Secured Ethereum Transactions using Smart Contracts & Solidity

Aarush Kumar
School of Computing Science and Engineering (B.Tech CSE(Hons.))
Galgotias University
Greater Noida, Uttar Pradesh, India
aarushkumar100616@gmail.com

Sauranh Kumar
School of Computing Science and Engineering (B.Tech CSE)
Galgotias University
Greater Noida, Uttar Pradesh, India
sau29@gmail.com

Abstract—
Today cryptocurrency, like Bitcoin, is gaining popularity in the middle people and we cannot deny its role in the global economy. Although cryptocurrency does not instead of fiat money in the near future, it will take a significant part of the market. Despite the fact that it has no special laws in many countries yet, there are already ways people can use cryptocurrency. While the crypto markets have been banned in China, Japan, on the contrary, officially allows people to pay differently services or goods via Bitcoin. Currently, there are a large number of different cryptocurrencies but this thesis describes a service that works with Ether (Eth) in its blockchain - Ethereum. In contrast Bitcoin, the Ethereum blockchain is not just a cryptocurrency support tool for it, it is a a powerful platform dedicated to a country that brings several development opportunities: new markets, loans, registrations and other resources. This makes Ethereum a good replacement by known payment methods. However, the current method of transmitting eth is still difficult for ordinary people and requires a certain technical background. This is one of the current issues for Ethereum platform. The process of creative creation is very complex moment; therefore, most people do not understand or are afraid to use it. That is slow reduces the process of eth integration in society and makes it less enjoyable to compare in standard payment systems. In addition, there is another important factor to consider - risk of loss of property. In order to create a job, the sender must know public address, 42-character long hexadecimal character unit, recipient. If any a error in any characters, money will be sent to the wrong address and can not to be downloaded. As the blockchain network has no centralized management in the system, there are no conditions that can help recover the lost money. The platform on which the service operates is called Ethereum. It can be defined as a a world-class powerful machine that uses different systems - smart contracts - using a custom blockchain. These applications work as planned and cannot checked, closed, rigged or interrupted. Created by Vitalik Buterin in 2015 an open source forum helps to streamline Blockchain (BC) technology process. integration. Therefore, it attracts interests from a variety of sources and major one's companies such as Microsoft and IBM. The Ethereum platform can

be used in a variety of business or financial sectors. It ensures security and prevents any intrusion into the system. Companies and services based on Ethereum they can do business with other companies and services that they do not knowing without the risk of fraud. Ethereum allows you to register any type of trade for any asset without the need to use trial. This makes it easier compared to current methods of trade registration.

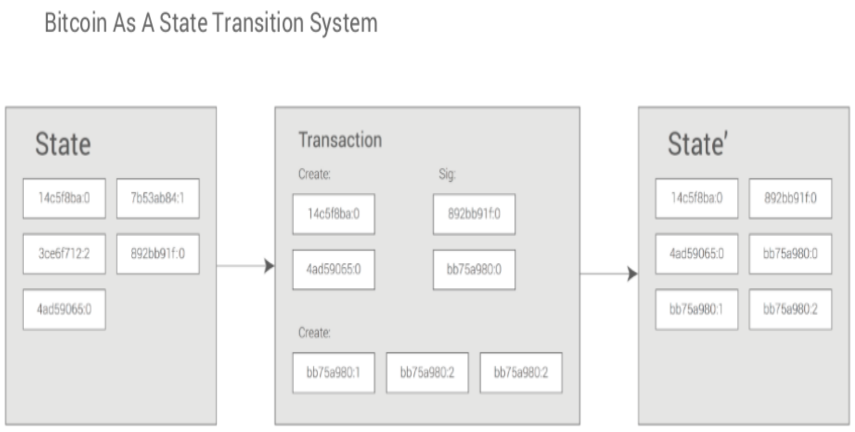
Keywords: Ethereum, Solidity, Smart-contracts, Blockchain, Ethers, State Transition System, Merkle Trees.

LIST OF ABBREVIATIONS: BC: Blockchain, Ether, EVM: Ethereum Virtual Machine

I. INTRODUCTION

The concept of digital currency shared with the country, as well as other applications such as real estate registrations, it has been in existence for decades. Anonymous e-cash agreements of the 1980s and 1990s, in particular reliance on cryptographic primitive known as Charmian blinding, provided high quality privacy, but protocols have largely failed to find trends due to their reliance on one site mediator. In 1998, Wei Dai's b-money became the first proposal to introduce a creative idea. money by solving computer puzzles and contextual compromises, but a suggestion there was little detail on how to make a national agreement. In 2005, Hal Finney introduced the concept of "practical proof of work", a program that uses ideas from b-money and Adam Back's difficult Hash cash puzzles to form the concept of a cryptocurrency, but he also failed miserably by relying on a reliable computer as a backend. Because the currency is the first application to the file, where the order of operations is often critical Importantly, allocated funds need a solution to agree on spatial distribution. Large roadblock that all pre-Bitcoin currency deals are subject to the fact that, although there has been a lot of research to create secure Byzantine consensus-tolerance systems for many years, everything The guidelines outlined were only part of the solution. Protocols assume that all participants in the system was known, and produced security genes in the form of "if N's participated, then i the system can tolerate cruel players who reach N / 4 ". However, the problem is that in an unknown area such safety genes are at risk of sybil attacks, when a single attacker creates thousands the nodes do not mimic a server or botnet and use these nodes to jointly protect a large part. The innovation offered by Satoshi is the idea of combining a very simple distribution consensus A protocol, based on nodes that combine transactions into a "block" every ten minutes creates an an ever-growing blockchain, with proof of function as a means by which nodes acquire rights participate in the program. While nodes have a large amount of integration power they have an equally powerful influence, which comes from the most powerful computer power in the entire network combined is much harder than copying a million nodes. Except for the Bitcoin blockchain model green and simple, has proven to be good enough, and will be so for the next five years the origin of more than two hundred currencies and protocols around the world.

Blockchain (BC), first created as a tool to support Bitcoin, has become independent technology used not only in the field of cryptocurrency. The process in the background BC is similar to peer-to-peer networks. BC stands for distributed book contains” account balances, reputation, trust plans, related data global knowledge; in short, anything that is currently represented by it computer is acceptable”- as stated in“Ethereum Yellow Paper ”. The ledger contains of blocks contains a copy of information about transactions or contracts, and so on it is constantly updated. Each block contains a cryptographic hash linked to the previous one block to build a series of blocks - Blockchain. Data held on Blockchain can be presented as a shared repository not only in one location, but in millions of computers at the same time. There is no single institution that makes it almost impossible relaxing information stored in BC. There are basically two main types of BC: public and private. Their main ideas are the same but the big difference is that the public BCs are completely open, while in order to to participate in private BC, the user needs to have permission from the creator or certification by a set of special rules. Both have advantages too inconsistencies and different areas to be used. For example, public BCs are numerous it is safe but requires a lot of integration power to maintain it to a large extent. BCs are secret they are usually business-oriented and do not provide access to information for non-consumers network participants. At the same time, the small size of such networks does are at high risk of hacker attacks. The next section examines BC again security.

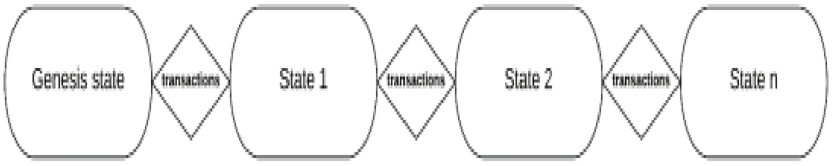


II. LITERATURE SURVEY

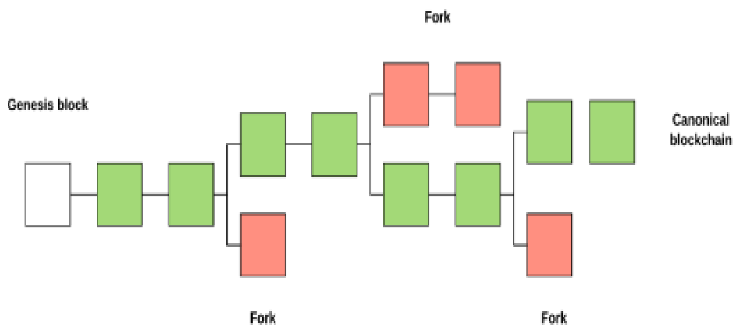
This clause describes the key concepts that lie behind the security of BC. There are two in them: secret fingerprints for each block and” agreement protocol ". Each block is signed with a unique cryptographic hash. To produce that hash computer power and miner time (miner is a special node of the BC network which enables network integration) a device is used. The miner secretly writes a block with a generated hash and adds it to the network. This hash stands for a confirmation that the block is valid. Hash generation requires a large amount of integration to solve a mathematical problem. Anyone can try to produce hash, but only the fastest miner wins the race. After that, the nodes are verified hash produced a new block, and this block has been added to the network. Verification the process, on the contrary, is easily compared to the hash generation, and does not take much energy and time. As soon as a new block was added, "as proof of work", the miner did produced the hash of this block is provided with a specially used digital token network (Bitcoin, Ether, etc.). The process listed above is called compliance Protocol. When a miner secretly writes a block with a hash, this hash closes the block. To break the mark, it is necessary to produce a new hash. Therefore, to change one record it is necessary to change the hash of the whole block. However, the hash also contains links to neighboring blocks. This means that to change the hash of a block, it is necessary to change the hashes of the following blocks again. In addition, each record within the block is protected by cryptography. Every participant of the network has a secret key, which is used to sign the transaction. In the event of a record changed, the signature becomes invalid. When this happens, the peer network is notified that something unusual happened. This allows for early entry on stage and prevent further damage. As BC is fragmented and distributed across the network to permanent peers updated, BC does not have a single point of failure. In other words, there is no central BC location, therefore, hacking such a network is necessary to gain access to everything for example, or at least 51 percent of the majority of the network. This means that large network, where at low risk. This is why private BCs are so vulnerable compared to social.

Ethereum BC can be described as a state-based marketing machine. Everything it begins with "genesis" - blank - state, empty and empty transactions in it. Whenever the action is added to the government, a new switch status occurs:

new status = old status + new transaction

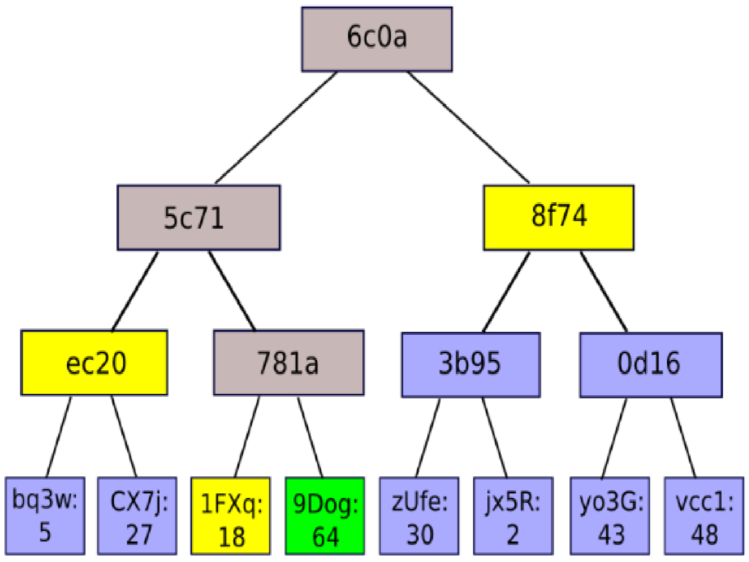


As mentioned earlier, new transactions can be added to the government only if they occur they are allowed. The verification process (mines) is performed by a group of existing nodes using their computer resources to verify mathematical transactions and perform a a new chain block. A valid block can be created by any miner and created instantly block added to chain. In a block post, the miner introduces i "statistical" proof that the block is valid. This raises the question: How do miners stick to the same series of blocks?
Since there is only one form of truth that everyone can accept, there will not be many chains. Otherwise, different chains may have different regions, and that leads to chaos. However, sometimes many regions are produced and are called “forks”.



In the event of a fork, the network nodes must determine which method is allowed. The decision was made using the “GHOST protocol”: The word "GHOST" stands for "heavily considered under a tree".

Ether is a cryptocurrency used by the Ethereum platform to pay for gas. Ether is produced whenever a valid new block is added to the network and delivered the node that created this block. Gas is a computational unit used throughout Ethereum regional change such as transactions and accounting services.The smart Ethereum contract also has an ether account, but it also has a code the part performed by EVM (Ethereum Virtual Machine) under certain circumstances. This part of the code can be defined as an editing class that contains various functions. If the tokens are transferred to a smart contract, it can automatically select the action to execute: to transfer money to one person or another, to make a return or to do something other. Wise contracts are written using different programming languages such as Solidity, Snake, LLL, etc. Smart contracts are created by planners and implemented Ethereum BC.EVM is a working time platform used by Ethereum blockchain for smart contract execution. It is separated from the main blockchain and, therefore, has no access to it network, accounts or processes, and has limited access to smart contracts. Whenever a code or function is generated by EVM, a certain amount gas is used. Gas price and gas price define a function creation is required to limit the amount of computer power EVM has to use in sequence to process this process. Therefore, the more consumers of gas use, the faster it is created work done.



The world of Global Ethereum represents the mapping of the regions of the accounts and their locations, addresses. The map contains the structure of the Merkle Patricia tree. The root of the tree horse consists of three elements: the root, the root horse and The "branch" of the tree. The branch contains all the hashes and creates a path from the chunk to the root. Every value stored in a tree is required to have a unique key that contains a link in neighboring blocks. When it is necessary to check information about a certain amount, this key tells you which path to follow to get the right block.

III. PROPOSED SYSTEM

After analyzing the current Ethereum transaction issues my team created i the idea of creating a tool that can simplify the work process for new users. At the same time making transactions secure. It was decided to build a web application with the potential to send ether directly to a the recipient's phone number required for public use fund address. In addition, the user gets the opportunity to cancel a job post using this web site application before tokens are accepted. The whole process must be protected therefore, in the event of a burglary, the stolen information cannot be used to accept the transfer tokens. To start the project three things have been developed: the web system, server and smart contract. The web application written in React, the JavaScript framework, provides its conclusion user. Controls all user input, generates recipient code as well transfers information to server. The Node JS server is used for SMS user verification. It also communicates with the wise agreement to withdraw tokens from the recipient's address. Smart Contract is written in the language of Solidity. This language is used for creative writing Ethereum network contract. The solution can be called a certified representative. The agent here means a smart contract used as a middle ground during work. Recipient confirmation by hosted server which is also used as the last data transfer point for the keystore.

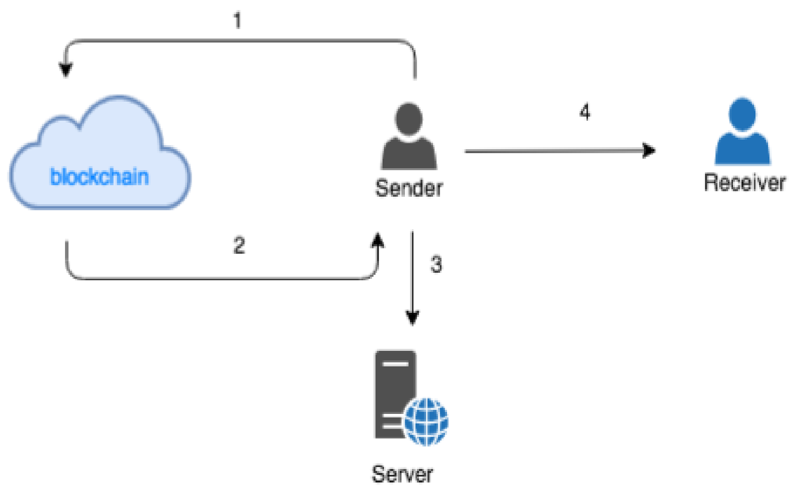
A. Back-end work flow

The back-end work flow can be divided into two parts: sending and receiving.

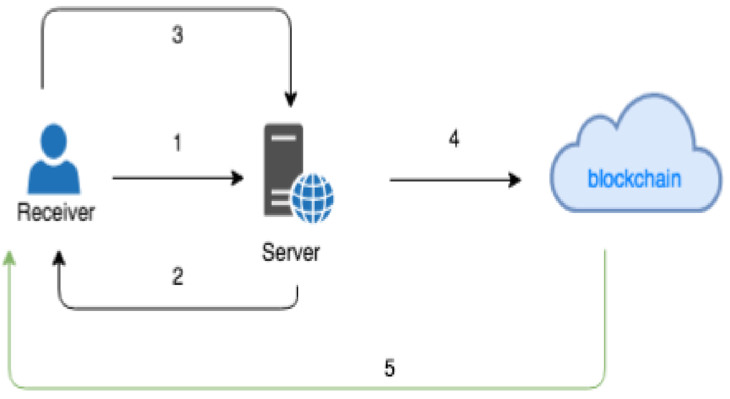
→ *Sending:*

First of all, private-public key pairs are generated on the sender device. Required signing job to get tokens. After that, ether is added to a smart contract, and the public key of the produced pair is given a deposit. Then a paired private key is used to create a randomly created keystore secret code. The secret code is transmitted by the sender to the recipient, the keystore - to server.

1. The sender produces a pair of private-public transport keys.
2. The sender enters the ether into a smart escrow contract and assigns transportation public deposit key. With the withdrawal of the escrow smart-contract confirms that the recipient's address is signed with the transit private key.
3. The sender encrypts the shipping secret key with a random and secret code sends a private transit key to the verification server.
4. The sender transmits the secret code to the recipient in the manner he or she chooses (voice, SMS, email, etc.)



→Receiving:
To get tokens the recipient needs to unlock the transit key pair. User gets keystore - secret pear-locked key originally created - from the server after being verified via SMS. Secret unlock code The keystore is passed on to the recipient by the sender. After unlocking the secret key, i The recipient's wallet address is signed by that secret key and returned to the server. The server then sends the request to a smart ether extraction contract. If all is well OK, ether is transmitted to the recipient's address.



1. The recipient types his or her phone number and secret code. Hash a phone verification request is sent to the server. (So not at any time in time the verification server has access to the verification private key.)
2. The server sends the verification code via SMS to the installed phone.
3. Recipient receives the code via SMS and type it. If the code is correct, the server returns the encrypted key store data to the recipient.
4. Recipient removes encryption keystore data with secret code provided by sender also receives a secret verification key. The recipient signs his or her address or his choice with a secret verification key. The recipient sends the signature address on the verification server.
5.Verification server attempts to withdraw ether via smart-contract to signed address. If the signature is correct, the transaction is repeated the receiver receives ether.

B. Security

The smart escrow contract installed on the Ethereum network cannot be hijacked due to high level of blockchain security (listed in Blockchain Security chapter). Therefore, it is a safe place to store tokens during the transfer. As all Ether transmissions made in BC, they are also unreliable. Despite the central server being used for phone verification and keystore transfer, at any point of the transfer process Verification server has no ether control locked in escrow smart Contract. Even if the verification server is in danger, in in the worst case, the recipient will not be able to receive the transfer. At the same time time, the sender can cancel the transfer by telephone to issue a smart contract and receive ether back.

In the event of a shared link being stolen, the ether cannot be retrieved without a phone verification: the the attacker must find a traffic keystore from the verification server in order to withdraw ether. In addition, the verification server does not store unripe phone numbers - instead to use a hash form produced by telephone number and salt:

phoneHash = hash(phone, salt)

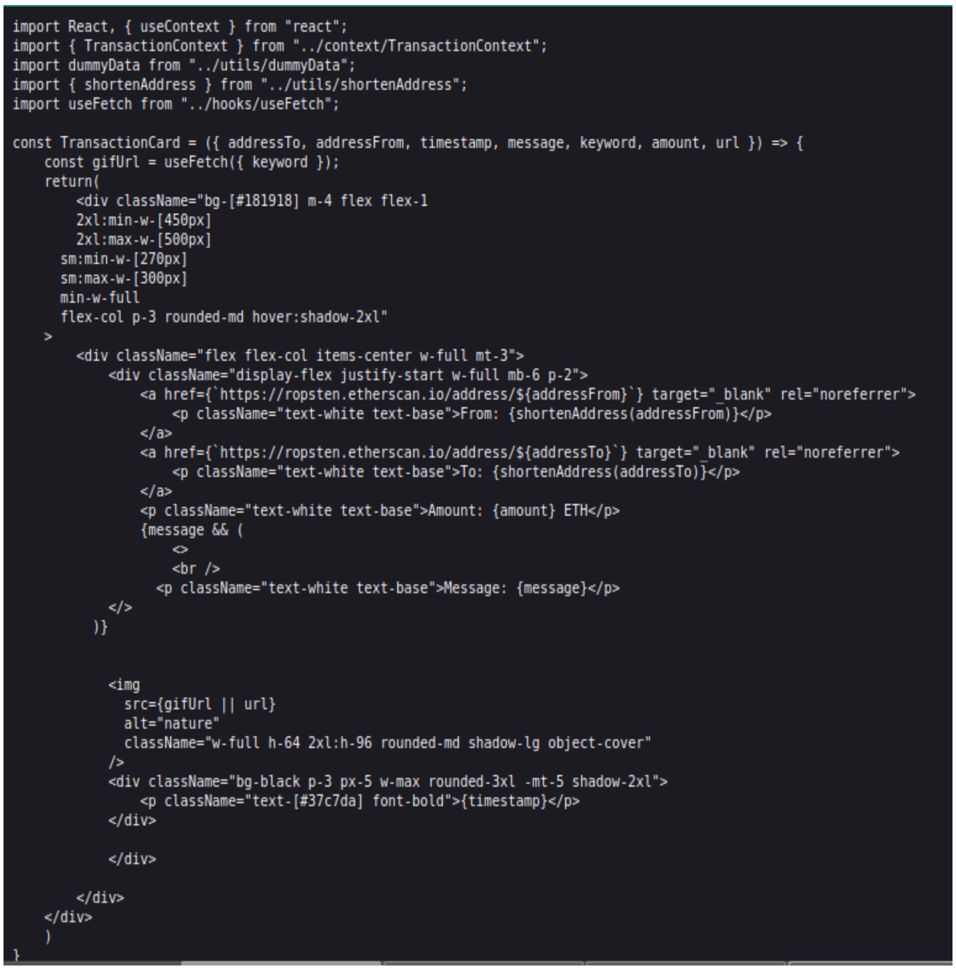
Salt is generated on the sender device:

salt = hash(transferId, code)

transferId = hash(phone, secret code)

C. Send transfer function code

The following code is used for sending transactions:



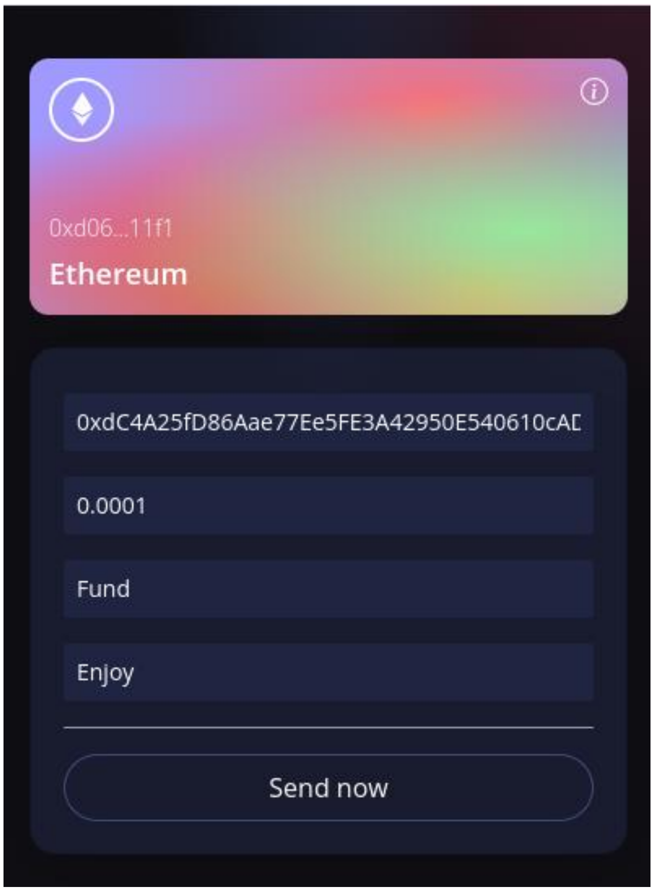
```
import React, { useContext } from "react";
import { TransactionContext } from "../context/TransactionContext";
import dummyData from "../utils/dummyData";
import { shortenAddress } from "../utils/shortenAddress";
import useFetch from "../hooks/useFetch";

const TransactionCard = ({ addressTo, addressFrom, timestamp, message, keyword, amount, url }) => {
    const gifUrl = useFetch({ keyword });
    return(
        <div className="bg-[#181918] m-4 flex flex-1
        2xl:min-w-[450px]
        2xl:max-w-[500px]
      sm:min-w-[270px]
      sm:max-w-[300px]
      min-w-full
      flex-col p-3 rounded-md hover:shadow-2xl"
    >
        <div className="flex flex-col items-center w-full mt-3">
            <div className="display-flex justify-start w-full mb-6 p-2">
                <a href={`https://ropsten.etherscan.io/address/${addressFrom}`} target="_blank" rel="noreferrer">
                    <p className="text-white text-base">From: {shortenAddress(addressFrom)}</p>
                </a>
                <a href={`https://ropsten.etherscan.io/address/${addressTo}`} target="_blank" rel="noreferrer">
                    <p className="text-white text-base">To: {shortenAddress(addressTo)}</p>
                </a>
                <p className="text-white text-base">Amount: {amount} ETH</p>
                {message && (
                    <>
                    <br />
                  <p className="text-white text-base">Message: {message}</p>
            </>
          )}


            <img
              src={gifUrl || url}
              alt="nature"
              className="w-full h-64 2xl:h-96 rounded-md shadow-lg object-cover"
            />
            <div className="bg-black p-3 px-5 w-max rounded-3xl -mt-5 shadow-2xl">
                <p className="text-[#37c7da] font-bold">{timestamp}</p>
            </div>

            </div>

        </div>
    </div>
    )
}
```

D. Front-end workflow:

The front-end workflow is also divided into two parts.

→ *Sending*

In order to use the web application the user needs a web3 wallet to be included: this could be a Metamask browser extension with any Ethereum wallet with a built-in web browser. From a user's point of view, the sending and receiving process is simple and straightforward. The first screen of the app has input fields for this unique wallet address the recipient and the amount of ether to be sent .



0xdC4A25fD86Aae77Ee5FE3A42950E540610cAE

0.0001

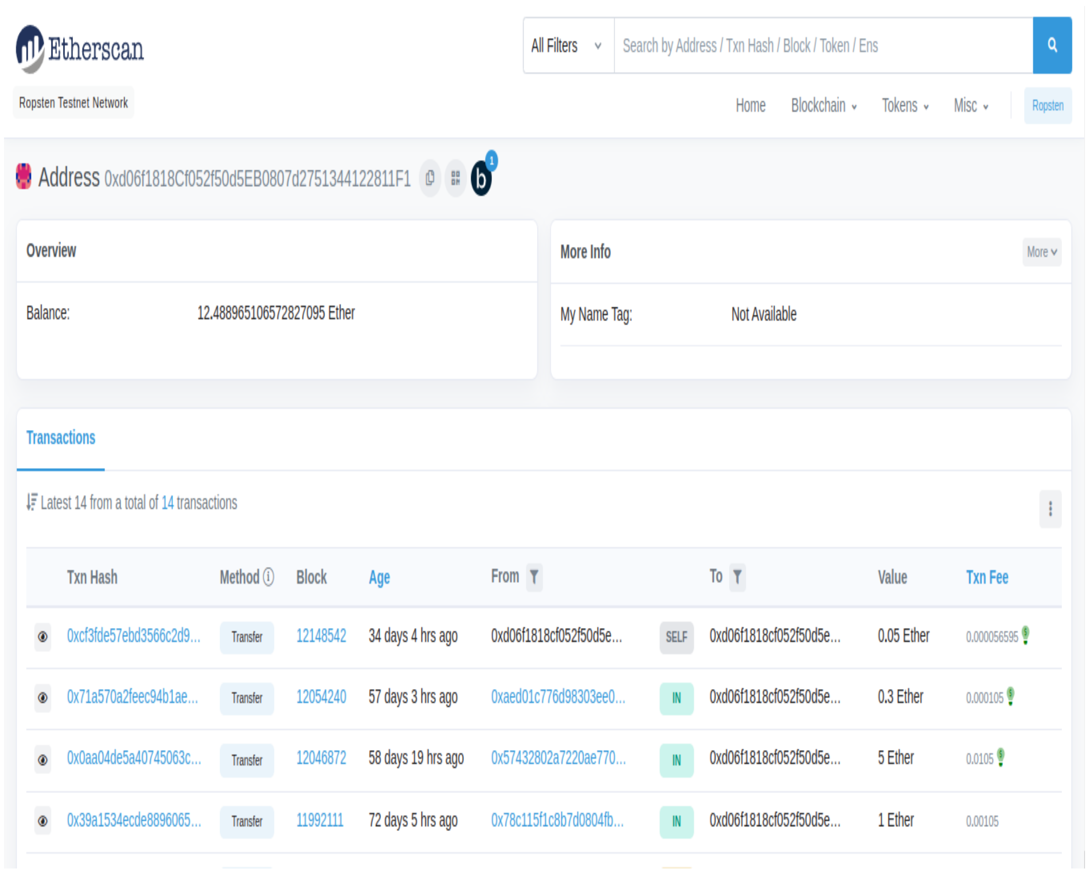
Fund

Enjoy

After clicking the "Submit" button the user needs to confirm the generated work wallet or Metamask browser extension. While the transaction is being sent to block for verification, the transaction bar screen screen shows the steps for this process.

The user can also check the performance process on the ethers can. When the action is confirmed the accept link is generated and displayed on screen. The user needs to send this link to the person who wants to receive the tokens. After clicking on that link, the recipient is redirected to the receiving screen.

The process can be re-evaluated on etherscan. The activity history screen contains a list of submitted and received tasks. Unrecognized activity can be canceled on this screen.



Address 0xd06f1818Cf052f50d5EB0807d2751344122811F1

Overview

Balance: 12.488965106572827095 Ether

More Info

My Name Tag: Not Available

Transactions

Latest 14 from a total of 14 transactions

Txn Hash	Method	Block	Age	From		To	Value	Txn Fee
0xcf3fde57ebd3566c2d9...	Transfer	12148542	34 days 4 hrs ago	0xd06f1818cf052f50d5e...	SELF	0xd06f1818cf052f50d5e...	0.05 Ether	0.000056595
0x71a570a2feec94b1ae...	Transfer	12054240	57 days 3 hrs ago	0xaed01c776d98303ee0...	IN	0xd06f1818cf052f50d5e...	0.3 Ether	0.000105
0x0aa04de5a40745063c...	Transfer	12046872	58 days 19 hrs ago	0x57432802a7220ae770...	IN	0xd06f1818cf052f50d5e...	5 Ether	0.0105
0x39a1534ecde8896065...	Transfer	11992111	72 days 5 hrs ago	0x78c115f1c8b7d0804fb...	IN	0xd06f1818cf052f50d5e...	1 Ether	0.00105

IV. Testing:

The testing phase was performed on the Ropsten testing network. It plays a key role Ethereum network but no need to pay for gas. This allows developers testing their smart contracts and creating free transactions. In addition, test cases were written to test coding.

V. CURRENT PROGRESS AND PLANS:

The first goal of the thesis was to create an application that was intended to simplify the process of ether transmission to the general public, thereby increasing the popularity of BC technology as well Ethereum platform.

After that, the development process had to be documented and provide information about the technology used in the project. Currently the active part of the project can be considered a success done. Already received constructive feedback and development ideas from Ethereum Community. The written text contains all the necessary information: detailed theater background and wrote the launch phase.

5.1.Current state of project:

Currently (May 2022) the web application is fully functional and accessible using the address WebApp Recent redesign has been successfully implemented. The service operates in two ways: main (main Ethereum network) and test (Ropsten network). To use the service, the user must have a web3 wallet (list of supported wallets can be found on the website). A trust fund is recommended: this wallet partnered with eth2phone service to perform the token process easy access to new users.

5.2.Future plans

The next step for the project is to create different service options that will allow sending eth via email address, twitter and other social media accounts. More than ether, the next version of the app will incorporate ERC20 tokens transmitted over the phone and other alternatives. Plans are also being made to build an independent mobile service application. Steps forward this direction has already been made: another team project is to make a transaction system for all cryptocurrencies.

REFERENCES:

[1] Ethereum platform website, https://www.ethereum.org, May 2022
[2] "Yellow Paper ': The Official Specification of Ethereum", https://github.com/ethereum/yellowpaper, quoted in May 2020
[3] Curtis Miles, "Blockchain Security: What Keeps Your Purchasing Data Safe?", Published on IBM (https://www.ibm.com/blogs/blockchain/2017/12/blockchain-security-what-keeps-your-transaction-data-safe /), December 2017
[4] Preeti Kasireddy, "How does Ethereum work, anyway?", Published in Medium (https://medium.com/@preethikasireddy/how-does-ethereum-work-anyway-22d1df506369), September 2017
[5] Preeti Kasireddy, "How does Ethereum work, anyway?", Published in Medium (https://medium.com/@preethikasireddy/how-does-ethereum-work-anyway-22d1df506369) , September 2017
[6] Vitalik Buterin, "Merkling in Ethereum", published on Ethereum blog (https://blog.ethereum.org/2015/11/15/merkling-in-ethereum/) , November 2015
[7] Etherscan Website, https://etherscan.io , May 2022
[8] Trust Fund website, https://trustwalletapp.com , May 2022
[9] Branch website, https://branch.io , May 2022
[10] Quid wallet website, http://quidwallet.com, May 2022