VIETNAM NATIONAL UNIVERSITY – HO CHI MINH CITY

THE INTERNATIONAL UNIVERSITY

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING



**CHARITY FUNDRAISING APPLICATION BUILT ON ETHEREUM BLOCKCHAIN**

By

Ho Minh Huy

A thesis submitted to the School of Computer Science and Engineering

in partial fulfillment of the requirements for the degree of

Bachelor of Computer Science

Ho Chi Minh city, Vietnam

2018

**CHARITY FUNDRAISING APPLICATION BUILT**

**ON ETHEREUM BLOCKCHAIN**

APPROVED BY:

Tran Thanh Tung, Ph.D.

THESIS COMMITTEE

**ACKNOWLEDGMENTS**

I would like to express my very great appreciation to Dr. Tran Thanh Tung for his valuable and constructive suggestions and guidance during the planning and development of this research work. His constant encouragement and support helped me to achieve my goal.

Table of Contents

[LIST OF TABLES 5](#_Toc512869996)

**[LIST OF FIGURES](#_Toc512869997)** [6](#_Toc512869997)

[ABSTRACT 7](#_Toc512869998)

**iv**

# LIST OF TABLES

# **LIST OF FIGURES**

# ABSTRACT

Blockchain is a recent technology which provides a alternative type of database, structured as a chain of blocks of data , which is managed autonomously using a peer-to-peer network and distributed timestamping server. By design, a blockchain is unchanged. At the time researching this study, there are 3 major blockchain protocols such as Bitcoin, Ethereum and Ripple Consensus Network.

Current approaches of making payment through applications, especially on websites, is that payers have to trust and provide their private information to the middleman which is a bank or a payment service to valid transactions from their accounts and transfer their money to the final destinations. It is too risky. Not only in making payment but also in funding, the fundraiser owner does not directly receive funds from donators. Both sender and receiver are able to keep track of their transactions. But, it is not the way they expected.

This research approach is to apply Blockchain technology to create a decentralized application running on Ethereum private blockchain. This application has its own custom Token exchange and fund between owners and fundraisers. Besides, it is also required HTML including CSS, Javascript, Jquery, Bootstrap and Web3js for the front-end. The local server is run using Express module in Nodejs and a distributed database, which is MongoDB, to store fundraisers contents.

# CHAPTER I

## INTRODUCTION

The current fundraising system is very centralized. Kickstarter, one of the largest fundaraising platform, is a conspicuous example. Product teams can go to <https://www.kickstarter.com/> to create a project, set a goal and start collecting money from others who interested in their ideas. Kickstarter is essentially a third party that sits between product teams and supporters. If the project successfully funded, the project team expects Kickstarter to give them the money. On the other hand, supporters want their money to go to the project if it was funded or to get a refund when it hasn’t reach its goals. Therefore, both of them need to trust Kickstarter to handle their money correctly.

There is always a middleman sits between the donators and the fundraiser’s owner. It makes the fundraising system becomes centralized and depends too much on the third-party such as a bank or a trusted organization ( like Kickstarter ) to manage and control the fund. Moreover, the donators must provide their personal information to the bank in order to prove their identity each time they want to use their money. Personal information leaking is a constant worry of internet users these days. Actually, it is one of the world’s most valuable commodity. It can be mostly used in business marketing, surveys, or event hacking bank accounts and take control on someone’s property. The intermediary often promises to keep the personal information safe by providing an array of protection strategies. But somehow, they are not totally efficient. The Facebook’s Data Leak, which is a most serious problem of the company in 2018, has been revealed that the data of up to 87 million users was leaked. It made the reputation of Facebook be seriously damaged. It has also aroused controversy over the data branch of the intermediaries. Internet users then have an awareness of providing their information on the public network.

The risk of personal information leaking and middlemen role has been proved. In Vietnam, from 2016 to 2018, there were a dramatically increase of reports of cases that the money was withdrawn from bank accounts without any permissions of the owners. Most of cases reasonable occurred due to the greed of a person or a party, who worked for the bank. Those people were able to steal the money from an account because they could keep track of it using the information that the owner had provided. Even they finally paid for their crime, but it was not sufficient to make up for the loss and the reputation of the bank.

A question of whether the role of the intermediary is necessary or not. And, is it possible if people are able to manage their own property even when it is being traded or transfered from one to another on a public network or online market ? Yes, it is possible.

One of the best way to prevent those things happened and remove the role of the middlemen or third-party is to build a fundraising system which uses **cryptocurrency** and runs on **a blockchain**. The main goal of this research is to build a Charity Fundraising System called Decentralized Application, which is a combination of smart contracts, and deployed to Ethereum blockchain testnet - a simulation of real Ethereum blockchain - in order to make all the fundraisers and their funding transactions in this system become immutable, indelible and transparent. The system also use cryptocurrency to do transactions in order not to collect or require donators personal information. But, the solution is not as 100% as it is expected because the world is still using banknote in trading. Hence, it requires a final exchange from cryptocurrency to cash.

# CHAPTER II

## BACKGROUND

### **Blockchain technology**

Blockchain technology is amazingly trendy nowadays. This technique was authentically described in 1991 by a group of researcher and its purpose was to “timestamp” digital documents in order to prevent them from being tampered or put on a date that was earlier than which they were written. But until 2009, Satoshi Nakamoto adapted this technology to run a the digital cryptocurrency called Bitcoin, Blockchain has become widespread.

As its name, a blockchain is a database structure that can be described as a chain of blocks. The significant design of blockchain technology comes from these two techniques: Hash function and Merkle Tree.

* **Hash Function**

00111100...10101

INPUT

e1703fa96e70xa....

OUTPUT

HASH

This is a one-way function. It takes an input, usually a sequence of bits which could be any digital data, and produces a fixed size values,such as an alphanumeric string, as output. However, if just only a single bit of the input is changed, the output would be completely different. Moreover, there is noway to figure out the input if there is only the output.

* **Merkle Tree**

Merkle Tree is a data structure where each layer is a combination of hashes.

1

2

12

3

4

34

R

1234

To construct the Merkle tree, each data is hashed and then coupled by hashing their hashes together. For example, from the figure......, [1], [2], [3] and [4] are hashes of data. They are then coupled and produce the output of two new hashes [12] and [34]. To form the Root of the tree, [12] and [34] are hashed together to produce the Top hash [R]. The root will be a representation of this data structure and it is impossible to figure out each individual data from the root.

The blockchain can be seen as a huge Merkel Tree. Each block contains some data, the hash of that block and the hash of the previous block. The data that is stored inside a block depends on the type of the blockchain. The hash of a block can be comparable to a fingerprint. It identifies a block and all of its contents. Just as a fingerprint, a hash is always unique. Once a block is created, its hash is calculated. If there is any change inside the block, it will be easily detected because the hash of that block is also changed. Therefore, it no longer is the same block. The third element inside each block is the hash of the previous block, which acts as a link to the previous one. This technique dramatically creates a chain of blocks and makes blockchain is so secure. In a blockchain, if any middle block is tampered, the hash of that block will be recalculated. In turn, that will make its adjacent block and all the following blocks invalid since they no longer store a valid hash of the previous one. Hence, changing a single block will cause all the hashes of other blocks to be recalculated in order to make the blockchain valid again.

In the area of technological breakthroughs, computers these days are so sophisticated that can calculate hundreds of thousands of hashes per second. The recalculation of all hashes of other blocks as a consequence of meddling with a middle block can be done easily. To due with this situation, blockchain has a special mechanism. This mechanism provides a difficult mathematical problem based on a cryptographic hash algorithm, which requires to spend lots of resources and time to solve . The solution for this puzzle is call Proof-Of-Work. To prevent the disturbances from the outside, this mechanism will automatically adjust the difficulty of the problem so that it can remain the time required to solve the problem. So if a person want to tamper with block N in a blockchain requiring 5 minutes to find the proof-of-work, he or she has to recalculate the hash of ( N-1) previous block and take at least T = ( N-1)\*5 minutes. Before finishing those work, another new valid block has been confirmed and link to that blockchain within 5 minutes. Therefore, cheating with a blockchain is now become harder than ever. Besides, it is wasted because energy and resources consumption cost a lots !

There is one more way that most of blockchains secure themselves: being distributed. Instead of using a central entity to manage the chain, blockchain uses a peer-to-peer network. Anyone is allowed to join and became a node of the network. When someone runs as a full-node, he or she gets the full copy of the blockchain. The node can use this to verify that everything is still in order. When someone in the network successfully creates a new block by solving for the solution ( Proof-Of Work ) and add it to the chain, it is called “mining”, and he or she is a miner. Mining process will proves that a miner did spend a lot of time and resources to solve the problem. That new block is sent to every nodes on the network. Each node then verifies the block to make sure that it has not been tampered with. All the node in this network create consensus. They make agreement about which blocks are valid and which are not. Blocks that are tampered with will be rejected by other nodes in the network. To successfully tamper with a blockchain, it requires to tamper with all blocks on the chain, fulfill the proof-of-work for each block and take control of more than 50% number of nodes on the network so that the tampered block could become accepted. And it is almost impossible to do so. In the end, after everything is checked out, each node update their copy by adding this block to the end of the chain. The one who mines the block is called miner. The security of blockchain comes from its creative use of hasing, proof-of-work mechanism and the consensus among all nodes in the peer-to-peer network.

In cryptocurrency, the functionality of blockchain helps to prevent the double spending. If a person want to spend his or her money or a coin twice, they would do a double-spending transactions. One of the transaction is real, and the others are the tricks. But, each transaction needs an amount of time to be confirmed. So, there is no way to spend a same money ( or coin ) at the same time.

Let take a closer look into one of the biggest cryptocurrency blockchain, Bitcoin, to investigate how the blockchain technology is applied. Bitcoin is a decentralized application for transfering Bitcoin, value. If someone one to send an amout of bitcoins to others, their bitcoins would go through Bitcoin blockchain network without any checking of intermediary. The power of Bitcoin blockchain comes from its properties:

* **Components in a block**

*+ A magic number*: it is a 4-byte-long arbitrary number that helps to identify the type of data supported by a chain. Each time a new block is created, the blockchain network will check its magic number whether that block is a BItcoin block or the others’ block.

*+ Blocksize:* 4-byte-long, which shows how long a block is with all the transactions.

*+ Header:* consisits of

***.****Version: The current version number of a* chain, which also relatively lets the user know the format of the block.

***.****Hash of the previous block:* 32 bytes. It is the hash of the Header from the previous block.

***.****Hash Merkel Root:* The root hash of merkel tree consisting of the transactions in the block.

***.****Time:*  Seconds (4-byte) from Unix Epoch (when the block was created). This is the number of seconds elapsed since the January 1, 1970.

***.****Bit target: Proof of Work algorithm target for this block.* This tell the miners the difficulty of the current block. The Bit target is automatically adjusted by the Bitcoin network to ensure that a new block would be mined at least every 10 minutes or longer.

***.****Nonce:* Counter used for the Proof of Work algorithm

*+ Transaction counter:* Show how many transactions a block has.

*+ Transactions:* a list of all transactions in a block.

* **Mining and Network Validating mechanism**

In the Bitcoin world, it takes approximately 10 minutes to validate a new block. Bitcoins transaction would be firstly put into a transaction pool. Before mining a new block, a node must verify the validation of the Proof-Of-Work of the current block. If it is valid, miners now starts constructing a candidate block by gatherung the transactions in the transaction pool. It removes the transactions which already presented in the previous block, if there are any. The mining software that is running on the node has the responsibility to create a proper Header for the candidate block. The mining process mainly find the nonce which is appended together with the block header to produce the output by running a SHA256 hash function. The output is then compared to the Bit target. If it is less than the target, the mining of this block is now done and the miner get the mining reward. If not, the nonce must be increased to recalculate the solution again. This calculation must be done until the solution is found. To find the solution which is less than the Bit target is challenging because it is impossible to predict what the nonce will be, mining also act as a proof that the miner worked to get a valid hash. Hence, that is why the result is called Proof-Of-Work. The block than transmits to all peers on the network. They need to validate the new block before propagating it to its peers. This is the parts where a dishonest miner can be found out. If there is any invalidation, the block will be unaccepted and the miner would have wasted his time and computing power. Validation requires:

* Block header hash is less than the target
* Block size is within acceptable limits
* Block timestamp is less than two hours in the future.
* The first transaction is a coinbase transaction ( and only the first )
* The coinbase transaction has a valid reward.
* All transactions within the blocks are valid ( also have a checklist on their own )

After successfully mining a new block, miner get a total of reward for mining block and transactions fees.

Blockchain technology is constantly evolving. The born of Smart Contracts is one of the recent developments in this field. They are very popular nowadays. The term “smart contract” was first used by Nick Szabo in 1997, long before Bitcoin. He wanted to use a distributed ledger to store contracts. Now, smart contracts are similar to contracts in real world. The only difference is that they are completely digital because they are actually programs stored inside a blockchain. Bitcoin coin can be seen as a blockchain smart contract in some way. The purpose of creating smart contracts is to rebuild a community exchange system that does not require a third-party such as bank, middle organizations, etc. Programmers can write a smart contract which contains conditions and put them into a blockchain. For example, a sender A want to send an amount of money to receiver B, A and B have to trust a middleman ( a bank ) to handle their money correctly, ensure transactions are valid, non-fraudulent and successful. Neither sender nor receiver has to pay transaction fee to the bank. With a smart contract, transactions can be validated if all conditions are reached and the money will be sent directly to the receiver account address within a micro transaction fee. Otherwise, transactions will not be confirmed. And because the smart contract are stored on a blockchain, everything is completely distributed. This technique could remove the role of the middlemen or a third-party, which means no one is in control of the someone’s trading properties, and make a trust-less system become possible.

Are smart contracts trustful ? Because they are stored on a blockchain they inherit some sophisticated properties. Smart contracts are immutable and distributed. Once a smart contract is created, it can never be changed. Hence, no one can go behind the back and edit the code of a smart contract. The output of a smart contract in a blockchain is validated by everyone on the network, which means nobody could force one smart contract to release a wanted output because other nodes on the network will spot this attempt and mark it as invalid. This ability is related to smart contract distributed property. Tampering with smart contracts becomes almost impossible. Smart contracts can be applied to many different things: on crowdfunding, automatic payments, processing claims of insurances, delivery payments and so on. Present, there are a handful of blockchains which support smart contracts, but the biggest one is Ethereum.

Bitcoin blockchain is not what this report is focusing on, Ethereum is the one. Programmers can push their program as a combination of smart contracts to the Ethereum blockchain and execute them. Ethereum has the same mechanism with Bitcoin. It also has its cryptocurrency named Ether.

\_What is ethereum ?

\_what are its characteristics ?

\_What are in the ethereum Block ?

\_What does a transaction contain ?

\_The workflows of pushing a transaction to a block and valid that block.

\_What make it more powerful than Bitcoin

\_ SmartContract and ERC20 token

**Geth:**

\_What is geth

\_How it works, how it connects ( communicate ) to the ethereum blockchain

**Metamask:**

\_What is metamask ?

\_What is the different between metamask and geth ?

Solidity and Remix IDE:

\_Ethereum Virtual Machine

\_Remix IDE with representative features. and some Solidity basic syntax.

**HTML:**

\_basic knowledge of HTML

**Javascript and Web3js:**

\_Introduce Javascript

\_describe in details Web3js : Functions and their return results.

**Summary**

.

# CHAPTER III

## METHODOLOGY

**Overview Of Methodology :**

Describe the workflows along with the reality

**Detail Of Methodology:**

A private blockchain

Solidity code compile

SmartContract Deployment

ERC223 token and ERC223 SmartContrarct

Dapp front-end

# CHAPTER IV

## IMPLEMENTATION

**Input/Output data:**

**Implementation settings:**

\_Install Programming environment( Ethereum, Geth, required module,…)

**Implementation steps:**

\_Running a private node

\_Mining some ether

\_Deploy SmartContract ( Customized Token, Charity SmartContract, )

\_Running Dapp from its front-end and using metamask to interact with it.

# CHAPTER V

## RESULT / DISCUSSION

Showing result

# CHAPTER VI

## CONCLUSION

**1. Summary**

Advantages and disadvantages of using blockchain technology .

**2.Future work:**

Some promising improvements.

# LIST OF REFERENCES

# APPENDICES