 DOI: 10.35382/TVUJS.14.4.2024.92

LEVERAGING BLOCKCHAIN TECHNOLOGY IN SMART LAND MANAGEMENT

Ngoc Hong Tran[1*], Bao Gia Pham[2], Quoc-Binh Nguyen[3], Hoan Le[4], Duy-Hung Le[5], Dung Hai Dinh[6]

Abstract – *Land and real estate, being high-value assets, it is crucial to have authentic documents to identify and prove who is indeed the owner. However, the existing land registration processes and the use of paper certification in Vietnam and most other countries involve many vulnerabilities, leading to many land dispute cases and illegal manipulation of public land. Also, the current titling and transferring ownership procedure is quite laborious since it involves manually storing and safekeeping a large volume of registers in written form. To address this problem, alternative solutions have been proposed, including digitalizing records and storing data in a centralized database. Nonetheless, with such approaches, data transparency and its availability may not be guaranteed since data tampering and unauthorized fraudulent change could occur in poorly maintained databases, or the system could be suspended for unexpected reasons if the servers are not appropriately backed up. In this study, a system that accelerated the land titling process and met all the required characteristics in terms of transparency and availability of data was developed. The proposed system was implemented on Ethereum blockchain, which kept all of the transactions during the land assets ownership transfer on a distributed network, thereby decreasing the cases of fraud in the process. The study then analyzed and presented several limitations of the current implementation and finally provided some possible suggestions for future improvement.*

Keywords: blockchain, Ethereum, hash, secure sustainable management, smart land management.

[1,2,5,6]Vietnamese-German University, Vietnam
[3]Ton Duc Thang University, Vietnam
[4]Mien Dong Innovative Technology University, Vietnam
*Corresponding author: ngoc.th@vgu.edu.vn

Received date: 30th June 2024; Revised date: 29th November 2024; Accepted date: 02nd December 2024

I. INTRODUCTION

Real estate (RE) is a unique and diverse asset class consisting of land and improvements in itself, such as buildings, roads, and structures. In recent years, RE, especially lands, have grown increasingly scarce and precious in Vietnam as the country's industrialization and modernization have accelerated. As well, Vietnam has had one of the world's fastest-growing economies and a significant reduction in poverty over the previous two decades. With such a fast-paced development focusing on services and manufacturing, Vietnam is promoting the transfer of agricultural land to more profitable non-agricultural land [1]. Statistical evidence in 2021 shows that roughly 37% of the total population, or 36.6 million people out of 98.9 million people, live in urban regions [2]. Together with that transition, land or real estate in general, is considered a key source of money that assures a better living quality as well as brings high-profit investment opportunities. As a result, ownership of any piece of land is highly contested, which also means the need for precise documents or a transparent system to certify ownership records is crucial. However, the prior land registration process involves manually safekeeping a large registration volume in written form. The main issue with this technique of land ownership preservation is that any future reference that needs to be taken from these hard copies would require too much time and effort. In addition, with such manual procedure, the ownership records are prone to mistakes, which has resulted in many cases of land disputes, illegal manipulation of public land, unauthorized fraudulent change, and others. Due to the lack of transparency in the land registration process, land administration has been identified as one of the most corrupt sectors in Vietnam. According to several studies (e.g., the Asia Foundation – T&C Consulting, 2014 [3]; World Bank 2012; 2013 [4, 5]), various flaws from this procedure were misused by land speculators in cooperation with high-position authorities to alter documents and afterward claim fake ownership or coercive land grabbing, resulting in the highest number of a civil dispute in the legal system.

Since 2010, twenty severe cases of land conflict have gained international media attention, according to the Armed Conflict Location and Event Data Project (ACLED) (2020) dataset [6]. In response to these above-mentioned problems, the register data maintenance has to be automated by removing the need for paper records. Like many other countries, the Vietnamese government has been making use of technological advancements to transform administrative data into a more digitalized form to adapt to the greater demand for transparency and security of data. Though the government has taken some initial moves towards digitalizing administrative formalities, it is still in the early stages of development, and its use case is just limited to archival tasks. Moreover, a significant proportion of data in the public sector in Vietnam is stored on disparate systems, which are still not yet synchronized [7]. This still could be a great potential for the incidence of inaccuracies or modification of legal documents by corrupted people. As a result, possessing a sophisticated land registration system can be a good foundation for integrating with other land-related activities (e.g., purchase, transfer, sell by auction, inheritance, and partition deeds) while highly prioritizing transparency and integrity is extremely important.

Blockchain, a distributed ledger technology (DLT), can be a good candidate for implementing a reliable land registration solution. Blockchain is a distributed database of records of all transactions that have been verified and agreed upon by a majority of the participants in the system following its consensus algorithm (e.g., Proof-of-Work, Proof-of-Stake). This protocol works in such a way that it is, in practice, infeasible to alter or falsify the already appended data by a single node or group of nodes [8]. A land title's intrinsic nature is linked to its historical ownership records and how it was altered over time, and its integrity is preserved by the chain of documents, which perfectly matches the fundamental concept that Blockchain technology is striving for. Implementing Blockchain in the land titling system promotes the following aspects:

1) Immutability: With the use of cryptography, data in Blockchain can only be updated in the 'append-only' manner. Hence, storing ownership data on Blockchain is a way to create immutable records, which could prevent the unauthorized fraudulent change issue.

2) Reliability: Data in Blockchain is stored in a distributed ledger that keeps all historical records of transactions to be shared across a peer-to-peer network. Therefore, implementing applications using Blockchain guarantees the quality and availability of data.

In this work, the proposed implementation is developed on Ethereum public blockchain, which describes itself as a blockchain-based, open-source, distributed platform that features smart contract functionality [9]. Within the scope of the work, the application is limited to the archival phase in the complete titling procedure. Moreover, this paper also includes explanations of the overall architecture, as well as technologies, methodologies, and tools used across the entire development process. Please refer to Table 1 for detailed contribution.

The paper is organized into six sections. Section I expresses the motivation and authors' contribution to this work. Related works are presented in Section II. Section III describes the details of blockchain technology. The land management systems are demonstrated in Section IV. In Section V, the implementation of the proposed land management system using the Amsterdam's

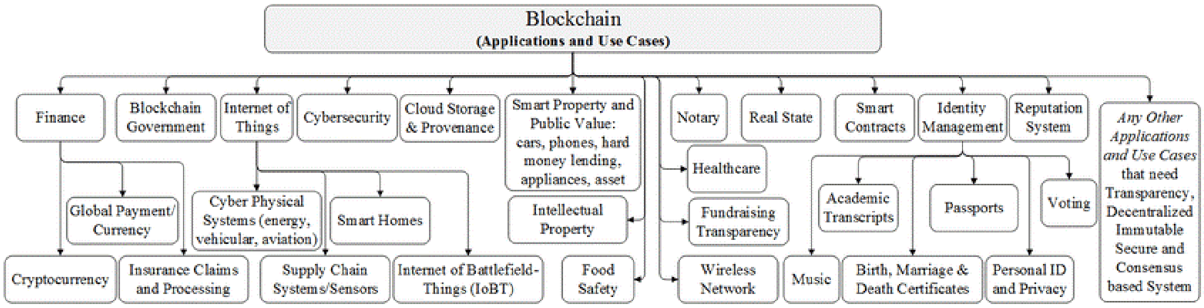

Fig. 1: Applications of blockchain in different domains

Table 1. Comparison between current technique and blockchain-based proposal

Parameters	Current techniques	Proposed blockchain solution
Methods	Manual process, keep track of ownership on paper documentation	Digital process, ownership of land is securely stored on blockchain
Time/Effort	Tedious process, require lots of time and efforts	As the manual process condenses in the digital application, the registration could be quicker, and the manipulation of records would save lots of time/efforts
Intermediaries /Brokers Need	Many intermediaries involved	Only data processors (administrators) are involved
Fraud/Error	Huge cases of fraud, manipulation of flaws in the current system	Ownership and lands specification are stored on the blockchain, which could be easily referenced and secured from being tampered

land dataset is mentioned. The future works and conclusion are presented in Section VI.

II. RELATED WORKS

The first revolution of blockchain emerged in 2008 when Nakamoto [10], an anonymous person or organization behind Bitcoin, introduced how blockchain technology, a peer-to-peer distributed timestamp server with a linked structure could be used resolve to maintain the transaction order and avoid the double-spending problem [10]. At the time of this writing (May 2022), it is almost a decade since the advent of Bitcoin; blockchain applications in the cryptocurrency have been widely accepted by the community and gradually become a part of the mainstream economy [11]. In recent years, the progress of smart contracts in blockchain, as well as the sequence of their function extension, has enabled the application to expand into many different fields. The world is becoming more familiar with smart contract-based blockchain applications in various domains, including cryptocurrency, finance, healthcare applications, supply chain management, voting and fundraising mechanisms, Internet of Things (IoT), public sector (e-government), insurance processing, and real estate [12], some of them are just in the phase of research and development. Yet, this research is still in the early stage of adoption. Figure 1 shows applications of blockchain in different domains. As a result, countries and researchers are taking first steps in investigating and implementing blockchain into e-government specifically; one of these fields is land title management. The following is a list of existing works and studies on applications of blockchain in real estate management and other related sectors will be presented.

In an article from Khulna University in 2020, Alam et al. [13] have conveyed a comprehensive study in which they presented a blockchain-based prototype system implemented on the Ethereum public network for land titling system. Besides, the study also examined the existing land registration procedure in Bangladesh along with its loopholes. According to their analysis, the proposed model could resolve the problems in the existing method by decreasing the number of travels, and the overall cost of operation. Moreover, the model providing easy access to the information. In another paper in 2018 from Graglia et al. [14], the authors provided the methods and requirements for implementing blockchain technology in real estate and land registries. They also present a conceptual framework for gradually implementing blockchain solutions at different integration levels. However, as recommended in the paper of Eder [15], it is not practicable to transfer an existing model to blockchain without setting out an incremental policy for real-world adoption. The authors then gave their suggestions for policy changes for the local networks, institutional reform, and cost analysis of making these moves.

In addition to theoretical models, in recent years, selected countries have put blockchain -based land registration solutions into reality and functioning in real life. In 2016, to resolve the problem of corruption and land disputes, the Republic of Georgia in cooperation with Bitfury, which is a full-scale blockchain company, launched a large-scale pilot project to migrate their land ownership registry system to a blockchain platform. Following its success in the first phase, in 2017, a more extensive system with further integration in the process of land sales and transfers was initiated. As a result, in 2018, 1.5 million land titles were reported to be published and operated on blockchain, which provides the immutability and transparency of the data. Their system supports land registration, purchases, and mortgage features, which makes the overall time necessary to handle land tiling operations considerably decreased [16]. In other fields that require immutability and transparency of the data, researchers also have been investigating the integration of blockchain into their solutions. In an article published in 2021, Haleem et al. [17] presented a blockchain model in healthcare that is integrated with the help of IoT and wearable devices to store and update patient data in real-time. They also depicted that in healthcare systems, blockchain could be effectively utilized as a solution to preserve and exchange patient data across hospitals, diagnostic laboratories, pharmacies, and clinicians, which can eliminate the deception in clinical trials, avoid the fear of data manipulation, and maintain synchronized correctness of data.

III. BLOCKCHAIN TECHNOLOGY

A. *Structure*

Data after validating and propagating across nodes in blockchain are stored in a block. Each block within the network is identified by a hash produced by the cryptographic hash of its header. The chosen hashing algorithm depends on the implementation of different Blockchains, e.g., SHA-256 in Bitcoin or Keccak-256 in Ethereum [10, 18]. As shown in Figure 2, every single block is a composite of header and body in which a set of transactions or data is stored [19]. The block header contains three sets of metadata.

First, the 'Hash of Previous Block' is the most crucial factor in ensuring the immutability of the entire network. The field 'Hash of Previous Block', as the name suggested, affects the current blocks' hash. In other words, each block's header contains the hash value of its parent, and this connection extends back until the first block, which is known as the genesis block. The entire identity of the current and the subsequent blocks would change if any piece of information in any previous block changes [20]. The second set of data in the header is 'Time Stamp' and 'Nonce' which relate to the approximate creation time of the block and the solution for the proof-of-work algorithm, respectively. The final piece of data stored in the header is the root value of the Merkle tree. Merkle tree is a binary hash tree used for efficiently summarizing and verifying the entire set of transactions. Each value in a node of the current level is the concatenation of the hash of its child nodes. Since only a small change in transaction can result in a completely different value in the Merkle root, the verification of transactions could be performed by only comparing the root rather than comparing all transactions in the block [19].

This underlying connection of the blocks assures that if a block has several subsequent blocks, it is infeasible to be modified without causing all successive blocks to be recalculated. Since such recalculation would require massive computation, data in blockchain could be considered immutable within this mechanism cosidered

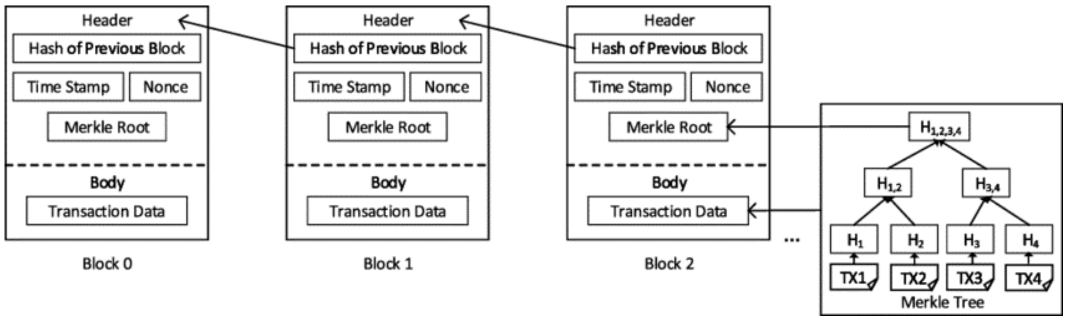

Fig. 2: Block structure [19]

a major characteristic of blockchain or Bitcoin's security [20].

B. Applied-cyrptography for blockchain

Such immutability and transparency of blockchain are guaranteed thanks to the use of cryptography at the lowest level and the collaboration between different approaches. The main cryptographic tools that were initially utilized by the early blockchain technology are public key cryptography and hash function [10].

Public key scheme

The public key algorithm, also known as asymmetric encryption, is used to verify that the identity of the transaction was indeed from the right person. In this algorithm, the encryption key and decryption key, also known as private and public key, respectively, are used. The private key is generated by a random number algorithm, and the associated public key is calculated by an irreversible algorithm. The private key in the blockchain is kept secretly in a blockchain wallet, which can be either a physical device or a software wallet. The private key is used to sign the message, which is called the digital signature, and the associated public key is used to confirm the identity of the transaction. As presented in Figure 3, every time users send a transaction, they first sign the hashed message with their private key, then the result will be sent to miners and afterward is verified by decrypting using their public key. Miners finally check if the hashed value of the transaction is equal to the decrypted value or not. If they are identical, the transaction is verified. The algorithm enables users to create digital signatures in every transaction without revealing their real-world identity [10].

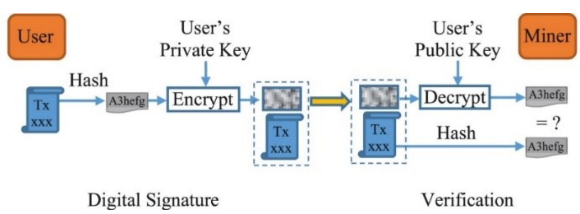

Fig. 3: Digital signature used in blockchain transaction [21]

Hash function

Hash function, as previously mentioned, is the crucial technology of blockchain. A cryptographic hash function is a mathematical function that accepts data input of an arbitrary size and provides a unique fixed-size output usually represented as a string in hexadecimal notation. Though the input of the hash function could be at any length, the output of the function always maintains the same length. The length of the output depends on the specific hash function. For example, SHA-256, one of the secure hashing algorithm family members, always produces 56 bits of data regardless of how short or long the given input is [21]. In addition, the output of the hash function is deterministic; in other words, the same input provided to the same hash function always produces the same result. If there are any modifications or changes even in a single bit of data, the hash result would be entirely different. Finally, the last characteristic of the hash function that made the foundation of the blockchain is preimage resistance, which means it is not difficult to calculate to produce a hash value given an input. However, it is computationally infeasible to reverse engineer the original input that maps to the given output.

C. Consensus algorithm

One of the essential features that ensure the integrity and synchroneity of blockchain is the consensus algorithm. The algorithm is used to ensure that transactions are appropriately propagated across all nodes in the networks following the set of predefined rules. In addition, it is also used to validate data within a block to avoid problems like double-spending [22]. Proof-of-Work (PoW), proposed by Nakamoto in 2008 [10], was used to implement the first consensus algorithm. Due to the drawbacks of the algorithm and the growing interest in blockchain, various new algorithms have emerged fit for different applications. The following sections will elaborate on the basic idea of the most common consensus algorithm, which is Proof-of-Work (PoW) and Proof-of-Stake (PoS).

Proof-of-Work (PoW)

Blockchain in the early stage, including Bitcoin and Ethereum both rely on the PoW algorithm. Whenever transactions are distributed to the blockchain they will be added to the newly created block. The process of a new block was created called the mining process and miner nodes would be involved in this process. In PoW, all mining nodes on the network try to compete to solve a difficult mathematical problem based on a cryptographic hash algorithm. Given the header of the previous block, the nonce which stands for ‘number only once’, the transactions included in the new block, miners find the hashed value of these inputs with an array of initial consecutive zeros to successfully create a new block and earn a reward. With the preimage resistance characteristic of the hashing method as mentioned in the Hash function section, the only way to find the hashed output is by randomly trying different values of nonce until the satisfying output is found. Once the miner calculates the correct hash value for the new block, they immediately broadcast it to the networks. Other miners then verify the authenticity of the block by comparing the block address and the header (described in the Block structure section), including the nonce [23].

The processing of the algorithm on all participating nodes on the network is required for all actions on blockchain using PoW, which necessitates a large number of computer resources. This algorithm was totally ideal in the beginning when transactions and node participants were few. However, at present, it requires the processing of thousands of computers just to compute for a single nonce. This would result in unnecessary waste of energy and resources. In addition, with the increasing demand for usage over time, the difficulty of solving the hash value is expected to rise, resulting in higher processing requirements [23]. As shown in Figure 4, in May 2021, mining difficulty on Bitcoin blockchain was at its highest point, approximately 25 trillion terahashes. Figure 4 presents such huge processing power is being used to produce a new block for this particular cryptocurrency and the required computational power would be even higher due to the increasing difficulty of the algorithm over time.

Proof-of-Stake (PoS)

PoS is another prominent consensus algorithm, which requires much less computational power compared to PoW. Rather than competing to find the hashed result that satisfies the predefined rule, miners must stake a certain amount of money to be picked as the next block creator. The entrusted miner will be randomly selected afterward, and this process changes frequently, making it infeasible for others to figure out who is the next miner and manipulate its operation. The chance of getting chosen is proportional to the size of the stake; thus, a greater deposit increases the chances of winning. Nonetheless, the miner loses all his deposit if he breaks the

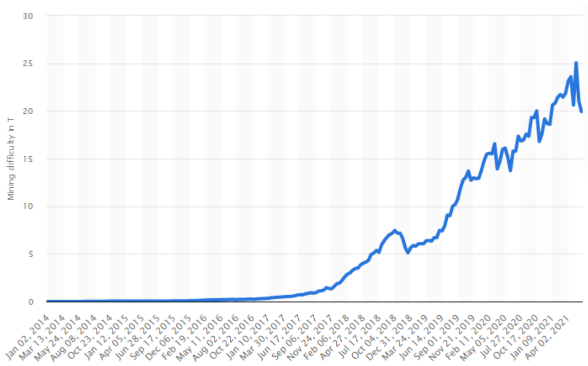

Fig. 4: Average mining difficulty of Bitcoin form January 2014 to May 13, 2021 (In Terahash) [24]

rule by tampering with the transaction included in the block. In addition, the rewards are always less than the staking amount, so there would be less chance and motivation for cheaters to compromise with the process. Though using this consensus mechanism, to some extent, goes against the nature of blockchain which is decentralization, e.g. some intensive capital corporations could deposit a much larger stake, resulting in an unbalanced situation, the consequences are still less severe than wasting unnecessary resources [23]. This algorithm will be promising in the future since it is about to be implemented in the first phase of the Ethereum 2.0 upgrade [25].

D. *Smart contract*

With such a fast pace revolution of blockchain in recent years, researchers have found a way to integrate customized programmable logic into blockchain networks in order to automate the contract process with less human intervention and third-party intermediate and enable faster transactions without supervision at every step of the whole procedure. The idea of a smart contract was initiated for the first time in 1994 by an American computer scientist Nick Szabo. The definition, according to Szabo's article was 'A computerized transaction protocol that executes the terms of a contract' [26]. The concept behind this was to turn the terms of a typical contract into a software program, which would then be embedded into hardware or software that enforced users to follow the contract's conditions; hence, eliminating the need for third-party agencies. Bitcoin could be regarded as the first blockchain to implement basic smart contracts, though the feature was just transferring money from one to another via its network. However, its initial idea was just barely a cryptocurrency, a smart contract with complicated logic cannot be supported. Later, Ethereum utilized this idea and made the first blockchain -based, Turing-complete network that supports advanced smart contract features and could be programmable using high-level programming language [9]. The following sections present the overview of knowledge, structures, and examples of smart contracts in Solidity, a language to write smart contracts on the Ethereum platform.

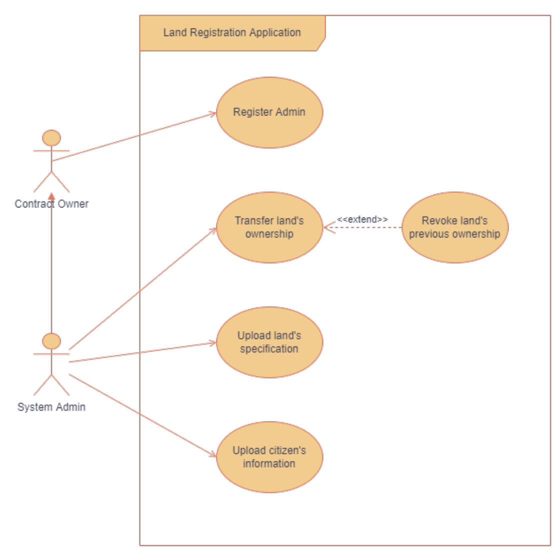

Fig. 5: Propose application's use case diagram

IV. BLOCKCHAIN-BASED LAND MANAGEMENT SYSTEM

This section goes through the operations of the application and the use case diagram, which shows how different types of users interact with the system. Then an overall architectural design and finally, the smart contract design and the utilized data model in the application will be presented.

A. *Roles and operators*

The application involves the record archival phase of the land registration process. Hence, the main users of the proposed application are administrators from authorities or authorized admins working in the land ministry, who directly be responsible for processing and manipulating the titling information. Figure 5 shows the use case diagram of the application. There are two types of users that are authorized in the application, which are ‘Contract Owner’ and ‘System Admin’. The following sections describe the role and interaction of these two types of users.

Contract Owner: This role represents the only account that deployed the smart contract to blockchain, which is allowed to register for the ‘System Admin’ role. The registered accounts are then stored directly on the Ethereum blockchain. Since the proposed application is limited to only authorized accounts that have been registered before, every operation in the smart contract will check for the ‘admin’ privilege before being executed. As shown in the diagram, ‘Contract Owner’ inherits all the actions of the ‘System Admin’, which means ‘Contract Owner’ has all the privileges of ‘System Admin’ who can create more admin users for the application.

System Admin: ‘System Admin’ is the main role that operates the application. The first action is to upload the land specification JSON file. The specification file contains the normalization geometry coordinator, the area, and the center of the land (see Figure 6), which then infer the location, the height, the boundary of the land plot, and the width in every direction. Coordinator land could be a polygon with several connected points in two dimensions. The uploaded file will then be stored in Interplanetary File System (IPFS).

Other actions that ‘System Admin’ can perform are to upload citizen profiles and transfer land ownership. The citizen profile uploading process requires the admin to fill in all information of citizens that will be uploaded and stored directly on the blockchain. With the given available land and citizen profile, then admins can transfer or register the ownership of the land to the citizen by ‘transfer land’s ownership’ action. If the transaction happens on land that has a prior owner, ‘revoke land’s previous ownership’ will be invoked before issuing the new ownership.

```json
{
    "id": 0,
    "area": 621,
    "paths": [
        [52.3743401602888, 4.8862385854049],
        [52.37425836757968, 4.886554134703491],
        [52.374206332690235, 4.88675492200334],
        [52.37420503591525, 4.886754319866858],
        [52.37414823931281, 4.886979008722971],
        [52.37421318873124, 4.88703745659352],
        [52.374283676641475, 4.886832599277177],
        [52.374305020928595, 4.886770564070238],
        [52.374352113275165, 4.886633711882074],
        [52.37444569059899, 4.886335518854553],
        [52.3743401602888, 4.8862385854049]
    ],
    "center": [52.37428163511373, 4.886648127889457]
}
```

Fig. 6: Sample land specification file

B. *Architecture*

This section provides a high-level architecture of the proposed application. The overview of the technologies implemented in the application will be listed. As demonstrated in Figure 7, there are three main components that interact with each other, which are the ‘Client Browser’, ‘Blockchain’ and ‘Storage’. The following paragraphs will go through each component and explain their connection within the system.

Client Browser: Client Browser allows the admin to interact with the system in a more intuitive way. The application on the client side is developed by NextJS, which is a framework that is built on top of ReactJS for developing single-page applications. In order to interact with Ethereum blockchain, browsers need to install the Metamask extension, which is an Ethereum provider that has the responsibility for connecting the client code with the Ethereum account,

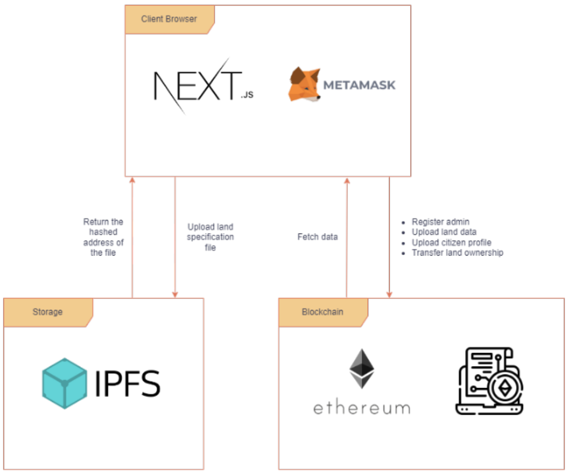

Fig. 7: System architecture

signing transactions, and interacting between the smart contract and the client browser.

Blockchain: Blockchain is the core component of the entire system. The proposed solution is built on the Ethereum platform and uses Solidity as the programming language for developing smart contracts. The main functionalities of the blockchain tier are handling the logic of the application, receiving the data input from the client side or vice versa, sending the data back to the front-end tier, authorizing and validating the privilege of the account, and finally archiving the record. The more specific explanations of the smart contract implementation will be presented in the smart contract design section.

Storage: One of the major drawbacks of Ethereum blockchain is that any operation performed, or any memory stored on-chain will be proportionally charged money based on computational power and amount of data, respectively [27]. Hence, to store thousands of ownership records, the application must be economical and cost-efficient in the long run. One alternative for data preservation will be IPFS, which is a peer-to-peer hypermedia protocol designed to store archival data. When files are uploaded using this protocol, they are split into smaller chunks, and their content is hashed and produces a unique fingerprint, namely content identifier (CID) [28]. In the proposed application, this technology is utilized to store land specifications (as presented in Figure 6.) attached to every land documentation. IPFS then produces a hashed value of CID, which is later stored on the blockchain.

C. *Smart contract*

This section presents the implementation of the smart contract used in the application. The first subsection provides some of the important data structures that are used to store and efficiently manage the logic, and then a set of functions handling the core logic of the main functionalities will be shown.

Figure 8 shows different variables with their associated data structures utilized in the smart contract. There are six major variables including 'lands', 'citizens', 'admins', 'landIdToTrxId', 'landTransferedTrxs', and 'citizenIdToOwnedTrxIds'. In the smart contract implementation, they are represented by mapping data structure which is mapped by an auto incremental unsigned integer as the key for accessing. This is a prevailing practice amongst Solidity developers since using mapping with auto incremental key, data when retrieving from the client side can be easily iterated and dynamically added.

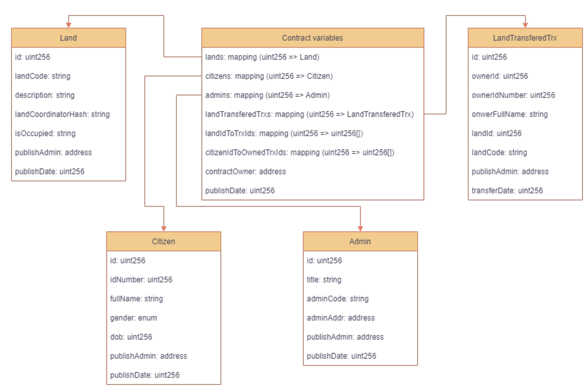

Fig. 8: Variables and data structures used in smart contract

Structure datatype

For each data structure, the unique fields of each 'struct' datatype are used in the smart contract.

- Land represents the land profile, which has the following properties:

+ *landCode*: String is the unique code that refers to the land plot.

+ *description*: String is the short description of the land plot.

+ *landCoordinatorHash*: String is the hashed value return from IPFS (mentioned in the Storage section) used to allocate to the file that contains the geometric coordinator of the land.

+ *isOccupied*: Book showsbook shows the occupation status of the land.

- Citizen represents the citizen entity, which has the following properties:

+ *idNumber*: uint256 is the ten-digit unique identification of the citizen.

+ *fullName*: String is the full name of the citizen.

+ *gender*: Enum uses the enumeration datatype, which has an intuitive naming value in the code base representing the gender of the citizen.

+ *dob*: uint256 is the Unix date of birth of the citizen.

- Admin represents the admin having access privileges to the application, which has the following properties:

+ *title*: String is the organizational title name for the position of admin in the government administrative system.

+ *adminCode*: String is the unique code representing the admin.

+ *adminAddr*: Address is the account address in Ethereum of the admin.

LandTransferedTrxis the detail of the transaction that is archived when ownership of the land is transferred. This includes the following properties:

+ *ownerIdNumber*: uint256 is the 10-digit identification number of the new owner.

+ *ownerFullName*: String is the new owner's full name.

+ *landId*: uint256 is the internal number id of the land.

+ *landCode*: String is the unique code that refers to the land code.

Smart contract variables

The following variables are defined to store the state of the smart contract:

+ *lands*: Mapping (uint256 => Land) is used to store all uploaded land profiles.

+ *citizens*: Mapping (uint256 => Citizen) is used to store all citizen profiles.

+ *admins*: Mapping (uint256 => Admin) is used to store the admin profile for the use of access control.

+ *landTransferedTrxs*: Mapping (uint256 => LandTransferTrx) is used to store all ownership transferring transactions of the system.

+ *contractOwner*: address is the address of the account that deploys the smart contract to Ethereum blockchain.

+ *publishDate*: uint256 is the Unix date that the contract is deployed.

To connect two different data types of land and citizen and manipulate their relationship, the following variables have been used:

+ *landIdToTrxIds*: Mapping (uint256 => uint256[]) is used to map the internal id of the uploaded land to an ids array of the ownership transactions. By using this variable, the system can keep track of all ownership records in the history of any land.

+ *citizenIdToOwnedTrxId*: Mapping (uint256 => uint256[]) is used to map the internal id of citizens in the application to an id array of the ownership transaction that is possessed by the citizen.

Core functions

Core functions are functions that handle the main functionality of the applications. In the proposed application, those are related to uploading land and citizen profile, registering for an admin account, and transferring land ownership. The upload functions are responsible for digitalizing the documentation, in which data is validated and refined, then archived in the blockchain. The admin register function is limited to just the contract owner to grant the admin permission for the given account. Finally, the land transfer ownership is used to grant the new ownership of a given land that was previously uploaded on the system to the new owner.

Figure 9 presents an example of a simplified flowchart diagram of the transfer ownership function. When clients send the request with the necessary data, including the Ethereum account address, the function first checks if the account calling the function has the admin privilege, it can process further. Otherwise, the function returns an error message and is terminated. Next, since the data is passed from the client-side, regardless of validation on the front end, the smart contract still has to revalidate the input again to ensure the accuracy of the data. Then, the function revokes the current ownership of the land if it is currently possessed. Otherwise, it directly grants new ownership of the land to the new owner and sends the response status back to the client-side.

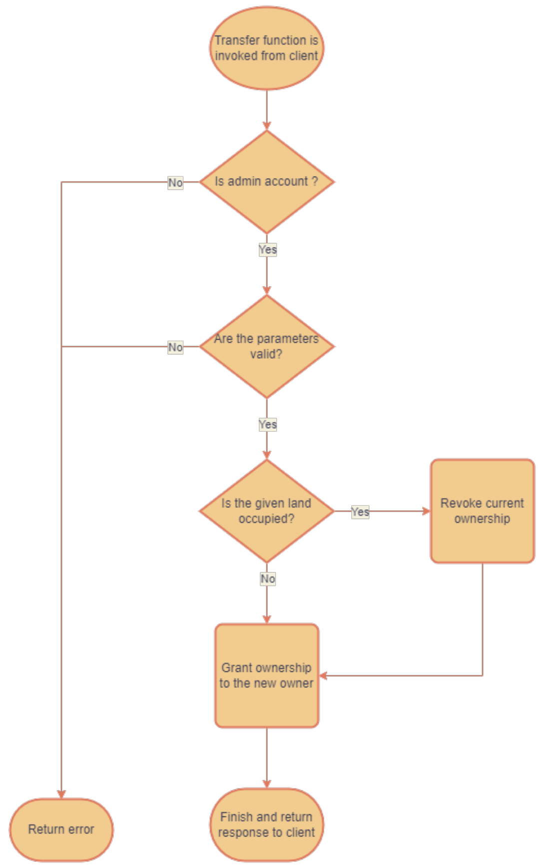

Fig. 9: Simplified flow chart diagram of transfer ownership function

V. IMPLEMENTATION

A. *Technological methodologies*

This work leverages technologies including Ethereum, Ganache, Truffle, and IPFS to implement the land management system.

Ethereum

Ethereum is an open-source, public blockchain-based that is built for multi-purpose decentralized applications. Launched in 2015, Ethereum is one of the very first blockchain to have a smart contract feature that focuses on the capability of automatic digital asset management, which makes the development of the asset managing application much more accessible [29]. Initially, Ethereum is a public permissionless blockchain-based platform implementing PoW (mentioned in the Proof-of-Work (PoW) section) consensus protocol called Ethash. Ethereum also supports a private platform as a configurable feature [9]. As PoW requires too much computational power and the transaction cost is noticeably high as the demand increases, Ethereum has been upgrading to version 2.0, in which PoS (explained in the Proof-of-Stake (PoS) section) is used as the consensus algorithm. The upgrade aims to enhance throughput efficiency, transaction cost, and the scalability of the Ethereum network [25]. Moreover, a comparison between Ethereum and other popular blockchain platforms such as Hyperleger Fabric and R3 Corda is demonstrated in Table 2.

In addition, smart contract in Ethereum blockchain, which can be developed with the Solidity programming language, is compiled and run on any node with preinstalled Ethereum virtual machine (EVM) [9]. Since blockchain is a technology that has just emerged and gained the community's attention in recent years, its popularity and community support must be crucial

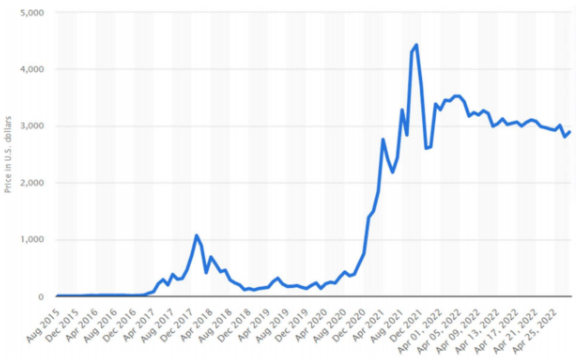

Fig. 10: Ethereum price per day from August 2015 to April 28, 2022 (in US dollars)

Table 2. Comparison of the three popular blockchain platforms: Ethereum, Hyperledger, R3 Corda [31]

Feature	Ethereum	HyperLedger	R3 Corda
Targeted industry	Cross Industry	Cross Industry	Financial
Mode of operation	Public	Private	Private
Decentralization	Decentralized	Partially	Partially
Consensus protocols	PoW / PoS	Pluggable	Notary-based
Smart contract Support	Yes	Yes	Yes
Cryptocurrency	ETH	N/A	N/A

Source: ConsenSys Software Inc

factors to consider when choosing a blockchain platform to implement [29]. Figure 10 shows the Ethereum token Ether (ETH) price. Released in 2015, Ethereum reached a peak and stayed steadily at a price of around 3,000 USD in the period of 2021–2022. This blockchain platform has been proven to be trusted and supported by the community. In terms of technical development, more mature and widely used blockchain networks will ensure the time and effort development efficiency of applications. Furthermore, Ethereum is the best candidate for a blockchain platform to be chosen for the proposed application as it has many tools and frameworks supporting a wide range of programming languages used for testing and development. More importantly, Ethereum's developers are actively researching and developing ways to improve the current chain. This will guarantee long-term support and improvement of the application in the long run.

```
module.exports = {
    initLands: async (contract) => {
        await contract.publishLand(
            "Land 1",
            "QmQi3QgPXjqCovd54CJH7fopaNq8h18ZSXcH7bxFBKYYVy"
        );
        await contract.publishLand(
            "Land 2",
            "QmUGRVskbKfjU7LBM7PtkvQYDyZC6mbUh8aKpPWiPdAGDK"
        );
    },
    initCitizens: async (contract) => {
        await contract.publishCitizen(
            1000000000,
            "Citizen Full Name 1",
            0,
            1645574400
        );
        await contract.publishCitizen(
            1000000001,
            "Citizen Full Name 2",
            0,
            1645574400
        );
    },
};
```

Fig. 11: Truffle console script used in the proposed application

Ganache

Ganache is a personal blockchain for rapid Ethereum distributed application development. Came with a set of development tools called Truffle Suit, Ganache could be considered a blockchain simulator that runs locally on the computer [32]. In the development process of the proposed application, Ganache was utilized during the entire development process, including development, deployment, and testing. Upon selecting a workspace for the project, Ganache provides a list of 10 Ethereum accounts with a default balance of 100 ETH, which can just be used locally for testing and developing. Instead of setting up the configurations and deploying smart contracts to Ethereum testnets, Ganache helps the development process that can be started instantly without too many configurations. Ganache virtual blockchain behaves the same as the real network, however, it is cost-efficient by using a local account with virtual money while the deployment and transaction time is significantly reduced compared to developing on other testing networks of Ethereum, e.g., Rinkeby, Kovan, Ropsten, etc.

Truffle

Truffle declares itself as a world-class development environment, testing framework, and asset pipeline for blockchain networks using EVM that aims to make blockchain development easier [33]. In the development process of the proposed application, Truffle is used as a framework to manage the code base, and compile, and deploy smart contracts. With Truffle, developers can configure multiple network setups and deploy and migrate their smart contracts to any declared public or private blockchain network. Truffle also allows developers to effectively write tests in JavaScript with a given smart contract predefined in the environment. In addition, Truffle has a configurable build pipeline and customized scripts injected into the console environment, which is used for setting up and effectively managing the testing data during the development process. Figure 11 shows a setup script that is used to upload mock data for lands and citizens on every contract deployment via the 'initLands' and 'initCitizens' functions. With Truffle, smart contract developers are also allowed to organize the folder structures of the project. Figure 12 is the Truffle configuration file used in the application; the smart contract code base, deployment script, and output of the smart contract are defined to be stored in their associated folder. Also, the setup deployment network, Solidity compiler, and the setup script for the Truffle console are also declared in the project configuration.

Interplanetary file system

IPFS is a decentralized file system for storing and accessing files, websites, applications, and data. IPFS is built on fundamental principles of peer-to-peer networking and content-based addressing. In order words, the protocol uses content addressing to identify the data by the content itself rather than by where it is located. Every piece of information achieved using this protocol has a CID, which is its hashed result. IPFS works by connecting all computer nodes on the network to the same file structure, which is organized as a Merkle DAG. The structure is the combination of the Merkel tree (used in blockchain to ensure immutability) and directed acyclic graphs (used in Git version control which allows users to see different versions of the content on IPFS) [34]. In

```
module.exports = {
    contracts_directory: "./src/blockchain/contracts",
    migrations_directory: "./src/blockchain/migrations",
    contracts_build_directory: "./abis",
    networks: {
        development: {
            host: "127.0.0.1",
            port: 7545,
            network_id: "*",
        },
        private: {
            host: "127.0.0.1",
            port: 8545,
            network_id: "*",
            from: "0x8411f48086e9097520920F61754Be32f3Cb0849B",
        },
    },
    compilers: {
        solc: {
            version: "^0.8.0",
        },
    },
    console: {
        require: "./src/blockchain/setup-scripts/setupFuncs.js",
    },
};
```

Fig. 12: Truffle console script used in the proposed application

the implementation of the proposed application, IPFS was used to store the land specification. By using IPFS, the application can scale up when thousands of land records are uploaded to the system, and the authenticity of documents is always ensured since the cryptographic hash identifier is retrieved from the content itself.

Then, the website system using NextJS was built up for the client side. NextJS is a framework built on top of ReactJS for developing single-page applications. Figure 13 presents the application GUI homepage.

B. *Discussion*

Although being a good candidate as an administrative tool for managing land ownership because of its transparency and immutability, blockchain still has some limitations. In this section, two of the critical drawbacks of the current implementation of Ethereum blockchain, which are high transaction cost and high transaction waiting time, will be analyzed. Then some feasible solutions for the current implementation are finally suggested.

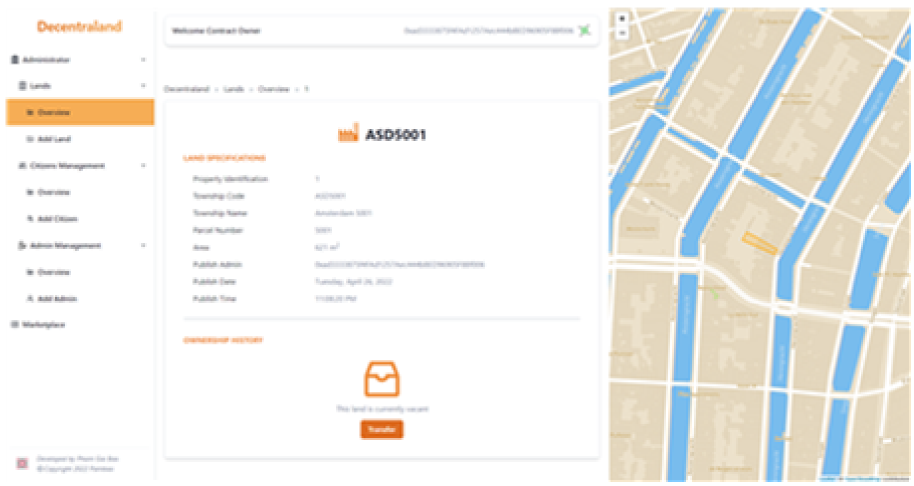

Fig. 13: User Interface of the proposed application

Ethereum limitations

- High transaction fee

With the current implementation of Ethereum using the PoW consensus mechanism, every transaction or interaction with the blockchain must pay for some gas as a reward for miners. Transaction in Ethereum has an algorithmically predefined gas requirement, which is a specified execution cost in the unit of gas. The total gas, which is paid by users, is calculated by the accumulation of all operations required in the transaction [35]. In other words, the more complex the operations of the transaction or the smart contract are, the higher the cost is required to pay for. For example, the simplest and typical transaction type is a transfer of Ethereum (ETH) between two Ethereum addresses. This type of transaction requires a fee of 21,000 gas * gas price to be paid, and it is even around 10 times higher for the case of smart contract creations and complex smart contract calls [36]. The gas price is a composite of (base fee + tip). The base fee is the minimum price for the inclusion of a transaction into a block, which is independently determined by the demand for block space (based on the size of the previous block). This base fee is automatically burnt when the new block is mined. The tip is the compensation for miners. The more tips users are willing to pay for, the more chance the transaction will be executed before others in the block [35].

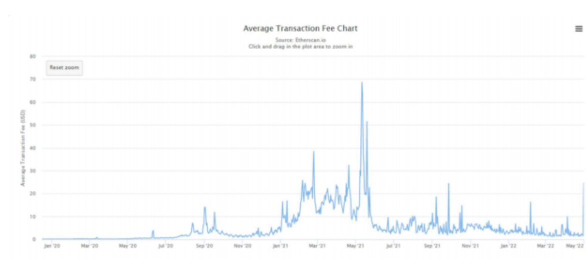

Fig. 14: Ethereum average transaction fee in USD from January 2021 to May 2022 [36]

Hence, with the mechanism explained above, the gas price depends on the supply/demand [35], which makes the total transaction fee extremely high at some point. As shown in Figure 14, during the period from the beginning of 2021 to June 2021, the average transaction fees mainly were between 15 USD to 50 USD per transaction and reached the highest point at nearly 70 USD per transaction in May 2021 [37]. Recently, at the time of this report (May 2022), the average transaction was approximately 5 USD per transaction. However, it was up to 25 USD per transaction in May 2022. The current transaction price is unpredictable and becomes too high because of the increasing demand, as stated above. This makes the operation fees on the full scale of the proposed application too costly and economically unstable to operate in the long run.

- High transaction waiting time

In addition to the high transaction fee that affects the scalability of applications implemented on Ethereum blockchain, high transaction time is also another issue of this network. The transaction waiting time is defined as the difference between the time the transaction is first added to the mempool, and the time the new block containing the transaction is mined. In addition, whenever the previous transaction of a particular user is waiting in the mempool, that user cannot submit a new transaction until the previous transaction is successfully included in the mined block or canceled. Therefore, the delay time of a specific transaction can entirely block the user from executing or performing any other actions in the network [38]. In Ethereum, the block time, which is the average time taken to mine a new block, is between 12 and 14 seconds. The expected block time in Ethereum is set as a constant at the protocol level to protect the network's security when miners suddenly add much more computational power. The average block time is automatically adjusted to be equal to that expected block time. In other words, if the block time is higher than the expected one, the difficulty of the mining algorithm will increase and vice versa [39]. Although the Ethereum network has the mechanism to reasonably adjust the block time, such an average of 14 seconds is still too high for an application to scale up when the demand significantly increases.

Suggested solutions

- Transition to Ethereum 2.0

Due to the limitation of the current Ethereum version, which is not environmentally sustainable (explained in the Proof-of-Work (PoW) section) and the negative effects on the scalability of application implemented on it, Ethereum developers have been actively working on the new upgrade (at the time of this writing in May 2022). The upgrade which is divided into several phases aims to improve the speed, efficiency, network scalability, and security through several changes to the underlying infrastructure and most noticeably, the transition from the current Proof-of-work (PoW) to Proof-of-Stake (PoS) consensus mechanism model (explained in Proof-of-Stake (PoS) section) [25]. As estimated in Ethereum documentation, the transition to PoS significantly reduces the energy consumption by 99% of the energy used in the current implementation of PoW [40]. In addition, the transition to the PoS consensus mechanism also allows the network to implement the sharding method, which is traditionally a process of horizontally partitioning a database to share to load. The term 'shards' is also a widespread concept used in computer science [41]. This will give Ethereum more capacity to store and access data and the capability to process many more transactions in a specific time compared to PoW. According to statistics provided on the Ethereum website, when the sharding is completely implemented in Ethereum, the transaction rate will be increased by at least 64 times the current rate. Specifically, from 15 transactions per second to the estimated interval between 25,000 and 100,000 transactions per second [40]. Although Ethereum was still in the process of upgrading to version 2.0, the transition significantly improved the limitations of the prior version. More importantly, applications that were previously implemented on the old infrastructure are automatically migrated to the new version without losing any transaction records and data. The new chain is promising and could potentially be a sustainable solution for the proposed application.

- Implementing on a different chain

Besides Ethereum 2.0, there are already blockchain networks improving the above limitation that implement other consensus algorithms and operate in the real solution. The most noticeable network could be Solana, which gained the attention of the community in recent years. As presented on their official website, the transaction rate is approximately 2,500 transactions per second. Also, their fee for executing one transaction is just 0.00025 USD, which is remarkably low compared to Ethereum 1.0 [42]. In addition, another blockchain that resolves the current issues of Ethereum by the use of PoS and sharding method, namely NEAR Protocol, could be another potential solution. There is a solution to migrate any application on Ethereum to NEAR blockchain called Aurora, which is a layer 2 scaling solution intended for developers to launch Ethereum decentralized applications on the NEAR network. This enables developers to run their implementation on an EVM compatible while the application can also benefit from an improved layer 1 blockchain with less transaction cost and a high throughput of only 2–3 seconds of transaction finalization [43]. These solutions of using different blockchain or layer 2 technology could be potentially promising options for improving the current issue of the implementation of Ethereum 1.0. However, such technologies are still in their early ages, which need more time to be proven as stable, trustworthy technologies with adequate research and studies conducted.

VI. CONCLUSION AND FUTURE WORKS

This work provides foundation knowledge and essential concepts of blockchain technology as well as an investigation of some of the widely used consensus algorithms. Additionally, this work presents a detailed smart contract implementation of application on Ethereum blockchain in the public sector of managing and archiving land ownership. Using blockchain technology, any related activities, e.g., archiving, transferring, and inheriting of land ownership records could be much more transparent and secured from fraudulent manipulation.

Although blockchain has not yet been applied to land management in Vietnam, there are currently no regulations or policies governing its employment in this sector. However, it has been demonstrated that blockchain can significantly enhance the effectiveness and efficiency of land management, as evidenced by its implementation in Dubai [44]. This advancement suggests a future trend where other countries may adopt blockchain technology for land management.

In future studies, various blockchain platforms (e.g., Solana, Polkadot, etc.), and expand the blockchain architecture to a hybrid mode will be explored. This study aims to continuously ensure transparency, security, privacy, and confidentiality in our system. More tests to achieve time cost per use case and performance evaluation will also be executed. Finally, the implementation details and code might be provided as requested.

REFERENCES

[1] Nguyen TN, Nguyen TN, Vuong DD. *Vietnam – Affordable housing : A way forward [Nhà ở giá hợp lý ở Việt Nam: Con đường phía trước]*. World Bank Group; 2015.

[2] Nguyen MN. *Urban population in Vietnam from 2018 to 2023*. https://www.statista.com/statistics/603397/vietnam-urban-population/ [Accessed 18th March 2024].

[3] Gillespie J. *Public land disputes in Vietnam: a multi-actor analysis of five case studies with an East Asian comparative*. https://thuvienso.quochoi.vn/handle/11742/23630 [Accessed 18th March 2024].

[4] The World Bank . *Corruption from the perspective of citizens, firms, and public officials: result of sociological survey*. Hanoi, Vietnam: National Political Publishing House; 2013.

[5] Dang HV, Nguyen VT, T&C Consulting. *Improving land sector governance in vietnam: implementation of land governance assessment framework (LGAF)*. Washington, D.C.: World Bank Group.; 2013.

[6] ACLED. *Bringing clarity to crisis datasets for 2020*. https://www.acleddata.com/data/ [Accessed 13th November 2023].

[7] Linh Tong. *Vietnam's experiment with e-government*. *E-government survey in media*. https://publicadministration.un.org/egovkb/en-us/Resources/E-Government-Survey-in-Media/ID/1913/Vietnams-Experiment-With-E-Government [Accessed 16th April 2024].

[8] Crosby M, Nachiappan, Pattanayak P, Verma S, Kalyanaraman V. Blockchain technology beyond bitcoin. *Applied Inovation Review*. 2016;2: 6–19.

[9] Ethereum. *Ethereum Whitepaper*. https://ethereum.org/en/whitepaper/ [Accessed 16th April 2024].

[10] Nakamoto S. *Bitcoin: A peer-to-peer electronic cash system*. https://www.ussc.gov/sites/default/files/pdf/training/annual-national-training-seminar/2018/Emerging_Tech_Bitcoin_Crypto.pdf [Accessed 16th April 2024].

[11] John E. *Bitcoin's price history*. https://www.investopedia.com/articles/forex/121815/bitcoins-price-history.asp#citation-35 [Accessed 17th April 2024].

[12] Rawat DB, Chaudhary V, Doku R. Blockchain technology: Emerging applications and use cases for secure and trustworthy smart systems. *Journal of Cybersecurity and Privacy*. 2021;1(1): 4–18. https://doi.org/10.3390/jcp1010002.

[13] Alam KM, Rahman JMA, Tasnim A, Akther A. A blockchain-based land title management system for Bangladesh. *Journal of King Saud University - Computer and Information Sciences*. 2022;34(6): 3096–3110. https://doi.org/10.1016/j.jksuci.2020.10.011.

[14] Graglia JM, Mellon C. Blockchain and property in 2018: at the end of the beginning. *Innovations: Technology, Governance, Globalization*. 2018;12: 90–116. https://doi.org/10.1162/inov_a_00270.

[15] Eder G. Digital transformation: blockchain and land titles. In: *Proceedings of the OECD global anti-corruption & integrity forum. 20th–21st March 2019; Paris, France*. OECD Global Anti-Corruption and Integrity Forum; 2019. p.12. https://www.scribd.com/document/550713727/Georg-Eder-Blockchain-Ghana-verified [Accessed 23rd November 2022].

[16] Shang Q, Price A. A blockchain-based land titling project in the Republic of Georgia: rebuilding public trust and lessons for future pilot projects. *Innovations: Technology, Governance, Globalization*. 2019;12(3–4): 72–78. https://doi.org/10.1162/inov_a_00276.

[17] Haleem A, Javaid M, Singh RP, Suman R, Rab Sh. Blockchain technology applications in healthcare: An overview. *International Journal of Intelligent Networks*. 2021;2: 130–139. . https://doi.org/10.1016/j.ijin.2021.09.005.

[18] Antonopoulos AM, Wood G. *Mastering Ethereum*. United States: O'Reilly Media, Inc.; 2018.

[19] Liang YC. *Blockchain for dynamic spectrum management. Dynamic spectrum management. Signals and Communication Technology*. Singapore: Springer; 2020. p.121–146. https://doi.org/10.1007/978-981-15-0776-2_5.

[20] Antonopoulos AM. *Mastering Bitcoin*. 2nd ed. United States: O'Reilly Media, Inc.; 2017.

[21] Huaqun G, Xingjie Y. A survey on blockchain technology and its security. *Blockchain: Research and Applications*. 2022;3(2): 100067. https://doi.org/10.1016/j.bcra.2022.100067.

[22] Corluka D, Lindh U. *Blockchain: a new technology that will transform the real estate market*. Master's thesis. Stockholm, Sweden: KTH University; 2017. https://kth.diva-portal.org/smash/record.jsf?pid=diva2%3A1124675&dswid=4709 [Accessed 24th April 2024].

[23] Lissaker F, Sjoberg J. *Architecture framework for blockchain implementation*. Master's thesis. Gothenburg, Sweden: Department of Computer Science and Engineering, Chalmers University of Technology and University of Gothenburg; 2019 https://odr.chalmers.se/server/api/core/bitstreams/c8972656-afec-440c-b857-b0458c99533b/content [Accessed 24th April 2024].

[24] Best Rd. *Average mining difficulty of Bitcoin from January 2014 to May 13, 2021*. https://www.statista.com/statistics/1224431/mining-difficulty-bitcoin/ [Accessed 24th April 2024].

[25] Ethereum. *Upgrading Ethereum to radical new heights*. https://ethereum.org/en/upgrades/ [Accessed 24th April 2024].

[26] Szabo N. *Smart contract: building blocks for digital markets*. https://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart_contracts_2.html [Accessed 25th April 2024].

[27] Ethereum. *Introduction to smart contracts*. https://docs.soliditylang.org/en/v0.8.13/introduction-to-smart-contracts.html [Accessed 25th April 2024].

[28] Protocol Labs. *IPFS powers the distributed web*. https://ipfs.io/#how [Accessed 29th April 2024].

[29] Polge J, Robert J, Le Traon Y. Permissioned blockchain frameworks in the industry: A comparison. *ICT Express*. 2021;7(2): 229–233. https://doi.org/10.1016/j.icte.2020.09.002.

[30] Best Rd. *Ethereum (ETH) price per day from August 2015 to April 28, 2022 (in U.S. dollars)*. https://www.statista.com/statistics/806453/price-of-ethereum/ [Accessed 13th November 2022].

[31] ConsenSys Software Inc. *Ganache*. https://trufflesuite.com/docs/ganache/ [Accessed 15th May 2024].

[32] ConsenSys Software Inc. *Truffle*. https://trufflesuite.com/docs/truffle/ [Accessed 15th May 2024].

[33] IPFS. *How IPFS works*. https://docs.ipfs.io/concepts/how-ipfs-works/#content-addressing [Accessed 15th May 2024].

[34] Ethereum. *Gas and fees*. https://ethereum.org/en/developers/docs/gas/ [Accessed 15th May 2024].

[35] Karaivanov A, Donmez A. *Transaction fee economics in the ethereum blockchain*. https://ideas.repec.org/p/sfu/sfudps/dp21-02.html [Accessed 15th May 2024].

[36] Etherscan. *Average transaction fee chart*. https://etherscan.io/chart/avg-txfee-usd [Accessed 15th May 2024].

[37] Liu Y, Lu Y, Nayak K, Zhang F, Zhang L, Zhao Y. Empirical analysis of EIP-1559: transaction fees,

waiting times, and consensus security. In: *2022 ACM SIGSAC Conference on Computer and Communications Security (CCS '22)*. 2022. New York, NY, USA: Association for Computing Machinery; 2022. p.2099–2113. https://doi.org/10.1145/3548606.3559341.

[38] Ethereum. *Block time*. https://ethereum.org/en/developers/docs/blocks/#block-time [Accessed 15th May 2024].

[39] Ethereum. *Ethereum energy consumption*. https://ethereum.org/en/energy-consumption/. [Accessed 15th May 2024].

[40] Ethereum. *Shard chains*. https://ethereum.org/en/upgrades/shard-chains/. [Accessed 15th May 2024].

[41] Solana. *Powerful for developers. Fast for everyone*. https://solana.com/news/solana-summer [Accessed 15th May 2024].

[42] Aurora. *What is Aurora?*. https://doc.aurora.dev/ [Accessed 15th May 2024].

[43] Merlec M, Lee Y, Hong SP, In H. A smart contract-based dynamic consent management system for personal data usage under GDPR. *Sensors*. 2021;21(23): 7994. https://doi.org/10.3390/s21237994.

[44] Vibhu U. *The use of blockchain technology is transforming land management*. https://www.linkedin.com/pulse/use-blockchain-technology-transforming-land-vibhu-upadhyay-zumuc/ [Accessed 15th May 2024].