Application of Blockchain Technology in Civil Registration Systems

Vennis Shah, Karnika Padia, and Vivian Brian Lobo

Abstract Smart conurbation treatments are surfacing the route for urbanization, which indicates that government or administrative bodies, citizenries, and corporations need to incorporate smart solutions to expound systems that can implement a specific task in a disciplined way supplanting outmoded systems. Civil registration systems are one of the outmoded systems that can be made straightforward through the application of blockchain. Transparency, immutability, protection, and confidentiality—being some of the conspicuous blockchain characteristics—make it a podium of choice for digitizing civil registration systems. This study aims to design a decentralized application (DApp), i.e., a government or an administrative gateway, to perform various activities of civil registration systems using blockchain. The designed system generates birth as well as death certificates by way of smart contracts on a permission less network using Ethereum—a blockchain-based distributed processing platform. Moreover, the paper presents DApp implementation accompanied by several frameworks such as Truffle, Ganache, MetaMask, Remix Editor, Solidity, and Web3.js. This designed system guarantees a simplified registration process and offers increased data transparency and effective record maintenance.

Keywords Blockchain · Civil registration · Decentralized application

V. Shah (✉) · K. Padia
Department of Information Technology, St. Francis Institute of Technology, Mumbai, India
e-mail: vennis.shah98@gmail.com

K. Padia
e-mail: kpadia05@gmail.com

V. B. Lobo
Department of Computer Engineering, St. John College of Engineering and Management, Palghar, Mumbai, India
e-mail: vbl2781991@gmail.com

D. Patel et al. (eds.), *IC-BCT 2019*, Blockchain Technologies,
https://doi.org/10.1007/978-981-15-4542-9_16

1 Introduction

Blockchain is an emergent record list, known as blocks, that is, linked using cryptography. Every single block comprises transaction data, timestamp, and cryptographic hash of a previous block [1]. Transaction entails receivers' public key and is signed by a sender using his/her private as well as public keys. A transaction is placed in a ledger—a list of all transactions—only after the signature of a legitimate user.

1.1 Characteristics of Blockchain

The main characteristics of blockchain are as follows:

- *Inalterable*: Once a transaction is completed, a block is appended to the chain and that blocks' content remains unchanged. In other words, this indicates high data security.
- *Falsifiable*: When a block is generated, previous blocks' hash is stored in current block—which can be referenced. In addition, the current block contains its own hash that can be used to connect to the next block. This means that every single block can be authenticated autonomously.
- *Distributed unanimity*: It facilitates recording of every single transaction and distributing it throughout the network. All users in the network can verify transactions and have a ledger's duplicate print. Modifications to the ledger are exhibited in all prints in a fraction of seconds or sometimes minutes. Security and precision of assets are cryptographically preserved by means of keys and digital signatures, which are controlled by users [2].

1.2 Classification of Network in Blockchain

- *Centralized network in blockchain*: Herein, all nodes are linked by a single node, and an organization that deploys the blockchain network can determine which node can connect to the network, thereby ensuring security. Nonetheless, data maintenance at a single node is problematic. Attacking one node is simpler than attacking numerous nodes instantaneously. Once an attack is performed on the central node, there is no substitute. This consecutively affects the complete network that can lead to data loss or addition of untruthful data (Fig. 1).
- *Decentralized network in blockchain*: Herein, ledgers are amassed at different locations; therefore, to attack a decentralized blockchain network, an attacker has to maintain a track of every single ledger. In decentralized network, a transaction can be added to a network only when all nodes verify and sign it using their digital signatures. Consequently, unethical ledger modification is practically impossible since legitimate nodes will oppose attacker's transaction, and the attacker's block will not be added to the ledger. This feature of decentralized network in

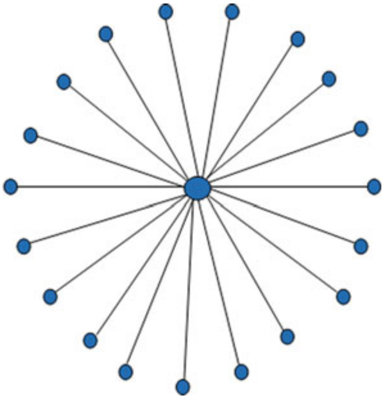


Fig. 1 Centralized network

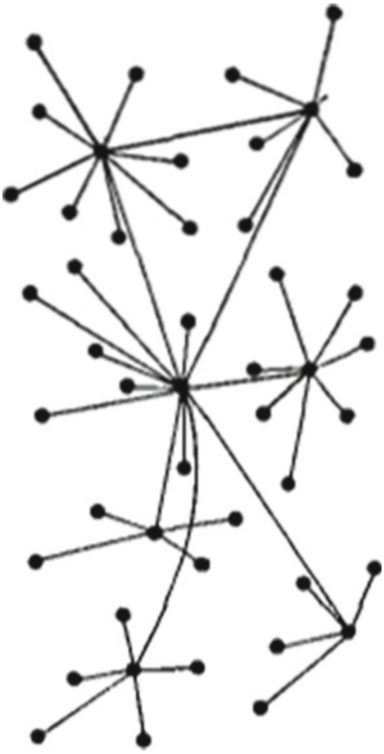


Fig. 2 Decentralized network

blockchain makes it much more secure when compared with centralized network in blockchain [3] (Fig. 2).

In this study, a decentralized smart contract mechanism is used.

- *Decentralized smart contract*: Ethereum is one of the most widely used smart contract platform to create decentralized smart contracts. Ethereum uses ether as its currency. A smart contract [4] represents the idea that legal contracts can be automatically notarized and executed. Solidity programming is the most favorable language that is used to develop smart contracts with Ethereum.

1.3 Confronts in Existing Civil Registration Systems

Absence of public consciousness and less requirement for civil registration papers, i.e., birth and death certificates, are one of the main confronts for low level of registrations. For instance, when an infant is born, parents need to visit a government organization to register an infant's birth. Conventional government paper work includes collecting a registration form, filling appropriate details, and submitting photocopies of required documents over and above carrying original documents for verification. Just in case there is an error, the entire wearisome procedure needs to be performed again, which ultimately leads to a troublesome task of registering an event in government records—thereby making the entire process inefficient and unproductive.

1.4 Resolution

A handful of problems in birth and death registrations can be put to an end by blockchain-based digitization. As mentioned above, blockchain proves to be inalterable, it provides true origin and conclusiveness, which makes it a platform of selection for digitizing birth and death records. Application of blockchain technology in civil registration systems can lessen parent's effort to an enormous extent. Rather than visiting government organizations and adhering to the conventional process, hospitals can give authorities to certain members to verify documents and enter details of an infant during birth. These details can be sent to government organizations (herein, nodes) through smart contracts. By means of transaction details and smart contract address, a government node can mine the block. Such a digital implementation of civil registrations will reduce manual documentation, decrease human errors, save time, keep data tamper-proof, and increase pellucidity between government organizations and hospitals. Similarly, death certificates and other registrations can be carried out without visiting government organizations.

2 Literature Review

Bayu et al. [5] studied up-to-the-minute applications related to blockchain and provided perceptions about blockchain and its associated terms. Friðrik et al. [6] assessed blockchain-as-a-service to execute distributed e-voting systems and proposed a blockchain-based e-voting system that discourses existing system limitations. Moreover, they assessed some blockchain backgrounds for constructing a blockchain-based e-voting system. Heng [7] discussed about blockchain applications in e-governance, its advantages, challenges, and other areas of application. Ahmed et al. [8] explained about blockchain components and its use cases in public sectors. Pinyaphat et al. [9] conducted a review on blockchain technology and determined its challenges. In addition, they provided an overview about Bitcoin. Few existing financial and non-financial blockchain applications were also discussed. Chengjun et al. [10] recapitulated existing blockchain technologies and elaborated the philosophies of designing and instigating secure distributed applications and analyzed security concerns. Peng et al. [11] focused on blockchain requirement in healthcare domain to resolve challenges such as gapped interactions, incompetent medical report distribution, and disjointed health documents. Wei et al. [4] reviewed the history of blockchain and clarified common definitions. Ting et al. [12] conducted a study on Ethereum by graphical examination to exemplify money transfer and smart contract creation and invocation. A cross-graph scrutiny facilitated them in addressing a few security issues in Ethereum.

3 Proposed System for Digitizing Civil Registrations

Figure 3 shows the block diagram of the proposed system for civil registrations that makes such systems uncomplicated and reachable to individual citizenries. Data will be accumulated in the form of blocks (i.e., smart contracts). This data can be straightforwardly mined by government organizations for record purposes. The flow of the proposed system can be described as follows:

- *For birth certificate registrations*:

  1. A citizen *(i.e., an infant)* is born at a hospital.
  2. Hospital authorities verify basic documents.
  3. The authorities login into the registration portal.
  4. Relevant information is placed into a smart contract and an event is registered.
  5. The block is mined by government organizations, which then generates a digital copy of birth certificate.

  Similar procedure is followed for death certificate registrations.

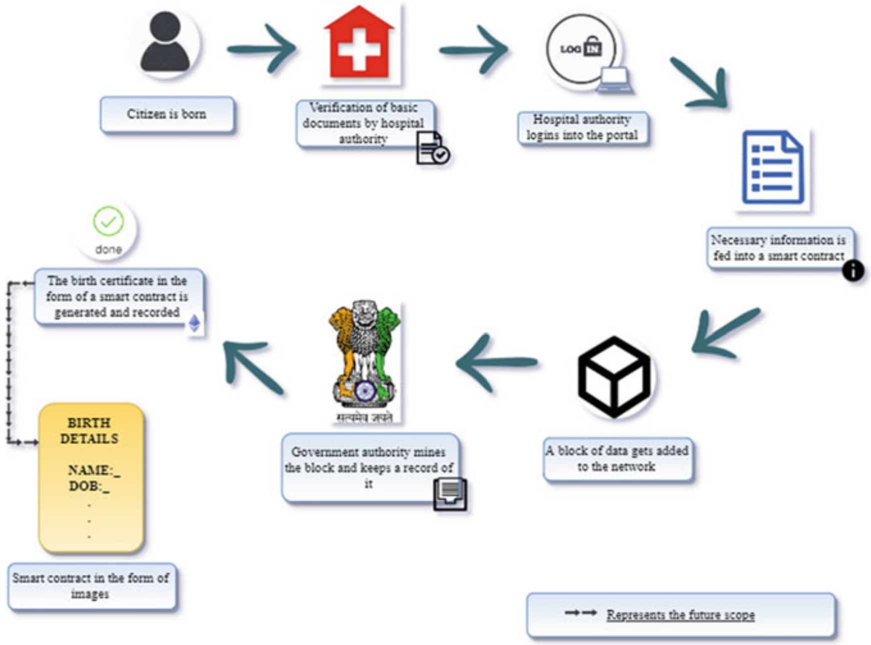

Citizen is born
Verification of basic documents by hospital authority
LOG IN
Hospital authority logins into the portal
Necessary information is fed into a smart contract
A block of data gets added to the network
सत्यमेव जयते
Government authority mines the block and keeps a record of it
done
The birth certificate in the form of a smart contract is generated and recorded
BIRTH DETAILS
NAME:_
DOB:_
Smart contract in the form of images
Represents the future scope

Fig. 3 Proposed system for civil registrations

4 System Design

For designing the proposed system, the following frameworks were used.

4.1 Truffle

It is used for estimating blockchain using Ethereum virtual machine. Some of its qualities include:

- *A migrations framework.*
- *Network supervision for deploying smart contracts to numerous public and private networks and a console for unproblematic contract communication.*

4.2 Ganache

It is a specific blockchain that is used for Ethereum progression. When Ganache is launched, a screen appears that displays server-related aspects. Ganache offers subsequent options, i.e., *accounts option* displays generated accounts and resultant account balances. *Blocks option* shows mined block factors accompanied by block number, amount of gas expended, and business transactions. *Transactions option* provides a list of transactions that are completed, and *logs option* enlists logs for the server.

4.3 MetaMask

It is a suspension bridge that permits a user to visit distributed Web of tomorrow in today's browser. MetaMask for diverse browsers can be operated to execute Ethereum decentralized applications (DApps) in a user's browser devoid of unambiguously running an entire Ethereum node. MetaMask consists of a tamper-proof identity crypt that offers a UI to uphold an individual's identity on a variety of sites and alphanumerically signed blockchain transactions. MetaMask add-on can be installed in Google Chrome, Mozilla Firefox, and Opera. Moreover, it provides a user the freedom to connect to a private or public network.

4.4 Remix Editor and Solidity

Smart contracts can be efficiently written in solidity using Remix tool. Remix Editor registers code and generates corresponding application binary interface (ABI) and byte code. With the help of Remix Editor, a user can deploy smart contracts to a particular address. Once the smart contract is deployed, the editor creates a smart contract address.

Solidity is a high-level programming language that is used for smart contract execution. Smart contracts help in understanding the behavior of Ethereum accounts.

4.5 Web3.js

Web3.js is a compendium of libraries, wherein one can communicate with both a local and remote Ethereum node using hypertext transfer protocol or interprocess communication connection.

5 System Implementation

```
pragma solidity ^0.4.2; //indicates the compiler version
contract Coursetro { //defining variables of type string
   string Name;
   string fName;
   string mName;
   string dob;
   string sex;
   string pob;
   string add
/*function to set the values of variables in the smart
contract*/
function setInstructor (string _Name, string _fName,
string _mName, string _dob, string _sex, string _pob,
string _add) public
{
     Name=_Name;
     fName = _fName;
     mName=_mName;
     dob=_dob;
     sex=_sex;
     pob=_pob;
     add=_add;
}
/*function to get the values of variables in the smart
Contract*/
function getInstructor () public constant returns
(string, string, string, string, string, string, string)
{
     return (Name, fName, mName, dob, sex, pob, add) ;}}
```

Figure 4 shows the home page of government portal and details that need to be filled in government portal for the process of civil registrations. From here, hospital authorities can either register themselves if they wish to get associated with government authorities or already registered hospitals can login to carry out further

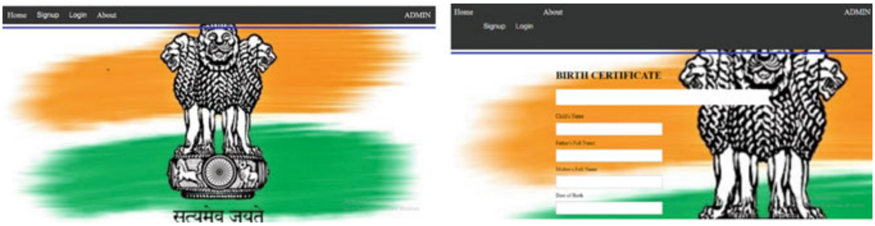


Fig. 4 Home page of government portal and details that need to be filled in government portal

processes. Also, it shows the details that need to be filled in government portal Web page where hospital authorities feed an infant's details, which in turn will be fed into a smart contract.

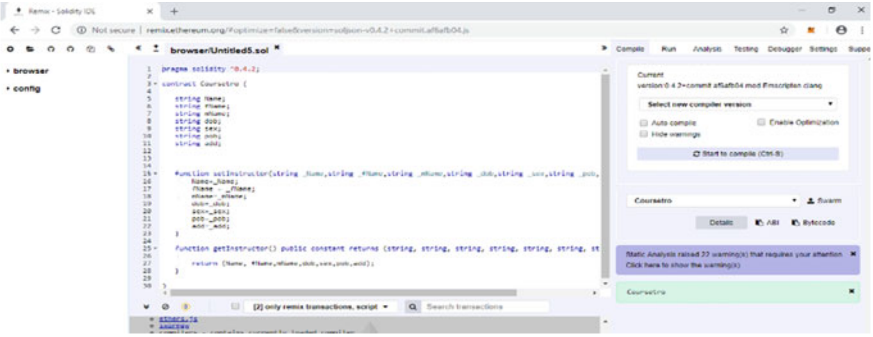
```
if (typeof web3 !== 'undefined') {
            web3 = new Web3(web3.currentProvider);
        } else {
        //set the provider you want from Web3.providers
web3 = new Web3(new Web3.providers.HttpProvider("http://localhost:7545"));
        }
web3.eth.defaultAccount = web3.eth.accounts [0];
//ABI stands for application binary interface
var CoursetroContract = web3.eth.contract([ABI]);
var Coursetro = CoursetroContract.at('0xc5ae4282b30402fbde783237d55532a599783870');
        console.log(Coursetro);
    Coursetro.getInstructor(function(error, result){
            if(!error)
                {
                    $("#instructor").html('Childs
Name:'+result[0]+','+'Fathers Name:'+result[1]/*+','+'Mothers name:'+result[2]+'Date of
birth:'+result[3]+','+'Sex:'+result[4]+','+'Place
ofbirth:'+result[5]+','+'Address:'+result[6]*/);
          console.log(result);
                }
            else
                console.error(error); });

        $("#button").click(function() {
       Coursetro.setInstructor($("#name").val(),
$("#ffn").val(),$("#mfn").val(), $("#dob").val(),
$("#sex").val(),$("#pob").val(),$("#add").val() );
        });
```

The above code is a smart contract code, which is implemented using solidity programming. This code is used for inserting values entered by hospital authorities into a smart contract using *setInstructor()* and is later fetched using *getInstructor()*. Solidity 0.4.2 compiler version is used.

Figure 5 shows the snapshot of remix editor, which is used to compile smart contract code. ABI is generated by the editor, which is data encoding strategy used in Ethereum to work with smart contracts. The associated byte code is generated.

The above code is the pseudo code for Web3.js. The contract address is also added in the contract. For example, in the above code snippet, the contract address is *0xc5ae4282b30402fbde783237d55532a599783870*. This piece of code is written inside the script tag in an HTML file for the birth certificate form.



```
pragma solidity ^0.4.2;

contract Coursetro {

    string Name;
    string fName;
    string mName;
    string dob;
    string sex;
    string pob;
    string add;



    function setInstructor(string _Name,string _fName,string _mName,string _dob,string _sex,string _pob,
        Name=_Name;
        fName = _fName;
        mName=_mName;
        dob=_dob;
        sex=_sex;
        pob=_pob;
        add=_add;
    }

    function getInstructor() public constant returns (string, string, string, string, string, string, st

        return (Name, fName,mName,dob,sex,pob,add);
    }

}
```

Current
version:0.4.2+commit.af6afb04.mod.Emscripten.clang

Coursetro

Static Analysis raised 22 warning(s) that requires your attention. Click here to show the warning(s)

Coursetro

Fig. 5 Snapshot of Remix Editor

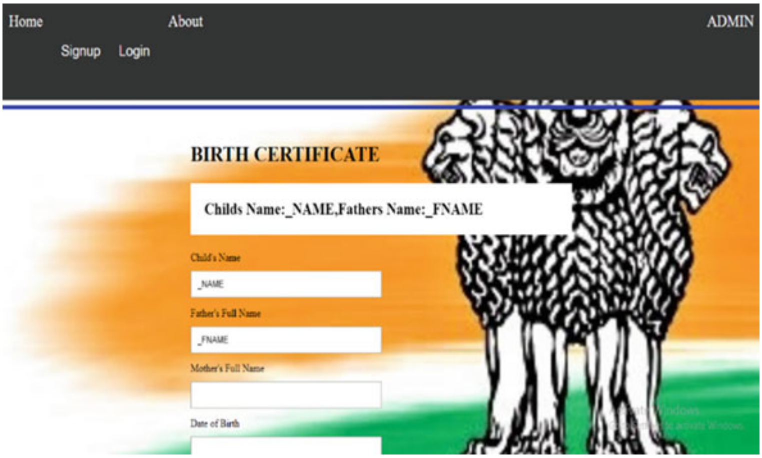
Figure 6 shows that the details that are fed into the smart contract are later fetched and displayed.

Figure 7 shows the snapshot of Ganache, which is a personal blockchain that can be used for Ethereum development. Ganache deploys smart contracts as well as a user can perform specific tests. Herein, ten accounts are created with 100.00 ETH each.

Figure 8 shows the addition of a custom network address http://localhost:7545 that is used by Ganache by means of custom remote procedure call option in MetaMask. Also, connectivity to http://localhost:7545, which is supported by Ganache is shown.

Figure 9 shows that by clicking on the account from Ganache, the account's private key can be obtained, which can be pasted in MetaMask to import this account in MetaMask. It also shows that the account is successfully imported

Figure 10 shows the snapshot of Ganache where list of blocks that are mined



BIRTH CERTIFICATE

Childs Name:_NAME,Fathers Name:_FNAME

Child's Name

_NAME

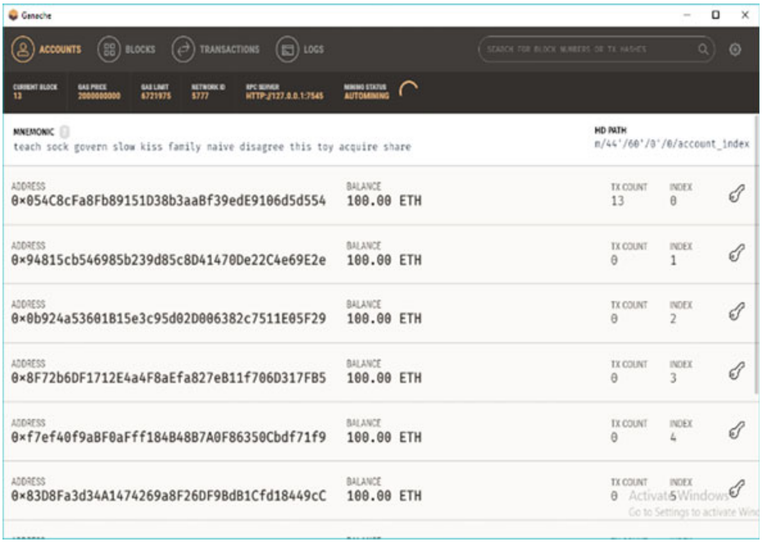
Father's Full Name

_FNAME

Mother's Full Name

Date of Birth

Fig. 6 Displaying entered values



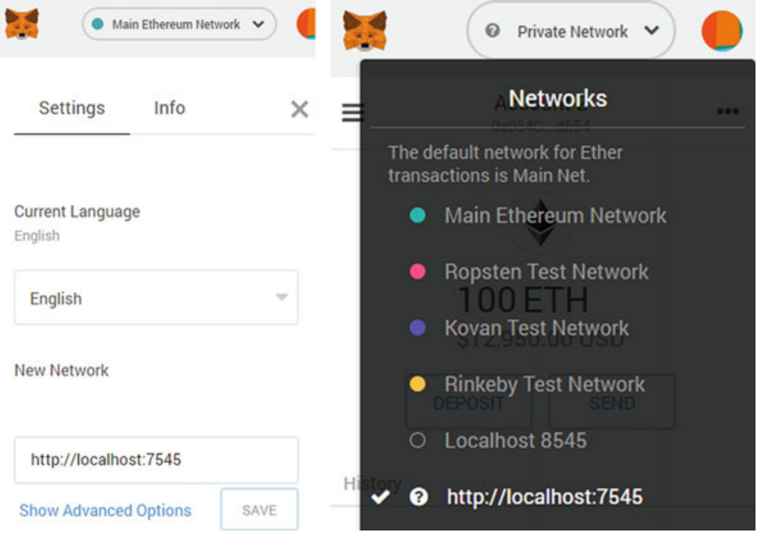
CURRENT BLOCK 13 | GAS PRICE 20000000000 | GAS LIMIT 6721975 | NETWORK ID 5777 | RPC SERVER HTTP://127.0.0.1:7545 | MINING STATUS AUTOMINING

MNEMONIC
teach sock govern slow kiss family naive disagree this toy acquire share

HD PATH
m/44'/60'/0'/0/account_index

ADDRESS	BALANCE	TX COUNT	INDEX
0×054C8cFa8Fb89151D38b3aaBf39edE9106d5d554	100.00 ETH	13	0
0×94815cb546985b239d85c8D41470De22C4e69E2e	100.00 ETH	0	1
0×0b924a53601B15e3c95d02D006382c7511E05F29	100.00 ETH	0	2
0×8F72b6DF1712E4a4F8aEfa827eB11f706D317FB5	100.00 ETH	0	3
0×f7ef40f9aBF0aFff184B48B7A0F86350Cbdf71f9	100.00 ETH	0	4
0×83D8Fa3d34A1474269a8F26DF9BdB1Cfd18449cC	100.00 ETH	0	5

Fig. 7 Snapshot of Ganache accounts



Current Language
English

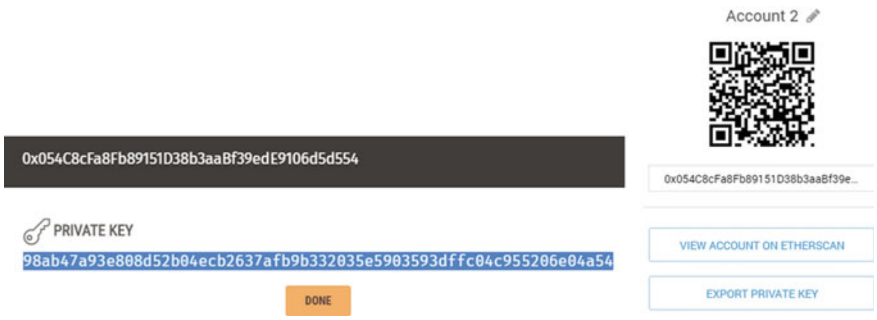
English

New Network

http://localhost:7545

The default network for Ether transactions is Main Net.

Fig. 8 Addition of a custom remote procedure call option to the network and selecting the added network, which is a private network

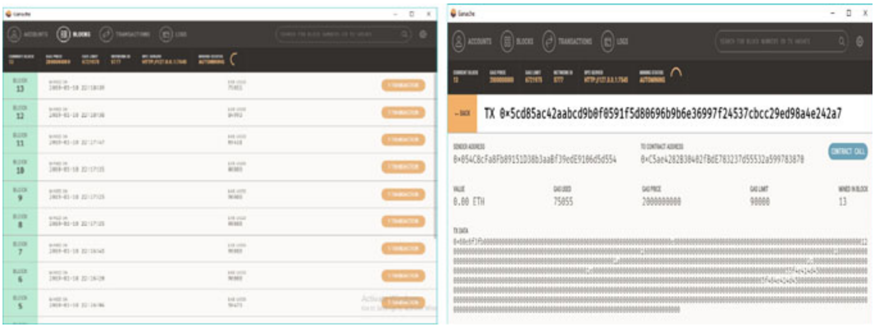


0x054C8cFa8Fb89151D38b3aaBf39edE9106d5d554

PRIVATE KEY

98ab47a93e808d52b04ecb2637afb9b332035e5903593dffc04c955206e04a54

Fig. 9 Private key of account to be imported and successfully importing the account



TX 0x5cd85ac42aabcd9b0f0591f5d80696b9b6e36997f24537cbcc29ed98a4e242a7

0x054C8cFa8Fb89151D38b3aaBf39edE9106d5d554

0.00 ETH

75055

2000000000

90000

13

Fig. 10 Blocks after they are mined and the transaction details of blocks that are mined

can be viewed. It also shows details such as *block number*, *amount of gas used*, and *the date and time when a block is mined*. It also shows transaction details of the block that is mined. It shows details such as sender's address, i.e., the address of the account that is imported in MetaMask to carry out the transaction. This address can be verified with the account address, which is displayed above in Fig. 9.

Table 1 shows an example of a transaction in our developed system for civil registrations.

Table 1 Example of a transaction

TxHash	Block number	Sender address	To contract address	Gas used
0x5cd85ac42a..	13	0x054C8cFa….	0xC5ae42…..	75,055

6 Results and Discussion

As block size increases, blockchain network increases, which results in scalability issues. Note that the scalability can be reduced by reducing block interval to achieve high throughput and efficiency. Moreover, blockchain possesses overhead problems in terms of bandwidth as well as storage space, which is a challenge. In addition, it is assumed that basic document verification is performed by trusted hospital authorities, and there are no malicious activities taking place. In the proposed system, it is stated that government officials will have to mine data blocks, but since Ganache is used, which auto-mines blocks, it is impossible to explicitly illustrate how government officials mine data blocks.

7 Conclusion and Future Scope

This paper presented an application of blockchain technology in civil registration systems that generate both birth as well as death certificates through smart contracts and assures safety, confidentiality, and makes smart contracts tamper-proof. Smart contracts are written using solidity programming and compiled via Remix Editor. The main advantage of using the developed system includes minimum interaction with government organizations, thereby making the entire procedure quicker and reduces human errors. In the near future, the developed system could be improved by including smart contracts in the form of images. In addition, similar concepts could be used to develop other registration applications in land transactions, marriage certificates, and many more sectors.

References

1. Blockchain [Online] https://en.wikipedia.org/wiki/Blockchain (Accessed 15th January 2019)
2. Tech Racers, 4 key features of blockchain, [Online] https://www.techracers.com/blogs/blockchain-key-features/ (Accessed 15th January 2019)
3. Records Keeper, Centralized vs. Decentralized Blockchain, [Online] https://www.recordskeeper.co/blog/centralized-vs-decentralized-blockchain/ (Accessed 15th January 2019)
4. Wei C, Wang Z, Ernst JB, Hong Z, Feng C, Leung VCM (2018) Decentralized applications: the blockchain-empowered software system. IEEE Access 6:53019–53033
5. Bayu Adhi T, Kweka BJ, Park Y, Rhee K-H (2017) A critical review of blockchain and its current applications. In: 2017 International Conference on Electrical Engineering and Computer Science (ICECOS). IEEE, pp 109–113
6. Friðrik ÞH, Hreiðarsson GK, Hamdaqa M, Hjálmtýsson G (2018) Blockchain-based E-voting system. In: 2018 IEEE 11th international conference on cloud computing. IEEE, pp 983–986
7. Heng H (2017) The application of blockchain technology in E-government in China. In: 2017 26th International Conference on Computer Communication and Networks (ICCCN). IEEE, pp 1–4

8. Ahmed A, Nasir Q, Talib M A (2018) Blockchain for government services—use cases, security benefits and challenges. In: 2018 15th international conference on Learning and Technology (L&T). IEEE, pp 112–119
9. Pinyaphat T, Techapanupreeda C (2018) Blockchain: challenges and applications. In: 2018 International Conference on Information Networking (ICOIN). IEEE, pp 473–475
10. Chengjun C, Duan H, Wang C (2018) Tutorial: building secure and trustworthy blockchain applications. In: 2018 IEEE international conference on Cybersecurity Development (SecDev), IEEE, pp 120–121
11. Peng Z, Walker MA, White J, Schmidt DC, Lenz G (2017) Metrics for assessing blockchain-based healthcare decentralized apps. In: 2017 IEEE 19th international conference on e-health networking, applications and services (Healthcom). IEEE, pp 1–4
12. Chen T, Zhu Y, Li Z, Chen J, Li X, Luo X, Lin X, Zhange X (2018) Understanding Ethereum via graph analysis. In: 2018 IEEE conference on computer communications. IEEE, pp 1484–1492