JOIV : Int. J. Inform. Visualization, 8(2) - May 2024 845-853

INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

journal homepage : www.joiv.org/index.php/joiv

A Prototype of Decentralized Applications (DApps) Population Management System Based on Blockchain and Smart Contract

Septovan Dwi Suputra Saian [a,*], Irwan Sembiring [a], Daniel H. F. Manongga [a]

[a] *Faculty of Information Technology, Universitas Kristen Satya Wacana, Sidorejo, Salatiga, Indonesia*

Corresponding author: [*]septovan.ovan@gmail.com

***Abstract*—The Indonesian population reached 270,20 million in 2020. Each resident is equipped with various secret identities. The COVID-19 pandemic has made all activities use technology as a basis, causing residents' identities to be stored digitally. Some applications that keep these identities experience data leaks. However, with the advent of Web3 and its emphasis on decentralization through blockchain, a new era of secure data management is possible. Blockchain, with its inherent security features, ensures that data stored is secure, difficult to damage or lose due to mutual consensus. Every transaction is recorded, making it easy to carry out the audit process. Therefore, this research will design and implement prototype dApps for secure population management, leveraging the superior security of blockchain technology. The initial stage of research is to conduct a literature study. Furthermore, it is to create designs such as system, infrastructure, and activity diagrams. Then do the development of the dApps prototype. The last is testing using OWASP ZAP and cost analysis. A dApps prototype was implemented on a blockchain. Every transaction is recorded and publicly viewable through the Etherscan platform. Other data stored on a blockchain have gone through an AES-256 encryption process with the data owner's account key so that the owner can only see the data. The results of the tests performed show that there is no high-level warning. The cost analysis results show that the most used costs are when deploying smart contracts and making new data. For further development, it is implementing permissionless blockchain and multi-accounts.**

***Keywords*—Decentralized applications; blockchain; smart contract.**

Manuscript received 29 May 2023; revised 16 Oct. 2023; accepted 3 Nov. 2023. Date of publication 31 May 2024.

I. INTRODUCTION

In 2020, the total population of Indonesia reached 270,20 million [1]. Each resident is equipped with various identities such as KIA (Child Identity Card), KTP (Resident Identity Card), KK (Family Card), passport, etc. These identities are confidential and need to be kept secret [2]. Data for 2022 shows that 66,48% of Indonesia's population has internet access [3]. The COVID-19 pandemic also catalyzes almost all sectors using technology for business processes. So internet use has increased to 52% since the pandemic [4].

For example, e-commerce applications such as Tokopedia, Bukalapak, and Bhinneka are in the buying and selling sector. Another government sector is the PeduliLindungi application, created in collaboration with Indonesian ministries. One of its functions is to store COVID-19 vaccination certificate data [5]. Another example is the JKN Mobile application developed by the Health Social Security Administration Agency (BPJS). The eHAC (electronic Health Alert Card) application is also in the government sector. In the financial industry, the Kreditplus application is used.

These applications store confidential data that should be guaranteed security. However, the PeduliLindungi application, which holds the National Identity Number (NIK) and vaccine registration number on the COVID-19 vaccination certificate [6] is suspected that there was a data leak due to the circulation of data by the President of the Republic of Indonesia, Joko Widodo [7]. Apps like 1) Tokopedia; 2) Bukalapak; 3) Bhinneka; and 4) JKN Mobile also experienced data leaks [7]. In August 2020, Kreditplus also experienced a data leak, namely as many as 890 thousand customer data in the form of names, e-mail addresses, passwords, home addresses, and KK data were sold on the Raid forums website [8]. The eHAC application also experienced a leak of 1.3 million user data [9]The data includes leaked names, home addresses, ID numbers, the hospital where the COVID-19 test was carried out, and others.

Web3 combines the concept of decentralization, blockchain technology, and token-based economics [10].

Web3 is a further development of Web1 & Web2. Web1 is known as a read-only because information moves in one direction. Meanwhile, Web2 is known as a read-write because the information has moved in both directions. Then Web3 can be called a read-write-own. It is a new idea that Web3 focuses on eliminating third parties with a decentralized process. Applications that run on blockchains are called decentralized applications (dApps) [11].

Blockchain provides several advantages in several aspects. For example, in parts (1) accounting settlement and crowdfunding; (2) data storage and sharing; (3) supply chain management; and (4) smart trading [12]. At number 2, the blockchain can be used as a secure repository of digital assets. It is because blockchain has the concept of decentralization and a secure ledger. Decentralization means not placing data in one centralized agent but placing it in everyone in the world. So, the data is difficult to tamper with. Besides that, blockchain can increase transparency and, of course, security. With these advantages, blockchain can solve personal data leaks in several applications.

Several business sectors have started developing blockchain as their business foundation; for example, telemedicine utilizes blockchain to store patient data where patients can view and authenticate it [13]. In the transportation sector, a prototype was developed as a user identity management for public transportation in Europe [14].

Therefore, this research will focus on designing dApps for population data management applications to prevent leakage and misuse of confidential identity data. The system encrypts the data using AES with a key only owned by the data owner. After that, the blockchain stores that data. So, this design is expected to create a system that can prevent leaks and misuse of confidential identity data.

II. Materials and Method

A. Blockchain

Blockchain is a decentralized and distributed database system with various combinations of interconnected blocks [15]. Blockchain decentralization means that the control or management of the system is not carried out by a single entity but by people who use the blockchain worldwide. This makes the blockchain secure because data becomes challenging to tamper with due to a consensus process. One of the most famous examples of blockchain implementation is cryptocurrencies such as Bitcoin and Ethereum.

Fig. 1 [15] is an overview of the blockchain structure. Each block consists of data, the previous block's hash, and the current hash resulting from the data hashing process. A new block will appear if there is a data change, which also stores the previous block's hash. When there is a difference between the last block's hash and the current hash in the previous block, this indicates a damaged block that could be caused by someone wanting to destroy the data.

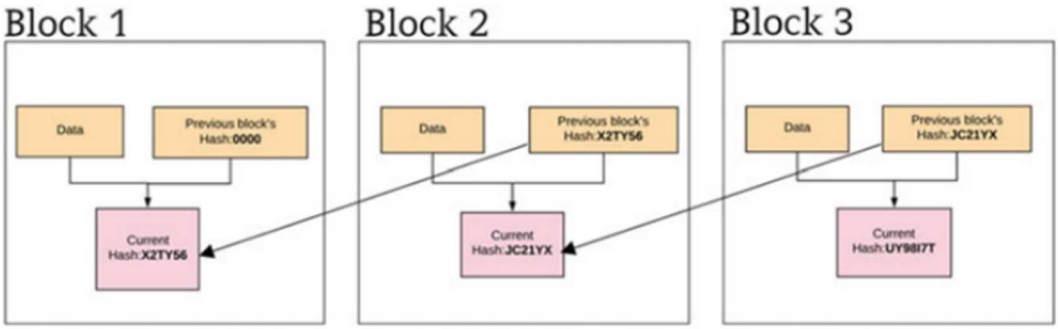

Fig. 1 Blockchain structure

Every time a state or data changes in a block, a transaction fee is required, known as a gas fee [16]. The gas fee results from multiplying the gas price and gas used. Changing the state or data means forming a new block. The transaction does not require a gas fee if there are no changes. The gas price unit used is not Ether (ETH) but Wei or Gwei. This unit is smaller than ETH: 1 *ETH* = 1018 *wei* = 109 *Gwei*. The nominal gas fee cannot be determined, meaning that each transaction can have a different price.

When viewed from its application, blockchain has experienced development. Starting from Blockchain 1.0, Blockchain 2.0, and Blockchain 3.0 [12]. Blockchain 1.0 deals with virtual currencies like Bitcoin, so blockchain is used as a payment system. Blockchain 2.0 has developed into a system that is not only payment but also operates in other sectors, such as security trading, supply chain finance, banking instruments, payment clearing, etc. In this phase, smart contracts, smart property, dApps, decentralized autonomous organizations (DAOs), and decentralized autonomous corporations (DACs) emerged. While Blockchain 3.0 is developing again and being used in various aspects such as government, health, science, culture, and art. Blockchain focuses on regulation and governance in a decentralized ecosystem in this era.

Blockchain has two types [17], permissionless and permissioned blockchain. Permissionless blockchain means writers and readers can interact anytime, like Bitcoin. Meanwhile, permissioned blockchain means only those with access can write and read processes. It makes permissioned blockchains like centralized databases.

B. Smart contract

A smart contract is a program code installed on a blockchain [18] Nick Szabo first coined this term in the late 1990s. At that time, smart contracts were described as vending machines. Someone inserts the appropriate coin, and then the machine will dispense something as requested. This example wants to show that there is no third-party influence.

It shows that the machine will issue something requested (assuming the number of coins is appropriate).

Currently, the most popular developer platform for installing smart contracts is Ethereum. The smart contract will be installed on a virtual machine (for example, Ethereum Virtual Machine or EVM). The programming language used to develop smart contracts is Solidity, which is an object-oriented programming language.

Smart contracts have several advantages [19],

1) Reduce the risk. Once installed, a smart contract cannot be changed. In addition, smart contracts store all historical data, making them easy to track and audit. This process reduces the risk of financial fraud.

2) Reduce administrative and service costs: Third-party costs can be eliminated because they do not involve a third party.

3) Improve the efficiency of business processes: As with the second point, it does not involve third parties, so business processes can be more efficient because they no longer go through third parties.

Blockchain and AES are used as systems for recording financial transactions used by parents, students, and schools [20]. Parents or students can top up, and then students can transact for school needs. The nominal entered when topping up will be encrypted using AES and stored in the blockchain. Other data, such as student ID, transaction ID, transaction code, and others, are stored in a database without encryption.

In this research, the data encrypted with AES before being stored on the blockchain is the entire data, not just part of the data. In addition, it does not use a regular database to store data. This aims to reduce data leakage. The use of blockchain with encryption in the health business sector [21], [22]. Blockchain is used to store patient data. Before being stored, the data is encrypted with a key generated by the application. In this study, the key is a wallet key. It shows that only people who own the data can decrypt and view it.

Blockchain and AES encryption were utilized to store data [23]. Encryption uses finger vein reading technology, which can be used during verification. Meanwhile, in this research, verification is still carried out by comparing the data entered with the stored data. Blockchain and encryption are used to carry out Know Your Customer (KYC) verification [24]. Blockchain stores customer data entered into a company, so customers can verify their data at a different company without entering it again. However, the system generates the key to encoding the data. In this study, the key uses a wallet key owned only by the user.

Based on previous studies, this research will develop a design to secure population data from data leaks. The security scenario is to store complete population data on a blockchain. Previously, the data had been encrypted using the AES algorithm with the data owner's wallet key. The smart contract that will be deployed is also equipped with a modifier function to check ownership of the smart contract. The application's encryption key is no longer generated through this design and is attached to the data owner. So, the data can only be seen by the owner. The modifier function provides additional security by ensuring that only the smart contract owner can access it.

This research will develop a population system dApp prototype, described in three stages (Fig. 2).

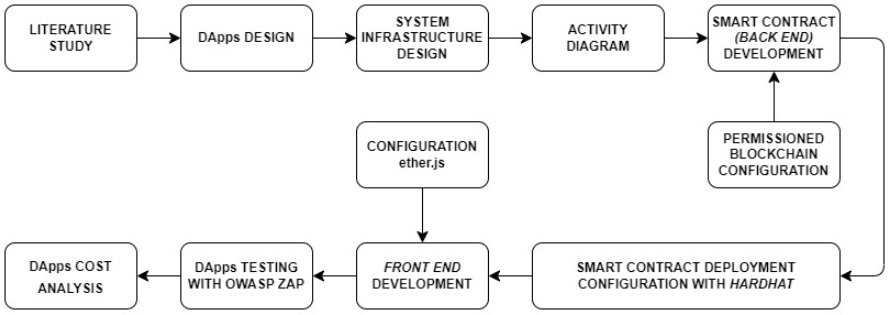

Fig. 2 Research stages

The stage of this research begins with conducting a literature study, namely by reading previous studies and collecting data regarding data leakage. The next stage is to design DApps. This stage is to make three designs: system design, system infrastructure design, and activity diagram. The next stage is the development or coding of DApps. This stage is divided into several parts: smart contract development, which simultaneously configures permissioned blockchain, smart contract deployment configuration, and front-end development along with ether.js configuration. Further testing of dApps using the OWASP ZAP tool. The last is to analyze the resulting DApps.

Open Web Application Security Project Zed Attack Proxy (OWASP ZAP) is an open-source tool that is useful for finding vulnerabilities in a website application [25][26]. The stages of its use consist of 5 steps, namely 1) Input - enter the hostname/host ID as input to attack; 2) Scan – application scan; 3) Discover – find application loopholes; 4) Analysis – the process of analyzing the findings by categorizing them into low, medium, high, and information; 5) Result – final result [27], [28].

National Identification Data		
+nik: string	+desa: string	+nameOfBiologicalMother: string
+name: string	+kecamatan: string	+issuingProvince: string
+placeOfBirth: string	+kelurahan: string	+issuingCity: string
+dateOfBirth: uint256	+religion: string	+issuedDate: uint256
+gender: string	+maritalStatus: string	
+bloodType: string	+occupation: string	
+address: string	+citizenship: string	
+rtRw: string	+validityPeriod: uint256	

Fig. 3 Data scheme

In this study, population data is used as data stored on the blockchain. This population data is depicted in Fig. 3. This data is some of the confidential data stored on a KTP. Data on the biological mother's name will be used in the verification process.

III. Result and Discussion

The DApps that will be developed are permissioned blockchains, meaning access restrictions exist. The limitation is access to attributes on smart contracts that can only be accessed by two actors, the government (Government) and residents (Citizens). Further is shown in Fig. 4.

The government has two features: creating and updating citizen national identification. In comparison, Citizens have features for data claims (claim) and data re-claims (re-claim). In addition, when data is updated, Citizens will receive a notification before finally re-claiming. Fig. 5 illustrates the infrastructure of the dApps being developed. DApps is a website-based application with 2 JavaScript libraries, Ethers.js and Crypto.js. Ethers.js is used to communicate between dApps and smart contracts [29]. Meanwhile, Crypto.js [30] encrypts and decrypts population data stored in smart contracts.

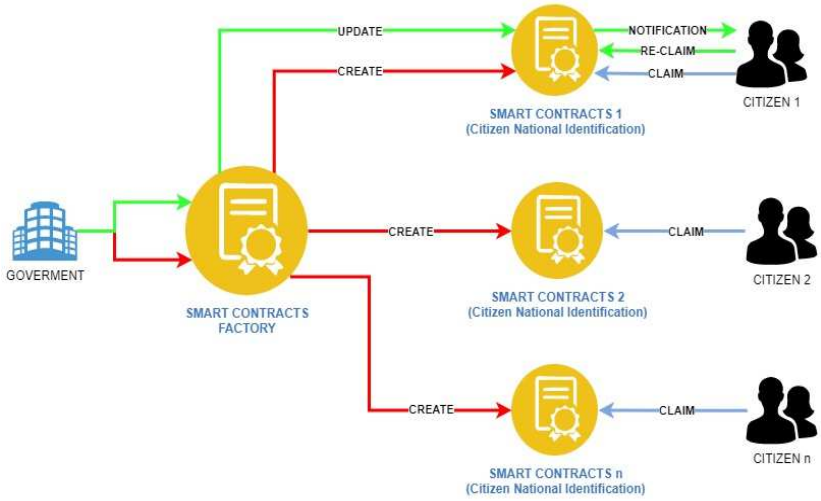

Fig. 4 dApps design

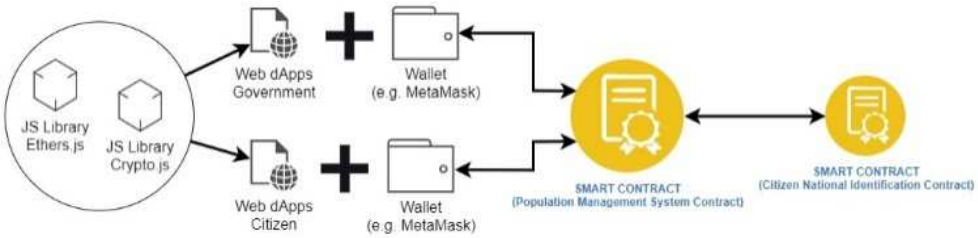

Fig. 5 dApps infrastructure

Fig. 6 is an activity diagram for making citizen data. The only actor involved is the Government, which interacts with the dApps. The government fills in the required data, which will then be encrypted with Crypto.js. The encrypted data is JSON data (Fig. 10), while the NIK value, birth mother's name, and date of birth are stored in a private attribute on the smart contract without encryption. These three data are helpful for the verification process. Fig. 7 shows an activity diagram illustrating the Citizen data verification and claim process. The only actor involved is Citizen, who is connected directly to dApps. This is an application of blockchain that eliminates third parties. Only Citizens can verify directly through the system without going through a third party. The data used to verify is NIK, biological mother's name, and date of birth. After the successful verification process, the Citizen can continue the claim process.

Figure 6 and Figure 7 are the main features of this developed model. The first part of the process is encrypting data before being stored on the blockchain. Next is the data verification process. The first stage of verification is locking the data with the Citizen's wallet key after successfully matching the data. The next verification is when a Citizen wants to see the data. The data can only be seen by the data owner, namely the Citizen.

When interacting between dApps and smart contracts, users need a wallet (e.g., MetaMask). In this dApps prototype, 1 MetaMask account can only be used by one user. For example, using Government can only be done with 1 MetaMask account and cannot be changed. As with Citizen users, it can only use 1 MetaMask account. This wallet also pays the gas fees required when using dApps. Not all operations require a gas fee.

Smart contracts are developed using the HardHat framework [31], which is a framework for developing smart contracts. In this study, HardHat is used to install (deploy) smart contracts on EVM. 2 smart contracts are being developed, the MainContract (Population Management System Contract), which will be fully connected to dApps. Then the MainContract will relate to the SubContract (Citizen National Identification Contract) used to store data.

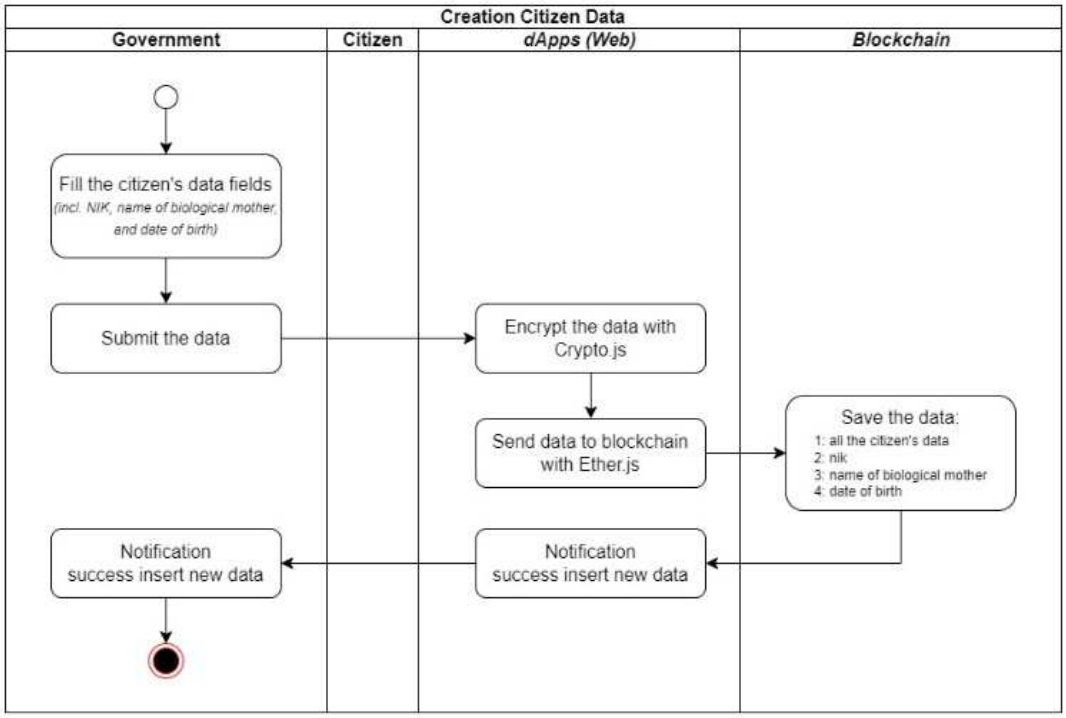

Fig. 6 Activity diagram - Creation of citizen data

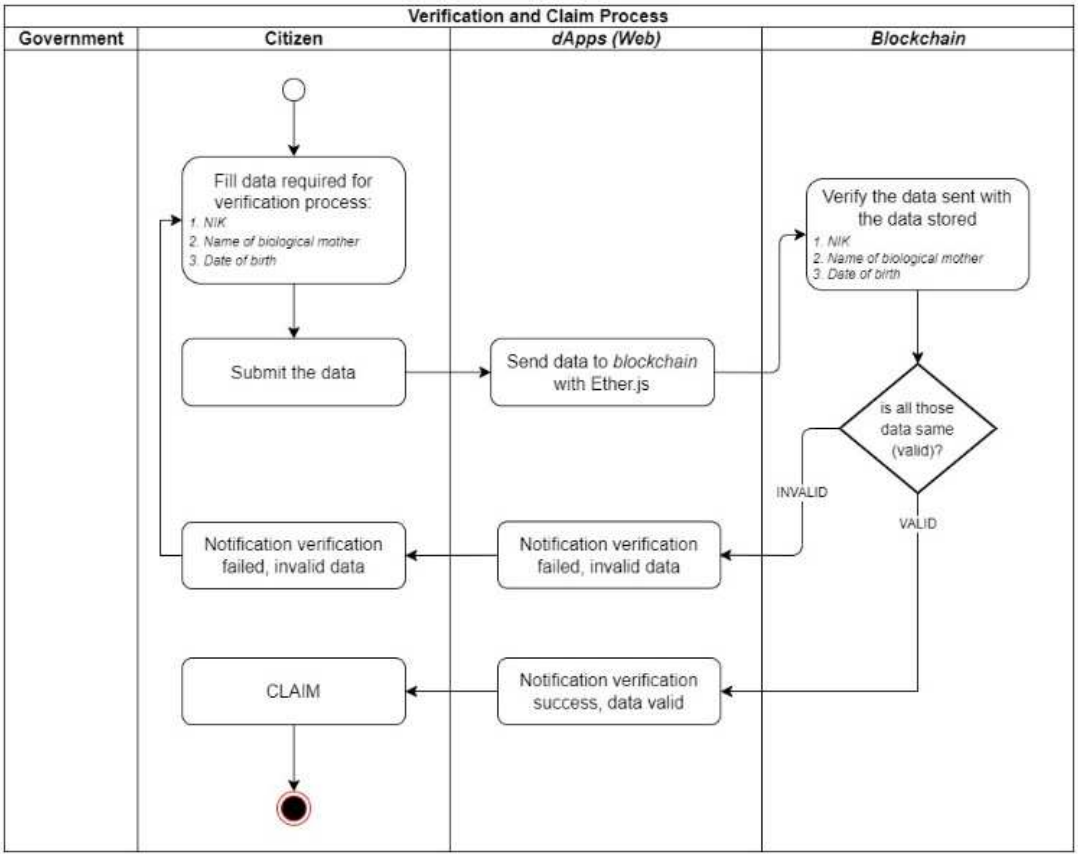

Fig. 7 Activity diagram - Citizen data verification and claim

Several configurations are done to ensure that the permissioned blockchain is implemented. The first is using the private modifier (Fig. 8) on all attributes to ensure that they cannot be accessed publicly. The second is creating an attribute owner (Fig. 8), which contains data on the contract owner. The third is creating a modifier function (Fig. 9) to check whether the owner/account who calls the smart contract function has access.

```
address private _owner;
```

Fig. 8 Example of private attribute

Fig. 9 shows that the addMasterData() function can only be accessed when the function caller is the smart contract's owner. Blockchain has the concept of openness, meaning anyone can access the smart contract installed on an EVM. Therefore, this configuration is required to restrict access.

```
modifier onlyOwner() {
  require(
    _owner == msg.sender,
    "Only Owner Allowed to Perform This
Action"
  );
  _;
}

function addMasterData(…) public onlyOwner
{
  …
}
```

Fig. 9 Example of modifier function and the implementation

Fig. 10 illustrates the status of population data which helps show the state of population data. There are four statuses, (1) UNCLAIMED, meaning that the Government has just formed the data; (2) VERIFIED, meaning the Citizen has verified his data by entering NIK data, date of birth, and birth mother's name; (3) CLAIMED, meaning that Citizen has made a data claim by making his account the owner of the contract so that he has access to the data; (4) UPDATED, meaning that the Government changes data. In the data claim process, there is a change of the SubContract owner, previously owned by the Government, which becomes owned by the Citizen.

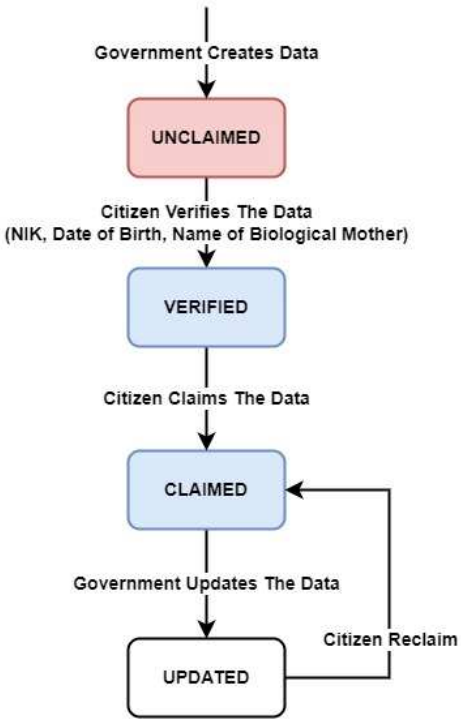

Fig. 10 Population data statuses

Fig. 11 illustrates the process of data encryption and decryption. Process a) is a data encryption process, and process b) is a data decryption process. The account wallet's key is helpful when encrypting and decrypting. Each data is locked with a wallet key, meaning only the account that locks it can access it. It maintains the privacy of user data.

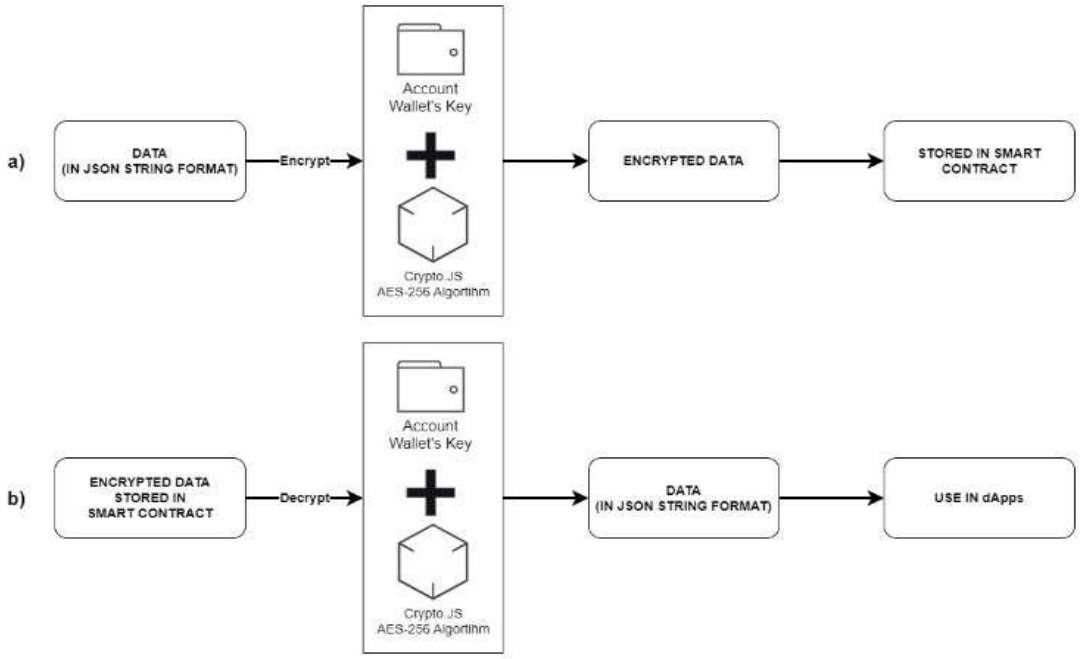

Fig. 11 Encryption-decryption process

Fig. 12 shows what data (plain text) will be encrypted on dApps. In addition, it offers data (encrypted text) stored in smart contracts. The plain text contains JSON string resident data. At the same time, the key is an example of a wallet's key from a MetaMask account. The encrypted text is data stored in the SubContract. The encryption and decryption process uses the Crypto.JS library with the AES-256 algorithm.

Plain Text	Key	Encrypted Text
{"nik":"001122334455667788",..., issuedDate":"2023-03-28"} *(Citizen data in string JSON format)*	0x50239e16...Bd4ab39d26 *(42 characters consist number and letter)*	U2FsdGVkX1+...0HL7Kh3Q==

Fig. 12 Example of encrypted data

In this study, smart contracts are installed on the Sepolia Testnet. To establish a smart contract, a wallet account is required. The account used is a government account. Fig. 13 shows the Government dashboard dApps connected to the Sepolia Testnet. This image also shows that a MetaMask account is required to access the dashboard page.

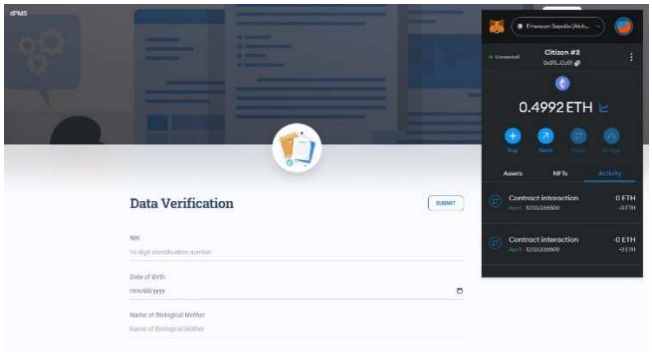

Fig. 13 Government dashboard

Meanwhile, Fig. 14 shows the Citizen form for data verification. As with the Government, access requires a MetaMask account. Based on the dApps infrastructure design (Fig. 5), the Government and Citizen websites are made differently.

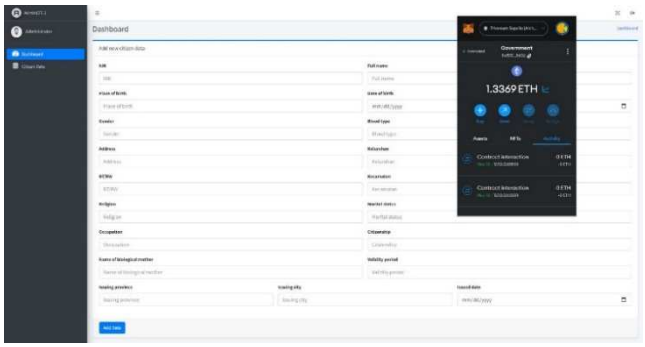

Fig. 14 Citizen verification form page

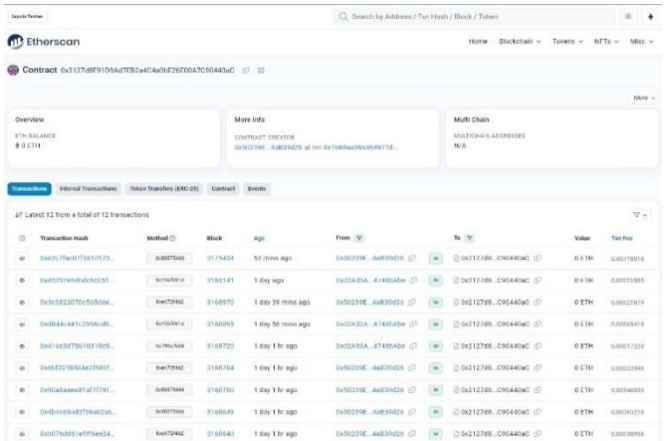

Fig. 15 Etherscan

One of blockchain's advantages is that it has a recorded history of changes/access (transactions). Fig. 15 shows the transaction history of the contracts made. Every recorded transaction is publicly viewable on Etherscan (https://sepolia.etherscan.io/) by entering the platform or smart contract address.

Vulnerability testing using OWASP ZAP was carried out to determine the vulnerability to attacks from this design. Testing is carried out by entering the website address into an automated scan tool. Then OWASP ZAP will automatically scan the website.

TABLE I
OWASP ZAP SUMMARY OF ALERTS REPORT

No	Risk level	Number of alerts
1	High	0
2	Medium	6
3	Low	3
4	Informational	4
5	False Positives	0

Table 1 shows the scanning report on vulnerability testing results using the OWASP ZAP tool. No high alerts or false positives were found.

TABLE II
ALERTS DETAIL

No	Name	Risk level	Number of instances
1	Absence of Anti-CSRF Tokens	Medium	12
2	Application Error Disclosure	Medium	2
3	CSP: Wildcard Directive	Medium	12
4	Content Security Policy (CSP) Header Not Set	Medium	168
5	Missing Anti-clickjacking Header	Medium	168
6	Vulnerable JS Library	Medium	11
7	Cross-Domain JavaScript Source File Inclusion	Low	11
8	Timestamp Disclosure - Unix	Low	816
9	X-Content-Type-Options Header Missing	Low	669
10	Content-Type Header Missing	Informational	156
11	Information Disclosure - Suspicious Comments	Informational	407
12	Modern Web Application	Informational	5
13	User Agent Fuzzer	Informational	1872

Several warnings are present for the middle and lower levels. These warnings are caused by the libraries used, such as Bootstrap, data tables, font-awesome, etc. Table 2 provides more complete information.

TABLE III
COMPARISON OF GAS USAGE FEES PER METHOD

No	Method	Ethereum Sepolia Gas fee (ETH)	Ethereum Sepolia Gas fee (USD)	Polygon Mumbai Gas fee (MATIC)	Polygon Mumbai Gas fee (MATIC)
1	Deployment main contract (1 smart contract)	6.40 E-03	1.14 E+01	4.24 E-02	2.54 E-02
2	The government creates new citizen data (sub-main contract created)	3.52 E-03	6.24 E+00	4.27 E-03	2.56 E-03
3	Government updates citizen data	2.49 E-04	4.41 E-01	3.71 E-04	2.22 E-04
4	Citizen verifies data	1.43 E-04	2.53 E-01	1.58 E-04	9.46 E-05
5	Citizen claims data	6.89 E-04	1.22 E+00	8.25 E-04	4.95 E-04
6	Citizen re-claims data	2.23 E-04	3.96 E-01	2.96E-04	1.77 E-04

Table 3 compares gas fee usage per method on 2 blockchains, Ethereum Sepolia and Polygon Mumbai. On Ethereum, Sepolia uses ETH with a conversion of 1 ETH = 1773.93 USD (data on May 13, 2023, at 01.17). Meanwhile, Polygon Mumbai uses MATIC with a conversion of 1 MATIC = 0.6 USD (data on May 23, 2023, at 22.38). Both chains are similar in using gas fees (ETH/MATIC); the method that requires the most gas fees is contract installation (number 1), which occurs once. However, there is a visible difference in method number 1; Polygon Mumbai is more numerous than Ethereum Sepolia. When converted to USD, Polygon Mumbai is much lower. Fig. 16 illustrates the comparison of gas fees between Ethereum Sepolia and Polygon Mumbai.

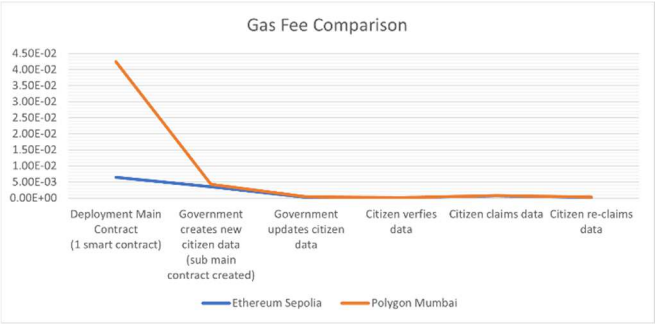

Fig. 16 Comparison of Gas Fee Usage for Ethereum Sepolia and Polygon Mumbai

IV. CONCLUSION

The population management system dApp prototype can be implemented with the design made. Therefore, the advantages of blockchain can be seen, one of which is a visible transaction history. Next, data storage in the blockchain is carried out with AES-256 encryption with the owner's account key first so the owner can only see the data. Based on the test results, no dangerous warnings are found. From the gas fee usage data, the process of creating new data by the Government requires many gas fees compared to other methods that cannot happen once. This developed model still uses permissioned blockchain, meaning access restrictions exist. You can implement a permissionless blockchain for population information systems for further development. In addition, multi-accounts should be implemented, especially for government users.

REFERENCES

[1] Badan Pusat Statistik, "Potret Sensus Penduduk 2020 Menuju Satu Data Kependudukan Indonesia," 2021. [Online]. Available: https://www.bps.go.id/publication/2021/01/21/213995c881428fef20a18226/potret-sensus-penduduk-2020-menuju-satu-data-kependudukan-indonesia.html.

[2] Tatyana, E. Oktaviani, and Syafwan, "Sistem Pengelolaan Kearsipan Debitur di Unit ADC PT. Bank Negara Indonesia (Persero) Tbk. Kantor Cabang Margonda," *Kompleksitas*, vol. 09, no. 2, pp. 12–19, 2020.

[3] Badan Pusat Statistik, "Statistik Telekomunikasi Indonesia 2022," 2023. [Online]. Available: https://www.bps.go.id/publication/2023/08/31/131385d0253c6aae7c7a59fa/statistik-telekomunikasi-indonesia-2022.html.

[4] K. Siste et al., "The Impact of Physical Distancing and Associated Factors Towards Internet Addiction Among Adults in Indonesia During COVID-19 Pandemic: A Nationwide Web-Based Study," Frontiers in Psychiatry, vol. 11, Sep. 2020, doi:10.3389/fpsyt.2020.580977.

[5] A. E. Istiqoh, A. Nurmandi, I. Muallidin, M. J. Loilatu, and D. Kurniawan, "The Successful Use of the PeduliLindungi Application in Handling COVID-19 (Indonesian Case Study)," in *Proceedings of Seventh International Congress on Information and Communication Technology*, vol. 3, 2023, pp. 353–363.

[6] F. Illia, M. P. Eugenia, and S. A. Rutba, "Sentiment Analysis on PeduliLindungi Application Using TextBlob and VADER Library," Proceedings of The International Conference on Data Science and Official Statistics, vol. 2021, no. 1, pp. 278–288, Jan. 2022, doi:10.34123/icdsos.v2021i1.236.

[7] R. Dhianty, "Kebijakan Privasi (Privacy Policy) dan Peraturan Perundang-Undangan Sektoral Platform Digital vis a vis Kebocoran Data Pribadi," *Scr. J. Kebijak. Publik dan Huk.*, vol. 2, no. 1, pp. 186–199, 2022.

[8] B. Napitupulu, "Supreme Court Decisions on Public Information and Personal Data Protection," Indonesia Private Law Review, vol. 3, no. 1, pp. 25–40, Jun. 2022, doi: 10.25041/iplr.v3i1.2559.

[9] M. B. Zaman, I. B. Pamungkas, and W. A. Wibowo, "Pengaruh Privasi Dan Keamanan Terhadap," *Sci. J. Reflect. Econ. Accounting, Manag. Bus.*, vol. 5, no. 4, pp. 891–902, 2022.

[10] D. Sheridan, J. Harris, F. Wear, J. Cowell, E. Wong, and A. Yazdinejad, "Web3 Challenges and Opportunities for the Market," pp. 1–7, Sep. 2022, [Online]. Available: http://arxiv.org/abs/2209.02446.

[11] K. Wu, Y. Ma, G. Huang, and X. Liu, "A first look at blockchain-based decentralized applications," Software: Practice and Experience, vol. 51, no. 10, pp. 2033–2050, Oct. 2019, doi: 10.1002/spe.2751.

[12] M. Xu, X. Chen, and G. Kou, "A systematic review of blockchain," *Financ. Innov.*, vol. 5, no. 1, 2019, doi: 10.1186/s40854-019-0147-z.

[13] I. T. Javed, F. Alharbi, B. Bellaj, T. Margaria, N. Crespi, and K. N. Qureshi, "Health-ID: A Blockchain-Based Decentralized Identity Management for Remote Healthcare," Healthcare, vol. 9, no. 6, p. 712, Jun. 2021, doi: 10.3390/healthcare9060712.

[14] L. Stockburger, G. Kokosioulis, A. Mukkamala, R. R. Mukkamala, and M. Avital, "Blockchain-enabled decentralized identity management: The case of self-sovereign identity in public transportation," Blockchain: Research and Applications, vol. 2, no. 2, p. 100014, Jun. 2021, doi: 10.1016/j.bcra.2021.100014.

[15] S. K. Panda and S. C. Satapathy, "An Investigation into Smart Contract Deployment on Ethereum Platform Using Web3.js and Solidity Using Blockchain," in *Advances in Intelligent Systems and Computing*, 2021, pp. 549–561.

[16] Y. Faqir-Rhazoui, M.-J. Ariza-Garzón, J. Arroyo, and S. Hassan, "Effect of the Gas Price Surges on User Activity in the DAOs of the Ethereum Blockchain," Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems, May 2021, doi:10.1145/3411763.3451755.

[17] K. Wust and A. Gervais, “Do you Need a Blockchain?,” 2018 Crypto Valley Conference on Blockchain Technology (CVCBT), Jun. 2018, doi: 10.1109/cvcbt.2018.00011.
[18] P. De Filippi, C. Wray, and G. Sileno, “Smart contracts,” Internet Policy Review, vol. 10, no. 2, Apr. 2021, doi: 10.14763/2021.2.1549.
[19] Z. Zheng et al., “An overview on smart contracts: Challenges, advances and platforms,” Future Generation Computer Systems, vol. 105, pp. 475–491, Apr. 2020, doi: 10.1016/j.future.2019.12.019.
[20] A. Fadlil, I. Riadi, and A. Nugrahantoro, “Data Security for School Service Top-Up Transactions Based on AES Combination Blockchain Technology,” Lontar Komputer : Jurnal Ilmiah Teknologi Informasi, vol. 11, no. 3, p. 155, Dec. 2020, doi:10.24843/lkjiti.2020.v11.i03.p04.
[21] M. Abraham, H. Am, C. Srinivasan, and D. K. Namboori, “Healthcare security using blockchain for pharmacogenomics,” *J. Int. Pharm. Res.*, vol. 6, pp. 529–533, 2019.
[22] A. G. de Moraes Rossetto, C. Sega, and V. R. Q. Leithardt, “An Architecture for Managing Data Privacy in Healthcare with Blockchain,” Sensors, vol. 22, no. 21, p. 8292, Oct. 2022, doi:10.3390/s22218292.
[23] A. H. Mohsin et al., “Based blockchain-PSO-AES techniques in finger vein biometrics: A novel verification secure framework for patient authentication,” Computer Standards & Interfaces, vol. 66, p. 103343, Oct. 2019, doi: 10.1016/j.csi.2019.04.002.
[24] N. Sundareswaran, S. Sasirekha, I. J. Louis Paul, S. Balakrishnan, and G. Swaminathan, “Optimised KYC Blockchain System,” 2020 International Conference on Innovative Trends in Information Technology (ICITIIT), Feb. 2020, doi:10.1109/icitiit49094.2020.9071533.
[25] B. Mburano and W. Si, “Evaluation of Web Vulnerability Scanners Based on OWASP Benchmark,” 2018 26th International Conference on Systems Engineering (ICSEng), Dec. 2018, doi:10.1109/icseng.2018.8638176.
[26] Nurbojatmiko, A. Lathifah, F. Bil Amri, and A. Rosidah, “Security Vulnerability Analysis of the Sharia Crowdfunding Website Using OWASP-ZAP,” 2022 10th International Conference on Cyber and IT Service Management (CITSM), Sep. 2022, doi:10.1109/citsm56380.2022.9935837.
[27] R. S. Devi and M. M. Kumar, “Testing for Security Weakness of Web Applications using Ethical Hacking,” 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184), Jun. 2020, doi: 10.1109/icoei48184.2020.9143018.
[28] I. F. Ashari, V. Oktarina, R. G. Sadewo, and S. Damanhuri, “Analysis of Cross Site Request Forgery (CSRF) Attacks on West Lampung Regency Websites Using OWASP ZAP Tools,” Jurnal Sisfokom (Sistem Informasi dan Komputer), vol. 11, no. 2, pp. 276–281, Aug. 2022, doi: 10.32736/sisfokom.v11i2.1393.
[29] T. R. Chowdhury, Md. Yusuf, P. Kundu, S. Chakraborty, and N. Biswas, “Crypto Pay: Design of Public Blockchain Platform,” American Journal of Electronics & Communication, vol. 3, no. 3, pp. 11–15, Jan. 2023, doi: 10.15864/ajec.3303.
[30] S. Sharma, K. Singla, G. Rathee, and H. Saini, “A Hybrid Cryptographic Technique for File Storage Mechanism Over Cloud,” 2020, pp. 241–256.
[31] S. M. Jain, “Hardhat,” in *A Brief Introduction to Web3*, Berkeley, CA: Apress, 2023, pp. 167–179.