Smart City

Volume 2
Issue 1 *Digital Transformation in the Everyday Urban Life*

Article 2

August 2022

Towards Public Vaccination Data Resilience During Natural Disaster Using Blockchain-Based Decentralized Application

Riri Fitri Sari
University of Indonesia, riri@ui.ac.id

Mokhammad Rizqi Herdiawan
University of Indonesia

Muhammad Hamzah
University of Indonesia

Muhammad Aljundi
University of Indonesia

Jauzak Hussaini Windiatmaja
University of Indonesia, jauzak.hussaini@ui.ac.id

Recommended Citation

Sari, Riri Fitri; Herdiawan, Mokhammad Rizqi; Hamzah, Muhammad; Aljundi, Muhammad; and Windiatmaja, Jauzak Hussaini (2022) "Towards Public Vaccination Data Resilience During Natural Disaster Using Blockchain-Based Decentralized Application," *Smart City*: Vol. 2: Iss. 1, Article 2.
DOI: 10.56940/sc.v2.i1.2
Available at: https://scholarhub.ui.ac.id/smartcity/vol2/iss1/2

TOWARDS PUBLIC VACCINATION DATA RESILIENCE DURING NATURAL DISASTER USING BLOCKCHAIN-BASED DECENTRALIZED APPLICATION

[1]Riri Fitri Sari, [1]Mokhammad Rizqi Herdiawan, [1]Muhammad Hamzah, [1]Muhammad Aljundi and [1]Jauzak Hussaini Windiatmaja

[1]Department of Electrical Engineering, Faculty of Engineering, University of Indonesia

E-mail: riri@ui.ac.id

ABSTRACT

This paper presents a platform development plan for a public vaccination decentralized data base based on Blockchain technology. Vaccination is a medical practice that allows a human body to produce a specific antibody as a means of developing a preventive measure and reducing the risk of contracting a particular disease. Systematic vaccination data recording is an utmost important system that is required particularly during the Covid-19 pandemic. The recording mechanism should utilize the state-of-the-art information and communication technology (ICT). For faster vaccination data recording, it is common to use digital technology, making the health care process faster thus decreasing time needed for the vaccination process. Data related to medical records, i.e. vaccination data records, are essential for the continuity of care of the patients. For health professionals, medical records provide an insight on the clinical judgment being exercised at the time. The presence of a complete, up-to-date and accurate medical record including all vaccination received by a patient can make all the difference to the outcome of a treatment. Although today the digital medical records have been implemented in different traditional database, these database is still very much disintegrated and prone to the happening of natural disaster. Access to medical records will become increasingly important as medical treatments become more complicated and the increase of aging population, while disasters will likely continue to occur with regular frequency. In order to tackle the problem, we propose a blockchain-based application with decentralized storage to provide easy tracing of vaccinated people while maintaining the availability and the resilience of the data stored. Blockchain technology is applied to ensure that the data remain available and secure even during natural disaster occurrence. The decentralized application proposed in this paper is built using Hyperledger Fabric, Django, and cURL. The application is deployed in the cloud using Amazon Web Services (AWS).

Keywords: *Vaccine; Patient Medical Records; Natural Disaster; Blockchain; Decentralized Application*

INTRODUCTION

A vaccine is a biological preparation that provides active acquired immunity to a particular infectious disease. A vaccine typically contains an agent that resembles a disease-causing microorganism and is often made from weakened or killed forms of the microbe, its toxins, or one of its surface proteins [1]. Complete vaccination will produce specific antibodies that provide immunity.

To facilitate the realization of the vaccination program, it is necessary to record systematic vaccination data. This recording can be supported by utilizing information and communication technology (ICT). This utilization can make recording easier, making the health service process faster. Immunization need not require a long process that causes long queues.

Data related to medical records, i.e. vaccination data records, are essential for the continuity of care of the patients. For health professionals, medical records provide an insight on the clinical judgment being exercised at the time. The presence of a complete, up-to-date and accurate medical record including all vaccination received by a patient can make all the difference to the outcome of a treatment. Although today the digital medical records have been implemented in different traditional database, these database is still very much disintegrated and prone to the happening of natural disaster. Access to medical records will become increasingly important as medical treatments become more complicated and the increase of aging population, while disasters will likely continue to occur with regular frequency.

In order to tackle the problem, we propose a blockchain-based application with decentralized storage to provide easy tracing of vaccinated people while maintaining the availability and the resilience of the data stored. Blockchain technology is applied to ensure that the data remain available and secure even during natural disaster occurrence. The decentralized application proposed in this paper is built using Hyperledger Fabric, Django, and cURL. The application is deployed in the cloud using Amazon Web Services (AWS).

The rest of the paper is organized as follows. The literature review and our proposed system architecture as well as implementation results with smart contracts are presented in Section II and III respectively. Section IV discusses deployment and testing for our proposed work. Finally, conclusions and future work are given in Section V.

BLOCKCHAIN

Blockchain is an open-source distributed ledger technology that was first applied in the financial sector with its most popular application being the Bitcoin cryptocurrency [5] which has proven that blockchain technology is stable, secure, and robust [6]. There are already notable implementation examples in healthcare such as the pharmaceutical supply chain [7][8], clinical trial management [9][10], and medical record management [11][12].

Sylim et al. proposed blockchain technology for detecting falsified and substandard drugs in the pharmaceuticals distribution system. They test the feasibility of applying the technology and its principles in a pharmaceuticals surveillance system and its resistance to tampering [7].

Zhuang et al. implemented blockchain technology to provides a feasible solution of Health Information Exchange challenges by utilizing the unique features of blockchain. Utilizing the smart-contract feature, which is a programmable selfexecuting protocol running on a blockchain, they developed a blockchain model to protect data security and patients' privacy, ensure data provenance, and provide patients full control of their health records [8].

Zhuang et al. investigated the current workflow of clinical trials. They used coded Smart Contract regulations to simulate several scenarios in healthcare processes. This proof-of-concept work provides a feasible simulation for potential solutions to monitor clinical trials across different census regions persistently [9]. Furthermore, Zhuang et al. also proposed a blockchain model containing multiple trial-based contracts for trial management and patient engagement and a master smart contract for automated subject matching, patient recruitment, and trial-based contracts management [10].

Soni and Singh address the blockchain impact on the healthcare and biomedical industry for security and privacy purpose [11]. They stated that blockchain allows users to verify public health information integrity, ensure data safety, maintain the integrity of clinical research reports, immutable auditing of medical records, reduction of audit expenses and regulatory compliance. Furthermore, Zhang also define the applicability of Blockchain technology in healthcare [12].

Based on these challenges, disruptive technologies such as blockchain may provide feasible solutions by utilizing blockchain features. Blockchain is a database form where data is stored in the form of a connected "chain of blocks" where each block contains data consisting of value and reference to the previous block which "chained" them together. These blocks are usually used to store transactions. Blockchain is considered to be an "un-hackable" system that can protect data security and patients' privacy [14]. All blockchain users use unique pairs of public and private keys to represent their identities which can be used to map the patient across multiple healthcare facilities thus allowing anonymity [15]. The public key is similar to a user account and the private key is similar to a user password. When patients use blockchain to give permission to clinicians to access prior records, they need to sign this transaction using a private key. This can be done through mobile devices and biometric verifications. Blockchain is an immutable system. Any transaction written into a blockchain is unchangeable and can be checked at any time. This feature keeps all data consistent and eliminates any chance of tampering the data [16].

Smart Contract, a self-executing protocol coded using Solidity [17], which is a Turing-complete language that provides the ability to solve any computational problem, was added to the Ethereum blockchain [18]. All the transactions that happen in the system will automatically follow smart contracts' regulations with mutual consent from all users in the blockchain. Using smart contracts can strengthen the feasibility of applying blockchain for patient-centric information exchange. There are previous examples of blockchain models using smart contracts to ensure all the transactions are following the different policies and all the data are in the same interoperability standard instantaneously [15][19][20][21][22].

Because blockchain is a peer-to-peer network that is fully distributed and does not use third-party intervention, every transaction is broadcasted to all the users to audit whether the input is legitimate or not [9]. The auditing process runs automatically through blockchain's validation process. The data inside the transaction (authorization of the data access in this case) are only accessible to the sender and the receiver. This method ensures users anonymity and ensures all transactions have the senders' authorizations without third-party involvement.

DESIGN AND IMPLEMENTATION

A. System Design

Waterfall Model [2] approach is used in developing the application. Waterfall models consist of phases that separate the system development process from the project. Each phase has an output which will be the input for the next phase. These phases are Requirements, Design, Implementation, Integration and Testing, Deployment, and Maintenance.

In the requirements phase, we determined that the system needs to keep records of vaccinations, which consists of vaccinated patients, vaccinated doctors, types of vaccines, vaccination providers, and vaccination times. All that data needs to be able to be stored on the blockchain, ensure that it is in the record, add it, and be able to display a record of vaccinations that have been carried out.

The design is applied to program code with different programming languages. The program is divided into three, namely frontend, backend, and blockchain. The frontend is the user side (client side) of the system or the part of the system that deals directly with users. This part of the program is structured in the Python and HTML programming languages. The program for this section uses HTML. The backend is the server side for data administration (creating data models) and application pages (page templates). Django framework is used with Model-View-Template (MVT) based architecture [3]. The blockchain part is a separate part of the application but is the place where the application retrieves and stores data records. The blockchain is made using the Hyperledger Fabric [4] which consists of a blockchain network and smart contracts (also known as chain code).

We design a system of database and interface that uses Blockchain to secure the vaccination data by providing the means to access, change and store data in the database. The system design is described with various UML diagrams, i.e. class diagram, use case diagram, and sequence diagram.

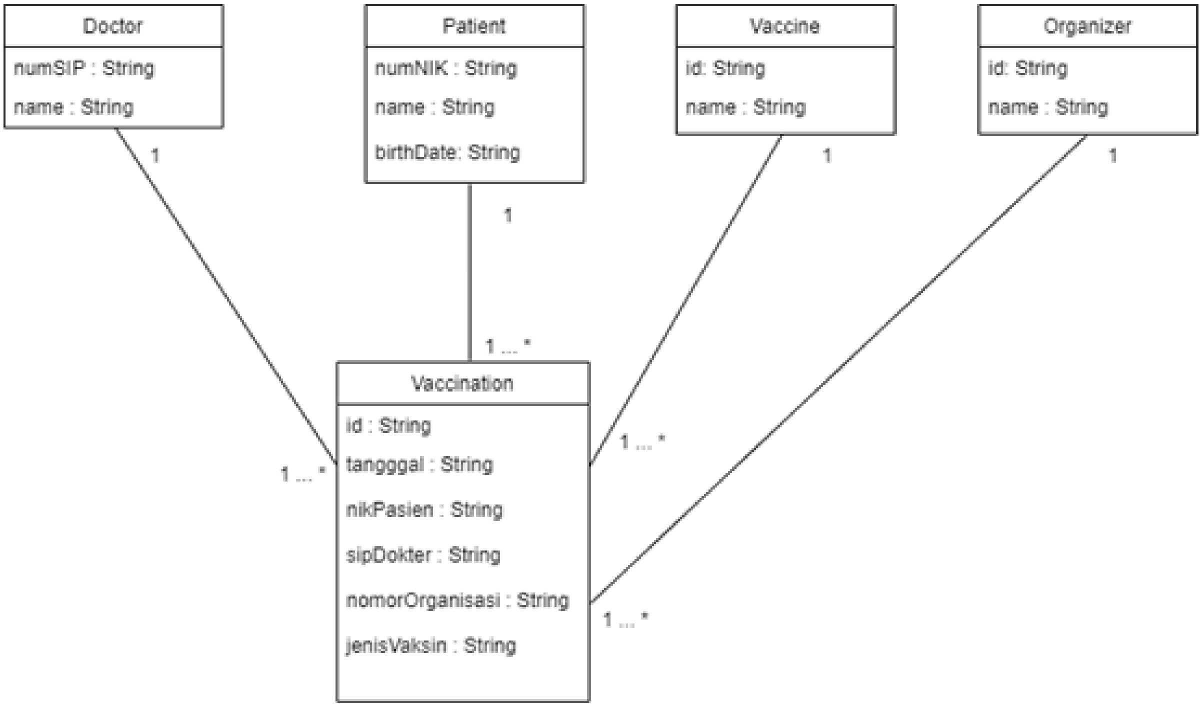


Figure 1. Class Diagram

The class diagram is shown in Figure 1. Class diagram describe the systems structure by showing its class, attribute, method, and connection between classes. There are 5 classes that make up the system structure, i.e. Vaccination, Doctor, Patient, Vaccine, and Organizer. Each of those classes has a main identifying attribute used when the system needs to call the object of the class and the

Vaccination class object uses the identifiers of Doctor, Patient, Vaccine, and Organizer objects as values for some of its attributes to show that those objects are related to the Vaccination object.

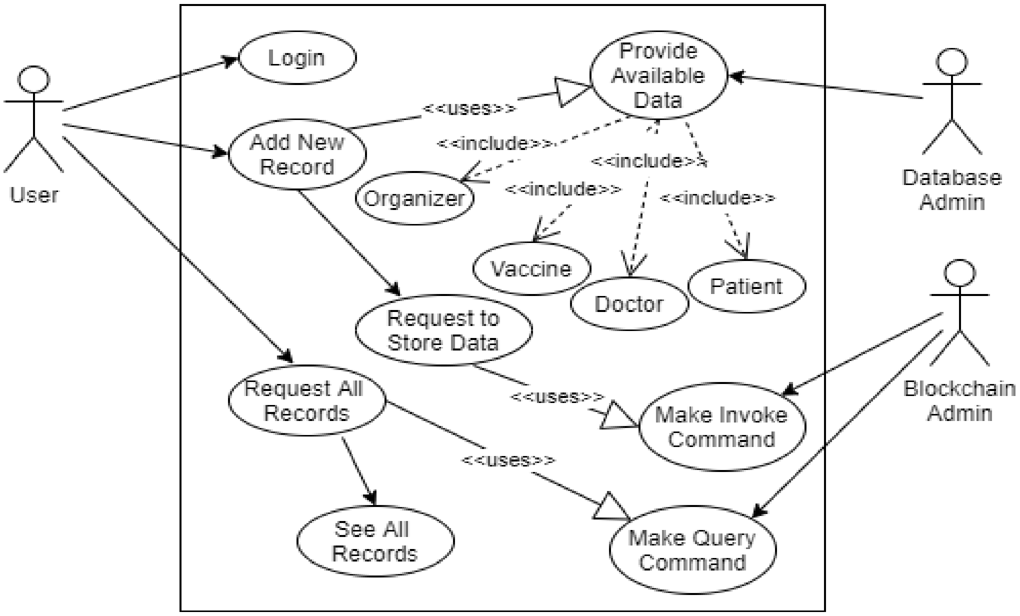


Figure 2. Use Case Diagram

Use case diagram is shown in Figure 2. that describes the parties that are involved with the system and their interactions with each other. User interacts with the application by login, add new record that uses the available data provided by the database admin and makes a request to store record that uses the invoke command made by the blockchain admin that allows the user to input new records to the database, and Request All Records that uses the Query command made by the blockchain admin that allows the user to see all records in the database.

Sequence diagram is shown in Figure 3. It describes the system in the form of how the objects in the system work together and in what order. First, the user needs to login to the application. Subsequently, when the user attempts to input new records. The application will send an invoke for new entry to the blockchain and if allowed, the blockchain will grant permission and the application will tell the user that the new record is successfully added. Last, if the user requests the records in the database as shown then the application will send a query for all entry to the blockchain, and if allowed the blockchain will grant permission and send all records in the database to the application. The application will show the user a table of records filled with records from the database.

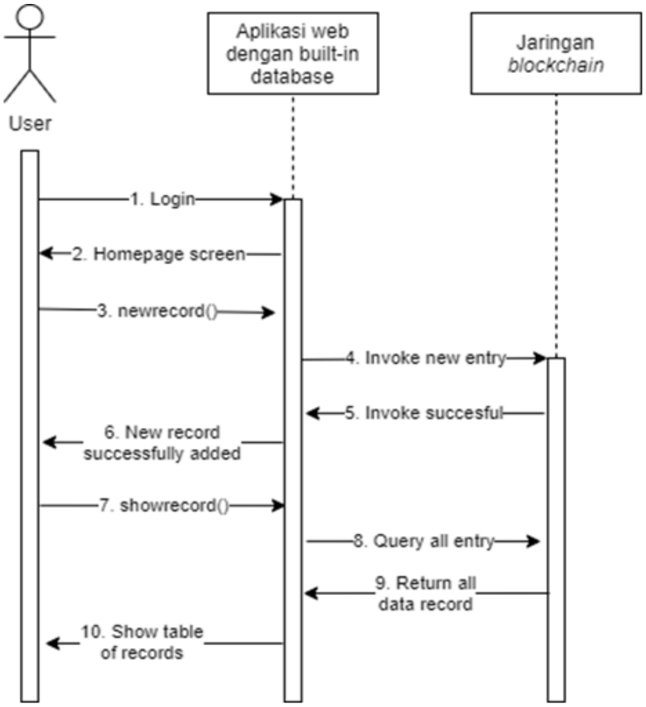


Figure 3. Sequence Diagram

B. Implementation

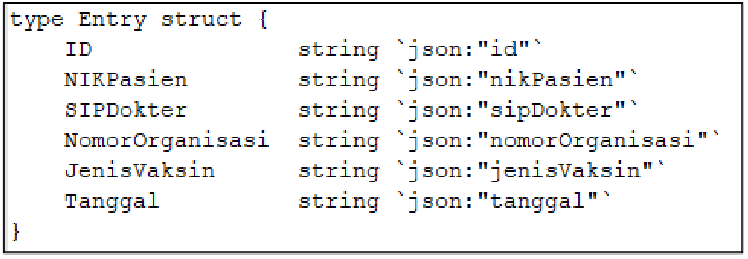
The implement phase is divided into three main sections, i.e. Blockchain implementation, Web Application implementation, and Functional implementation. The application is built using Hyperledger Fabric, Django, and cURL. The application is deployed in the cloud using Amazon Web Services (AWS). The system can provide the decentralized application for the vaccine using the blockchain technology.

To ensure that the proposed system is applicable, we observe the deployability of the system in a country. We took Indonesia as the case study. Yodi et al. reports that provide a detailed description of a health system and policy initiatives in Indonesia [23]. The report stated that Indonesia has a well-developed and extensive network of public health facilities, which reach from community level through to district, provincial and national level hospitals. Increased government and private funding has resulted in an increase in health infrastructure, including primary and referral health facilities. By knowing this, we believe that the proposed system can be applied in Indonesia's current system. Furthermore, with the growing interest of company in Indonesia to adopt blockchain technology, we think the proposed system will be deployed in such a good environment.

1. Blockchain

Blockchain is implemented using Hyperledger Fabric. Hyperledger Fabric is an open source, private blockchain framework. As we aware that healthcare data is related to patient’s privacy, to comply the ethical and legal aspect, we opted to use private blockchain. Within the chain code, there are declarations of variables or data structures that are stored in the blockchain. It is equipped with method functions to add, change, or display data. In this project, the functions used in chain code are functions to retrieve all data and to add new data to records. The chain code used in this project can be divided into three parts.

Table 1. ‘Entry’ Variable



```
type Entry struct {
    ID                string `json:"id"`
    NIKPasien         string `json:"nikPasien"`
    SIPDokter         string `json:"sipDokter"`
    NomorOrganisasi   string `json:"nomorOrganisasi"`
    JenisVaksin       string `json:"jenisVaksin"`
    Tanggal           string `json:"tanggal"`
}
```

Table 1 shows variable declaration Entry. This section serves to define the form of records that will be stored in the blockchain. The variable for this record is named 'Entry' with the data type struct.

Table 2. Smart Contract Algorithm: CreateEntry Function and GetAllEntry Function

```
Algorithm 1. SmartContract
Function CreateEntry
        input: contractapi.TransactionContextInterface as ctx, sting as id, string as nikPasien
        output: ctx GetStub, PutState id

        entry <- Entry{
                ID: id,
                NIKPasien: nikPasien,
                SIPDokter: sipDokter,
                NomorOrganisasi: nomorOrganisasi,
                JenisVaksin: jenisVaksin,
                Tanggal: tanggal
        }

Function GetAllEntry
        input: contractapi.TransactionContextInterface as ctx
        output: entries, nil

        defer close resultsIterator
        declare entries Entry

        if resultsIterator has next:
                declare entry Entry
                entries <- append entries to entry
```

Table 2 shows the Smart Contract Algorithm. There are 2 functions within this algorithm, i.e. CreateEntry and GetAllEntry. CreateEntry function serves to create new records according to the structure defined in the variable declaration. GetAllEntry retrieves all records, this section serves to display all records stored on the blockchain as the variable.

2. Web Application

Web Application implementation that uses the framework of Django that is based on Python. The purpose of implementing this section is to establish an easier interface. The Django framework uses a model-view-template (MVT) architecture base [6]. Models are used to build data structures, in this case, create data interfaces so that input can enter the blockchain. A view is a collection of functions that make up the interface of a Django web application. The template serves as a layout provider for static and dynamic HTML web pages that will be used by the frontend and backend of the web application. So according to the architecture used, the Web Application implementation will be divided into three sections.

Table 3. Models

```
Algorithm 2. Models

Object Patient:
        input: models.Model
        output: name
        numNIK <- models.CharField(max_length <- 50, primary_key <- True)
        name <- models.CharField(max_length <- 200, null <- True)

Object Doctor:
        input: models.Model
        output: name
        numSIP <- models.CharField(max_length <- 50, primary_key <- True)
        name <-models.CharField(max_length <- 200, null <- True)

Object Organizer:
        input: models.Model
        output: name
        id <- models.CharField(max_length <- 50, primary_key <- True)
        name <-models.CharField(max_length <- 200, null <- True)

Object Vaccine:
        input: models.Model
        output: name
        id <- models.CharField(max_length <- 50, primary_key <- True)
        name <-models.CharField(max_length <- 200, null <- True)

Function Vaccination:
        id <- models.CharField(max_length <- 50, primary_key <- True)
        patient <- models.ForeignKey(Patient, null <- True, on_delete <- models.SET_NULL)
        doctor <- models.ForeignKey(Doctor, null <- True, on_delete <- models.SET_NULL)
        organizer <- models.ForeignKey(Organizer, null <- True, on_delete <- models.SET_NULL)
        vaccine <- models.ForeignKey(Vaccine, null <- True, on_delete <- models.SET_NULL)
        date_created <- models.DateTimeField(auto_now_add <- True, null <- True)
```

Table 3 shows models algorithm. The model is implemented via the models.py file to model the data structure of the design. The data model is defined by class type. All data used is modeled in this file, both data stored on the blockchain (vaccinations) and those stored in databases/outside the blockchain (patients, doctors, vaccines, organizers). The vaccination class takes data attributes from other data classes with the aim of limiting the data to only those registered in the database.

Table 4. Views algorithm: NewRecord function and ShowRecords function

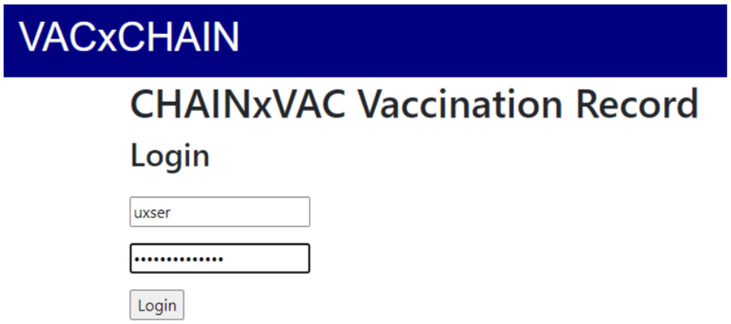
```
Algorithm 3. Views

Function NewRecord
        input: request, form
        output: render request, new record view, form

        form <- NewRecordForm()
        if request method == 'POST':
                form <- NewRecordForm(request POST)
                if form is valid:
                        insert form to fabric
                        redirect NewRecord

Function ShowRecord
        input: request
        output: render request, show records view, vaccinations array

        declare array vaccinations
        try:
                vaccination <- get all entrries from hyperledger fabric
                for each vaccination:
                        nomor_organisasi <- get name object from Organizer with id
                        "nomor_organisasi"
                        jenis_vaksin <- get name object from Organizer with id "jenis_vaksin"
```

3. Functional Implementation

The functional implementation is described in Figure 4 – Figure 7.

a) Login page

The page appears at the first time opening the web application. It requires the user to enter the valid username and the password like it’s shown in Figure 4.



VACxCHAIN

CHAINxVAC Vaccination Record

Login

uxser

••••••••••••••

Figure 4. Interface of login page

After the user logs in, the application will show a home page like Figure 4.

b) Home page

In this page, users can choose to add a new record, show all records, or log out from the application. The choice is shown like in Figure 5.

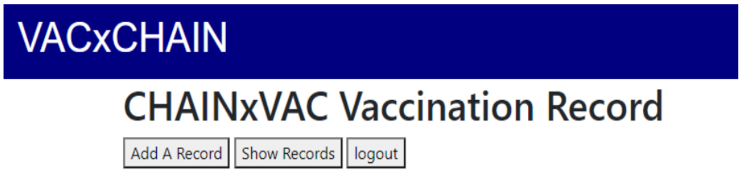


Figure 5. Interface of home page

c) New record page

From the home page, if the user chooses to add a new record, it will show the page like in Figure 6. The user can add a single entry or can add multiple records by uploading a CSV file. Adding a single data will use data in the database so the record added will have only registered data. Also, the user can go back to the home page by clicking the ‘Go Back’ button.

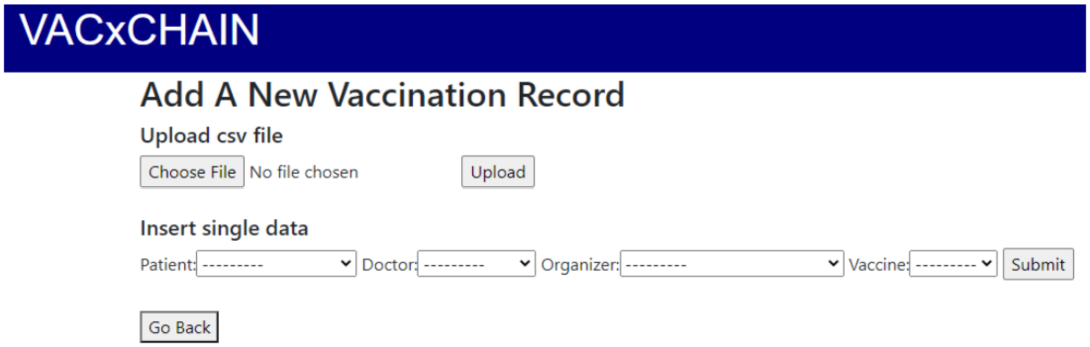
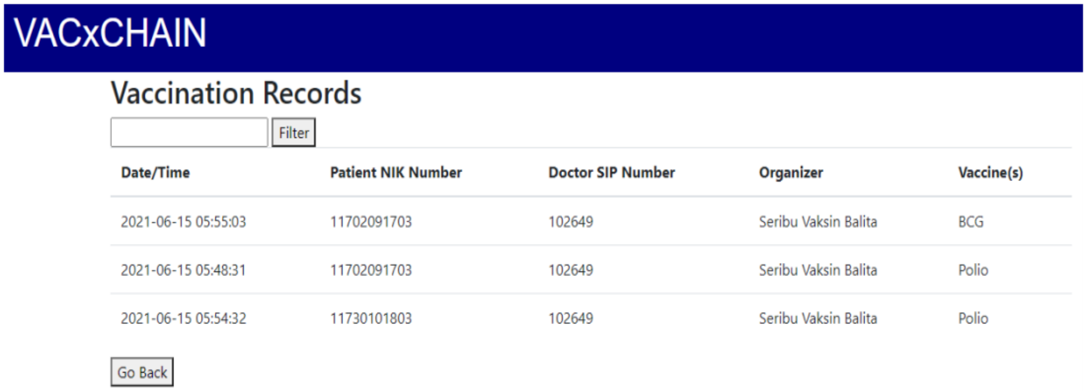


Figure 6. Interface of new record page

d) Show records page

From the home page, if the user chooses to see all records, the application will show a page containing a table full of vaccination records that have been stored in the blockchain like shown in Figure 7. The display can be filtered so the table can only show the records that contain the data in the filter. After seeing the table, the user can go back by clicking the ‘Go Back’ button.



VACxCHAIN

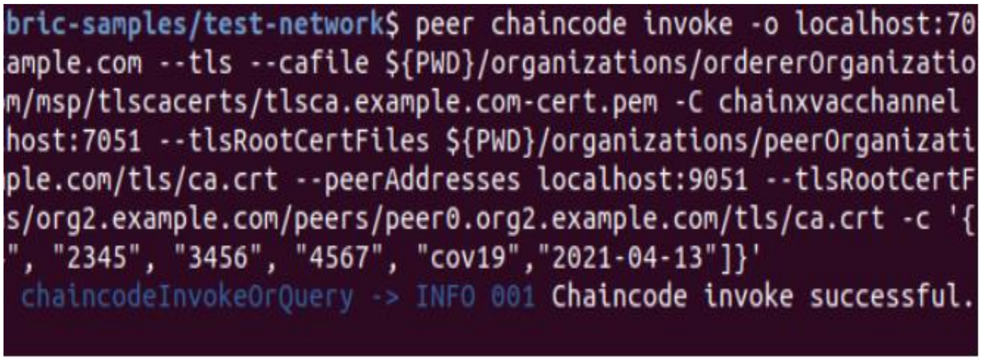
Vaccination Records

Date/Time	Patient NIK Number	Doctor SIP Number	Organizer	Vaccine(s)
2021-06-15 05:55:03	11702091703	102649	Seribu Vaksin Balita	BCG
2021-06-15 05:48:31	11702091703	102649	Seribu Vaksin Balita	Polio
2021-06-15 05:54:32	11730101803	102649	Seribu Vaksin Balita	Polio

Figure 7. Interface of show records page

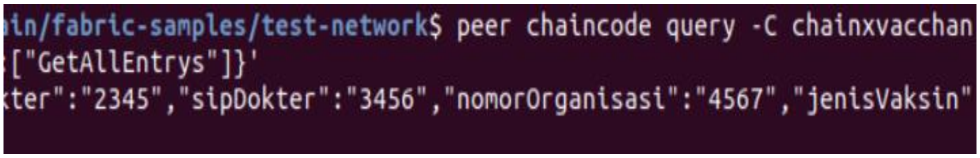
TESTING

The functions in blockchain are tested by using Linux Terminal to see if the results are as expected. There are two functions to be tested, CreateEntry and GetAllEntrys. As the result shown in Figure 8, it can be concluded that the blockchain function can add a new record. The result in Figure 9 shows that blockchain can return all stored data with the function GetAllEntrys.



```
bric-samples/test-network$ peer chaincode invoke -o localhost:70
ample.com --tls --cafile ${PWD}/organizations/ordererOrganizatio
m/msp/tlscacerts/tlsca.example.com-cert.pem -C chainxvacchannel
host:7051 --tlsRootCertFiles ${PWD}/organizations/peerOrganizati
ple.com/tls/ca.crt --peerAddresses localhost:9051 --tlsRootCertF
s/org2.example.com/peers/peer0.org2.example.com/tls/ca.crt -c '{
", "2345", "3456", "4567", "cov19","2021-04-13"]}'
 chaincodeInvokeOrQuery -> INFO 001 Chaincode invoke successful.
```

Figure 8. The results of testing CreateEntry function in Linux Terminal



```
in/fabric-samples/test-network$ peer chaincode query -C chainxvacchan
["GetAllEntrys"]}'
ter":"2345","sipDokter":"3456","nomorOrganisasi":"4567","jenisVaksin"
```

Figure 9. The results of testing GetAllEntrys function in Linux Terminal

CONCLUSION AND FUTURE WORK

In this paper, we proposed a data recording system that uses blockchain technology to secure the database from tampering by outside parties. We use blockchain to accomplish this due to its nature to decentralize its database so that there are multiple copies of the database allowing for cross reference of data in every database and immutable changes that ensures that any instance inserted into the system is unable to be changed. We analyzed the blockchain function and web application interface using the available testbench and discovered that the blockchain functions behaved appropriately. The application interface was able to show the page that was designed in the template file while the application also responded correctly according to our design.

There are several things that can be improved in the project. The interface in the application is still minimal so the layout of its page can be reshaped as a regular website page. The data control can be improved. Some work to ensure the opening procedure to be ready for a disaster condition in which blockchain technology is used for public vaccination data will help the prevention of larger pandemic situation. Data resilience of public vaccination information will help the society at large to face some emergency condition.

ACKNOWLEDGMENT

This work is supported by the Ministry of Education, Culture, Research, and Technology (Kemendikbudristek) of Indonesia under PDD Grant with contract number NKB-1016/UN2.RST/HKP.05.00/2022.

REFERENCES

[1] World Health Organization, "Vaccines and immunization: What is vaccination?," 19 July 2021. [Online]. Available: https://www.who.int/news-room/q-a-detail/vaccines-and-immunization-what-is-vaccination/. [Accessed 17 August 2021].

[2] "SDLC - Waterfall Model," [Online]. Available: https://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.htm. [Accessed 14 June 2021].

[3] "Django Project MVT Structure," [Online]. Available: https://www.geeksforgeeks.org/django-project-mvt-structure/. [Accessed 14 June 2021].

[4] "Develop Hyperledger Fabric Chaincode," [Online]. Available: https://docs.aws.amazon.com/managed-blockchain/latest/hyperledger-fabric-dev/managed-blockchain-hyperledger-develop-chaincode.html. [Accessed 14 June 2021].

[5] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” 2008.

[6] R. Grinberg, “Bitcoin: An innovative alternative digital currency,” Hastings Sci. & Tech. LJ, vol. 4, pp. 159, 2012.

[7] P. Sylim, F. Liu, A. Marcelo, and P. Fontelo, “Blockchain technology for detecting falsified and substandard drugs in distribution: Pharmaceutical supply chain intervention,” JMIR Res. Protocols, vol. 7, no. 9, pp. e10163, 2018.

[8] Y. Zhuang, L. R. Sheets, Y. -W. Chen, Z. -Y. Shae, J. J. P. Tsai and C. -R. Shyu, "A Patient-Centric Health Information Exchange Framework Using Blockchain Technology," in IEEE Journal of Biomedical and Health Informatics, vol. 24, no. 8, pp. 2169-2176, Aug. 2020, doi: 10.1109/JBHI.2020.2993072.

[9] Y. Zhuang, L. Sheets, Z. Shae, J. J. P. Tsai, and C. R. Shyu, “Applying blockchain technology for health information exchange and persistent monitoring for clinical trials,” inAnnu. Symp. Proc. vol. 2018, pp. 1167–75, 2018.

[10] Y. Zhuang et al., “Applying blockchain technology to enhance clinical trial recruitment,” in AMIA Annu Symp Proc, 2019, vol. 2019, pp. 1277–1285.

[11] T.-T. Kuo, H.-E. Kim, and L. Ohno-Machado, “Blockchain distributed ledger technologies for biomedical and health care applications,” J. Amer. Med. Inform. Assoc., vol. 24, no. 6, pp. 1211–1220, 2017.

[12] P. Zhang, D. C. Schmidt, J. White, and G. Lenz, “Blockchain technology use cases in healthcare,” in Advances in Computers, vol. 111, Amsterdam, The Netherlands; New York: Elsevier, 2018, pp. 1–541.

[13] T.-T. Kuo, H. Zavaleta Rojas, and L. Ohno-Machado, “Comparison of blockchain platforms: A systematic review and healthcare examples,” J. Amer. Med. Inform. Assoc., vol. 26, no. 5, pp. 462–478, 2019.

[14] R. Burstall and B. Clark, “Blockchain, IP and the fashion industry,” Managing Intell. Prop., vol. 266, pp. 9, 2017.

[15] G. Zyskind and O. Nathan, “Decentralizing privacy: Using blockchain to protect personal data,” in Security and Privacy Workshops (SPW), 2015 IEEE, 2015: IEEE, pp. 180–184.

[16] V. Patel, “A framework for secure and decentralized sharing of medical imaging data via blockchain consensus,” Health Inform. J., pp. 1460458218769699, 2018.

[17] G. Wood, “Ethereum: A secure decentralised generalised transaction ledger,” Ethereum Project Yellow Paper, vol. 151, no. 2014, pp. 1–32, 2014.

[18] R. C. Holt and J. R. Cordy, “The turing programming language,” Commun ACM, vol. 31, no. 12, pp. 1410–1424, 1988.

[19] X. Liang et al., “Provchain: A blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability,” in Proc. 17th IEEE/ACM Int. Symp. Cluster, Cloud and Grid Computing, 2017: IEEE Press, pp. 468–477.

[20] K. Peterson, R. Deeduvanu, P. Kanjamala, and K. Boles, “A blockchainbased approach to health information exchange networks,” in Proc. NIST Workshop Blockchain Healthcare, 2016, vol. 1, pp. 1–10.

[21] C. Esposito, A. De Santis, G. Tortora, H. Chang, and K.-K. R. Choo, “Blockchain: A panacea for healthcare cloud-based data security and privacy?,” IEEE Cloud Computing, vol. 5, no. 1, pp. 31–37, 2018.

[22] P. Zhang, J. White, D. C. Schmidt, G. Lenz, and S. T. Rosenbloom, “FHIRChain: Applying blockchain to securely and scalably share clinical data,” Comput. and Structural BioTechnol. J., vol. 16, pp. 267–278, 2018.

[23] Mahendradhata Y, Trisnantoro L, Listyadewi S, Soewondo P, Marthias T, Harimurti P, Prawira J. The Republic of Indonesia health system review.