International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 09 Issue: 04 | Apr 2022 www.irjet.net p-ISSN: 2395-0072

Real-life Application of a Decentralized System

PARTHIV JANI[1], ASHMIT SRIVASTAVA[2], ANUPAMA JAWALE[3]

[1,2]UG Student, Dept. Of Information Technology, Narsee Monjee College, Mumbai, Maharashtra, India
[3]Assistant Professor, Dept. Of Information Technology, Narsee Monjee College, Mumbai, Maharashtra, India

---

Abstract - *Blockchain is the technology that can lead to significant changes in our current technology and will have a great impact on the next few decades. It can change the way we perceive technological processes, and can also transform our economy. The majority of current Blockchain Technology research is focused on its application for cryptocurrencies such as Bitcoin and Ethereum, with only a few studies looking at the use of Blockchain Technology in other settings or sectors. Blockchain Technology is more than just cryptocurrency, it can have several applications in governmental activities, finance and banking industries, accounting, and many business processes. Therefore, this paper aims to explain how Blockchain technology works, demonstrate its real-world application by looking into the current lottery picking system, and propose an E-lottery picking system using smart contracts in the Blockchain ecosystem, as well as how it can be deployed on the Ethereum Network.*

Key Words: **Blockchain, Decentralization, Smart Contract, Token, Solidity**

1. SYSTEM FUNDAMENTALS

1.1 Blockchain Technology

Blockchain, also known as Distributed Ledger Technology (DLT), makes the history of any digital asset immutable and transparent by utilizing decentralization and cryptographic hashing.

The three key components of blockchain are Blocks, Nodes, and Miners.

1.1.1 Blocks

Blockchain is essentially a chain of blocks, each of which contains the three pieces of information listed below:

-The data stored in the block.

-The nonce is a 32-bit whole number. When a new block iscreated, a completely random nonce is generated, which then generates a block header hash.
-The nonce is related to the hash, which is a 256-bit value. It has to start with a lot of zeros (i. e., it must be extremely small). A nonce generates the cryptographic hash when the first block of a chain is created. The data in the block is regarded signed and irreversibly connected to the nonce and hash unless it is mined.

1.1.2 Miners

Miners use a technique known as mining to insert fresh blocks to the chain. Every block in a blockchain has its own unique nonce and hash, but it also refers to the previous block's hash, attempting to make mining a block difficult, particularly on large chains. Miners employ specialized methods to solve the enormously difficult math problem of producing a valid hash using a nonce. Because the nonce is only 32 bits long and the hash is 256 bits long, there are roughly four billion nonce-hash combinations to mine. When this happens, miners are said to have stumbled upon the "golden nonce," and their block is added to the chain.

Changing any previous block in the chain necessitates re-mining not only the damaged block, but all subsequent blocks as well. As a result, manipulating blockchain technology is extremely difficult. Consider it "safety in math," because identifying golden nonces requires a significant amount of time and computational abilities.

Whenever a block is successfully dug up, all nodes in the network recognize the alteration, and the miner is financially rewarded.

1.1.3 Nodes

One of the most fundamental concepts in blockchain technology is decentralization. A single machine or Organization cannot own or manipulate the chain. Instead, the nodes that connect to the chain create a distributed ledger. A node is any type of electronic device that saves copies of the blockchain and keeps the network running. Each node has its own copy of the blockchain, and in order for the chain to be updated, trusted, and confirmed, the network should algorithmically approve any new mined block. Because blockchain are transparent, every transaction in the ledger can be easily reviewed and investigated. Each participant is given a unique alphanumeric identification number that is used to keep track of their transactions.

1.2 Dapps (Decentralized Applications)

The issue of privacy and security has been of utmost importance in the last decade for consumers and businesses alike, as more users are being aware of their rights, businesses need to be more cautious about handling

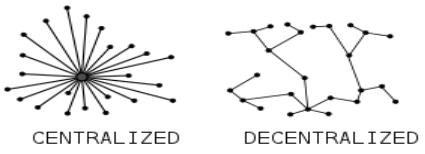
their user's data. Let's see a brief history of various technologies used to ensure privacy and security.

The concept of cryptography was first used in 1900 BC in Egypt. Since then it has become an essential tool in the arsenal of computer scientists and mathematicians to design robust systems which are secure and can not be compromised very easily. From the password of your Google account to the messages you send on WhatsApp, everything is encrypted, so only those with the encryption key can see them.

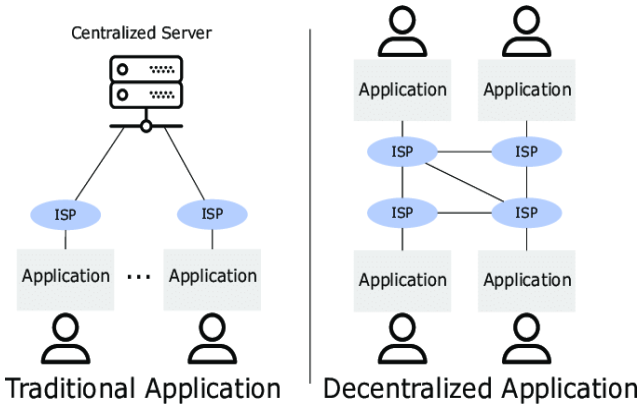
But when we talk about privacy, we are not always talking about a data breach or a hack, we also talk about the companies that are responsible for encrypting and handling this data, all these companies exploit the gathereddata in some or the other way.

That's when decentralization comes into play, 24 years after the blockchain was invented in 1991, In 2015, Vitalik Buterin invented Ethereum which gave birth to decentralization. Decentralization as the word implies is a terminology that is used to describe a system which is not controlled by any single entity. It is controlled by anyone and everyone who is the part of that system for e.g.- if an app like Facebook is decentralized, then all the data of all the users, will be stored in the billions of devices that are using Facebook instead of a single server or the servers of a company, only the user who own that data can access thatdata and each of those billions of devices is called a node.

The apps which are decentralized are called DApps.



They have a slightly different architecture as compared to a traditional app. The complete architecture will be discussed in the project in detail.



A lottery system is perfect use case for designing a DApp as users need to have trust on some person/company to take part in a lottery, a DApp can be designed as a transparent lottery system in which all the data and money will be stored on the blockchain till the time a winner is announced and then the money will be securely sent to the winner.

1.3 Ethereum Network

Ethereum is a decentralized, opensource blockchain with smart contract functionality. Ether is the native crypto currency of the platform. After Bitcoin, by market capitalization, it is the most valuable cryptocurrency. The Ethereum blockchain is the most widely utilized blockchain.

Vitalik Buterin, a programmer, presented Ethereum in 2013. Crowdfunding was used to support development in 2014, and the network started operating on July 30, 2015. Developers can use the platform to deploy permanent and immutable decentralised applications with which users caninteract.

It is a network management protocol that allows users to build and execute smart contracts on a decentralised network. A smart contract is made up of code that performs specific operations and communicates with other smart contracts, and it must be written by a developer. In comparison to Bitcoin, which stores numbers, Ethereum stores executable code.

With their ability to decentralise information and services, Ethereum DApps provide a platform for Web 3.0 to supply a totally free (as in freedom) and accessible Internet for everybody. Because there will be no middlemen to facilitatethe flow of information and services, there will be no need for a central point of control.

Ethereum is a permission-free, non-hierarchical group of networks (nodes) that construct and find consensus on an ever-growing series of "blocks," or batches of transaction data, known as the blockchain. Each block contains an identifier for the block that it must immediately follow in the chain in order to be valid. Whenever a node introducesa block to its chain, it implements the transaction records in the order they were added, causing the ETH balances and other storage values of Ethereum accounts to change. These balances and values, known collectively as the state, are kept separate from the blockchain on the node's computer in a Merkle tree.

1.4 Smart Contract

Smart Contracts are the most essential building blocks of our application which enables us to code our business logicand our in-game crypto currency into a piece of code which is then deployed on to a public blockchain network. Solidityis the name of the programming language which is used to create this smart contract. There will be two smart contracts that will be used on our lottery application.

- Token Smart Contract

This smart contract is used to create the currency that will be used in the lottery game. It will be an Ethereum token following the ERC20 standard and will be deployed separately on the blockchain network. This will specify all the functionalities of our in-game currency. From the currency name to its supply etc. This contract uses solidity language.

- Lottery Smart Contract

This smart contract consists of the business logic of our application, the logic behind picking the lottery winner, the logic of accepting the funds and storing it in the contract, the logic of transferring funds when a winner is declared. This will be implemented separately from the token smart contract on the Ethereum network. For development and debugging, this contract additionally employs the Solidity programming language and the Remix IDE.

1.5 Solidity Language

Solidity is an object-oriented programming language used to create smart contracts. It is used to implement smart contracts on various blockchain platforms, the most prominent of which is Ethereum. It was created by Christian Reitwiessner, Alex Beregszaszi, and several former Ethereum core contributors to allow the creation of smart contracts on blockchain platforms like Ethereum. The Solidity-compiled programs are designed to run on the Ethereum Virtual Machine.

Solidity is an object-oriented, high-level programming language used to implement smart contracts. Smart contracts are programs that govern how accounts behave in the Ethereum state.

Solidity is written in curly brackets. It is inspired by C++, Python, and JavaScript and is intended for use with the Ethereum Virtual Machine (EVM).

Solidity is statically typed and, among other things, supports inheritance, libraries, and complex user-defined types. Solidity makes it possible to create contracts for things like casting a vote, crowdfunding, blind auctions, and multi-signature wallets.

2. ISSUE WITH CURRENT LOTTERY SYSTEM

The key issue in the existing lottery systems is that the whole process is very vulnerable to various kinds of scams, frauds, and other attacks; and to participate in a lottery, a person must have some level of trust. The constitutional laws related to these types of systems are very ambiguous and are easily exploited, all these may lead to loss of money, identity theft, loss of privacy etc.

The overall problem can be further categorized into various aspects such as lack of privacy, the requirement of trust, possibility of technical failures, lack of accessibility, unclear procedures, long wait times, or sometime no payment at all.

Taking part in a lottery usually entails obtaining a physical lottery ticket with a serial number imprinted on it from a store, and then waiting for the winning serial number to be announced and comparing it to yours to see if you have won anything. The whole process is prone to human errors, from misplacing the serial number to not matching the serial number correctly; there are many things that can go wrong in such a system. If it's an online lottery, many of the issues gets resolved but new ones also arise such as lack of privacy, you need to provide your details along with your payment method details to purchase a ticket, the possibility of it being a scam etc.

All these problems need to be tackled if we want to democratize the whole process of taking part in a lottery and try our luck without worrying about the trivial stuff.

3. PROPOSED SYSTEM

Removal of the need to trust and increased privacy and robustness throughout the whole process of taking part in a lottery is the main goal of the proposed system.

The use of blockchain can help us achieve all these desired goals in a very efficient manner, use of blockchain is synonymous with privacy and anonymity and the overall rules and regulations governing such a lottery is system is available in the public domain as all the code written has to be deployed on a public blockchain like Ethereum, cardano etc.

Blockchain also does the job removing most of the technical issues such as risk of hacking, loss of credentials, identity theft etc. The smart contract written to manage the lottery system makes sure that there is no unauthorized access to the funds at any point of time and all the winnings are automatically transferred to the winner after the lottery manager has executed the function which picks a random person from the pool of participants.

The transfer of payments is done in the form of a cryptocurrency so that the privacy and anonymity of the participant is maintained and the transaction is immutable which means that the reversal of transaction is impossible.

The front-end of this system which is built in react is responsive and very intuitive. The back-end of this system can also be used to build a mobile-app, desktop app or any other type of app for any other platform.

More features can be added later, such as the ability to withdraw your money from the lottery, or ability to run multiple lotteries at one time etc.

4. WORKING OF THE SYSTEM

4.1 Architecture Flowchart

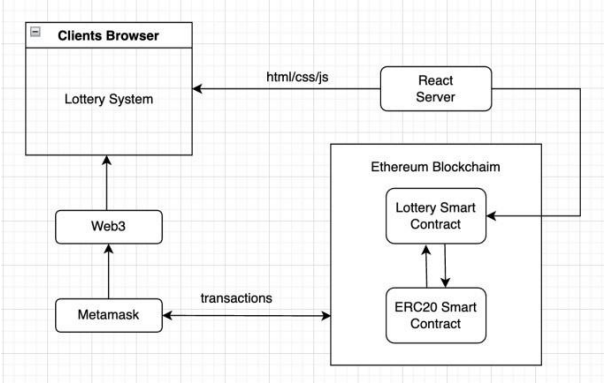


Chart-1: Architecture of Decentralized Application

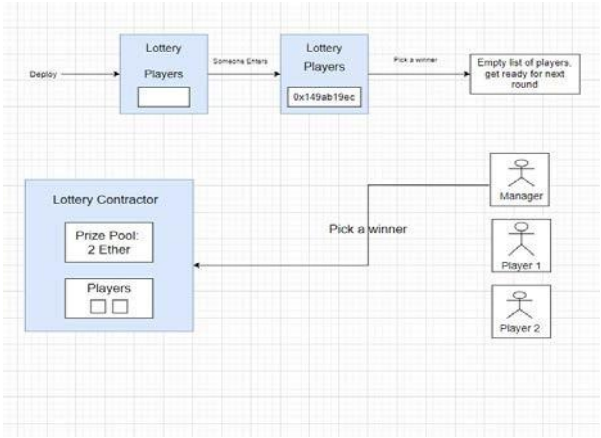


Chart-2: Working of the System

4.2 Flowchart Explanation

The architecture of our decentralized application is explained through above diagram:

There are two separate smart contracts namely Lottery Contract and ERC20 Contract deployed on the Ethereum blockchain network. The React server is responsible for serving the web-page to the client and also interacting with the deployed lottery contract to fetch the list of players, adding a new player and enabling the manager to execute the pickWInner function. The lottery contract then interacts with the ERC20 smart contract for transfer of funds, getting the total balance of an address, sending some token to another address.The react web-page also containsinjected web3 which means that we allow Metamask to take control of all the blockchain integration. This enables us to offload a bunch of tasks to metasmask. Things like gas fees calculation, account management and integration with various API's are taken care of.

The 2nd diagram explains the working of the system.

Suppose, there are two people who went to our lottery website and took part in it by paying some token amount. So the enterLottery function will add them to the players array. Now let's say the pickWinner function is executed by the contract manager, based on the random number generator, and either of them has an equal chance of winning the lottery. Let us say player 1 won the lottery, then, the combined token amount of player 1 and player 2 will be sent to player 1 as his winnings

4.3 Code

```
pragma solidity >=0.4.22 <0.9.0;import './ERCToken.sol';
contract Lottery {
string public name = 'Lottery Contract';ERCToken public
ERCtoken;
address public manager;address[] public players;uint256
totalMoney = 0;
constructor(ERCToken _ERCtoken) {
ERCtoken = _ERCtoken;manager = msg.sender;
}
modifier restriction(){
require(msg.sender == manager);
_;
}
function enterLottery(uint _amount) public {
require(_amount == 10000000000000000000);
ERCtoken.transferFrom(msg.sender,
address(this)
,
_amount);
players.push(msg.sender);
totalMoney = totalMoney +
_amount;
}
function random() private view returns
(uint) {return
uint(keccak256(abi.encodePacked(block.di
fficulty,block.timestamp, players)));
}
function pickWinner() public
restriction {require(players.length
> 0);
uint index = random() % players.length;
ERCtoken.transfer(players[index], totalMoney);

}
function getPlayers() public view returns(address[]
memory) {
return players;
```

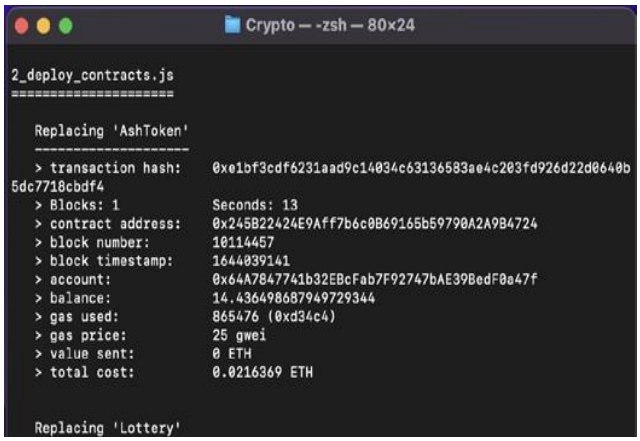


Fig-1: Token Contract Deployment with address.

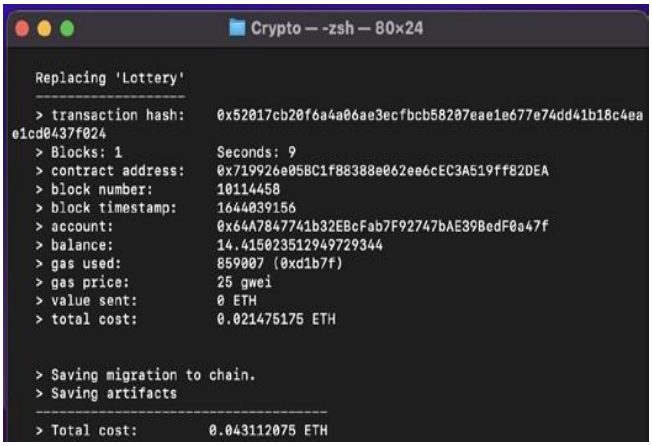


Fig-2: Lottery Contract Deployment with address.

4.4 Key Function Explanation

- The constructor functions accept the address of the deployed ECR20 token to initiate the lottery contract.

- The enterlottery function accepts 100 tokens from the sender account and the respective sender's address to players arrays.

- In solidity, the random function is used to generate a pseudorandom number.

- The pickWinner function is used to execute the winner picking logic of the Lottery. This function can only be executed by the manager of the lottery contract.

- The getPlayers() function is used to get the list of addresses which are part of the players array.

5. Benefits of Using Proposed System

No single point of failure:

Unlike a traditional web-app where there is usually a single point of failure like a web server, a decentralized app has no single point of failure as it runs on a blockchain which has a great number of nodes all around the world.

Transparency**:**

The complete code for the ERC20 token as well as the lottery system is deployed on a public blockchain network so any person can have a look at it and decide for themselves before using the system.

Security**:**

A blockchain system is virtually hack-proof as the computing resources required to manipulate a blockchain system is exponentially higher than the most powerful computer on this planet.

Privacy:

The lottery participant's identity is kept secret at all times; only the public key is displayed, which cannot be traced back to the owner in any way.

Decentralized:

As the application is completely decentralized, no single entity has any kind of access to any kind of data.

Block proof:

DApps aren't restricted to a single IP address. It is far more difficult for external authorities to stop a DApp because there is less authority owning the DApp's network.

6. CONCLUSION

In this paper, we develop a token and a lottery system to demonstrate a blockchain system based decentralized application which solves several issues that people are facing with current lottery games and systems. This paper introduces a new methodology using Smart Contract and Ethereum network in the process of selling andpurchasing lottery tickets. This system will replace the original lottery operations in every aspect such as the removal of third-parties in the buying process, ensuring the efficiency in prize declaration and the claiming of prizes. The proposed lottery system can ensure that the system is working with all fairness, transparency, and security.

With the help of this application, it is shown that Blockchain is not just about cryptocurrency and it has a lot of real- world applications that can help in solving major trust and security issues.

REFERENCES

[1] Vitalik Buterin et al. 2014. A next-generation smart contract and decentralized application platform. white paper (2014).

[2] Vitalik Buterin et al. 2013. Ethereum white paper. GitHub repository (2013).

[3] Nakamoto, S. (2008) Bitcoin: A Peer-to-Peer ElectronicCash System. https://bitcoin.org/bitcoin.pdf

[4] Chris Dannen. 2017. Solidity Programming. InIntroducing Ethereum and Solidity. Springer

[5]N Szabo. 1997. Nick Szabo - the idea of smart contracts. Nick Szabo's Papers and Concise Tutorials. http://szabo.best.vwh.net/smart_contracts_idea.html (1997).

[6]PlayDapp (August 2019) WhitePaper_v1.0