DECORRUPTING CHARITIES
-USING BLOCKCHAIN

Team :

Members:

Aastha Khare

Hardik Chawla

Guntas Singh

Alakh Singh Sethi

Mentor

Dr. P.S.Rana

Problem Statement

5. Pick any problem you see in your daily life & figure out how blockchain fits into it and why that problem should be addressed by blockchain. Give a solution using only blockchain or using blockchain in conjunction with any other technology of your choice (AI, Data Science, etc).

General Problem:

The report, titled 'Nothing to Lose (But Your Chains)', threw light on potential applications of blockchain technology for charities and social enterprises. Major problem of this era is the looming threat people face of their hard -earned money being donated without any transparency. This problem attracted our attention while going through "MILAP" fund raising. Hereby we plan to "decorrupt charities" using smart contracts and blockchain. This will help to shoot over the growth of charities by philanthropic organizations.

IDEA/SOLUTION/PROTOTYPE

The serious issue which charitable organizations as well the philanthropic organizations are facing in current scenario has its solution hidden in “GIVING UNCHAINED” I.e. giving digital currency. There are myriad benefits of giving a digital currency:

1. Transparency: blockchain technology offer a form of radical transparency that can overcome public skepticism and lack of trust
2. New way for charities to conveniently raise money: self-governing smart contracts offer new opportunities for business to embed philanthropy at their core and new ways for charities to raise money and address social platforms
3. IOT; underpinned by blockchain technology can lead to a world in which smart machines emerge as new; hyper rational donors I.e. AI PHILANOTHROPISTS
4. Use colored coins to make donations of intangible assets such as intellectual property.
5. Increasing trust in donations: blockchain removes third party and makes charity more trustable.

WHY BLOCKCHAIN?

Blockchain acts as public record of ownership and transactions; It exists as a public ledger that records ownership at any given point of time. Now why we used blockchain:

1. Trust (most precious commodity for NGO's): This keys down to two features

Decentralized nature of system I.e. no reliance on third parties hence we no longer have to trust bank or law firms. This is possible because blockchain is not owned by single party but rather maintained and owned by all users, who contribute their own computer power to perform cryptographic calculations necessary to maintain the ledger.

Security: this is possible because the cryptographic protocols used to maintain the blockchain are extremely strong and the sequential nature of chain means the transactions actually become more secure over time as the no of blocks in the chain grows.

2. Easy to maintain and cut the fund-raising cost of charity because no bank fee and blockchain itself is a public record.

TECHNOLOGY STACK:

It is the skeleton of the project and it is important because it influences the scalability of the product. Hence we have been very careful in choosing the right technology stack

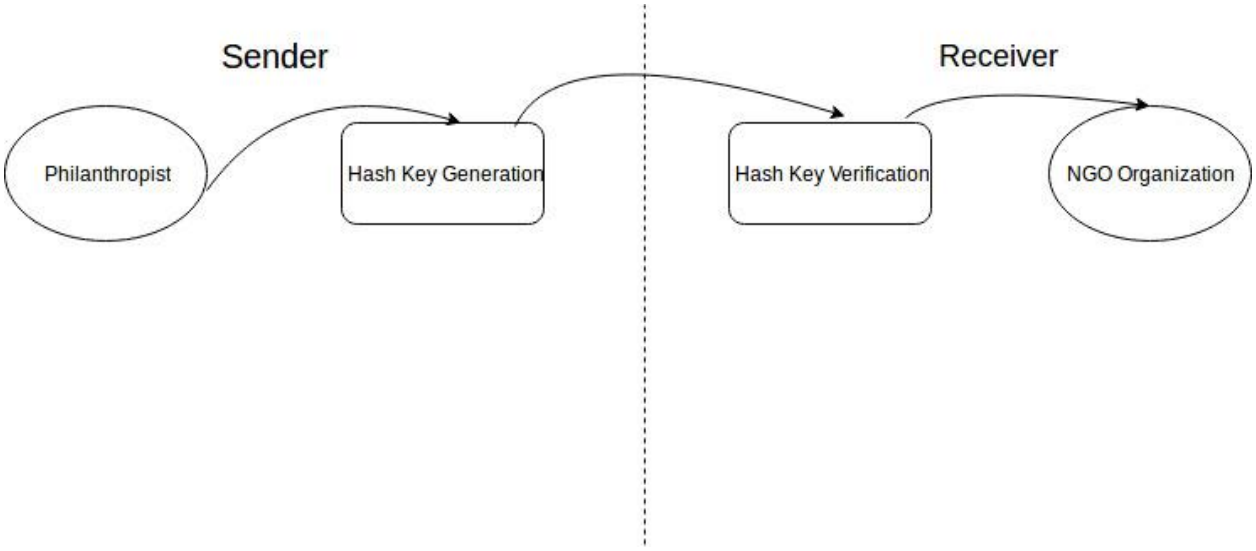
- CLIENT SIDE STACK

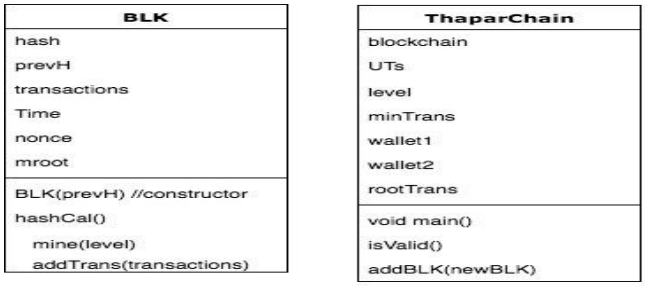
★ HTML
★ CSS
★ JAVASCRIPT
★ BOOTSTRAP

- SERVER SIDE STACK

★ JAVA
★ FLASK/PHP
★ MONGO DB
★ APACHE SERVER
★ PHUSION PASSENGER

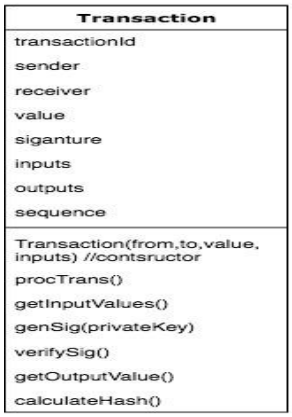
DATA FLOW:

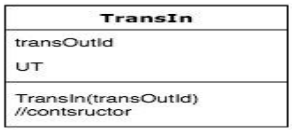

Sender
Receiver
Philanthropist
Hash Key Generation
Hash Key Verification
NGO Organization

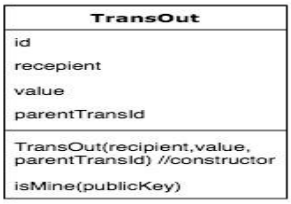

BLK
hash
prevH
transactions
Time
nonce
mroot
BLK(prevH) //constructor
hashCal()
mine(level)
addTrans(transactions)

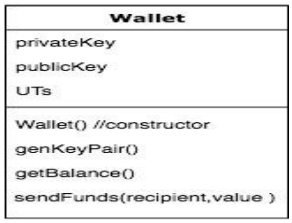
ThaparChain
blockchain
UTs
level
minTrans
wallet1
wallet2
rootTrans
void main()
isValid()
addBLK(newBLK)


StringUtil
applyESig(privateKey,input)
verifyEsig(publicKey,data,sig)
getDifString(level)
getStringFromKey(key)
gen_mroot(transactions)
applySha256(input)


Transaction
transactionId
sender
receiver
value
siganture
inputs
outputs
sequence
Transaction(from,to,value, inputs) //contsructor
procTrans()
getInputValues()
genSig(privateKey)
verifySig()
getOutputValue()
calculateHash()


TransIn
transOutId
UT
TransIn(transOutId) //contsructor


TransOut
id
recepient
value
parentTransId
TransOut(recipient,value, parentTransId) //constructor
isMine(publicKey)


Wallet
privateKey
publicKey
UTs
Wallet() //constructor
genKeyPair()
getBalance()
sendFunds(recipient,value)

hash-Hash of the current block calculated by the hashCal function

prevH-Hash of the previous Block

transactions-transactions for the particular Block

Time-The time of generation of the block (no of milliseconds from 1/1/1970

nonce-mined by the miners

mroot

BLK(prevH) //constructor-initializes prevH, Time and calls hashCal for hash

hashCal()-calculates new hash based on elements

mine(level)-calculates the nonce for the givel level of difficulty

addTrans(transactions)-add transaction to the block

ThaparChain-

blockchain

UTs

level-level of difficulty

minTrans

wallet1

wallet2

rootTrans

void main()-the main function

isValid()-checks the validity of the chain by comparing hash and prevH

addBLK(newBLK)-adds new block to the chain

StringUtil

applyESig(privateKey,i)-applies edsa signature and returns the result as bytes

verifyEsig(publicKey,data,sig)-verify the string signature

getDifString(level)-returns difficulty target string to compare with hash

getStringFromKey(key)-gets encoded string with base64

gen_mroot(transactions)-

applySha256(input)-apply Sha256 to a given string and returns the result

Transaction

transactionId-hash of transactions

sender-sender address public key

receiver-recipient address public key

value-amount to be sent

signature-preventing use by anyone else

inputs-arrayList of input transactions

outputs-arrayList of output transactions

sequence-rough count of the number of transactions generated

Transaction(from,to,value,inputs)//constructor

procTrans()-process transactions

getInputValues()-get the input transactions

getSig(privateKey)-generates signature using a private key

verifySig()-verify the signature

getOutputValue()-get the output transactions

calculateHash()-calcute hash

TransIn

transoutId-reference to TransOut->id

UT-contains unspent transaction output

TransIn(tranOutId)//constructor – copies tranOutId to transoutId

TransOut

id

recipient-new owner

value-amount of coins

parentTransId-the id of the transaction this was created in

TransOut(recipient,value,parentTransId)//contructor

isMine(publicKey)-check if coins belong to that wallet

Wallet

privateKey-to spend coins

publicKey-to receive money

UTs

Wallet() //constructor

genKeyPair()-generate the private and public key

getBalance()-shows the balance amount

sendFunds(recipient,value)-sending funds to another wallet

DEPENDENCIES/LIMITATIONS

Blockchain in charity is not a silver bullet to be fired off it has its own limitations.

1. Lack of understanding regarding governmental rules and regulations, regarding investing and receiving investment capital.
2. **I**t's Too New**:** concepts of Blockchain, cryptocurrency, and ICOs are still unknown to many charitable organizations, and certainly to NGO and philanthropists' who have money to invest but see this cryptocurrency "stuff" as somehow fake money. It will take time for the entire concept to become mainstream and for weaknesses to be eliminated.
3. Anonymity: blockchain uses digital signatures to prove identity which is traced down to cryptographic identities which are theoretically anonymous; hence are concerns of criminal exploitations.

FUTURE ASPECTS

As we might expect, the major blockchain innovations in the charity sector are currently being led by those at the very top (international organizations like UNO and countries like USA and UK can monitor the use of their aid inn corruption prone countries). However, in the future, we can expect to see a trickle down of such technologies, with the potential to change how we interact with (and donate to) charities, enabling charities and organizations to deliver remarkable new projects. Once the smaller organizations embrace blockchain the impact will be extraordinary.

Though currently it, may seem as a science fiction but in future it could be used as:

1.AI philanthropists

2.Donation through clauses in smart contracts

3.Donations of intellectual properties

4.use of blockchain by international organizations to monitor the aids provided to the corruption prone countries.