A Project Report on

Implementing Comprehensive Blockchain based Framework for Transparent Real Estate Transactions

Submitted in partial fulfillment of the requirements
for the award of the degree of

Bachelor of Engineering

in

Information Technology

by

Aditya Shinde

18104034

Under the Guidance of

Prof. Neha Deshmukh

Department of Information Technology
NBA Accredited
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI

Academic Year 2021-2022

Approval Sheet

This Project Report entitled ***"Implementing Comprehensive Blockchain based Framework for Transparent Real Estate Transactions"*** Submitted by ***"Aditya Shinde"(18104034)***is approved for the partial fulfillment of the requirenment for the award of the degree of ***Bachelor of Engineering*** in ***Information Technology*** from ***University of Mumbai***.

(Prof. Neha Deshmukh)
Guide

Prof. Kiran Deshpande
Head Department of Information Technology

Place:A.P.Shah Institute of Technology, Thane
Date:

CERTIFICATE

This is to certify that the project entitled ***"Implementing Comprehensive Blockchain based Framework for Transparent Real Estate Transactions"*** submitted by ***"Aditya Shinde" (18104034)*** for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in ***Information Technology***,to the University of Mumbai,is a bonafide work carried out during academic year 2021-2022.

(Prof.Neha Deshmukh)
Guide

Prof. Kiran Deshpande
Head Department of Information Technology

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane
Date:

Acknowledgement

I have great pleasure in presenting the report on **Implementing Comprehensive Blockchain based Framework for Transparent Real Estate Transactions** I take this opportunity to express my sincere thanks towards my guide **Prof. Neha Deshmukh** Department of IT, APSIT thane for providing the technical guidelines and suggestions regarding line of work. I would like to express my gratitude towards his constant encouragement, support and guidance through the development of project.

I thank **Prof. Kiran B. Deshpande** Head of Department,IT, APSIT for his encouragement during progress meeting and providing guidelines to write this report.

I thank **Prof. Vishal S. Badgujar** BE project co-ordinator, Department of IT, APSIT for being encouraging throughout the course and for guidance.

I also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. I wish to express my deep gratitude towards all my colleagues of APSIT for their encouragement.

Student Name: Aditya Shinde

Student ID: 18104034

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature)

(Aditya Shinde - 18104034)

Date:

Abstract

The onset of blockchain generation within bitcoin has generated significant interest by showing an opportunity to eliminate the middle ground need and transform communication between people and machines by increasing trust. Initially restricted to the integrated currency domain, people began to see the power of a generation beyond just the cryptocurrencies, which brought the acceptance of the blockchain era to erase the world's problems. One such situation is problems for e-governance companies in other areas of the public sphere. In the scope of this thesis, we have specifically addressed the issues within the traditional property registration system. This thesis discusses the new design and architecture for real estate transactions and implements it using a blockchain-based solution and addressing issues including record integrity, privacy, and most importantly, the lack of common platforms among concerned government organizations. The advent of the blockchain era led to the creation of blockchain-enabled platforms like Hyperledger Fabric. It is one of the most popular open-source permissioned blockchain frameworks, created and supported by the Linux Foundation and IBM, used in many industrial scenarios. So, it is used to create a network with one ordering organization with an ordering node and one peer organization with two peer nodes to prove the concept. Chaincode similar to Ethereum's smart contract contains the logic to perform all operations and modify the ledger data. All the methods of chaincode are accessed using the Fabric gateway in the web application to perform various operations.

Contents

List of Figures

List of Tables

List of Abbreviations

RoR:	Record of Rights
ICT:	Information and Communication Technologies
IPFS:	InterPlanetary File System
DILRMP:	Digital India Land Record Modernisation Programme
MMP:	Mission Mode Project
NeGP:	National e-Governance Plan
DoSR:	Department of Stamps & Registry
BoR:	Board of Revenue
SRO:	Office of Sub Registrar
CFT:	Crash Fault Tolerant
MSP:	Membership Service Provider
CA:	Certification Authority
SDK:	Software Development Kit
TLS:	Transport Layer Security
MFA:	Multi-factor Authentication
IGR:	Inspector General of Registration
ULPIN:	Unique Land Parcel Identification Number
OPS:	Operations per Second

Chapter 1

Introduction

Technology in its relationship to globalization has always been at the forefront of disrupting standard issues and producing a brand-new clean look in certain objects. Blockchain is one of those technologies, it is a disruptive innovation with the potential to revolutionize the way governments and other non-profit, or for-profit organizations handle themselves, as well as how they communicate with collaborating parties [1]. However, it was not very widespread until a person or a group going by the name of Satoshi Nakamoto in 2009 created the Bitcoin cryptocurrency [2].

According to the general public-related industries, technological changes are always in line with integration guidelines and are provided with an amazing deal of problems and even growth. In this thesis, we delve deeper into the knowledge of property registration while conveying it to the Indian context and current related issues. The primary purpose of this concept is to show how integration with blockchain technology is the best solution. We can look at blockchain-era property registration technology that makes it an effective solution to current problems, and finally, we can present to you the results obtained from all the tests of the system used. One of the significant challenges facing various provinces in India is the issue of land ownership, which is due to the fact that there is no land records management system to date. The issue of land ownership is so important, because almost all financial institutions rely heavily on the ownership of land-based assets to obtain collateral. Due to uncertainty about patent applications, financial institutions are inefficient and hinder the country's development.

Land ownership in India is primarily based on deeds of sale (acquired at the time of local registration) and other deeds. It involves following steps:

1. Buyer - Verification of ownership
2. Buyer - Pay negotiated amount to seller
3. Buyer - Preparation of sell deed
4. Buyer - Payment of stamp duty and registration charges
5. Seller - Sign the sell deed
6. Registrar - Execution of sell deed

Both buyer and seller need to follow the workflow to complete a property registration. However, not all of these documents confirm the ownership claim but instead confirm the fact that the transfer of the property will be recorded, and these are the recorded activities known as "world records". The term "world records" is a common term and conclusion of many other documents that provide property maps, the record of rights (ROR), sales documents, and much more information and are organized into various categories. Due to the lack of good governance and communication in departments, we often find inconsistencies in records, outdated and unscheduled data. This discrepancy is the cause of the patent problem as many years of records have to go back to confirm the ownership claim. The need for clear ownership and determining land ownership is explained in this thesis. So, over the past three decades, various schemes have been used to revitalize world records to move from the ownership of imaginative sites to the clearing and clarification of land ownership.

1.1 Blockchain

Blockchain is widely distributed to make business processes safer and more transparent. Blockchain is actually a series of blocks connected by a cryptographic hash, as the name implies [3]. Each block contains data structure and a cryptographic hash of the previous block, activity data, and timestamp. Consensus algorithms are defined in intelligent blockchain protocols, which are programs that automatically perform predefined actions under certain conditions.

1.2 Peer-to-peer Network

Computers connected to each other without servers should form a peer-to-peer network, which increases security because there is no single point of failure. Transactions are accepted on the block only if all partners agree and all peers have access to all network transaction records. In this thesis, literacy access is given to real estate owners and restricted access to government officials responsible for keeping records of land records.

One of the general motives for the authorities to introduce blockchain-based system is the fact that it will increase the speed of registration without compromising the security and integrity of the data [4]. With the evolution of information and communication technologies (ICT), government authorities are challenged to provide more efficient and effective services to citizens, yet guaranteeing a high level of data security, transparency, auditability, and privacy [1]. As discussed by authors of [5], blockchain can be used in improving the security of smart vehicles; a similar approach can be used for IoT devices. When real estate transactions are managed by using Blockchain, we do not need to depend on third parties to verify them and trust is of much less concern when the registration is forced to be done in a transparent way and cryptographic protocols are used to check the integrity of the information supplied by the person [6].

Chapter 2

Literature Review

The authors of [6] proposed a solution using Ethereum blockchain, a permissionless blockchain, to implement the proposed solution, and the working logic is present in the smart contract. The end-users will use a web application, and they need to use the Metamask wallet to perform any operation and pay some ether cryptocurrency as gas. Because as of now, Ethereum works on a proof of work consensus mechanism, the blockchain miners charge the gas to ensure that the smart contract is executed and the transaction is recorded in the blockchain. Although the decentralization of the system is achieved, the use of cryptocurrency and crypto wallets makes it inefficient and unnecessarily difficult to use for naive users.

The authors of [7] proposed a solution using the Ethereum blockchain and smart contract written in solidity. The core principle used is that the physical record for every land is converted into a certain amount of tokens based on the ERC20 token standard, so the tokens owned for a unique id of land represent the ownership of the land. When the ownership is transferred, the tokens with the unique id are transferred to a new owner. The token-based ownership makes it easy to keep records of shared ownership. Still, The major disadvantage in this solution is similar to the previously mentioned paper, i.e. wastage of the Ethereum cryptocurrency for every transaction. It will make use of this solution expensive for the general public.

The authors of [8] proposed a solution using Hyperledger Fabric (for simplicity Fabric) and IPFS (InterPlanetary File System). Fabric is a permissioned blockchain, and it doesn't require any cryptocurrency or mining to perform any operations. The working logic is present in Chaincode, which is similar to the smart contract of Ethereum. The IPFS is used to store documents in a decentralized way. Although the authors have proposed a more cost-efficient solution than previously discussed solutions, they haven't truly digitized the process. Since, to complete the registration process, there is a need to upload a scanned paper-based document.

The authors of [9] have discussed a potential solution with Multichain technology. The core principle is ensuring the integrity and availability of the documents used in the land registration process. The generated hash of the document is stored when the document is uploaded, and it is used whenever the user needs to verify the integrity and authenticity. The problem with the solution is the involvement of documents because it makes representing and accessing data of land or transfer rugged. Also, It risks privacy by exposing unwanted details to the public.

Digital India Land Record Modernisation Programme (DIL-RMP) [10, 11], an initiative by the Government of India, it mainly focuses on digitizing land records. A separate online web portal is available for citizens to access electronic certificates of land records and other related documents. This cuts down the time for accessing records as it reduces the interaction between the citizen and the government officials. This curtails a significant number of fraudulent transactions and acts as evidence in case of disputes. Likewise, Land Records Mission Mode Project (MMP) [12] is a program under National e-Governance Plan (NeGP) which aims to accelerate easy maintenance and updating of land records by keeping track of mutation cases. However, this lacks the digitization of the land registration process since the preparation and execution of sale deeds happen in a paper-based format. So, the land record needs to be updated manually, making the whole process time consuming and error-prone. Since the data storage is centralized, it poses threats to the privacy and integrity of the data.

After reviewing the already proposed solutions, it was observed that currently, no solution could truly decentralize real estate transactions and, at the same time, digitize the complete process. The solution should be Secure, fast, and it should be easy to use for the general public. Although using Ethereum, Fabric and Multichain technologies comes with its own pros and cons. The major setback for Ethereum is the high cost for each transaction and low frequency of transactions. The Multichain technology provides a great blockchain platform, but the cost of implementing the solution in production is very high. On the other hand, Fabric is open source and provides full flexibility over blockchain architecture and scalability. With Fabric, there is no need to use cryptocurrency and no cost for executing the transactions.

Chapter 3

Existing system

In this section, we will describe the current state of local registration and land transfer in India and in particular the state of Uttar Pradesh. We have found that almost all the provinces of India have adopted the same workflow, so we can assume that the results we have given for the state of Uttar Pradesh can be extended to other provinces. In almost every province of India, we have found that usually two or three parts carry land titles [13] and so on.

1. Tickets and Registration Category – DoSR (Department of Stamps & Registry): Manages document verification during registration. There is also an additional obligation to collect stamp duty for the deed of sale to be registered.
2. Finance Department or Finance Board – BoR (Board of Revenue): Handles the preparation and maintenance of transactional records or transaction details.
3. Department of Surveillance and Residence: Responsible for the preparation and maintenance of maps.

In this article, we ignored the Survey Division and Settlement Division as we found that the irregularities were large and there was no record of geographical mapping. For the purposes of this thesis, we exclude title deeds relating to land/property registration. Listed below are the different types of opportunities where you can find a stable place in India:

1. Sell
2. Will/Gift/Swap
3. Inheritance
4. Procurement
5. Provided by the government or a court

All of the above transactions go under different types of actions and thus differ in the course of the transaction. In this sense, the focus is on the registration of acquired assets. In the state of Uttar Pradesh, as shown in Fig. 3.1 the three-dimensional framework is in place in the DoSR [13].

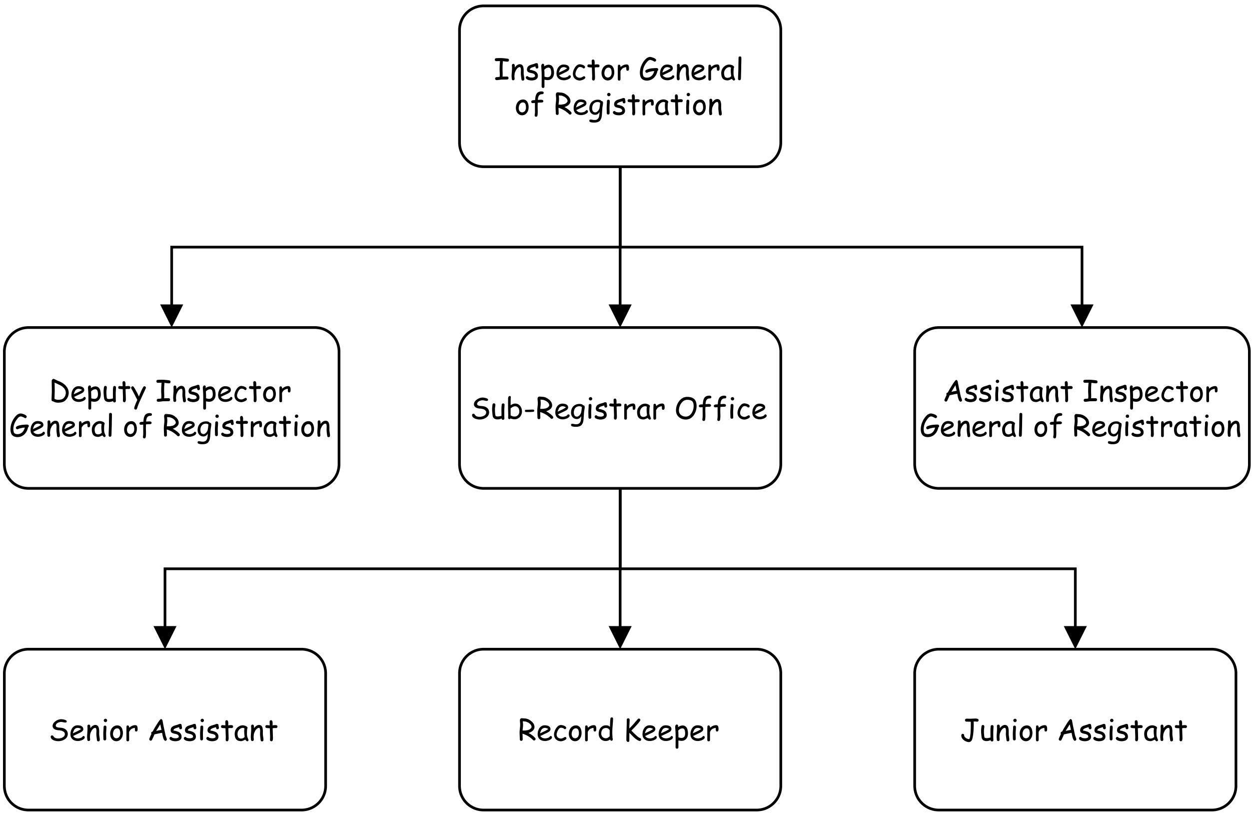


Figure 3.1: Hierarchical structure of DoSR

This divine structure was later used to describe the role-based authority Information contained in the blockchain, thus limiting access to sensitive information. The office of Sub Registrar (SRO) is the place where all registration activities take place. Various office functions are provided below [13]:

1. Estimation of property value
2. Collection of stamp duties
3. Registration of title deed
4. Maintenance of copies of title deeds
5. Issuing registration certificates

Listed below are the steps during the registration of immovable property in Uttar Pradesh:

1. Asset Verification: This feature of registration usually lasts until the consumer's consumer title is carefully considered. There are many variations of land rights in India and switching to online sites has not yet resolved the issues, so it is up to the buyer to verify ownership of the previous property.
2. Property Valuation: The value of a property is measured by the payment of the registration stamp tax, which is calculated as a percentage of the circular average for that area.

3. Preparation of Stamp Paper: Non-judicial stamp paper equal to the value of stamp tax is required for purchase, now available online.
4. Stamp Work Payment: This section can now be done online
5. Registration to the SRO: This part of the registration process requires personal intervention from the SRO, as well as buyer and seller witnesses for verification.

As mentioned earlier, in the state of Uttar Pradesh the BoR is responsible for the preparation and maintenance of transaction details. Departmental activities vary widely in the area where property is immovable i.e., whether the property belongs to the urban or rural category.

Chapter 4

Proposed Solution

We now propose a blockchain-based framework to execute real estate transactions. It involves the Fabric network architecture as well as a chaincode. It will enable users to perform all the operations digitally and temper proof. The entities involved and working of the framework is based on the existing system in place in India to perform real estate transactions. The users of this system will be the general public and government officials.

4.1 Considerations

There are a few primary processes that are needed to be considered in order to make the framework ready for real-world scenarios. In order to allow users to use this framework, all the users need to register themselves on the system, which can be done using signup functionality and later verifying the identity. Also, every Sub-Registrar needs to register all the estates under his jurisdiction on the system. If there is a need to put old record on the system, then local government officials who keep records of old real estate transactions will need to insert all the old transaction details in the system. After this, the framework will be ready to use. So, to explain the system's working, we have considered that all the procedures mentioned earlier are completed beforehand.

4.2 Architecture

Orderer

The ordering service will be implemented and administered by the central government. The orderer will run the Raft consensus mechanism with multiple nodes, which can be achieved even if the nodes are present at different physical locations. Raft is a crash fault tolerant (CFT) consensus mechanism used for the ordering service, and it is based on an implementation of Raft protocol in etcd key-value store [14].

Member/Peer Organizations

Every state of the country will be a separate Member Organization in the network, and it will act as a membership service provider (MSP) for that organization's entities.

Peer

Every member organization will take the responsibility of setting up peers for the use of their organization and joining the respective channel on every peer. The peers will involve anchoring, endorsing, and committing peers.

Ledger

Every peer has its own copy of the ledger that the peer maintains. It contains world state (data in the form of key-value pairs) and blockchain (log of transactions responsible for current world state). There are currently two options to use for ledger databases, LevelDB and CouchDB, but by default, LevelDB is used.

Channel

Every state government will be present in a separate channel, and one common participant in every channel will be the central government.

Certification Authority (CA)

A root CA will distribute digital identities to peers, orderer nodes, client applications, and administrators. The central government will administer the root CA to secure the blockchain network from unwanted access.

Chaincode

Every state government follows different rules to perform real estate transactions, so it will not be feasible for every state to implement the same chaincode. Fabric offers a solution to this problem by allowing every organization to develop and write the chaincode according to their needs. Then they can install the chaincode on endorsing peers, so they can use it to connect them to applications using Fabric gateway.

Application

To interact with smart contract and blockchain network, organizations will use the application software development kit (SDK) to develop the applications using the technology of their choice. The general public and government officials will use the application to execute real estate transactions.

In Fig. 4.1, the "Org 0" is the central government, i.e. the orderer organization. "Org 1" and "Org 2" are two state governments having their separate channels, "Channel 1" and "Channel 2" respectively. Both "Org 1" and "Org 2" have their peers in their respective channels. Applications A1, A2, and peers "Peer 1" and "Peer 2" belong to "Org 1" and "Org 2", respectively. According to their need, all the organizations can have more than one peer and application. Every peer has its copy of the ledger, and every endorsing peer has its instance of the chaincode.

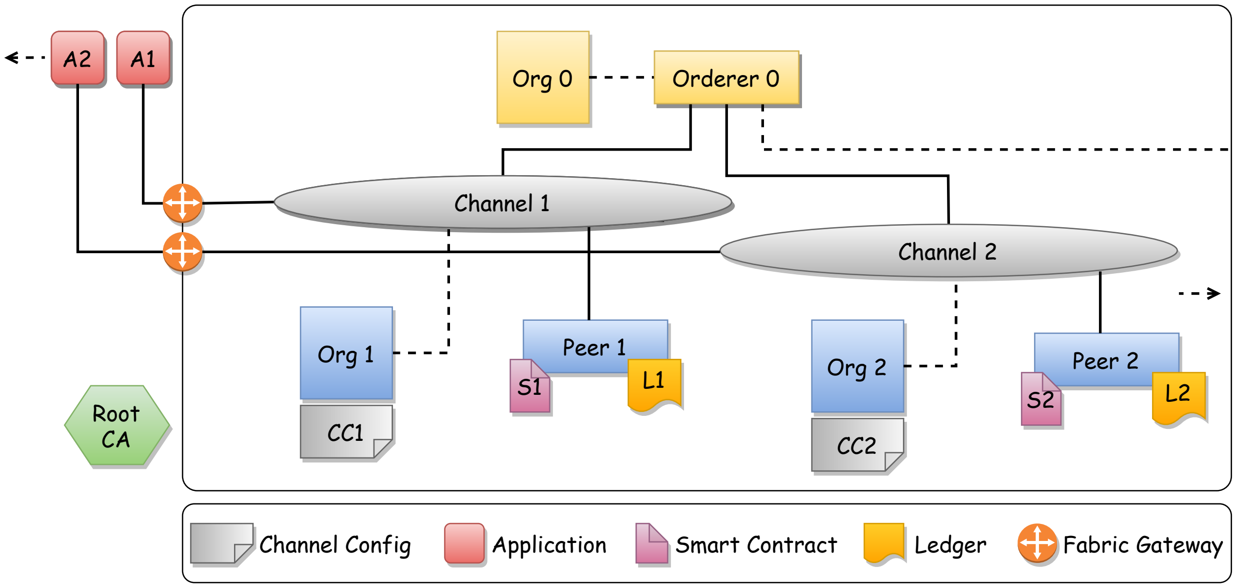


Figure 4.1: Proposed Fabric architecture.

4.3 Security

The security of the proposed framework is handled in two parts, i.e. security of blockchain and security of web server.

Security of web server

The web server uses TLS (Transport Layer Security) to secure the communication between the web server and users by encrypting the information. There is no data saved on the web server except users' sessions to prevent data from being accessed by intruders in case the security of the web server is compromised. The users must complete the multi-factor authentication (MFA) process and identity verification while signing up or logging in. It ensures that only legitimate users use the web application. Similarly, users need to go through MFA while performing the estate's final transfer.

Security of blockchain

The integrity of data in the blockchain is not a primary concern. As more nodes are added to the network, the probability of any unwanted entity modifying the data drops exponentially. But, to secure the data in transit between nodes, TLS is used. As discussed by the authors of [15], the security of Fabric can be improved further without affecting performance.

4.4 Chaincode Development

The significance of discussing chaincode development is that, since the framework is considered a solution for the whole country, if any state has special requirements or needs any different workflow, then the only thing they need to modify is the chaincode. Chaincode development involves developing the framework's logic for executing real estate transactions.

The first important part is choosing the programming language to develop the chaincode. There are currently three programming languages to choose from: Go, Java, and Node.js. The optimal choice is Go because it is the native language to Fabric and according to many analyses published on all three languages, Go is the best choice. The results show that Go is the most performing one for almost all the tests realized and that in the case of the update of the ledger, the latency follows a linear trend as the number of nodes in the network increase. In contrast, in the case of query, the latency is approximately constant [16].

The structures are used as the template for storing data related to IGR (Inspector General of Registration), Sub-Registrar, users, and estate transaction records. Following is the example of structure used for user data:

```
type User struct {
    Password_Hash string   `json:"passwordSalt"`
    Password_Salt string   `json:"passwordHash"`
    UID           string   `json:"uid"`
    Name          string   `json:"name"`
    Status        int      `json:"status"`
    Owned_Estates []string `json:"ownedEstates"`
}
```

The data in the ledger is stored as key-value pairs, and the keys are used to add, update and read data. The pattern used for keys for users is "user_" + uid, where uid is a unique identification number, i.e. Aadhaar number of a user. Similarly, structures and key patterns are used for storing and retrieving the data, such that data can be mapped uniquely. The chaincode has many functions to perform various operations like making land available to sell, requesting to buy a new estate, accepting buyers' requests, and other functions.

```
func (s *SmartContract) RequestToBuy_Estate
    (ctx contractapi.TransactionContextInterface,
    buyer_uid string, name string, ulpin string,
    proposedPrice int, dateTime string)
    error {
    // logic
}
```

The "InitLedger" function initializes the ledger with the credentials of IGR, then IGR will create the office wise sub-registrar accounts to administer the real estate transactions.

Chapter 5

Experimentation and Results

The implementation is only for a single state of the country, so it involves the central government and one state government. The setup will be similar for every other state. The thing that may or may not differ is the chaincode, because a state government may have some additional requirements so that they may make the necessary changes. The implementation involves deploying all the nodes involved in the Fabric network in docker-based containers. The installation procedure for docker is mentioned in Appendix-I. The docker containers are launched using docker-compose. The docker images and required binaries are installed by following the procedure mentioned in Appendix-II. Instead of using a root CA, cryptogen tool is used to simplify the process with configuration specified in Appendix-III. The cryptogen is a utility for generating Fabric key material, and it takes the "crypto-config.yaml" file that contains the information about the entities for which the digital certificates or identity is needed to access the blockchain network. It is not recommended to use cryptogen in the production network. The genesis block is generated using configtxgen with the configuration specified in Appendix-III. The order uses the genesis block to create and join a channel.

The Fabric network is started using docker-compose, and it uses the configuration specified in the "docker-compose.yaml" file. It launches a single orderer node in Raft consensus, two peers, and a cli container. The peers belong to the member organization. The cli container is used to interact with orderer and peers. The Fig. 5.1 illustrates the blockchain network design used for implementation.

"Peer 1" is anchor, endorsing, and committing peer, "Peer 2" is only a committing peer. The chaincode is installed on "Peer 1". The Fabric gateway connects the web application "A1" to the blockchain network using connection shown in Appendix-III; the gateway enables the web application to submit the chaincode transaction requests to peers, wait for the response, and many other operations. The web application exposes the RESTful API which is used in the frontend of the application. All the users of the real estate transaction framework like IGR, Sub-Registrars, and the general public will use the web application to perform the various operations implemented in the smart contract. The users can only see the simple user interface of the web application; they are not bothered by the complexity of the Fabric technology.

The workflow to follow while executing a real estate transaction is as follows:

1. Buyer - Verification of ownership

2. Buyer - Send request with proposed price to seller, using ULPIN (Unique Land Parcel identification Number) of estate

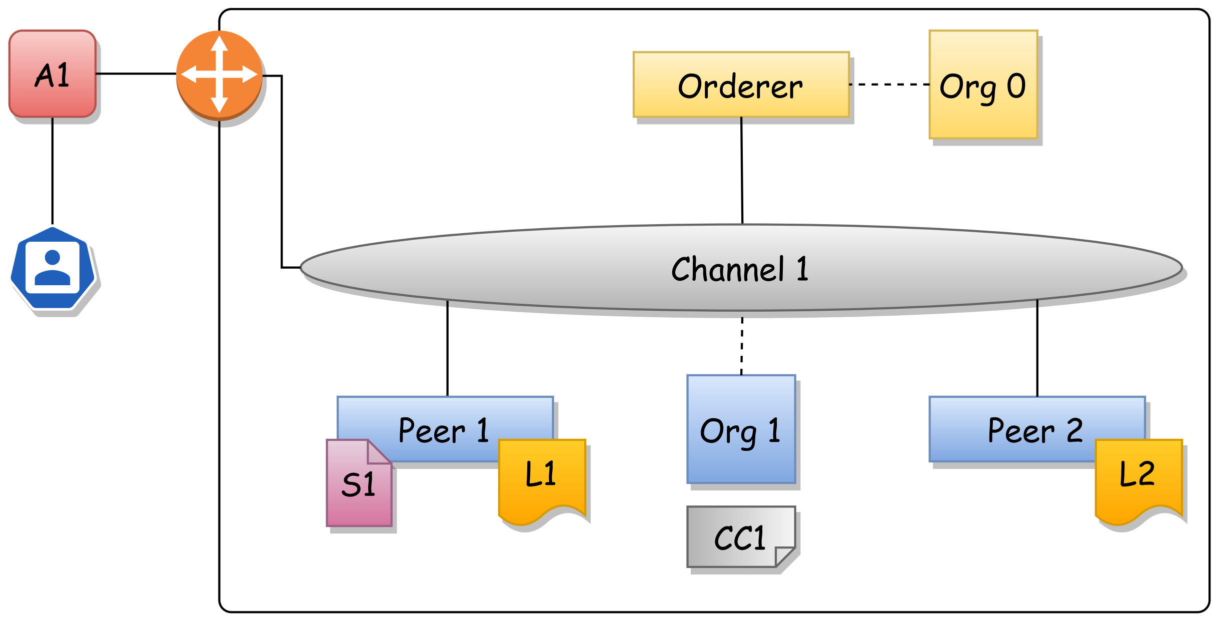


Figure 5.1: Implemented Fabric architecture.

3. Seller - Accept the request
4. Buyer - Pay stamp duty and registration charges
5. Registrar - Verify details and execute the transaction

Workflow involves a seller, a buyer, and a sub-registrar. If the status of the property is set to "available for sell", then any buyer can request to buy the property with the price he wants to propose. If the seller accepts that request, the buyer must pay the registration charges to move forward in the process. After this, the request will go to the sub-registrar under whose jurisdiction the property is present, and then he can verify and execute the final step. Immediately after that, the ownership of the property will change, and the transaction will be recorded on the blockchain for future reference. In every step of the workflow, the consensus is achieved for achieving transparency and integrity. A simple example of the consensus process is shown in Fig. 5.2, and it consists of all sub-processes involving entities of the blockchain network.

The IGR, Registrar, buyer and seller can perform the operations by logging in into the web portal. The details of any user and real estate with it's history can be viewed on the home page without logging in, as illustrated in Fig. 5.3. The Registrar's and users' dashboards are illustrated in Fig. 5.3 and Fig. 5.4 respectively.

Every operation performed by a user on the web application is completed by submitting a query or a invoke request for a chaincode function, using fabric gateway on the web server. The query request is useful for only reading the data from the ledger, so it is not recorded on the blockchain, which is why it is much faster than invoke. On the other hand, invoke request is useful for reading as well as writing and updating the ledger data, so it is recorded on the blockchain. It makes it slower because an operation will be complete when the consensus process is over. Both query and invoke requests serve different purposes, so comparing them isn't very meaningful, but using them efficiently helps make the system efficient. The framework is tested on Ubuntu (20.04.3) with 8 GB RAM using a Docker-based implementation

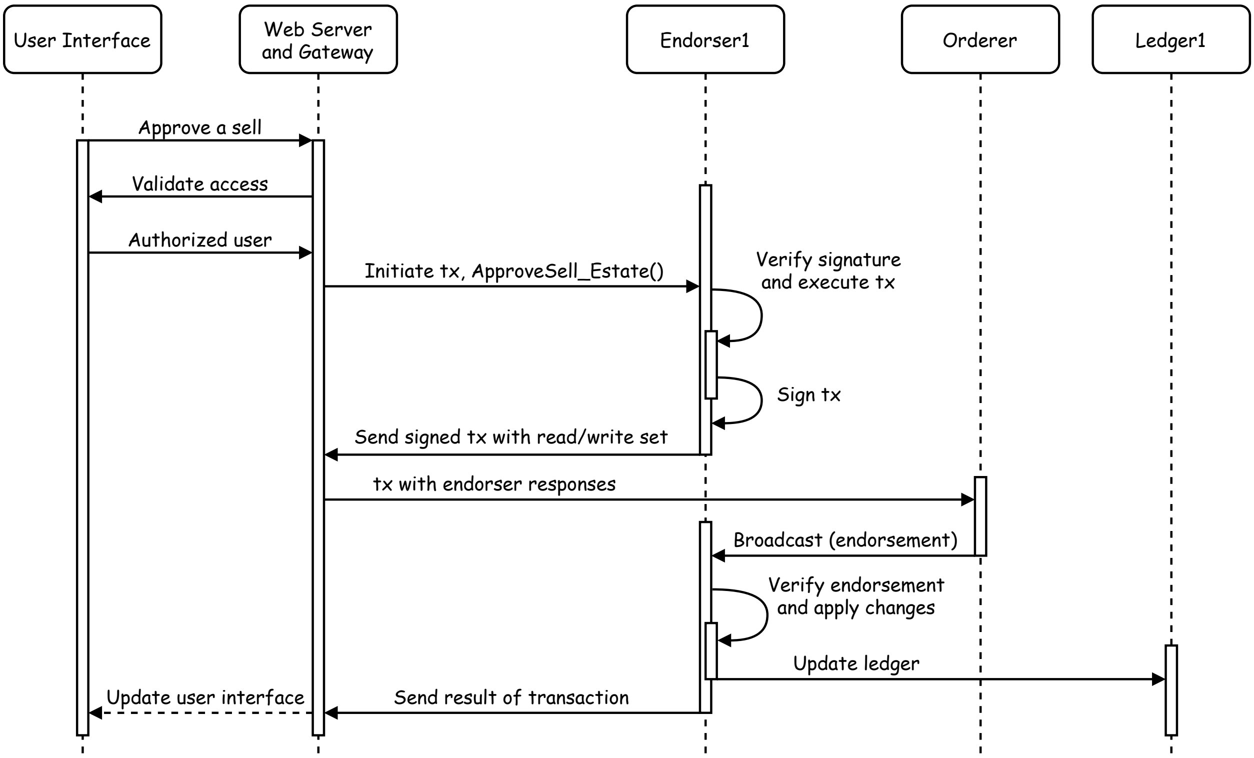
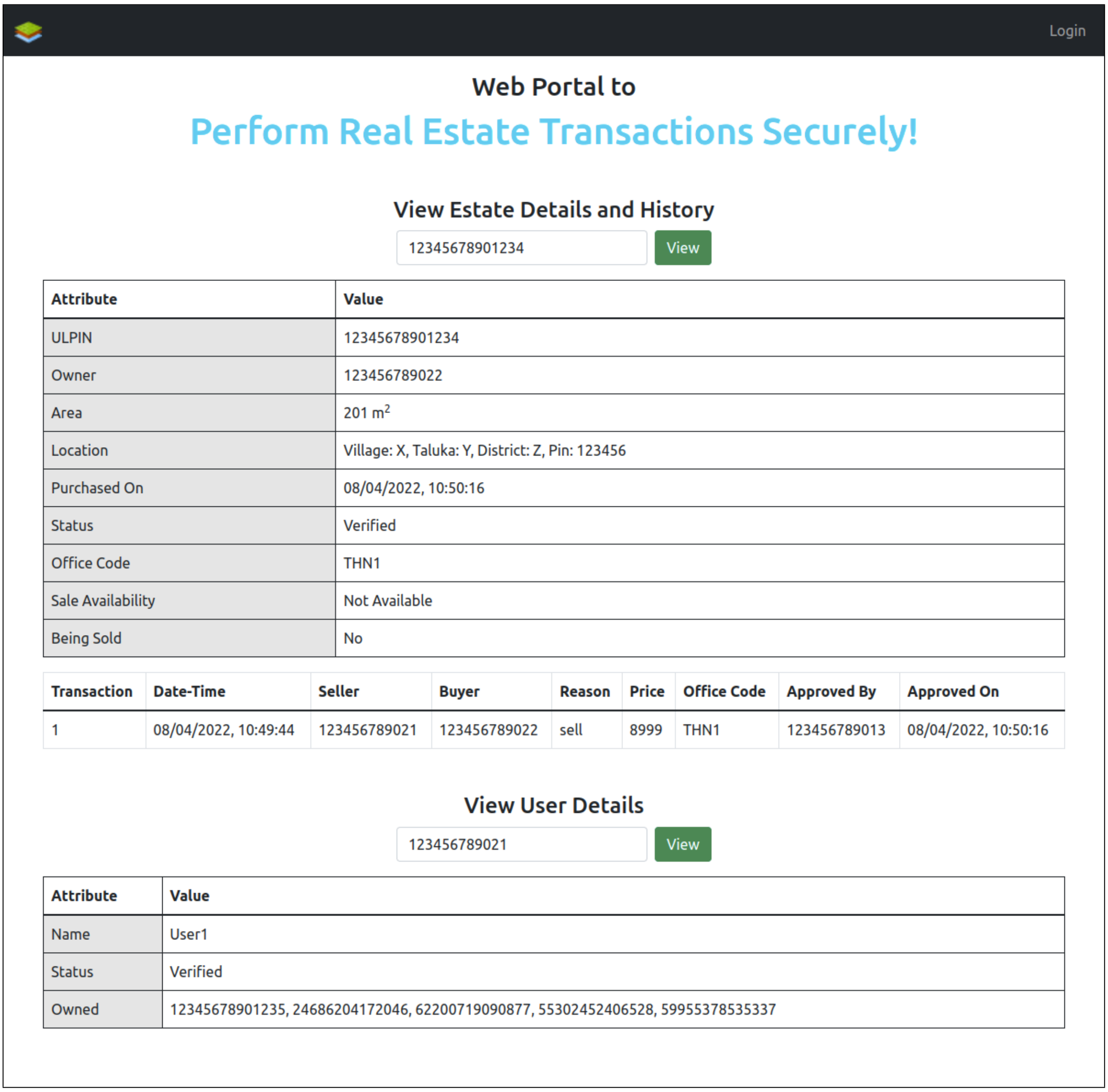


Figure 5.2: Sequence of events to achieve consensus, when the estate is transferred.

of Fabric architecture (Fig. 5.1). As illustrated in Fig. 5.6, the results obtained show how increasing the number of endorsing nodes (assuming that orderer nodes will be increased accordingly) can improve the throughput of the system, i.e. users' operations per second (OPS). It shows that scaling the nodes can largely improve the performance.


Login

Web Portal to

Perform Real Estate Transactions Securely!

View Estate Details and History

12345678901234 View

Attribute	Value
ULPIN	12345678901234
Owner	123456789022
Area	201 m^2
Location	Village: X, Taluka: Y, District: Z, Pin: 123456
Purchased On	08/04/2022, 10:50:16
Status	Verified
Office Code	THN1
Sale Availability	Not Available
Being Sold	No

Transaction	Date-Time	Seller	Buyer	Reason	Price	Office Code	Approved By	Approved On
1	08/04/2022, 10:49:44	123456789021	123456789022	sell	8999	THN1	123456789013	08/04/2022, 10:50:16

View User Details

123456789021 View

Attribute	Value
Name	User1
Status	Verified
Owned	12345678901235, 24686204172046, 62200719090877, 55302452406528, 59955378535337

Figure 5.3: Home Page

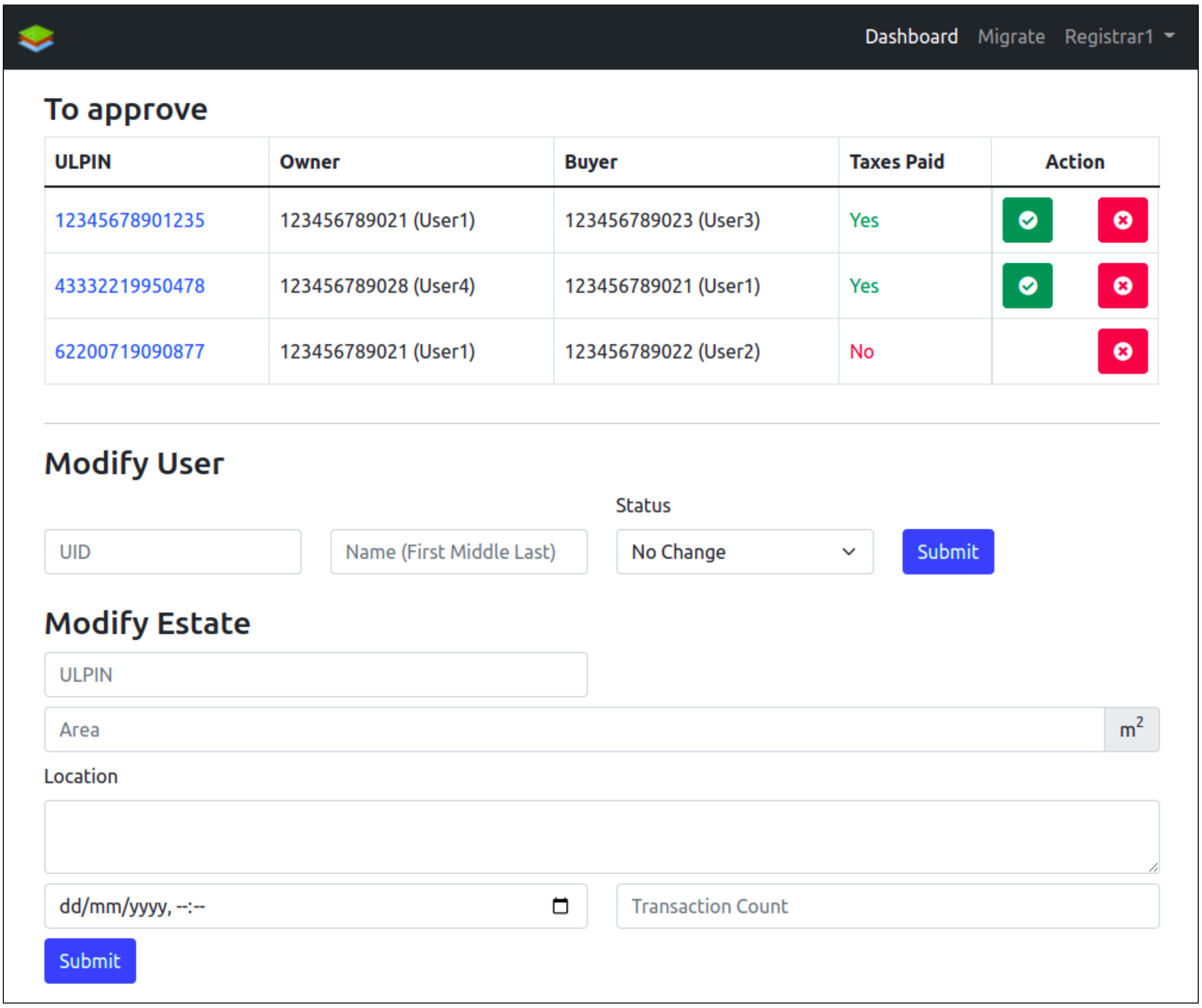


Figure 5.4: Registrar Dashboard

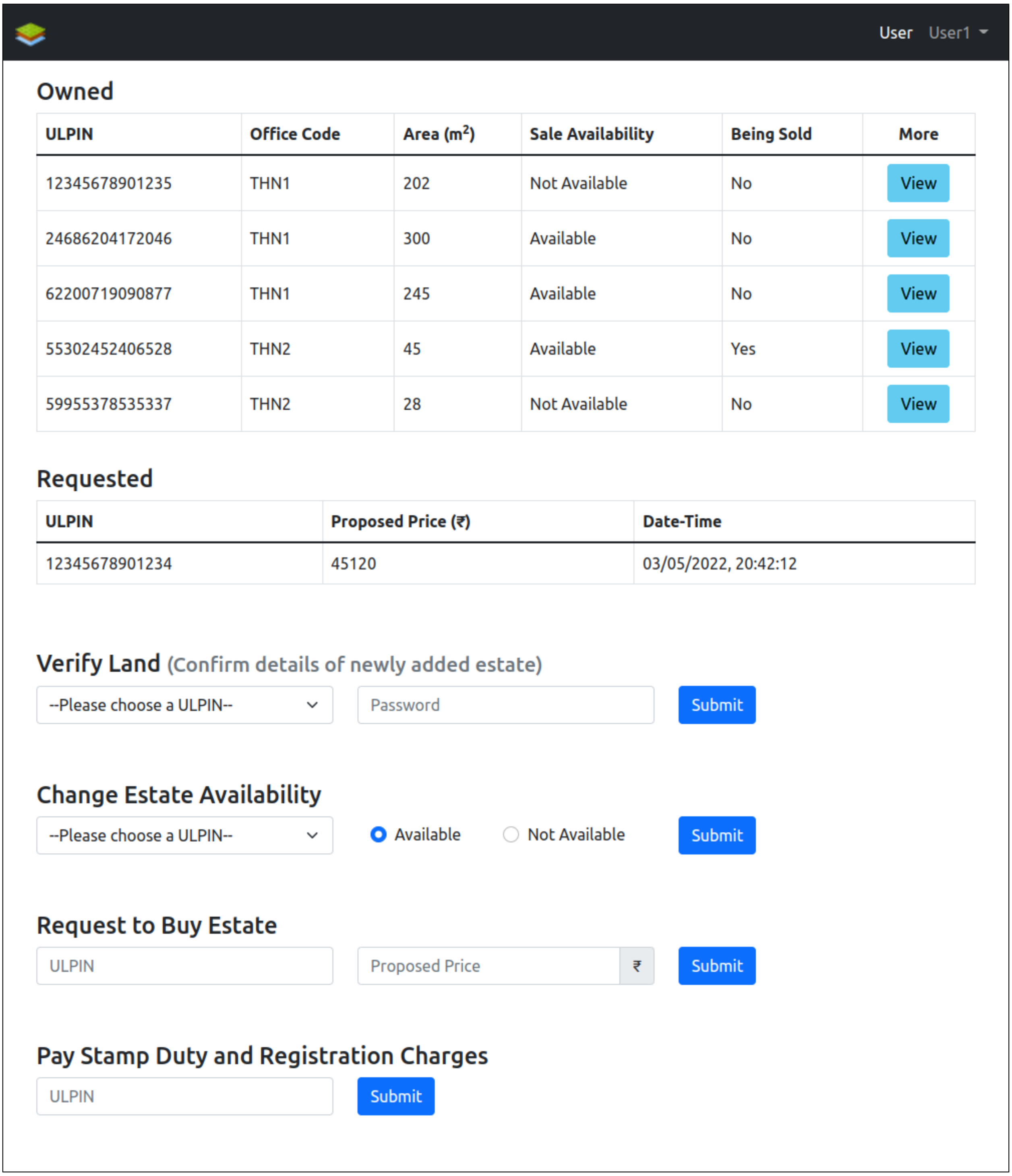


Figure 5.5: User Dashboard

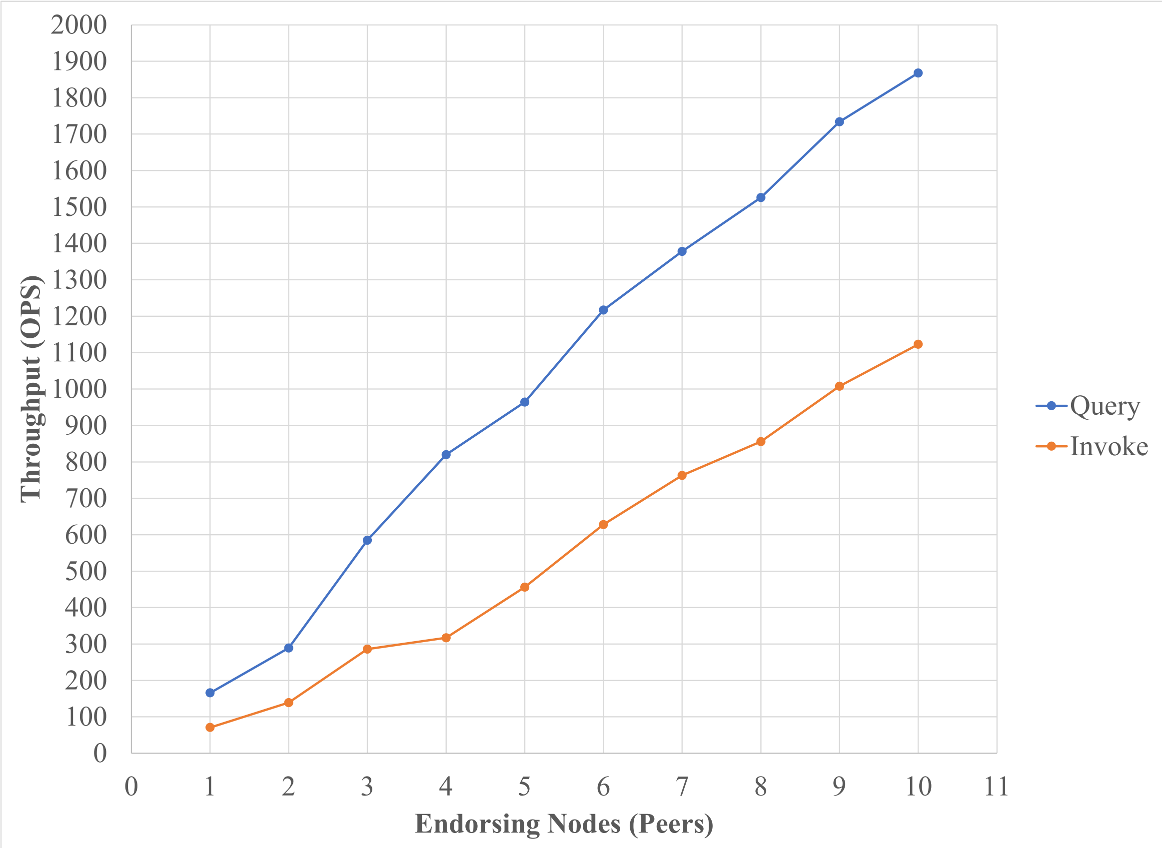


Figure 5.6: Throughput for operations involving query and invoke requests

After further analysis of the results, as detailed in the Table 5.1 demonstrated that scaling the nodes increases the latency of operations since the majority endorsement policy is used in the consensus mechanism. Because more and more endorsing nodes need to verify the transaction before the ledger is updated. So, to achieve the best performance for the framework, the balance between horizontal scaling and vertical scaling is needed, i.e. number of nodes and capacity of a single node.

Table 5.1: Average latency for operations involving invoke requests

Endorsing Nodes	Orderer Nodes	Invoke (OPS)	Avg. Latency(s)	Max Latency(s)
1	1	71	0.34	0.96
2	1	139	0.47	1.02
3	2	286	0.60	1.25
4	2	317	0.73	1.37
5	2	456	0.86	1.51
6	3	628	0.99	1.66
7	3	763	1.12	1.80
8	3	856	1.25	1.95
9	3	1008	1.38	2.09
10	3	1123	1.51	2.24

Chapter 6

Conclusions and Future Scope

This paper presents a modern solution to prevent and overcome the problems the current World Records Management System in India is facing. The installation of blockchain in the system makes it secure, decentralized and, free from fraud. The program provides access to all stakeholders involved in world record keeping, and builds a strong World Record Management platform. Distributing it via blockchain, makes the record transparent, validated and secure. These records can be used as tangible evidence in any land-based legal proceedings or disputes.

Therefore, we have proposed an easy-to-use and seamless platform that can be used to facilitate property registration. In current system there are many problems such as vendor engagement or intermediate involvement and time delays. This framework will eliminate the problems associated with property registration in India and in many parts of the world. Digitally registering land titles will not only simplify the process but also protect land titles from various man-made and natural disasters. Blockchain technology is emerging very quickly due to the secure features it provides. So, using a blockchain to save world record transactions is a way to create static records. Nowadays, land is not a commodity, the scope is therefore broad and there are many use cases for the created platform. However, for any successful implementation of the technology, seamless acceptance by stakeholders is required. Therefore, the main challenge for this solution will be to move existing world records management systems to the proposed one.

Many additional features can be added to this framework, like more details of the property and its geography. The chaincode can be improved and tested to make it more efficient. There is a need to handle cases where the property has shared ownership. To make the framework a complete solution a native cryptocurrency can be added to do all types of payments.

Bibliography

[1] Lazuashvili, N., Norta, A., Draheim, D.: Integration of blockchain technology into a land registration system for immutable traceability: A casestudy of georgia. In: Business Process Management: Blockchain and Central and Eastern Europe Forum. pp. 219–233. Springer International Publishing, Cham (2019)

[2] Nakamoto, S.: Bitcoin: A peer-to-peer electronic cash system (2008), https://bitcoin.org/bitcoin.pdf

[3] Swan, M.: Blockchain: Blueprint for a New Economy. O'Reilly Media, Inc. (2015)

[4] Verheye, B.: Land Registration in the Twenty-First Century: Blockchain Land Registers from a Civil Law Perspective, pp. 123–135. Springer International Publishing, Cham (2020)

[5] Smys, S., Wang, H.: Security enhancement in smart vehicle using blockchain-based architectural framework. In: Journal of Artificial Intelligence and Capsule Networks. vol. 3, pp. 90–100 (2021). https://doi.org/10.36548/jaicn.2021.2.002

[6] Suganthe, R.C., Shanthi, N., Latha, R.S., Gowtham, K., Deepakkumar, S., Elango, R.: Blockchain enabled digitization of land registration. In: 2021 International Conference on Computer Communication and Informatics (ICCCI). pp. 1–5 (2021)

[7] Nandi, M., Bhattacharjee, R.K., Jha, A., Barbhuiya, F.A.: A secured land registration framework on blockchain. In: 2020 Third ISEA Conference on Security and Privacy (ISEA-ISAP). pp. 130–138 (2020). https://doi.org/10.1109/ISEA-ISAP49340.2020.235011

[8] Mukne, H., Pai, P., Raut, S., Ambawade, D.: Land record management using hyperledger fabric and ipfs. In: 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT). pp. 1–8 (2019). https://doi.org/10.1109/ICCCNT45670.2019.8944471

[9] Ramya, U.M., Sindhuja, P., Atsaya, R., Bavya Dharani, B., Manikanta Varshith Golla, S.: Reducing forgery in land registry system using blockchain technology. In: Luhach, A.K., Singh, D., Hsiung, P.A., Hawari, K.B.G., Lingras, P., Singh, P.K. (eds.) Advanced Informatics for Computing Research. pp. 725–734. Springer Singapore, Singapore (2019)

[10] Dilrmp.in: Digital india land record modernization programme, https://dolr.gov.in/programme-schemes/dilrmp/digital-india-land-record-modernization-programme

[11] Vikaspedia.in: Digital india land record modernisation programme, https://vikaspedia.in/social-welfare/rural-poverty-alleviation-1/schemes-department-of-land-resources/computerisation-of-land-records

[12] Meity.gov.in: Land records - ministry of electronics and information technology, https://meity.gov.in/content/land-records

[13] Gunda, A., Shukla, S.: Property Registration and Land Record Management via Blockchain. Master's thesis, Indian Institute Of Technology Kanpur (2019)

[14] Hyperledger-fabric.readthedocs.io: Hyperledger fabric documentation, https://hyperledger-fabric.readthedocs.io

[15] Brotsis, S., Kolokotronis, N., Limniotis, K., Bendiab, G., Shiaeles, S.: On the security and privacy of hyperledger fabric: Challenges and open issues. In: 2020 IEEE World Congress on Services (SERVICES). pp. 197–204 (2020). https://doi.org/10.1109/SERVICES48979.2020.00049

[16] Foschini, L., Gavagna, A., Martuscelli, G., Montanari, R.: Hyperledger fabric blockchain: Chaincode performance analysis. In: ICC 2020 - 2020 IEEE International Conference on Communications (ICC). pp. 1–6 (2020)

Appendices

Appendix-I: Install Docker

Set up the repository

1. Update the `apt` package index and install packages to allow `apt` to use a repository over HTTPS:

```
$ sudo apt-get update

$ sudo apt-get install \
  ca-certificates \
  curl \
  gnupg \
  lsb-release
```

2. Add Docker's official GPG key:

```
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg \
  | sudo gpg --dearmor \
  -o /usr/share/keyrings/docker-archive-keyring.gpg
```

3. Use the following command to set up the stable repository.

```
$ echo \
  "deb [arch=$(dpkg --print-architecture) \
 signed-by=/usr/share/keyrings/docker-archive-keyring.gpg]\
  https://download.docker.com/linux/ubuntu \
  $(lsb_release -cs) stable" \
  | sudo tee /etc/apt/sources.list.d/docker.list \
  > /dev/null
```

Install Docker Engine

1. Update the apt package index, and install the latest version of Docker Engine, containerd, and Docker Compose:

```
$ sudo apt-get update

$ sudo apt-get install docker-ce docker-ce-cli containerd.
   io docker-compose-plugin
```

2. To install a specific version of Docker Engine, list the available versions in the repo, then select and install:

   (a) List the versions available in your repo:

   ```
   $ apt-cache madison docker-ce
   ```

   (b) Install a specific version using the version string from the second column, for example, 5:18.09.1 3-0 ubuntu-xenial.

   ```
   $ sudo apt-get install \
       docker-ce=<VERSION_STRING> \
       docker-ce-cli=<VERSION_STRING> \
       containerd.io docker-compose-plugin
   ```

Verify Docker Installation

1. Make sure the docker daemon is running.

```
$ sudo systemctl start docker
```

2. Add your user to the docker group.

```
$ sudo usermod -a -G docker <username>
```

3. Verify that Docker Engine is installed correctly by running the hello-world image.

```
$ sudo docker run hello-world
```

This command downloads a test image and runs it in a container. When the container runs, it prints a message and exits.

Appendix-II: Install Fabric Samples, Binaries, and Docker Images

Determine a location on your machine where you want to place the fabric-samples repository and enter that directory in a terminal window. The command that follows will perform the following steps:

1. If needed, clone the hyperledger/fabric-samples repository
2. Checkout the appropriate version tag
3. Install the Hyperledger Fabric platform-specific binaries and config files for the version specified into the /bin and /config directories of fabric-samples
4. Download the Hyperledger Fabric docker images for the version specified

Once you are ready, and in the directory into which you will install the Fabric Samples and binaries, go ahead and execute the command to pull down the binaries and images.

```
$ curl -sSL https://bit.ly/2ysbOFE | bash -s
```

The command above downloads and executes a bash script that will download and extract all of the platform-specific binaries you will need to set up your network and place them into the cloned repo you created above. It retrieves the following platform-specific binaries:

- configtxgen,
- configtxlator,
- cryptogen,
- discover,
- idemixgen
- orderer,
- peer,
- fabric-ca-client,
- fabric-ca-server

and places them in the bin sub-directory of the current working directory.

You may want to add that to your PATH environment variable so that these can be picked up without fully qualifying the path to each binary. e.g.:

```
$ export PATH=<path to download location>/bin:$PATH
```

Finally, the script will download the Hyperledger Fabric docker images from Docker Hub into your local Docker registry and tag them as ‘latest’.

The script lists out the Docker images installed upon conclusion.

Appendix-III: Hyperledger Fabric Configuration Files

Configuration for a channel:

The configuration given below is used with configtxgen tool to create the genesis block for a channel.

```
$ configtxgen -profile OneOrgApplicationGenesis \
  -outputBlock ./channel-artifacts/genesis.block \
  -channelID channel1
```

configtx.yaml

```yaml
Organizations:
    - &OrdererOrg
        Name: OrdererOrg
        ID: OrdererMSP
        MSPDir: ./crypto-material/ordererOrganizations/example.com/msp

        Policies:
            Readers:
                Type: Signature
                Rule: "OR('OrdererMSP.member')"
            Writers:
                Type: Signature
                Rule: "OR('OrdererMSP.member')"
            Admins:
                Type: Signature
                Rule: "OR('OrdererMSP.admin')"

        OrdererEndpoints:
            - orderer.example.com:7050

    - &Org1
        Name: Org1MSP
        ID: Org1MSP
        MSPDir: ./crypto-material/peerOrganizations/org1.example.com/msp

        Policies:
            Readers:
                Type: Signature
                Rule: "OR('Org1MSP.admin', 'Org1MSP.peer',
                'Org1MSP.client')"
            Writers:
                Type: Signature
                Rule: "OR('Org1MSP.admin', 'Org1MSP.client')"
            Admins:
```

```yaml
                Type: Signature
                Rule: "OR('Org1MSP.admin')"
            Endorsement:
                Type: Signature
                Rule: "OR('Org1MSP.peer')"



Capabilities:
    Channel: &ChannelCapabilities
        V2_0: true

    Orderer: &OrdererCapabilities
        V2_0: true

    Application: &ApplicationCapabilities
        V2_0: true



Application: &ApplicationDefaults

    Organizations:

    Policies:
        Readers:
            Type: ImplicitMeta
            Rule: "ANY Readers"
        Writers:
            Type: ImplicitMeta
            Rule: "ANY Writers"
        Admins:
            Type: ImplicitMeta
            Rule: "MAJORITY Admins"
        LifecycleEndorsement:
            Type: ImplicitMeta
            Rule: "MAJORITY Endorsement"
        Endorsement:
            Type: ImplicitMeta
            Rule: "MAJORITY Endorsement"

    Capabilities:
        <<: *ApplicationCapabilities


Orderer: &OrdererDefaults
```

```yaml
    OrdererType: etcdraft
    Addresses:
        - orderer.example.com:7050

    EtcdRaft:
        Consenters:
        - Host: orderer.example.com
          Port: 7050
          ClientTLSCert: ./crypto-material/ordererOrganizations/example.c\
          om/orderers/orderer.example.com/tls/server.crt
          ServerTLSCert: ./crypto-material/ordererOrganizations/example.c\
          om/orderers/orderer.example.com/tls/server.crt

    BatchTimeout: 2s
    BatchSize:
        MaxMessageCount: 10
        AbsoluteMaxBytes: 99 MB
        PreferredMaxBytes: 512 KB

    Organizations:
    Policies:
        Readers:
            Type: ImplicitMeta
            Rule: "ANY Readers"
        Writers:
            Type: ImplicitMeta
            Rule: "ANY Writers"
        Admins:
            Type: ImplicitMeta
            Rule: "MAJORITY Admins"
        BlockValidation:
            Type: ImplicitMeta
            Rule: "ANY Writers"



Channel: &ChannelDefaults
    Policies:
        Readers:
            Type: ImplicitMeta
            Rule: "ANY Readers"
        Writers:
            Type: ImplicitMeta
            Rule: "ANY Writers"
        Admins:
```

```yaml
            Type: ImplicitMeta
            Rule: "MAJORITY Admins"

    Capabilities:
        <<: *ChannelCapabilities



Profiles:
    OneOrgApplicationGenesis:
        <<: *ChannelDefaults
        Orderer:
            <<: *OrdererDefaults
            Organizations:
                - *OrdererOrg
            Capabilities: *OrdererCapabilities
        Application:
            <<: *ApplicationDefaults
            Organizations:
                - *Org1
            Capabilities: *ApplicationCapabilities
```

Configuration file to generate the crypto material:

crypto_config.yaml

```yaml
OrdererOrgs:
  - Name: Orderer
    Domain: example.com
    EnableNodeOUs: true
    Specs:
      - Hostname: orderer


PeerOrgs:
  - Name: Org1
    Domain: org1.example.com
    EnableNodeOUs: true
    Template:
      Start: 1
      Count: 2
    Users:
      Start: 1
      Count: 1
```

This configuration is used initially to generate crypto material using cryptogen tool. The crypto material will contain credentials for all the entities i.e, admins, orderer and peers. It contains Digital Certificates, Public and Private Keys, etc. A digital certificate is a document which holds a set of attributes relating to the holder of the certificate. The most common type of certificate is the one compliant with the X.509 standard, which allows the encoding of a party's identifying details in its structure.

```
$ cryptogen generate --config="./crypto-config.yaml" \
  --output="crypto-material"
```

Connection configuration file for the Fabric Gateway:

connection-org1.yaml

```yaml
name: "simplecc.org1.profile"
version: "1.1"

channels:
  channel1:
    orderers:
      - orderer.example.com
    peers:
      - peer1.org1.example.com

organizations:
  Org1MSP:
    mspid: Org1MSP
    peers:
      - peer1.org1.example.com

orderers:
  orderer.example.com:
    url: grpcs://localhost:7050
    grpcOptions:
      ssl-target-name-override: orderer.example.com
    tlsCACerts:
      path: ../crypto-material/ordererOrganizations/example.com/orderers/\
      orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem

peers:
  peer1.org1.example.com:
    url: grpcs://localhost:7051
    grpcOptions:
      ssl-target-name-override: peer1.org1.example.com
    tlsCACerts:
```

```
      path: ../crypto-material/peerOrganizations/org1.example.com/peers/p\
      eer1.org1.example.com/msp/tlscacerts/tlsca.org1.example.com-cert.pem
```

This configuration helps the Fabric Gateway to communicate with the orderer and peers to carry out a transaction. It defines all the available choices of orderer and peer endpoints and, all file system paths to required digital certificates to form a secure connection, are also specified.

Publication

Paper entitled **"Blockchain based Web Framework for Real Estate Transactions"** was presented at **"5th International Conference on Computer Networks and Inventive Communication Technologies"** for the Scopus indexed Springer journal by **"Rajan Khade, Amit Pandey, Aditya Shinde and Neha Deshmukh".**