

“Transaction of cryptocurrency with Blockchain”



COMPUTER NETWORKS

SUBMITTED BY:

PARTICIPANT 01: SRISTI MITRA

ROLL: 2K19/CO/389 (A6)

PARTICIPANT 02: ZISHNENDU SARKER

ROLL: 2K19/CO/450 (A6)

SUBMITTED TO:

VINAY DUBEY SIR

COMPUTER ENGINEERING Dept.

DELHI TECHNOLOGICAL UNIVERSITY

MAY, 2021

ACKNOWLEDGMENT

I would like to express my deepest appreciation towards all the resources that have provided me the possibility to make progress in our report. A special gratitude I give to our Computer Networks teacher Vinay Dubey Sir, whose stimulating suggestions and encouragement helped to coordinate in writing this project. We were inspired by our subject teacher who taught us encryption and cryptography in the class . According to the knowledge we gained in the class and from some online resources , we are able to finish this project paper . We are also grateful that the project enhances our knowledge towards cryptography and hash algorithms and some related topics in Computer Networks .

TABLE OF CONTENTS:

SL NO.	NAME OF THE TOPIC	PAGE NO.
01	INTRODUCTION	03
02	RELATED WORK	04
03	WORK EVIDENCE	07
04	PROPERTIES	08
05	PROBLEM DESCRIPTION	09
06	ALGORITHM	10
07	FRAMEWORK AND SOFTWARE USED	10
08	IMPLEMENTATION	11
09	RESULTS	12
10	CONCLUSION	16
11	REFERENCES	17

Transaction of cryptocurrency with Blockchain

Sristi Mitra¹, Zishnendu Sarker² ^{[1],[2]}Department of Computer Science and Engineering
Delhi Technological University, Delhi-110042

¹sristimitra_2k19co389@dtu.ac.in

²zishnendusarker_2k19co450@dtu.ac.in

Abstract. The blockchain has been producing cryptocurrency, and the transaction process takes place through the smart contract, which is the blockchain's second layer (cryptocurrency transaction) and third layer (smart contract). Nakamoto Satoshi created the first decentralized cryptocurrency, Bitcoin, in the late twentieth century, and several concepts about the protocol termed Bitcoin have remained unclear since. We created a blockchain that is feasible for blockchain beginners and explain the protocol in this paper, as well as implement Smart Contracts that are both user-friendly and decentralized using Flask as a web node to decentralize our blockchain with Python and create smart contracts with Truffle Grenache provided by the Ethereum protocol of Crypto.

Keywords: Blockchain, Smart Contract, Cryptocurrency, Peer to Peer(P2P), Decentralization, SHA256, Block Mining, Consensus Protocol.

INTRODUCTION:

With a market worth of \$786.32 billion, Bitcoin has made other cryptocurrencies more well-known across the world. The blockchain is a technology that uses timestamps, nonces, and Hash(SHA256) as the identifier to access the block (the cryptography part has been used in hash). The blockchain will be decentralized by the Genesis-Block and will distribute the hash of each block to the other by Peer-To-Peer Network(P2P) that uses the Consensus Protocol to authenticate each blockchain. As a result, Genesis-Block will communicate with other nodes (Miner or Other PC or MemPool). For user friendliness and feasibility, we utilized Flask as the decentralized web to describe the steps of creating the blockchain.

- **BACKGROUND:** Every blockchain has a unique hash and nonce that distinguishes it from the others. If an intruder tries to modify a single bit of data in the blockchain, SHA256 (Hash cryptography) will update the whole hash document and remove the fraudulent blockchain from the chain using the Consensus Protocol. Every day, bitcoin transactions are carried out. The blockchain technology, which will be discussed later in this article, was used to create the transaction data. The transactions are carried out via a smart contract, which is a blockchain application that assists in the management of bitcoin and token transactions.

RELATED WORK:

- A smart-contract is a piece of blockchain software that verifies itself, executes itself, and is unaffected by manipulation. Nick Szabo proposed the concept of smart contracts in 1994 [5]. It enables third parties to run programming code without requiring their involvement. The following are the components of a smart contract: address, status, function, and value. Taking transaction data, running applicable code, and triggering output events. The logic implementation states fluctuate depending on the function. Since the launch of blockchain technology with the Bitcoin cryptocurrency in 2008, Since it permits P2P transactions as well as databases that may be maintained publicly in a safe system in a trustworthy system, blockchain integration in smart-contracts has been a focus as a development area.

The smart-contract has the following characteristics:

- A smart-contract is a machine-readable code that is running on a computer blockchain platform.
- Smart-contracts are a component of a single software application.
- Smart-contracts are programs that are triggered by an event.
- Once developed, smart contracts are self-contained and do not require monitoring.
- The availability of smart contracts is made available.

Solidity is a high-level programming language for developing intelligent systems. contracts. Developing a solid blockchain platform Smart contracts like Ethereum, Zeppelin, Eris DB, and Counterparty may be traced and are permanent.

- A blockchain is a distributed data storage system that is shared across nodes (connected computers). A blockchain is an electronic database. Bitcoin is well-known, E.g.Bitcoin, for retaining a decentralized and secure record of transactions, which is a vital feature in the cryptocurrency ecosystem. The blockchain's creative idea is to ensure the accuracy and security of an information record while also generating confidence without the need for a trusted intermediary..
 - The data structuring technique is constructed differently than a traditional database and blockchain. It gathered information in the form of blocks, which include data in sets, within the network. When the block's storage capacity is reached, it is encased and attached to the block that was filled first..
 - A database, on the other hand, organizes data into tables, but a blockchain, as the name implies, organizes data into chunks (blocks) that are then joined together.

This data structure creates an irreversible data chronology when generated in a decentralized manner. When a block is finished, it is carved into the stone and becomes part of the chronology. A unique time stamp is applied as each block in the chain is completed. Every block includes the SHA256 cryptographic mechanism.

- Method of Secure Hashing (SHA) The hash function of the Bitcoin Protocol and algorithm for mining is 256, which refers to a cryptographic hash function that produces a 256-bit result. SHA-256 controls the generation and administration of addresses. It's also used for transaction verification. In Bitcoin, the hash function is employed twice, implying that SHA-256 is used.
- The technique is based on the SHA-2 algorithm developed by the National Security Agency (Secure Hash Algorithm 2). (NSA). SHA-256 is also used in popular encryption protocols like SSH, SSL, and TLS, as well as operating systems that are open source, like Linux. Hash's algorithm is extremely safe, and its inner workings are kept secret from the general public. The entire hash will change if the block is changed..

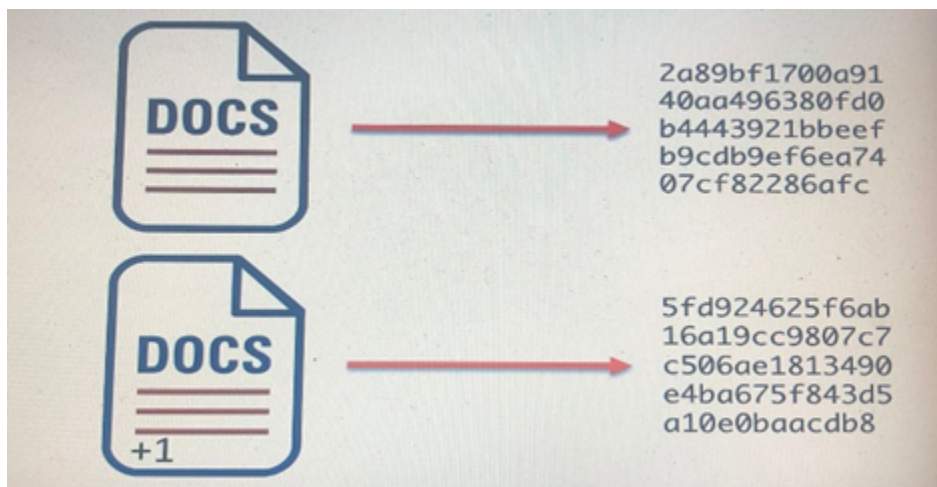


Figure 1: Change in Hash if one bit change in Block

- Mining The blockchain is a method that adds transactions to an existing ledger of transactions that is shared among all blockchain users. While most people identify bitcoin mining with it, other blockchain-based systems use it as well. Mining entails generating a hash of a block of transactions that can't be easily manipulated; hence, without a centralized system, ensuring the integrity of all blocks on the blockchain will be impossible. That is why some hackers mine bitcoins on workstations they have hacked, leading an unwary victim to pay the price of mining while receiving no benefits. As a result, the protocol is used to determine whether or not the block has been confirmed. The protocol is Consensus Protocol, and the chain uses authentic.

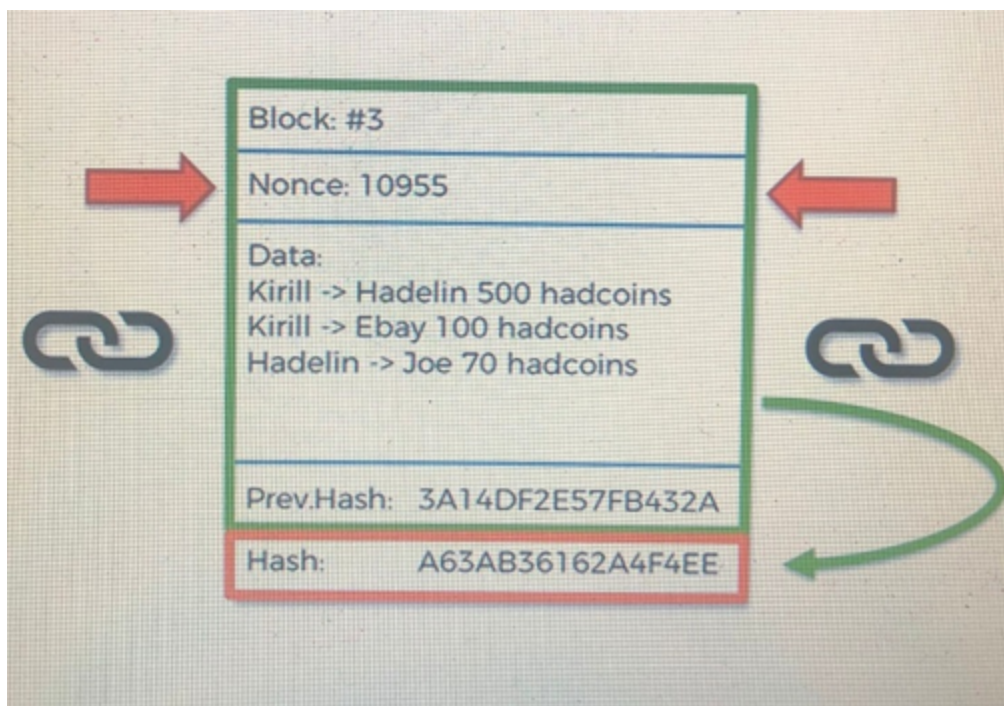


Figure 2: Example of Blockchain which have been mined

- The Consensus Protocol is the core of the blockchain, allowing every participating node to validate transactions at the same time. Proof of Work (Pow) is used to implement the Bitcoin Consensus Protocol, which takes time and resources. Bitcoin's transaction verification rate is sluggish in comparison to Visa and MasterCard, driving the creation of alternative consensus techniques.
 - In terms of how networks attain consensus, all or any other blockchain apps, also known as decentralized applications (dApps), differ. dApps employ peer-to-peer (P2P) computer networks instead of centralized nodes or servers. The lack of centralized power is another property of decentralized apps. Most of the common applications we use today are run by a collection of individuals or corporations that characterized dApps as the process of creating a decentralized system of scale. Consensus protocol types
 - Before we go into the consensus protocols, let's have a look at a statistical fact about them.
 - In theory, the blockchain is considered hacked if a hacker gains access to 51 percent or more of the network.
 - Different types of consensus procedures address the 51 percent attack problem in different ways.

WORK EVIDENCE:

Proof of Work was one of the earliest blockchain consensus techniques to be used. It operates by computing hash values and verifying transactions until the hash value contains a specific amount of trailing zeros. A nonce is a random integer that generates the necessary amount of trailing zeros in the hash function. Properties Proof of Work is a consensus algorithm designed for permissionless public ledgers that uses the processing capacity of the node's systems. The blocks are shown using a linear framework. Each block consists of a number of transactions.

By aiding all nodes in the network in confirming transactions, consensus mechanisms serve as the backbone of blockchain. Proof of work (PoW) is Bitcoin's consensus protocol, which is both energy and time intensive. The rate of transaction verification in Bitcoin is considerably sluggish in compared to Visa and MasterCard. Other consensus procedures have as a result been proposed.

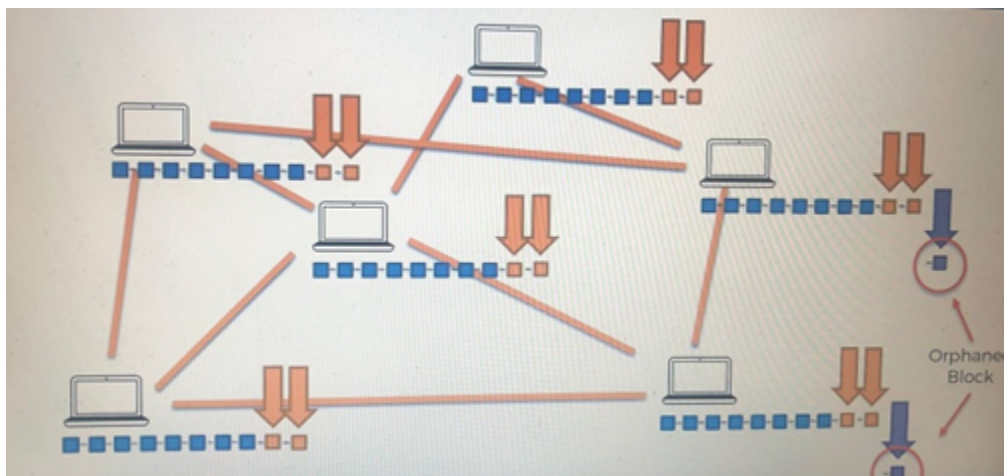


Figure 3: Process of Proof of Work of Consensus Protocol Proof Of Stake(Ethereum)

In terms of how the network obtains consensus, all crypto-currencies and other blockchain apps, also known as decentralized applications (dApps), differ. dApps employ a peer-to-peer (P2P) network of computers instead of a centralized node or server. The lack of centralized authority is another property of decentralized apps. The bulk of today's standard programmes are controlled by a group of people or corporations who set the terms of use.

Proof-of-stake consensus was adopted by Ethereum, one of the first significant cryptocurrencies. Let's investigate this matter further. Assume we're transaction validators. A person who validates bitcoin transactions by computing the hash value with a specific number of leading zeros receives the amount of bitcoins assigned to them. As proof of stake consensus, a validator is selected and given a block. The miner must first allocate a fraction of his bitcoin to begin validating. If the miner is successful in invalidating the transaction, the payout is equal to the stake they initially invested plus transaction expenses. This is a technique for punishing poor conduct while rewarding good.

- **PROPERTIES:**

Validators are picked depending on the amount of money they have invested in the network. The purpose is to prevent mining hubs from becoming centralized and to allow all miners to validate. There is no computational challenge, thus it is environmentally friendly. Specialized equipment is not required in mining what will safeguard cryptocurrency transactions.

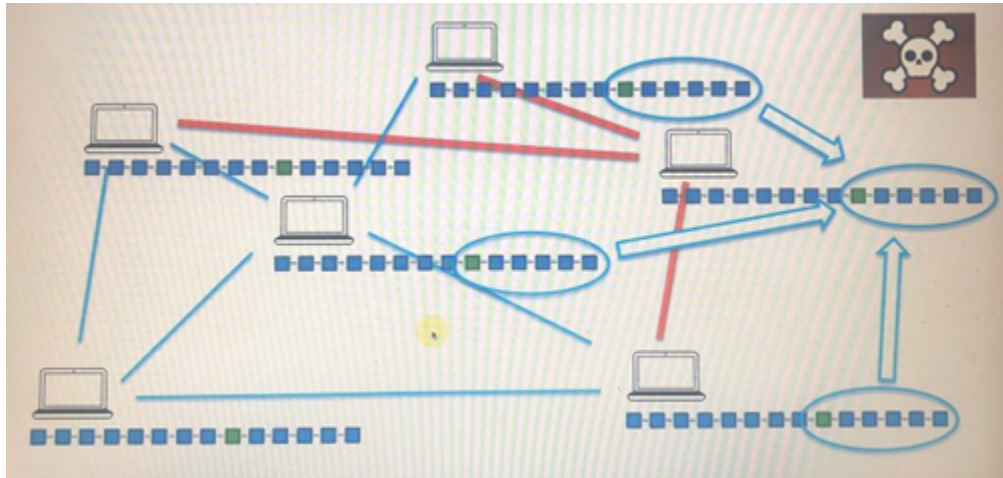


Figure 4: Identify Malicious attack by Consensus Protocol

- **Transactions in cryptocurrencies** It's a digital payment system that doesn't rely on banks for transaction verification. It's a peer-to-peer payment system that lets anyone send and receive money from anyone else in the world. It's nothing more than a digital record in an internet database used to verify the legitimacy of a certain transaction. Instead of moving and exchanging physical money in the actual world. A publicly available ledger records all currency transactions, including bitcoin. Digital wallets are used to store cryptocurrency. Bitcoin was the first cryptocurrency and is still the most well-known today. The majority of cryptocurrency investment is speculative, with speculators occasionally driving prices to absurdly high levels.

The use of encryption to verify transactions is referred to as "cryptocurrency." This implies that bitcoin is stored and sent between wallets and public ledgers using advanced encryption. Security and safety are the goals of encryption. After the next block of the chain has been mined, each transaction will proceed. The person that mines it will receive a payment from the block since they assisted the provider in locating the hash of the block, which will process thousands of transaction requests from other nodes. All of these transaction requests will be saved in MemPool and subsequently allocated to a block by an algorithm to process them.

- Unspent transaction output (UTXO) is the amount of digital money that remains after a bitcoin transaction (UTXO). It's comparable to the change you get after buying something, but it's not a lower denomination of money—it's database transaction output created by the network to accommodate non-exact change transactions. The fraction of the total bitcoin that is not spent in a transaction is used as an accounting metric. Each transaction contains an input and an output, similar to double-entry bookkeeping. 1 BTC is equivalent to a bucket of coins. Each coin represents one UTXO.
- All legal transactions are kept in MemPool while the Bitcoin network confirms them. A large mempool size indicates increased network traffic, which means the average confirmation time is longer and priority expenditures are higher. The capacity of the Mempool chart tells us how long the congestion will last, whilst the transaction of the Mempool Count picture shows how frequently Transactions are jammed. Transactions from the mempool must be included in the block for us to validate. Each transaction has its own weight, which is determined by the kind of UTXO transaction used (input) and the address provided to it in the block (output).

By joining the Bitcoin network, every Bitcoin node creates its own version of the mempool. The mempool is made up of some of the technical team's most recent Bitcoin nodes. Blockchain.com allows us to gather as much data as possible in order to build a reliable meme pool.



Figure 5: Transaction Process with MemPool

PROBLEM DESCRIPTION:

The blockchain implementation will provide insight into how each protocol works and demonstrate how each node may connect to one another using the unique HTTP that will be implemented.

Initially, we'll use Python and Flask to build a site to decentralize the Genesis block, which will link to other nodes via a Flask site using the HTTP standard. The OSI Model, which is a computer network concept, will be implemented as a node connection across Subnet <http://127.0.0.1:5000/>. After connecting each node, we'll use the Grenache Ethereum Framework to write a smart contract.

ALGORITHM:

START-> create block -> proof_of_work Consensus Protocol create -> check_if_chain_valid -> connect to Flask -> Get_mine_block ->Get_chain ->add_node ->add_transaction -> replace_chain

A.) Building Blockchain

Building blockchain with the algorithm which is required to be perform; create_block, get_previous_block, proof_of_work, hash, is_chain_valid, add_transaction, add_node, replace_chain which all these functions contain the algorithm of building Blockchain will be provided.

B.) Request Decentralization from Server

Deploy the Genesis blockchain which will be the initial block which will be chain if the next block is mined and chain to it.

C.) Connect Node

Pull request to HTTP server to connect the node with other node with different device which will mine the blockchain

D.) Start Mining

The algorithm which have been included will run and will mine the block itself

FRAMEWORK AND SOFTWARE USED:

Framework:

Flask Framework server Environment

Software used:-

Ganache , Postman, Spyder(Python IDE),Solidity compiler

IMPLEMENTATION:

Languages:- Python, Solidity

Operating System:- Windows

Library Packages or APIs Used:-

- Flask : To set up HTTP framework server environment
- Requests: To perform function request from Postman server environment
- Uuid : To show HTTP format message
- urllib.parse : To relate the HTTP message
- hashlib : To perform SHA256 algorithm
- datetime: To make a timestamp to Blockchain
- JSON : To get the json file for framework server environment

A. Decentralized Blockchain

Deploy the blockchain that was built by building the Spyder, which will connect the node to the Postman location at <http://127.0.0.1:5000/> and construct the Genesis Block, which will be joined by another block later.

The data from the previous blockchain will be included in the block that will be mined and joined to the chain.

The transaction would be completed once the software was executed. Each transaction's data will be stored in a blockchain that will be mined after the request has been given to MemPool, and when a block to connect to the chain exists, the transaction will take place in that block.

B. Create Smart Contract

With Ganache software and solidity and MyEtherWallet We create the smart contract Server location which will provide you a Hash256 address as your address for wallet and with MyEtherWallet perform with the address from Ganache we could create the wallet with will contain the data as the solidity languages which have been developed by C++ languages

RESULTS:

Given screenshots , provide the results that we have obtained.

A. Generate Genesis Block and connect node

The screenshot shows a REST client interface with a POST request to `http://127.0.0.1:5002/connect_node`. The response is a JSON object indicating that all nodes are now connected and listing the total nodes.

KEY	VALUE	DESCRIPTION
Key	Value	Description

```

1  {
2    "message": "All the nodes are now connected. The Hadcoin Blockchain now contains the following",
3    "nodes": "total_nodes": [
4      "127.0.0.1:5003",
5      "127.0.0.1:5001"
6    ]
7  }
  
```

Response status: 201 CREATED, 41 ms, 305 B.

Figure 6: Connect Node

B. Mine the block

The screenshot shows a REST client interface with a GET request to `http://127.0.0.1:5003/mine_block`. The response is a JSON object indicating that a block has been mined successfully, including details like index, message, previous hash, proof, timestamp, and transactions.

```

1  {
2    "index": 6,
3    "message": "Congratulations, you just mined a block!",
4    "previous_hash": "31dae7fa05e8680104d59d6c05d909e3482c74b33bb472ea5c3cb4019f3e9714",
5    "proof": 48191,
6    "timestamp": "2022-04-21 05:37:36.561161",
7    "transactions": [
8      {
9        "amount": 1,
10       "receiver": "Dhruva",
11       "sender": "f596258738ef43a58e40c2ec984744ee"
12     }
13   ]
14 }
  
```

Response status: 200 OK, 123 ms, 443 B.

Figure 7: Mine the Blockchain

C. Longest Chain win the chain

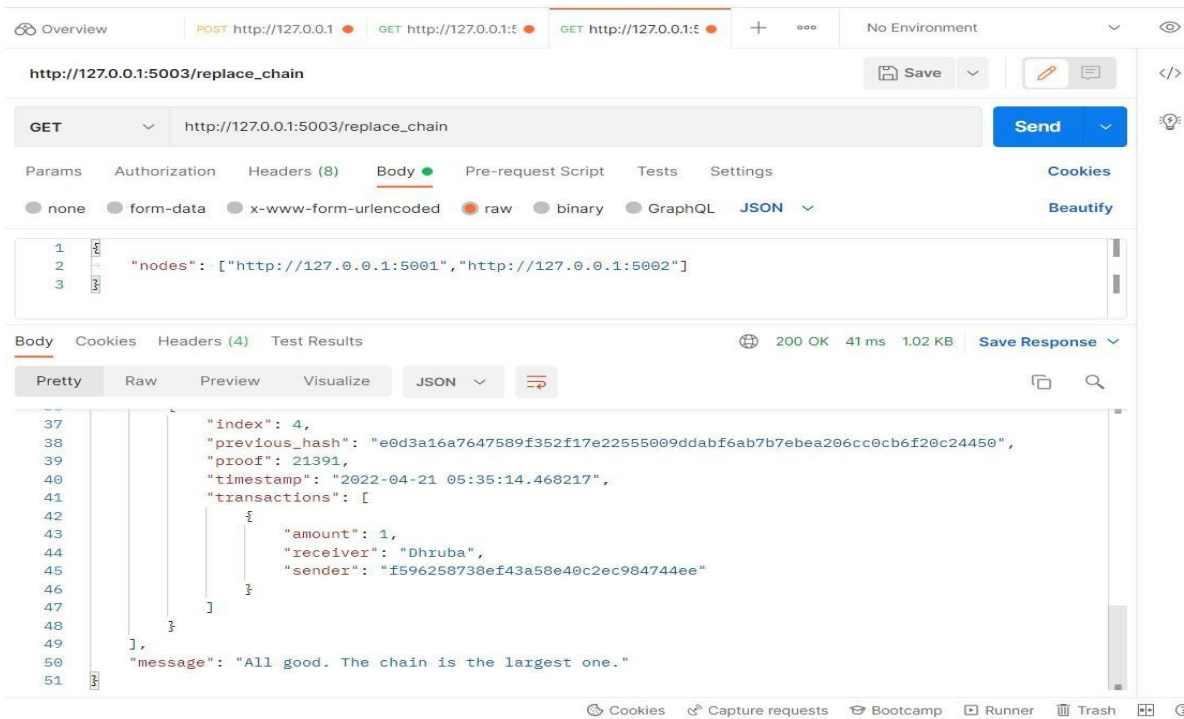


Figure 8: Longest Node win the chain

D. Make Transaction

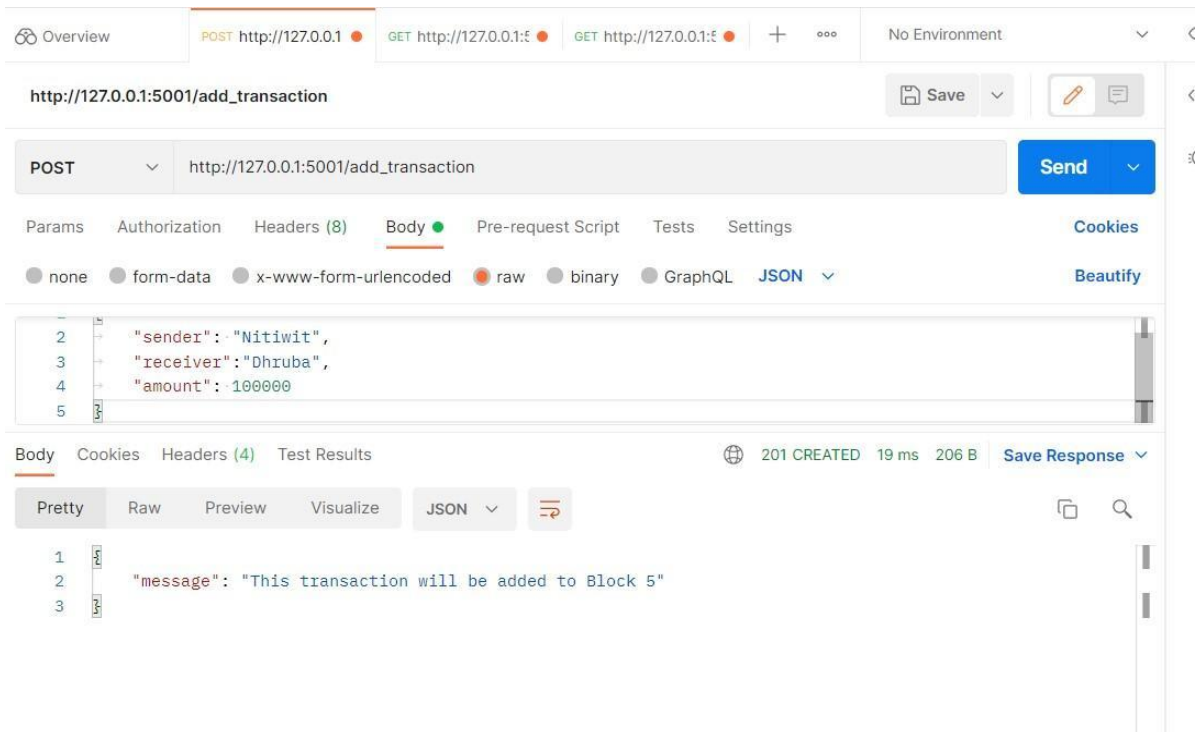


Figure 9: Make Transaction from port 5001 Nitiwit to 5003 Dhruba

E. Mine Block with Transaction

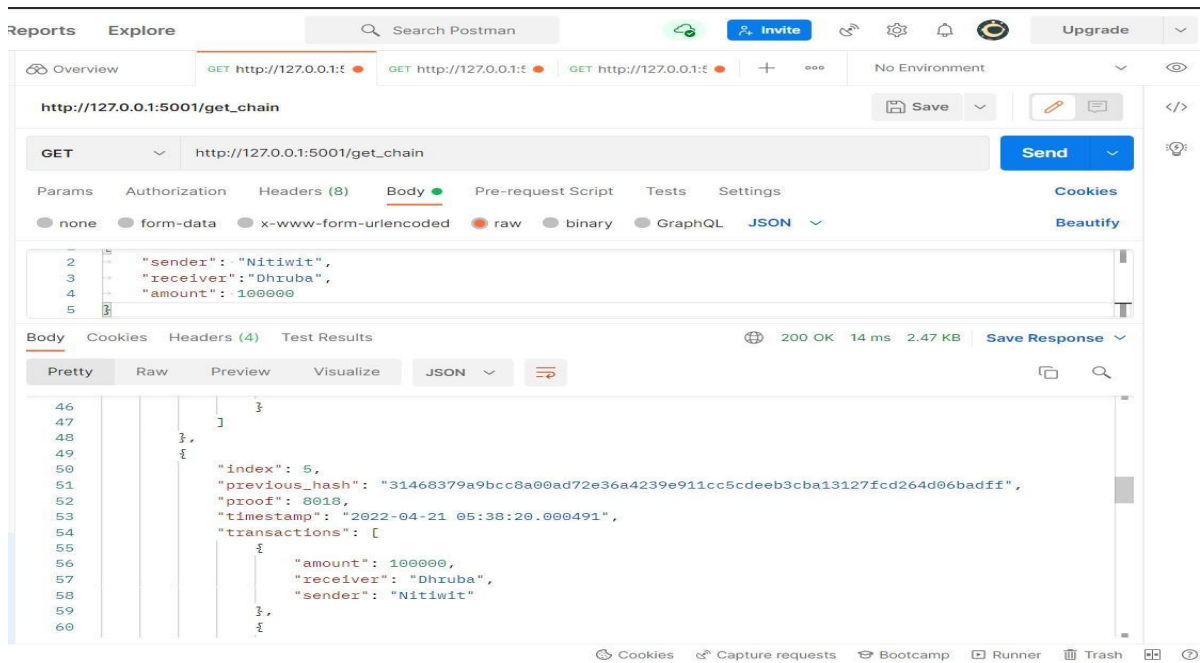


Figure 10: The blockchain with the transaction

F. Create Smart Contract

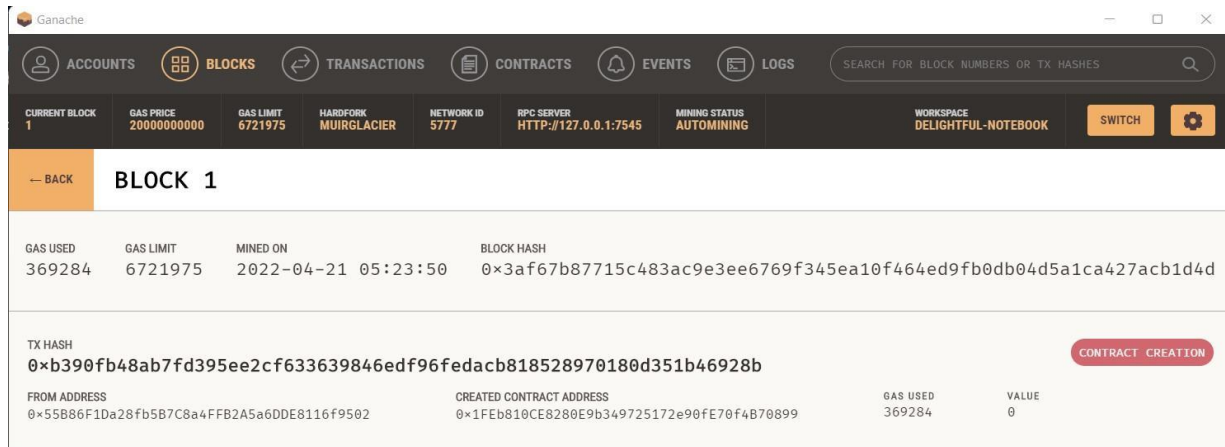


Figure 11: Smart Contract

H. Transaction with Smart Contract (Buy Had coin)

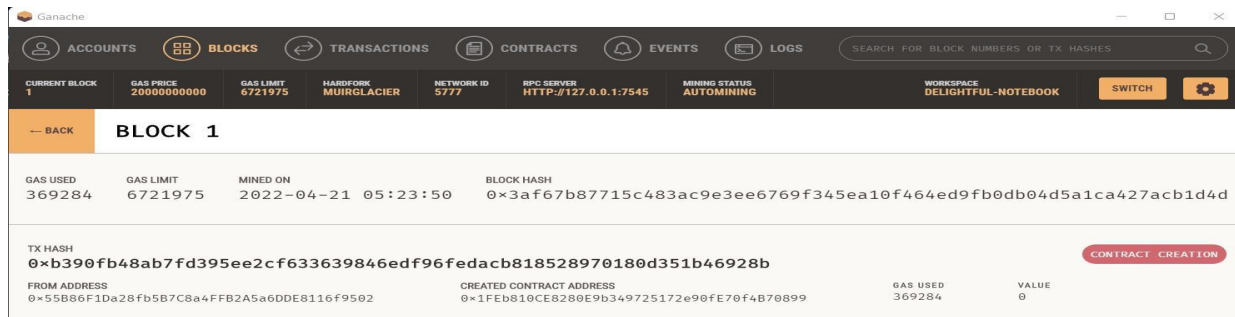


Figure 12: Ganache first transaction


I. Check value of Had coin

Read / Write Contract

0x1FEb810CE8280E9b349725172e90fE70f4B70899

equity_in_hadcoins ▾

investor address

0x55B86F1Da28fb5B7C8a4FFB2A5a6DDE8116f9502 

uint256

100000

Figure 13: Value of Had coin in USD


J. Sell had Coin via Smart Contract

Read / Write Contract

0x1FEb810CE8280E9b349725172e90fE70f4B70899

sell_hadcoins ▾

investor address

0x55B86F1Da28fb5B7C8a4FFB2A5a6DDE8116f9502 

hadcoins_sold uint256

500

WRITE

Figure 14: Sold had coin

K. Had coin Transaction in smart Contract

BLOCK 5	MINED ON 2022-04-21 05:31:19	GAS USED 39691	1 TRANSACTION
BLOCK 4	MINED ON 2022-04-21 05:30:40	GAS USED 42864	1 TRANSACTION
BLOCK 3	MINED ON 2022-04-21 05:30:31	GAS USED 24265	1 TRANSACTION
BLOCK 2	MINED ON 2022-04-21 05:28:24	GAS USED 87864	1 TRANSACTION
BLOCK 1	MINED ON 2022-04-21 05:23:50	GAS USED 369284	1 TRANSACTION
BLOCK 0	MINED ON 2022-04-20 05:57:55	GAS USED 0	NO TRANSACTIONS

Figure 15: List of total transaction of Had Coin in our Etherwallet

CONCLUSION:

The blockchain is the new way to do business. Thousands of transactions are made every minute using blockchain, smart contracts, and even E-wallets.

We went through everything in depth, including the fundamentals of blockchain and cryptocurrencies, as well as smart contracts, which will be incredibly helpful in the future of blockchain since they are simple to grasp.

The implementation and use of smart contracts and blockchain will vary in the future, thus the goal of our project is to present the core notion here, which is connected to computer networks in terms of Peer-to-Peer protocol. Finally, the topic's concept might be evolved into another Bitcoin in the near future or utilized as the money itself.

REFERENCES:

- [1] E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne. Finding high-quality content in social media. In Proceedings of the First ACM International Conference on Web Search and Data Mining (WSDM '08), 2008.
- [2] J. Allan, editor. Topic Detection and Tracking: Event-based Information Organization. Kluwer Academic Publisher, 2002
- [3] Matches the Skype Network Traffic Forensics. 3 Internet Criminal Procedures and Trusted Computer Workshop, October 29-30, IEEE Xplore Press, Ballarat, pages: 19-27. Segment: 10.1109/CTC.2012.14 Bardis, N.G. furthermore, K. Ntaikos, 2008.
- [4] H. Becker, F. Chen, D. Iter, M. Naaman, and L. Gravano. Automatic identification and presentation of Twitter content for planned events. In Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM '11), 2011
- [5] Al-Riyami, SS and K.G. Paterson, 2003. Uncertified public key cryptography. Methods for the Ninth World Theoretical Conference furthermore, Use of Cryptology and Information Security, November 30- Dec. 4, Springer Berlin Heidelberg, Taiwan, pages: 452-473. DOI: 10.1007/978-3-540-40061-5_29 Azab, A., P. Watters and R. Layton, 2012.
- [6] Building security AES cryptographic-based visit application calculation and key administration. Methodology for tenth World WSEAS Conference on Mathematics Methods, Numeracy Methods and Intelligent Systems, (TIS '08), ACM, USA, pages: 486-49
- [7] D. Sheiko, "Persistent Full Duplex Client-Server Connection via WebSocket,"2010. <http://dsheiko.com/weblog/persistent-full-duplex-client-server-connection-via-web-socket>
- [8] Makoto, "Living on the Edge of the WebSocket Protocol," 2010. <http://blog.new-bamboo.co.uk/2010/6/7/living-on-the-edge-of-the-web-socket-protocol>