade and customize with new parts, skins, and abilities. As players progress and win races, their vehicles evolve, reflecting their achievements and style.

Dynamic NFTs on the Sui network allow vehicles to securely update attributes and metadata over time. Every upgrade and customization is recorded on-chain, ensuring each vehicle's uniqueness and authenticity.

SuiNS: Personalized racer profiles and teams

Players can register unique names for their racer profiles and teams using SuiNS, like speedster@suiforspeed.sui or dragonracer@suiforspeed.sui. This simplifies social interactions, team coordination, and improves the community aspect of the game.

SuiNS provides a decentralized domain naming system, allowing memorable and personalized names on the blockchain.

Tokenized circuits and earnings from circuit usage

Race circuits are tokenized as unique NFTs allowing players to own, design, and vote to enable track customization and drive better changes. When other players race on these circuits, the owners earn \$RALLY tokens as usage fees or royalties. This system incentivizes creativity and allows players to monetize their track designs.

Asset tokenization on the Sui network enables minting of circuits as unique assets with secure ownership. Smart contracts automatically distribute earnings to circuit owners when their tracks are used, enhancing the game's economy through player-driven content.

Decentralized storage for game data

The game world includes extensive data such as track designs and leaderboards. Leveraging decentralized and efficient storage, combined with asset tokenization, enables true decentralization of these terrains and models.

Walrus offers scalable off-chain storage for large amounts of game data. This ensures high availability and security, protecting against

data loss and enhancing player trust.

Competitive events and betting mechanisms

The game hosts regular competitive events and tournaments where players can participate individually or as teams. Additionally, players can place bets on race outcomes using \$RALLY tokens, adding an extra layer of excitement and engagement.

On-chain logic enables secure and transparent management of events and betting systems. The Sui network ensures fairness, with immutable records of bets and outcomes.

ArcaneBattles is a strategic, multiplayer card game inspired by classics like Hearthstone. Players collect, trade, and battle with a variety of magical cards representing spells, creatures, and heroes. By integrating with the Sui blockchain, ArcaneBattles aims to enrich the gameplay experience through decentralized features that promote true ownership, fairness, and a dynamic in-game economy.

Dynamic in-game economy with two closed loop tokens

ArcaneBattles implemented two stable in-game currencies to facilitate various transactions, enhancing player engagement and economic depth.

Closed loop token (CLT)

ArcaneBattles utilizes two Closed Loop Tokens within its ecosystem:

The dual-token system adds depth to the in-game economy, encouraging players to engage in various activities and make strategic decisions about resource allocation.

True ownership and NFT card rental

ArcaneBattles allows players to own their cards as NFTs and provide a rental marketplace for rare or powerful cards.

Each card in ArcaneBattles is represented as an NFT on the Sui blockchain, granting players true ownership of their digital assets. The NFT Rental feature allows players to rent out their rare or high-level cards to others for a fee. This creates a community-driven economy where new or casual players can access powerful cards temporarily, while owners earn passive income from their collections.

Fair and unpredictable gameplay

ArcaneBattles ensures randomness in card draws and in-game events to prevent manipulation and enhance fairness.

By leveraging Sui's on-chain randomness, ArcaneBattles introduces unpredictable elements such as random card draws, critical hit chances, and random effects from certain cards. This randomness is verifiable and secure, preventing manipulation by any party and maintaining fairness across all gameplay aspects.

Enhanced card visualization and dynamic wear mechanism

ArcaneBattles provides rich, dynamic displays of card information and introduces a wear-and-tear mechanic to simulate card degradation over time.

ArcaneBattles uses Sui's Display Standard to offer detailed metadata for each card NFT, including stats, abilities, and artwork. Beyond static information, the game introduces a dynamic display mechanism where cards visually show signs of wear as they are used in battles. Over time, frequently used cards may appear scratched, faded, or have other visual cues indicating wear. After extensive use, cards have a chance to be destroyed entirely.

This wear-and-tear mechanic simulates the experience of physical card games, where rare cards are often kept in pristine condition and used sparingly. It encourages players to strategically decide when to use their valuable cards and adds a layer of depth to the game's economy and strategy.

Players can mitigate or repair wear on their cards by using Mystic Dust to restore them or upgrade them to gold versions, which are more durable and feature enhanced visuals. This system adds a strategic resource management element, as players must balance the benefits of using powerful cards against the potential cost of their degradation.

By integrating these Sui blockchain features, ArcaneBattles not only enhances the gaming experience but also pioneers the next generation of digital card games. The dual-token economy adds complexity and depth to in-game transactions, encouraging strategic decision-making. The wear-and-tear mechanic introduces a novel layer of strategy and realism, as players must consider the

longevity of their cards. Together, these features create a rich, engaging, and immersive experience that leverages the full potential of blockchain technology within a gaming context.

Game economies

Designing and managing tokens and coins on the Sui blockchain is crucial to creating a viable game economy.

GameFi (Gaming Finance) combines gaming with blockchain-based financial incentives. It provides players with economic benefits through token rewards for in-game achievements. The recent rise of GameFi has led to significant growth in token launches to support gaming ecosystems.

The terms that follow are frequently used when discussing GameFi, so it's important to make sure your definition of the terms match the documentation.

Tokenomics refers to the model and design of the rules that govern tokens for a Web3 ecosystem. In the case of GameFi, it defines how a token is created, distributed, utilized, and maintained within your gaming platform. Some of the features that comprise a token's economics include supply details, distribution mechanisms, staking and vesting.

There are several types of token supply that are considered when discussing tokenomics for an on-chain ecosystem.

There is more than one type of distribution mechanics, but they are not necessarily all used.

Ongoing distribution refers to staking and vesting of GameFi tokens. Not all game economies include staking or vesting models. When deciding whether yours should, consider the benefits:

To learn more about token vesting strategies, see [Token Vesting Strategies](#).

When deciding how best to launch your tokens, it's important to consider some key points:

On Sui, owned objects are either freely transferable or non-transferable. To ensure royalties, Sui provides a standard called Kiosk. A kiosk is a shared object that restricts access to a single address or user. See [Sui Kiosk](#) for an in-depth look at the Kiosk standard.

The kiosk "owner" (although a shared object has no owner from the perspective of Sui, the smart contract ensures that only one address is permitted to access it) is allowed to:

Any other address is allowed to:

The adoption of Kiosks implies that marketplaces become aggregators of "listed" items inside different Kiosks.

An Asset that has undergone the "lock" operation cannot undergo the "take" operation anymore; it can only undergo the "list" operation.

The "buy" operation requires the use of another Object called Transfer Policy. This is usually a Shared Object and contains Rules that govern the "buy" operation, the most common Rule being "Royalties."

A "buy" operation cannot be completed for an Asset without a defined Transfer Policy.

An empty Transfer Policy, one that does not have Rules, means that the Asset is freely tradable. Since Kiosk only allows the "list" and "buy" pair of operations, a transfer is possible by setting the price to 0 SUI.

Rules can be anything programmable with Move. To use marketplaces, an Asset creator should use Rules defined in <https://github.com/MystenLabs/apps/tree/main/kiosk/sources/rules>

Common Rules include:

The combination of the Lock Rule and the Royalty Rule enforces royalties to be paid to the creator. The Lock Rule ensures an Asset cannot be "taken" out of a Kiosk (to be freely traded), while the Royalty Rule ensures that any Asset traded through Kiosk will have royalties deducted from the transaction.

Adding the Lock Rule is recommended when royalties are a strict requirement. As long as marketplaces support only Kiosk on Sui, even without the Lock Rule, users may not have other options. It is safe to assume that peer-to-peer trading is unsafe, and most, if not all, users will avoid it since there is no way to ensure the transaction will take place smoothly. In peer-to-peer transactions, someone has to initiate either the Asset transfer or the payment transfer, and there are no guarantees that the follow-up will take

place.

The most important thing is during the initial airdrop or minting of the asset to ensure the Asset is put inside a Kiosk and not sent to an address directly.

There are a number of tools available in the Sui ecosystem to help you realize your Sui game vision.

Playtron GameOS is a Linux-based operating system that seeks to turn PCs, handhelds, and desktops into dedicated gaming consoles. It supports multiple game stores like Steam and Epic Games, offering a seamless gaming experience across devices such as Steam Deck, ROG Ally, and Lenovo Legion Go.

<https://www.playtron.one/playtron-os>

E4C: Ludus is a cross-platform gaming layer designed to unify Web2 and Web3 gaming experiences on a single platform. Leveraging the Sui blockchain, it offers developers access to dynamic NFTs and zkLogin, facilitating integration of blockchain features into games across various platforms. The native E4C token is its primary currency, enabling in-game purchases and transactions. E4C: Ludus also provides a unified frontend for players to access a range of games, with the aim to enhance user engagement and simplify the gaming experience. The platform is set to launch globally in 2025, with a demo currently available for users to explore.

<https://ludus.ambrus.studio/>

Sui Coins is the utility layer for tokens and NFTs on the Sui network, offering asset management tools that include token swaps, automated dollar-cost averaging, airdrops, an incinerator for deleting assets, zkSend for private transfers, and a merger tool to consolidate small balances. Sui Coins also features an open-source SuiCoins Terminal for integrating crypto swaps across platforms.

<https://www.suicoins.com/> <https://terminal.suicoins.com/>

Beamable is a development platform that helps you integrate live services and backend features into your games. It offers SDKs for both Unity and Unreal Engine, facilitating development and deployment of online game functionalities. You can incorporate features such as player authentication, inventory management, and microservices within the environments of your chosen game engines. Beamable provides support for the Sui blockchain, allowing for the integration of Web3 elements like NFTs and on-chain assets into games. The Beamable SDKs offer tools and sample projects to help you build on the Sui network.

<https://beamable.com/>

Forge is a platform that enables game developers to create custom loyalty programs, rewarding players for engaging in community activities and in-game challenges. Players earn loyalty points by completing actions you define, which can be redeemed for in-game items and digital content. Forge also offers analytics tools to help developers understand their audience and improve monetization strategies.

<https://forge.gg/>

Snag Solutions provides white-label loyalty and marketplace platforms to enhance community engagement and control your digital ecosystems. Their solutions enable you to track and reward user contributions, create customizable marketplaces, and integrate social features like peer-to-peer trading and user profiles. Snag Solutions offers customization options, APIs, and SDKs to align with your brand's identity.

<https://www.snagsolutions.io/>

Venly is a developer platform that aids blockchain integration for businesses through secure digital wallets, tokenization services, and payment solutions. It offers APIs and SDKs for management of digital assets. Venly enables you to create, trade, and manage NFTs, tokens, and payments securely while maintaining full ownership of your assets.

<https://www.venly.io/>

Cycle through the available videos using the thumbnails, then tap or click the video to play.

There aren't real-world implementations for the integrations described in this section. These examples are meant to be a thought exercise to showcase the possibilities for viable game integration on the Sui network.

ShadowQuest is a multiplayer game that combines fantasy with RPG battle mechanics. To enhance the gaming experience with Web3 technologies, ShadowQuest is integrating with Sui offering seamless blockchain interactions to players without compromising the overall gaming experience.

Seamless player onboarding and wallet integration

ShadowQuest wants to onboard players without adding complexity, especially those unfamiliar with Web3. By using [zkLogin](#), users can sign in using social platforms like Google, Facebook, Twitch, and Apple. This automatically creates a Sui wallet linked to their ShadowQuest account, making blockchain interactions seamless.

Simplified transaction handling

Players in ShadowQuest earn or use in-game assets such as NFTs or \$SHADOW tokens. To attract users unfamiliar with Web3, ShadowQuest manages game transactions to avoid Web3 friction, like wallets popping up for signing transactions. By sponsoring user transactions, the friction is minimized because ShadowQuest users do not directly pay for transaction costs and gas fees.

Enoki transactions can be signed without requiring confirmation from the user to approve the transaction.

ShadowQuest uses Enoki Gas Pool to sponsor transactions, covering gas fees for players. This ensures all in-game transactions are seamless and cost-free for players, providing a better user experience.

NFT marketplace and royalty enforcement

ShadowQuest allows players to buy, sell, or trade in-game items, such as weapons, armor, and cosmetics. The items should respect royalties to ensure that creators benefit from each transaction.

Kiosk provides a decentralized marketplace solution, ensuring royalties are enforced on all NFT trades. This helps both the game developers and creators maintain control over secondary sales, ensuring revenue generation throughout the asset's lifecycle.

NFT usage for game access

ShadowQuest uses NFTs as entry tickets for different game modes and events. Players can acquire or earn various Runes, which grant access to specific game challenges or seasonal competitions. These NFTs cannot be traded or transferred to other players.

Soulbound NFTs represent different Runes that are either earned through gameplay or purchased. These NFTs grant exclusive access to matches and seasonal challenges but cannot be traded after bound to a player.

Sui for Speed is a racing game set in the Sui ecosystem. Players pilot customizable vehicles through fantastical terrains, competing in races, time trials, and exploration challenges. By integrating with the Sui network's blockchain technology—including features such as Walrus, dynamic NFTs, SuiNS, and asset tokenization—the game offers players true ownership of their vehicles and in-game assets, along with a vibrant, player-driven economy.

Customizable vehicles with dynamic NFTs

In Sui for Speed, players own racing vehicles represented by dynamic NFTs that you can upgrade and customize with new parts, skins, and abilities. As players progress and win races, their vehicles evolve, reflecting their achievements and style.

Dynamic NFTs on the Sui network allow vehicles to securely update attributes and metadata over time. Every upgrade and customization is recorded on-chain, ensuring each vehicle's uniqueness and authenticity.

SuiNS: Personalized racer profiles and teams

Players can register unique names for their racer profiles and teams using SuiNS, like speedster@suiforspeed.sui or dragonracer@suiforspeed.sui. This simplifies social interactions, team coordination, and improves the community aspect of the game.

SuiNS provides a decentralized domain naming system, allowing memorable and personalized names on the blockchain.

Tokenized circuits and earnings from circuit usage

Race circuits are tokenized as unique NFTs allowing players to own, design, and vote to enable track customization and drive better changes. When other players race on these circuits, the owners earn \$RALLY tokens as usage fees or royalties. This system incentivizes creativity and allows players to monetize their track designs.

Asset tokenization on the Sui network enables minting of circuits as unique assets with secure ownership. Smart contracts automatically distribute earnings to circuit owners when their tracks are used, enhancing the game's economy through player-driven content.

Decentralized storage for game data

The game world includes extensive data such as track designs and leaderboards. Leveraging decentralized and efficient storage, combined with asset tokenization, enables true decentralization of these terrains and models.

Walrus offers scalable off-chain storage for large amounts of game data. This ensures high availability and security, protecting against data loss and enhancing player trust.

Competitive events and betting mechanisms

The game hosts regular competitive events and tournaments where players can participate individually or as teams. Additionally, players can place bets on race outcomes using \$RALLY tokens, adding an extra layer of excitement and engagement.

On-chain logic enables secure and transparent management of events and betting systems. The Sui network ensures fairness, with immutable records of bets and outcomes.

ArcaneBattles is a strategic, multiplayer card game inspired by classics like Hearthstone. Players collect, trade, and battle with a variety of magical cards representing spells, creatures, and heroes. By integrating with the Sui blockchain, ArcaneBattles aims to enrich the gameplay experience through decentralized features that promote true ownership, fairness, and a dynamic in-game economy.

Dynamic in-game economy with two closed loop tokens

ArcaneBattles implemented two stable in-game currencies to facilitate various transactions, enhancing player engagement and economic depth.

Closed loop token (CLT)

ArcaneBattles utilizes two Closed Loop Tokens within its ecosystem:

The dual-token system adds depth to the in-game economy, encouraging players to engage in various activities and make strategic decisions about resource allocation.

True ownership and NFT card rental

ArcaneBattles allows players to own their cards as NFTs and provide a rental marketplace for rare or powerful cards.

Each card in ArcaneBattles is represented as an NFT on the Sui blockchain, granting players true ownership of their digital assets. The NFT Rental feature allows players to rent out their rare or high-level cards to others for a fee. This creates a community-driven economy where new or casual players can access powerful cards temporarily, while owners earn passive income from their collections.

Fair and unpredictable gameplay

ArcaneBattles ensures randomness in card draws and in-game events to prevent manipulation and enhance fairness.

By leveraging Sui's on-chain randomness, ArcaneBattles introduces unpredictable elements such as random card draws, critical hit chances, and random effects from certain cards. This randomness is verifiable and secure, preventing manipulation by any party and maintaining fairness across all gameplay aspects.

Enhanced card visualization and dynamic wear mechanism

ArcaneBattles provides rich, dynamic displays of card information and introduces a wear-and-tear mechanic to simulate card degradation over time.

ArcaneBattles uses Sui's Display Standard to offer detailed metadata for each card NFT, including stats, abilities, and artwork. Beyond static information, the game introduces a dynamic display mechanism where cards visually show signs of wear as they are used in battles. Over time, frequently used cards may appear scratched, faded, or have other visual cues indicating wear. After extensive use, cards have a chance to be destroyed entirely.

This wear-and-tear mechanic simulates the experience of physical card games, where rare cards are often kept in pristine condition and used sparingly. It encourages players to strategically decide when to use their valuable cards and adds a layer of depth to the game's economy and strategy.

Players can mitigate or repair wear on their cards by using Mystic Dust to restore them or upgrade them to gold versions, which are more durable and feature enhanced visuals. This system adds a strategic resource management element, as players must balance the benefits of using powerful cards against the potential cost of their degradation.

By integrating these Sui blockchain features, ArcaneBattles not only enhances the gaming experience but also pioneers the next generation of digital card games. The dual-token economy adds complexity and depth to in-game transactions, encouraging strategic decision-making. The wear-and-tear mechanic introduces a novel layer of strategy and realism, as players must consider the longevity of their cards. Together, these features create a rich, engaging, and immersive experience that leverages the full potential of blockchain technology within a gaming context.

Tools

There are a number of tools available in the Sui ecosystem to help you realize your Sui game vision.

Playtron GameOS is a Linux-based operating system that seeks to turn PCs, handhelds, and desktops into dedicated gaming consoles. It supports multiple game stores like Steam and Epic Games, offering a seamless gaming experience across devices such as Steam Deck, ROG Ally, and Lenovo Legion Go.

<https://www.playtron.one/playtron-os>

E4C: Ludus is a cross-platform gaming layer designed to unify Web2 and Web3 gaming experiences on a single platform. Leveraging the Sui blockchain, it offers developers access to dynamic NFTs and zkLogin, facilitating integration of blockchain features into games across various platforms. The native E4C token is its primary currency, enabling in-game purchases and transactions. E4C: Ludus also provides a unified frontend for players to access a range of games, with the aim to enhance user engagement and simplify the gaming experience. The platform is set to launch globally in 2025, with a demo currently available for users to explore.

<https://ludus.ambrus.studio/>

Sui Coins is the utility layer for tokens and NFTs on the Sui network, offering asset management tools that include token swaps, automated dollar-cost averaging, airdrops, an incinerator for deleting assets, zkSend for private transfers, and a merger tool to consolidate small balances. Sui Coins also features an open-source SuiCoins Terminal for integrating crypto swaps across platforms.

<https://www.suicoins.com/> <https://terminal.suicoins.com/>

Beamable is a development platform that helps you integrate live services and backend features into your games. It offers SDKs for both Unity and Unreal Engine, facilitating development and deployment of online game functionalities. You can incorporate features such as player authentication, inventory management, and microservices within the environments of your chosen game engines. Beamable provides support for the Sui blockchain, allowing for the integration of Web3 elements like NFTs and on-chain assets into games. The Beamable SDKs offer tools and sample projects to help you build on the Sui network.

<https://beamable.com/>

Forge is a platform that enables game developers to create custom loyalty programs, rewarding players for engaging in community activities and in-game challenges. Players earn loyalty points by completing actions you define, which can be redeemed for in-game items and digital content. Forge also offers analytics tools to help developers understand their audience and improve monetization strategies.

<https://forge.gg/>

Snag Solutions provides white-label loyalty and marketplace platforms to enhance community engagement and control your digital ecosystems. Their solutions enable you to track and reward user contributions, create customizable marketplaces, and integrate social features like peer-to-peer trading and user profiles. Snag Solutions offers customization options, APIs, and SDKs to align with your brand's identity.

<https://www.snagsolutions.io/>

Venly is a developer platform that aids blockchain integration for businesses through secure digital wallets, tokenization services, and payment solutions. It offers APIs and SDKs for management of digital assets. Venly enables you to create, trade, and manage NFTs, tokens, and payments securely while maintaining full ownership of your assets.

<https://www.venly.io/>

Cycle through the available videos using the thumbnails, then tap or click the video to play.

There aren't real-world implementations for the integrations described in this section. These examples are meant to be a thought exercise to showcase the possibilities for viable game integration on the Sui network.

ShadowQuest is a multiplayer game that combines fantasy with RPG battle mechanics. To enhance the gaming experience with Web3 technologies, ShadowQuest is integrating with Sui offering seamless blockchain interactions to players without compromising the overall gaming experience.

Seamless player onboarding and wallet integration

ShadowQuest wants to onboard players without adding complexity, especially those unfamiliar with Web3. By using [zkLogin](#), users can sign in using social platforms like Google, Facebook, Twitch, and Apple. This automatically creates a Sui wallet linked to their ShadowQuest account, making blockchain interactions seamless.

Simplified transaction handling

Players in ShadowQuest earn or use in-game assets such as NFTs or \$SHADOW tokens. To attract users unfamiliar with Web3, ShadowQuest manages game transactions to avoid Web3 friction, like wallets popping up for signing transactions. By sponsoring user transactions, the friction is minimized because ShadowQuest users do not directly pay for transaction costs and gas fees.

Enoki transactions can be signed without requiring confirmation from the user to approve the transaction.

ShadowQuest uses Enoki Gas Pool to sponsor transactions, covering gas fees for players. This ensures all in-game transactions are seamless and cost-free for players, providing a better user experience.

NFT marketplace and royalty enforcement

ShadowQuest allows players to buy, sell, or trade in-game items, such as weapons, armor, and cosmetics. The items should respect royalties to ensure that creators benefit from each transaction.

Kiosk provides a decentralized marketplace solution, ensuring royalties are enforced on all NFT trades. This helps both the game developers and creators maintain control over secondary sales, ensuring revenue generation throughout the asset's lifecycle.

NFT usage for game access

ShadowQuest uses NFTs as entry tickets for different game modes and events. Players can acquire or earn various Runes, which grant access to specific game challenges or seasonal competitions. These NFTs cannot be traded or transferred to other players.

Soulbound NFTs represent different Runes that are either earned through gameplay or purchased. These NFTs grant exclusive access to matches and seasonal challenges but cannot be traded after bound to a player.

Sui for Speed is a racing game set in the Sui ecosystem. Players pilot customizable vehicles through fantastical terrains, competing in races, time trials, and exploration challenges. By integrating with the Sui network's blockchain technology—including features such as Walrus, dynamic NFTs, SuiNS, and asset tokenization—the game offers players true ownership of their vehicles and in-game assets, along with a vibrant, player-driven economy.

Customizable vehicles with dynamic NFTs

In Sui for Speed, players own racing vehicles represented by dynamic NFTs that you can upgrade and customize with new parts, skins, and abilities. As players progress and win races, their vehicles evolve, reflecting their achievements and style.

Dynamic NFTs on the Sui network allow vehicles to securely update attributes and metadata over time. Every upgrade and customization is recorded on-chain, ensuring each vehicle's uniqueness and authenticity.

SuiNS: Personalized racer profiles and teams

Players can register unique names for their racer profiles and teams using SuiNS, like [speedster@suiforspeed.sui](#) or [dragonracer@suiforspeed.sui](#). This simplifies social interactions, team coordination, and improves the community aspect of the game.

SuiNS provides a decentralized domain naming system, allowing memorable and personalized names on the blockchain.

Tokenized circuits and earnings from circuit usage

Race circuits are tokenized as unique NFTs allowing players to own, design, and vote to enable track customization and drive better changes. When other players race on these circuits, the owners earn \$RALLY tokens as usage fees or royalties. This system incentivizes creativity and allows players to monetize their track designs.

Asset tokenization on the Sui network enables minting of circuits as unique assets with secure ownership. Smart contracts

automatically distribute earnings to circuit owners when their tracks are used, enhancing the game's economy through player-driven content.

Decentralized storage for game data

The game world includes extensive data such as track designs and leaderboards. Leveraging decentralized and efficient storage, combined with asset tokenization, enables true decentralization of these terrains and models.

Walrus offers scalable off-chain storage for large amounts of game data. This ensures high availability and security, protecting against data loss and enhancing player trust.

Competitive events and betting mechanisms

The game hosts regular competitive events and tournaments where players can participate individually or as teams. Additionally, players can place bets on race outcomes using \$RALLY tokens, adding an extra layer of excitement and engagement.

On-chain logic enables secure and transparent management of events and betting systems. The Sui network ensures fairness, with immutable records of bets and outcomes.

ArcaneBattles is a strategic, multiplayer card game inspired by classics like Hearthstone. Players collect, trade, and battle with a variety of magical cards representing spells, creatures, and heroes. By integrating with the Sui blockchain, ArcaneBattles aims to enrich the gameplay experience through decentralized features that promote true ownership, fairness, and a dynamic in-game economy.

Dynamic in-game economy with two closed loop tokens

ArcaneBattles implemented two stable in-game currencies to facilitate various transactions, enhancing player engagement and economic depth.

Closed loop token (CLT)

ArcaneBattles utilizes two Closed Loop Tokens within its ecosystem:

The dual-token system adds depth to the in-game economy, encouraging players to engage in various activities and make strategic decisions about resource allocation.

True ownership and NFT card rental

ArcaneBattles allows players to own their cards as NFTs and provide a rental marketplace for rare or powerful cards.

Each card in ArcaneBattles is represented as an NFT on the Sui blockchain, granting players true ownership of their digital assets. The NFT Rental feature allows players to rent out their rare or high-level cards to others for a fee. This creates a community-driven economy where new or casual players can access powerful cards temporarily, while owners earn passive income from their collections.

Fair and unpredictable gameplay

ArcaneBattles ensures randomness in card draws and in-game events to prevent manipulation and enhance fairness.

By leveraging Sui's on-chain randomness, ArcaneBattles introduces unpredictable elements such as random card draws, critical hit chances, and random effects from certain cards. This randomness is verifiable and secure, preventing manipulation by any party and maintaining fairness across all gameplay aspects.

Enhanced card visualization and dynamic wear mechanism

ArcaneBattles provides rich, dynamic displays of card information and introduces a wear-and-tear mechanic to simulate card degradation over time.

ArcaneBattles uses Sui's Display Standard to offer detailed metadata for each card NFT, including stats, abilities, and artwork. Beyond static information, the game introduces a dynamic display mechanism where cards visually show signs of wear as they are used in battles. Over time, frequently used cards may appear scratched, faded, or have other visual cues indicating wear. After extensive use, cards have a chance to be destroyed entirely.

This wear-and-tear mechanic simulates the experience of physical card games, where rare cards are often kept in pristine condition and used sparingly. It encourages players to strategically decide when to use their valuable cards and adds a layer of depth to the

game's economy and strategy.

Players can mitigate or repair wear on their cards by using Mystic Dust to restore them or upgrade them to gold versions, which are more durable and feature enhanced visuals. This system adds a strategic resource management element, as players must balance the benefits of using powerful cards against the potential cost of their degradation.

By integrating these Sui blockchain features, ArcaneBattles not only enhances the gaming experience but also pioneers the next generation of digital card games. The dual-token economy adds complexity and depth to in-game transactions, encouraging strategic decision-making. The wear-and-tear mechanic introduces a novel layer of strategy and realism, as players must consider the longevity of their cards. Together, these features create a rich, engaging, and immersive experience that leverages the full potential of blockchain technology within a gaming context.

Videos

Cycle through the available videos using the thumbnails, then tap or click the video to play.

There aren't real-world implementations for the integrations described in this section. These examples are meant to be a thought exercise to showcase the possibilities for viable game integration on the Sui network.

ShadowQuest is a multiplayer game that combines fantasy with RPG battle mechanics. To enhance the gaming experience with Web3 technologies, ShadowQuest is integrating with Sui offering seamless blockchain interactions to players without compromising the overall gaming experience.

Seamless player onboarding and wallet integration

ShadowQuest wants to onboard players without adding complexity, especially those unfamiliar with Web3. By using [zkLogin](#), users can sign in using social platforms like Google, Facebook, Twitch, and Apple. This automatically creates a Sui wallet linked to their ShadowQuest account, making blockchain interactions seamless.

Simplified transaction handling

Players in ShadowQuest earn or use in-game assets such as NFTs or \$SHADOW tokens. To attract users unfamiliar with Web3, ShadowQuest manages game transactions to avoid Web3 friction, like wallets popping up for signing transactions. By sponsoring user transactions, the friction is minimized because ShadowQuest users do not directly pay for transaction costs and gas fees.

Enoki transactions can be signed without requiring confirmation from the user to approve the transaction.

ShadowQuest uses Enoki Gas Pool to sponsor transactions, covering gas fees for players. This ensures all in-game transactions are seamless and cost-free for players, providing a better user experience.

NFT marketplace and royalty enforcement

ShadowQuest allows players to buy, sell, or trade in-game items, such as weapons, armor, and cosmetics. The items should respect royalties to ensure that creators benefit from each transaction.

Kiosk provides a decentralized marketplace solution, ensuring royalties are enforced on all NFT trades. This helps both the game developers and creators maintain control over secondary sales, ensuring revenue generation throughout the asset's lifecycle.

NFT usage for game access

ShadowQuest uses NFTs as entry tickets for different game modes and events. Players can acquire or earn various Runes, which grant access to specific game challenges or seasonal competitions. These NFTs cannot be traded or transferred to other players.

Soulbound NFTs represent different Runes that are either earned through gameplay or purchased. These NFTs grant exclusive access to matches and seasonal challenges but cannot be traded after bound to a player.

Sui for Speed is a racing game set in the Sui ecosystem. Players pilot customizable vehicles through fantastical terrains, competing in races, time trials, and exploration challenges. By integrating with the Sui network's blockchain technology—including features such as Walrus, dynamic NFTs, SuINS, and asset tokenization—the game offers players true ownership of their vehicles and in-game assets, along with a vibrant, player-driven economy.

Customizable vehicles with dynamic NFTs

In Sui for Speed, players own racing vehicles represented by dynamic NFTs that you can upgrade and customize with new parts,

skins, and abilities. As players progress and win races, their vehicles evolve, reflecting their achievements and style.

Dynamic NFTs on the Sui network allow vehicles to securely update attributes and metadata over time. Every upgrade and customization is recorded on-chain, ensuring each vehicle's uniqueness and authenticity.

SuiNS: Personalized racer profiles and teams

Players can register unique names for their racer profiles and teams using SuiNS, like speedster@suiforspeed.sui or dragonracer@suiforspeed.sui. This simplifies social interactions, team coordination, and improves the community aspect of the game.

SuiNS provides a decentralized domain naming system, allowing memorable and personalized names on the blockchain.

Tokenized circuits and earnings from circuit usage

Race circuits are tokenized as unique NFTs allowing players to own, design, and vote to enable track customization and drive better changes. When other players race on these circuits, the owners earn \$RALLY tokens as usage fees or royalties. This system incentivizes creativity and allows players to monetize their track designs.

Asset tokenization on the Sui network enables minting of circuits as unique assets with secure ownership. Smart contracts automatically distribute earnings to circuit owners when their tracks are used, enhancing the game's economy through player-driven content.

Decentralized storage for game data

The game world includes extensive data such as track designs and leaderboards. Leveraging decentralized and efficient storage, combined with asset tokenization, enables true decentralization of these terrains and models.

Walrus offers scalable off-chain storage for large amounts of game data. This ensures high availability and security, protecting against data loss and enhancing player trust.

Competitive events and betting mechanisms

The game hosts regular competitive events and tournaments where players can participate individually or as teams. Additionally, players can place bets on race outcomes using \$RALLY tokens, adding an extra layer of excitement and engagement.

On-chain logic enables secure and transparent management of events and betting systems. The Sui network ensures fairness, with immutable records of bets and outcomes.

ArcaneBattles is a strategic, multiplayer card game inspired by classics like Hearthstone. Players collect, trade, and battle with a variety of magical cards representing spells, creatures, and heroes. By integrating with the Sui blockchain, ArcaneBattles aims to enrich the gameplay experience through decentralized features that promote true ownership, fairness, and a dynamic in-game economy.

Dynamic in-game economy with two closed loop tokens

ArcaneBattles implemented two stable in-game currencies to facilitate various transactions, enhancing player engagement and economic depth.

Closed loop token (CLT)

ArcaneBattles utilizes two Closed Loop Tokens within its ecosystem:

The dual-token system adds depth to the in-game economy, encouraging players to engage in various activities and make strategic decisions about resource allocation.

True ownership and NFT card rental

ArcaneBattles allows players to own their cards as NFTs and provide a rental marketplace for rare or powerful cards.

Each card in ArcaneBattles is represented as an NFT on the Sui blockchain, granting players true ownership of their digital assets. The NFT Rental feature allows players to rent out their rare or high-level cards to others for a fee. This creates a community-driven economy where new or casual players can access powerful cards temporarily, while owners earn passive income from their collections.

Fair and unpredictable gameplay

ArcaneBattles ensures randomness in card draws and in-game events to prevent manipulation and enhance fairness.

By leveraging Sui's on-chain randomness, ArcaneBattles introduces unpredictable elements such as random card draws, critical hit chances, and random effects from certain cards. This randomness is verifiable and secure, preventing manipulation by any party and maintaining fairness across all gameplay aspects.

Enhanced card visualization and dynamic wear mechanism

ArcaneBattles provides rich, dynamic displays of card information and introduces a wear-and-tear mechanic to simulate card degradation over time.

ArcaneBattles uses Sui's Display Standard to offer detailed metadata for each card NFT, including stats, abilities, and artwork. Beyond static information, the game introduces a dynamic display mechanism where cards visually show signs of wear as they are used in battles. Over time, frequently used cards may appear scratched, faded, or have other visual cues indicating wear. After extensive use, cards have a chance to be destroyed entirely.

This wear-and-tear mechanic simulates the experience of physical card games, where rare cards are often kept in pristine condition and used sparingly. It encourages players to strategically decide when to use their valuable cards and adds a layer of depth to the game's economy and strategy.

Players can mitigate or repair wear on their cards by using Mystic Dust to restore them or upgrade them to gold versions, which are more durable and feature enhanced visuals. This system adds a strategic resource management element, as players must balance the benefits of using powerful cards against the potential cost of their degradation.

By integrating these Sui blockchain features, ArcaneBattles not only enhances the gaming experience but also pioneers the next generation of digital card games. The dual-token economy adds complexity and depth to in-game transactions, encouraging strategic decision-making. The wear-and-tear mechanic introduces a novel layer of strategy and realism, as players must consider the longevity of their cards. Together, these features create a rich, engaging, and immersive experience that leverages the full potential of blockchain technology within a gaming context.

Example integrations

There aren't real-world implementations for the integrations described in this section. These examples are meant to be a thought exercise to showcase the possibilities for viable game integration on the Sui network.

ShadowQuest is a multiplayer game that combines fantasy with RPG battle mechanics. To enhance the gaming experience with Web3 technologies, ShadowQuest is integrating with Sui offering seamless blockchain interactions to players without compromising the overall gaming experience.

Seamless player onboarding and wallet integration

ShadowQuest wants to onboard players without adding complexity, especially those unfamiliar with Web3. By using [zkLogin](#), users can sign in using social platforms like Google, Facebook, Twitch, and Apple. This automatically creates a Sui wallet linked to their ShadowQuest account, making blockchain interactions seamless.

Simplified transaction handling

Players in ShadowQuest earn or use in-game assets such as NFTs or \$SHADOW tokens. To attract users unfamiliar with Web3, ShadowQuest manages game transactions to avoid Web3 friction, like wallets popping up for signing transactions. By sponsoring user transactions, the friction is minimized because ShadowQuest users do not directly pay for transaction costs and gas fees.

Enoki transactions can be signed without requiring confirmation from the user to approve the transaction.

ShadowQuest uses Enoki Gas Pool to sponsor transactions, covering gas fees for players. This ensures all in-game transactions are seamless and cost-free for players, providing a better user experience.

NFT marketplace and royalty enforcement

ShadowQuest allows players to buy, sell, or trade in-game items, such as weapons, armor, and cosmetics. The items should respect royalties to ensure that creators benefit from each transaction.

Kiosk provides a decentralized marketplace solution, ensuring royalties are enforced on all NFT trades. This helps both the game

developers and creators maintain control over secondary sales, ensuring revenue generation throughout the asset's lifecycle.

NFT usage for game access

ShadowQuest uses NFTs as entry tickets for different game modes and events. Players can acquire or earn various Runes, which grant access to specific game challenges or seasonal competitions. These NFTs cannot be traded or transferred to other players.

Soulbound NFTs represent different Runes that are either earned through gameplay or purchased. These NFTs grant exclusive access to matches and seasonal challenges but cannot be traded after bound to a player.

Sui for Speed is a racing game set in the Sui ecosystem. Players pilot customizable vehicles through fantastical terrains, competing in races, time trials, and exploration challenges. By integrating with the Sui network's blockchain technology—including features such as Walrus, dynamic NFTs, SuiNS, and asset tokenization—the game offers players true ownership of their vehicles and in-game assets, along with a vibrant, player-driven economy.

Customizable vehicles with dynamic NFTs

In Sui for Speed, players own racing vehicles represented by dynamic NFTs that you can upgrade and customize with new parts, skins, and abilities. As players progress and win races, their vehicles evolve, reflecting their achievements and style.

Dynamic NFTs on the Sui network allow vehicles to securely update attributes and metadata over time. Every upgrade and customization is recorded on-chain, ensuring each vehicle's uniqueness and authenticity.

SuiNS: Personalized racer profiles and teams

Players can register unique names for their racer profiles and teams using SuiNS, like speedster@suiforspeed.sui or dragonracer@suiforspeed.sui. This simplifies social interactions, team coordination, and improves the community aspect of the game.

SuiNS provides a decentralized domain naming system, allowing memorable and personalized names on the blockchain.

Tokenized circuits and earnings from circuit usage

Race circuits are tokenized as unique NFTs allowing players to own, design, and vote to enable track customization and drive better changes. When other players race on these circuits, the owners earn \$RALLY tokens as usage fees or royalties. This system incentivizes creativity and allows players to monetize their track designs.

Asset tokenization on the Sui network enables minting of circuits as unique assets with secure ownership. Smart contracts automatically distribute earnings to circuit owners when their tracks are used, enhancing the game's economy through player-driven content.

Decentralized storage for game data

The game world includes extensive data such as track designs and leaderboards. Leveraging decentralized and efficient storage, combined with asset tokenization, enables true decentralization of these terrains and models.

Walrus offers scalable off-chain storage for large amounts of game data. This ensures high availability and security, protecting against data loss and enhancing player trust.

Competitive events and betting mechanisms

The game hosts regular competitive events and tournaments where players can participate individually or as teams. Additionally, players can place bets on race outcomes using \$RALLY tokens, adding an extra layer of excitement and engagement.

On-chain logic enables secure and transparent management of events and betting systems. The Sui network ensures fairness, with immutable records of bets and outcomes.

ArcaneBattles is a strategic, multiplayer card game inspired by classics like Hearthstone. Players collect, trade, and battle with a variety of magical cards representing spells, creatures, and heroes. By integrating with the Sui blockchain, ArcaneBattles aims to enrich the gameplay experience through decentralized features that promote true ownership, fairness, and a dynamic in-game economy.

Dynamic in-game economy with two closed loop tokens

ArcaneBattles implemented two stable in-game currencies to facilitate various transactions, enhancing player engagement and

economic depth.

Closed loop token (CLT)

ArcaneBattles utilizes two Closed Loop Tokens within its ecosystem:

The dual-token system adds depth to the in-game economy, encouraging players to engage in various activities and make strategic decisions about resource allocation.

True ownership and NFT card rental

ArcaneBattles allows players to own their cards as NFTs and provide a rental marketplace for rare or powerful cards.

Each card in ArcaneBattles is represented as an NFT on the Sui blockchain, granting players true ownership of their digital assets. The NFT Rental feature allows players to rent out their rare or high-level cards to others for a fee. This creates a community-driven economy where new or casual players can access powerful cards temporarily, while owners earn passive income from their collections.

Fair and unpredictable gameplay

ArcaneBattles ensures randomness in card draws and in-game events to prevent manipulation and enhance fairness.

By leveraging Sui's on-chain randomness, ArcaneBattles introduces unpredictable elements such as random card draws, critical hit chances, and random effects from certain cards. This randomness is verifiable and secure, preventing manipulation by any party and maintaining fairness across all gameplay aspects.

Enhanced card visualization and dynamic wear mechanism

ArcaneBattles provides rich, dynamic displays of card information and introduces a wear-and-tear mechanic to simulate card degradation over time.

ArcaneBattles uses Sui's Display Standard to offer detailed metadata for each card NFT, including stats, abilities, and artwork. Beyond static information, the game introduces a dynamic display mechanism where cards visually show signs of wear as they are used in battles. Over time, frequently used cards may appear scratched, faded, or have other visual cues indicating wear. After extensive use, cards have a chance to be destroyed entirely.

This wear-and-tear mechanic simulates the experience of physical card games, where rare cards are often kept in pristine condition and used sparingly. It encourages players to strategically decide when to use their valuable cards and adds a layer of depth to the game's economy and strategy.

Players can mitigate or repair wear on their cards by using Mystic Dust to restore them or upgrade them to gold versions, which are more durable and feature enhanced visuals. This system adds a strategic resource management element, as players must balance the benefits of using powerful cards against the potential cost of their degradation.

By integrating these Sui blockchain features, ArcaneBattles not only enhances the gaming experience but also pioneers the next generation of digital card games. The dual-token economy adds complexity and depth to in-game transactions, encouraging strategic decision-making. The wear-and-tear mechanic introduces a novel layer of strategy and realism, as players must consider the longevity of their cards. Together, these features create a rich, engaging, and immersive experience that leverages the full potential of blockchain technology within a gaming context.

Related links