

Billion

A Blockchain-based Accounting Application

Billion

is a blockchain-based accounting application designed specifically for small to medium size charities and donors.

Industry Overview

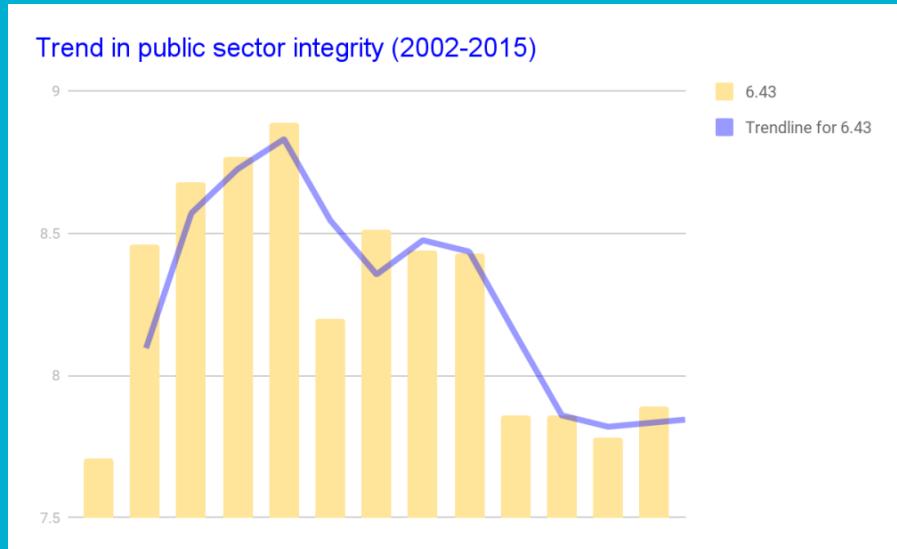
Nonprofit sector has a large market volume. The nonprofit sector contributed an estimated \$905.9 billion to the US economy in 2013, composing 5.4 percent of the country's gross domestic product (GDP). (Bureau of Economic Analysis)

Nonprofit sector also has a large user group. In 2014, 1.4 billion people worldwide donated money to nonprofit organizations. (Charities Aid Foundation)

The Issues

1. Integrity

The integrity of auditing in public sector is especially crucial as it measures how transparent public organizations process public budgets and funds.



The Issues

2. Professionalism

Normally crowd funding charities and small Kick starters do not have technical and financial abilities to conduct large-scale auditing. Some organizations rely on basic statistical software, such as Microsoft Excel and Google Drive, to manage their funds.



The Issues

3. Accountability

Due to limited tracking and tracing features of those applications, it is very difficult and time-consuming to find out the source of mistakes and frauds during auditing processes.



Billion

It is necessary for small charities to adopt a more advanced structure that is highly traceable and strictly accountable. Our team believe that blockchain technology is a good solution.





Blockchain

We believe that **blockchain technology** has congenital advantages to prevent fundraising frauds.

1. Blockchain prevents anyone from amending previous blocks. This eliminates unlawful changes before and after audit trail.
 2. Blockchain gives each participant a unique identity and such identity is not shareable. This allows participants to hold full accountability to their activities.
 3. blockchain technology is highly efficient and affordable. This lowers financial barriers that prevent small charities from upgrading their structures.
-



Vision

With a significant growing market size of small charities, we believe that our application of blockchain for charity, compared to traditional trusted authority, is more trustworthy by adding transparency towards all donators. More and more people participate in online kick starting events via social media these days. A peer-to-peer network will give public more clearance to supervise and engage.

Clients



Charities

which do not have financial ability of supporting a comprehensive accounting system that is highly transparent and accountable.



Non-profit Organizations

That are relying on unprofessional and unsafe applications to maintain their daily operations and finances.



Small Amount Donors

Who cannot comprehensively supervise the usage of their donations.



Partners

Billion is here to alleviate their frustrations by offering a highly transparent accounting application for all related fund management.

go-to-market strategy

Organizations

Recommendation

Social Media

Promotion

Users

Reference

Billion will be partner with target organizations to create a pre-launch testing period. This means that Billion will first secure several small charities and organizations which are willing to replace their current accounting system with us. The second step Billion will take is to get users through recommendations of organizations and users' references. Billion will also run various campaigns on social media.

Technical Specifications

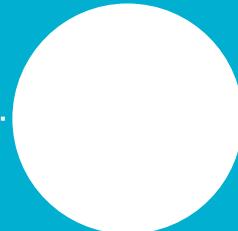
1

Python on Ethereum



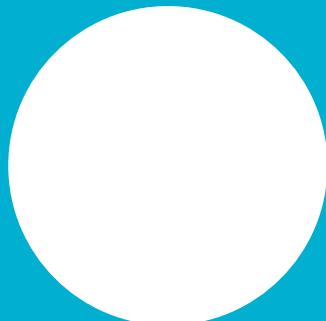
2

In Billion, a charity will be defined as a blockchain recording all information and progress of this charity. When the blockchain is created, it holds the information, encrypted by cryptographic hash function, about the sponsor, the content, the time, and the purpose of the charity.



3

Then, every donation on this charity and every usage of the funds will automatically update the blockchain. The blockchain is updated in discrete chunks called ‘blocks’, including information about donation or usage of funds, which are ‘chained’ together using hashes of the previous block contents.



UI Samples

Script Samples

```
1 contract BillionCoin {
2     uint256 public totalSupply;
3     mapping (address => uint256) public balanceOf;
4     mapping (address => mapping (address => uint256)) public allowance;
5     event Transfer(address indexed from, address indexed to, uint256 value);
6     ...
7     function BillionCoin( uint256 initialSupply ) { balanceOf[msg.sender] = initialSupply;    }
8     function _transfer(address _from, address _to, uint _value) internal {
9         require(_to != 0x0);
10        require(balanceOf[_from] >= _value);
11        require(balanceOf[_to] + _value > balanceOf[_to]);
12        uint previousBalances = balanceOf[_from] + balanceOf[_to];
13        balanceOf[_from] -= _value;
14        balanceOf[_to] += _value;
15        Transfer(_from, _to, _value);
16        assert(balanceOf[_from] + balanceOf[_to] == previousBalances);
17    }
18    function transfer(address _to, uint256 _value) public { _transfer(msg.sender, _to, _value); }
19    function transferFrom(address _from, address _to, uint256 _value) public returns (bool success) {
20        _transfer(_from, _to, _value);
21    }
22}
23
24
25 function Billion(address sendTo, uint goal, uint duration, uint etherCost, address reward ) {
26     beneficiary = sendTo;
27     fundingGoal = goal * 1 ether;
28     deadline = now + duration * 1 minutes;
29     price = etherCost * 1 ether;
30     tokenReward = token(reward);
31 }
```

```
1 interface token { function transfer(address receiver, uint amount);}
2 contract Billion{
3     address public beneficiary;
4     uint public fundingGoal;
5     uint public amountRaised;
6     uint public deadline;
7     uint public price;
8     token public tokenReward;
9     mapping(address => uint256) public balanceOf;
10    bool fundingGoalReached = false;
11    bool crowdsaleClosed = false;
12
13    event GoalReached(address beneficiary, uint amountRaised);
14    event FundTransfer(address backer, uint amount, bool isContribution);
15    function Billion(address sendTo, uint goal, uint duration, uint etherCost, address reward )
16        payable() {
17        modifier afterDeadline() { ... }
18        function checkGoalReached() afterDeadline { ... }
19        function safeWithdrawal() afterDeadline { ... }
20    }
21
22    function payable() {
23        require(!crowdsaleClosed);
24        uint amount = msg.value;
25        balanceOf[msg.sender] += amount;
26        amountRaised += amount;
27        tokenReward.transfer(msg.sender, amount / price);
28        FundTransfer(msg.sender, amount, true);
29    }
30}
```

Billion

We believe that Billion is a promising accounting application for next generation nonprofit fund management. Potential impacts of our solution include a significant increase of fund base and numbers of participating people. It also includes the increase of public trust of nonprofit organizations. Moreover, Billion helps the whole auditing and accounting industry to change.

Team

Fanyun Peng

pengf603@newschool.edu

Zexin Liao

16liao@gmail.com

Zexi Yuan

yuanzexi63@gmail.com

Tong Jin

jintong5@gmail.com

Xiaojie Yang

yangx066@newschool.edu