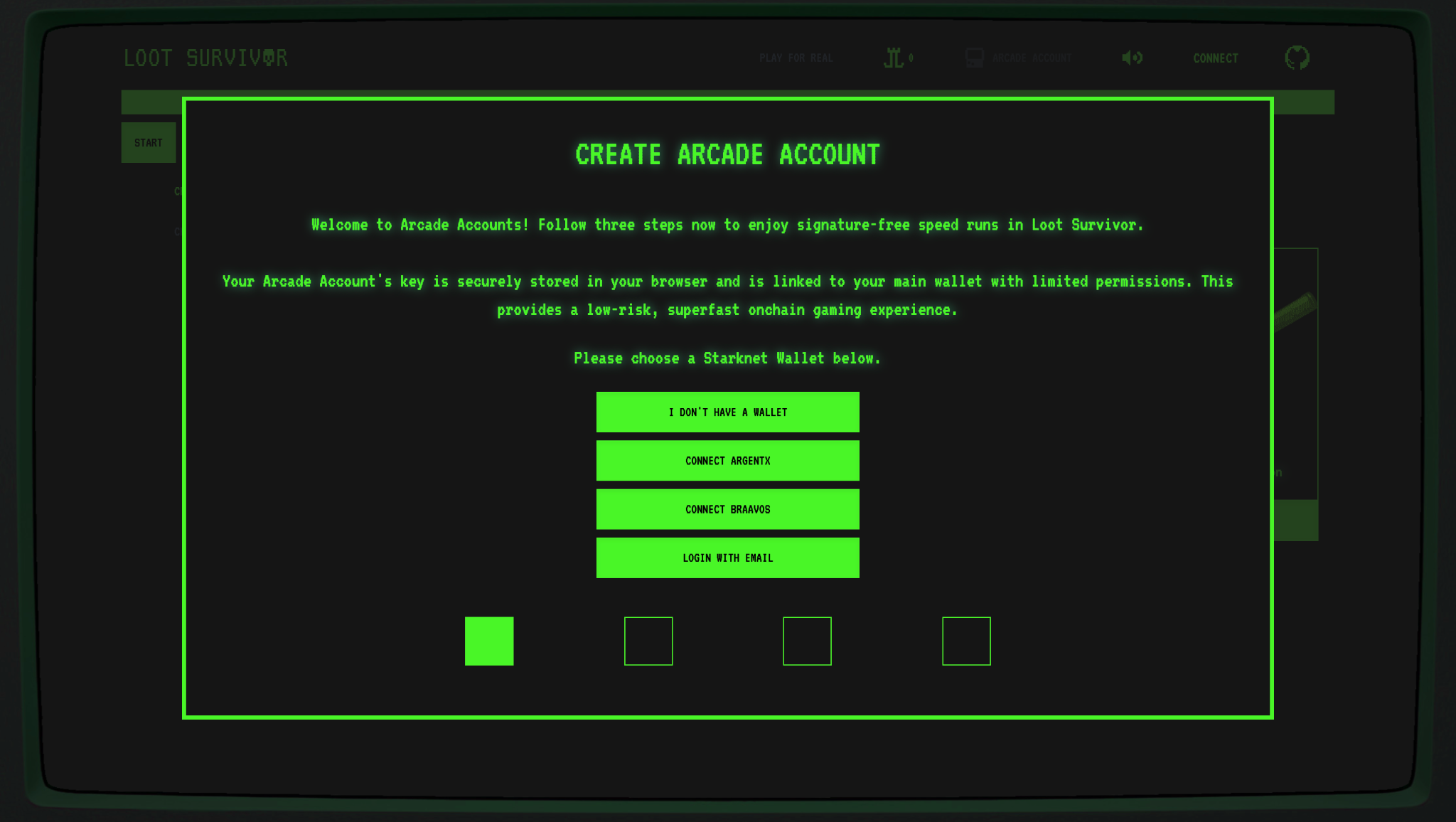
**Part 2. Building complex, on-chain gaming with StarkNet**

In an [earlier post](https://medium.com/starkware/let-the-games-begin-redefining-on-chain-gaming-with-starknet-1e6e6d6c37fa) on the status of on-chain gaming with Starknet, we were introduced to some innovative projects exploring the limits of game development on the Ethereum L2 - [Realms.Eternum](https://medium.com/starkware/let-the-games-begin-redefining-on-chain-gaming-with-starknet-1e6e6d6c37fa), [Loot Survivor](https://loot-survivor.vercel.app/), and [Influence](https://www.influenceth.io/). Cairo powers these first truly eternal on-chain games – a STARK-proof-based Rust-inspired language for writing smart contracts on Starknet. The developers behind Realms and Influence share with us their review of why it would be difficult to build a similar product in a decentralized manner on a different platform and what exciting features are added in the first release of the respective projects!

**Loot Survivor**

Loot Survivor is a session-based game where the player is granted only a life to find and improve weapons, fight beasts, and earn rewards for their skill/luck. Players are challenged to fight over 70 beasts, 60 obstacles, and dozens of weapons, with many ways to improve them upon winning a battle.

Loot Survivor is a pay-per-single-session strategy game such that the player has to pay in LORDs tokens to play a single round. If the player's hero dies, the player can easily start a new in-game session without having any large investments into hero NFTs and further speculation. Every round of the game is decentralized so that bots and cheaters do not get any advantage in playing over naive human players.



When players log onto the home page, they are prompted with a screen to log in. They can connect to their Argent or Braavos wallet or even log in with an email - powered by Argent, generating a wallet with the registered email address behind the scenes. Easy accessibility to web3 games has been a key requirement since the very beginning, and the unique access by Argent via email addresses is a testament to the same. Once logged in, the player has to choose a single weapon, and his game journey begins.

Behind the intricate plot of the game and many possibilities for the direction in which the game goes lies the complex architecture of the game. Let us now explore what new ideas and concepts the team behind Loot Survivor uses to create the first truly autonomous on-chain game.

**Arcade accounts**. When the player begins the game, they are prompted to create an arcade account for the session, similar to a "burner" account. The "use-and-throw" account thus created runs on the client side (in the browser). Account abstraction (discussed in detail [here](https://medium.com/starknet-edu/account-abstraction-on-starknet-part-i-2ff84c6a3c30)) offered by Cairo makes arcade accounts feasible for development. The advantage of having arcade accounts is that certain types of transactions can be sent on-chain without asking the user for a signature every time an action is taken. For someone who has transacted with blockchain wallets before, pressing "Sign" and "Send" on the wallet widget whenever an action is taken is an extra deterrent from anyone else stealing your funds. However, in gaming, many players' choices are optional, so bypassing the repeated signature step offers added convenience rather than posing a security risk.

**Optimistic client rendering**. On the path of bringing traditional games on-chain, the abstraction of blockchain constraints simplifies adoption. A major blocker of on-chain gaming is that once a user sends a transaction, they also have to wait for the transaction to be accepted on the network before proceeding. Due to the previously mentioned reason for in-game actions not always being of critical importance, this step can be skipped for a better user experience. This is exactly what Loot Survivor achieves by implementing optimistic updates on the client side. When a user completes an action, the transaction is simulated on a local client implementation, and if the transaction does not fail, the game optimistically goes on.

**Single storage contracts**. The Loot Survivor game is a nascent example of a completely on-chain game such that no part of the game state is stored off-chain. In Loot Survivor, a complete player-related state (such as health, XP, weapons, and attachments) is stored in a 252-bit Cairo [felt](https://book.starknet.io/chapter_9/felts.html). As a result, any action that changes the player state results in a single storage update. For those unfamiliar with the general architecture of layer two rollups like Starknet, the benefit of a single storage update is that multiple state changes can be aggregated into a single storage update. Since the transaction's fee only [depends on the number of unique storage updates](https://docs.starknet.io/documentation/architecture_and_concepts/Network_Architecture/fee-mechanism/) (if the same storage cell is updated multiple times within the transaction, the fee remains that of a single update), associated gas fees of in-game actions are reduced drastically.

**Entropy-based bot resilience**. On-chain games oftentimes have financial incentives for the players to win the game (in most cases – token rewards). The existence of financial incentives also may encourage users to try and exploit the system. To counteract this, Loot Survivor uses global game entropy and the player's unique entropy to decide the game's plot. The global entropy is a variable that anyone can adjust by invoking the external public function called rotate\_game\_entropy(). It's assumed that the community around the game starts running bots that periodically call the rotate\_game\_entropy() function to ensure no one is cheating the system. Every adventurer is given a distinct hash when starting a game to guarantee diverse outcomes for players despite a common game entropy. Whenever an action is taken, the global entropy is combined with the player entropy to generate a distinctive outcome every time the game is played.

[INSERT CHART DIAGRAM FOR ENTROPY]

Loot Survivor is a first-of-a-kind strategic game already released on testnet and is ready to be experimented with [here](https://goerli-survivor.realms.world/). Beyond the opportunity to be among the early testers for one of the inaugural fully on-chain games, players are incentivized to participate. A percentage of the tokens inserted into the Loot Survivor arcade machine are distributed to the top three scorers on the global leaderboard as a reward.

**Influence**

Influence is a player-driven space strategy massively multiplayer online (MMO) game that revolves around the plot of space colonization many years into the future. War and global warming on Earth forced humans to leave Earth in pursuit of life-sustaining resources.

The game has three stages: Exploitation, Discovery, and Conflict. The [latest game update](https://influenceth.substack.com/p/flight-path-to-game-launch?utm_source=profile&utm_medium=reader2), Limited Mainnet Release, allows players to explore the vast space of asteroids that will be made available in the first stage of the game - Exploitation. In the current release, players can bridge their assets from Ethereum mainnet to Starknet, name their asteroids, form a crew, and complete long-range asteroid scans (which release generic information about the asteroid, such as what resources are available on the asteroid, but not a detailed summary of the amounts and location of these resources).



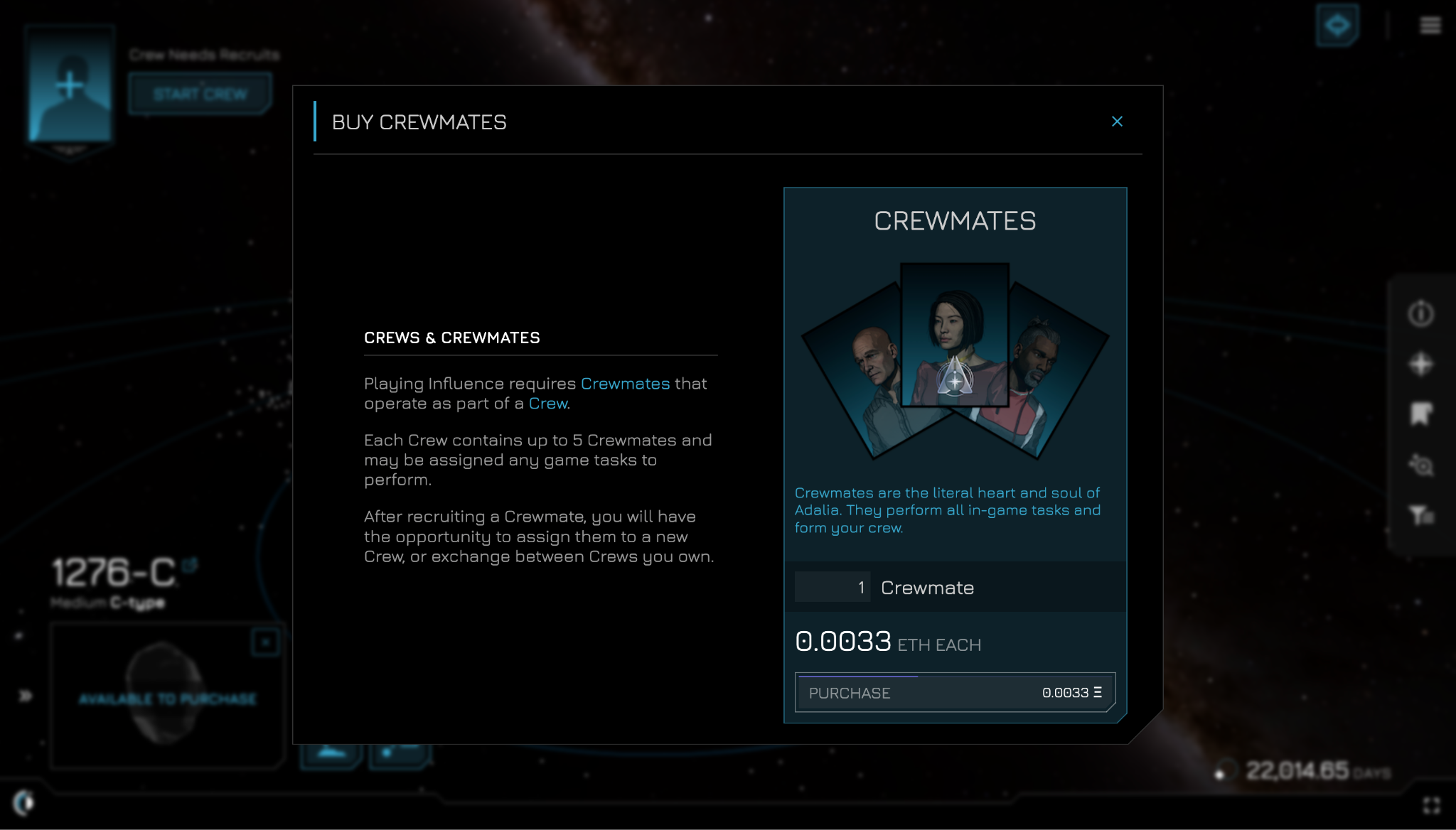
Since the testnet release of the pre-release is in the final stages of production, Cairo's crucial role in development started being noticed. Most of the main [features](https://wiki.influenceth.io/en/docs/press-kit) that an on-chain game must have become much more efficient to build when using Starknet and Cairo.

**Account Abstraction**. The Influence team is working closely with Argent to deliver account abstraction benefits using session keys. Session keys enable temporary sessions that can be used to send transactions on behalf of the user without asking for a signature. The session keys allow only specific functions to be called on behalf of the user; however, they allow multiple transactions to be sent without troubling the player repeatedly. Below is an example code of how session keys are created and what parameters are to be taken care of

| import { Signer, ec } from "starknet"  // gets signer with random private key you need to store if you want to reuse the session const sessionSigner = new Signer()  const requestSession: RequestSession = {  key: await sessionSigner.getPublicKey(),  expires: Math.floor((Date.now() + 1000 \* 60 \* 60 \* 24) / 1000), // 1 day in seconds  policies: [  {  contractAddress: "0x...",  selector: "doAction"  }  ] } |
| --- |

([Source](https://docs.argent.xyz/starknet/session-keys-experimental-feature))

**Action log**. Whenever players submit an action in the game, the action is immediately reflected in the action log of the game. This provides immediate feedback to players as they engage with the game and make decisions. Thus, players are shielded from the intricate details of blockchain technology, where a transaction can exist in various stages like sent, pending, confirmed on L2, or approved on L1. In the new age of on-chain gaming, it is optional for players to grasp these technicalities to enjoy the game.



**Autonomous Worlds**. Both Influence and Loot Survivor are building games compatible with autonomous worlds. In an autonomous world game, the code must be written in such a way that even if the original developers behind the game stop releasing new features, it should be possible for anyone else to build on top of the smart contracts that the game uses. This means that multiple user interfaces could be made available for the same game (and the same in-game assets/resources). To incentivize teams to develop frontend interfaces of their own, the developers of these games generally include some incentive for the interface developer to build on top of existing smart contracts. This could mean that a specific percentage of in-game SWAY tokens (or LORDS in the case of Loot Survivor) are transferred to a wallet specified by the UI developer, thus funding their research and development.

**The frontier of on-chain gaming**

The frontier of on-chain gaming is vast and unexplored. Teams like Influence and Realms are constantly iterating over new paradigms to create truly eternal on-chain games such that the legacy of the games can go on even without the original developers steering the ship's wheel. Nevertheless, it's no wonder that tools like Cairo and Starknet have emerged as the key tools in shaping this sustainable gaming future.