TECHNICAL WHITEPAPER

ySignAPP

Based on NEM Technology

Author:

ySign Team

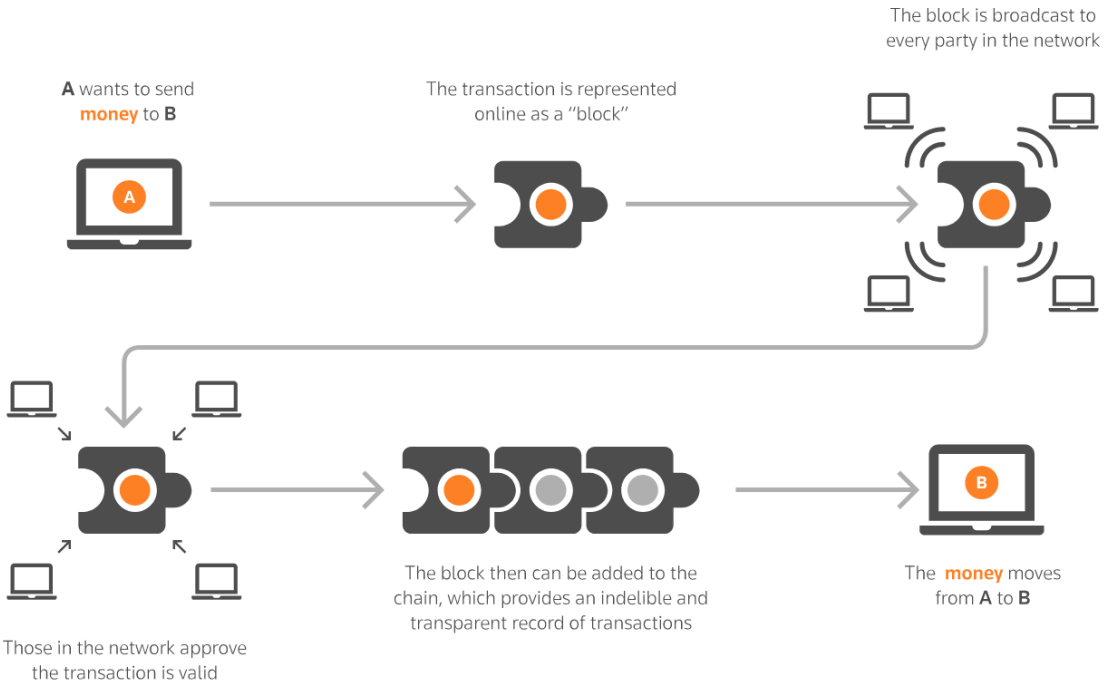
Version: V.0.5 – Date: 29.1.2018.

Contents

BACKGROUND AND SIGNIFICANCE OF BLOCKCHAIN

Before this time, TCP/IP protocol has been an important method used to transfer information globally. A significant development that transcends that spectrum was the development and introduction of the Interconnection technology known as the internet, as well as the virtual reality or augmented reality. These technologies have introduced more ways for people to interact and connect, which made more entities within the communication industry to become digitized and tokenized. Users would have more need for a P2P, open, decentralized, trustless communication platform. Users have often been in search for a communication platform that doesn't reveal their identity. People have often sought that freedom in online communication that meet the peer-to-peer protocol. One of the past peer-to-peer solution where value and asset can be transferred without the involvement of third parties was the Bitcoin network. With the communication industry going global and getting connected, an increasing number of companies leverage the power of the internet to find new customers worldwide. Majority of the traditional communication service providers are actively creating and innovating around the web to provide more appealing solutions. However, the central, slow, trust-based protocol peculiar to online communication is something worrisome and requires urgent attention especially as we anticipate the boom in the industry come 2021. If the industry must attain the expected boom, then it must adapt to the internet 3.0 which, in other words, is adopting the blockchain protocol. As a result, the ySign team which is currently developing a platform that will create a paradigm shift in the

online communication industry is set to bridge the gap identified above. After the ICO, ySign will fully implement and run an online based communication platform that will be powered by the blockchain system. Satoshi Nakamoto announced the Bitcoin Whitepaper on the 31st of October 2008 titled “Bitcoin, A Peer to Peer Electronic Cash System” where he/she/they introduced a decentralised system of value transfer. Every participant of the network is a relevant stakeholder; whereby value is transferred between two parties without necessarily creating a trust relationship. The technology has changed the way value is obtained and shared with the help of a decentralised, peer to-peer powered system.



BLOCKCHAIN REVOLUTION

We are proud to become the enablers of change in this industry by building the next generation of communication app that focuses on the end user and not the service provider. Even though we will be providing the platform on a smart contract with other third-party developers to leverage on, the end goal is to ensure a decentralised system where any barrier and central authority is eliminated, and where users would enjoy a fast and secure online communication. With the act of creating a new blockchain, based on the Ethereum protocol, with custom features to meet nowadays communication apps requirements, we believe we can remove the risk inherent in the current system by making it decentralized, transparent and trustless.

The ySign team introduces a platform that is borderless, secure, and fast for online communication powered by the blockchain technology.

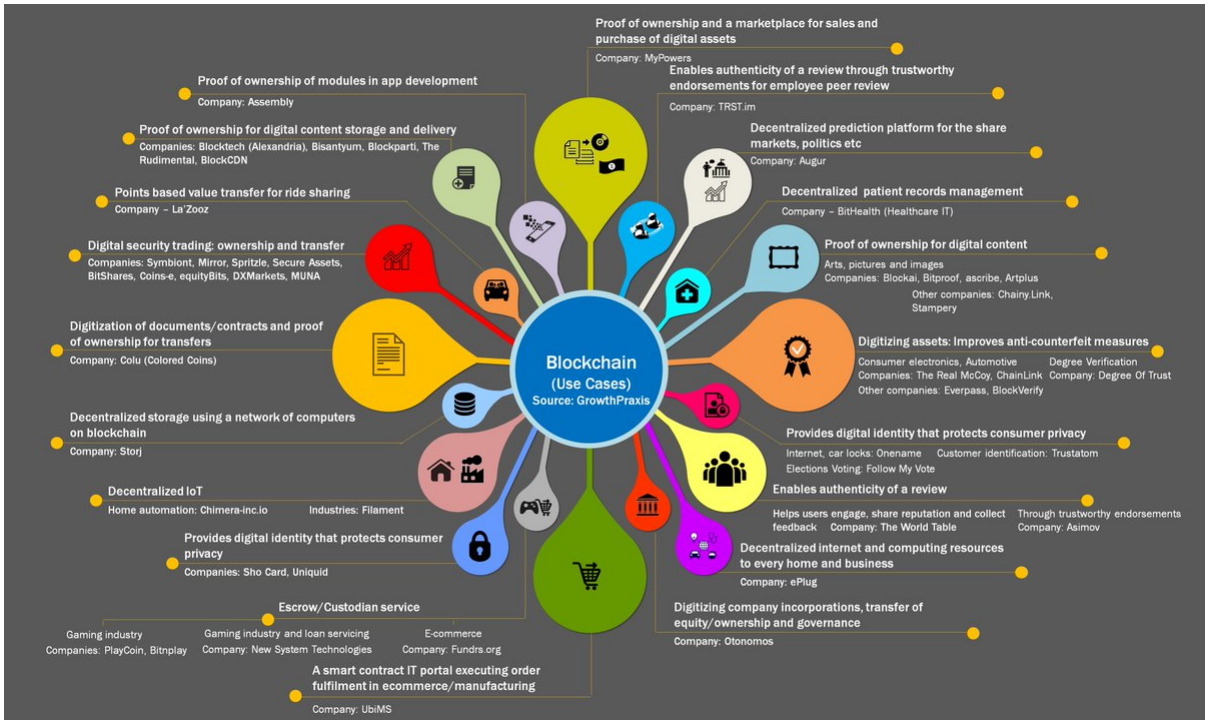
A Trustless platform: You don't need to trust the system, everything is written in Smart Contract. A Privacy-Protected Platform: You don't need to share any personal information. What you need is just a wallet.

An Anonymous Platform: You are hidden from the internet, our platform is only interested in your optimal satisfaction rather than focusing on knowing your name, your favorites and so on.

A Reliable Platform: Thanks to Blockchain technology and our decentralised systems, you can use it whenever you want and cease the worry about corruption or trusting a third party

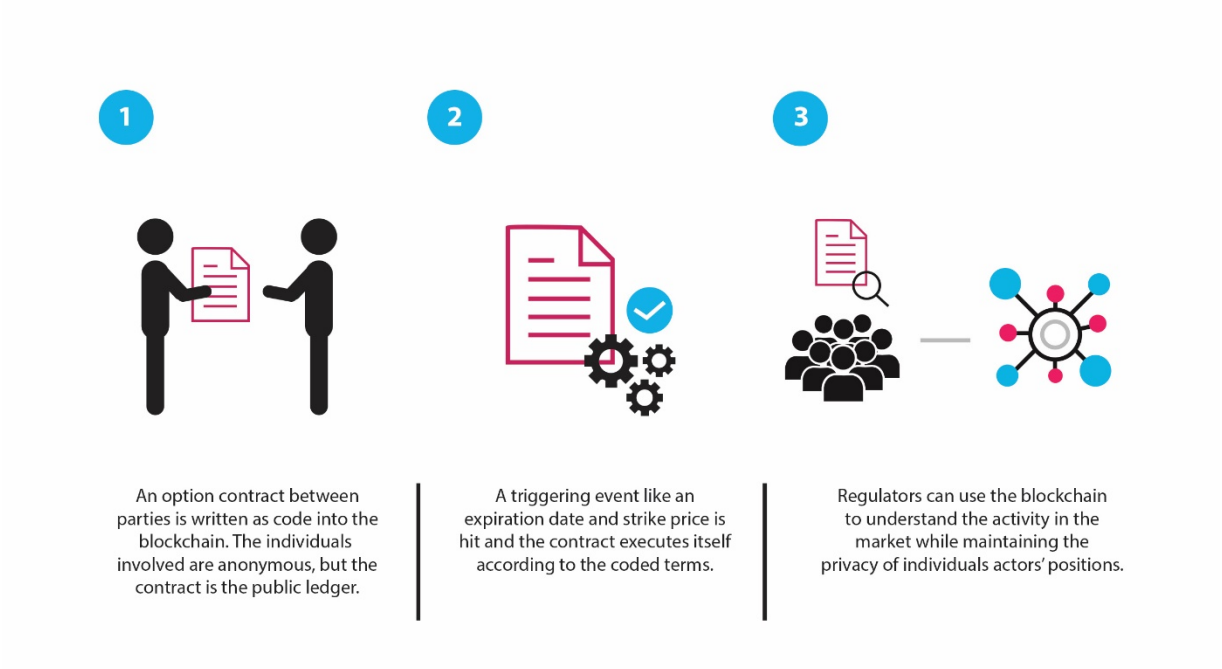
A Secure Platform: Online chat with ySign is secure. This means that no one has prior knowledge or access to your online wallet.

A Fast Platform: Through hard work, we've created an optimized platform that ensures fast content and service delivery anytime.



WHAT ARE SMART CONTRACTS?

One of the best things about the blockchain is that, because it is a decentralized system that exists between all permitted parties, there's no need to pay intermediaries (Middlemen) and it saves you time and conflict. Blockchains have their problems, but they are rated, undeniably, faster, cheaper, and more secure than traditional systems, which is why banks and governments are turning to them.



The best way to describe smart contracts is to compare the technology to a vending machine. Ordinarily, you would go to a lawyer or a notary, pay them, and wait while you get the document. With smart contracts, you simply drop a bitcoin into the vending machine (i.e. ledger), and your escrow, driver's license, or whatever drops into your account. More so, smart contracts not only define the rules and penalties around an agreement in the

same way that a traditional contract does, but also automatically enforce those obligations.

```
/* Allow another contract to spend some tokens in your behalf */
function approve(address _spender, uint256 _value)
    returns (bool success) {
    allowance[msg.sender][_spender] = _value;
    return true;
}

/* Approve and then comunicate the approved contract in a single tx */
function approveAndCall(address _spender, uint256 _value, bytes _extraData)
    returns (bool success) {
    tokenRecipient spender = tokenRecipient(_spender);
    if (approve(_spender, _value)) {
        spender.receiveApproval(msg.sender, _value, this, _extraData);
        return true;
    }
}

/* A contract attempts to get the coins */
function transferFrom(address _from, address _to, uint256 _value) returns (bool success) {
    if (balanceOf[_from] < _value) throw;                 // Check if the sender has enough
    if (balanceOf[_to] + _value < balanceOf[_to]) throw;  // Check for overflows
    if (_value > allowance[_from][msg.sender]) throw;   // Check allowance
    balanceOf[_from] -= _value;                           // Subtract from the sender
    balanceOf[_to] += _value;                             // Add the same to the recipient
    allowance[_from][msg.sender] -= _value;
    Transfer(_from, _to, _value);
    return true;
}

/* This unnamed function is called whenever someone tries to send ether to it */
function () {
    throw;     // Prevents accidental sending of ether
`
```

SMART CONTRACT

ySign is an Ethereum token. It complies with and extends ERC-20 - a de facto standard and widely used token API. The ySign Smart Contract guarantees:

Transparency

Balance. The information on the number of tokens held by any user is public.
Transfers. All information on transfers is public and can be traced back in time.

Ownership
Scope. Only Ethereum users and contracts can be token holders.
Uniqueness. Each token belongs to one user-owner. There are no shared tokens.

Token Supply
Single issuance. Tokens are issued only once, at the time of deployment.
Supply. The token supply is set at the time of deployment

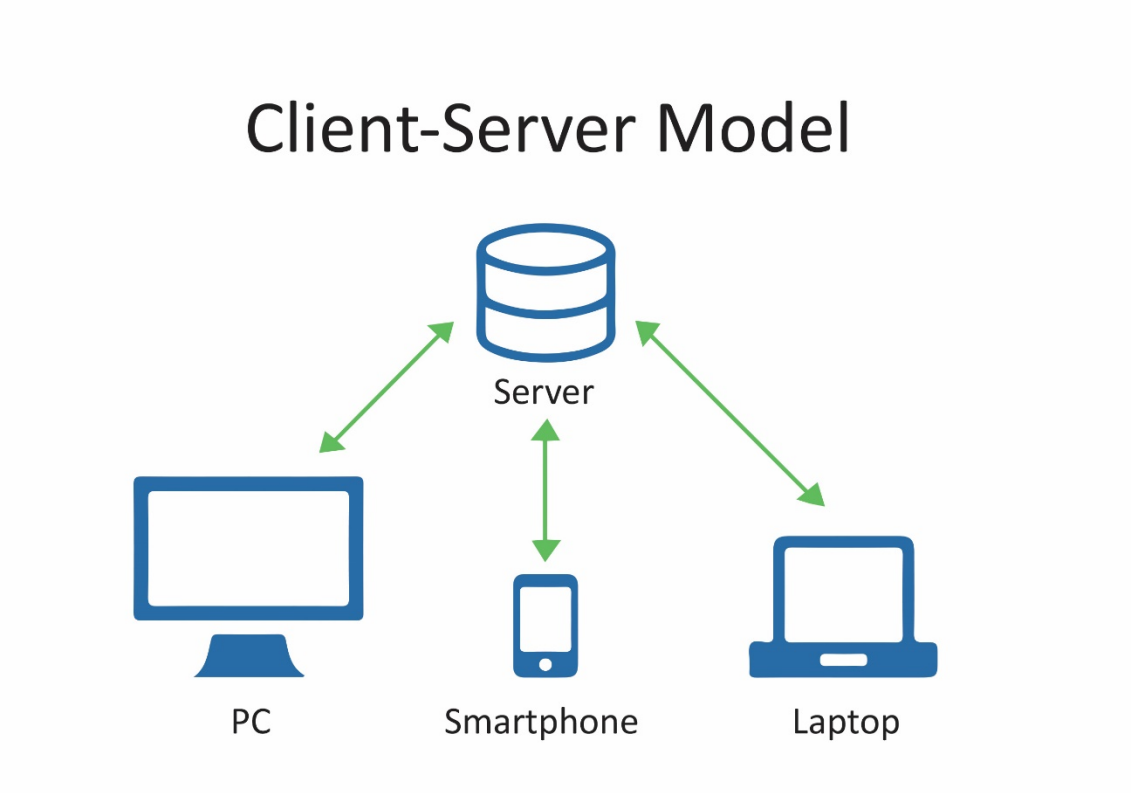
Contract Management
Replacement. The contract owner can relinquish the ownership in favor of any other Ethereum user or contract.

Miscellaneous
Recovery. Any call to the contract which results in an error does not change the users' tokens or Ether balance, except for the gas spent on the transaction.
Safety
No-overflowing. Our contract use Safe math that provide it with maximum safety of overflowing attacks.

```
function mul(uint256 a, uint256 b) internal pure returns (uint256) {
    if (a == 0) {
      return 0;
    }
    uint256 c = a * b;
    assert(c / a == b);
    return c;
  }

  /**
  * @dev Integer division of two numbers, truncating the quotient.
  */
  function div(uint256 a, uint256 b) internal pure returns (uint256) {
    // assert(b > 0); // Solidity automatically throws when dividing by 0
    uint256 c = a / b;
    // assert(a == b * c + a % b); // There is no case in which this
doesn't hold
    return c;
  }

  /**
  * @dev Substracts two numbers, throws on overflow (i.e. if subtrahend is
greater than minuend).
  */
  function sub(uint256 a, uint256 b) internal pure returns (uint256) {
    assert(b <= a);
    return a - b;
  }

  /**
  * @dev Adds two numbers, throws on overflow.
  */
  function add(uint256 a, uint256 b) internal pure returns (uint256) {
    uint256 c = a + b;
    assert(c >= a);
    return c;
  }
```

HOW IT WORKS

The transfer of messages in classic messengers is carried out according to the principle «client-server-client»



This technology undoubtedly has its advantages: quick search of users on a single database, storage of correspondence and synchronization between all devices of one user, service for transfer and storage of files, quick connection even if clients work through NAT. But in certain cases, the pros become cons. The storage of correspondence and data about users on the servers of the service jeopardize the confidentiality of this data. Data and correspondence can reach third parties as a result of server hacking, can be disclosed by the owners of the service at the request of government agencies or special services, and can simply be available to the employees of the service.

Another drawback of this architecture is a single point of failure - the server. If the server stops working for some reason, then the service also stops working.
The core of the ySign messenger is the concept of decentralized messaging. This concept lies in creating an instant messenger, which will ensure the anonymity of users, confidentiality of correspondence and the absence of a single point of failure.

WHY NEM BLOCKCHAIN?

Performance:

Blockchain technology offers a fundamentally streamlined method of maintaining a secure ledger of transactions compared to a traditional database.

How does NEM push blockchain performance further?

NEM's blockchain platform was designed and coded from the ground up for scale and speed. NEM's permissioned private blockchain delivers industry-leading transaction rates for internal ledgers. And its revolutionary consensus mechanismand the Supernode program ensure that NEM's open, public blockchain can grow without ever compromising throughput or stability.

Ease of Development:

Blockchain technology offers the potential to drastically simplify an enormous variety of secure ledger, transactional, and tracking systems compared with traditional databases.

How does NEM make blockchain development easier?

NEM's blockchain exposes its functionality through a powerful API interface that can be used with any programming language, not a specific "smart contract" language. Existing business logic code can easily mate up and use blockchain where it's strongest: secure transactions and ledger keeping. In short, configure NEM for your business, and then deploy blockchain incrementally and without forced retooling of existing infrastructure.

Deep Customization:

Blockchain technology is expanding to address many real world uses and applications beyond just "cryptocoins".

How does NEM customize blockchain to your needs?

Unlike other blockchain technologies, NEM is built from the ground up with powerful modular customization for virtually any application. We call it our Smart Asset system. With it, NEM lets you focus on building exactly what you need, whether that's a fintech system, tracking logistics, an ICO, document notarization, decentralized authentication, or much more.

Security:

Blockchain technology naturally provides a secure method of recorded transactions through its unique consensus-driven ledger concept. But every blockchain's implementation is different and has different weaknesses and vulnerabilities.

How does NEM make blockchain use even more secure?
NEM's architecture provides an incredibly secure and stable platform through its use of Eigentrust++ and an incentivized public node network based on its two-tier architecture. The significant risks inherent in on-blockchain "smart contracts" are eliminated by providing

building block customization to NEM functionality that keeps application security in your hands, not on the blockchain. And NEM's private blockchain option allows complete control over internal data privacy when a public blockchain isn't the right solution.

IMPLEMENTATION OF NEM BLOCKCHAIN IN ySignAPP

Our team will be developing IOS and Android apps with new type of messenger which gives user full privacy, without data leaking or any storing of your personal data. Your chat is only yours and the person you are messaging to. Also we will provide futures like calls, data sharing and video call. In this part we will demonstrate you some of our code which will make our app come to life.

Messages: To build our app 100% private messages will be stored on your device memory that means **no data is stored in any database on our server**. Every user will get his unique code, and find other people by their code. After finding user you want to message, you send messages just like any other messenger. For this we will use NEM secure message classes:

fromDecodedPayload

```
public static SecureMessage fromDecodedPayload(Account sender,
                                               Account recipient,
                                               byte[] payload)
```

Creates a new secure message around a decoded payload that should be encrypted.

Parameters:

`sender - The message sender.`

`recipient - The message recipient.`

`payload - The unencrypted payload.`

Returns:

`The secure message.`

fromEncodedPayload

```
public static SecureMessage fromEncodedPayload(Account sender,
                                               Account recipient,
                                               byte[] payload)
```

Creates a new secure message around an encoded payload that is already encrypted.

Parameters:

`sender - The message sender.`

`recipient - The message recipient.`

`payload - The encrypted payload.`

Returns:

`The secure message.`

- ***getEncodedPayload***

```
public byte[] getEncodedPayload()
```

Description copied from class: Message

Gets the encoded message payload.

Specified by:

getEncodedPayload in class Message

Returns:

The encoded message.

- ***getDecodedPayload***

```
public byte[] getDecodedPayload()
```

Description copied from class: Message

Gets the decoded message payload.

Specified by:

getDecodedPayload in class Message

Returns:

The decoded message.

To simple things up, we make your messages safe! And now this is how we do it for android and IOS

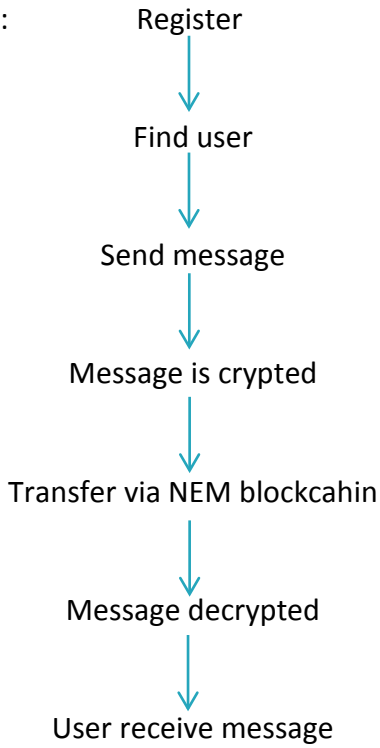
```
public void sendMessage(String message, String tos) throws XMPPException
        {
            System.out.print("in send messgae");
            chat.sendMessage(message);
            scrollar.getCorner(ScrollPaneConstants.LOWER_LEFT_CORNER);
        }


public void processMessage(Chat chat, Message msg) {
    System.out.println(Thread.currentThread());
    System.out.print("in the process message of newchat");
    if(msg.getType() == Message.Type.chat)
        System.out.println(chat.getParticipant() + " says: " +
msg.getBody());
    scrollar.getCorner(ScrollPaneConstants.LOWER_LEFT_CORNER);
    text.setForeground(Color.BLUE);
    text.append(chat.getParticipant() + ":  "  + msg.getBody() + "\n");

}
```

*Android version-java

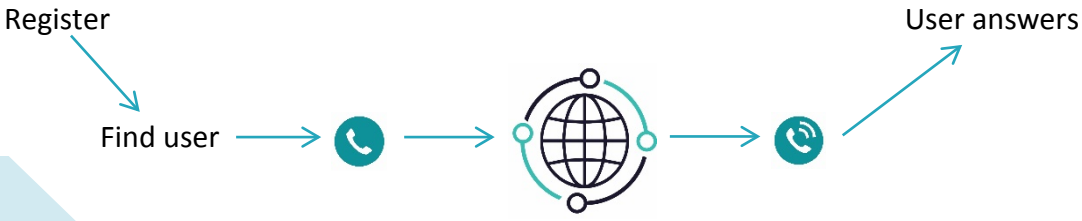
```
@IBOutlet var message: UITextField!
@IBAction func send(_ sender: Any) {

       if(message.hasText){
           postMessage(name: twitterHandle, message: message.text!)
       }
}
func postMessage(name: String, message: String){
        let params: Parameters = [
            "name": name,
            "text": message
        ]
        Alamofire.request(ChatViewController.MESSAGES_ENDPOINT, method:
.post, parameters: params).validate().responseJSON { response in
            switch response.result {
            case .success:
                print("Validation successful")
            case .failure(let error):
                print(error)
            }
        }
}
```

*IOS version-Swift

In other words it works like this:



Call: The process is almost the same as sending a message



NEM BLOCKCHAIN

UPCOMING UPDATES AND FEATURES

Data Sharing: We use secure blockchain P2P option to transfer your data.

Video call: the feature we are looking forward to the most.

Websites / References

ySign ICO

ICO Website:
https://www.ysign.io/

ySignAPP

APP Website:
https://www.ysign.online/

Sources / Links:

Nem Technical Reference:
https://www.nem.io/NEM_techRef.pdf