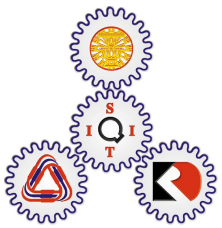


Capstone Project

Smart Contract and Decentralized Application

Gaming Reward Systems

Presented by

Sirimata Suthamma 6322770700

Napat Soodla 6322772318

Napon Jirapakatawee 6322772359

Lecturer: Dr. Watthanasak Jeamwatthanachai

Blockchain Development

CSS484 Section 1

Sirindhorn International Institute of Technology

Thammasat University

18 December 2023

Table of Contents

Introduction **3**
- Project Overview 3
- Objective 3

Daily Login Reward System **4**
- Design 4
- Progressive Reward Structure 4

System Architecture **5**
- Decentralized Architecture 5
- Smart Contract Backend System 5
- Frontend System 5
  - Frontend Framework 5
  - User Interface Design 5
  - Functionality 5

Security Measures **6**
- Smart Contract Security 6
- Libraries for Security 6

Conclusion **7**

Introduction

Project Overview

In the dynamic landscape of gaming and digital entertainment, the decentralized gaming reward system project emerges as a groundbreaking initiative designed to redefine user engagement through the integration of blockchain technology. Traditional gaming ecosystems have often been characterized by centralized structures, where user identity, rewards, and interactions are mediated through proprietary systems. This project seeks to challenge this model by embracing the principles of decentralization, transparency, and user empowerment.

As the gaming industry continues to evolve, the demand for more inclusive, secure, and engaging experiences has grown. Blockchain technology, with its inherent features of immutability and transparency, has emerged as a transformative force capable of reshaping the way users interact with gaming platforms. This project acknowledges the limitations of traditional centralized gaming models and responds with a decentralized approach that leverages the Ethereum blockchain.

The decentralized gaming reward system project introduces an innovative approach to incentivizing user engagement in gaming through the integration of blockchain technology. This system represents a paradigm shift from traditional centralized gaming models, embracing the principles of decentralization, transparency, and user empowerment.

Objective

The primary objectives of this project were to create a secure and transparent gaming environment while enhancing user engagement through a decentralized reward system. By shifting from traditional username and password authentication to Ethereum wallet addresses, the project aims to establish a decentralized identity system, placing control directly in the hands of users. The integration of smart contracts further ensures trustless and tamper-resistant reward distribution, providing users with a fair and transparent gaming experience.

Daily Login Reward System

Design

To give this project more specification, we decided to make a daily login as a reward system. It is an in-game system that will award the players with in-game money for 7 days that each player logged into the game. However, giving the reward directly will be too abstract and hard to visualize. Therefore, we also implemented the authentication system and broad structure of in-game currency to help visualize the developer better on how it will looks like when implementing the feature in their own game.

Progressive Reward Structure

A progressive reward structure is designed to motivate players by providing increasingly valuable rewards as they engage with the platform, in this case is game, consistently over time. The longer a user remains active, the more significant the rewards they receive. This structure encourages user retention and loyalty.

The following are key features of a progressive reward structure:

- Incremental Value: Each consecutive interaction or login results in a reward with incrementally higher value. This could be in the form of in-game currency, virtual items, or other benefits.
- Streak Multipliers: Introduce multipliers that amplify the rewards for users who maintain a streak of consecutive logins. The longer the streak, the more substantial the multiplier, providing an additional incentive for daily engagement.
- Special Milestones: Incorporate special milestones that trigger bonus rewards. For example, a user who logs in for seven consecutive days might receive a bonus reward to celebrate the weekly achievement.
- Seasonal or Event-Based Rewards: Implement special rewards during seasons or events to keep the reward structure dynamic. Limited-time rewards can create a sense of urgency and excitement.

Daily login rewards system is design according to the incremental value feature where a progressive structure is applied by designing a schedule where users receive increasing amounts of in-game currency for each consecutive day they log in. Additionally, reaching milestones like a one-week streak could trigger bonus rewards or unlock special features.

System Architecture

Decentralized Architecture

Blockchain technology serves as the backbone of this project, providing a decentralized and secure infrastructure. Smart contracts, deployed on the Ethereum blockchain, govern the daily login reward system, eliminating the need for centralized authorities in reward distribution. The transparency and immutability inherent in blockchain contribute to a trustworthy environment where users can verify their interactions independently.

This project is grounded in the belief that decentralized applications (DApps) have the potential to redefine user relationships with gaming platforms. By offering users greater control over their identities, assets, and rewards, the project aims to enhance the overall gaming experience. The rationale extends beyond technological innovation to address the evolving expectations of a user base increasingly inclined towards decentralized and community-driven ecosystems.

Looking at our application, we have 2 sections that are decentralized: Authentication and Game Currency System. Authentication serves as a decentralized authentication mechanism, ensuring that user access is controlled by the rules encoded in the smart contract, while Game Currency System decentralizes the handling of in-game currency transactions. The rules for depositing, withdrawing, and transferring funds are encoded in the smart contract

For Deployment, Ganache is used as the local network for testing, and truffle is used as the main compiler. MetaMask is used as a wallet provider for the frontend.

Smart Contract Backend System

-detail about code- *Provide insights into the backend system that supports the decentralized gaming reward infrastructure. Detail the backend components and their roles in the overall architecture.*

Overall, the system's architecture leverages Ethereum smart contracts for the distribution of rewards, ensuring transparency and fairness. User authentication is facilitated through Ethereum wallet addresses, aligning with the core principles of decentralization. Gamification elements, such as daily login rewards, contribute to a vibrant and engaging user experience.

There are 3 smart contracts running as the base of the system: Authentication, GameCurrency and GameReward.

Authentication is designed to be the access control of the user. Users are required to fill their password upon entering the game. There is a requirement of username when register

in the first time. All in all, this contract is used to authenticate users before take action in the game.

To understand how the game works, Game currency is implemented to show how the framework will acutally process in the game. This included deposit function, withdraw function, transfer function, etc.

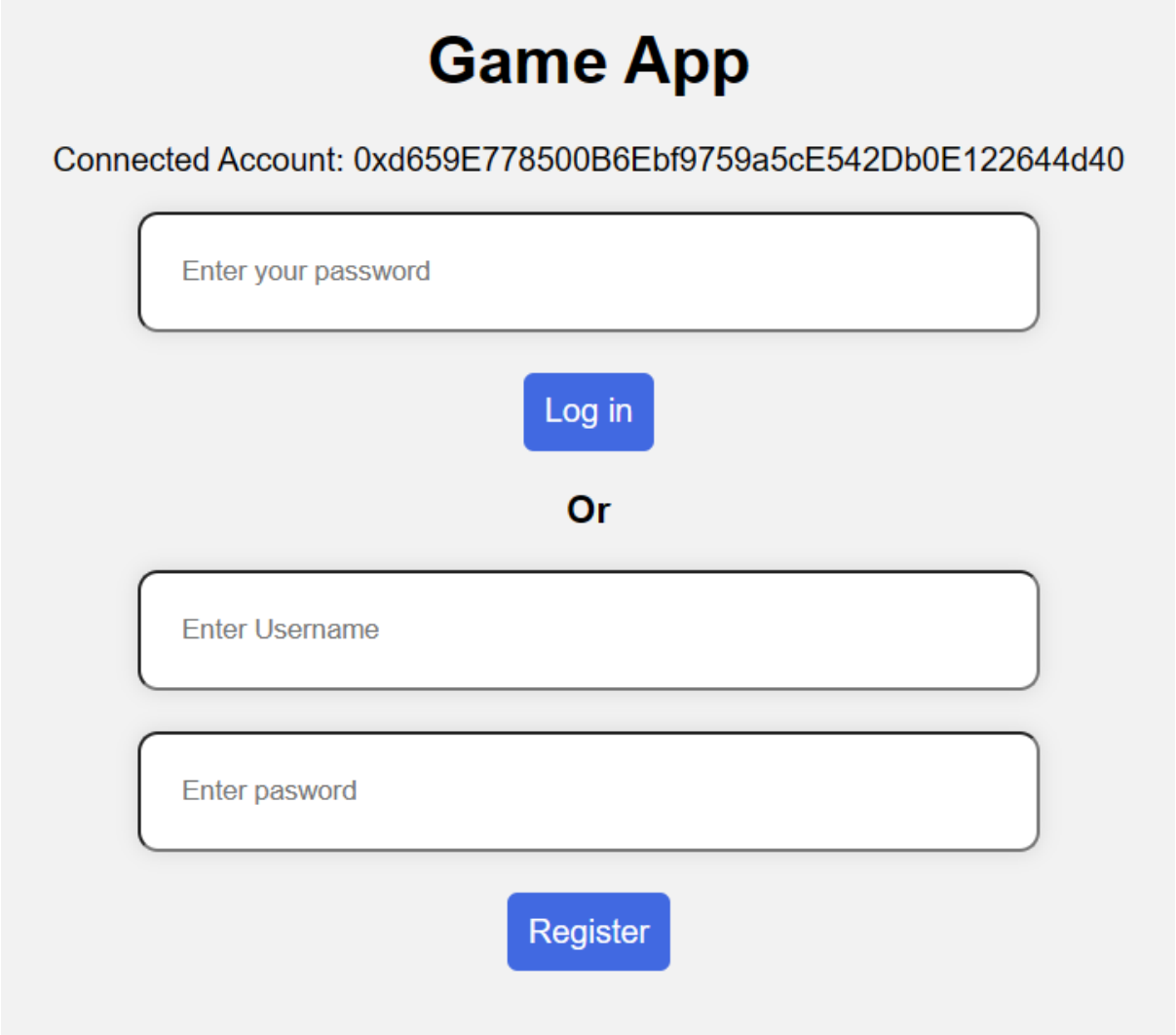
GameReward take the role as the main script of this project to distribute game rewards to player, daily rewards to be specific. The role of the script is to keep track of the amount given per reward and track the time to check if the next reward is available. It is also be the gateway of other contracts; all functions of the written smart contract will pass through this script.

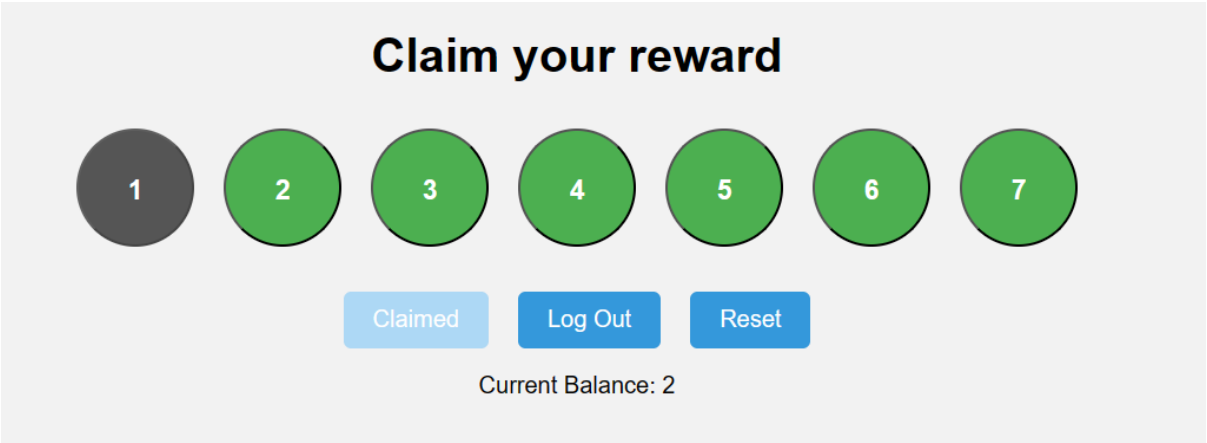
Frontend System

Frontend Framework

Framework that is used in the project is React framework. It is preferable for this project because React enables the creation of components that can manage their state and efficiently update the UI when needed. Furthermore, since React.js supported NodeJS, implementation and adatpation of Web3 and Smart Contract is much more easier compared to legacy HTML or other frameworks.

User Interface Design





Functionality

-optional, Explain the implemented functionality that allows users to view and claim daily rewards seamlessly.-

The images above show the interfaces of our application. There are 2 pages: the login page and the reward claiming page. The login page have 2 sections: the login area (contains

only password) and register area (contain both username and password). The account is already bound with the wallet address, so only password is required for login. For the reward claiming page, the green circle with the number of days on it displays how many days the reward is claimed. If the player didn’t claim the reward yet, claim button will be available and click it to claim reward. In case everything finished, click “Log out”. Reset button is not supposed to be deployed, as it resets the daily reward. It is implemented for testing purpose.

Security Measures

Smart Contract Security

Explain the implementation of best practices for smart contract security, ensuring the integrity and safety of the reward system.

Authentication

- Password Hashing: The contract hashes passwords using the keccak256 function before storing them.
- Login State Management: There is an “isLoggedIn” flag used to track the login state of users. This prevents unauthorized access to certain functions and ensures that actions like claiming daily rewards can only be performed by authenticated users.
- Input Validation: The require statement is implemented to ensures that the user must be logged in before logging out.

Game Currency

- Input Validation: The require statements ensure that users cannot transfer or withdraw more tokens than they have in their balance.

Game Reward

- Dependency Injection: The addresses of the Authentication and GameCurrency contracts are passed during deployment, ensuring that the GameReward contract interacts with trusted instances of these contracts.
- Modifiers for Access Control: The onlyLoggedIn and onlyAfterRewardInterval modifiers provide access control to functions like claiming daily rewards. This ensures that only logged-in users can claim rewards, and rewards can only be claimed after a certain time interval.
- Fallback Function: The fallback function rejects unexpected Ether transfers, preventing accidental loss of funds.

Conclusion

In conclusion, the decentralized gaming reward system project has successfully achieved its goal of creating a more engaging and user-friendly gaming experience. The transition to Ethereum wallet addresses for user authentication has improved security and empowered players with decentralized identity. This shift has resonated positively with the gaming community, resulting in increased daily interactions and a favorable response.

The incorporation of blockchain technology, particularly utilizing the Ethereum blockchain, has ensure transparency and decentralization. Smart contracts govern the reward distribution, ensuring automated and tamper-resistant processes. This commitment to decentralization aligns with the fundamental principles of blockchain technology, fostering a trustless and open environment within the gaming platform.

Security has been a top priority throughout the project's development. The implementation of robust security measures, including adherence to smart contract best practices, ensures the protection of user accounts and data.